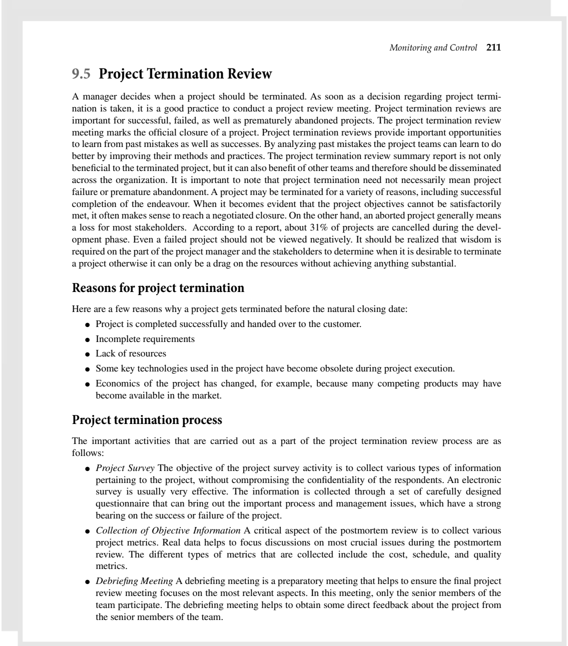
Guided Tour



Monitoring and Control **211**

9.5 Project Termination Review

A manager decides when a project should be terminated. As soon as a decision regarding project termination is taken, it is a good practice to conduct a project review meeting. Project termination reviews are important for successful, failed, as well as prematurely abandoned projects. The project termination review meeting marks the official closure of a project. Project termination reviews provide important opportunities to learn from past mistakes as well as successes. By analyzing past mistakes the project teams can learn to do better by improving their methods and practices. The project termination review summary report is not only beneficial to the terminated project, but it can also benefit of other teams and therefore should be disseminated across the organization. It is important to note that project termination need not necessarily mean project failure or premature abandonment. A project may be terminated for a variety of reasons, including successful completion of the endeavour. When it becomes evident that the project objectives cannot be satisfactorily met, it often makes sense to reach a negotiated closure. On the other hand, an aborted project generally means a loss for most stakeholders. According to a report, about 31% of projects are cancelled during the development phase. Even a failed project should not be viewed negatively. It should be realized that wisdom is required on the part of the project manager and the stakeholders to determine when it is desirable to terminate a project otherwise it can only be a drag on the resources without achieving anything substantial.

Reasons for project termination

Here are a few reasons why a project gets terminated before the natural closing date:

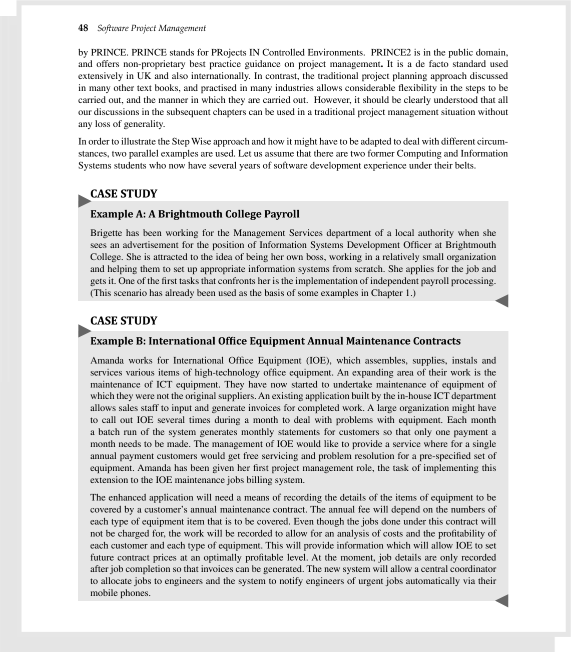
- Project is completed successfully and handed over to the customer.
- Incomplete requirements
- Lack of resources
- Some key technologies used in the project have become obsolete during project execution.
- Economics of the project has changed, for example, because many competing products may have become available in the market.

Project termination process

The important activities that are carried out as a part of the project termination review process are as follows:

- *Project Survey* The objective of the project survey activity is to collect various types of information pertaining to the project, without compromising the confidentiality of the respondents. An electronic survey is usually very effective. The information is collected through a set of carefully designed questionnaire that can bring out the important process and management issues, which have a strong bearing on the success or failure of the project.
- *Collection of Objective Information* A critical aspect of the postmortem review is to collect various project metrics. Real data helps to focus discussions on most crucial issues during the postmortem review. The different types of metrics that are collected include the cost, schedule, and quality metrics.
- *Debriefing Meeting* A debriefing meeting is a preparatory meeting that helps to ensure the final project review meeting focuses on the most relevant aspects. In this meeting, only the senior members of the team participate. The debriefing meeting helps to obtain some direct feedback about the project from the senior members of the team.

Additional new and important topics such as Software Project Management Tools, Software Project Termination, and Process Automation



48 *Software Project Management*

by PRINCE. PRINCE stands for PRojects IN Controlled Environments. PRINCE2 is in the public domain, and offers non-proprietary best practice guidance on project management. It is a de facto standard used extensively in UK and also internationally. In contrast, the traditional project planning approach discussed in many other text books, and practised in many industries allows considerable flexibility in the steps to be carried out, and the manner in which they are carried out. However, it should be clearly understood that all our discussions in the subsequent chapters can be used in a traditional project management situation without any loss of generality.

In order to illustrate the Step Wise approach and how it might have to be adapted to deal with different circumstances, two parallel examples are used. Let us assume that there are two former Computing and Information Systems students who now have several years of software development experience under their belts.

CASE STUDY

Example A: A Brightmouth College Payroll

Brigette has been working for the Management Services department of a local authority when she sees an advertisement for the position of Information Systems Development Officer at Brightmouth College. She is attracted to the idea of being her own boss, working in a relatively small organization and helping them to set up appropriate information systems from scratch. She applies for the job and gets it. One of the first tasks that confronts her is the implementation of independent payroll processing. (This scenario has already been used as the basis of some examples in Chapter 1.)

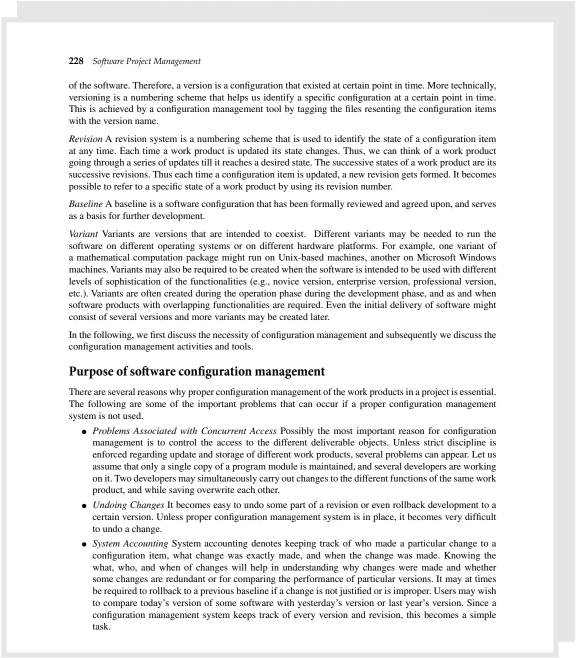
CASE STUDY

Example B: International Office Equipment Annual Maintenance Contracts

Amanda works for International Office Equipment (IOE), which assembles, supplies, instals and services various items of high-technology office equipment. An expanding area of their work is the maintenance of ICT equipment. They have now started to undertake maintenance of equipment of which they were not the original suppliers. An existing application built by the in-house ICT department allows sales staff to input and generate invoices for completed work. A large organization might have to call out IOE several times during a month to deal with problems with equipment. Each month a batch run of the system generates monthly statements for customers so that only one payment a month needs to be made. The management of IOE would like to provide a service where for a single annual payment customers would get free servicing and problem resolution for a pre-specified set of equipment. Amanda has been given her first project management role, the task of implementing this extension to the IOE maintenance jobs billing system.

The enhanced application will need a means of recording the details of the items of equipment to be covered by a customer's annual maintenance contract. The annual fee will depend on the numbers of each type of equipment item that is to be covered. Even though the jobs done under this contract will not be charged for, the work will be recorded to allow for an analysis of costs and the profitability of each customer and each type of equipment. This will provide information which will allow IOE to set future contract prices at an optimally profitable level. At the moment, job details are only recorded after job completion so that invoices can be generated. The new system will allow a central coordinator to allocate jobs to engineers and the system to notify engineers of urgent jobs automatically via their mobile phones.

Numerous new examples of important principles through a running case study



228 *Software Project Management*

of the software. Therefore, a version is a configuration that existed at certain point in time. More technically, versioning is a numbering scheme that helps us identify a specific configuration at a certain point in time. This is achieved by a configuration management tool by tagging the files resenting the configuration items with the version name.

Revision A revision system is a numbering scheme that is used to identify the state of a configuration item at any time. Each time a work product is updated its state changes. Thus, we can think of a work product going through a series of updates till it reaches a desired state. The successive states of a work product are its successive revisions. Thus each time a configuration item is updated, a new revision gets formed. It becomes possible to refer to a specific state of a work product by using its revision number.

Baseline A baseline is a software configuration that has been formally reviewed and agreed upon, and serves as a basis for further development.

Variant Variants are versions that are intended to coexist. Different variants may be needed to run the software on different operating systems or on different hardware platforms. For example, one variant of a mathematical computation package might run on Unix-based machines, another on Microsoft Windows machines. Variants may also be required to be created when the software is intended to be used with different levels of sophistication of the functionalities (e.g., novice version, enterprise version, professional version, etc.). Variants are often created during the operation phase during the development phase, and as and when software products with overlapping functionalities are required. Even the initial delivery of software might consist of several versions and more variants may be created later.

In the following, we first discuss the necessity of configuration management and subsequently we discuss the configuration management activities and tools.

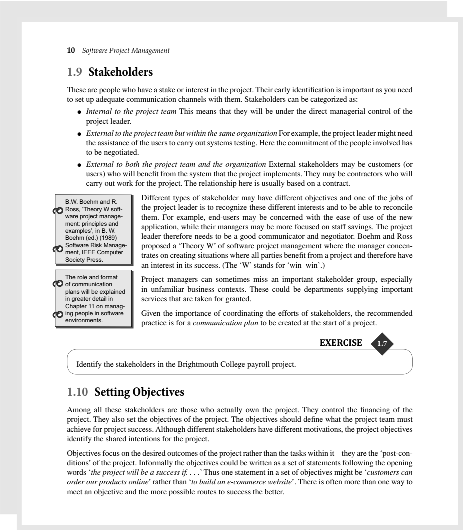
Purpose of software configuration management

There are several reasons why proper configuration management of the work products in a project is essential. The following are some of the important problems that can occur if a proper configuration management system is not used.

- *Problems Associated with Concurrent Access* Possibly the most important reason for configuration management is to control the access to the different deliverable objects. Unless strict discipline is enforced regarding update and storage of different work products, several problems can appear. Let us assume that only a single copy of a program module is maintained, and several developers are working on it. Two developers may simultaneously carry out changes to the different functions of the same work product, and while saving overwrite each other.
- *Undoing Changes* It becomes easy to undo some part of a revision or even rollback development to a certain version. Unless proper configuration management system is in place, it becomes very difficult to undo a change.
- *System Accounting* System accounting denotes keeping track of who made a particular change to a configuration item, what change was exactly made, and when the change was made. Knowing the what, who, and when of changes will help in understanding why changes were made and whether some changes are redundant or for comparing the performance of particular versions. It may at times be required to rollback to a previous baseline if a change is not justified or is improper. Users may wish to compare today's version of some software with yesterday's version or last year's version. Since a configuration management system keeps track of every version and revision, this becomes a simple task.

Enhanced coverage of topics such as Software Models, Configuration Management, Software Economics, Testing, and Software Project Quality

Margin notes highlighting terms, concepts and important references



10 *Software Project Management*

1.9 Stakeholders

These are people who have a stake or interest in the project. Their early identification is important as you need to set up adequate communication channels with them. Stakeholders can be categorized as:

- *Internal to the project team* This means that they will be under the direct managerial control of the project leader.
- *External to the project team but within the same organization* For example, the project leader might need the assistance of the users to carry out systems testing. Here the commitment of the people involved has to be negotiated.
- *External to both the project team and the organization* External stakeholders may be customers (or users) who will benefit from the system that the project implements. They may be contractors who will carry out work for the project. The relationship here is usually based on a contract.

Different types of stakeholder may have different objectives and one of the jobs of the project leader is to recognize these different interests and to be able to reconcile them. For example, end-users may be concerned with the ease of use of the new application, while their managers may be more focused on staff savings. The project leader therefore needs to be a good communicator and negotiator. Boehm and Ross proposed a 'Theory W' of software project management where the manager concentrates on creating situations where all parties benefit from a project and therefore have an interest in its success. (The 'W' stands for 'win–win'.)

B.W. Boehm and R. Ross, 'Theory W software project management: principles and examples', in B. W. Boehm (ed.) (1989) Software Risk Management, IEEE Computer Society Press.

Project managers can sometimes miss an important stakeholder group, especially in unfamiliar business contexts. These could be departments supplying important services that are taken for granted.

Given the importance of coordinating the efforts of stakeholders, the recommended practice is for a *communication plan* to be created at the start of a project.

The role and format of communication plans will be explained in greater detail in Chapter 11 on managing people in software environments.

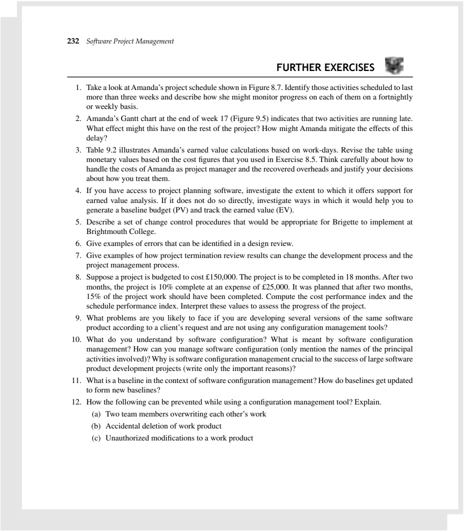
EXERCISE 1.7

Identify the stakeholders in the Brightmouth College payroll project.

1.10 Setting Objectives

Among all these stakeholders are those who actually own the project. They control the financing of the project. They also set the objectives of the project. The objectives should define what the project team must achieve for project success. Although different stakeholders have different motivations, the project objectives identify the shared intentions for the project.

Objectives focus on the desired outcomes of the project rather than the tasks within it – they are the 'post-conditions' of the project. Informally the objectives could be written as a set of statements following the opening words '*the project will be a success if. . . .*' Thus one statement in a set of objectives might be '*customers can order our products online*' rather than '*to build an e-commerce website*'. There is often more than one way to meet an objective and the more possible routes to success the better.



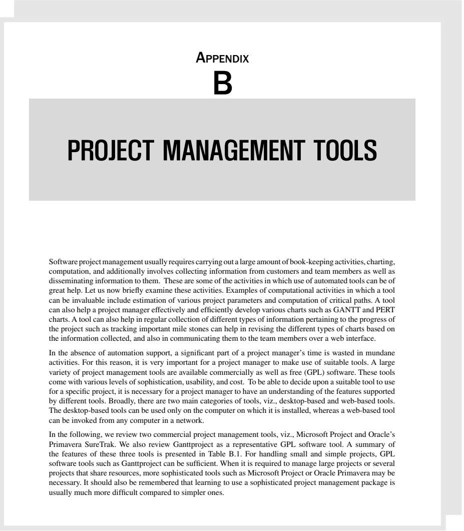
232 *Software Project Management*

FURTHER EXERCISES

1. Take a look at Amanda's project schedule shown in Figure 8.7. Identify those activities scheduled to last more than three weeks and describe how she might monitor progress on each of them on a fortnightly or weekly basis.
2. Amanda's Gantt chart at the end of week 17 (Figure 9.5) indicates that two activities are running late. What effect might this have on the rest of the project? How might Amanda mitigate the effects of this delay?
3. Table 9.2 illustrates Amanda's earned value calculations based on work-days. Revise the table using monetary values based on the cost figures that you used in Exercise 8.5. Think carefully about how to handle the costs of Amanda as project manager and the recovered overheads and justify your decisions about how you treat them.
4. If you have access to project planning software, investigate the extent to which it offers support for earned value analysis. If it does not do so directly, investigate ways in which it would help you to generate a baseline budget (PV) and track the earned value (EV).
5. Describe a set of change control procedures that would be appropriate for Brigette to implement at Brightmouth College.
6. Give examples of errors that can be identified in a design review.
7. Give examples of how project termination review results can change the development process and the project management process.
8. Suppose a project is budgeted to cost £150,000. The project is to be completed in 18 months. After two months, the project is 10% complete at an expense of £25,000. It was planned that after two months, 15% of the project work should have been completed. Compute the cost performance index and the schedule performance index. Interpret these values to assess the progress of the project.
9. What problems are you likely to face if you are developing several versions of the same software product according to a client's request and are not using any configuration management tools?
10. What do you understand by software configuration? What is meant by software configuration management? How can you manage software configuration (only mention the names of the principal activities involved)? Why is software configuration management crucial to the success of large software product development projects (write only the important reasons)?
11. What is a baseline in the context of software configuration management? How do baselines get updated to form new baselines?
12. How the following can be prevented while using a configuration management tool? Explain.
    (a) Two team members overwriting each other's work
    (b) Accidental deletion of work product
    (c) Unauthorized modifications to a work product

Enhanced pedagogy featuring additional solved examples, solved and unsolved exercises for holistic study

New Appendix on Project Management Tools



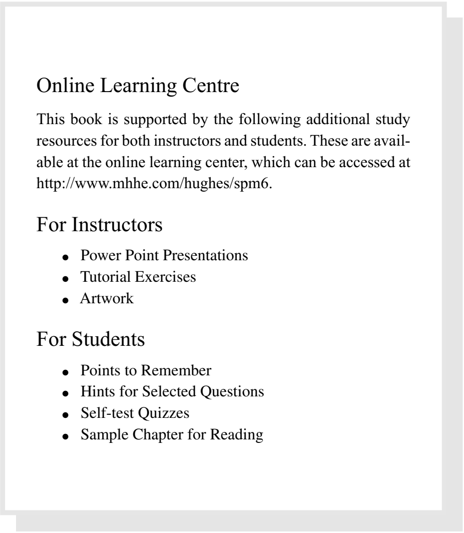
APPENDIX
B

PROJECT MANAGEMENT TOOLS

Software project management usually requires carrying out a large amount of book-keeping activities, charting, computation, and additionally involves collecting information from customers and team members as well as disseminating information to them. These are some of the activities in which use of automated tools can be of great help. Let us now briefly examine these activities. Examples of computational activities in which a tool can be invaluable include estimation of various project parameters and computation of critical paths. A tool can also help a project manager effectively and efficiently develop various charts such as GANTT and PERT charts. A tool can also help in regular collection of different types of information pertaining to the progress of the project such as tracking important mile stones can help in revising the different types of charts based on the information collected, and also in communicating them to the team members over a web interface.

In the absence of automation support, a significant part of a project manager's time is wasted in mundane activities. For this reason, it is very important for a project manager to make use of suitable tools. A large variety of project management tools are available commercially as well as free (GPL) software. These tools come with various levels of sophistication, usability, and cost. To be able to decide upon a suitable tool to use for a specific project, it is necessary for a project manager to have an understanding of the features supported by different tools. Broadly, there are two main categories of tools, viz., desktop-based and web-based tools. The desktop-based tools can be used only on the computer on which it is installed, whereas a web-based tool can be invoked from any computer in a network.

In the following, we review two commercial project management tools, viz., Microsoft Project and Oracle's Primavera SureTrak. We also review Ganttproject as a representative GPL software tool. A summary of the features of these three tools is presented in Table B.1. For handling small and simple projects, GPL software tools such as Ganttproject can be sufficient. When it is required to manage large projects or several projects that share resources, more sophisticated tools such as Microsoft Project or Oracle Primavera may be necessary. It should also be remembered that learning to use a sophisticated project management package is usually much more difficult compared to simpler ones.



Online Learning Centre

This book is supported by the following additional study resources for both instructors and students. These are available at the online learning center, which can be accessed at http://www.mhhe.com/hughes/spm6.

For Instructors

- Power Point Presentations
- Tutorial Exercises
- Artwork

For Students

- Points to Remember
- Hints for Selected Questions
- Self-test Quizzes
- Sample Chapter for Reading

New and exhaustive Online Learning Centre with additional study resources for both Instructors and Students