Chapter 3
Information Technology Project and Systems Life Cycles: Project Management and Team Activities

Basic project management tools and techniques are the same regardless of whether one is involved with hardware, software, services, or in an IT project that combines them all. To successfully plan, implement, and complete a project, work breakdown structures, schedules, risk analyses, and other commonly used tools of the trade are needed. The tools are the same—but the projects are different.

Different projects require different technical and management approaches. The application of traditional project management tools and techniques is generally less successful in the IT industry. This is not because these techniques are inapplicable but rather because the unique characteristics of IT projects are not taken into account. In other words, we apply the tools assuming that IT projects have the same characteristics as engineering or construction projects and expect them to respond in the same way. Therein, lies the problem—IT characteristics such as risks, schedule requirements, customer needs, market-driven pressures, and even the competitive environment differ from those found in the traditional project world. So the challenge is not to learn unique tools and techniques but to learn how to apply the traditional ones in a different environment. To do that, we have to understand the uniqueness of IT projects and their products.

A project has both a life cycle and a systems development life cycle during which a number of typical activities occur. The key to planning and managing IT projects is to understand these life cycles, how they fit together to accomplish the project and product requirements, and what activities the project manager is responsible for during the entire process. The project life cycle (PLC) encompasses all the activities of the project, while the systems development life cycle (SDLC) is focused on accomplishing the product requirements.

This chapter discusses how the systems development life cycle fits into the project life cycle and the differences between IT projects and traditional engineering or construction projects. Since one of the principal differences between IT projects and others are the risks involved, a comparison of the risk differences is made in this chapter. A more complete discussion of risk and risk management is found in Chapter 7.

Understanding the environment of any project requires a good understanding of its life phases, what occurs during each phase, and what the project manager is required to do to successfully accomplish these phases.

The Project and Systems Development Life Cycles

Project managers have long known that projects have a life cycle much like a biological life cycle. That is, the project starts slowly, builds steadily to a peak resource and activity level, and then rapidly decreases in resources and activity to the closeout point.

There is no standard nomenclature for naming phases. Often, the initiation phase will be called the concept phase. Planning is sometimes called development, monitor and control are often referred to as implementation, and the word *termination* can replace the term *closeout*. The phase names I use in this book are chosen as a convenient way to differentiate between the project and systems life cycles, but you or your organization may use a different set of names. The important point to remember is that the activities, regardless of the phase name, are the same.

The project life cycle shown in Exhibit 3-1. is typical of an IT project. However, different types of projects can and often do have more or fewer phases than these. For example, a research project might have basic research and proof-of-concept phases before the initiation phase. Also, some organizations view certain activities, for example, project selection or customer services, as being life cycle phases, while others do not. Some authorities argue that customer service in particular and system maintenance

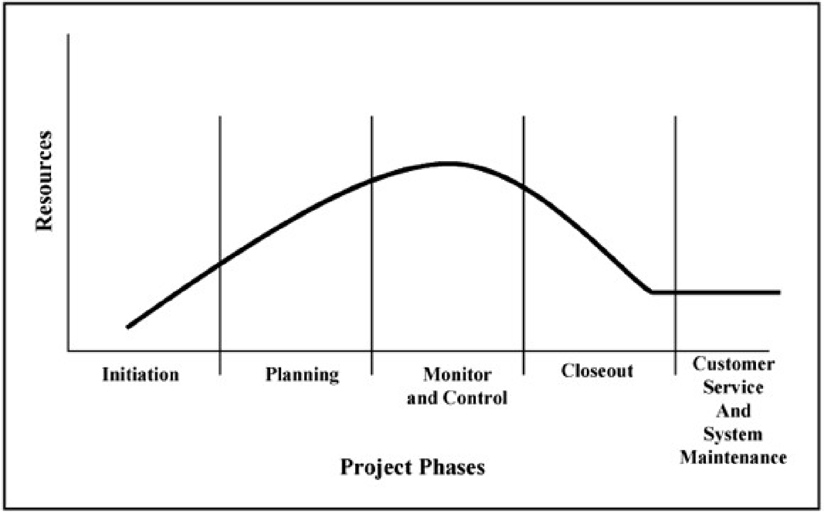


Exhibit 3-1. Typical IT project life.

to a significant degree do not meet the strict definition of a project because these are ongoing activities. However, most IT projects have customer service and maintenance commitments that are contractual, which make them projects. In any event, the long-term customer service and system maintenance activities should be treated as new projects, or as ongoing events. But here, this phase in the life cycle is a legitimate phase because it is the transition between the end of the product development cycle and its following service life.

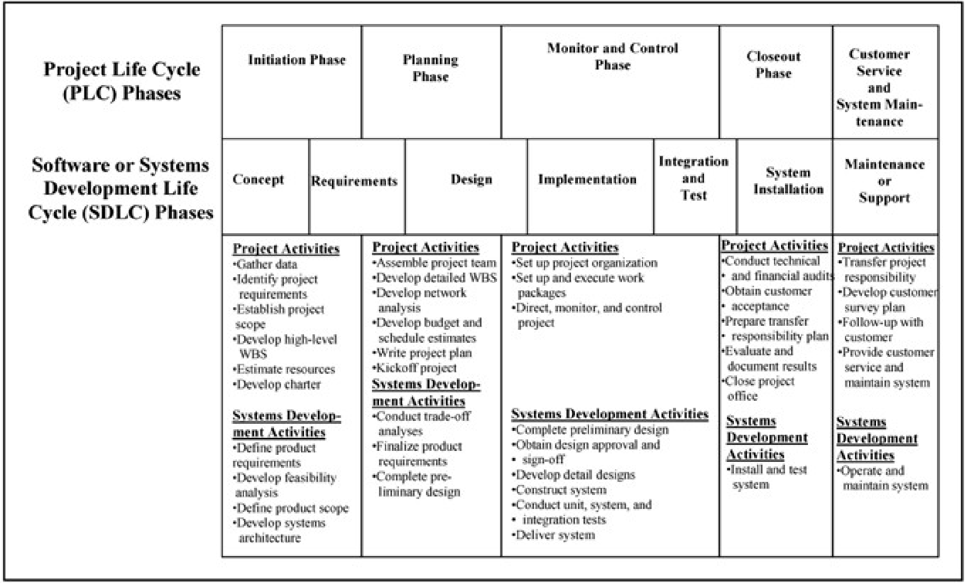
The SDLC is an underlying or coincident part of the project life cycle. Note that the SDLC is also used in a software development activity to mean software development life cycle, but the term *systems development life cycle* is more appropriate to the IT environment because IT is a systems project, not just a software development project.

Many organizations attempt to manage projects using a SDLC model alone, but neither the project life cycle nor the SDLC is solely sufficient to successfully manage a project; both are required, and it is a serious mistake to ignore either. A close study of Exhibit 3-2 (discussed in the next section) should make it clear why both life cycle models are important to project success.

The Systems Development Life Cycle

Exhibit 3-2 shows the relationship of the project and systems life cycles, along with the typical project manager and project team activities during each of the phases. From this graphic, it should be clear that the project life cycle activities have more to do with planning, administration, and leadership—all those activities necessary to ensure that plans and processes are in place to ensure a smoothly running project. The SDLC activities, on the other hand, are focused on the technical aspects of producing the project deliverables.

Although there is a definite overlap and merging of activities



Project Life Cycle (PLC) Phases	Initiation Phase	Planning Phase	Monitor and Control Phase	Closeout Phase	Customer Service and System Maintenance

Software or Systems Development Life Cycle (SDLC) Phases	Concept	Requirements	Design	Implementation	Integration and Test	System Installation	Maintenance or Support

Initiation Phase	Planning Phase	Monitor and Control Phase	Closeout Phase	Customer Service and System Maintenance
Project Activities •Gather data •Identify project requirements •Establish project scope •Develop high-level WBS •Estimate resources •Develop charter	**Project Activities** •Assemble project team •Develop detailed WBS •Develop network analysis •Develop budget and schedule estimates •Write project plan •Kickoff project	**Project Activities** •Set up project organization •Set up and execute work packages •Direct, monitor, and control project	**Project Activities** •Conduct technical • and financial audits •Obtain customer • acceptance •Prepare transfer • responsibility plan •Evaluate and document results •Close project office	**Project Activities** •Transfer project responsibility •Develop customer survey plan •Follow-up with customer •Provide customer service and maintain system
Systems Development Activities •Define product requirements •Develop feasibility analysis •Define product scope •Develop systems architecture	**Systems Development Activities** •Conduct trade-off analyses •Finalize product requirements •Complete preliminary design	**Systems Development Activities** •Complete preliminary design •Obtain design approval and • sign-off •Develop detail designs •Construct system •Conduct unit, system, and • integration tests •Deliver system	**Systems Development Activities** •Install and test system	**Systems Development Activities** •Operate and maintain system

Exhibit 3-2. The project and systems development life cycles and activities.

between the models, it is important that the project manager and his organization realize that the project life cycle encompasses all the activities of the project, whereas the SDLC focuses on the activities of the product. Each of the PLC and SDLC phase activities is explained in more detail below.

Project Life Cycle Activities in the Concept Phase

The first phase of the PLC is the data-gathering phase. The project manager often is not aware of the project until it has been selected and the company is committed to pursuing it. When this happens, there may be very little available information about the project, at least in written form, and the project manager has to scramble to gather as much information as possible in a very short time.

Every project manager has experienced or will experience the scenario described above—a project handed to her with virtually no background or requirements list. When this happens, the only recourse is to find out where or how the project originated and then to question everyone associated with its selection. The objective is to determine why the project was selected, who is for and against the project, the basis of any budgetary and schedule estimates, and what the expected product functions are. If the project was initiated because of a contract award, then much of the needed data can be found within the contract documents or in any proposal that was submitted. In either case, the project manager's job is the same—ferret out the data he needs to develop the project plans.

Project Requirements and Scope

Initially, the project manager must interpret the requirements and plan the project with minimum help because the project team cannot be formed until the level of effort is established. The first task of the project manager, then, is to quickly determine the general scope of the project and the skill sets needed to accomplish the requirements. This task is not as daunting as it seems because, according to her experience, the project manager will quickly develop a sense of the project size and complexity. The requirements will describe the project deliverables or products the customer expects. From this, the project manager can determine the types of skills needed. However, the wise project manager will first determine the general scope of the project and then solicit the help of her peers to develop the details. There are three reasons for this approach.

First, it is always better to have the combined experience and knowledge of a team than to try to know or remember or discover everything alone. Second, as the team members develop an understanding of the project, they will not only have suggestions about the needed skill sets but also specific recommendations about the people in the organization who are best suited to each task. Perhaps the most important reason for using your peers at this stage of project definition is their collective "lessons learned" experience. They will be able to help identify the risk areas in both the customer's stated needs and the organization's ability to respond to them.

The most critical activity in this phase is to identify the customer's requirements. In fact, I have devoted Chapter 4 to a detailed discussion of the subject. At this point, however, it is sufficient to understand that identification of the customer's requirements is vital because it defines not only the product, but also the actions and activities of the project team and the functional organizations supporting the effort. In short, identifying the requirements is the first step in defining the project scope.

Resource Needs

Once the general scope is determined, the project manager can began estimating the number and types of resources needed for the project. It usually is not realistic to have all the requirements identified. As you will see in the next chapter, requirements identification is not particularly difficult, but it does demand careful attention to all the project documentation available, and it requires a disciplined process to ensure that all the requirements are identified. Thus, a general understanding of the project scope is about the best that can be accomplished this early in the project. Of course, the more information that is available on the project, the more detailed the scope definition. For example, if the project results from a bidding action and a contract, then the requirements and the scope will be well understood from the start. It is those projects that are internally generated, no matter how good or important the reason, that typically lack written descriptions or requirements definitions and are, therefore, difficult to scope.

Whether the project is well defined or the project manager is working with a general scope statement, the next order of business is to develop a high-level WBS. Again, the more detailed the requirements, the better the WBS will be. However, remember that the WBS is needed before any final estimates of the types and numbers of resources can be made, or the cost and schedule estimates can be developed.

As with defining the requirements and scope, it is generally better for the project manager to use a team of experienced people to help with developing the WBS. This decreases the likelihood of overlooking a task and increases the speed of getting the WBS completed. In addition, experienced people will have a wealth of lessons learned, knowledge that can aid in determining resources, usually by name. For those project managers who have to negotiate for resources, having the names of the individuals they want when approaching a functional manager is important. The individuals you want may not be available for your project, but presenting the functional manager with the request puts him on notice about the skill levels you need. In other words, it is a good negotiating strategy that often eliminates the problem of having a functional manager provide you with people who just happen to be available and who may not even have the required skill sets. To repeat, using an initial team of experienced colleagues can provide valuable information and save the project from future problems.

This initial team may or may not become part of the project team. In fact, its members often don't become part of the final team because they are likely to be too senior to be assigned task responsibilities. In addition, some of them may be running projects of their own. When the WBS is sufficiently developed to have an accurate view of what the project entails, what resources are required, and, most important, which functional groups in the organization are needed to support the project, then the major output of the concept phase can be developed—the project charter.

The Project Charter

Project charters are extremely important to a project manager's success, and, thus, the success of the project. Yet, most organizations do not prepare project charters. This may be because of a lack of project management training. Or, it may be because these organizations have not embraced project management techniques and concepts as part of their corporate culture. In other words, it is a lack of education on the importance of this tool that likely prohibits its use.

A project charter is not a legal document; it is an internal document that is usually prepared by the project manager and signed by a person who is senior enough to have functional authority over the project and all the supporting functional areas. The primary purpose of the project charter is to name the project manager and to give him the authority to initiate the project. Many organizations will argue that they already have project charters because they send out an announcement each time a project begins. But the charter is more than just an initiation announcement; it is also a commitment to support the project. Each functional manager who is expected to supply resources to the project should sign the charter document. The charter is an excellent way of getting senior buy-in before the project begins.

The project charter usually is short—not more than three pages and often only one page long. The format varies from company to company, but generally it contains a brief scope statement describing the project, how the project supports the strategic goals of the company, who the project manager is, and finally, if possible, the project's priority within the company. The priority is the single most difficult thing to obtain because most companies view all their projects as having the same priority—number one. So even if you do manage to get a project charter written and signed, it is unlikely that you will have a priority assigned to it. However, if you can get a priority assigned, your life as a project manager will be a lot easier when you negotiate for resources. Exhibit 3-3 shows a sample project charter format that can be adapted to any organization in any industry.

I. Purpose (Scope statement)

II. Project Establishment (Business reason for the project and how it supports the company's strategic goals)

III. Project Manager Designation and Authority (Names the project manager and provides the range and limit of his or her authority)

IV. Project Team Organization (Describes where the project team is within the organizational hierarchy)

V. Project Manager's Reporting Chain

VI. Project Organization and Structure

VII. Project Team Composition

VIII. Support Organizations and Support Requirements (De-scribes which functional groups are required to support the project)

IX. Special Communication Requirements (Used if special reports or unique reporting data or cycles are necessary)

X. Appendixes (Some companies attach a more detailed scope statement to the charter than is usually given in the Purpose, above)

Exhibit 3-3. A sample project charter outline.

Systems Development Life Cycle Activities in the Concept Phase

The SDLC activities in this phase overlap significantly with those of the project life cycle, which may be why some organizations try to use the SDLC to manage their projects—the overlapping activities give a false sense of doing everything that needs to be done to manage the project. But there also are some different activities as well. In both models, this is the data-gathering phase—trying to get a handle on what the project is all about and, in the case of the SDLC, what the product is.

Product Requirements

It is important to realize that in project management we deal with two scopes: the project scope and the product scope. Whereas the project scope is general in that it addresses all those activities required to support developing the product—the reason for the project—the product scope is very specific and focuses only on the product (or service). The basis of the product scope, as with the project scope, is its requirements.

The challenge in requirements identification is understanding what the customer wants. Generally, requirements are poorly written or incomplete, or the customer simply does not know what he wants. Even when the requirements seem clear and straightforward, the wise project manager will restate them back to the customer to ensure they both understand the direction the product definition is taking.

Product requirements are best stated in terms of functionality desired. Functional requirements relieve the customer of the burden of developing detailed specifications, and it makes describing her product needs easier. Likewise, functional requirements free the provider to develop creative, and usually more cost-effective, solutions. Again, this subject is treated in detail in the next chapter.

Once the product requirements are identified and verified with the customer, the next step in the SDLC is the feasibility analysis.

The Feasibility Analysis

A major problem in any project, but particularly in the IT industry, is that too few organizations actually do a feasibility analysis of the requirements to determine whether they have the resources or technical capability to meet the customer's needs. All too often, the organization simply takes the position that "we can do any project" with no consideration about whether they have the right amount of expertise.

A recent Government Accounting Office (GAO) study revealed that every contract they reviewed failed, or was late and over budget, if the providing organization did not map the customer's requirements against the provider's capability *before* design approval. A feasibility study is crucial to the success of a project, and it should be a part of the requirements identification and high-level WBS development.

Feasibility studies demonstrate to a prospective project owner or investor that a given concept is financially viable and whether further study and/or a business plan is warranted.

For a feasibility study, basic data is obtained from the client through a series of queries, questions, and meetings, wherein the client provides some of the research. Other data and facts need to be gained from a variety of sources.

The typical feasibility study contains, among other items, notes on financial projections, a general description of the business, general details describing how the company/project will be formed, managed, and marketed, statements concerning the competition, and a cash-flow projection based on averages. Further notes can be included as to general details of the project and revelations discovered during the research stage. The study will normally be completed quickly, and it will be presented in a very general format (unlike a business plan). A feasibility study should answer five questions:

1. Will it work?

2. Do we have the expertise and resources to do it?

3. Will it benefit the company?

4. What will it cost to start?

5. Does it fit into the company's strategic plan?

The feasibility analysis serves several purposes. Not only does it help to determine whether the company has the technical and resource capabilities to do the project, but perhaps more important, it answers the question of whether the project would contribute to the company's long-term growth plans. If the project doesn't fit the strategic plan, then whether or not there is sufficient expertise and resources is a moot point—the project should not be started. One further major benefit of a feasibility analysis is that it helps identify and reduce business risks. Technical risks, to some degree, can also be identified by the feasibility analysis, but a more thorough risk assessment is done when the systems architecture is developed.

If the feasibility analysis indicates that the project does fit the company's strategic goals but reveals shortcomings in the provider's capability or resource pool, then the opportunity exists to either hire additional capability, contract with a consultant, or team with another company. Another legitimate strategy is to negotiate with the customer to reduce the product scope, or at least postpone some of the options until the company can develop the requisite capability.

With the feasibility analysis completed, the product scope can be definitized. Product scope is simply the amalgamation of the requirements identification and feasibility study into a statement of product definition.

Product Scope

Product scope can be defined as the features and functions that characterize a product or service. With the requirements definition completed and a well-researched feasibility analysis in hand, the scope of the work to design, develop, and implement the product is well under way.

The product scope is different from the project scope in that the emphasis here is on defining the functions and characteristics of the product and the technical considerations for building it. This is not the process of actually designing the product but defining the parameters within which the product is likely to be built. In other words, the product scope defines the boundaries around the product (i.e., how big, what color, how responsive, and how reliable it is), but it does not dictate a solution. In fact, a good product scope definition won't even suggest a technical approach: It will just specify the product's minimum functionality and characteristics. The technical approach is determined by developing the systems architecture, which determines the product specifications.

Systems Architecture

The National Aeronautics and Space Administration (NASA) defines a systems architecture as:

> ... How functions are grouped together and interact with each other. The architecture applies to the mission and to inter- and intra-system, segment, element, and subsystem.[1]

In other words, the architecture includes every aspect of the system. And "system" is the keyword when defining the IT project. All too often, the emphasis is on the software development component, with the rest of the system being designed almost as an afterthought.

The key elements of architectural design are:

- ✓ Requirements
- ✓ Functional design of alternatives
- ✓ Analysis of alternatives
- ✓ Evaluation criteria
- ✓ Formulation of a preferred system architecture

The architectural design process then is one of identifying the requirements and developing several technical alternative solutions to meeting the customer's needs. The obvious reason for this approach is to determine the most efficient and cost-effective solution. Once the analysis of these alternatives yields the preferred solution, then it is important to establish evaluation criteria to measure whether the alternative is indeed the proper one and how well it meets the requirements.

The architectural design process ultimately results in the formulation of a preferred architecture, which means a detailed analysis and description or specification of the system. With the preferred architecture identified, the serious project and product planning can begin.

Project Life Cycle Activities in the Planning Phase

Many of the activities in this phase will have been started in the concept phase and, perhaps, even completed, depending on how well the customer's requirements are stated. But generally, this is the phase during which project plans and project team composition is refined and finalized. The first task during this phase is to complete and assemble the project team.

The Project Team

Resource requirements are determined from the requirement definition exercise and the high-level WBS development. The earlier the team is assembled, the better, but how fast this activity can be accomplished is purely a function of how well the project requirements are stated and understood from the beginning. It usually takes longer to assemble the team for projects that are generated from within the organization because of the lack of detailed and well-stated requirements. On the other hand, projects that result from competitive bids are better defined. Consequently, it is easier to establish minimum skill and experience levels for them. In fact, some customers, particularly from the public sector, often attach a key personnel clause to the contract, which specifies the minimum skill and experience qualifications the key members of the team must possess. The key personnel requirements, incidentally, become a part of the evaluation criteria when the customer is deciding who the bid winner will be.

Once the team is assembled, their first task should be to finish development of the WBS, because it is the tool upon which all the rest of the project planning depends.

The Detailed Work Breakdown Structure

The WBS may only require some refinements. As with every other aspect of the project at this stage of development, however, the completeness of the WBS is a function of how well stated the requirements are. But with most projects, the best that can be expected at this point is a high-level WBS. The team should finish its development. Even if the project manager and her peers have managed a rather detailed treatment of the WBS, it should be reviewed once more by the team, project manager, and the customer to ensure that all tasks have been captured. Remember, if it is not in the WBS, it is not in the project.

The WBS is the single most important tool in the project manager's arsenal because it is the basis of everything the project team will do for the remainder of the project. Without it, or without a complete one, good budget and schedule estimates cannot be developed, executable and achievable plans cannot be written, and there will be no accurate baseline against which to measure the project's progress.

It is important that the entire project team be involved in this final push to complete the WBS to ensure completeness and accuracy. This effort also serves as the first team buy-in and team-building opportunity. Finally, the customer should always approve the WBS because it is the instrument that captures all the requirements as well as the supporting project activities. If the customer decides that he does not want to pursue any part of the project, as a result of seeing the effort described by task, then now is the time to make adjustments in the project direction.

The next project team activity is the network analysis, which must be accomplished before budget and schedule estimates can be developed.

The Network Analysis

The network analysis is a tool that is basic to determining the schedule and, consequently, the cost of a project. Incredibly, many project teams and organizations do not do a network analysis: They just make an estimate at the schedule and build their project plans around that estimate. Without a network analysis, it is not possible to optimize the schedule, nor is it possible to identify points of resource conflict that may cause delays in the schedule. If everyone clearly understood its uses, no one would attempt a project without a complete network analysis.

The network analysis has several functions. First of all, it graphically shows all the task dependencies in a project or even in smaller components, such as a phase. The network should be constructed using task-level activities. If it is constructed at a higher level, say at the summary level, it will not reveal the source of problems, should they exist. Second, the network analysis is often the first opportunity to identify risk areas, such as resource conflicts or task dependencies, that might contribute to schedule delays if one or more of the tasks are late. Third, the network analysis identifies the critical path, which is the shortest duration that the project can be accomplished. The critical path, furthermore, defines the project schedule. From the network analysis, the project schedules can be developed.

Project Schedules

Project schedules are estimated from the network analysis, and, in fact, are basically a bar chart depiction of the network itself. It is crucial to develop as accurate a schedule as possible because the other key elements for project success—budget estimation and resource allocation—are dependent on it.

The project can have several schedules. If the project is a large and complex one, the project manager will need a master schedule that provides an overview of the project milestones. The master schedule is particularly important for communicating with the stakeholders and for providing general progress updates. The project manager will also need a schedule that shows each task so that actual progress can be tracked. In addition, the Gantt chart can be used to show milestones, meetings, deliverable dates, and any other information that assists the management process.

The reason that an accurate schedule is important is not so much because of the necessity of accurate progress tracking, although that is certainly important, but rather because there is no way to develop a reasonably accurate budget without it. Hence, the sequence is clear: network analysis to determine the critical path and other task dependencies, schedules by which the progress of each task, deliverable, milestone, and any other requirement can be monitored, and the budget for every aspect of the project. Once these key elements of the project are determined, the project plan can be written.

The Project Plan

The project plan is in some ways a misnomer because it implies one simple, straightforward document. The fact is, the project plan is neither simple nor straightforward. To be sure, there are templates from which a plan can be developed, and there are even software programs that aid in the writing of such plans. But although each plan should have pretty much the same parts, every project, being unique and different, will have to be described in a way that reflects its unique nature. So there really is no way to reduce the project to a routine activity. If that were possible, then project management would be simply an administrative function. Furthermore, the project plan is an amalgamation of several plans. That is, the project plan consists of the general description of how the project will be accomplished, and it also contains all the other plans necessary to support the project.

A key project management activity is the kickoff meeting. Ideally, the project kickoff occurs after the project team has completed defining the project to a significant degree, but that almost never happens.

The Kickoff Meeting

When is the best time to have a kickoff meeting? The right time is when there is enough information available to make the meeting meaningful, informative, and when most, if not all, the stakeholders can attend. This "right time" needs to be early in the project life cycle, ideally at the end of the concept phase or the beginning of the planning phase. However, the tendency is to schedule a kickoff meeting immediately after project approval. Often the pressure to schedule the meeting early comes from senior management because a meeting gives an appearance of activity that the planning function just cannot provide. Having the kickoff meeting too soon, though, can be counterproductive.

The problem with having a kickoff meeting too soon is that it defeats the purpose of the meeting. A kickoff meeting is an opportunity to provide a statement of the project's scope and general schedules, an indication of the most likely technical approach, an approximation of the resources needed, and an introduction of the stakeholders to the team members. But often the team is well into the planning phase before enough information about the project and project deliverables is available or clear enough to warrant discussing them at a kickoff meeting. However, waiting too late to have the meeting is just as dangerous as having the meeting too early.

Some organizations insist that the kickoff meeting should not be conducted until all the detailed planning is completed. Their rationale is that until the details are known, the project cannot be discussed with any authority. The problem with this kind of thinking is that by the time all the detailed planning is completed, the meeting is no longer a kickoff to the project—it becomes a status meeting. In addition, one of the primary purposes of a kickoff meeting is to get buy-in from the stakeholders.

The project manager and the project team are busy during the planning phase putting the finishing touches on the plans that will ensure project success. Key to developing these plans is the final product architecture.

Systems Development Life Cycle Activities in the Planning Phase

The activities of the SDLC during the planning phase are focused on defining the product alternative selection and, at least, the preliminary design features. Ideally, the activities of the concept phase will have yielded a general understanding and agreement about what the systems architecture will look like. But the final decision about the technical approach is done during the planning phase. One of the key considerations is the trade-off analysis, which reveals the most economic and feasible technical approach.

The Trade-Off Analysis

A critical element of project definition, schedule, and cost estimating is in the consideration of different technical approaches. Although a trade-off analysis can be performed anytime during the project's life cycle, it is most often done during the early stages of requirement definition.

The trade-off analysis is done in conjunction with the technical or functional experts, and it is usually the responsibility of the project manager to decide which approach is the best. The tradeoff analysis is used to determine which approach is the most efficient and effective in meeting the customer's requirements. The selection parameters can vary widely. The parameters selected, though, should relate directly to the problem statement. For instance, the problem might be to design an IT system that performs with a prespecified degree of effectiveness and at a minimum life cycle cost. Therefore, the parameters to consider will pertain to system effectiveness and cost. On the other hand, there may be a need to evaluate different off-the-shelf components. In this case, the primary considerations might be supportability, interchangeability, or mean time between failures. The problem will dictate the parameters, and the project manager is responsible for ensuring that the different alternatives are assessed against the proper parameters.

Generally, there are two classes of evaluation parameters: cost and system effectiveness. Obviously, the schedule will be dictated by cost and system effectiveness and vice versa. Within each of these two classes, there are a number of different parameters. Each parameter will be more or less important depending on the customer and the customer's needs. For example, if the project is to deliver a piece of equipment for the Department of Defense, operational necessity and speed of implementation may be the driving criteria, and cost may be inconsequential. Usually, the customer will indicate the criteria that are most important. In that event, these criteria should be a part of the evaluation of technical alternatives. Otherwise, the project manager must determine the criteria she thinks are important, based on what is known of the customer and the customer's stated requirements, and apply a weighting factor to each of the criteria as they are applied in the alternative selection evaluation. Exhibit 3-4 provides a list of parameters that are commonly considered, and it shows an order of evaluation parameters, as the system is decomposed into subsystems and the details of subsystems. Note that Exhibit 3-4 lists no first-order parameters. Usually, we consider system costs to be the first-order parameter because that is the single most important parameter in any project. Even if the overall cost is not the princi-

TECHNICAL ALTERNATIVE EVALUATION PARAMETERS

Second-Order Parameter	• System effectiveness	• Life-cycle cost
Third-Order Parameter	• System Performance	• Research and development costs
	• Operational availability	• Investment costs
	• Dependability	• Operation and support cost
	• Capacity for growth	• Phase-out costs
Fourth-order Parameter	• Accuracy	• Design costs
	• Range	• Data costs
	• Size and Weight	• Test and evaluation costs
	• Reliability and maintain-ability	• Manufacturing costs
	• Speed	• Inventory costs
	• Supportability	• Maintenance costs
	• Transportability	
Fifth-Order Parameter	• Accessibility	
	• Diagnostic aids	
	• Handling	
	• Interchangeability	
	• Logistics requirements	
	• Producibility	
	• Operator skills	
	• Safety	
	• Storage	
	• Utilities	
	• Test requirements	

Exhibit 3-4. System alternative evaluation parameters.

pal evaluation criteria in determining contract award, cost is the ultimate discriminator because the customer will have a budget cap. Also, in evaluating technical approaches, the efficiency of a system is measured, in part, against its cost-effectiveness.

The problem is to select the best approach possible through an iterative process of system analysis. This process is demonstrated in Exhibit 3-5. The process can be very time-consuming and, in some cases, very subjective. One of the major problems for many project managers or project team members is that they often forget the ripple effect of changes in a system. That is, each time a change is made to a system or each time a new alternative is considered, a sensitivity analysis is required to determine the overall effect to the system.

A process for managing project time, cost, and performance trade-offs should emphasize the systems approach to management. To manage this trade-off process, the following steps should be taken:

- ✓ Identify the basis for project conflict or possible redirection.
- ✓ Review the project objectives and requirements.
- ✓ Analyze the project status.
- ✓ Identify any alternative courses of action.
- ✓ Analyze and select the best alternative.
- ✓ Document the actions taken.
- ✓ Revise the project plan.

The objective of performing a trade-off analysis is to find an alternative course of action. A good approach to trade-off analysis is to identify several courses of action by brainstorming with the project team and other functional experts. With several approaches identified, the best approach, in terms of an efficient technical solution and costs, can be determined.

It is important to remember that each alternative must be weighed against the project requirements and objectives. A good

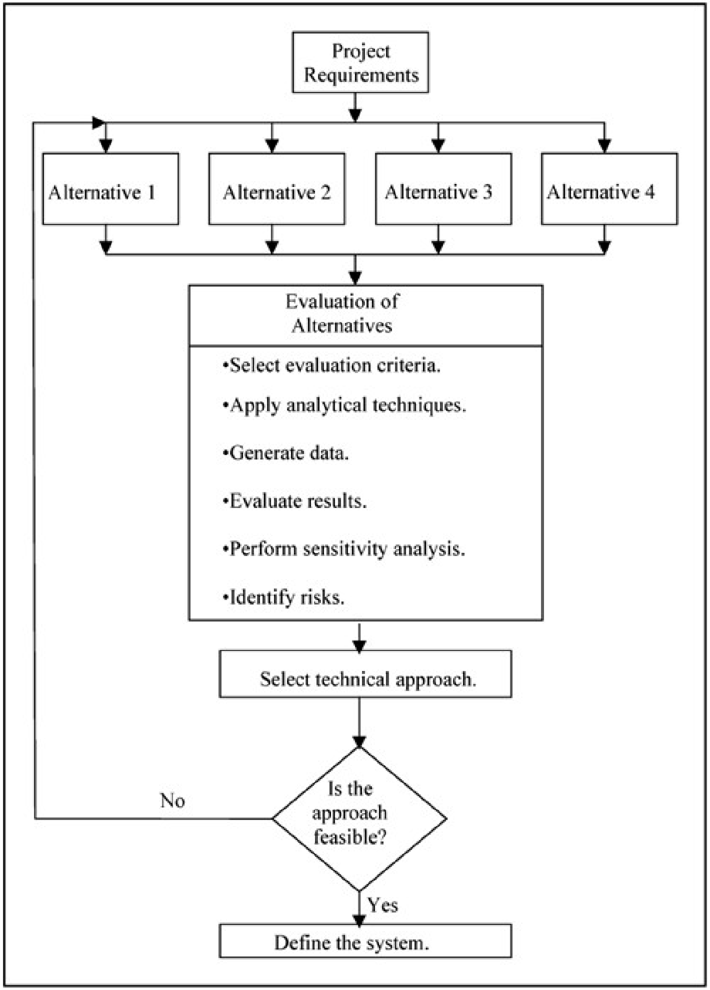

Project Requirements
Alternative 1
Alternative 2
Alternative 3
Alternative 4
Evaluation of Alternatives
•Select evaluation criteria.
•Apply analytical techniques.
•Generate data.
•Evaluate results.
•Perform sensitivity analysis.
•Identify risks.
Select technical approach.
Is the approach feasible?
No
Yes
Define the system.

Exhibit 3-5. Trade-off analysis process.

technical approach is not always the most cost-effective one or the best in terms of schedule. Every change or potential change has a ripple effect throughout the system, and sensitivity analyses must be performed.

Exhibit 3-5 shows that several alternatives are analyzed before the best one is selected. In fact, all the alternatives must be analyzed and tested for their sensitivity to the system before the best one can be chosen. That is why the trade-off analysis process is such a time-consuming one. But because this process is so time-consuming, it is often skipped, often with disastrous results.

Once the trade-off analysis is completed and all the stake-holders agree upon the best technical approach, then the product requirements can be finalized.

Finalizing Product Requirements

It should be obvious by now that the process of defining requirements, both for that project and the product, is iterative in nature. That is, analyzing what you think the customer wants and verifying it requires open and active communication channels. With the proper attention on developing a comprehensive WBS and rechecking that the systems architecture definition still accurately reflects the customer's needs, the product requirements can be fleshed out and the product requirements list and/or specifications can be finalized. It is at this point that it will be clear, both to your organization and to the customer, that the project is achievable or not and that the product can be designed with the requested functionality or with any approved changes in the scope.

The next step in this phase of the SDLC is to complete the preliminary design.

The Preliminary Design

In our model of Exhibit 3-2, this step is listed as "complete preliminary design." You will notice that there is no "start preliminary design" step because the preliminary design has been evolving from the requirements analysis, feasibility studies, systems architecture development, and trade-off analysis. In fact, the preliminary design should be pretty much completed by this point except for any refinements necessitated by finalizing the WBS and product requirements. In other words, the final touches can now be put to the design and to any plans that need updating. The next step in the process is to obtain stakeholder and especially customer approval so that the detailed design can be developed.

Design Approval and Sign-Off

Design approval and sign-off by the customer and other key stakeholders is critical to a successful project. One of the major reasons for project failure in the IT industry is that this key step is often bypassed in deference to getting on with the coding. The thought seems to be that the design will naturally evolve as the code is written—it won't. It could, if the only component of the project were to design software. But the preliminary design must include the system—that is, software, hardware, communications, training, and whatever other product component—and it must include the integration of all of these elements. So this design thing is not trivial; it is the heart of the project.

Develop Detailed Designs

The phrase "Develop detailed designs" is very descriptive of the activity—it means to develop to the greatest detail possible the design for each component of the product, its construction, integration, test, and final delivery. This element of the system development cycle often includes several critical milestones set by the customer. Even if the customer doesn't require specific design review checkpoints, it is a good idea for the project manager to have the customer review them. The reason for these reviews is to ensure adherence to the design parameters, but just as important, it is the best way to assess the viability of the technical approach. What seemed like a good idea in theory may prove unworkable once the pieces are put together.

Once the design is complete, and again, approved, the construction of the product can begin in earnest.

Constructing the System

It would seem that an inordinate amount of time has passed between the point of project go-ahead and the start of constructing the product. Relatively speaking, there has been a considerable amount of time spent in the planning and designing stages. And this is the very thing that makes many senior managers, as well as project managers, nervous—spending time planning and designing when they could be "coding like hell." The problem is that the planning/designing functions do not have the "feel" of making progress, whereas the action involved in coding or building does. Hence, the tendency is to skip right to the actual building cycle. As hard as it may be to do, the project manager *must* resist the temptation to shortchange the planning and design cycles, and she *must*, to the best extent possible, resist senior management's urgings to get on with it. It is not unusual for world-class organizations to spend half their project's budget before the project is actually implemented, that is, engaging in the build, test, and install phases. As Dwight D. Eisenhower said, "The plan is worthless; the planning is priceless."

So it is with IT projects. If the right amount of planning and design work has been accomplished, the construction of each of the project components can now progress— usually with little or no snags. Bear in mind that, in an IT project, there will be at least two components and probably three or four that have to be built, and some or all of these components will need building parallel to meeting the schedule. This concept will be explored more fully in Chapter 9.

Unit, System, and Integration Tests

Testing in any project, and especially in IT projects, is often one place where significant delays can occur. The delays are not usually the fault of the testers, although they can be, but rather because testing organizations often have a limited number of available, highly skilled testers on staff. If the organization typically has several projects running at the same time, then scheduling the test window is critical. The wise project manager will factor this potential schedule problem into her project plan and ensure that "begin testing" is one milestone that is met.

Testing project products is a project within itself. The essential steps in the testing phase are:

- Write a test plan incorporating functional requirements and features of the product(s).
- Develop a test suite based on the test plan so that all the product functions and features are tested individually, as subsystems where appropriate, and as an integrated and complete system.
- Execute the test suite and report any discrepancies from the plan. Resolve discrepancies as bugs, test case errors, or deficiencies, as appropriate.
- After discrepancies are corrected, prepare the final product for release and installation. Archive all product documentation and test components.

It is important to note that there usually are three different tests run on the product—the tests on individual components, tests on subsystems, and tests on the integrated final product. These tests don't include ongoing software code testing. So the testing function is not only critical to the overall quality of the finished product, but is one that can be in and of itself an intensive and sometimes grueling endeavor. To skimp on the testing, though, is asking for trouble when the system is delivered to the customer.

Page 66

Delivering the System

The delivered system is the culmination of the project's technical effort and represents the reality of the customer's stated needs and requirements. The system with all its components is the final act of both the SDLC and the PLC. It is why the project was initiated.

The delivered system may be physically delivered to a customer site or, if the customer is internal to the organization, it may simply be delivered by powering it up. Regardless of whether the customer is internal or external, the process should be treated the same—as if delivery is the conclusion of a written contract. Otherwise, there will be problems with the final phase of the project life and systems development life cycles.

Project Life Cycle Activities in the Termination Phase

The activities in this phase are basically administrative for the project manager. She has to ensure the project requirements have all been met and that the customer agrees that the product is acceptable. This phase is often the most difficult phase of a project. The pressure is on to finish the work and meet the budget, schedule, and performance goals set at the beginning of the project. But because of the nature of business, the functional managers supporting your project need those resources for beginning other projects. The project manager often finds himself trying to keep enough of the team together to complete the work that is left. Without completing these termination activities, the project could continue unfinished for months or even years.

Technical and Financial Audits

There are two audits that are absolutely required before closing the project. They are the technical audit and the financial audit.

The technical audit is performed to determine whether the requirements of the project have been completely met. The project manager and team accomplish this audit by reviewing the scope and specification documents and comparing the requirements against the WBS. If all the requirements are met and the final project deliverables are completed, then the technical audit can be successfully closed.

The financial audit is precisely what one would expect it to be—an evaluation of the finances to ensure that all vendors are paid and that all invoices to the customer are prepared and submitted. The financial audit also compares the actual costs against the planned cost of the project to determine how closely the initial estimates were. This part of the audit becomes extremely important in one of the other termination activities—evaluating and documenting the lessons learned.

Once the audits are completed, the very important step of obtaining customer acceptance can be accomplished.

Obtaining Customer Acceptance

Project managers, and organizations in general, often make the serious mistake of not ensuring that the original contract or internal memorandum of understanding, if not covered by contract, contains a completion criterion for the project. Completion criteria are established so that the customer can state clearly what the provider must do to satisfy the terms of the agreed project parameters. Completion criteria protect both the customer and the provider because they each know precisely what is expected of the other. Without these criteria, the customer can always claim that the provider has not met the project requirements, and the project will never be closed. Generally speaking, this is not a scenario that is often played out because the customer wants the project finished as much as, if not more than, the provider.

Getting customer sign-off on the project officially completes the project, but there are still activities that have to be finished.

Preparing a Transfer Responsibility Plan

In most cases, IT projects are transferred to another organizational group or team for care and maintenance. The actual maintenance of an IT system is not a project but rather becomes a functional activity. Nevertheless, all the documentation and work of the original project can be viewed as the statement of work (SOW) or scope for the maintenance phase. So the closeout of the project should be accomplished with this view in mind. The successful maintenance and continued customer satisfaction of the product depends in large part on the quality of the information handed over to the maintenance team. Hence, a thorough and well-documented, supported transfer plan is a major activity during the project termination; this is yet another reason for holding the team together until *all* phase activities are completed.

Evaluating and Documenting Results

One of the most neglected activities of the termination phase is the "lessons learned" effort. The reason so few organizations support this activity is simply that there is a rush to start the next project, and the resources are desperately needed elsewhere. In short, the feeling is that the organization can't afford to lose its resources to this kind of activity. The irony is that a lessons learned meeting generally only takes two to three hours because most of the data are already available in the status reports. Basically all that remains is to collect individual thoughts, opinions, and comments about what went well and what didn't during the project. It is also important for the project manager to record his evaluations of the project team members. The objective in a lessons learned exercise is not to place blame but to evaluate actions and skills, people and technical skills, so that future project teams have the benefit of knowing how best to utilize the talents of the resource pool.

After the lessons learned activity is completed, the only thing left is for the project manager and team to close the project office.

Closing the Project Office

Closing a project office is difficult or relatively easy, depending on whether the office is a loose structure within the organization or an actual office on the customer's site. If the latter, there can actually be a considerable amount of work involved. There may be a rental agreement that has to be paid off or terminated. Or, a transfer of responsibility has to be made. There also may be utilities and telephones to disconnect, and office furniture that must be disposed of. In many instances, the customer will have furnished equipment, usually computers, for the development effort. If so, these customer-owned assets have to be properly accounted for, returned, or disposed of in accordance with the client's direction.

In addition to disposing of the physical assets of the project office, the project manager is responsible for archiving the project documentation. This documentation usually is of two types: the project control book, which contains all the plans, status reports, and all other technical information, and the legal documentation. The legal documentation will contain such things as the contract, if from an outside customer, or a memorandum of understanding, if from an internal customer. It may also include memoranda between the project manager and customer and other stakeholders, in addition to any other documentation that has to do with the contractual aspects of the project.

One final but very important project manager activity in this phase is the reassignment of people resources. The project manager may not have functional responsibility for the team members. In fact, she probably won't have that responsibility unless the project operates in a projectized organizational structure. But at the very least, the project manager should make recommendations about how best to utilize the team members. This action not only engenders trust and loyalty on future projects. It also helps the organization maximize the use of individual skill sets.

Systems Development Life Cycle Activities in the Termination Phase

The SDLC activities of this phase are not necessarily simple, but they are straightforward—install the system and make sure it runs as advertised.

Usually, installation of an IT system is fairly simple, if the components have been built according to the plan and if they have been thoroughly tested. In fact, the integration tests will have required that the components all are connected and working together, so the system has already been installed once. Still, the proof of the pudding is in demonstrating that the system works as the customer expects. So, although this is the only activity in the SDLC for this phase, it is actually the crucial one, because if it doesn't work as the customer expects it to work, then it may be back to the drawing board.

Once the system is installed, the provider tests it to ensure it functions as expected. In addition, the customer is likely to have had a contract clause specifying that he be allowed to operate the system to determine its functionality and adherence to the stated requirements. In fact, this often is the test of acceptability by the customer. Only upon the successful conclusion of this test is the system accepted. But once it is accepted, the whole project is completed and closed. Now a new phase of customer relationship begins—operation and maintenance of the system and continued customer satisfaction.

Project Life Cycle Activities in the Customer Service and System Maintenance Phase

This area is not, traditionally, a phase of the project life cycle, nor is it an SDLC phase. However, it is my belief that it should be a phase of both cycles. In reality, the follow-on maintenance and customer service functions occur here. However, the tendency is to treat these activities as separate from the original project. In one sense they *are* separate—the project of developing and delivering the product *is* complete. But in a very real sense the project is never over while the customer is using, and expecting good service from, the product. Thus, it is logical that we consider follow-on activities as part of our project responsibilities.

The customer service and maintenance functions do not fit the definition of a project. That is, they don't have a defined end date, unless there is one dictated by contract. But the work of the project—training materials, manuals, operating procedures, specifications, and so on—serves as the SOW (statement of work) for those who are responsible for maintaining the system and ensuring the customer is satisfied with how the product operates. This SOW should be transferred to the maintenance/customer service team by formally passing the responsibility from the project team to a service team.

Transferring Project Responsibility

A memorandum of understanding should be used to transfer the project responsibility, regardless of whether it is for an internal or external customer. Once the project is installed and tested to be operationally ready, the project team is essentially through with the project work, except for the remaining administrative details. But unless the system continues to function in a manner pleasing to the customer, the effort has been wasted. Therefore, it is crucial that a smooth transfer from the project team to the maintenance and customer service teams (usually these functions are separate) be made. It is best to have in place a formal transfer procedure that includes training on and about the system for those who will continue the customer interface. In addition, it is my recommendation that the project manager of the original project contact the customer two to three months after the product is installed to inquire about how well the system is functioning. This inquiry serves two purposes; first, it gives the customer time to operate the system and to determine whether it truly meets all the stated requirements. Second, it demonstrates to the customer that the provider is sincerely interested in providing quality products.

Developing Customer Survey Plans, Following Up with the Customer, and Providing Customer Service and System Maintenance

Every organization should prepare good customer survey plans. This subject is discussed more fully in Chapter 12. Surveying customers about products you have provided is the fastest way to determine whether your project management processes are adequate. Customer surveys, and occasional visits to the customer, are the best ways to follow up with the customer.

The sole objective of this phase of the project life cycle is to maintain the product and to ensure that the customer is satisfied with the work your organization has done in providing the IT product of her dreams.

Systems Development Life Cycle Activities in the Customer Service and System Maintenance Phase

The SDLC activities in this phase are focused on one thing only—maintaining the system.

Too many organizations do not differentiate this activity from the project. This is particularly true in IT efforts for internal customers. That is, the thinking is that the project continues as long as the system is in place. The problem with this approach is that there is never any closure from one activity to another. It is important to the project manager and the team members to have a sense of accomplishment. I have already stated that the maintenance and customer service activities are not truly projects in the traditional sense, and they are not. But these activities should be appended to the project and systems development life cycle so that the relationship between the project and the follow-on responsibilities is clear. And there should be a definite and distinct demarcation between the project and follow-on efforts.

Summary

There are two distinct life cycles at work in a project—the project life cycle, which encompasses all the project work including the systems development, and the systems development life cycle, which focuses on the product or products of the project. Many organizations make the mistake of trying to manage their project using only the SDLC. It is a mistake to assume that either the PLC or the SDLC is sufficient by itself—both are needed to successfully manage any project, but particularly one as complex as any IT project.

The PLC and the SDLC models have overlapping activities. It is not possible to completely separate the two any more than it is possible to delineate between the phases of either model. But thinking of the models as having distinct phase activities is helpful in understanding what the project manager and his team have to do in order to successfully complete the project on time, within budget, and to the customer's performance specifications.

Note

1. National Aeronautics and Space Administration, NASA Engineering Management Council,*The NASA Mission Design Process* (Washington, D.C., 1992).