13

INTEGRATION AND EVALUATION

13.1 INTEGRATING, TESTING, AND EVALUATING THE TOTAL SYSTEM

As its name implies, the integration and evaluation phase has the objectives of assembling and integrating the engineered components of the new system into an effectively operating whole, and demonstrating that the system meets all of its operational requirements. The goal is to qualify the system's engineering design for release to production and subsequent operational use.

As previously noted, the systems engineering life cycle model defines integration and evaluation as a separate phase of system development because its objectives and activities differ sharply from those of the preceding portion called the engineering design phase. These differences are also reflected in changes in the primary participants engaged in carrying out the technical effort.

If all of the building blocks of a new system were correctly engineered, and if their design was accurately implemented, their integration and subsequent evaluation would be relatively straightforward. In reality, when a team of contractors develops a complex

Systems Engineering Principles and Practice, Second Edition. Alexander Kossiakoff, William N. Sweet, Samuel J. Seymour, and Steven M. Biemer
© 2011 by John Wiley & Sons, Inc. Published 2011 by John Wiley & Sons, Inc.

system during a period of rapidly evolving technology, the above conditions are never fully realized. Hence, the task of system integration and evaluation is always complex and difficult and requires the best efforts of expert technical teams operating under systems engineering leadership.

The success of the integration and evaluation effort is also highly dependent on the advance planning and preparation for this effort that was accomplished during the previous phases. A detailed test and evaluation master plan (TEMP) is required to be formulated by the end of concept exploration and elaborated at each step thereafter (see Chapter 10). In practice, such planning usually remains quite general until well into the engineering design phase for several reasons:

1. The specific test approach is dependent on just how the various system elements are physically implemented.
2. Test planning is seldom allocated adequate priority in either staffing or funding in the early phases of system development.
3. Simulating the system operational environment is almost always complicated and costly.

Hence, the integration and evaluation phase may begin with very considerable preparation remaining and may therefore proceed considerably slower than originally planned. The purpose of this chapter is to describe the essential activities that are typically required in this phase, a number of the problems that are commonly encountered, and some of the approaches to helping overcome the resulting obstacles.

Place of the Integration and Evaluation Phase in the System Life Cycle

It was seen in previous chapters that the general process of test and evaluation is an essential part of every phase of system development, serving as the validation step of the systems engineering method. It can be generally defined as embodying those activities necessary to reveal the critical attributes of a product (in this case a system element, such as a subsystem or component) and to compare them to expectations in order to deduce the product's readiness for succeeding activities or processes. In the integration and evaluation phase, the process of test and evaluation becomes the central activity, terminating with the evaluation of the total system in a realistic replica of its intended operational environment.

Figures 13.1 and 13.2 show two different aspects of the relation between the integration and evaluation phase and its immediately adjacent phases in the system life cycle. Figure 13.1 is a functional flow view, which shows the integration and evaluation phase to be the transition from engineering design to production and operation. Its inputs from the engineering design phase are an engineered prototype, including components, and a test and evaluation plan, with test requirements. The outputs of the integration and evaluation phase are system production specifications and a validated production system design. Figure 13.2 is a schedule and level-of-effort view, which

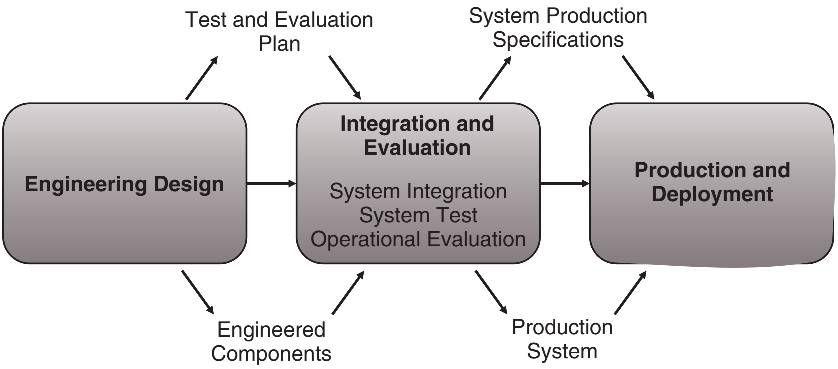

Figure 13.1. Integration and evaluation phase in a system life cycle.

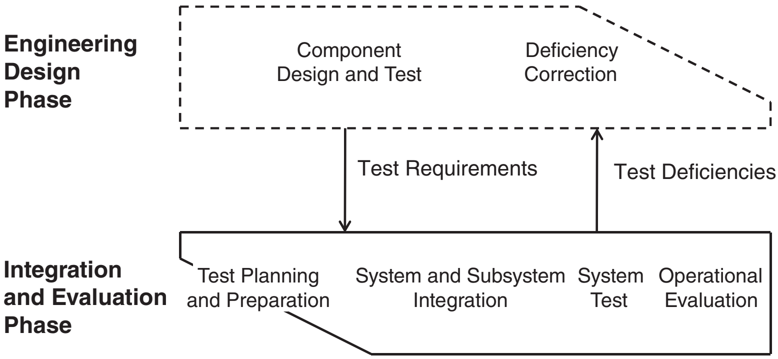

Figure 13.2. Integration and evaluation phase in relation to engineering design.

shows the overlap of the integration and evaluation phase with the engineering design phase.

The differences in the primary objectives, activities, and technical participants of the integration and evaluation phase from those of the engineering design phase are summarized in the following paragraphs.

Program Focus. The engineering design phase is focused on the design and testing of the individual system components and is typically carried out by a number of different engineering organizations, with systems engineering and program management oversight being exercised by the system developer. On the other hand, the integration and evaluation phase is concerned with assembling and integrating these engineered components into a complete working system, creating a comprehensive system test environment and evaluating the system as a whole. Thus, while these activities overlap in time, their objectives are quite different.

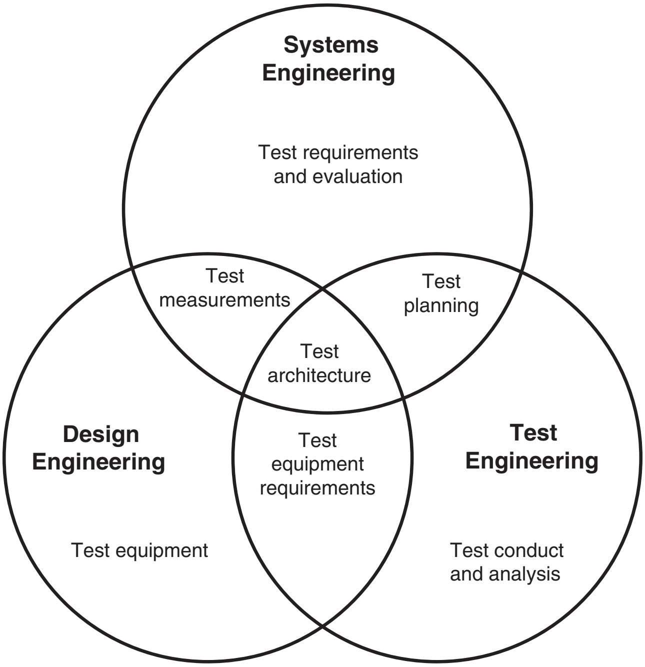

Figure 13.3. System test and evaluation team.

Program Participants. The primary participating technical groups in the integration and evaluation phase are systems engineering, test engineering, and design engineering. Their functions are pictured in the Venn diagram of Figure 13.3, which shows the activities that are primary ones for each technical group and those that are shared. Systems engineering is shown as having the prime responsibility for defining the test requirements and evaluation criteria. It shares the responsibility for test planning with test engineering and the definition of test methodology and data to be collected with design engineering. Test engineering has responsibility for test conduct and data analysis; it usually provides a majority of the technical effort during this period. In many programs, design engineering has the prime responsibility for test equipment design. It is also responsible for component design changes to eliminate deficiencies uncovered in the test and evaluation process.

Critical Problems. The system integration process represents the first time that fully engineered components and subsystems are linked to one another and are made to perform as a unified functional entity. Despite the best plans and efforts, the integration of a system containing newly developed elements is almost certain to reveal unexpected incompatibilities. At this late stage in the development, such incompatibilities must be resolved in a matter of days rather than weeks or months. The same is true when deficiencies are discovered in system evaluation tests. Any crash program to resolve such critical problems should be led by systems engineers working closely with the project manager.

Management Scrutiny. A large-scale system development program represents a major commitment of government and/or industrial funds and resources. When the development reaches the stage of system integration and testing, management scrutiny becomes intense. Any real or apparent failures are viewed with alarm, and temptations to intervene become strong. It is especially important that the program management and systems engineering leadership have the full confidence of top management, and the authority to act, at this time.

Design Materialization Status

The status of system materialization in the integration and evaluation phase is shown in Table 13.1. The table entries identifying the principal activities in this phase are seen to be in the upper right-hand corner, departing sharply from the downward progression of activities in the previous phases. This corresponds to the fact that in the other phases, the activities referred to the stepwise materialization of the individual component building blocks, progressing through the states of visualization, functional definition, and physical definition to detailed design, fabrication, and testing. In contrast, the activities in the integration and evaluation phase refer to the stepwise materialization of the entire system as an operational entity, proceeding through the integration and test of physically complete components into subsystems, and these into the total system.

A very important feature of the materialization status, which is not explicitly shown in Table 13.1, is the characterization of interactions and interfaces. This process should have been completed in the previous phase but cannot be fully validated until the whole system is assembled. The inevitable revelation of some incompatibilities must therefore be anticipated as the new system is integrated. Their prompt identification and resolution is a top priority of systems engineering. Accomplishing the integration of interfaces and interactions may not appear to be a major increase in the materialization of a system, but in reality, it is a necessary (and sometimes difficult) step in achieving a specified capability.

This view of the activities and objectives of the integration and evaluation phase can be further amplified by expanding the activities pictured in the last column of Table 13.1. This is demonstrated in Table 13.2, in which the first column lists the system aggregation corresponding to the integration level as in Table 13.1; the second column indicates the nature of the environment in which the corresponding system element is evaluated; the third column lists the desired objective of the activity; and the fourth defines the nature of the activity, expanding the corresponding entries in Table 13.1. The sequence of activities, which proceeds upward in the above table, starts with tested components, integrates these into subsystems, and then into the total system. The process then evaluates the system, first in a simulated operational environment and finally in a realistic version of the environment in which the system is intended to operate. Thus, as noted earlier, in the integration and evaluation phase, the process of

TABLE 13.1. Status of System Materialization at the Integration and Evaluation Phase

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

TABLE 13.2. System Integration and Evaluation Process

Integration level	Environment	Objective	Process
System	Real operational environment	Demonstrated operational performance	Operational test and evaluation
System	Simulated operational environment	Demonstrated compliance with all requirements	Developmental test and evaluation
System	Integration facility	Fully integrated system	System integration and test
Subsystem	Integration facility	Fully integrated subsystems	Subsystem integration and test
Component	Component test equipment	Verified component performance	Component test

materialization refers to the system as a whole and represents the synthesis of the total operational system from the previously physically materialized components.

Systems Engineering Method in Integration and Evaluation

Since the structure of the integration and evaluation phase does not conform to the characteristics of the preceding phases, the application of the systems engineering method is correspondingly different. In this phase, the requirements analysis or problem definition step corresponds to test planning—the preparation of a comprehensive plan of how the integration and evaluation tests are to be carried out. Since the functional design of the system and its components has been completed in previous phases, the functional definition step in this phase relates to the test equipment and facilities, which should be defined as a part of test preparation. The physical definition or synthesis step corresponds to subsystem and system integration, the components having been implemented in previous phases. The design validation step corresponds to system test and evaluation.

The organization of the principal sections in this chapter will follow the order of the above sequence. However, it is convenient to combine test planning and test equipment definition into a single section on test planning and preparation and to divide system test and evaluation into two sections: developmental system testing, and operational test and evaluation. These sections will be seen to correspond to the processes listed in the right-hand column of Table 13.2, reading upward from the fourth row.

Test Planning and Preparation. Typical activities include

- reviewing system requirements and defining detailed plans for integration and system testing, and
- defining the test requirements and functional architecture.

System Integration. Typical activities include

- integrating the tested components into subsystems and the subsystems into a total operational system by the sequential aggregation and testing of the constituent elements, and
- designing and building integration test equipment and facilities needed to support the system integration process and demonstrating end-to-end operation.

Developmental System Testing. Typical activities include

- performing system-level tests over the entire operating regime and comparing system performance with expectations,
- developing test scenarios exercising all system operating modes, and
- eliminating all performance deficiencies.

Operational Test and Evaluation. Typical activities include

- performing tests of system performance in a fully realistic operational environment under the cognizance of an independent test agent and
- measuring degree of compliance with all operational requirements and evaluating the readiness of the system for full production and operational deployment.

13.2 TEST PLANNING AND PREPARATION

As described earlier, planning for test and evaluation throughout the system development process begins in its early phases and is continually extended and refined. As the system design matures, the test and evaluation process becomes more exacting and critical. By the time the development nears the end of the engineering design phase, the planning and preparation for the integration and evaluation of the total system represents a major activity in its own right.

TEMP

It was noted in Chapter 10 that acquisition programs often require the preparation of a formal TEMP. Many of the principal subjects covered in the TEMP are applicable to the development of commercial systems as well. For reference purposes, the main elements of the TEMP format, described more fully in Chapter 10, are listed below:

1. *System Introduction:* describes the system and its mission and operational environment and lists measures of effectiveness;
2. *Integrated Test Program Summary:* lists the test program schedule and participating organizations;

3. *Developmental Test and Evaluation:* describes objectives, method of approach, and principal events;
4. *Operational Test and Evaluation:* describes objectives, test configuration, events, and scenarios; and
5. *Test and Evaluation Resource Summary:* lists test articles, sites, instrumentation, and support operations.

Elements 3 and 4 will be referred to in somewhat greater detail in the final sections of this chapter.

Analogy of Test and Evaluation Planning to System Development

The importance of the test and evaluation planning process is illustrated in Table 13.3, which shows the parallels between this process and system development as a whole. The left half of the table lists the principal activities involved in each of four major steps in the system development process. The entries in the right half of the table list the corresponding activities in developing the test and evaluation plan. The table shows that the tasks comprising the test and evaluation planning process require major decisions regarding the degree of realism, trade-offs among test approaches, definition of objectives, and resources for each test event, as well as development of detailed procedures and test equipment. In emphasizing the correspondence between these activities, the table also brings out the magnitude of the test and evaluation effort and its criticality to successful system development.

As may be inferred from Table 13.3, specific plans for the integration and evaluation phase must be developed before or concurrently with the engineering design

TABLE 13.3. Parallels between System Development and Test and Evaluation (T&E) Planning

System development	T&E planning
Need:	Objective:
Define the capability to be fielded.	Determine the degree of sophistication required of the test program.
System concept:	Test concept:
Analyze trade-offs between performance, schedule, and cost to develop a system concept.	Evaluate trade-offs between test approaches, schedule, and cost to develop a test concept.
Functional design:	Test plan:
Translate functional requirements into two level specification for the (sub)system(s).	Translate test requirements into a description of each test event and the resources required.
Detailed design:	Test procedures:
Design the various components that comprise the system.	Develop detailed test procedures and test tools for each event.

process. This is necessary in order to provide the time required for designing and building special test equipment and facilities that will be needed during integration and system testing. Costing and scheduling of the test program is an essential part of the plan since the costs and duration for system testing are very often underestimated, seriously impacting the overall program.

Review of System Requirements

Prior to the preparation of detailed test plans, it is necessary to conduct a final review of the system-level operational and functional requirements to ensure that no changes have occurred during the engineering design phase that may impact the system test and evaluation process. Three potential sources for such changes are described below:

1. *Changes in Customer Requirements.* Customer needs and requirements seldom remain unchanged during the years that it takes to develop a complex new system. Proposed changes to software requirements seem deceptively easy to incorporate but frequently prove disproportionately costly and time-consuming.
2. *Changes in Technology.* The rapid advances in key technologies, especially in solid-state electronics, accumulated over the system development time, offer the temptation to take advantage of new devices or techniques to gain significant performance or cost savings. The compulsion to do so is heightened by increases in the performance of competitive products that utilize such advances. Such changes, however, usually involve significant risks, especially if made late in the engineering design phase.
3. *Changes in Program Plans.* Changes that impact system requirements and are unavoidable may come from programmatic causes. The most common is funding instability growing out of the universal competition for resources. Lack of adequate funds to support the production phase may lead to a slip in the development schedule. Such events are often beyond the control of program management and have to be accommodated by changes in schedules and fund allocations.

Key Issues

There are several circumstances that require special attention during test planning and preparation for system integration and evaluation. These include the following:

1. *Oversight.* Management oversight is especially intense during the final stages of a major development. System tests, especially field tests, are regarded as indicators of program success. Test failures receive wide attention and invite critical investigation. Test plans must provide for acquisition of data that are necessary to be able to explain promptly and fully any mishaps and remedial measures to program management, the customer, and other concerned authorities.

2. *Resource Planning.* Test operations, especially in the late stages of the program, are costly in manpower and funds. Too frequently, overruns and slippages in the development phases cut into test schedules and budgets. Serious problems of this type can be avoided only through careful planning to assure that the necessary resources are made available when required.
3. *Test Equipment and Facilities.* Facilities for supporting test operations must be designed and built concurrently with system development to be ready when needed. Advance planning for such facilities is essential. Also, the sharing of facilities between developmental and operational testing, wherever practicable, is important in order to stay within program funding limits.

Test Equipment Design

As noted in Chapter 11, the testing of system elements, as well as the system as a whole, requires test equipment and facilities that can stimulate the element under test with external inputs and can measure the system responses. This equipment must meet exacting standards:

1. *Accuracy.* The inputs and measurements should be several times more precise than the tolerances on the system element inputs and responses. There must be calibration standards available for ensuring that the test equipment is in proper adjustment.
2. *Reliability.* The test equipment must be highly reliable to minimize test discrepancies due to test equipment errors. It should be either equipped with self-test monitors or subjected to frequent checks.
3. *Flexibility.* To minimize costs where possible, test equipment should be designed to serve several purposes, although not at the expense of accuracy or reliability. It is frequently possible to use some of the equipment designed for component tests also for.

Before designing the test equipment, it is important to define fully the test procedures so as to avoid later redesign to achieve compatibility between test equipment and the component or subsystem under test. This again emphasizes the importance of early and comprehensive test planning.

The paragraphs below discuss some of the aspects of test preparation peculiar to the integration, system test, and operational evaluation parts of the test and evaluation process.

Integration Test Planning

Preparing for the system integration process is dependent on the manner in which the system components and subsystems are developed. Where one or more components of a subsystem involve new technical approaches, the entire subsystem is often developed by the same organization and integrated prior to delivery to the system contractor. For example, aircraft engines are usually developed and integrated as units before delivery to the airplane developer. In contrast, components using mature technologies are often acquired to a specification and delivered as individual building blocks. The integration process at the system contractor's facility must deal with whatever assortment of components, subsystems, or intermediate assemblies is delivered from the respective contractors.

As stated previously, it is important to support the integration process at both the subsystem and system levels by capable integration facilities. These must provide the necessary test inputs, environmental constraints, power and other services, output measurement sensors, as well as test recording and control stations. Many of these must be custom designed for each specific use. The facilities must be designed, built, and calibrated before integration is to begin. A typical physical test configuration for is described in Section 13.3, System Integration.

Developmental System Test Planning

Preparing for system-level tests to determine that the system performance requirements are met and that the system is ready for operational evaluation is more than a normal extension of the integration test process. Integration testing is necessarily focused on ensuring that the system's components and subsystems fit together in form and function. System performance tests go well beyond this goal and measure how the system as a whole responds to its specified inputs and whether its performance meets the requirements established at the outset of its development.

The success or failure of a test program is critically dependent on the extent to which the total effort is thoughtfully planned and precisely detailed, the test equipment is well engineered and tested, and the task is thoroughly understood by the test and data analysis teams. Problems in system testing are at least as likely to be caused by faults in the test equipment, poorly defined procedures, or human error as by improper system operation. Thus, it is necessary that the test facilities be engineered and tested under the same rigorous discipline as that used in system development. Many programs suffer from insufficient time and effort being assigned to the testing process, and pay for such false economy by delays and excessive costs during system testing. To minimize the likelihood of such consequences, the test program must be planned early and in sufficient detail to identify and estimate the cost of the required facilities, equipment, and manpower.

Operational Evaluation Planning

Because operational evaluation is usually conducted by the customer or a test agent, its planning is necessarily done separately from that for integration and development testing. However, in many large-scale system developments, the costs of system-level testing compel the common use of as much development test equipment and facilities as may be practicable.

In some cases, a joint developer–customer test and evaluation program is carried out, in which the early phases are directed by the developer and the later phases by the customer or the customer's agent. Such collaborative programs have the advantage of providing a maximum exchange of information between the developer and customer, which is to their mutual benefit. This also helps to avoid misunderstandings, as well as to quickly resolve unanticipated problems encountered during the process.

At the other extreme are operational test and evaluation programs that are carried out in a very formal manner by a special system evaluation agent and with maximum independence from the developer. However, even in such cases, it is important for both the developer and the system evaluation agent to establish channels of communication to minimize misinformation and unnecessary delays.

13.3 SYSTEM INTEGRATION

In the engineering of new complex systems with many interacting components, testing at the system level cannot begin until the system has been fully assembled and demonstrated to operate as a unified whole. The likelihood that some of the interfaces among the elements may not fit or function properly, or that one or more interactions among them may fall outside prescribed tolerances, is usually high. It is only the very simplest systems that are assembled without testing at several intermediate levels of aggregation. Thus, experience has shown that no matter how thoroughly the individual components have been tested, there almost always remain unforeseen incompatibilities that do not reveal themselves until the system elements are brought together. Such discrepancies usually require changes in some components before the integrated system works properly. These changes, in turn, frequently require corresponding alterations in test equipment or procedures and must be reflected in all relevant documentation. This section describes the general process and problems involved in integrating a typical complex system.

The successful and expeditious integration of a complex system depends on how well it has been partitioned into subsystems that have simple interactions with one another and are themselves subdivided into well-defined components. The integration process can be thought of as the reverse of partitioning. It is normally accomplished in two stages: (1) the individual subsystems are integrated from their components, and (2) the subsystems are assembled and integrated into the total system. At intervals during both stages, the assembled elements are tested to determine whether or not they fit and interact together in accordance with expectations. In the event that they do not, special test procedures are instituted to reveal the particular design features that need to be corrected. Throughout the entire process, system integration proceeds in an orderly, stepwise manner with system elements added one or two at a time and then tested to demonstrate proper operation before proceeding to the next step. This procedure maintains control of the process and simplifies diagnosis of discrepancies. The price for this stepwise integration of the system is that at every step, the test equipment must simulate the relevant functions of the missing parts of the system. Nevertheless, experience in the development of large systems has repeatedly demonstrated that the provision of this capability is, in the long run, quite cost-effective. In the integration of large software programs, this is frequently done by connecting the "program executive" to "stubbed-off" or nonfunctioning modules, which are successively replaced one at a time by functioning modules.

Determining the most effective order of assembly and selecting the optimum test intervals are critical to minimizing the effort and time needed to accomplish the integration process. Since both system-level knowledge and test expertise are essential to the definition of this process, the task is normally assigned to a special task team composed of systems engineers and test specialists.

Physical Test Configuration

Integration testing requires versatile and readily reconfigurable integration facilities. To understand their operation, it is useful to start with a generic model of a system element test configuration. Such a model is illustrated in Figure 13.4 and is described below.

The *system element* (component or subsystem) under test is represented by the block at the top center of the figure. The *input generator* converts test commands into exact replicas, functionally and physically, of the inputs that the system element is expected to receive. These may be a sequence of typical inputs covering the range expected under operational conditions. The input signals in the same or simulated form are also fed to the element model. The *output analyzer* converts any outputs that are not already in terms of quantitative physical measures into such form. Whether or not the data obtained in the tests are compared in real time with predicted responses from the element model, they should also be recorded, along with the test inputs and other conditions, for subsequent analysis. In the event of discrepancies, this permits a more detailed diagnosis of the source of the problem and a subsequent comparison with results of

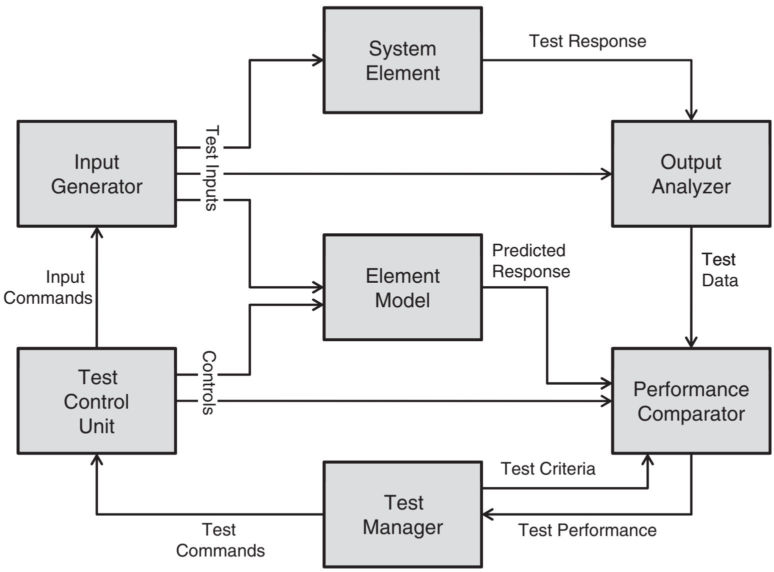

Figure 13.4. System element test configuration.

suitably modified elements. The physical building blocks in the top row of Figure 13.4 may be seen to implement the corresponding functional elements of Figure 13.3.

The *element model*, pictured in the center of the Figure 13.4, has the function of reproducing very precisely the response that the component or subsystem under test is expected to produce to each input, according to its performance specifications. The element model may take several forms. At one extreme, it may be a specially constructed and validated replica of the system element itself. At the other, it may be a mathematical model of the element, perhaps as simple as a table lookup if the predicted performance is an explicit function of the input. How it is configured determines the form of input required to drive it.

The *test manager* introduces a function not represented in the basic test architecture of Figure 13.3. Because the testing of most elements of complex systems is a complicated process, it requires active supervision by a test engineer, usually supported by a control console. This allows critical test results to be interpreted in real time in terms of required performance so that the course of testing can be altered if significant deviations are observed.

The *performance comparator* matches the measured system element outputs with the expected outputs from the element model in accordance with test criteria provided by the test manager. The comparison and assessment is performed in real time whenever practicable to enable a rapid diagnosis of the source of deviations from expected results, as noted previously. The evaluation criteria are designed to reflect the dependence of the operational performance on individual performance parameters.

Most actual test configurations are considerably more complex than the simplified example in Figure 13.4. For example, tests may involve simultaneous inputs from several sources involving various types of system elements (e.g., signal, material, and mechanical), each requiring a different type of signal generator. Similarly, there are usually several outputs, necessitating different measuring devices to convert them into forms that can be compared with predicted outputs. The tests may also involve a series of programmed inputs representing typical operating sequences, all of which must be correctly processed.

It is clear from the above discussion that the functionality embodied in the test configuration of a system element is necessarily comparable to that of the element itself. Hence, designing the test equipment is itself a task of comparable difficulty to that of developing the system element. One factor that makes the task somewhat simpler is that the environment in which the test equipment operates is usually benign, whereas the system operating environment is often severe. On the other hand, the precision of the test equipment must be greater than that of the system element to ensure that it does not contribute significantly to measured deviations from the specified element performance.

Subsystem Integration

As noted previously, the integration of a subsystem (or system) from its component parts is normally a stepwise assembly and test process in which parts are systematically aggregated, and the assembly is periodically tested to reveal and correct any faulty interfaces or component functions as early in the process as practicable. The time and effort required to conduct this process is critically dependent on the skillful organization of the test events and the efficient use of facilities. Some of the most important considerations are discussed below.

The order in which system components are integrated should be selected to avoid the need to construct special input generators for simulating components within the subsystem, that is, other than those simulating inputs from sources external to the subsystem being integrated. Thus, at any point in the assembly, the component that is to be added should have inputs that are derivable from either generators of external inputs or the outputs of components previously assembled.

The above approach means that subsystem integration should begin with components that have only external inputs, either from the system environment or from other subsystems. Examples of such components include

1. subsystem support structures,
2. signal or data input components (e.g., external control transducers), and
3. subsystem power supplies.

The application of the above approach to the integration of a simple subsystem is illustrated in Figure 13.5. The figure is an extension of Figure 13.4, in which the subsystem under test is composed of three components. The configuration of components in the figure is purposely chosen so that each component has a different combination of inputs and outputs. Thus, component A has a single input from an external subsystem and two outputs—one an internal output to B and the other to another subsystem. Component B has no external interfaces—getting its input from A and producing an output to C. Component C has two inputs—one external and the other internal, and a single output to another subsystem.

The special features of the test configuration are seen to be

1. a compound input generator to provide the two external inputs to the subsystem —one to A and the other to C;
2. internal test outputs from the interfaces between A and B, and between B and C; these are needed to identify the source of any observed deviation in the overall performance and are in addition to the external subsystem outputs from A and C; and
3. a compound element model containing the functions performed by the constituent components and providing the predicted outputs of the test interfaces.

Following the integration sequence approach described above, the configuration in Figure 13.5 would be assembled as follows:

1. Start with A, which has no internal inputs. Test A's outputs.
2. Add B and test its output. If faulty, check if input from A is correct.
3. Add C and test its output.

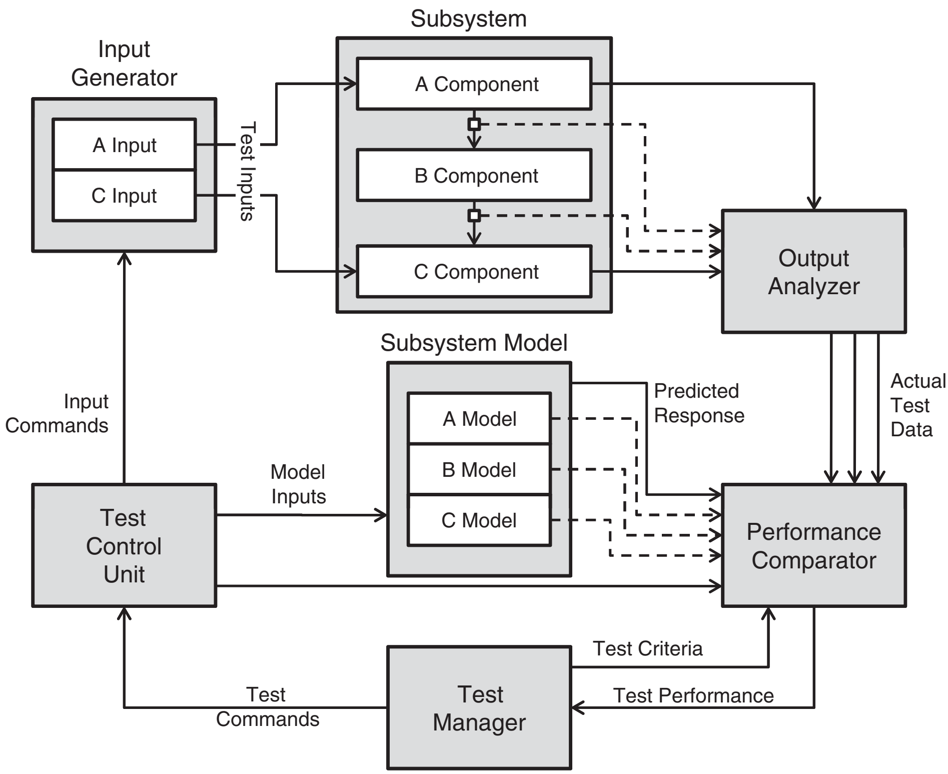

Figure 13.5. Subsystem test configuration.

The above integration sequence does not require the construction of input generators to provide internal functions and should rapidly converge on the source of a faulty component or interface.

The approach described above works in the great majority of cases but must, of course, be carefully reviewed in the light of any special circumstances. For example, there may be safety issues that make it necessary to leave out or add steps to circumvent unsafe testing conditions. The temporary unavailability of key components may require a substitution or simulation of elements. Particularly critical elements may have to be tested earlier than in the ideal sequence. Systems engineering judgment must be applied in examining such issues before defining the integration sequence.

Test Conduct and Analysis. The determination of whether or not a given step in the integration process is successful requires matching the outputs of the partially assembled components against their expected values as predicted by the model. The effort required to make this comparison depends on the degree of automation of the test configuration and of the analytical tools embodied in the performance comparator block in Figure 13.5. The trade-off between the sophistication of the test and analysis tools and the analysis effort itself is one of the critical decisions to be made in planning the integration process.

In scheduling and costing the integration effort, it must be expected that numerous deviations will be observed in the measured performance from that predicted by the model, despite the fact that all components presumably have previously passed qualification tests. Each discrepancy must be dealt with by first documenting it in detail, identifying the principal source(s) of the deviations, and devising the most appropriate means of eliminating or otherwise resolving the discrepancy.

It should be emphasized that in practice, most failures observed during the integration process are usually due to causes other than component malfunctions. Some of the most frequently occurring problem areas are faulty test equipment or procedures, misinterpretation of specifications, unrealistically tight tolerances, and personnel error. These are discussed in the succeeding paragraphs.

There are several reasons why faults are frequently found in the test equipment:

1. The amount of design effort allocated to the design and fabrication of test equipment is far smaller than the effort spent on component design.
2. The test equipment must be more precise than the components to ensure that its tolerances do not contribute significantly to observed deviations from predictions.
3. The equipment used to test separately an individual component may not be exactly the same as that incorporated into the integration test facility, or its calibration may be different.
4. The predictions of expected performance of the element under test by the element model may be imperfect due to the impossibility of modeling exactly the behavior of the test element.

Not infrequently, the specifications of interfaces and interactions among components permit different interpretations by the designers of interfacing components. This can result in significant mismatches when the components are assembled. There is no practical and foolproof method of entirely eliminating this source of potential problems. Their number can, however, be minimized through critical attention to and review of each interface specification prior to its release for design of the associated hardware or software. In most cases, establishing an interface coordination team, including all involved contractors, has proven to be advantageous.

To ensure that interfacing mechanical, electrical, or other elements fit together and interact properly, the specifications for each separate element must include the permitted tolerances (deviations from prescribed values) in the interacting quantities. For example, if the interfacing components are held together by bolts, the location of the holes in each component must be specified within a plus/minus tolerance of their nominal dimensions. These tolerances must allow for the degree of precision of production machinery, as well as normal variations in the size of standard bolts. If the specified tolerances are too tight, there will be excessive rejects in manufacture; if too loose, there will be occasional misalignments, causing fit failures.

Personnel errors are a common source of test failure and one that can never be completely avoided. Such failures may occur because of inadequate training, unclear or insufficiently detailed test procedures, overly complex or demanding test methods, fatigue, or simple carelessness. Errors of this type can occur at any point in the planning, execution, and support of the testing process.

Changes. If the diagnosis of a faulty test traces the problem to a component design feature, it is necessary to undertake a highly expedited effort to determine the most practical and effective means of resolving the problem. At this stage of development, the design should be under strict configuration management. Since any significant change will be costly and potentially disruptive, all means of avoiding or minimizing the change must be explored and several alternatives examined. The final decision will have to be made at the program management level if significant program cost and schedule changes are involved.

If there is no "quick fix" available, consideration may be given to seeking a waiver to deviate from a certain specification for an initial quantity of production units so as to afford adequate time to design and validate the change prior to its release for production. Not infrequently, careful analysis reveals that the effect of the deviation on operational performance is not sufficient to warrant the cost of making the change, and a permanent waiver is granted. Systems engineering analysis is the key to determining the best course of action in such circumstances, and to advocating its approval by management and the customer.

Total System Integration

The integration of the total system from its subsystems is based on the same general principles as those governing the integration of individual subsystems, described in the preceding paragraphs. The main differences are those of relative scale, complexity, and hence criticality. Faults encountered at this stage are more difficult to trace, costly to remedy, and have a greater potential impact on overall program cost and schedule. Hence, a more detailed planning and direction of the test program are in order. Under these conditions, the application of systems engineering oversight and diagnostic expertise are even more essential than in the earlier stages of system development.

System Integration Test Facility. It was noted that specially designed facilities are normally required to support the integration and test of systems and their major subsystems. This is even more true for the assembly and integration of total systems. Often, such a facility is gradually built up during system development to serve as a "test bed" for risk reduction testing and may be assembled in part from subsystem test facilities.

As in the case of subsystem integration test facilities, the system integration facility must provide for extracting data from test points at internal boundaries between subsystems, as well as from the normal system outputs. It should also be designed to be flexible enough to accommodate system updates. Thus, the design of the integration facilities needed to achieve the necessary test conditions, measurements, and data analysis capabilities is itself a major systems engineering task.

13.4 DEVELOPMENTAL SYSTEM TESTING

The system integration process was seen to be focused on ensuring that component and subsystem interfaces and interactions fit together and function as they were designed. Once this is accomplished, the system may, for the first time, be tested as a unified whole to determine whether or not it meets its technical requirements, for example, performance, compatibility, reliability, maintainability, availability (RMA), safety, and so on. The above process is referred to as verification that the system satisfies its specifications. Since the responsibility for demonstrating successful system verification is a necessary part of the development process, it is conducted by the system developer and will be referred to as developmental system testing.

System Testing Objectives

While the primary emphasis of developmental system-level testing is on the satisfaction of system specifications, evidence must also be obtained concerning the system's capability to satisfy the operational needs of the user. If any significant issues exist in this regard, they should be resolved before the system is declared ready for operational evaluation. For this reason, the testing process requires the use of a realistic test environment, extensive and accurate instrumentation, and a detailed analysis process that compares the test outputs with predicted values and identifies the nature and source of any discrepancies to aid in their prompt resolution. In a real sense, the tests should include a "rehearsal" for operational evaluation.

In the case of complex systems, there are frequently several governing entities in the acquisition and validation process that must be satisfied that the system is ready for full-scale production and operational use. These typically include the acquisition or distribution agency (customer), which has contracted for the development and production of the system, and in the case of products to be used by the public, one or more regulatory agencies (certifiers) concerned with conformance with safety or environmental regulations. In addition, the customer may have an independent testing agent who must pass favorably on the system's operational worth. In the case of a commercial airliner, the customer is an airline company and the certification agencies are the Federal Aviation Administration (FAA) and the Civil Aeronautics Board (CAB).

An essential precondition to system-level testing is that component and has been successfully completed and documented. When system test failures occur in components or subsystems because of insufficient testing at lower levels, the system evaluation program risks serious delays. A required "stand-down" at this point in the program is time-consuming, expensive, and may subject the program to a critical management review. It is axiomatic, therefore, that the system test program should not be started unless the developer and customer have high confidence in the overall system design and in the quality of the test equipment and test plans.

Despite careful preparation, the test process should be conducted with the expectation that something may go wrong. Consequently, means must be provided to quickly identify the source of such unexpected problems and to determine what, within the bounds of acceptable costs in money and time, can be done to correct them. Systems engineering knowledge, judgment, and experience are crucial factors in the handling of such "late-stage" problems.

Developmental Test Planning

The provisions of the defense TEMP regarding developmental test and evaluation state that, in part, plans should

- define the specific technical parameters to be measured;
- summarize test events, test scenarios, and the test design concept;
- list all models and simulations to be used; and
- describe how the system environment will be represented.

System Test Configuration

System testing requires that the test configuration be designed to subject the system under test to all of the operational inputs and environmental conditions that it is practical to reproduce or simulate, and to measure all of the significant responses and operating functions that the system is required to perform. The sources for determining which measurements are significant should be found largely in system-level requirements and specifications. The principal elements that must be present in a system test configuration are summarized below and are discussed in the subsequent paragraphs of this section.

- *System Inputs and Environment*
  1. The test configuration must represent all conditions that affect the system's operation, including not only the primary system inputs but also the interactions of the system with its environment.
  2. As many of the above conditions as practicable should be exact replicas of those that the system will encounter in its intended use. The others should be simulated to realistically represent their functional interactions with the system.
  3. Where the real operational inputs cannot be reproduced or simulated as part of the total test configuration (e.g., the impact of rain on an aircraft flying at supersonic speed), special tests should be carried out in which these functions can be reproduced and their interaction with the system measured.
- *System Outputs and Test Points*
  1. All system outputs required for assessing performance should be converted into measurable quantities and recorded during the test period.

2. Measurements and recordings should also be made of the test inputs and environmental conditions to enable correlation of the variations in inputs with changes in outputs.
3. A sufficient number of internal test points should be monitored to enable tracing the cause of any deviations from expected test results to their source in a specific subsystem or component.

- *Test Conditions*
  1. To help ensure that contractor system testing leads to successful operational evaluation by the customer, it is important to visualize and duplicate, insofar as possible, the conditions to which the system is most likely to be subjected during operational evaluation.
  2. Some system tests may intentionally overstress selected parts of the system to ensure system robustness under extreme conditions. For example, it is common to specify that a system degrade "gracefully" when overstressed rather than suddenly crash. This type of test also includes validating the procedures that enable the system to recover to full capability.
  3. Wherever practicable, customer operating and evaluation agent personnel should be involved in contractor system testing. This provides an important mutual exchange of system and operational knowledge that can result in better planned and more realistic system tests and more informed test analyses.

Development of Test Scenarios

In order to evaluate a system over the range of conditions that it is expected to encounter in practice, as defined in top-level system requirements, a structured series of tests must be planned to explore adequately all relevant cases. The tests should seek to combine a number of related objectives in each test event so that the total test series is not excessively prolonged and costly. Further, the order in which tests are conducted should be planned so as to build upon the results of preceding tests, as well as to require the least amount of retesting in the event of an unexpected result.

Composite system tests of the type described above are referred to as test events conducted in accordance with test scenarios, which define a series of successive test conditions to be imposed on the system. The overall test objectives are allocated among a set of such scenarios, and these are arranged in a test event sequence. The planning of test scenarios is a task for systems engineers with the support of test engineers because it requires a deep understanding of the system functions and internal as well as external interactions.

The combination of several specific test objectives within a given scenario usually requires that the operational or environmental inputs to the system must be varied to exercise different system modes or stress system functions. Such variations must be properly sequenced to produce maximum useful data. Decisions have to be made as to whether or not the activation of a given test event will depend on a successful result of the preceding test. Similarly, the scenario test plan must consider what test results outside expected limits would be cause for interrupting the test sequence, and if so, when the sequence would be resumed.

System Performance Model

In describing the testing and integration of system components, a necessary element was stated to be a model of the component that predicted how it is expected to respond to a given set of input conditions. The model is usually either a combination of physical, mathematical and hybrid elements, or wholly a computer simulation.

In predicting the expected behavior of a complex system in its totality, it is usually impractical to construct a performance model capable of reproducing in detail the behavior of the whole system. Thus, in system-level tests, the observed system performance is usually analyzed at two levels. The first is in terms of the end-to-end performance characteristics that are set forth in the system requirements documents. The second is at the subsystem or component level where certain critical behavior is called for. The latter is especially important when an end-to-end test does not yield the expected result and it is required to locate the source of the discrepancy.

Decisions as to the degree of modeling that is appropriate at the system test level are very much a systems engineering function, where the risks of not modeling certain features have to be weighed against the effort required. Since it is impractical to test everything, the prioritization of test features, and hence of model predictions, must be based on a system-level analysis of the relative risks of omitting particular characteristics.

The design, engineering, and validation of system performance models is itself a complex task and must be carried out by the application of the same systems engineering methods used in the engineering of the system itself. At the same time, pains must be taken to limit the cost of the modeling and simulation effort to an affordable fraction of the overall system development. The balance between realism and cost of modeling is one of the more difficult tasks of systems engineering.

Engineering Development Model (EDM)

As mentioned earlier, the system test process often requires that essentially all of the system be subjected to testing before the final system has been produced. For this reason, it is sometimes necessary to construct a prototype, referred to as an "EDM," for test purposes, especially in the case of very large complex systems. An EDM must be as close as possible to the final product in form, fit, and function. For this reason, EDMs can be expensive to produce and maintain, and must be justified on the basis of their overall benefit to the development program.

System Test Conduct

The conduct of contractor system tests is usually led by the test organization, which is also involved in the integration-testing phase, and is intimately familiar with system design and operation. There are, however, numerous other important participants.

Test Participants. As shown in Figure 13.3, systems engineers should have been active in the planning of the test program from its inception and should have approved the overall test plans and test configurations. An equally critical systems engineering function is that of resolving discrepancies between actual and predicted test results. As mentioned previously, those may arise from a variety of sources and must be quickly traced to the specific system or test element responsible; a system-level approach must be taken to devise the most effective and least disruptive remedy.

Design engineers are also key participants, especially in the engineering of test equipment and analysis of any design problems encountered during testing. In the latter instance, they are essential to effect quickly and expertly such design changes as may be required to remedy the deficiency.

Engineering specialists, such as reliability, maintainability, and safety engineers, are essential participants in their respective areas. Of particular importance is the participation of specialists in the testing of human–machine interfaces, which are likely to be of critical concern in the operational evaluation phase. Data analysts must participate in test planning to ensure that appropriate data are acquired to support performance and fault diagnostic analysis.

As noted earlier, while system testing is under the direction of the developer, the customer and/or the customer's evaluation agent will often participate as observers of the process and will use this opportunity to prepare for the coming operational evaluation tests. It is always advantageous for customer test personnel to receive some operation training during this period.

Safety. Whenever system testing occurs, there must be a section of the test plan that specifically addresses safety provisions. This is best handled by assigning one or more safety engineers to the test team, making them responsible for all aspects of this subject. Many large systems have hazardous exposed moving parts, pyrotechnic and/or explosive devices, high voltages, dangerous radiation, toxic materials, or other characteristics that require safeguards during testing. This is particularly true of military systems.

In addition to the system itself, the external test environment may also pose safety problems. The safety engineers must brief all participating test personnel on the potential dangers that may be present, provide special training, and supply any necessary safety equipment. Systems engineers must be fully informed on all safety issues and must be prepared to assist the safety engineers as required.

Test Analysis and Evaluation

Test analysis begins with a detailed comparison of system performance, as a function of test stimuli and environments, with that predicted by the system performance model. Any deviations must trigger a sequence of actions designed to resolve the discrepancies.

Diagnosing the Sources of Discrepancies. In all discrepancies in which the cause is not obvious, systems engineering judgment is required to determine the most promising course of action for identifying the cause. Time is always of the essence, but never as much so as in the middle of system-level evaluation. The cause of a test discrepancy can be due to a fault in (1) test equipment, (2) test procedures, (3) test execution, (4) test analysis, (5) the system under test, or (6) occasionally, to an excessively stringent performance requirement. As noted previously, faults are frequently traceable to one of the first four causes, so that these should be eliminated before contemplating emergency system fixes. However, since there is seldom time to investigate possible causes one at a time, it is usually prudent to pursue several of them in parallel. It is here that the acquisition of data at many test points within the system may be essential to rapidly narrow the search and to indicate an effective priority of investigative efforts. This is also a reason why test procedures must be thoroughly understood and rehearsed well in advance of actual testing.

Dealing with System Performance Discrepancies

If a problem is traced to the system under test, then it becomes a matter of deciding if it is minor and easily corrected, or serious, and/or not understood, in which case delays may be required, or not serious and agreeable to the contractor and customer that corrective action may be postponed.

The above decisions involve one of the most critical activities of systems engineers. They require a comprehensive knowledge of system design, performance requirements, and operational needs, and of the "art of the possible." Few major discrepancies at this stage of the program can be quickly corrected; any design change initiates a cascade of changes in design documentation, test procedures, interface specifications, production adjustments, and so on. In many instances, there may be alternative means of eliminating the discrepancy, such as by software rather than hardware changes. Many changes propagate well beyond their primary location. Dealing with such situations usually requires the mobilization of a "tiger team" charged with quickly reaching an acceptable resolution of the problem.

Any change made to the system raises the question whether or not the change requires the repetition of tests previously passed—another systems engineering issue with a serious impact on program schedule and cost.

In cases where the system performance discrepancy is not capable of being eliminated in time to meet established production goals, the customer has the option of choosing to accept release of the system design for limited production, assuming that it is otherwise operationally suitable. Such a decision is taken only after exhaustive analysis has been made of all viable alternatives and usually provides for later backfitting of the initial production systems to the fully compliant design.

13.5 OPERATIONAL TEST AND EVALUATION

In previous periods of subsystem and system testing, the basis of comparison was a model that predicted the performance expected from an ideal implementation of the functional design. In system operational evaluation, the test results are compared to the operational requirements themselves rather than to their translation into performance requirements. Thus, the process is focused on *validation* of the system design in terms of its operational requirements rather than on *verification* that it performs according to specifications.

The operational evaluation of a new system is conducted by the customer or by an independent test agent acting on the customer's behalf. It consists of a series of tests in which the system is caused to perform its intended functions in an environment identical or closely similar to that in which it will operate in its intended use. The satisfactory performance of the system in meeting its operational requirements is a necessary prerequisite to initiation of production and deployment. In the case of systems built for public use, such as commercial aircraft, there will also be special tests or inspections by government agents responsible for certifying the product's safety, environmental suitability, and other characteristics subject to government regulation.

Operational Test Objectives

Operational test and evaluation is focused on operational requirements, mission effectiveness, and user suitability. The subject of operational evaluation is usually a preproduction prototype of the system. The expectation is that all obvious faults will have been eliminated during development testing, and that any further significant faults may cause suspension of evaluation tests, pending their elimination by the developer. The limitations of time and resources normally available for operational evaluation require careful prioritization of test objectives. A generally applicable list of high-priority areas for testing includes the following:

1. *New Features.* Features designed to eliminate deficiencies in a predecessor system are likely to be the areas of greatest change and hence greatest uncertainty. Testing their performance should be a top priority.
2. *Environmental Susceptibility.* Susceptibilities to severe operational environments are areas least likely to have been fully tested. Operational evaluation is sometimes the first opportunity to subject the system to conditions closely resembling those that it is designed to encounter.
3. *Interoperability.* Compatibility with external equipment, subject to nonstandard communication protocols and other data link characteristics, makes it essential to test the system when it is connected to the same or functionally identical external elements as it will be connected to in its operational condition.
4. *User Interfaces.* How well the system users/operators are able to control its operations, that is, the effectiveness of the system human–machine interfaces, must be determined. This includes assessing the amount and type of training that will be required, the adequacy of training aids, the clarity of displays, and the effectiveness of decision support aids.

Example: Operational Evaluation of an Airliner. The function of a commercial airliner is to transport a number of passengers and their luggage from a given location to remote destinations, rapidly, comfortably, and safely. Its operational con-

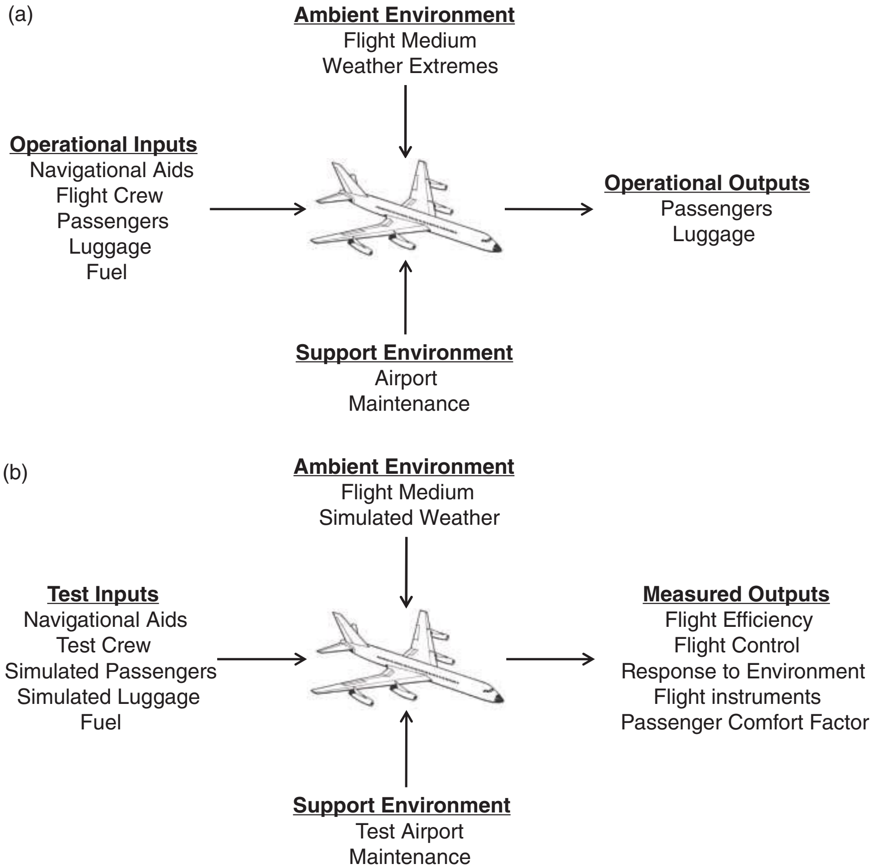

Figure 13.6. (a) Operation of a passenger airliner. (b) Operational testing of an airliner.

figuration is illustrated by a so-called context diagram in Figure 13.6a. The diagram lists the principal operational inputs and outputs, together with the ambient and support environments, that contribute to and affect the operation of the system. The principal inputs besides passengers and luggage are fuel, flight crew, and navigation aids. Numerous secondary but important functions, such as those relating to the comfort of the passengers (food, entertainment, etc.) that must also be considered are omitted from the figure for the sake of clarity. The operational flight environment includes the flight medium, with its variation in pressure, temperature, wind velocity, and weather extremes, which the system must be designed to withstand with minimum effect on its primary functions.

Figure 13.6b is the corresponding diagram of the airliner in its operational test mode. A comparison with Figure 13.6a shows that the test inputs duplicate the operational inputs, except that most of the passengers and luggage are simulated. The measured outputs include data from the plane's instruments and special test sensors to enable the evaluation of performance factors relating to efficiency, passengers comfort, and safety, as well as to permit the reconstruction of the causes of any in-flight abnormalities. The operational test environment duplicates the operational environment, except for conditions of adverse weather, such as wind shear. To compensate for the difficulty of reproducing adverse weather, an airplane under test may be intentionally subjected to stresses beyond its normal operating conditions so as to ensure that sufficient safety margin has been built in to withstand severe environments. In addition, controllable severe flight conditions can be produced in wind tunnel tests, in specially equipped hangars, or in system simulations.

Test Planning and Preparation

Test plans and procedures, which are used to guide operational evaluation, must not only provide the necessary directions for conducting the operational tests but should also specify any follow-up actions that, for various reasons, could not be completed during previous testing, or need to be repeated to achieve a higher level of confidence. It should also be noted that while there are general principles that apply to most system test configurations, each specific system is likely to have special testing needs that must be accommodated in the test planning.

The extensive scope of test planning for the operational evaluation of a major system is illustrated by the provisions of the TEMP. It requires that plans for operational test and evaluation should, in part,

- list critical operational issues to be examined to determine operational suitability,
- define technical parameters critical to the above issues,
- define operational scenarios and test events,
- define the operational environment to be used and the impact of test limitations on conclusions regarding operational effectiveness,
- identify test articles and necessary logistic support, and
- state test personnel training requirements.

Test and Evaluation Scope. Evaluation planning must include a definition of the appropriate scope of the effort, how realistic the test conditions must be, how many system characteristics must be tested, what parameters must be measured to evaluate system performance, and how accurately. Each of these definitions involves trade-offs between the degree of confidence in the validity of the result, and the cost of the test and evaluation effort. Confidence in the results, in turn, depends on the realism with which the test conditions represent the expected operational environment. The general relationship between test and evaluation realism and evaluation program cost is pictured in Figure 13.7. It obeys the classic law of diminishing returns, in which cost escalates as the test sophistication approaches full environmental reality and complete parameter testing.

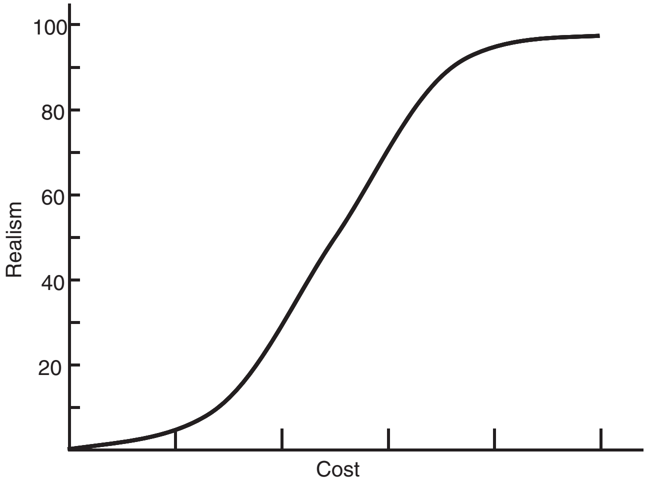

Figure 13.7. Test realism versus cost.

The decision of "how much testing is enough" is inherently a systems engineering issue. It requires a basic knowledge of the operational objectives, how these relate to system performance, what system characteristics are most critical and least well proven, how difficult it would be to measure critical performance factors, and other equally vital elements of the trade-offs that must be made. It also requires the inputs of test engineers, design engineers, engineering specialists, and experts in the operational use of the system.

Test Scenarios. System operational evaluation should proceed in accordance with a set of carefully planned test scenarios, each of which consists of a series of events or specific test conditions. The objective is to validate all of the system requirements in the most efficient manner, that is, involving the least expenditure of time and resources.

The planning of the test events and their sequencing must not only make the most effective use of test facilities and personnel but also must be ordered so that each test builds on the preceding ones. The proper functioning of the links between the system and external systems, such as communications, logistics, and other support functions, is essential for the successful testing of the system itself and must, therefore, be among the first to be tested. At the same time, all test equipment, including data acquisition, should be recalibrated and recertified.

Test Procedures. The preparation of clear and specific test procedures for each test event is particularly important in operational testing because the results are critical for program success. Also, the user test personnel are generally less familiar with the detailed operation of the system under test than development test personnel. The test procedures should be formally documented and thoroughly reviewed for completeness and accuracy. They should address the preparation of the test site, the configuration of the test equipment, the setup of the system, and the step-by-step conduct of each test. The required actions of each test participant should be described, including those involved in data acquisition.

Analysis Plan. An analysis plan must be prepared for each test event specifying how the data obtained will be processed to evaluate the proper performance of the system. The collective test plans should be reviewed to ensure that they combine to obtain all of the measures needed to establish the validity of the system in meeting its operational requirements. This review requires systems engineering oversight to provide the necessary system-level perspective.

Personnel Training

The fact that these tests are performed under the direction of personnel who have not been part of the system development team makes the evaluation task especially challenging. An essential part of the preparation for operational evaluation is, therefore, the transfer of technical system knowledge from the development organization and the acquisition agency to those responsible for planning and executing the evaluation process. This must be started at least during the developmental system test period, preferably by securing the active participation of the evaluation agent's test planning and analysis personnel. The developer's systems engineering staff should be prepared to take the lead in effecting the necessary transfer of this knowledge.

While it is to everyone's benefit to effect the above knowledge transfer, the process is too often inadequate. Significant funding is seldom earmarked for this purpose, and the appropriate personnel are often occupied with other priority tasks. Another common obstacle is an excessive spirit of independence that motivates some evaluation agents to avoid becoming involved in the preevaluation testing phase. Therefore, it usually remains for an experienced program manager or chief systems engineer in either organization to take the initiative to make it happen.

Test Equipment and Facilities

Since the focus of operational evaluation is on end-to-end system performance, only limited data are strictly required regarding the operation of individual subsystems. On the other hand, it is essential that any system performance discrepancy be quickly identified and resolved. To this end, the system developer is often permitted to make auxiliary measurements of the performance of selected subsystems or components. The same equipment as was employed in developmental testing is usually suitable for this purpose. It is to the advantage of both the evaluation agent and the developer to monitor and record the outputs from a sufficient number of system test points to support a detailed posttest diagnosis of system performance when required.

As stated previously, the conditions to which each system is subjected must be representative of its intended operational environment. In the above example of a commercial airliner, the operational environment happens to differ from readily reproducible flying conditions only in the availability of adverse weather conditions that the airliner must be able to handle safely. This fortunate circumstance is not typical of the evaluation of most complex systems. Operational testing of ground transport vehicles requires a specially selected terrain that stresses their performance capabilities over a broad range of conditions. Systems depending on external communications require special auxiliary test instrumentation to provide such inputs and to receive any corresponding output.

Test Conduct

If system developer personnel participate, they do so either as observers, or more commonly, in a support capacity. In the latter role, they assist in troubleshooting, logistic support, and provision of special test equipment. In no case are they allowed to influence the conduct of the tests or their interpretation. Nonetheless, they often can play a key role in helping quickly to resolve unexpected difficulties or misunderstandings of some feature of the system operation.

As a preliminary to conducting each test, the operational personnel should be thoroughly briefed on the test objectives, the operations to be performed, and their individual responsibilities. As noted previously, personnel and test equipment errors are often the most prevalent causes of test failures.

Test Support. Operational and logistic support of evaluation tests is critical to their success and timely execution. Since these tests are in series with key program decisions, such as authorization of full-scale production or operational deployment, they are closely watched by both developer and customer management. Thus, adequate supplies of consumables and spare parts, transportation and handling equipment, and technical data and manuals must be provided, together with associated personnel. Test equipment must be calibrated and fully manned. As noted earlier, support should be obtained from the system developer to provide engineering and technical personnel capable of quickly resolving any minor system discrepancies that may invalidate or delay testing.

Data Acquisition. It was noted in the previous paragraphs that data acquired during operational evaluation are usually much more limited than that which was collected during developer system tests. Nevertheless, it is essential that the end-to-end system performance be measured thoroughly and accurately. This means that the "ground truth" must be carefully monitored by instrumenting all external conditions to which the system is subjected and the measurements recorded for posttest analysis. The external conditions include all functional system inputs as well as significant environmental conditions, especially those that may interfere with or otherwise affect system operation.

Human–Machine Interfaces. In most complex systems, there are human–machine interfaces that permit an operator to observe information and to interact with the system, serving as a critical element in achieving overall system performance. A classic example is an air traffic controller. While data input from various sensors is automatic, the controller must make life-and-death decisions and take action based on information displayed on a control console and received from reporting pilots. A similar operator function is part of many types of military combat systems.

In such operator interactions, system performance will depend on two interrelated factors: (1) effectiveness of operator training and (2) how well the human interface units have been designed. During operational testing, this aspect of system performance will be an important part of the overall evaluation because improper operator action often results in test failures. When such errors do occur, they are often difficult to track down. They can result from slow reaction time of the operator (e.g., fatigue after many hours on station), awkward placement of operator controls and/or display symbology, or many other related causes.

Safety. As in the case of development system tests, special efforts must be exerted to ensure the safety of both test personnel and inhabitants neighboring the test area. In the case of military missile test ranges, instrumentation is provided to detect any indication of loss of control, in which case a command is sent to the missile by the range safety officer to actuate a self-destruct system to terminate the flight.

Test Analysis and Evaluation

The objectives of operational evaluation have been seen to determine whether or not the system as developed meets the needs of the customer, that is, to validate that its performance meets the operational requirements. The depth of evaluation data analysis varies from "go no-go" conclusions to a detailed analysis of the system and all major subsystems.

Under some circumstances, an independent evaluation agent may judge that a new system is deficient in meeting the user's operational goals to a degree not resolvable by a minor system design or procedural change. Such a situation may arise because of changes in operational needs during the development process, changes in operational doctrine, or just differences of opinion between the evaluator and the acquisition agent. Such cases are usually resolved by a compromise, in which a design change is negotiated with the developer through a contract amendment, or a temporary waiver is agreed upon for a limited number of production units.

Test Reports

Because of the attention focused on the results of the operational evaluation tests, it is essential to provide timely reports of all significant events. It is customary to issue several different types of reports during the evaluation process.

Quick-Look Reports. These provide preliminary test results immediately following a significant test event. An important purpose of such reports is to prevent misinterpretation of a notable or unexpected test result by presenting all the pertinent facts and by placing them in their proper perspective.

Status Reports. These are periodic reports (e.g., monthly) of specific significant test events. They are designed to keep the interested parties generally aware of the progress of the test program. There may be an interim report of the cumulative test findings at the conclusion of the test program while the data analysis and final report are being completed.

Final Evaluation Report. The final report contains the detailed test findings, their evaluation relative to the system's intended functions, and recommendations relative to its operational suitability. It may also include recommendations for changes to eliminate any deficiencies identified in the test program.

13.6 SUMMARY

Integrating, Testing, and Evaluating the Total System

The objectives of the integration and evaluation phase are to integrate the engineered components of a new system into an operating whole and to demonstrate that the system meets all its operational requirements. The outputs of the integration and evaluation phase are

- validated production designs and specifications, and
- qualification for production and subsequent operational use.

The activities constituting integration and evaluation are

- *Test Planning:* defining test issues, test scenarios, and test equipment;
- *System Integration:* integrating components into subsystems and the total system;
- *Developmental System Testing:* verifying that the system meets specifications; and
- *Operational Test and Evaluation:* validating that the system meets operational requirements.

Test Planning and Preparation

Integration and evaluation "materializes" the system as a whole and synthesizes a functioning total system from individual components. These activities solve any remaining interface and interaction problems.

Defense systems require a formal TEMP, which covers test and evaluation planning throughout system development.

System requirements should be reviewed prior to preparing test plans to allow for customer requirements changing during system development. Late injection of technology advances always poses risks.

Key issues during system integration and evaluation include

- intense management scrutiny during system testing,
- changes in test schedules and funding due to development overruns, and
- readiness of test equipment and facilities.

System test equipment design must meet exacting standards and accuracy must be much more precise than component tolerances. Reliability must be high to avoid aborted tests. Finally, the design must accommodate multiple use and failure diagnosis.

System Integration

A typical test configuration consists of

- the system element (component or subsystem) under test,
- a physical or computer model of the component or subsystem,
- an input generator that provides test stimuli,
- an output analyzer that measures element test responses, and
- control and performance analysis units.

Subsystem integration should be organized to minimize special component test generators, to build on results of prior tests, and to monitor internal test points for fault diagnosis.

Test failures are often not due to component deficiencies, but test equipment may be inadequate. Additionally, interface specifications may be misinterpreted or interface tolerances may be mismatched. And finally, inadequate test plans, training, or procedures may lead to personnel errors.

Integration test facilities are essential to the engineering of complex systems and represent a significant investment. However, they may be useful throughout the life of the system.

Developmental System Testing

Developmental system testing has the objectives of verifying that the system satisfies all its specifications and of obtaining evidence concerning its capability to meet operational requirements.

The system test environment should be as realistic as practicable—all external inputs should be real or simulated. Conditions expected in operational evaluation should be anticipated. Moreover, effects impractical to reproduce should be exercised by special tests. However, the entire system life cycle should be considered.

Test events must be carefully planned—related test objectives should be combined to save time and resources. Detailed test scenarios need to be prepared with sufficient flexibility to react to unexpected test results.

A predictive system performance model must be developed. This is a major task requiring systems engineering leadership and effort; however, an EDM is excellent for this purpose.

Developmental tests are carried out by a coordinated team consisting of

- systems engineers, who define test requirements and evaluation criteria;
- test engineers, who conduct test and data analysis; and
- design engineers, who design test equipment and correct design discrepancies.

System performance discrepancies during developmental testing must be accounted for in test scheduling, quickly responded to by a remedial plan of action.

Operational Test and Evaluation

System operational test and evaluation has the objectives of validating that the system design satisfies its operational requirement and of qualifying the system for production and subsequent operational use.

Typical high-priority operational test issues are

- new features designed to eliminate deficiencies in a predecessor system,
- susceptibilities to severe operational environments,
- interoperability with interacting external equipment, and
- user system control interfaces.

The essential features of an effective operational evaluation include

- familiarity of the customer's or the customer agent's test personnel with the system;
- extensive preparation and observation of developmental testing;
- test scenarios making effective use of facilities and test results;
- clear and specific test procedures and detailed analysis plans;
- thorough training of test operation and analysis personnel;
- fully instrumented test facilities replicating the operational environment;
- complete support of test consumables, spare parts, manuals, and so on;
- accurate data acquisition for diagnostic purposes;
- special attention to human–machine interfaces;
- complete provisions for the safety of test personnel and neighboring inhabitants;
- technical support by system development staff; and
- timely and accurate test reports.

PROBLEMS

13.1 Figure 13.3 pictures the individual and common responsibilities of design engineers, test engineers, and systems engineers. In addition to differences in their responsibilities, these classes of individuals typically approach their tasks with significantly different points of view and objectives. Discuss these differences, and emphasize the essential role that systems engineers play in coordinating the total effort.

13.2 Figure 13.4 diagrams the test configuration for a component or a subsystem in which it is subjected to controlled inputs and its response is compared in real time with that of a computer model of the element under test. When a real-time simulation of the element is not available, the test configuration records the test response to be analyzed at a later time. Draw a diagram similar to Figure 13.4 representing the latter test configuration, as well as that of the subsequent test analysis operation. Describe the functioning of each unit in these configurations.

13.3 Test failures are not always due to component deficiencies; sometimes, they result from an improper functioning of the test equipment. Describe what steps you would take before, during, and after a test to enable a quick diagnosis in the event of a test failure.

13.4 The systems engineering method in the integration and evaluation phase is outlined in the introduction to this chapter. Construct a functional flow diagram for the four steps in this process.

13.5 In designing system tests, probes are placed at selected internal test points, as well as at system outputs, to enable a rapid and accurate diagnosis of the cause of any discrepancy. List the considerations that must be applied to the selection of the appropriate test points (e.g., what characteristics should be examined). Illustrate these considerations using the example of testing the antilock brake system of an automobile.

13.6 Describe the differences in objectives and operations between developmental test and evaluation and operational test and evaluation. Illustrate your points with an example of a lawn tractor.

13.7 Define the terms “verification” and “validation.” Describe the types of tests that are directed at each, and explain how they meet the definitions of these terms.

FURTHER READING

B. Blanchard and W. Fabrycky. *System Engineering and Analysis*, Fourth Edition. Prentice Hall, 2006, Chapters 6, 12, and 13.

W. P. Chase. *Management of Systems Engineering*. John Wiley & Sons, Inc., 1974, Chapter 6.

D. K. Hitchins. *Systems Engineering: A 21st Century Systems Methodology*. John Wiley & Sons, Inc., 2007, Chapters 8, 11, and 12.