Software Project Management

Edited By

Mandeep Kaur

Printed by
EXCEL BOOKS PRIVATE LIMITED
A-45, Naraina, Phase-I,
New Delhi-110028
for
Lovely Professional University
Phagwara

SYLLABUS

Software Project Management

Objectives: To Impart the skills needed to the management of software project. Student will learn how to calculate cost and time estimation; Risk management; project planning and project evaluation; monitoring and control project development; various software quality standards.

Sr. No.	Description
1.	**Introduction to Software Project Management:** What is project? Software Project Vs. Other Types. Activities by Software Project Mgt. Plans, Methods and Methodologies. Problems with Software Projects.
2.	**Step Wise Project Planning:** Project Scope, Objectives, Infrastructure, Characteristics, Effort, Estimation, Risk Identification.
3.	**Programme Management & Project Evaluation:** Meaning, Managing Allocation of resources. Creating Programme. Individual Projects. Technical Assessment, Cost Benefit Analysis & Risk Evaluation.
4.	**Project Approach:** Intro. Technical Plan, Choice of Process Models: Waterfall, V-Process, Spiral. Prototyping. Incremental Delivery.
5.	**Effort Estimation:** Meaning, Problems with Estimation, Basis, Estimation Techniques. Albrecht Function Point Analysis. Functions Mark II. COCOMO Model.
6.	**Activity Planning:** Objectives, Project Schedule, Network Planning Model. Time Dimension. Indentifying Critical Path.
7.	**Risk Management:** Categories of Risk. Identification. Assessment. Schedule Risk. Applying PERT Technique.
8.	**Resource Allocation:** Indentifying Resource Requirements. Scheduling Resources. Publishing the Resource Schedule & Cost Schedule. Scheduling Sequence.
9.	**Monitoring & Control:** Creating Frameworks. Data collection. Visualising Progress. Cost Monitoring. Change Control.
10.	**Software Quality:** Introduction, Defining Software Quality. ISO 9126. Software Measures. Product Vs. Process Quality Management. External Standards. **Small Projects:-** Introduction, Problems with Student Projects, Content of project plan

CONTENT

Unit 1:	Introduction to Software Project Management *Mandeep Kaur, Lovely Professional University*	1
Unit 2:	Step Wise Project Planning *Mandeep Kaur, Lovely Professional University*	40
Unit 3:	Work Breakdown Structure *Mandeep Kaur, Lovely Professional University*	64
Unit 4:	Programme Management & Project Evaluation *Mandeep Kaur, Lovely Professional University*	77
Unit 5:	Project Approach *Mandeep Kaur, Lovely Professional University*	98
Unit 6:	Effort Estimation *Mandeep Kaur, Lovely Professional University*	114
Unit 7:	Activity Planning *Deepak Mehta, Lovely Professional University*	140
Unit 8;	Network Planning Model *Deepak Mehta, Lovely Professional University*	149
Unit 9:	CPM /PERT *Deepak Mehta, Lovely Professional University*	166
Unit 10:	Risk Management *Deepak Mehta, Lovely Professional University*	178
Unit 11:	Resource Allocation *Mandeep Kaur, Lovely Professional University*	209
Unit 12:	Monitoring & Control *Mandeep Kaur, Lovely Professional University*	229
Unit 13:	Software Quality *Deepak Mehta, Lovely Professional University*	258
Unit 14:	Small Projects *Mandeep Kaur, Lovely Professional University*	270

Unit 1: Introduction to Software Project Management

Notes

CONTENTS

Objectives

Introduction

1.1 What is Project?

1.1.1 Sequence of Activities

1.1.2 One Goal

1.1.3 Specified Time

1.1.4 Within Budget

1.1.5 According to Specification

1.2 Software Projects vs. Other Types

1.3 Activities by Software Project Management

1.3.1 Planning

1.3.2 Methodologies

1.3.3 Project Methods/Techniques

1.4 Problems with Software Project

1.5 Summary

1.6 Keywords

1.7 Review Questions

1.8 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize project
- Describe software project vs. other types
- Explain activities by software project management
- Discuss problems with software projects

Introduction

Software project management includes the tools, techniques, and knowledge essential to deal with the growth of software products. In Software Project Management, the end users and developers require to know the cost of the project, duration and length. It is a process of managing, allocating and timing resources to develop computer software that meets necessities. It consists of eight tasks:

- Problem Identification
- Problem Definition

Notes

- Project Planning
- Project Organization
- Resource Allocation
- Project Scheduling
- Tracking, Reporting and Controlling
- Project Termination

In problem identification and definition, the conclusions are made as approving, declining or prioritizing projects. In problem identification, project is recognized, defined and justified. In problem definition, the use of the project is clarified. The main product is project proposal.

In project planning, it explains a series of actions or steps that are needed to for the growth of work product. In project organization, the functions of the personnel are incorporated. It is done in corresponding with project planning.

In resource allocation, the resources are allocated to a project in order that the goals and objectives are attained. In project scheduling, resources are allocated so that project objectives are attained within a sensible time span.

In tracking, reporting and controlling, the process engage whether the project results are in accordance with project plans and performance specification. In controlling, suitable action is taken to correct improper deviations. In project termination, the concluding report is submitted or a release order is signed.

Did u know? What is Project Management?

The methods and regulation used to define goals, plan and monitor tasks and resources, identify and resolve issues, and control costs and budgets for a specific project is known as project management.

1.1 What is Project?

A project is a sequence of unique, complex, and connected activities having one goal or purpose and that must be completed by a specific time, within budget, and according to specification. This definition tells us quite a bit about a project. To appreciate just what constitutes a project let's look at each part of the definition.

1.1.1 Sequence of Activities

A project includes a number of activities that must be completed in some particular order, or sequence. An activity is a defined chunk of work. The chain of the activities is based on technical requirements, not on management concern. To conclude the sequence, it is helpful to think in terms of inputs and outputs as follows:

- What is needed as input in order to begin working on this activity?
- What activities produce those as output?

Notes The output of one activity or set of activities becomes the input to another activity or set of activities.

Notes

Unique Activities

The activities in a project must be unique. A project has never happened before, and it will never happen again under the same conditions. Something is always different each time the activities of a project are repeated. Usually, the variations are random in nature—for example, a part is delayed, someone is sick, a power failure occurs. These are random events that can happen, but we never are sure of when, how, and with what impact on the schedule. These random variations are the challenge for the project manager.

Complex Activities

The activities that make up the project are not simple, repetitive acts, such as mowing the lawn, painting the house, washing the car, or loading the delivery truck. They are complex. For example, designing an intuitive user interface to an application system is a complex activity.

Connected Activities

Connectedness implies that there is a logical or technical relationship between pairs of activities. There is an order to the sequence in which the activities that make up the project must be completed. They are considered connected because the output from one activity is the input to another. For example, we must design the computer program before we can program it.

Unconnected Activities

You could have a list of unconnected activities that must all be complete in order to complete the project. For example, consider of developing a payroll system. With some exceptions, the different modules of payroll system like, data entry module, updation module, calculation module, etc. can be developed separately in any order. But the pay roll system as a whole cannot be completed until all its modules are completely developed, but the different modules may be developed in any order. So developing a payroll system with various modules in which the modules can be developed in any order is not considered a project according to the definition.

1.1.2 One Goal

Projects must have a single goal.

Example: Consider the development of Management Information System (MIS) project for an institute. Each module of the MIS like, Personal information system accounting system, administration control system, examination system can be consider as a subproject, each of which is a project in its own right.

This division makes for better management control. This artificial decomposition of a complex project into subprojects often simplifies the scheduling of resources and reduces the need for interdepartmental communications while a specific activity is worked on. The downside is that the projects are now interdependent. Even though interdependency adds another layer of complexity and communication, it can be handled.

1.1.3 Specified Time

Projects have a specified completion date. This date can be self-imposed by management or externally specified by a customer or government agency. The deadline is beyond the control of anyone working on the project. The project is over on the specified completion date whether or not the project work has been completed.

Notes

1.1.4 Within Budget

Projects also have resource limits, such as a limited amount of people, money, or machines that are dedicated to the project. While these resources can be adjusted up or down by management, they are considered fixed resources to the project manager.

Example: Suppose a company has only one Web designer at the moment. That is the fixed resource that is available to project managers. Senior management can change the number of resources, but that luxury is not available to the project manager. If the one Web designer is fully scheduled, the project manager has a resource conflict that he or she cannot resolve.

1.1.5 According to Specification

The customer, or the recipient of the project's deliverables, expects a certain level of functionality and quality from the project. These expectations can be self-imposed, such as the specification of the project completion date, or customer-specified, such as producing the sales report on a weekly basis.

Although the project manager treats the specification as fixed, the reality of the situation is that any number of factors can cause the specification to change. For example, the customer may not have defined the requirements completely, or the business situation may have changed (this happens in long projects). It is unrealistic to expect the specification to remain fixed through the life of the project. Systems specification can and will change, thereby presenting special challenges to the project manager.

Task Connectedness implies that there is a logical or technical relationship between pairs of activities. Explain

Self Assessment

Fill in the blanks:

1. In project planning, it explains a series of actions or steps that are needed to for the growth of ..... product.
2. A project has never happened before, and it will never happen again under the …… conditions.
3. In project scheduling, resources are allocated so that project objectives are attained within a sensible .....
4. The chain of the activities is based on technical requirements, not on management .....

1.2 Software Projects vs. Other Types

Software Projects

Software projects are disreputably hard to define. Unlike a house, you can't see software or touch it or feel it or visualize it and it's hard for the layman to get a theoretical grip of its size or cost or how long it might take to construct. Software projects have numerous properties that make them exceptionally different to other kinds of engineering project. The properties of Software Projects are as follows:

Notes

- ***The technology changes very rapidly:*** Most large software projects utilize new technology; for many projects.
- ***The product is intangible:*** It's very difficult to declare a bridge is 90% complete if there is not 90% of the bridge there. It is easy to declare that a software project is 90% complete, even if there are no visible outcomes.
- ***Large software projects are generally modified:*** Most large software systems are one-off, with experience gained in one project being of little help in another.
- ***We don't have much knowledge:*** Software engineering is a new discipline, and so we basically don't have much understanding of how to engineer large scale software projects.

Examples:

- Freshmeat
- Apache Software Foundation
- CPAN
- CTAN
- CRAN

Construction Projects

The project produces an artifact. The worth generated by the project is implanted in the artifact. The artifact may be a compound system with human and mechanical components.

Examples:

- Warship
- Jubilee line extension
- Millennium dome
- Customer call centre
- Method guidebook
- IT system

Research Projects

The project generates information. The information may be formally symbolized as models, patterns or patents. Or the information may be embedded in a working process or artifact.

Examples:

- Business modeling
- Developing a model of the UK economy
- Developing a new species of wheat
- Developing novel approaches to project management.
- Military intelligence/code breaking.
- The analysis, testing, QA or evaluation portions of a larger project.

Notes

Reengineering Projects

The project generates a desired change in some system or process.

Examples:

- Taking sterling into the Euro
- Renumbering the UK telephone system
- Implementing PRINCE project management practices into a large organization.
- Designing and installing an Intranet.

Procurement Projects

The project create a business relationship contractually based with a chosen supplier for a describe product or service based on a fixed specification and/or a described specification process.

Examples:

- Outsourcing a specific construction or research project
- Outsourcing a complete business function (such as IT).
- Imposing new rules and measures on a regulated industry.

Business Implementation Projects

The project constructs an operationally effective process. The value generated by the project is embedded in the process.

Examples:

- Developing a new business process to repackage and exploit existing assets.
- Installing e-commerce

Some Projects are difficult to classify under this Scheme

Nationalized symbolic programmes:

- Putting a man on the moon by the end of the decade.
- Mitterrand's Grandes Projects.
- New Labour

Huge medical programmes:

- Creating an artificial heart.
- Mass inoculation programmes.

Other hybrid or interdisciplinary projects

- Pilot projects
- Moving offices

Notes

Caution In most cases, this complexity occurs from an ambiguity about the primary purpose of the project.

Where do we start? How do we know when to stop? When can (should) we evaluate the results?

Each type of project yields miscellaneous answers to these questions - and this implies that each type of project needs a somewhat different process and management style.

	Start	Stop	Evaluate
Construction	With a set of requirements. With a defined solution.	When the artifact is "complete". When the requirements are satisfied.	On delivery of the artifact. Over the lifetime of the artifact.
Research	With a hypothesis. With a problem.	When the time runs out. When we detect diminishing returns.	When the knowledge is confirmed or disconfirmed by later work. When the knowledge is used by later work.
Reengineering	With a problem. With an opportunity. With an (imported) solution.	When we seem to be ahead of the game. When some higher process changes the game we're playing.	At any time.
Procurement	With a set of requirements. With a defined solution.	We construct a tender document that is "complete". We sign a contract with the supplier who seems to be ahead at the end of the tender period.	Over the lifetime of the contract. On completion of the contract.
Business Implementation	With an opportunity. With a business concept.	When the process is operational. When the process has been running smoothly for a defined period. When the business benefits are starting to become visible.	When the process has been running smoothly for a defined period. When the business benefits are starting to become visible. Over the lifetime of the process.

1.3 Activities by Software Project Management

1.3.1 Planning

Project planning is a feature of Project Management, which includes of various processes. The aim of these processes is to make sure that various Project tasks are well coordinated and they meet the various project goals including timely completion of the project. Project Planning is a feature of Project Management that focuses a lot on Project Integration. The project plan shows the current status of all project activities and is used to observe and control the project.

The Project Planning tasks ensure that various elements of the Project are coordinated and therefore guide the project execution.

Notes

Project Planning assists in

- Facilitating communication
- Provides overall documentation of assumptions/planning decisions
- Monitoring/measuring the project progress.

The Project Planning Phases can be generally classified as follows:

- Development of the Project Plan
- Execution of the Project Plan
- Change Control and Corrective Actions.

Project Planning is an ongoing effort all through the Project Lifecycle.

Caution "If you fail to plan, you plan to fail."

Project planning is vital to the success of the Project.

Careful planning exact from the beginning of the project can help to keep away from costly mistakes. It provides an assurance that the project execution will achieve its goals on schedule and within budget.

Steps in Project Planning

Project Planning covers the various features of the Project. Usually Project Planning is believed to be a process of estimating, scheduling and assigning the projects resources in order to bring an end product of proper quality. Though, it is much more as it can assume a very strategic role, which can determine the very success of the project. A Project Plan is one of the decisive steps in Project Planning in General!

Typically Project Planning can contain the following types of Project Planning:

1. Project Scope Definition and Scope Planning
2. Project Activity Definition and Activity Sequencing
3. Time, Effort and Resource Estimation
4. Risk Factors Identification
5. Cost Estimation and Budgeting
6. Organizational and Resource Planning
7. Schedule Development
8. Quality Planning
9. Risk Management Planning
10. Project Plan Development and Execution
11. Performance Reporting
12. Planning Change Management
13. Project Rollout Planning

Notes

We now briefly study each of the above steps:

1. ***Project Scope Definition and Scope Planning:*** In this step we document the project work that would facilitate us to attain the project goal. We document the supposition, constraints, business requirements, user expectations, technical requirements, project objectives, project deliverables and everything that describes the final product requirements. This is the foundation for a successful project completion.

2. ***Quality Planning:*** The pertinent quality standards are determined for the project. This is an significant aspect of Project Planning. Based on the inputs captured in the previous steps such as the Project Scope, Requirements, deliverables, etc. a range of factors influencing the quality of the final product are determined. The processes required to deliver the Product as assured and as per the standards are defined.

3. ***Project Activity Definition and Activity Sequencing:*** In this step we describe all the specific activities that must be performed to deliver the product by producing the various product deliverables. The Project Activity sequencing recognizes the interdependence of all the activities defined.

4. ***Time, Effort and Resource Estimation:*** One time the Scope, Activities and Activity interdependence is evidently defined and documented, the next vital step is to decide the effort required to complete each of the activities. The Effort can be calculated using one of the many techniques obtainable such as Function Points, Lines of Code, Complexity of Code, Benchmarks, etc. This step clearly estimates and documents the time, effort and resource required for each activity.

5. ***Risk Factors Identification:*** "Expecting the unexpected and facing it". It is significant to recognize and document the risk factors associated with the project based on the assumptions, constraints, user expectations, specific situation, etc.

6. ***Schedule Development:*** The time plan for the project can be arrived at based on the activities, interdependence and effort required for each of them. The schedule may power the cost estimates, the cost benefit analysis and so on. Project Scheduling is one of the most significant task of Project Planning and also the most complex tasks. In very large projects it is possible that several teams work on developing the project. They may work on it in equivalent. However their work may be mutually dependent. Again various factors may impact in effectively scheduling a project

   - Teams not directly under our control
   - Resources with not enough experience

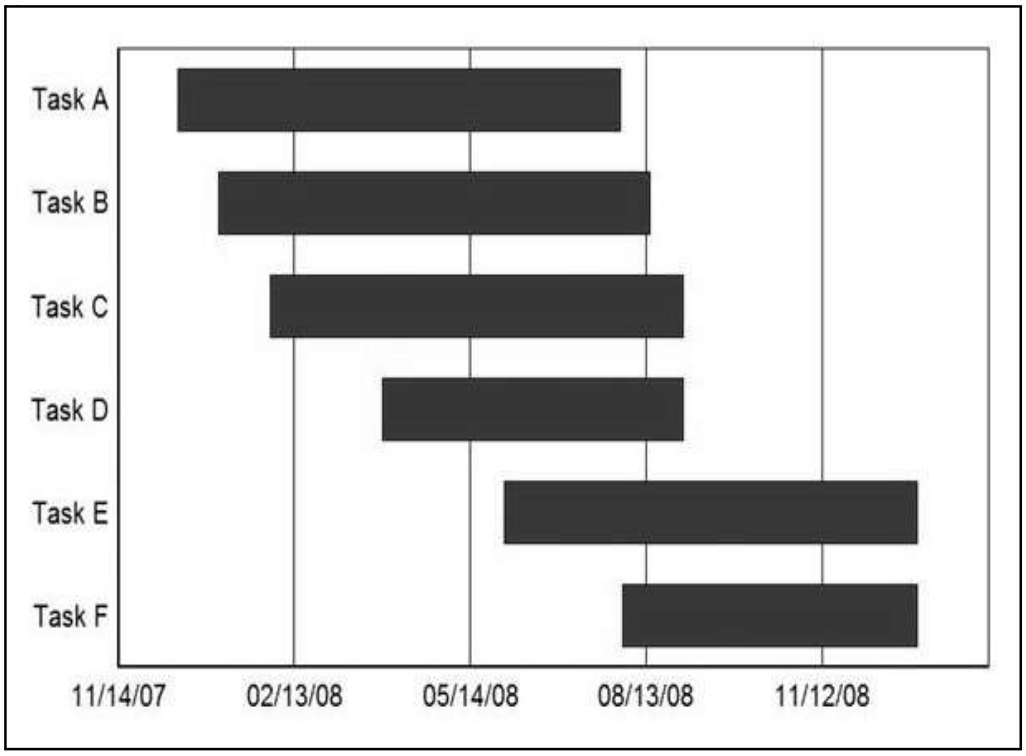
   Popular Tools can be used for creating and reporting the schedules such as Gantt Charts

   Did u know? What is Gantt Charts?

   A Gantt chart is a type of bar chart that demonstrates a project schedule. Gantt charts show the start and finish dates of the terminal elements and outline elements of a project.

7. ***Cost Estimation and Budgeting:*** Based on the information composed in all the previous steps it is possible to estimate the cost concerned in executing and implementing the project. A Cost Benefit Analysis can be inwards for the project. Based on the Cost Estimates Budget allocation is done for the project.

Notes



8. ***Organizational and Resource Planning:*** Based on the actions identified, schedule and budget allocation resource types and resources are acknowledged. One of the primary goals of Resource planning is to ensure that the project is run proficiently. This can only be achieved by keeping all the project resources fully utilized as possible. The success depends on the correctness in predicting the resource demands that will be placed on the project. Resource planning is an iterative process and essential to optimize the use of resources throughout the project life cycle thus making the project execution more competent. There are various types of resources — Equipment, Personnel, Facilities, Money, etc.

9. ***Risk Management Planning:*** Risk Management is a procedure of identifying, analyzing and responding to a risk. Based on the Risk factors Identified a Risk resolution Plan is formed. The plan analysis each of the risk factors and their impact on the project. The probable responses for each of them can be planned. All through the lifetime of the project these risk factors are monitored and acted upon as essential.

10. ***Project Plan Development and Execution:*** Project Plan Development employs the inputs collected from all the other planning processes such as Scope definition, Activity identification, Activity sequencing, Quality Management Planning, etc. A complete Work Break down structure including of all the activities acknowledged is used. The tasks are scheduled based on the inputs captured in the steps formerly described. The Project Plan documents all the assumptions, activities, schedule, timelines and drives the project. Each of the Project tasks and activities are occasionally monitored. The team and the stakeholders are well-versed of the progress. This serves as an outstanding communication mechanism. Any delays are analyzed and the project plan may be adjusted consequently.

11. ***Performance Reporting:*** As explained above the progress of each of the tasks/activities illustrated in the Project plan is monitored. The progress is compared with the schedule and timelines documented in the Project Plan. Different techniques are used to measure and report the project performance such as EVM (Earned Value Management). A broad

Notes

variety of tools can be used to report the performance of the project such as PERT Charts, GANTT charts, Logical Bar Charts, Histograms, Pie Charts, etc.

12. ***Planning Change Management:*** Study of project performance can demand certain aspects of the project be changed. The Requests for Changes need to be analyzed cautiously and its impact on the project should be analyzed. Considering all these aspects the Project Plan may be customized to accommodate this request for Change. Change Management is also essential to accommodate the implementation of the project at present under development in the production environment. When the novel product is implemented in the production environment it should not negatively impact the environment or the presentation of other applications sharing the same hosting environment.

13. ***Project Rollout Planning:*** In Enterprise environments, the achievement of the Project depends a huge deal on the success of its rollout and implementations. Whenever a Project is rolled out it may influence the technical systems, business systems and sometimes even the way business is run. For an application to be effectively implemented not only the technical environment should be ready but the users should accept it and use it efficiently. For this to happen the users may have to be trained on the new system. All this requires planning.

1.3.2 Methodologies

In this section, we emphasize a number of commonly used Software development methodologies. We consider both thick and thin (also called 'lightweight') methodologies. The thick methodologies we consider are RUP, SSADM and PRINCE2. XP, SCRUM and Crystal Clear are measured as thin methodologies. When discussing each methodology, we will focus on the management and business features of the methodology.

Rational Unified Process (RUP)

The Rational Unified Process (RUP) is a software design methodology formed by the Rational Software Company. The Rational Software Company was obtained by IBM in 2003. RUP is a thick methodology; the entire software design process is explained with high detail. RUP is hence particularly applicable on larger software projects. The RUP methodology is common enough to be used out of the box, but the modular nature of RUP-it is designed and documented utilizing Unified Modeling Language (UML)–also makes it easy to adapt the methodology to the particular needs of a single project or company. One of the main differences between RUP and other methodologies like SSADM is that RUP doesn't use a waterfall approach for software development. The phases of requirements, analysis, design, implementation, integration and testing are not done in strict sequence. In RUP, an iterative approach is used: a software product is designed and built in a series of incremental iterations. Figure 1.1 shows one iteration of a RUP project in a graphical way.

Application Area

Due to the modular nature of RUP, it can be employed for all sorts of software projects. It is even possible to use RUP for non-software projects. Though, because of the complexity of the RUP methodology, it is used mostly for larger software projects.

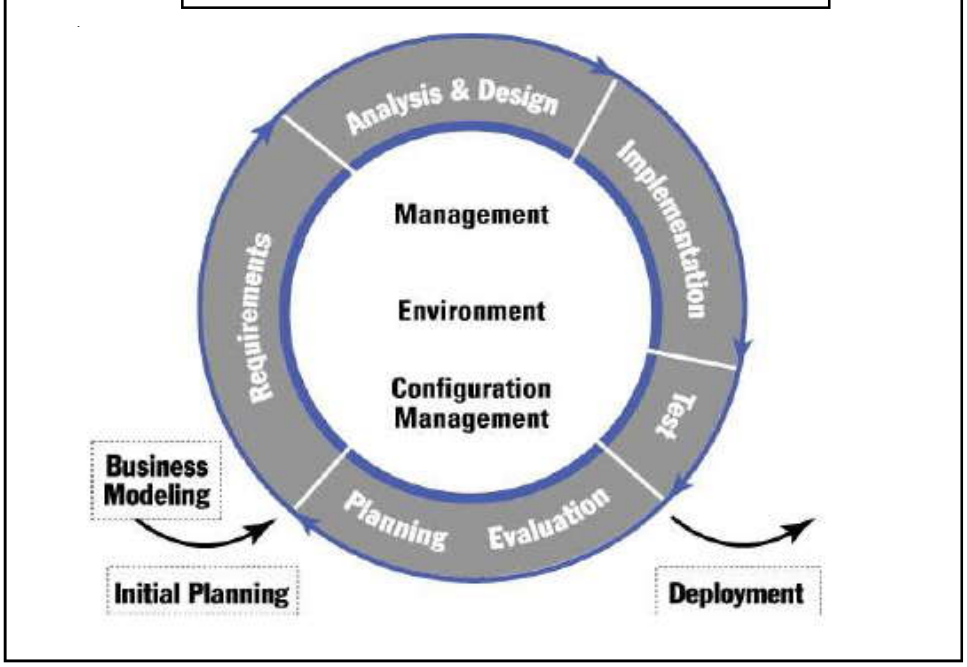
Advantages

- The iterative approach leads to higher effectiveness. Testing takes place in each iteration, not just at the end of the project life cycle. This way, problems are noticed earlier, and are therefore easier and cheaper to determine. When using a waterfall approach, it can happen

Notes

that, for example, software programmers have to wait for the completion of the design phase before starting implementing and integrate the design. Designing and building a software project with an iterative approach resolves this problem. Integration and implementation will not only happen at the end of the project, but in every iteration. This saves time, as more team members can work more of the time.

Figure 1.1: The Iterative Approach to Software Development



- Managing changes in software requirements will be made simpler by using RUP. If a software project is very small, it is nearly unfeasible to define all the software requirements at the beginning of a project. It will almost always take more than one step to know what the final software product will look like, for the customer as well as for the project members. Developing with iterations makes this process of changing, requirements, that often leads to missed schedules and disappointed customers, less troublesome.
- RUP itself is software, too, and is dispersed in an electronic and online form. Team members don't need to leave their computers for RUP related activities. No more searching in big, dusty books. All facts about the software development methodology is available at the project members' fingertips. Also, the newest version of RUP is always present on the computer of each team member. And even more significant, it makes sure that every team member is using the same version of RUP. RUP is designed and documented using UML, in an object oriented way. This makes it easy to adapt RUP to the special needs of a single project or organization.

Disadvantages

- RUP is a commercial product, no open or free standard. Previous to RUP can be used, the RUP has to be bought from IBM, as an electronic software and documentation package. The RUP only exists in an electronic form, which can occasionally limit its use.
- RUP, as said before, explains the whole software design process with high detail; it is a very complex methodology, tricky to comprehend for both project managers and project members. Therefore, it is not the most appropriate software design methodology for most small projects.

Notes

- Starting to use RUP as software growth methodology is difficult. Everyone participating in the project will have to learn working with RUP. Coupling with other methodologies RUP is a thick methodology, and coupling with other methodologies is therefore not always probable. There is a way to couple the Crystal Clear methodology with RUP.

Notes Each iteration comprises some, or most, of the development disciplines (requirements, analysis, design, implementation, testing, and so on).

SSADM

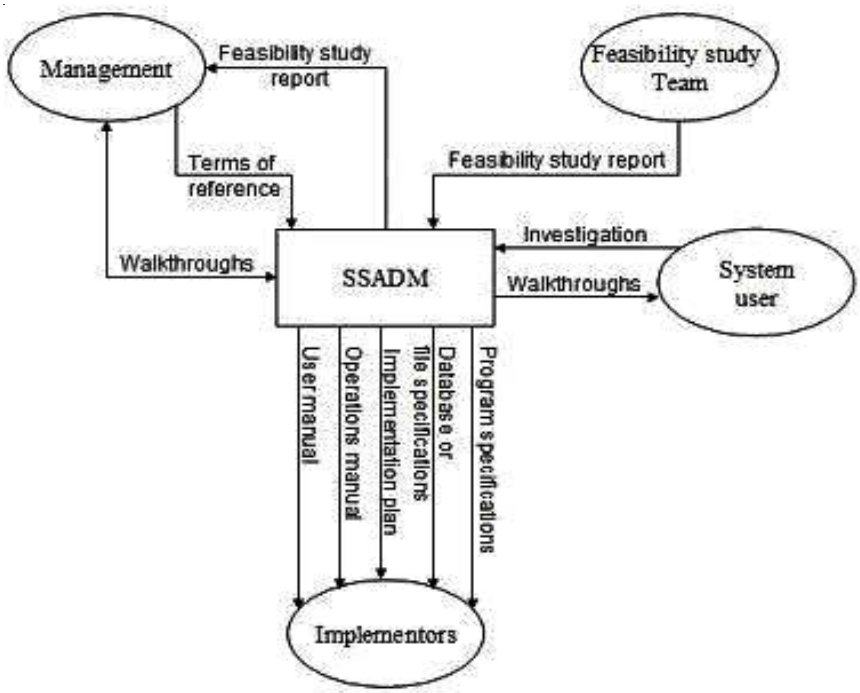
Structured Systems Analysis and Design Methodology (SSADM) is an extensively used computer application development method in the UK. Just like PRINCE, its use is often specified as a requirement for government computing projects. Today, it is more and more being adopted by the public sector in Europe.

> "SSADM has been employed by the government in computing since its launch in 1981. It was commissioned by the Central Computing and Telecommunications Agency (CCTA) in a proposal to normalize the many and varied IT projects being developed across government departments. The CCTA examined a number of approaches before accepting a tender from Learmonth & Burchett Management Systems to develop a method."

Since 1981 SSADM has been further urbanized and refined and in 1990 version 4 of it was launched. SSADM is an open standard, which means that it is freely obtainable for use in industry and many companies offer support, training and Computer Aided Software Engineering (CASE) tools for it.

In detail, SSADM sets out a cascade or waterfall view of systems development, in which there are a series of steps, each of which leads to the next step (see the model in Figure 1.2). SSADM's steps are:

Figure 1.2: SSADM Process Model



Notes

1. ***Feasibility Study:*** The feasibility study comprise of one single stage, which involves conducting a high level analysis of a business area to decide whether a system can cost effectively support the business requirements. In the Feasibility Study an overview Data Flow Diagram (DFD) is produced together with a high level Logical Data Structure (LDS). At this phase the DFD will represent the existing system and the LDS may be incomplete and contain unresolved many-to-many relationships.

2. ***Requirements Analysis:***

   - *Investigation of the current environment:* All through this stage the systems requirements are identified and the present business environment is modeled in terms of the processes carried out and the data structures involved. In this DFDs and LDSs are used to produce detailed logical models of the current system.

   - *Business Systems Options (BSO):* All through this stage up to six business system options are produced and presented. As a result, one of these options is adopted and polished. DFDs and LDSs are produced to support each business system option and the last chosen option. The transition from the former stage to this stage is a key part of SSADM: this is where we move from a logical model of the current system to a logical model of the required system. This means that here the DFDs and LDSs have to be refined to cater to new or changed requirements.

3. ***Requirements Specification:*** The Requirements Specification comprises of a single stage which involves further developing the work carried out in the Requirements Analysis: detailed functional and non-functional requirements are recognized and new techniques are introduced to define the required processing and data structures.

4. ***Logical System Specification:***

   - *Technical system options:* In this phase up to six technical options (specifying the development and implementation environments) are formed, one being selected.

   - *Logical design:* In this phase the logical design of update and enquiry processing and system dialogues (menus, etc.) is carried out.

5. ***Physical Design:*** The Physical Design contains of a single stage in which the logical system specification and technical system specification is employed to create a physical database design and a set of program specifications.

SSADM revolves around the use of three key techniques:

1. ***Logical Data Modelling (LDM):*** This is the procedure of identifying, modelling and documenting the data requirements of a business information system. A LDM consists of a LDS and the associated documentation. LDSs represent Entities (things about which a business needs to record information) and Relationships (necessary associations between entities).

2. ***Data Flow Modelling (DFM):*** This is the procedure of identifying, modeling and documenting how data flows around a business information system. A Data Flow Model consists of a set of integrated DFDs supported by suitable documentation. DFDs represent processes (activities which transform data from one form to another), data stores (holding areas for data), external entities (things which send data into a system or receive data from a system) and finally data flows (routes by which data can flow).

3. ***Entity/Event Modelling (EM):*** This is the procedure of identifying, modeling and documenting the business events which influence each entity and the sequence in which these events occur. An EM consists of a set of Entity Life Histories (ELHs) (one for each entity) and appropriate supporting documentation.

Notes

Application Area

SSADM was initially developed to standardize the many and varied IT projects being developed across government departments. Nowadays, SSADM Version 4 can be used in all kinds of analysis and design stages of system development. SSADM can be used for practically any size of project: small (1–2 persons, less than one man year), medium (4–10 persons, 1–20 man years) and large projects. In addition SSADM can be used to develop new projects, but it can also be used to maintain existing systems.

Advantages

- As stated before SSADM is an open standard, which means that it is freely available for use in industry and many companies offer support, training and CASE tools for it.
- SSADM separates an application development project into modules, stages, steps, and tasks, and offers a framework for describing projects in a fashion suited to managing the project.
- SSADM's objectives are to:
  - Advance project management and control
  - Make more effectual use of experienced and inexperienced development staff
  - Develop improved quality systems
  - Make projects flexible to the loss of staff
  - Allow projects to be supported by computer based tools such as computer aided software engineering systems
  - Create a framework for good communications between participants in a project.
- SSADM can decrease the chances of initial requirements being misunderstood and of the systems functionality straying from the requirements through the use of inadequate analysis and design techniques.

Disadvantages

- SSADM is a typical instance of a structured methodology, which means that the purpose of it is to:
  - Formalize the requirements elicitation process to reduce the chances of misunderstanding the requirements.
  - Introduce best practice procedure to the analysis and design process.
- As stated before SSADM can reduce the chances of initial requirements being misunderstood and of the systems functionality straying from the requirements through the use of inadequate analysis and design techniques. Though, SSADM assumes that the requirements (in the form of an agreed requirements specification) will not change during the development of a project. Following each step of SSADM thoroughly can be time consuming and there may be a considerable delay between inception and delivery (which is typically the first time the users see a working system). The longer the development time the more chance of the system meeting the requirements specification but not satisfying the business requirements at the time of delivery.

Notes

Task Analyze the application area of SSADM.

PRINCE2

Projects in Controlled Environments (PRINCE) are a project management technique covering the organization, management and control of projects. A project has a apparent beginning, middle and end, a understandable organizational structure and defined objectives. You can use a managing methodology like PRINCE to ensure that a project is successful, which means that it finishes on time, within budget and offers the customer with what they have asked for. PRINCE was first urbanized by the CCTA, which is now part of the Office of Government Commerce (OGC), in 1989 as a UK Government standard for IT project management. Because its introduction, PRINCE has become widely used in both the public and private sectors and is now the de facto standard for project management in the UK. Although PRINCE was originally developed for the needs of IT projects, the methodology has also been used on many non-IT projects. The latest version of the methodology, PRINCE2, is designed to include the requirements of existing users and to enhance the methodology towards a generic, best practice approach for the management of all types of projects.

PRINCE2 is a process-based approach for project management providing a simply tailored and scalable methodology for the management of all types of projects. Each process is described with its key inputs and outputs together with the specific objectives to be achieved and activities to be carried out. The methodology describes how a project is divided into manageable stages enabling efficient control of resources and usual progress monitoring throughout the project. The various roles and responsibilities for managing a project are completely described and are adaptable to suit the size and complexity of the project, and the skills of the organization.

PRINCE2 summarize eight processes that are required to effectively carry out a project. These are:

1. ***Starting up a Project:*** To be confident about the project has a very clear beginning, this process occurs even before the project has really started. All decision making persons have to come together and will appoint a Project Manager. Together they will discuss the project and outline reasons for it and how decide how the project is to be carried out. All this information will be put together in a 'Project Brief'.

2. ***Initiating a Project:*** Before a project can be permitted during the 'Directing a Project' process it must be carefully planned to ensure that it meets its objectives. Detailed estimations of costs, needed time and other resources have to be made and these are put together by the Project Manager into a so called Project Initiation Document (PID) for approval by the Project Board (PB).

3. ***Directing a Project:*** After the Project Brief and the PID have been put together, the project has to be permitted by a group of senior managers, called the Project Board (PB). During the rest of the project this PB has the in general responsibility for the success of the project whereas the Project Manager has the day to day accountability. He will inform the PB about the project's progress with the help of regular reports.

4. ***Controlling a Stage:*** One of the advantages of PRINCE2 is that projects are separated into manageable stages to make sure the project remains manageable and controlled. How many stages are used, will depend on the size of the project and the level of risk. In PRINCE2 each project stage must be completed before the next stage can be started and each new stage is planned in the stage preceding it. Also the Stage Plans will be approved by the PB to help ensure that the project remains within budget and delivers its objectives.

Notes

5. ***Managing Stage Boundaries:*** This process engages preparing for the next stage and reviewing the current stage. The Project Manager makes suggestions to the PB about the likelihood of the project achieving its business objectives and any changes in the business case, project plan, risks and issues. When a project has clear stage boundaries it can be simply controlled and managed by permitting the project to continue only once the PB is satisfied with the current stage end and next stage plan.

6. ***Planning:*** Each project plan, stage plan and team plan must believe key planning aspects. These include what products to produce, the activities required to produce these products, estimated resources (including costs and time), scheduling the activities and analyzing risks. By following the PRINCE2 planning process all these points are conducted in a sensible, logical sequence. Ensuring consistency enables plans to be compared and streamlines the planning process.

7. ***Managing Product Delivery:*** The objective of a PRINCE2 project is to deliver products. A product can be a physical thing such as a poster or it could be an intangible deliverable such as a service or sales agreement. In fact everything produced in PRINCE2 (even a document) is called a product. Often a Project Manager does not create the product. A third party supplier and/or their colleagues may do some or all of the work. It is the Project Manager's responsibility to ensure that the supplier produces the correct products at the right time by providing a description of the work to be done.

8. ***Closing a Project:*** At the end of the project, after its products have been delivered, the project is closed down with approval of the PB. The Project Manager plans what will be done to assess the project's outcome, which is called the Post Project Review (PPR). A controlled close down is in effect the last provable PRINCE2 project action. Any lessons learned are recorded, resources are released and the Post Project Review Plan (PPRP) is created.

The key concepts that are fundamental to PRINCE2 are:

- ***Control:*** Being able to control your project is key to its achievement. For this reason PRINCE2 breaks down projects into easily managed stages, essentially breaking a large project into 'bite size chunks'.
- ***Quality:*** To ensure that a product (or service) meets the customer's quality prospect these must be defined and agreed when a project is being planned.
- ***Planning:*** In PRINCE2 planning does not end once the project has started. Of the eight PRINCE2 processes all but one engage planning, even the final process.
- ***Lesson Learned:*** Every time we carry out a project we learn something. All lessons, mistakes, ideas or successes are detained in the Lessons Learned Log. At the end of the project these are collated into a Lessons Learned Report, allowing others in the organization to bene fit from them.

Application Area

PRINCE2 is a project management methodology owned and preserved by the OGC in the UK. It summarizes best practice from a variety of industries and backgrounds. PRINCE2 has been adopted by the National Health Service as its favored methodology and a number of governments' worldwide are looking at adopting it as their standard project management methodology. PRINCE2 is one of the few government standards that have grown organically to be adopted by both private and public organizations. Some organizations that use PRINCE2 as a project management methodology are the UK Police Forces, Rolls Royce, the British Medical Association, Norwich Union, the UK Department of Justice and London Underground.

Notes

Advantages

Besides the key idea of PRINCE2, there are some other advantages of the use of PRINCE2:

- PRINCE2 is a structured methodology as long as organizations with a standard approach to the management of projects. The methodology embodies proven and established best-practice in project management. It is widely recognized and unstated, and so provides a common language for all participants in the project, also PRINCE2 is very useful for educative use.
- PRINCE2 enables projects to have:
  - A controlled and planned start, middle and end
  - Usual reviews of progress against plan and against the Business Case
  - Flexible decision points
  - Automatic management control of any deviations from the plan
  - The involvement of management and stakeholders at the right time and place during the project
  - Good communication channels between the project, project management, and the rest of the organization.
- Since there are no chapters on test methods in the PRINCE2 handbook, the choice which test method you want to use, is totally without restrictions. PRINCE2 gives you a free choice of test method, but does ask attention for it when putting together the project quality plan, which is part of the PID.

Disadvantages

- Every person who works on a PRINCE2 project should be quite well-known with every aspect of PRINCE2 to know how to play the game. It often happens that this is not the case, because it is very expensive to give everyone concerned a course to study PRINCE2.
- Using PRINCE2 means that a lot of documents and lists have to be written, and because Project Managers only have to inform the PB about the status of the project, when something goes wrong they can simply blame others. It is also very easy to blame other project groups when something goes wrong. This leads to the so called "cover your ass behavior".
- Splitting up a PRINCE2 project often results in a lack of knowledge of the project by accountable persons like the Project Manager. Also it's not useful to make use of expensive Project Managers when the only work they have to do is to administrate and inform the PB.

Did u know? What is Project Initiation Document (PID)?

The Project Initiation Documentation bundles the information which was acquired through the Starting up a Project and Initiating a Project processes in a PRINCE2 controlled project environment. Since PRINCE2's 2009 revision it explicitly renamed the Document part of the initial definition into Documentation, which means that the methodology not so much seeks a master document describing all of the project at hand, but more the collection of documents until Authorizing a Project up to then created.

Notes

eXtreme Programming (XP)

XP is a software engineering technique that has been formulated in 1996 by Kent Beck. XP has received fair media attention, and is most renowned for its practices that are sometimes regarded as contentious, such as pair programming and test-driven development. In this document, we will not concern ourselves with these aspects of eXtreme Programming, but instead we will focus on the management part.

Principles of XP

XP aims to decrease the risk involved in software development. In particular, it aims to reduce the cost of delaying design decisions. In XP, Beck gives a treatment of the cost and revenues of design decisions and aspect implements (which he calls 'options'), and he concludes that it is more beneficial to delay options of which it is uncertain whether they will generate revenue (i.e. there is a certain amount of risk involved in implementing the option).

Usually, the cost of making decisions about (and therefore changes to) a software project would rise exponentially during the course of development. It would therefore be costly to defer options, because implementing them later on might be too costly, and perhaps even cost more than the value of the option would be.

XP decrease the cost of making modifications later on during development, and thereby allows decisions that entail high risk to be deferred until a sound judgment can be made on them.

Planning

An XP project is made up of releases. The first release aspires to produce an initial, working version of the product. The succeeding releases add functionality to the project, change behavior and fix bugs. An XP project typically lasts the entire lifetime of the application: the software is constantly tweaked and updated to be as useful as probable. Of course, this approach is not required: the project can be ended when the customer decides the product is 'finished'. There is typically one to three months time between releases. Each release is divided up iterations. Generally, they are one to three weeks in length. In an XP project, requirements are not fixed in advance. At the start of the project, or whenever he can think of one, the customer writes down a desired aspect in a so{called user story, which clarifies the feature by means of a typical 'use case'.

At the start of the project, a release arrangement is drawn up. First, all stories are written by the customer. The growth team then assigns a cost to each story. This cost should be one, two or three 'ideal programming weeks' for a single developer. If the cost is better, the story should be split up. If the cost is less, multiple stories should be merged together. The stories are then divided over a number of releases. Release dates can then be calculated from the stories assigned to each release, or the stories can be divided such that fixed release dates will be met. For this, you will need to estimate how to convert ideal programming weeks to calendar weeks. The first time, this will be hard (and estimates might be off), but as the project progresses the estimates will become better.

At the start of each iteration, the customer chooses the stories from that release that are of the greatest value to him, which will be implemented during the iteration. The stories are then broken up into smaller units called tasks. Each developer has the opportunity to assume accountability for a number of tasks. The tasks are then estimated, in ideal programming days, by the developer that chose them (making sure no-one has too much or too little to do), and implemented during the course of the iteration. Experience from an iteration or a release, such as the perfect programming time realized, can be taken into account to estimate better next release or iteration planning.

Notes

Application Area

XP is a frivolous methodology for small-to-medium-sized teams developing software in the face of vague or rapidly changing requirements. XP is a good choice when requirements are unclear (which might occur when because himself does not know precisely what he wants), or prone to change (because of changing business situations, or as a result of external conditions). Because in XP the development of a product is divided into many small cycles, and each cycle is planned separately, changes to the planning can be made constantly, rapidly and easily. Team size is an issue when implementing XP. XP is meant for small-to-medium sized teams. In practice, this means that teams should be maximum ten people. A few more is probably okay, but twenty is too many.

Advantages

- An XP project is very impressionable. A usable product can be released very quickly, at which point the business can already take advantage of the product, and the product can and will be improved continually after that, with feedback that stems from live use. Especially when the project is exploratory for the customer as well, having feedback from live use and adapting to changing minds, wishes and circumstances can be invaluable.
- Additionally, the process is very clear. Progress, position and direction of the project are very transparent, which will make management happy as well.

Disadvantages

- The main roadblock to implementing XP in any given environment will usually be the customer. The 'customer' or a person that plays the role of the customer, has to be an integral part of the development team. This means that the customer will have to be obtainable on-site at all times. Sometimes, this is just not feasible, or the customer will refuse to assign an employee to the development team full-time. In such cases, XP will not be able to work properly, and should be abandoned.

Task What do you understand from the statement "All practices of XP work jointly to achieve this goal"?

Scrum

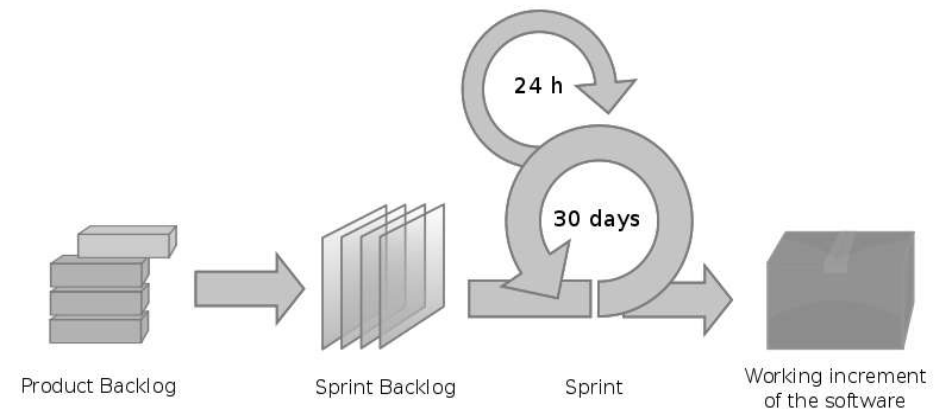
Scrum is an agile technique for project management, in use since at least 1990. It has been called a "hyper-productivity tool", and has been documented to considerably improve productivity in teams previously paralyzed by heavier methodologies-quickly producing results where there had been little or none.

Scrum uses the following concepts:

- ***Sprint:*** A period of 30 days or less where a set of work will be execute to create a deliverable.
- ***Backlog:*** All work to be performed in the predictable future, both well defined and requiring further definition.
- ***Sprint backlog:*** The work that should be done through the current sprint.
- ***Product backlog:*** The work that should be done for the entire product as desired by the customer.
- ***Scrum:*** A daily meeting at which progress and obstruction to progress are reviewed.

Notes

Figure 1.3: The Scrum Work Flow



Scrum is an iterative, incremental process for developing a product. The dissimilar iterations are shown in Figure 1.3.

Scrum works by first important a backlog of things that need to be done; this list is usually maintained by one person. Other interested parties can request things to be put on a backlog. For each sprint a subset of the backlog is chosen to be done. All through the sprint the team will only work on the things that are in the sprint's backlog to keep people focused and creative. Each day of a sprint there is a Scrum in which the team members respond to the following questions:

- What did you do since the last Scrum meeting?
- Do you have any obstacles?
- What will you do before the next meeting?

The sprint is lead by the so called Scrum Master. It's his job to eliminate all the obstacles that the team has encountered as soon as possible. This ensures that the team itself can stay focused on the task itself. A Scrum ensures that the team as a entire stays in touch with all parts of the sprint. During a Scrum, management and the customer may be present but only the Scrum Master and the team members are allowed to talk. This is to be sure that the Scrum is short and focus on the task at hand. After each sprint there is usually a demonstration of what has been done. Before the start of a new sprint a discussion with the team and management is held to establish a new sprint backlog based on the result of the last sprint and changes to the environment/requirements.

Scrum's main focus points are on team empowerment and flexibility. During sprint a team itself is responsible for doing the given work. The only interaction with management is to tell them what's getting in their way and what needs to be removed to get better productivity. Because after each sprint, the next increment can be changed according to the accomplishments and the changes in the environment, the project is very flexible and adaptable.

Application Area

Tentatively Scrum can always be applied when a group of people should work together to archive a common goal. It has even been used as a project management approach, in a so called "Scrum of Scrums". Of course to work correctly the teams should be small, but this can be solved by dividing projects into sub-teams. In the ideal situation all team members should be at the same location for optimal communications among the members. But when this is not the case the Scrum meetings and is held as a teleconference.

Notes

Advantages

- During a Scrum sprint there are no turbulence from the outside; this keeps the team focused and creative. Which is very good for the productivity?
- At the end of each sprint, what has been done and what should be done in the next sprint can be evaluated. This keeps the process very supple.

Disadvantages

Because a Scrum team should be responsible for itself during a sprint, it's important that management doesn't interfere with how the work is being done.

Caution The management needs to fully trust the team to do the right thing, which could potentially be problematic.

Crystal Clear

Crystal Clear is a highly optimized way to use a small, collocated team, prioritizing for safety in delivering a satisfactory outcome, efficiency in development, and habitability of the working conventions." The Crystal Clear methodology is part of the Crystal family of methodologies, where every methodology is characterized by a color (Clear, Yellow, Orange, Red, Maroon, Blue, Violet). That color represents the number of people for which the methodology is suited; Crystal Clear is the lightest color and is meant for the smallest project groups, of two to eight people. Darker colors are for larger groups-these will not be discussed here.

Crystal Clear has at its core seven properties that should be established for every project that wishes to adhere to the methodology. While all of these are desired, only the first three are mandatory; the other four will get the project further into the safety zone. The seven properties are:

1. ***Frequent Delivery:*** When delivering working, tested code to the actual software users once every few months (or more often, if possible), users will be able to deliver feedback on implemented requirements, sponsors will see progress and developers will get a morale boost.
2. ***Reflective Improvement:*** Taking time to let the team reflect on what works and what doesn't work for the project, and improving the things that don't work.
3. ***Osmotic Communication:*** Having the entire team so close together (if possible in the same room, otherwise in adjacent rooms) that people don't have to go to a lot of trouble to raise or answer questions, but can do so instantly, will make people work together naturally, inspect each others' work and pick up relevant information as if by osmosis.
4. ***Personal Safety:*** If people feel safe to speak up without fear of reprisal, they can give constructive criticism on other people's work and admit their own mistakes, leading to honesty and ultimately to trust.
5. ***Focus:*** If everybody has time to focus on their primary objectives for two hours a day, for two consecutive days every week, without any distractions that can make them lose their train of thought (like meetings or other work), people will be more focused and work will be finished quicker.
6. ***Easy Access to Expert Users:*** If expert users are available to the team, they can answer questions and deliver feedback on quality and design decisions.

Notes

7. ***Technical Environment with Automated Tests, Configuration Management & Frequent Integration:*** A proper technical environment where testing and configuration management/version control tasks (like making backups and merging changes) do not have to be done by hand will make life easier for developers.

Crystal Clear offers several concrete procedures/techniques that can help establish these critical properties, but these are optional: If the team knows of other ways to satisfy the properties, there is nothing that stands in their way. In general, it can be said that Crystal Clear values properties over techniques. This also makes Crystal Clear a low threshold methodology: project groups can carry over their established methods and techniques-which the group has either grown into or were developed to fit their specific situation-to Crystal Clear, and thus will not have to learn a set of new ones before coming up to speed.

Application Area

As explained above, Crystal Clear is meant for project groups consisting of two to eight people working at the same physical location, with one or more expert users available. In general, this means any setting where the first three (but ideally, all seven) of the properties can be fulfilled are applicable.

Though, the above does not have to be strictly remained to. All methodologies in the Crystal family support the stretch to fit principle, which states that when a prospective project does not fit within the aimed methodology, the principles and practices to be carried out by the methodology can be stretched to fit the particular case.

Example: Teams that is significantly larger than eight people have carried out Crystal Clear successfully by stretching it to fit their needs.

Advantages

- Since the seven properties are based upon performance that has been observed in successful project groups, those practicing Crystal Clear might well be on the right track to bringing the project to an end effectively. While this is of course no assurance of success-there are always other factors that contribute to or detract from a project's success-it is likely that these properties contain at least some quality that does indeed make the disparity between a successful project and an unsuccessful one. In fact, [CC-BOOK] cites numerous stories that demonstrate the achievement of the properties and techniques provided as well as Crystal Clear in general.
- Unlike conventional, thick methodologies like SSADM or PRINCE, Crystal Clear is flexible as to what project teams are supposed to do and how to do it. This is expressed in the properties over procedure and stretch to fit principles. In fact, Crystal Clear was clearly designed to be usable by as many project groups as probable, with the least number of new techniques to learn.

Disadvantages

One of Crystal Clear's major strengths is also its main disadvantage. It tries to be a methodology that is applicable in as many cases as probable. This obviously prevents it from ever being a "best" methodology (like XP strives to be) in any specific case.

Another drawback might be that Crystal Clear is still comparatively new: the only book written on the subject is not yet published, so it may not have a lot of real-world usage yet. On the other hand, the main beliefs behind the methodology are all based on real experiences drawn from real projects, so perhaps wider experience will reveal that Crystal Clear indeed works"as advertised".

Notes

1.3.3 Project Methods/ Techniques

This section confers a number of project management techniques. These procedures can be employed as an aid to estimate, track and evaluate different features of the project. We start with a discussion of PMBOK, which is really not a technique in itself, but rather a collection of industry-standard techniques. After that, we discuss COCOMO, MTA, EV and Critical Path.

Project Management Body of Knowledge (PMBOK)

The Project Management Body of Knowledge is a broad term that explains the sum of knowledge within the profession of Project Management (PM). As with additional professions such as law, medicine, and accounting, the body of knowledge rests with the practitioners and academics that apply and advance it. The full Project Management Body of Knowledge (PMBOK) comprise knowledge of proven traditional practices that are extensively applied, as well as knowledge of innovative and advanced practices that have seen more limited use. The PMBOK framework splits the project processes into five discrete process groups: initiating, planning, executing, controlling and closing.

Notes Note that these groups do not imply that the project has to go through each one in this order; they are only provided in order to be able to structure and categorize the different project processes.

PMBOK also recognize several project knowledge areas: integration management, scope management, time management, cost management, quality management, human resource management, communications management, risk management and procurement management. By using this twin categorization in process groups and knowledge areas, we can categorize project processes. The PMBOK Guide comprises summaries of generally accepted techniques and methodologies that can be used to implement these project processes.

Application Area

PMBOK tries to imitate the growth of knowledge and practices in the field of project management by capturing those practices, tools, techniques and other relevant items that have become generally accepted.

Usually accepted does not mean that the knowledge and practices described in the PMBOK framework are or should be applied uniformly on all projects; the project management team is always answerable for determining what is appropriate for any given project. A few well-known procedures included in the PMBOK framework are Earned Value (EV) management, Program Evaluation and Review Technique (PERT) [PMBOK-PERT] and Critical Path Method (CPM).

Advantages

- PMBOK provides a general project management framework in the form of process groups and knowledge areas.
- PMBOK proposes a unified project management terminology.
- PMBOK gives a concise summary of and reference to generally accepted project management principles.

Notes

Disadvantages

- PMBOK is only a framework; the definite needs of the project in question should be determined by a knowledgeable managerial team.
- PMBOK provides nominal coverage of various project management methodologies and techniques. One definitely needs to consult specialized texts on these subjects in order to learn the ins and outs.
- PMBOK only incorporates those features of the project management process that are profession independent.

Notes PMBOK actually is a collection of usually accepted project management techniques, these techniques can easily be integrated in other methodologies when applicable.

Constructive Cost Model (COCOMO)

COCOMO is an experiential, algorithmic model for estimating the effort, schedule and costs of a software project. It was derived by collecting relevant data from a large number of software projects, then analyzing the data to discover the formulae that were the best-fit to the observations.

The first version of the COCOMO model (now known as COCOMO 81) was a three-level model where the levels imitated the detail of the analysis of the cost estimate. The first level (basic) provided an initial, rough estimate; the second level modified this using a number of project and process multipliers and the most detailed level produced estimates for different phases of the project.

COCOMO 81 makes a variety of assumptions about the software development process in order to produce its estimates. The latter will only be somewhat accurate when the project uses the waterfall process model and every line of code is produced from scratch. It also fails to take into account that nowadays higher-level programming languages are employed, supported by various automated tools. We will not elaborate on this version, since it has been obsolete by COCOMO 2.

COCOMO 2 comprises support for various development methodologies such as component-based development and prototyping, fourth generation programming languages and CASE support tools. COCOMO 2 still consists of three levels, but these have been given somewhat different interpretations:

- ***The early prototyping level:*** Size estimates are based on object points. These object points are a simple way of quantifying the perceived complexity of requirements that need to be implemented. The required effort is then computed by applying a simple extrapolation from the object points and programmer productivity. Object points are based on the number of screens, reports and modules in third generation programming languages, and can be weighed by the perceived complexity of the screen, report or module in question.
- ***The post-architecture level:*** Once the system architecture has been designed a reasonably accurate estimate of the software size can be made. The estimate as this level uses a more extensive set of multipliers reflecting personnel capabilities, product and project characteristics.
- ***The early design level:*** This level corresponds to the completion of the system requirements with (perhaps) some initial design. Estimates are based on function points, which are obtained by working out the object points in detail. More specifically, the total number of points is computed by measuring or estimating the following program features: external

Notes

inputs and outputs, user interactions, external interfaces and files used by the system. The function points are then converted to number of lines of source code using the tables provided by the COCOMO model.

Application Area

COCOMO is a well-known experiential algorithmic cost estimation procedure. It is well-documented, in the public domain and is supported by public domain and commercial tools. It has been extensively used and has a long pedigree from its first instantiation in 1981. The application of the first instantiation of the model was limited due to the quite large constraints on the development process. This issue has been mitigated by continued improvements on, and extensions of the model, resulting in COCOMO 2. A refinement of the model for the Ada programming language is available as well.

Advantages

- Though it's hard to pinpoint the exact cost of any given project, one can still obtain usable data by calculating optimistic and pessimistic estimates.
- Implementation and execution of the model is very easy and proficient. As a result, it is supported by public as well as commercial tools.
- COCOMO is a well-known and well-documented method.

Disadvantages

- It is quite difficult to come up with acceptable estimates for the size of a project when the latter still in an early stage of development.
- The use of the number of lines of source code as a measure of difficulty is highly disputable. Even though COCOMO tries to take this into account by providing different tables for all major programming languages, there are still lots of discrepancy such as: expressivity differences between programmers, usage of subroutines, general code reuse, and etcetera.
- Some input parameters in the model cannot by determined quantitatively; they need be estimated as well. A few examples: experience and productivity of the programmers, maturity and capability of CASE tools. The accuracy of the ultimate estimates of the COCOMO model depends considerably on the exactness of the initial ones.
- The COCOMO model has not been revised since 1995. Consequently, it is likely the model fails to take into account new theories and practices in the Software Engineering field, resulting in worse estimates.

Usage in Methodologies

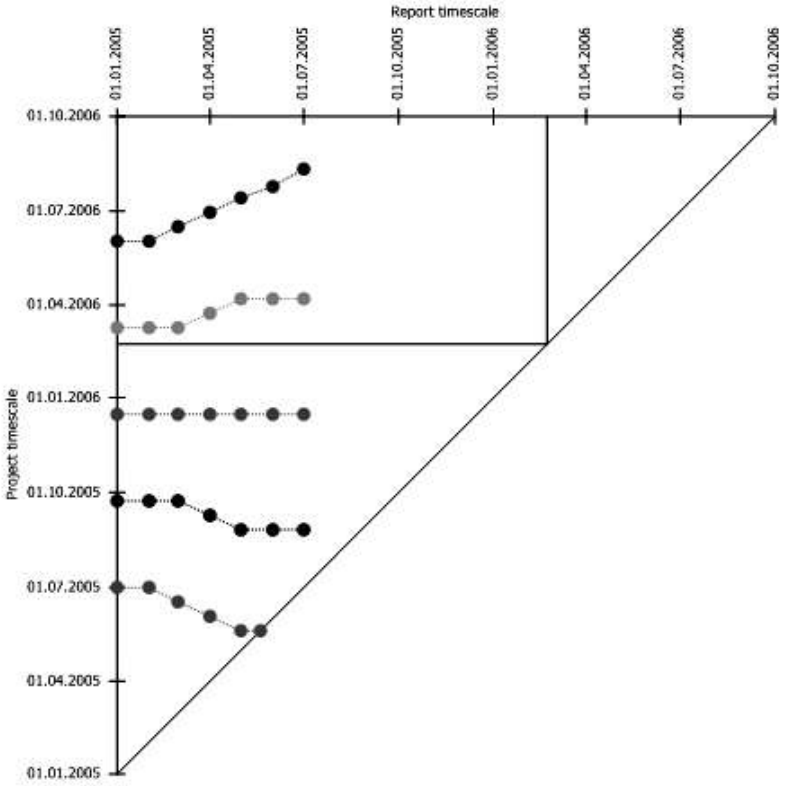
As confirmed above, the COCOMO model can only be applied when the project in question satisfies a given number of criteria. Moreover, it is advisable to try out other estimation techniques, as to get a feeling of the accuracy of the estimates that have been obtained. Other probable techniques include:

- Expert judgment
- Estimation by analogy
- Other algorithmic cost estimation models

Notes

Each of these methods has advantages and disadvantages, and none is appropriate in all circumstances, since cost estimation of software engineering projects is a very difficult task due to the highly dynamic character of the profession.

Figure 1.4: MTA Chart [MTA-SAP]



Tools

Implementing the COCOMO model comes down to assessing some easy mathematical formulae wherein the variables should be chosen so as to match the characteristics of the project under scrutiny as strongly as possible. As such, one can easily find implementations on the Internet by using a decent search engine such as Google.

Did u know? What is Ada programming language?

Ada is a structured, statically typed, imperative, wide-spectrum, and object-oriented high-level computer programming language, extended from Pascal and other languages. It has strong built-in language support for explicit concurrency, offering tasks, synchronous message passing (via guarded task entries), protected objects (a monitor-like construct with additional guards as in conditional critical regions) and nondeterminism (via select statements)

Milestone Trend Analysis (MTA)

MTA is a software engineering method for evaluating the actual progress of a project in relation to its planning. This comparatively simple technique consists of recording the dates of the

Notes

milestone deadlines at the times they are changed, i.e. when they are delayed or advanced. This way one gets a matrix of data: the columns of the matrix delimit the project milestones, the rows the dates on which the deadlines were reevaluated, as an actual cell contains the new deadline estimate for the milestone in question. Of course, one can really enhance insight in these data by using some simple visualization techniques. This can be done by plotting the estimated deadlines against the dates on which they were evaluated. The latter are generally placed on the X-axis, the former on the Y-axis. The evolution of a project milestone deadline is thus observable as a curve on the graph: downward movement of the curve signifies that the deadline in question was advanced, while upward movement means postponement. One can also easily spot milestone completion: this is the case when the curve intersects the line y = x. The general shape of the graph is often approximately triangular: this is the result of the fact that we stop plotting a curve when the milestone in question has reached completion, i.e. when it intersects with the angular bisector of the first quadrant.

An example of a typical MTA chart can be seen in figure 1.4.

Application Area

MTA can be applied to each project that utilizes milestones as the major indicators of progress. It is in essence a very simple and graceful technique that can easily be applied to assess progress. Of course, MTA is an evaluation technique that is to be used during the execution of a project. Its main uses are avoiding and correcting schedule slippage, and post-mortem schedule evaluation.

Advantages

- MTA is a simple, elegant and effectual technique.
- MTA is extensively used and supported.
- MTA has a big application area.

Disadvantages

- MTA in itself does not keep track of inter-package addiction. Therefore, when a certain milestone completion date is altered, one need to make sure its dependencies is altered as well. This does not prove to be much of a problem in practice though, since MTA is available as a plug-in for more complete project management tools that can keep track of dependencies.
- The inputs of the MTA method are of course estimates of milestone completion targets. As such, it is essential these estimates are made by knowledgeable and experienced engineers. MTA will not be of much use if these estimates are not sensibly accurate.

Usage in Methodologies

As declared above, the only prerequisite is that the project under study uses milestones. MTA does not impose any further restrictions on the process model and can help to explain progress assessment in almost any project.

Tools

An MTA plugin for the well-known Microsoft Project management tool is obtainable.

Notes

Earned Value (EV) Management

In earned value management the development of a project is estimated by comparing what already has been done with the estimates that were made at the beginning of a project. By extrapolating these measurements, a project manager can judge how much resources will be used at the end of a project. Some ordinary acronyms that are used use in the EV management:

BCWS Budgeted Cost for Work Scheduled

BCWP Budgeted Cost for Work Performed

ACWP Actual Cost of Work Performed

BAC Budget at Completion

EAC Estimate at Completion

Budgeted Cost for Work Performed (BCWP) is also identified as the earned value. This value shows what a project actually has earned at a certain point in time. The cost of an amount of work can be expressed in diverse ways.

Example: In dollars or in hours.

Also one has to choose when something has been earned. It can be chosen to only set something to be earned when the full task is done. Or say that the part of the task that by now had been done has been earned. In the last case the problem is that estimating how far a task has progressed is complex.

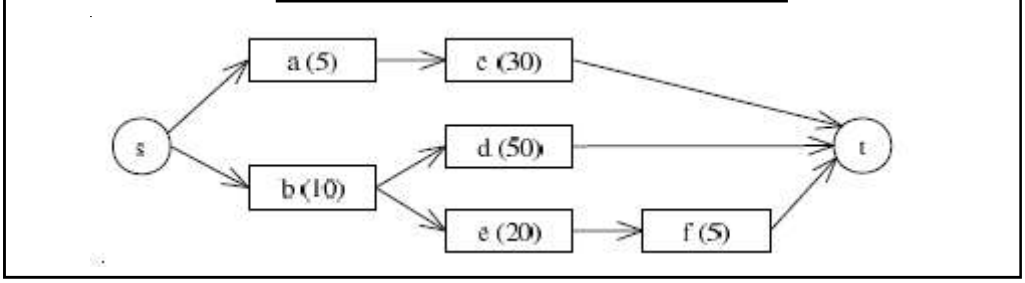
Advantages

In a project a manager can judge if the projects are on schedule/budget. If that's not the case an estimate can be made how far the project is over budget.

Disadvantages

It's very tricky to estimate the real Earned Value at a certain point in time. Wrong estimates of this value can make a project look like it's doing a lot better then it really is (or the other way around).

Figure 1.6: An Example of a Project Network



Usage in Methodologies

EV management can be utilized to monitor projects where there is planning beforehand when assured goals should be reached. This encompasses most thick methods.

Notes

Examples: PRINCE2 and SSADM.

Tools

Insight is a tool for earned value management that incorporates with a broad range of project management tools.

Critical Path

The critical path method operates on a directed acyclic graph that successively orders all tasks that need to be completed in the project. We term this graph the project network. An instance of a project network can be seen in Figure 1.5. The tasks connected in a project network are classically the terminal elements of a Work Breakdown Structure. The graph states the order in which the different tasks need to be completed, and the dependencies between them. Each task has an associated cost in time. The critical path is the longest path from the start of the project to the finish, and its cost is the shortest period in which the project can be completed. Any delay on tasks on the critical path will delay the entire project. In our example, the critical path is (s; b; d; t), with a cost of 60 days. A related concept is slack; this is the time that a single activity can be delayed, without delaying the project. By description, the slack of all activities on the critical path is 0.

Application Area

Critical Path can be employed for task scheduling in just about any project management scheme. Though, the grade of dependencies between the tasks must be high enough to make critical path calculation useful. Calculating the critical path for all the deliverables in a (linear) waterfall methodology just won't be all that amazing.

Advantages

Critical Path analysis is very obvious and unambiguous. It can be used to identify the most important activities, and make sure additional care is given to them. In addition, for activities that are not on the critical path, the slack can be calculated and taken into account.

Disadvantages

Critical path was planned for routine activities, which can be estimated effortlessly and correctly. Doubt about the duration of a task cannot be expressed in the critical path model, and reality can then sometimes deviate from the model's predictions.

Usage in Methodologies

For Critical Path scheduling to be effectual, tasks must be known early in advance, and for analysis to be useful, the tasks must have visible dependencies. This makes it inappropriate for methodologies like XP, where activities are small, scheduled only shortly in advance, and tasks have few to no need upon each other.

Tools

Critical Path analysis is a basic project management technique that is extensively supported by a variety of project management applications. A very well acknowledged one is Microsoft Project [CP-MSPROJ]. Another means that can do Critical Path analysis is PlanBee [CP-PLANBEE]. Another one is Open Plan [CP-OPENPL]. These are all commercial applications. A Free Software

Notes

application that supports Critical Path Analysis is for example Manage-XPS [CP-MAN-XPS]. There are certainly many more.

Self Assessment

Fill in the blanks:

5. Project planning is a feature of Project Management, which includes of various .....
6. Usually Project Planning is believed to be a process of estimating, ... and assigning the projects resources in order to bring an end product of proper quality.
7. Risk Management is a procedure of identifying, analyzing and responding to a .....
8. The Project Plan documents all the assumptions, activities, schedule, ... and drives the project.
9. ..... Management is also essential to accommodate the implementation of the project at present under development in the production environment.
10. PRINCE2 is a process-based approach for project management providing a simply tailored and ..... methodology for the management of all types of projects.
11. The tasks connected in a project network are classically the ..... elements of a Work Breakdown Structure.
12. MTA is a software engineering method for evaluating the actual progress of a project in relation to its .....

1.4 Problems with Software Project

The below list of software project failure reasons is not prioritized. Some of the reasons are claims that were measured by researchers.

The maturity of the software engineering field

- The software engineering field is much younger than the other engineering fields and that, in time, will get more established.
- The field is young and thus most of the field engineers and managers are also young. Young people have less experience and therefore tend to fail more.
- Young people are more optimistic and tend to estimate roughly.

Shortage of Knowledge Base

As a comparatively young engineering field, software engineering is short of accumulative knowledge bases. For example, the well-known gang of four book "Design Patterns: Elements of Reusable Object-Oriented Software" was first published in late 1994. The book suggests design patterns to ordinary software design problems and it is one of the well-known knowledge base materials in the software engineering field. "Software engineering has evolved progressively from its founding days in the 1940s", but it is still short of accumulative knowledge base as opposed to other engineering fields. Another example is OOP (Object Oriented Paradigm). OOP is considered to be more effectual than the previous procedural paradigm. OOP was only embraced by the Software industry in the 1990s. "Even though it originated in the 1960s, OOP was not generally used in mainstream software application development until the 1990s."

Notes

Software is not Tangible

As contrasting to other engineering fields like civil engineering, the software engineering building blocks are much less tangible and therefore hard to measure and estimate. "Software is conceptual"

Competition: Harsh Deadlines

The competition in the software industry is cruel. The Time-To-Market (TTM) is vital and the drive to meet harsh deadlines is massive. This characteristic, along with other methodologic al anomalies like "Code first; think later" and "Plan to throw one away; you will, anyhow," makes competition cruel. The hard competition in the software industry causes not only the need to deliver ASAP, but also the requirement to catch as many probable customer eyes as possible. Firing in every direction causes disorganization, fast coding and projects that are not well planned.

Technology Changes Rapidly

"Software development skill change faster than other construction technologies." Until lately, Microsoft was regularly bombarding the industry with new technologies. Rapid technology changes introduce liability for software manufactures.

Example: New Operating Systems obligate a company like Ahead to release a new adaptable version of Nero.

A few years ago, Microsoft had determined to change the way it introduced new technologies to the software industry. It introduced the wave method. In this methodology, Microsoft agreed to release a bundle of technologies (tools, Framework, programming languages, etc.) in waves, every several years and by that, let the software industry adapt and digest the new upcoming technologies. Lots of popular software like Ulead Video Studio and Nero that used to run on Windows XP do not run on Windows Vista.

Change is Tempting

A building architect will not choose to add extra floors during the building construction. The result would be dreadful, as the building foundations were not constructed for it. The software architect's hand, though, will be much more loose on the trigger. Irresponsible changes like adding new features and redefining existed ones may cause deadline disobedience and/or bad planning and coding (patch). Given the harsh competition, it looks like changes are expected.

Bad Time Management

Estimating the development time should associate to the employees ("resources") on hand. In some cases, managers estimate and then impose a time table as if they were the ones who were going to do the developing. This type of enforcement yields pressure on growth and may harm it. Moreover, violating deadlines in this condition is common.

Bad or no Managing Skills

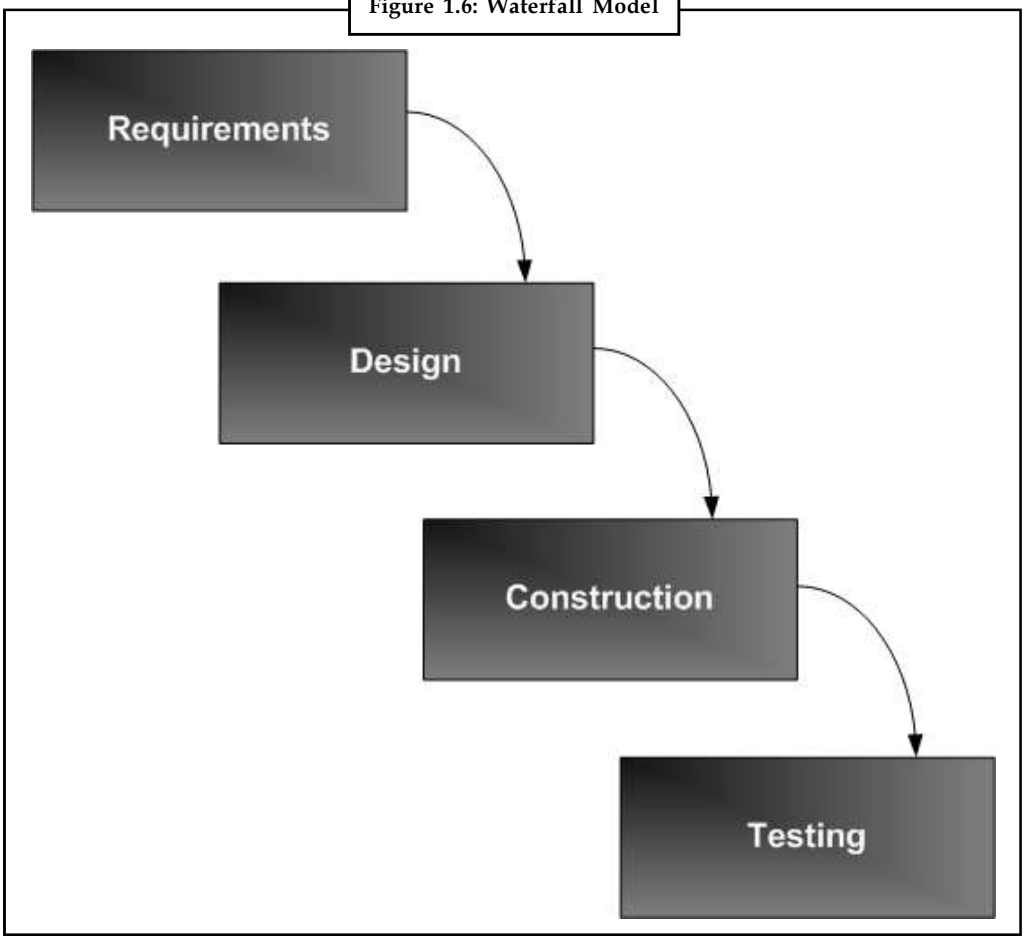
It is ordinary that software managers are used to being excellent, successful and professional software engineers. Regrettably, the skills are not the same when it comes to successful managing. Great engineering skills do not guarantee great managing. Newborn software managers do not obtain the right, or any, guidance.

Notes

Wrong or no Software Development Life Cycle (SDLC) Methodology

Growing life cycle methodology must be part of software project management. Nevertheless, it should not be forced into the R&D environment. The software engineering field is comparatively young, but still there are already well-known developing life cycle methodologists (Agile, Crystal, Spiral, Waterfall, etc.), successful stories and case studies. Software project managers may adopt one of the existing methodologies, but usually there is also a need to adapt the methodology to the company on hand. The adaptation includes: company culture, employees, marker, managers, etc. This is the Waterfall Model:

Figure 1.6: Waterfall Model



Bad or no Documentation

Documentation should be measured as a "must have" and not "nice to have". Documentation is an integral part of developing the life cycle process. It should not be caught as a nagging tedious task, done for the sake of some strict QA manager. There are a variety of types of software project documentation, each related to a certain stage in the development life cycle of the project. For example:

- Statement of Work: preliminary requirements and project frame, usually written by the customer
- Marketing Requirements Document (MRD)

Notes

- Software Requirements Specifications (SRS)
- High Level Design (HLD), written by R&D
- Low Level Design, written by the R&D
- Project Schedule
- Software Quality Assurance Plan (SQA)

There are many templates and different names to the above documentation. Nevertheless, the important thing is that their survival requires the position holder to think before working on the project. The documents need to be stored and updated during the life of the project as it is done in a source code case (out of date document is a bug).

Caution Badly written or no MRD or SRS document can cause project failure.

Bad or no Software Requirements

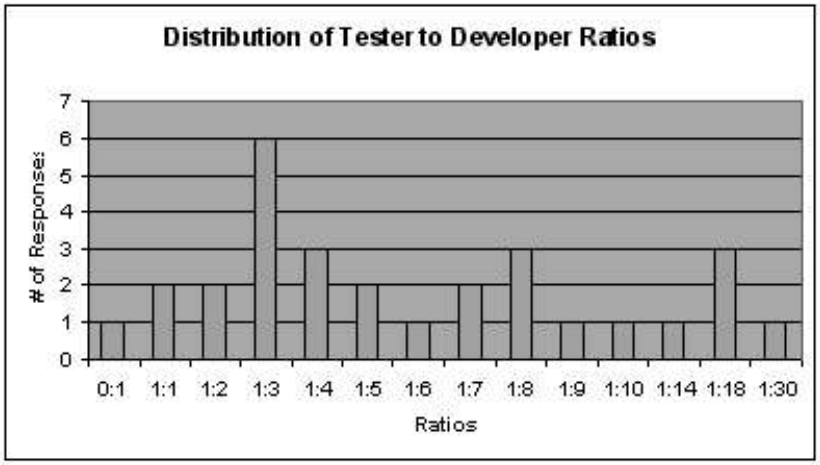
As much as it sounds strange, in some software projects SRSs (Software Requirement Specifications) do not exist or are badly written. There are many types of SRS formats and even if it was only one common template, the content would vary from company to company. It is a question of how well-defined the requirements are. We have never heard about a well-defined SRS that caused projects to fail, but I am familiar with the opposite. A terse requirements document affects the ability to break the software complexity, generate tasks and estimate time. Furthermore, inadequate definitions cause misunderstandings and wrong implementation. Changes to the project during the development become inevitable and, in time, project deadlines will be missed.

Lack of Testing

- Those who develop the software should not test it. The developer should run unit testing, but it is not a replacement to an objective QA test.
- Testing only at the end of a long milestone raise harms due to the load of testing and inherent problems that should have been caught at earlier stages.
- Furthermore, managers tend to rush the testing period at the end of the milestone in order to release on time.
- QA that does not bite and has no real power does not have the right effect on the R&D department and is there for the project itself.
- QA should be started as soon as the software project starts. Hence, even in the planning stage. QA participation in early stages is important for its preparation for the software. For example, QA should also check the SRS document and make sure that the software was implemented according to it.
- The following Professor Brooks rule of thumb might seem radical, but being given no proportional time for planning and testing is indeed problematic: "1/3 of the schedule for design, 1/6 for coding, 1/4 for component testing, and 1/4 for system testing."
- Tester to developer ratio: there is no rule of thumb that defines the number of QA engineers per software engineer. The reason for that is that it depends on many variables and more specifically on the characteristics of the software. For example, if "multilingual" is a software requirement, then the number of tests increases. Another example will be the

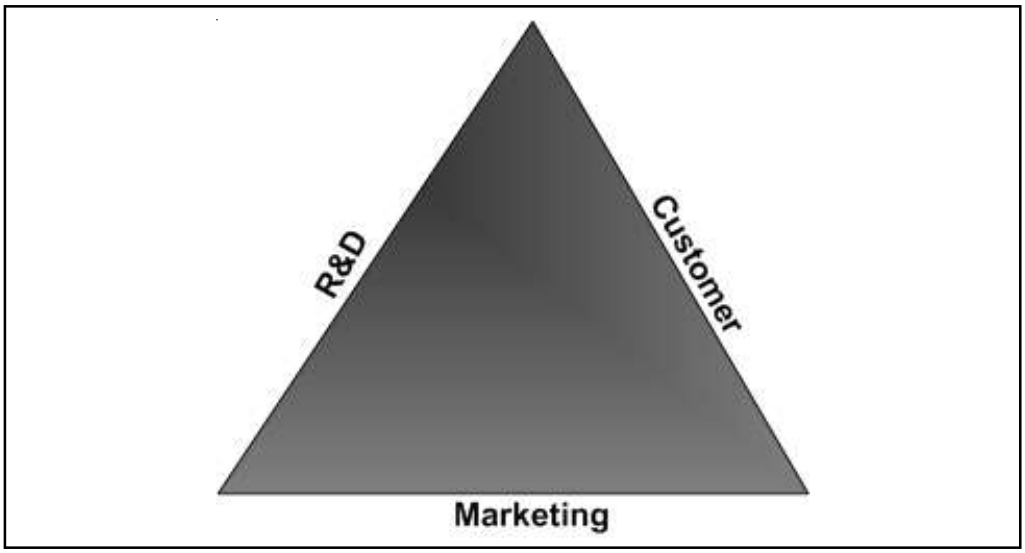
Notes

number of supported Operating Systems. The number of testers required to test the software requires estimation. Bad estimation can cause project failure. There are several models that help with Tester to developer ratio. According to a recent informal survey held at QAI's 20th Annual Software Testing Conference in September of 2000:



Poor Communication among the "Holy Triangle:" Customers, R&D and Marketing

The "Holy Triangle," as we describe it, describes the important relationships between the customers, marketing and R&D. As seen in the picture below, the marketing side combines the Customer and R&D. Marketing interviews the customers and picks at their needs continuously. Then it brings the important knowledge to the R&D department. Strange as it may seem, in a number of commercial software companies, the customer requirements and needs are not gathered. This anomaly can happen if the company suffers from "Hero base project." In this case, a certain persona, generally the company CTO, enforces the project requirements without taking into thought the market and the customer's real needs. The result of this behavior might be the creation of software that lacks the market needs and, in time, is a project failure. "? The communication of requirements from customers to developers is a common source of problems, as is the communication from developers to customers of the repercussions of those requirements?"



Notes

Human Resources Management

It is given information that lots of software project manager's start working without the basic guidance of how to motivate people to succeed. Software managers tend to manage their software engineers only in the professional engineering aspects. Though, software engineers are people too. Learning what motivates them requires time and will from the manager side. No two men are alike, both in terms of management and motivation.

No Version/Source Control

Astonishing as it may sound, some software projects are not backed up in source control. Sources get lost; versions cannot be regained; products on customer's side can not be reconstructed.

No or Bad Risk Management

"A project risk is an unsure event or condition that has consequences for the project? The purpose of risk management is to identify, analyze, and respond to project risks? ". Given the above items and the fact that software projects tend to fail, it would be absurd not to manage risks. The Risk Management Document is the leading design to enforce the software company to think about what can go wrong. The thinking process itself can solve problems before they even happened. Examples of risk:

- Unfinished or badly written requirements
- Choosing a technology that is not known by the current developers
- Relying on difficult third party software.

The Risk Management Document needs to be updated throughout project life cycles. It should not be too general or vague, but address real details of problems that might occur.

Self Assessment

Fill in the blanks:

13. The …….. is vital and the drive to meet harsh deadlines is massive.
14. Documentation should be measured as a "must have" and not "……".
15. The "Holy Triangle," as we describe it, describes the important relationships between the customers, marketing and …….

Caselet

Software Study in Complexity

The road to hell is paved with works-in-progress. How bad! But that's a Philip Roth quote that greets you right at the start of Software Project Management, by Bob Hughes and Mike Cotterell. Now into its third edition, the book from Tata McGraw-Hill Publishing Co. Ltd. (www.tatamcgrawhill.com) handles "the more agile approaches to software projects such as Dynamic System Development Method (DSDM) and Extreme Programming (XP)", apart from giving inputs on Project Management Institute of the US and Association of Project Management of the UK.

Contd.....

Notes

How are software projects special? The authors speak of four qualities, viz. invisibility, complexity, conformity and flexibility. "With software, progress is not immediately visible"; and "per dollar, pound or euro spent, software products contain more complexity than other engineered artefacts." To add to the woes of software creation, "Organisations, because of lapses in collective memory, in internal communication or in effective decision-making, can exhibit remarkable `organisational stupidity' that developers have to cater for." Software is so easy to change; but flexibility is both a strength and source for trouble because "software will change to accommodate the other components rather than vice-versa."

Often projects fail because of faulty estimation of effort required. An over-estimate can cause the project to take longer, because two laws come into play: Parkinson's Law, that work expands to fill the time available, and Brooks' Law, that putting more people on a late job makes it later! What happens if there is an underestimate? Your staff may respond to deadlines with substandard work, and this is 'Weinberg's zeroth law of reliability' in action - "if a system does not have to be reliable, it can meet any other objective."

A parametric model you'd read about in the book is COCOMO (short for COnstructive COst MOdel). The basic equation is effort = c x sizek where effort is measured in pm, or the number of 'person-months' consisting of units of 152 working hours, size is measured in kdsi, thousands of delivered source code instructions, and c and k are constants. "Boehm originally used mm (for man-months) when he wrote Software Engineering Economics," states the book, and that's another book you can catch up with. Also, there is now a newer version called COCOMO II, like a movie sequel. For Boehm, the constants depended on the mode of the system, which was organic, embedded or semi-detached.

Take my suggestion: Better be attached to completing the software project because an unfinished one is only a ticket to hell.

Thinking Hat for Red Hat

READ, practice and pass the test. Thus screams the cover of RHCE, Exam Study Guide by Michael Jang, published by Dreamtech Press (www.wileydreamtech.com). The abbreviation stands for Red Hat Certified Engineer Linux. "Major corporations, from Home Depot to Toyota, and governments such as Germany, the Republic of Korea, and Mexico have made the switch to Linux," states the preface. "Major movie studios such as Disney and Dreamworks use Linux to create the latest motion pictures."

About the exam, the author cautions that it is a gruelling five-and-a-hour exercise (twice the length of a world-class marathon). "The most important thing that you can take to the exam is a clear head."

Okay, it's time for some teasers: Which of the following services works to connect Linux to a Microsoft Windows-based network - NFS, SMB, DNS or Windows for Workgroups? Which of the following commands would you use to write an ISO file to a CD - cdburn, cdrecord, isorecord, or xcdrecord?

Some queries are detailed: "You are running an ISP service and provide space for users' Web pages. You want them to use no more than 40MB of space, but you will allow up to 50MB until they can clean up their stuff. How could you use quotas to enforce this policy? (a) Enable grace periods; set the hard limit to 40MB and the soft limit to 50MB; (b) Enable grace periods; set the hard limit to 40MB and the soft limit to 50MB; (c) Enable grace periods; set the soft limit to 40MB and the hard limit to 50MB; or (d) None of the above." Are there answers? Yes, in this problem, 'c' is the right answer because "this will warn users they are over their limit after the grace period, but will make sure they do not exceed the 50MB true maximum barrier." Option `a' is wrong because "the soft limit must be less than the hard limit," and `b' is same as 'a'. Option 'd' is incorrect because 'c' does the job.

Notes

1.5 Summary

- A project includes a number of activities that must be completed in some particular order, or sequence.
- Projects have a specified completion date.
- The customer, or the recipient of the project's deliverables, expects a certain level of functionality and quality from the project.
- Software projects are disreputably hard to define.
- The aim of these processes is to make sure that various Project tasks are well coordinated and they meet the various project goals including timely completion of the project.
- The thick methodologies we consider are RUP, SSADM and PRINCE2. XP, SCRUM and Crystal Clear are measured as thin methodologies.
- The Project Management Body of Knowledge is a broad term that explains the sum of knowledge within the profession of Project Management (PM).
- In a project a manager can judge if the projects are on schedule/budget.

1.6 Keywords

CCTA: Central Computing and Telecommunications Agency

EVM: Earned Value Management

PM: Project Management

PMBOK: Project Management Body of Knowledge

PRINCE: Projects in Controlled Environments

RUP: Rational Unified Process

SSADM: Structured Systems Analysis and Design Methodology

UML: Unified Modeling Language

1.7 Review Questions

1. "A project has never happened before, and it will never happen again under the same conditions". Discuss.
2. A project risk is an unsure event or condition that has consequences for the project? Justify.
3. "Software development skill change faster than other construction technologies." Do you agree? Justify.
4. Explain the various project planning projects.
5. In project scheduling how the resources are allocated?
6. In your words explain the advantages and disadvantages of RUP.
7. Do you think SSADM is a methodology? Explain how?
8. Discuss how the PMBOK tries to imitate the growth of knowledge and practices?
9. Explain the uses of Holy Triangle.
10. Explain the various problems with the software project.

Notes

Answers: Self Assessment

1. Work
2. Same
3. time span
4. considerations
5. processes
6. scheduling
7. risk
8. timelines
9. Change
10. Scalable
11. Terminal
12. Planning
13. Time-to-Market (TTM)
14. nice to have
15. R&D

1.8 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://en.wikipedia.org/wiki/Software_project_management

http://www.comp.glam.ac.uk/staff/dwfarthi/projman.htm

Notes

Unit 2: Step Wise Project Planning

CONTENTS

Objectives

Introduction

2.1 Project Scope

2.1.1 Scope Planning

2.1.2 Scope Definition

2.1.3 Create WBS (Work Breakdown Structure)

2.1.4 Scope Verification

2.2 Project Objectives

2.3 Project Infrastructure

2.3.1 How to start building a Project Infrastructure

2.3.2 Project Infrastructure and Governance

2.4 Project Characteristics

2.5 Effort Estimation

2.6 Risk Identification

2.6.1 Qualitative Risk Identification Techniques

2.7 Summary

2.8 Keywords

2.9 Review Questions

2.10 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize project scope
- Describe the objectives, infrastructure and characteristics of project
- Explain the effort estimation
- Discuss various risk identification

Introduction

Planning is the trickiest process in project management. This unit describes a framework of basic steps in project planning. Many dissimilar techniques can be used but this unit tells the overview of the steps and activities in each step of project planning. A main step in project planning is to plan in outline first and then in more detail. The key to a effectual project is in the planning. Generating a project plan is the first thing you should do when undertaking any kind of project. Often project planning is ignored in favor of getting on with the work. Though, many people fail to realize the value of a project plan in saving time, money and many problems.

Notes

2.1 Project Scope

Project scope is the total amount of work done to attain a project management team's project. It is the sum total of all the products, necessities and outcome as one project. In terms of project management, scope is the process preferred to make sure that the project is done. In the planning process, the scope of the project is examined. A project's requirements, products, and results are defined. The definition of the scope is the most essential part of project management. It assist the team know what needs to be done and what achieving the project will entail. Specially, the project management team discusses the needs, risks, strategies, effort and cost. Then the project management team splits the project scope into short-term objective to be accomplished. The project managers then allocate different people to accomplish the goals. After the scope is done, it is checked and approved to make sure the project meets the product requirements, services and results. It is the final step where project objectives are analyzed critically to make sure that the project meets the quality outcome.

When people talk about scope, they instantly think time and cost. Time and cost are outputs of scope. Determining scope is a different exercise. When we talk about identifying the scope, we are talking about developing a ordinary understanding as to what is included in, or excluded from, a project. We are not talking about deciding how long it will take, or how much it will cost. That comes after the scope is described.

Anyone who has ever done a software project will have stories of how scope changes caused pain. Scope is bound to change, and this is to be expected. As the detail becomes clearer, more complications creep in. These are not foreseeable at the start and hopefully we build in a contingency for what we cannot see. The scope changes that usually cause problems are those where the perception of what was in and out of scope was different between various parties. The Project Manager assumed there would only be four or five reports, and the business assumed ten to twenty. No one felt it was worth talking about because they assumed the other person thought the same way they did.

Example: If we were looking for an economical car, we would first define the scope. For example, we want a 4-cylinder front wheel drive with seating for 2 adults and 2 children, and less than 2 years old. Maybe you also want it to be a red convertible. Having defined the scope, you can calculate cost and time. How much you will have to spend and how long you will take to buy it. If you get the scope wrong, the time and cost will be wrong.

The project scope contains the following aspects to be managed:

1. Scope Planning
2. Scope Definition
3. Create WBS
4. Scope Verification

This sort of organization contains the processes required to guarantee that the project includes all the work required, and only the work required, to complete the project effectively. It is essentially concerned with the defining and controlling of what is or is not included in the project.

The processes, tools, and procedure used to administer product's scope differ by application area and are often described as part of the project life cycle. The project's output is a single product but that product may comprise subsidiary components, each with its own separate but interdependent product scopes.

Example: A new computer system would generally include four subsidiary components—hardware, software, training, and implementation.

Notes

The significant thing to remember is that "Completion of the project scope is calculated against the project plan, but completion of the product scope is measured against the product requirements". We are also well aware of the fact that if project is multifaceted and larger in size, the different team & different components are required to complete concurrently. There exists trade offs among the competing objects and alternatives are there to meet the needs of stakeholders.

2.1.1 Scope Planning

Scope planning refers to a project management process that describes boundaries and deliverables. A deliverable is an input/output term that refers specially to the unique and individual products, elements, results, or items that are produced for delivery at the conclusion of a specific project component, or at the conclusion of the project as a whole. Deliverables can come in a number of different variations. Deliverables can be in the form of a written report, which can be extremely lengthy and can encompass extensive amounts of information and data. Deliverables can also be much shorter. In some cases, the end deliverable may be a short report, a slide presentation, a poster, a short blurb, or even a slogan. In these cases, the deliverable may actually be very short, and rather than containing all information gathered over the course of a project, may contain a succinct message that is meant to represent the sum total of the information without actually presenting it directly. Deliverables towards the end of a project life are typically referred to as external deliverables, and these typically require the review and/or approval of the customer or financially responsible party. The basic matrix of a scope planning analysis consists of three main categories: Initiation, planning, and definition, with two control categories: Verification and change control mix together between the three main categories.

Beginning inputs contain program deliverable description, strategic planning, program selection criteria, and historical information. Tools and techniques comprise program selection methods and expert judgment. The output of the initiation phase will contain a program charter, the identification and assignment of a program director, and the identification of known constraints and assumptions. The planning category covers descriptions on deliverables, the program charter, constraints, and assumptions. Tools and techniques involved in this group contain deliverable analysis, a benefit/cost analysis, the identification of alternatives. The final main category contains a statement of scope, a definition of assumptions and constraints, and other planning outputs and historical information. Tools and techniques involved contain work breakdown structure templates and decomposition. The output of definition is work breakdown structure, and the defined scope section of the project management plan.

Two control devises, verification and change control are mix together between the main categories of scope planning. Verification's inputs are work results, and deliverable documentation. Inspection is its sole tools and technique. Formal acceptance is verification's output, and is necessary to advance the project management plan to the next level.

Change control is positioned between planning and definition. Its inputs are comprised of work breakdown structure, performance reports, change requests, and the scope management plan. Tools and techniques contain the scope change control system, performance measurements and additional planning when indicated. The outputs of change control are changes in scope, corrective actions, and lessons learned entered in the information base for other project management considerations.

Did u know? What is Deliverables?

Deliverables can come in a number of different variations. Deliverables can be in the form of a written report, which can be extremely lengthy and can encompass extensive amounts of information and data.

Notes

2.1.2 Scope Definition

Scope explanation often account for a paragraph or two in a Business Case or Project Charter. Regularly, they are qualitative and/or focus on general statements. "We will develop service by providing an information system to answer to customer inquiries." Is it a real time system? Is it all screen-based? What reports can be produced? Where does the knowledge come from? What management is required for the data? Is all the data compatible? Do you want to generate standard letters? How many letters? How customizable are the letters? Do you want to store the questions? Do you want to store the answers? In order to define the scope, there will be supposition that need to be made. There is no point in waiting until everything is clear to define scope. By that time, the project will possibly be finished. Each of these assumptions should be documented and followed up at a later date to validate the scope. If the supposition is false, it may have an impact on the scope.

There are plentiful ways to define. Ideally several ways should be used. Each looks at the situation from a different perception and will elicit different information. We look at three main ways in this paper. They are:

- Define Deliverables
- Define Functionality and Data
- Define Technical Structure

Define Deliverables

One method to focus people on the scope is to describe the internal and external deliverables:

- External deliverables are things the project delivers to the users, e.g., screens and reports. Users usually think of a system in these terms. It also comprises any hardware or software required by the users or the project team.
- Internal deliverables are things the project makes internally, e.g., Project Charter, Business Requirement Specification, etc.

It is likely that the users will not be completely clear on all the deliverables. In this situation you can make generic assumptions. For example, you might not know precisely what reports are required but you allow for 12 unspecified reports.

Once the external deliverables are described, the Project Manager can define the internal deliverables.

Example External Deliverables

Name	Description
License Detail Screen.	Screen to enter and view license details
Company Summary Screen	Screen to view all licenses issued by a particular company. Facility to drill down to License Detail Screen.
License Due Report	Report listing all licenses due in the next period. Facility to select a period e.g. 1 week, 4 weeks, quarter
5 Reports	Allow for 5 unspecified reports
Server	Server to run the application
Etc.	

Notes

Name	Description
Project Charter	Document identifying how the project will be managed
Business Requirement Specification	Document identifying the requirements for the project
Weekly Reports	Status reports to be issued weekly
Prototypes x 3	Three prototypes will be allowed for in the development.
Etc.	

Define the Functionality

Another method is to define the functionality. This should not be either a lengthy or detailed process. Usually, depending on project size, the exercise can be completed in a one hour to half-day seminar. A good system is to use a functional decomposition. If using a spreadsheet and a projector, a scribe can make the scope as it is discussed. Remember to start all functionality with a verb. It is valuable to do the functional decomposition in conjunction with a data definition. If this is unfeasible, once the scope is discussed, it will become reasonably clear what data is required.

The Project Manager can conclude if there are any situations that need to be clarified with the users, and finalize the scope definition. If for example, in defining the functionality it becomes apparent that considerable information will need to be move from a legacy system, which is known to be inaccurate, data cleansing can be factored into the scope.

Task External deliverables are things the project delivers to the users. Analyze

Example Functional Decomposition

1.0 Capture License details

1.1 Set up companies

1.2 Set up products

1.3 Create licenses

1.4 Modify licenses

1.5 Delete licenses

2.0 Generate payments

2.1 Create payment report

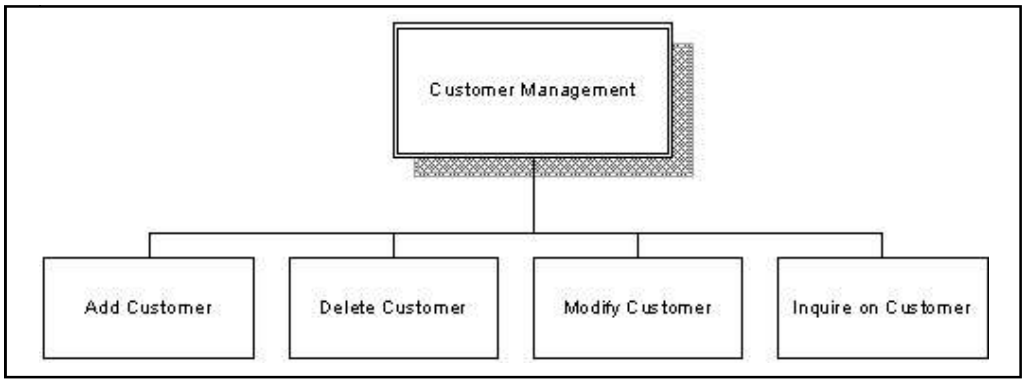
2.2 Authorise payments

2.3 Notify accounts

Etc.

It can also be defined as a diagram:

Notes



Defining the Data

Defining the data is comparable to functionality, and should be employed in conjunction with functionality. The process is probable to capture what users expect to see in a system. The intention is not to make the business users, data modelers. The purpose is to get the business users to verbalize their requirements for information in a structured manner. Ask the users what are the people, places and thing they want to keep track of. In this case, the focus is on nouns.

This way will not capture data that may be required to technically make the system work. For example, it will not capture things like transaction log files, archive files, SQL script files, etc. Post workshop, the Project Manager will need to sit with a data modeler to sort out what else is required. The hardest part is to stop doing a data model. Keep the focus on where the data is to come from, and recognize what is new, where the interfaces are likely to be, is existing data suitable, is the data currently captured, etc.

Data Definition Example

Name	Description
Companies	Details of the company including address, overseas offices, and up to ten contacts
Licenses	Licenses for all software and hardware used in the organisation. Include contracts, correspondence, quotes and any other related documents. Does not include manuals
Renewal dates	Dates the license is due for renewal and the cost of the renewal.
Etc.	

Technical Structure Definition

This procedure of scope definition can be useful for describing scope where the project is focused on infrastructure. It can also be valuable in a situation where an existing system is being modified. The output can be either a table, or a diagram. A table might just list the components to be customized and the modification. The structure diagram might recognize the whole system and highlight which components are being modified and how they are being modified. It may also be appropriate to indicate the purpose of each component; however it will probably be vague at this stage of development.

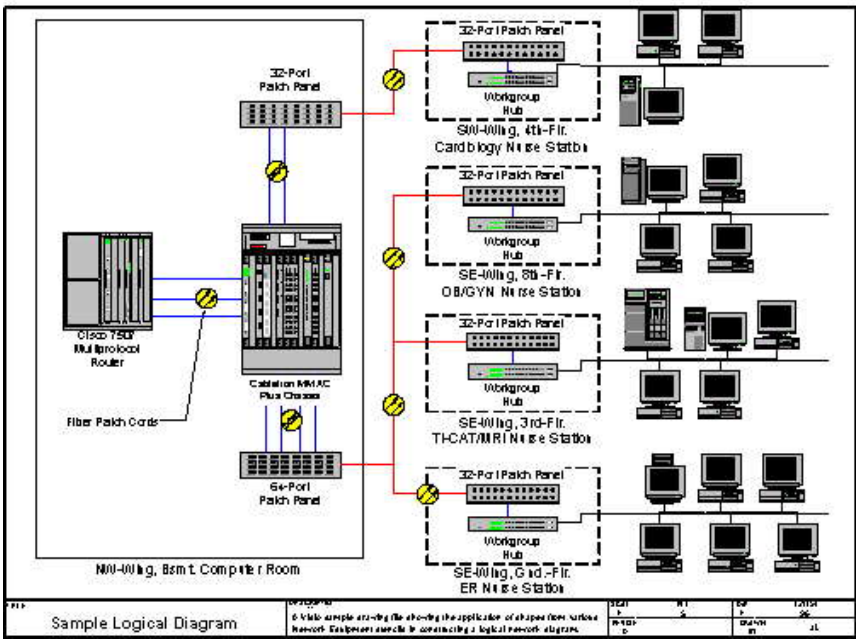
Example: The 'outputs HTML' module takes information retrieved from the database and inserts it into an .asp document for output to the server. It also updates a transaction log with the database information and time of the output. If an error occurs in retrieving data from the database, an error log is updated and an error page sent to the server.

Notes

Example Technical Structure Table

Component	Description
Subsystem1	Handles all customer processing and interfaces to CMS (Customer Management System).
Subsystem2	Carries out inquiries on billing systems (2) and combines data into common format. Sorts data by date of payment.
Etc.	

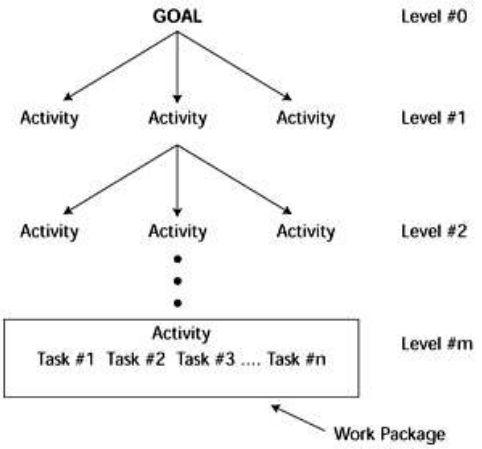
Figure 2.1: Example Technical Structure Diagram



2.1.3 Create WBS (Work Breakdown Structure)

The Work Breakdown Structure (WBS) is a hierarchical description of the work that must be done to complete the project as defined in the Project Overview Statement (POS).

Figure 2.2: Hierarchical Visualization of the Work Breakdown Structure



Notes

To understand WBS, we must know about activity.

An activity is nothing but simply a chunk of work. Activities turn to tasks at some level in the hierarchy. A task is a smaller chunk of work. The terms activity and task have been used interchangeably among project managers and project management software packages.

Work Package

A work package is a complete description of how the tasks that make up an activity will actually be done. It includes a description of the what, who, when, and how of the work.

Decomposition

Breaking down work into a hierarchy of activities, tasks, and work packages is called decomposition. For example, take a look at the top of the WBS in Figure 2.2. Notice that:

- The goal statement from the POS is defined as a Level 0 activity in the WBS.
- The next level, Level 1, is a decomposition of the Level 0 activity into a set of activities defined as Level 1 activities. These Level 1 activities are major chunks of work. When the work associated with each Level 1 activity is complete, the Level 0 activity is complete. For this example, that means that the project is complete. As a general rule, when an activity at Level n is decomposed into a set of activities at Level n+1 and the work associated with those activities is complete, the activity at Level n, from which they were defined, is complete.

Notes Importance of decomposition: Decomposition is important to the overall project plan because it allows you to estimate the duration of the project, determine the required resources, and schedule the work. The complete decomposition will be developed by using the completeness criteria discussed later in this unit. By following those criteria, the activities at the lowest levels of decomposition will possess known properties that allow us to meet planning and scheduling needs.

WBS is explained briefly in Unit-3

2.1.4 Scope Verification

Verification is a quality guarantying procedure or technique applied by Project Management whereby a valuation of a component, product or service is finished at the end of a phase or project to verify or confirm that it assures all of the regulations or specification requirements. Verification can take place during production or development, and is usually an internal process. In contrast, validation is the process by which the Project Management set up that the component; product or service meets the needs of the intended end-user or customer. It can be said that validation assures that the correct component, product or service was created, and verification guarantee that it was built or produced correctly. Building the correct component speaks to end-users needs, while doing it correctly checks that the actual development process was followed suitably. In some applications, requirements are such that it is necessary to have written requirements for verification and validation, as well as formal protocols for determining compliance. Project Management efficiently implementing a verification/validation process can save the producer from the financial burden of expensive recall campaigns, costly product rework, and unexpected delays in product releases.

Notes

With software project management, once the scope of the project has been defined, it is significant to get the endorsement of stakeholders before proceeding further. If the project is small it might only require a single signature from the main sponsor or client for scope verification however with larger projects there may be a number of stakeholders involved.

With key stakeholders it is essential to get explicit scope verification which could be a formal signature on paper or via an email that specially states project approval. Whichever the method, a record of the scope verification should be retained. Of course, the stakeholders should already have been exposed to draft copies of the proposal making the verification process a mere procedure.

With other, less concerned, stakeholders it is acceptable to get implicit approval or scope verification. In other words the project manager presents the project plan to them for inspection with an ending date for voicing any questions or concerns. After that date if the project management team has not heard from the stakeholder it is assumed that approval has been granted.

With other interested parties the project manager require only deliver to them a copy of the project plan letting them know the document is for informational purposes only. These parties are not in a position to request changes and by this point the project has probably already been permitted by the sponsor or main client.

Caution The project management team should however be available to answer questions or clarify the parameters of the project.

Self Assessment

Fill in the blanks:

1. The project managers then allocate different ..... to accomplish the goals.
2. Specially, the project management team discusses the needs, risks, strategies, effort and .....
3. ..... can be in the form of a written report, which can be extremely lengthy and can encompass extensive amounts of information and data.
4. ..... control is positioned between planning and definition.
5. Verification can take place during production or ....., and is usually an internal process.
6. The terms activity and ..... have been used interchangeably among project managers and project management software packages.
7. Post workshop, the Project Manager will need to sit with a ..... modeler to sort out what else is required.

2.2 Project Objectives

Successful objectives in project management are specific. A specific objective increases the chances of leading to a specific outcome. Consequently objectives shouldn't be vague, such as "to improve customer relations," because they are not measurable. Objectives should show how effective a project has been, for instance "to reduce customer complaints by 50%" would be a good objective. The measure can be, in some cases, a simple yes or no answer, for example, "did we reduce the number of customer complaints by 50%?"

Notes

While there may be one main project objective, in pursuing it there may be temporary project objectives. In lots of instances, project teams are tasked with achieving a series of objectives in pursuit of the final objective. In many cases, teams can only proceed in a stair step fashion to attain the desired outcome. If they were to proceed in any other manner, they may not be able to develop the skills or insights along the way that will allow them to progress in a productive manner.

Objectives can often be set under three headings:

- ***Performance and Quality:*** The end consequence of a project must fit the principle for which it was intended. At one time, quality was seen as the blame of the quality control department. In more recent years, the concept of total quality management has come to the fore, with the responsibility for quality shared by all staff from top management downwards.
- ***Budget:*** The project must be completed without more than the authorized expenditure. Financial sources are not always infinite and a project might be abandoned altogether if funds run out before completion. If that was to happen, the money and effort invested in the project would be forfeited and written off. In tremendous cases the project contractor could face ruin. There are many projects where there is no direct profit motive, however, it is still important to pay proper attention to the cost budgets, and financial management remains essential.
- ***Time to Completion:*** Definite progress has to match or beat planned progress. All significant stages of the project must take place no later than their specified dates, to result in total completion on or before the planned finish date. The timescale purpose is very significant because late completion of a project is not very likely to please the project purchaser or the sponsor.

Project objectives are often tough to track, vague, and lacking in depth. In project objectives, people require details to help know where they are in the process, and data helps them make conversant decisions. We like to recommend "DISCO" when forming objectives. "DISCO" can be spelled out to point us in the proper direction for creating project objectives and tracking their progress.

D - Detail Specifics

Give as much knowledge as possible and make these objectives very specific. Far too many objectives have been set, which are very grey in nature and lack data to help team members understand all specifics.

I - Include Qualitative and Quantitative Measurements

Objectives must be deliberate. When you look at an objective, you must ask, "Can we measure this?" If not, it needs to be rewritten so that it can be measured and tracked for successful completion. The only way to do this is to make sure qualitative and quantitative components are set.

Qualitative measurements calculate a project based on quality standards, quality indicators, or quality characteristics. Defect ratio, break down ratio, and improvement needs are all to be measured. Each of these can be prioritized and broken down into a specific tracking mechanism to follow and monitor.

Quantitative measurements calculate the project based on numerical indicators. Some of the most ordinary quantitative measurements are time, budget, production, work hours, process time, and development progress. Quantitative measurements normally include the need to set a series of benchmarks as a starting point to begin tracking.

Notes

S - Seek Consensus with the Team

Making sure the team agrees with the measurement is very significant. Sometimes objectives are set at the commencement of the project, and they are very loose. When the team sets a standard of measurement, it will usually be detailed and explicable. It is important because the team needs to be on the same page during planning. They must agree that these standards are the best possible measurements considering the project.

C - Create a Reasonable Approach in Obtaining Those Objectives

The approach for reaching objectives is tremendously significant. Unless the approach is understood by the entire team and supported, there will be conflict in the team's processes. Conflict means you will have people going in different directions and using various methods.

O - Operate in a Methodical Timeframe

Setting up a timeline and follow it. This timeline must make sense and be exposed to the entire team. You must constantly focus on maintaining clarity.

Example: An example of a great DISCO objective is, "We will design 15 training courses that meet organisational development guidelines by June 30 with a budget of $483,000. We will include courses on supervision, communication, performance appraisals, and creating an optimistic workplace." DISCO objectives can be very successful in pushing the project forward and bridging the gap for communication. However, good objectives will never write themselves, nor will they track themselves.

2.3 Project Infrastructure

Project Infrastructure refers to the organizational structure, processes, tools, techniques and training an organization puts in place to make projects more successful.

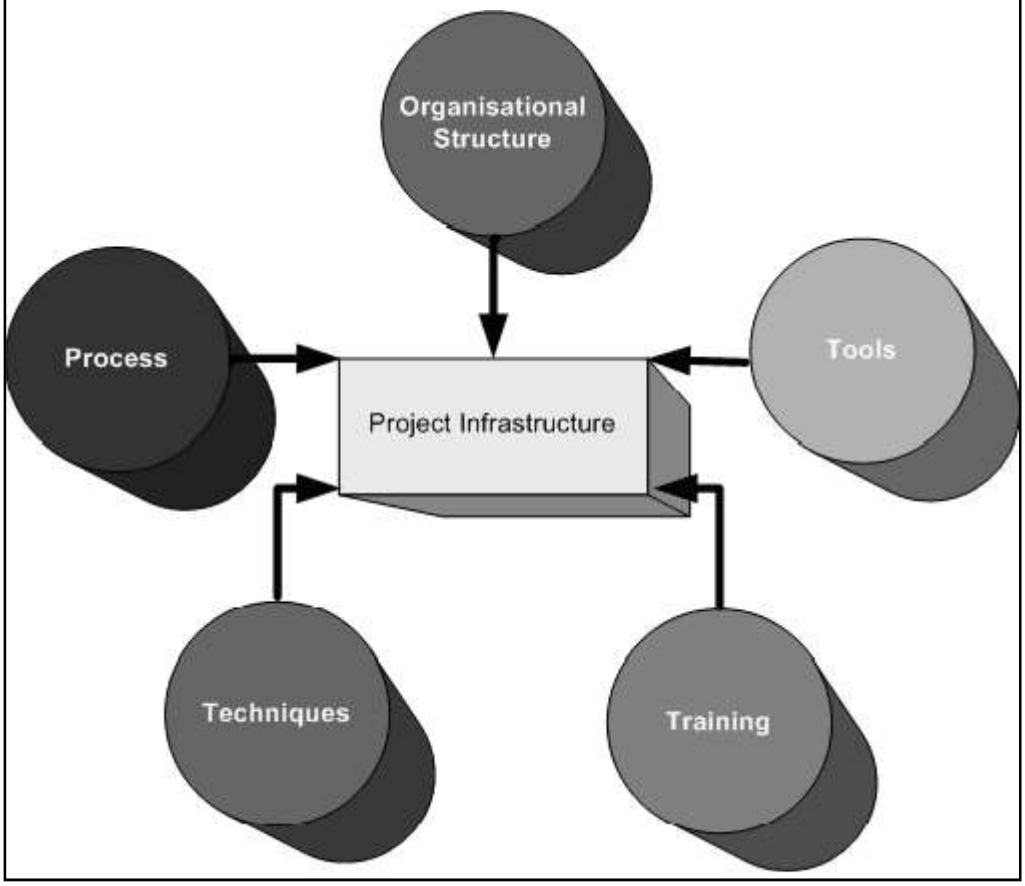
- Organizational Structure – Organizational structure including such support mechanisms as project management office, project recruiting function, financial monitoring area, etc. It also covers lines of communication and escalation.
- Processes – Typically methodologies, checklists and guidelines
- Tools – Software and templates
- Techniques – Repeatable processes such as kick off meetings, PIRs, analysis techniques, etc.
- Training – Formal and informal training and reference documentation.

Most people have never considered of having a project infrastructure. They might think of having methodologies and pattern, but not taken a holistic view of their project infrastructure. To go back to our analogy with the telco, if the review of the business area only looks at automation, or only looks at roles and responsibilities it is going to give an incomplete result. A more integrated approach is not only required. It is absolutely essential for success.

Project infrastructure is about analysis the whole environment and finding out how to put in place an environment that will work in an integrated manner to support projects. Looking more closely at each area, we can see the following.

Notes

Figure 2.3: Project Infrastructure



Organizational Structure

Structure refers to how a team is organized to be most effectual. Understanding what a team is best left to do is a good starting point. For example, is it a successful use of time and effort for a project manager to update resources requirement projections and recruit resources? Maybe it is more effective for a project coordinator working across several projects to undertake this task. Maybe a recruitment area could shortlist project resources.

Another area might be project financials. Is there explanation for a central project financial area? One quick test is to say what would be the cost of all the project managers tracking financials and what would be the cost of having a particular project financial analyst function. We have even seen a small organization combine both resource and financial roles into one person.

Processes

A project management process is not a collection of guide or templates. Templates are the output of processes. If you are doing a risk evaluation, you should not start off with a template and fill it out. You start off by understanding what is involved in carrying out a risk evaluation. You understand the techniques to generate a list of risks. You recognize who should be involved. You understand rating risks for impact and probability. After you put all this together into a process, you end up with information that finds its way into a template.

Notes

An organization needs to put in place processes in order to bring some uniformity to the way in which projects are managed. There will always be a need to tailor the process but the aim of the process is to ensure there is a minimum "reinventing of the wheel". It also means the organization has a clear understanding of what is happening because it has happened that way on previous projects. Projects don't exist in a fog.

Tools

We have seen organizations with a number of tools to do the same job. Almost surely this will result in problems. For example, one organization used both Artemis and Microsoft Project to create schedules. Those with Project could not integrate with the Artemis schedules. They didn't even have access to Artemis. There were lots of lists of dependencies maintained manually in spreadsheets.

Having a steady set of tools is fundamental to the creation of a project infrastructure. Tools may include:

- Scheduling tools
- Risk and issue management tools
- Financial management tools
- Document management tools
- Action Item management tools
- Databases for recording anything from benefits, to progress reporting, to resources.

Techniques

Techniques are the general, reusable process that an organization expands, or that an organization subscribes to. For instance, an organization may use JAD sessions. There are a number of techniques around JAD that need to be applies in a steady way. People should be trained to apply the techniques, and participants will become familiar with the techniques. It makes life much easier if people can quickly slot into an environment because they have undertaken a similar activity previously.

One exacting area where techniques are important is in the development of requirements. It should not be up to the project manager or business analyst as to what method they use to gather requirements. The organization should make a decision as to the technique they will use and every project uses the same techniques.

We have seen many association use a range of techniques such as UML, Data Flow diagrams, Functional Definition etc. to document system requirements. Each new project required a learning curve for participants where they had to become familiar with a new technique. Another factor was where people preferred another technique so you end up with resistance, or a blended technique. Typically it results in a new set of problems.

Training

Communication does not take place by osmosis. There needs to be a training program in place to communicate the way in which projects should be assumed. Training will likely range from classroom to CBT (Computer Based Training) to "one on one" training for new project managers.

The training should not end at project managers. It is significant that project participants also receive training so they can understand how the project will be managed and what they are expected to contribute.

Notes

Task Explain the difference between tools and techniques for the project infrastructure?

2.3.1 How to start building a Project Infrastructure

It is improbable you will start with a blank slate. If you are building a project infrastructure, you will almost surely have some things in place. There may be templates, processes, techniques and documentation already in existence – if not in use. There are bound to be a few tools – perhaps some of them home grown.

The starting point should be to carry out an audit of what previously exists. Look at the categories identified above, and try to collect anything that fits into a category. Also do a quick consideration of the potential usability of the item. For example, in one company we looked at, they had an old project management intranet site. We were told "There is lots of good stuff there." When we were able to resurrect the site, most of the links were broken, and much of the documentation attached to the site no longer existed. It did though provide a starting point. By talking to people who had been around long enough to have used the site, we were able to dig out a number of documents and screen prints from the site when it was in existence.

Define the Scope

Just like any project, we need to describe the scope. The limits of a project infrastructure can be blurred unless they are defined. For example, where does the project infrastructure take you in terms of financial management? Where do the usual company financial management start in relation to projects? Does the project infrastructure comprise a skills register, or is that part of the HR function? Who allocates PCs to project teams? Is this part of a project infrastructure or is it part of your normal facilities management? Is space allocation for teams part of your infrastructure?

Project Process Modeling

Once you have the scope recognized, it comes down to a business process modeling exercise. Take a new project, and navigate it through the organization to completion. It is likely there will be multiple paths. For example, you may decide that there is a different path for a small project than for a large project. There may be different steps for projects requiring capital approvals.

The purpose of this will be to understand what the optimal infrastructure is that you need to put in place. The project process model will emphasize where project infrastructure is required. For example, one step in the process will be to define scope. When taking a project infrastructure view this may mean you need one or more of the following:

- Guidelines for running a scoping workshop
- A template to record scope
- Examples of previous scope documents
- Training on how to create a scope statement
- Guidelines to identify the various components of scope (Outcome, internal and external deliverables, objectives etc.)
- Checklist of common deliverables (Training Materials, Product Documents, Operations Manuals, etc.)

Notes

By taking each step of the process, and asking "What infrastructure would make this more efficient?" a model of the infrastructure can be built.

Standing Infrastructure

The project view is not the only view. In order to undertake projects, there is also a standing infrastructure. This is analogous to having a HR department to recruit staff for a line of business. One cannot exist without the other. The standing infrastructure will include things like:

- Knowledge management
- Skills optimization
- Resource Management
- Training
- QA
- Reporting

The projects will draw on the standing infrastructure in order to complete their work. Some will be visible when you look at your project process model, however not all will be visible.

Caution A view needs to be take from a Project Management Office perspective to understand what other standing infrastructure needs to be included.

Gap Analysis

Given we now have a much clearer view of what is required, we can compare this with what already exists. What we will see will be:

- Existing materials that fit the proposed project infrastructure
- Existing material that does not fit but could be reworked to fit
- Existing material that does not fit, and will never fit
- Gaps
- Duplication of materials (e.g. In one organization I found 4 templates to request seed funding for a project. They were all in use).

Implementing the Infrastructure

It is likely there will be well-established positions when it comes to giving up the old. Some people will use approach A and see no reason to use approach B. Some will use B and see no reason to use A. If there are valid reasons for only using one approach, then it will need to be enforced. You will need the support of senior management to make it happen.

Before you start the process, there should be activities around selling the concept of a project infrastructure. There will be benefits to the company, but will there be benefits to individuals? Here are some suggestions to help acceptance.

Working Teams

Form working teams to expand parts of the project infrastructure. For example, you might have a "Project Initiation Team" who put together the entire infrastructure to establish a

Notes

project. They will need to work closely with other teams to ensure integration. Clearly defined scope and a sign-off process are important if you don't want to end up with a patchwork quilt. Another group might focus on Tools; another on Risk Management. If anything looks too controversial, then it is probably best taken out of a workshop and managed by Project Infrastructure implementers.

Look at Individual Motivations

Think what might inspire individuals and find benefits that will appeal to each person. For example:

- A "tidy" person will see appeal in having a clear concise path to follow with a consistent look and feel to the entire infrastructure.
- A "busy" person will focus on the efficiency of a clear process.
- A "less experienced" person will be motivated by having training and a clear path to follow.

Recognize Contributions

If you use a template someone has developed, identify them for the effort. In the version control section make sure the person who developed the template is identified. Recognition will encourage ownership.

Single Source of Truth

How often have we all seen numerous versions of documents? Which one should you use? Start out be making a single source of documents that is under tight control. Anyone can read, but only a select few can update – and then in a controlled way. If a document management system is available, it is a good option.

Also consider making the documents obtainable through the intranet if it exists. If people get used to going to the intranet to find master documents, it will put a control over morphing versions. Combined with a document management system where the link is probably cryptic, this can be a good way of managing documents. Converting reference documents to PDF is also useful.

As a temporary measure, we once created a web page with a listing of templates with hyperlinks. We put this in a central location on the LAN and sent everyone a link to the page. Rather than people wandering through directories looking for the most recent version, they opened the browser page from their desktop and clicked through to the version we wanted them to use.

It is Never Finished

There is always room for development. Make sure you put in place review mechanisms. Many organizations have a project management group who look at how project management can be improved, or provide training to other PMs. It is good to have a feedback forum to look at improvements to the project infrastructure.

Ownership

Make sure it is clear who owns the infrastructure, and who is in charge for authorizing changes. If the ground rules are set prior to the infrastructure being built, it will make it a lot easier to reach decisions.

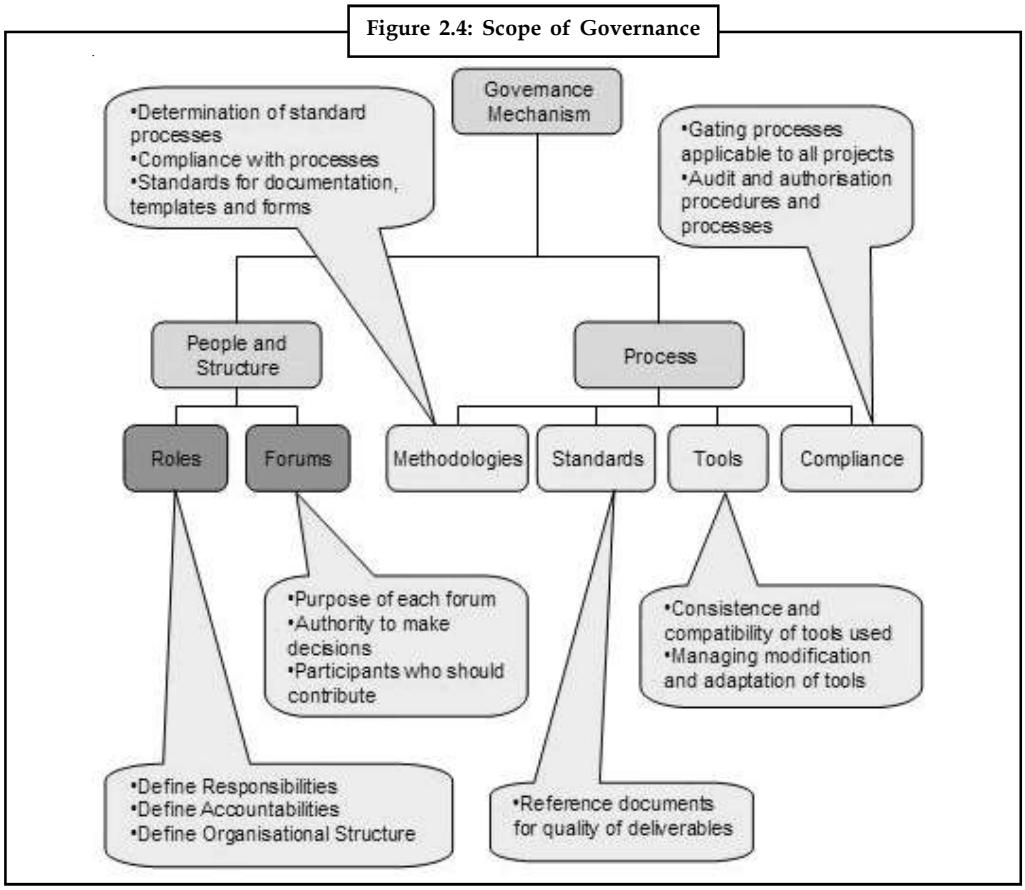
Notes

Task Explain how recognizing contributions are important in starting the project infrastructure.

2.3.2 Project Infrastructure and Governance

AS 8000 defines Corporate Governance as "The system by which Entities are directed and controlled". The following diagram shows the scope of governance.

Figure 2.4: Scope of Governance



As you can see, a project infrastructure fits contentedly as a mechanism for managing project governance. The infrastructure covers both "People and Structure" and "Process". Good governance is both a driver of the project infrastructure and an outcome of having a proper project infrastructure.

Self Assessment

Fill in the blanks:

8. Project objectives are often tough to track, vague, and lacking in ……..
9. Quantitative measurements calculate the project based on …….. indicators.
10. A project management process is not a collection of guide or……..
11. Techniques are the general, …… process that an organization expands, or that an organization subscribes to.

Notes

12. The …… purpose is very significant because late completion of a project is not very likely to please the project purchaser or the sponsor.

13. Converting reference documents to …….. is also useful.

2.4 Project Characteristics

1. ***Temporary:*** Temporary means that every project has a exact beginning and a definite end. The end is reached when the project's objectives have been attained, or it becomes clear that the project objectives will not or cannot be met, or the need for the project no longer exists and the project is terminated. Temporary does not essentially mean short in duration; many projects last for several years. In every case, however, the duration of a project is finite; projects are not ongoing efforts.

2. ***Unique, Product Service or Result:*** Projects engage creating something that has not been done in exactly the same way before and which is, therefore, unique and distinct. Projects create:

   - A product or artifact that is produced, is experimental and can be either an end item in itself or a component item.
   - A capability to perform a service, such as business functions supporting production or distribution.
   - A result, such as new knowledge. For instance, a research and development project develops knowledge that can be used to determine whether or not a trend is present or a new process will benefit society.

   The presence of recurring elements does not change the fundamental uniqueness of the project work. For example:

   - A project to develop a new commercial airliner may require numerous prototypes.
   - A project to bring a new drug to market may require thousands of doses of the drug to support clinical trials.
   - A real estate expansion project may include hundreds of individual units.
   - A growth project (e.g., water and sanitation) may be implemented in five geographic areas.

3. ***Aims/Tasks/Purpose:*** The projects are designed to achieve exact targets defined in terms of aims, tasks or a purpose. The nature and size of the project depends upon complexity of the task, realization of the aims and scope of the purpose any organization wants to achieve. In short project has to be aimed for achieving certain tasks in a given time frame.

4. ***Limited Time Scale:*** The projects are always planned considering time constraints. Extension to the project completion dead lines are always discouraged as time overrun, costs extra and in some cases opportunity cost for not completing a project is too high.

2.5 Effort Estimation

Effort estimation contains in predict how many hours of work and how many workers are needed to develop a project. The effort invested in a software project is most likely one of the most important and most analyzed variables in recent years in the process of project management. The determination of the value of this variable when initiating software projects allows us to

Notes

plan adequately any forthcoming activities. As far as estimation and prediction is concerned there is still a number of unsolved problems and errors. To obtain good results it is necessary to take into consideration any previous projects. Estimating the effort with a high grade of dependability is a problem which has not yet been solved and even the project manager has to deal with it since the beginning.

Example: The estimation of the effort invested in the development of software projects can turn into a complicated problem to be solved if the appropriate models are not available. Unfortunately, until this moment this is the situation, since there are not the necessary records in the software development companies. Years of investigation are required in order to obtain the volumes of information needed to carry out a prediction with a good level of reliability and with a low error margin.

The domains are not the most suitable, due to their size and limited number of variables, and because of the fact that they depend on the particular casuistry of each company. The quality of the prediction can improve if more appropriate sets of data are available and more deep study of the methods is performed.

Sets of data are provided bellow. Each set shows information about certain amount of software development projects. For each project, there are two variables: one, (independent variable) that refers to the size of the generated code -measured in lines of code or function points-, and the other (dependant variable) that indicates the effort (time) invested in the development of projects. Columns "Size" and "Effort" show the measure used. Column "Projects" shows the number of projects in the data.

Data	Projects	Size	Effort
Set 1	21	function points	person-days
Set 3	18	thousands of lines of code	man-months
Set 4	33	lines of code	man-months
Set 6	35	lines of code	person-hours
Set 11	33	function points	man-months

Effort Estimation is briefly explained in Unit-6.

2.6 Risk Identification

Risk identification is utilized in risk management to answer the questions: What can happen? How can it happen? Risk identification is the process of distinguishing the opportunities opened up by each activity or phase of the project and clarifying where the risk lies. The agreed lenience of risk should help identify the amount of time should be spent in identifying risk, but, at least the 20% of the risks that would have 80% of the potential impact should be identified. There are many techniques to aid in risk identification and they generally fall under the heading of either quantitative or qualitative risk identification techniques.

2.6.1 Qualitative Risk Identification Techniques

Following are the major qualitative techniques of Risk Identification:

- Assumptions Analysis
- Check Lists

Notes

- Prompt Lists
- Brainstorming
- Facilitated workshops
- Interviews

Assumptions Analysis

It is evitable that when you start scheduling or outlining your project you will be making assumptions. Making supposition within a project will always create risks and a way to help manage this is to list all the assumptions of each phase or stage of your project against a timeline. Then think about the result of the assumptions and how they influence the other parts of the project. Try creating a high level storyboard of the project showing risk and assumptions are associated, this can show the effect of decisions and should provide a better understanding about the risks in a program or project.

Check Lists

Risk checklists are often constructed upon a Project or Program Managers past experience or the Project Management experience of an organization. They will normally take into account:

- Quality experience
- Formality of development
- Novelty of application
- Impact on business
- Requirements standards
- Software identification
- Projects concurrency
- Dependencies
- Project duration
- Flexibility of delivery
- Planning estimates
- Stability of suppliers
- Range of sites
- Impact upon status quo

Prompt Lists

Risks here will be identified by rational examination of each Program or Project aspect. Following is a list of common areas from which risk can arise:

Notes

External	
Infrastructure	Relating to infrastructures such as computer networks, transport systems for staff, power supply systems
Economic	Relating to economic factors such as interest rates, exchange rates, inflation
Legal & Regulatory	Relating to the laws and regulations which if complied with should reduce hazards (E.g. - Health and Safety at Work Act)
Environmental	Relating to issues such as fuel consumption, pollution
Political	Relating to possible political constraints such as a change of government
Market	Relating to issues such as competition and supply of goods
Act of God	Relating to issues such as fire, flood earthquake
Financial	
Budgetary	Relating to the availability of resources or the allocation of resources
Fraud or Theft	Relating to the unproductive loss of resources
Insurable	Relating to the potential areas of loss which can be insured against
Capital Investment	Relating to the making of appropriate investment decisions
Liability	Relating to the right to sue or be sued in certain circumstances
Activity	
Policy	Relating to the appropriateness and quality of policy decisions
Operational	Relating to the procedures employed to achieve particular objectives
Information	Relating to the adequacy of information which is used for decision making
Reputational	Relating to the public reputation of the organisation and consequent effects
Transferable	Relating to risks which can be transferred or the transfer of risks at inappropriate cost
Technological	Relating to the use of technology to achieve objectives
Project	Relating to project planning and management procedures
Innovation	Relating to the exploitation of opportunities to make gains
Human Resources	
Personnel	Relating to the availability and retention of staff
Health and Safety	Relating to the well-being of people

Brainstorming or SWOT Analysis

Within SWOT analysis risks are identified by staring at the Strengths (or perceived Strength), Weakness, Opportunity & Threats to the success of the project or program. This is usually done within a Workshop environment.

Facilitated Workshops

A facilitated workshop is a structured approach that allows a group of people to work together to reach a predetermined objective. The participants explore risks and make decisions in the workshop. A skilled facilitator to ensure that prearranged objectives are achieved supports these workshops. Facilitated workshops can be used at any time in a project. It is up to the project team to decide whether a workshop is necessary, or whether another technique, such as interviewing or research is more applicable.

Notes Used properly, facilitated workshops are a useful tool for effecting cultural change in an organization because they promote buy-in and empowerment of participants.

Interviews

Interviews conducted to recognize risk will only be successful where there is:

Notes

- Good preparation;
- Clear objectives;
- A positive & supportive environment;
- Proper time management;
- Use of open questions; and
- Challenging not confrontational atmosphere.

The results of interviews should be well documented.

Did u know? What is SWOT?

SWOT analysis (alternately SLOT analysis) is a strategic planning method used to evaluate the Strengths, Weaknesses/Limitations, Opportunities, and Threats involved in a project or in a business venture.

Caselet

'Project Antariksha' Plans Climate Atlas of Kerala

Vinson Kurian'

Thiruvananthapuram, March 17 Project Antariksha, a pioneering effort at networking Kerala through an array of satellite-based automatic weather stations, intends to bring out a weather and climate atlas of the State in the near future.

It is a collaborative project among the Indian Space Research Organisation, the Kerala State Planning Board and the Centre for Monsoon Studies of the Cochin University for Science and Technology (Cusat).

Two years into implementation, the project has seen 58 automatic weather stations being set up over Kerala and two on the Lakshadweep Islands.

Project Antariksha seeks to reach real-time data, including rainfall, on a Web-based system to reach the user groups. Advisories on precipitation and temperature anomalies would be issued to the State Government.

Workshops planned

List of the districts and the respective number of AWSs (in brackets) they house is as follows: Alappuzha and Thiruvananthapuram (4 each); Ernakulam (8); Idukki, Kannur and Wynad (6 each); Kasargod, Malappuram, Pathanamthitta and Kozhikode (3 each); Kollam and Kottayam (1 each); Palakkad and Thrissur (5 each).

Interactive workshops are being planned at the district-level to make people aware of the changing weather conditions and on how to use the AWSs as a tool for the purpose. Representatives of local bodies, political organisations, school/college students, teachers and local NGOs would be target groups.

Steps have already been initiated by the Kerala Agricultural University to develop suitable techniques for weather data analysis with region-specific models to generate weather advisories for various applications such as agriculture, irrigation and water management.

Contd.....

Notes

Using the latest GIS software, the hourly meteorological data received will be converted into user-friendly format for short and long-term planning by various departments under the State Government.

AWS data collected, quality checked and thus converted and pertaining to various districts is available for year 2007. Steps are being taken to ensure uninterrupted reception of quality data. A few more AWSs are planned to be installed at the locations where the need is being felt. – V.K.

Self Assessment

Fill in the blanks:

14. A project to develop a new commercial airliner may require numerous ……..
15. The projects are always planned considering …….. constraints.

2.7 Summary

- Planning is the trickiest process in project management.
- Project scope is the total amount of work done to attain a project management team's project.
- Scope planning refers to a project management process that describes boundaries and deliverables.
- A work package is a complete description of how the tasks that make up an activity will actually be done.
- Verification is a quality guarantying procedure or technique applied by Project Management.
- A specific objective increases the chances of leading to a specific outcome.
- Project infrastructure is about analysis the whole environment and finding out how to put in place an environment that will work in an integrated manner to support projects.
- Structure refers to how a team is organized to be most effectual.
- Effort estimation contains in predict how many hours of work and how many workers are needed to develop a project.

2.8 Keywords

CBT: Computer Based Training

LAN: Local Area Network

POS: Project Overview Statement

2.9 Review Questions

1. A project management process is not a collection of guide or templates. Discuss
2. Techniques are the general, reusable process that an organization expands, or that an organization subscribes to. Analyze
3. What are checklists? Explain the various points that it covers.

Notes

4. The project scope contains the various features. Explain in detail.
5. Internal deliverables are things the project makes internally. In this statement what type of things they are talking about? Explain
6. Completion of the project scope is calculated against the project plan, but completion of the product scope is measured against the product requirements". Justify
7. Defining the data is comparable to functionality, and should be employed in conjunction with functionality. Explain how?
8. Explain the aspects responsible for the project infrastructure.
9. Explain the various project objectives.
10. Discuss the creation of WBS (Work Breakdown Structure).

Answers: Self Assessment

1.	people	2.	cost
3.	Deliverables	4.	Change
5.	Development	6.	task
7.	data	8.	depth
9.	numerical	10.	templates
11.	reusable	12.	timescale
13.	PDF	14.	prototypes
15.	time		

2.10 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://en.wikipedia.org/wiki/Project_planning

http://www.mindtools.com/pages/main/newMN_PPM.htm

Notes

Unit 3: Work Breakdown Structure

CONTENTS

Objectives

Introduction

3.1 Uses for the WBS

3.2 Generating the WBS

3.2.1 Top-down Approach

3.2.2 Bottom-up Approach

3.3 Criteria to Test for Completeness in the WBS

3.4 Using a Joint Project Planning Session to Build the WBS

3.5 Approaches to Building the WBS

3.5.1 Noun-type Approaches

3.5.2 Verb-type Approaches

3.5.3 Organizational Approaches

3.6 Representing the WBS

3.7 Summary

3.8 Keywords

3.9 Review Questions

3.10 Further Readings

Objectives

After studying this unit, you will be able to:

- Discuss the Work Breakdown Structure
- Explain the approaches to building the Work Breakdown Structure
- Determine which of the approaches to use for generating the Work Breakdown Structure for a given project
- Generate a complete Work Breakdown Structure
- Use a Joint Project Planning session to generate a Work Breakdown Structure
- Explain top-down versus bottom-up processes for building the Work Breakdown Structure in the Joint Project Planning session

Introduction

In this unit we will study about the Work Breakdown Structure (WBS). WBS is a hierarchical description of the work that must be done to complete the project. Several processes can be used to create this hierarchy, which we discuss in this unit.

Notes

3.1 Uses for the WBS

The WBS has four uses:

1. ***Thought process tool:*** The manager and the project team use WBS as a thought process tool that helps them to visualize exactly how the work of the project can be defined and managed effectively.
2. ***Architectural design tool:*** WBS can be used as a design tool as it is a picture of the work of the project that shows how the items of work are related to one another.
3. ***Planning tool:*** In the planning phase, the WBS gives the project team a detailed representation of the project as a collection of activities that must be completed in order for the project to be completed. It is at the lowest activity level of WBS that we will estimate effort, elapsed time, and resource requirements; build a schedule of when the work will be completed; and estimate deliverable dates and project completion.
4. ***Project status reporting tool:*** The WBS is used as a structure for reporting project status. The project activities are consolidated (that is, rolled up) from the bottom as lower-level activities are completed. As work is completed, activities will be completed. Completion of lower-level activities causes higher-level activities to be partially complete. Some of these higher-level activities may represent significant progress whose completion will be milestone events in the course of the project.

Task The WBS defines milestone events that can be reported to senior management and to the customer. Analyze

WBS and the Project Manager

It is the project manager who decides on the architecture of the WBS and the level of detail required. This detail is important because the project manager is accountable for the success of the project. The WBS must be defined so that the project manager can manage the project. Apart from any senior management requirements for reporting or organizational requirements for documentation or process, the project manager is free to develop the WBS according to his or her needs and those of management.

3.2 Generating the WBS

The best way to generate the WBS is as part of the Joint Project Planning (JPP) session. One of two simple decomposition processes is used to identify the activities that must be performed from the beginning to the completion of the project. These activities are the lowest level of managed work for the project manager. At this point in the planning process, you should have completed the Project Overview Statement. You may have to go back and reconsider the POS as a result of further planning activities, but for now let's assume the POS is complete. Our technique for generating the WBS will reduce even the most complex project to a set of clearly defined activities. The WBS will be the document that guides the remainder of the planning activities.

Two approaches can be used to identify the project activities:

- The first is the top-down approach.
- The second is the bottom-up approach.

Notes

3.2.1 Top-down Approach

The top-down approach begins at the goal level and successively partitions work down to lower levels of definition until the participants are satisfied that the work has been sufficiently defined.

Once the project activities have been defined using the top-down approach, they will be defined at a sufficient level of detail to allow you to estimate time, cost, and resource requirements first at the activity level and then aggregate to the project level.

Caution Because the activities are defined to this level of detail, project time, cost, and resource requirements are estimated much more accurately.

Once the activities are described, you can sequence the project work so that as many activities as possible are performed in parallel, rather than in sequence.

There are two variations of the top-down approach:

- Team approach
- Sub-team approach

Team Approach

In this approach the entire team works on all parts of the WBS. For each Level 1 activity, appoint the most knowledgeable member of the planning team to facilitate the further decomposition of that part of the WBS. Continue with similar appointments until the WBS is complete. This approach allows all members of the planning team to pay particular attention to the WBS as it is developed, noting discrepancies and commenting on them in real time.

The team approach is better then the sub-team approach but it requires more time to complete.

Sub-team Approach

When time is at a premium, the planning facilitator will prefer the sub-team approach. The first step is to divide the planning team into as many sub-teams as there are activities at Level 1 of the WBS. Then follow these steps:

1. The planning team agrees on the approach to building the first level of the Work Breakdown Structure.
2. The planning team creates the Level 1 activities.
3. A subject matter expert leads the team in further decomposition of the WBS for his or her area of expertise.
4. The team suggests decomposition ideas for the expert until each activity within the Level 1 activities meets the WBS completion criteria.

Did u know? What is decomposition?

Decomposition (or rotting) is the process by which organic material is broken down into simpler forms of matter.

Notes

3.2.2 Bottom-up Approach

Bottom-up approach is more like an organized brainstorming session used to build the WBS. It works as follows.

The first steps are the same as those for the top-down approach. Namely, the entire planning team agrees to the first-level breakdown. The planning team is then divided into as many groups as there are first-level activities. Each group then makes a list of the activities that must be completed in order to complete the first-level activity. To do this, they proceed as follows:

1. Someone in the group identifies an activity and announces it to the group. If the group agrees, then the activity is written on a slip of paper and put in the middle of the table. The process repeats itself until no new ideas are forthcoming.
2. The group then sorts the slips into activities that seem to be related to one another. This grouping activity should help the planning team add missing activities or remove redundant ones.
3. Once the team is satisfied it has completed the activity list for the first-level breakdown, the members are finished. Each group then reports to the entire planning team the results of its work.
4. Final critiques are given, missing activities added, and redundant activities removed.

Self Assessment

Fill in the blanks:

1. It is at the lowest activity level of WBS that we will estimate effort, ……. , and resource requirements.
2. The best way to generate the WBS is as part of the …… session.
3. The project activities have been defined using the ……. approach.
4. Continue with similar appointments until the WBS is …….
5. The team approach is better than the ……. but it requires more time to complete.
6. The planning team is then divided into as many groups as there are …….. activities.
7. Once the team is satisfied it has completed the activity list for the first-level breakdown, the members are ……...
8. The technique for generating the WBS will …….. even the most complex project to a set of clearly defined activities.

3.3 Criteria to Test for Completeness in the WBS

Developing the WBS is the most critical part of the JPP. If we do this part right, the rest is comparatively easy. How do you know that you've done this right? Each activity must possess six characteristics to be considered complete—that is, completely decomposed. The six characteristics are as follows:

- Status/completion is measurable.
- Start/end events are clearly defined.

Notes

- Activity has a deliverable.
- Time/cost is easily estimated.
- Activity duration is within acceptable limits.
- Work assignments are independent.

If the activity does not possess these six characteristics, decompose the activity and ask the questions again. As soon as an activity possesses the six characteristics, there is no need to further decompose it. As soon as every activity in the WBS possesses these six characteristics, the WBS is defined as complete. The following unit look at each of these characteristics in more detail.

Measurable Status

The project manager can ask for the status of an activity at any point in time during the project. If the activity has been defined properly, that question is answered easily. For example, if a system's documentation is estimated to be about 300 pages long and requires approximately four months of full-time work to write, here are some possible reports that your activity manager could provide regarding the status:

1. Let's see, the activity is supposed to take four months of full-time work. I've been working on it for two months full-time. I guess I must be 50 per cent complete.
2. I've written 150 pages, so I guess I am 50 per cent complete.
3. I've written, and had approved, 150 pages and estimate that the remaining work will require two more months. I am 50 per cent complete.

The answer (1) is not acceptable, as it does not indicate that after working for two-months full time, the half work is completed. The answer (2) is a little better but it doesn't say anything about the quality of the 150 pages that have been written, nor does it say anything about the re-estimate of the remaining work. And so we see that an acceptable answer must state what has been actually completed (approved, not just written, in our example) and what remains to be done, along with an estimate to completion. Therefore, after working through about half of the activity, the activity manager should be able to give a very accurate estimate of the time required to complete the remaining work.

A simple metric that has met with some success is to compute the proportion of tasks completed as a percentage of all the tasks that make up the activity. For example, suppose the activity has six tasks associated with it and four of the tasks are complete; the ratio of tasks complete to total tasks is 4/6, that is, the activity is 66 per cent complete. Even if work had been done on the fifth task in this activity, because the task is not complete on the report date, it cannot be counted in the ratio. This metric certainly represents a very objective measure. Although it may seem somewhat inaccurate, it is a good technique. Best of all, it is quick. Project manager and activity manager do not have to sit around mired in detail about the percentage complete. This same approach can be used to measure the earned value of an activity.

Bounded

Each activity should have a clearly defined start and end event. For example, using the systems documentation example, the start event might be notification to the team member who will manage the creation of the systems documentation that the final acceptance tests of the system are complete. The end event would be notification to the project manager that the customer has approved the system documentation.

Notes

Deliverable

The result of completing the work that makes up the activity is the production of a deliverable. The deliverable is a visible sign that the activity is complete. This sign could be an approving manager's signature, a physical product or document, the authorization to proceed to the next activity, or some other sign of completion.

Cost/Time Estimate

Each activity should have an estimated time and cost of completion. Being able to do this at the lowest level of decomposition in the WBS allows you to aggregate to higher levels and estimate the total project cost and the completion date. By successively decomposing activities to finer levels of granularity, you are likely to encounter primitive activities that you have performed before. This experience at lower levels of definition gives you a stronger base on which to estimate activity cost and duration for similar activities.

Acceptable Duration Limits

While there is no fixed rule for the duration of an activity, it is recommended that activities should have a duration of less than two calendar weeks. This seems to be a common practice in many organizations. Even for long projects where contractors may be responsible for major pieces of work, they will generate plans that decompose their work to activities having this activity duration.

Activity Independence

It is important that each activity be independent. Once work has begun on the activity, it can continue reasonably without interruption and without the need of additional input or information until the activity is complete. The work effort could be contiguous, but it can be scheduled otherwise for a variety of reasons.

Notes You can choose to schedule it in parts because of resource availability, but you could have scheduled it as one continuous stream of work.

3.4 Using a Joint Project Planning Session to Build the WBS

The best way to build a WBS is as a group activity. To create the WBS, assemble a facilitator, the project manager, the core members of the project team, and all other managers who might be affected by the project or who will affect the project. The important thing is to have the expertise and the decision makers present in this part of the planning session who can give input into the WBS. This exercise should be continuous. The steps to build WBS are as follows:

- The first step is for the whole planning team to decide on the first-level decomposition of the goal statement. One obvious approach would be to use the objective statements from the POS as the first-level decomposition.
- Once the first-level decomposition is developed, the team has two choices on how to proceed:
    - Without a doubt, the best way (from a WBS completion point of view) is to have the entire planning team remain intact and complete the WBS together. Often that will not work simply from the standpoint of it taking more of everybody's time and also tying up the time of several high-level managers more than in the second choice.

Notes

- The second choice is to divide the planning team into groups and let each group take one or more of the first-level activities and complete the WBS for that part only.

- Whichever approach is used, it is essential that the entire group have the opportunity to review the final WBS and offer critiques of it. It is important that these efforts be made, because the WBS must be complete, and all the members of the team will see the final result of the JPP.

Self Assessment

Fill in the blanks:

9. Each activity should have a clearly defined start and ……..... event.

10. Developing the WBS is the most critical part of the …….....

3.5 Approaches to Building the WBS

There are many ways to build the WBS but none can be said as the correct one. Hypothetically, if we put each member of the JPP session in a different room and ask that person to develop the project WBS, they might all come back with different renditions, as there is no single best answer. The choice is subjective and based more on the project manager's preference than on any other requirements.

There are three general approaches to building the WBS:

- ***Noun-type approaches:*** Noun-type approaches define the deliverable of the project work in terms of the components (physical or functional) that make up the deliverable.

- ***Verb-type approaches:*** Verb-type approaches define the deliverable of the project work in terms of the actions that must be done to produce the deliverable. Verb-type approaches include the design-build-test-implement and project objectives approaches.

- ***Organizational approaches:*** Organizational approaches define the deliverable of the project work in terms of the organizational units that will work on the project. This type of approach includes the department, process, and geographic location approaches.

3.5.1 Noun-type Approaches

There are two noun-type approaches:

(a) ***Physical Decomposition:*** In projects that involve building products, it is tempting to follow the physical decomposition approach. Take a mountain bike, for example. Its physical components include a frame, wheels, suspension, gears, and brakes. If each component is to be manufactured, this approach might produce a simple WBS. As mentioned previously, though, you have to keep in mind the concern of summary reporting.

(b) ***Functional Decomposition:*** Using the bicycle example, we can build the WBS using the functional components of the bicycle. The functional components include the steering system, gear-shifting system, braking system, and pedaling system. The same cautions that apply to the physical decomposition approach apply here as well.

Task What have you understood from physical and functional decomposition? Write the difference in your words.

Notes

3.5.2 Verb-type Approaches

There are two verb-type approaches:

(a) ***Design-build-test-implement:*** The design-build-test-implement approach is commonly used in those projects that involve a methodology. Application systems development is an obvious situation. Using our bicycle example again, a variation on the classic waterfall categories could be used. The categories are design, build, test, and implement.

Remember, the WBS activities at the lowest levels of granularity must always be expressed in verb form. After all, we are talking about work, and that implies action, which, in turn, implies verbs.

(b) ***Objectives:*** The objectives approach is similar to the design-build-test-implement approach and is used when progress reports at various stages of project completion are prepared for senior management. Reporting project completion by objectives gives a good indication of the deliverables that have been produced by the project team. Objectives will almost always relate to business value and will be well received by senior management and the customer as well. There is a caveat, however. This approach can cause some difficulty because objectives often overlap. Their boundaries can be fuzzy.

Caution You'll have to give more attention to eliminating redundancies and discovering gaps in the defined work.

3.5.3 Organizational Approaches

The deployment of project work across geographic or organizational boundaries often suggests a WBS that parallels the organization. The project manager would not choose to use this approach but rather would use it out of necessity. In other words, the project manager had no other reasonable choice. These approaches offer no real advantages and tend to create more problems than they solve. We list them here only because they are additional approaches to building the WBS.

(a) ***Geographic:*** If project work is geographically dispersed (our space program, for example), it may make sense from a coordination and communications perspective to partition the project work first by geographic location and then by some other approach at each location.

(b) ***Departmental:*** On the other hand, departmental boundaries and politics being what they are, we may benefit from partitioning the project first by department and then within department by whatever approach makes sense. We benefit from this structure in that a major portion of the project work is under the organizational control of a single manager. Resource allocation is simplified this way. On the other hand, we add increased needs for communication and coordination across organizational boundaries in this approach.

(c) ***Business Process:*** The final approach involves breaking the project down first by business process, then by some other method for each process may make sense. This has the same advantages and disadvantages as the departmental approach but the added complication that integration of the deliverables from each process can be more difficult than in the former case.

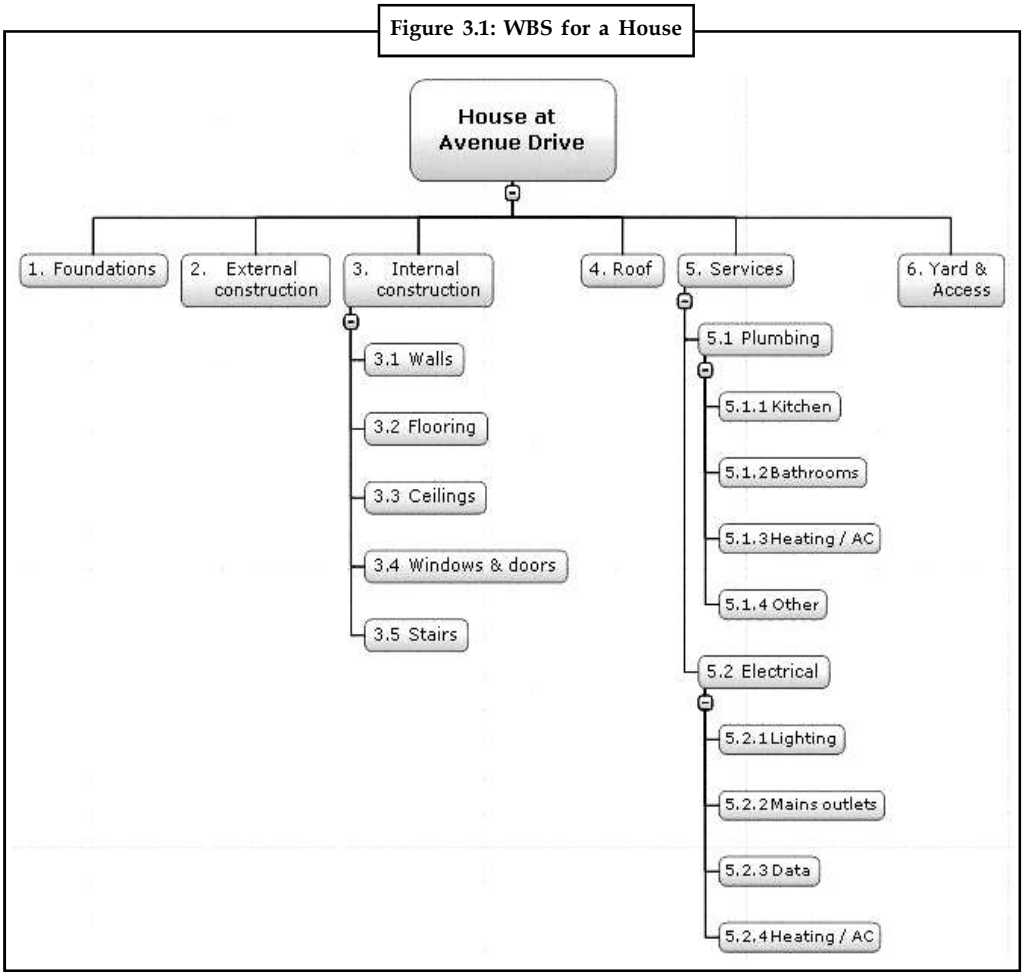
Did u know? What are organizational boundaries?

Notes

3.6 Representing the WBS

Whatever approach you use, the WBS can be generically represented, as shown in Figure 3.1. The goal statement represents the reason for doing the project.

Figure 3.1: WBS for a House



The Level 1 partitioning into some number of activities (also known as chunks of work) is a necessary and sufficient set of activities. That is, when all of these first-level activities are complete, the project is complete. For any activity that does not possess the six characteristics, we partition it into a set of necessary and sufficient activities at Level 2. The process continues until all activities have met the six criteria. The lowest level of decomposition in the WBS defines a set of activities that will each have an activity manager, someone who is responsible for completing the activity.

The lowest-level activities are defined by a work package. A *work package* is simply the list of things to do to complete the activity. The work package may be very simple, such as getting management to sign off on a deliverable.

Notes A work package may be a mini project and may consist of all the properties of any other project, except that the activity defining this project possesses the six criteria and need not be further partitioned.

Self Assessment

Notes

Fill in the blanks:

11. The ..... is a hierarchical description of the work that must be done to complete the project as defined in the Project Overview Statement (POS).
12. A ..... is a complete description of how the tasks that make up an activity will actually be done.
13. Breaking down work into a hierarchy of activities, tasks, and work packages is called .....
14. .....approach and ..... approach are the two variations of the top-down approach.
15. There are ..... general approaches to building the WBS.

Case Study

Blogs are designed for People who don't want to learn

Get online. Create your blog. Grow your own business.

These are the mantras that Andy Wibbels offers in Blogwild! from Nicholas Brealey (www.nbrealey-books.com). "Blogwild is for businesses and entrepreneurs who have heard about blogs, know that there's something special going on, but aren't quite sure what the big deal is," notes the introduction. So, first, `what is a blog?'

Short for Web log; it is an online diary, explains the author. "For the elegant simplicity and beauty that blogs create, they certainly have an ugly, ungainly name," he says.

Blawg (the way it is pronounced) sounds like "some kind of gurgling sound your cat makes prior to coughing up a giant hairball."

Yuk! But wait. "Blogs are designed for people who don't want to learn (or don't have the time to learn)." It is as simple as sending an e-mail, entices the author, before giving his updated description of blog thus: "An easily, instantly, and frequently updated Website, focussed around a topic, industry, or personality."

Start reading blogs, exhorts Wibbels, because that's "the easiest way to get acquainted with the conversations of the blog format".

Gradually, you'd pick up new words. Such as: 'blogosphere', which is 'the collective hive of blogs'; and 'blogroll', which is a list of the blogger's favourite blogs and Websites.

Get to know how to use blog search engines and directories such as Blogwise, Blogarama and Globe of Blogs.

'Blog ecosystems' are sites that monitor millions of blogs every minute of every day, "searching for the top key words being mentioned in blog posts or the most popular links, books, movies, and people."

Examples are Blogdex, IceRocket, BlogPulse and Technorati. Wibbels likens visiting these sites to "being in a huge conventional hall eavesdropping on millions of conversations at the same time." A fantastic way to do market research and to see what's buzzing, he says.

Contd.....

Notes

"Blogging provides a way for companies and customers to meet on common ground to talk about what excites them and what makes them tick. It also allows them to closely track where, and under what circumstances, their products are being talked about online."

Each day, seventy thousand new blogs are created, informs Wibbels. "More than a million posts are added every single day. Millions of people look to blogs for a good laugh, a great idea, a fantastic tip, or an instant analysis."

Blog was 'the word of the year' in 2004, according to Merriam-Webster's Collegiate Dictionary. But Blogger (www.blogger.com) had been launched in 1999. "Blogger provided both the blog platform and hosting space for blogs... The world's largest conversation began."

Wibbels sees in blogs the rise of 'citizen journalism', a.k.a. 'consumer-generated media'. This is because blogging supports "the idea that customers and consumers are not passive receptors of media and marketing."

Since bloggers are able to publish text, pictures, audio and video, "often, blogs are the first source of information for breaking events." Like it or not, `blogs drive transparency'.

Disturbingly for mainstream TV or newspapers, "blogs create a greater sense of trust and reputation than traditional media." With barriers gone for consumers to get online, and give vent to praise or criticism, "even grandma can wreck your ad campaign".

3.7 Summary

- The Work Breakdown Structure (WBS) is a hierarchical description of the work that must be done to complete the project as defined in the Project Overview Statement.
- An activity is nothing but simply a chunk of work. Activities turn to tasks at some level in the hierarchy. The terms activity and task have been used interchangeably among project managers and project management software packages.
- A work package is a complete description of how the tasks that make up an activity will actually be done. It includes a description of the what, who, when, and how of the work.
- Breaking down work into a hierarchy of activities, tasks, and work packages is called decomposition. Decomposition is important to the overall project plan because it allows you to estimate the duration of the project, determine the required resources, and schedule the work.
- The WBS can be used as Thought process tool, Architectural design tool, Planning tool and Project status reporting tool. It is the project manager who decides on the architecture of the WBS and the level of detail required.
- The best way to generate the WBS is as part of the Joint Project Planning (JPP) session. One of two simple decomposition processes is used to identify the activities that must be performed from the beginning to the completion of the project.
- These activities are the lowest level of managed work for the project manager. There are two approaches can be used to identify the project activities.
- There are Six Criteria to Test for Completeness in the WBS. The best way to build a WBS is as a group activity.
- To create the WBS, assemble a facilitator, the project manager, the core members of the project team, and all other managers who might be affected by the project or who will affect the project.
- There are many ways to build the WBS but none can be said as the correct one.

Notes

3.8 Keywords

Activity: An activity is nothing but simply a chunk of work.

Decomposition: Breaking down work into a hierarchy of activities, tasks, and work packages is called decomposition.

Task: A task is a smaller chunk of work. The terms activity and task have been used interchangeably among project managers and project management software packages.

Work Breakdown Structure (WBS): WBS is a hierarchical description of the work that must be done to complete the project as defined in the POS.

Work Package: A work package is a complete description of how the tasks that make up an activity will actually be done. It includes a description of the what, who, when, and how of the work.

3.9 Review Questions

1. What is WBS? Explain with the help of example.
2. Differentiate between a task and an activity.
3. What is work package?
4. Why it is important to decompose a WBS?
5. Write down the uses for the WBS.
6. How do you generate WBS? Discuss the different approaches to generate WBS.
7. What are the differences between the top-down approach and bottom-up approach to generate WBS?
8. What are the six criteria to test for completeness in the WBS?
9. What do you mean by activity independence?
10. Write a short note on "Using a Joint Project Planning Session to Build the WBS".
11. Discuss the general approaches to build the WBS.
12. How do you represent a WBS?

Answers: Self Assessment

1. elapsed time
2. Joint Project Planning (JPP)
3. top-down
4. complete
5. sub-team approach
6. first-level
7. finished
8. reduce
9. end
10. JPP
11. WBS
12. Work package
13. Decomposition
14. Team, sub-team
15. three

Notes

3.10 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://en.wikipedia.org/wiki/Work_breakdown_structure

http://www.netmba.com/operations/project/wbs/

Unit 4: Programme Management & Project Evaluation

Notes

CONTENTS

Objectives

Introduction

4.1 Meaning

4.1.1 Programme Management

4.1.2 Project Evaluation

4.2 Managing Allocation of Resources

4.3 Creating Programme

4.3.1 The Golden Thread

4.4 Individual Projects

4.5 Technical Assessment

4.6 Cost Benefit Analysis

4.6.1 Mistakes and Problems with Cost-benefit Analysis

4.6.2 History of Cost Benefit Analysis

4.7 Risk Evaluation

4.8 Summary

4.9 Keywords

4.10 Review Questions

4.11 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize allocation of resources
- Describe creating programs
- Explain the individual projects
- Discuss technical assessment
- Describe cost benefit analysis and risk evaluation

Introduction

Program management or programme management is the procedure of organizing numerous related projects, often with the meaning of improving an organization's performance. In practice and in its aims it is often intimately related to systems engineering.

Project evaluation is an organized and objective measurement of an ongoing or completed project. The aim is to conclude the relevance and level of achievement of project objectives, development

Notes

effectiveness, efficiency, impact and sustainability. Evaluations also feed lessons learned into the decision-making process of the project stakeholders, including donors and national partners.

4.1 Meaning

4.1.1 Programme Management

Programme Management is the procedure of managing numerous ongoing interdependent projects. An instance would be that of designing, manufacturing and providing support infrastructure for an automobile manufacturer. This necessitates hundreds, or even thousands, of different projects. In an organization or enterprise, Program Management also reflects the emphasis on coordinating and prioritizing resources across projects, departments, and entities to make sure that resource contention is managed from a global focus. Five major features of programme management are:

- ***Governance:*** Defining roles and responsibilities, and providing oversight.
- ***Management:*** Planning and administering both projects and the overall program.
- ***Financial Management:*** Implementation of specific fiscal practices and controls.
- ***Infrastructure:*** The program office, technology, and other factors in the work environment supporting the program effort.
- ***Planning:*** Activities that take place at multiple levels, with different goals. The program plan is not a traditional plan.

Program Governance

Program governance is the feature of the discipline that creates both the structure and practices to guide the program and offer senior-level leadership, oversight, and control. Strategically, it encompasses the relationship between the oversight effort and the enterprise's overall business direction.

Notes It also encompasses all the decision-making roles and responsibilities involved in executing the program effort.

Projects are usually governed by a simple management structure. The project manager is responsible for day-to-day direction, a senior IT executive integrates technology with business interests, and a business sponsor is accountable for ensuring that the deliverables align with business strategy.

Programs require a more compound governing structure because they involve fundamental business change and expenditures with important bottom-line impact. In fact, in some instances their outcomes determine whether the enterprise will survive as a viable commercial/ governmental entity.

Example: As we can see in Figure 4.1, unlike most projects, programs generally have a steering committee or other group that represents different interests and provides executive-level oversight. As the program evolves, this governing body ensures that it maintain to align with the enterprise's strategic direction and makes decisions that may eventually filter up to the board of directors. Defining the role and decision-making powers of the steering committee is a significant part of the program governance effort and should be done with an eye toward facilitating rapid decisions and promoting a clear, unified direction.

Notes

Figure 4.1: Sample Program Governance Structure

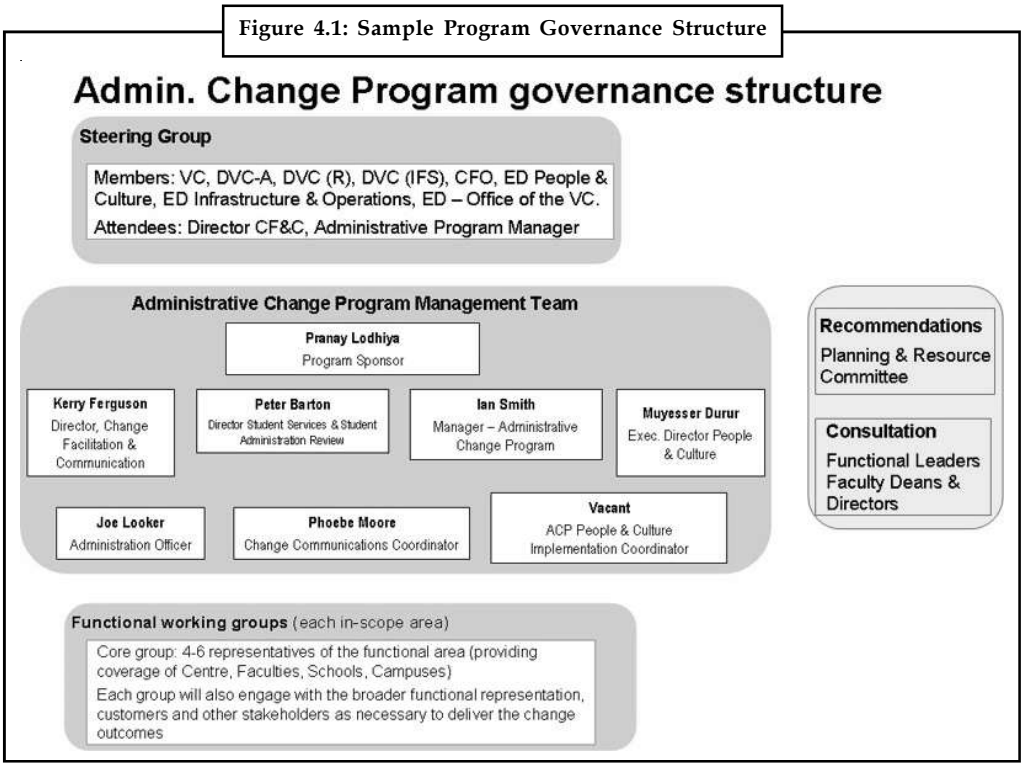


Figure 4.1 also show a typical program management structure, which is more complex than that of a project. Creating this structure engage defining specific roles with specific decision-making authority, and making clear to all who "owns" certain program functions.

Good governance is dangerous to program success. A poorly articulated management structure, overlapping roles and decision-making authority, and roles filled by the wrong people (or not filled at all) can prevent a program from achieving persistent momentum or bog it down with endless attempts to achieve consensus on every decision.

Management

Project management is the planning, organizing, directing, and controlling of corporation resources for a comparatively short-term purpose. It is clear from this description that project management is concerned with the dynamic allocation, utilization, and direction of resources (both human and technical), with time — in relation to both individual efforts and product delivery schedule — and with costs, relating to both the acquisition and consumption of funding. As a corollary, it is safe to say that without the direction project management give, work would have to go on via a series of negotiations, and/or it would not align with the goals, value proposition, or needs of the enterprise.

Within a program, these same errands (i.e., allocation, utilization, and direction) are allocated to people at three levels in the management hierarchy; the higher the level, the more common the responsibilities. For example, at the bottom of the management hierarchy, project managers are assigned to the different projects within the overall program. Each manager carries out the management responsibilities we described above.

At the core or middle of the hierarchy is the program manager/director, 2 whose major responsibility is to ensure that the work effort attains the outcome specified in the business and IT strategies. This involves setting and reviewing objectives, coordinating activities across projects, and controlling the integration and reuse of interim work products and results. This person spends more time and effort on integration activities, negotiating changes in plans, and

Notes

communicating than on the other project management activities we described (e.g., allocating resources, ensuring adherence to schedule, budget, etc.).

At the top of the program management hierarchy are the program sponsor(s) and the program steering committee. Their main responsibility is to own and oversee the implementation of the program's underlying business and IT strategies, and to define the program's connection to the enterprise's overall business plan(s) and direction. Their management activities contain providing and interpreting policy, creating an environment that fosters sustainable momentum for the program (i.e., removing barriers both inside and outside the enterprise), and occasionally reviewing program progress and interim results to ensure alignment with the overall strategic vision.

These individuals receive periodic summary reports and briefings on funding consumption, resources and their utilization, and delivery of interim work products and results. Typically, they will focus on these reports only if there is significant deviation from the plan.

Program Financial Management

The financial feature of a program comprises the need to conform to internal (and sometimes external) policies and/or regulations for significant expenditures. It also includes development and use of program-specific procedures for making and reporting expenditures.

Overall costs for programs are usually considerably greater than those for projects. For example, projects that consume one to five man-years of attempt might have an internal cost range of $250,000 to $1,000,000, assuming the resources are employees (not contractors) with an hourly charge-back rate of $100 to $150 per hour. A program to upgrade and rewrite the core software applications of a large financial services company might require between 750,000 and 1,000,000 work hours, a staff of 175 consultants and 225 employees, and expenses ranging between $160,000,000 and $200,000,000.

The costs are bigger not only because the program is larger, but also as it entails more types of expenditures. In a project of the size we just explained, most — if not all — the expenditures are for labor, from an accountancy viewpoint. The program costs would include labor (both internal chargeback and consulting fees, and travel and living expenses, including short-term apartment leases), hardware, packaged software applications (which may be capitalized and depreciated), work space (perhaps construction, too), and furnishings/equipment, such as computers, servers, printers, desks, chairs, cubicles, and so on. Enterprises have different ways to treat these expenditures, outlined in financial policies and procedures. Government agencies and regulated industries may also have laws or regulations regarding spending and expense reporting.

From an administrative point of view, the responsibilities associated with authorizing, recording, and reporting program expenditures go well beyond those typically exercised by an individual project manager. Typically, the office of the Chief Financial Officer (CFO) will be involved during the strategic definition and financial justification phases of a program. Financial analysts will construct and/or use complex financial models, see that the enterprise's financial policies are interpreted and applied correctly, and ensure that the program's financial impact is accurately represented to executives at key decision points.

The CFO's engagement will continue, with different responsibilities, throughout the program's lifecycle. The program office will typically include a role for a budget administrator who assists the program manager/director in ensuring conformance to financial policies and guidelines. A best practice requires the CFO to fill this role with a full-time or part-time financial analyst.

Early in the program, you should plan and conduct a checkpoint review of the financial management apparatus and identify needs and requirements that are specific to the program.

Implementing the program's financial practices may require nothing more than educating people about how to apply them. However, in some instances you may need to tailor and adopt policies, create new cost centers and/or a chart of accounts, and outline financial procedures and assign decision authority unique to the specific program.

Notes

In any case, the skills required to create and ensure program-wide application of sound financial practices are typically not required for a project effort.

Caution To succeed, program financial management demands early and active engagement on the part of the CFO and his or her staff.

Program Infrastructure

Infrastructure is a useful term to describe collections of roles, tools, and practices that organizations assemble and integrate in order to provide services and support for software development. To understand the infrastructure required for a successful program, let's first explore the management and administrative roles, tools, and practices that constitute the Program Management Office, or PMO. Then we will look at requirements for the technical environment and tools.

Administrative Infrastructure

Of course, simply creating and operating a PMO — which can assume many forms — differentiates programs from projects. Our discussion will focus primarily on PMOs that support a single program — one that will be disbanded at the close of the program effort. However, we should keep in mind that in some IT organizations, an Enterprise PMO is a permanent fixture, providing services to multiple (and changing) programs.

The PMO provides administrative and management support to the program manager/director and all other program participants. It also provides specialized staff expertise for specific work areas.

The PMO involves many roles covering numerous areas and activities. In addition to serving the program manager/director, the staff members, a group of senior specialists, fill essential program roles. For large, complex programs, the PMO helps establish and maintain appropriate work processes, controls, and reporting functions to keep management apprised of the program's progress. It also defines, plans, and completes various work efforts.

As an example, let's examine just one role in the PMO — facilities administration — and how it contributes to program success. Whoever takes on this role must identify, plan, and deliver all necessary facilities for either a program-specific or permanent PMO. To do this, the facilities administrator must:

- Work with the PMO manager and program manager to define what should be included in facilities and define and prioritize facility needs.
- Develop and gain approval for a facilities plan.
- Manage execution of the facilities plan and associated deliveries, construction, and installation.
- Collaborate closely with the infrastructure and technical environment coordinator.

Let's compare the value of this role within a project versus a program. For a single, small project with a maximum of seven employees on the construction team, this role would add little value. The team members would likely have offices or cubicles and the ability to reserve meeting rooms through a reservation system.

But suppose you have a program for which mobilization will take four weeks. Over this period, 200 consultants are to become resident at the principal office campus, 260 IT staff will be assigned

Notes

to the program, and the three project managers insist that their project teams be co-located for efficiency. There is a need for twelve dedicated photocopiers, five dedicated conference rooms, a space for the program office, and so forth. The daily billing rate for all 200 consultants is $360,000. So if the operation of these facilities is delayed by five days and the consultants cannot work on site — well, you can do the math. Clearly, someone must "own" the responsibility to set up the proper space and tools to get the job done.

In truth, we could devote an entire article to the work performed by the PMO — and that office does not even cover every responsibility. For now, let us just say that the infrastructure the PMO provides enables all the project teams involved in the program to be productive.

Technical Environment and Tools

A program infrastructure also includes both hardware — for desktop and network devices for storage and communication — and software, including desktop software and shared platforms with development tools, modeling software, planning tools, communication tools (email, Internet browser, virtual meeting/collaboration programs, telecommunications programs), and software for document retention and reproduction.

An individual project, especially a pioneering effort, may introduce new tools or hardware partly in order to understand their capabilities and limitations. The project manager may become involved in technical support or infrastructure functions, to acquire, install, and/or "tune" the hardware and software. Typically, this will involve a small number of installations for a small number of IT staff. Periodic changes and/or additions to the development environment will affect larger numbers of IT staff, but these are typically defined and managed as separate projects.

Program technical activities, in contrast, usually include large numbers of staff from a variety of sources (internal and external) and various technology backgrounds. As managers identify and staff component projects in the program, they must also specify, acquire, and install technology environments and tools for each project, which collectively form the program's technical infrastructure. This effort might encompass creating a new, remote development site or integrating two companies' technologies following a merger, for example.

This infrastructure effort should be treated as an internal program project (as opposed to an external project, which delivers components or results to clients). Managers should plan a well-defined, rapid, and brief lifecycle for creating the technology environment. The effort should include defining needs and requirements, setting a scope, and installing, testing, and implementing all technologies. If some tools will be new to some portion of the program staff, it may also be necessary to define a rapid-delivery training effort.

Managers should also consider how the infrastructure's hardware and tools will be used beyond the program's boundaries. If they felt compelled to select technologies different than those in the current enterprise IT architecture, then supporting and maintaining new software applications built with those technologies may require additional personnel, software, and training. Managers should always carefully evaluate the potential impact of their program technology selections upon existing IT architecture and resources (and perhaps future direction) before actually making the acquisitions.

Task The PMO involves many roles covering numerous areas and activities. What are areas and activities they are talking about?

Notes

Program Planning

For program planning, most managers will typically use a bottom-up approach that identifies and executes planning iterations for the program's individual component projects. First, each project manager constructs a plan that estimates and allocates resources required to deliver the project's products or results, using the same techniques and practices they would employ in planning a standalone project.

Then, in the next planning iteration, managers identify connections and dependencies among the program's projects, and refine and rework their project plans to integrate them with others. Often this integration effort requires adjustments to the products planned for each project, the numbers and types of resources required, and — naturally — the schedule. The managers' ability to continuously manage and adjust to inter-project dependencies is a significant determinant of program success. This ability is also a major differentiator between the requirements of project planning and program planning.

The Program Plan

Once the individual project plans are integrated, it is time to initiate the program planning effort. What exactly is a program plan? American Heritage Dictionary defines a plan as "A scheme, program, or method worked out beforehand for the accomplishment of an objective: a plan of attack." But when we look at how we develop and use program plans, we discover that they do not fit neatly into this definition.

First of all, in contrast to the planning for the program's projects, the program plan typically is not developed through a series of iterations. Instead, the planning effort involves conducting a series of reviews of the individual project plans, and then creating a digest of their contents. During this process, conflicts between projects may become apparent and require resolution. A goal of the digest effort is to produce a concise, usable view of all program work, timeframes, and required results. A program plan describing 10,000 activities, for example, would not have these qualities.

You don't use the program plan to direct work and allocate resources. That is the purpose of the individual project plans. It may be helpful to think of the program plan as a seismograph that seeks to detect and measure the potential impact of any trembling in the ground underneath the program effort. As component projects proceed and individual project plans record completion percentages, expenditure of resources, and interim (or final) dates for work activities, the program plan integrates these measures and shows their collective impact. This enables managers to assess the program's progress against plan and detect potential problems. For example, if a client asks for additional functionality in a component that one project is building, that may delay the component's delivery to other projects and slow them down as well.

In short, the program plan's integrated representation of significant planned activities and results of individual projects provides managers with a window into the cumulative work effort of the program. Managers use it to verify that the program is moving in the right direction to meet business goals, identify where unplanned changes may be occurring and assess their potential impact, and to model and/or test the impact of possible adjustments and corrections.

4.1.2 Project Evaluation

Project Evaluation is a step-by-step process of collecting, recording and organizing information about project results, including short-term outputs (immediate results of activities, or project deliverables), and immediate and longer-term project outcomes (changes in behaviour, practice or policy resulting from the project).

Notes

Common rationales for conducting an evaluation are:

- response to demands for accountability;
- demonstration of effective, efficient and equitable use of financial and other resources;
- recognition of actual changes and progress made;
- identification of success factors, need for improvement or where expected outcomes are unrealistic;
- validation for project staff and partners that desired outcomes are being achieved

The project planning stage is the best time to identify desired outcomes and how they will be measured. This will guide future planning, as well as ensure that the data required to measure success is available when the time comes to evaluate the project.

Why is Project Evaluation important?

Evaluating project results is helpful in providing answers to key questions like:

- What progress has been made?
- Were the desired outcomes achieved? Why?
- Are there ways that project activities can be refined to achieve better outcomes?
- Do the project results justify the project inputs?

What are the Challenges in Monitoring and Evaluation?

- getting the commitment to do it;
- establishing base lines at the beginning of the project;
- identifying realistic quantitative and qualitative indicators;
- finding the time to do it and sticking to it;
- getting feedback from your stakeholders;
- reporting back to your stakeholders.

Task In your words discuss with your class mates that what project evaluation is important?

4.2 Managing Allocation of Resources

Powerful resource allocation is what sets Project Insight, project management software, apart from other solutions. Project managers can effectively monitor and balance resource workloads across multiple projects from one report.

Graphical displays give you visibility over the entire organization to ensure that resources have the right amount of work now, and in the future. With Project Insight technology, you are always in control and ahead of the curve.

- Assign team members to projects by group, department or company
- Mass assign team members to tasks by skill set
- Swap resources on tasks with just a few mouse clicks

Notes

- View resource allocation on work scheduled across all projects
- Drill down to see projects and tasks in resource allocation reports
- Use drag and drop to balance workload of under and over burdened resources
- Suggest the most available resource by skill set
- Forecast and run 'what if' scenarios on projects in planning phase
- Set default burden and bill rates at the system, project, or task level

Project Insight gives project managers power over the management of resource allocation for software development, marketing, product development teams and more. Assigning team members to business goals, projects and individual tasks is simple and easy with our PMI and PMBOK® Guide compliant solution. Mass assign team members' tasks grouped by skill set, department or resource type, or handle resource allocation management for a single person. It is equally simple to change a resource on a set of project tasks as well.

Our portfolio system allows resource allocation managers and project managers to use project level and/or cross project resource allocation to manage workloads in order to achieve their goals. The software application reports evenly divide the work (hours) among the workdays (duration) scheduled for the tasks to calculate the total work or effort assigned to a resource within a specified date range.

Efficient Resource Allocation and Workload Management

Resource information may be accessed from the 'Resources' tab within a project to review the availability of resources. Project Insight, web project management software provides real-time resource allocation data based on the allocation of their assignments to project tasks system-wide.

Project managers can also view all resources across all projects in Project Insight. This information is accessed in 'My Reports,' 'Cross Project Resource Allocation.' Data may be hidden or displayed according to each person's preferences, supporting a wide variety of applications for these reports. Hundreds of permutations of resource allocation reports are available.

Other project management software applications claim to have extensive resource allocation capabilities in their marketing materials; however, they often fall short. Project Insight not only allows resource managers or project managers to see the total workload each resource has per day, week or other time period, it allows them to drill down on all of the projects and tasks that are causing the over allocation in one view. Tasks can easily be reassigned using Project Insight's simple drag and drop functionality. It's perfect for the management of all kinds of goals, tasks and projects including IT projects, interactive or marketing projects, product development projects, professional services and more.

Notes All tasks are efficiently managed with proper resource allocation and tracking, down to the last detail.

Self Assessment

Fill in the blanks:

1. Project evaluation is an …….. and objective measurement of an ongoing or completed project.
2. …… and administering both projects and the overall program.

Notes

3. ……are usually governed by a simple management structure.
4. At the top of the program management hierarchy are the program sponsor(s) and the program ……...
5. The costs are bigger not only because the program is larger, but also as it entails more types of ……...
6. …….. provides specialized staff expertise for specific work areas.
7. Resource information may be accessed from the 'Resources' tab within a project to review the availability of ……..

Case Study

Practicing Project Management: A Case Study, 2005 BWF-HHMI Course in Scientific Management

The ability to allocate resources to achieve a goal is the hallmark of project management. This case study is designed to help you practice the allocation of resources in project management and identify the effects of resource allocation on achieving specific goals. Read through and complete this case study, and examine the final outcome to understand the potential problems and pitfalls that can occur in projects. Through these types of exercises and personal experience you will build an experience base of skills that will help you better manage your own research laboratory.

Statement of Work

Background. You are an Assistant Professor at Enormous State University (ESU) and perform research on Yersinia pestis (plague) and pulmonary cell interactions. Recently you have found a very interesting proton pump in Y. pestis that when blocked by a drug inhibitor in vitro, kills the bacteria but not human cells in culture (you have a paper submitted on this work). You have been wondering if this drug could be an antidote for a plague infection? The NIH has just released an RFA for "Saving our soldiers from dangerous pathogens" that is due in 13 weeks. This grant could fully fund your lab for five years. Your goal is to prepare yourself for a successful RFA proposal.

Case study: You need to define the scope of the work you would like to achieve to reach your goal. Yet you are constrained by the available research funds and can only choose two of the following for your scope of work.

Scope of Work

- Propose a clinical trial for the existing drug inhibitor to demonstrate its efficacy in stopping Y. pestis.
- Create additional candidate drugs with similar structure to your inhibitor that have high potential to kill Y. pestis.
- Make an antibody to pump that will help in pump characterization.
- Identify gene homologs and physiologic pathways of current pump identified.
- Define role of immune system and its potential to improve cell killing with respect to pump inhibition.

Contd.....

Notes

Choices: Scope of Work

Allocation of Resources: Assignment of Personnel:

You have sufficient personnel in your laboratory to attempt the scope of work. Allocate your personnel according to the following rules:

- Choose only two scope projects.
- You must do the ongoing lab work.
- Each person can only do one scope project/ongoing lab work.
- You need at least one person per scope project.
- You can have up to two people per project.
- You must have people covering the ongoing lab work (one for each column).

Different personnel have different research strengths and can or cannot function in different projects. Their abilities are listed in the following table. You can consider a degree of redundancy in the work assignments that you may want in case of personnel loss or turnover.

Name	Person	Scope 1	Scope 2	Scope 3	Scope 4	Scope 5	Ongoing Lab Work 1	Ongoing Lab Work 2
Rudy	Physician-scientist	Best	Poor	Adequate	Best	Poor	Poor	Poor
Chia-Ling	Graduate student	Poor	Best	Adequate	Poor	Poor	Adequate	Best
Suzanne	Post-doc	Poor	Poor	Best	Poor	Adequate	Adequate	Best
Hortence	Post-doc	Best	Adequate	Poor	Best	Adequate	Best	Adequate
Pradip	Post-doc	Adequate	Best	Adequate	Poor	Best	Best	Adequate

Things Happen!

In the course of all events it is certain that things will go wrong. While you are working on your projects the following things have happened in the laboratory. These will affect your work, and the ability to complete your goal.

- Rudy quits the laboratory to join a private practice medicine group. If you want to continue Scope 1 you need to place another person on the project (if you do not already have another person covering the project).
- Bob's Brain Foundation, the group funding your brain tumor research, decides to perform a review of your program. To keep funding you need to show significant progress. You need your "best" people doing the ongoing lab work.

Modify your available resources to best accomplish you goals. You may also change your scope projects, just realize it is a "bit late in the game" to be changing scope projects.

Name	Person	Scope 1	Scope 2	Scope 3	Scope 4	Scope 5	Ongoing Lab Work 1	Ongoing Lab Work 2
Chia-Ling	Graduate student	Poor	Best	Adequate	Poor	Poor	Adequate	Best
Suzanne	Post-doc	Poor	Poor	Best	Poor	Adequate	Adequate	Best
Hortence	Post-doc	Best	Adequate	Poor	Best	Adequate	Best	Adequate
Pradip	Post-doc	Adequate	Best	Adequate	Poor	Best	Best	Adequate

Contd.....

Notes

More Things Happen!

Ongoing events affect your final productivity.

- Hortence is sick and cannot complete her assigned tasks her productivity goes from "Best" to "Adequate", or from "Adequate" to "Poor" depending on the task she has been assigned.

DO NOT make any additional changes to the table. Copy your previous results onto the new table and calculate your point score.

Final Productivity

Name	Person	Scope 1	Scope 2	Scope 3	Scope 4	Scope 5	Ongoing Lab Work 1	Ongoing Lab Work 2
Chia-Ling	Graduate student	Poor 1 Point	Best 3 Points	Adequate 2 Points	Poor 1 Point	Poor 1 Point	Adequate 2 Points	Best 3 Points
Suzanne	Post-doc	Poor 1 Point	Poor 1 Point	Best 3 Points	Poor 1 Point	Adequate 2 Points	Adequate 2 Points	Best 3 Points
Hortence (sick)	Post-doc	Adequate 2 Points	Poor 1 Point	Poor 1 Point	Adequate 2 Points	Poor 1 Point	Adequate 2 Points	Poor 1 Point
Pradip	Post-doc	Adequate 2 Points	Best 3 Points	Adequate 2 Points	Poor 1 Point	Best 3 Points	Best 3 Points	Adequate 2 Points

Examine the table above. Using the columns, copy the check boxes of the scope projects and personnel you used from the previous table. Record the points you received for each project at the bottom. If you chose a project and at least one of the personnel you placed on the project had a "Best" score it as 3 points. If the personnel had "Adequate" score it as 2 points. Add up your points and place the sum in the "TOTAL POINTS" Box. If the personnel had "Poor" score it as 1 point. If you switched any projects between table one and table 2 (i.e. after the misfortune) subtract 2 points from your total.

Total Points	
Subtract 2 Points If You Changes Scopes	
Final Points	

Final Point Scoring:

Final Points	Outcome
10-12	Excellent- Kept current lab funding and added new grant
8-9	Good-Kept current lab funding
0-7	Poor – Too high risk, lost all lab funding

The goal of this case study is not to win, but to see how ongoing events can affect your laboratory, and how project management can help you control your resources for your benefit. Key things to consider as you analyze your projects.

- After the initial things went wrong, how did you reallocate your resources?
- How much risk were you willing to take? If you lost a scope project, did you refocus on protecting the current lab funding? Were you willing to risk your current lab funding to get more funding?
- What would you do differently next time?

This case study was developed by Milton Datta, M.D., Emory University, for the session on project planning at the 2005 BWF-HHMI Course in Scientific Management

Notes

4.3 Creating Programme

Across the India at the moment there will be hundreds of programmes being run, but how well are they being run and how does the sponsor know that his/her programme is in a healthy shape? There are a number of ways to find out, most of them costing money from consultants. Most programmes are complex and are being run using a methodology that fits one of the following descriptions:

1. Home grown - the organisation has taught itself how to run large complex change initiatives based on its project experience.
2. Proprietary methodologies from one of the big consultancies at a significant cost.
3. OGC Managing Successful Programmes methodology available for public use and owned by the OGC. This is by far the dominant approach in the current market place.

From our experience there is a thread that runs through most proven methodologies, and is often missing from the home grown approach. We call it The Golden Thread, as all the successful programmes we have come across use it, whilst less successful ones don't.

4.3.1 The Golden Thread

- ***The Vision Statement:*** clearly sets out the direction and purpose of the programme.
- ***Blueprint:*** A detailed description and understanding of what the future state of the organisation will be like, an important step often missed. If you don't know where you are going, how will you know when you have arrived?
- ***Quantified Benefits:*** A detailed description of the benefits to be delivered, not a set of bullet points buried in a business case. This is the core of an effective programme, and can't effectively be done until you know where you are going.
- ***Project Portfolio:*** Well structured to maximize the efficient use of resources, designed to deliver dividends early and effective control points to stop the programme running away with itself.
- ***Business Transition:*** Effective plans to minimize the instability caused to the business whilst remaining ambitious about the intended achievements.
- ***Outcomes Achieved:*** Focus remains, the new systems have been installed and moved to the new building, you have reached the new state, but you still will not have achieved the benefits.
- ***Benefits Realization:*** It's all too easy to declare victory before the benefits have been achieved. People still think that a miracle will occur that turns outcomes into benefits which is wrong - management intervention will still be required.

With this information, you can do your own health check to test the quality of your programme. If the key components of The Golden Thread are not in place you will undoubtedly run into trouble. It's all too easy to bypass the blueprint or benefits through expediency to move forward, but if we had £10 for every senior manager who said, if we'd have done this in the first place we would have headed off all the problems later, we'd be very rich people by now.

Task Most programmes are complex and are being run using a methodology. Discuss

Notes

4.4 Individual Projects

An individual project is an economically indivisible series of works fulfilling a precise technical function and with clearly identifiable aims. An individual project may include one or more sub-projects.

Example: The funding of individual R&D projects oriented to the market represents by far the core activity of the EUREKA network. Challenged now by the Eurostars programme for Small and Medium sized entreprises also ran by EUREKA, it remains the main European funding tool for large companies and research Institutes willing to do business through technology. EUREKA ongoing projects in figures.

Project Participants

Large companies	476
SMEs	1174
Research Institutes	491
Universities	459
Government/National Administration	40
Total number of organisations involved in EUREKA projects	**2640**

Budget

Number of running projects	722
Total budget for these projects	€ 1.3 billion
Average budget per project	**€ 1.8 million**

4.5 Technical Assessment

Assessment (analysis and normative evaluation) of a particular technical device, system, or procedure with regard to a defined set of criteria, goals and objectives.

The terms assessment and evaluation are often used interchangeably, and one glossary of evaluation (NSF 97-153) defines assessment as "often used as a synonym for evaluation." Diane Ebert-May offers that assessment is "data collection with a purpose," and often addresses the "what questions about teaching and learning—what do students know and what can they do?" Assessment provides evidence that things are working or not (e.g. Do students actually learn better?). Evaluation may be defined as "the systematic investigation of the merit or worth of an object" (NSF 93-152) and is often used in the context of "what value has been added through this project, and how do you know?" Project evaluation is often used to demonstrate accountability (e.g. have the project goals been met?).

Assessment of education and outreach programs may be done for many reasons, on many scales and the results may be utilized by different interested groups-including top-to-bottom project reviews, evaluation of the effectiveness of specific materials or methods, indicators of student learning, long-term impacts of a project, confirmation that the goals of a project have been met.

Notes

Assessment activities may also lead to more expansive research on learning projects. The scholarship of teaching and learning provides many exciting (and much needed) opportunities to form partnerships with the cognitive and social sciences.

There are a few basic principles that will help you effectively develop your own assessment plans to best meet the needs of your project:

- Clearly define project goals and expected outcomes at the start.
- What is the purpose of the assessment? Who will use the results and in what way?
- Identify the baseline data you will need to document change.
- There is an arsenal of assessment techniques that are available; pick the right tools and metrics that will provide the information required to meet your needs.
- Assessment is done throughout the course of a project for varying reasons: formative assessment is done to provide feedback for ongoing activities, and to inform any needed mid-course corrections; summative assessment is done to measure a project's overall success; longitudinal assessment tracks impacts beyond the duration or initial scope of the project.
- The assessment plans should be integral to the development and management of the project, not just added on as an afterthought.
- Develop partnerships with colleagues who have knowledge and expertise in assessment.

Self Assessment

Fill in the blanks:

8. The …… clearly sets out the direction and purpose of the programme.
9. If the …….. of The Golden Thread are not in place you will undoubtedly run into trouble.

4.6 Cost Benefit Analysis

Cost-benefit analysis (CBA) is the weighing-scale approach for decision-making. All the positive elements (cash-flows and other intangible benefits) are put on one side of the balance and all the negative elements (the costs and disadvantages) are put on the other. Whichever weighs the heavier wins.

You may have been intensely creative in generating solutions to a problem, and rigorous in your selection of the best one available. This solution may still not be worth implementing, as you may invest a lot of time and money in solving a problem that is not worthy of this effort.

Cost Benefit Analysis or cba is a relatively simple and widely used technique for deciding whether to make a change. As its name suggests, to use the technique simply add up the value of the benefits of a course of action, and subtract the costs associated with it.

Costs are either one-off, or may be ongoing. Benefits are most often received over time. We build this effect of time into our analysis by calculating a payback period. This is the time it takes for the benefits of a change to repay its costs. Many companies look for payback over a specified period of time – e.g. three years.

In its simple form, cost-benefit analysis is carried out using only financial costs and financial benefits.

Notes

Example: A simple cost/benefit analysis of a road scheme would measure the cost of building the road, and subtract this from the economic benefit of improving transport links. It would not measure either the cost of environmental damage or the benefit of quicker and easier travel to work.

4.6.1 Mistakes and Problems with Cost-benefit Analysis

A frequently made mistake in the CBA method is to use non-discounted amounts for calculating the costs and benefits. A method like NPV or Economic Value Added or CFROI is strongly recommended, because all of these account for the time value of money.

A frequent problem with CBA is that typically the costs are tangible, hard and financial, while the benefits are hard and tangible, but also soft and intangible. Caution should be taken here against people who claim that "if you can't measure it does not exist/it has no value".

Especially in more strategic investments, frequently the intangible benefits clearly outweigh the financial benefits.

Risk must often be considered as a factor in making the decision.

Did u know? What is NPV?

In finance, the net present value (NPV) or net present worth (NPW)[1] of a time series of cash flows, both incoming and outgoing, is defined as the sum of the Present Values (PVs) of the individual cash flows of the same entity. In the case when all future cash flows are incoming (such as coupons and principal of a bond) and the only outflow of cash is the purchase price, the NPV is simply the PV of future cash flows minus the purchase price (which is its own PV).

4.6.2 History of Cost Benefit Analysis

The idea of this methodology originated with Jules Dupuit, a French engineer whose 1848 article is still worth reading. The British economist, Alfred Marshall, conceived some of the formal concepts that are at the foundation of CBA. But the practical development of CBA came as a result of the impetus provided by the Federal Navigation Act of 1936. This act required that the U.S. Corps of Engineers carry out projects for the improvement of the waterway system when the total benefits of a project exceed the costs of that project. Thus, the Corps of Engineers had create systematic methods for measuring such benefits and costs. The engineers of the Corps did this without much assistance from the economics profession. It wasn't until about twenty years later in the 1950s that economists tried to provide a rigorous, consistent set of methods for measuring benefits and costs and deciding whether a project is worthwhile.

Example: A sales director is deciding whether to implement a new computer-based contact management and sales processing system. His department has only a few computers, and his salespeople are not computer literate. He is aware that computerized sales forces are able to contact more customers and give a higher quality of reliability and service to those customers. They are more able to meet commitments, and can work more efficiently with fulfillment and delivery staff.

Notes

His financial cost/benefit analysis is shown below:

Costs:

New computer equipment:

- 10 network-ready PCs with supporting software @ $2,450 each
- 1 server @ $3,500
- 3 printers @ $1,200 each
- Cabling & Installation @ $4,600
- Sales Support Software @ $15,000

Training costs:

- Computer introduction – 8 people @ $400 each
- Keyboard skills – 8 people @ $400 each
- Sales Support System – 12 people @ $700 each

Other costs:

- Lost time: 40 man days @ $200/day
- Lost sales through disruption: estimate: $20,000
- Lost sales through inefficiency during first months: estimate: $20,000
- Total cost: $114,000

Benefits:

- Tripling of mail shot capacity: estimate: $40,000/year
- Ability to sustain telesales campaigns: estimate: $20,000/year
- Improved efficiency and reliability of follow-up: estimate: $50,000/year
- Improved customer service and retention: estimate: $30,000/year
- Improved accuracy of customer information: estimate: $10,000/year
- More ability to manage sales effort: $30,000/year

Total Benefit: $180,000/year

Payback time: $114,000 / $180,000 = 0.63 of a year = approx. 8 months

The payback time is often known as the break even point. Sometimes this is more important than the overall benefit a project can deliver, for example because the organization has had to borrow to fund a new piece of machinery. The break even point can be found graphically by plotting costs and income on a graph of output quantity against $. Break even occurs at the point the two lines cross.

Inevitably the estimates of the benefit given by the new system are quite subjective. Despite this, the Sales Director is very likely to introduce it, given the short payback time.

Example of a Cost-benefit Analysis

A company that would like to buy Business Intelligence software to improve its business might use a CBA to make up its mind.

Notes

On the minus (cost) side would be:

- the price of the software,
- the cost of consultants to install and implement the software, and
- the cost of training for the users of the software.

However on the plus (benefits) side would be:

- improved business processes (leading to an annual cost decrease),
- due to better available information, the company will be able to take better decisions (leading to additional cash-flows), and
- increased staff moral, due to using these modern tools to support the business.

4.7 Risk Evaluation

A risk evaluation can be defined as a problem that could cause some loss or threaten the success of project. The potential risk factors may have an adverse impact on the cost, schedule, or technical success of the project. Thus, risk management is the process of identifying, determine and solve the potential problems before they can damage the project.

Offshore program and project management involves four critical activities:

1. Transition Management
2. Governance
3. Performance Management
4. Quality Management

Transition Management

The process of managing assets during a period of restructuring is defined as transition management. The role of the Transition Manager is to minimize the costs and risks thus ensuring that the process runs smoothly.

The real process starts when the contract is signed. Smooth transition management is the next issue to manage. Transition management is considered to be a critical success factor of offshore activities. Transition management is defined as the detailed knowledge of transfer and documentation of all relevant tasks, technologies and workflows The transition period is perhaps the most difficult stage of an offshore endeavor, taking anywhere from three months to a year to complete.

Transition management involves the following:

- Develop an initial transition plan (involving activities such as milestones, assets and benchmarks)
- Initiation of projects
- Internal procedures and processes
- Manage employees
- Document lessons learned to improve vendor management

Task Analyze in group of four why Transition management is considered to be a critical success factor of offshore activities?

Notes

Governance

After managing the issue of transition put the skills into practice by governing the offshore relationship - client management, third party contract. The approach focuses on the evolution of services provided, ongoing communication processes, and overall project management. Governance activities pertaining to areas of off shoring can make or break a project.

The Ongoing governance involves the following:

- Project management
- Relationship building and management
- Risk management - describes the processes concerned with identifying, managing, and correction of outsourcing partnership risks.

Performance Management

As offshore outsourcing has become an important aspect for multiple business processes, the types and complexity of contracts and sourcing alliances are bound to explode. With organizations outsourcing almost every aspect of their operations, multiple vendors participating in sourcing deals, and activities occurring 24x7, it's a nonstop challenge to coordinate interactions, manage performance, monitor contract terms, track financial metrics, and maintain alignment.

Ongoing governance involves the following:

- Measures outsourcing effectiveness using appropriate metrics
- Implementing improvements and adjustments
- Evaluating feasibility of additional outsourcing

Quality Management

Quality is a huge concern with offshore outsourcing. Errors are more costly to fix and debugging becomes essential. A strict quality assurance and control program forms an integral part of every offshore delivery project.

Performance management aims at reviewing and continuous improvement of software development and business processes, validation and verification of work products and customized status reports.

Self Assessment

Fill in the blanks:

10. Cost Benefit Analysis is a relatively simple and widely used technique for deciding whether to make a ……...
11. Cost-benefit analysis is carried out using only financial costs and financial …….
12. Risk management is the process of identifying, determine and solve the potential problems before they can …… the project.
13. The role of the Transition Manager is to …….. the costs and risks.
14. ……. activities pertaining to areas of off shoring can make or break a project.
15. Quality is a huge concern with …… outsourcing.

Notes

4.8 Summary

- Program management or programme management is the procedure of organizing numerous related projects, often with the meaning of improving an organization's performance.
- Project Evaluation is a step-by-step process of collecting, recording and organizing information about project results, including short-term outputs.
- We call it The Golden Thread, as all the successful programmes we have come across use it, whilst less successful ones don't.
- Cost/Benefit Analysis is a powerful, widely used and relatively easy tool for deciding whether to make a change.
- To use the tool, firstly work out how much the change will cost to make. Then calculate the benefit you will from it.
- Where costs or benefits are paid or received over time, work out the time it will take for the benefits to repay the costs.
- You may, however, decide to include intangible items within the analysis. As you must estimate a value for these, this inevitably brings an element of subjectivity into the process.
- The potential risk factors may have an adverse impact on the cost, schedule, or technical success of the project.

4.9 Keywords

CBA: Cost Benefit Analysis

CFO: Chief Financial Officer

ESU: Enormous State University

PMO: Project Management Officer

4.10 Review Questions

1. Explain why Programme Management is said to be the procedure of managing numerous ongoing interdependent projects.
2. Program governance creates both the structure and practices to guide the program. Discuss in detail with an example.
3. At the core or middle of the hierarchy is the program manager/director. Who are present in the upper most hierarchies? Explain the roles and responsibilities of Program Manager.
4. The financial feature of a program comprises the need to conform to internal. Analyze this statement and explain in detail.
5. Project managers can also view all resources across all projects in Project Insight. What types of resources they are talking about? Explain briefly.
6. Explain how to create the programmes using the golden threads? Explain the features of golden threads in detail.
7. Discuss the individual projects with an example.
8. Good governance is dangerous to program success. Why?

Notes

9. Cost/Benefit Analysis can be carried out using only financial costs and financial benefits. What do you understand from this statement?
10. Offshore program and project management involves many critical activities. Explain.

Answers: Self Assessment

1.	organized	2.	Planning
3.	Projects	4.	steering committee
5.	expenditures	6.	PMO
7.	Resources	8.	Vision Statement
9.	key components	10.	change
11.	benefits	12.	damage
13.	minimize	14.	Governance
15.	offshore		

4.11 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://en.wikipedia.org/wiki/Program_evaluation

http://managementhelp.org/evaluation/program-evaluation-guide.htm

Notes

Unit 5: Project Approach

CONTENTS

Objectives

Introduction

5.1 Project Approach

5.2 Technical Plans

5.2.1 Technical Documentation

5.3 Choice of Process Models

5.3.1 Water Fall Model

5.3.2 V-Process Model

5.3.3 Spiral Model

5.3.4 Prototype Model

5.4 Incremental Delivery

5.5 Summary

5.6 Keywords

5.7 Review Questions

5.8 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize technical plan
- Describe the process models
- Explain the incremental delivery

Introduction

The purpose of the project approach is to identify the project management methodologies that will be used on the project. The approach should be based on the project management framework but should also contain reference to and iterative development, or prototyping methodologies that will be utilized. It is imperative to offer enough description of the approach but the description should not be too detailed. It should also be noted here if no specific or a general approach will be used.

5.1 Project Approach

Approaches vary from highly planned and controlled to fast supple and adaptable. Traditional approaches focus on predictability where as more fresh approaches prioritize flexibility to changing environments and requirements, in order to maximize business benefits. Approaches vary according to the amount of overlap between phases and how often and rapidly they are carried out. There is no one right approach. A best approach should be

Notes

chosen according to environment and the type of project being undertaken. Approaches should be chosen according to:

- How well understood the objectives are (by the client and the team)
- How stable the technology is (well proven or need testing and adaptation)
- How experienced the team is (in the technology, in development and in understanding the objectives set by the organizations management)
- How much control management want.

Generic Project Phases

1. Initiation/Analysis
2. Design
3. Construction/Implementation/Execution/Coding
4. Testing/Pilot/Debugging/Verification)
5. Deployment
6. Review & Maintenance

Approach Overviews

- ***Waterfall:*** Waterfall has extremely strict separate stages, with no overlap, and high levels of planning and process control. This is suitable for projects with a well understood scope that can be built using proven technology, before it goes out of date. For example, this is the favored approach in the construction industry.
- ***Iterative:*** In iterative the team produces succeeding releases that eventually evolve into a complete product. This is a good approach when there are many unknowns, or when speed is of the essence. This might be a good approach to implementing a leading edge product or developing for a user who is not sure what they want.
- ***Agile:*** Agile is the same as Iterative but with particularly small/fast releases in a high performance environment with low levels of process control. Good for motivated experienced teams with strict deadlines and good understanding of the organisation's strategy.

The project approach should also list exact tools or techniques that are not part of the project management framework or not included in the approved information management technology list.

The approach should be based on the decided information management project methodology.

Notes Managed exceptions to the approved methodology should be documented in the project management approach.

The project management approach should comprise the following key tasks or phases:

Business Case Analysis

The objective of the business case analysis phase is to determine the business case for investing in a data warehouse or business intelligence solution. It identifies the projected return on investment.

Notes

Planning and Analysis

The objectives of the planning and analysis phase are to:

- Recognize and secure approval for information management requirements; and
- Complete project planning in enough detail to commence technical architecture and design.

Architecture and Design

The objective of the architecture and design phase is to describe the technical architecture for the entire solution including:

- Database architecture and infrastructure;
- Database sizing and performance expectations;
- Identification of the extract, transform and load process;
- Recommendations for the sizing and configuration of hardware; and
- Access control, backup and recovery guidelines.

Build and Test

The objectives of the build and test stage is to deliver:

- Code required to extract data from the source systems and load it into a temporary staging area;
- An automated process that will extract data from source systems;
- Programs that will load the data warehouse with the source data from the staging area including all data base scripts for views, indexes, synonyms and aggregations;
- Processes to clean the data in the staging area and ensure that it is fully consistent with business rules;
- Programs and scripts that ensure backup/recovery, access control and archiving;
- System processes to handle the data warehouse manager;
- A query manager to support business intelligence requirements;
- Pre-developed queries and reports and user access tools to access information;
- A set of repeatable system integration, volume and aggregation tests; and
- A set of repeatable manual reconciliation tasks.

User Acceptance Test (UAT)

The objective of the UAT stage is to ensure that the information management solution meets agreed requirements.

Release to Production

The objective of this stage is to migrate the information management solution into a production environment. The initial data should be pre-loaded and the Data Warehouse should be tuned for performance, if necessary.

Notes

Self Assessment

Fill in the blanks:

1. A best approach should be chosen according to …….. and the type of project being undertaken.
2. The project approach should also list exact tools or techniques that are not part of the project management ……
3. Complete project planning in enough detail to commence …….. architecture and design.
4. The initial ……. should be pre-loaded and the Data Warehouse should be tuned for performance, if necessary.
5. The objective of the architecture and design phase is to describe the technical …….

5.2 Technical Plans

The project technical plan can have two meanings, depending on the management:

1. ***Non-schematic representation of the WBS:*** Yes, the project technical plan is more or less the WBS, it is a non-graphical non-schematic representation of the WBS. All the information that is incorporated in the WBS can still be known by reading the project technical plan, including the dependencies. The project technical plan and also have sub-technical plans, which are further breakdowns to activities in the main project technical plan (sub-project technical plans exist only in very large projects).
2. ***A part of the project plan listing the technical issues/challenges pertaining to the project and how to address them:*** For instance, the project technical plan of a web project will state that the web application will receive a lot of visitors, which will drain the server's resources, so the best way to address this problem is by adding more web and database servers to the infrastructure to handle the potential load and to ensure that the written code is efficient.

The project technical plan illustrates a detailed breakdown of the relationships between the major activities required by the project. Succeeding sub-project technical plans are based on it and any later amendments should be highlighted. Prepared at the same time as the project resource plan; these two plans together show a high level view of the projects products, activities and resources. Once approved by the project owner the project technical plan becomes the top level production schedule for the major products required throughout the project.

The technical plan will differ greatly dependent on the particular industry or business sector. A technical plan should be created as part of the business plan, this allow more accurate budgeting for things such as capital equipment, land, energy costs, etc. The would contain items such as initial building plan, ground requirement, machinery or equipment, Energy usage, waste treatment, in fact anything that clarifies the technical part of the project together with a Gantt (timing) chart and possibly some sketches of the product, building and possibly a flow chart of the process.

An environmental Impact report should also be completed for any manufacturing processes you may be involved in. This is usually a specialized document showing how "Green" your company is. In today's business climate it is very important.

Task With your tutor analyze why WBS is important in project technical plans?

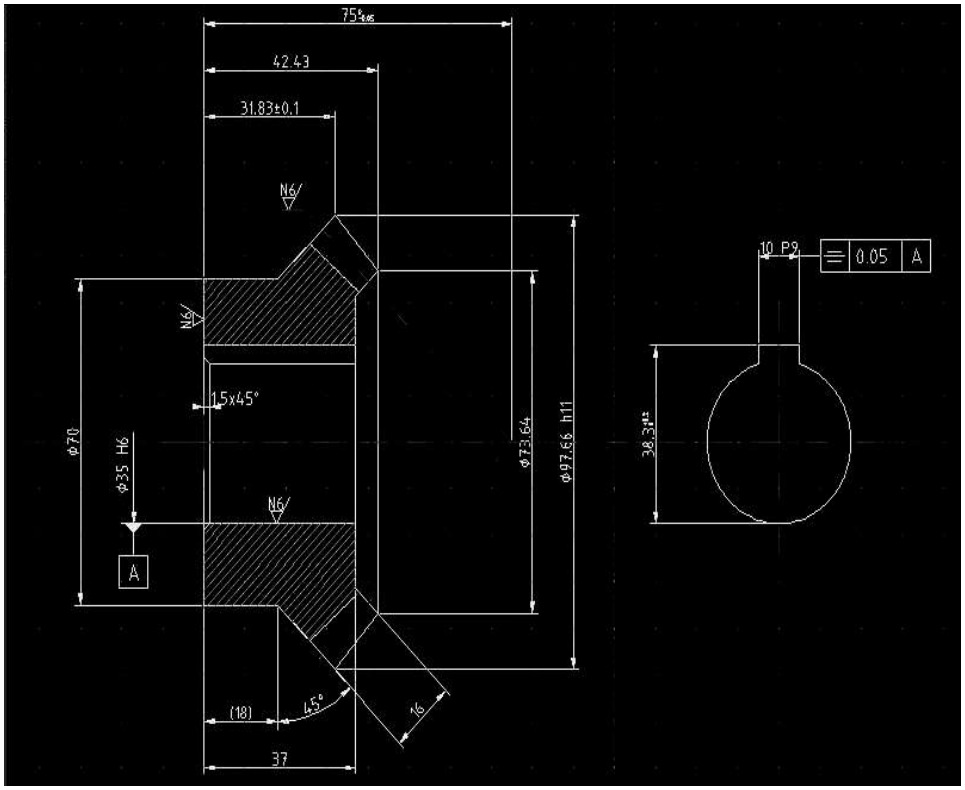
Notes

5.2.1 Technical Documentation

In many situations, the input of technical information will not be required, in particular for smaller projects, none manufacturing projects or internet projects.

If the project involves land, building or substantial capital equipment then it is better to include information that is more technical. This is mainly due to the need for accurate figures for land, buildings and capital equipment, which cannot be estimated precisely. These are required for the Financial Reports. Often the cost of building, if guessed can vary widely, once you know the approximate area then a local building contractor should be able to give you a cost per square meter.

This process needs some initial design work consequently for a project including buildings and work areas you would need some form of initial plan showing the land area required and any construction such as this example.

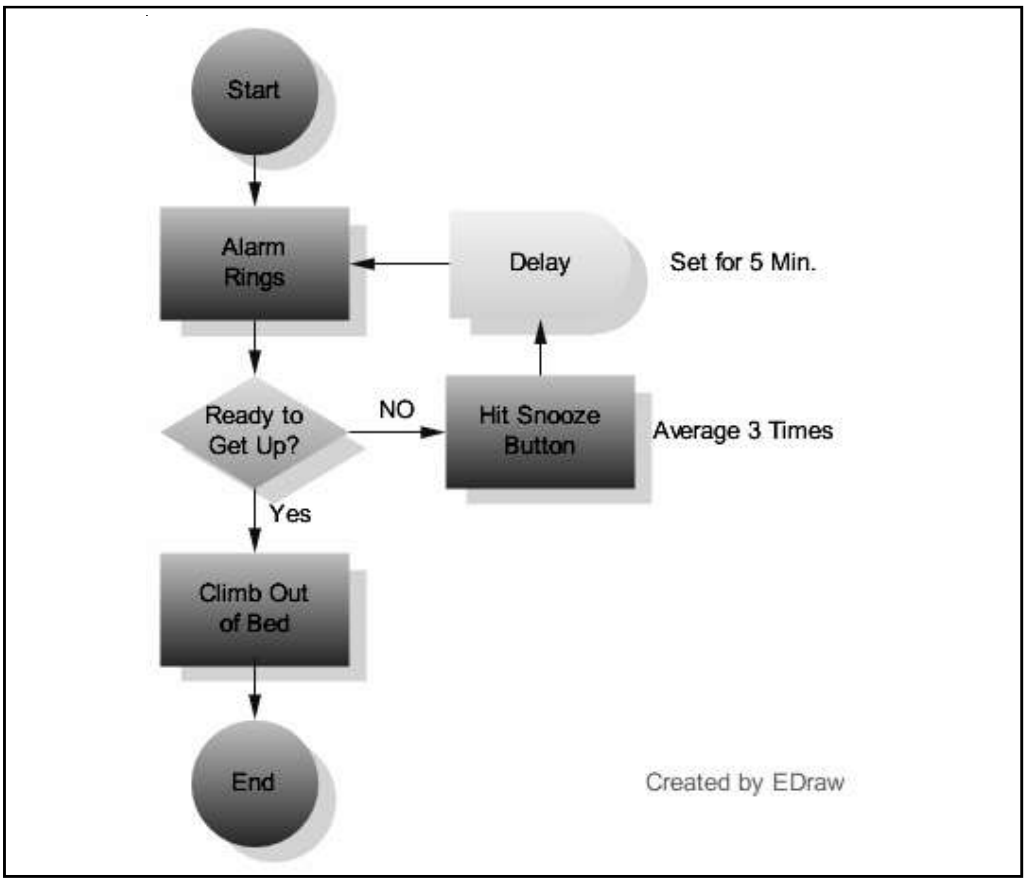


Above is a plan of a planned building, giving the area of construction required, for the square meters it is possible to estimate the total cost of the construction.

Below is a 3D drawing of a planned project, whilst not essential to the business plan a drawing such as this does help sell the project to the financiers. Also for liaising with potential clients/ customers before the date of first production.

Notes

Other items referring to the capital costs should be additional to the Technical Plan, such as a listing of capital equipment, a timing chart to show cash flow and, if the project relates to a manufactured item, process flow diagrams such as below. This use my theory that if you can't put it on paper than there is little chance of being able to put the process together in actuality.



It is also valuable including initial product costing, product specifications and any other information that will clarify the profitability of the project.

If the suggested technical information is incorporated then the plan becomes a more valuable document containing the information needed to complete the project.

Notes

Self Assessment

Fill in the blanks:

6. The project technical plan illustrates a detailed ……. of the relationships between the major activities required by the project.
7. An environmental Impact report should also be completed for any …… processes.
8. If the project involves land, building or substantial capital equipment then it is better to include information that is more ……...

5.3 Choice of Process Models

Building software is a continuous learning process and the outcome is nothing but a refined version of knowledge that has been collected and processed in the process. A process model can be called a framework of tasks required to develop and build high-quality software. Can we call the process model as software engineering? The answer is "yes" and "no". Process model is techniques that are involved while software is being engineered. Also, the technical methods and tools that comprise in the software engineering form a part of the model. Software must be developed keeping in mind the demands of the end-user using defined software. We will discuss the various stages that go into the making of software and the software models involved therein, in this lesson. A structured set of activities required to develop a process model:

- Specification
- Design
- Validation/Verification
- Evolution

A process model is an abstract representation of a process. It presents a description of a process from some particular perspective.

A process model or process model is a simplified representation of a software process, presented from a specific perspective.

The linear sequential model or the waterfall model is based on the assumption that a complete system is delivered at the end of the complete software development cycle. It is designed for linear development. The prototype model is designed to make the developer and the customer understand the requirements in a better way. It does not deliver a production system at once. However, these models are not evolutionary i.e. iterative in nature. Iterative models help the software engineers to develop more complex software.

Did u know? What are the various examples of process perspectives?

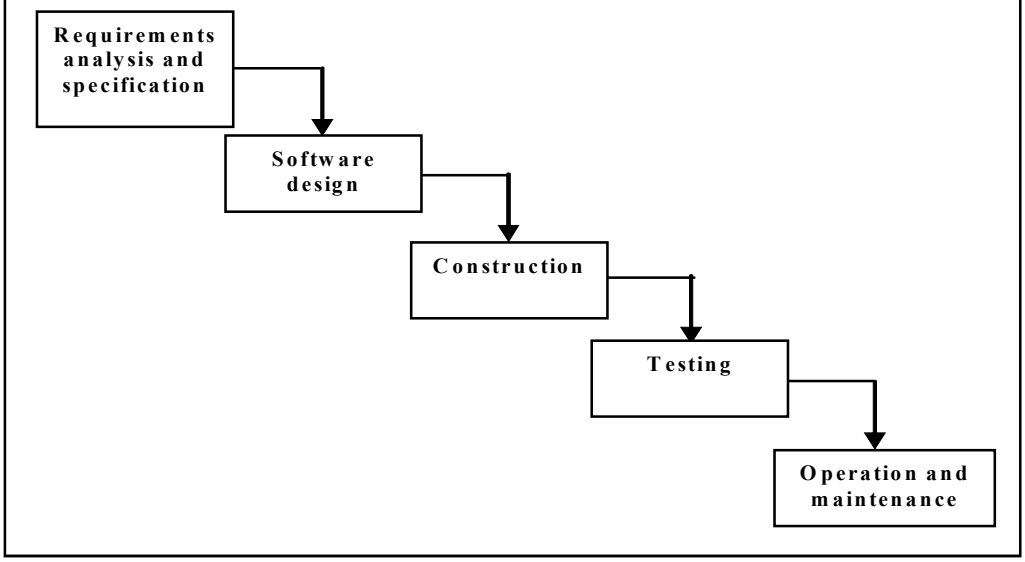
- Workflow perspective - sequence of activities
- Data-flow perspective - information flow
- Role/action perspective - who does what

5.3.1 Water Fall Model

The waterfall model derives its name due to the cascading effect from one phase to the other as is illustrated in the Figure 5.1. In this model each phase is well defined, has a starting and ending point, with identifiable deliveries to the next phase.

Notes

Figure 5.1: Waterfall Model



This model has five phases: (i) requirement analysis and specification, (ii) design, (iii) implementation and unit testing, (iv) integration and system testing and (v) operation and maintenance. All the phases occur one-by-one without overlapping. The next phase begins only after the end of the previous phase.

- ***Requirement Analysis and Specification:*** This phase aims at understanding the exact requirements of the customer and documents them. Both the customer and the software developer work together so as to document all the functions, performance and interfacing requirements of the software. It describes the "what" of the system to be produced and not "how". In this phase a large document called Software Requirement Specification document (SRS) is produced which contains the detailed description of what the system will do in the common language.
- ***Design Phase:*** The aim of this phase is to transform the requirements gathered in the SRS into a suitable form which permits further coding in a programming language. It defines the overall software architecture together with high level and detailed design. All this work is documented as Software Design Description document (SDD).
- ***Implementation and Unit Testing Phase:*** The actual coding begins at his stage. The implementation goes smoothly if the SDD has complete information required by the software engineers. During testing, the code is thoroughly examined and modified. Small modules are testes in isolation initially. Thereafter, these modules are tested by writing some overhead code in order to check the interaction between these modules and the flow of intermediate output.
- ***Integration and System Testing:*** This phase is highly crucial as the quality of the end product is determined by the effectiveness of the testing carried out. Better output will lead to satisfied customers, lower maintenance costs and accurate results. Unit testing determines the efficiency of individual modules. However, in this phase the modules are tested for their interactions with each other and with the system.
- ***Operation and Maintenance Phase:*** Maintenance is the task performed by every user once the software has been delivered to the customer, installed and operational.

Notes

Thus, the delivery of software initiates the maintenance phase. The time and efforts spent on the software to keep it operational after release is important. It includes activities like error correction, removal of obsolete functions, optimization and enhancements of functions. It may span for 5 to 50 years.

Thus, in this model the requirements must be clear at the very initial stages. The end product is available relatively late which in turn delays the error recovery.

Task All the phases occur one-by-one without overlapping. Discuss in a group of four why it happens so in water fall model?

Project Outputs in Waterfall Model

As we have seen, the output of a project employing the waterfall model is not just the final program along and documentation to use it. There are a number of intermediate outputs that must be produced to produce a successful product. Though the set of documents that should be produced in a project is dependent on how the process is implemented, the following is a set of documents that generally forms the minimum set that should be produced in each project:

- Requirements document
- Project plan
- System design document
- Detailed design document
- Test plan and test reports
- Final code
- Software manuals (e.g., user, installation, etc.)
- Review reports

Except for the last one, these are the outputs of the phases, and they have been briefly discussed. To certify an output product of a phase before the next phase begins, reviews are often held. Reviews are necessary, especially for the requirements and design phases, because other certification means are frequently not available. Reviews are formal meetings to uncover deficiencies in a product and will be discussed in more detail later. The review reports are the outcome of these reviews.

Limitations of the Waterfall Model

Software life-cycle models of the waterfall variety are among the first important attempts to structure the process. However, the waterfall model has limitations. Like a waterfall, progress flows in one direction only, towards completion of the project (from one phase to the next). Schedule compression relaxes this requirement but introduces new complexities.

Well-defined phases, frozen deliverables, and formal change control make the waterfall model a tightly controlled approach to software development. The waterfall model's success hinges, however, on the ability:

- To set the objectives.
- To state requirements explicitly.
- To gather all the knowledge necessary for planning the entire project in the beginning.

Notes

The waterfall model's original objectives were to make small, manageable, individual development steps (the phases) and to provide enough control to prevent runaway projects. There are a few problems that have caused dissatisfaction with this phased life-cycle approach.

- The new software system becomes useful only when it is totally finished, which may create problems for cash flow and conflict with organizational (financial) objectives or constraints. Too much money may be tied up during the time the software is developed.
- Neither user nor management can see how good or bad the system is until it comes in. The users may not have a chance to get used to the system gradually.
- Changes that "aren't supposed to happen", are not viewed kindly, neither for requirements nor during the operational life of the system. This can shorten the software's useful life.

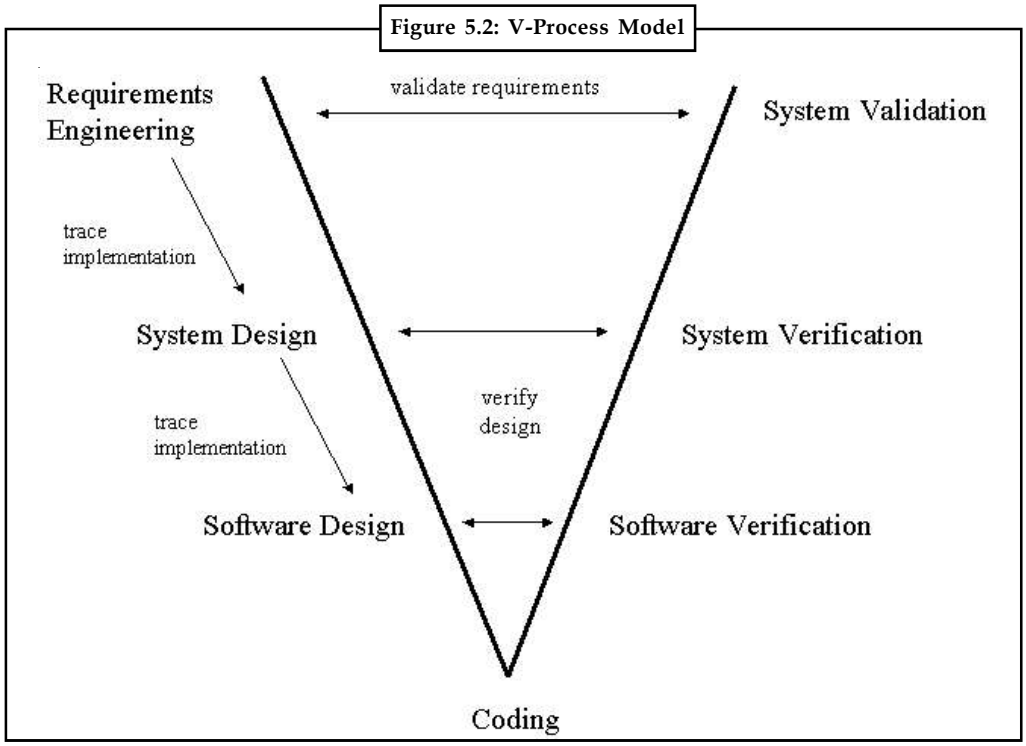
Because of these shortcomings, other process models have appeared. They are based on the concept of iteration and evolution. Despite these limitations, the waterfall model is the most widely used process model. It is well suited for routine types of projects where the requirements are well understood.

Caution If the developing organization is quite familiar with the problem domain and the requirements for the software are quite clear, the waterfall model works well.

5.3.2 V-Process Model

The V-Process Model is an improved version of the classic waterfall model whereby each level of the development life cycle is verified before moving on to the next level. With this model, testing explicitly starts at the especially beginning, i.e. once the requirements are written. Here, by testing we mean verification by means of reviews and inspections, i.e. static testing. This helps in recognizing errors very early in the life cycle and reducing potential future defects appearing in the code later in the life cycle.

Figure 5.2: V-Process Model



Notes

Every level of the development lifecycle has a equivalent test plan. i.e. as each phase is being worked on, a test plan is developed to organize for the testing of the products of that phase. Be developing the test plans, we can also define the expected results for testing of the products for that level as well as defining the entry and exit criteria for each level.

In the V-Model the test activities are spelled out to the similar level of detail as the design activities. Software is designed on the left-hand (downhill) part of the model, and built and tested on the right-hand (uphill) part of the model.

Notes That diverse organization may have different names for the development and testing phases.

The communications between the left and right hand activities are shown by the lines across the middle of the V, showing the test levels from component testing at the bottom, integration and system testing, and acceptance testing at the top level.

Advantages

- Each phase has specific deliverables.
- Higher chance of success over the waterfall model due to the development of test plans early on during the life cycle.
- Time concern in comparison with the waterfall model is low or even we can say 50% less.
- Works well for small projects where requirements are easily understood.
- Utility of the resources is high.

Disadvantages

- Very rigid, like the waterfall model.
- Little flexibility and adjusting scope is difficult and expensive.
- Software is developed during the implementation phase, so no early prototypes of the software are produced.
- Model doesn't provide a clear path for problems found during testing phases.

5.3.3 Spiral Model

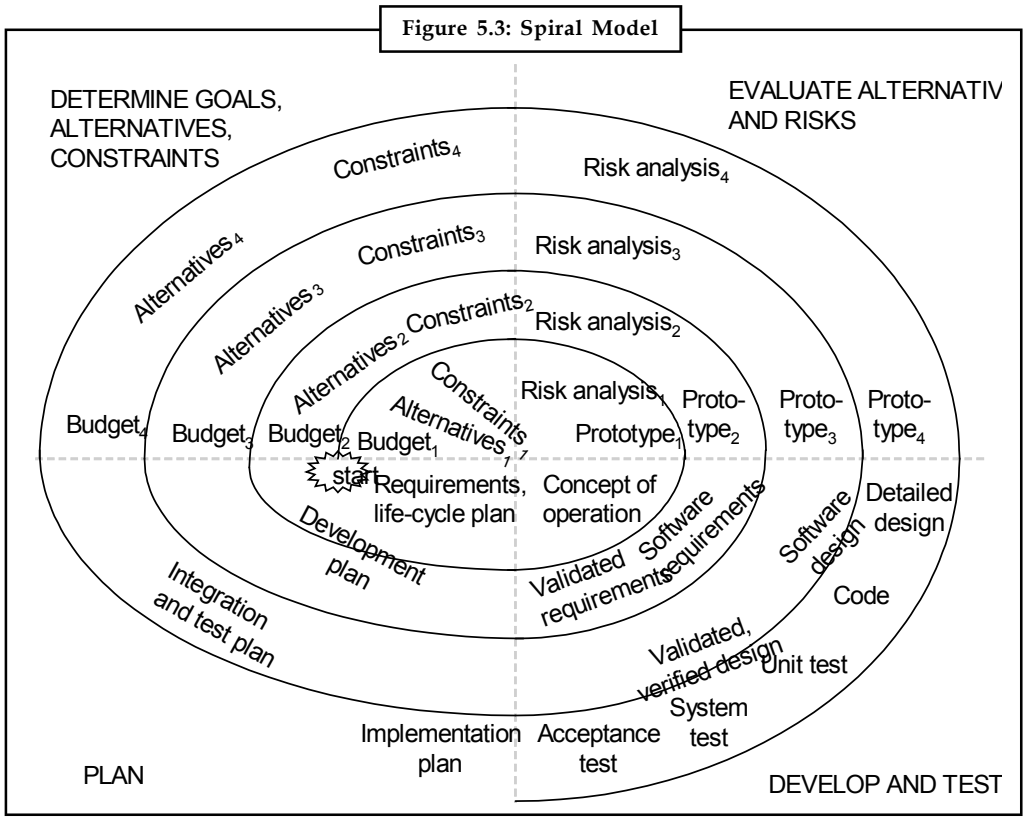
The classical models do not deal with the uncertainty with the software projects. A lot risk assessment and analysis form a part of the software development. This was first realized by Barry Boehm, who introduced the factor of "project risk" into the life cycle model which resulted in the spiral model in 1986 as shown in Figure 5.3.

The radial dimension represents the cumulative cost and each path around the spiral indicates the incremented cost. The angular dimensions depict the progress made in each cycle completion. Each loop of the spiral, clockwise from the X-axis, through 360 degrees represents one phase. Each phase is split into four sectors namely:

- Planning: Determining aims, alternatives and constraints.
- Risk Analysis: Analyzing alternatives and identify and resolution of risks.
- Development: Product development and testing.
- Assessment: Customer review.

Notes

Figure 5.3: Spiral Model



The essential concept of the Spiral Model is "to minimize risks by the repeated use of prototypes [emphasis added] and other means. Unlike other models, at every stage risk analysis is performed. The Spiral Model works by building progressively more complete versions of the software by starting at the center of the spiral and working outwards. With each loop of the spiral, the customer evaluates the work and suggestions are made for its modification. Additionally, with each loop of the spiral, a risk analysis is performed which results in a 'go/no-go' decision. If the risks are determined to be too great then the project is terminated" [Frankovich 1998]. Thus, the Spiral Model addresses the problem of requirements engineering through development of prototypes, and it addresses the need for risk management by performing risk analysis at each step of the life cycle.

Did u know? What are prototypes?

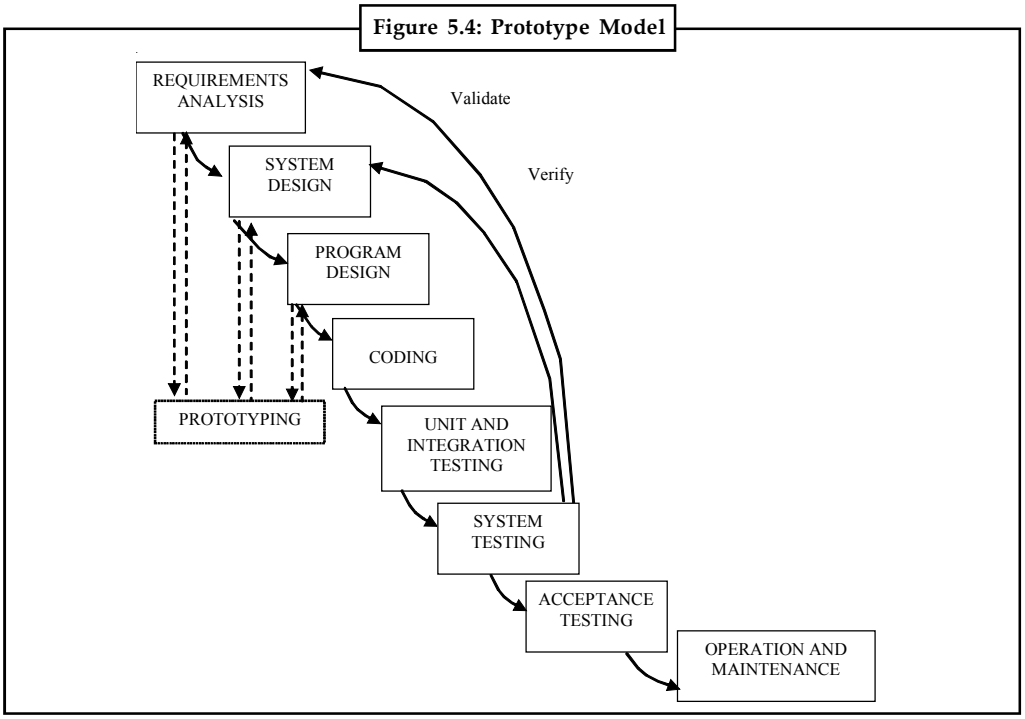
A prototype is an early sample or model built to test a concept or process or to act as a thing to be replicated or learned from.

5.3.4 Prototype Model

A prototype model is beneficial when the customer requirements are dynamic and keep on changing with time and the developer is unsure about the software adaptability with the system and the operating system. Thus in a prototype model, a working prototype is built with the available set of requirements such that it has limited functionalities, low reliability and performance.

Notes

Figure 5.4: Prototype Model



This prototype is further enhanced by the developer with better understanding of the requirements and preparation of a final specification document. This working prototype is evaluated by the customer and the feedback received helps the developers to get rid of the uncertainties in the requirements and to start a reiteration of requirements for further clarification. The prototype can be a usable program with limited functionality but cannot be used as a final product. This prototype is thrown away after preparing the final SRS; however, the understanding obtained from developing the prototype helps in developing the actual system.

The development of prototype is an additional cost overhead but still the total cost is lower than that of the software developed using a waterfall model. The earlier the prototype is developed the speedier would be the software development process. This model involves a lot of customer interaction which is not always possible.

Self Assessment

Fill in the blanks:

9. A process model is an abstract representation of a process. It presents a description of a ……. from some particular perspective.
10. A process model or process model is a simplified representation of a ……, presented from a specific perspective.
11. The delivery of software initiates the …….. phase.
12. In the …….. the test activities are spelled out to the similar level of detail as the design activities.

5.4 Incremental Delivery

A project lifecycle plan used to decrease risk of project failure by dividing projects into lesser, more manageable pieces. The resultant sub-projects may deliver parts of the full product, or

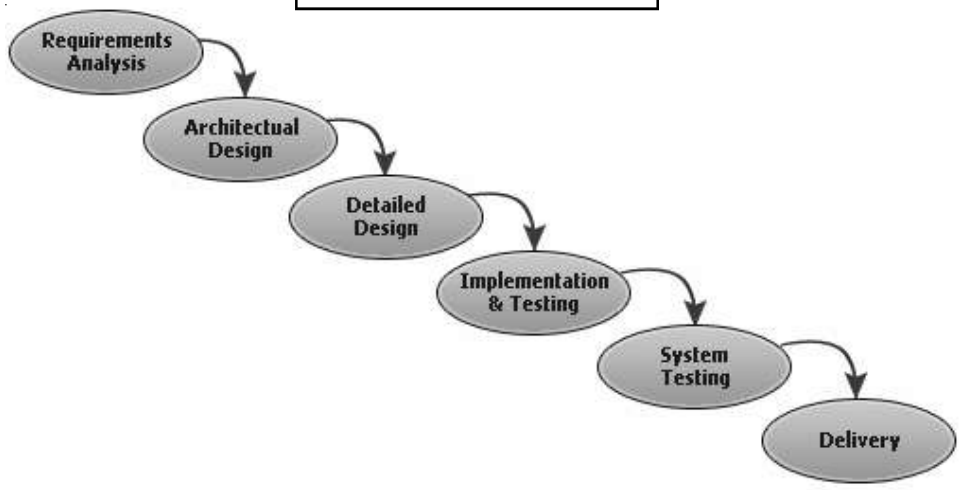
Notes

product versions. These will be improved to increase functionality or improve product quality in subsequent sub-projects. Incremental Delivery is the delivery of increments to the customer/ users at intervals throughout the project timescale.

Incremental delivery is a chief component of most software projects today – it permits us to deliver the *most valuable* elements of a system first, which allows our customers to start getting benefit from the system earlier. As extra features are developed, and additional use cases are enabled, they are delivered to the customers, who get incremental value from those features. This can have a significant impact on ROI projections for a project – and can be the dissimilarity between getting the deal and losing it.

Visualize that we had an application with four main features providing 50, 25, 15 and 10 units of ROI, and each taking one calendar quarter to develop. If we constrain the analysis for our project to a two year payback period (not untypical with software projects), the return versus time is both faster and higher if we delivered each feature incrementally than if we delivered all features when they were all complete. Figure 5.5 represents the incremental delivery.

Figure 5.5: Incremental Delivery



Task In a group of four explain the incremental delivery in your words.

Self Assessment

Fill in the blanks:

13. The radial dimension represents the cumulative …… and each path around the spiral indicates the incremented cost.
14. The development of prototype is an additional cost overhead but still the total cost is lower than that of the software developed using a ……. model.
15. Incremental delivery is a chief component of most software projects today – it permits us to deliver the *most valuable* elements of a …….

Notes

Caselet

New Delhi, Oct 3

Highway developers have approached the Competition Commission of India (CCI) against an NHAI move to restrict the number of bidders for EPC projects to be developed in Uttar Pradesh.

In other words, financial bids from top seven bidders in the technical score list will be invited. This is evident in the tenders for two projects to be developed under the engineering procurement contract (EPC) mode.

This is not the first time such a clause has made it to the NHAI bidding process. In September 2008, a similar clause –which sought to restrict the number of bidders to five — had been dropped from the model Request for Qualification (RFQ) document for highway projects.

The two EPC projects with this clause involve development of about 305 km stretch of NH connecting Raibareilly in Uttar Pradesh. They are Tanda-Raibareilly section (165.5 km stretch with an estimated total project cost of ₹ 692 crore); and Raibareilly-Banda section (140.2 km stretch with ₹ 525 crore project cost).

The National Highways Authority of India (NHAI) has stated in the qualifying tender document for these two projects, "The Authority (NHAI) expects to shortlist up to seven pre-qualified Applicants for participation in the Bid Stage. The Authority, however, reserves the right to increase the number of short-listed pre-qualified Applicants by adding one additional Applicant."

Highway builders lobby body — National Highways Builders Federation (NHBF) — has recently written to the CCI against this move. The NHBF argues that the clause "goes against the prevailing Competition Act".

5.5 Summary

- Traditional approaches focus on predictability where as more fresh approaches prioritize flexibility to changing environments and requirements, in order to maximize business benefits.
- Succeeding sub-project technical plans are based on it and any later amendments should be highlighted.
- In many situations, the input of technical information will not be required, in particular for smaller projects, none manufacturing projects or internet projects.
- Building software is a continuous learning process and the outcome is nothing but a refined version of knowledge that has been collected and processed in the process.
- A project lifecycle plan used to decrease risk of project failure by dividing projects into lesser, more manageable pieces.

5.6 Keywords

SDD: Software Design Description Document

UAT: User Acceptance Test

WBS: Work Breakdown Structure

5.7 Review Questions

1. Approaches vary from highly planned and controlled to fast supple and adaptable. Discuss
2. Explain the project management approaches. Discuss its key tasks or phases in detail.

Notes

3. The project technical plan illustrates a detailed breakdown of the relationships between the major activities required by the project. What do you understand from this statement?
4. A process model can be called a framework of tasks required to develop and build high-quality software. Do you agree? Justify
5. The waterfall model derives its name due to the cascading effect from one phase to the other. Discuss and explain what the project inputs in waterfall model are?
6. What is the difference between the spiral model and prototype model?
7. Explain why Incremental delivery is a chief component of most software projects today?
8. The prototype can be a usable program with limited functionality but cannot be used as a final product. Why?
9. In the V-Model the test activities are spelled out to the similar level of detail as the design activities. Do you agree with this statement why?
10. Write down the advantages and disadvantages of all process models.

Answers: Self Assessment

1.	environment	2.	framework
3.	technical	4.	data
5.	architecture	6.	breakdown
7.	manufacturing	8.	technical
9.	process	10.	software process
11.	maintenance	12.	V-Model
13.	Cost	14.	Waterfall
15.	system first		

5.8 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://www.projectapproach.org/

http://www.projectapproach.org/project_approach.php

Notes

Unit 6: Effort Estimation

CONTENTS

Objectives

Introduction

6.1 Effort Estimation

6.1.1 Elements of a Successful Estimate

6.2 Problems with Estimation

6.3 Estimation Basis

6.3.1 Phase 1: Create an Initial Estimate

6.3.2 Phase 2: Track EQF to Understand the Project Estimate

6.3.3 Phase 3: Use EQF to Manage Project Concerns

6.4 Estimation Techniques

6.5 Albrecht Function Point Analysis

6.5.1 Function Point Analysis

6.6 Functions Mark II

6.7 COCOMO Model

6.7.1 Brief Characteristic of the Method

6.7.2 Levels of the Model

6.7.3 Appraisal of the Model

6.7.4 Modes

6.8 Summary

6.9 Keywords

6.10 Review Questions

6.11 Further Readings

Objectives

After studying this unit, you will be able to:

- Understand RedHat Linux
- Install RedHat Linux
- Prepare RedHat Linux for Installation
- Boot from CD
- Know Graphical Installation Launch
- Set disk partition levels
- Set Boot Loader

Notes

- Discuss the First Boot
- Create the User Account

Introduction

Among the important practical problems in software engineering is software estimation – the estimation of development schedules and the assessment of productivity and quality. Is it possible to apply mathematical and scientific principles to software estimation, so that development schedules, productivity, and quality might be objectively ascertained or estimated rather than being a matter of opinion?

The debate over this question is familiar. In the case of development schedules, for example, many programmers find it self evident that accurate and objective estimates are not possible. One reader of an early version of this paper commented, "Software practitioners know about poor predictability from empirical evidence. I don't need to prove it..."

On the other hand, there are a large number of design methods, development processes, and programming methodologies that claim or hint at objective estimation of development schedules, project complexity, and programmer productivity.

6.1 Effort Estimation

Many People Have Referred to Estimation as a "BLACK ART." This makes some intuitive sense: at first glance, it might seem that estimation is a highly subjective process. One person might take a day to do a task that might only require a few hours of another's time. As a result, when several people are asked to estimate how long it might take to perform a task, they will often give widely differing answers. But when the work is actually performed, it takes a real amount of time; any estimate that did not come close to that actual time is inaccurate.

To someone who has never estimated a project in a structured way, estimation seems little more than attempting to predict the future. This view is reinforced when off-the-cuff estimates are inaccurate and projects come in late. But a good formal estimation process, one that allows the project team to reach a consensus on the estimates, can improve the accuracy of those estimates, making it much more likely that projects will come in on time.

Notes A project manager can help the team to create successful estimates for any software project by using sound techniques and understanding what makes estimates more accurate.

6.1.1 Elements of a Successful Estimate

A sound estimate starts with a Work Breakdown Structure (WBS). A WBS is a list of tasks that, if completed, will produce the final product. The way the work is broken down dictates how it will be done. There are many ways to decompose a project into tasks. The project can be broken down by feature, by project phase (requirements tasks, design tasks, programming tasks, QA tasks, etc.), or by some combination of the two. Ideally, the WBS should reflect the way previous projects have been developed.

A useful rule of thumb is that any project can be broken down into between 10 and 20 tasks. For large projects (for example, a space shuttle), those tasks will be very large ("Test the guidance system"); for small projects (like writing a simple calculator program), the tasks are small ("Build the arithmetic object that adds, multiplies, or divides two numbers"). The team must

Notes

take care in generating the WBS—if the tasks are incorrect, they can waste time going down a wrong path.

Once the WBS is created, the team must create an estimate of the effort required to perform each task. The most accurate estimates are those that rely on prior experience. Team members should review previous project results and find how long similar tasks in previous projects took to complete. Sources of delays in the past should be taken into account when making current estimates. Postmortem reports are a good source of this information.

No estimate is guaranteed to be accurate. People get sick or leave the organization; teams run into unforeseen technical problems; the needs of the organization change. The unexpected will almost certainly happen. Therefore, the goal of estimation is not to predict the future. Instead, it is to gauge an honest, well-informed opinion of the effort required to do a task from those people in the organization who have the most applicable training and knowledge.

If two people widely disagree on how long a task will take, it's likely that the source of that disagreement is that each person made different assumptions about details of the work product or the strategy for producing it. In other words, any disagreement is generally about what is required to perform the task itself, not about the effort required to complete it. For example, given the same vision and scope document for a tool that sets the computer clock, two different developers might come up with wildly different estimates. But it might turn out that one developer assumed that the implementation would have a simple command line interface, while the other assumed that there would be a complete user interface that had to integrate tightly with the operating system's control panel. By helping the programmers discuss these assumptions and come to a temporary resolution about their differences, the project manager can help them agree on a single estimate for the task.

A project manager can help the team create more accurate estimates by reducing the uncertainty about the project. The most effective way to do this is to do a thorough job creating a vision and scope document —the more accurate and detailed it is, the more information the team has to work with when generating their estimate. The project manager can also ensure that the team has reached a consensus on the tasks that must be performed. Finally, the project manager can lead the team in a discussion of assumptions.

Did u know? What is the QA task?

1. Retrieving, Reviewing and analyzing QI data.
2. Identifying barriers and root causes of QI issues and performance gaps.
3. Writes, revises, and verifies quality standards and test procedures for program design and product evaluation to attain quality of software economically and efficiently.
4. Reviews new or modified program, including documentation, diagram, and flow chart, to determine if program will perform according to user request and conform to guidelines.
5. Participating in committees as needed.
6. Collecting and producing written documentation for reporting findings.

Assumptions Make Estimates More Accurate

Once the team has agreed upon a WBS, they can begin to discuss each task so they can come up with an estimate. At the outset of the project, the team members do not have all of the information

Notes

they need in order to produce estimates; nevertheless, they need to come up with numbers. To deal with incomplete information, they must make assumptions about the work to be done. By making assumptions, team members can leave placeholders for information that can be corrected later, in order to make the estimate more accurate.

For the estimates to be most effective, the assumptions must be written down. Important information is discovered during the discussion that the team will need to refer back to during the development process, and if that information is not written down, the team will have to have the discussion all over again. If an assumption turns out to be incorrect, the schedule will need to be adjusted; they will be able to point to the exact cause of the delay by showing that a documented assumption turned out to be incorrect. This will help the project manager explain any resulting schedule delay to others in the organization and avoid that source of delays in the future. The assumptions also provide a way to keep a record of team decisions, share those decisions with others, and find errors in their decisions.

The team should hold a brainstorming session to try to identify as many assumptions as possible. The bigger the list of assumptions, the lower the overall risk for the project. A project manager may get better results from this session by helping the team see how these assumptions can work to their benefit. Any software engineer who has had a bad experience with an estimate that has turned out to be inaccurate will appreciate the value of assumptions: they serve as a warning to the rest of the organization that the estimate is contingent on the assumptions being true. If even one of these assumptions turns out to be incorrect, the team cannot be "blamed" for the incorrect estimate that resulted.

While identifying assumptions is a skill that improves with experience, there are a set of questions that can help a novice team member figure out what assumptions he or she needs to make in order to properly estimate the software. The project manager can use these questions to help lead the discussion to identify the assumptions:

- Are there project goals that are known to the team but not written in any documentation?
- Are there any concepts, terms, or definitions that need to be clarified?
- Are there standards that must be met but will be expensive to comply with?
- How will the development of this project differ from that of previous projects? Will there be new tasks added that were not performed previously?
- Are there technology and architecture decisions that have already been made?
- What changes are likely to occur elsewhere in the organization that could cause this estimate to be inaccurate?
- Are there any issues that the team is known to disagree on that will affect the project?

The last bullet point is especially important. If one team member believes that the project will go down one path while another team member believes the project will go down a different path, the estimates could vary significantly, depending on which team member is correct. For example, one team member may think that a certain off-the-shelf component should be used because that is cheaper than building it, while another team member may believe that they must build that component themselves because they cannot locate one for sale which suits their particular needs. Instead of reaching an impasse, the team can make an assumption—either assume that they will buy the component, or assume that they will build it—which will enable them to move forward with the estimate. It should be easier to reach an agreement at this point because it is not the final decision. By writing down the assumption, the team keeps a record of the disagreement and leaves open the possibility that this will change in the future. The written assumption will be especially useful later while doing a risk assessment for the project plan because there is a risk that the assumption is incorrect.

Notes

In other words, assumptions can help find a compromise to resolve disagreements. If two team members disagree, the team can agree to write down an assumption to temporarily resolve the issue for the purposes of the estimate. It's much easier to get people to agree on one answer temporarily by agreeing to revisit the issue later.

Discussing and writing down the assumptions in a team setting helps the team to identify potential roadblocks. For example, the team may have a genuine disagreement about whether or not to develop a user interface for their clock-setting application. The assumption allows the team to reach a temporary decision, knowing that the final decision is still open. Writing down the assumption allows the team to go back and revise the estimate later if it turns out the assumption is wrong—which means that it is vital that everyone understands that the assumptions are allowed to be incorrect. That way, if the team estimated that they would build a command-line program but later the decision was made to go with a full user interface, everyone will be able to explain why the schedule is delayed.

Another benefit of discussing assumptions is that it brings the team together very early on in the project to make progress on important decisions that will affect development. It's all too common for a developer to make estimates after reading the vision and scope document but before ever talking to anyone about the details of the project. Even if she writes down her assumptions, she has almost certainly missed many others. A moderated discussion of assumptions gives the project team a very effective forum to discuss the unknowns of the project. Identifying unknowns eliminates the source of many inaccuracies in the estimates.

One side effect of writing down the assumptions is that it puts pressure on the senior managers to allow the project to be estimated again if any of the assumptions prove to be incorrect. This is why the project manager should plan on having the vision and scope document updated to include any new assumptions that were identified during the estimation session. This gives the stakeholders and management a chance to review those assumptions and accept or reject them very early on, before they have had a chance to interfere with the development of the software. By having the senior managers review the assumptions, a project manager can reduce a source of future project delays.

Task Explain how assumptions can help find a compromise to resolve disagreements.

Distrust Can Undermine Estimates

Estimates can either be a source of trust or distrust between the project team and their managers. If the team knows that they are given the opportunity to fully justify their estimates, and that they will be allowed to reestimate if the scope of the project changes, that they won't be punished for making an honest mistake in estimation, then each team member will work very hard to produce accurate estimates. In this case, estimation can be an effective tool for team motivation. Estimates are most accurate when everyone on the project team feels that he was actively part of the estimation process. Every team member feels a personal stake in the estimates, and will work very hard to meet any schedule based on those estimates.

Estimation is, by its nature, a politically charged activity in most organizations. When a team is asked to create estimates for work, they are essentially being asked to define their own schedule. Stakeholders need the project completed but usually do not have software engineering experience, so they may not be equipped to understand why a project will take, say, six months instead of three. For this reason, project managers must take care to make the estimation process as open and honest as possible so that the stakeholders can understand what's going on.

Notes

It is common for non-technical people to assume that programmers pad their estimates. They often have a "rule" by which they cut off a third or half of any estimate that they hear, and expect that to be the "real" deadline. They often feel, fairly or not, that the engineering team is "putting one over" on them, mostly because the entire engineering process is, to them, a mystery. This lack of trust causes engineers to automatically pad their estimates, because they know they won't be given enough time otherwise. And even when the situation is not this bad (although it often is), some environment of distrust still exists to a lesser extent in many organizations.

In many of these organizations, there are some kinds of estimates—especially those for quality and requirements tasks—that are particularly likely to not be taken seriously.

Senior managers are often willing to take the programmers' estimates at face value, even when those estimates are clearly padded. This is because, to them, programming is opaque: managers and stakeholders don't understand how code is written, so they assume that all programming tasks are difficult. They are more likely to trust programmers' estimates, even when those estimates are highly padded. Requirements analysts, on the other hand, often produce specifications using nothing more than a word processor. A manager or stakeholder is much more likely to trivialize that work and distrust the estimate because he (incorrectly) feels that he has an intuitive grasp on the work being done. Even worse, in some organizations there is a "rule" that testing should always take exactly one-third (or some other fixed ratio) of the programming time, which causes the testing effort to be shortchanged by only allowing exactly that much time for it instead of the actual amount of time testing would require.

Distrust in a software organization can be a serious, endemic problem. It starts with a kernel of distrust between management and the engineering team; the distrust grows until management simply won't accept the team's estimates. For example, a senior manager may decide that the team plans to spend too much time testing the software, even though the team reached consensus and all team members stand behind the estimates. A project manager must be especially careful to explain this and support that consensus when senior managers start to pick apart the team's estimates. If deadlines are handed down that do not allow enough time for the team to complete the work, it can lead to serious morale problems—and the project manager will be blamed for the delay, often by the same people who caused it in the first place.

An important part of running successful software projects is reaching a common understanding between the engineers, managers, and stakeholders. One way to do that is with a consistent set of practices. This allows the engineers' work to be transparent to the rest of the organization. Similarly, the managers' and stakeholders' needs and expectations must be transparent to the engineers. By having key managers attend the estimation session, a project manager can show them that the estimates are made systematically, using an orderly and sensible process, and that they are not just made up on a whim. When the team is having trouble reaching convergence on a task, team members should bring up examples of past results for tasks of similar size and complexity. This transparency helps everyone present (especially the observers) understand why these estimates come out as they do.

Estimation is the calculated approximation of a result which is usable even if input data may be incomplete or uncertain.

In project management (i.e., for engineering), accurate estimates are the basis of sound project planning. Many processes have been developed to aid engineers in making accurate estimates, such as:

- compartmentalization (i.e., breakdown of tasks),
- parametric estimating,
- structured planning,
- educated assumptions,

Notes

- delphi method,
- identifying dependencies,
- examining historical data,
- estimating each task,
- documenting the results.

Popular estimation processes for software projects include:

- Cocomo
- Proxy Based Estimation (PROBE)
- Wideband Delphi
- The Planning Game
- Program Evaluation and Review Technique (PERT)
- Event chain methodology

When and Why is Estimation Done?

When

- The first estimate is necessary before the start of the project.
- Estimation is a process of gradual refinement.
- It does not finish until the project finishes.

Why

- Estimates are needed and relied upon early
- The functional requirements do not provide a solid background
- It is not immediately known how long it will take to develop the features
  - Particularly if the desired outcomes are genuinely novel.
- Feature Creep is a killer
  - It is the unpredictable yet near-certain change of the functionality as the project progresses.
- Staff ability
  - Estimators
  - Programmers
- Code reuse
  - Is code reused?
  - Is code to be reused?
- Programming language used

Self Assessment

Fill in the blanks:

1. A useful rule of …….. is that any project can be broken down into between 10 and 20 tasks.
2. A project manager can help the team create more accurate …… by reducing the uncertainty about the project.

Notes

3. The …… should hold a brainstorming session to try to identify as many assumptions as possible.
4. Discussing and writing down the assumptions in a team setting helps the team to identify potential……...
5. In project management accurate estimates are the basis of …….. project planning.

6.2 Problems with Estimation

How Inaccurate Estimates lead to Failed Projects

Inaccurate estimates and misconceptions about the estimating process often contribute to failed projects.

An inaccurate estimate leads to infeasible plans. When infeasible plans are implemented, the result is often missed deadlines, inadequate performance and/or poor quality.

Inaccurate estimation can be attributed to:

- Lack of past data on which to base estimates
- Lack of estimating experience
- Lack of a systematic estimation process, sound techniques, or models suited to the project's needs
- Failure to include essential project activities and products within the scope of the estimates
- Unrealistic expectations or assumptions
- Failure to recognize and address the uncertainty inherent in software estimates.

Accommodating Reuse in the Estimates

Estimates for the amount of code that can be reused are a source of error and risk in estimating software cost. PL/PM tends to be overly optimistic about the effort that will be necessary to adapt the existing code. They often assume only a 10% change to a program's design and code will be necessary when in fact the changes are often more than 50%.

Other factors that commonly contribute to size, cost and schedule risks are:

- Optimistic assessments of the software development environment and resources, and
- Misunderstood or constantly changing requirements.

The risk management should be revisited each time the project estimate is updated. The risk management plan update should not concentrate only on the risk areas identified originally.

Example: An off-the-shelf-software package may not be performing as originally assumed.

Each time an estimate is done, the assumptions and inputs should be updated to reflect the most current information.

Use of Past Data

Past data from the PKB and/or from the same project should be used to improve the accuracy of the estimates.

Notes

Granularity of Estimates

Estimates are to be given for each phase in the project.

For maintenance projects,

- For work requests >= (x) days, estimates are to be given for each phase in the work request.
- For work requests < (x) days, estimates may be given for the entire work request. However, it is desirable to give the estimates for each phase even for these work requests.

In addition to this, effort estimates are to be given for the following activities:

- Project management
- Documentation effort
- Reviews
- Configuration Management

Understanding the Tradeoffs

Once a project estimate is generated, some combination of functionality, schedule, cost and resources that can be accepted by management and customers should be produced. This requires a good understanding of the relationships between these variables. Here are some instances of tradeoffs:

- If the schedule is lengthened, the overall cost can be reduced and fewer resources can be used. Sometimes, the schedule has to be lengthened only by a few weeks, to get a benefit. Usually, management and customers may not want a delay, but it is worth to negotiate an acceptable delay. A schedule can be shortened only in three ways:
  - Reducing the functionality (reducing the effort by doing less)
  - Increasing the team size (but only if there are tasks that can be done in parallel to take advantage of this), or
  - Keeping the team size constant but get them to work overtime
- If the functionality cannot be reduced, choosing one of the two remaining alternatives will cost more.
- The difference between the nominal schedule and the shortest schedule for the project may be a little over two months. But to achieve the shortest schedule, the peak team size may have to be increased by almost 10 and this would increase the cost by over₹ 50,00,000.

The decision should be taken considering

- If a 2-month decrease in the schedule is worth the cost
- If 10 additional team members can be found in time to help achieve it.

The decision is project specific; For some projects, a schedule decrease may be required at any cost; for others, it won't be.

All projects may not have such dramatic differences between the estimate options. The size-effort-schedule-resource-cost relationship follows some basic rules.

Notes

Caution Where certain options are available, PL/PM should ensure that everyone involved are properly informed regarding the possible decisions so as to arrive at a feasible and optimistic solution.

Tracking and Reporting Estimation Activities

Comparing planned versus actual estimates over time allows the PL/PM to see how well they are estimating and also to see how their project is changing during project execution. If estimates are never tracked, it is not possible to determine how good the estimates were. The estimation work sheets should form a database of estimates.

This database can be used by the PL/PM to either calibrate estimation models or for purposes of comparison when performing estimates for future projects. Estimation data, both planned and actual, should be provided to the PKB for the benefit of other similar projects.

6.3 Estimation Basis

Project work estimation has three phases: the initial first cut, commonly known as SWAG (scientific wild-ass guess), tracking the estimate against the actual numbers, and using the schedule to see what's happening in your project.

If you've been assigned project estimates, or if your project estimates aren't particularly close to reality, don't fret. Try these techniques to make and learn about your estimates.

6.3.1 Phase 1: Create an Initial Estimate

If you're a project manager, you probably try to estimate the work at the beginning of the project, even if you're assigned a project end date. Sometimes senior managers have trouble hearing what you've said in your estimate. I use one of these three alternatives to creating estimates for the entire project:

1. Provide a date range for the estimate: "We'll be able to release between May 1 and June 15." Some senior managers can't hear the second half of that statement; they only hear May 1. If you work for a manager like that, try either of these other two suggestions.
2. Use the word about to describe the precision of the estimate: "Five people for about nine months or 10 people for about six months." You haven't described an end date, but you have explained the resources you'll require.
3. Provide a confidence level to describe the range of dates: "We have 90% confidence in June 1, and 100% confidence in Aug. 1." In my experience, even the managers who can't hear the "between" estimate can hear my confidence levels.

Once you have a gross estimate at the beginning of the project, you can drill down and create estimates for each of the project components. Whether you try to create precise estimates or choose to use slack buffers to deal with incomplete estimates, you will have some project estimate total.

The problem with estimates is that they are guesses. They're the best guesses we can make, but they're still guesses.

Notes As the project unfolds, you'll be able to acquire feedback on how well you estimated using the second part of estimation, the EQF, or estimation quality factor.

Notes

6.3.2 Phase 2: Track EQF to Understand the Project Estimate

As you continue to manage the project, track your initial completion date estimate. Each month (or in a short project, each week), take five minutes out of your project team meeting and ask, "When do you think we will finish the project?" Track that estimate on a chart set up with the release dates on the Y axis, and the date that you asked the question on the X axis.

There are two good reasons for asking this question. First, you continue to focus your project staff on completing the project. People tend to work on what you, the project manager, focus on. Second, by asking your project staff, you can discover the various confidences they have in the release date. When you look at the EQF chart, you can see if people are concerned that the project won't meet its release date, or if they're feeling confident about meeting or beating the release date. Then you can deal with their concerns or your own.

When you track EQF with your project team, you're learning more about the project and using EQF to learn how good your initial estimate was.

6.3.3 Phase 3: Use EQF to Manage Project Concerns

We use the slope of the EQF to make queries like, "Tell me what's happened in the project to make you think we will meet/beat/miss the date." When people become more optimistic or pessimistic, we want to know why. The EQF not only gives me feedback on my initial estimate; it also gives me another technique to discuss the project state with the project team.

And once we understand the project team's concerns, we can deal with them or elevate those concerns to my management.

If you're using only one of these techniques to estimate and manage your projects, consider adding the other two. Every project worth completing has some uncertainty. EQF is a great technique for displaying project uncertainty and for understanding why the team is uncertain about the project.

Self Assessment

Fill in the blanks:

6. An …….. estimate leads to infeasible plans.
7. …….. tends to be overly optimistic about the effort that will be necessary to adapt the existing code.
8. All projects may not have such dramatic differences between the …… options.
9. Once you have a gross estimate at the …… of the project, you can drill down and create estimates for each of the project components.

6.4 Estimation Techniques

For selecting an estimation technique, the following factors need to be considered:

- Whether the assumptions of the estimation technique match the project.
- Whether the data required by the method is available from a reliable sources.
- Whether the activities covered by the method match the planned activities of the project.

Some of the estimation methods being followed by different types of projects are given in Table 6.1. Please note that these are indicative and not exhaustive. As and when more information is acquired from the projects, this table would be updated.

Notes

Table 6.1: Example of Project Types and Relevant Estimation Methods

Type of Project	Estimation Methods Used	Performance Factors Used for Adjusting Estimates
Development	• Size – KLOC/ FP/ Use Case Points • Effort / schedule is derived from Size • Activity Based Approach • Number of requirements used to estimate calendar time for systems requirements specification, design and integration / system testing phases • Parametric Model Approach • Size – KLOC/ FP/ Use Case Points • Effort / schedule – using COCOMO • Effort / schedule / resources – using Knowledge Plan	• Productivity (person days required to produce a screen of given complexity) • Application domain familiarity • Familiarity with the technology
Development (GUI)	Simple Estimating Relationships • Size measure – no. of screens (of complexity Low/ Medium/ Very High) used for estimating effort / schedule in Construction (Coding/ Unit testing) phase • Function Point • Activity Based Approach • Number of requirements used to estimate calendar time for systems requirements specification, design and integration / system testing phases • Effort / schedule / resources – using Knowledge PLAN	• Productivity (person days required to produce a screen of given complexity) • Application domain familiarity • Familiarity with the technology
Development (Reverse Engineering)	Analogy • Size estimation in terms of total LOC • Productivity (in LOC/day) from a previous reverse engineering project on similar platform and LOB was used • Effort / schedule / resources – using Knowledge PLAN(Refer Estimation Binder under Information Sharing for more details on Knowledge PLAN)	• The two projects were compared in terms of tool usage, knowledge transfer from customer, etc. and productivity value was adjusted accordingly.
Development (Conversion)	Simple Estimating Relationships • Size (in LOC) from the existing Source Code • Effort estimation using the obtained size and the Conversion Language Table • Effort / schedule / resources – using KnowledgePLAN	

Contd.....

Notes

Partial Life Cycle (Coding and Unit Testing) and Regular maintenance	Simple Estimating Relationships for coding new programs • Size estimation in terms of LOC • Use productivity (LOC/day) to get effort estimate • Derive schedule based on effort estimate, customer's schedules and manpower available. • Effort / schedule / resources – using Knowledge PLAN	• Productivity values from earlier work requests in the same project and / or from other similar projects are used
Regular maintenance	Initial Team size estimation Capture following parameters • Database used • # of Interfaces • Total Number of programs / Screens / Reports • Total number of Tables / Files • Lines of code • Tech Arch • FTE (Full Time Employees) currently maintaining the system (includes all employees on the system) • System Documentation quality • Backlog in terms of man months • Average enhancements per month • Average production calls per month Based on number of FTEs and the back log determine the number of resources needed to maintain the application Activity Based approach for modifying existing programs • Size estimation in terms of programs / total LOC • Identify the activities to be performed for the modification • Use performance data from earlier work requests in the same project • Derive schedule based on effort estimate, customer's schedules and manpower available. • Use Knowledge PLAN to derive FTE required for support and corrective maintenance activity for a given period	The following may be used to adjust the effort/ schedule estimates: • Extent of testing required • Interdependencies / impact on other modules • Complexity of the modification • Skills of the project team • Wait time expected based on previous work requests data • Rework expected based on previous work requests data

Task In a group of four explain what have you understood by this table.

6.5 Albrecht Function Point Analysis

6.5.1 Function Point Analysis

Software systems, unless they are thoroughly understood, can be like an ice berg. They are becoming more and more difficult to understand. Improvement of coding tools allows software

Notes

developers to produce large amounts of software to meet an ever expanding need from users. As systems grow a method to understand and communicate size needs to be used. Function Point Analysis is a structured technique of problem solving. It is a method to break systems into smaller components, so they can be better understood and analyzed.

Function points are a unit measure for software much like an hour is to measuring time, miles are to measuring distance or Celsius is to measuring temperature. Function Points are an ordinal measure much like other measures such as kilometers, Fahrenheit, hours, so on and so forth

Human beings solve problems by breaking them into smaller understandable pieces. Problems that may appear to be difficult are simple once they are broken into smaller parts — dissected into classes. Classifying things, placing them in this or that category, is a familiar process. Everyone does it at one time or another — shopkeepers when they take stock of what is on their shelves, librarians when they catalog books, secretaries when they file letters or documents. When objects to be classified are the contents of systems, a set of definitions and rules must be used to place these objects into the appropriate category, a scheme of classification. Function Point Analysis is a structured technique of classifying components of a system. It is a method to break systems into smaller components, so they can be better understood and analyzed. It provides a structured technique for problem solving.

In the world of Function Point Analysis, systems are divided into five large classes and general system characteristics. The first three classes or components are External Inputs, External Outputs and External Inquires each of these components transacts against files therefore they are called transactions. The next two Internal Logical Files and External Interface Files are where data is stored that is combined to form logical information. The general system characteristics assess the general functionality of the system.

Function Point Analysis was developed first by Allan J. Albrecht in the mid 1970s. It was an attempt to overcome difficulties associated with lines of code as a measure of software size, and to assist in developing a mechanism to predict effort associated with software development. The method was first published in 1979, then later in 1983 . In 1984 Albrecht refined the method and since 1986, when the International Function Point User Group (IFPUG) was set up, several versions of the Function Point Counting Practices Manual have been published by IFPUG. The current version of the IFPUG Manual is 4.1. A full function point training manual can be downloaded from this website.

Objectives of Function Point Analysis

Frequently the term end user or user is used without specifying what is meant. In this case, the user is a sophisticated user. Someone that would understand the system from a functional perspective — more than likely someone that would provide requirements or does acceptance testing.

Since Function Points measures systems from a functional perspective they are independent of technology. Regardless of language, development method, or hardware platform used, the number of function points for a system will remain constant. The only variable is the amount of effort needed to deliver a given set of function points; therefore, Function Point Analysis can be used to determine whether a tool, an environment, a language is more productive compared with others within an organization or among organizations. This is a critical point and one of the greatest values of Function Point Analysis.

Function Point Analysis can provide a mechanism to track and monitor scope creep. Function Point Counts at the end of requirements, analysis, design, code, testing and implementation can be compared. The function point count at the end of requirements and/or designs can be compared to function points actually delivered. If the project has grown, there has been scope creep. The amount of growth is an indication of how well requirements were gathered by and/or

Notes

communicated to the project team. If the amount of growth of projects declines over time it is a natural assumption that communication with the user has improved.

Characteristic of Quality Function Point Analysis

Function Point Analysis should be performed by trained and experienced personnel. If Function Point Analysis is conducted by untrained personnel, it is reasonable to assume the analysis will done incorrectly. The personnel counting function points should utilize the most current version of the Function Point Counting Practices Manual.

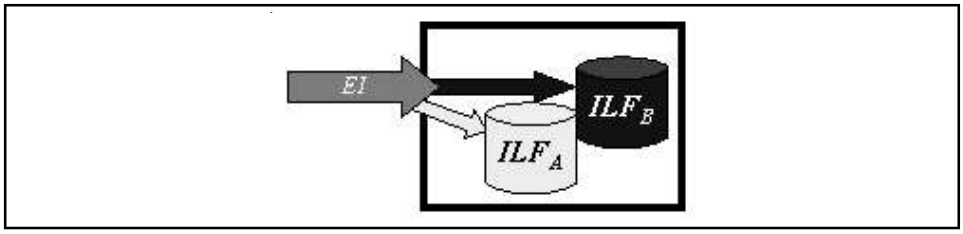
Current application documentation should be utilized to complete a function point count. For example, screen formats, report layouts, listing of interfaces with other systems and between systems, logical and/or preliminary physical data models will all assist in Function Points Analysis.

The task of counting function points should be included as part of the overall project plan. That is, counting function points should be scheduled and planned. The first function point count should be developed to provide sizing used for estimating.

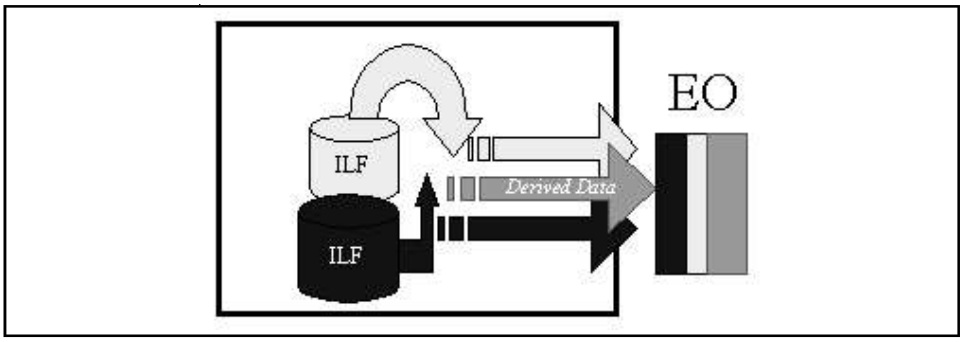
The Five Major Components

Since it is common for computer systems to interact with other computer systems, a boundary must be drawn around each system to be measured prior to classifying components. This boundary must be drawn according to the user's point of view. In short, the boundary indicates the border between the project or application being measured and the external applications or user domain. Once the border has been established, components can be classified, ranked and tallied.

- ***External Inputs (EI):*** An elementary process in which data crosses the boundary from outside to inside. This data may come from a data input screen or another application. The data may be used to maintain one or more internal logical files. The data can be either control information or business information. If the data is control information it does not have to update an internal logical file. The graphic represents a simple EI that updates 2 ILF's (FTR's).

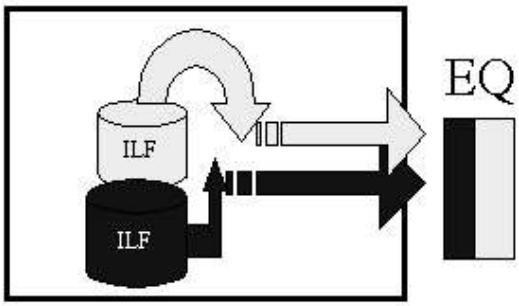


- ***External Outputs (EO):*** An elementary process in which derived data passes across the boundary from inside to outside. Additionally, an EO may update an ILF. The data creates reports or output files sent to other applications. These reports and files are created from one or more internal logical files and external interface file. The following graphic represents on EO with 2 FTR's there is derived information (green) that has been derived from the ILF's



Notes

- ***External Inquiry (EQ):*** An elementary process with both input and output components that result in data retrieval from one or more internal logical files and external interface files. The input process does not update any Internal Logical Files, and the output side does not contain derived data. The graphic below represents an EQ with two ILF's and no derived data.



- ***Internal Logical Files (ILF's):*** A user identifiable group of logically related data that resides entirely within the applications boundary and is maintained through external inputs.
- ***External Interface Files (EIF's):*** A user identifiable group of logically related data that is used for reference purposes only. The data resides entirely outside the application and is maintained by another application. The external interface file is an internal logical file for another application.

All components are rated as Low, Average or High

After the components have been classified as one of the five major components (EI's, EO's, EQ's, ILF's or EIF's), a ranking of low, average or high is assigned. For transactions (EI's, EO's, EQ's) the ranking is based upon the number of files updated or referenced (FTR's) and the number of data element types (DET's). For both ILF's and EIF's files the ranking is based upon record element types (RET's) and data element types (DET's). A record element type is a user recognizable subgroup of data elements within an ILF or EIF. A data element type is a unique user recognizable, non recursive, field.

Each of the following tables assists in the ranking process (the numerical rating is in parentheses). For example, an EI that references or updates 2 File Types Referenced (FTR's) and has 7 data elements would be assigned a ranking of average and associated rating of 4. Where FTR's are the combined number of Internal Logical Files (ILF's) referenced or updated and External Interface Files referenced.

Did u know? What is Internal Logical Files?

ILF stands for "Internal Logical File". In my words, ILFs represent data that is stored and maintained within the boundary of the application you are counting. When counting ILFs you are basically counting the data functions that your application is being built to maintain.

EI Table

FTR's	DATA ELEMENTS		
	1-4	5-15	> 15
0-1	Low	Low	Ave
2	Low	Ave	High
3 or more	Ave	High	High

Notes

Shared EO and EQ Table

FTR's	DATA ELEMENTS		
	1-5	6-19	> 19
0-1	Low	Low	Ave
2-3	Low	Ave	High
> 3	Ave	High	High

Values for Transactions

Rating	VALUES		
	EO	EQ	EI
Low	4	3	3
Average	5	4	4
High	7	6	6

Like all components, EQ are rated and scored. Basically, an EQ is rated (Low, Average or High) like an EO, but assigned a value like and EI. The rating is based upon the total number of unique (combined unique input and out sides) data elements (DET) and the file types referenced (FTR) (combined unique input and output sides). If the same FTR is used on both the input and output side, then it is counted only one time. If the same DET is used on both the input and output side, then it is only counted one time.

For both ILF's and EIF's the number of record element types and the number of data elements types are used to determine a ranking of low, average or high. A Record Element Type is a user recognizable subgroup of data elements within an ILF or EIF. A Data Element Type (DET) is a unique user recognizable, non recursive field on an ILF or EIF.

RET's	DATA ELEMENTS		
	1-19	20 - 50	> 50
1	Low	Low	Ave
2-5	Low	Ave	High
> 5	Ave	High	High

Rating	Values	
	ILF	EIF
Low	7	5
Average	10	7
High	15	10

The counts for each level of complexity for each type of component can be entered into a table such as the following one. Each count is multiplied by the numerical rating shown to determine the rated value. The rated values on each row are summed across the table, giving a total value for each type of component. These totals are then summed across the table, giving a total value for each type of component. These totals are then summoned down to arrive at the Total Number of Unadjusted Function Points.

Notes

Type of Component	Complexity of Components			
	Low	Average	High	Total
External Inputs	___ x 3 = ___	___ x 4 = ___	___ x 6 = ___	
External Outputs	___ x 4 = ___	___ x 5 = ___	___ x 7 = ___	
External Inquiries	___ x 3 = ___	___ x 4 = ___	___ x 6 = ___	
Internal Logical Files	___ x 7 = ___	___ x 10 = ___	___ x 15 = ___	
External Interface Files	___ x 5 = ___	___ x 7 = ___	___ x 10 = ___	
		Total Number of Unadjusted Function Points		___
		Multiplied Value Adjustment Factor		___
		Total Adjusted Function Points		___

The Value Adjustment Factor (VAF) is based on 14 General System Characteristics (GSC's) that rate the general functionality of the application being counted. Each characteristic has associated descriptions that help determine the degrees of influence of the characteristics. The degrees of influence range on a scale of zero to five, from no influence to strong influence. The IFPUG Counting Practices Manual provides detailed evaluation criteria for each of the GSC'S, the table below is intended to provide an overview of each GSC.

General System Characteristic		Brief Description
1.	Data communications	How many communication facilities are there to aid in the transfer or exchange of information with the application or system?
2.	Distributed data processing	How are distributed data and processing functions handled?
3.	Performance	Was response time or throughput required by the user?
4.	Heavily used configuration	How heavily used is the current hardware platform where the application will be executed?
5.	Transaction rate	How frequently are transactions executed daily, weekly, monthly, etc.?
6.	On-Line data entry	What percentage of the information is entered On-Line?
7.	End-user efficiency	Was the application designed for end-user efficiency?
8.	On-Line update	How many ILF's are updated by On-Line transaction?
9.	Complex processing	Does the application have extensive logical or mathematical processing?
10.	Reusability	Was the application developed to meet one or many user's needs?
11.	Installation ease	How difficult is conversion and installation?
12.	Operational ease	How effective and/or automated are start-up, back-up, and recovery procedures?
13.	Multiple sites	Was the application specifically designed, developed, and supported to be installed at multiple sites for multiple organizations?
14.	Facilitate change	Was the application specifically designed, developed, and supported to facilitate change?

Once all the 14 GSC's have been answered, they should be tabulated using the IFPUG Value Adjustment Equation (VAF)

14 where: Ci = degree of influence for each General System Characteristic

VAF = 0.65 + [(Ci) / 100] .i = is from 1 to 14 representing each GSC.

i =1, A = is summation of all 14 GSC's.

Notes

The final Function Point Count is obtained by multiplying the VAF times the Unadjusted Function Point (UAF).

FP = UAF * VAF

Albrecht Function Point Analysis

Count the following parameters of the problem:

A: number of external inputs

B: number of external outputs

C: number of inquiries (interactive external inputs)

D: number of external (i.e. permanent) files

E: number of internal (i.e. temporary) files

and decide whether each is Simple, Average, or Complex,

then calculate their weighted sum according to this table

	Simple	Average	Complex
A	3	4	6
B	4	5	7
C	3	4	6
D	7	10	15
E	5	7	10

That gives the UFC (Unadjusted Function Count)

Then, rate each of these factors on a scale of 0 to 5

where 0=irrelevant and 5 = essential, and add all the factors up.

1	Reliable backup and recovery required
2	Data communications
3	Distributed functions
4	High Performance (time efficiency) required
5	? Heavily used configuration
6	On-line (interactive) data entry
7	Must be easy to use
8	On-line update
9	Complex user interface
10	Complex process
11	? Reusability
12	Ease of installation
13	Multiple installation sites
14	Easy to modify

and compute 0.65 + 0.1 × the-sum-of-all-those-weights

That gives the TCF (Technical Complexity Factor)

Then the number of Function Points, FP = UFC × TCF

Notes

and FP is supposed to be proportional to effort or time required.

(One Function Point = 2 days !)

Self Assessment

Fill in the blanks:

10. Function Point Analysis is a …….. technique of problem solving.
11. The value adjustment factor (VAF) is based on 14 general system characteristics (GSCs) that rate the general ……. of the application being counted.
12. The task of counting ……. should be included as part of the overall project plan.

6.6 Functions Mark II

The MK II (Mark 2) Method is one of the software sizing methods in functional point group of measurements. This is a method for analysis and measurement of information processing applications based on end user functional view of the system. The MK II Method (ISO/IEC 20968 Software engineering—MK II Function Point Analysis—Counting Practices Manual) is one of five currently recognized ISO standards for functionally sizing software.

The MK II Method were defined by Charles Symons in book published in 1991. UK Software Metrics Association is now responsible for the method and its continuing development. The functional user requirements of the software are identified and each one is categorized into one of there types: inputs, exits and objects. In order to determine functional size of system these functional requirements are counted.

Nolan, Norton & Co., part of KPMG Management Consulting, hired Charles Symons in 1984 to advise clients on methods to improve their systems development performance. In the course of doing this he claims to have discovered weaknesses in Albrecht's approach to function point analysis and developed the MK II approach to overcome them. By 1987 it became a licensed product and is actively marketed.

Symons claims that the Albrecht approach suffers from the following weaknesses:

- It is often difficult to identify the components of an application. For example, what is a logical file? In 1984, there was insufficient guidance to make these determinations.
- Albrecht had assigned weights to function point components based on "debate and trial."
- The above two criticisms were also leveled at the identification and weighting of Value Adjustment Factors.
- Albrecht did not provide a means of accounting for internal complexity. This is the same problem regarding algorithms that the feature points technique was developed to address.
- When small systems are combined into larger applications, Albrecht's approach makes the total function point count less than the sum of the components.

The principal conceptual difference between the two methods is the calculation of Information Processing Size, which corresponds to the Albrecht Unadjusted Function Points. Symons decomposes the application being counted into a collection of logical transactions. Each transaction consists of an input, a process and an output. For each transaction, Unadjusted Function Points (UFP) become a function of the number of input data element-types, entity-types referenced and output data element-types. The UFPs for the entire system are then summed.

Notes

Detractors of Symons' work say that MK II is simply a way to overstate the value of the monolithic systems that Nolan, Norton & Co.'s clients were building at the time. They consider it a political concession that will undermine the competitiveness of those clients in the long run. More moderate critics believe that Symons identified some function point issues that practitioners were already addressing. However, he chose to deal with them in a manner that would lead to a proprietary Nolan, Norton & Co. product, instead of strengthening the original Albrecht work.

Grant Rule supplied information on the current state of MK II. He reported that the technique has been in the public domain since 1991. The design authority is the United Kingdom Software Metrics Association (UKSMA) who were formerly known as the United Kingdom Function Point Users Group (UFPUG).

6.7 COCOMO Model

Constructive Cost Model (COCOMO) is a method for assessing the cost of a software package. COCOMO, Constructive Cost Model is static single-variable model. Barry Boehm introduced COCOMO models. There are three levels of COCOMO model basic, immediate and detailed.

6.7.1 Brief Characteristic of the Method

- COCOMO (Constructive Cost Model) is a combination of parametric estimation equation and weighting method. Based on the estimated instructions (Delivered Source Instructions DSI), the effort is calculated by taking into consideration both the attempted quality and the productivity factors.
- COCOMO is based on the conventional top-down programming and concentrates on the number of instructions.

6.7.2 Levels of Model

- ***Basic COCOMO:*** Basic COCOMO model is static single-valued model that computes software development effort (and cost) as a function of program size expressed in estimated lines of code. By means of parametric estimation equations (differentiated according to the different system types) the development effort and the development duration are calculated on the basis of the estimated DSI. The breakdown to phases is realized in percentages. In this connection it is differentiated according to system types (organic-batch, semidetached-on-line, embedded-real-time) and project sizes (small, intermediate, medium, large, very large).
- ***Intermediate COCOMO:*** Intermediate COCOMO model computes software development effort as a function of program size and a set of "cost drivers" that include subjective assessments of product, hardware, personnel, and project attributes. The estimation equations are now taking into consideration (apart from DSI) 15 influence factors; these are product attributes (like software reliability, size of the database, complexity), computer attributes (like computing time restriction, main memory restriction), personnel attributes (like programming and application experience, knowledge of the programming language), and project attributes (like software development environment, pressure of development time).

Notes The degree of influence can be classified as very low, low, normal, high, very high, extra high; the multipliers can be read from the available tables.

Notes

- ***Detailed COCOMO:*** Advanced COCOMO model incorporates all characteristics of the intermediate version with an assessment of the cost driver's impact on each step, like analysis, design, etc. In this case the breakdown to phases is not realized in percentages but by means of influence factors allocated to the phases. At the same time, it is differentiated according to the three levels of the product hierarchy (module, subsystem, system); product-related influence factors are now taken into consideration in the corresponding estimation equations.

6.7.3 Appraisal of the Model

Applications of COCOMO

- ***Medium and Large Projects:*** For small projects, the attempt for an estimation according to intermediate and detailed COCOMO is too high; but the results from basic COCOMO alone are not sufficiently exact.
- ***Technical Application:*** For software projects developing commercial applications, COCOMO usually comes up with overstated effort estimation values therefore COCOMO is only applied for the development of technical software. This circumstance is due to the fact that the ratio DSI and man months implemented in the COCOMO estimation equation fits the efficiency rate in a technical development; with regard to commercial software development a higher productivity rate DSI/man-month can be assumed.

Strong and Weak Points of the Model and Possible Remedial Measures

- ***Estimation Base "Delivered Source Instructions":*** By means of estimation base instructions (DSI) it was attempted to diminish the great uncertainties and problems in connection with the traditional estimation base LOC. However, some problems remain: the ambiguity of a DSI estimation and for the development effort the DSI are-based on modern software engineering methods-no longer of great importance since the effort increasingly occurs during the early activities and DSI will only be effective towards the end of the development process; DSI as well as LOC depends on the selected programming language (an Ada adoption to COCOMO is already available, however). A remedy can be achieved by the weighting of instructions according to their various types compiler, data description, transformation, control, and I/O instruction, data description instructions (differentiated according to integration degree, message/data object, modification degree) and processing instructions (differentiated according to batch/on-line, modification degree, complexity, language)).
- ***Macro and Micro Estimation:*** By means of the different levels of the model, COCOMO makes it possible to realize both a macro estimation by means of Basic COCOMO and a micro estimation by means of Intermediate COCOMO and Detailed COCOMO. The micro estimation allows the effort allocation to activities and functional units. However, method COCOMO is not only based on a software life cycle deviating from the V-Model but also on another system structure. Therefore, in order to list individual efforts for sub models, (sub-) activities, and (sub-) products, it is necessary to adjust these items of method COCOMO to the V-Model concept.
- ***Influence Factors/Objectivity:*** In the effort estimation, COCOMO takes into consideration the characteristics of the project, the product, and the personnel as well as of the technology. In order to achieve an objective evaluation of these influence factors, COCOMO offers exact definitions. The quantification of influence factors represents a certain problem, though which has a strong impact on the quality of the estimation method and on the required DSI information.

Notes

- ***Range of Application:*** By differentiating the estimation equations according to project sizes and system types, the range of application for method COCOMO is a wide one. It is also one of the few estimation methods offering-apart from the support for development projects-support for the effort estimation of SWMM tasks as well (also by parametric estimation equations) as for the estimation of the project duration.
- ***Tool Support:*** Computer-based support is required for Intermediate and Detailed COCOMO, based on the quantity problem (differentiation of influence factors on phases and sub products.

6.7.4 Modes

COCOMO can be applied to the following software project's categories.

Organic Mode

These projects are very easy and have small team size. The team has a good application experience work to a set of less than inflexible/rigid requirements. A thermal analysis program developed for a heat transfer group is an example of this.

Semi-detached Mode

These are intermediate in size and complexity. Here the team has mixed experience to meet up a mix of rigid and less than rigid requirements. A transaction processing system with fixed requirements for terminal hardware and database software is an example of this.

Embedded Mode

Software projects that must be developed within a set of tight hardware, software, and operational constraints. For example, flight control software for aircraft.

Self Assessment

Fill in the blanks:

13. The …….. Method were defined by Charles Symons in book published in 1991.
14. Basic …….. is static single-valued model that computes software development effort (and cost) as a function of program size expressed in estimated lines of code.
15. The micro estimation allows the effort allocation to activities and……..

Caselet

Nasscom Reiterates 16-18% Growth Estimate in IT Sector

Thiruvananthapuram Sept. 30:

The National Association of Software and Service Companies has stuck to its 16-18 per cent annual growth estimate in the IT sector and termed the outlook positive.

Speaking to newspersons here after the Executive Council met in Kerala for the first time, the Nasscom Chairman, Mr Rajendra Pawar, and the President, Mr Som Mittal, announced the setting up of a regional chapter in the State.

Contd.....

Notes

This is the first time that the grouping has gone to a town other than metros and three bellwether industry locations in south India.

According to Mr Mittal, the term "recession" is being loosely bandied about to qualify the times we are in.

This is unwarranted, he said, adding that unlike in 2008, the situation can be only described "uncertain" at best. The private sector in the US is still doing well.

Mr Pawar concurred with this view, substantiating it in terms of numbers. In 1990, software exports amounted to $1 billion.

This was followed by a decade described as the "golden age" when, during 1990-2000, the exports figure jumped to $8.2 billion.

The subsequent decade, which has just passed, witnessed a growth nine times over to $75 billion.

In doing so, it survived the 9/11 tragedy, the Mumbai terrorist attack aftermath and the year 2008 recession.

So there's no reason to doubt the resilience of the industry in doing what it has been till now.

According to Mr V. K. Mathews, Executive Chairman, IBS Group, and a member of the Executive Council, the India BPO sector still enjoyed the cost advantage.

"Uncertain" times also call for belt tightening, which will bring the US and European companies looking to offshore work even more.

Mr Mittal conceded that the deal sizes may have shrunk in size, but the volumes are being maintained.

"We are finding traction in transformational deals as companies increasingly look to move applications to cloud, mobile and social media platforms," he said.

6.8 Summary

- MANY PEOPLE HAVE REFERRED TO ESTIMATION AS A "BLACK ART." This makes some intuitive sense: at first glance, it might seem that estimation is a highly subjective process.
- Inaccurate estimates and misconceptions about the estimating process often contribute to failed projects.
- When infeasible plans are implemented, the result is often missed deadlines, inadequate performance and/or poor quality.
- Project work estimation has three phases: the initial first cut, commonly known as SWAG (scientific wild-ass guess), tracking the estimate against the actual numbers, and using the schedule to see what's happening in your project.
- If you've been assigned project estimates, or if your project estimates aren't particularly close to reality, don't fret. Try these techniques to make and learn about your estimates.
- Software systems, unless they are thoroughly understood, can be like an ice berg. They are becoming more and more difficult to understand.
- Improvement of coding tools allows software developers to produce large amounts of software to meet an ever expanding need from users.

Notes

- MK II for analysis and measurement of information processing applications based on end user functional view of the system.
- Constructive Cost Model is static single-variable model. Barry Boehm introduced COCOMO models.

6.9 Keywords

COCOMO: Constructive Cost Model

EI: External Inputs

EIF's: External Interface Files

EO: External Outputs

EQ: External Inquiry

IFPUG: International Function Point User Group

ILF's: Internal Logical Files

PERT: Program Evaluation and Review Technique

PROBE: Proxy Based Estimation

6.10 Review Questions

1. How Inaccurate Estimates lead to Failed Projects?
2. A sound estimate starts with a work breakdown structure (WBS). Explain.
3. An inaccurate estimate leads to infeasible plans. Discuss.
4. Project work estimation has three phases, explain each in detail.
5. Give characteristic of Quality Function Point Analysis.
6. Function Point Analysis was developed first by Allan J. Albrecht in the mid 1970s. Discuss.
7. For selecting an estimation technique, what are the factors need to be considered? Explain
8. The MK II (Mark 2) Method is one of the software sizing methods in functional point group of measurements. Analyze.
9. Examine how Constructive Cost Model (COCOMO) is a method for assessing the cost of a software package?
10. Explain what are the Strong and Weak Points and possible Remedial Measures of a model?

Answers: Self Assessment

1.	Thumb	2.	Estimates
3.	Team	4.	Roadblocks
5.	Sound	6.	Inaccurate
7.	PL/PM	8.	Estimate
9.	Beginning	10.	Structured
11.	Functionality	12.	function points

13. MK II
14. COCOMO model
15. functional units

Notes

6.11 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://www.chemuturi.com/Test%20Effort%20Estimation.pdf

http://scrummethodology.com/scrum-effort-estimation-and-story-points/

Notes

Unit 7: Activity Planning

CONTENTS

Objectives

Introduction

7.1 Aim of Activity Planning

7.2 Project Schedule

7.2.1 Schedule Inputs

7.2.2 Scheduling Tools

7.2.3 Project Review

7.3 Summary

7.4 Keywords

7.5 Review Questions

7.6 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize the meaning and scope of activity planning
- Describe the goals and objectives of project schedule

Introduction

Careful planning up front in your software project reduces the risk of costly rework or failure down the road. A series of planning activities must be completed before the project moves forward. Project sponsors frequently minimize the value of planning, desiring instead to get the project underway immediately. As project manager, your job is to act in the best interest of your sponsor, even if that means standing firm while you develop your project plan.

7.1 Aim of Activity Planning

Project time management includes two high-level groups of processes for planning and scheduling project activities and tasks necessary for timely completion of the project. Project activities planning and scheduling is the first process group of project time management. Developing the project implementation schedule is the second group. In this unit we will review the planning and scheduling process group. We will talk about the key actions the project manager must take to undertake project activities planning and scheduling in an efficient manner.

Review Scope

Dick Billows, project management consultant and trainer at 4PM, recommends nailing down the project scope before anything else. The scope outlines the main objectives and establishes boundaries for the project. Boundaries are important to frame the project by detailing what things are inside and outside the scope of the project.

Notes

Caution Skipping this step increases the risk of project failure due to insufficient information from which to formulate your overall project plan.

Define Success

An often overlooked software project planning activity is defining what success looks like before the project begins. In a 2007 Gartner research study on why information technology projects succeed or fail, survey participants indicated that, in part, failure resulted from inadequately defining "critical success factors" at the start of the project. Without stakeholder agreement, some degree of failure is likely, especially in complex projects.

Select Team

Selecting the right team members depends on a clear understanding of the project scope and success criteria. Team members define the work that needs to be done throughout the course of the project. They provide estimates of how much time each piece of work takes and what they need in order to complete their assignments. Team members define all dependencies between each work product. Highly experienced and qualified experts make the planning process more reliable.

Finalize Project Plan

As project manager, your job is to tie together all the information you have collected from your sponsor, team members and other stakeholders. Combine this information with your project management requirements into a comprehensive document or computerized project management system, and present the overall project plan to your sponsor. Negotiate resolutions to outstanding issues and fine-tune the plan. Obtain sponsor approval, funding and team members, and then commence with the project.

Project Planning Iterations

Project planning activities are never fully done until the project is complete. Throughout the project, issues arise that require you to adjust the plan. Just as with the initial planning phase, you must seek input from team members to understand the nature of the work and how much time to assign to it, defining requirements and dependencies accordingly. Present the revised plan to your sponsor for approval. Never fear negotiating things you need to complete the project successfully.

Task In your words explain what have you understood by aims in activity planning.

Self Assessment

Fill in the blanks:

1. Developing the project implementation schedule is the …….. group.
2. Without …….. agreement, some degree of failure is likely, especially in complex projects.

Notes

3. Team members define all dependencies between each work product. Highly experienced and qualified experts make the …….. more reliable.
4. Obtain sponsor approval, funding and……, and then commence with the project.
5. Present the revised plan to your …….. for approval. Never fear negotiating things you need to complete the project successfully.

7.2 Project Schedule

Can you imagine starting a long car trip to an unfamiliar destination without a map or navigation system? You're pretty sure you have to make some turns here and there, but you have no idea when or where, or how long it will take to get there. You may arrive eventually, but you run the risk of getting lost, and feeling frustrated, along the way.

Essentially, driving without any idea of how you're going to get there is the same as working on a project without a schedule. No matter the size or scope of your project, the schedule is a key part of project management. The schedule tells you when each activity should be done, what has already been completed, and the sequence in which things need to be finished.

Luckily, drivers have fairly accurate tools they can use. Scheduling, on the other hand, is not an exact process. It's part estimation, part prediction, and part 'educated guessing.'

Because of the uncertainty involved, the schedule is reviewed regularly, and it is often revised while the project is in progress.

Notes It continues to develop as the project moves forward, changes arise, risks come and go, and new risks are identified. The schedule essentially transforms the project from a vision to a time-based plan.

Schedules also help you do the following:

- They provide a basis for you to monitor and control project activities.
- They help you determine how best to allocate resources so you can achieve the project goal.
- They help you assess how time delays will impact the project.
- You can figure out where excess resources are available to allocate to other projects.
- They provide a basis to help you track project progress.

With that in mind, what's the best way of building an accurate and effective schedule for your next project?

Project managers have a variety of tools to develop a project schedule – from the relatively simple process of action planning for small projects, to use of Gantt Charts and Network Analysis for large projects. Here, we outline the key tools you will need for schedule development.

To plan and schedule project activities and tasks the project manager needs to take the next four steps:

- Set up activities.
- Define relationships between activities.

Notes

- Estimate resources required for performing activities.
- Estimate durations for activities.

Step #1: Set up Activities

The first step of project activities planning and scheduling requires the project manager to define what amount of actions and tasks are necessary for producing project deliverables in a timely manner. The input for this process will be the project deliverables statement. The project manager can use this document to define high-level activities that will be used later in creating the project implementation schedule. The project manager should also work on developing project activities templates that help simplify the process of project scheduling and planning.

In cooperation with experts and the project team, the manager should make project activities lists that will be the output of the process for project activities planning and scheduling. For each of the listed activities accurate milestones should be identified and approved. All the identified milestones should be gathered into a single milestones list.

Step #2: Define Relationships

The next step for planning project activities and tasks requires the project manager to make a sequence of all the activities identified at the previous step. The manager will use project activities lists, the milestones list and the product scope statement to define relationships among the activities. With help of project management software that person can set up priorities for each of the project activities and make task sequences organized and sorted by importance and urgency.

There is also a need to define dependencies between the activities. Dependencies can be internal and external. Activities with internal dependencies refer to any actions that the project team will take to produce the deliverables within the existing working environment. Activities with external dependencies refer to non-project factors that define success of project-related activities.

Both types of activity dependencies should be identified and added to sequenced and prioritized activity lists. Once the relationships are defined, the project manager should update project activities templates, outline the dependencies and link them to the product scope statement.

Step #3: Estimate Resources

At this step, the project manager needs to review stakeholder requirements and the product scope statement to estimate an amount of resources required for performing project activities and tasks. Also expert judgments and alternatives analysis should be used for this purpose.

The constraint of time needs to be considered when estimating activity resources. The project manager in cooperation with experts and the team should develop resource calendars and define types of required resources. Once all this information is collected and analyzed, it should be used to make a decomposition of activity resources categorized by types, priorities and time. This decomposition is critical to creating the project implementation schedule.

Step #4: Estimate Durations

The final step in project activity planning and scheduling requires the project manager to define and estimate an amount of working time required for accomplishing each identified activity. This is about setting up durations for project activities and tasks. Durations will depend on (1) the amount of work effort and (2) available of activity resources.

Notes

The project manager should review the resource decomposition and project activities templates to estimate the number of work periods required for completing the identified activities and producing the deliverables. The output of this process is activity estimates that are linked to resource calendars. This information will be used later in developing the implementation schedule.

Did u know? What is the resource calendar?

A resource calendar is a calendar that is used to reflect specific working hours, vacations, leaves of absence, and planned personal time for individual resources. A resource calendar differs from a base calendar in that it only reflects working and non-working times for individual resources.

7.2.1 Schedule Inputs

You need several types of inputs to create a project schedule:

- ***Personal and project calendars:*** Understanding working days, shifts, and resource availability is critical to completing a project schedule.
- ***Description of project scope:*** From this, you can determine key start and end dates, major assumptions behind the plan, and key constraints and restrictions. You can also include stakeholder expectations, which will often determine project milestones.
- ***Project risks:*** You need to understand these to make sure there's enough extra time to deal with identified risks – and with unidentified risks (risks are identified with thorough Risk Analysis).
- ***Lists of activities and resource requirements:*** Again, it's important to determine if there are other constraints to consider when developing the schedule. Understanding the resource capabilities and experience you have available – as well as company holidays and staff vacations – will affect the schedule.

A project manager should be aware of deadlines and resource availability issues that may make the schedule less flexible.

7.2.2 Scheduling Tools

Here are some tools and techniques for combining these inputs to develop the schedule:

- ***Schedule Network Analysis:*** This is a graphic representation of the project's activities, the time it takes to complete them, and the sequence in which they must be done. Project management software is typically used to create these analyses – Gantt charts and PERT Charts are common formats.
- ***Critical Path Analysis:*** This is the process of looking at all of the activities that must be completed, and calculating the 'best line' – or critical path – to take so that you'll complete the project in the minimum amount of time. The method calculates the earliest and latest possible start and finish times for project activities, and it estimates the dependencies among them to create a schedule of critical activities and dates. Learn more about Critical Path Analysis.
- ***Schedule Compression:*** This tool helps shorten the total duration of a project by decreasing the time allotted for certain activities. It's done so that you can meet time constraints, and still keep the original scope of the project. You can use two methods here:

Notes

- *Crashing:* This is where you assign more resources to an activity, thus decreasing the time it takes to complete it. This is based on the assumption that the time you save will offset the added resource costs.
- *Fast-Tracking:* This involves rearranging activities to allow more parallel work. This means that things you would normally do one after another are now done at the same time. However, do bear in mind that this approach increases the risk that you'll miss things, or fail to address changes.

Did u know? Use of Project Stages.

One of the biggest reasons that projects overrun is that the 'final' polishing and error-correction takes very much longer than anticipated. In this way, projects can seem to be '80% complete' for 80% of the time! What's worse, these projects can seem to be on schedule until, all of a sudden, they overrun radically.

A good way of avoiding this is to schedule projects in distinct stages, where final quality, finished components are delivered at the end of each stage. This way, quality problems can be identified early on, and rectified before they seriously threaten the project schedule.

7.2.3 Project Review

Once you have outlined the basic schedule, you need to review it to make sure that the timing for each activity is aligned with the necessary resources. Here are tools commonly used to do this:

- ***'What if' scenario analysis*** – This method compares and measures the effects of different scenarios on a project. You use simulations to determine the effects of various adverse, or harmful, assumptions – such as resources not being available on time, or delays in other areas of the project. You can then measure and plan for the risks posed in these scenarios.
- ***Resource leveling:*** Here, you rearrange the sequence of activities to address the possibility of unavailable resources, and to make sure that excessive demand is not put on resources at any point in time. If resources are available only in limited quantities, then you change the timing of activities so that the most critical activities have enough resources.
- ***Critical Chain Method:*** This also addresses resource availability. You plan activities using their latest possible start and finish dates. This adds extra time between activities, which you can then use to manage work disruptions.
- ***Risk Multipliers:*** Risk is inevitable, so you need to prepare for its impact. Adding extra time to high-risk activities is one strategy. Another is to add a time multiplier to certain tasks or certain resources to offset overly optimistic time estimation.

After the initial schedule has been reviewed, and adjustments made, it's a good idea to have other members of the team review it as well. Include people who will be doing the work – their insights and assumptions are likely to be particularly accurate and relevant.

Notes

Notes Scheduling aims to predict the future, and it has to consider many uncertainties and assumptions. As a result, many people believe it's more of an art than a science.

But whether you're planning a team retreat, or leading a multimillion-dollar IT project, the schedule is a critical part of your efforts. It identifies and organizes project tasks into a sequence of events that create the project management plan.

A variety of inputs and tools are used in the scheduling process, all of which are designed to help you understand your resources, your constraints, and your risks. The end result is a plan that links events in the best way to complete the project efficiently.

Self Assessment

Fill in the blanks:

6. No matter the size or …….. of your project, the schedule is a key part of project management.
7. ……., on the other hand, is not an exact process.
8. Project managers have a variety of tools to develop a ……...
9. The input for this process will be the project ……. statement.
10. The constraint of time needs to be considered when ……. activity resources.
11. Once all this information is collected and analyzed, it should be used to make a …… of activity resources categorized by types, priorities and time.
12. A good way of avoiding this is to schedule projects in distinct stages, where final quality, ……. are delivered at the end of each stage.
13. …….. is inevitable, so you need to prepare for its impact.
14. The project manager should ……. the resource decomposition and project activities templates.
15. Activities with …… dependencies refer to non-project factors that define success of project-related activities.

Caselet

Planning is the basis of all Corporate Activity'

Chennai , Jan. 4

"Planning is fundamental and forms the basis of any corporate activity. Sixty per cent accuracy with planning is better than a blindfolded approach. More importantly, one needs to plan for the planning to be successful and yield desired results," Mr. Hari Prakash, General Manager, Concorde Motors Pvt. Ltd., told post-graduate management students of Stansfield School of Business after inaugurating the Business Line Club at the college.

Stressing the importance of planning, Mr. Prakash said India has 25 crore families. Each one of them has needs and each one of them is different. This is a huge market for any marketer. A corporate can offer different products to these households for which planning

Contd.....

Notes

with respect to their tastes, preferences, income level, purchasing power, present necessities and future needs is vital.

A sales management job might be frustrating for a bright young management graduate, but a product management line is better and will be a challenging one, he said. Every organisation has product managers, though they exist in different names.

Elaborating on rural markets, Mr. Prakash said, "In rural markets, emotions are much stronger and therefore the loyalty to a product. The farmer who bought a brand wants his son also to buy the same brand. It will take more than 15 years to get a foothold in rural areas which means a marketer has to invest for 15 years to get a share of the rural markets."

In his closing remarks, Mr. Prakash said, "There are many functions in a corporate. An MBA should know all these functions as he is expected to be a multi-tasker."

Besides the post-graduate students of management, faculty members Mr. K. Sai Manohar, Mr. S.K. Shankar and Ms. Usha were also present on the occasion.

7.3 Summary

- Project time management includes two high-level groups of processes for planning and scheduling project activities and tasks necessary for timely completion of the project.
- Developing the project implementation schedule is the second group.
- Project managers have a variety of tools to develop a project schedule – from the relatively simple process of action planning for small projects, to use of Gantt Charts and Network Analysis for large projects.
- Here, we outline the key tools you will need for schedule development.

7.4 Keywords

Selecting Team: Selecting the right team members depends on a clear understanding of the project scope and success criteria.

Risk Multipliers: Risk is inevitable, so you need to prepare for its impact. Adding extra time to high-risk activities is one strategy.

Crashing**:** This is where you assign more resources to an activity, thus decreasing the time it takes to complete it.

7.5 Review Questions

1. Outline the essential steps involved in carrying out a activity planning for software project management.
2. Analyze how project activities planning and scheduling is the first process group of project time management.
3. Examine how team members define the work that needs to be done throughout the course of the project.
4. Examine the impact of project planning iterations on activity planning.
5. What do you think would be the main factors regarding project scheduling?
6. Analyze the work of area in which project scheduling helps you.

Notes

7. To plan and schedule project activities and tasks the project manager needs to take the certain steps. What are the steps they talking about?
8. Scrutinize several types of inputs to create a project schedule.
9. Discuss the tools and techniques used for combining the inputs to develop the project schedule.
10. Explain the tools used for the project review.

Answers: Self Assessment

1.	second	2.	stakeholder
3.	planning process	4.	team members
5.	sponsor	6.	scope
7.	Scheduling	8.	project schedule
9.	deliverables	10.	estimating
11.	decomposition	12.	finished components
13.	Risk	14.	Review
15.	external		

7.6 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://www.whoguidemhpcuk.org/downloads/primary_care/06-2_activity_planning.pdf

http://en.wikipedia.org/wiki/Schedule_(project_management)

Unit 8: Network Planning Model

Notes

CONTENTS

Objectives

Introduction

8.1 Network Planning Model

8.1.1 Benefits to Network-based Scheduling

8.2 Building the Network Diagram using the Precedence Diagramming Method

8.2.1 Dependencies

8.2.2 Constraints

8.2.3 Using the Lag Variable

8.2.4 Creating an Initial Project Network Schedule

8.3 Analyzing the Initial Project Network Diagram

8.3.1 Compressing the Schedule

8.4 Management Reserve

8.5 Time Dimension

8.6 Summary

8.7 Keywords

8.8 Review Questions

8.9 Further Readings

Objectives

After studying this unit, you will be able to:

- Determine task relationships
- Construct a network representation of the project activities
- Describe the types of activity dependencies
- Recognize the types of constraints that create activity sequences
- Compute the Earliest Start (ES), Earliest Finish (EF), Latest Start (LS), and Latest Finish (LF)
- Explain lag variables and their uses
- Identify the critical path in the project
- Define free slack and total slack and know their significance
- Analyze the network for possible schedule compression
- Use advanced network dependency relationships for improving the project schedule
- Understand and apply management reserve

Notes

Introduction

In this unit, we think about what needs to be done first and what can be done at the same time. We want to capture the logical relationships that exist between the tasks in our WBS. The traditional technique used to capture these relationships is the network diagram.

8.1 Network Planning Model

A Network Planning Model is a pictorial representation of the sequence in which the project work can be done. The whole idea here is look at your work visually and think about in what order (sequence) the work needs to occur. This is an exercise in logic. In many cases, this step is an excellent team activity. At this time, you don't want to concern yourself with resource constraints: just focus on logical sequence of the work. When you complete this task, you want to be clear on three things:

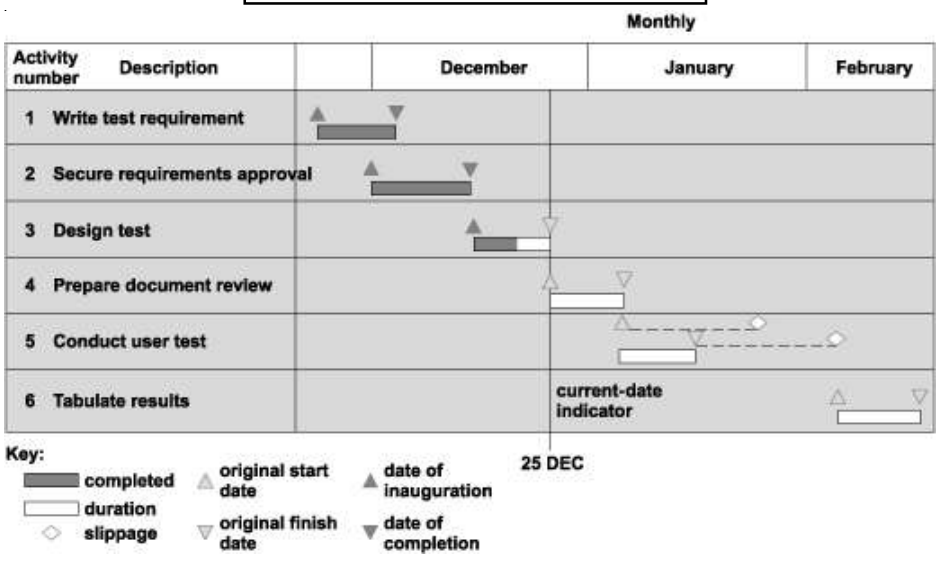
- For each task, what others tasks must be completed first?
- For the project, what tasks could be done at the same time (concurrently, in parallel)? For the project, where are your external dependencies? What tasks need an external event or task to complete, before it can start?

8.1.1 Benefits to Network-based Scheduling

There are two ways to build a project schedule:

- Gantt chart
- Network diagram

Figure 8.1: Example of a Gantt Bar Chart



Gantt Chart

The Gantt chart is the older of the two and is used effectively in simple, short-duration types of projects. To build a Gantt chart (see figure 8.1), the project manager begins by associating a rectangular bar with every activity. The length of the bar corresponds to the duration of the activity. He or she then places the bars horizontally along a time line in the order in which the activities should be completed. There can be instances in which activities are located on the time

line so that they are worked on concurrently with other activities. The sequencing is often driven more by resource availability than any other consideration.

There are two drawbacks to using the Gantt chart:

- Because of its simplicity, the Gantt chart does not contain detailed information. It reflects only the order imposed by the manager and, in fact, hides much of that information. Unless you are intimately familiar with the project activities, you cannot tell from the Gantt chart what must come before and after what.
- Second, the Gantt chart does not tell the project manager whether the schedule that results from the Gantt chart completes the project in the shortest possible time or even uses the resources most effectively.

Task The Gantt chart reflects only when the manager would like to have the work done. Explain

Network Diagram

The network diagram provides a visual layout of the sequence in which project work flows. It includes detailed information and serves as an analytical tool for project scheduling and resource management problems as they arise during the life of the project. In addition, the network diagram allows you to compute the earliest time at which the project can be completed. That information does not follow from a Gantt chart.

Network diagrams can be used for detailed project planning, during implementation as a tool for analyzing scheduling alternatives, and as a control tool:

Planning: Even for large projects, the Network Planning Model gives a clear graphical picture of the relationship between project activities. It is, at the same time, a high-level and detailed-level view of the project.

Implementation: For those project managers who use automated project management software tools, you will update the project file with activity status and estimate-to-completion data. The network diagram is then automatically updated and can be printed or viewed.

Control: While the updated network diagram retains the status of all activities, the best graphical report for monitoring and controlling project work will be the Gantt chart view of the network diagram. This Gantt chart cannot be used for control purposes unless you have done network scheduling or incorporated the logic into the Gantt chart.

Notes Comparing the planned schedule with the actual schedule, the project manager will discover variances and, depending on their severity, will be able to put a get-well plan in place.

Self Assessment

Fill in the blanks:

1. A ……. is a pictorial representation of the sequence in which the project work can be done.
2. The …… is the older of the two and is used effectively in simple, short-duration types of projects.

Notes

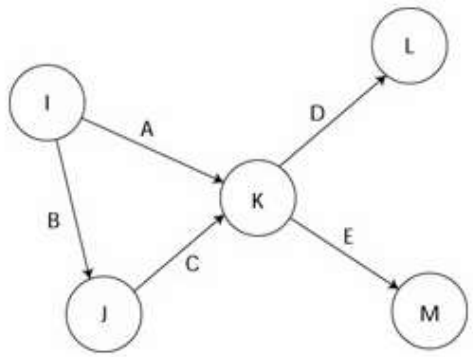
3. The …….. allows you to compute the earliest time at which the project can be completed.

4. Even for large projects, the Network Planning Model gives a clear …….. picture of the relationship between project activities.

8.2 Building the Network Diagram using the Precedence Diagramming Method

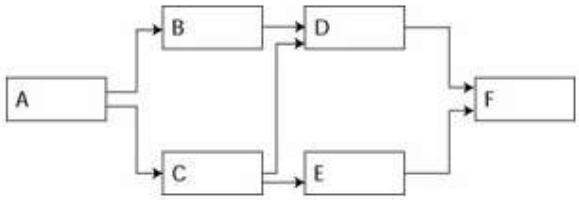
It is also called as the *Activity-on-the-Arrow (AOA) method* (See figure 8.2). An arrow represents each activity. The node at the left edge of the arrow is the "begin the activity," while the node at the right edge of the arrow is the "end the activity." Every activity is represented by this configuration. Nodes are numbered sequentially, and the sequential ordering had to be preserved, at least in the early versions. Because of the limitations of the AOA method, ghost activities had to be added to preserve network integrity.

Figure 8.2: The Activity-on-the-Arrow Method



Later on, the AOA method lost its appeal, and a new method known as the *Activity-on-the-Node (AON) method* replaced it. The term more commonly used to describe this approach is *Precedence Diagramming Method (PDM)* (see figure 8.3).

Figure 8.3: PDM Format of a Project Network Diagram

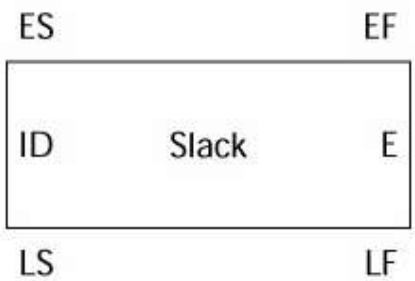


Each activity in the network diagram is represented by a rectangle that is called an *activity node* (see Figure 8.4). The entries in the activity node describe the time-related properties of the activity. Some of the entries describe characteristics of the activity, such as its expected duration (E), while others describe calculated values (ES, EF, LS, LF) associated with that activity. These terms will be defined later on in this unit.

In order to create the network diagram using the PDM, you need to determine the predecessors and successors for each activity. Here, you are looking for the technical dependencies between activities. Once an activity is complete, it will have produced an output, a deliverable, which becomes input to its successor activities. Work on the successor activities requires only the output from its predecessor activities.

Notes

Figure 8.4: Activity Node



Reading the Network Diagram

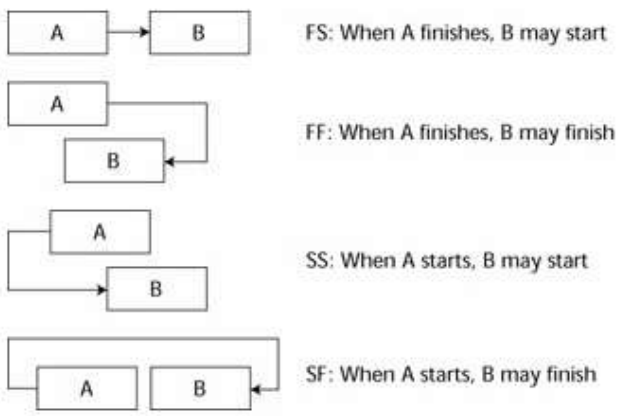
The network diagram is logically sequenced (see figure 8.3). It is read from left to right. Every activity in the network, except the 'start' must have at least one activity that comes before it (its immediate predecessor). Similarly, every activity in the network, except the 'end' must have at least one activity that comes after it (its immediate successor).

Caution An activity begins when its predecessors have been completed. The start activity has no predecessor, and the end activity has no successor.

8.2.1 Dependencies

A *dependency* is simply a relationship that exists between pairs of activities. To say that activity B depends on activity A means that activity A produces a deliverable that is needed in order to do the work associated with activity B. There are four types of activity dependencies, illustrated in Figure 8.5:

Figure 8.5: Dependency Relationships



Finish-to-start: The finish-to-start (FS) dependency is displayed with an arrow emanating from the right edge of the predecessor activity and leading to the left edge of the successor activity. It says that activity A must be completed before activity B can begin. For example, activity A can represent the collection of data, and activity B can represent entry of the data into the computer. It means that once we have finished collecting the data (Activity A), we may begin entering the data (Activity B).

Notes

Start-to-start: The start-to-start (SS) dependency is displayed with an arrow emanating from the left edge of the predecessor (A) and leading to the left edge of the successor (B). It says that activity B may begin once activity A has begun. It means both the activity A and B could start at the same time. For example, we could alter the data collection and data entry dependency: As soon as we begin collecting data (activity A), we may begin entering data (activity B).

Start-to-finish: The start-to-finish (SF) dependency is displayed with an arrow emanating from the left edge of activity A to the right edge of activity B. It is little more complex than the FS and SS dependencies. Here activity B cannot be finished sooner than activity A has started. For example, suppose you have built a new information system. You don't want to eliminate the legacy system until the new system is operable. When the new system starts to work (activity A) the old system can be discontinued (activity B). SF dependencies can be used for just-in-time scheduling between two tasks, but they rarely occur in practice.

Finish-to-finish: The finish-to-finish (FF) dependency is displayed with an arrow emanating from the right edge of activity A to the right edge of activity B. It states that activity B cannot finish sooner than activity A. For example, let's refer back to our data collection and entry example. Data entry (activity B) cannot finish until data collection (activity A) has finished.

Notes To preserve the connectedness property of the network diagram, the SS dependency on the front end of two activities should have an accompanying FF dependency on the back end.

8.2.2 Constraints

The type of dependency that describes the relationship between activities is determined as the result of *constraints* that exist between those activities. Each type of constraint can generate any one of the four dependency relationships. There are four types of constraints:

- Technical constraints
- Management constraints
- Interproject constraints
- Date constraints

Technical Constraints

Technical dependencies between activities arise when one activity (the successor) requires output from another (the predecessor) before work can begin on it. In the simplest case, the predecessor must be completed before the successor can begin.

Management Constraints

A second type of dependency arises as the result of a management-imposed constraint. For example, suppose the product manager on a software development project is aware that a competitor is soon to introduce a new product with similar features to theirs. Rather than following the concurrent design-build strategy, the product manager wants to ensure that the design of the new software will yield a product that can compete with the competitor's new product. He or she expects design changes in response to the competitor's new product and, rather than risk wasting the programmers' time, imposes the FS dependency between the design and build activities.

Notes

Interproject Constraints

Interproject constraints result when deliverables from one project are needed by another project. Such constraints result in dependencies between the activities that produce the deliverable in one project and the activities in the other project that require the use of those deliverables. For example, suppose the new piece of test equipment is being manufactured by the same company that is developing the software that will use the test equipment. In this case, the start of the testing activities in the software development project depends on the delivery of the manufactured test equipment from the other project. The dependencies that result are technical but exist between activities in two or more projects, rather than within a single project.

Date Constraints

Date constraints impose start or finish dates on an activity that force it to occur according to a particular schedule. In our date-driven world, it is tempting to use the requested date as the required delivery date. These constraints generally conflict with the schedule that is calculated and driven by the dependency relationships between activities.

Task What does FS dependency signifies?

8.2.3 Using the Lag Variable

Pauses or delays between activities are indicated in the network diagram through the use of *lag variables*. Lag variables are best defined by way of an example. Suppose that the data is being collected by mailing out a survey and is entered as the surveys are returned. Imposing an SS dependency between mailing out the surveys and entering the data would not be correct unless we introduced some delay between mailing surveys and getting back the responses that could be entered.

8.2.4 Creating an Initial Project Network Schedule

To establish the project schedule, you need to compute two schedules:

- *The early schedule*, which we calculate using the forward pass (This schedule consists of the earliest times at which an activity can start and finish. These are calculated numbers that are derived from the dependencies between all the activities in the project).
- The *late schedule*, which we calculate using the backward pass (The late schedule consists of the latest times at which an activity can start and finish without delaying the completion date of the project. These are also calculated numbers that are derived from the dependencies between all of the activities in the project).

The combination of these two schedules gives us some additional pieces of information about the project schedule:

- The window of time within which each activity must be started and finished in order for the project to complete on schedule.
- The sequence of activities that determine the project completion date.
- The sequence of activities that determine the project completion date is called the *critical path*. The critical path can be defined in several ways:
    - The longest duration path in the network diagram

Notes

- The sequence of activities whose early schedule and late schedule are the same
- The sequence of activities with zero slack or float.

The activities that define the critical path are called *critical path activities*. Any delay in a critical path activity will delay the completion of the project by the amount of delay in that activity. Critical path activities represent sequences of activities that warrant the project manager's special attention.

Earliest Start Time (ES) and Early Finish Time (EF)

The *Earliest Start (ES) time* for an activity is the earliest time at which all of its predecessor activities have been completed and the subject activity can begin. The ES time can be set as follows:

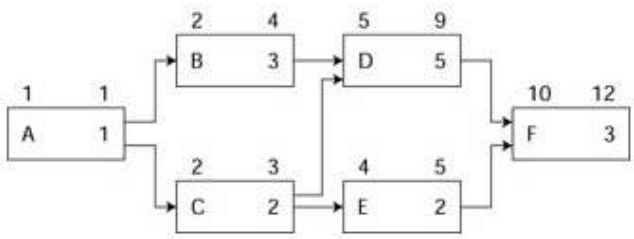
- The ES time of an activity with no predecessor activities is arbitrarily set to 1, the first day on which the project is open for work.
- The ES time of activities with one predecessor activity is determined from the *earliest finish (EF)* time of the predecessor activity.
- The ES time of activities having two or more predecessor activities is determined from the latest of the EF times of the predecessor activities.
- The ES can also be used to calculate the earliest finish time of an activity. The earliest finish (EF) of an activity is calculated as [(ES + Duration) – One time unit]. The reason for subtracting the one time unit is to account for the fact that an activity starts at the beginning of a time unit (hour, day, and so forth) and finishes at the end of a time unit. In other words, a one-day activity, starting at the beginning of a day, begins and ends on the same day.

Example: **Understand the Dependencies and Network Schedule**

Look at Figure 8.6 and note that:

- Activity E has only one predecessor, activity C. The EF for activity C is the end of day 3. Because it is the only predecessor of activity E, the ES of activity E is the beginning of day 4.
- Activity D has two predecessors, activity B and activity C. When there are two or more predecessors, the ES of the successor, activity D in this case, is calculated based on the maximum of the EF dates of the predecessor activities. The EF dates of the predecessors are the end of day 4 and the end of day 3. The maximum of these is 4, and therefore, the ES of activity D is the morning of day 5.
- Similarly, the EF and ES for other activities may also be calculated.

Figure 8.6: Forward Pass Calculations



Notes

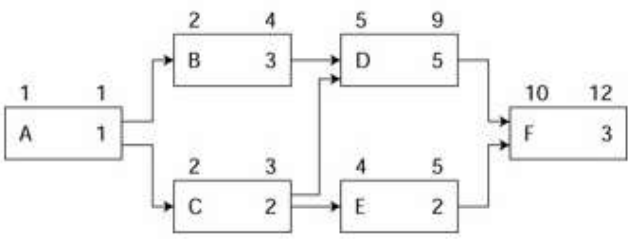
Latest Start Time (LS) and Latest Finish Time (LF)

The *latest start (LS)* and *latest finish (LF)* times of an activity are the latest times at which the activity can start or finish without causing a delay in the completion of the project. Knowing these times is valuable for the project manager, who must make decisions on resource scheduling that can affect completion dates. The steps are as follows:

- The window of time between the ES and LF of an activity is the window within which the resource for the work must be scheduled or the project completion date will be delayed.
- To calculate these times, you work backward in the network diagram. First set the LF time of the last activity on the network to its calculated EF time. Its LS is calculated as [(LF – Duration) + One time unit]. Again, you add the one time unit to adjust for the start and finish of an activity within the same day.
- The LF time of all immediate predecessor activities is determined by the minimum of the LS, minus one time unit, times of all activities for which it is the predecessor.

For example, let's calculate the late schedule for activity E in Figure 8.7. Its only successor, activity F, has an LS date of day 10. The LF date for its only predecessor, activity E, will therefore be the end of day 9. In other words, activity E must finish no later than the end of day 9 or it will delay the start of activity F and hence delay the completion date of the project. The LS date for activity E will be, using the formula, 9 – 2 + 1, or the beginning of day 7. On the other hand, consider activity C. It has two successor activities, activity D and activity E. The LS dates for them are day 5 and day 7, respectively. The minimum of those dates, day 5, is used to calculate the LF of activity C, namely, the end of day 4.

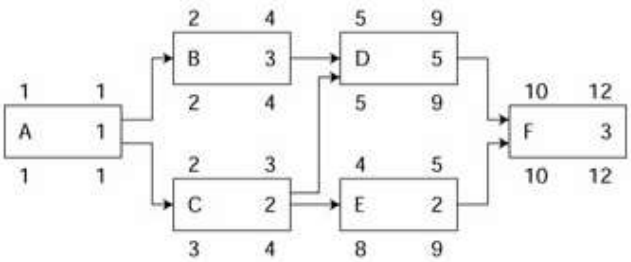
Figure 8.7: Backward Pass Calculations



Calculating Critical Path

As mentioned, the critical path is the longest path or sequence of activities (in terms of activity duration) through the network diagram. The critical path for the example problem we used to calculate the early schedule and the late schedule is shown in Figure 8.8.

Figure 8.8: Critical Path



Notes

One way to identify the critical path in the network diagram is to identify all possible paths through the network diagram and add up the durations of the activities that lie along those paths. The path with the longest duration time is the critical path. For projects of any size, this method is not feasible, and we have to resort to the second method of finding the critical path—computing the slack time of an activity.

Did u know? What are the uses of critical path method?

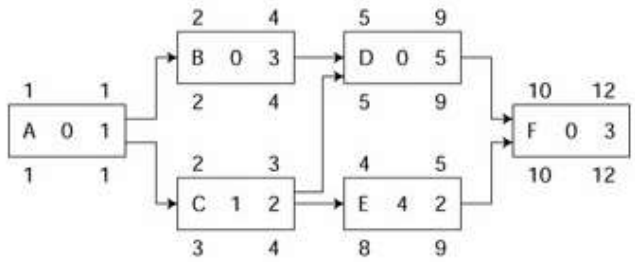
CPM can help you figure out:

- how long your complex project will take to complete
- which activities are "critical," meaning that they have to be done on time or else the whole project will take longer

Computing Slack

The second method of finding the critical path requires us to compute a quantity known as the activity slack time. *Slack time* (also called *float*) is the amount of delay expressed in units of time that could be tolerated in the starting time or completion time of an activity without causing a delay in the completion of the project. Slack time is a calculated number. It is the difference between the late finish and the early finish (LF – EF). If the result is greater than zero, the activity has a range of time in which it can start and finish without delaying the project completion date, as shown in Figure 8.9.

Figure 8.9: ES to LF Window of An Activity



Because weekends, holidays, and other non working periods are not conventionally considered part of the slack, these must be subtracted from the period of slack.

There are two types of slack:

1. ***Free Slack:*** This is the range of dates in which an activity can finish without causing a delay in the early schedule of any activities that are its immediate successors. Notice in (Figure 8.8) that activity C has an ES of the beginning of day 2 and a LF of the end of day 4. Its duration is two days, and it has a day 3 window within which it must be completed without affecting the ES of any of its successor activities (activity D and activity E). Therefore, it has free slack of one day. Free slack can be equal to but never greater than total slack. When you choose to delay the start of an activity, possibly for resource scheduling reasons, first consider activities that have free slack associated with them. By definition, if an activity's completion stays within the free slack range, it can never delay the early start date of any other activity in the project.

2. ***Total Slack:*** This is the range of dates in which an activity can finish without delaying the project completion date. Look at activity E in Figure 8.8. Activity E has a free slack

Notes

(or float) of four days, as well as a total slack (or float) of four days. In other words, if activity E were to be completed more than three days later than its EF date, it would delay completion of the project. We know that if an activity has zero slack, it determines the project completion date. If an activity with total slack greater than zero were to be delayed beyond its late finish date, it would become a critical path activity and cause the completion date to be delayed.

Notes Based on the method you used to compute the early and late schedules, the sequence of activities having zero slack is defined as the critical path. If an activity has been date-constrained using the on-this-date type of constraint, it will also have zero slack. However, this constraint usually gives a false indicator that an activity is on the critical path. Finally, in the general case, the critical path is the path that has minimum slack.

Self Assessment

Fill in the blanks:

5. ……. are numbered sequentially, and the sequential ordering had to be preserved, at least in the early versions.
6. Each activity in the network diagram is represented by a rectangle that is called an ……. node.
7. A …….. is simply a relationship that exists between pairs of activities.
8. The type of dependency that describes the relationship between activities is determined as the result of …….. that exist between those activities.
9. …… dependencies between activities arise when one activity (the successor) requires output from another (the predecessor) before work can begin on it.
10. …… (also called float) is the amount of delay expressed in units of time.

8.3 Analyzing the Initial Project Network Diagram

After you have created the initial project network diagram, one of two situations will be present:

- The initial project completion date meets the requested completion date. Usually this is not the case, but it does sometimes happen.
- The more likely situation is that the initial project completion date is later than the requested completion date. In other words, we have to find a way to squeeze some time out of the project schedule.

8.3.1 Compressing the Schedule

In situation where, the initial project completion date is later than the requested completion date, you must find ways to reduce the total duration of the project to meet the required date. What you will now have to do is adjust and readjust the critical path. This is known as *schedule compression*. By making adjustments to when tasks begin or by adding additional resources, you can complete the same work in less time.

Notes

There are four processes you can do to affect the flow of project schedule:

- ***Fast tracking:*** This method allows activities to be done in parallel that would normally be done in sequence.

  Example: You may allow two phases of the project to overlap slightly where normally you'd have quality control events, walkthroughs, or other events scheduled before the second phase of the project would be allowed to begin.

  This approach usually increases project risk.

- ***Crashing:*** Crashing allow the project manager to add more resources to effort-driven activities in an attempt to shorten their duration. For example, if you have to physically install 1,000 workstations and you've only eight people assigned to the task, it may take them months to complete. If you crash the project, you might assign 16 more people to this task to complete it in a matter of weeks. Crashing doesn't always work because some activities are a fixed duration and additional labor won't ensure the activities will finish faster. Crashing usually increases project costs because of the expense of the labor.

- ***Lead time:*** Lead time is negative time because it brings activities closer together–even allowing them to overlap. For example, you may have to install a new network cable throughout a campus. Your schedule calls for all of the network cables to run before any PCs plug into the new network. To speed up the schedule, you elect to allow the activity to connect the PCs to the new network as soon as half of the new cables are ready. The first activity, to run the network cables, does not have to be complete for the second activity, connecting to the new network, to begin.

- ***Lag time:*** Lag time is waiting time. It's often applied to activities where there must be an added duration between the tasks. For example, after installing a database, you have to wait four hours for all of the records from other databases in the network to recognize the database and synch with this database server. Lag time adds time to the project schedule.

To begin schedule compression, do the following:

- Analyze the critical path to move tasks earlier in the workflow—where possible.
- Consider relationships between tasks to change FS to SS.
- Identify tasks that require lag time and evaluate the predecessor task to move it earlier in the workflow.
- Consider any tasks and the level of acceptable risks by changing relationship types.
- Consider adding additional resources to tasks to shorten the duration required to complete tasks.

Caution Not all tasks can be shortened with additional resources.

8.4 Management Reserve

You and your project team will no doubt be tempted during the creation of each task to overstate the estimated amount of time for it to be completed. Don't yield to this temptation. Always reflect the accurate amount of time it should take to complete a task.

The reason is explained in Parkinson's Law. Parkinson's Law states that work will expand to the fill amount of time allotted to it. In other words, if your project team says an activity will take

Notes

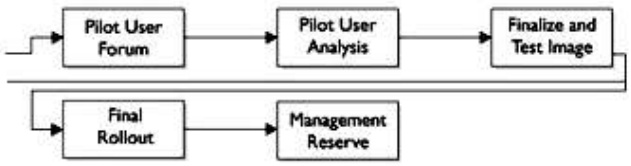
them 24 hours to complete, but they know the work will probably only take 16 hours to complete, it'll magically take 24 hours. Really, it's no magic. When people overestimate their time to account for expected troubles, just-in-case scenarios, and other time-munching issues, they rarely take advantage of the time they've created for themselves. They'll find other work to complete or simply wait until the time they've reserved for issues has passed and then hop into the work and hope for perfection.

Think of your own experiences. How many times have you had some small task to complete but spent hours cleaning your desk, organizing your materials, and researching the best mode of attack rather than just hopping in and completing the assignment? But how do you work on the day before your vacation? You are able to complete considerably more work on that particular day because the tasks must be completed before you're able to escape.

The same experience will be transferred to your team if you allow them two generous weeks for a task that should typically only require one. Your team will quickly discover that it will take every moment of the two weeks to complete the task you've assigned them.

Instead, what you should do is use management reserve. *Management reserve* is an artificial task that is added at the end of the project. The time allotted to the reserve is typically 10 to 15 per cent of the total amount of time to complete all the tasks in a project. When a task runs over its allotted time, the overrun is applied to the management reserve at the end of the critical path rather than on each lagging task. Figure 8.10 demonstrates the benefit of using management reserve.

Figure 8.10: Management Reserve Accounts for Task Overruns



Management reserve allows a project manager to use percentages to see how the overall project is coming along.

Example: If the project is only 40 per cent complete but the management reserve is 65 per cent used, then the project is in trouble if the remaining tasks follow the trend of the project thus far.

8.5 Time Dimension

As a manager, you have to regulate the pressure and work load which is imposed upon your team; you must protect them from the unreasonable demands of the rest of the company. Once you have arrived at what you consider to be a realistic schedule, fight for it. Never let the outside world deflect you from what you know to be practical. If they impose a deadline upon you which is impossible, clearly state this and give your reasons. You will need to give some room for compromise, however, since a flat NO will be seen as obstructive. Since you want to help the company, you should look for alternative positions.

You could offer a prototype service or product at an earlier date. This might, in some cases, be sufficient for the customer to start the next stage of his/her own project on the understanding that your project would be completed at a later date and the final version would then replace the prototype.

Notes

The complexity of the product, or the total number of units, might be reduced. This might, in some cases, be sufficient for the customer's immediate needs. Future enhancements or more units would then be the subject of a subsequent negotiation which, you feel, would be likely to succeed since you will have already demonstrate your ability to deliver on time.

You can show on an alternative schedule that the project could be delivered by the deadline if certain (specified) resources are given to you or if other projects are rescheduled. Thus, you provide a clear picture of the situation and a possible solution; it is up to your manager then how he/she proceeds.

Self Assessment

Fill in the blanks:

11. The ……. project completion date meets the requested completion date.
12. …… Law states that work will expand to the fill amount of time allotted to it.
13. Building the Network Diagram Using the Precedence Diagramming Method is also called as the …… method.
14. A …… is simply a relationship that exists between pairs of activities.
15. Pauses or delays between activities are indicated in the network diagram through the use of …… variables.

Caselet

Tejas Networks Plans to Invest ₹ 100 cr in R&D Next Year

Balaji Narasimhan

Bangalore, Dec. 24

Tejas Networks, a Bangalore-based telecom equipment products company, plans to invest ₹ 100 crore in R&D in India in 2011, said Mr. Sanjay Nayak, CEO and Managing Director.

"We invest around 15 per cent of our revenues every year in R&D and have spent around ₹ 500 crore in R&D over the last 10 years," he told Business Line in an interview.

This works out to an average of around ₹ 50 crore a year ; however, this year, the doubling of investment is primarily because of the growth of 3G networks in India. Tejas gets around 70 per cent of its revenues from India and is betting on the 3G revolution to boost its performance.

Explaining this, Mr. Nayak said, "With a 2G phone, your data transfers are in the range of Kbps, but with 3G, it will touch several Mbps. This means that a mobile base station that handles traffic of around 5 Mbps will have to handle a hundred times more data," he pointed out.

Tejas Networks creates solutions built on optical networking technology, which are used by mobile service providers to manage data transfers between the mobile phones and the base stations they communicate with. When telecom providers upgrade their infrastructure to support 3G, Tejas would be the equipment vendor in the picture, thus boosting its revenues, felt Mr. Nayak.

Contd.....

Notes

Giving an international example, he said that the demand on AT&T's network in the US went up by over 5,700 per cent over a period of three years immediately following the introduction of Apple's iPhone.

IPO delayed

While Mr. Nayak was keen on talking about the overall industry scenario, he was noncommittal about the IPO plans of his company. In 2009, the company had plans of an IPO in 2010, but these plans did not materialise.

Now, Mr. Nayak said, while it is still important, it is just one of the things that the company is interested in. "IPO is not the focus of the company right now," he said, but refused to state whether or not his company would have an IPO next year.

8.6 Summary

- A Network Planning Model is a pictorial representation of the sequence in which the project work can be done. There are two ways to build a project schedule: (a) Gantt chart and (b) Network diagram.
- The Gantt chart is the older of the two and is used effectively in simple, short-duration types of projects. The network diagram provides a visual layout of the sequence in which project work flows.
- It includes detailed information and serves as an analytical tool for project scheduling and resource management problems as they arise during the life of the project.
- Network diagrams can be used for detailed project planning, during implementation as a tool for analyzing scheduling alternatives, and as a control tool.
- You can build the Network Diagram using the Precedence Diagramming Method which is also known as the activity-on-the-arrow (AOA) method. Each activity in the network diagram is represented by a rectangle that is called an activity node.
- The entries in the activity node describe the time-related properties of the activity. Some of the entries describe characteristics of the activity, such as its expected duration (E), while others describe calculated values (ES, EF, LS, LF) associated with that activity.
- The network diagram is logically sequenced. It is read from left to right. Every activity in the network, except the 'start' must have at least one activity that comes before it (its immediate predecessor). Similarly, every activity in the network, except the 'end' must have at least one activity that comes after it (its immediate successor).
- There are four types of constraints:
  - Technical constraints
  - Management constraints
  - Interproject constraints and
  - Date constraints.
- Pauses or delays between activities are indicated in the network diagram through the use of lag variables. In situation where, the initial project completion date is later than the requested completion date, you must find ways to reduce the total duration of the project to meet the required date.

Notes

- What you will now have to do is adjust and readjust the critical path.
- This is known as schedule compression. Management reserve allows a project manager to use percentages to see how the overall project is coming along.

8.7 Keywords

A Project Network Diagram: A Network Planning Model is a pictorial representation of the sequence in which the project work can be done. The whole idea here is look at your work visually and think about in what order (sequence) the work needs to occur.

Gantt Chart: The Gantt chart is the older of the two and is used effectively in simple, short-duration types of projects.

Network Diagram: The network diagram provides a visual layout of the sequence in which project work flows.

Dependency: A dependency is simply a relationship that exists between pairs of activities. To say that activity B depends on activity A means that activity A produces a deliverable that is needed in order to do the work associated with activity B.

Critical Path: The sequence of activities that determine the project completion date is called the critical path.

Fast tracking: This method allows activities to be done in parallel that would normally be done in sequence.

Crashing: Crashing allow the project manager to add more resources to effort-driven activities in an attempt to shorten their duration.

Lead time: Lead time is negative time. It brings activities closer together—even allowing them to overlap.

Lag time: Lag time is waiting time.

Management Reserve: Management reserve allows a project manager to use percentages to see how the overall project is coming along. For example, if the project is only 40 per cent complete but the management reserve is 65 per cent used, then the project is in trouble if the remaining tasks follow the trend of the project thus far.

8.8 Review Questions

1. What is a Project Network Diagram?
2. Why do we create a Gantt chart and a network diagram?
3. What are the drawbacks in using the Gantt chart?
4. Explain how a network diagram can be used as a tool for analyzing scheduling alternatives.
5. How do you build a Network Diagram using the Precedence Diagramming Method?
6. What do you understand by dependencies? Explain the different types of activity dependencies.
7. What is the relation between a dependency and a constraint?
8. What is Lag Variable?
9. How do you create an initial project schedule?

Notes

10. What is a critical path? How do you calculate it?
11. What is the difference between a free slack and total slack?
12. Explain the following terms:
    (a) Fast tracking
    (b) Crashing
    (c) Lead time
    (d) Lag time
13. How do you compress a schedule?
14. What is the purpose of management reserve?

Answers: Self Assessment

1.	Project Network Diagram	2.	Gantt chart
3.	network diagram	4.	graphical
5.	Nodes	6.	Activity
7.	Dependency	8.	Constraints
9.	Technical	10.	Slack time
11.	initial	12.	Parkinson's
13.	AOA	14.	dependency
15.	lag		

8.9 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management*, Addison-Wesley, 1998.

Online links

http://ideas.repec.org/p/iim/iimawp/wp00253.html

http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=AD0673528

Notes

Unit 9: CPM /PERT

CONTENTS

Objectives

Introduction

9.1 Understanding PERT/CPM for Project Scheduling and Management

9.1.1 Framework for PERT and CPM

9.1.2 Drawing the CPM/PERT Network

9.2 CCPM vs. Traditional Project Management

9.3 Critical Chain Project Management Approach

9.4 Defining Buffers

9.4.1 Types of Buffers

9.4.2 Using Buffers

9.4.3 Managing Buffers

9.5 Summary

9.6 Keywords

9.7 Review Questions

9.8 Further Readings

Objectives

After studying this unit, you will be able to:

- Explain the difference between the critical path and the critical chain
- Identify resource constraints and know how to resolve them
- Use the critical chain approach to project management for single projects

Introduction

In this unit we will study about the Critical Chain Project Management (CCPM) which is method is planning and managing projects that puts more emphasis on the resources required to execute project tasks developed by Eliyahu M. Goldratt.

9.1 Understanding PERT/CPM for Project Scheduling and Management

Basically, CPM (Critical Path Method) and PERT (Programme Evaluation Review Technique) are project management techniques, which have been created out of the need of Western industrial and military establishments to plan, schedule and control complex projects.

There are many variations of CPM/PERT which have been useful in planning costs, scheduling manpower and machine time. CPM/PERT can answer the following important questions:

- How long will the entire project take to be completed? What are the risks involved?
- Which are the critical activities or tasks in the project which could delay the entire project if they were not completed on time?
- Is the project on schedule, behind schedule or ahead of schedule?
- If the project has to be finished earlier than planned, what is the best way to do this at the least cost?

9.1.1 Framework for PERT and CPM

Essentially, there are six steps which are common to both the techniques. The procedure is listed below:

1. Define the Project and all of it's significant activities or tasks. The Project (made up of several tasks) should have only a single start activity and a single finish activity.
2. Develop the relationships among the activities. Decide which activities must precede and which must follow others.
3. Draw the "Network" connecting all the activities. Each Activity should have unique event numbers. Dummy arrows are used where required to avoid giving the same numbering to two activities.
4. Assign time and/or cost estimates to each activity
5. Compute the longest time path through the network. This is called the critical path.
6. Use the Network to help plan, schedule, monitor and control the project.

The Key Concept used by CPM/PERT is that a small set of activities, which make up the longest path through the activity network control the entire project.

Notes If these "critical" activities could be identified and assigned to responsible persons, management resources could be optimally used by concentrating on the few activities which determine the fate of the entire project.

Non-critical activities can be replanned, rescheduled and resources for them can be reallocated flexibly, without affecting the whole project.

Five useful questions to ask when preparing an activity network are:

- Is this a Start Activity?
- Is this a Finish Activity?
- What Activity Precedes this?
- What Activity Follows this?
- What Activity is Concurrent with this?

Some activities are serially linked. The second activity can begin only after the first activity is completed. In certain cases, the activities are concurrent, because they are independent of each other and can start simultaneously. This is especially the case in organizations which have supervisory resources so that work can be delegated to various departments which will be responsible for the activities and their completion as planned.

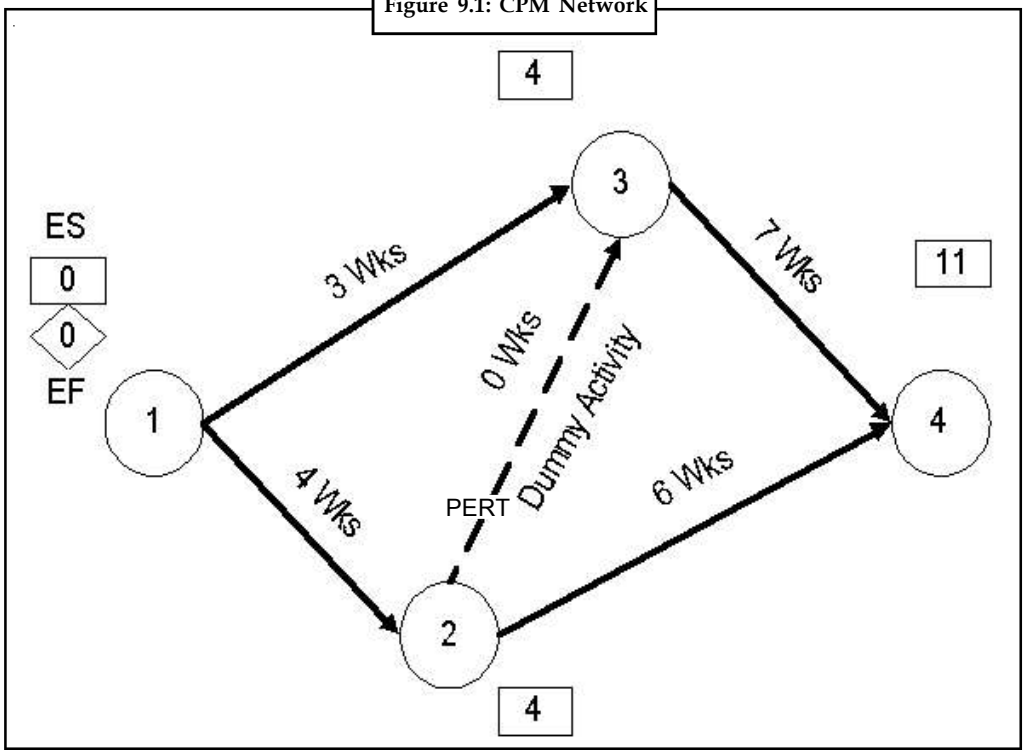
Notes

Caution When work is delegated like this, the need for constant feedback and co-ordination becomes an important senior management preoccupation.

9.1.2 Drawing the CPM/PERT Network

Each activity (or sub-project) in a PERT/CPM Network (see figure 9.1) is represented by an arrow symbol. Each activity is preceded and succeeded by an event, represented as a circle and numbered.

Figure 9.1: CPM Network



At Event 3, we have to evaluate two predecessor activities - Activity 1-3 and Activity 2-3, both of which are predecessor activities. Activity 1-3 gives us an Earliest Start of 3 weeks at Event 3. However, Activity 2-3 also has to be completed before Event 3 can begin. Along this route, the Earliest Start would be 4 + 0 = 4. The rule is to take the longer (bigger) of the two Earliest Starts. So the Earliest Start at event 3 is 4.

Similarly, at Event 4, we find we have to evaluate two predecessor activities - Activity 2-4 and Activity 3-4. Along Activity 2-4, the Earliest Start at Event 4 would be 10 wks, but along Activity 3-4, the Earliest Start at Event 4 would be 11 wks. Since 11 wks is larger than 10 wks, we select it as the Earliest Start at Event 4. We have now found the longest path through the network. It will take 11 weeks along activities 1-2, 2-3 and 3-4. This is the Critical Path.

Did u know? CPM was the discovery of M.R. Walker of E.I. Du Pont de Nemours & Co. and J.E. Kelly of Remington Rand, circa 1957. The computation was designed for the UNIVAC-I computer. The first test was made in 1958, when CPM was applied to the construction of a

Notes

new chemical plant. In March 1959, the method was applied to a maintenance shutdown at the Du Pont works in Louisville, Kentucky. Unproductive time was reduced from 125 to 93 hours.

Self Assessment

Fill in the blanks:

1. Compute the longest time path through the network. This is called the ……...
2. In certain cases, the activities are……., because they are independent of each other and can start simultaneously.
3. Non-critical activities can be replanned, rescheduled and resources for them can be …… flexibly, without affecting the whole project.
4. Use the Network to help plan, schedule, ……. and control the project.

9.2 CCPM vs Traditional Project Management

CCPM (Critical Chain Project Management) puts more emphasis on the resources required to execute project tasks in contrast to the more traditional Critical Path and PERT methods, which emphasize task order and rigid scheduling.

Secondly, using the Critical Chain Method, projects can be completed more quickly and with greater scheduling reliability. The difference between traditional and Critical Chain scheduling is in how uncertainty is managed. In traditional project scheduling, uncertainty is managed by padding task durations, starting work as early as possible, multi-tasking, and focusing on meeting commitment dates. The following points illustrate some of the problems associated with traditional project scheduling:

- Padding task durations (providing worst-case estimates) is done to ensure a high probability of task completion. The knowledge that there is so much safety time built into tasks results in various time wasting practices, e.g., waiting until the last moment to complete a task. As a result, all the safety time can be wasted at the start of the task so that, if problems are encountered, the task overruns.
- Starting work as early as possible, even when not scheduled, is a response to worst-case estimates. When workers give worst-case estimates, they don't expect to stay busy with just one task – so they multi-task, working on several tasks at once by switching between them. The result is that everything takes a long time to complete and very little completes early.
- With the focus on meeting commitment dates (start and finish), output from a task completed early will rarely be accepted early by the next person needing this output. So, any effort spent in finishing early will be wasted.

Given the above issues, the most projects are always late.

In critical chain scheduling, uncertainty is primarily managed by:

(a) using average task duration estimates

(b) scheduling backwards from the date a project is needed (to ensure work that needs to be done is done, and it is done only when needed)

(c) placing aggregate buffers in the project plan to protect the entire project and the key tasks and using buffer management to control the plan. The key tasks are those on which the ultimate duration of the project depends, also known as the Critical Chain.

Notes

Critical Chain approach is perhaps the most important new development in project scheduling in the last 30 years. Used properly, the Critical Chain approach is an extremely powerful means of gaining more predictability, productivity and speed from your project plans. It has been found to be an effective tool to protect projects from uncertainty.

Task Explain in a group of four in your words that by using the Critical Chain Method, how the projects can be completed more quickly?

Self Assessment

Fill in the blanks:

5. CCPM (Critical Chain Project Management) puts more emphasis on the resources required to …….. project.
6. …… durations (providing worst-case estimates) is done to ensure a high probability of task completion.
7. Critical Chain approach is perhaps the most important new development in ……. in the last 30 years.

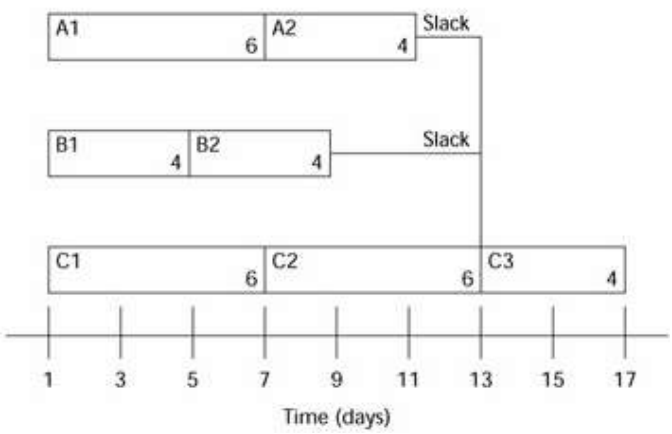
9.3 Critical Chain Project Management Approach

The CCPM approach is identical to the TPM approach up to the point where the project network diagram is defined and the critical path is identified. The traditional project manager would next conduct a resource leveling exercise targeting resource usage on the critical path. At this point, the critical chain project manager develops the critical chain plan. In the discussion that follows, we describe each of the CCPM planning steps for you by way of a simple example. We will use as an example the project shown in Figure 9.2.

Step 1: Creating the Early Schedule Project Network Diagram

Figure 9.2 shows the early schedule for a simple seven-task project same as TPM might make.

Figure 9.2: The Early Schedule



Notes

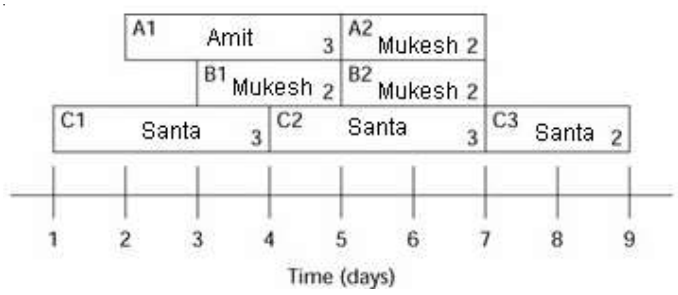
Notes Note that the project duration is estimated at 16 days using the original task duration estimates, which include contingency for each task.

Figure 9.2 shows the critical path (C1-C2-C3) for a project manager using TPM, and it is the starting point for resolving any resource contention problems.

Step 2: Converting the Early Schedule to the Late Schedule and Adding Resources

The first thing that a project manager using CCPM does is convert the task schedule to the late schedule. This is shown in Figure 9.3. Note that this conversion removes the slack associated with the sequence defined by tasks A1–A2 and B1–B2. In fact, it removes all of the free slack and total slack associated with any task or task sequence in the project. Note also that the 50 percent duration estimates have replaced the original estimates, which included contingency. In doing that, the project duration has been reduced from the original 16 days to 8 days. We have also added the three resources (Amit, Mukesh, and Santa) to the tasks to which they have been assigned. Note that there is a resource conflict with Mukesh on tasks A2 and B2. Also note that the project duration reduces to eight days when the contingencies are removed.

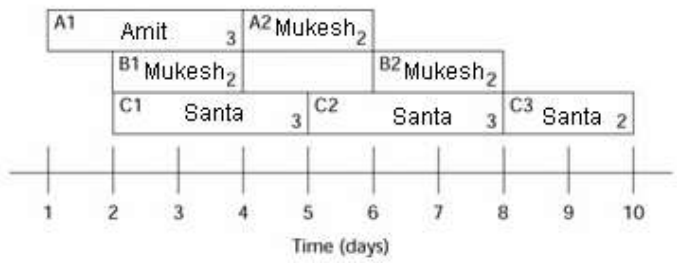
Figure 9.3: The Late Schedule with Resource Assignments



Step 3: Resolving Resource Conflicts

In general, resource conflicts are removed by beginning with the task sequence that has the least slack. After resolving that conflict, move to the task path that now has the least slack. Continue in this fashion until all resource conflicts have been resolved. In our example, the critical chain (C1-C2-C3) does not have any resource conflicts. The next task path to consider is A1-A2. In this case, Mukesh would be scheduled to work on A2, and that means pushing his work on B1 to an earlier date. This resolution is illustrated in Figure 9.4.

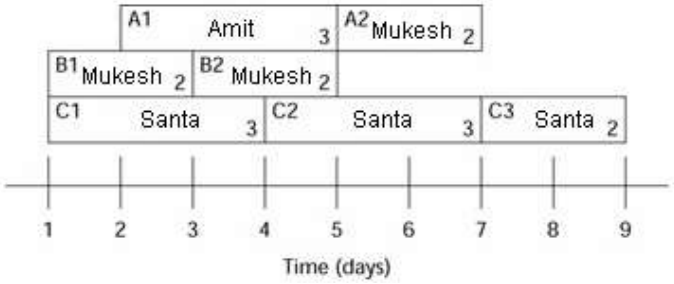
Figure 9.4: One way to Resolve the Resource Conflict



Notes

The other way to resolve the resource conflict is to have Ernie work on B2; then after Amit has completed A1, Mukesh can work on A2. This resolution is illustrated in Figure 9.5. The second choice extends the duration of the project.

Figure 9.5: Another way to Resolve the Resource Conflict



This simple example illustrates the major difference between TPM and CCPM. TPM uses the early schedule as the base for all management decisions. CCPM uses the late schedule. TPM focuses only on the critical path and manages in accordance with that. CCPM focuses on the paths with resource constraints and manages in accordance with the best use of the resources.

Notes It does so by using the critical path but only to identify the chains with the least slack and prioritizes resource use based on the minimum slack paths. To protect the scarce resources, CCPM uses the concept of buffers.

Self Assessment

Fill in the blanks:

8. The traditional project manager would next conduct a resource …… exercise targeting resource usage on the critical path.
9. The first thing that a project manager using CCPM does is …….. the task schedule to the late schedule.
10. In general, resource conflicts are removed by beginning with the task …… that has the least slack.

9.4 Defining Buffers

Buffers are segments of time that are placed at the end of a sequence of tasks for the purpose of protecting the schedule of those tasks. Buffers can also be used to protect cost, much like a contingency for unexpected expenses in a budget. The size of time buffers is based on the total duration of the sequence of tasks to which they are attached. The size of the buffer is determined by calculating the total of the contingencies in the tasks that make up the sequence.

9.4.1 Types of Buffers

The three main types of buffers are as follows:

- Project buffers
- Feeding buffers
- Resource buffers

Notes

Project Buffers

The *project buffer* is a time buffer placed at the end of the critical chain to protect the overall project schedule. Its size can be calculated as the square root of the sum of the squared differences between the original task duration estimate and the reduced task duration estimate.

Feeding Buffers

The *feeding buffer* is a time buffer placed at the end of a sequence of tasks that lead into the critical chain. Its size is calculated the same way as the project buffer size.

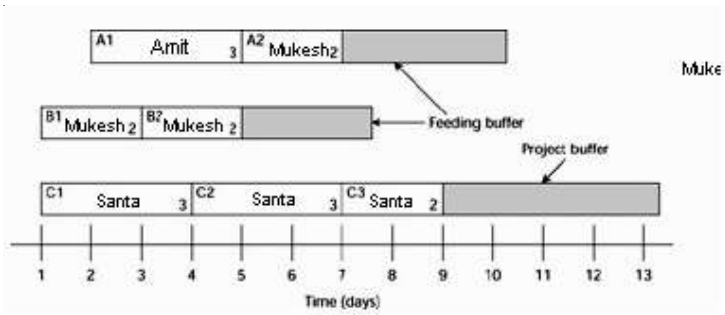
Resource Buffers

The *resource buffer* is different than the previous buffers. First, it is not a time buffer. It is a flag, usually placed on the critical chain to alert a resource that it is needed. The flag can be placed at intervals such as one week before the resource is needed, three days before the resource is needed, or one day before the resource is needed. Because it does not contain any time interval, it does not affect the project scheduled completion date. It serves merely to protect the critical chain.

9.4.2 Using Buffers

Let's return to the example project in Figure 9.4. So far, the project can be completed at the end of the eighth day. Resolving the resource conflict with Mukesh did not add any duration to the project. First, we put a project buffer after C3, the final task in this project. Its size is the square root of the square of the contingencies in tasks C1, C2, and C3 (or the square root of 9 + 9 + 4), which is approximately 4.7 days. Next, we add feeding buffers at the end of the sequences A1-A2 and B1-B2. The calculated buffer sizes are 3.6 days for A1-A2 and 2.8 days for B1-B2. Incorporating these buffers into Figure 9.4 results in Figure 9.6.

Figure 9.6: Project Schedule with Buffers Added



Note that the CCPM approach gives us a project duration (including contingency buffers) of 13.7 days, compared to the TPM approach of 16 days. In other words, extracting the contingency from each task, collecting it at the end of the task sequences, and managing it has saved us an average of 2.3 days out of a 16-day schedule. That's about a 14 percent schedule improvement.

9.4.3 Managing Buffers

The TPM will see the buffer as nothing more than management reserve. The only real similarity between a management reserve and a buffer may be in how they are managed. Managing management reserve and managing buffers can follow very similar logic. The three decision trigger levels in buffer management are as follows:

Notes

- When the sequence of tasks schedule slips and penetrates into the first third of the buffer.
- When the sequence of tasks schedule slips and penetrates into the middle third of the buffer.
- When the sequence of tasks schedule slips and penetrates into the final third of the buffer.

Let's take a closer look at each of these trigger points and the appropriate action that the CCPM project manager should take.

Penetration into the First Third of the Buffer

Unlike the project manager using TPM who chases every slippage on the critical path or change in the critical path, the project manager using CCPM is looking at the performance of a sequence and is less likely to act in haste. Penetration into the first third of the buffer means that the cumulative slippage in the sequence is less than one third of the buffer. In this case, the CCPM project manager will not take any action.

Penetration into the Middle Third of the Buffer

Penetration into the middle third of the buffer does call for some action on the part of the CCPM project manager. In this case, the correct action is to investigate the cause of the slippage and put a get-well plan in place. The earlier in the sequence this occurs, the more serious the problem.

Penetration into the Final Third of the Buffer

Penetration into the final third of the buffer is serious regardless of when in the sequence it occurs. Obviously, if it is in the first third of the duration of the sequence, it is very serious. If it occurs late in the final third, there may be little that can be done. In any case, action is called for.

Task Explain with the help of an example that how do you use buffers?

Self Assessment

Fill in the blanks:

11. …… is method is planning and managing projects that puts more emphasis on the resources required to execute project tasks.
12. Using …… Method, projects can be completed more quickly and with greater scheduling reliability.
13. Penetration into the …… of the buffer does call for some action on the part of the CCPM project manager.
14. The CCPM approach is identical to the …… approach up to the point where the project network diagram is defined and the critical path is identified.
15. …… are segments of time that are placed at the end of a sequence of tasks for the purpose of protecting the schedule of those tasks.

Notes

Caselet

PERT Moots steps to Rectify National Fiscal Situation

Chennai, July 1

A LONG-TERM fiscal outlook coupled with a national fiscal consolidation mission is the country's imperative need, according to a Chennai-based think-tank, Public Expenditure Round Table (PERT).

PERT has proposed a time-frame for Central and State governments to set right the fiscal situation. The time-frame is broken into two parts: from the present juncture to 2010 and from 2010 to 2020.

A recent PERT-Madras University seminar resulted in a report, "FISC2020: India's Long-term Fiscal Outlook." The report highlighted measures that need to be taken to rectify the country's fiscal situation.

The report set reference points for PERT's time-frame to improve the fiscal situation. For instance, by 2020, the tax-GDP ratio should be stepped up by 3 per cent to stand at 18 per cent and social sector spending by at least 2 per cent of GDP.

Among the key measures suggested are a halt and subsequent reversal in the trend of the falling tax-GDP ratio. PERT's report added that the solution is not restricted to calibrating tax rates, but that legal and administrative structures needed correction and consolidation.

Public-private partnerships would be needed on a larger scale, and off-budget resources and institutional credit would also have to be harnessed in a more significant way added the report.

PERT's report addressed the issue of transfer of resources between the Centre and the States. The report said that States needed to be empowered with greater taxation powers, as they had to bear the lion's share of expenditure commitments on education, health and urban development.

The PERT Chairman, Mr K. Venkataraman, said the think-tank would monitor actions taken by the Central Government and the States. He added that PERT would even form small expert groups to suggest concrete and specific administrative actions.

9.5 Summary

- CCPM puts more emphasis on the resources required to execute project tasks in contrast to the more traditional Critical Path and PERT methods, which emphasize task order and rigid scheduling.
- The difference between traditional and Critical Chain scheduling is in how uncertainty is managed.
- In traditional project scheduling, uncertainty is managed by padding task durations, starting work as early as possible, multi-tasking, and focusing on meeting commitment dates.
- The CCPM approach is identical to the TPM approach up to the point where the project network diagram is defined and the critical path is identified.
- The traditional project manager would next conduct a resource leveling exercise targeting resource usage on the critical path.

Notes

- At this point, the critical chain project manager develops the critical chain plan. Buffers are segments of time that are placed at the end of a sequence of tasks for the purpose of protecting the schedule of those tasks.
- Buffers can also be used to protect cost, much like a contingency for unexpected expenses in a budget.
- The three main types of buffers are Project buffers, Feeding buffers and Resource buffers. The TPM will see the buffer as nothing more than management reserve.
- The only real similarity between a management reserve and a buffer may be in how they are managed.

9.6 Keywords

Critical Chain Project Management (CCPM): It is a method of planning and managing projects that puts more emphasis on the resources required to execute project.

Buffers: *Buffers* are segments of time that are placed at the end of a sequence of tasks for the purpose of protecting the schedule of those tasks.

Project Buffers: The *project buffer* is a time buffer placed at the end of the critical chain to protect the overall project schedule.

Feeding Buffers: The *feeding buffer* is a time buffer placed at the end of a sequence of tasks that lead into the critical chain.

Resource Buffers: The *resource buffer* is different than the previous buffers. First, it is not a time buffer. It is a flag, usually placed on the critical chain to alert a resource that it is needed.

9.7 Review Questions

1. What is CCPM?
2. Make distinctions between CCPM and TPM.
3. What are the problems associated with traditional project scheduling?
4. How uncertainty is managed by Critical Chain approach?
5. How CCPM approach is identical to TPM?
6. Define buffers and explain their types.
7. Examine the impact of CPM/PERT on project scheduling.
8. Make an analysis and write a note on Project Buffers and Feeding Buffers. Examine their differences.
9. Analyze the distinctive points between CPM vs. PERT.
10. Explain the ways in which the buffers are managed.

Answers: Self Assessment

1.	critical path	2.	concurrent
3.	reallocated	4.	monitor
5.	execute	6.	Padding task

Notes

7.	project scheduling	8.	leveling
9.	convert	10.	sequence
11.	Critical Chain Project Management (CCPM)		
12.	Critical Chain	13.	middle third
14.	TPM	15.	Buffers

9.8 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://www.interventions.org/pertcpm.html

http://www.amiestudycircle.com/sample_5.pdf

Notes

Unit 10: Risk Management

CONTENTS

Objectives

Introduction

10.1 Risk Management

10.1.1 What is Risk?

10.1.2 Risk Analysis

10.1.3 Purpose of Risk Management

10.2 Software Risk Management Steps and Techniques

10.2.1 Risk Identification

10.2.2 Risk Estimation

10.2.3 Risk Exposure

10.2.4 Risk Migration

10.3 Risk Management Plans

10.4 Why is Risk Management Important?

10.5 Risk Management Cycle

10.5.1 Risk Identification

10.5.2 Risk Quantification

10.5.3 Risk Monitoring

10.5.4 Risk Mitigation

10.6 Practical Techniques in Risk Management

10.6.1 Risks in Software Project Management

10.7 Metrics in Risk Management

10.7.1 Identifying Risks with Metrics

10.7.2 Tracking Risks with Metrics

10.8 Categories of Risk

10.9 Assessment

10.9.1 Schedule Risk Assessment Plan

10.10 PERT Technique

10.11 Summary

10.12 Keywords

10.13 Review Questions

10.14 Further Readings

Notes

Objectives

After studying this unit, you will be able to:

- Explain the Risk Management
- Describe categories of risk
- Explain Schedule Risk Assessment Plan
- Discuss applying PERT technique

Introduction

In risk management, a prioritization process is followed whereby the risks with the greatest loss and the greatest probability of occurring are handled first, and risks with lower probability of occurrence and lower loss are handled in descending order. In practice the process can be very difficult, and balancing between risks with a high probability of occurrence but lower loss versus a risk with high loss but lower probability of occurrence can often be mishandled.

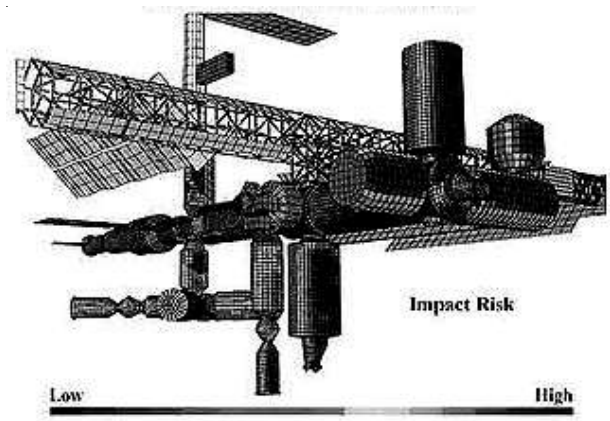
Intangible risk management identifies a new type of a risk that has a 100% probability of occurring but is ignored by the organization due to a lack of identification ability. For example, when deficient knowledge is applied to a situation, a knowledge risk materialises. Relationship risk appears when ineffective collaboration occurs. Process-engagement risk may be an issue when ineffective operational procedures are applied. These risks directly reduce the productivity of knowledge workers, decrease cost effectiveness, profitability, service, quality, reputation, brand value, and earnings quality. Intangible risk management allows risk management to create immediate value from the identification and reduction of risks that reduce productivity.

Risk management also faces difficulties allocating resources. This is the idea of opportunity cost. Resources spent on risk management could have been spent on more profitable activities. Again, ideal risk management minimizes spending while maximizing the reduction of the negative effects of risks.

10.1 Risk Management

Risk management is the discipline of identifying, monitoring and limiting risks. In some cases the acceptable risk may be near zero. Risks can come from accidents, natural causes and disasters as well as deliberate attacks from an adversary.

Figure 10.1: International Space Station



Notes

In businesses, risk management entails organized activity to manage uncertainty and threats and involves people following procedures and using tools in order to ensure conformance with risk-management policies.

Risk management is also used in the public sector to identify and mitigate risk to critical infrastructure. For the most part, these methodologies consist of the following elements, performed, more or less, in the following order:

1. identify assets and identify which are most critical
2. identify, characterize, and assess threats
3. assess the vulnerability of critical assets to specific threats
4. determine the risk (i.e. the expected consequences of specific types of attacks on specific assets)
5. identify ways to reduce those risks
6. prioritize risk reduction measures based on a strategy.

The strategies include transferring the risk to another party, avoiding the risk, reducing the negative effect of the risk, and accepting some or all of the consequences of a particular risk.

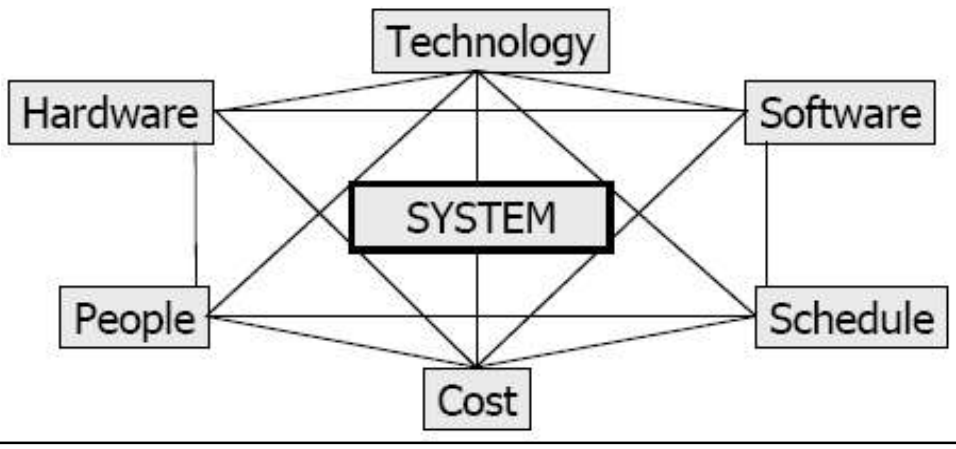
Example: Example of risk management:

High impact risk areas for the International Space Station.

10.1.1 What is Risk?

Risk is defined as "The possibility of suffering harm or loss; danger." Even if we're not recognizable with the formal definition, most of us have an innate sense of risk. We are aware of the potential dangers that permeate even simple daily activities, from getting injured when crossing the street to having a heart attack because our cholesterol level is too high. Although we prefer not to dwell on the myriad of hazards that surround us, these risks shape many of our behaviors. Experience (or a parent) has taught us to look both ways before stepping off the curb and most of us at least think twice before ordering a steak. Indeed, we manage personal risks every day.

Figure 10.2: Source of Software Risk (system context)



10.1.2 Risk Analysis

Risk analysis is essentially a "what if" analysis where various scenarios are visualized. It's a systematic use of known information and data to determine how and when incidents can or may

Notes

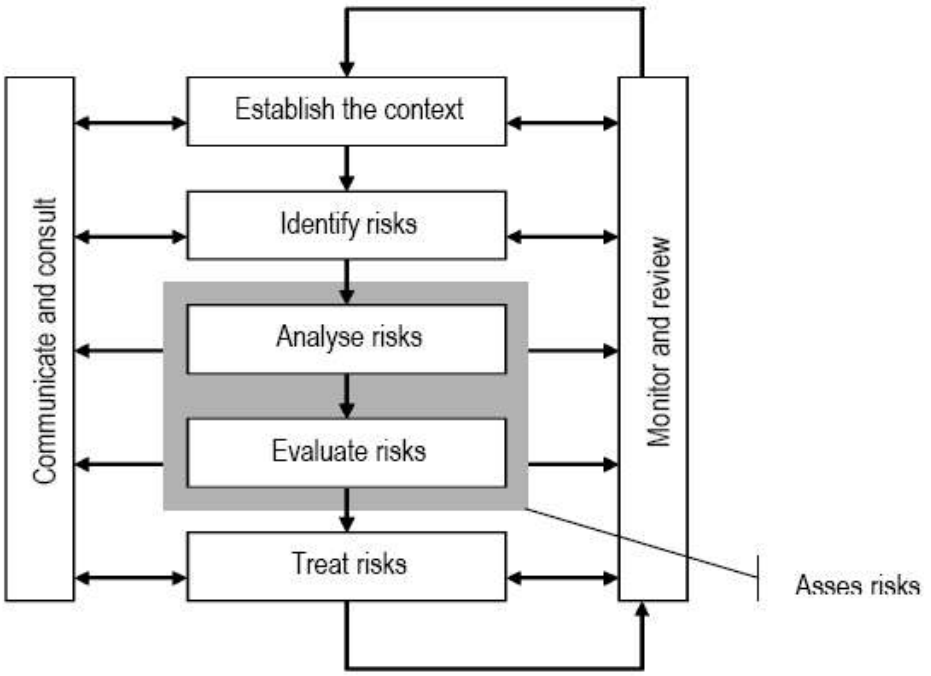
occur and the size of their consequences. The management of these risks are a very important part of the management process. This process includes wide aspects of managing and is easiest solved in teams, divided by their skill and knowledge in particular areas.

Notes The process of risk analysis and management is a process of continual improvement, which means that there is never only one solution of a problem, there are always improvements that can be done to upgrade (improve) the quality of the treatment of the risks.

10.1.3 Purpose of Risk Management

The risk management process includes seven steps that has to be followed Figure 10.3 shows what the concept continues improvements mean. When the risk has been treated it is not put a side, but instead the risk management process starts all over again to come up with better solutions. The seven steps can be explained as the following:

Figure 10.3



1. Establish the context sets the boundaries for which within the risk are managed and helps to set guidelines for how to get started with the process. The context includes five sublevels:
   - The strategic context
   - The organizational context
   - The risk management context
   - Develop criteria
   - Decide the structure
2. Identifying risks is the same as identifying how, what or why incidents may occur.

Notes

3. The risk analysis includes the magnitude of consequences and the chance that these consequences may come to life.
4. In the evaluation of risks the consequences are leveled (ranked) after their magnitude, so if needed the right treatment will be applied.
5. If the risk is low or has a low-priority it can be taken care of (by routine knowledge) or be accepted at this stage. But if it is a high-priority risk, a plan for managing is instantly laid out. According to AS/NZS 4360 the following plan should be used:
   - Identify treatment options
   - Evaluate treatment options
   - Select treatment options
   - Prepare treatment plan
   - Implement plans
6. Monitoring and reviewing are used to overlook the risk management cycle and track changes within it so new contexts (continues improvements) can be made and the final treatment of the risks improves.
7. Communication and consultation at every new step in the process is very important to people on the inside of the company as for people on the outside (stakeholders, investors).

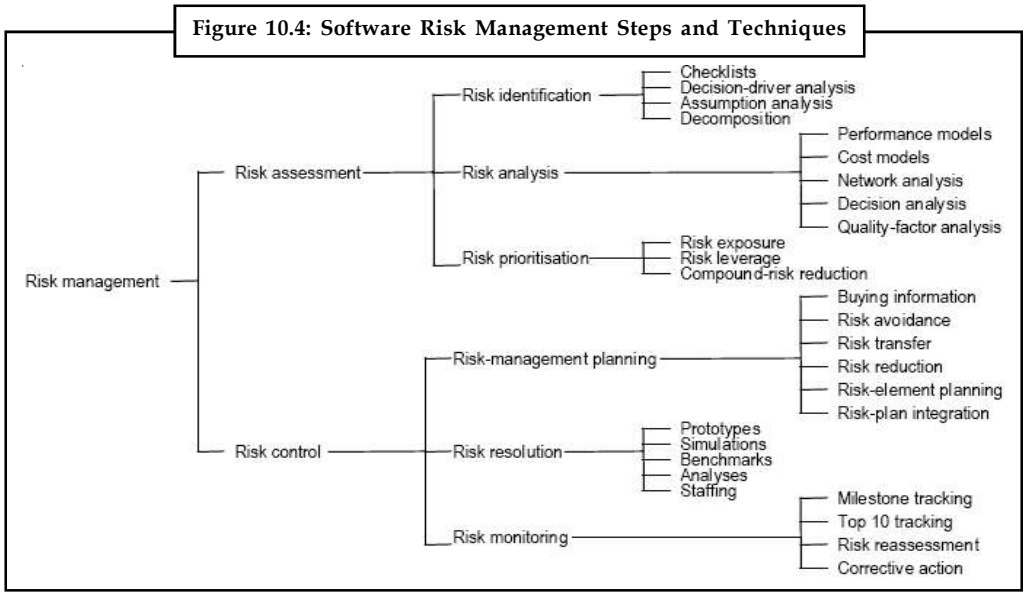
Did u know? What is AS/NZS 4360?

AS/NZS 4360 is a generic guide for risk management so that it applies to all forms of organizations. Risk management" is defined as 'the culture, processes and structures that are directed towards realizing potential opportunities whilst managing adverse effects.'

The model of the risk management process AS/NZS 4360 consists of three major elements. The risk management workflow, monitor and review, and finally communication and consult.

10.2 Software Risk Management Steps and Techniques

Figure 10.4: Software Risk Management Steps and Techniques



Notes

10.2.1 Risk Identification

Risk identification is used in risk management to answer the questions: What can happen? How can it happen? Risk identification is the process of recognizing the opportunities opened up by each activity or phase of the project and clarifying where the risk lies. The agreed tolerance of risk should help identify the amount of time should be spent in identifying risk, but, at least the 20% of the risks that would have 80% of the potential impact should be identified. There are many techniques to aid in risk identification and they generally fall under the heading of either quantitative or qualitative risk identification techniques.

Following are the main *Qualitative Techniques of Risk Identification*

- Assumptions Analysis
- Check Lists
- Prompt Lists
- Brainstorming
- Facilitated workshops
- Interviews

Assumptions Analysis

It is evitable that when you start planning or outlining your project you will be making assumptions. Making assumptions within a project will always create risks and a way to help manage this is to list all the assumptions of each phase or stage of your project against a timeline. Then think about the consequence of the assumptions and how they affect the other parts of the project. Try creating a high level storyboard of the project showing risk and assumptions are associated, this can show the effect of decisions and should provide a better understanding about the risks in a program or project.

Check Lists

Risk checklists are often built upon a Project or Program Managers past experience or the Project Management experience of an organization. They will normally take into account:

- Quality experience
- Formality of development
- Novelty of application
- Impact on business
- Requirements standards
- Software identification
- Projects concurrency
- Dependencies
- Project duration
- Flexibility of delivery
- Planning estimates
- Stability of suppliers
- Range of sites
- Impact upon status quo

Notes

Prompt Lists

Risks here will be identified by logical examination of each Program or Project aspect. Following is a list of common areas from which risk can arise:

External	
Infrastructure	Relating to infrastructures such as computer networks, transport systems for staff, power supply systems
Economic	Relating to economic factors such as interest rates, exchange rates, inflation
Legal & Regulatory	Relating to the laws and regulations which if complied with should reduce hazards (E.g. - Health and Safety at Work Act)
Environmental	Relating to issues such as fuel consumption, pollution
Political	Relating to possible political constraints such as a change of government
Market	Relating to issues such as competition and supply of goods
Act of God	Relating to issues such as fire, flood earthquake
Financial	
Budgetary	Relating to the availability of resources or the allocation of resources
Fraud or Theft	Relating to the unproductive loss of resources
Insurable	Relating to the potential areas of loss which can be insured against
Capital Investment	Relating to the making of appropriate investment decisions
Liability	Relating to the right to sue or be sued in certain circumstances
Activity	
Policy	Relating to the appropriateness and quality of policy decisions
Operational	Relating to the procedures employed to achieve particular objectives
Information	Relating to the adequacy of information which is used for decision making
Reputational	Relating to the public reputation of the organisation and consequent effects
Transferable	Relating to risks which can be transferred or the transfer of risks at inappropriate cost
Technological	Relating to the use of technology to achieve objectives
Project	Relating to project planning and management procedures
Innovation	Relating to the exploitation of opportunities to make gains
Human Resources	
Personnel	Relating to the availability and retention of staff
Health and Safety	Relating to the well-being of people

Brainstorming or SWOT Analysis

Within SWOT analysis risks are identified by looking at the Strengths (or perceived Strength), Weakness, Opportunity & Threats to the success of the project or program. This is usually done within a Workshop environment.

Facilitated Workshops

Within SWOT analysis risks are identified by looking at the Strengths (or perceived Strength), Weakness, Opportunity & Threats to the success of the project or program. This is usually done within a Workshop environment.

Interviews

Interviews conducted to identify risk will only be successful where there is:

- Good preparation;
- Clear objectives;

Notes

- A positive & supportive environment;
- Proper time management;
- Use of open questions; and
- Challenging not confrontational atmosphere.

The results of interviews should be well documented.

Quantitative Risk Management Techniques

There are three main quantitative techniques:

- Decision trees
- Influence diagrams
- Monte Carlo Simulation

Decision Trees

Rather like flowchart diagrams these represent a method of looking at, for example, two options and making a decision. By analyzing the impact each decision will have, the risks of taking that decision can be forecast and used to anticipate problems or inform the direction the project takes. This technique is best suited to simpler situations. In complex scenarios they can become confusing and complicated.

Influence Diagrams

This technique results in a diagram, which is similar to a project network diagram or Microsoft Project PERT charts. In this case each box will contain a variable or decision, which will have an influence on future progress. By analyzing the impact each variable will have, the risks of taking one path over another can be forecast and used anticipate problems or inform the direction the project takes.

Monte Carlo Simulation (or 3-Point Estimation)

Looking at both best and worse case scenarios as well as most likely scenario and then planning what the impact of each is. This can be plotted against the Project Baseline and the Critical Path to show the consequence of risk and allow you to anticipate suitable response to risk.

10.2.2 Risk Estimation

Risk, at the general level, involves two major elements: the occurrence probability of an adverse event and the consequences of the event. Risk estimation, consequently, is an estimation process, starting from the occurrence probability and ending at the consequence values.

Risk estimation involves following activities:

- Discussion of source, exposure issues
- Communication of results with stakeholders
- Assess changes in knowledge/perception in light of new information

During the Risk Estimation step of risk management, the frequency and consequences associated with each risk scenario are estimated and communicated with stakeholders. Stakeholders may have important knowledge of sources and patterns of exposure that analysts will need to integrate into a risk assessment.

Notes

Caution However conflict is most likely to arise at this step as stakeholders are not typically involved in the risk estimation process, and the uncertainties and value assumptions associated with the methods may not be clearly communicated.

During the Risk Estimation stage, stakeholder's knowledge and perceptions are assessed in light of receiving new information resulting from the risk estimates and the stakeholder analysis is updated. Third party review by third party experts and explicit communication of the methods, assumptions and uncertainties will contribute to credibility and trust in the technical analyses.

Software Estimation Risks

The effects of inaccurate software estimation and schedule overruns are well known. The problem stems from an inability to accurately assess risks associated with various software development projects. Software estimation errors generally result from four major risk areas, which are:

1. The inability to accurately size the software project. This results in poor implementations, emergency staffing, and cost overruns caused by underestimating project needs.
2. The inability to accurately specify a development environment which reflects reality. This results in defining cost drivers which may be inappropriate, underestimated or overestimated.
3. The improper assessment of staff skills. This results in misalignment of skills to tasks and ultimately miscalculations of schedules and level of effort required, as well as either underestimating or overestimating project staffing requirements.
4. The lack of well defined objectives, requirements, and specifications, or unconstrained requirements growth during the software development life cycle. This results in forever changing project goals, frustration, customer dissatisfaction, and ultimately, cost overruns.

All potential risks associated with the proposed software development project should be defined and weighed, and impacts to project cost should be determined. This information should always be included in the software estimation process.

10.2.3 Risk Exposure

Quantifying the effects of a risk by multiplying the risk impact by the risk probability yields risk exposure.

Risk-exposure = Risk-impact x Risk-probability

For each risk, the *Risk Exposure* is defined as the probability of the undesirable outcome times the size of the loss involved. Risk exposure helps us to list the risks in priority order, with the risks of most concern given the highest priority. Next, we must take steps to control the risks. The notion of control acknowledges that we may not be able to eliminate all risks.

Identifying Risk Exposure

It is the responsibility of project manager to ensure that team has an understanding of the project's exposure to risk. Gaining this understanding can be achieved through identifying categories of risk and then answering questions associated with each of these categories. The next section provides project managers with a set of questions to ask about their projects to help categorize risk.

Notes

Task The notion of control acknowledges that we may not be able to eliminate all risks. Analyse

Risk Categories and Questions

- Business/Strategic
- External Factors
- Procurement
- Organizational Factors
- Management
- Technical

Strategic

1. Do the project objectives fit into the organizations overall business strategy?
2. When is the project due to deliver?
3. What would be the result of late delivery?
4. What would be the result of limited success (functionality)?
5. What is the stability of the business area?

External Factors

1. Is the project exposed to requirements due to international interests (foreign legal implications or foreign company involvement)?
2. Could there be political implications of the project failure?
3. Is this project a part of the larger program? IF so, what constraints are set for the project by the program?

Procurement

1. Does the supplier have a reputation for delivery of high quality?
2. Is the contract sufficiently detailed to show what the supplier is going to provide?
3. Are the acceptance criteria clear to both the parties?
4. Is the contract legally binding/enforceable?

Organizational Factors

1. What consideration needs to be given to security of the project?
2. Does the project have wholehearted support from senior management?
3. What is the commitment of the user management?
4. Have training requirements been identified? Can these requirements be met?

Notes

Management

1. How clearly are the project objectives defined?
2. Will the project be run using well-documented approach to project management?
3. Does this approach cover aspects of quality management, risk management and development activities in sufficient depth?
4. How well does the project team understand chosen methodology?
5. What is the current state of project plans?
6. Is the completion of project dependent on completion of other projects?
7. Are the tasks in project plan interdependent?
8. Can a critical path through the project tasks be identified?
9. What is the availability of appropriate resources?
10. What are the skills and experience of project team?
11. Will people be available for training?
12. How many separate users are involved?
13. How much changes will there be for user's operation or organization?

Technical

1. Is the specification clear, concise accurate and feasible?
2. How have the technical options been evaluated?
3. What is the knowledge of equipment (hardware/software)?
4. Does the experience of project manager cover a similar application?
5. Is this a new application?
6. What is the complexity of system?
7. How many sites will the system are implemented in?
8. Is the proposed equipment new/leading edge?
9. Who is responsible for defining system testing?
10. Who is responsible for defining acceptance testing?
11. On what basis is the implementation planned?
12. What access will the project team have to development/testing facilities?
13. Will users of data processing staff use the system?
14. Have requirements for long-term operations, maintenance and support been identified?

With risk identified, it is much easier to develop a plan to eliminate or manage risk.

10.2.4 Risk Migration

Risk Migration is the process of reducing risk exposure, either by decreasing the probability of the risk occurring, or by finding ways to reduce the possible impact if it does occur.

Notes

10.3 Risk Management Plans

The Risk Management Plan (RMP) presents the process for implementing proactive risk management as part of overall project management. The RMP describes techniques for identifying, analyzing, prioritizing & tracking risks; developing risk-handling methods; & planning for adequate resources to handle each risk, should they occur. The RMP also assigns specific risk management responsibilities & describes the documenting, monitoring & reporting processes to be followed.

Risk Management Plan defines how risks will be managed during the life cycle of the program. It is used to plan the way risks are handled within the program. The Risk Strategy and supporting Plan must acknowledge actual and potential threats to the successful delivery of a project and determines the activities required to minimize or eliminate them. The risk plan needs to be capable of integration into or coordination with the project plan.

Self Assessment

Fill in the blanks:

1. …….. can come from accidents, natural causes and disasters as well as deliberate attacks from an adversary.
2. Risk analysis is essentially a "what if" analysis where various scenarios are …….
3. The lack of well defined objectives, ……., and specifications, or unconstrained requirements growth during the software development life cycle.
4. The Risk Management Plan (RMP) presents the process for implementing proactive risk management as part of overall ……...
5. The ……. and supporting Plan must acknowledge actual and potential threats to the successful delivery of a project and determines the activities required to minimize or eliminate them.
6. The problem stems from an inability to ……. assess risks associated with various software development projects.

10.4 Why is Risk Management Important?

Risk exists in everything that we do. It is only through the effective management of risk that we can be assured that we are doing our reasonable best to manage ourselves so as to protect patients, staff, the public and other stakeholders against risks of all kinds. It can also help us to meet our objectives and improve performance by contributing to:

- Increased certainty and fewer surprises
- Better service delivery
- More effective management of change
- More efficient use of resources
- Better management at all levels through improved decision making
- Innovation
- Improved working environment
- Patient/Staff safety

Notes

Task In a group of three write in your words that why risk management is important?

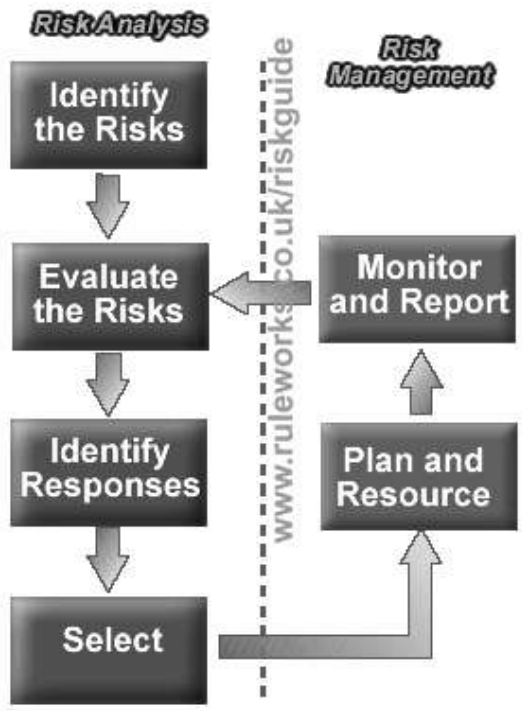
10.5 Risk Management Cycle

Every project is subject to constant change in its business and wider environment. The risk environment is constantly changing too. The project's priorities and relative importance of risks will shift and change. Assumptions about risk have to be regularly revisited and reconsidered, for example at each end stage assessment.

The Figure 10.5 shows the main steps through the risk management cycle:

- Identify the risks
- Evaluate the risks
- Identify suitable responses to risk
- Select
- Plan and resource
- Monitor and report

Figure 10.5: Steps of Risk Management Cycle



There are four stages to risk management planning. They are:

- Risk Identification
- Risk Quantification

Notes

- Risk Monitoring and Control
- Risk Mitigation

10.5.1 Risk Identification

In this stage, we identify and name the risks. The best approach is a workshop with business and IT people to carry out the identification. Use a combination of brainstorming and reviewing of standard risk lists.

There are different sorts of risks and we need to decide on a project by project basis what to do about each type.

Business risks are ongoing risks that are best handled by the business. An example is that if the project cannot meet end of financial year deadline, the business area may need to retain their existing accounting system for another year. The response is likely to be a contingency plan developed by the business, to use the existing system for another year.

Generic risks are risks to all projects. For example the risk that business user might not be available and requirements may be incomplete. Each organization will develop standard responses to generic risks.

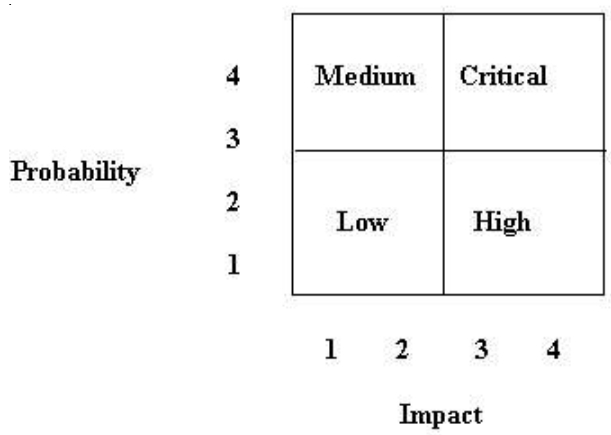
Risks should be defined in two parts. The first is the cause of the situation (Vendor not meeting deadline, Business users not available, etc.). The second part is the impact (Budget will be exceeded, Milestones not achieved, etc.).

Notes Hence a risk might be defined as "The vendor not meeting deadline will mean that budget will be exceeded". If this format is used, it is easy to remove duplicates, and understand the risk.

10.5.2 Risk Quantification

Risk need to be quantified in two dimensions. The impact of the risk needs to be assessed. The probability of the risk occurring needs to be assessed. For simplicity, rate each on a 1 to 4 scale. The larger the number, the larger the impact or probability. By using a matrix, a priority can be established.

Figure 10.6



Notes

Notes Note that if probability is high, and impact is low, it is a Medium risk. On the other hand, if impact is high, and probability low, it is High priority. A remote chance of a catastrophe warrants more attention than a high chance of a hiccup.

10.5.3 Risk Monitoring

The ongoing risk management task of monitoring the success and status of the other risk management tasks.

Classification

Risk monitoring is part of the following inheritance hierarchy:

- Type: Abstract
- Superclass: Engineering Task
- Subclasses: None

Responsibilities

The typical responsibilities of Risk monitoring are to Determine if:

- any aspect of the risk analysis has changed and therefore should be repeated.
- any undesirable event defining a risk has actually occurred.
- the other risk tasks are being performed effectively and efficiently.

Preconditions

Risk monitoring can typically begin when the following preconditions hold:

- The endeavor is started.
- The associated teams are initially staffed.
- At least one of these teams is adequately trained in risk monitoring including associated techniques and work products.
- The risk management plan exists and has passed evaluation.
- At least some risks have been identified, analyzed, and documented.

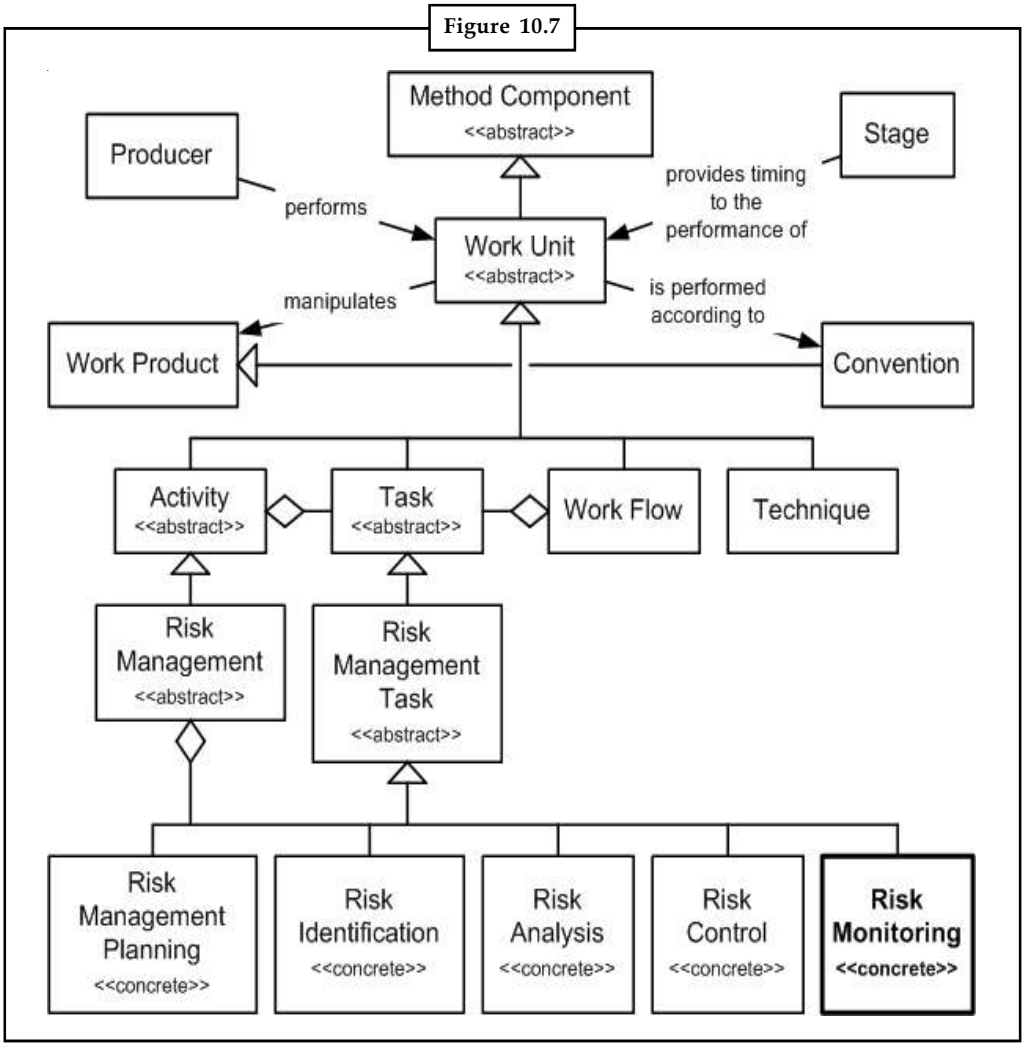
Completion Criteria

Risk monitoring is typically complete when the following post conditions hold:

- The risk monitoring reports have all been:
  - Produced
  - Evaluated for quality
  - Updated based on the quality control evaluation

Notes

Figure 10.7



- Approved
- Published to its stakeholders.

Because risk monitoring is an ongoing task, it is technically complete only when

- The endeavor has completed.
- The system, application, or center has been retired.

Steps

Risk monitoring typically involves members of the endeavor's teams performing the following steps in an iterative, incremental, parallel, time boxed, and ongoing manner:

- Determine if any risks have changed.
- Determine risk controls being used.
- Determine the effectiveness of the risk control actions and techniques.
- Develop the risk management plan.

Notes

Techniques

Risk monitoring can typically be performed using the following techniques:

- Assessment of current situation
- Auditing of current situation
- Cross Functional Teams to provide multiple viewpoints so that all aspects of risk management can be monitored
- Inspecting of current development or usage tasks
- Interviews with stakeholders, domain experts, and members of the development and operations organizations
- Incremental Development of the risk monitoring report
- Iteration of the risk monitoring report
- Observation of current development or usage tasks
- Parallel Development of the risk monitoring with other tasks
- Reviewing of current development situation

Work Products

Risk monitoring typically results in the production of all or part of the following work products:

- Risk monitoring Report

Guidelines

- Risk monitoring is an ongoing task that is not completed until the endeavor is finished.

10.5.4 Risk Mitigation

Intelligent Risk Mitigation Solutions

Acxiom's suite of advanced, information-based risk management solutions help organizations manage risk and make intelligent decisions. Acxiom's accurate, comprehensive database maximizes information currency and helps keep companies and their customers secure. Specific uses include fraud prevention, investigation, verification and authentication of individuals, and debt recovery.

Our portfolio includes:

- Fraud prevention solutions for the financial, insurance, health care, higher education, automotive, and telecommunications industries.
- Real-time transaction monitoring and detection for fraudulent behaviors.
- Investigative tools for governmental agencies, law enforcement and security firms.
- Debt-recovery solutions for financial institutions.
- A precise, accurate service for identifying the best address and telephone number for individuals.

Notes

- Background screening solutions that include criminal data, national watch list feeds, identity verification, court records and driver histories, as well as a comprehensive research network for in-person county court checks.

Risk Mitigation Solutions

Risk API: In today's economy, we understand the challenges you face and know how important it is for you to have reliable and user-friendly solutions to assist you in finding consumers.

InsightCollect: InsightCollect is a high quality tool used for collections, fraud management and risk mitigation. This online tool combines an easy-to-use format with in-depth data and advanced analytics, providing accurate, actionable information that can be used to locate and contact individuals.

Task Answers the following questions

1. What is risk monitoring?
2. What are the steps of risk management cycle?

10.6 Practical Techniques in Risk Management

In risk management, one has to identify clearly the goal and timeline of any risk mitigation planning, the "Exit Strategy." The Exit Strategy takes into account:

(a) the probability of various outcomes,

(b) their associated costs and benefits,

(c) method of measuring and reporting on the progress (transparency),

(d) have built in contingency plans to work around potential problems that may occur, and

(e) have a clear and easily measurable action plan for achieving best possible outcome.

The previous administration completely bumbled in implementing the first half of the TARP. There was no clear "Exit Strategy" for the $350 billion funding as the TARP did not did not have a clear and easily measurable action plan, method of measuring progress, or even contingency plans to work-around potential problems.

The complete ineptitude of the previous administration's handling of the TARP was laid bare by The Congressional Oversight Panel (COP) report, which came out a few days ago. The Congressional Oversight Panel lambasted the previous administration's shortsightedness of purpose, poor planning, fumbled execution, lack of transparency, and bad deal making by asking the Treasury in December 2008 the following 10 questions:

1. What is Treasury's Strategy?
2. Is the Strategy Working to Stabilize Markets?
3. Is the Strategy Helping to Reduce Foreclosures?
4. What Have Financial Institutions Done With the Taxpayers' Money Received So Far?
5. Is the Public Receiving a Fair Deal?
6. What is Treasury Doing to Help the American Family?
7. Is Treasury Imposing Reforms on Financial Institutions that are taking Taxpayer Money?

Notes

8. How is Treasury Deciding Which Institutions Receive the Money?
9. What is the Scope of Treasury's Statutory Authority?
10. Is Treasury Looking Ahead?

The Treasury should have had answers to these questions PRIOR to releasing the first $350 billion of the TARP. Even now, it appears that the Treasury is still unwilling (or unable) to address these basic issues, which only heightens the risk that the remaining $350 billion will be squandered yet again with no measurable benefit.

Since the Treasury was unable to answer any of the 10 questions, it would be very interesting to hear what the new Treasury Secretary Geithner will say next week. Whatever it is, I doubt that any of the questions posed by the Congressional Oversight Panel would be answered. The problem lies not only with the Treasury Department but also with President Obama. The Administration needs clearly state that lax accounting, poor planning, and insufficient results will not tolerated and, going forward, there will be strict performance and reporting requirements associated with the bailout. Without these directions, the risk of additional $350 billion being wasted is great.

10.6.1 Risks in Software Project Management

Unlike the hazards of daily living, the dangers in the young and emerging field of software engineering must often be learned without the benefit of lifelong exposure.

Notes A more deliberate approach is required. Such an approach involves studying the experiences of successful project managers as well as keeping up with the leading writers and thinkers in the field. One such writer in the area of risk is Dr. Barry W. Boehm.

In his article "Software Risk Management: Principles and Practices" he lists the following top 10 software risk items:

1. Personnel Shortfalls
2. Unrealistic schedules and budgets
3. Developing the wrong functions and properties
4. Developing the wrong user interface
5. Gold-plating
6. Continuing stream of requirements changes
7. Shortfalls in externally furnished components
8. Shortfalls in externally performed tasks
9. Real-time performance shortfalls
10. Straining computer-science capabilities

Risk management as being comprised of the following activities:

- Risk Assessment
  - making a list of all of the potential dangers that will affect the project
  - assessing the probability of occurrence and potential loss of each item listed
  - ranking the items (from most to least dangerous)

Notes

- Risk Control
  - coming up with techniques and strategies to mitigate the highest ordered risks
  - implementing the strategies to resolve the high order risks factors
  - monitoring the effectiveness of the strategies and the changing levels of risk throughout the project.

Task Discuss with your class fellows about different types of risks items and we can define lists the following top 10 software risk items:

1. Personnel Shortfalls
2. Unrealistic schedules and budgets
3. Developing the wrong functions and properties
4. Developing the wrong user interface
5. Gold-plating
6. Continuing stream of requirements changes
7. Shortfalls in externally furnished components
8. Shortfalls in externally performed tasks
9. Real-time performance shortfalls
10. Straining computer-science capabilities

10.7 Metrics in Risk Management

There are many metric sets proposed throughout in Software Engineering literature, and following is one very simple set of "core" attributes which drive the list of measures to collect:

- Size
- Effort
- Schedule
- Software quality
- Rework

Most metric sets deal with a variation of these attributes and are chosen to help project managers gain insight into their product (size, software quality, rework), process (rework, software quality) and project (effort, schedule).

10.7.1 Identifying Risks with Metrics

One simple measure that can be collected is staff level, sometimes called personnel, which counts the total number of software personnel available for a project. At initial risk identification, this measure can be used in a feasibility analysis and compared against the estimated staffing level. The results can give managers an indication of whether they will have enough personnel for a project or whether they will have to start looking for new team members.

Notes

However, this measure by itself will probably not give sufficient insight to help identify all personnel shortfall risks. Two further refinements of this measure are software development staff profile and software development personnel qualification, sometimes also called staff experience.

Software development staff profile is composed of several components which managers can measure and assess during the risk identification period:

- Staff level, which deals with actual availability for the project and considers whether a team member is full or part time, will be transferring before the project is over, going on family leave, etc. All this data is measurable and can be plotted against estimated staff requirements.
- Staff availability, which deals with whether the team member is actually available, in place and trained at the appropriate time of the development life cycle.
- Historic project and company retention rates, which can help project managers predict whether their team will be intact throughout the project, or whether they should plan for turnover and build sufficient slack in their estimates.
- Staff mix, which measures distribution by activity such as Quality Assurance (QA) and Configuration Management (CM), and helps managers determine whether they have enough people for each task.

Another measure is staff experience, or software personnel qualification, which deals with individual team members' proficiencies. Referring back to the list of underlying risk factors developed by the authors in "Toward an Assessment of Software Development Risk," team expertise is listed in four different categories as potential risk factors.

Staff experience or qualification level can refer to several different things:

- educational level
- years of experience with the company, indicating a knowledge of company standards as well as loyalty and dedication
- years of software development experience
- years of experience in the domain
- years of experience in the language
- years of experience on similar projects
- amount of specialty training
- years of relevant specialty training

All of this measurement data is available to managers who are trying to assess whether their team has sufficient experience to complete the planned project. None of this experience necessarily guarantees capability, but these types of measurements give managers a tool to determine whether their proposed team members can perform the tasks required of them.

Task Explain another measure is staff experience, or software personnel qualification, which deals with individual team members' proficiencies.

10.7.2 Tracking Risks with Metrics

Once managers have built their teams and the project has begun, their metrics programs allow them to track the progress of the project, product and process. The metrics programs also offer insight into those areas that were identified as potential risk items in the early planning stage.

Notes

Once they have begun to collect data, managers can begin to use performance analysis and look for trends to indicate whether or not a risk item is under control, as well as to indicate that a new, previously unidentified item may be becoming a risk.

The following is an example that discusses tracking the two personnel issues discussed above, staff level and staff experience.

Assume that personnel shortfalls were originally identified as a risk item, and that the project manager is closely tracking his personnel. The project manager should have plotted the planned staffing profiles for the total staff and for the experienced staff at the beginning of the contract. As time passes, some deviation from the curve is expected, but too great a deviation is cause for alarm. A program that does not have enough experienced personnel or that tries to bring too many into the project toward the end of the schedule is a project that is at risk. When looking at the shape of the planned software staff curve, it should grow through the requirements and design phases, peak in code and early test, and begin to fall as acceptance and integration tests are completed. The profile of the experienced staff should be high in the beginning of the project, decrease slightly during coding and increase again during test. The ratio of experienced personnel should be near to 3:1, but never exceed 6:1.

As discussed above, the staff level refers to the ability of the developer to maintain a sufficient level of staff to complete the project timely. In addition to tracking total staff, tracking experienced staff is also important as they are crucial to maintaining schedule and product quality. Finally tracking staff losses is important because staff turnover can impact the stability of the work force. Even though a team member leaves and is quickly replaced, there is usually an impact due to the earning curve while that new person is trained and acclimates to the existing team.

This sample chart is made up of only three measures, total staff level, experienced staff level, and unexpected staff losses. Nevertheless, it shows several things to a manager. Initially, the total number of personnel on the project was lagging behind the estimate, but the number of experienced personnel working on it was higher than planned. This could mean that there were problems getting enough personnel to work on the project at first and the shortage was being covered by a higher than planned number of experienced staff. It could also mean that the schedules can be maintained and the project is on track, but it may be at the expense of cost, since experienced personnel are usually more expensive. The manager should continue to monitor this.

The number of unplanned losses is nominal and does not seem to indicate any problems or risks. However, the number of experienced personnel is beginning to fall faster than the unplanned turnover rate. This may be a risk if the project is starting into the testing phase. Again, the manager may want to look at other measures or begin to track this item more closely.

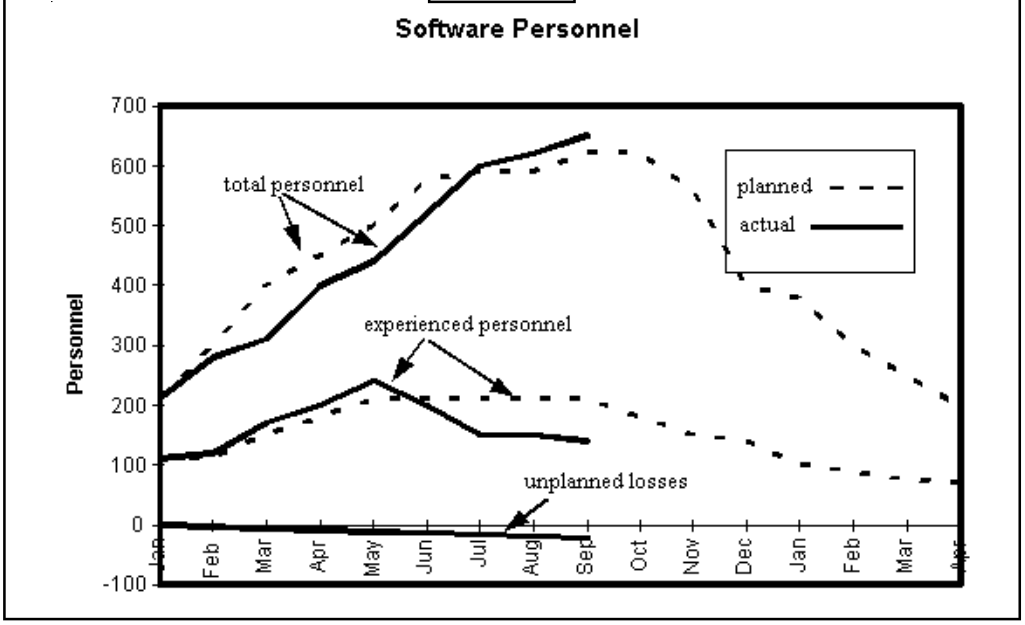
There are variations of this metric that can also be used to give the manager further insights. These include reporting staffing separately for each development task, e.g., QA, CM, or testing; and reporting staffing separately for special skills, e.g., Ada, client-server, or database development.

Normally, understaffing as seen in Figure indicates a possible schedule slippage that must be further tracked. The manager would do well to also look at the schedule and other indicators to assess the impact. If the project does continue to slip due to a personnel shortage, adding new personnel is not necessarily the answer, as this may add further delays due to the learning curve. If the turnover rate becomes too high, this could also become a major risk due to lack of continuity, impairing project knowledge and eroding the knowledge-base.

Since risk identification is an ongoing process, measures like the above can periodically be reviewed if other indicators point to staffing problems. Managers must be careful to include any experience obtained on the current project is an interim analysis is done.

Notes

Figure 10.8



Self Assessment

7. The results can not give managers an indication of whether they will have enough personnel for a project or whether they will have to start looking for new team members. True/ False

8. A program that does not have enough experienced personnel or that tries to bring too many into the project toward the end of the schedule is a project that is at risk. True/False

9. The metrics programs also offer ……..... into those areas that were identified as potential risk items in the early planning stage.

10. The number of …….....… losses is nominal and does not seem to indicate any problems or risks.

10.8 Categories of Risk

Are you developing any Test plan or test strategy for your project? Have you addressed all risks properly in your test plan or test strategy?

As testing is the last part of the project, it's always under pressure and time constraint. To save time and money you should be able to prioritize your testing work. How will prioritize testing work? For this you should be able to judge more important and less important testing work. How will you decide which work is more or less important? Here comes need of risk-based testing.

"Risk are future uncertain events with a probability of occurrence and a potential for loss".

Risk identification and management are the main concerns in every software project. Effective analysis of software risks will help to effective planning and assignments of work.

Notes

Schedule Risk

Project schedule get slip when project tasks and schedule release risks are not addressed properly.

Schedule risks mainly affect on project and finally on company economy and may lead to project failure.

Schedules often slip due to following reasons:

- Wrong time estimation
- Resources are not tracked properly. All resources like staff, systems, skills of individuals, etc.
- Failure to identify complex functionalities and time required to develop those functionalities.
- Unexpected project scope expansions.

Budget Risk

- Wrong budget estimation.
- Cost overruns
- Project scope expansion

Operational Risks

Risks of loss due to improper process implementation, failed system or some external events risks.

Causes of Operational risks

- Failure to address priority conflicts
- Failure to resolve the responsibilities
- Insufficient resources
- No proper subject training
- No resource planning
- No communication in team.

Technical Risks

Technical risks generally leads to failure of functionality and performance.

Causes of technical risks are:

- Continuous changing requirements
- No advanced technology available or the existing technology is in initial stages.
- Product is complex to implement.
- Difficult project modules integration.

Notes

Programmatic Risks

These are the external risks beyond the operational limits. These are all uncertain risks are outside the control of the program.

These external events can be:

- Running out of fund
- Market development
- Changing customer product strategy and priority
- Government rule changes.

Did u know? What is Risk-based Testing

Testing is used in software development to reduce risks associated with a system. By testing, we hope to identify many of the problems before they get to the customer, thereby reducing the system's risk. Unfortunately, even through testing alone can't find all of the bugs, the rapid pace of application development today, challenges even the act of just completing testing.

10.9 Assessment

10.9.1 Schedule Risk Assessment Plan

A plan typically consists of a series of iterations and each iteration consists of work items that are scheduled to be completed within an iteration. You provide the estimate for each work item. Given the number of work items and their estimates within the duration of an iteration, work items can be categorized from very probably to improbably able to complete in an iteration. The Schedule Risk Assessment plan helps in determining the work items that might go in or slip away in an iteration.

The Schedule Risk Assessment plan displays the risk assessment for each team member of a team area or project area. The work items in the plan are sequenced in the same order as in the Current Work section of the My Work view. The risk assessment is calculated in the same sequence. The work items in the Inbox folder are not assessed for risk.

In the Schedule Risk Assessment plan, in addition to the original estimate, you can provide a minimal and maximal estimate.

Example: Assess the risk of a sprint

Work item owners can specify minimal and maximal estimates in addition to the already existing estimates. This information is used to simulate the probability of work items being completed on time.

If the minimal estimate is not provided, the original estimate is divided by 2 to get the minimal estimate. Similarly, if the maximum estimate is not provided, the original estimate is multiplied by 2 to get the maximal estimate. These estimates are used to simulate the probability of work items being completed on time.

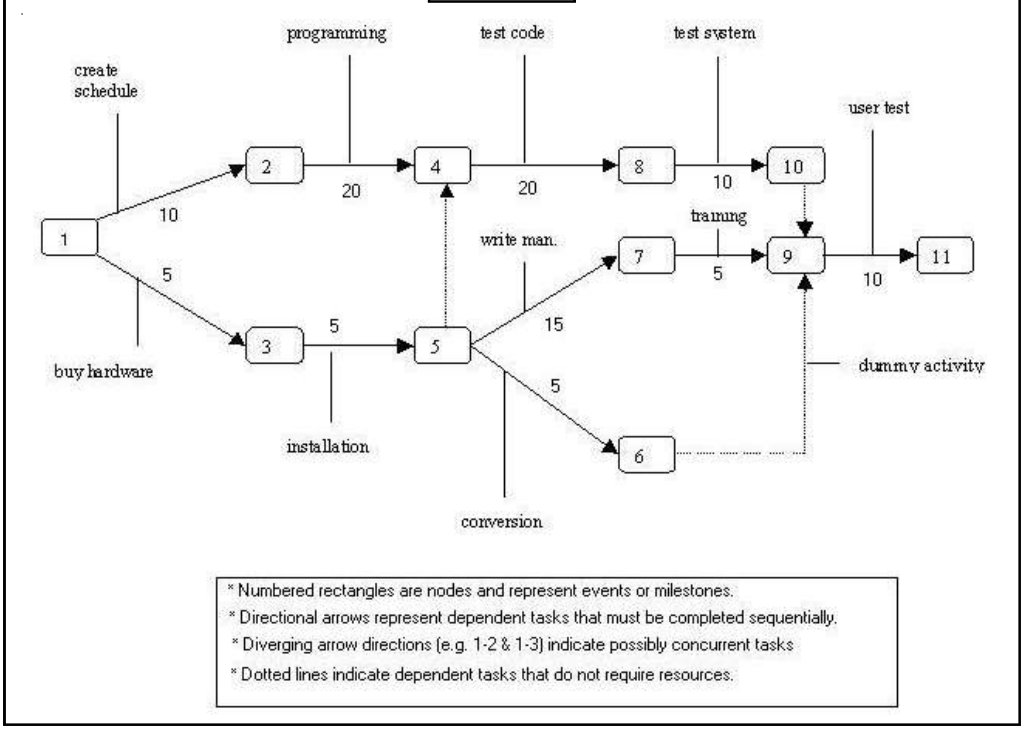
To provide these estimates, click in the plan. Click **Assess Schedule Risk** in the sidebar to simulate the probability. The simulation assumes that each contributor completes the work items in the same order as specified in the Current Work section of the My Work view. The probability of completion for each work item is indicated by a color gradient from white to red; from very probable to improbable.

Notes

10.10 PERT Technique

A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. PERT stands for *Program Evaluation Review Technique*, a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program. A similar methodology, the *Critical Path Method* (CPM) was developed for project management in the private sector at about the same time.

Figure 10.9



A PERT chart presents a graphic illustration of a project as a network diagram consisting of numbered *nodes* (either circles or rectangles) representing events, or milestones in the project linked by labelled *vectors* (directional lines) representing tasks in the project. The direction of the arrows on the lines indicates the sequence of tasks. In the diagram, for example, the tasks between nodes 1, 2, 4, 8, and 10 must be completed in sequence. These are called *dependent* or *serial* tasks. The tasks between nodes 1 and 2, and nodes 1 and 3 are not dependent on the completion of one to start the other and can be undertaken simultaneously.

These tasks are called *parallel* or *concurrent* tasks. Tasks that must be completed in sequence but that don't require resources or completion time are considered to have *event dependency*. These are represented by dotted lines with arrows and are called *dummy activities*. For example, the dashed arrow linking nodes 6 and 9 indicates that the system files must be converted before the user test can take place, but that the resources and time required to prepare for the user test (writing the user manual and user training) are on another path. Numbers on the opposite sides of the vectors indicate the time allotted for the task.

The PERT chart is sometimes preferred over the Gantt chart, another popular project management charting method, because it clearly illustrates task dependencies. On the other hand, the PERT chart can be much more difficult to interpret, especially on complex projects. Frequently, project managers use both techniques.

Notes

PERT is a variation on Critical Path Analysis that takes a slightly more skeptical view of time estimates made for each project stage. To use it, estimate the shortest possible time each activity will take, the most likely length of time, and the longest time that might be taken if the activity takes longer than expected.

Use the formula below to calculate the time to use for each project stage:

$$\frac{\text{shortest time} + 4 \times \text{likely time} + \text{longest time}}{6}$$

Self Assessment Questions

Fill in the blanks

11. As testing is the last part of the project, it's always under pressure and .....…… constraint.
12. …….. risks generally lead to failure of functionality and performance.
13. A …… presents a graphic illustration of a project as a network diagram.
14. The …… was developed for project management in the private sector at about the same time.
15. The Schedule Risk Assessment plan displays the …….. for each team member of a team area or project area.

Caselet

Operational Risk Management — How Banks can Manage the Unknown

WHAT if suddenly ATMs stopped vending crisp notes, bank branches closed for few days, the data centre of major banks shut down, busy operations in dealing rooms of major banks come to a halt and banking personnel don't reach their offices.

This is not a doomsday scenario but what actually happened during the Mumbai floods. Uncertainty has crept into our lives. In technical parlance, we can call the risk involved in running daily operations "operational risk", or op-risk, for short.

In the banking industry, op-risk is as old as banking itself. The banking landscape has undergone a sea change and is becoming more complex in terms of volume of business, product innovation, financial engineering, new market practices, fast and rapid technology innovation, deregulation, consolidation of banks and increasing competition among banks.

This has increased probability of failure or mistakes from the operations point of view; it has increased the focus on managing op-risk.

The new BIS guidelines, generally known as "Basel II Accord", recognises this and places increased emphasis on op-risk management. The Basel committee defines op-risk as the risk "of loss resulting from inadequate or failed internal processes, people and systems or from external events". This definition includes legal risk, but excludes strategic and reputation risk. As banks move towards implementing Basel II norms, they need to evolve an internal framework for effective management of op-risk.

Contd.....

Notes

Depending on the size, complexity and organisational structure of bank, a five-step approach can be used for building a robust op-risk framework:

- Identification of operational risk through event framework
- Analysing the causes of events
- Risk mapping
- Risk measurement and control
- Management of operational risk and, thereby, capital management.

To start with, banks, as part of identification, should classify and capture all operational losses in the form of "events". Events are nothing but "occurrences" or "happenings". Banks should start accumulating data on events that have occurred in the past and also identify potential events.

All events should be defined with attributes, such as, frequency of event, severity, loss amount, reason for loss, date of discovery of loss and date of occurrence.

Banks can adopt the seven type of events suggested by Risk Management Group (RMG) of Basel committee for one of their quantitative studies (QIS-2) which includes internal fraud, external fraud, employment practices and work safety, client products and business services, damages to physical assets, business disruption and system failures and execution delivery.

The second step involves doing a causal analysis to understand the exact cause for the above events and estimate the actual loss as well as potential loss in case the events are repeated. This analysis on cause of events can make the bank understand the level of exposure and the op-risk management strategy it needs to adopt.

Once banks have developed an event database and done the causal analysis, they can start risk mapping. Risk mapping is a tool wherein banks can map the above risk events and losses to any specified set of business lines.

Basel has come out with eight set of business lines — corporate finance, trading and sales, retail banking, commercial banking, payment and settlement, agency and custody services, asset management and retail brokerage — to which the events collected by bank can be mapped.

Op-risk measurement is still evolving in terms of tools and techniques that can be used for effective measurement and management. Banks can follow either or both of qualitative risk measurement or quantitative risk measurement.

The generic ways of measuring op-risk include qualitative risk measurement techniques such as critical assessment method, which involves questionnaire format and interviews with all line managers to identify the op-risk events.

Another widely used approach, which is a combination of qualitative as well as quantitative approaches, is the Key Risk Indicators (KRI) approach, which involves identifying indicators, which convey good idea about the scope of business and thereby the risk involved.

For instance, portfolio size, volume of transactions traded, volume of deals routed through payment and settlement systems, etc., form one set of predictive indicators. KRI is more a predictive model than a cause-and-event approach.

A common quantitative approach used is Loss Distribution Approach (LDA), which involves arriving at a right fit distribution of historical loss events and, thereby, at quantitative results like expected loss and finally operational value at risk.

Contd.....

Notes

Another forward-looking scenario generation approach for op-risk measurement is Loss Scenario Modelling, which involves generating simulations for loss scenarios based on the events and losses captured in the first step.

Basel II norms suggest three approaches for measurement of op-risk. The simplest approach, best suited for less sophisticated and small balance-sheet banks, is the Basic Indicator Approach (BIA). BIA requires banks to allocate capital based on a single indicator of operational risk, which in this case will be average gross income of past three years multiplied by factor called alpha, which is set at 15 per cent.

The second approach is the Standardised Approach (SA), which involves mapping the bank's business lines to the set of eight business lines and use multiplier (Beta) of average gross income to compute capital charge.

Also, there is the Alternative Standardised Approach (ASA), which uses loans and advances, instead of gross income, for retail banking and commercial banking business lines multiplied by fixed factor which results in capital charge to be set aside.

The most sophisticated approach suggested is advanced measurement approach (AMA). Under the AMA, the regulatory capital requirement will equal the risk measures generated by the bank's internal operational risk measurement system using quantitative and qualitative criteria for the AMA. Internal data used must be based on a minimum historical observation period of five years. However, when a bank first moves to AMA, a three-year period is acceptable.

Banks need to employ the quantitative approaches like Internal Measurement Approach (IMA) or Loss Distribution Approach (LDA) or Balance Scorecard Approach (BSA) for adopting AMA. All AMA approaches compute the expected and unexpected loss. The most significant aspect for a bank to graduate from Basic Indicator Approach (BIA) to Advanced Measurement Approach (AMA) is the potential benefit of less capital allocation for operational risk.

As op-risk involves failures during operations in daily business, the key steps in op-risk management involve improving internal control environment, designing and developing procedures to implementing the risk management processes and employing risk transfer techniques, such as insurance, to mitigate the loss arising from operational risk. Credit rating agencies have started rating banks based on their risk control and management frameworks. Investor awareness has also increased to the extent that banks with robust risk management frameworks are able to attract strategic investments with less effort.

Given the known benefits of implementing the provisions of the Basel II accord, banks should prioritise their strategy towards op-risk management. A constructive approach in this direction could be to automate the suggested five-step approach and, as a first step, to start developing a loss event database.

10.11 Summary

- Software project managers need to manage risk and use every tool available to them for this management.
- If they can use a tool that is already being used on their project for other purposes, they save themselves time and money.
- Most managers use some form of a metrics program to track their project for cost, schedule, effort, and quality.

Notes

- Many of the measures used to help them with this project management can also serve additional use in identifying and tracking risk.
- This unit used a common software risk, personnel shortfall, to show how managers can use measures and metrics to help identify and track risk items.
- This technique can be applied to other common risk items, such as requirements changes and unrealistic schedules and budgets, to help managers have visibility into and control over their overall projects in addition to identifying and monitoring their risk items.

10.12 Keywords

QA: Quality Assurance

CM: Configuration Management

Risk: The possibility of suffering harm or loss; danger

COP: Congressional Oversight Panel

10.13 Review Questions

1. What is risk management? Examine the various components of risk management.
2. What do you think are the common uses of PERT technique?
3. What are categories of risk? Explain how "Risk are future uncertain events with a probability of occurrence and a potential for loss."
4. Make distinctions between operational and budget risk.
5. Risk management is the discipline of identifying, monitoring and limiting risks. Analyse.
6. Analyze the distinctive points between Risk Assessment and Risk Control.
7. What is Schedule Risk Assessment Plan? Explain how the plan typically consists of a series of iterations and each iteration consists of work items that are scheduled to be completed within an iteration.
8. Make an analysis and write a note on Schedule Risk and Technical Risk. Examine their differences.
9. Risk, at the general level, involves two major elements: the occurrence probability of an adverse event and the consequences of the event. Explain
10. Examine the impact of SWOT analysis on risk management.

Answers: Self Assessment

1.	Risks	2.	Visualized
3.	Requirements	4.	project management
5.	Risk Strategy	6.	accurately
7.	False	8.	True
9.	insight	10.	unplanned
11.	time	12.	Technical
13.	PERT chart	14.	Critical Path Method (CPM)
15.	risk assessment		

Notes

10.14 Further Readings

Books

B. W. Boehm, *Software Risk Management: Principle and Practices*, IEEE Software, Vol. 8, No. 1, January 1991, pp. 32-41.

B. W. Boehm, *A Spiral Model of Software Development and Enhancement*, Computer, May 1988, pp. 61-72.

Online links

http://en.wikipedia.org/wiki/Risk_management

http://www.risk-management-india.com/

Notes

Unit 11: Resource Allocation

CONTENTS

Objectives
Introduction
11.1 Resource Allocation
11.2 Identifying Resource Requirements
11.3 Scheduling Resources
11.3.1 Resource Scheduling – Example
11.3.2 Computerized Resource Scheduling
11.3.3 Benefits of Resource Scheduling
11.4 Resource Schedule and Cost Schedule
11.4.1 Resource Schedule
11.4.2 Cost Schedule
11.5 Scheduling Sequence
11.6 Summary
11.7 Keywords
11.8 Review Questions
11.9 Further Readings

Objectives

After studying this unit, you will be able to:

- Describe resource requirement
- Explain scheduling resource
- Discuss resource schedule
- Explain cost schedule

Introduction

Up to this point, we have identified the activities in the project and developed a schedule that meets the expected end date of the project. We have also estimated the resources to be used in the project and cost of the project. Now, this unit discusses the final step of assigning the resources according to the schedule developed (by leveling resources) and to determine if you can accomplish this schedule with the resources available. The impact on the cost of the project is also considered on the basis of leveled resources.

11.1 Resource Allocation

Effective Resource Management for Team Projects and Goals

Project Insight gives project managers power over the management of resource allocation for software development, marketing, product development teams and more. Assigning team

Notes

members to business goals, projects and individual tasks is simple and easy with our PMI and PMBOK® Guide compliant solution. Mass assign team members' tasks grouped by skill set, department or resource type, or handle resource allocation management for a single person. It is equally simple to change a resource on a set of project tasks as well.

Our portfolio system allows resource allocation managers and project managers to use project level and/or cross project resource allocation to manage workloads in order to achieve their goals. The software application reports evenly divide the work (hours) among the workdays (duration) scheduled for the tasks to calculate the total work or effort assigned to a resource within a specified date range.

Efficient Resource Allocation and Workload Management

Resource information may be accessed from the 'Resources' tab within a project to review the availability of resources. Project Insight, web project management software provides real-time resource allocation data based on the allocation of their assignments to project tasks system-wide.

Project managers can also view all resources across all projects in Project Insight. This information is accessed in 'My Reports,' 'Cross Project Resource Allocation.' Data may be hidden or displayed according to each person's preferences, supporting a wide variety of applications for these reports. Hundreds of permutations of resource allocation reports are available.

Other project management software applications claim to have extensive resource allocation capabilities in their marketing materials; however, they often fall short. Project Insight not only allows resource managers or project managers to see the total workload each resource has per day, week or other time period, it allows them to drill down on all of the projects and tasks that are causing the over allocation in one view. Tasks can easily be reassigned using Project Insight's simple drag and drop functionality. It's perfect for the management of all kinds of goals, tasks and projects including IT projects, interactive or marketing projects, product development projects, professional services and more. All tasks are efficiently managed with proper resource allocation and tracking, down to the last detail.

Notes If your goal is to make sure that you're on top of resource allocation, then Project Insight's powerful resource management applications will make it happen.

11.2 Identifying Resource Requirements

As you plan for application compatibility testing, keep in mind the future state of your computing environment. Are you planning to upgrade some of your software to versions that fully use new Windows 2000 features? Are you planning to implement new standard desktop configurations or use Terminal Services? Issues such as these determine the resources that are required and the applications that are to be tested as a suite.

If you plan to deploy new applications with Windows 2000 during the rollout, test these applications with the current applications.

You can facilitate testing by setting up a lab where testers can conduct their tests. In such a lab, you can have the necessary tools and equipment available at all times. Some organizations have a lab for testing applications that is separate from the Windows 2000 lab. If you do not have the budget for a separate lab, you might share a lab with another project or with training. If you share a lab, try to choose one that has compatible scheduling and equipment requirements.

Notes

In the lab, set up the test computers for dual or triple startup so that testers can quickly access the mode they need to install and test their applications.

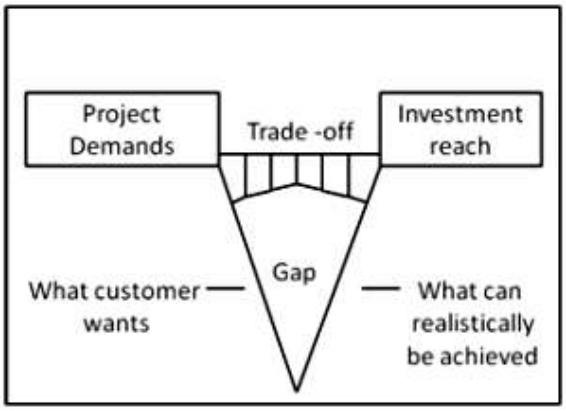
Example: If you follow the strategy suggested in "Testing Applications" later in this unit, you might need Windows NT 4.0 and Windows 2000 to test the applications through the upgrade path. To make it easy for testers to restore the computers to their prior state, make disk images of the drives with the base operating systems.

Consider whether you need to connect the lab to the corporate network. For example, you might need access to network shares for installing applications from the network or to your corporate intranet if you develop a Web-based test tracking system. If you need such access, first verify that the protocol stacks used on the client computers are compatible with your production network.

If your test lab is large, you might decide to assign a lab manager. Because the skills required to run a lab and to manage testing are so different, consider selecting different people for the two roles. While the lab manager needs to have strong technical skills, the testing manager needs to have strong managerial and communication skills.

Resources are the means we use to achieve project objectives. The primary resource is obviously people with applicable skills and competencies. The other main grouping of resources we need includes capital, facilities, equipment, material and information. There is usually a gap between the investment reach of a project and the project demands. See diagram below

Figure 11.1



In order to ensure a cost effective application of required resources we must perform a proper needs analysis in order to define the project goals and objectives. The project's baseline that must be resourced will be known after a requirement specification has been completed. The WBS is the base document for determining resource requirements.

Task The requirement specification will spell out the real requirements to achieve through the specific project. Explain

Inputs to Resource Planning

- The WBS identifies the project elements that will need resources. It is therefore the primary input to resource planning.

Notes

- Historical information as to what types of resources were required for similar work on previous projects is valuable input. In many instances there are industry standards available that you can consult.
- The scope statement contains the project justification and the project objectives which should be considered.
- A resource pool description is another useful input. It constitutes knowledge of what resources (people, equipment and material) are potentially available.

Resource Estimation

Resource estimation is a structured prediction of the cost and other resources required to execute a task. One of the primary functions of the process is to establish a control basis. Therefore, the more accurate the estimation, the more reliable the control system becomes.

The required accuracy and effort going into resource estimation can be influenced by the element of uncertainty and risk involved because of technical complexity and novelty of the project. The expected price basis of the contract could make resource estimation a critical factor. The higher the risk the more important it becomes to have a realistic estimation, which implies a more detailed process.

The amount of effort that it will take to complete a task is also important. Effort is determined by the time it will take to complete a task. Once you understand the effort that's required for a project, you can assign resources to determine how long the project will take and estimate human resources and other resource costs.

A properly documented Resource Plan will specify the exact quantities of human resources, equipment and materials needed to complete your project.

Resource Acquisition

Resource acquisition refers to the process of physically securing the necessary inputs. All resources required have to be paid for in some way or another. The financing of a project therefore plays an essential role in the acquisition process.

The acquisition process must be managed properly to take care of possible seasonal shortages, labour disputes, equipment breakdowns, competing demands, delayed deliveries and other things that may go wrong. The project plan may have to be modified to accommodate or work around supply problems.

Resource Levelling

Resource levelling is the process that ensures resource demand does not exceed resource availability and vice versa. We don't want extreme peaks and valleys in the execution of the tasks. The first step is usually to move non critical tasks with float to a later date. If you are forced to reschedule a task on the critical path it would influence the completion dates of successive tasks. Check to see whether it will have an influence on meeting the project deadline.

Human Resource Planning

We use a responsibility assignment matrix to allocate roles and responsibilities. The WBS is the primary document for doing this. List the project activities on the left-hand side of the matrix and role-players on top of the matrix. List a brief description of the work to be done in the matrix cells. After completion of the RAM you will know exactly who will do what on the project and

Notes

what their respective roles are. If it is a large project you can then draw up an organisation chart for the project. Once the staff have been assigned you can plan for briefings, coaching, training and all the activities re the management of the project teams and staff.

Resource Allocation

The different types and quantities of resources must be allocated to the project as needed to perform the activities. A schedule for this must be maintained to ensure that the allocation process takes place on time and in the right quantities. In many disciplines there are specific methods and documentation to ensure this e.g. a bill of materials. There may also be some conflicting needs for equipment that must be managed.

The delivery of resources is a very important point. During contracting you must know exactly where resources will be originated and where it will be delivered to. Sometimes delivery to site is a good option. You don't want to be liable for transportation costs of equipment that may be sourced more than thousand kilometres away.

Notes Although we have discussed the most important aspects in this article there are many other tasks to perform in order to deal with the resource requirement process fully. The resourcing of a project will also be dependent on the procurement process which includes selection criteria, resource cycles, payment and storage. Examine your project and decide how the project should be resourced and what the critical factors are.

We am currently developing an online project management course based on the Moodle *Learning Management System*. It is a platform with a virtual classroom where you have a learning space with different resources e.g. text lessons, discussion forum, templates, video and chat. Interaction makes the learning interesting and much more dynamic than traditional methods. I will also cover the resourcing of a project fully in the online course.

Self Assessment

Fill in the blanks:

1. Mass assign team members' tasks grouped by skill set, …….. or resource type.
2. The software application reports evenly divide the work (hours) among the workdays (duration) scheduled for the tasks to calculate the ……. or effort assigned to a resource within a specified date range.
3. Project managers can also view all …… across all projects in Project Insight.
4. The scope statement contains the project ……. and the project objectives which should be considered.
5. The different types and ……. of resources must be allocated to the project as needed to perform the activities.
6. The required accuracy and effort going into resource ……. can be influenced by the element of uncertainty and risk.

Notes

11.3 Scheduling Resources

In this unit we will learn about scheduling resources in projects. We will begin by discussing the nature of resource requirements (both people and machines) and the problems associated with managing resources in a project environment. Given the finite nature of resource availability, a project plan may have to be modified so that it is practical. This is the major thrust of resource planning and management. In this unit, we will examine, at some length, the four major stages of the resource scheduling process. These stages are **resource definition**, **resource allocation**, **resource aggregation**, and **resource leveling**. *Resource definition* involves identifying the critical resources that need to be planned and managed for the successful completion of the project. In a multi-project environment as projects are competing for scarce resources, *resource allocation* addresses the problem of the optimum use and timing of the assignment of these resources to the various project activities. *Resource aggregation* involves determining the aggregate resources that will be needed, period by period, to complete all project activities. Having identified the necessary resource requirements, the last stage in the process is *resource leveling*. In this stage, we attempt to ensure that the demand for resources does not exceed availability. Specifically, demand for resources is smoothed to ensure that the peaks and valleys are reduced. In this lesson, we will also learn about the "critical chain approach" to tackle resource dependencies that occur in projects due to reduced slack.

Caution This unit is a critical topic in project management as careful planning and management of resources can prevent cost overruns in the future.

Project Constraints

In this unit, we will discuss the essential features of resource planning and management in projects. We begin this lesson by first understanding the different kinds of project constraints, in particular, the types and nature of resource constraints.

The primary impact of project constraints is the likelihood of delaying the completion of the project. There are three types of project constraints: technological, resource and physical. The technological constraints relate to the sequence in which individual project activities must be completed. For example, in constructing a house, pouring the foundation must occur before building the frame. Resource constraints relate to the lack of adequate resources which may force parallel activities to be performed in sequence. The consequence of such a change in network relationships is delay in the completion date of the project. We will examine the nature of resource constraints in much greater detail in the next section. Physical constraints are caused by contractual or environmental conditions.

Example: Due to space limitations an activity such as painting a wall may have to be performed by only one person (Gray and Larson, 2003).

In general, from a scheduling perspective, projects can be classified as either time constrained or resource constrained. A project is classified as time constrained in situations where the critical path is delayed and the addition of resources can bring the project back on schedule and the project completed by the required date. However, the additional resource usage should be no more than what is absolutely necessary. The primary focus, for purposes of scheduling, in time constrained projects is resource utilization. On the other hand, a project is resource constrained if the level of resource availability cannot be exceeded. In those situations where resources are inadequate, project delay is acceptable, but the delay should be minimal. The focus of scheduling in these situations is to prioritize and allocate resources in such a manner that there is minimal project delay.

Notes

Notes However, it is also important to ensure that the resource limit is not exceeded and the technical relationships in the project network are not altered.

Resource Constraints

The most important resources that project managers have to plan and manage on day-to-day basis are people, machines, materials, and working capital. Obviously, if these resources are available in abundance then the project could be accelerated to achieve shorter project duration. On the other hand, if these resources are severely limited, then the result more likely will be a delay in the project completion time. Depending on the type of resources, the costs of providing an abundance of such resources to accelerate project completion time can be very high. However, if resources are readily available and excess premiums are not incurred to use them on the project, then project cost should be low, as some project costs are resource related while others are likely to be time dependent. In general, projects with a shorter duration are less expensive. The longer the duration of the project, the higher will be overall project cost due to the increase in fixed costs such as overheads. The reality is that as long as the work on a project is ongoing it will continue to draw resources into its orbit. Whatever the parameters of the project, it is unlikely that the relationship between cost and duration is linear. For any particular project, the decision to place the project on the curve between the point of least duration with its associated higher resource requirements and a point of increased duration with its associated lower resource requirements depends on the particular parameters of the project.

When a project plan is first devised it is likely that the plan will identify peaks of resource requirements. However, given the finite nature of resource availability, it may be impractical to meet such peak resource needs. Ideally, there should be an even demand for resources over the entire project duration, with a smooth increase at the beginning of a project and a smooth decrease at the end. Given the limited nature of resources, thoughtful consideration should be given to the project resource requirements; the project plan should be refined when necessary so that it is practical.

Did u know? What are the four stages?

The process of refining the plan to effectively manage and schedule resources (sometimes referred to as resource modeling) comprises four major stages: resource definition, resource allocation, resource aggregation, and resource leveling (which includes resource smoothing).

Resource Allocation

Resource allocation, also called resource loading, is concerned with assigning the required number of those resources identified in the previous step to each activity identified in the plan. More than one type of resource may be attributed to a specific activity. For example, fixing the plates on a ship's hull may require 10 fitters, 20 welders, 15 laborers and a certain type of welding machine. From a practical standpoint, resource allocation does not have to follow a constant pattern; some activities may initially require fewer resources but may require more of the same resources during the later stages of the project. At this stage, the impact of any resource allocation decision is not known and we cannot yet answer questions such as:

- Is lack of resources on this particular activity having an adverse effect on the duration of the whole project? Such an activity is more likely to be on the critical path.

Notes

- By excessive use of resources are we completing this activity more quickly than necessary in terms of the overall project duration? Such an activity is not likely to be on the critical path.

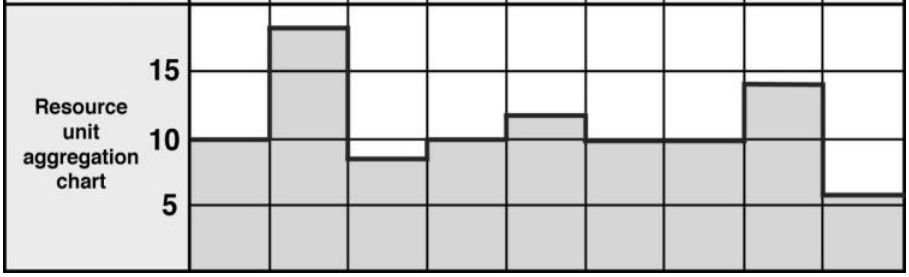
These questions will be answered later in the resource modeling process, specifically during the resource leveling and smoothing stage.

Resource Aggregation

Resource aggregation, or resource loading, is simply the summation, on a period-by-period basis, of the resources required to complete all activities based on the resource allocation carried out in the previous stage. The results are usually shown graphically as a histogram. Such aggregation may be done on an hourly, daily, or weekly basis, depending on the time unit used to allocate resources. When a bar chart is used as the planning tool, the resource aggregation is fairly simple and straightforward. For a given bar chart, there is a unique resource unit aggregation chart which can be drawn underneath the bar chart. However, a separate graph will be required for each resource unit. An example is shown in Figure 11.2 below, where, for a particular resource, the required resource units for each time period are annotated on the bar chart. The total number of resource units for each time period can then be summed and a resource aggregation or load chart can be produced.

Figure 11.2: Resource Unit Aggregation Chart Derived from a Bar

Activity \ Week	1	2	3	4	5	6	7	8	9
A	10	10							
B		8	8	6					
C				2	6				
D				2	6	10	10	8	
E								6	6
Total resource units requirement	10	18	8	10	12	10	10	14	6

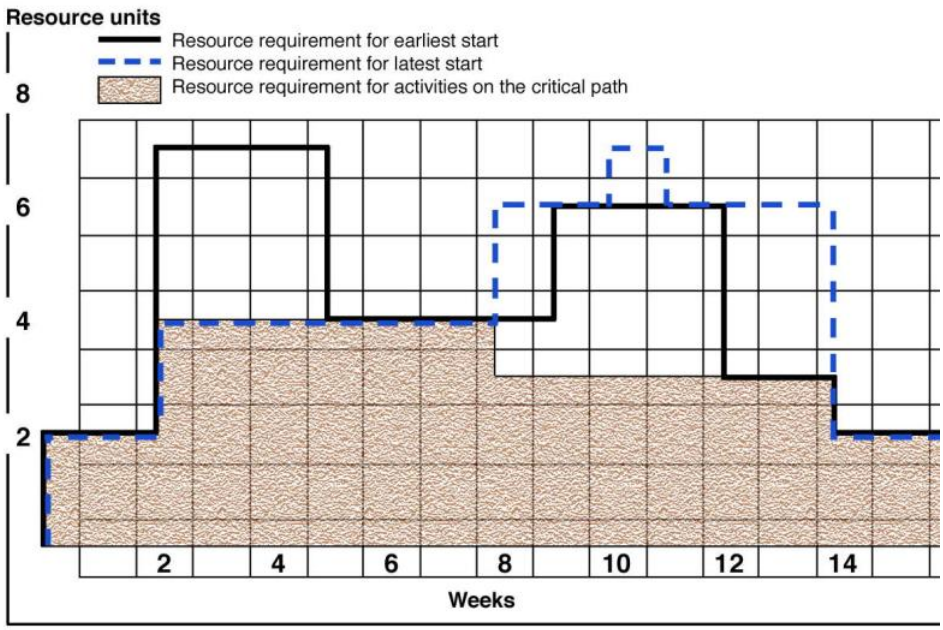


However, when a network is used for planning, the resource aggregation procedure is not so simple or straightforward. As the network is not drawn to a time-scale, there is not a direct link between the network and the demand for resources. Therefore, a schedule must be prepared which tabulates activities in terms of time. However, this highlights another difficulty, namely

Notes

that those activities which are not on the critical path do not have fixed starting and finishing times but are constrained by the earliest and latest starting and finishing times. However, this seeming difficulty offers the planner considerable scope for adjusting the demand for resources. This will be discussed in more detail later, but the limits within which resources can be adjusted, without extending the overall project duration, are the resource requirements between the earliest starting times and the latest starting times. This is illustrated in Figure 11.3, which shows the differing resource requirements that arise when both earliest and latest start times are considered and also highlights the resource requirements for those activities which are on the critical path.

Figure 11.3: Resource Unit Aggregation



Resource Leveling

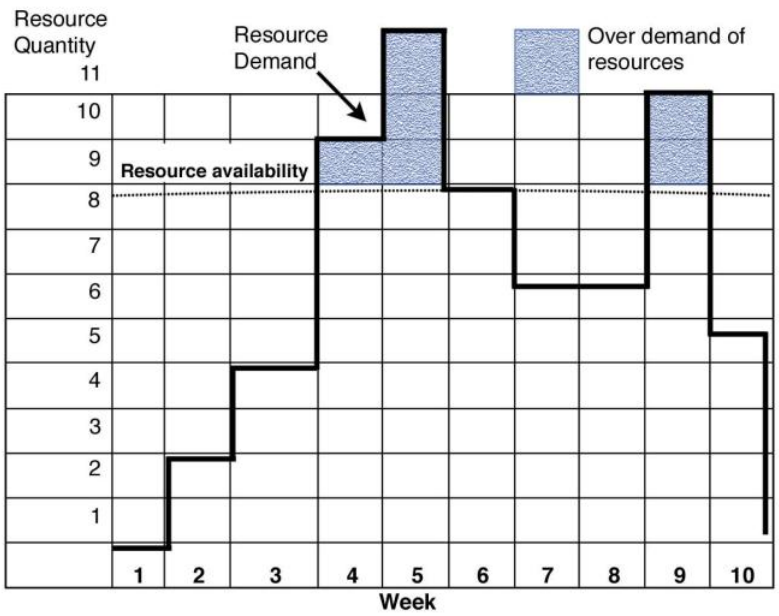
Having established the resource requirements through resource allocation and aggregation, we will now examine the next phase of the planning and resource management process—resource leveling. We will now compare those requirements with resource availability by developing resource profiles. However, the most likely scenario is that, at some point, demand will exceed supply. Such a scenario is illustrated in Figure 11.4.

Notes Disregarding factors such as economic considerations, if sufficient resources are available so that supply always exceeds demand then, we should have no problem.

Resource leveling is the process that ensures resource demand does not exceed resource availability. The ideal scenario would be a build up of resource usage at the beginning of the project and a reduction at the end of the project. However, the approach to resource leveling will also depend on whether resources are dedicated to a particular project or shared across several projects and whether there is a need to keep all resources fully utilized.

Notes

Figure 11.4: Resource Demand Compared to Resource Availability



We will begin by analyzing the issues involved in resource leveling for a situation where a bar chart has been used as the primary planning technique for a simple project. The reason for this is that resource leveling must be considered within a time framework and bar charts are drawn to a time scale while networks are not. Examine Figure 11.2 in this figure, the time-scale for the activities comprising the project are shown in a bar chart, which also shows resource requirements for one particular resource unit. An examination of the bar chart and its associated resource chart in Figure 11.2 shows that improvements can be made to the level of resource requirements by:

- Delaying or bringing forward the start of certain activities.
- Extending the duration of certain activities and so reducing the demand for resources over the duration of the activity or by a combination of both of these adjustments.

However, there are problems with using the simple bar chart as a tool for resource leveling. For example, we do not have any information about the interdependency of tasks. Therefore, if we delay a task by starting later than originally planned or by extending the duration of the task, we cannot evaluate the exact impact this will have on the overall project. Referring to Figure 11.2 again, if we assume that the maximum amount of resource availability is 14 units, then we have a problem in week 2 because 18 units of resources are required in that week. In order to reduce the resource demand in week 2, we may have to extend Activity A into week 3 (if this is possible) and spread the resource demand over three weeks, or delay the commencement of Activity B. However, the exact impact of these changes on the overall project duration cannot be easily determined.

Another issue is that the critical path(s) cannot be easily determined, although we may be able to deduce which activities are critical by inspection. Clearly, if we do not wish to extend the overall duration of the project we must avoid extending or delaying activities which are on the critical path.

Notes Finally, the availability of slack or float is not clear. Knowing this is important because it is this attribute that can be utilized to adjust our resource requirements.

Notes

Resource leveling can be accomplished more easily if resource requirements to complete an activity are expressed in terms of hours or days required. The definition of resource requirements using such units of measure can help us determine if an activity should be completed in a short time through the use of many resources or over a longer period of time through the use of fewer resources. In practice, however, there is a limit to the number of resources that can be deployed and, therefore, a limit to the amount by which any activity duration can be shortened.

We will now examine situations where networks are used as the primary planning method. Generally, there are two approaches to leveling and smoothing the resources required:

- ***Time-limited resource considerations:*** In this case emphasis will be placed on completing the project within a specified time. This time will usually have been determined by network analysis. Adjustments in the timing of any activity, and the resources required at a given time, must be undertaken within the float (slack) available. Obviously there can be no adjustment of activities which are on the critical path.
- ***Resource-limited resource considerations:*** In this case the project must be completed with the resources available even if this means extending the project duration. If the total resource demand exceeds the resource availability at any time then some of the activities must be delayed until there is sufficient resource availability.

For both of the above approaches, information concerning the earliest and the latest.

Resource Smoothing

Resource smoothing is part of the resource leveling process. In itself, resource smoothing is the process that, not withstanding any constraints imposed during the leveling process, attempts to determine a resource requirement that is "smooth" and where peaks and troughs are eliminated. For example, even if 7 units of a given resource are available at any one time, utilizing 5 of these units each week is preferable to 4 one week, 7 the next, 2 the next and so on. Even if there is no limit to the amount of any one resource available, it is still desirable that resource usage is as smooth as possible. Given that the resource requirements of those activities on the critical path are fixed, some order or priority needs to be established for selecting which activity and which particular resource associated with this activity should be given priority in the smoothing process. In determining which activity should be given priority, a subjective judgment should be made about the type of resource (or resources) associated with each activity; priority should be given to the activities whose resources are considered to be most important. Beyond this consideration, activities should be ranked in order of total work content and total float or slack available for that activity.

Did u know? What is the useful device for prioritizing?

A useful device for prioritizing is to consider the ratio of total work content/total float remaining and give priority to activities with the highest value of this ratio.

Solving the resource scheduling problem for optimal solutions is extremely complex, particularly for large project networks with many different resource types. However, several heuristics are available to solve such problems. These heuristics allocate resources to activities to minimize project delay based on certain priority rules. The two most commonly used heuristics are the serial and the **parallel** methods. In the serial method of resource allocation, activities are sorted into a list and resources are allocated to each of these activities one at a time until resources are allocated to all activities. In the parallel method, however, resources are allocated on a period by period basis rather than each activity. In this method only those activities whose preceding activities have been completed will be considered. If two or more activities compete for the same resources, then allocation of resources is based on certain prescribed priority rules.

Notes

Compared to the serial method, the parallel method has been the most widely used heuristic. The following priority rules, in the order presented, have been found to be the most effective in minimizing project delay.

- Minimum slack
- Smallest duration
- Lowest activity identification number

Regardless of the scheduling heuristic used, the primary impact of resource constrained scheduling is the loss of flexibility due to the reduction in slack. Furthermore, the reduction in slack also increases the number of critical or near-critical activities.

11.3.1 Resource Scheduling – Example

We will now go through an example to demonstrate the resource scheduling process for a resource constrained scenario using the parallel heuristic and the priority rules given above. We will solve two problems. In solving these problems, two critical assumptions are made.

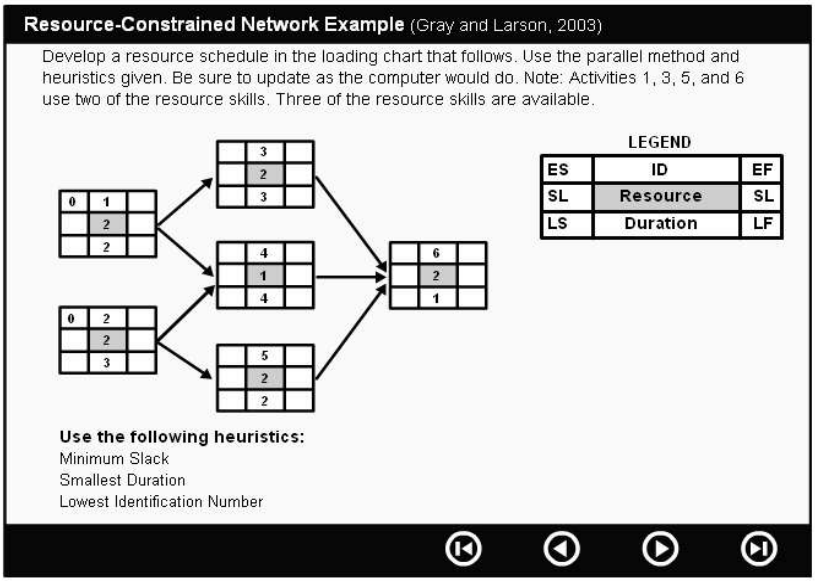
- No splitting activities are allowed, i.e., once an activity is placed on the schedule; it will be worked on continuously until it is finished.
- The resource level used for an activity cannot change.

In actual practice, however, these limiting assumptions do not exist.

Resource Constrained Network Example (Gray and Larson, 2003)

Develop a resource schedule in the loading chart that follows. Use the parallel method and heuristics given. Be sure to update as the computer would do. Note: Activities 1, 3, 5, and 6 use two of the resource skills. Three of the resource skills are available. To proceed through the example, press the next icon in the Flash slide show below.

Figure 11.5: Resource Loading Chart Example

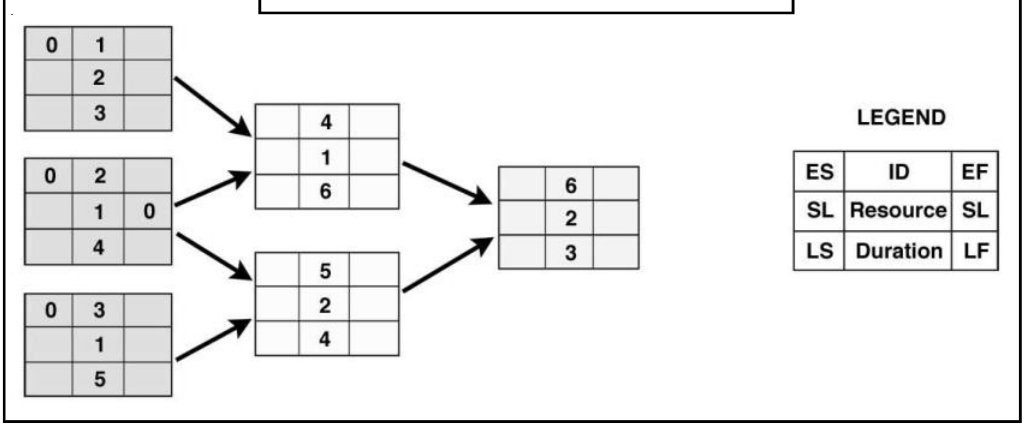


Time-Constrained Network - Example (Gray and Larson, 2003)

Notes

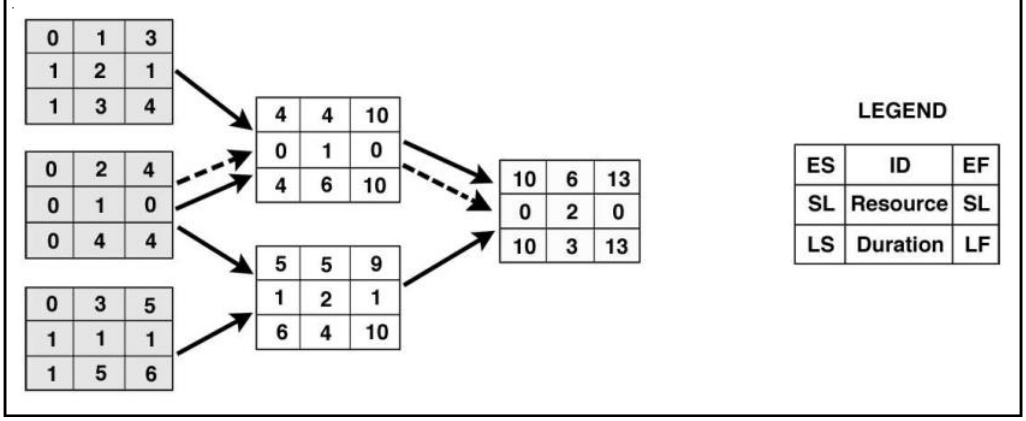
This next example has several parts. We will discuss each of them in some detail.

Figure 11.6: Time-Constrained Network Example



First, compute the early, late, and slack times for the activities in the network in Figure 11.7, assuming a time-constrained network. Which activities are critical? What is the time constrained project duration?

Figure 11.7: Time-Constrained Network Example Showing Early, Late, and Slack Times



Now, assume you are a computer using a software that schedules projects by the parallel method and the following heuristics. Schedule only one period at a time!

- Minimum slack
- Smallest duration
- Lowest activity identification number

Caution Remember to maintain the technical dependencies of the network.

Notes

Figure 11.8: Scheduled Resource Load Chart with ES and Slack Updates

ID	RES	DUR	ES	LF	SL	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	2	3	0	4	1	2	2	2													
2	1	4	0	4	0	1	1	1	1												
3	1	5	0,1,2,3	6	1,0, -1.-2	X	X	X	1	1	1	1	1								
4	1	6	4	10	0					1	1	1	1	1	1						
5	2	4	5,6,7,8	10	1,0, -1,-2						X	X	X	2	2	2	2				
6	2	3	10,11,12	13	0,1, -2											X	X	2	2	2	
Resources Scheduled					3	3	3	2	2	2	2	2	3	3	2	2	2	2	2	2	
Resources Available					3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	

Notes The parallel method schedules resources to various activities through leveling and smoothing. This is accomplished in the above problem by delaying and reducing the slack on activities 3, 5 and 6. Using the load profiles presented above, graphical resource aggregation charts, similar to the ones presented earlier in this lesson, can be developed.

Next, keep a log of each activity change and the update you make each period—e.g., period 0-1, 1-2, 2-3, etc. The log should include any changes or updates in ES and slack times each period, activities scheduled and activities delayed. (Hint: Remember to maintain the technical dependencies of the network.) The log is shown in Table 11.1 below.

Table 11.1: Log of the Parallel Method of Scheduling

Period	Activity	Changes
0-1	2	Schedule activity 2 first by the minimum slack rule
	1	Schedule activity 1
	3	Delay activity 3 ES to period 1. Reduce slack to 0.
	5	Delay activity 5 ES to period 6. Reduce slack to 0.
1-2	3	Delay activity 3 ES to period 2. Reduce slack to -1.
	5	Delay activity 5 ES to period 7. Reduce slack to -1.
	6	Delay activity 6 ES to period 11. Reduce slack to -1.
2-3	3	Delay activity 3 ES to period 3. Reduce slack to -2.
	5	Delay activity 5 ES to period 8. Reduce slack to -2.
	6	Delay activity 6 ES to period 12. Reduce slack to -2.
3-4	3	Schedule activity 3
4-5	4	Schedule activity 4
5-6	---	No changes
6-7	---	No changes
7-8	---	No changes
8-9	5	Schedule activity 5
9-10	---	No changes
10-11	---	No changes
11-12	---	No changes
12-13	6	Schedule activity 6

Notes

Notes The parallel method schedules resources to various activities through leveling and smoothing. The log presented above shows how this was accomplished in the above problem by delaying and reducing the slack on activities 3, 5, and 6.

Now, list the order in which you scheduled the activities of the project. Which activities of the schedule are now critical?

The order is (2, 1, 3, 4, 5, 6) and the critical activities are 2, 3, 5, and 6 as these are the activities with the least or negative slack.

Finally, recompute the slack for each activity given the new schedule. What is the slack for activity 1, 4, 5?

For this, see the answer to the second question. The slack for 1 = (0), 4 = (2), and 5 = (0).

11.3.2 Computerized Resource Scheduling

The resource scheduling examples that we solved above used the manual approach. For large networks, however, this is not a feasible approach. Fortunately, all the better quality computer packages for project management, besides network analysis, also feature the ability to schedule activities taking into account resource constraints. Thus for most people engaged in project management today, a computerized approach is preferred. While a full description of the way in which current packages perform resource scheduling is beyond the scope of this topic, these packages commonly use the serial scheduling and parallel scheduling algorithms. The basic features of both these algorithms were described earlier in this lesson, even though the computer packages use more sophisticated versions of these algorithms. Given the nature of the resource constraints, each method may produce a feasible schedule; however, the schedules need not necessarily be the same. Furthermore, if the same problem is solved by two different project management packages, different schedules may be generated. A computer demonstration of resource constrained scheduling is also presented in your textbook.

11.3.3 Benefits of Resource Scheduling

The process of scheduling resources before the project begins provides the following benefits:

1. If project delay is unacceptable, it allows sufficient time for considering alternatives such as cost-time trade-offs and changing of priorities.
2. Provides information to prepare time-phased work package budgets with dates.
3. Enables project managers to determine the amount of flexibility they have over certain resources.

Self Assessment

Fill in the blanks:

7. ……, also called resource loading, is concerned with assigning the required number of those resources identified in the previous step to each activity identified in the plan.
8. The most important resources that project managers have to plan and manage on day-to-day basis are people, machines, materials, and ……..

Notes

9. …….. involves identifying the critical resources that need to be planned and managed for the successful completion of the project.

10. In a ……. environment as projects are competing for scarce resources, *resource allocation* addresses the problem of the optimum use and timing of the assignment of these resources to the various project activities.

11. …….. involves determining the aggregate resources that will be needed, period by period, to complete all project activities.

11.4 Resource Schedule and Cost Schedule

11.4.1 Resource Schedule

It helps you to identify all of the resources required to complete your project successfully.

Using this *Resource Plan*, you will be able to identify the quantity of labor, equipment and materials needed to deliver your project.

You will then create a resource schedule, which enables you to plan the consumption of each type of resource, so that you know that you will have enough resources to complete the project.

This **Resource Planning** template will help you identify the:

- Types of labor required for the project
- Roles and key responsibilities for each labor type
- Number of people required to fill each role
- Items of equipment to be used and their purposes
- Types and quantities of equipment needed
- Total amount of materials needed

This Resource Plan template will also help you to:

- Plan the dates for using or consuming these resources
- Identify the amount of resource required per project activity
- Create a detailed resource utilization schedule.

11.4.2 Cost Schedule

Cost planning ensures value for money and responsible management of public monies of a capital project's finances, including the requirements set out in the code of practice for the building and construction industry.

Utilising all the cost planning estimates templates (CPA, CPB, CPC1, CPC2, CPD), guarantees that a project can be costed as accurately as possible, as well as assist in the determining of the economic building lifespan and all future redevelopments that may occur and their estimated costing.

Cost planning is to allow for options to be analysed and may be required to determine best value for money by assessing recurrent costs over the life of the facility, as well as the capital cost. The cost planning and analysis may include a comprehensive cost-benefit analysis or cost-effectiveness analysis for the preferred option. Throughout the life of a project; project objectives, client requirements or social, economic and environmental conditions may change. This can

Notes

affect the budget and costs of the identified delivery options. If there is significant change to the service plan, asset and property configurations or similar, that impact on the preferred option(s), consideration should be given to reworking the options analysis process.

Cost plans for Capital Works Projects are prepared by consultants in conjunction with the development of successive stages of planning and design. Project managers and PCGs use the cost plans to exercise financial monitoring and control.

It is intended that the Total End Cost budget elements be assessed and modified on an individual project basis as set out in the *section Allowances* and are dependant upon the risks assessed at each phase of the project.

Did u know? What is CMB and PCG?

CMB and PCGs will then be able to use the cost plan categories set out in the section *Allowances* to exercise better financial monitoring and control. It is expected that sufficient funds will be available to introduce energy efficiency and waste management measures and cover additional items such as FFE and IT requirements.

11.5 Scheduling Sequence

You use the scheduling sequence to specify in which sequence the system executes the scheduling or rescheduling of an operation group or order group. You can set up the sequence using various criteria, for example:

- Order priority
- Resource
- Product number

The scheduling sequence is relevant for:

- Manual scheduling and rescheduling of operations and order groups using drag and drop.
- Scheduling and rescheduling of operations and order groups using the planning function Reschedule.
- Scheduling with detailed scheduling heuristics.

The scheduling sequence is not relevant for scheduling an order. The scheduling sequence is valid for all detailed scheduling strategies in a strategy profile.

Self Assessment

Fill in the blanks:

12. Using this ……, you will be able to identify the quantity of labor, equipment and materials needed to deliver your project.
13. Cost plans for ……. are prepared by consultants in conjunction with the development of successive stages of planning and design.
14. Project managers and …… use the cost plans to exercise financial monitoring and control.
15. It is intended that the Total End …….. budget elements be assessed and modified on an individual project.

Notes

Caselet

Kalam for Efficient Resource Allocation

Our Bureau

NEW DELHI, April 9

THE President, Dr A.P.J. Abdul Kalam, on Wednesday expressed concern over the mounting debt burden of both the Centre and the States, which, he felt, would have to be tackled in order to ensure higher economic growth and significant poverty reduction.

"Many States are burdened with debt. The Union Budget also shows that one-fourth of the Centre's total receipts go for interest payments," Dr Kalam said, while addressing the Golden Jubilee function of the Finance Commission.

According to Dr Kalam, if India is to become a "developed country", its economy would have to register an annual growth of nine per cent, so that the 26 crore people currently below the poverty line emerge out of it by the year 2020.

The President directed the Twelfth Finance Commission (TFC) to evolve a mechanism for efficient resource allocation to ensure poverty reduction through proper fiscal management.

The Finance Minister, Mr Jaswant Singh, said more than the fiscal deficit, it was the revenue deficit and primary deficit (i.e. fiscal deficit net of interest payments) of the Government, which needed to be addressed seriously.

He felt that for a 'continental-sized' economy, the average annual growth rate of six per cent registered during the past decade was 'remarkable', though "we have to continue and reach the horizon".

The TFC Chairman, Dr C. Rangarajan, said there was a need for all State Governments to keep their fiscal deficit within 'reasonable limits' while simultaneously meeting developmental responsibilities.

For this, there was a need to raise additional revenues, especially from non-tax sources such as levying cost-based user charges on power, transport and water, and also prune non-developmental.

11.6 Summary

- Project Insight gives project managers power over the management of resource allocation for software development, marketing, product development teams and more.
- Assigning team members to business goals, projects and individual tasks is simple and easy with our PMI and PMBOK® Guide compliant solution.
- Mass assign team members' tasks grouped by skill set, department or resource type, or handle resource allocation management for a single person. It is equally simple to change a resource on a set of project tasks as well.
- In this unit we will learn about scheduling resources in projects. We will begin by discussing the nature of resource requirements (both people and machines) and the problems associated with managing resources in a project environment.
- Given the finite nature of resource availability, a project plan may have to be modified so that it is practical.

Notes

- This is the major thrust of resource planning and management.
- In this lesson, we will examine, at some length, the four major stages of the resource scheduling process.

11.7 Keywords

Resource Definition: Resource definition involves identifying the critical resources that need to be planned and managed for the successful completion of the project.

Resource Allocation: In a multi-project environment as projects are competing for scarce resources, resource allocation addresses the problem of the optimum use and timing of the assignment of these resources to the various project activities.

Resource Aggregation: Resource aggregation involves determining the aggregate resources that will be needed, period by period, to complete all project activities.

Resource Leveling: Having identified the necessary resource requirements, the last stage in the process is resource leveling.

11.8 Review Questions

1. Describe resource requirement system with an example.
2. Analyze the performance of resource allocation and resource aggregation systems.
3. Evaluate the concept of resource schedule and cost schedule.
4. Project Insight gives project managers power over the management of resource allocation for software development, marketing, product development teams and more. Comment.
5. The project's baseline that must be resourced will be known after a requirement specification has been completed. Discuss.
6. The WBS is the base document for determining resource requirements. Comment.
7. Resource leveling is the process that ensures resource demand does not exceed resource availability. Explain.
8. The resource scheduling examples that we solved above used the manual approach. Give the list of examples and explain.
9. Cost planning ensures value for money and responsible management of public monies of a capital project's finances, including the requirements set out in the code of practice for the building and construction industry. Discuss.
10. The most important resources that project managers have to plan and manage on day-to-day basis are people, machines, materials, and working capital. Analyse.

Answers: Self Assessment

1.	Department	2.	total work
3.	resources	4.	justification
5.	quantities	6.	estimation
7.	Resource allocation	8.	working capital

Notes

9. Resource definition
10. Multi-project
11. Resource aggregation
12. Resource Plan
13. Capital Works Projects
14. PCGs
15. Cost

11.9 Further Readings

Books

B. W. Boehm, *Software Risk Management: Principle and Practices*, IEEE Software, Vol. 8, No. 1, January 1991, pp. 32-41.

B. W. Boehm, *A Spiral Model of Software Development and Enhancement*, Computer, May 1988, pp. 61-72.

Online links

http://en.wikipedia.org/wiki/Resource_allocation

http://www.mdm.gwu.edu/Forman/Chapter8.pdf

Unit 12: Monitoring & Control

Notes

CONTENTS

Objectives
Introduction
12.1 Creating Frameworks
12.2 Data Collection
12.2.1 Types of Tools
12.2.2 Construction of Schedules and Questionnaires
12.2.3 Measurement Scales and Indices
12.2.4 Pilot Studies and Pre-tests
12.3 Visualizing Progress by Progress Report
12.4 Change Control
12.4.1 Change Control Activities
12.4.2 Usage of Metadata
12.5 Purpose of Controls
12.6 Balancing the Control System
12.7 Progress Reporting System
12.7.1 Types of Project Status Reports
12.7.2 How and What Information to Update
12.7.3 Variances
12.8 Graphical Reporting Tools
12.8.1 Using the WBS to Report Project Status
12.9 Managing Project Status Meetings
12.9.1 What is their Format?
12.9.2 Problem Management Meetings
12.10 Summary
12.11 Keywords
12.12 Review Questions
12.13 Further Readings

Objectives

After studying this unit, you will be able to:

- Describe Creation of frameworks
- Explain data collection
- Discuss Visualizing Progress
- Explain Cost Monitoring
- Describe Change Control

Notes

Introduction

At this point, you have put considerable effort into building and getting approval for a project plan that describes in great detail how you will accomplish the goal of the project. The project work has begun, and you want to make sure that it is progressing as planned. To do this, you will institute a number of reports that are designed to tell exactly how well the project is doing with respect to the plan and how to correct variances from this plan. The question to consider is the extent to which you want to maintain control through the reports you require. In this unit we will study about the monitoring and controlling progress of the project.

12.1 Creating Frameworks

Considering that the Project Management Plan is the baseline for the project. This is guide for monitoring and controlling the project. As a project manager, one will need access to work performance information, performance reports, and change request. This information will need to be at your fingertips as inputs to yield project performance indicators. After analyzing and reviewing the information, it is time to decide whether corrective or preventative actions are needed.

The four inputs to monitoring and controlling project work are:

1. ***Project Management Plan:*** The project management plan is the main source of information about how the project will be executed, monitored, and controlled. It is the plan, with all additional subsidiary plans as needed.
2. ***Work Performance information:*** Work performance information is the information about project activities. This includes status information about progress, deliverables, expenses, and quality assurance validations.
3. ***Rejected Change Requests:*** Reject change request can be enlightening when reviewed in the context of determining how the progress of the project is fairing.

Remember the quality of the measurement is only as good as the data gathered to ascertain the measurement. While it may take time, being vigilant and reviewing this information will save one headaches in the long run.

So how do you sort through the granular details and formulate it into information? Well the good news is this isn't a single problem, it affects all project managers, and there are standardized tools and techniques for monitoring and controlling project work.

These tools and techniques are:

- ***Expert Judgment:*** On the basis of current project information and experience with similar projects, project managers and team members can use expert judgment to make decisions, such as whether to take corrective or preventive actions.
- ***Earned Value Technique (EVT):*** Earned Value Technique (EVT) provides project managers with a means of calculating current project schedule and cost performance. Project managers can then use this information to forecast future schedule and cost performance.
- ***Project Management Methodology:*** The organizational project management methodology provides project managers with detailed guidance and procedures to enable effective monitoring and control through each stage of a project.
- ***Project Management information system (PMIS):*** A PMIS allows for monitoring and controlling parameters such as cost and resource usage. A PMIS can also enable project managers to calculate and manage earned value information, as well as request and update project information automatically.

Using the correct tools and techniques will help keep you in the loop of what's going on with your project.

Notes

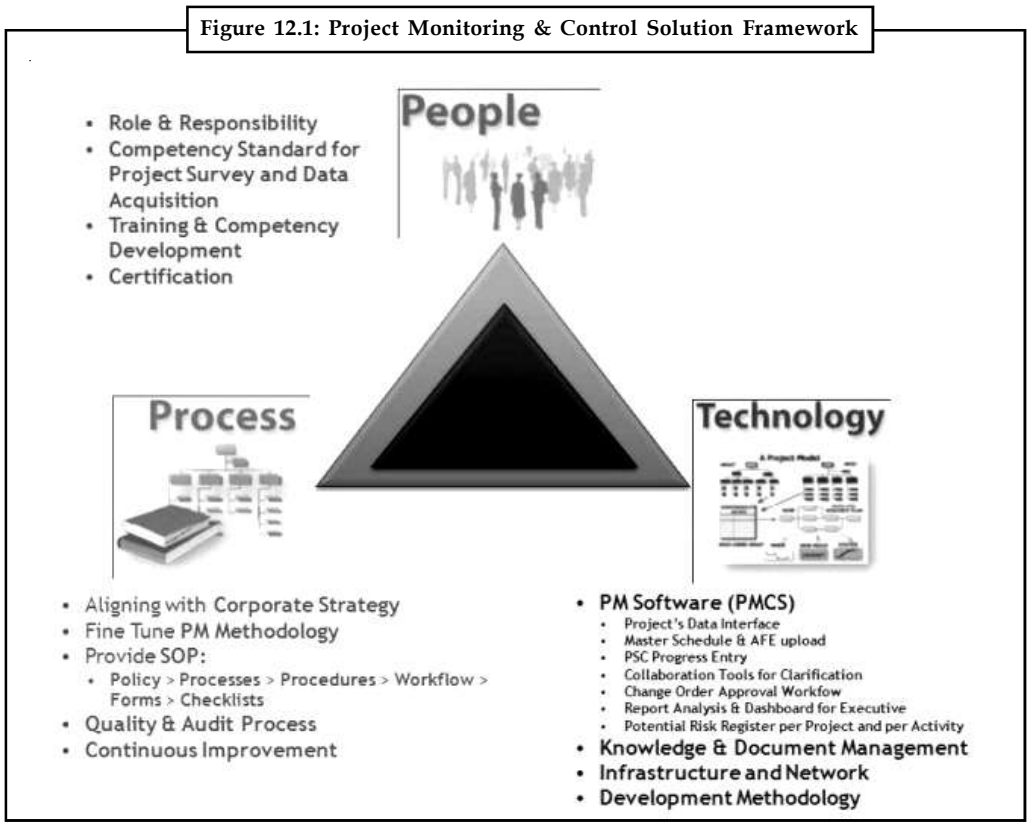
Notes Obviously as one monitors and controls there will be some action items that occur. These action items cycle back into the project execution. So what are some outputs or action items that occur during the Monitor and Control Project Work process?

The outputs of monitoring and controlling project work are:

- ***Recommended Corrective Actions:*** These are based on project work performance information. By comparing this information to the project plan, the project manager or team uses expert judgment to put forward ideas to remedy problems that have arisen.
- ***Recommended Preventive Actions:*** These are based on project work performance information. By comparing this information to the project plan, the project manager or team uses expert judgment to suggest ways of avoiding project risks.
- ***Forecasts:*** Based on work performance information received during the Monitor and Control Project Work process, forecasts allow the prediction of successful or unsuccessful project outcomes.
- ***Recommended Defect Repairs:*** These are an output of monitoring and controlling project work. This output recommends the remedial work necessary when a product does not meet quality requirements.
- ***Requested Changes:*** These are revised actions that are necessary for meeting project objectives. The requests are often made by the project manager or members of the project team as a way of improving methods or overcoming problems.

Just keep in mind that every organization is different, and your experiences of monitoring and controlling project work will depend on your company's project management methodology and the type of project.

Figure 12.1: Project Monitoring & Control Solution Framework



Notes

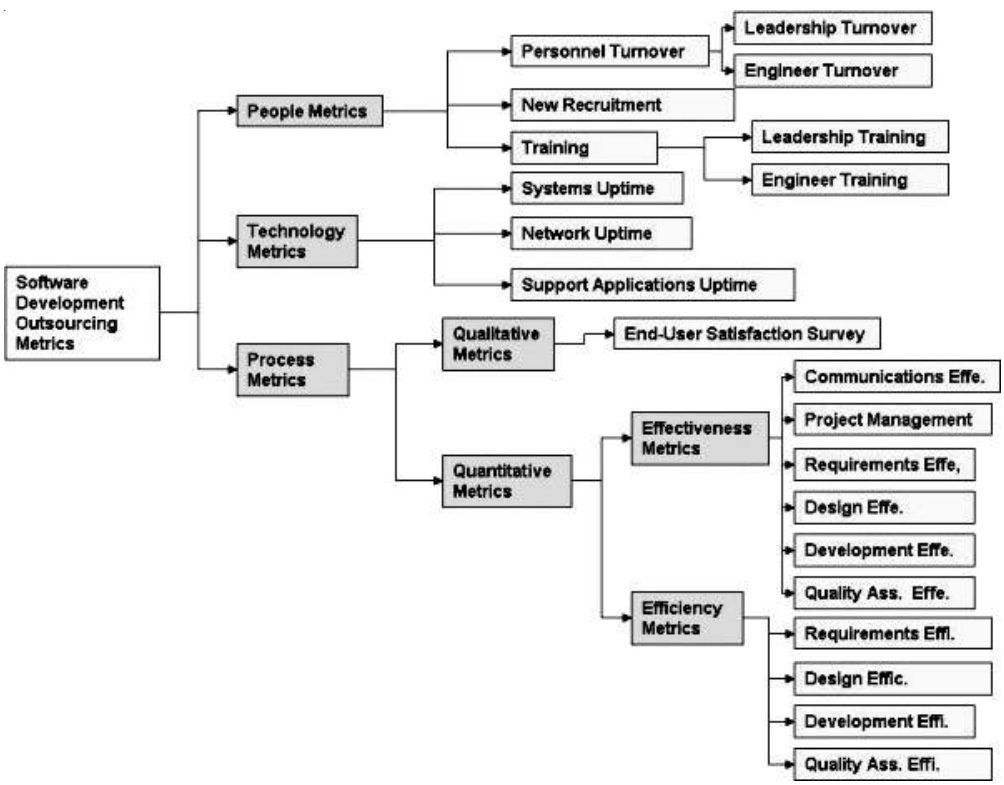
There are a number of frameworks available for defining projects and for managing their implementation. At Data Perceptions we use our own project methodology that we have developed through years of experience in the development and deployment of IT projects.

The following diagram illustrates our Entrepreneurial Culture – Iterative System Development IT project framework. The EC-ISD project framework is a practical approach that provides a scalable structure, which encourages discipline and good practices. It offers an overview of the steps involved in each IT project, although the level of detail required for each step will depend on the complexity and criticality of the individual project.

For smaller, less critical projects, for example, all steps in the first two columns may be completed in a two hour meeting followed by an email summary. For larger projects, multiple meetings and revisions may be required to fully define the project requirements and scope.

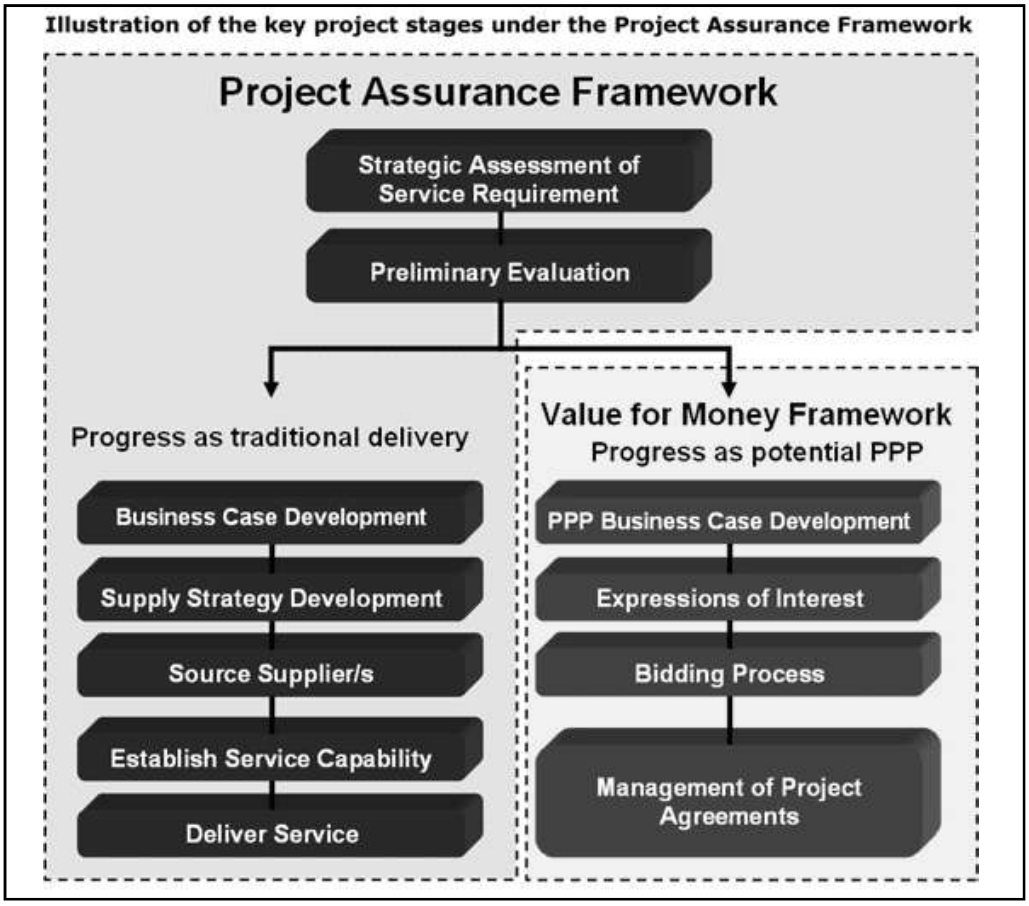
Caution In either case, the methodology provides an inclusive structure to follow, which ensures that no element essential to the success of your IT project is overlooked.

Figure 12.2: Frameworks



Notes

Example:



Self Assessment

Fill in the blanks:

1. Considering that the …….. is the baseline for the project.
2. The ……. project framework is a practical approach that provides a scalable structure.
3. A …….. allows for monitoring and controlling parameters such as cost and resource usage.
4. There are a number of ……. available for defining projects and for managing their implementation.

12.2 Data Collection

Data Collection is an important aspect of any type of research study. Inaccurate data collection can impact the results of a study and ultimately lead to invalid results.

Data collection methods for impact evaluation vary along a continuum. At the one end of this continuum are quantitative methods and at the other end of the continuum are Qualitative methods for data collection.

Notes

The question of "what data to collect?" The answer is to collect all of the data required to provide the metrics primitives and the additional qualifiers.

In most cases, the "owner" of the data is the best answer to the question of "who should collect the data?" The data "owner" is the person with direct access to the source of the data and in many cases is actually responsible for generating the data. Table 12.1 illustrates the owners of various kinds of data.

Benefits of having the data owner collect the data include:

- Data is collected as it is being generated, which increases accuracy and completeness.
- Data owners are more likely to be able to detect anomalies in the data as it is being collected, which increases accuracy.
- Human error caused by duplicate recording (once by data recorder and again by data entry clerk) is eliminated, which increases accuracy.

Once the people who gather the data are identified, they must agree to do the work. They must be convinced of the importance and usefulness of collecting the data. Management has to support the program by giving these people the time and resources required to perform data collection activities. A support staff must also be available to answer questions and to deal with data and data collection problems and issues.

Table 12.1: Examples of Data Ownership

Owner	Examples of Data Owned
Management	• Schedules • Budgets
Engineers	• Time spent per task • Inspection data including defects • Root cause of defects
Testers	• Test cases planned/executed/passed • Problem reports from testing • Test coverage
Configuration Management Specialists	• Lines of code • Modules changed
Users	• Problem reports from operations • Operational hours

A training program should be provided to help insure that the people collecting the data understand what to do and when to do it. As part of the preparation for the training program, suitable procedures must be established and documented. For simple collection mechanisms, these courses can be as short as one hour. I have found that hands-on, interactive training, where the group works actual data collection examples, provides the best results.

Without this training, hours of support staff time can be wasted answering the same questions repeatedly. An additional benefit of training is that it promotes a common understanding about when and how to collect the data. This reduces the risk of collecting invalid and inconsistent data.

If the right data is not collected accurately, then the objectives of the measurement program cannot be accomplished. Data analysis is pointless without good data. Therefore, establishing a good data collection plan is the cornerstone of any successful metrics program. Data collection must be:

- ***Objective:*** The same person will collect the data the same way each time.
- ***Unambiguous:*** Two different people, collecting the same measure for the same item will collect the same data.

Notes

- ***Convenient:*** Data collection must be simple enough not to disrupt the working patterns of the individual collecting the data. Therefore, data collection must become part of the process and not an extra step performed outside of the workflow.
- ***Accessible:*** In order for data to be useful and used, easy access to the data is required. This means that even if the data is collected manually on forms, it must ultimately be included in a metrics database.

There is widespread agreement that as much of the data gathering process as possible should be automated. At a minimum, standardized forms should be used for data collection, but at some point the data from these forms must be entered into a metrics database if it is to have any long-term usefulness. I have found that information that stays on forms quickly becomes buried in file drawers never to see the light of day again.

Dumping raw data and hand tallying or calculating metrics is another way to introduce human error into the metrics values. Even if the data is recorded in a simple spreadsheet, automatic sorting, data extraction, and calculation are available and should be used. Using a spreadsheet or database also increases the speed of producing the metrics over hand tallies.

Automating metrics reporting and delivery eliminates hours spent standing in front of copy machines. It also increases usability because the metrics are available on the computer instead of buried in a pile of papers on the desk. Remember, metrics are expensive.

Did u know? What is automation?

Automation can reduce the expense, while making the metrics available in a timelier manner.

12.2.1 Types of Tools

The various methods of data gathering involve the use of appropriate recording forms. These are called tools or instruments of data collection. They consist of:

- Observation schedule
- Interview guide
- Interview schedule
- Mailed questionnaire
- Rating scale
- Checklist
- Document schedule/data sheet
- Schedule for institutions

Each of the above tools is used for a specific method of data gathering: Observation schedule for observation method, interview schedule and interview guide for interviewing, questionnaire for mail survey, and so on.

The tools of data collection translate the research objectives into specific questions/items, the responses to which will provide the data required to achieve the research objectives. In order to achieve this purpose, each question/item must convey to the respondent the idea or group of ideas required by the research objectives, and each item must obtain a response which can be analysed for fulfilling the research objectives.

Notes

Information gathered through the tools provides descriptions of characteristics of individuals, institutions or other phenomena under study. It is useful for measuring the various variables pertaining to the study.

Notes The variables and their interrelationships are analysed for testing the hypothesis or for exploring the content areas set by the research objectives.

A brief description of the various tools of data collection is given below.

Observation Schedule

This is a form on which observations of an object or a phenomenon are recorded. The items to be observed are determined with reference to the nature and objectives of the study. They are grouped into appropriate categories and listed in the schedule in the order in which the observer would observe them.

The schedule must be so devised as to provide the required verifiable and quantifiable data and to avoid selective bias and misinterpretation of observed items. The units of observation must be simple, and meticulously worded so as to facilitate precise and uniform recording.

Interview Guide

This is used for non-directive and depth interviews. It does not contain a complete list of items on which information has to be elicited from a respondent: it just contains only the broad topics or areas to be covered in the interview.

Interview guide serves as a suggestive reference or prompter during interview. It aids in focussing attention on salient points relating to the study and in securing comparable data in different interviews by the same or different interviewers.

Interview Schedule and Mailed Questionnaire

Both these tools are widely used in surveys. Both are complete lists of questions on which information is elicited from the respondents. The basic difference between them lies in recording responses. While the interviewer fills out a schedule, the respondent completes a questionnaire.

Rating Scale

This is a recording form used for measuring individual's attitudes, aspirations and other psychological and behavioural aspects, and group behaviour.

Checklist

This is the simplest of all the devices. It consists of a prepared list of items pertinent to an object or a particular task. The presence or absence of each item may be indicated by checking 'yes' or 'no' or multipoint scale. The use of a checklist ensures a more complete consideration of all aspects of the object, act or task. Checklists contain terms, which the respondent understands, and which more briefly and succinctly express his views than answers to open-ended question. It is a crude device, but careful pre-test can make it less so. It is at best when used to test specific hypothesis. It may be used as an independent tool or as a part of a schedule/questionnaire.

Notes

Document Schedule/Data Sheet

This is a list of items of information to be obtained from documents, records and other materials. In order to secure measurable data, the items included in the schedule are limited to those that can be uniformly secured from a large number of case histories or other records.

Schedule for Institutions

This is used for survey of organisations like business enterprises, educational institutions, social or cultural organisations and the like. It will include various categories of data relating to their profile, functions and performance.

Caution These data are gathered from their records, annual reports and financial statements.

12.2.2 Construction of Schedules and Questionnaires

Schedule v. Questionnaire

Schedules and questionnaires are the most common instruments of data collection. These two types of tools have much in common. Both of them contain a set of questions logically related to a problem under study; both aim at eliciting responses from the respondents; in both cases the content, response structure, the wordings of questions, question sequence, etc. are the same for all respondents. Then why should they be denoted by the different terms: 'schedule' and 'questionnaires'? This is because the methods for which they are used are different. While a schedule is used as a tool for interviewing, a questionnaire is used for mailing.

This difference in usage gives rise to a subtle difference between these two recording forms. That is, the interviewer in a face-to-face interviewing fills a schedule, whereas the respondent himself fills in a questionnaire. Hence the need for using two different terms.

The tool is referred to as a schedule when it is used for interviewing; and it is called a questionnaire when it is sent to a respondent for completion and return.

The Process of Construction

The process of construction of a schedule and a questionnaire is almost same, except some minor differences in mechanics. This process is not a matter of simply listing questions that comes to researchers mind. It is a rational process involving much time, effort and thought. It consists of the following major steps:

1. ***Data need determination:*** As an interview schedule or a mailed questionnaire is an instrument for gathering data for a specific study, its construction should flow logically from the data required for the given study.
2. ***Preparation of "Dummy" tables:*** The best way to ensure the requirements of information is to develop "dummy" tables in which to display the data to be gathered.
3. ***Determination of the respondents' level:*** Who are our respondents? Are they persons with specialized knowledge relating to the problem under study? Or are they lay people? What is their level of knowledge and understanding? The choice of words and concepts depends upon the level of the respondents' knowledge.

Notes

4. ***Data gathering method decision:*** Which communication mode is most appropriate - face-to-face interview or mailing? The choice of question structure depends largely on the communication mode chosen.

5. ***Instrument drafting:*** After determining the data required for the study, first, a broad outline of the instrument may be drafted, listing the various broad categories of data. Second, the sequence of these groupings must be decided. Third, the questions to be asked under each group heading must be listed. All conceivable items relevant to the 'data need' should be compiled.

6. ***Evaluation of the draft instrument:*** In consultation with other qualified persons, the researcher must rigorously examine each question in the draft instrument.

7. ***Pre-testing:*** The revised draft must be pre-tested in order to identify the weaknesses of the instrument and to make the required further revisions to rectify them.

8. ***Specification of procedures/instructions:*** After the instruction is finalised after pre-tests, the procedures or instructions, relating to its use must be specified.

9. ***Designing the format:*** The format should be suited to the needs of the research. The instrument should be divided into different sections relating to the different aspects of the problem.

Question Construction

A survey instrument — interview schedules or questionnaire — is useful for collecting various types of information, viz., (a) factual information - facts about the respondents: sex, age, marital status, education, religion, caste or social class, income and occupation; and facts about events and circumstances, (b) psychological information such as attitudes, opinions, beliefs, and expectations, and (c) behavioural information, like social participation, and so on.

Once the information need is determined as explained in the previous topic, we can begin question construction. This involves four major decision areas. They are: (a) question relevance and content, (b) question wording, (c) response form, and (d) question order or sequence.

Question Relevance and Content

My question to be included in the instrument should pass certain tests. Is it relevant to the research objectives? Can it yield significant information for answering an investigative question? If not, it should note be included in the instrument.

Question Wording

This is a difficult task. The function of a question in a schedule/questionnaire is to elicit particular information without distortion. "Questioning people", says Oppenheim, "is more like trying to catch a particular elusive fish, by hopefully casting different kinds of bait at different depths, without knowing what goes on beneath the surface." As the meaning of words differs from person to person, the question designer should choose words which have the following characteristics:

a. Shared vocabulary

b. Uniformity of meaning

c. Exactness

Notes

d. Simplicity

e. Neutrality. The words to be used must be neutral ones, i.e., free from the distorting influence of fear, prestige, bias or emotion.

Certain other problem areas of question wording are:

a. Unwarranted assumptions,

b. Personalization,

c. Presumptions,

d. Hypothetical question,

e. Questions in embarrassing matters.

Some of the approaches to deal with this problem are:

a. to express the question in the third person; instead of asking the respondent for his views, he is asked about the views of others;

b. to use a drawing of two persons in a certain setting with 'balloons' containing speech coming from their mouths, as in a cartoon - leaving one person's balloon empty and asking the respondent to put himself in the position of that person and to fill in the missing words; and

c. to use sentence completion tests.

Response Form or Types of Questions

The third major area in question construction is the types of questions to be included in the instrument. They may be classified into open questions and closed questions. Closed questions may be dichotomous, multiple choice or declarative ones.

Types of Questions to be Avoided

The question designer should avoid the following types of questions: (a) Leading questions, (b) 'Loaded' questions, (c) Ambiguous questions, (d) Double-barreled, (e) Long questions, (t) Avoid double negative.

Question Order or Sequence

The order in which questions are arranged in a schedule/questionnaire is as important as question wording. It has two major implications. First, an appropriate sequence can ease the respondent's task in answering. Second, the sequence can either create or avoid biases due to context effects, i.e., the effects of preceding questions on the response to later questions.

Mechanics of the Schedule and Questionnaire

In addition to question wording and question construction, the mechanics of the form should also be considered in the design of a schedule/questionnaire. The mechanics of the form has several aspects: items of the form, instruction, pre-coding, sectionalisation, spacing, paper, printing, margins, etc.

Notes

Items of the form: The following items are mandatory for schedules and questionnaires:

1. The name of the organization collecting the data should appear at the top of front -page. The name of the sponsor, of the study, if any should also be shown.
2. The title of the study should appear in large print next to the name of the organization on the first page. Below this title, the title of the tool - e.g., 'Schedule for consumers; - may be noted.
3. The confidentialness of the data should be made cleat.
4. A place for writing the date of filling in the form should be provided.
5. A serial number to each copy of the tool may be assigned.
6. The pages of the instrument should be numbered.

Instructions: In the face sheet below the title of the questionnaire, a brief statement of the objective of the study, the confidentialness of the data, and instructions relating to answering the questions may be provided.

Pre-coding: Items in the tool should be pre-coded so as to facilitate transcription of data.

Sectionalisation: There should be a separate section for each topical area.

Spacing: For each open-ended question, an adequate space should be provided for answer. There should, indeed more space than seems necessary, for some interviewers/respondents may write in a large script for legibility. Moreover, liberal spacing is a stimulus for the questionnaire respondent to write more fully. Even short-answer questions should be spaced, so that the interviewer/respondent will not easily confuse the line, from which he is reading.

Paper: The paper used for mimeographing/printing should be of good quality.

Printing: Mailed questionnaire should necessarily be printed in order to make it attractive and to minimise the postal expenditure.

Margins: One inch margin on the left side of the sheet and one-half inch margin on other sides may be provided. If the instrument is to be bound, left-side margin should conform to the type of binding used.

Indentation: This is required for 'yes' or 'no' questions. If the respondent's answer is 'yes', then a series of questions is offered. If the answer is 'no' a different series of questions is offered.

Note of Thanks: A final note or comment of thanks for the cooperation of the respondent should be included at the end of the instrument.

Caution Question designing remains primarily a matter of common sense and experience and of avoiding known pitfalls, as there arc no hard and fast rules relating to it. Hence alternative versions of questions must be rigorously tested in pre-tests. Test-revision-retests play a crucial role in questionnaire construction.

12.2.3 Measurement Scales and Indices

Scales are devised for measuring variables in social science research. During the past few decades thousands of scales have been designed by researchers in sociology, psychology, education, psychiatry, ethics, behavioural science, economics, administration and other fields.

Notes

Task Indices and scales are often used interchangeably to refer to all sorts of measures, absolute or relative, single or composite, simple or elaborate. "Scaling" refers to the procedure by which numbers or scores assigned to the various degrees of opinions, attitude and other concepts. Explain

12.2.4 Pilot Studies and Pre-tests

Pilot Study

It is difficult to plan a major study or project without adequate knowledge of its subject matter, the population it is to cover, their level of knowledge and understanding and the like. What are the issues involved? What are the concepts associated with the subject matter? How can they be operationalised? What method of study is appropriate? How long the study will take? How much money it will cost? These and other related questions call for a good deal of knowledge of the subject matter of the study and its dimensions. In order to gain such pre-knowledge of the subject matter of an extensive study, a preliminary investigation is conducted. This is called a pilot study.

Pre-test

While a pilot study is a full-fledged miniature study of a problem, pre-test is a trial test of a specific aspect of the study such as method of data collection or data collection instrument - interview schedule, mailed questionnaire or measurement scale.

Need for Pre-testing

An instrument of data collection is designed with reference to the data requirements of the study. But it cannot be perfected purely on the basis of a critical scrutiny by the designer and other researchers. It should he empirically tested. As emphatically pointed by Goode and Hatt, "no amount of thinking, no matter how logical the mind or brilliant the insight, is likely to take the place of careful empirical checking". Hence pre-testing of a draft instrument is indispensable. Pre-testing-means trial administration of the instrument to a sample of respondents before finalising it.

Purposes of Pre-testing

Pre-testing has several purposes: (1) to test whether the instrument would elicit responses required to achieve the research objectives, (2) to test whether the content of the instrument is relevant and adequate, (3) to test whether wording of questions is clear and suited to the understanding of the respondents, (4) to test the other qualitative aspects of the instrument like question structure and question sequence, and (5) to develop appropriate procedure for administering the instrument with reference to field conditions.

Self Assessment

Fill in the blanks:

5. Two different people, collecting the same measure for the same item will collect the …….....

Notes

6. The …….. must be so devised as to provide the required verifiable and quantifiable data and to avoid selective bias and misinterpretation of observed items.
7. Data collection methods for impact ……. vary along a continuum.
8. After the instruction is finalised after pre-tests, the …….. or instructions, relating to its use must be specified.
9. The process of construction of a schedule and a is almost same, except some minor differences in mechanics.
10. Dumping raw data and hand tallying or calculating …… is another way to introduce human error into the metrics values.

12.3 Visualizing Progress by Progress Report

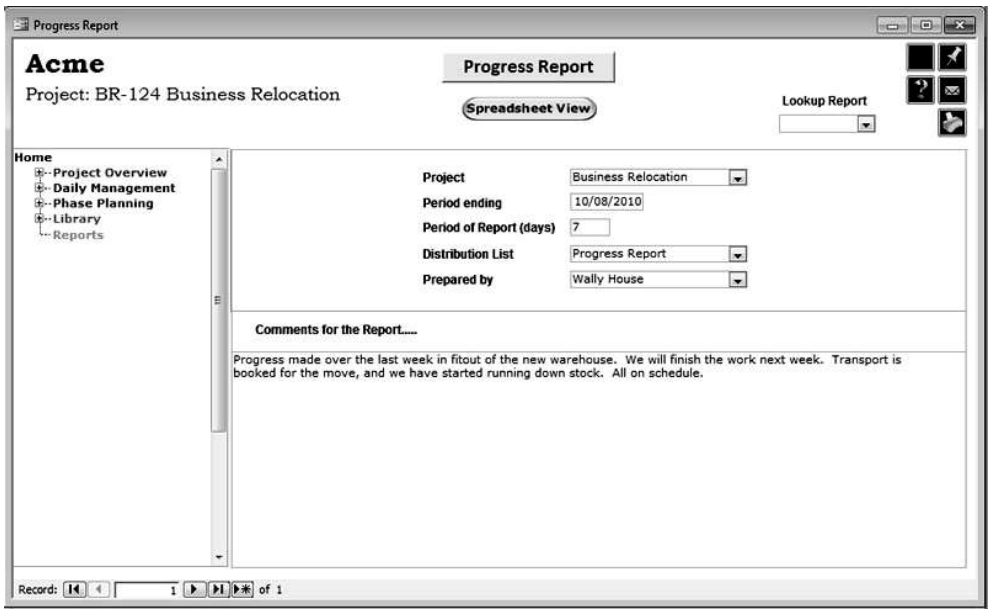
To generate a project status report for the project, the Project Manager need only add a few comments and the project management report will be produced automatically showing:

- Milestones in the last period, and if they are completed
- Milestones due next period
- Action items not completed by their due date
- Action items due to be completed in the next period
- The commentary prepared by the Project Manager.

The report can be e-mailed directly to a set distribution list (using Microsoft Outlook). The period can be set by the project manager.

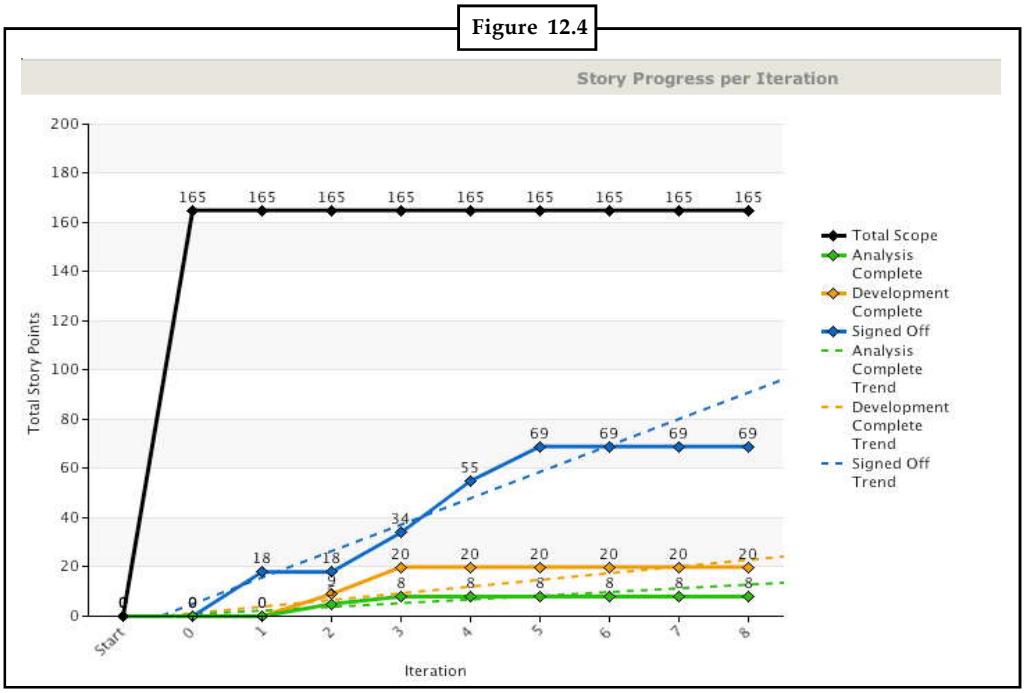
The value is not so much in the report. The value comes from ensuring that at least once a reporting period, the team is forced to review due and outstanding items. Usually this will force them to focus on completing actions and milestones so they do not appear in the report. This function acts as a project status report template or a piece of period status reporting software.

Figure 12.3



Notes

Figure 12.4



Did u know? What are the benefits of visualizing progress?

- Easy to see information because it is in a single consistent format
- Project staff are forced to review progress on a period basis
- Little work involved to generate a report
- Periods can be set to match company requirements.

Cost Monitoring

Having accurate project estimates and a robust project budget is necessary to deliver within the project budget. Both Estimate Costs and Determine Budget are project planning processes. Without keeping an eye on the actual costs while the project is being implemented, the project will most likely never be delivered on-budget. There are several techniques as stated in the PMBOK used to monitor and control the cost of a project:

- Earned Value Management
- Forecasting
- To-Complete Performance Index (TCPI)
- Variance Analysis
- Performance Reviews

Earned Value Management or Analysis

Earned Value Management (EVM) is a mathematical method by which you can measure the actual performance of a project. You will use EVM to monitor your project in terms of schedule and cost. For example, suppose your project is on track as per the schedule. Through EVM, you

Notes

will be able to understand whether the project is also on-budget. If it is not, you can take corrective action. EVM principles can be extended to Forecasting, TCPI, and Variance Analysis. EVM is an input to project performance reviews. Therefore, it is critical for you to understand EVM formulae so that you can use them as inputs to other cost control techniques.

Earned Value Management (EVM) is a well known project management technique which measures the integration of technical performance, cost and schedule against planned performance within a given project. The result is a simple set of metrics providing early warnings of performance issues, allowing for timely and appropriate adjustments. In addition, EVM improves the definition of project scope, and provides valuable metrics for communicating progress to stakeholders. The information generated helps to keep the project team focused on making progress.

Forecasting

EVM provides formulae to forecast the future performance of a project. The forecast is based on the current actual performance. As a project manager, having the ability to tell whether your project will be delivered on-time and on-budget is critical. Let's take an example to understand this.

Suppose you have completed 25% of your project. As per the schedule you are on track. However, after completing 50% of the project, you realize your project is delayed. By using forecasting formulae you can determine the degree of delay. This will also enable you to investigate the cause of delay and the corrective action, such as Crashing, required to get the project back on track. In addition, to the schedule delay you can use EVM Forecasting formulae to determine the actual cost of the project on completion and take measures to rectify any anomaly before it is too late.

To-Complete Performance Index (TCPI)

If the project is delayed or over-budget, you can use TCPI to determine the project performance required to complete the project as budgeted or estimated. TCPI also leverages the EVM formulae.

Variance Analysis

Variance analysis is the comparison of expected project performance to the actual cost performance. This analysis helps you understand the causes of variance, if any. Preventative and corrective actions are determined based on the variance analysis.

Performance Review

Performance reviews in projects are required to check the health of a project. This usually involves Cost and Schedule as the main parameters to assess.

Notes

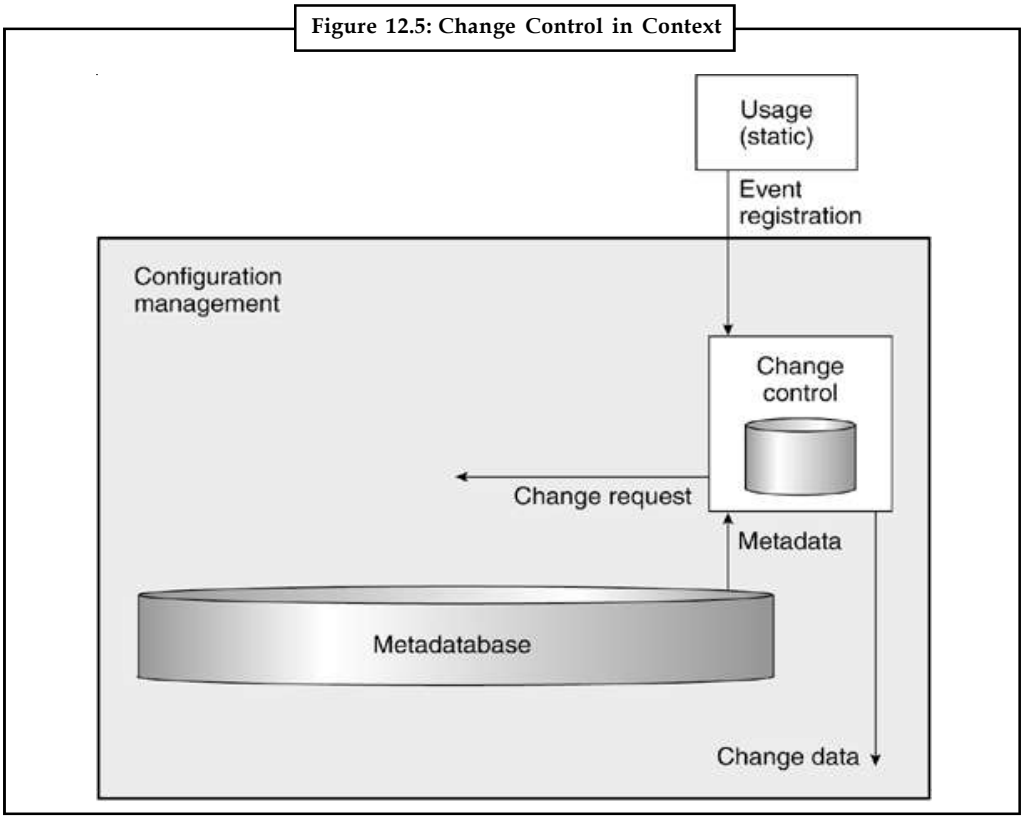
Notes However, other parameters, such as Scope, Quality, Team Morale, may be used. Reviews may include the client, Product Owner, other Project Managers or ScrumMasters.

12.4 Change Control

When developing and maintaining a product, changes are inevitable. People make mistakes, customers need changes, and the environment in which the product operates evolves. In addition, people constantly develop their knowledge of the problem and their ability to solve it. In software development, it's generally said that the solution of a problem will create new problems. In other words, we get wiser all the time.

The purpose of change control is to be fully in control of all change requests for a product and of all implemented changes. For any configuration item, it must be possible to identify changes in it relative to its predecessor. Any change should be traceable to the item where the change was implemented. Figure 12.5 shows how change control affects and is affected by its environment.

Figure 12.5: Change Control in Context



Inputs

Change control is initiated by an event. An event may also be called a wish for modification but need not be expressed as a clearly formulated wish. In this context, an event is any observation of something surprising, unexpected, inconvenient, or directly wrong during usage of the configuration item. It may, for instance, be:

- A wrong formulation, caught during the review of a document.
- A coding mistake found during a walk-through of a piece of source code.

Notes

- An enhancement request arising from a new idea from the customer during work on the project.
- A mistake found in the integration test.
- A wish to expand or enhance the finished product, arising once the product is in operation.
- An inquiry to a helpdesk about a problem in connection with usage of a system.
- A change required in the code because of an upgrade to a new version of the middleware supporting the system, which may not be backward compatible.

An event should be documented in an event registration, which is the input to the change control activity. Some changes, such as those due to a review, can be foreseen and planned, while those due to, for instance, a new customer request cannot.

Outputs

The result of change control is documented events and change requests derived from these events. Both should be securely maintained, as in a database, so that relationships between change requests and configuration items can be reliably maintained. Event registration and change requests may be put under configuration management, but this happens rarely, except where configuration management has to be very formal.

12.4.1 Change Control Activities

A change process is a miniature development project in itself. An event registration should have a written and controlled life cycle, consisting roughly of the phases described in Table 12.2. Each phase should be described in detail, stating the responsibility and specific actions in the company. It may be necessary for a company to describe different kinds of life cycles, depending on the types of events to be handled.

Table 12.2: Overview of Change Control Phases

Phase	Description
Creation of the event registration.	The event registration is created, and the event is described.
Analysis of the event registration.	Configuration item(s) affected by possible changes are determined, and the extensiveness of these changes is estimated.
Rejection or acceptance of the event registration.	If the event registration is accepted, a change request is created for each configuration item affected.
The change request initiates a new configuration item.	A new configuration item is identified and created, and the change is implemented. In the course of accepting the new item and placing it in storage, feedback is given to the configuration control board.
Closing of the change request.	The change request can be closed when the change has been implemented and accepted.
Closing of the event registration.	The event registration can be closed when all corresponding change requests are closed.

Quite often the change request is joined with the event registration, so no independent change requests are created.

Notes This is not a very good idea, unless it remains possible to extract statistics and status information on individual change requests as well as on the event. This is especially true if an event causes changes in several configuration items, which is often the case.

Notes

12.4.2 Usage of Metadata

When performing the change process, metadata is used for analytical purposes. This may be in the form of reports or a direct search in the database or the databases where metadata is maintained. Trace information is often used—for instance, to determine in which configuration item changes are required due to an event. Also information about variants or branches belonging to a configuration item is used to determine if a change has effects in several places.

Finally metadata may be used to determine if a configuration item has other outstanding event registrations, such as whether other changes are in the process of being implemented or are awaiting a decision about implementation.

Consequence Analysis

When analyzing an event, you must consider the cost of implementing changes. This is not always a simple matter. The following checklists, adapted from a list by Karl Wiegers, may help in analyzing the effects of a proposed change. The lists are not exhaustive and are meant only as inspiration.

Identify

- All requirements affected by or in conflict with the proposed change.
- The consequences of not introducing the proposed change.
- Possible adverse effects and other risks connected with implementation.
- How much of what has already been invested in the product will be lost if the proposed change is implemented—or if it is not.

Check if the proposed change

- Has an effect on nonfunctional requirements, such as performance requirements (ISO 9126, a standard for quality characteristics, defines six characteristics: functional, performance, availability, usability, maintainability, and portability. The latter five are typically referred to as nonfunctional.)
- May be introduced with known technology and available resources.
- Will cause unacceptable resource requirements in development or test.
- Will entail a higher unit price.
- Will affect marketing, production, services, or support.

Follow-on effects may be additions, changes, or removals in

- user interfaces or reports, internal or external interfaces, or data storage,
- designed objects, source code, build scripts, include files,
- test plans and test specifications,
- help texts, user manuals, training material, or other user documentation,
- project plan, quality plan, configuration management plan, and other plans,
- other systems, applications, libraries, or hardware components,

Roles

The configuration (or change) control board (CCB) is responsible for change control. A configuration control board may consist of a single person, such as the author or developer

Notes

when a document or a piece of code is first written, or an agile team working in close contact with users and sponsors, if work can be performed in an informal way without bureaucracy and heaps of paper. It may also—and will typically, for most important configuration items—consist of a number of people, such as the project manager, a customer representative, and the person responsible for quality assurance.

Process Descriptions

The methods, conventions, and procedures necessary for carrying out the activities in change control may be;

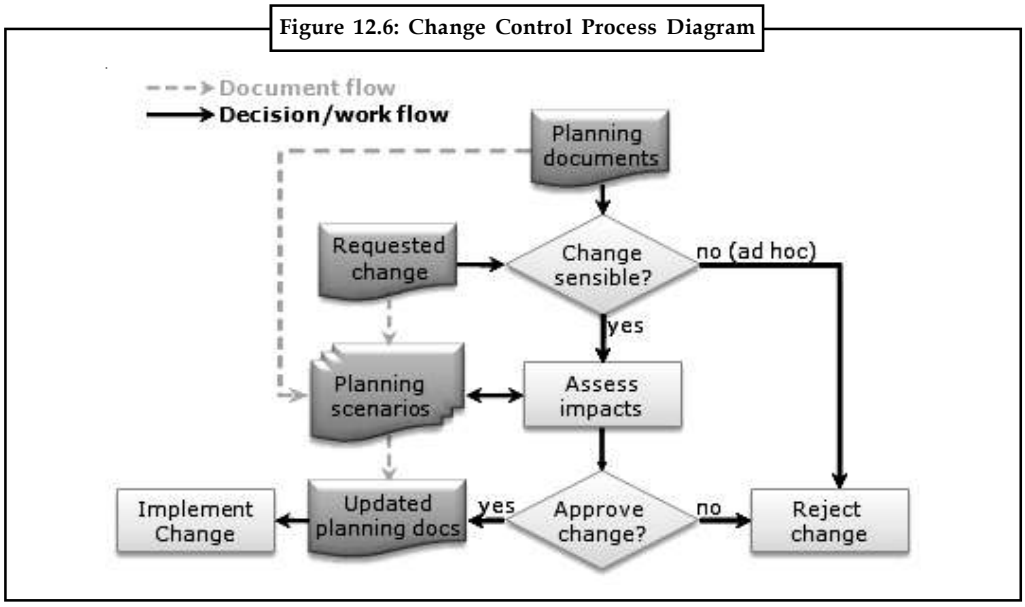
- description of the change control process structure,
- procedures in the life cycles of events and changes,
- convention(s) for forming different types of configuration control boards,
- definition of responsibility for each type of configuration control board,
- template(s) for event registration,
- template(s) for change request.

Connection with Other Activities

Change control is clearly delimited from other activities in configuration management, though all activities may be implemented in the same tool in an automated system. Whether change control is considered a configuration management activity may differ from company to company. Certainly it is tightly coupled with project management, product management, and quality assurance, and in some cases is considered part of quality assurance or test activities. Still, when defining and distributing responsibilities, it's important to keep the boundaries clear, so change control is part of configuration management and nothing else.

Example: Figure 12.6 shows an example of a process diagram for change control. A number of processes are depicted in the diagram as boxes with input and output sections (e.g., "Evaluation of event registration"). All these processes must be defined and, preferably, described.

Figure 12.6: Change Control Process Diagram



Notes

12.5 Purpose of Controls

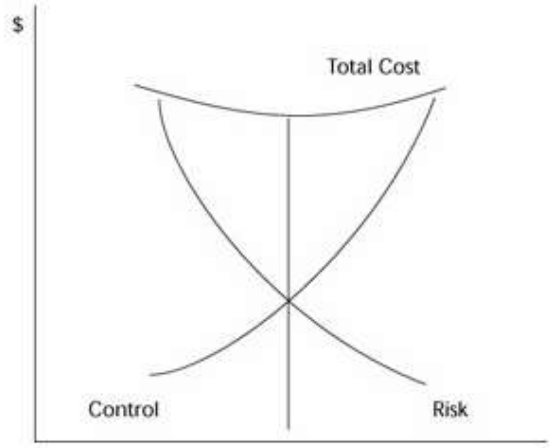
Controls are actions taken as a result of reports. When implemented, controls are designed to bring actual project status back into conformance with the project plan. These reports are designed to support control activities by drawing attention to certain aspects or characteristics of the project, such as planned versus actual schedule, trends in the schedule, and actual versus planned resource use.

We typically track performance levels, costs, and time schedules. There are three reasons to use reports in your project.

1. ***To track progress:*** The project manager will want to use a periodic (at least biweekly, but weekly is best) reporting system that identifies the status of every activity scheduled for work since the last progress report. These reports summarize progress for the current period as well as the cumulative progress for the entire project.
2. ***To detect variance from plan:*** Variance reports are of particular importance to management. They are simple and intuitive, and they give managers an excellent tool by which to quickly assess the health of a project. To detect variance, the project manager needs to compare planned performance to actual performance.
3. ***To take corrective action:*** To take corrective action, it is necessary to know where the problem is and to have that information in time to do something about it. Once there is a significant variance from plan, the next step is to determine whether corrective action is needed and then act appropriately.

12.6 Balancing the Control System

Figure 12.7: The Total Cost of Control and Risk



It is very easy to get carried away with controls and reports. The more controls that are put in place, the lower the project risk, and the less likely it will be for the project to get in trouble. As figure 12.7 shows, however, there is a point of diminishing returns. Cost aside, there is another impact to consider. To comply with the project controls, project team members will have to spend time preparing and defending progress reports. This subtracts from the time spent doing project work.

The project manager needs to strike a balance between the extent of the control system and the risk of unfavorable outcomes. Figure 12.7 shows the relationship between risk and control. Conceptually, there is a balance point that minimizes the total cost exposure for having chosen a particular level of control.

Notes

Did u know? Control versus Quality

Control and quality are positively correlated with one another. If we do not take steps to control the product and the process, we will not enjoy the benefits that quality brings to the equation.

12.7 Progress Reporting System

To make sure that the project proceeds according to plan, you need to establish a reporting system that keeps you informed of the many variables that describe how the project is proceeding as compared to the plan.

A reporting system has the following characteristics:

- Provides timely, complete, and accurate status information.
- Doesn't add so much overhead time as to be counterproductive.
- Is readily acceptable to the project team and senior management.
- Warns of pending problems in time to take action.
- Is easily understood by those who have a need to know.

Most project management software tools allow you to customize their standard reports to meet even the most specific needs.

12.7.1 Types of Project Status Reports

There are five types of project status reports:

1. ***Current Period Reports:*** These reports cover only the most recently completed period. They report progress on those activities that were open or scheduled for work during the period. Reports might highlight activities completed and variance between scheduled and actual completion dates.
2. ***Cumulative Reports:*** These reports contain the history of the project from the beginning to the end of the current report period. They are more informative than the current period reports because they show trends in project progress.
3. ***Exception reports:*** Exception reports report variances from plan. These reports are typically designed for senior management to read and interpret quickly.
4. ***Stoplight reports:*** Stoplight reports are a variation that can be used on any of the previous report types. Here stickers of the different colors are put on the top right of the first page of the project status report. The purpose of the different color of stickers are as follows:
   (a) *Green sticker:* It means the project is on schedule and everything seems to be moving as planned. This sticker will signal to senior managers that everything is progressing according to plan, and they need not even read the attached report.
   (b) *Yellow sticker:* It means that the project has encountered a problem or there is a schedule slippage. That is a signal to upper management that the project is not moving along as scheduled but that you have a get-well plan in place.
   (c) *Red stickers:* It means that a project is out of control. Red reports are to be avoided at all costs because they mean that the project has encountered a problem, and you don't have a get-well plan or even a recommendation for upper management.
5. ***Variance Reports:*** Variance reports do exactly what their name suggests—they report differences between what was planned and what actually happened. The report has three columns:

Notes

- The planned number
- The actual number

Typical variance reports are snapshots in time (the current period) of the status of an entity being tracked. Most variance reports do not include data points that report how the project reached that status.

Task Project variance reports can be used to report project as well as activity variances. Explain in group of four.

12.7.2 How and What Information to Update

As input to each of these report types, activity managers and the project manager must report the progress made on all of those activities that were open for work during the period of time covered by the status report. Recall that your planning estimates of activity duration and cost were based on little or no information. Now that you have completed some work on the activity, you should be able to provide a better estimate of the duration and cost exposure. This reflects itself in a re-estimate of the work remaining to complete the activity. That update information should also be provided.

The following is a list of what should actually be reported:

1. Determine a set period of time and day of week by which by which all updated information is to be submitted.
2. Report actual work accomplished during this period.
3. Record historical and re-estimate remaining (in-progress work only).
4. Actual start and finish dates of activities started or completed during the report period.
5. Record days of duration accomplished of so far working re-estimated duration as reflected in the time-to-completion number.
6. Report resource effort (hours/day) spent and remaining (in-progress work only).

Did u know? What is Frequency of Gathering and Reporting Project Progress?

A logical frequency for reporting project progress is once a week, usually on Friday afternoon. There are some projects, such as refurbishing a large jet airliner, where progress is recorded after each shift, three times a day.

12.7.3 Variances

Variances are deviations from plan. Think of a variance as the difference between what was planned and what actually occurred. There are two types of variances:

Positive Variances

Positive variances are deviations from plan that indicate that an ahead-of-schedule situation has occurred or that an actual cost was less than a planned cost. This type of variance is good news to the project manager.

Notes

Positive variances bring their own set of problems, which can be as serious as negative variances. Positive variances can allow for rescheduling to bring the project to completion early, under budget, or both. Resources can be reallocated from ahead-of-schedule projects to behind-schedule projects.

Negative Variances

Negative variances are deviations from plan that indicate that a behind-schedule situation has occurred or that an actual cost was greater than a planned cost.

Negative variances, just like positive variances, are not necessarily bad news. For example, you might have overspent because you accomplished more work during the report period than was planned. But in overspending during this period, you could have accomplished the work at less cost than was originally planned. You can't tell by looking at the variance report.

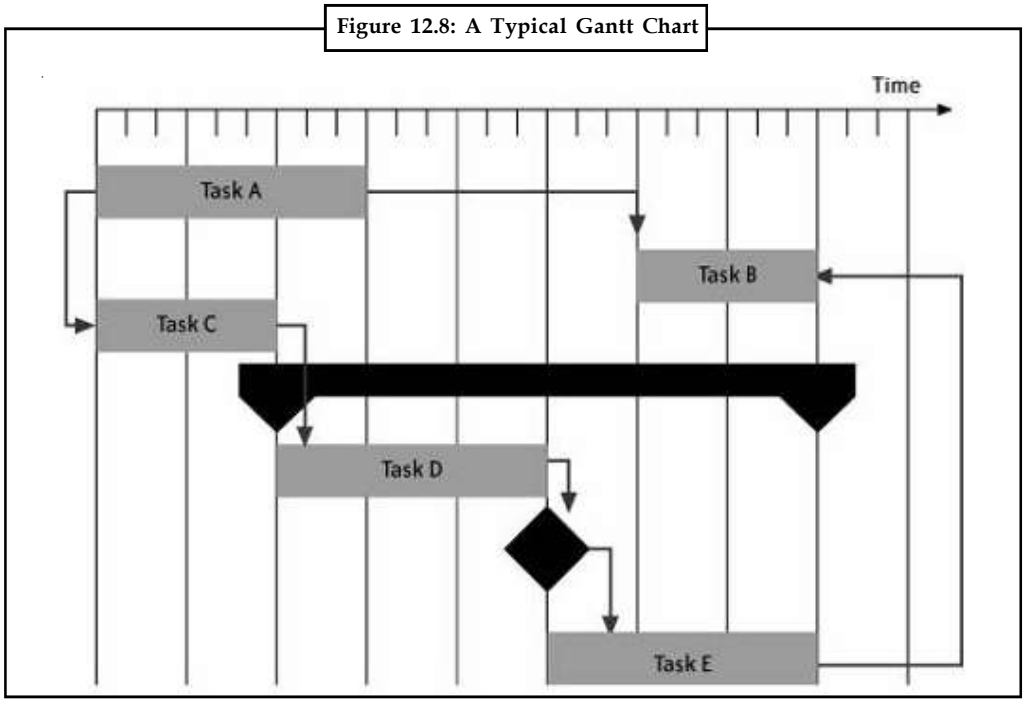
In most cases, negative time variances affect project completion only if they are associated with critical path activities or if the schedule slippage on non-critical path activities exceeds the activity's total float. Variances use up the float time for that activity; more serious ones will cause a change in the critical path.

Negative cost variances can result from uncontrollable factors such as cost increases from suppliers or unexpected equipment malfunctions. Some negative variances can result from inefficiencies or error.

12.8 Graphical Reporting Tools

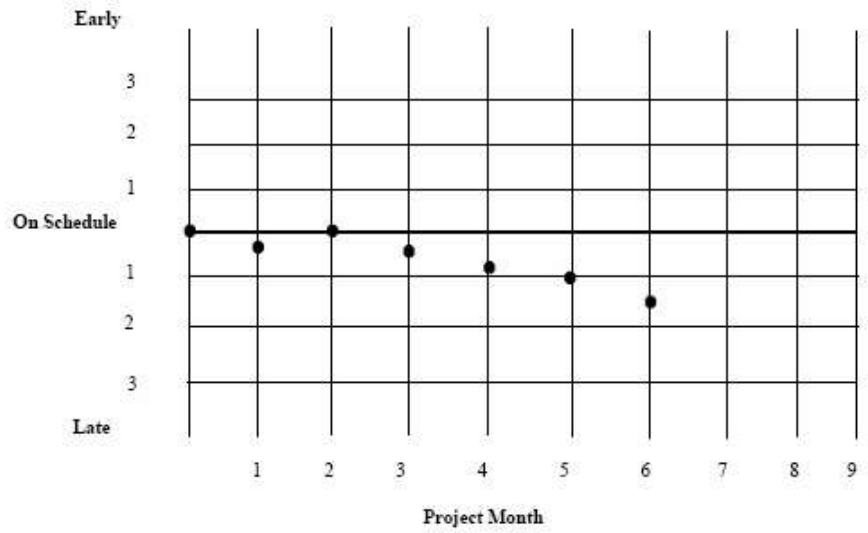
Usually senior managers have only a few minutes of uninterrupted time to digest your report. Having to read several pages only to find out that the project is on schedule is frustrating and a waste of valuable time. For this it is better to present them the report in the form of graphics. You can use the Gantt charts (See Figure 12.8) and Milestone Trend Charts (See Figure 12.9) for this purpose.

Figure 12.8: A Typical Gantt Chart



Notes

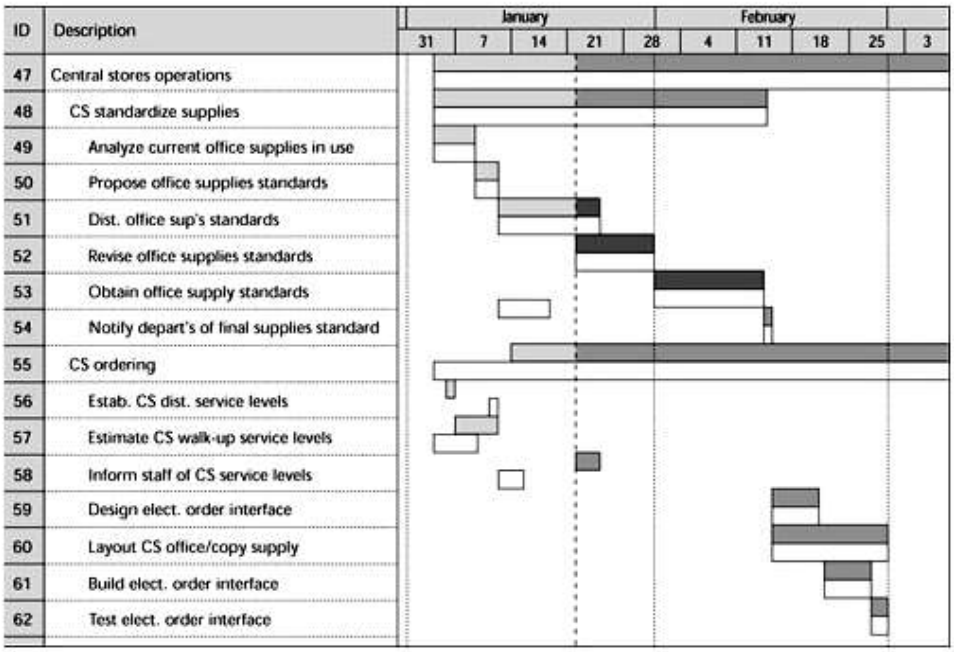
Figure 12.9: A Typical Milestone Trend Chart



12.8.1 Using the WBS to Report Project Status

Because the Work Breakdown Structure (WBS) shows the hierarchical structure of the work to be done, it can be used for status reporting, too. In its simplest form, each activity box can be shaded to reflect completion percentages. As lower-level activities are completed, the summary activities above them can be shaded to represent percent complete data. Senior managers will appreciate knowing that major parts of the project are complete. Unfortunately, the WBS does not contain scheduling or sequencing information. To the extent that this adds to the value of the report, narrative data or brief tabular data might be added to the report. Figure 12.10 shows an example status report using the WBS.

Figure 12.10: Status Reporting with the WBS



Notes

Although this report is rather intuitive, it does not contain much detail. It would have to be accompanied by an explanatory note with schedule and cost detail.

Task List down reasons that how the graphical reporting system is better than the written one for senior managers?

12.9 Managing Project Status Meetings

To keep close track of progress on the project, the project manager needs to have information from his or her team on a timely basis. This information will be given during a project status meeting. At a minimum, you need to have a status meeting at least once a week. On some of the major projects on which we've worked, daily status meetings were the norm for the first few weeks, and then as the need for daily information wasn't as critical, we switched to twice a week and finally to weekly status.

12.9.1 What is their Format?

While the format of the status review meetings should be flexible, as project needs dictate, certain items are part of every status meeting. We recommend that you proceed in a top-down fashion:

1. The project champion reports any changes that may have a bearing on the future of the project.
2. The customer reports any changes that may have a bearing on the future of the project.
3. The project manager reports on the overall health of the project and the impact of earlier problems, changes, and corrective actions as they impact at the project level.
4. Activity managers report on the health of activities open or scheduled open for work since the last status meeting.
5. Activity managers of future activities report on any changes since the last meeting that might impact project status.
6. The project manager reviews the status of open problems from the last status meeting.
7. Attendees identify new problems and assign responsibility for their resolution (the only discussion allowed here is for clarification purposes).
8. The project champion, customer, or project manager, as appropriate, offers closing comments.
9. The project manager announces the time and place of the next meeting and adjourns the meeting.

Minutes are part of the formal project documentation and are taken at each meeting, circulated for comment, revised as appropriate, distributed, and filed in the project notebook. Because there is little discussion, the minutes contain any handouts from the meeting and list the items assigned for the next meeting.

Notes The minutes should also contain the list of attendees a summary of comments made, and assigned responsibilities.

12.9.2 Problem Management Meetings

Problem management meetings provide an oversight function to identify, monitor, and resolve problems that arise during the life of a project. This is an important function in the management of projects, especially large projects.

Self Assessment

Notes

Fill in the blanks:

11. …… are actions taken as a result of reports.
12. The more controls that are put in place, the …… the project risk.
13. To make sure that the project proceeds according to plan, you need to establish a …… system.
14. …… reports cover only the most recently completed period.
15. …… are deviations from plan.

Caselet

Controlling Fiscal Deficit is Big Challenge for Govt.: FICCI

Our Bureau

New Delhi, Jan. 25

The Government would face a stiff challenge to bring down the fiscal deficit, even as inflation would continue to remain a concern in 2011, according to FICCI's Economic Outlook Survey.

The industry body's survey that interviewed economists revealed that bringing down fiscal deficit to the project levels in 2011-12 and 2012-13 would be a challenging task as the cushion of windfall gains from 3G/BWA auction will not be there in the next two years. The situation becomes even more taxing as expenditure is expected to go up steeply on account of rising crude oil prices and the implementation of the proposed National Food Security Act.

The survey thus pegs the handling of the fiscal situation as a priority for the country in the next two years. The panel of economists surveyed believe that deficit for the 2010-11 financial year, fiscal deficit is likely to be under 5.5 per cent of the GDP. Large resources raised by the Government from 3G/BWA auction, proceeds from disinvestment, better tax collections, and decontrol of petrol prices lowering the subsidy burden are the main factors cited for this expected improvement of the fiscal situation.

The survey also cautioned that if the Government wants to maintain fiscal discipline in the coming years then it must carry forward the tax reforms agenda as well as complete the reforms initiated in the pricing of petroleum products.

Results of the FICCI Economic Outlook Survey show that inflation would continue to remain a cause for concern throughout 2011. The surveyed economists have pointed out that a structural element – of rising food prices on account of continuously increasing food demand - has been built into inflation and dealing with this situation requires more effective supply side measures.

The survey suggests that to bring food inflation under control, the Government should focus on enhancing overall agriculture production and agri-productivity levels.

FICCI's Economic Outlook Survey also predicts that GDP growth of 2010-11 is likely to be at 8.7 per cent, agriculture and allied activities will grow at 4.4 per cent while industrial growth is likely to be around 8.6 per cent. The survey pegs the fiscal deficit at around 5.3 per cent of the GDP and the WPI inflation rate to be at 7 per cent.

Notes

12.10 Summary

- Controls are actions taken as a result of reports. When implemented, controls are designed to bring actual project status back into conformance with the project plan.
- There are three reasons to use reports in your project: (i) To track progress, (ii) To detect variance from plan, and (iii) To take corrective action.
- The more controls that are put in place, the lower the project risk, and the less likely it will be for the project to get in trouble.
- The project manager needs to strike a balance between the extent of the control system and the risk of unfavorable outcomes. Control and quality are positively correlated with one another.
- To make sure that the project proceeds according to plan, you need to establish a reporting system that keeps you informed of the many variables that describe how the project is proceeding as compared to the plan.

12.11 Keywords

Controls: Controls are actions taken as a result of reports.

Current Period Reports: These reports cover only the most recently completed period.

Cumulative Reports: These reports contain the history of the project from the beginning to the end of the current report period.

Exception Reports: Exception reports report variances from plan.

Variance Reports: Variance reports, report differences between what was planned and what actually happened.

Variances: Variances are deviations from plan.

12.12 Review Questions

1. What are controls in project management?
2. What are the reasons to use reports in your project?
3. Write a short note on "balancing the control system."
4. What are the characteristics of a reporting system?
5. Discuss the various types of project status reports.
6. What is the purpose of different colors of stickers in spotlight reports?
7. What are variance reports?
8. What should be the frequency of gathering and reporting project progress?
9. Differentiate between the positive and negative variances.
10. Why do we need Graphical Reporting Tools for reporting system?
11. How can WBS be used to report project status?

Notes

Answers: Self Assessment

1. Project Management Plan
2. EC-ISD
3. PMIS
4. Frameworks
5. same data
6. schedule
7. evaluation
8. procedures
9. questionnaire
10. metrics
11. Controls
12. Lower
13. reporting
14. Current period
15. Variations

12.13 Further Readings

Books

Robert K. Wyzocki, Rudd McGary, *Effective Project Management*, WILEY-Dreamtech India Pvt. Ltd., 2000.

Roger S. Pressman, *Software Engineering a Practitioner's Approach*, Fourth Edition, McGraw Hill International, 2000.

Lan Somerville, *Software Engineering*, Fifth Edition, Addison Wesley Publications, 1996.

Bob Hughes, Mike Cotterell, *Software and Project Management*, Tata McGraw-Hill Publishing Company Limited, Third Edition, 2004.

Walker Royce, *Software Project Management,* Addison-Wesley, 1998.

Online links

http://itl.nist.gov/div898/handbook/pmc/pmc.htm

http://www.workflow-research.de/Tutorials/AMCIS2004/MIZU-BPM-AMCIS2004-Section5.pdf

Notes

Unit 13: Software Quality

CONTENTS

Objectives

Introduction

13.1 Software Quality

13.2 ISO-9126

13.3 Measuring Software

13.4 Product Quality Management

13.4.1 Process Quality Management

13.5 External Standards

13.6 Summary

13.7 Keywords

13.8 Review Questions

13.9 Further Readings

Objectives

After studying this unit, you will be able to:

- Explain software quality
- Discuss ISO-9126
- Describe Software Measures
- Differentiate between Product and Process Quality
- Explain External Standards

Introduction

Software Quality Assurance (SQA) as a 'planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established technical requirements'. SQA does this by checking that:

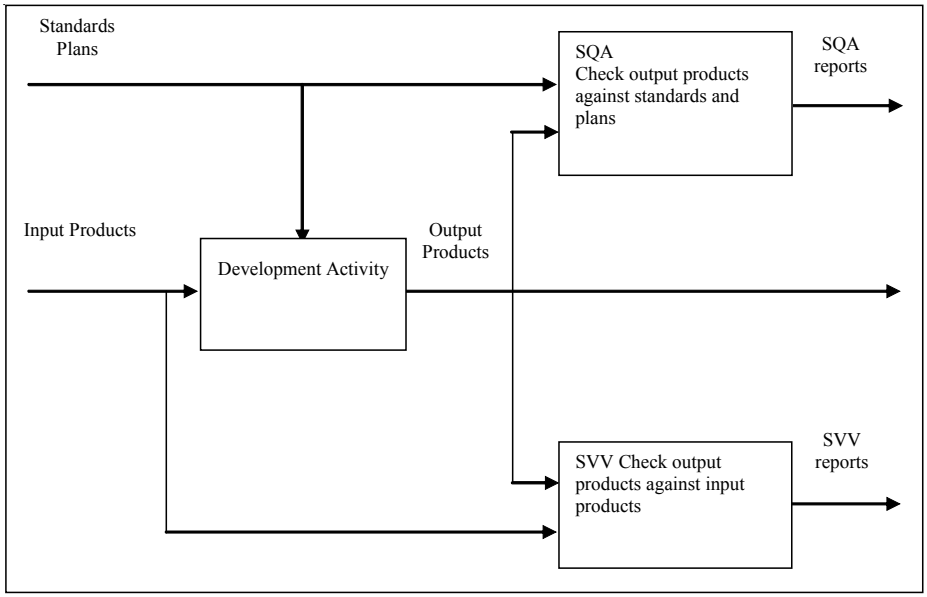
- plans are defined according to standards;
- procedures are performed according to plans;
- products are implemented according to standards.

A procedure defines how an activity will be conducted. Procedures are defined in plans, such as a Software Configuration Management Plan. A product is a deliverable to a customer. Software products are code, user manuals and technical documents, such as an Architectural Design Document. Examples of product standards are design and coding standards, and the standard document templates.

Notes

SQA is not the only checking activity in a project. Whereas SQA checks procedures against plans and output products against standards, Software Verification and Validation (SVV) checks output products against input products. Figure 13.1 illustrates the difference.

Figure 13.1: Differences between SQA and SVV



13.1 Software Quality

Formal SQA Definition

The correct definition of Software Quality Assurance goes something like:

The function of software quality that assures that the standards, processes, and procedures are appropriate for the project and are correctly implemented.

The problem with this, and similar, definitions for commercial SQA practitioners are:

- It tells us little about what SQA is other than repeating the definition. That is, it uses the defined terms "assures" and "software".
- It doesn't provide a scope for someone responsible for Software Quality Assurance.
- It doesn't address the role, or relationship, with Software Testing.
- In its pure form under which a separate 'audit' style group needs to be established it is difficult to apply to a small development environment.

There are many definitions of these Software Quality Attributes but a common one is the FURPS+ model which was developed by Robert Grady at Hewlett Packard.

Under the FURPS, the following characteristics are identified:

1. ***Functionality:*** The F in the FURPS+ acronym represents all the system-wide functional requirements that we would expect to see described. These usually represent the main product features that are familiar within the business domain of the solution being developed. For example, order processing is very natural for someone to describe if you

Notes

are developing an order processing system. The functional requirements can also be very technically oriented. Functional requirements that you may consider to be also architecturally significant system-wide functional requirements may include auditing, licensing, localization, mail, online help, printing, reporting, security, system management, or workflow. Each of these may represent functionality of the system being developed and they are each a system-wide functional requirement.

2. ***Usability:*** Usability includes looking at, capturing, and stating requirements based around user interface issues, things such as accessibility, interface aesthetics, and consistency within the user interface.

3. ***Reliability:*** Reliability includes aspects such as availability, accuracy, and recoverability, for example, computations, or recoverability of the system from shut-down failure.

4. ***Performance:*** Performance involves things such as throughput of information through the system, system response time (which also relates to usability), recovery time, and startup time.

5. ***Supportability:*** Finally, we tend to include a section called Supportability, where we specify a number of other requirements such as testability, adaptability, maintainability, compatibility, configurability, installability, scalability, localizability, and so on.

Why don't we apply these proven Quality Management Processes to Software?

The definition still refers back to the traditional manufacturing QA world. There are, however, some notable differences between software and a manufactured product.

These differences all stem from the fact that the manufactured product is physical and can be seen whereas the software product is not visible. Therefore its function, benefit and costs are not as easily measured.

The following differences highlight some of the issues in taking the manufacturing QA model and applying it to software development.

- The manufactured product is a physical realization of the customer requirements.
- The function of the product can be verified against this physical realization.
- The costs of manufacture, including rework, repairs, recalls etc., are readily categorized and visible.

The benefit of the product to its user customer is readily categorized and visible.

In order to overcome these types of issues, and reap the benefit of QA applied to software, other terms, models and paradigms needed to be (and were) developed.

In order to identify the Software Costs and Benefits, remembering Fujitsu's term with cost and performance as prime consideration, a number of Software Characteristics where defined.

Notes These characteristics are sometimes referred to as Quality Attributes, Software Metrics or Functional and Non-Functional Requirements.

13.2 ISO-9126

ISO 9126 is a worldwide standard for the evaluation of software. The standard is split into four parts: quality model; external metrics; internal metrics; and quality in use metrics.

Notes

ISO 9126-1 symbolizes the latest research into characterizing software for the purposes of software quality control, software quality assurance and software process improvement (SPI). The ISO 9126-1 software quality model recognizes 6 main quality characteristics:

- Functionality
- Reliability
- Usability
- Efficiency
- Maintainability
- Portability

These characteristics are broken down into sub characteristics; a high level table is shown below. It is at the sub characteristic level that measurement for SPI will occur. The main characteristics of the ISO9126-1 quality model can be defined as follows:-

Functionality

Functionality is the essential purpose of any product or service. For certain items this is relatively easy to define, for example a ship's anchor has the function of holding a ship at a given location. The more functions a product has, e.g. an ATM machine, then the more complicated it becomes to define it's functionality. For software a list of functions can be specified, i.e. a sales order processing systems should be able to record customer information so that it can be used to reference a sales order.

Did u know? What is sales order system?

A sales order system should also provide the following functions:

- Record sales order product, price and quantity.
- Calculate total price.
- Calculate appropriate sales tax.
- Calculate date available to ship, based on inventory.
- Generate purchase orders when stock falls below a given threshold.

The list goes on and on but the main point to note is that functionality is expressed as a totality of essential functions that the software product provides. It is also important to note that the presence or absence of these functions in a software product can be verified as either existing or not, in that it is a Boolean (either a yes or no answer). The other software characteristics listed (i.e. usability) are only present to some degree, i.e. not a simple on or off. Many people get confused between overall process functionality (in which software plays a part) and software functionality. This is partly due to the fact that Data Flow Diagrams (DFDs) and other modeling tools can depict process functionality (as a set of data in\data out conversions) and software functionality. Consider a sales order process, that has both manual and software components. A function of the sales order process could be to record the sales order but we could implement a hard copy filing cabinet for the actual orders and only use software for calculating the price, tax and ship date. In this way the functionality of the software is limited to those calculation functions. SPI, or Software Process Improvement is different from overall Process Improvement or Process Re-engineering, ISO 9126-1 and other software quality models do not help measure overall Process costs/benefits but only the software component. The relationship between

Notes

software functionality within an overall business process is outside the scope of ISO 9126 and it is only the software functionality, or essential purpose of the software component, that is of interest for ISO 9126.

Following functionality, there are five other software attributes that characterize the usefulness of the software in a given environment.

Each of the following characteristics can only be measured (and are assumed to exist) when the functionality of a given system is present. In this way, for example, a system can not possess usability characteristics if the system does not function correctly (the two just don't go together).

Reliability

Once a software system is functioning, as specified, and delivered the reliability feature describes the capability of the system to maintain its service provision under defined conditions for defined periods of time. One aspect of this characteristic is fault tolerance that is the ability of a system to withstand component failure. For example if the network goes down for 20 seconds then comes back the system should be able to recover and continue functioning.

Usability

Usability only exists with regard to functionality and refers to the simplicity of use for a given function. For example a function of an ATM machine is to dispense cash as requested. Placing common amounts on the screen for selection, i.e. $20.00, $40.00, $100.00 etc., does not impact the function of the ATM but addresses the Usability of the function. The ability to learn how to use a system (learn ability) is also a major sub characteristic of usability.

Efficiency

Efficiency is concerned with the system resources used when providing the necessary functionality. The amount of disk space, memory, network etc. provides a good indication of this characteristic. As with a number of these characteristics, there are overlaps. For example the usability of a system is influenced by the system's Performance, in that if a system takes three hours to respond the system would not be easy to use although the essential issue is a performance or efficiency characteristic.

Maintainability

The capability to recognize and fix a fault within a software component is what the maintainability characteristic addresses. In other software quality models this characteristic is referenced as supportability. Maintainability is impacted by code readability or complexity as well as modularization. Anything that helps with identifying the cause of a fault and then fixing the fault is the concern of maintainability. Also the ability to verify (or test) a system, i.e. testability, is one of the sub characteristics of maintainability.

Portability

This characteristic refers to how well the software can accept to changes in its environment or with its requirements. The sub characteristics of this characteristic include adaptability. Object oriented design and implementation practices can contribute to the extent to which this characteristic is present in a given system.

Notes

Notes **ISO 9126 Observations**

For the most part, the overall structure of ISO9126-1 is similar to past models, McCall (1977) and Boehm (1978), although there are a couple of notable differences. Compliance comes under the functionality characteristic, this can be attributed to government initiatives like SOX. In many requirements specifications all characteristics, that are specified, that are not pure functional requirements are specified as Non-Functional requirements. It is interesting to note, with ISO9126, that compliance is seen as a functional characteristic.

Using the ISO 9126 (or any other quality model) for derivation of system requirements brings clarity of definition of purpose and operating capability.

For example, a rules engine approach to compliance would enable greater adaptability, should the compliance rules change. The functionality for compliance could be implemented in other ways but these other implementation methods may not produce as strong an adaptability characteristic as a rules, or some other component based, architecture.

Also, a designer typically will need to make trade offs between two or more characteristics when designing the system. Consider highly modularized code, this code is usually easy to maintain, i.e. has a good changeability characteristic, but may not perform as well (for CPU resource, as unstructured program code). On a similar vein a normalized database may not perform as well as a not normalized database. These trade offs need to be identified, so that informed design decisions can be made.

Although ISO 9126-1 is the latest proposal for a useful Quality Model, of software characteristics, it is unlikely to be the last. One thing is certain, the requirements (including compliance) and operating environment of software will be continually changing and with this change will come the continuing search to find useful characteristics that facilitate measurement and control of the software production process.

Self Assessment

Fill in the blanks:

1. SQA is not the only …….. activity in a project.
2. A procedure defines how an activity will be conducted. …….. are defined in plans, such as a Software Configuration Management Plan.
3. Software products are ……...…… , user manuals and technical documents, such as an Architectural Design Document.
4. The ……..... of software quality that assures that the standards, processes, and procedures are appropriate for the project and are correctly implemented.

13.3 Measuring Software

Measuring software is a powerful way to track progress towards project goals.

As Grady states, "Without such measures for managing software, it is difficult for any organization to understand whether it is successful, and it is difficult to resist frequent changes of strategy." Appropriately selected metrics can help both management and engineers maintain their focus on their goals.

Notes

The use of measurement is common. We use measurements in everyday life to do such things as weigh ourselves in the morning or when we check the time of day or the distance we have traveled in our car. Measurements are used extensively in most areas of production and manufacturing to estimate costs, calibrate equipment, assess quality, and monitor inventories. Science and engineering disciplines depend on the rigor that measurements provide, but what does measurement really mean?

According to Fenton, "measurement is the process by which numbers or symbols are assigned to attributes of entities in the real world in such a way as to describe them according to clearly defined rules". An entity is a person, place, thing, event or time period. An attribute is a feature or property of the entity.

To measure, we must first determine the entity.

Example: We could select a car as our entity. Once we select an entity, we must select the attribute of that entity that we want to describe. For example, the car's speed or the pressure in its tires would be two attributes of a car. Finally, we must have a defined and accepted mapping system. It is meaningless to say that the car's speed is 65 or its tire pressure is 75 unless we know that we are talking about miles per hour and pounds per square inch, respectively.

13.4 Product Quality Management

Delivering products with high quality — defined as "meeting specifications at the lowest possible cost" as well as "delivering the value a customer derives from a product or service" — is a top priority for manufacturers and industrial operations. Quality has many aspects and Invensys Operations Management applications have provided integrated quality management functionality to tens of thousands of companies worldwide.

Invensys quality applications cover the gamut: from real-time process visibility to methods for data monitoring and alarming; historian data capture and reporting functionality to store and display voluminous process data for quality analysis; enterprise-wide SPC; integrated MES for specification management, genealogy, BOM enforcement, OEE and downtime monitoring; batch recipe management software to collect information on batch quality and recipe specifications; and underlying ArchestrA framework for monitoring and configuring data levels and application templates to deliver nearly any quality capability.

Benefits

- Regulatory adherence and compliance
- Respond to process variation in real-time
- Productivity and cost of manufacturing improvements
- Actionable, real-time quality-related information.

Did u know? What are the Key Capabilities of PQM?

- Data monitoring and alarming
- Best-in-class enterprise SPC, with database backbone
- Advanced, interactive trending and charting
- Government and regulatory reporting capabilities

Notes

- Historical data collection
- Integration with Laboratory Information Management Systems (LIMS)

13.4.1 Process Quality Management

Technology without great design will not achieve the marketplace advantage that is a requirement for business success. When technology *is* supported by sophisticated design, the result is a supercharged and extremely potent business tool. Quality Process specializes in the efficient development and delivery of appealing and satisfying customer experiences that aid our clients in their success.

An effective Quality Management process provides support for rapid adoption and consistent and repeatable application of quality improvement methodologies. As manufacturers across the globe are experiencing increased competitive pressure, price erosion and shorter time-to-market requirements, they also face the challenge of products being developed and manufactured by dispersed teams and locations. Quality can suffer in this distributed environment if it's not made a priority. As a result, many companies are now focused on improving overall product quality and streamlining the processes across the product development and manufacturing organizations.

By adopting quality management processes, companies will be able to implement a formalized quality process that is optimized for their unique requirements, use a common language and shared terminology across the company, expose performance bottlenecks and drive continuous, yet controlled, process improvements. Ultimately, quality management process improvements lead to time-to-market benefits, increased competitive advantage, and happier customers.

Example: The quality processes that we follow are based on iterative development methodologies like XP & RUP. We make use of formalized review mechanisms derived from Fagan's Peer Reviews. A formalized communication process with well-defined methodologies with clients and customers helps in developing applications and systems, which meet user requirements to the dot.

13.5 External Standards

"International Organization of Standardization" Instead of using an acronym (IOS) they used the Greek word for equal, which is ISO.

ISO 9000

The SEI CMM is an attempt to improve software quality by improving the underlying software processes. Another attempt based on International Standards Organization (ISO) 9000 series is based on software quality improvement. This standard although being used in over 130 countries is not industry specific and can be applied to a wide range of products e.g. automobiles, televisions, refrigerators, etc. Thus, we can conclude that ISO 9000 series is a set of documents dealing with quality systems that can be used for software quality assurance purposes. Within the ISO 9000 series, standard ISO 9001 is most applicable to software development.

Contrasting ISO 9001 and CMM

Although ISO 9001 and CMM are related in a lot of ways there are some issues that are covered in one but not in the other. These differences are listed in Table 13.1.

Notes

Table 13.1: Differences between ISO 9001 and CMM

ISO 9001	CMM
It determines the minimum criteria for an acceptable quality system.	It emphasizes on continuous process improvement
ISO has a wider scope including software, hardware, materials and services.	ISO focuses strictly on software.

ISO 9001

ISO 9001 is a standard for quality management systems and CMMI is a model for process improvement. If an ISO-certified organization wishes to improve its processes continuously, implementing CMMI would be a good choice, as it provides more detailed practices for process improvement than the ISO standards. However, there are two issues that need to be resolved when an ISO-certified organization implements CMMI. First, it is not easy to identify any reusable parts of the ISO standards, and it would be advantageous to be able to reuse selected portions of the ISO standards during CMMI adoption in order to use existing resources to their best advantage. Second, it is difficult for an ISO-certified organization to implement CMMI in a straightforward, easy manner because of the differences in the language, structure, and details of the two sets of documents. In this paper, we present our unified model for ISO 9001:2000 and CMMI that resolves these two issues. Our model would be an extremely useful tool for ISO-certified organizations that plan to implement CMMI.

According to the Carnegie Mellon University Software Engineering Institute, CMM is a common-sense application of software or business process management and quality improvement concepts to software development and maintenance. It's a community-developed guide for evolving towards a culture of engineering excellence, model for organizational improvement. The underlying structure for reliable and consistent software process assessments and software capability evaluations.

The Capability Maturity Model for Software (CMM) is a framework that describes the key elements of an effective software process. There are CMM's for non software processes as well, such as Business Process Management (BPM). The CMM describes an evolutionary improvement path from an ad hoc, immature process to a mature, disciplined process. The CMM covers practices for planning, engineering, and managing software development and maintenance. When followed, these key practices improve the ability of organizations to meet goals for cost, schedule, functionality, and product quality. The CMM establishes a yardstick against which it is possible to judge, in a repeatable way, the maturity of an organization's software process and compare it to the state of the practice of the industry. The CMM can also be used by an organization to plan improvements to its software process. It also reflects the needs of individuals performing software process, improvement, software process assessments, or software capability evaluations; is documented; and is publicly available.

ISO 9002

ISO 9002 is no longer in use. It was the standard that applied to organizations that did not do design or development. It was made obsolete with the 2000 year revisions. Now companies that do not do design are registered to ISO 9001; they include a "Permissible Exclusion" in the Quality Manual stating that design and development do not apply and are not included in the Quality System. ISO 9000 is often used to refer to the set of ISO Quality Management System documents. ISO 9001 is the document that contains the requirements for the Quality Management System. You will register to ISO 9001. ISO 9000 is a guidance document on the fundamentals and vocabulary for quality management systems. ISO 9002 and ISO 9003 are no longer in use. All companies register to ISO 9001.

Notes

Notes The SEI CMM is an attempt to improve software quality by improving the underlying software processes. Explain

Self Assessment

Fill in the blanks:

5. ISO 9001 is a standard for quality management systems and ……...……is a model for process improvement.
6. ……...……. is no longer in use. It was the standard that applied to organizations that did not do design or development.
7. Within the ……series, standard ISO 9001 is most applicable to software development.
8. Delivering products with high quality — defined as "meeting ……...….. at the lowest possible cost" as well as "delivering the value a customer derives from a product or service"
9. As……., "Without such measures for managing software, it is difficult for any organization to understand whether it is successful, and it is difficult to resist frequent changes of strategy."
10. The capability to recognize and fix a fault within a software component is what the …….. characteristic addresses.
11. Usability includes looking at, capturing, and stating requirements based around user interface issues, things such as accessibility, ……...……, and consistency within the user interface.
12. ……. symbolizes the latest research into characterizing software for the purposes of software quality control, software quality assurance and software process improvement (SPI).
13. Efficiency is concerned with the system resources used when providing the necessary…….....
14. There are many definitions of these Software Quality Attributes but a common one is the …….. model which was developed by Robert Grady at Hewlett Packard.
15. ……..... is a common-sense application of software or business process management and quality improvement concepts to software development and maintenance.

Caselet

Software Quality and Car Racing

Change four tyres, add more than 80 litres fuel, tweak the car for better efficiency, clean the windshield, clear debris from the front grill, and give the driver water. All in less than 13 seconds.

Such breakthrough pit-stop performance in car racing events is possible only with a highly capable crew that has continuously improved its processes, much the same way as

Contd....

Notes

in software quality assurance, says Raghav S. Nandyal in Making Sense of Software Quality Assurance (www.tatamcgrawhill.com).

"Beyond winning and losing, poor pit stop performance can result in injuries, lost earnings, lost sponsorship, and lost jobs. Along similar lines, without an optimising software development process assisted with an equally capable competency management process, software initiatives will wrap up without a trace in today's competitive environment."

In racing, you are only as fast as the slowest team member and only as successful as the last pit-stop performance, reads a quote, cited in the book. Likewise, "with software development, you are only as successful as your team's ability to handle the steepest learning curve limited by the slowest learner on the team and leveraging from a strong software quality assurance competency that enables reuse of competency assets backed by the efficiencies of a mature development process."

Nandyal rues that software quality assurance is one of the first tasks to be axed when the company experiences tight cash flow. He traces the myopic reaction to bizarre and cynical staffing practices, which fill the assurance teams with non-performers!

The book has instructive chapters on classes of audits and appraisals, information capture, verification of evidence, and synthesis of final findings. An important chapter speaks about competencies required of professionals in this sphere of work.

13.6 Summary

- The correct definition of Software Quality Assurance goes something like:
- The function of software quality that assures that the standards, processes, and procedures are appropriate for the project and are correctly implemented.
- ISO 9126 is a worldwide standard for the evaluation of software. The standard is split into four parts: quality model; external metrics; internal metrics; and quality in use metrics.
- ISO 9126-1 symbolizes the latest research into characterizing software for the purposes of software quality control, software quality assurance and software process improvement (SPI).
- Measuring software is a powerful way to track progress towards project goals.
- As Grady states, "Without such measures for managing software, it is difficult for any organization to understand whether it is successful, and it is difficult to resist frequent changes of strategy."
- Appropriately selected metrics can help both management and engineers maintain their focus on their goals.
- "International Organization of Standardization" Instead of using an acronym (IOS) they used the Greek word for equal, which is ISO.

13.7 Keywords

ISO: International Organization of Standardization

SVV: Software Verification and Validation

SPI: Software Process Improvement

13.8 Review Questions

1. What testing activities you may want to automate in a project?
2. How to find that tools work well with your existing system?

Notes

3. How will you test the field that generates auto numbers of AUT when we click the button 'NEW" in the application?
4. How will you evaluate the fields in the application under test using automation tool?
5. Can we perform the test of single application at the same time using different tools on the same machine?
6. Difference between Web application Testing and Client Server Testing. State the different types for Web application Testing and Client Server Testing types?
7. What is 'configuration management'?
8. How to test the Web applications?
9. What are the problems encountered during the testing the application compatibility on different browsers and on different operating systems?
10. How exactly the testing the application compatibility on different browsers and on different operating systems is done?
11. How testing is proceeded when SRS or any other document is not given?
12. How do we test for severe memory leakages?
13. What is the difference between quality assurance and testing?

Answers: Self Assessment

1.	Checking	2.	Procedures
3.	Code	4.	Function
5.	CMMI	6.	ISO 9002
7.	ISO 9000	8.	Specifications
9.	Grady states	10.	Maintainability
11.	interface aesthetics	12.	ISO9126-1
13.	Functionality	14	FURPS+
15.	CMM		

13.9 Further Readings

Books

B. W. Boehm, *Software Risk Management: Principle and Practices*, IEEE Software, Vol. 8, No. 1, January 1991, pp. 32-41.

B. W. Boehm, *A Spiral Model of Software Development and Enhancement*, Computer, May 1988, pp. 61-72.

Online links

http://en.wikipedia.org/wiki/Software_quality

http://www.ocoudert.com/blog/2011/04/09/what-is-software-quality/

Notes

Unit 14: Small Projects

CONTENTS

Objectives

Introduction

14.1 Managing Small Projects

14.2 Problem with Student Projects

14.3 Content of Project Plan

14.4 Summary

14.5 Keywords

14.6 Review Questions

14.7 Further Readings

Objectives

After studying this unit, you will be able to:

- Recognize the management of small projects
- Explain the problems with student projects
- Describe the content of project plan

Introduction

Projects are defined by their goals, objectives, boundaries and constraints. Getting a detailed but clear picture of your project scope will put you a long way toward completing your project successfully.

Goals and objectives define what has to be done. Not everyone agrees on what is meant by goals and objectives, so for our purposes we'll define them as follows: A goal is simply a broad statement of what you want to do. The objectives are sub-goals, more detailed, that explain what must be done to achieve the goal. Your project should have only one goal, but may have several objectives. If it looks like you have more than one goal, all but one should be reclassified as an objective or else handled as the goal of a separate project. Defining the project scope can be challenging because you need to get everyone to agree on the final product.

Here is an example of a goal with several objectives:

- Goal (more broad): We want to move the office to Houston, Texas.
- Objective (more specific): Locate an office in Houston.
- Objective (more specific): Arrange for personnel and equipment transfer
- Objective (more specific): Transfer equipment and furnishings
- Objective (more specific): Transfer personnel
- Objective (more specific): Maintain business-as-usual during the transition

Notes

Project Boundaries identify inclusions and exclusions — things that we do or don't want to do in conjunction with the goals and objectives. These are things that may not be related to the project, but that must be considered anyway. For example:

- Personnel not wishing to transfer will be replaced and arrangements made for outplacement.
- Company vehicles will be sold and replaced with new ones in Houston.
- Disposal of current facilities is not part of this project.

Project Constraints define cost, schedule, or quality requirements. These may include budget limitations or schedule requirements or minimum acceptability. For example:

- The cost of the entire move can't exceed $50,000.
- The move must be completed during the month of June, next year.
- Payments on outplacement services can't exceed $1200 per employee.
- The new Offices must support 200 office workers.
- Involve others in defining the project scope. Unless you are the only one involved, you shouldn't be defining the project scope by yourself. You can't go off and start putting down ideas in a vacuum. You need to get the sponsor, the customer, and other stakeholders involved. If it's a family project, you'll want to get family members involved. A business project may require a more formal approach with meetings and written documentation. You may even want one or more of your team members involved to validate the definition and to begin translating it into requirements and workload.
- Be prepared to multi-task. Remember that project management is not done step by step. If you're working with a tight deadline then managing your project may involve several activities that may need to take place at the same time and be part of a repeating process. For example, if your goal is to relocate the company headquarters, you may not know the location that you are moving to until the first phase of the project, an analysis, is finished. You may have to perform the same steps several times until you find a suitable location — and you may be doing that at the same time you are making arrangements for the move.
- Beware of the predefined project. You may be handed a "project scope definition" when you first pick up the assignment. If that's the case, you'll need to go through the motions of defining the scope, just to be sure that the definition is complete and accurate. You should always verify information that is given to you that you haven't helped develop.
- Guard against creeping project scope. Once your project definition is completed you will want to guard against scope creep. This happens when work is added that wasn't in the original definition, requirements or tasks list. The project scope is developed as you define the project. It includes goals and objectives, descriptions, constraints and boundaries. The scope is further defined by the list of requirements and the task list. You'll want to avoid adding requirements or tasks that are outside the original goals and objectives, but just as important, you'll want to watch for work being performed that isn't in the requirements or the task list. Make sure you keep a copy of the original project scope for comparison purposes. You can use it to call attention to work that is out of scope and initiate a redefinition if necessary.
- Get the ball rolling early. When it comes to defining the project scope, you should get started early, get everyone involved that should be involved, and get agreement on the goal, objectives, boundaries and constraints.

Notes

Caution Not until this is done can you start listing requirements, building a task list and moving forward in earnest.

14.1 Managing Small Projects

As both an active project manager and project management trainer, we often get asked whether the project management best practices that are applicable for large projects can be applied on smaller projects. This is a really important question and one which all project managers must face up to when managing small projects.

Focusing on Project Delivery

One of the arguments against using project management methodologies is that they are very process-centric resulting in vast quantities of project documentation which are simply not practical or desirable on small projects. This is a powerful argument and any method which focuses on producing documentation at the expense of delivering the real business benefits of the project will be a hindrance rather than a benefit. After all, the name of the game in project management is delivering business objectives, not producing reams of documents.

There is an ongoing and active discussion within the software development community about the best way to produce software on projects. More recently, some software professionals have argued for more agile methods of producing software rather than the more traditional heavyweight methods which focused on producing vast quantities of documentation.

Agile methods focus on delivery of software rather than documentation. With this in mind, I think project managers everywhere can learn something from the agile methods employed in software development.

Task In short, this leads us to focus on project delivery rather than project documentation, although the critical choice project managers everywhere need to make is how much documentation is really necessary?

Apply the Best Practices

We believe in only producing as much as is required by the project. Nothing more and nothing less. A simple rule of thumb is: if it's useful in helping us to deliver the business objectives of the project then produce it, if it isn't useful in helping us to deliver the business objectives of the project then don't waste time to produce it. With this in mind, I believe that in all projects, at a minimum it is best to apply project management best practices.

Let's consider the best practices in turn and see whether or not the overhead lost in applying best practices is worth the benefits which can be gained.

Defining Objectives and Scope

Even on the smallest project there will be objectives which must be achieved. As a project manager, it is in your interest to define what these objectives are since you are likely to be assessed on whether the project meets those objectives. It is your responsibility to ensure the project meets those objectives and you are accountable for this. In short, the buck stops with you.

Notes

Now suppose you don't define and write down what the objectives are, you are always going to be at the mercy of any boss who decides he's got it in for you. The defined and documented set of objectives is your insurance policy against your manager later coming along and saying you didn't meet the objectives.

However, there is another reason why you still need to define and document the objectives even on a small project. You want to satisfy the needs of the stakeholders since that is what you are paid to do as a project manager. If the objectives aren't defined, then you won't be able to meet those needs through your project.

Similarly with defining the scope. The scope forms the boundary of your project. If you don't define what it is, the likelihood is that it will grow and grow as the project progresses and although you might have started managing a very small project, before long your project could become very much bigger than when you set out.

Notes You still need to document who are the stakeholders on a small project as well. By defining who these are, you can ensure that you cover all of their needs when you define the objectives and deliverables.

Defining Deliverables

Somebody is going to have to carry out the actual work to produce whatever is delivered from your project. Even if the deliverables might be small and don't take much time to produce, they should still be written down. By documenting these things and then having them reviewed by others allows errors to be found. Your aim should be to document a detailed enough set of descriptions of the products to be delivered.

These descriptions will then be used by the people who will produce the deliverables. Even if these descriptions take no more than a page of text, it is important to write them in a clear and unambiguous way. If you don't write down a description, it means that the person making the deliverable can interpret what is required in unexpected ways which will only result in work being done later to correct the mistakes. So, always define and document the deliverables.

Project Planning

If you were to walk up Mount Everest, you would never do it without a considerable amount of planning. Even if you walk up the hill at the back of your house, there is probably some planning involved - what time do you go? What should you take with you? It is the same on even the smallest project where you will still need to work out which activities are required to produce a deliverable, estimate how long the activities will take, work out how many staff and resources are required and assign activities and responsibilities to staff.

All of these things need to be written down and communicated effectively to the project team members. I've seen lots of people become unstuck because they think they need to use some kind of project management planning software such as Microsoft Project. This is an unnecessary overhead. I've noticed that people tend to waste too much time making their Microsoft Project Gantt charts look pretty, so that they lose sight of the reason why they are using the tool.

Instead, for small projects we find that creating a bar chart in Microsoft Excel is the best. It is simple and more than adequate for small projects. Just make each column a sequential date, write your tasks in the first column, and fill in the cells to represent the time the activity takes.

Notes

In addition to the bar chart, you will need to document the milestones on the project. Milestones are the dates by which you need to deliver certain things, or may be the date on which a major activity ends.

Task The responsibilities of each project member must also be documented in the project plan. Explain

Communication

Even in the smallest project team comprised of just a project manager and one other person, the project manager will still need to assign tasks and responsibilities to the other person. It can't be assumed that they will know what they should do without it being effectively communicated from the project manager. If the project manager doesn't assign them specific activities, then the chances are they will go ahead and work on things which are not needed by the project. So, either the project will end up delivering the wrong things, or the project will get delayed since time will need to be spent later on doing the activities which should have been done earlier.

You can communicate the plans via email, or give a print out of the plan to your project team member(s), or better still, call a meeting and run through the plan with the project team members. Remember, if the plan changes, you will also need to communicate the changes to your team as well.

Tracking and Reporting Progress

If we still consider our two person project team - the project manager and one other person - the project manager will need to know the progress of the activities which the other person is working on. This can be done in a variety of ways: a short daily email detailing the work completed, the work still left to do, and a list of any issues/problems. In most cases this will be sufficient.

Alternatively a short 15 minute face to face catch up can accomplish the same thing. Or a combination of the two things might be best. In any event, the project manager still needs to be fully aware of the progress that is being made so that progress can be tracked effectively.

Change Management

Even on our two person project, changes are likely to occur. Requests for change usually come from stakeholders and it is your responsibility as project manager to assess the impact of accepting these into the project. To do this, you need a good estimate of the impact the change will have in terms of the extra effort and cost involved. This will often impact the schedule as well, so by having a clear understanding of how the schedule and budget will be affected you can make the decision as to whether or not you will accept the change into your project.

On a small project there shouldn't be any need for any fancy change control board to decide if the change is accepted. A quick discussion with the key stakeholder(s) should be sufficient for you to come to a decision providing you have worked out the impact on cost and schedule.

One thing you should never do is simply accept the change. Even if you think the change is small, you should never accept any change(s) without fully understanding what its impact will be on cost and schedule. That is a recipe for what we call "scope creep" where the project grows bigger and bigger as more and more changes are added into the project.

Notes

Notes Before you know it, your small project has become a much larger one and you will inevitably fail to deliver your project to your original budget and schedule.

Risk Management

There will be risks even on a small project. Make sure you have thought through all the potential risks at the beginning of the project, monitor the top ten risks each week (or top five if the number of risks is small) and keep looking out for new risks. Failing to manage risk properly is one the main causes for projects to fail.

The overhead in managing risks is very low. On a recent project, I drew up a list of what I considered to be all the risks on the project. It came to about 10 risks in all. Of these, five were serious risks. We worked out a plan to avoid or minimize each risk. In all, it took me little over a couple of hours to do this. Then, each week on the project, we would spend say half an hour reviewing all the risks and thinking of any new ones. At the end of the project, whilst some risks actually had materialized, because we'd identified a plan at the start of the project to minimize the impact of these risks, the impact of these risks on the project ended up being minimal.

So, with little up front and ongoing effort, you get a big pay back if you manage the risks throughout the project.

Self Assessment

Fill in the blanks:

1. ……. methods focus on delivery of software rather than documentation.
2. Instead, for small projects we find that creating a …… in Microsoft Excel is the best.
3. On a small project there shouldn't be any need for any fancy change …….. to decide if the change is accepted.
4. Failing to manage …… properly is one the main causes for projects to fail.
5. It is in your interest to define what these …….. are since you are likely to be assessed on whether the project meets those objectives.
6. Even on the …….. project there will be objectives which must be achieved.

14.2 Problem with Student Projects

Why Student Projects are Good

Before we take a look at some of the key issues with student projects, I'd like to take a moment to explain why student projects are beneficial. First off, it promotes teamwork, something that is crucial in the game industry, since games are built, in most cases, by diverse teams of talented people. Learning to work alongside others towards the same goal is a valuable skill you must learn if you want to make it in this industry. Along with teamwork, it also promotes good communication skills. Because you will be working with people who may speak a different language than you in terms of development and discipline (not actual languages), you will need to learn how to effectively communicate with them. Oftentimes, designers will have trouble communicating to programmers, while programmers may have trouble communicating with

Notes

artists, so learning how to communicate to different team members effectively before you actually get a job in the industry is important and will make your life easier. Working on a student project also shows that you are capable of making and, hopefully, finishing an actual game. Student projects can make great portfolio pieces, act as learning opportunities, and can even get you noticed by industry professionals. If a studio sees that you actually made a working game, it looks a lot better than the student with the same degree that only has documentation and schoolwork to show them; they can clearly see your skills in action, and any strengths or weaknesses you may have had during the project.

Did u know? Why student projects are not limited?

Student projects are not limited by all of the constraints typical game studios are, so they are a great way to show off your creativity.

The Next Big Hit

The Problem: Everyone wants to make the next big hit in games, but this is rather unfeasible, especially if it's a student project. The simple fact is that you most likely won't have the resources, money, people, or time to create the greatest game ever. That's not to say that you can't make a game that is successful, but be careful not to set your sights too high.

The Solution: Keep your scope small and simple. Large, complex games take a lot of time and resources, which you likely won't have at your disposal. Think of ways to make your game stand out in other ways, such as unique and interesting gameplay, and take a look at Independent Games Festival and Game Jam winners. Lower your expectations for the game in terms of extreme success, and keep the game itself small.

Notes If you start small, and create a fun, playable and polished game in a short amount of time, you can always add a few extra features afterward.

Scope/Feature Creep

The Problem: The team wants to make a great game, so they keep adding more features and expanding the scope until the game becomes a monster they can't tame. This goes along with the "Next Big Hit" section above, and can lead to complete failure of a project, as well as bad time management.

The Solution: Keep the game simple and expand on it once it's playable. Stick to the pre-production, and remember that you can always add new features later on. If any cool ideas for new features come up, jot them down and keep them stored somewhere for later use. The key to avoiding scope and feature creep is focus on what immediately has to be done to make the game even remotely playable; if the game isn't playable, there's no sense in trying to add new things to make it better. Follow the pre-production through to the end so that you have a completed game, then you can add more to it, but only after the game is complete.

Time Management

The Problem: Bad time management can easily lead to the failure of a project, and can be attributed to a number of things, including too large of a scope, scope creep, bottlenecks, individuals, and more. The majority of times we've witnessed bad time management were on projects with

Notes

deadlines (for IGF, etc.) and on projects that lasted too many semesters, where those that started the project had already graduated.

The Solution: Proper time management skills are extremely valuable, especially in the industry. If your project is having issues with this, take the time to figure out where the bulk of the problem lies. Use the "Where is the Time Spent" resource below to help evaluate what could be holding the game up. Once determined, talk with those team members about improving their time management; you may want to recommend the second link below. It the issue lies with the entire team (not the game), put together a meeting to discuss how to improve this. If there are deadlines involved, make sure that those deadlines are clearly communicated to the entire team, and that everyone understands what must be done before then. If the game seems farther behind than it should be, you may want to try making a schedule and/or reevaluating the scope.

Did u know? What does any time management software do?

- Capture, organize and manage all your projects and tasks in one central place
- Align your time, effort and energy with your highest priorities so you can focus on what really matters most to you
- Take control of your time and get your life organized
- Make great progress each day and complete your important projects in less time
- Be more focused, relaxed and productive while you work
- Never have to worry about missing an important deadline again
- Make time for your loved ones and the activities you really enjoy
- Import all your existing data from Outlook (2003, 2007, 2010) and keep it in sync.

Lack of Documentation

The Problem: Students seem to hate documenting anything, and will either not document at all, or not document effectively. These are bad habits to get into, since documenting your work is rather important in the industry. Without documentation, there is no clarified vision of the game, and miscommunication can lead to all sorts of problems during development. Not documenting effectively can also cause the game to fall apart.

The Solution: Get into the habit of documenting everything; it's good practice and will help solidify your game. You also need to keep all documentation up to date and it should be edited often to avoid discrepancies during development. Furthermore, ensure that everyone on the team has access to the most up to date documentation at all times, since there's no point to having it if no one on the team can see it. Good documentation is the backbone of the game, so take the time to do it, and do it right. If you're unsure of how to document something, talk with your leads, other team members, or professors.

Know Your Limitations

The Problem: Someone, or possibly everyone, on the team is trying to learn new skills at the same time the game is being built. While developing games is a learning experience, you need to know your limitations. If the team has to take the time to teach one member how to do everything, that's taking time away from working on the game. If no one on the team knows anything about developing games, all of the time will be spent learning rather than building. Another problem with limitations is that one team member will try to do everything; this leads

Notes

to bottlenecks in development where things get backed up because everything falls onto that one overwhelmed person.

The Solution: Know what your and your teammates' limitations are, and don't jump into a game project without knowing how to do anything. Learn development skills before joining a project and you can spend more time making the game work instead of learning how to make a game work. Don't take on tasks you can't deal with, especially if you don't actually have the skills involved. Putting on too many hats can be a common mistake, but it's important to divide work up so one person doesn't become bogged down or overwhelmed. Know what your limits are and abide by them to use these projects to fine tune and show off your skills.

Insufficient or Lack of Leadership

The Problem: Because most of the team members become friends through the process of development, leadership can often suffer, especially if there isn't a professor looking over their shoulders. Since the students become good friends, they don't want to tell their buddy that he needs to step up his game. Insufficient leadership on the students' part can really destroy the game from the inside out. On the other hand, it could be that there is no student leadership, and everything is run by a professor. This situation is okay for a class project, but for a student project, which takes place outside of the classroom, it really is best to have students at least partially leading each other.

The Solution: First off, let the students lead if they aren't already. That's not to say that professors can't help at times, but this is a learning experience for students to prepare them for the industry. If the friend issue is a problem with the leadership on the team, the leads need to realize that they have to put their foot down sometimes, for the good of the game. Good team leaders should be able to communicate effectively, in a friendly but professional manner. Since leads should be the ones the rest of the team goes to if they have any problems, it's also important for them to listen to and deal with the problems accordingly. For any leads on game projects, we'd definitely recommend reading the book below.

Task In a group of four analyze the solution for lack of leadership in small projects.

Not Enough Playtesting

The Problem: Sometimes the game is not playtested during development, or very rarely is. You have to playtest your game to ensure that everything is working as it should, to make sure that it is fun and playable, to discover and fix any bugs, and much more. If the team doesn't play their own game, how will they know if everything is working properly, or if it even works? It's also imperative to have other people play your game, and provide feedback. When the only people playing the game are the developers, it can be difficult to tell if the game is fun or not, and developers will often see what they want to see rather than what's really going on, especially towards the middle and end of development.

The Solution: If the team isn't play testing the game regularly, try setting up designated playtest times. If a particular team member isn't play testing, and their work is being affected by it, someone should talk to them and explain, in a friendly but professional manner, why they need to test their work. You should also get other people to play your game and give you some feedback. This should be fairly easy, especially if you are an on-campus student. So long as those who play provide honest feedback, almost anyone can playtest for you, even if they aren't a gamer.

Notes

Self Assessment

Fill in the blanks:

7. Along with……, it also promotes good communication skills.
8. If your project is having issues with this, take the …… to figure out where the bulk of the problem lies.
9. If the friend issue is a problem with the …… on the team, the leads need to realize that they have to put their foot down sometimes, for the good of the game.
10. The key to avoiding …….. and feature creep is focus on what immediately has to be done to make the game even remotely playable.
11. Another problem with …… is that one team member will try to do everything; this leads to bottlenecks in development where things get backed up because everything falls onto that one overwhelmed person.
12. Good ……. should be able to communicate effectively, in a friendly but professional manner.

14.3 Content of Project Plan

Project Name: What is the project called, if there is a job code includes it here.

Budget: How much money is available to deliver this project. On micro small projects this will all relate to expenditure and staff costs will not be included, but clarify this.

Completion Date: When must this project is completed by.

Description: A short description of the project, giving an outline of what it is and what it should achieve. It is important to nail this down clearly before starting the project.

Objective: We remove and add this element form the template on a regular basis. This is often captured in the description, however sometimes it may be beneficial to state the objectives separately.

Constraints: In this section any public holidays and annual leave of key people should be noted. Any issues that could impact on the delivery of the project should also be recorded here. This could include the budget or completion date not being confirmed; the name of an event to still be agreed; resources not confirmed.

Roles and Responsibilities: Record who is delivering the project, who is supervising the project, the budget holder, key people who need to be involved and the role each person will play.

Main deliverables: What is the main output of this project, a report, an event, a display?

External Dependencies: What resources and/or people from outside your organisation/ department are required to deliver this project successfully?

Self Assessment

Fill in the blanks:

13. On micro small projects this will all relate to …….. and staff costs will not be included, but clarify this.
14. Any issues that could impact on the ……. of the project should also be recorded.
15. ……. is important to nail this down clearly before starting the project.

Notes

Caselet

A Primer on Project Management

All of us have heard, some read, the famous best seller on Execution: The discipline of getting things done by Larry Bossidy and Ram Charan. Execution translates visions of the organizations in to reality. In the software industry, it often translates to effective project management. The topic assumes more relevance today especially for software firms that are at "tipping point" in their growth stories.

The Project Manager (PM) and the development team today deal with many pressures - senior management, marketing, finance, customers, and users - during the software development process. These pressures impact the cost and the quality of the software produced. There are generally more than one or two reasons for a software project to fail, and it usually is a combination of technical, project management and business decisions.

Opting to Outsource

Project life cycles have become shorter, thanks to rapid evolution of technologies and markets. This has resulted in short-term contract agreements, with customers expecting the firm to ramp up in capabilities in short time to meet their needs. The clients demand more value for the outsourced work as cost arbitrage of the Indian firms continue to be eroded and are looking "to squeeze every dollar spent". Hence, there's an increasing need to manage projects effectively and efficiently – leaving no slack on cost, quality and schedule.

Software outsourcing projects are of two types: fixed price (FP) contracts and time & material (T&M) contracts. In FP contracts the software firm gets a fixed price and pays for all realised costs. This keeps the software firm interested in managing the projects effectively, with available resources at minimal costs to meet schedules.

In T&M contracts, though the client pays for all realized costs, the onus of finding resources (mainly competent engineers) for the projects rest with the firm.

This poses a challenge in executing effective staff management in a highly attrition-afflicted industry. All these factors compel companies to increasingly focus on the following project management capabilities.

Meticulous Methods

First, define and defend the boundaries of the project well with the client. . Effective requirements, scope management, defining exact deliverables and committing to all these become important. Research has often showed that badly defined requirements are one of the main reasons why software projects fail. These can be better managed by shortening the cycles for delivery and periodically delivering smaller work products that allows clients to provide timely feedback. Methods such as agile project management are tailored for this context. These methods reduce risks for both the client and the firm .

Second, have robust methodologies and frequent cross-checks to ensure that effort estimates are appropriate, especially in FP projects. While optimistic, underestimates often lead to unmanageable projects, pessimistic overestimates end up as a losing proposition. Hence, the need for reliable statistics.

Third, focus continually on people's productivity and skills . This becomes a challenge as firms tend to increase the bottom of the "pyramid model" of resourcing to reduce costs. Project managers often have to deal with partially trained fresher's, mentor and motivate

Contd....

Notes

them and promote them to get them to the level of co-owners of the project . Apart from technical capability building, this also requires the firms to invest in project management capabilities. The likes of Project Management Institute (PMI) have developed exhaustive framework for accessing and developing project management capabilities. But the unique nature of the software industry requires a comprehensive project management training programme that focuses on experiential learning through internal case studies and sessions conducted by in-house experts. Since each firm is unique with respect to its organizational climate and the types of projects they handle , best practices need to be tailored to become effective in the organizational context with greater emphasis on practices that has worked well in the organization.

Risk Planning

The need for appropriate risk planning and management, as technologies and business environment continues to evolve, cannot be emphasized enough. If identified risks are not reassessed and controlled , there is no insight to the problems within the project. These problems could entail inadequate engineering resources or delayed third party components, situations that can result in unpleasant surprises for the project managers.

CXOs looking for levers to pull the organization to the next levels of performance while meeting higher customer expectations need look no further – the one labelled "Project Management" is the one to try first.

14.4 Summary

- So, in summary, applying the best practices to even a small project can be done without creating too much paperwork or overhead.
- The best practices are the things which countless project managers have done on thousands of projects and are deemed to be the "best practice" because they tend to help you to achieve the best results.
- Don't think that because you're managing a small project that you can ditch these best practices because if you do, you will regret it later when your project gets in a mess.
- First off, it promotes teamwork, something that is crucial in the game industry, since games are built, in most cases, by diverse teams of talented people.
- Learning to work alongside others towards the same goal is a valuable skill you must learn if you want to make it in this industry.
- Along with teamwork, it also promotes good communication skills.
- Because you will be working with people who may speak a different language than you in terms of development and discipline (not actual languages), you will need to learn how to effectively communicate with them.

14.5 Keywords

Constraints: A constraint is a restriction on the degree of freedom you have in providing a solution. Constraints are effectively global requirements, such as limited development resources or a decision by senior management that restricts the way you develop a system.

PM: Project Manager

Scope: opportunity for exercising the faculties or abilities; capacity for action.

Notes

14.6 Review Questions

1. You have worked on various types of projects ranging from infrastructure to software. How are infrastructure projects different from software projects?
2. What are the critical factors that measure the success of capital projects?
3. In the present scenario of frequent failure of software projects, can you put forward some strategies that facilitates successful software projects?
4. Scope and deliverables of software projects are changed frequently. This has severe implications on the projects. How can a project manager minimize their impact on the project?
5. Project reviews and audits are important in project management. Can you share your experiences about their contribution in the success of a project?
6. You have valuable experiences in the field of project risk management. Can you share your views on how an organization following project management should equip to cope with and overcome project risks?
7. Involvement of stakeholders is crucial for the success of the project. What strategies would you suggest to project managers so that they can involve all the stakeholders in the various stages of a project?
8. What role do you envisage for future project managers in improving the discipline of project management?
9. How important do you think on the job training is for the project managers and team members? How can training needs be identified?
10. How you can deliver the product fast to costumers?

Answers: Self Assessment

1.	Agile	2.	bar chart
3.	control board	4.	risk
5.	objectives	6.	smallest
7.	teamwork	8.	time
9.	leadership	10.	scope
11.	limitations	12.	team leaders
13.	expenditure	14.	delivery
15.	Small description		

14.7 Further Readings

Books

B. W. Boehm, *Software Risk Management: Principle and Practices*, IEEE Software, Vol. 8, No. 1, January 1991, pp. 32-41.

B. W. Boehm, *A Spiral Model of Software Development and Enhancement*, Computer, May 1988, pp. 61-72.

Online links

http://www.seminarprojects.com/tag/small-software-project-ideas

http://www.seminarprojects.com/Thread-software-project-ideas