4

PLANS AND PLANNING

A plan in the mind of a man is not a plan.
—Richard H. Thayer

4.1 INTRODUCTION TO THE PLANNING PROCESS

By definition, every project of every kind is an endeavor of limited duration that uses resources to achieve stated objectives. A project plan specifies, among other things, the duration of the project, the resources needed, and how the resources will be applied to achieve the stated objectives. Software requirements (discussed in Chapter 3) provide the objectives for the product to be developed or modified. The planning process is concerned with developing the various elements of a project plan and documenting the plan in a specified format.

Your software project management plan must be a written document; otherwise, various stakeholders in the project will have differing interpretations of how the project will be conducted, and there will be no documentation of plans for effort, cost, schedule, resources, and supporting activities. The project plan also provides a vehicle for trade studies and for negotiating trade-offs among cost, schedule, and requirements, both initially and as changes occur. Baseline control of the written project plan supports systematic updating of the plan and communication of changes.

In the best case, your planning process will begin with tailoring of your organization's standard processes to fit the management, software development, and supporting processes of your project. In that case the information in this chapter can be used as a checklist against which you can compare your organization's planning processes and document templates.

Managing and Leading Software Projects, by Richard E. Fairley

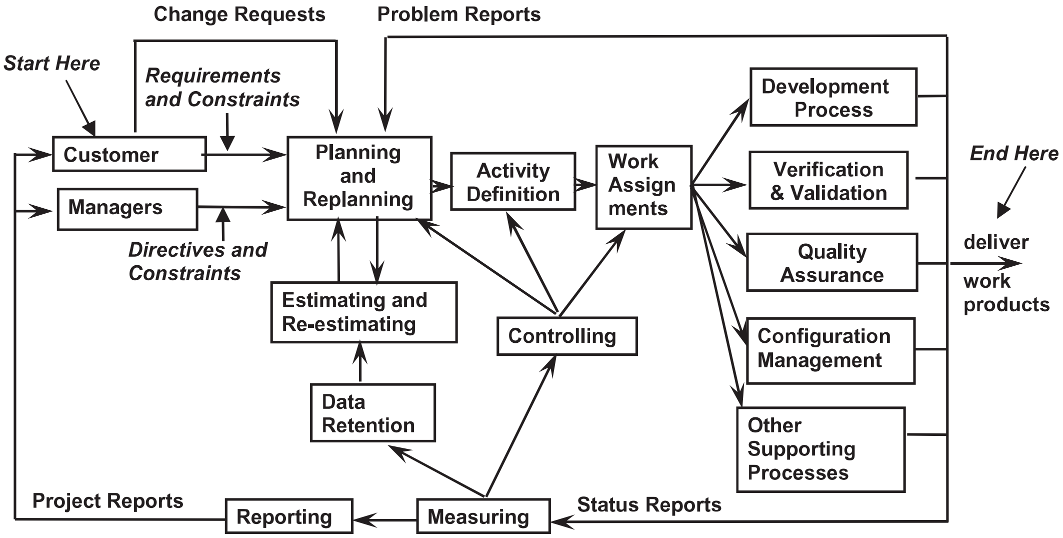

FIGURE 4.1 A workflow model for software projects

In the worst case, you will have to develop your project plan without any organizational structures or guidelines. In the absence of organizational structures and guidelines, the workflow model for managing software projects presented in Figure 1.1 (repeated here as Figure 4.1), the system engineering model presented in Figure 2.1*b*, and the development models and supporting processes described in Chapter 2, plus the information in this chapter can provide a tailorable framework for planning and executing software projects.

Project planning, like all elements of software development, is best accomplished in an iterative manner; details are added as understanding grows.

4.2 OBJECTIVES OF THIS CHAPTER

After reading this chapter and completing the exercises, you should understand:

- the planning process for software projects
- the project planning process area of CMMI-DEV-v1.2
- an approach to planning Agile projects
- a template for software project management plans (SPMPs)
- tailoring the SPMP template
- techniques for preparing a SPMP

The planning process presented in this chapter is informed by the Project Planning process area of the CMMI-DEV-v1.2 process framework, the planning elements of ISO and IEEE Standards 12207, IEEE Standard 1058, and the PMI Body of Knowledge. These elements are described in Appendix 4A to this chapter. An annotated version of IEEE Standard 1058 is presented in Appendix 4B.

An electronic copy of the annotated version of IEEE Standard 1058, presentation slides for this chapter, and other supporting material are available at the URL listed in the Preface to this text. Terms used in this chapter and throughout this text are defined in the Glossary at the end of the text. Mechanisms and techniques of project planning and estimation are presented in Chapters 5 and 6 of this text.

4.3 THE PLANNING PROCESS

As depicted in the workflow model of Figure 1.1 in Chapter 1 of this text, repeated here as Figure 4.1, inputs to planning include the customer's requirements and constraints as well management directives and constraints. The system requirements, system design, and software requirements may also be available or they are developed during project initiation. As discussed in Chapter 3, customer's requirements include operational features, quality attributes, and design constraints for the envisioned product. Constraints imposed by the customer may include both product and process constraints.

A product constraint might require that the system be developed using a specified version of an operating system or that the new or modified system provide an SQL interface to an existing database. A process constraint might require that the system be delivered in a staged sequence of increasing capabilities or that the source code for the deliverable software plus the requirements and design documentation be delivered to an independent agent for final verification and validation.

Management directives may include a policy statement that all projects must produce design documentation and verify it for completeness, correctness, and consistency using peer reviews. A management constraint might limit your project resources to a staffing level of 10 software developers.

Some of your first tasks as project manager are to establish a pattern of ongoing communication with the designated customer representative (your primary point of contact), and to clarify with him/her/them the operational requirements, development constraints, and success criteria for the project.

As stated in Chapter 3, each operational requirement must be prioritized as Essential, Desirable, or Optional to facilitate achievement of a balance among requirements, schedule, and budget. Sufficient time and resources must be provided to implement all of the Essential requirements and as many of the Desirable requirements as desired by the customer.

Depending on the nature and scope of your project, clarifying the operational requirements and developing the system requirements, system architecture, and software requirements may be your task. Alternatively, you may delegate it to one or more members of your planning team, or they may be provided as the starting point for your planning process.

Your understanding of the operational requirements and development constraints will influence your choice of the development model to be used and the procedures to be followed. Considerations include:

- Development of a user-intensive system may require prototyping to clarify the operational requirements and to provide information for design of the user interface.
- Development of the software for an embedded system may require the participation of you and your technical leader on the system engineering team.

- Development of staged delivery of system capabilities based on stable requirements and a stable architecture may indicate that an incremental build strategy is appropriate.
- Development of a first-of-its-kind system may require an evolutionary development strategy.
- An Agile process may be appropriate for development and ongoing enhancement of a Web-based application or in cases where the requirements are evolving or changing rapidly.

Development models are presented in Chapter 2.

An external customer (an acquirer) will typically specify the amount of money and the time available for the project, which may have to be negotiated to achieve a balance with the requirements. An internal customer may or may not provide money and/or resources for the project but will undoubtedly specify a schedule constraint for completion of the project, which may have to be negotiated. In any case, a contractual agreement in the form of a Statement of Work or a Memo of Understanding that contains items such as those listed in Section 3.3.2 of this text should be negotiated and accepted by you and your customer. Other planning activities include establishing an initial baseline of requirements, preparing estimates, and negotiating constraints to obtain a balance among requirements, cost, and schedule.

According to IEEE Standard 12207.1, every kind of plan, whether it is a project plan, a configuration management plan, a quality assurance plan, a training plan, or other kind of plan should contain the following information [12207]:

- needs to be satisfied
- success criteria
- work activities to be accomplished
- schedule, budget, and resources
- quality control measures
- change procedures and tracking of project history
- interfaces to relevant stakeholders
- roles to be played
- responsibilities and authorities
- resource acquisition plan
- skills acquisition plan, as needed

In addition every kind of plan must undergo a formal review and be accepted by the appropriate stakeholders, including the initial version of and subsequent changes to the plan.

In conjunction with preparing the generic information listed above, planning the specifics of a software project should include the activities contained in Tables 4.1 and 4.2.

Although the activities in Tables 4.1 and 4.2 are in sequential order, it should be understood that they, like most activities in software engineering, are best accomplished in an iterative manner. It should also be understood that the planning

TABLE 4.1 Preplanning activities for software projects

Preplanning activities
• Establish a working relationship with your customer/acquirer and other project stakeholders • Develop and/or clarify operational requirements and development constraints • Prioritize operational requirements • Establish the initial baseline of operational requirements • Develop system requirements and system architecture, as appropriate • Develop technical specifications for the software requirements • Establish traceability among operational requirements, system requirements, and software requirements • Obtain commitment to an initial version of the requirements by customer/acquirer and other appropriate stakeholders • Establish an initial baseline of operational requirements and technical specifications • Identify the resources needed and a schedule for developing the initial version of the project plan

TABLE 4.2A Comprehensive scope of planning activities for software projects (part 1)

Planning activities
• Plan for ongoing interactions with the customer in reviews, demonstrations, approvals, and acceptance of the delivered product • Plan for ongoing interactions with the user community in requirements elicitation, prototype demonstrations, and operational evaluations • Prepare a preliminary estimate of effort, cost, and schedule to determine feasibility of the project within the constraints on those factors • Refine the technical specifications for the system or product • Specify a development process and supporting processes • Develop an architecture decomposition view (ADV) of the product architecture and allocate requirements to the elements of the ADV • Specify the interfaces among modules in the ADV and the interfaces between modules and the external environment • Develop a work breakdown structure that includes work elements for the ADV modules • Develop work packages for the tasks in the work breakdown structure (WBS) • Define a schedule of objectively measurable milestones • Prepare a schedule network and identify the critical path(s) • Prepare a PERT (Program Evaluation and Review Technique) estimate of project duration • Identify numbers and kinds of resources needed, when they will be needed, and for how long • Prepare an estimate of optimal effort, cost, schedule, and resources • Negotiate with the customer to obtain a balance among requirements, cost, resources, and project duration that satisfies the project constraints • Finalize a contractual agreement with the customer that provides a balance among requirements, schedule, resources, and cost

TABLE 4.2B Comprehensive scope of planning activities for software projects (part 2)

Planning activities

- Define the organizational structure of the project team and specify roles, responsibilities, and authorities
- Establish the engineering environment to include standards, procedures, and tools for software development, verification, and validation
- Specify a version control process and a version control tool
- Establish a change control board for the project
- Identify work products to be placed under version control
- Establish a change control process that includes an impact analysis process
- Specify objective acceptance criteria for placing new and modified work products under version control
- Plan for verification and validation of work products
- Develop a measurement plan to measure and report quantity and quality of work products, effort, cost, progress, defects and other quality measures
- Develop a risk management plan to identify and confront risk factors on an on-going basis
- Develop plans, as appropriate, for the following kinds of activities:
  - ◦ management of subcontractors and vendors
  - ◦ coordination with associated projects and programs
  - ◦ coordination with the Independent (big "I") verification and validation organization
  - ◦ information security, including security clearances and access to information within various organizational entities
  - ◦ approvals as required by regulations, licensing agreements, and rights-in-data
  - ◦ installation, user training, and transition
  - ◦ ongoing maintenance activities
  - ◦ management of computing resources, facilities management, physical security
  - ◦ backup protection of product and process data
- Prepare a plan for updating the project plan on a periodic basis and as events dictate
- Document the project plan using the organization's standard format, a tailored format based on IEEE Standard 1058 [IEEE1058], or the format in Table 4.4 of this text
- Review the project plan with the customer, higher level managers, and other appropriate stakeholders; revise as needed
- Obtain commitment to the plan by the appropriate stakeholders
- Place the plan under version control, thus establishing the initial baseline of the plan

activities in Tables 4.1 and 4.2 may be more comprehensive than necessary for your project; they should be tailored to fit the needs of the project.

The items tagged "as appropriate" in Table 4.2 may not apply to your project; all other items in the list should be addressed at a level of detail appropriate for the nature and scope of your project and the criticality of the system or product to be developed. Items in Table 4.2 that are not included in your planning activities should be noted in the project plan, and brief justifications should be provided for not including them.

If you are fortunate to work in a well-managed organization most of the activities Tables 4.1 and 4.2 will have standard processes, procedures, and tools that will require little, if any, tailoring for your project. For example, the configuration management and independent testing processes may be standardized; there may be a set of development process models to choose from; there may be organizational units that have trained personnel, procedures, and tools for most or all of the supporting processes; and there may be internal consultants to assist in tailoring a template and preparing the project plan.

If you are not so fortunate, the planning process can require a large amount of effort. In this case it is tempting to circumvent most of the planning activities listed above. The risk to project success must be assessed for the items that are not planned; for example:

- What risks are incurred if you don't have a process for managing changes to requirements?
- What risks are incurred if you don't have a process for assessing the impact of changes to requirements, cost, schedule, or technology?
- What risks are incurred if you don't have a schedule with objective milestones?
- What risks are incurred if you don't have a process for measuring effort and defects?
- What risks are incurred if you don't practice risk management?

These issues and other aspects of project risk are presented in Chapter 9.

Of course, the level of detail in your plan should be appropriate to the scope and criticality of your project. The plan may consist of a few pages for a small project or many pages for a large project. An additional consideration is that the planning activities listed above may not be accomplished in the order listed. For example, some elements of the contractual agreement may be specified in a legally binding contract before you, the project manager, become involved in the project.

Planning activities should occur in a manner that fits the needs of the situation; for example, some planning activities may occur in an evolutionary manner, as in situations where requirements evolve and schedule milestones indicate that a working version of the system is to be demonstrated to the customer on a weekly basis. In these situations planning what to do next will evolve as the situation evolves.

4.4 THE CMMI-DEV-V1.2 PROCESS AREA FOR PROJECT PLANNING

According to CMMI-DEV-v1.2, the purpose of the project planning process is to establish and maintain plans that define project activities. Other CMMI-DEV-v1.2 process areas related to project planning include:

- requirements development,
- requirements management,
- risk management, and
- the technical solution process areas.

Specific goals of project planning in CMMI-DEV-v1.2 include establishing estimates, developing a project plan, and obtaining commitment to the plan. Specific practices related to these specific goals are listed in Table 4.3. The nature of those

TABLE 4.3 CMMI-DEV-v1.2 specific goals and practices of project planning

	Specific Practices	Chapters of This Text
	SG 1 establish estimates	
SP 1.1-1	Estimate the scope of the project	Chapter 3
SP 1.2-1	Establish estimates of work product and task attributes	Chapter 5
SP 1.3-1	Define project life cycle	Chapter 2
SP 1.4-1	Determine estimates of effort and cost	Chapter 6
	SG 2 develop a project plan	
SP 2.1-1	Establish the budget and schedule	Chapter 6
SP 2.2-1	Identify project risks	Chapter 9
SP 2.3-1	Plan for data management	Chapters 7 & 8
SP 2.4-1	Plan for project resources	Chapter 5
SP 2.5-1	Plan for needed knowledge and skills	Chapter 5
SP 2.6-1	Plan stakeholder involvement	Chapter 2
SP 2.7-1	Establish the project plan	Chapter 4
	SG 3 obtain commitment to the plan	
SP 3.1-1	Review plans that affect the project	Chapter 4
SP 3.2-1	Reconcile work and resource levels	Chapter 6
SP 3.3-1	Obtain plan commitment	Chapter 4

specific practices is discussed here. Techniques for accomplishing these practices are presented in subsequent chapters, as indicated in the third column of Table 4.3.

An estimate of effort, schedule, and resources based on requirements and constraints is an essential element of a project plan; stated differently, it is not possible to develop a plan without estimates, and it is not possible to develop estimates without requirements. The CMMI-DEV-v1.2 process areas of requirements development and requirements management (presented in Chapter 3) are thus closely related to the project planning process area.

Estimating the scope of a project is concerned with identifying all of the work activities to be accomplished. A work breakdown structure is typically used to document the scope of a project; work breakdown structures are presented in Chapter 5. Product size and complexity are the primary factors typically used to determine the amount of effort that will be required to develop a software product. Other factors include required performance, reliability, safety, and security. Therefore establishing estimates of product attributes such as size and complexity is a specific practice of specific goal SG 1 (SP 1.2-1). Other factors that will impact effort and schedule should also be considered.

An appropriate development model for a software project depends on the scope of work to be accomplished, the attributes of the product, and the phases of development to be included. SP 1.3-1 is concerned with defining a software development model that includes a set of development phases appropriate for the project scope and product attributes. Based on the outcomes of specific practices 1.1-1, 1.2-1, and

1.3-1, an estimate of effort and cost is developed using historical data, expert judgment, and other techniques presented in Chapter 6 of this text.

Achieving specific goal SG 1 in Table 4.3 provides the foundation for achieving SG 2, Develop a Project Plan. A project plan that satisfies SG 2 will contain a schedule and a budget that satisfy the effort and cost estimates developed in satisfying SG 1; said another way, effort, schedule, and cost estimates provide constraints that cannot be exceeded in the project plan.

Because effort is the product of people and time, a schedule can be derived from an effort estimate; for example, 54 person-months of effort might be scheduled as 6 persons for 9 months. The schedule specifies the predecessor tasks that must be completed, and the work products produced by those tasks that must be available, before subsequent tasks can begin; the schedule also specifies the successor tasks that can be performed after each task is completed. Sequencing constraints among work activities are thereby specified, and opportunities for concurrent work activities are identified. The budget is then allocated to each of the tasks to be accomplished.

A risk is a potential problem that (should it become a problem) will adversely impact a successful outcome of delivering an acceptable product on time and within budget. Risk factors should be identified in the project plan and appropriate mitigation actions planned. Risk management for software projects is presented in Chapter 9.

SP 2.3-1 in Table 4.3 involves developing a plan for management of project data, which includes all of the data in all areas of the project (project management, development processes, and supporting processes). The plan should specify the project data to be collected, a schedule for collecting and validating the data, report formats and distribution lists, and any requirements for privacy and security of project data.

SP 2.4-1 and SP 2.5-1 are concerned with identifying and planning for the resources necessary to perform a project. Resources include quantities of people and required skill levels, software tools, computing hardware, facilities, travel budget, and all other resources needed to conduct a project (SP 2.4-1). Because people are typically the most important resource for a software project, it is important to identify the knowledge and skill levels needed to perform a project (SP 2.5-1).

Stakeholders are people whose involvement in a project is necessary or desirable to ensure a successful outcome. Different kinds of people may have different kinds and amounts of involvement during different phases of a project. For example, involvement of user representatives is more important during requirements definition and product acceptance than during detailed design and coding. SP 2.6-1 is concerned with planning for involvement of identified stakeholders.

Achievement of SG 2 in Table 3.1 culminates in SP 2.7-1 (Establish the Project Plan). Format and contents of a software project management plan are presented and discussed in the following section of this chapter.

Obtain Commitment to the Plan (SG 3) includes three specific practices. SP 3.1-1 involves reviewing all plans that affect the project to understand project commitments. For example, documentation of requirements, plans for some or all of the supporting processes, and plans for activities such as installing the delivered system and training of users are typically developed and documented separately, and referenced in the project plan.

SP 3.2-1 (Reconcile Work and Resource Levels) is concerned with reconciling differences between estimates and available resources. An estimate might, for example, indicate that 10 people having specified skills will be needed to complete a project in the required 12 months. Perhaps the budget will support only 7 people at the requisite skill level.

Acceptable options for reconciling differences between the work to be done, resources, and available time include:

- reducing the requirements (de-scoping),
- increasing the quantity of resources (and the corresponding budget),
- using more productive resources, and
- extending the schedule.

Unacceptable options for achieving a balance among the work to be done, resources, and available time (i.e., requirements, effort, and schedule) in the project plan include descoping plans for measurement and control, peer reviews, and verification and validation; and planning for overtime.

The final step in the CMMI® Project Planning process area is obtaining commitment to the plan by the stakeholders who are responsible for performing and supporting execution of the plan. Commitments to each work activity identified in SP 1.1-1 should be obtained from the relevant stakeholders internal to your project as well as any external stakeholders such as senior management, external customer, and associated projects. Organizational interfaces and technical interface specifications should be specified and commitments must be obtained from the appropriate stakeholders to participate in maintaining the interfaces. At minimum, stakeholders will include you, your software architect, the quality assurance group, your manager, and a customer representative (e.g., the marketing department, or an external customer).

The approach to project planning indicated in Tables 4.1 and 4.2 and by the Project Planning process area of CMMI-DEV-v1.2 is the basis of the so-called plan-driven approach to managing software projects. It must be emphasized that the comprehensive set of tasks in Tables 4.1 and 4.2 and the specific goals and practices of CMMI-DEV-v1.2 are sufficient for the largest and most complex software projects. They must be tailored and adapted to fit the needs of each project. Unfortunately, some people misinterpret the plan-driven approach and reject it as being too cumbersome and bureaucratic without understanding that a plan must be tailored and adapted to the needs of each situation. The plan-driven approach to project planning is appropriate in two situations:

1. when there is a formal contractual agreement between an acquirer and a supplier, and/or
2. for large, complex projects internal to an organization.

4.4.1 Planning Agile Projects

An Agile approach may be appropriate for small projects (e.g., 10 or fewer software developers) when formal contractual conditions do not apply and in cases where the requirements are evolving or changing on a continuing basis and frequent delivery of evolving capabilities are to be delivered to users, for example, in a web-based application. The Agile development process is described in Section 2.5.3. As related there, planning for an Agile project involves:

- working with the customer to develop the product vision,
- determining project duration and level of effort to be applied,
- obtaining the commitment of a knowledgeable customer representative for ongoing involvement in the project,
- establishing the development environment,
- planning the frequency of iterations, and
- planning the frequency of delivery of evolving capabilities to users.

In addition a design metaphor must be established by the developers and the particular version of an Agile process to be used must be adopted and accepted by the project stakeholders; the Scrum version of Agile development is discussed in Section 2.5.3. A plan for ongoing reviews with customers, developers, and other stakeholders must be established, as must plans for periodically reviewing the planned and actual state of affairs and for reconciling differences. As with all software projects, initial assessment of risk factors and plans for ongoing risk management must be established. Planning an Agile project thus involves:

- developing the product vision;
- determining the project duration and level of effort;
- obtaining commitment of a knowledgeable customer representative;
- establishing the development environment;
- planning the frequency of iterations;
- planning the frequency of deliveries;
- establishing a design metaphor;
- adopting a version of Agile development;
- planning for ongoing reviews by the stakeholders;
- planning for periodic reviews of project status;
- conducting an initial risk assessment and risk mitigation; and
- planning for ongoing risk assessments and mitigation activities.

4.4.2 Balancing Agility and Discipline

As related in Chapter 2, the text *Balancing Agility and Discipline* by Boehm and Turner contrasts plan-driven and Agile approaches to software development and presents a middle-ground approach to achieving a balance that incorporates aspects of both approaches based on each particular situation [BOEHM04].

4.5 A MINIMAL PROJECT PLAN

At minimum, a plan for a software project, whether plan-driven or Agile, must include the following information:

- a statement of the purpose and objectives of the project
- identification of stakeholders and their objectives
- software development model to be used
- software development environment to be used
- platform technology to be used
- scope of work activities to be completed
- schedule of work activities including periodic, objective milestones
- skill levels and numbers of software personnel needed
- when various numbers and kinds of software personnel will be needed
- resources in addition to software personnel
- a plan for periodically reporting project status
- a risk management plan

As has been repeatedly emphasized, these elements of a project plan must be based on the development model to be used and scaled to the size and complexity of the project. Techniques for developing and documenting these elements of a project plan are presented in subsequent chapters.

The following sections of this chapter provide a template for and a description of the elements of a comprehensive plan-driven plan, an example of tailoring the template, and techniques for reducing the effort required to develop a software project management plan.

4.6 A TEMPLATE FOR SOFTWARE PROJECT MANAGEMENT PLANS

In the absence of a standard format for project plans in your organization, the template presented in Tables 4.4*a*, *b*, and *c* can be used. This template is similar to the format for project plans specified in IEEE/EIA Standard 1058. The template is comprehensive and is intended for the largest and most complex projects. It can be, and should be, tailored to fit the needs of each project; an example of tailoring is presented later in this chapter. The lengthy template is presented in three tables for ease of presentation. The topics in the tables are discussed in numerous places throughout the text; the “Discussed In” column of each table indicates the chapters and sections where the primary discussion of each topic is presented.

An annotated version of the template is contained in Appendix 4B of this chapter. In the appendix a series of questions is posed to assist you in preparing your project plan. An on-line version of the template that provides an easy-to-use outline for preparing project plans can be obtained at the URL listed in the Preface to this text.

An overview of the various elements of the template is presented in the following sections.

4.6.1 Front Matter

There are nine major sections in the template for software project management plans based on IEEE Standard 1058 plus the “front matter,” which includes a title

TABLE 4.4A Template for a Software Project Management Plan (part 1)

Contents	Discussed In
Front matter	Section 4.4.1
Title Page	
Revision History	
Preface	
Table of Contents	
List of Figures	
List of Tables	
Project summary	Section 4.4.2
1. Project Summary	
1.1 Purpose, Scope, and Objectives	
1.2 Assumptions	Section 4.4.2
1.3 Constraints	
1.4 Project Deliverables	Chapter 1
1.5 Schedule and Budget Summary	
Evolution, references, definitions	Section 4.4.3
2 Evolution of the Plan	
3 References	
4 Definitions	
Project organization	Section 4.4.4
5. Project Organization	
5.1 Project Interfaces	
5.2 Project Structure	Chapter 1
5.3 Roles and Responsibilities	Chapter 8

page, a revision history, a preface, and perhaps a table of contents, a list of figures, and a list of tables. The title page should contain:

- the project name,
- the version number of the plan,
- the date of issue,
- the name of the responsible party (you),
- your organization, and
- your contact information (telephone numbers, e-mail address).

Your project plan must be placed under version control as soon as commitment to it is obtained from the appropriate stakeholders.[18] As the plan evolves, the revision history will include an entry for each prior version of the plan. Each entry should include:

[18] see Section 3.2.5 for a discussion of version control

- the version number,
- date of release,
- sections changed, and
- the nature of the changes made.

In some situations it may be appropriate that each version of the plan (including the initial version) include the names, signatures, and titles of the persons who are authorized and responsible for approving the initial plan and changes to the plan. This person might be an external customer (the acquirer), or you, the project manager, in the case of an internal project.

The Preface should address the following issues:

- the purpose of the project,
- the context in which the project will occur, and
- the intended audience of the plan

Depending on the scope and formality of your project plan, it may be appropriate to include a table of contents, a list of figures, and a list of tables.

4.6.2 Project Summary

Section 1 of a software project management plan provides a summary of the project (sometimes referred to as the "executive summary"). The summary, as indicated in Table 4.4*a*, includes the purpose, scope, and objectives of the project.

Purpose, scope, and objectives of your project plan should address the following issues:

- purpose: the reason your organization is doing the project and the business needs or contractual agreements to be satisfied by the outcomes of the project
- scope: the scope of the project specifies the major work activities to be conducted and the relationship of this project to other projects and other ongoing activities
- objectives: the success criteria for the project; the objectives that must be satisfied to ensure an acceptable outcome; the work products to be delivered; and methods to be used in determining that the objectives have been satisfied
- exclusions: scope and objectives that are explicitly excluded from this project and/or from the resulting work products

The purpose should present the motivation for conducting the project, which might be, for example, to replace an existing system, to upgrade an existing system, to provide an automated system to replace a manual process, or to conduct a feasibility study and build a prototype of a future product. The purpose of the ATM project, for example, might be to replace or upgrade an existing ATM system, to provide a first-time ATM system for a financial institution, or to conduct a feasibility study and build a prototype of an advanced user interface for an ATM system. The prototype might involve using thumb prints, RFID cards, retinal scans, facial scans, or voice recognition and spoken commands for the user interface.

The scope of work activities for a project (e.g., the ATM system) might include refinement of the operational requirements, development of the technical specifications, design and implementation of the software, validation by an independent group, training of users, installation of the software at multiple sites. Or, it might be limited to modifying the design of an existing product and re-implementing some features, which would result in a new version of the product.

The objectives of your project should specify, as clearly as possible, the success criteria for the project. It may be that the delivery date is the most critical success factor, even if fewer features than desired are included in the delivered product. Or, it may be that developing an architectural structure for a family of products (a product line) that will maximize reuse of components in future systems is of high priority even if it means extending the schedule beyond the planned completion date.

In some cases it is important to clearly state what activities are excluded from the scope of your project. It might be, for example, that based on sensitivity of the customer's data, your project will not include testing of the system in the users' environment. Or, because your project involves improving the performance of some elements of a customer's operational system, and because the users must not be impacted by the changes, the user interface must not be modified.

Assumptions are conditions on which your project plan is based that you have not verified or are unable to verify at this time. You might assume, for example, that sufficient numbers of personnel who have the necessary skills will be available when needed. Or, you might assume that product complexity will not be a problem because you expect to have software developers who are familiar with this kind of system. Section 1.2 should list the factors and conditions that you assume will be true.

Constraints (section 1.3 of the management plan) are externally imposed conditions that your project must satisfy. Constraints are categorized as design constraints and process constraints. A design constraint might require reuse of existing components or building specified interfaces to another system. A process constraint might limit the money, resources, and/or time available to conduct the project.

Section 2.3 should thus state limitations that have been imposed on factors such as:

- schedule,
- budget,
- resources,
- software to be incorporated,
- technologies to be used, and
- interfaces of the product to other systems.

Project deliverables should specify the following items:

- work products to be delivered to the customer (see section 1.3)
- when and where they will be delivered,

- quantities and media of delivery,
- any special packaging and handling instructions.

Project deliverables may be limited to object code and a users' manual or they might include source code, design documentation, and test suite, all under version control using a specified version control tool, perhaps because the customer intends to maintain and evolve the delivered work products. The deliverables listed in the project plan should reflect those listed in the contractual agreement (MOU or SOW) and other contractual documents. A reference to those documents should be included.

The final item in the executive summary (section 1.5) is a summary of the schedule and budget for the project. Topics to be addressed in section 1.5 include:

- the time frame for this project (stated in elapsed time or by start and end dates),
- the major milestones and when are they scheduled to occur (by elapsed time from start or by dates of occurrence),
- the overall cost (in dollars or staff-hours), and
- costs and schedules for supporting processes and additional plans that are not included in this plan, with references to the documentation for those plans.

The duration of the project can be stated in elapsed time (e.g., six months) or by start and end dates (e.g., 3/15/20xx to 9/15/20xx). Start and end dates are preferable because they make the plan more specific. Cost may be stated in monetary units or total units of effort (e.g., staff-months). The latter measure may be preferable because of organizational sensitivities. Major milestones, such as customer reviews and demonstrations that involve the user community, or planned deliveries of subset versions of the final system or product, should be included in the project summary.

Costs and schedules for supporting processes such as subcontractor management or verification and validation by an independent organization, and additional plans for the "as appropriate" activities listed in Table 4.4*a*, (user training, etc.) that are not contained in this project plan should be listed.

Because of the nature of the information in the project summary, this section is typically completed last.

4.6.3 Evolution, Definitions, and References

Section 2 of a project plan (evolution of the Plan) describes the plan for updating the project plan on a periodic basis and as events dictate. The following issues should be addressed:

- the planned schedule for periodic updating of the plan,
- conditions and events for which unscheduled updates will be made
- method of controlling changes to the plan, and
- methods used to issue updates to the appropriate stakeholders

You might, for example, plan to update the project plan on a monthly basis and, with the customer's involvement, revise the cost, schedule, and/or requirements when a customer's change request is out of scope for the current plan. Changes to the plan should be controlled by a (small) group of authorized individuals (the project CCB; see Section 3.2.5) and tracked using a version control tool. As explained in Section 3.2.5, the CCB should be scaled to fit the needs of the project; on a small project, the CCB may consist of you, the project manager, and the customer. Although it may not be thought of in this way, you and the customer are the (informal) CCB if your project uses an Agile development model.

On a large system-level program that involves multiple coordinated projects, there may be several CCBs; you may be a member of the large program control CCB in addition to being the chair of the software CCB. In any case, updated plans and a brief explanation of the changes made to the project plan must be communicated to appropriate stakeholders; notifications of changes might be distributed to an e-mail distribution list.

Section 3 of the template in Table 4.4*a* provides references to related documents. This section should list the documents that are related to the plan, such as the Concept of Operations and the Technical Specifications, and indicate where they can be found. As always, related documents (and the project plan) should be scaled to fit the needs of the project. The ConOps, for example, may be a vision statement of a few pages or a small set of use cases. The Technical Specifications may be a document of comparable size. On the other hand, the ConOps and Technical Specifications may each be large documents if the envisioned system is large and complex.

Documents to be referenced should include the product and process foundation documents listed in Table 3.1 (operational requirements, system requirements and architecture, software requirements, design constraints, and the contractual agreement). References should be provided for other applicable documents, such as additional contractual documents, and also references to plans for associated projects. References to organizational policies and procedures, and applicable standards and guidelines to be followed should be listed.

Note, in particular, that requirements documentation and the project plan should be cross-referenced and maintained for consistency among requirements, schedule, budget, resources, and risk factors; however, they should be separate documents because they address different issues and are intended for different audiences. Requirements documents should not contain any information related to schedules, budgets, resources, or facilities required to conduct the project. Similarly the project plan should not contain any product information other than a brief overview of the product to be developed or modified and a reference to the requirements documentation. Traceability matrices can be used to cross-reference the project plan and related documents (see Section 3.2.4). Physical locations where the related documents can be found should be provided and path names and passwords for accessing electronic files should be provided.

Section 4, definitions, provides explanations of terms and acronyms used in the plan that may not be familiar to the intended audience of the plan. The definitions section should indicate the meanings of terms and acronyms and include references to other documents that contain terminology needed to understand this plan (e.g., IEEE Standard 610.12™—IEEE Standard Glossary of Software Engineering Terminology).

4.6.4 Project Organization

Section 5 of the plan is concerned with the way in which the project is organized. It describes the project's communication interfaces, the organizational structure for the project, and the roles and responsibilities for those who will conduct the project work activities.

Project interfaces (section 5.1 of the SPMP) should indicate the organizational entities with which you and your project members will interact and the individuals who will be the points of contact in those organizations.

Project interfaces may exist between your project and supporting entities within your organization such as an independent testing group and the parent organization, and to external entities such as the acquiring organization, subcontractors, vendors, and affiliated projects. You can use organizational charts and diagrams to depict your project's organizational interfaces. Names, titles, phone numbers, and e-mail addresses should be listed for those with whom you will interact.

Section 5.2, project structure, addresses the following issues:

- how the development team will be organized;
- how the development team will interact with supporting entities such as configuration management, quality assurance, and verification and validation; and
- the points of contact and the lines of communication within the project.

In particular, section 5.2 of your project plan should indicate the ways in which you (the project manager), the software architect (who may be you), the team leaders, and the software developers will interact. Graphical devices such as organizational charts or diagrams can be used to illustrate the lines of authority, responsibility, and communication within the project. An example of an organization structure for a software project is depicted in Figure 1.3 of this text.

Section 5.3, roles and responsibilities, specifies:

- the roles that must be played to accomplish the various development activities and supporting processes,
- the organizational units that will play the roles, and
- the persons responsible for playing those roles within the organizational units.

This section should specify the job titles and necessary skills of individuals and organizational units that are responsible for the various work activities and supporting processes. The individuals whose names are known (perhaps now, perhaps later) can be assigned to the responsibilities. But first, the roles to be played in conducting the project are identified. A role (e.g., the designer role) may be played by one or more individuals. One individual may play multiple roles, concurrently and/or sequentially. For example, an individual may be a designer first and later become a programmer; an individual may concurrently be a tester and the keeper of evolving product versions on a small project. One or more matrices that trace roles to development activities and supporting processes can be used to depict project roles and responsibilities.

4.6.5 Managerial Processes

Section 6 of a software project management plan, managerial processes, is the essence of a project plan. It contains the start-up plan, the work plan, the project control plan, the risk management plan, and the closeout plan, as listed in Table 4.4*b*.

The start-up plan for your project (section 6.1) is concerned with developing a plan for making initial estimates, doing the estimates, and developing a staffing plan, a plan for acquiring other necessary resources, and a training plan for the project team (if needed). Depending on the size and scope of the project, these plans may be incorporated directly into the project plan, or the project plan may contain reference to other documents and electronic files that contain the start-up plans.

The project estimation plan (section 6.1.1) should address the following issues:

- the plan for making initial and ongoing estimates (who will do them?, when will they be done?, who will approve them?);
- the tools and techniques that will be used to make estimates;
- how the estimates will be documented;

TABLE 4.4B Template for a Software Project Management Plan (part 2)

Contents	Discussed In
Managerial processes	Section 4.4.5
6. Managerial Processes	
6.1 Start-up Plan	
6.1.1 Project Estimation	Chapter 6
6.1.2 Staffing Plan	Chapter 5
6.1.3 Resource Acquisition Plan	
6.1.4 Project Staff Training Plan	
6.2 Work Plan	
6.2.1 WBS and Work Packages	Chapter 5
6.2.2 Schedule Dependencies	Chapter 5
6.2.3 Resource Allocation	Chapter 5
6.2.4 Budget Allocation	Chapter 5
6.3 Project Control Plan	
6.3.1 Requirements	Chapters 3 and 7
6.3.2 Schedule	Chapter 8
6.3.3 Budget	Chapter 8
6.3.4 Quality	Chapter 7
6.3.5 Metrics Plan	Chapter 8
6.3.6 Reporting Plan	Chapters 7 and 8
6.4 Risk Management Plan	Chapter 9
6.5 Closeout Plan	

- plans for periodic re-estimation of cost, schedule, staffing, and other resources required to complete the project;
- frequency of re-estimation; and
- the plan for re-estimating when requirements or other project conditions change.

When an estimate is prepared, the following items should be documented:

- person(s) who made the estimate;
- methods, tools, and techniques used to make the estimate;
- historical data used as the basis of estimation; and
- the estimator's level of confidence in the estimate.

Estimation methods, tools, and techniques are presented in Chapter 6 of this text.

In some cases an estimate of cost, duration, and resources might be completed and commitments made before you develop the project plan at the level of detail indicated in Table 4.4*b*. In those cases you must still validate the estimate. If you think the estimate is not valid, you must re-negotiate requirements, schedule, and budget; otherwise, you risk failure before you start.

The staffing plan (section 6.1.2) should indicate:

- the kinds of skills required;
- the numbers of people needed who have those skills;
- when they will be needed;
- for how long;
- how they will be obtained; and
- the person, or persons, responsible for acquiring the necessary personnel.

Tools such as resource Gantt charts, resource histograms, spreadsheets, and tables can be used to depict the staffing plan by skill level, by project phase, and by aggregations of skill levels and project phases. These techniques are discussed in Chapter 5.

The resource acquisition plan (section 6.1.3) should address the following issues:

- resources, in addition to personnel, that will be needed;
- quantities of each kind of resource needed;
- when the resources will be needed;
- the person, or persons, responsible for obtaining the resources; and
- approvals needed.

The resources may include items such as computer hardware and software, service contracts, transportation, facilities, and administrative services.

The resource acquisition plan should specify the points in the project schedule when the various acquisition activities should occur. Constraints on acquiring the necessary resources should be specified. This section can be expanded into additional subsections (labeled as 6.1.3.x, etc.) to accommodate acquisition plans for the various types of resources to be acquired. References to resource acquisition plans contained in separate documents should be included here.

The project staff training plan (section 6.1.4) indicates the kind and extent of training needed to ensure that the necessary skill levels, in sufficient numbers, will be available to successfully conduct the software project.

The need for special training may depend on the nature of the product to be developed and the skills needed to do the work. If training is required, a training plan should include:

- the types of training to be provided,
- numbers of personnel to be trained,
- entry and exit criteria for training, and
- the training methods to be used (e.g., lectures, consultations, mentoring, or computer-assisted training).

As in all plans, the staff training plan should include:

- schedule,
- budget,
- milestones, and
- responsible parties.

The training plan should include training needed in both technical and managerial skills.

The work plan (section 6.2) describes the work activities and the details of schedules, resources, and budget for your software project. The four subsections contain the work breakdown structure (WBS) and work packages, the work activities to be performed, schedule dependencies, resource allocation, and budget allocation.

Section 6.2.1 (WBS and work packages) documents:

- the scope of work activities included in this project plan,
- partitioning of the work activities,
- the level of detail provided in the plan, and
- documentation of the work activities.

The scope of work activities and the partitioning of that work are specified at the top level of the WBS. A WBS is a hierarchical decomposition of work activities; it is a fundamental tool for planning and controlling software projects. During project planning an initial version of the WBS should be developed. Work activities in the initial WBS should be decomposed so that:

- accurate estimates of resource requirements and schedule duration for each major work activity can be made,

- opportunities for reuse of software components can be identified, and
- the project's risk factors are exposed (both technical and managerial).

The level of decomposition for different work activities in the work breakdown structure may be different depending on factors such as the quality of the requirements, familiarity of the work, novelty of the technology to be used, and software components to be reused.

Work packages are used to specify, for each work activity, factors such as:

- the resources needed,
- estimated duration,
- work products to be produced,
- acceptance criteria for the work products,
- predecessor and successor work activities, and
- risk factors for the work activity.

Techniques and guidelines for constructing work breakdown structures and preparing work packages are presented in Chapter 5.

Schedule dependencies (section 6.2.2) indicate:

- tasks must be completed before subsequent tasks can begin;
- tasks that can be accomplished concurrently with other tasks; and
- schedule constraints imposed by dependencies on external factors such as vendor-supplied equipment and software, subcontractor-supplied software, and interfaces to other system components.

Tasks are the lowest level work activities in a WBS hierarchy; they are also the elements of the schedule. The project schedule should include frequent milestones that can be assessed for attainment using objective indicators to assess the scope and quality of work products completed at those milestones. Techniques that can be used to specify scheduling relationships include milestone charts, activity lists, schedule networks, critical path networks, PERT charts, activity Gantt charts, and resource Gantt charts. Examples and illustrations are provided in Chapter 5.

Resource allocation (section 6.2.3) documents, for each work activity in the work breakdown structure, the following:

- kinds and numbers of resources needed (people and other resources),
- when they are needed, and
- for how long.

Resources to be allocated include personnel by skill level, and may include hardware elements, software tools, travel budget, testing and simulation facilities, and administrative support. A table that documents, for each task, the resources required and when the task is scheduled to occur should be provided, plus an inverse table that shows, for each resource, the tasks to which it is allocated and when the task is scheduled to occur. The necessary data for these tables can be obtained from the work packages and the schedule network.

Budget allocation (section 6.2.4) documents the budget components allocated to each work activity and task in the WBS. The task-by-task budget should include the estimated cost for personnel by skill level to accomplish each task (in monetary units or staff-hours) and may include, as appropriate, costs for items such as travel, meetings with customer and users, computing resources, software development tools, testing tools, and administrative support for each work activity.

Budgets for higher level activities in the WBS (the sum of budgets for lower level activities and tasks in the WBS) should be documented. The total budget for each type of resource and their sum (the overall project budget) should be provided. The budget allocation can be developed in tabular form using a spreadsheet.

Section 6.3 of the project management plan (the project control plan) specifies the control procedures to be used in meeting product requirements, schedule, budget, and the quality standards of work processes and work products. Also, a plan for collecting project data and a reporting plan must be developed. Each element of the control plan should be consistent with your organization's standards, policies, and procedures for controlling software projects and should satisfy any contractual agreements for project control.

The requirements control plan (section 6.3.1) should address the following issues:

- how the requirements will be initially accepted as a product baseline;
- control mechanisms that will be used to measure, report, and control changes to the requirements baseline; and
- how the impact of requirements changes on product scope and quality, and project schedule, budget, resources, and risk factors will be assessed.

Configuration management mechanisms for controlling the requirements should include change control procedures, a version control tool, and a change control board. Techniques that can be used to measure and control requirements include traceability, prototyping, impact analysis, and reviews. These and other techniques are discussed in Chapter 3 of this text.

The schedule control plan (section 6.3.2) indicates the techniques that will be used to:

- measure and report the progress of work completed at the major and minor project milestones,
- compare actual progress to planned progress at the milestones, and
- implement corrective action when actual progress does not conform to planned progress.

Achievement of schedule milestones should be assessed using objective criteria to measure the quantity and quality of work products completed at each major and minor milestone.

The budget control plan (section 6.3.3) is concerned with:

- how the cost of completed work is to be determined,
- how comparisons of budgeted costs to actual costs will be made,
- how the cost of corrective action will be tracked, and
- tools and techniques that will be used to track and control the budget.

The budget plan should include frequent milestones that can be assessed for achievement using objective indicators to assess the quantity and quality of work products completed at those milestones. Mechanisms such as binary tracking and earned value reporting should be used to measure and report schedule progress and the cost of work completed versus work planned for completion. These mechanisms are described in Chapter 8.

The quality control plan (section 6.3.4) documents the mechanisms that will be used to measure and control the quality of the work processes and the evolving work products. Quality control mechanisms may include audits of work processes, verification and validation of work products, reviews, root cause analysis, and process assessments. Technical performance measurement can be used to track technical parameters that are allocated to individual elements of the system or product, such as actual versus allocated memory bytes and execution time cycles. Details are provided in Chapters 7 and 8.

The metrics plan (section 6.3.5) addresses the following issues:

- process and product data to be collected;
- how the data will be collected and validated;
- who will collect and validate project data;
- methods, tools, and techniques to be used;
- frequency of collecting the various types of metrics data;
- mechanisms for validating the metrics data; and
- how will the data be retained for future use.

Process and product metrics to be collected and validated should be consistent with the needs of the project and the reporting plan.

The reporting plan (section 6.3.6) documents:

- the mechanisms, report formats, and information flows that will be used to communicate the status of requirements, schedule, budget, scope, quality, and other status metrics;
- the kinds of reports that will be prepared;
- who will prepare and distribute the reports;
- frequency of preparing and distributing each type of report;
- formats to be used;
- methods, tools, and techniques that will be used; and
- individuals who will receive copies.

The nature of and frequency of reporting project status should be consistent with the project scope, criticality, risk, visibility, organizational policies, and contractual requirements. Metrics and reporting are discussed in Chapters 7 and 8.

Section 6.4 of the plan contains the risk management plan for your project. A risk is a potential problem that, if it materializes, will have a negative impact on your project. The risk management plan documents the following topics:

- mechanisms that will be used to identify, analyze, and prioritize project risk factors;
- mechanisms for developing action plans and contingency plans;
- staff members who will implement the plans;
- methods to be used to track the identified risk factors, evaluate changes in the levels of risk factors, and respond to those changes;
- staff members who will be responsible for monitoring risk factors; and
- how risk factors will be continuously identified, assessed, and mitigated on an ongoing basis during the project.

The kinds of risk factors that should be considered include:

- risks in the acquirer-supplier relationship;
- contractual risks;
- technological risks;
- risks caused by the size and complexity of the product;
- risks in the development and target environments;
- risks in personnel acquisition, skill levels, and retention;
- risks to schedule and budget;
- risks in vendor and subcontractor relations; and
- risks in achieving customer and user acceptance of the product.

Risk management is covered in Chapter 9.

The project closeout plan (section 6.5) documents:

- conditions and events that will indicate completion of the project;
- postmortem meetings and lessons-learned briefings that will be held;
- how lessons learned and analysis of project objectives achieved (and not achieved) will be documented, distributed, and archived;
- the plan for archiving project work materials; and
- how project members will be reassigned.

The remaining elements of the template for software project management plans (technical processes, supporting processes, additional plans, appendices, and index) are listed in Table 4.4*c* and discussed in the following sections.

4.6.6 Technical Processes

Section 7 of the plan (technical processes) documents the development processes to be used. This is the section where you specify the technical methods, tools, and

TABLE 4.4C Template for a Software Project Management Plan (part 3)

Contents	Discussed In
Technical processes	Section 4.4.6
7. Technical Processes	
7.1 Development Process Model	Chapter 2
7.2 Methods, Tools, and Techniques	
7.3 Infrastructure Plan	
7.4 Product Acceptance Plan	
Supporting processes	Section 4.4.7
8. Supporting Processes	
8.1 Configuration Management	Chapter 3
8.2 Verification and Validation	Chapter 2
8.3 Documentation	Chapter 1
8.4 Quality Assurance	Chapter 1
8.5 Reviews and Audits	Chapter 2
8.6 Problem Resolution	Chapter 1
8.7 Subcontractor Management	Chapter 1
8.8 Process Improvement	
Additional plans, appendixes, index	Section 4.4.8
9. Additional Plans	
Appendices	
Index	

techniques to be used; plans for establishing and maintaining the project infrastructure; and the product acceptance plan.

Section 7.1 (development process model) specifies:

- the development process model that will be used to develop the software product, and
- tailoring of the process model for this project.

The development process model should be described in sufficient detail to document:

- the relationships among major development activities and supporting processes (by specifying the flow of information and work products among activities and tasks),
- sequencing constraints among work products to be generated,
- reviews to be conducted,
- major milestones to be achieved,
- baselines to be established,

- project deliverables to be completed, and
- required approvals that span the duration of the project.

A combination of graphical and textual notations can be used to describe the development model. Any tailoring of an organization's standard process model should be indicated in this section. Development process models are described in Chapter 2 of this text.

Methods, tools, and techniques to be used to develop or modify the software are specified in section 7.2. The issues that should be addressed in this section are:

- development methods, techniques, software tools, and programming languages and other notations that will be used to specify, design, build, test, integrate, document, deliver, and modify and maintain the work products;
- technical standards, policies, procedures, and guidelines that will be used to govern development and/or modification of work products; and
- government regulations and laws, if any, that must be observed.

The infrastructure plan (section 7.3) addresses:

- the plan for establishing and maintaining the development environment (hardware, operating system, network, software utilities); and
- facilities, policies, procedures, and standards.

Infrastructure resources may include workstations, local area networks, desks, office space, and provisions for physical security, administrative personnel, and janitorial services.

The product acceptance plan (section 7.4) documents:

- how user and customer acceptance of the deliverable work products will be obtained;
- objective criteria to be used in determining acceptability of the deliverable work products;
- technical processes, methods, and tools that will be used in obtaining product acceptance; and
- if appropriate, the formal agreement for the acceptance criteria to be prepared and signed by representatives of the development organization and the acquiring organization during initial planning.

Validation methods such as testing, demonstration, analysis, and inspection should be specified. The relationship among the requirements, requirements-based test plans, and the list of required deliverable work products should be indicated. Traceability matrices can be used for this purpose.

4.6.7 Supporting Processes

Section 8 of a project management plan contains plans for the supporting processes that span the duration of the software project. These plans may include, but are not limited to those listed in Table 4.4*c* (configuration management, verification and validation, software documentation, quality assurance, reviews and audits, problem resolution, and subcontractor management). The eight supporting processes in Table 4.4*c* are those specified in IEEE Standard 12207; tailoring of the template for project plans may result in deletion or modification of some supporting processes. Additional processes may be added as appropriate.

Plans for supporting processes should be developed to a level of detail consistent with the other sections of the plan. In particular, the plan for each supporting process plan should include:

- roles,
- responsibilities,
- authorities,
- schedule,
- budget,
- resource requirements,
- risk factors, and
- work products.

The nature of, and types of supporting processes required may vary from project to project. However, the absence of a configuration management plan, verification and validation plan, quality assurance plan, joint customer-developer review plan, or problem resolution plan should be explicitly justified in any software project management plan that does not include them.

Plans for some supporting processes may be separately developed by the organizational entities that will provide the support. Those plans may be incorporated directly into your software project management plan or incorporated by reference. Referenced plans are considered to be part of the project plan. Supporting plans may be based on the organization's standard support processes, which can be included by reference.

The configuration management plan (section 8.1 of the SPMP) addresses the following issues:

- work products to be placed under version control;
- how readiness of work products for baselining (placement under version control) will be determined;
- how change requests and problem reports will be handled (logged, analyzed, and tracked);
- change control procedures to be used;
- members of the change control board;
- how stakeholders will be notified of changes to baselines;
- who will track changes in work products and analyze change trends;
- automated tools to be used for version control; and
- methods, tools, and conventions that must be used to satisfy your organization's policies, the contractual agreement, and post-release product support requirements

The verification and validation plan (section 8.2) addresses:

- who will do verification and validation (V&V);
- scope of activities that will be included;
- methods, tools, and techniques that will be used;
- the degree of independence between the development entities and the V&V entities of the project;
- automated tools to be used for V&V; and
- How will interactions with an Independent V&V organization be coordinated, if applicable.

Verification planning should result in plans for techniques such as traceability, milestone reviews, progress reviews, peer reviews, prototyping, simulation, and modeling. Validation planning should result in plans for techniques such as testing, demonstration, analysis, and inspection.

The documentation plan (section 8.3) should indicate:

- nondeliverable and deliverable documents that will be generated;
- templates or standard formats that will be used;
- responsible individuals for providing the necessary information, generating the various documents, reviewing them, and accepting them;
- documents that will be placed under version control;
- when review copies and initial baseline versions will be required; and
- who will get copies of the review and baselines versions of the documents.

Nondeliverable documents may include:

- requirement specifications;
- design documentation;
- source code;
- traceability matrices;
- test plans, meeting minutes;
- review reports;
- action items;
- change requests; and
- defect reports.

Deliverable work products may include:

- source code,
- object code,
- users' manual,
- on-line help system,
- regression test suite,

- configuration library,
- principles of operation,
- maintenance guide, and
- any other items specified in section 1.4 of the project plan (project deliverables).

The quality assurance plan (section 8.4) addresses:

- how assurance will be provided that the software project is fulfilling its commitments to the planned software processes and work products as specified in the requirements, software project management plan, supporting plans, and any policies, standards, procedures, or guidelines to which the process or the product must adhere;
- who will be responsible for process and product assurance; and
- the authorities, responsibilities, and lines of communication for those who will be responsible for process and product assurance.

Quality assurance procedures may include analysis, reviews, audits, and assessments. The quality assurance plan should indicate the relationships among the quality assurance, verification and validation, review, audit, configuration management, and assessment processes. The quality assurance plan must be developed and executed by an organizational entity (or entities) independent of you, the project manager, and incorporated by reference into your project plan.

The plan for reviews and audits (section 8.5) documents:

- the kinds of reviews and audits that will be conducted;
- who will conduct them; and
- schedules, resources, methods, and procedures that will be used to conduct project reviews and project audits.

This plan should include plans for joint customer-developer reviews, management reviews, developer peer reviews, quality assurance audits, and customer audits. Elements of this plan should be consistent with organizational policies, the project's contractual agreement, and other contractual documents.

The problem resolution plan (section 8.6) indicates:

- how problems in the work processes and work products will be reported, analyzed, prioritized, and resolved;
- how problems will be tracked to closure;
- the roles of organizational entities such as development, configuration management, the change control board, verification and validation, and quality assurance in problem resolution;
- how the relationship between problem resolution and risk management (section 5.4) will be managed; and

- how effort devoted to problem reporting, analysis, and resolution will be separately reported so that rework can be tracked and needed process improvements identified.

Subcontractor management plans (section 8.7) address:

- how subcontractors will be selected;
- who will be responsible for preparing subcontractor management plans;
- who will be responsible for providing the technical and managerial interfaces to subcontractors; and
- mechanisms of measurement, reporting, and control that will be used.

Plans for subcontractor management should include the items necessary to ensure successful completion of each subcontract. In particular, plans for:

- requirements management,
- monitoring of technical progress,
- schedule and budget reporting,
- product acceptance criteria, and
- risk management procedures

should be included in each subcontractor plan. Additional topics should be added as needed for successful completion of each subcontract. A reference to the official subcontract and prime contractor/subcontractor points of contact should be provided.

A plan for process improvement (section 8.8) documents:

- the frequency of assessment to determine areas for improvement,
- who will do the project assessments,
- who will develop and implement improvement plans, and
- who will implement improvement plans.

The process improvement plan should be closely related to the risk management and problem resolution plans. For example, root cause analysis of recurring problems may lead to simple process improvements that can significantly reduce rework during the remainder of the project. Proposed improvements should be carefully examined to identify those processes that can be improved without serious disruptions to your ongoing project and to identify those processes that can best be improved by process improvement initiatives at the organizational level.

4.6.8 Additional Plans, Appendixes, Index

Section 9 provides additional plans that should be included in your software project management plan, as appropriate. The following issue should be addressed:

- additional plans needed to satisfy product requirements, organizational policies, and contractual terms;
- who will prepare them; and
- who will execute them.

Additional plans for a particular project may include plans for:

- assuring that special safety or security requirements for the product are met,
- special facilities or equipment,
- product installation,
- user training,
- system integration,
- data conversion,
- system transition,
- product maintenance, or
- product support plans.

Appendixes may be included in a project plan to provide supporting details that would detract from the plan if included in the body of the plan. An index to the key terms and acronyms used throughout the project plan is optional, but is recommended to improve the usability of the plan.

Details of the mechanisms used to prepare and execute a project plan are provided in subsequent chapters of this text.

4.7 TECHNIQUES FOR PREPARING A PROJECT PLAN

Preparing a software project management plan using the template presented in Tables 4.4*a, b*, and *c* and described in Section 4.4 will be overwhelming if you, alone, are faced with developing all of the elements of the plan for a large project. Several factors should reduce the time and effort you will have to invest in preparing a project plan.

4.7.1 Tailoring the Project Plan Template

Tailoring is concerned with adding, deleting, and modifying elements of the template for your project plan. If you are planning a project for an internal customer in a familiar and well-defined development environment using a small team of experienced software developers and a standard set of supporting processes your tailoring of Tables 4.4*a, b*, and *c* might result in the tailoring indicated by deleting the indicated elements:

Title Page
Revision History
~~Preface~~
~~Table of Contents~~

- ~~List of Figures~~
- ~~List of Table~~
- 1 Project Summary
  - 1.1 Purpose, Scope, and Objectives
  - 1.2 Assumptions and Constraints
  - 1.3 Project Deliverables
  - 1.4 Schedule and Budget
- ~~2 Evolution of the Plan~~
- 3 References
- ~~4 Definitions~~
- ~~5 Project Organization~~
  - ~~5.1 Project Interfaces~~
  - ~~5.2 Project Structure~~
  - 5.3 Roles and Responsibilities
- 6 Managerial Processes
  - 6.1 Start-Up Plan
    - 6.1.1 Project Estimation
    - ~~6.1.2 Staffing Plan~~
    - ~~6.1.3 Resource Acquisition Plan~~
    - ~~6.1.4 Project Staff Training Plan~~
  - 6.2 Work Plan
    - 6.2.1 WBS and Work Packages
    - 6.2.2 Schedule Dependencies
    - ~~6.2.3 Resource Allocation~~
    - ~~6.2.4 Budget Allocation~~
  - 6.3 Project Control Plan
    - 6.3.1 Requirements
    - ~~6.3.2 Schedule~~
    - ~~6.3.3 Budget~~
    - ~~6.3.4 Quality~~
    - ~~6.3.5 Metrics Plan~~
    - ~~6.3.6 Reporting Plan~~
  - 6.4 Risk Management Plan
  - ~~6.5 Closeout Plan~~
- ~~7 Technical Processes~~
  - ~~7.1 Development Process Model~~
  - ~~7.2 Methods, Tools, and Techniques~~
  - ~~7.3 Infrastructure Plan~~
  - 7.4 Product Acceptance Plan
- ~~8 Supporting Processes~~
  - ~~8.1 Configuration Management~~
  - ~~8.2 Verification and Validation~~
  - ~~8.3 Documentation~~
  - ~~8.4 Quality Assurance~~
  - ~~8.5 Reviews and Audits~~
  - ~~8.6 Problem Resolution~~
  - ~~8.7 Subcontractor Management~~
  - ~~8.8 Process Improvement~~

9 Additional Plans
~~Appendices~~
~~Index~~

The tailored plan would have the resulting format:

Title Page
Revision History
1 Project Summary
  1.1 Purpose, Scope, and Objectives
  1.2 Assumptions and Constraints
  1.3 Project Deliverables
  1.4 Schedule and Budget
3 References
  5.3 Roles and Responsibilities
6 Managerial Processes
    6.1.1 Project Estimation Plan
    6.2.1 WBS and Work Packages
    6.2.2 Schedule Dependencies
    6.3.1 Requirements Control Plan
  6.4 Risk Management Plan
  7.4 Product Acceptance Plan

Tailoring is not meant to imply that the deleted elements are unimportant but that they will be conducted in the usual, familiar way (e.g., configuration management, verification and validation) and do not need to be documented in the project plan, or that they are not applicable to this project (e.g., there is no subcontractor plan because there are no subcontractors). Cases where elements would not be deleted are, for example, cases where the process to be used (e.g., for CM or QA) differs from the standard organizational process.

To maintain consistency among project plans within your organization, you should retain the numbering scheme from the template for project plans.

4.7.2 Including Predefined Elements

Your organization may have policies, procedures, checklists, document templates, one or more standard process models, tailoring guidelines, and examples that you can use to guide your preparation of the initial version of your project plan. This can significantly reduce the time and effort required.

4.7.3 Using Organizational Support

Your organization may have internal consultants and experts who can help you with areas such as requirements definition, tailoring the organization's standard development process, cost and schedule estimation, risk management, configuration management, tailoring and preparing project plans, and specialty disciplines such as human factors, safety, security, and reliability.

4.7.4 Leading a Planning Team

Small projects have small plans because the number of work activities to be planned and coordinated is small and because the projects often occur in stable, well-defined environments; large projects have correspondingly large plans. If you are the project manager of a large project your primary planning activity may involve coordinating the efforts of a planning team and integrating their work into a comprehensive project management plan. Members of the team may include specialists in areas such as those mentioned above (requirements engineering, tailoring the organization's standard development process, cost and schedule estimation, risk management, configuration management, tailoring and preparing project plans, and specialty disciplines such as human factors, safety, security, and reliability).

In the cases of large, complex projects you may need a "plan for planning." Like all plans, a plan for planning should include items such as:

- roles,
- responsibilities,
- authorities,
- schedule,
- budget,
- resources,
- risk factors, and
- work products.

4.7.5 Incremental Planning

Your initial project plan should be sufficiently comprehensive to include all of the work activities within the scope of your project. The level of detail in your initial plan should satisfy the following criteria:

1. the scope of the plan includes all of the major work activities to be accomplished
2. opportunities for reuse of existing components are identified;
3. effort, schedule, and resources for each identified work activity can be estimated with confidence;
4. predecessor and successor activities for each work activity are specified and a schedule is determined; and
5. complexities and risk factors are identified.

Different work activities may be decomposed to different levels. Familiar work activities and components identified for reuse in the product may satisfy the criteria at a high level; unfamiliar work, risk factors based on uncertainties, and use of new technologies may indicate the need to incorporate prototyping and feasibility studies in the plan.

During project execution the plan is updated and elaborated as specified in section 2 of your plan (evolution of the plan). For example, you may plan to update your project plan on a monthly basis or when external factors such as changes in the customer's requirements, difficulties with subcontractors, or delays in delivery of hardware components dictate the need for replanning.

4.8 KEY POINTS OF CHAPTER 4

- Operational requirements, technical specifications, and process constraints provide the basis for project planning.
- A software project management plan is a baseline-controlled written document. Appendix 4B to this chapter provides a template for developing software project management plans based on IEEE Standard 1058; an electronic copy is available at the URL listed in the Preface to this text.
- The comprehensive template for software project management plans presented in Tables 4.4*a*, *b*, and *c* can be, and should be, tailored to fit the needs of each project, as in the example of tailoring.
- Developing a software project management plan, like all software engineering processes, is best accomplished in an iterative manner. The initial version of the plan should be updated on a periodic basis and as events require.
- The level of effort devoted to project planning, and the level of detail in a project plan, both initially and ongoing, are determined by the risk factors created by not doing more.
- The level of detail in your initial project plan should satisfy the following criteria: effort, schedule, and resources for each identified work activity can be estimated with confidence; predecessor and successor activities for each work activity can be determined; opportunities for reuse of existing components are identified; and complexities and risk factors are identified.
- Acceptable options for obtaining a balance among effort, schedule, and requirements in your project plan include descoping the requirements, increasing the quantity of resources, using more productive resources, extending the schedule, and combinations of these options.
- Unacceptable options for achieving a balance among effort, schedule, and requirements include descoping the plans for measurement and control, peer reviews, verification and validation, and planning for overtime effort.
- SEI, ISO, IEEE, and PMI provide frameworks, standards, and guidelines for project planning (see Appendix 4A to this chapter)

REFERENCES

[BOEHM04] Boehm, B., and R. Turner. *Balancing Agility and Discipline*. Addison Wesley, 2004.

[CMMI06] SEI, *CMMI® Models and Modules*. http://www.sei.cmu.edu/CMMI/models/, 2006.

[IEEE1058] IEEE Std 1058™–1998. *IEEE Standard for Software Project Management Plans*. Engineering Standards Collection. IEEE Product: SE113. Institute of Electrical and Electronic Engineers, August 2003.

[IEEE12207] IEEE/EIA 12207.0/.1/.2. *Industry Implementation of International Standard ISO/IEC 12207:1995 Standard for Information Technology–Software Life Cycle Processes*. Engineering Standards Collection. IEEE Product: SE113. Institute of Electrical and Electronic Engineers, August 2003.

[PMI04] *A Guide to the Project Management Body of Knowledge*, 3rd ed. (PMBOK® Guide). Project Management Institute, 2004.

[SACMM02] SEI, *Software Acquisition Capability Maturity Model (SA-CMM), Version 1.03*. http://www.sei.cmu.edu/publications/documents/02.reports/02tr010.html

EXERCISES

4.1. CMMI-DEV-v1.2 lists four related process areas in the project planning process area:

requirements development,
requirements management,
risk management, and
technical solution.

Access the CMMI Web site at http://www.sei.cmu.edu/publications/documents/06.reports/06tr008.html, review the project planning process area, and briefly explain how each of the four related process areas is related to project planning.

4.2. What risk factors are created if a project does not have a written project plan?

4.3. What risk factors are created if a project manager does not maintain baseline control of the project plan?

4.4. Briefly describe the ways in which each of the following provides a basis for project planning:

a. operational requirements
b. software specifications
c. process constraints
d. product constraints

4.5. Briefly explain why identifying opportunities for reuse of existing components is an important aspect of project planning.

4.6. Why is planning for overtime effort an unacceptable option in a project plan?

APPENDIX 4A

FRAMEWORKS, STANDARDS, AND GUIDELINES FOR PROJECT PLANNING

4A.1 THE CMMI-DEV-v1.2 PROJECT PLANNING PROCESS AREA

Project Planning is a level 2 process area in the staged representation of the CMMI-DEV–v1.2 process framework [CMMI06]. According to CMMI-DEV–v1.2:

> The purpose of Project Planning (PP) is to establish and maintain plans that define project activities.

The specific goals and specific practices of project planning are:

SG 1 Establish Estimates
- SP 1.1 Estimate the Scope of the Project
- SP 1.2 Establish Estimates of Work Product and Task Attributes
- SP 1.3 Define Project Life Cycle
- SP 1.4 Determine Estimates of Effort and Cost

SG 2 Develop a Project Plan
- SP 2.1 Establish the Budget and Schedule
- SP 2.2 Identify Project Risks
- SP 2.3 Plan for Data Management
- SP 2.4 Plan for Project Resources
- SP 2.5 Plan for Needed Knowledge and Skills
- SP 2.6 Plan Stakeholder Involvement
- SP 2.7 Establish the Project Plan

SG 3 Obtain Commitment to the Plan
- SP 3.1 Review Plans That Affect the Project
- SP 3.2 Reconcile Work and Resource Levels
- SP 3.3 Obtain Plan Commitment

Related process areas are:

- Requirements Development
- Requirements Management
- Risk Management
- Technical Solution

4A.2 ISO/IEC AND IEEE/EIA STANDARDS 12207

As discussed in Chapter 1, IEEE/IEA Standard 12207–1996 is the Industry Implementation of International Standard ISO/IEC 12207:1995; it is the umbrella standard for software lifecycle processes for the IEEE's suite of software engineering standards. 12207 consists of three documents: 12207.0, Software life cycle processes; 12207.1: Life cycle data; and 12207.2, Implementation considerations [IEEE12207].

Section 5.2 of 12207.1, states that the generic purpose of all plans is to specify the activities to be performed and state when, how, and by whom the activities will be performed.

According to 12207.1, every kind of plan, whether it is a project plan, a configuration management plan, a quality assurance plan, a training plan, or other kind of plan should contain the following generic information:

- needs to be satisfied;
- success criteria;
- work activities to be accomplished;
- schedule, budget, and resources;
- quality control measures;
- change procedures and tracking of project history;
- interfaces to relevant stakeholders;
- roles to be played;
- responsibilities and authorities; and
- resource acquisition plan.

According to 12207.1, the specific contents of a project management plan includes items such as

- software life cycle model;
- project structural relationships;
- authority and responsibility of each organizational unit;
- the engineering infrastructure to be used, including items such as the test environment, standards, procedures, and tools;
- a work breakdown structure;
- scheduling of activities and tasks;
- quality management plan;

- configuration management plan;
- subcontractor management plans, as appropriate;
- verification and validation plans;
- risk management plan;
- tracking and reporting plan;
- plans for involvement of the acquirer and users;
- training plan; and
- security policy.

4A.3 IEEE/EIA STANDARD 1058

IEEE/EIA Standard 1058–1998 is the IEEE Standard for Software Project Management Plans. The format and content of project plans based on 1058 are presented in this chapter [IEEE1058].

4A.4 THE PMI BODY OF KNOWLEDGE

Section 4.3 of A Guide to the Project Management Body of Knowledge [PMI04], Develop Project Management Plan, states that the project management plan integrates and coordinates all subsidiary plans. Subsidiary plans include, but are not limited to:

- project scope management plan,
- schedule management plan,
- cost management plan,
- quality management plan,
- process improvement plan,
- staffing management plan,
- communication management plan,
- risk management plan, and
- procurement management plan.

Each of the subsidiary plans is detailed to the extent required by the specific project. In section 1.1, Purpose of the PMBOK® GUIDE, it is emphasized that the project management team is responsible for determining what is appropriate for any given project.

APPENDIX 4B

ANNOTATED OUTLINE FOR SOFTWARE PROJECT MANAGEMENT PLANS, BASED ON IEEE STANDARD 1058

4B.1 PURPOSE

This outline describes the format and content of software project management plans based on IEEE Std 1058. The standard does not specify the exact techniques to be used in developing a software project management plan, nor does it provide examples of software project management plans. Each organization using this standard should develop a set of practices and procedures to provide detailed guidance for preparing and updating of software project management plans based on the standard. These practices and procedures should take into account the environmental, organizational, and political factors that influence application of the standard.

Not all software projects are concerned with development of source code for a new software product. Some software projects consist of a feasibility study and definition of product requirements. Other software projects terminate on completion of the product design, and some projects are concerned with major modifications to existing software products. The standard is applicable to all types of software projects; applicability is not limited to projects that develop source code for new products. Project size or type of software product does not limit application of this standard. Small projects may require less formality in planning than large projects, but all components of the standard should be addressed by every software project.

Software projects are sometimes component parts of larger projects. In these cases the software project management plan may be a separate component of a larger plan or it may be merged into a system-level or business-level project management plan. Various parts of a project plan may be adaptations of, or direct implementations of the development organization's policies, procedures, and guidelines. In these cases references to those documents can be included with the appropriate tailoring information. For example, the quality assurance procedures for the project may be “the same way we always do it.” In that case the QA planning section of the project plan might incorporate a reference to the organization’s QA policies, standards, and procedures plus a description of the schedules and resources required for QA on this project.

4B.2 EVOLUTION OF PLANS

Developing the initial version of the software project management plan should be one of the first activities to be completed for a software project. As the project evolves, the nature of the work to be done will be better understood and plans will become more detailed. In addition requirements will change, personnel will come and go, and project conditions will change. Thus the project plan should contain a plan for revising the plan at periodic intervals and on occurrence of unusual events. Each version of the project plan should be placed under version control, and each version should contain a schedule for subsequent updates to the plan.

4B.3 OVERVIEW

The format and typical contents of software project management plans are described in this document. A software project management plan is the controlling document for managing a software project; it defines the technical and managerial processes necessary to develop software work products that satisfy the product requirements.

Some organizations may have generic project plans based on this standard, so that development of a particular project plan will involve tailoring of the generic plan in areas such as the process model, supporting processes, and infrastructure and adding project-unique elements such as schedule, budget, work activities, and risk management plan.

4B.4 FORMAT OF A SOFTWARE PROJECT MANAGEMENT PLAN

The individual or organization responsible for conducting a software project should also be responsible for preparing the software project management plan (the SPMP). The outline of elements in an SPMP is provided in Table 4A1.1.

The ordering of elements presented in Table 4A1.1 is not meant to imply that the sections must be developed in that order. The order of elements is intended for ease of reading, presentation, and use, and not as a guide to the order of preparation of the various elements of a SPMP. The various sections and subsections of a SPMP may be included by direct incorporation or by reference to other plans and documents.

Each version of a SPMP based on this outline should contain a title page, a signature page, and a change history.

TABLE 4A1.1 Format of a Software Project Management Plan

- Title Page
- Signature Page
- Change History
- Preface
- Table of Contents
- List of Figures
- List of Tables
- 1 Overview
    - 1.1 Project Summary
        - 1.1.1 Purpose, Scope, and Objectives
        - 1.1.2 Assumptions and Constraints
        - 1.1.3 Project Deliverables
        - 1.1.4 Schedule and Budget Summary
    - 1.2 Evolution of the Plan
- 2 References
- 3 Definitions
- 4 Project Organization
    - 4.1 External Interfaces
    - 4.2 Internal Structure
    - 4.3 Roles and Responsibilities
- 5 Managerial Process Plans
    - 5.1 Start-up Plan
        - 5.1.1 Estimation Plan
        - 5.1.2 Staffing Plan
        - 5.1.3 Resource Acquisition Plan
        - 5.1.4 Project Staff Training Plan
    - 5.2 Work Plan
        - 5.2.1 Work Activities
        - 5.2.2 Schedule Allocation
        - 5.2.3 Resource Allocation
        - 5.2.4 Budget Allocation
    - 5.3 Control Plan
        - 5.3.1 Requirements Control Plan
        - 5.3.2 Schedule Control Plan
        - 5.3.3 Budget Control Plan
        - 5.3.4 Quality Control Plan
        - 5.3.5 Reporting Plan
        - 5.3.6 Metrics Collection Plan
    - 5.4 Risk Management Plan
    - 5.5 Closeout Plan
- 6 Technical Process Plans
    - 6.1 Process Model
    - 6.2 Methods, Tools, and Techniques
    - 6.3 Infrastructure Plan
    - 6.4 Product Acceptance Plan
- 7 Supporting Process Plans
    - 7.1 Configuration Management Plan
    - 7.2 Verification and Validation Plan
    - 7.3 Documentation Plan
    - 7.4 Quality Assurance Plan
    - 7.5 Reviews and Audits
    - 7.6 Problem Resolution Plan
    - 7.7 Subcontractor Management Plan
    - 7.8 Process Improvement Plan
- 8 Additional Plans
- Annexes
- Index

Title Page

The title page should contain:

project name
version number of the plan
issuing organization

Signature Page

The signature page should contain the signature(s) and title(s) of the persons responsible for approving the SPMP.

Change History

The change history should include a list of all prior versions of the plan:

version number
date of release
sections changed
nature of changes

Preface

The preface of the SPMP should describe:

scope and context of the plan
intended audience
table of contents
list of figures
list of tables

4B.5 STRUCTURE AND CONTENT OF THE PLAN

1 PROJECT OVERVIEW

This section of the SPMP contains the following information: the purpose, scope, and objectives of the project; the major assumptions and constraints, a list of project deliverables, a summary of the project schedule and budget, and the plan for evolution of the SPMP.

1.1 Project Summary

1.1.1 Purpose, Scope, and Objectives (Subclause 1.1.1 of the SPMP)

Purpose: why are we doing this project? What business or system needs are to be satisfied by the outcomes of the project?

Scope: what activities are included in this project? What is the relationship of this project to other projects and ongoing work processes?

Objectives: what outcomes do we desire? What work products are to be delivered? How will satisfaction of objectives be determined?

Exclusions: what scope factors and objectives are explicitly excluded from this project and/or the resulting work products.

1.1.2 Assumptions and Constraints

Assumptions: what are the conditions that we have assumed will be true for this project?

Constraints: what constraints have been imposed on factors such as the schedule, budget, available resources, software to be reused, technology to be employed, and/or interfaces of the product to other products?

1.1.3 Project Deliverables What work products will we deliver to the customer? When and where must we deliver them? In what quantities and on what media? Are there any special packaging or handling instructions? Is there another document, such as a CDRL (Contractor's Data Requirements List) or PPL (Program Parts List), that contains the deliverables list? If so, where can this document be found?

1.1.4 Schedule and Budget Summary What is the time frame for this project? What is the overall cost (in dollars or staff-hours)? When are the major milestones scheduled to occur? What are the major supporting processes and additional plans for the project?

1.2 Evolution of the SPMP

What is the planned schedule for periodically updating the SPMP? Under what conditions will unscheduled updates occur? How will changes to the plan be controlled? What methods will be used to issue updates to the appropriate stakeholders?

2 REFERENCES

Where can additional documents related to this plan be found? (e.g., the Concept of Operations, system requirements specification, software requirements specification, and/or CDRL). Applicable standards and guidelines, such as IEEE or corporate standards for the project plan and supporting processes, should be included. Path names should be provided for access to electronic files.

3 DEFINITIONS

What are the meanings of the terms and acronyms are used in this document? What other documents contain terminology needed to understand this plan (e.g., IEEE Standard 610-12).

4 PROJECT ORGANIZATION

This section of the SPMP identifies interfaces to organizational entities external to the project, describes the project's internal organizational structure, and defines roles and responsibilities for the project.

4.1 External Interfaces

What are the organizational entities external to the project and where are the points of contact between the project and those entities? External interfaces may exist between the project and the parent organization, the acquiring organization, subcontractors, and affiliated projects. Organizational charts and diagrams may be used to depict the project's organizational interfaces.

4.2 Internal Structure

How is the development team organized? How does the development team interact with supporting entities such as configuration management, quality assurance, and verification and validation? Where are the points of contact and what are the lines of communication? Graphical devices such as organizational charts or diagrams can be used to illustrate the lines of authority, responsibility, and communication within the project.

4.3 Roles and Responsibilities

Which organizational units are responsible for the various work activities and supporting processes? A matrix that relates work activities and supporting processes to organizational units can be used to depict project roles and responsibilities.

5 MANAGERIAL PROCESS PLANS

This section of the SPMP specifies the project start-up plan, the risk management plan, the project work plan, the project control plan, and the project closeout plan.

5.1 Project Start-up Plan

Project start-up involves developing an estimation plan and doing estimates, developing a staffing plan, a plan for other necessary resources, and a training plan for the project team. Depending on the size and scope of the project, these plans may be incorporated directly into the SPMP, or the SPMP may contain reference to other documents and electronic files that contain the start-up plans.

5.1.1 Estimation Plan What is the plan for making initial and ongoing estimates? What are the details of the project cost, schedule, staff requirements, and other resources? What methods, tools, and techniques were used to make the estimates? What historical information was used? What is the estimator's level of confidence in the estimate? How will periodic re-estimates be made of cost, schedule, staffing, and other resources required to complete the project? How frequently will re-estimation be done? What is the plan for re-estimating when requirements or other project conditions change?

5.1.2 Staffing Plan What skills are required? How many people having what skill levels are needed? When will they be needed? For how long? How will the people be obtained? Who is responsible for acquiring the necessary personnel? Techniques such as Gantt charts, resource histograms, spreadsheets, and tables can be used to depict the staffing plan by skill level, by project phase, and by aggregations of skill levels and project phases.

5.1.3 Resource Acquisition Plan What resources, in addition to people, are needed? When are they needed? Who is responsible for acquiring them? What approvals are required? The resource acquisition plan may include plans for items such as the equipment, computer hardware and software, service contracts, transportation, facilities, and administrative services. The resource acquisition plan should specify the points in the project schedule when the various acquisition activities will be required. Constraints on acquiring the necessary resources should be specified. This section can be expanded into additional subsections of the form 5.1.3.x to accommodate acquisition plans for the various types of resources to be acquired. References to resource acquisition plans contained in separate documents must be included here.

5.1.4 Project Staff Training Plan What training is needed to ensure that necessary skill levels, in sufficient numbers are available to successfully conduct the software project? The training schedule should include the types of training to be provided, numbers of personnel to be trained, entry and exit criteria for training, and the training methods; for example, lectures, consultations, mentoring, or computer assisted training. The training plan should include needed training in both technical and managerial skills.

5.2 Work Plan

This section of the SPMP describes the work activities, and the details of schedule, resources, and budget for the software project.

5.2.1 Work Activities This section contains the project work breakdown structure. Work activities in the WBS should be decomposed to a level that exposes the project risk factors and allows accurate estimation of resource requirements and schedule duration for each work activity. Work packages should be used to specify, for each work activity, factors such as the necessary resources, estimated duration, work products to be produced, acceptance criteria for the work products, and predecessor and successor work activities. The level of decomposition for different work activities in the work breakdown structure may be different depending on factors such as the quality of the requirements, familiarity of the work, novelty of the technology to be used, and software components to be reused.

5.2.2 Schedule Allocation The allocated schedule provides answers to questions such as: What are the time-sequencing constraints among work activities? Where are the opportunities for concurrent work activities? What schedule constraints are caused by dependencies on external factors such as vendor-supplied equipment and software, interfaces to hardware components, and subcontractor-supplied software? The allocated schedule should include frequent milestones that can be assessed for attainment using objective indicators to assess the scope and quality of work products completed at those milestones. Techniques that can be used to specify schedule relationships include milestone charts, activity lists, activity Gantt charts, activity networks, critical path networks, and PERT charts.

5.2.3 Resource Allocation What resources are allocated to the various work activities in the work breakdown structure and the project schedule? Resources specified may include personnel by skill level and factors such as computing resources, software tools, special testing and simulation facilities, and administrative support. A separate line item should be provided for each type of resource needed. Allocation of resources to activities should be indicated. A summary of resource requirements for the various work activities can be collected from the work packages of the work breakdown structure and presented in tabular form.

5.2.4 Budget Allocation What elements of the budget are allocated to each of the major work activities in the work breakdown structure? The activity budget should include the estimated cost for personnel (in dollars or staff-hours by skill level) to accomplish each activity and may include, as appropriate, costs for factors such as travel, meetings, computing resources, software tools, special testing and simulation facilities, and administrative support. A separate line item should be provided for each type of resource for each activity. The work activity budget may be developed using a spreadsheet and presented in tabular form.

5.3 Control Plan

This section of the SPMP specifies the metrics, reporting mechanisms, and control procedures to be used in measuring, reporting, and controlling the product requirements, the project schedule, budget, and resources, and the quality of work processes and work products. Each element of the control plan should be consistent with the organization's standards, policies, and procedures for controlling software projects and with any contractual agreements for project control.

5.3.1 Requirements Control Plan How will requirements be accepted as product baselines? What control mechanisms will be used to measure, report, and control changes to the requirements baseline? How will the impact of requirements changes on product scope and quality, and project schedule, budget, resources, and risk factors be assessed? Configuration management mechanisms should include change control procedures, version control, and a change control board. Techniques that can be used for requirements control include traceability, prototyping and modeling, impact analysis, and reviews.

5.3.2 Schedule Control Plan What techniques will be used to measure the progress of work completed at the major and minor project milestones, to compare actual progress to planned progress, and to implement corrective action when actual progress does not conform to planned progress? Achievement of schedule milestones should be assessed using objective criteria to measure the scope and quality of work products completed at each milestone.

5.3.3 Budget Control Plan How will the cost of work completed, comparisons of planned cost to budgeted cost, and the cost of corrective action (when actual cost, schedule, scope, or quality does not conform to plans) be accomplished? How frequently will budget/cost information be provided? What tools and techniques will be used? Who will get copies of the information? The budget plan should include frequent milestones that can be assessed for achievement using objective indicators to assess the scope and quality of work products completed at those milestones. A mechanism such as earned value tracking should be used to report the budget and schedule plan, schedule progress, and the cost of work completed.

5.3.4 Quality Control Plan How will the quality of work processes and evolving work products be measured and controlled? Quality control mechanisms may include audits of work processes, verification and validation of work products, joint reviews, root cause analysis, and process assessments. Technical performance measurement should be used.

5.3.5 Reporting Plan What are the reporting mechanisms, report formats, and information flows to be used in communicating the status of requirements, schedule, budget, scope, quality, and other desired or required status metrics? The methods, tools, and techniques of communication should be included in the reporting plan. The frequency and nature of project measurement and control should be consistent with the project scope, criticality, risk, visibility, and contractual requirements.

5.3.6 Metrics Collection Plan How will necessary metrics data be collected, validated, and retained? What methods, tools, and techniques will be used? How frequently will various types of metrics data be collated, analyzed, and reported?

5.4 Risk Management Plan

What mechanisms will be used to identify, analyze, and prioritize project risk factors? How will contingency plans be developed? What methods will be used to track the identified risk factors, evaluate changes in the levels of risk factors, and respond to those changes? How will risk factors be identified, assessed, and mitigated on an ongoing basis during the project? Risk factors that should be considered include risks in the acquirer-supplier relationship, contractual risks, technological risks, risks caused by the size and complexity of the product, risks in the development and target environments, risks in personnel acquisition, skill levels, and retention, risks to schedule and budget, and risks in achieving user, customer, and acquirer acceptance of the product.

5.5 Project Closeout Plan

How will the project be concluded? How will staff members be reassigned? What postmortem meetings and briefings will be held? What is the plan for archiving project work materials? How will lessons learned and analysis of project objectives achieved be documented?

6 TECHNICAL PROCESS PLANS

This section of the SPMP specifies the development process model, the technical methods, tools, and techniques to be used to develop the various work products; plans for establishing and maintaining the project infrastructure; and the product acceptance plan.

6.1 Process Model

What process model will be used to develop the software product? The process model should describe the relationships among major project work activities and supporting processes by specifying the flow of information and work products among activities and functions, the timing of work products to be generated, reviews to be conducted, major milestones to be achieved, baselines to be established, project deliverables to be completed, and required approvals that span the duration of the project. The process model for the project should include project initiation and project termination activities. To describe the process model, a combination of graphical and textual notations may be used. Any tailoring of an organization's standard process model for a project should be indicated in this section.

6.2 Methods, Tools, and Techniques

What development methodologies, tools, techniques, programming languages, and other notations will be used to specify, design, build, test, integrate, document, deliver, and modify and maintain the work products internal to the project and those to be delivered to the customer? In addition, what technical standards, policies, procedures, and guidelines will be used to govern development and/or modification of work products?

6.3 Infrastructure Plan

What are the plan for establishing and maintaining the development environment (hardware, operating system, network, and software), and the policies, procedures, standards, and facilities required to conduct the software project? These resources may include workstations, local area networks, software tools for analysis, design, implementation, testing, and project management, desks, office space, and provisions for physical security, administrative personnel, and janitorial services.

6.4 Product Acceptance Plan

What is the plan for user, customer, and acquirer acceptance of the deliverable work products generated by the software project? What objective criteria will be used to determine acceptability of the deliverable work products? Will a formal agreement for the acceptance criteria be prepared and signed by representatives of the development organization and the acquiring organization? Any technical processes, methods, or tools required for product acceptance should be specified in the product acceptance plan. Validation methods such as testing, demonstration, analysis, and inspection should be specified in this plan. The relationship among the requirements, requirements-based test plans, and the list of required deliverable work products should be indicated—a traceability matrix can be used.

7 SUPPORTING PROCESS PLANS

This section of the SPMP contains plans for the supporting processes that span the duration of the software project. These plans may include, but are not limited to, items such as configuration management, verification and validation, software documentation, quality assurance, reviews and audits, problem resolution, and subcontractor management. Plans for supporting processes should be developed to a level of detail consistent with the other sections of the SPMP. In particular, the roles, responsibilities, authorities, schedule, budgets, resource requirements, risk factors, and work products for each supporting process should be specified. The nature and types of supporting processes required may vary from project to project; however, the absence of a configuration management plan, verification and validation plan, quality assurance plan, joint acquirer-supplier review plan, problem resolution plan, or subcontractor management plan should be explicitly justified in any software project management plan that does not include them. Plans for supporting processes may be incorporated directly into the software project management plan or incorporated by reference to other plans. Referenced plans are considered to be part of the project plan. Supporting plans may be based on the organization's standard support processes, which can be included by reference.

7.1 Configuration Management Plan

What work products will be placed under configuration management (version control)? How will readiness of work products for baselining be determined? How will change requests be handled (logging, analysis, and tracking)? What will be the change control procedures? Who will be the members of the change control board? How will stakeholders be notified of changes to baselines? Who will track changes in progress and analyze change trends? What automated tools will be used for configuration management? What methods, tools, and conventions must be used to satisfy corporate policies and product support requirements?

7.2 Verification and Validation Plan

Who will do verification and validation (V&V)? What scope of activities will be included? What methods, tools, and techniques will be used? What will be the degree of independence between the development entities and the V&V entities of the project? What automated tools will be used for V&V? Verification planning should result in plans for techniques such as traceability, milestone reviews, progress reviews, peer reviews, prototyping, simulation, and modeling. Validation planning should result in plans for techniques such as testing, demonstration, analysis, and inspection. Automated tools to be used in verification and validation should be specified.

7.3 Documentation Plan

The documentation plan should answer the following questions: What documents will be generated? What templates or standards will be used? Who will be responsible for providing the necessary information, generating the documents, reviewing them, and accepting them? Which documents will be placed under version control? When will review copies and initial baseline versions be due? Who will get copies of the review and baselines versions of the documents? Nondeliverable work products may include requirements specifications, design documentation, traceability matrices, test plans, meeting minutes, review reports, action items, change requests, and defect reports. Deliverable work products may include source code, object code, users' manual, on-line help system, regression test suite, configuration library, principles of operation, a maintenance guide, and any other items specified in subclause 1.1.3 of the SPMP.

7.4 Quality Assurance Plan

How will assurance be obtained that the software project is fulfilling its commitments to the planned software processes as specified in the requirements specification, the software project management plan, supporting plans, and any standards, procedures, or guidelines to which the process or the product must adhere? Quality assurance procedures may include analysis, inspections, reviews, audits, and assessments. The quality assurance plan should indicate the relationships among the quality assurance, verification and validation, review, audit, configuration management, system engineering, and assessment processes.

7.5 Reviews and Audits Plan

What schedules, resources, methods, and procedures will be used to conduct project reviews and audits? This plan should include plans for joint acquirer-supplier reviews, management progress reviews, developer peer reviews, quality assurance audits, and acquirer-conducted reviews and audits.

7.6 Problem Resolution Plan

How will problems in the work processes and work products be reported, analyzed, prioritized, and resolved? What will be the roles of organizational entities such as development, configuration management, the change control board, and verification and validation in problem resolution? Effort devoted to problem reporting, analysis, and resolution should be separately reported so that rework can be tracked and process improvement accomplished.

7.7 Subcontractor Management Plans

How will subcontractors be selected and managed? Who will be responsible for preparing subcontractor management plans? Who will be responsible for providing the technical and managerial interfaces to subcontractors? Plans based on this standard should be prepared to include the items necessary to ensure successful completion of each subcontract. In particular, requirements management, monitoring of technical progress, schedule and budget control, product acceptance criteria, and risk management procedures should be included in each subcontractor plan. Additional topics should be added as needed to ensure successful completion of the subcontract. A reference to the official subcontract and prime contractor/subcontractor points of contact should be provided.

7.8 Process Improvement Plan

How and when will the project be periodically assessed to determine areas for improvement? Who will do the project assessments? Who will develop and implement improvement plans? The process improvement plan should be closely related to the problem resolution plan. For example, root cause analysis of recurring problems may lead to simple process improvements that can significantly reduce rework during the remainder of the project. Proposed improvements should be carefully examined to identify those processes that can be improved without serious disruptions to an ongoing project and to identify those processes that can best be improved by process improvement initiatives at the organizational level.

8 ADDITIONAL PLANS

What additional plans are needed to satisfy product requirements and contractual terms by systematically managing the software project? Who will prepare them and execute them? What forms will they have? Additional plans for a particular project may include plans for assuring that special safety, privacy, or security requirements for the product are met, plans for special facilities or equipment, product installation plans, user training plans, integration plans, data conversion plans, system transition plans, product maintenance plans, and product support plans.

APPENDIXES

Appendixes may be included, either directly or by reference to other documents, to provide supporting details that could detract from the SPMP if included in the body of the plan.

INDEX

An index to the key terms and acronyms used throughout the SPMP is optional. An index is nevertheless recommended to improve the usability of the SPMP.