Chapter **13**

Solution Implementation

KENNETH W. MCDONALD, Ph.D.
DANIEL J. MCCARTHY, Ph.D.

> The focus of a project manager during a solution implementation is to do everything possible to insure the system delivers the value expected, on-time, and within cost.
>
> —Mr. Jack Clemons, Lockheed-Martin Corp., 2008

13.1 INTRODUCTION

Once a decision is made, we focus our attention to implementing the chosen solution design. Simply deciding to implement the selected solution does not imply that the solution will be successfully implemented. The systems engineer hopes to encounter "blue skies and smooth sailing" as indicated by the blue color depicted in the systems decision process (SDP) when implementing the solution. However, solution implementation may be the most difficult and frustrating phase of the SDP if sufficient attention is not given to detailed planning [1]. Even the best of solution designs, if poorly implemented, can fail to meet the needs of our client. Successfully implementing a solution depends on the emphasis and consideration given to the eventual solution implementation during the three phases that precede it: problem definition, solution design, and decision making. Planning for implementation must begin in defining the problem in phase one, continue throughout the design of candidate solutions, and be a consideration in the decision making phase of the SDP.

Decision Making in Systems Engineering and Management, Second Edition
Edited by Gregory S. Parnell, Patrick J. Driscoll, Dale L. Henderson

> Implementing a successful solution depends on the emphasis and consideration given to the eventual solution implementation during the three phases that precede it.

The phases of the SDP are highly interdependent as are the stages of a system life cycle. Decisions we make in one phase of the SDP inevitably impact the decisions we make in other phases of the process. How we define our problem certainly impacts our solution design space. Similarly, the solution design selected during the Decision-Making phase of the SDP will shape the plan we develop in the Solution Implementation phase. The same interdependence is true between life cycle stages. For example, the decisions (e.g., design choices) made in the Design and Develop the System life cycle stage will impact the decisions and latitude available to the project manager (PM) responsible for manufacturing the system in the Produce the System life cycle stage. For this reason, it is imperative that neither the designer nor the PM carry out his or her role independently of the other. Both should work toward the same set of objectives in trying to ensure that the system solution chosen during the Decision-Making phase is met with success. In this sense, making decisions and implementing them are essentially indistinguishable, except in terms of their order [1].

A concept map of the solution implementation phase is shown in Figure 13.1. The map depicts the interrelationship of the activities and tasks as well as the inputs and outputs within the solution implementation phase. The tasks and relationships are described in greater detail in this chapter.

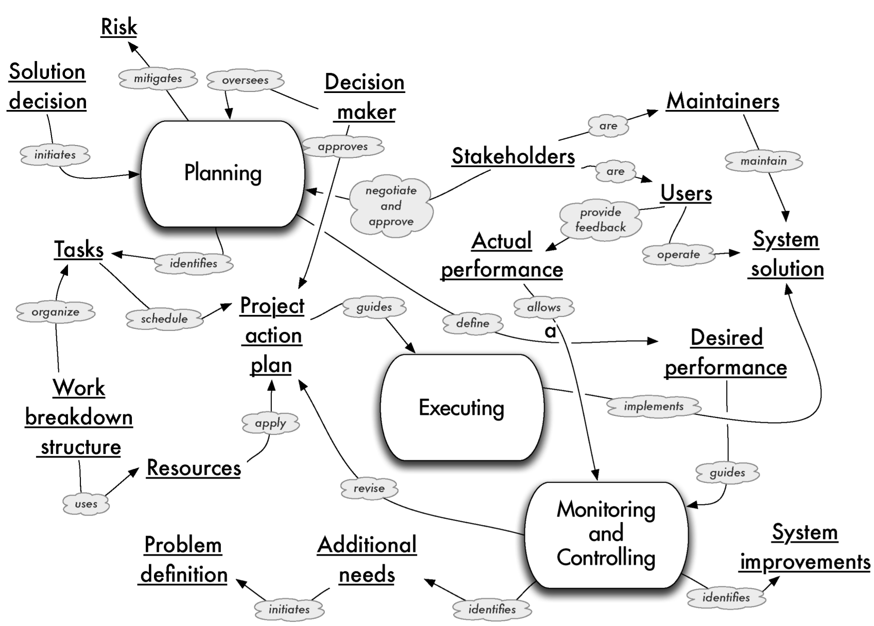

Figure 13.1 Concept map for Chapter 13.

This chapter focuses on the activities and elements that contribute to a successful implementation of the system solution. We begin by discussing the solution implementation phase and its relationship to specific stages of the system life cycle introduced in Chapter 3. Next, we recognize the implementation of a solution as a project. We then describe each of the critical solution implementation tasks and discuss various project management tools and techniques available for use in completing these tasks. Next, we describe critical solution implementation tasks as applied through selected system life cycle stages. Lastly, we provide a case study as an illustrative example to highlight many of the concepts covered in the chapter and some of the key challenges one might face in the Solution Implementation phase of the SDP.

13.2 SOLUTION IMPLEMENTATION PHASE

The Solution Implementation phase is the fourth phase of the SDP. It is probably one of the more difficult phases to accomplish successfully because the activities of this phase focus on turning the client's expectations for the system into reality. Entry into this phase may induce a false sense of completion as moving from the decision making phase to the solution implementation phase is often viewed as moving to completion. However, as mentioned earlier in the text, there always exists the possibility of returning to an earlier phase of the SDP based on evolving project conditions. Engaging the systems decision process is an iterative and cyclical in its progression as is intentionally depicted in the circular nature of the SDP diagram. Additionally, the Solution Implementation phase can be needed in any one of the seven life cycle stages listed in Table 13.1.

This phase of the SDP requires a substantial amount of coordinated effort and emphasis. The first step toward success is conceptualizing the action of "implementing a solution" as a project. By doing so, the complete arsenal of project management principles are available to help plan, execute, monitor, and control the conversion of the system solution into reality.

A *project* is a temporary endeavor undertaken to create a unique product, service, or result [1]. As a temporary endeavor, a project has a definite beginning and end, but this does not mean a project is short in duration. On the contrary, many projects may last for years, making the concept of "temporary" a flexible term. Here, conceptualizing the Solution Implementation phase as a project means that the project end is defined as producing a result in concert with the intended outcome of Solution Implementation phase. Properly moving a project through to completion requires expertise in project management capabilities. Project management encompasses the knowledge, skills, tools and techniques applied to activities in order to meet the project objectives [2]. Project management applies and integrates the project management processes: initiating, planning, executing, monitoring, controlling, and closing.

A broad, detailed exposition of project management in the context of the Solution Implementation phase exceeds the scope of this chapter. Indeed, entire texts are

TABLE 13.1 Life Cycle Stages

Life Cycle Stage	Purpose
Establish system need	• Define the problem • Identify stakeholder needs • Identify preliminary requirements
Develop system concept	• Refine system requirements • Explore concepts, examine technology readiness, and assess risks • Perform capability versus cost tradeoffs • Select concept
Design and develop the system	• Develop preliminary and final designs • Build development system(s) for test and evaluation • Test for performance, integration, robustness, effectiveness, etc. • Assess risk, reliability, maintainability, supportability, life cycle cost, etc.
Produce the system	• Acquire long-lead-time components • Develop production plan and schedule • Perform low-rate initial production (LRIP) • Perform full-rate production (FRP) • Monitor and test production items for conformance to specifications
Deploy the system	• Identify deployment locations • Provide training for installation, maintenance, and operation • Transport to chosen locations • Plan and execute logistical support
Operate the system	• Operate system to satisfy user needs • Gather and analyze data on system performance • Provide sustained system capability through maintenance, updates, or planned spiral developed enhancements
Retire the system	• Store, archive, or dispose of system

devoted to project management and in many cases full texts are devoted to particular tools and techniques used by project managers. As such, the focus of this chapter is to highlight select portions of project management processes in sufficient detail to enable a systems engineer or engineering manager to successfully complete the Solution Implementation phase.

There is more than one way to manage a project to successful completion, and one could argue that project management is more of an art than a science. Successful project managers (PM) tailor the five project management processes to fit the characteristics of a particular project. Initiating defines and authorizes the project; planning defines scope, objectives, and the course of action; executing integrates people and resources to execute the project management plan; monitoring and controlling track progress and identify shortcomings requiring action; and closing is the formal acceptance of the product, service or outcome which brings the project to an end [1].

Figure 13.2 depicts a model that illustrates how these different project management processes relate to one another. Viewing the model from left to right, the model begins with the initiating process. The initiating process develops a project charter and a scope statement. These two items establish the breadth of the project and the objectives that need to be accomplished by the project end. From the initiating process, the model flows into the planning process which requires a project team to collect and consolidate information from many sources to identify, structure, and finalize the project scope, cost and schedule. These three items along with a plan as to how, when, why, and where the available people, tools, and resources are going to be used define a complete project management plan.

The executing process that naturally follows consists of the activities used to complete the work identified in the project management plan. The planning process and the executing process are iterative, allowing the PM to reassess the plan as new information arises or the project scope requires adjusting. The monitoring and controlling processes observe project execution so that any potential problems and challenges to successful completion may be identified as early as possible. Once challenges/issues are identified, corrective action is taken to avoid schedule slippage, cost overruns, and other detrimental effects imposed by deviations from the plan. Monitoring and controlling must be performed frequently enough to allow the PM sufficient cognizance of the health of the project so that any corrective action required may be taken prior to events having an adverse impact on the project's cost, schedule, or performance. For systems decision problems, the performance element of the project plan is comprised of the total system value returned by properly implementing system functions of the qualitative and quantitative value

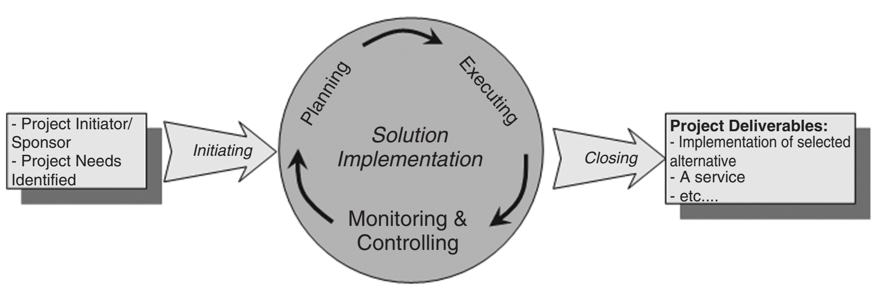

Figure 13.2 The project management process.

models. The closing process employs techniques to formally terminate the activities of project management plan and verify project completion [1].

13.3 THE INITIATING PROCESS

Initiating defines and authorizes the project by creating a project charter and a scope of work statement. The project charter is a document that provides authorization for the project. Using the project charter and the project statement of work, a preliminary project scope statement is developed. The project scope statement defines what needs to be accomplished and generally includes the following items [1, 3]:

1. Project and product objectives (clearly defined and achievable)
2. Project assumptions
3. Project constraints
4. Project requirements and deliverables
5. Project boundaries
6. Solution requirements and characteristics
7. Initial project organization
8. Initial project risks (risk register update)
9. Initial work breakdown structure (WBS)
10. Schedule milestones
11. Project configuration management requirements
12. Cost estimate
13. Solution acceptance criteria

Obviously, each project scope statement will vary and not all of the listed items are necessary for every Solution Implementation. The project scope statement may also be redefined as the situation dictates. However, changing the project scope requires the PM to review each of the subsequent processes as well, since the project scope statement is the basis from which subsequent processes unfold. If a project is large or complex enough, it may need to be decomposed into phases in order to be properly managed. This should again appear in the scope statement. Finally, a feasibility analysis is conducted during this process in order to assess the practicality of completing the project as planned.

As noted in the project scope list, identifying risks that might threaten the solution is vital to success. This activity's importance cannot be overemphasized. Identifying and analyzing project risks specific to this phase of the SDP in concert with the active risk register from earlier phases continues a common thread of vigilance through the project management process. By addressing risks at this point in the project specific to cost, schedule, logistics, liability, laws, regulations, and so on, the PM achieves a greater understanding of the impact of uncertainty going forward.

As discussed in Chapters 11 and 12, life cycle cost analysis continues into the Solution Implementation phase. An updated cost estimate that now includes many new elements specific to this phase will become part of the project scope statement. The PM uses this cost estimate to develop a realistic projected budget sufficient to carry the overall effort through to completion. As described further in Section 13.6, this budget serves as a primary indicator for tracking progress in comparison to schedule and performance.

13.4 PLANNING

The *planning* process is critical to setting the conditions for overall success of the implementation phase. Inadequate planning is a primary contributor to projects failing to achieve their schedule, cost and performance objectives. The planning process lays out the course of action to attain the scope and objectives of the project. There are several techniques and approaches to assist the systems engineer in this planning effort. The first step in the planning process is analyzing the preliminary project scope statement and project management processes in order to develop a *project management plan*. The project management plan includes the activities needed to identify, define, combine, unify and coordinate the various processes to successfully accomplish the project. The project management plan uses all the necessary subordinate plans and integrates them into a cohesive effort to accomplish the project. The subordinate plans include but are not limited to [1–3]:

1. Project scope management plan
2. Schedule management plan
3. Cost management plan
4. Quality management plan
5. Process improvement plan
6. Staffing management plan
7. Communication management plan
8. Document control management plan
9. Risk management plan
10. Procurement management plan

Each one of these subordinate plans includes a number of project management techniques which allow the specific plans to be implemented and monitored. For example, the project scope management plan is used to ensure all required work necessary to complete the project successfully is identified. It is just as important to identify what is not included in the project. The work breakdown structure (WBS) is one of the more effective techniques to help in this scoping process.

Similar to the logic of describing the functional structure of a system, the WBS is a hierarchical representation of all the tasks that must be accomplished in order to successfully complete a project. Four rules are used when developing a WBS. First, each task that is broken down to a lower level must have at least two subtasks. Second, if it is difficult to determine how long a task will take or who will do the task, it most likely requires further decomposition. Third, any task or activity that consumes resources or takes time should be included in the WBS. Fourth, the time needed to complete an activity at any level of the hierarchy should be the sum of the task times on branches below it. Properly completed, the WBS ultimately serves as the basis for identifying and assigning appropriate task responsibilities.

The WBS defines the exact nature of the tasks required to complete the project. While the hierarchical structure is certainly helpful, the WBS is not limited to one particular format but takes a number of different forms. A WBS can appear as a tree diagram (Figure 13.3) with level one tasks directly below the overall project objective followed by level two tasks [3]. In the case of the rocket example, Figure 13.3 illustrates a classic WBS.

The overall objective of this project is production of a mobile rocket. Under the level one task #5—plan production training—the two subtasks include identify test requirements and identify test location. This logical breakdown allows planners working the project scope management plan to accurately identify the requirements associated with every task supporting the production of the mobile rocket objective. Although this example may seem simplistic, the WBS tool is highly effective in supporting the project scope management plan. Other effective tools and techniques identified during the planning process include, but are not limited to, those shown in Table 13.2 [1–7].

The linear responsibility chart is an excellent technique that is used in conjunction with the WBS. When complex tasks are broken down to basic tasks, a linear responsibility chart takes those tasks and assigns personnel and organizations responsibility for each one. A linear responsibility chart shows the critical interfaces between tasks and organizations/individuals and highlights areas that require special management attention [3]. Such a chart is illustrated in Figure 13.4, which takes our rocket WBS in Figure 13.3 and assigns a number of tasks to different individuals and teams.

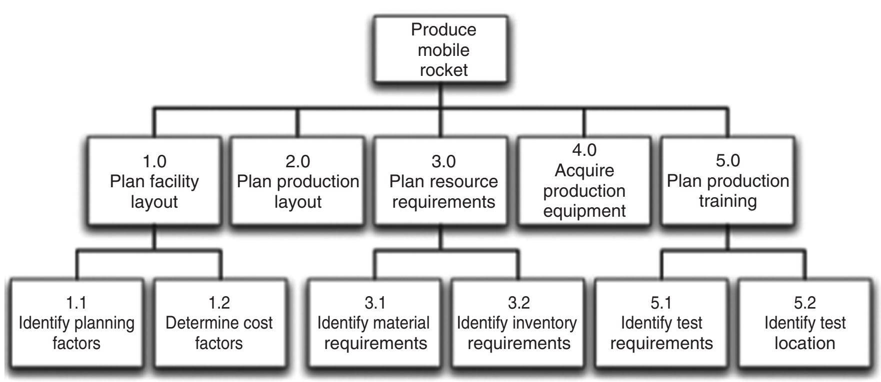

Figure 13.3 Work breakdown structure (WBS) for the rocket problem.

TABLE 13.2 Tools and Techniques Supporting the Planning Process

- Linear responsibility charts
- Scheduling
  - —Schedule milestone list
  - —Activity sequencing
  - —Activity resource estimating
  - —Activity duration estimating
- Project configuration management requirements
- Order of magnitude cost estimate
- Resource allocation
  - —Resource loading
  - —Resource leveling
  - —Constrained resource scheduling
- Staffing management plan
- Earned value analysis
- Schedule baseline
- Cost baseline
- Quality baseline
  - —Quality assurance
  - —Quality control
- Risk
  - —Risk identification
  - —Qualitative risk analysis
  - —Quantitative risk analysis
  - —Risk response
- Critical path method
- Value engineering
- Stakeholder communication plan
- Document control register
- Change control system

For clarification and understanding, a number of implied tasks are not listed on the WBS, including a project plan and budget. On the linear responsibility chart, the associated managers and teams are added to illustrate the complexity of the overall organization and the interfaces between departments. For example, establishing the project plan is the responsibility of the lead planner and his planning team. Moving from left to right in the chart, certain relationships between particular individuals and teams are evident. Obviously, the plan is exceptionally important because it is the foundation for executing the overall project. Therefore, formal approval is most likely to occur all the way through the chain of command from the project manager, through the program manager, to senior VP for programs.

Additionally, the linear responsibility chart also shows the relationships between the lead planner and his team and the other departments. For planning purposes, it is important to consult each department because they have valuable information that the planner uses. At a minimum, the lead planner needs to consult each department manager. Therefore a "3" is used to identify a mandatory consultation requirement on the part of the lead planner. You could argue that the lead planner has a mandatory requirement to consult with the team as well but that is not necessarily how the organization runs. In this case, it is assumed that the department managers are the "gatekeepers" to their departments and that the lead planner needs to consult with them versus going directly to the department team.

	Senior Vice President for Programs	Program Manager	Project Manager	Land Planner	Planning Team	Engineering Manager	Engineering Team	Operations Manager	Operations Team	Logistics Manager	Logistics Team	Lead Budget Analyst	Budget Team	Resource Manager	Resource Team
Establish Project Plan	5	6	2.4	1.2	1	3	4	3	4	3	4	3	4	3	4
Establish Project Budget	5	6	2.4	4	4	3	4	3	4	3	4	1.2	3	3	4
Plan Facility Layout	5	6	1.2	3	4	1.2	1	3	4	3	4	3	4	3	4
- Identify Planning Factors		5	6	6.2	3	3	1	3	4	3	4	3	4	3	4
- Determine Cost Factors		6	6	3	3	3	3	3	3	3	3	1.2	3	3	3
Plan Production Layout		6	6.2	3	4	1.2	3	3	3	4	4	4	4	3	3
Plan Resource Requirements		5	6.2	3	3	4	4	3	3	4	4	4	4	1.2	3
- Identify Material Requirements		5	6	3	4	4	4	3	3	4	4	4	4	1.2	3
- Identify Investory Requirements		5	6	3	4	4	4	3	3	4	4	4	4	1.2	3
Acquire Production Equipment		6	6.2	3	4					3	3	3	3	1.2	3
Plan Production Training		5	6.2	3	4	3	3	3	3			3	4	1.2	3
- Identify Test Requirements		5	6	3	4	3	3	3	3			3	4	1.2	3
- Identify Test Location		5	6	3	4	3	3	3	3			3	4	1.2	3

1 Responsible
2 Supervision
3 Consultation Mandatory
4 Consultation Possible
5 Must be notified
6 Formal Approval

Figure 13.4 Example linear responsibility chart.

Tremendous effort must go into the planning process to ensure success of the project. Many projects fail, regardless of size, when left to planners or individuals who have limited practical experience. Therefore, a successful plan must include experienced PMs on the planning team. Their expert advice brings a level of practical experience which equates to time and money savings when the final project management plan moves to execution. If possible, a PM should be identified to lead the project during the planning phase. It is not necessarily imperative that a PM be brought in this early, but it should be viewed as a “best practice” effort to improve the overall quality and efficiency of the project.

Figure 13.2 illustrates the planning process as an iterative process which includes the two processes: (a) executing and (b) monitoring and controlling. As execution begins, there are inevitable changes that occur, such as information updates, challenges with resource allocation, scheduling delays, value engineering, and so on. These changes are identified during either the executing or the monitoring and controlling processes. Once identified, the change or information is fed back into the planning phase to allow the project management plan to be updated accordingly. For the most part these changes are unpredictable. Therefore close integration of the planning, execution, and monitoring and controlling processes is essential for success. The PM must constantly seek this type of feedback in order to adjust the plan and execute accordingly.

13.5 EXECUTING

The *executing* process requires the PM team to perform a myriad of actions to put the project management plan into action. As discussed, the project management plan is made up of specific management plans that employ various techniques and tools to execute the overall management plan. The PM team must orchestrate the integration, sequencing, and interfacing of these plans. Additionally, the PM team must track the deliverables (products, results, and/or services) from each of the subordinate plans. The communications plan becomes exceptionally important here as well because distributing accurate information keeps the PM team informed of the project's progress and status.

Equally important is managing the expectations of stakeholders. The involvement of stakeholders during this process is troublesome at times. The PM should be aware of the type and amount of information that is passed along to stakeholders. Most projects do not proceed smoothly at all times because of natural occurring and frequently uncontrollable variation in project components affecting scheduling milestones. However, over the long haul these same projects are successful. Exposing stakeholders to a complete view of this variation may cause unwarranted celebration (upside variation), concern (downside variation), or over-reaction. Communicating information to stakeholders can be problematic if not properly conducted by the PM. Additional tasks during the Executing process include but are not limited to [1]:

1. Perform activates to accomplish project objectives.
2. Expend effort and spend funds to accomplish the project objectives.
3. Staff, train, and manage the project team members assigned to the project.
4. Obtain quotation, bids, offers or proposals as appropriate.
5. Obtain, manage, and use resources including materials, tools, equipment, and facilities.
6. Implement the planned methods and standards.
7. Create, control, verify, and validate project deliverables.
8. Manage risks and implement risk response activities.
9. Manage sellers.
10. Adapt approved changes into the project's scope, plans and environment.
11. Establish and manage project communication channels, both external and internal to the project management team.
12. Collect project data and report cost, schedule, technical and quality progress, and status information to facilitate forecasting.
13. Collect and document lessons learned and implement approved process improvement activities.

It is vital to ensure information that affects the project management plan be updated as quickly as possible to ensure appropriate corrective/improvement action can be implemented in a timely manner. Most project deliverables take the form of tangible items such as roads, buildings, systems, reports, devices, and so on. However, intangible deliverables such as professional training, information, professional image enhancement, and security, among others, can also be provided [1].

13.6 MONITORING AND CONTROLLING

The *monitoring and controlling* process serves the purpose to monitor the other project processes so that effective control measures can be directed to keep the project performing correctly, on time, and below cost [1]. One of the first steps to proper monitoring is identifying those essential elements requiring control, which for typical systems projects are performance, time, and cost. The PM must establish clear boundaries for control and identify the level of importance for each category. It is safe to say that the boundaries and level of importance are not the same for each project and are driven by the project's overall scope statement and stakeholder input [3, 8, 9]. Continuous monitoring in each of the subordinate plans (Section 13.4) allows the PM team to keep current with the changing dynamics of the project and to register the project's health through the prism of performance, time and cost.

The linear responsibility chart shown in Figure 13.4 underscores the need for good monitoring. Figure 13.5 demonstrates how continuous monitoring of a project motivates confidence for a PM that the project is on-track or that it requires action to restore it to this status. It is a three-dimensional "snapshot" of the project status at time τ_1 in comparison to the cost, schedule, and value estimates promised by the chosen solution at the decision gate just prior to Implementation. The dot inside the box represents the project plan's ideal state of these three project elements by time τ_1. The box imposed around this ideal location represent the acceptable levels of variation for these elements at time τ_1 under conditions in which "normal" (planned for) variation in these three elements occurs. If the current estimates for these three planning elements locate the project state within this box, then for all practical purposes the project is on-track at time τ_1 to deliver the total system value (typically represented by functionality or performance) on time and under or at cost by the end of project.

The dashed line in the figure illustrates the hypothetical development path that the project proceeded along up to time τ_1. Note that in this situation the initial cost outlay for the project at the start of the implementation phase was greater than zero. Unfortunately for the PM, the current state of the project at time τ_1, shown by the dot below the box, indicates that the project implementation has issues that must be addressed. While the project is a bit ahead of schedule ($S(\tau_1) < S^*(\tau_1)$, and cost is less than or equal to the planned estimate ($C(\tau_1) \leq C^*(\tau_1)$, the system is not achieving the planned for value return ($V(\tau_1) < V^*(\tau_1)$) at this time. Thus, some corrective or controlling action is required.

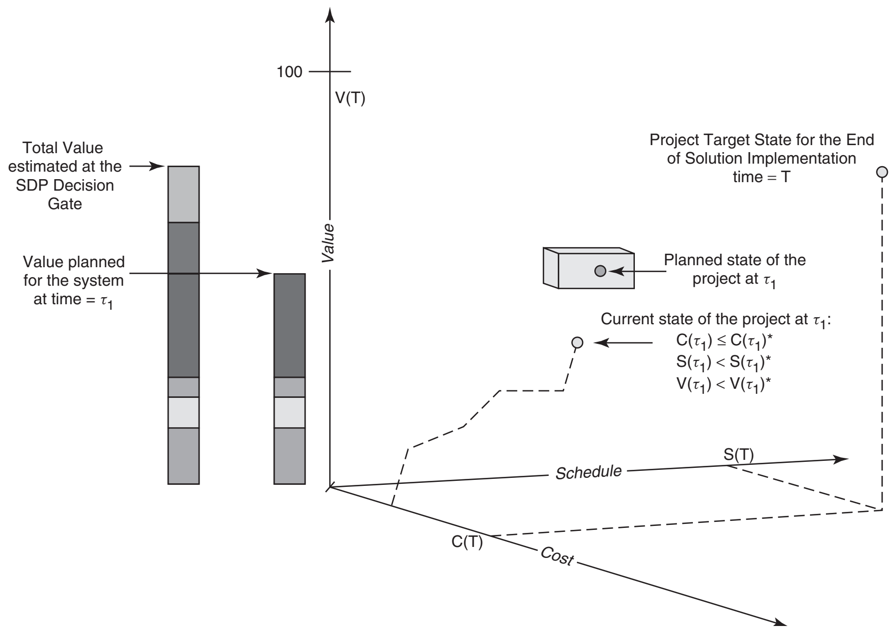

Figure 13.5 Conceptualization of project management during implementation.

An example of a corrective action taken when the schedule is behind is called "crashing the schedule." Crashing the schedule is a technique used in the critical path method to bring a project back on schedule if a particular task is going over the schedule time allowed. This technique requires placing money and/or resources (personnel and/or equipment) against the task in order to reduce its duration. The obvious outcome of using this technique is increasing the cost of resources which increases the overall project cost. The PM team understands the importance of these monitoring and controlling techniques and will ensure analysis of the results provide an accurate picture of the project's status.

Another method for monitoring a project is earned value. *Earned value* (EV) analysis is a commonly used method for measuring the overall performance of a project [1, 3]. It involves analysis of the actual work accomplished as compared to the projected budget and actual expenditures. Figure 13.6 illustrates an EV graph representing the facility layout portion of the rocket project in Figure 13.3. In this case we again see that the EV is lagging behind the budgeted amount and the actual expenditures. This leads the PM to conclude the project is behind schedule. The money spent on the project to-date exceeds the value accumulated by the system for the work performed. An earned value graph presents the PM with an effective means of monitoring critical project elements in a way that clearly highlights the links existing between cost, schedule, and value (performance).

There is more information in the EV chart shown in Figure 13.6 than simply a summary of goals being achieved or not. The stair-stepping pattern of the EV line

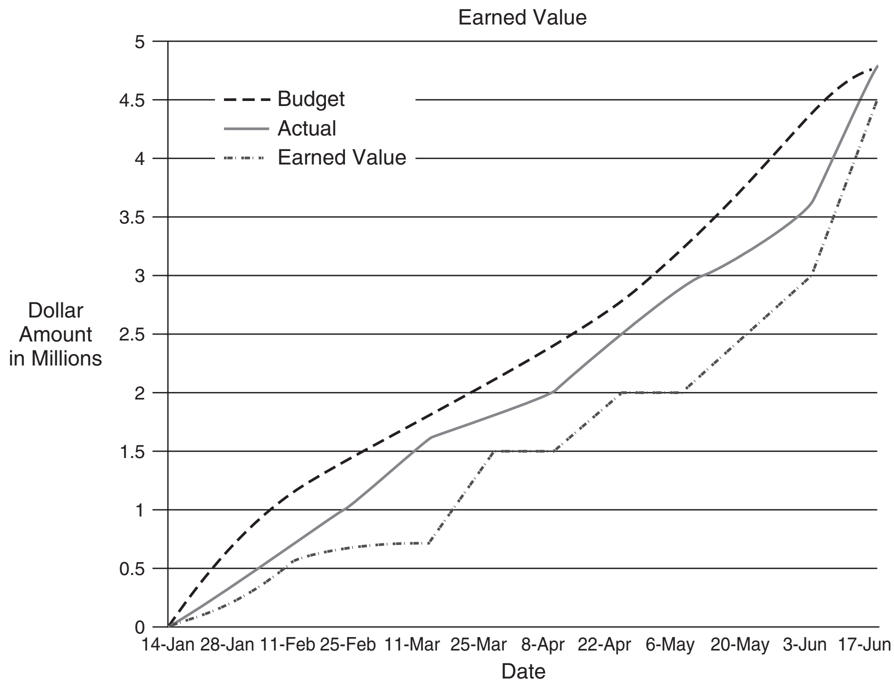

Figure 13.6 Earned value graph.

indicates surging by the contractor. While there are several reasons this pattern can occur, it often indicates challenges with the contractor. At the 11 February date, the contractor is stagnant (EV does not increase). The PM in this case responds with an increase in actual budget expenditure. More than likely, there is a possible cash flow problem with the contractor, and he required more money to get the work done. Even though the EV is below what the PM would want, there are situations where paying the contractor ahead of the EV is the best course of action for project success. This occurs most often when a good working relationship is established between the contractor and PM. Working with the contractor to keep maintaining progress instead of, for example, dismissing him from the project and pursuing litigation is probably the best course of action for the PM in this case.

The data past 11 March shows that the contractor responded with an upsurge in the EV. However, on 25 March, the contractor slows down again and EV begins to plateau. In this instance, the PM reacted differently and lowers the payment to the contractor in order to get him to respond. You will notice also that near the end of the facility layout portion of the project, there is great gain in EV for the overall project. This is a very typical pattern for a systems implementation project. Engineers have a tendency to estimate a concave shape over the project, indicating optimism in terms of how quickly value (functionality, performance) can be developed. In reality, the EV curve assumes a convex shape because of re-work imposed by test failures, delayed schedules and response surges in activity, and cost-conserving measures put in-place to mitigate the threat of running out of budget before the expected (or required) value has been delivered.

Overall, the monitoring and controlling process is concerned with but not limited to the following list of activities [1]:

1. Comparing actual project performance against the project management plan
2. Assessing performance to determine whether any corrective or preventive actions are indicate and then recommending those actions as necessary
3. Analyzing, tracking, and monitoring project risks to make sure the risks are identified, their status is reported, and the appropriate risk response plans are being executed
4. Maintaining an accurate, timely information base concerning the project's products and their associated documentation through project completion
5. Providing information to support status reporting, progress measurement and forecasting
6. Providing forecasts to update current cost and current schedule information
7. Monitoring implementation of approved changes when and as they occur
8. Updating the Project Management Plan accordingly
9. Updating stakeholders and their status with regards to the project as appropriate
10. Conducting PM team meetings to develop an updating rhythm for analysis, tracking, and monitoring
11. Conducting proper change control methods
12. Approving/rejecting change requests
13. Updating project management plan
14. Updating project scope statement
15. Approving corrective actions
16. Approving preventive actions
17. Monitoring deliverables

13.7 CLOSING

The *closing* process involves all the necessary administrative and contractual closing procedures to ensure proper project closeout. This process includes all the subordinate project plans as well as any phases (for complex/large projects) that are associated with the project. The administrative closing procedures are those procedures and project relationships dealing with different aspects of the project [1]. A major portion of the administrative work required here includes documentation archiving (see Section 13.4). This process is exceptionally important for possible future legal inquiries. Many contracts start out with good intentions on the part of both parties, but more often than not, there are lawsuits brought against a company by a contractor for numerous reasons. A good document control plan and register assists the PM team respond to litigation. The contract closure procedure includes those activities needed to complete contractual obligations. One aspect of this process that is important is ensuring proper contractor documentation completion. For example, near the end of most projects, the daily requirements of progress reports sometimes get overlooked, especially if a contractor is near 100% paid. If a PM fails to hold the contractor to full compliance for contractual requirements such as progress reporting, the PM is liable. It is imperative that the PM team ensure all aspects of close-out procedures are adhered to and followed.

13.8 IMPLEMENTATION DURING LIFE CYCLE STAGES

After careful analysis of the implementation phase, it is important to understand its role during all stages of a systems life cycle. The life cycle is a dynamic living model that requires thorough understanding of the SDP and how it is used in every stage of the life cycle. As such, the implementation phase of the SDP takes different approaches to meet the particular needs as dictated by the solution and the life cycle stage. The guidelines presented in the previous sections are still relevant here as well but are adjusted to meet the unique circumstances presented by the situation. The following sections will focus on the implementation phase as it applies to the Produce the System, Deploy the System, and Operate the System stages of the system life cycle.

13.8.1 Implementation in "Produce the System"

The fourth life cycle stage and the first to be discussed in this section is "Produce the System." One inherent and necessary objective of the Design and Develop stage is to create the design such that the system and all of its elements are produced effectively and efficiently. For most physical systems, the primary objective of the production stage is to turn the system solution into reality. During this time, the PM team handles any design changes justified by requirements or by market demands [6]. Inspection and testing of the product occurs in this stage. The PM team is required to validate that the product meets the specifications identified in the requirements. Project management techniques as discussed earlier can be integrated with the SDP and systems engineering procedures and practices to organize and implement a production or manufacturing requirement. This can be an exceedingly complex set of activities requiring an excellent PM team [10].

Planning for "Produce the System" To achieve success during the produce the system life cycle stage, detailed planning must precede the execution of production. Planning the implementation of the system solution helps ensure that the system solution does get implemented in such a way that the expected performance is realized [11]. For the producing the system stage to successfully occur, it is supported by specific planning actions such as those discussed in Section 13.4. The purpose of planning is not to eliminate uncertainties, but rather to prepare as much as possible for anticipated events and to adjust when unexpected events occur. By preparing a plan, we establish a common baseline from which to coordinate activities. Detailed planning is essential for the producing the system stage because it minimizes the risk inherent in projects.

The systems engineer must realize that although we are focused on producing the system, the majority of planning for deploying and operating the system also occurs in this stage. That is to say that the planning of one stage is performed in relation to other life cycle stages. For example, how the system solution is deployed is considered when planning how the system solution is produced. Likewise, how the system solution operates is considered when planning how the system solution is deployed. Planning must occur not in a vacuum but concurrently and with the end state in mind.

Executing for "Produce the System" When planning the production of the system solution is complete, the next step is to execute the production plan. Resources are required to execute the plan as well as support the deployment and operation of the system solution. Execution includes resource management for implementation of the system solution. Attention is now given to defining resource requirements for production to begin. Resources include everything needed to accomplish each task during production: people, money, supplies, inventory, equipment, facilities, infrastructure, external services, and technology, to name a few. Spreadsheets and cost-estimating models are techniques that help organize and layout resource requirements (amount and time).

As an example, Microsoft® Project has the ability to display all of this information. This enables the PM to identify a task, its duration, start and finish dates, resources, resource requirements, and when those resources are required. The products of this stage are successfully executed if resources are available. As this stage is carried out, assessment and control of the stage must occur. During the execution of this stage, the key tasks of monitoring and controlling are critical to ensure that the system will function as expected and executed as planned.

Monitoring and Controlling for "Produce the System" The SDP includes measures and methods that allow for monitoring and controlling system solution performance during all life cycle stages. A feature of Microsoft® Project is that it may be used to measure progress in terms of time, budget, and project performance. Since Microsoft® Project does not monitor progress in terms of system performance, other methods (simulations and testing) are used to assess system performance. The project action plan allows the PM team to compare actual and planned task durations, resource usage, and expenditures at any level of activity. The project action plan is used as a control document by measuring comparisons. These comparisons dictate what project performance information is monitored. This gives the PM team the ability to control the project and take corrective action if the project is not proceeding according to plan [5]. The elements that are controlled during the produce the system stage include but are not limited to:

1. Cost
2. Schedule
3. Risk
4. System performance requirements
5. Design changes
6. Production and manufacturing process
7. Quality
8. Reliability
9. Safety of product and personnel

Additionally, there may be others defined by the PM and/or stakeholders.

13.8.2 Implementation in "Deploy the System"

At the end of the produce the system stage, the system solution enters the fifth life cycle stage: Deploy the System. The PM team must receive prior approval from the decision maker that the system is ready to proceed to the next life cycle stage. Upon approval, the manufactured system solution becomes the deployment system, which delivers fully operational system solutions to users.

Planning for "Deploy the System" The purpose of the Deploy the System stage is to transfer the system solution from the development facility to the operational location and to establish full operational capability. Distribution facilities, marketing, and sales organizations are required to support the implementation of this stage. Use of these and other resources are planned in great detail. In this section, the planning elements and methods for successful deployment of the system are addressed.

Deploying the system solution is a process that must also be planned for in order to meet stakeholder objectives. Elements of the development process requiring planning include but are not limited to:

1. Geographical distribution
2. Deployment schedule
3. The type and number of system components at each location
4. Logistical support
5. Type training required (e.g., installation, maintenance, and operation)
6. Resource requirements to support the required training
7. Testing

Marketing elements are keys in determining the best locations to deploy the system solution. The strategy for deployment and support requirements are identified. Acceptance testing or a full operational testing occurs prior to moving the system solution into the next life cycle stage. Acceptance testing often results in minor adjustments to system operation [12]. This testing is conducted with stakeholders present. Stakeholders ensure that the system solution continues to operate as intended according to desired preferences. Risk is always present, and it is possible that the events and activities inherent in the deployment process may affect the system's operational performance. As such, a risk management plan is developed and contains elements as outlined earlier.

An operational test or demonstration is another type of test which is very useful in communicating the operability of the system. Before testing is complete, operational testing in the environment and under conditions in which the system solution operates are performed. Another critical activity in this stage that requires planning is training. Training produces trained installers, operators, and maintainers to support the operations of the system solution. Training resources are planned and phased into the project as required. The process planning methods described during the Produce the System stage are applied in this stage as well.

Executing for "Deploy the System" Deploying a system solution requires excellent documentation concerning how to install the system in its operational environment. In some cases, special analysis and testing is carried out so as to assure field operability. Training is also designed and delivered in formal well-developed programs. The required training is scheduled, not only in this stage but also in the Operate the System stage. Subsequently, training specialists are part of the team of personnel required to execute the project.

Upon completion of planning for deployment, the plan is carried out. Resources are present to execute the deployment process. Many products are late to market simply because of being starved for resources. When this occurs, the costs can be enormous. To execute the deployment plan correctly, timing and resources are critically important. The right kind and the right amount of resources are required and on time when needed. When an increase or adjustment in resources is needed, senior leaders must prepare to respond quickly. Some examples of resource requirements in this stage are listed but not limited to:

1. Competent trainees
2. Users to perform operational testing
3. Tentative user locations
4. Transportation assets
5. Training equipment

Monitoring and Controlling for "Deploy the System" The need to exert proper control over deploying the system mandates the necessity for monitoring and controlling the proper activities and elements during this stage. According to Table 13.1, we are verifying that the system solution meets system performance measures. In order for the system solution to perform during full operational capacity as intended, it is monitored, assessed, controlled. The details of the deployment plan identify additional elements to control commensurate to the unique deployment requirements. As the system performs outside of its intended functions, corrective action is taken to bring performance in conformity with stakeholder preferences. The fundamental items controlled are time, cost, and performance, which were discussed earlier in the chapter. It is prudent and necessary to perform testing of the system functions prior to the deployment process, during the deployment process, and upon arrival at user location if possible. These efforts ensure that the system operates as intended upon reaching the end user.

13.8.3 Implementation in "Operate the System"

A system is not considered successful until it is successfully implemented and is turned over to the user. A full-scale operation generally occurs in the sixth life cycle stage: Operate the System. This stage begins as users receive the first operational systems. The objective of the operation stage is to fulfill the stakeholder's need. The stakeholders' needs are fulfilled when the system solution is realized. Planning, monitoring and assessment and control during this stage is critical to ensure that the system solution is successfully implemented into full-scale operation. It is hoped that the events and activities preceding this stage were conducted thoroughly.

Planning for "Operate the System" In the Operate the System stage, the system has attained full operational capability. This means that the system solution operated and maintained in conformance with user requirements. This also includes satisfying the user, gathering data on system performance, sustaining and maintaining operability, adding enhancements to the system solution, and identifying improvements for future implementation. During this long period in which the system is operational in the field, emphasis should be placed on the continuous measurement of the system's performance.

The planning performed in this stage centers on operating and maintaining the system and identifying system improvements. Preliminary planning for retiring the system also occurs here. Since emphasis during this stage is on system performance, data are gathered to assess system performance. Measurement procedures range from simple manual data sheets to automated sensors that record operational status continuously [2]. Some companies try to maintain contact with consumers through hot lines, reports of satisfaction, and online usage monitoring. To support these methods, it is important that procedures on how to install and sustain a performance measurements program be explicitly defined. Emphasis should be placed on maintaining the system solution to ensure that it continues to function in accordance to its operational requirements. Maintainability is the ability of the system solution to be retained in or restored to a performance level when prescribed maintenance is performed [5].

Executing for "Operate the System" The system operators and maintainers execute the functions of operating and maintaining the system solution. The resources are adequately planned and acquired when needed. These individuals are involved in the collection of data to assess whether the system functions in accordance with its intended design. Trained individuals are assigned to perform periodic evaluations of the system as it operates in its natural environment and to perform the necessary maintenance needed to sustain system performance. Some examples of resource requirements include, but are not limited to, the following:

1. Data collection methods
2. Data collection equipment
3. Personnel resources

Monitoring and Controlling for "Operate the System" Once a system is operating, it is controlled; that is, its operation is regulated so that it continues to meet expectations [11]. Continual operational evaluation and testing of the system is a method used to identify what is controlled. System audits are also performed. This method is used after the recommended alternative is implemented to see how the actual system performs, whereas an operational evaluation can occur prior to operational implementation and during actual implementation. System improvement relies on the identification of deviations between the actual operation of the system and what is termed as normal or standard. After these deviations are pinpointed, their causes are identified in order to correct malfunctions. Feedback is another valuable tool. Users and maintainers provide feedback about what they like and do not like, which is used during refinement to make changes in the design, leading to upgrades of the system [13].

A set of specific evaluation test requirements and tests are evolved from the objectives and needs determined in the final requirements specifications. Each objective and critical evaluation component is measured by at least one evaluation test instrument. If it is determined that the resulting system product can no longer meet stakeholder needs, the problem enters phase one of the SDP Problem Definition and repeats the procedure set forth in this text.

Solution Implementation can often be one of the most difficult and time consuming challenges faced by the systems engineer. Even the best of solutions, if not properly implemented, can fail to meet the needs of the stakeholders. History is full of engineering projects that have come in late, over budget, or that failed to perform as intended. Thankfully, there are many project management tools and techniques that are available to assist the systems engineer during the Solution Implementation phase of the SDP. These tools and techniques can be used in any of the lifecycle stages of a system and are often tailored to fit the needs of the specific lifecycle stage as described in this chapter. However, as has been said project management is more of an art than a science. Many times, successful solution implementation comes down to the ability of the systems engineer to work with the many stakeholders involved in the Solution Implementation phase of the SDP. The importance of these people skills and the involvement of leaders early and often are highlighted in the case study that follows. The case study describes the implementation of an automated system for college applications in the early years of information systems.

CASE STUDY: DESIGN AND IMPLEMENTATION OF THE PROSPECTIVE STUDENT INFORMATION SYSTEM

Bobbie L. Foote, Professor Emeritus, University of Oklahoma

The Beginning

In 1967 the University of Oklahoma decided to venture into new computer technology to consider automating their student admissions system. The University at this time was making a commitment to use commercial computing power instead of continuing full-scale development of their own computer, the OSAGE, which was designed and built from conception by University faculty. They had realized that they did not have the resources to develop the myriad software applications necessary to run the University information needs based on the one of a kind operating system for the OSAGE. The University had bought an IBM 650 computer by 1957 and after 1962 had purchased an IBM 1620 and had started developing the software and operating specialists to run a punch card operation.

A young professor in Industrial Engineering was selected to design and implement the new system. A team of administrators was formed to advise him. A bright young graduate student who was an employee of a local air conditioning manufacturing firm and had an undergraduate degree in Industrial Engineering from the University of Oklahoma was hired to gather information and write prototype software to show feasibility.

In the early planning stages, a standard stakeholder analysis was performed. A survey was sent out to determine potential users. These users were interviewed as to what information they wanted. This led to a listing of actions and decisions that they needed to make. A standard flowchart was developed to look at timing.

This was an admissions system. So, the first day of enrollment in the summer or fall served as one of the drop-dead time goals. To be admitted, the student had to supply high school transcripts, a copy of a medical exam, records of immunizations, religious preferences, housing preferences, and their interest in various academic programs. There were deadlines for these submissions. If a student failed to submit the required information, the University manually went through the file and noted omissions and sent out a letter with boxes marked to indicate the information shortage. Some students went through several cycles. Students who showed late interest were a problem as the University needed to expedite their record analysis and communicate decisions quickly. Appeals had to be handled.

Another problem was the exponential growth of the applications as more and more people realized the value of a college education. This led to a big problem with applications for a scholarship as the donors and other supervisors had deadlines to make decisions. Scholarship applications required extra information such as letters of recommendation and student essays. This put a real burden on the manual system.

As a part of the routine information gathering, histories of other processing applications were sought. It was at this point an unpleasant realization occurred. Over 90 percent of the past projects had either failed or never even started. An investigation of these failures ensued.

The Pause

At this point a series of meetings with upper level managers including the Vice President (later Provost) for Academics was initiated. A series of surveys were sent out to get information about these failures. The returns indicated a variety of reasons: the system did not meet their needs, low budget, hence a requirement that people who used the system kick in from their resources, awkward means of interacting with the system, inability to modify the systems once flaws were found.

During this time the student research assistant created a system to handle the requirements as understood. This system consisted of: a set of decisions required by each user, the time deadlines, and the information required by the decisions, a set of actions required that were to be automated such as letters of acceptance, letters asking for more information, and letters reminding students of deadlines; and most importantly a system flow diagram that pinpointed interactions among future users of the system. Student letters asking for information were separated into two groups: (a) one that required a form answer and (b) more complex issues that required a personal answer. This list was reviewed by University software developers and feasibility was assessed. IBM had an interest and donated time to help assess feasibility.

The above study highlighted the amount of cooperation required from University functions and their clerical personnel to use the system successfully. Everyone realized that a training package had to be developed before the system could be activated.

The Implementation Plan

The chairman of the IE department and the study manager had a series of meetings to address the problem of implementation. The need for training was determined to be an easy problem to design and manage. The major problem was the financial issue and the perceived quality issues that plagued previous projects. A series of personal interviews with users and system designers revealed a further problem: the selective memory of users. After a new system was begun users would frequently say that they had unmet needs that had been communicated to designers but ignored in the final product. Users also had little concept of the changes in operating protocol that would be needed.

The following solution emerged. If you wanted help, you had to pay, but the Provost would match the dollars from the managers' budget from his own. The final part of the implementation protocol was to circulate the plan to the managers (users) and then meet personally with them. The provost required one of two responses: The plan suits my need or it fails and here is what is missing. Time deadlines for responses were given. A new plan was devised and the same process was carried out. When no gaps were found by users, then there was a final document circulated which required the signature of the user. They either signed saying that the system met their needs or the system failed. If the system failed, they had to give reasons. Those users with problems met with the study manager and the provost to discuss their problems. In front of the Provost, managers were candid and cooperative. After this meeting, every user who had not signed, signed. The head of the computer center took responsibility for developing and testing the software in time for implementation just after Christmas.

The Outcome

The resulting system was programmed and proved to be a rousing success. The offload on users and their clerical staff was huge. The growth of the University was enabled and further data processing projects grew rapidly. PSIP proved to be the basis of the complex information system that encompasses the University today.

ILLUSTRATIVE EXAMPLE: SYSTEMS ENGINEERING CURRICULUM MANAGEMENT SYSTEM (CMS)—IMPLEMENTATION

Robert Kewley, Ph.D. U.S. Military Academy

Planning For Action

The CMS project encompasses the design, development, deployment, training, use, maintenance, improvement, and retirement of the CMS system for the Department of Systems Engineering. The department's CIO has oversight of this project, and development will take place using a combination of internal IT staff and capstone students. The CIO, with approval from the department head, will schedule a phased development and deployment of the system to support the teaching calendar and ABET accreditation requirements. The goal is to have all components of the system deployed within 13 months in order to support ABET data collection beginning in August 2007. Deployment during the 2006–2007 academic year will allow the department faculty to learn the system, use it, and provide feedback to the development team as to how it can better support their needs. The development team created a phased project plan using project management software. Their first step was to load all of the functions from functional analysis into the project plan. The next step was to break those functions into phases so that development could proceed in accordance with the academic calendar and ABET requirements. In addition, some functions naturally precede others. For example, the Develop Program function relies on the use of course data. Therefore, it made sense to sequence the Develop Course and Execute Course functions before the Develop Program function. The Integrate Department Academic Operations function was least critical to ABET assessment, so it was sequenced last. Figure 13.7 shows a breakdown of the project by phases. Once the development team had developed a project plan, they took a detailed look at the Phase I activities and developed a work breakdown structure using project management software. They first had to look at each function and identify any additional development tasks needed in order to support implementation of each function. For example, the function Develop Course Strategy required the addition of two supporting tasks, Develop Course-Level Data Tables and Develop Course-Level Portal. Once all tasks were added, they had to be scheduled in order to meet Phase I deployment timelines—to include an acceptance test at the end of Phase I on 1 August 2006. In addition, some tasks required other tasks to be completed before they could be started. In order to implement the task Develop Interface for Course Objectives, the Develop Course-Level Data Tables task had to be completed so that the objectives could be stored in the database.

EXECUTION

In order to begin execution of the project plan, the team identified available development resources from the department's internal IT staff and assigned development engineers to the identified tasks. A portion of the resulting work breakdown structure with tasks assigned to different developers is shown in Fig. 13.8.

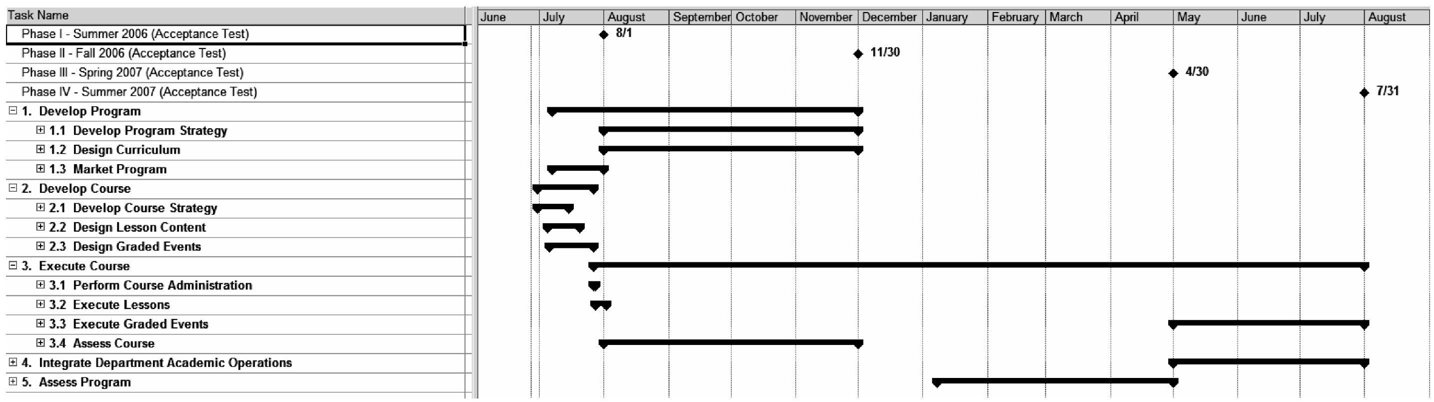

Figure 13.7 CMS project plan by phase.

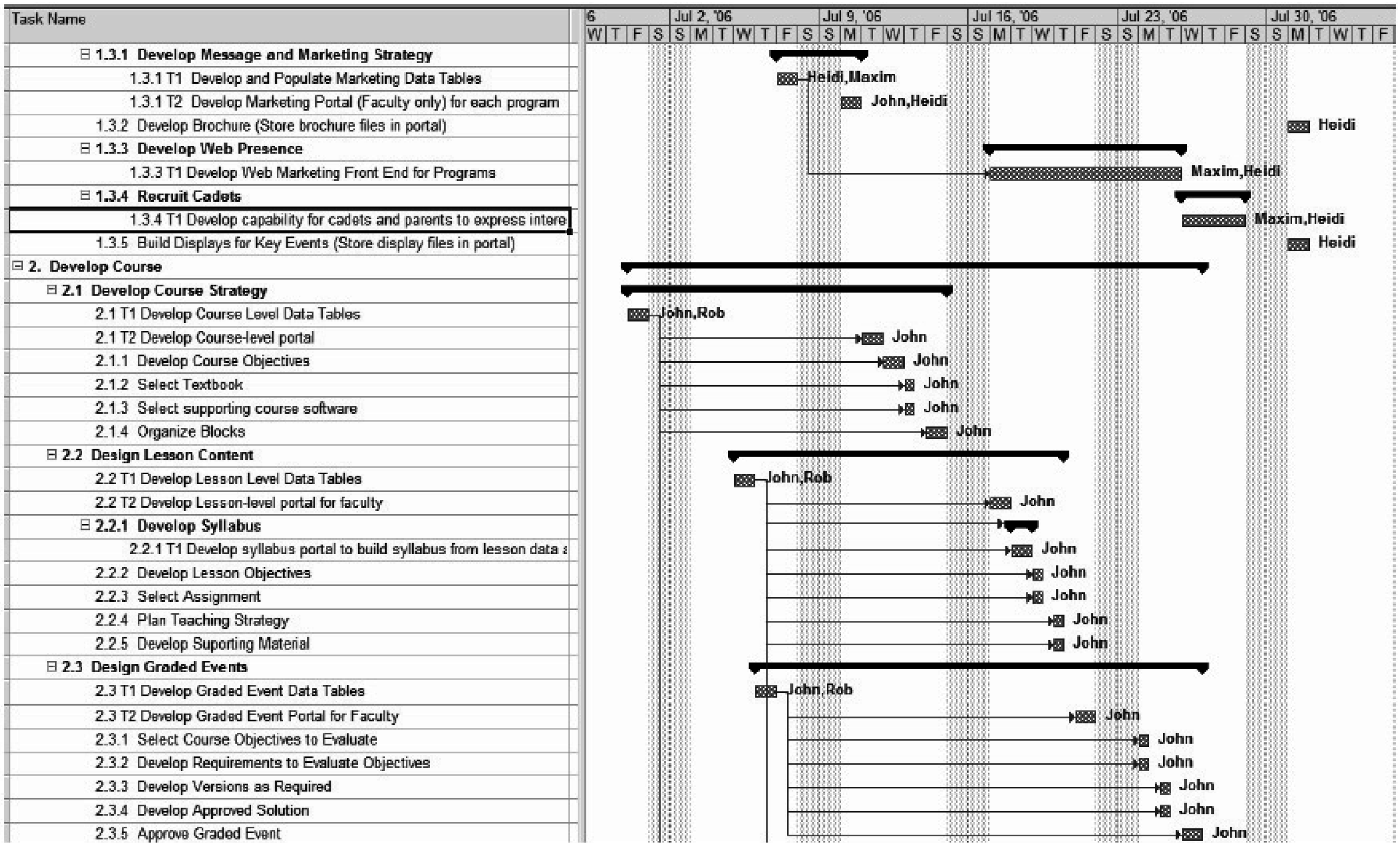

Figure 13.8 Detailed work breakdown structure for Phase I tasks.

ASSESSMENT AND CONTROL

The purpose of monitoring is to ensure that the system enables execution of the supported functions in accordance with the system design goals and expectations of the stakeholders. In order to monitor the performance of the CMS across all phases of the system life cycle, the development team incorporated three different techniques.

Acceptance Testing. At the end of each phase of the development process, the program directors will do acceptance testing of the system to ensure it meets the design requirements specified in the needs analysis. Deficiencies will be noted and corrected. If serious enough, these deficiencies could delay deployment of the system until its performance is acceptable.

Online Feedback. The development team will incorporate an online feedback capability for users of the CMS system. At any time, users of the system, including cadets, instructors, and program directors, can follow a link to a free text feedback page that allows them to provide feedback about how the system meets their needs. This feedback will be stored in the system database and reviewed monthly by the development team. Based on that feedback, they can change and update the CMS as required.

Focus Groups. At the end of each semester, the development team will conduct a focus group with CMS users in order to get a more structured assessmente of how the system is performing in accordance with stakeholder needs. The results of these focus groups will be integrated into the development and maintenance plan.

As the department continues to develop and use the CMS, the assessment feedback will guide decisions about maintaining or upgrading the system. They will also be able to assess adequacy of user training. At some point during the life cycle, information from the assessment process will indicate that the system is beyond its useful life. This will lead to retirement of the CMS and possible development of a new system to perform the curriculum management function.

13.9 EXERCISES

13.1. Describe the Solution Implementation phase.

13.2. During the Solution Implementation phase, what signifies the completion of the project? Explain your answer.

13.3. Draw and describe the project management process.

13.4. Why is the Initiating Process so important?

13.5. There are several techniques used to assist in the project manager in developing a project plan. List one method and explain how it works.

13.6. Why is project management considered more of an art than a science?

13.7. Define the tasks within the Solution Implementation phase. Which task is the most important? Explain your answer.

13.8. What is a work breakdown structure? Why is it important to a PM?

13.9. What is linear responsibility chart? Why is it important to a PM?

13.10. Why is the Closing Process so important?

13.11. Explain the relationship of the Solution Implementation phase and the project life cycle stages.

13.12. The case study reveals a common problem among users and stakeholders: unmet user needs. What do you recommend to ensure that user and stakeholder needs are met?

13.13. Why is the Solution Implementation phase considered the most difficult phase of the SDP? Explain your answer.

13.14. Reviewing the case study, what do you think was the primary reason for solution implementation success? Why?

REFERENCES

1. Project Management Institute. *A Guide to the Project Management Body of Knowledge*, 3rd ed. Newtown Square, PA: Project Management Institute, 2004.
2. Kerzner, H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 9th ed., Hoboken, NJ: John Wiley & Sons, 2006.
3. Meredith, JR, Mantel, SJ. *Project Management*. New York: John Wiley & Sons, 2006.
4. van Gigch, JP. *System Design Modeling and Metamodeling*. New York: Plenum Press, 1991.
5. Mantel, SJ, Jr., Meredith, JR., Shafer, SM, Sutton, MM. *Project Management in Practice*. Hoboken, NJ: John Wiley & Sons, 2001.
6. Forsberg, K, Mooz, H, Cotterman, H. *Visualizing Project Management*, 2nd ed. New York: John Wiley & Sons, 2000.
7. Smith, PG, Reinertsen, DG. *Developing Products in Half the Time*, 2nd ed. New York: John Wiley & Sons, 1998.
8. Fisk, ER, Rapp, RR. Engineering Construction Inspection. Hoboken, NJ: John Wiley & Sons, 2004.
9. Palmer, D. *Maintenance Planning and Scheduling Handbook*, 2nd ed. New York: McGraw-Hill, 2006.
10. Eisner, H. *Essentials of Project and Systems Engineering Management*, 2nd ed. New York: John Wiley & Sons, 2002.
11. Athey, TH. *Systematic Systems Approach: An Integrated Method for Solving Problems*. Boston, MA: Pearson Custom Publishing, 1982.
12. Sage, AP, Armstrong, JE, Jr. *Introduction to Systems Engineering*. New York: John Wiley & Sons, 2000.

13. Buede, DM. *The Engineering Design of Systems: Models and Methods*. New York: John Wiley & Sons, 2000.
14. Blanchard, BS, Fabrycky, WJ. *Systems Engineering and Analysis*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1998.