

Software Project and Configuration Management MANP 1433

Lecturer: Haslina Md Sarkan
haslinams@utm.my

Overview

Course Objectives

Course Materials

Course Evaluation

Course Contents

Course Objectives

- To develop a complete software project plan adhering to a software engineering standard, which includes supporting software quality plan and risk management plan.
- To organize and integrate the different roles and activities in a software development team based on the project plan.
- To predict and handle issues involved in software project management and relating them to the factors affecting software quality.
- To measure the software quality required adhering the software engineering standard in the project plan.
- To plan configuration management process to a small software project environment.

Course Materials

During the course the following material will be used:

- Module handouts in UTM e-learning system
- Reference books :
 - Software Project Management, 5th edition by B. Hughes and M. Cotterell, McGrawHill, 2009.
 - Mastering Software Project Management: Best Practices, Tools and Techniques, M. K. Chemuturi, T.M. Cagley Jr, J. Ross Publishing, 2010.
 - Configuration Management Best Practices: Practical Methods That Work in the Real World, Robert Aiello, Bob Aiello, Leslie Sachs, Addison Wesley, 2010.
 - Articles on Configuration Management, including: Software Configuration Management, Lcfg, Cfengine, Allfusion Harvest Change Manager, Quattor, Baseline (Configuration Management), Engineering Support, Component Repository Management, Haphaestus Books, 2011.⁴

Course Evaluation

- Coursework **60%**
 - Tests (2) 20%
 - Written Assignment (2) 20%
 - Oral Presentation of Assignment (1) 10%
 - Software Development Plan (1) 10%
- Examination **40%**

Course Contents

- Software project management overview (2h)
- Software project initiation and scope definition (2h)
- Software project planning, effort, schedule and cost estimation (3h)
- Software project enactment (2h)
- Software project review and evaluation (1h)
- Software project closure (1h)
- Software engineering measurement (2h)
- Software measurement tools (1h)
- Presentation Case Study & hands on with SDP (6h)
- Software Development Plan reporting, review and evaluation (1h)

Course Contents

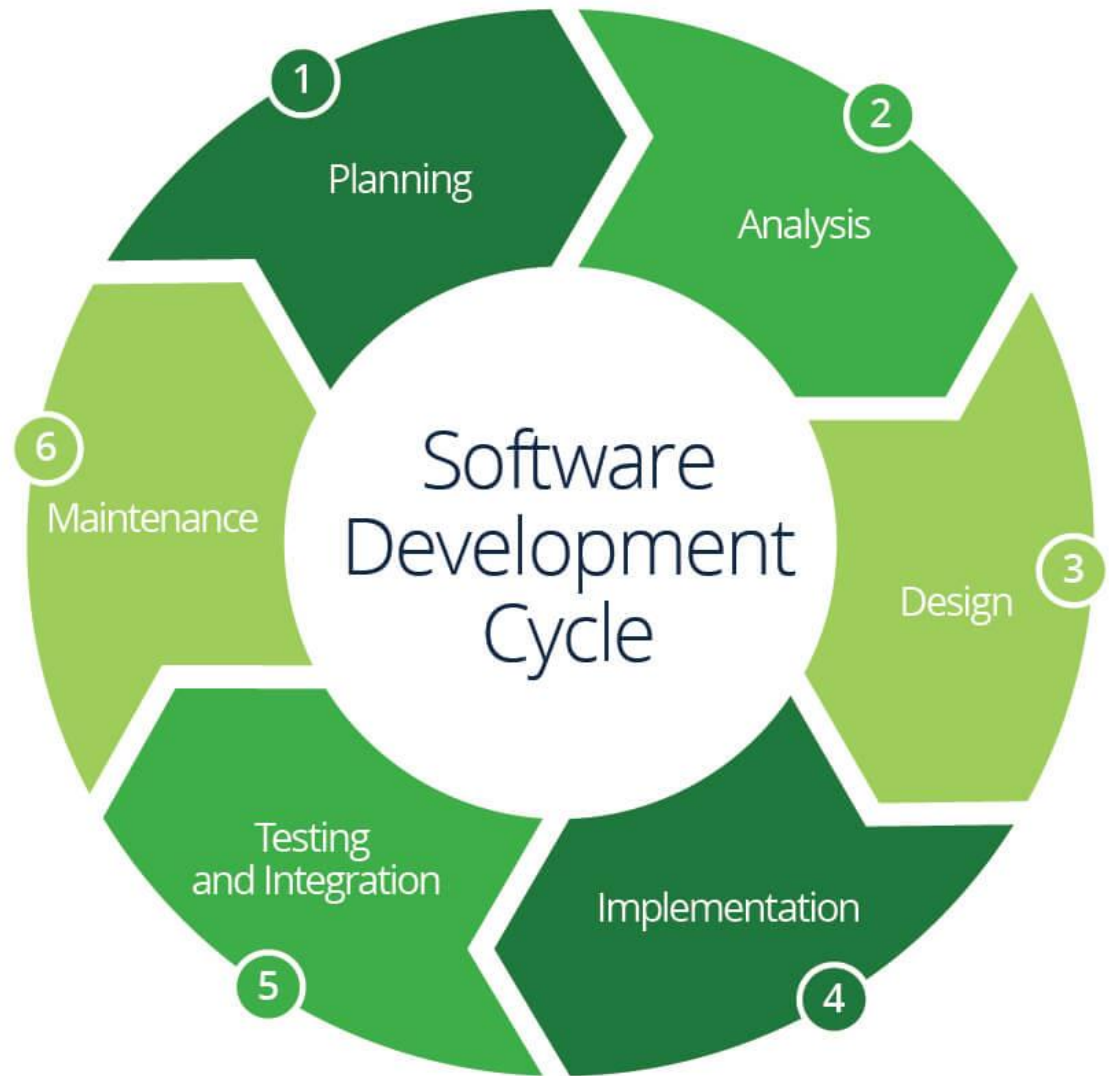
- Software configuration management (SCM) overview (2h)
- Management of SCM process (5h)
- Software configuration identification (2h)
- Software configuration control (2h)
- Software configuration status accounting (3h)
- Software configuration auditing (2h)
- Software release management and delivery (2h)
- Creating usage model using configuration management tool. (3h)



**“The most important software cost driver is
poor management”**

Barry Boehm

INTRODUCTION



What is Software Engineering?

- The engineering discipline was introduced to cater the poor quality of software, where projects exceeded time and budget and to ensure that software is built systematically, rigorously, measurably, on time, on budget and within specification.
- ISO/IEC/IEEE Systems and Software Engineering Vocabulary (SEVOCAB) defines software engineering as :

The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software - that is, the application of engineering to software

Managing Software Project

- Effective software project management has to take into account:
 - Client's understanding of the project needs and its feasibility.
 - Complexity especially regarding the impact of changing requirements.
 - Increased understanding and changing conditions will generate new or changed software requirements → iterative process to develop software.
 - A rapid rate of change in the underlying technology.

Project = Small business



- Both project and small business:
 - Consume resources
 - Have stakeholders
 - Have critical success factors
 - Have financial targets

Recipe for Project Success?



Ingredients

- Software development requires
 - Competent technologists
 - Competent managers
 - An effective organization structure



Method

- A software development method presents:
 - a set of guidelines or approaches for software development processes



- optimal choices for development tools and techniques, for high quality and cost-effective software products



TOPIC 1:

SOFTWARE PROJECT MANAGEMENT OVERVIEW

What is a project?

- A project is a **temporary** overlay on the standing organization
- It is created to achieve a **specific objective** defined by a **client** within **agreed time** and **budget**
- The standing organization is represented by the line-management
- **One project manager** will manage the project
- The project manager is responsible for the project and reports to the line-management

Project defined by PMI

- A temporary group activity designed to produce a unique product, service or result.
- A project is **temporary** in that it has a defined beginning and end in time, and therefore defined scope and resources.
- And a project is **unique** in that it is not a routine operation, but a specific set of operations designed to accomplish a singular goal.

In other words

- Projects have a purpose
- Projects are limited in time and space
- Projects have budget constraint
- Projects are collective
- Projects are unique
- Projects are realistic
- Projects are complex
- Projects are an adventure
- Projects can be assessed
- Projects are made up of stages

What is Project Management?

- Project management deals with the complex knowledge-based teamwork in organizations going through very fast changes in business environments.
- The purpose of project management is to carry out a series of activities as effective as possible.

Project management defined by PMI

- the application of knowledge, skills and techniques to execute projects effectively and efficiently. It's a strategic competency for organizations, enabling them to tie project results to business goals — and thus, better compete in their markets.

Work Package (PMI)

- It is the effort required to produce a deliverable within a project.
- This effort may be a single task or it could be several related tasks.

Software Work Package

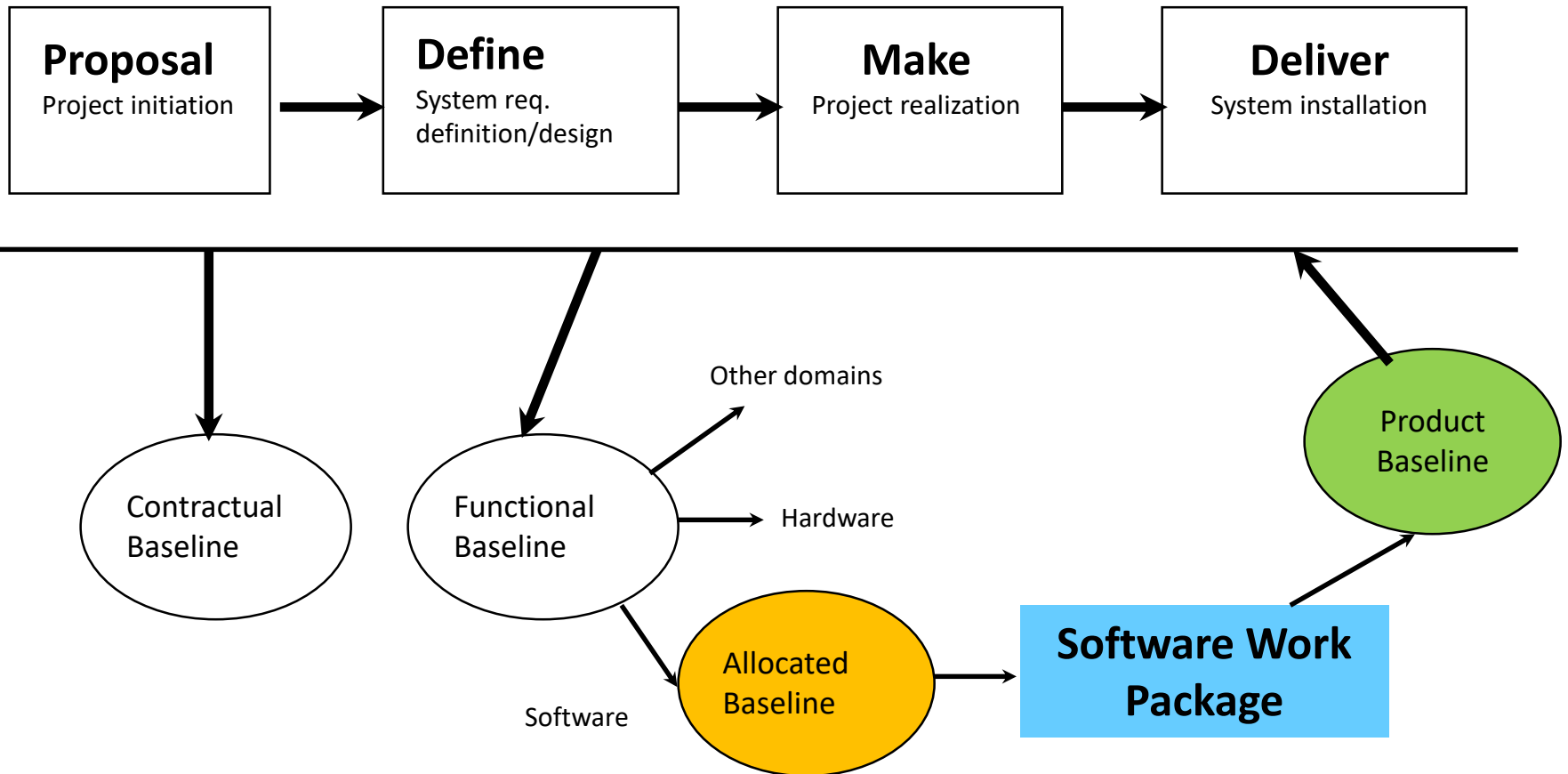
Purpose

To provide at an agreed price and schedule, software that meets the (functional) requirements, negotiated with the customer and verifiable by the customer

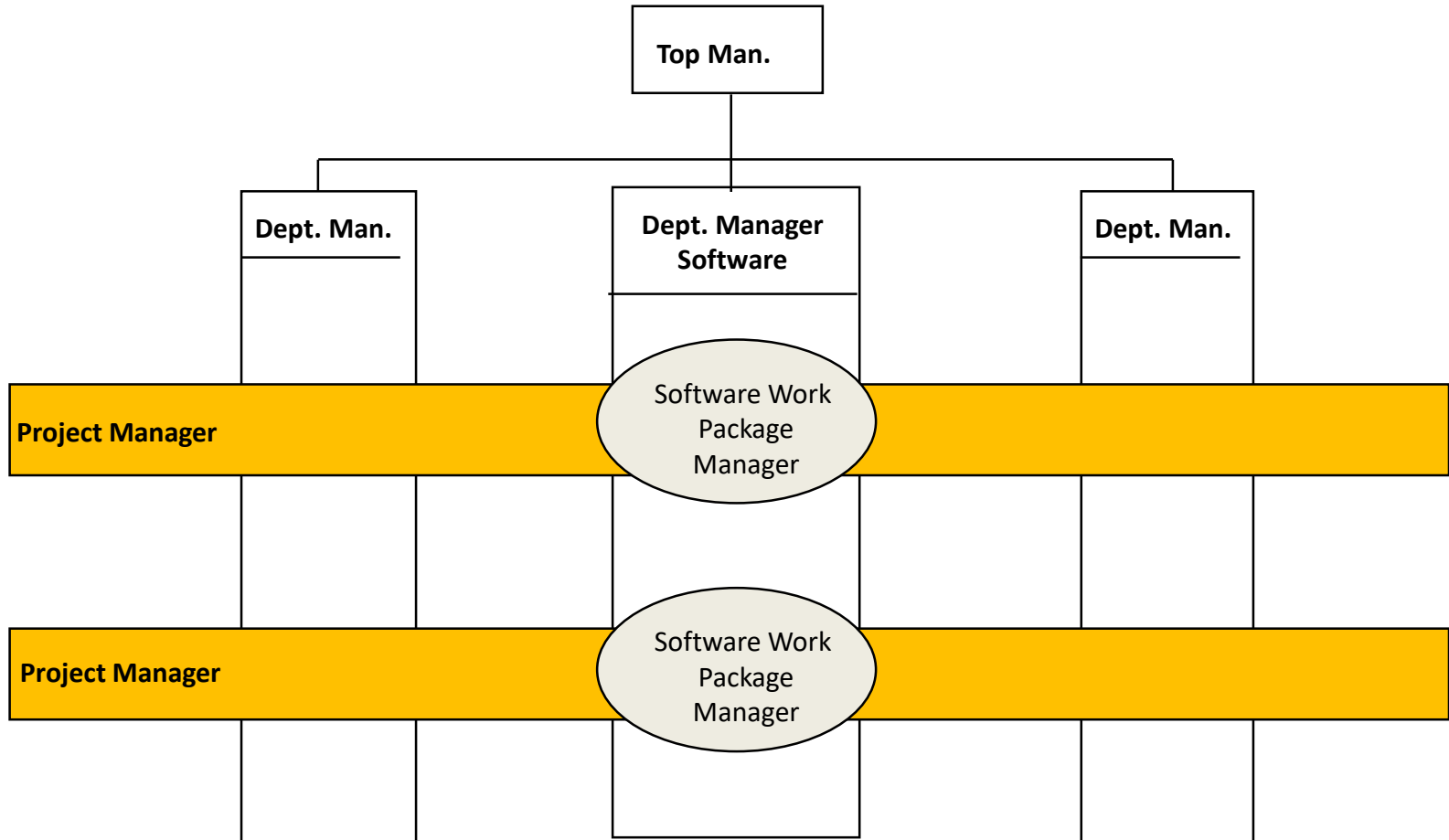
Size

The Software Work Package may consist of one or a number of CSCI's

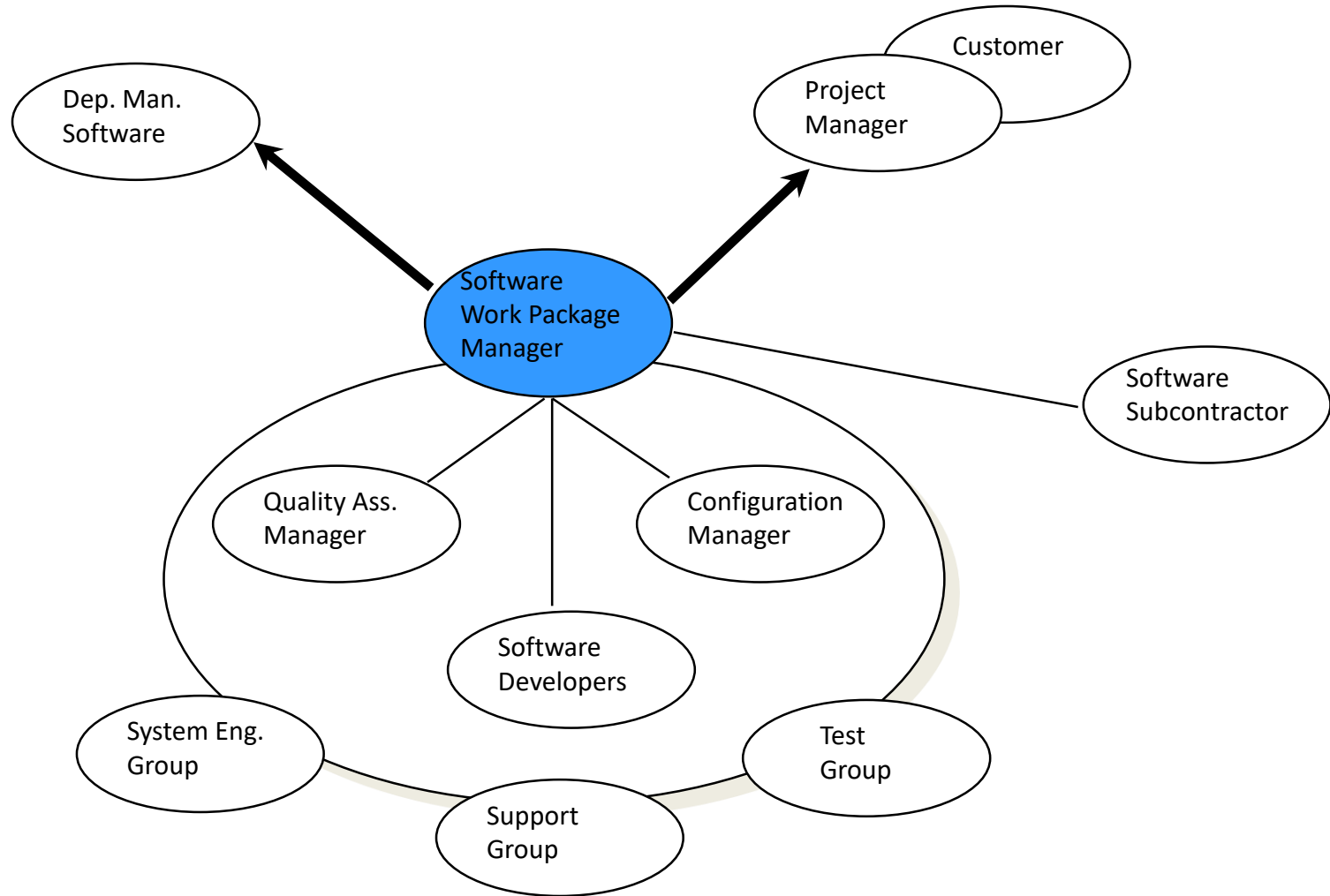
Software Work Package



Software Work Package



People involved



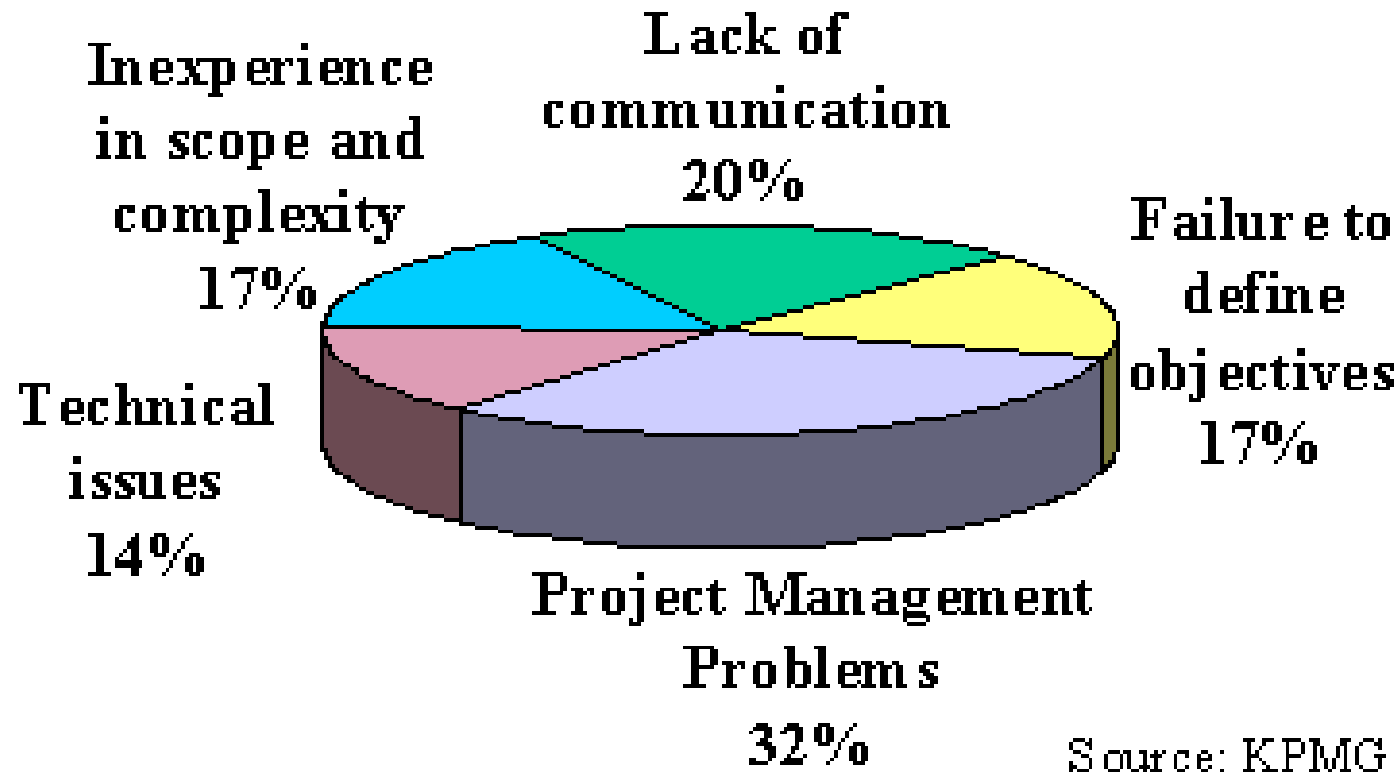
Software work package managing steps

- Setting-up and kick-off
 - Make sure that the input documents are ready, set up the environment, define the resources, enforce commitment
- Production
 - Drive and control the execution of the work and of each participants commitment
- Delivery
 - Carry out the product qualification, finalize the output documents, deliver them to the customer
- Closing down
 - Evaluate the project, transfer the knowledge and resources to the support group

Software work package documentation

- Management documentation
 - Software Development Plan (SDP)
 - Software Quality Program Planning (SQPP)
- Engineering documentation
 - Allocated baseline: SRS, IRS
 - Design documentation: SDD, IDD
 - Internal documentation: project notes, project log, etc.
 - Product baseline: Software Product Specification (SPS), Version Description Document (VDD)
- Quality assurance documentation
 - Audit and review reports
 - STP, STD, STR

Problems with software projects





What the user wanted



What the budget allowed for



What the timescale allowed for



What the technician designed



What the user finally got

**Software Product from
different perspectives**

Explain this scenario



How the customer explained it



How the Project Leader understood it



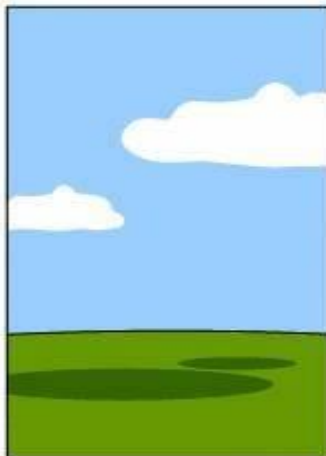
How the Analyst designed it



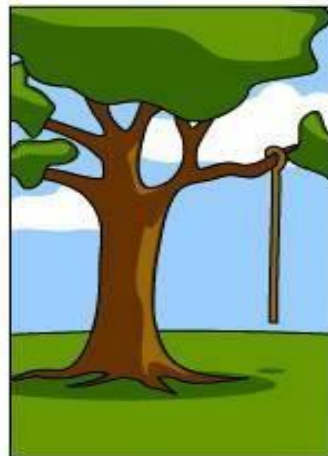
How the Programmer wrote it



How the Business Consultant described it



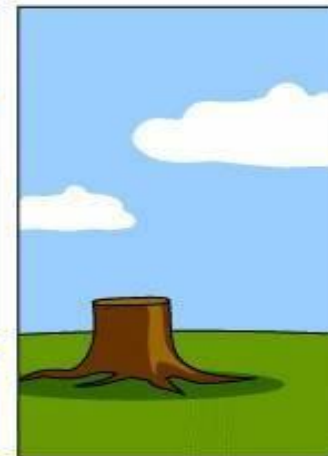
How the project was documented



What operations installed



How the customer was billed



How it was supported

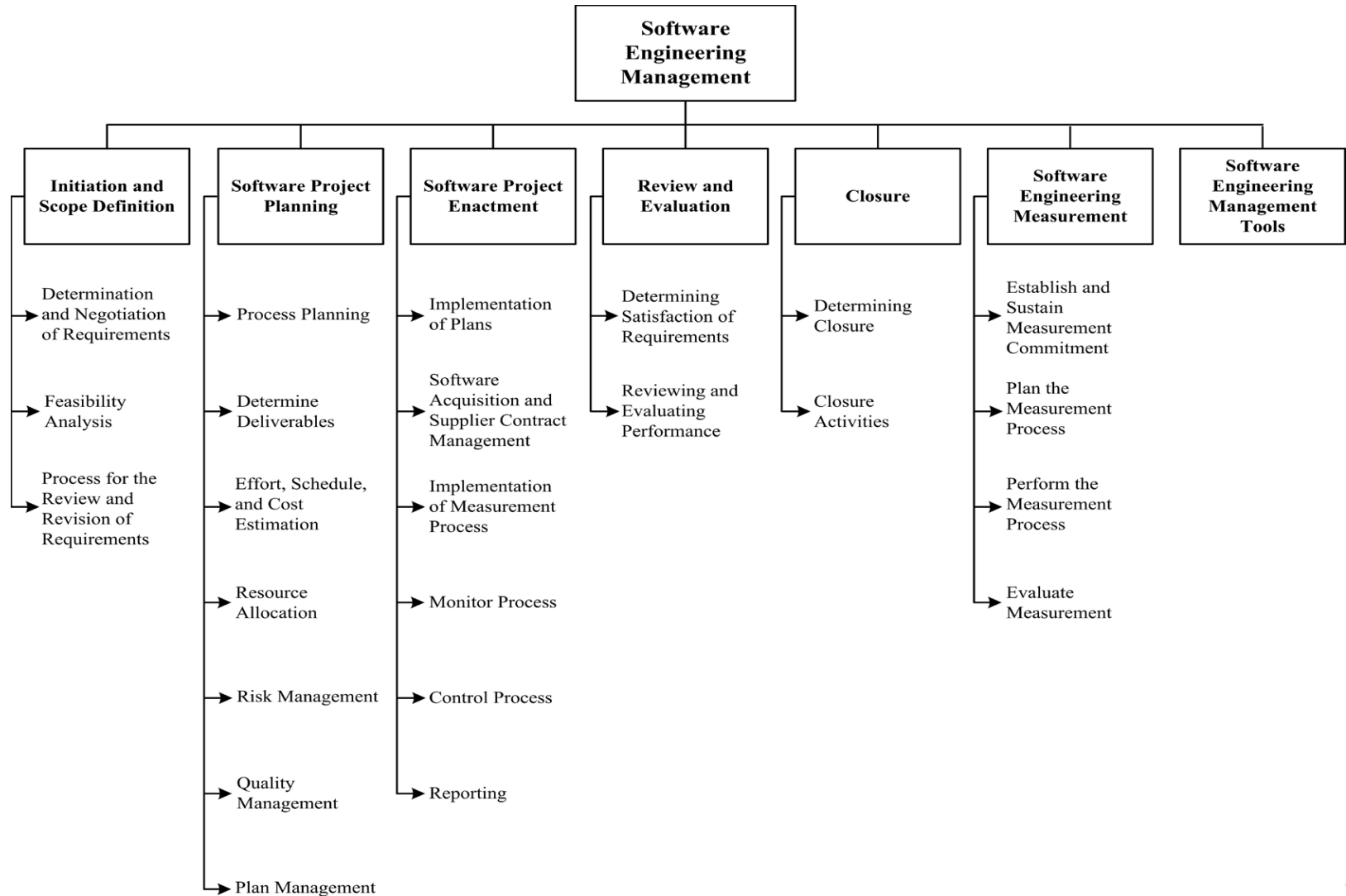


What the customer really needed

Software Project Management must Manage to a Future Target

- Business Changes
 - Market shifts and investment fluctuations
 - Portfolio changes
 - Mergers and Acquisitions
- Technology Changes
 - Hardware, software, networks, mobile
 - Fast, better, cheaper
- Process Changes
 - Agility vs. quality
 - Business and technology
- Staffing Changes
 - Shortages and gluts

Software Engineering Management according to SWEBOK



TOPIC 2:

SOFTWARE PROJECT INITIATION

AND SCOPE DEFINITION

- decision to embark on a software engineering project

Software Project Initiation

Initiation:

The process of formally recognizing that a new project exists
or
that an existing project should continue into its next phase.

This formal initiation links the project to the ongoing work of the standing organization.

Projects are typically authorized as a result of

A market demand

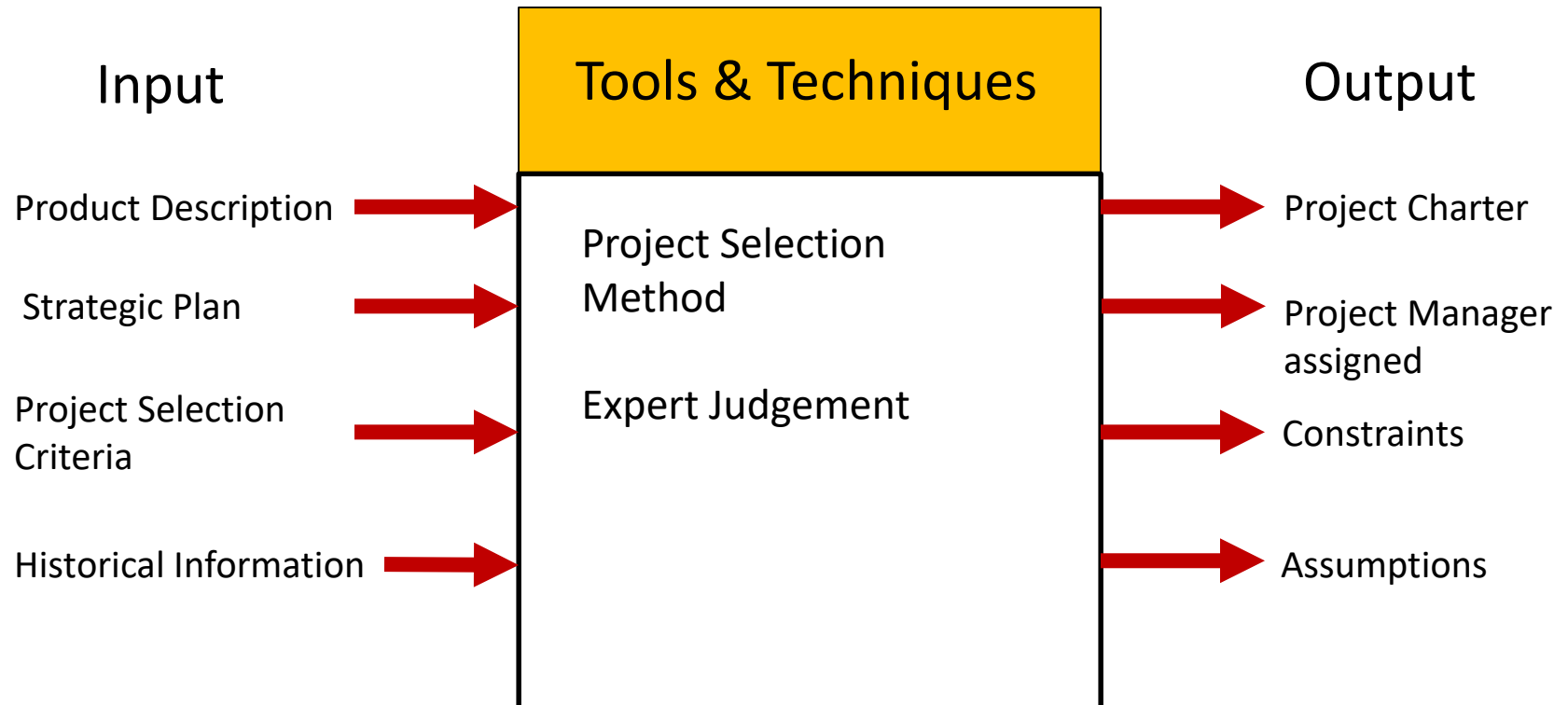
A business need

A customer request

A technological advance

A legal requirement

Initiation



Inputs to Initiation

- Product Description
 - Gather data
 - Identify need
- Strategic Plan
 - Establish goals and objectives
- Project Selection Criteria
 - E.g. financial returns
- Historical Information
 - Past projects
 - Previous phase

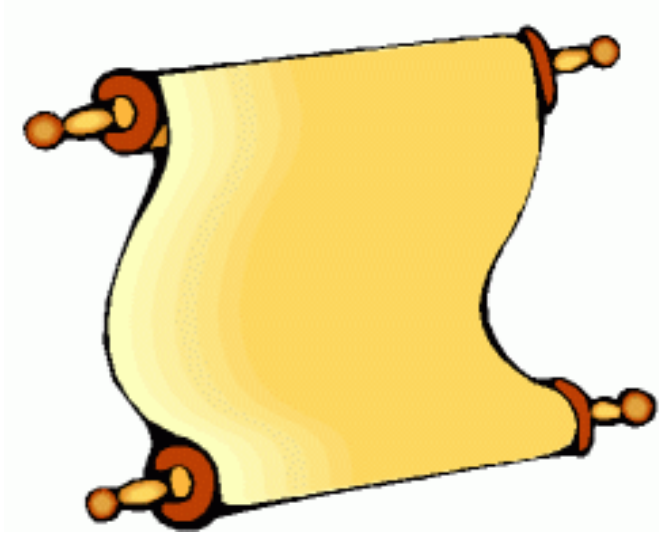
Tools and Techniques for Initiation

- Project Selection Methods (decision models)
 - Benefit measurement (e.g. through comparative approaches or scoring models)
 - Constrained optimization (mathematical based models)
- Expert Judgments
 - Expertise from other units within the organization
 - Consultants
 - Professional and technical associations
 - Industry groups

Outputs from Initiation

- Project Charter
 - A document formally recognizes the existence of a project.
- Project manager identified/assigned
 - Assigned prior to the start of the project plan execution, preferably before much project planning has been done.
- Constraints
 - Factors limiting the team's options.
- Assumptions
 - Factors that will be considered to be true, real or certain.

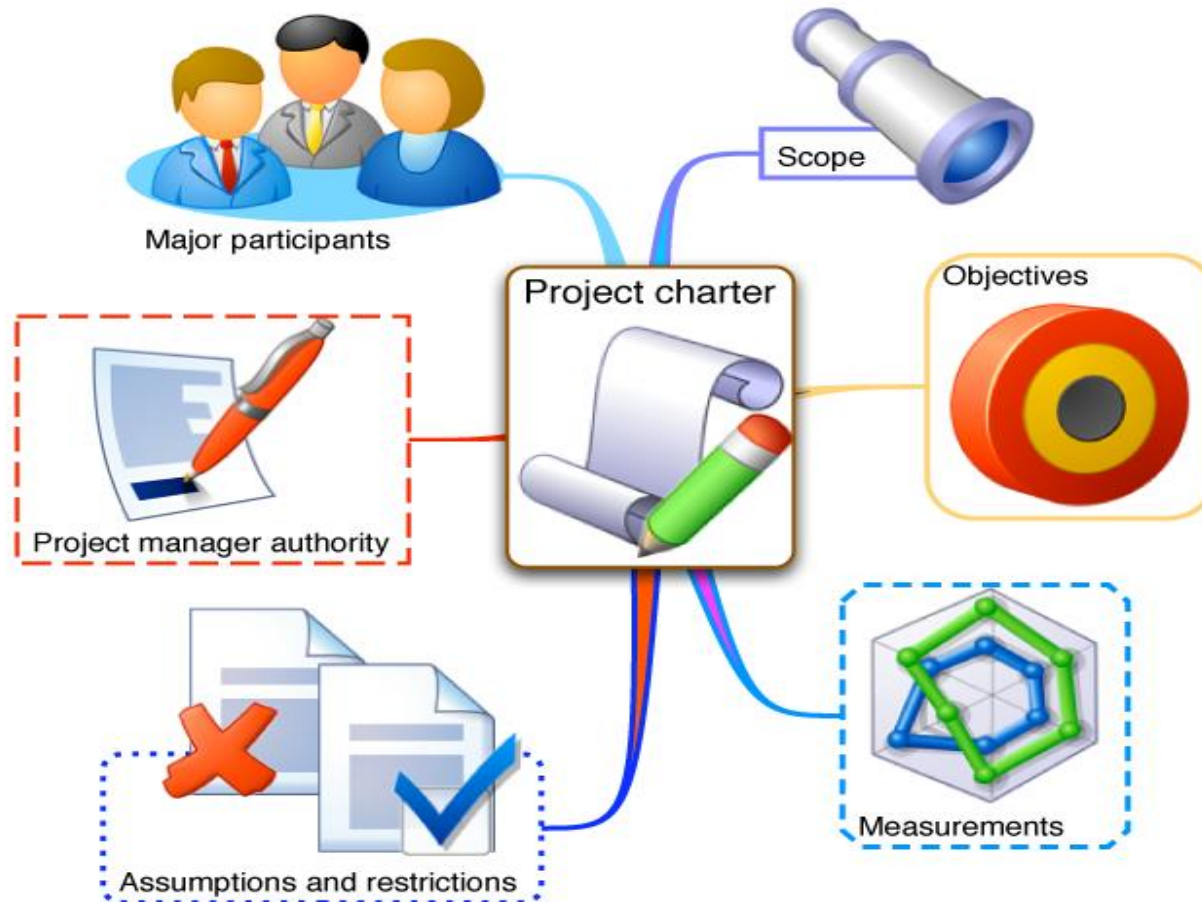
Project Charter



A document issued by senior management to the project manager to apply organizational resources to project activities.

- Includes the business need the project is to address
- Establishes the scope of the project
- Includes the product description
- Names the project manager as the responsible and authorized party
- Identifies the project deliverables, schedule and budget
- Is concise

Project Charter



<http://www.sixsigmadaily.com/tag/project-charter>

Role of the project manager

- A project manager acts as the single point of contact on a project.
- He/She is the individual responsible for:
 - planning and organizing the work
 - managing the day to day activities
 - delivering the project to the client

Responsibilities of a project manager

- Project planning
- Managing the project
- Lead project team
- Building client partnerships
- Targeting to the business



Responsibilities of a project manager

- Project planning
 - Develops preliminary study with project team, identifying business problem, requirements, project scope and benefits
 - Identifying key project results and milestones
 - Develops project plan and work breakdown structure and communicates to team and client
 - Determines needed resources including client involvement
 - Estimates timelines and phases
 - Influences selection of project team members
 - Assigns project responsibilities based on assessment of individual skills and development needs
 - Defines clear individual roles and performance expectations
 - Establishes acceptance criteria
 - Determines appropriate technological approach

Responsibilities of a project manager

- Managing the project
 - Continually reviews project status
 - Reviews work against key results criteria
 - Uses systematic method for logging project status, checking against schedule
 - Uses change management/request procedure
 - Uses project meetings to measure progress against plan, communicate changes and issues
 - Assesses skill-needed documentation of meetings, work, conversations, and decisions
 - Measures quality through testing against requirements
 - Conducts project reviews and walk-throughs with appropriate client involvement

Responsibilities of a project manager

- Lead project team
 - Involves team in planning
 - Uses both formal and informal methods to track project status
 - Recognizes individual and team accomplishments or results
 - Manages performance issues in a timely manner & delegates tasks effectively based on understanding individual strengths and weaknesses
 - Maintains open door for staff ideas and concerns
 - Sets performance and development objectives for staff
 - Schedules and holds regular team meetings

Responsibilities of a project manager

- Building client partnerships
 - Involves working jointly with client in defining project goals and key results
 - Works with client to ensure alignment of project to overall business goals
 - Listens and responds actively, documents client needs, changes and demands
 - Implements procedures for controlling and handling change
 - Develops client understanding of the system and trains in systems use
 - Presents and reports periodically to client
 - Establishes lines of responsibility and accountability to client

Responsibilities of a project manager

- Targeting to the business
 - Manages in accordance with visions and values
 - Links overall architecture principles
 - Interfaces effectively with business systems and processes
 - Plans for impacts on related systems departments to achieve maximum efficiency
 - Understands business needs, time and cost pressures
 - Keeps current with business and technology developments in competitors
 - Aligns project with corporate and business priorities and direction

Assessing Project Managers

- Skills and competencies are the two characteristics that determine the success or failure of a project manager
- Skills
 - At the visible level
 - Level of mastery can be measured
 - Can be acquired through trainings
 - Examples: estimating time & estimating cost
- Competencies
 - Traits that lie below the surface
 - Visible in practice
 - Immeasurable in the sense of determining whether a person has them and, if so, to what degree.
 - Difficult to develop through training
 - Some are hereditary

Criteria of a good Project Manager

- A well rounded project manager must have skills and competencies in the following categories:
 - Business: related to the business and business processes in general and do not involve specific business function knowledge
 - Personal: relate to individual. These skills do not involve another party in order to be practiced
 - Interpersonal: relate to the individual. Involve at least two people, neither one of which is the manager of the other
 - Management: relate to all aspects of management either people management or work management. Also related to the performance of strategic and tactical management functions not specific to any individual

Classification of Project Managers

- Team leader
 - Responsible for part of the project.
 - Generally assigned responsibility for an activity and can have a small number of staff assigned to the activity whose work they will manage
- Project manager
 - Responsible for projects that are classified as simpler, less complex, lower risk or not mission-critical
- Senior project manager
 - More experienced than the previous classes
 - Qualified to manage projects that are more complex, higher risk or mission-critical
- Program managers
 - Reserved for those who have achieved the highest level of professionalism and experience in project management
 - Often manages project managers on very complex or multi-project undertakings

Constraints

- For a contracted project, contractual provisions are generally the constraints.
- Example of constrain is predefined budget. This will limit the team's option regarding
 - Scope
 - Staff
 - schedule

Assumptions

- Normally involve a degree of risk.
- Can be identified early or they may be the result of risk identification.

Scope

Subdividing major project deliverables into more manageable components to:

- Improve accuracy of cost, time and resource estimates.
- Define a baseline for performance measurement and control
- Facilitate clear responsibility assignments

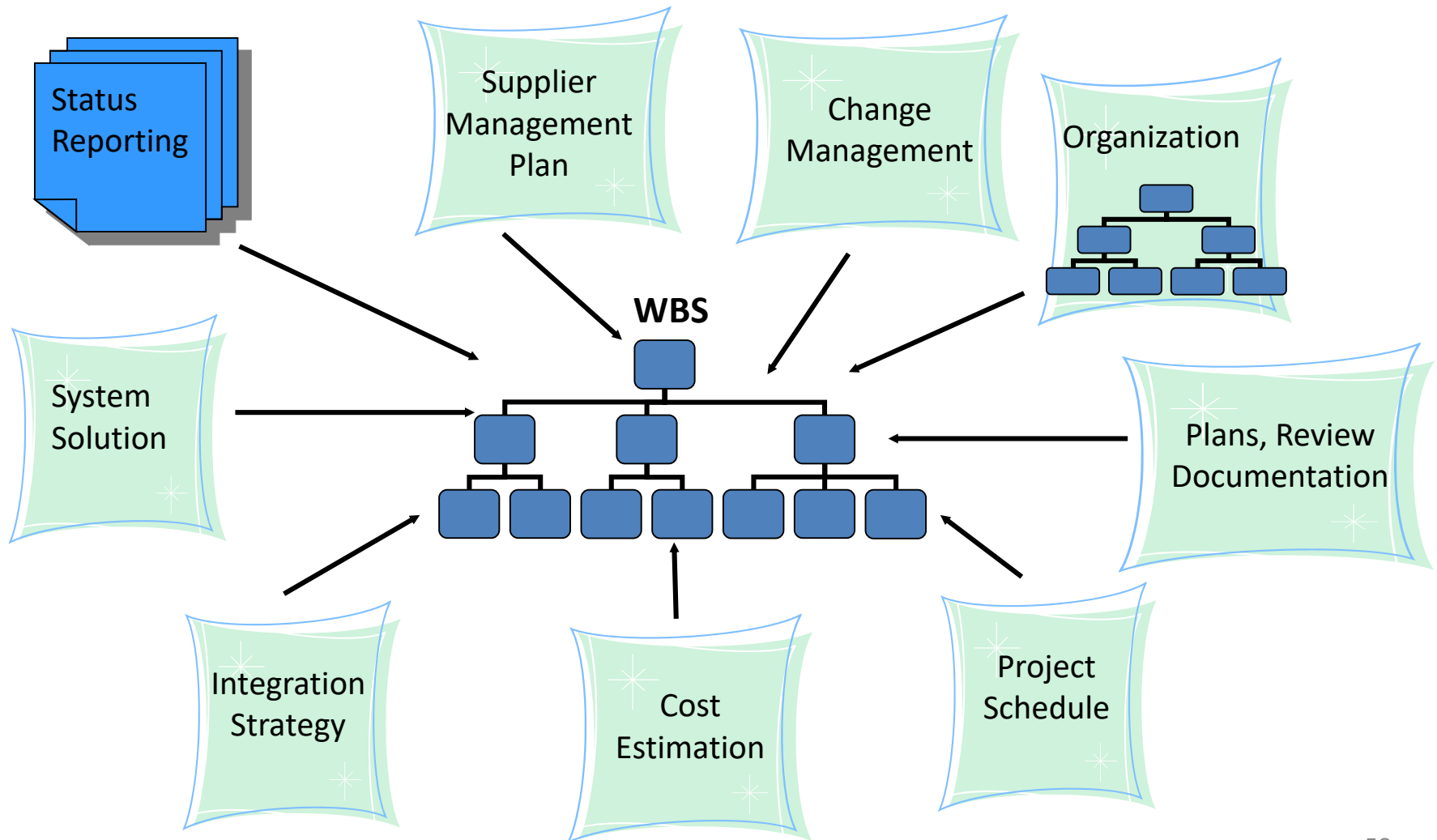
Scope

- Critical to project success.
- Poor scope definition leads to higher project costs due to inevitable changes that disrupt the project.
- The output of scope is the Work Breakdown Structure (WBS)

Work Breakdown Structure (WBS)

- Detailed representation of the scope of the project expressed in terms of work, resource and cost
- A hierarchical decomposition of work (tasks) needed to produce a system
- It provides:
 - Assistance in creating a clear allocation of responsibilities for work, resources and cost
 - A logical structure for cataloging all work elements
- Must be updated regularly (tasks can be added, modified or removed)

WBS Relationships

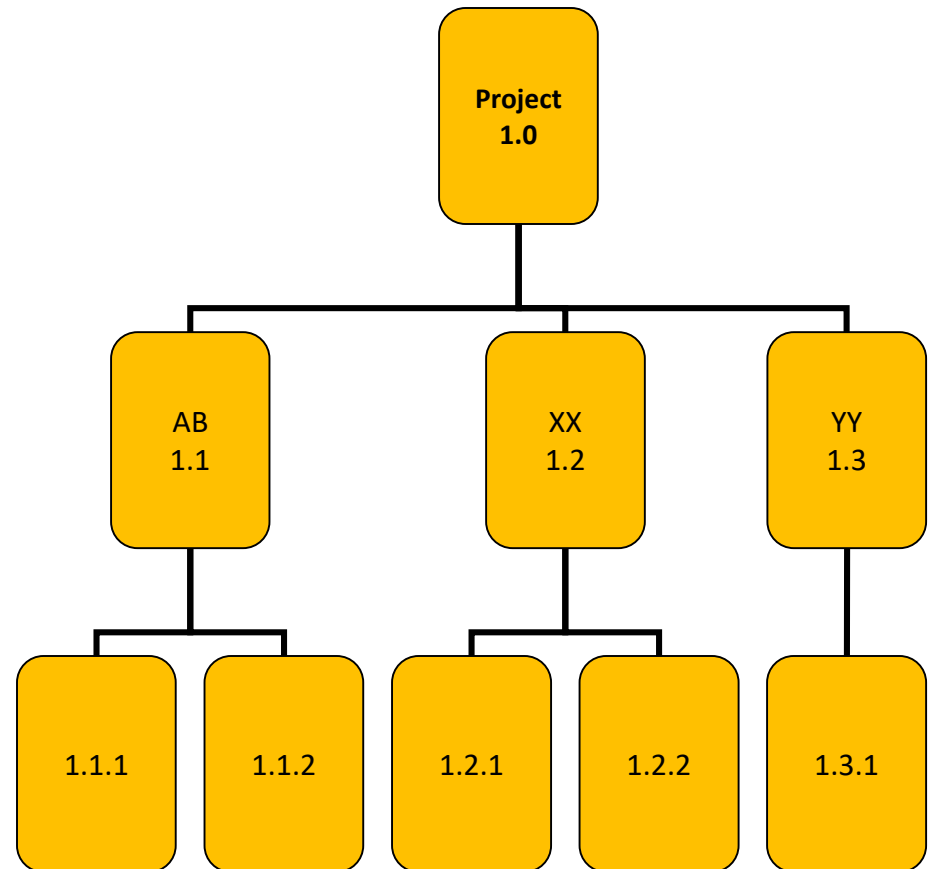


WBS Format

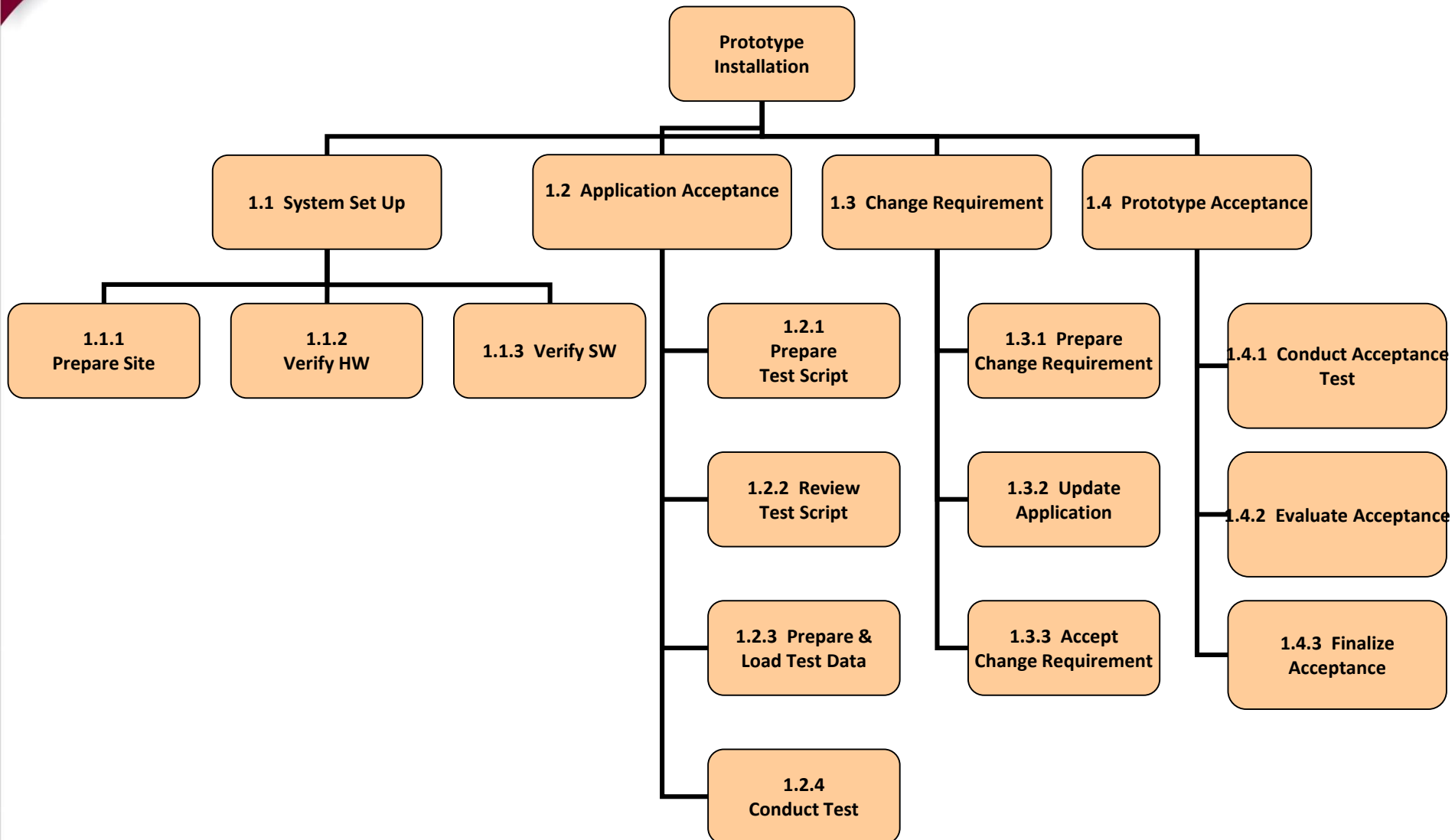
Indented List

- 1.1 XXXXXXXXXXXXXXXX
 - 1.1.1 XXXXXXXXXXXXXXXX
 - 1.1.2 XXXXXXXXXXXXXXXX
- 1.2 XXXXXXXXXXXXXXXX
 - 1.2.1 XXXXXXXXXXXXXXXX
 - 1.2.2 XXXXXXXXXXXXXXXX
- 1.3 XXXXXXXXXXXXXXXX
 - 1.3.1 XXXXXXXXXXXXXXXX
 - 1.3.2 XXXXXXXXXXXXXXXX
- 1.4 XXXXXXXXXXXXXXXX

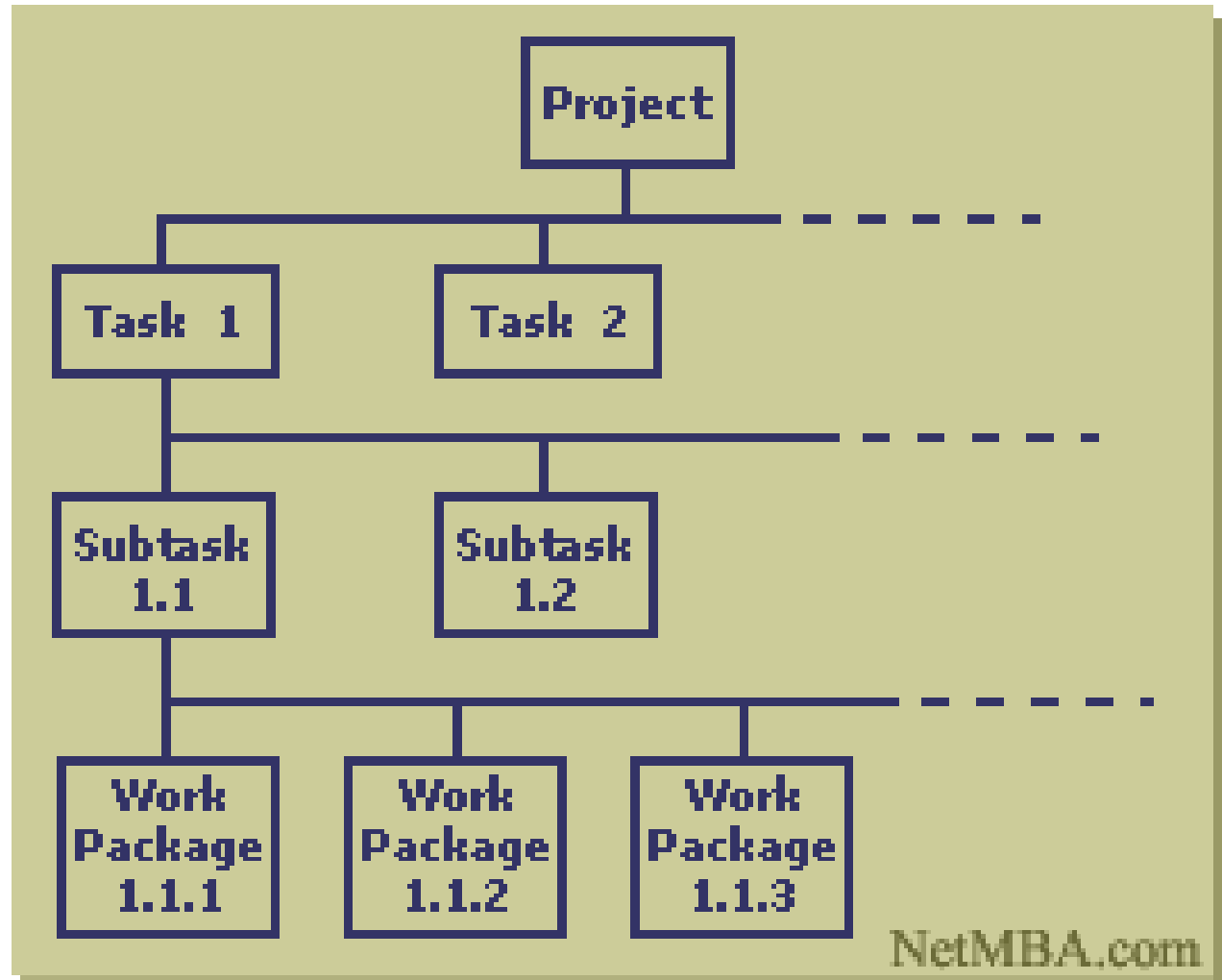
Tree Structure



Example of WBS view



Exercise on WBS



TOPIC 3:

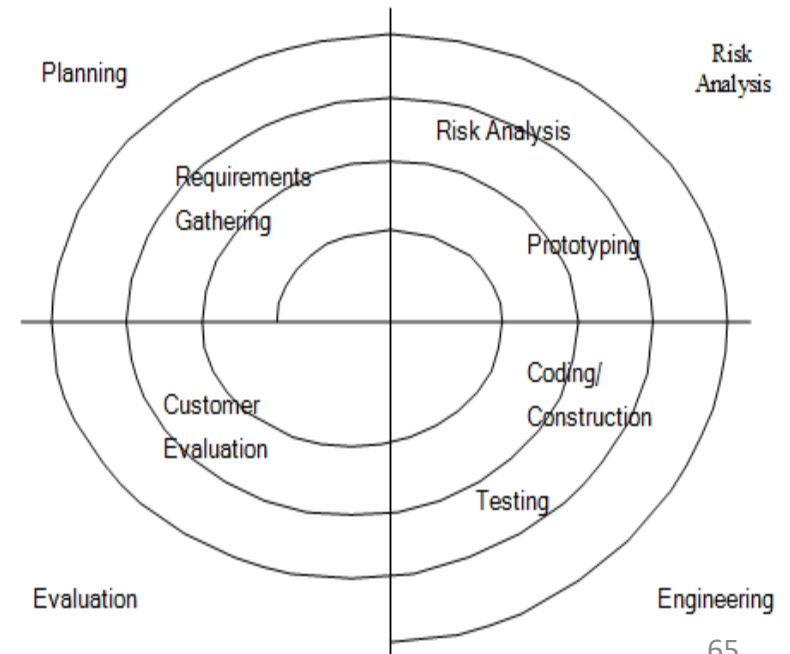
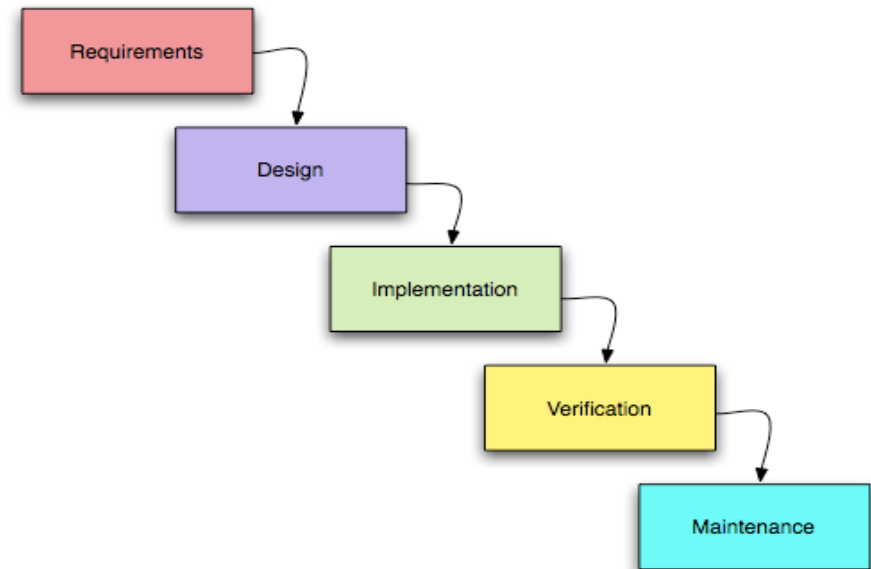
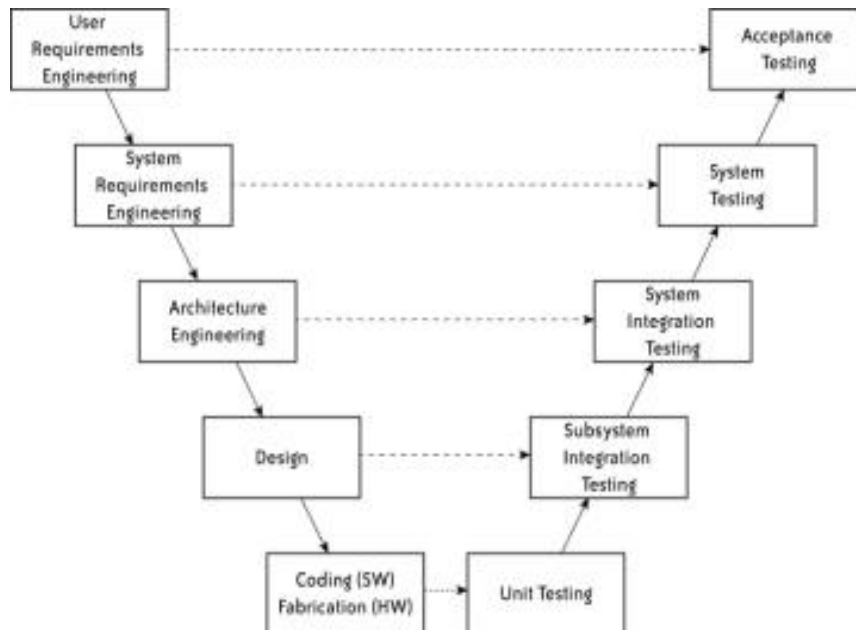
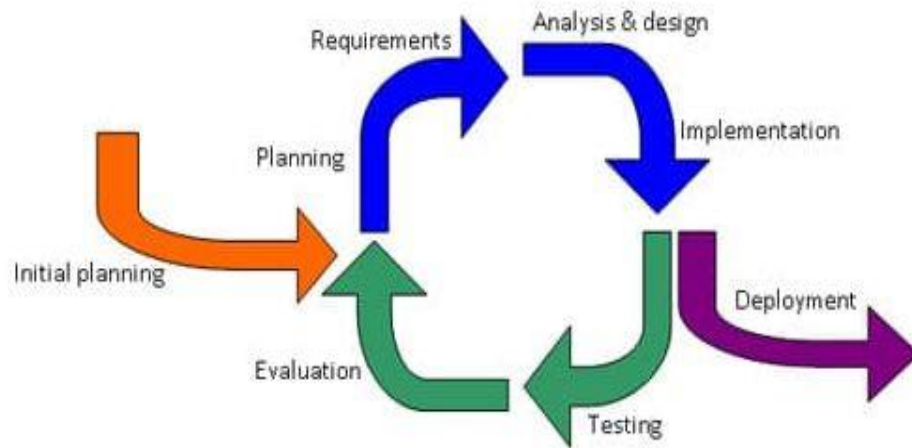
SOFTWARE PROJECT PLANNING, EFFORT, SCHEDULE AND COST ESTIMATION

Software Project Planning

- The first step should be selection of an appropriate software development life cycle model
 - perhaps tailoring it based on project scope, software requirements and a risk assessment.
- Other factors to be considered include
 - the nature of the application domain, functional and technical complexity
 - software quality requirements

Software development lifecycle (SDLC)

- No general rule specifying exactly the single phases, included in a software lifecycle or fixing their order
- Most software lifecycles are phased processes with clearly identifiable goals, milestones, and tasks, which can be summarized to general activities such as:
 - Requirements analysis
 - Design
 - Implementation
 - Testing
 - Maintenance



SDLC

- No ideal process model
- Hybrid solution - "Mix-and-match" lifecycle model
- Process model should be specified at the beginning of the project
- Risk analysis and resolution are significant for process control

Project Management Activities

- Planning
- Controlling
- Organizing
- Staffing
- Directing
- Monitoring
- Innovating
- Representing



Planning and control of a project

- Items of planning and control
- Work with a control cycle
- Execute the project in phases
- Define the baselines
- Management structure
(Project vs Line Management)



Items of planning and control

- Budget
- Time
- People
- Software Quality Assurance
- Commitment



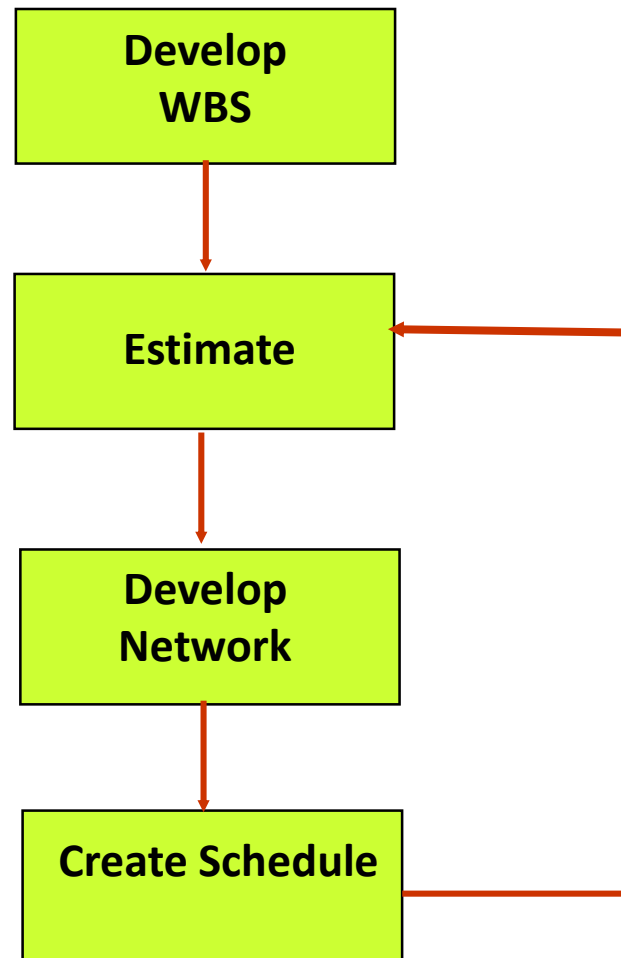
Project Plan

- An iterative activity, providing a means for effective management of a software project.
- It tries to avoid problems and prepare possible solutions.
- There is no fixed list how to do it, but a typical plan might include an analysis of project constraints (including, e.g. delivery date, available staff and expertise, budget) resulting to the specification of approximate project parameters, e.g.:
 - project structure,
 - project size,
 - product functions and their relationships.

Criteria of a Project Plan

- Goals and objectives of the product
- Work Breakdown Structure (WBS)
- Resource estimates
- Project schedule (when to deliver?)
- Risk analysis
- Quality considerations
- Configuration Management Issues
- Testing
- Contingency or exceptions

How to make a Project Plan?



Estimation

- It is about:
 - determining the project scope's constraints
 - investigating the quality as required by the client
 - preparing the project schedule and budget
 - adjusting to the available resources
 - taking into consideration the project risks

Estimation

- Is used to study and to predict the resources or time necessary to implement requirements
- Can be used to draw up project or even iteration plans
- Is used to analyse the viability of the investment analysis, to come up with a realistic pricing and to bid for a project competitively

Estimation

- Therefore proper project planning and control is not possible without a sound and reliable estimate
- Estimating size and cost are one of the most important topics in the area of software project management

Software Estimates

- Unlike other types of projects, software is an intangible product where specifications are not very obvious especially at its initial stages
- This characteristic puts software in a delicate position where inconsistency and vagueness of the project attributes makes the work estimation more difficult

Software Estimates

- In a project it is very important to know:
 - how long will it take
 - how much effort will be required
 - how many people will be involved
 - how many resources do I need
 - what are the risks

What to estimate?

- Effort
- Activity duration
- Resources:
 - **People**
 - **Equipment**
- Materials
- Cost
 - **Direct (e.g. space rental)**
 - **Indirect (e.g. management cost)**

Estimating Terms

- Effort: the number of labour units required to complete an activity or other project element
- Duration: number of work periods required to complete an activity or other project element.
 - Continuous duration is work time that is not interrupted.
 - Interruptible duration is work time that might be interrupted.
- Resources: any necessary item that might be directly used in & charged to the project
- Direct costs: labor, materials, equipment, services and fees that are charged to the project & must be traceable
- Indirect costs: overhead & admin expenses. Overhead might include user training (course development, classroom costs, etc)

When to estimate?

- Estimate is done many times throughout the project lifecycle
- Prepare estimates:
 - After WBS is developed
 - When determining whether to bid on an opportunity
 - When taking over a project to validate proposal estimates
 - When moving to the next phase of a project
 - When an assumption proves to be invalid
 - When the WBS changes to ascertain the effort and cost associated with the changes
 - When there are authorized changes in resources, materials or services

How to estimate?

- Software estimation is one of the most challenging and important activities in software development.
- Software estimation can never be an exact science
- Combination of good historical data and the use of a method can improve estimation accuracy

Estimation Types

- A few elements are to be considered when choosing the right method to estimate software development:
 - team's experience
 - the estimated lines of codes (LOC) to be delivered
 - the programming language used
 - the complexity of the software
 - the design methodologies
 - the environment for the tests to be conducted etc.

Estimation Types

- From the point of view of the input data, estimation methods can be divided into three groups:
 - data-intensive
 - expert-based
 - hybrid methods

Data-intensive method

- The prediction relies upon an accurate estimation of:
 - the software size
 - its complexness
 - the number of interfaces needed or the number of screens
- Made up of mathematics and statistics equations
- Fixed and predefined
- The input parameters have to be derived from various project properties in order to obtain a more precise estimation

Expert-based method

- The characteristics include learning, inferring and analyzing previous projects
- Among the well known expert-based software models are the ones using soft computing methodologies like artificial neural networks (ANN), fuzzy logic (FL) and evolutionary computing (EC) which cater for real life situations where things are vague and flexibility is required to handle each one of them
- Expert Judgement is the most commonly used technique but difficult to verify the accuracy of the results
- Analogy Based Estimation (ABE) is a simple & flexible

Hybrid

- Hybrid means combining the available data and expert knowledge in order to come up with a good estimate
- The models are built by integrating expert opinion and data-driven techniques
- Assumed to work better because there is no isolate technique that does not have any flaw nor one that can be accepted by everyone and hybridizing a few approaches is hoped to produce more realistic estimations

Planning Estimation

- Establish the estimating objectives
 - Why are they being prepared?
 - What is the required accuracy?
 - Who is the intended audience?
 - When are the estimates needed?
- Determine the project details
 - Identify required information to prepare estimates
- Select the appropriate model
- Develop the estimating strategy and plan
 - Identify estimators
 - Develop a plan to gather the estimates
 - Review history of similar projects

Key factors in Software Estimation

- Project Manager has to balance between:
 - Cost
 - Schedule
 - Quality
- Project costs involve:
 - Hardware costs
 - Travel and training costs
 - **Effort costs**

Software Cost Estimation

- Software cost estimation is the process of predicting the amount of effort required to build a software system.
- Cost estimates are needed throughout the software lifecycle.
- Examining projects & characteristics helps develop underlying theory of how software development works & what factors generate highest costs.
- Cost model gives us some idea of what project will cost.
- Cost modeling is imprecise and difficult – it is important as a planning & evaluation tool.

Software Cost Estimation

- Size is a primary cost factor in most models. There are two common ways to measure software size:
 - lines of code
 - function points
- Lines of Code: The most commonly used measure of source code program length is the number of lines of code (LOC)
- Function points (FP): measure size in terms of the amount of functionality in a system.

Software Cost Estimation

- Software project managers must estimate how much a software development is going to cost for the project budget.
- Types of costs:
 - Facilities costs
 - Hardware & software costs
 - Costs for physical space for team
 - Project management costs
 - Standards
 - Training
 - Travel
 - Management methods
 - People costs
 - Salary

Effort

- Effort is the number of labour units required to complete an activity or other project element
- The estimated range of effort required for a project, or parts of a project, can be determined using:
 - a calibrated estimation model based on historical size and effort data (when available)
 - other relevant methods such as expert judgment and analogy
- Task dependencies can be established and potential opportunities for completing tasks concurrently and sequentially can be identified and documented using a Gantt chart, for example.

Software Effort Estimation

- After sizing the product (functional size measurement), the effort to be expended shall be estimated in person months (or hours) using the size and additional estimation parameters
- The estimated effort is distributed across the phases of the project as the basis for project planning and scheduling
- For the total project plan, an estimate must also be made for the effort for project management and quality assurance
- Often the project management and quality assurance efforts are overlooked or forgotten, and this leads to severe miscalculations and underestimating

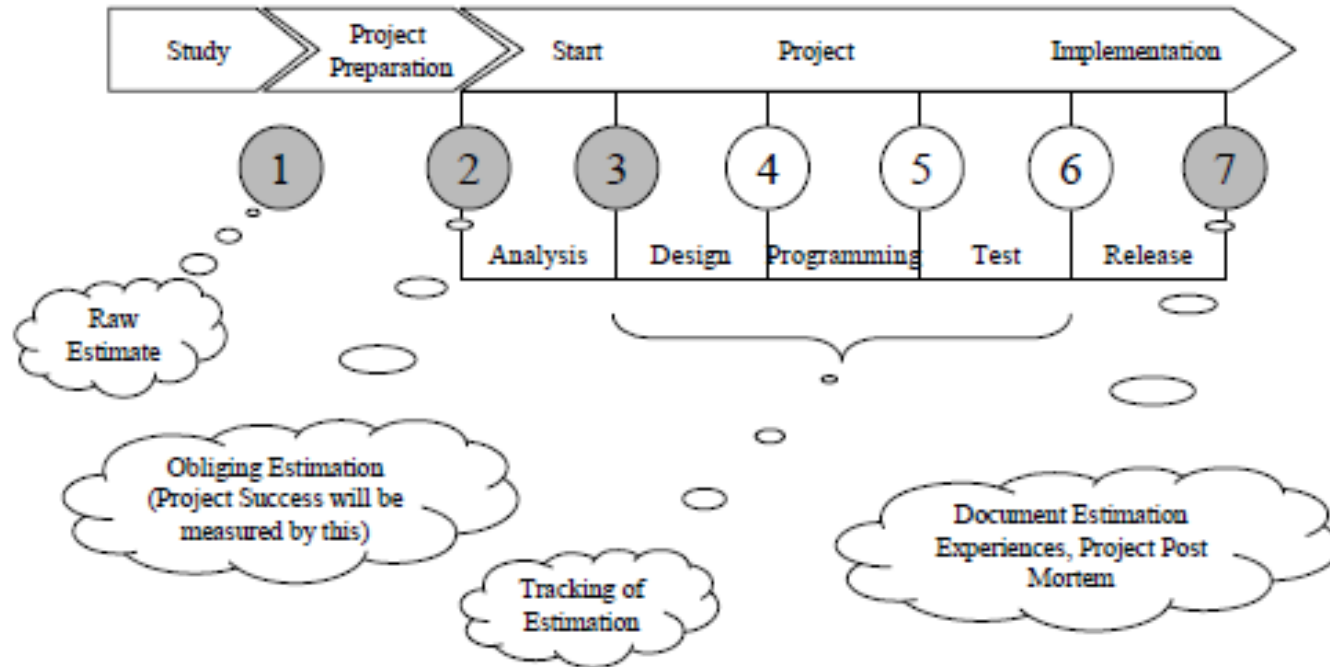
Schedule

- Determine the start and finish dates for project activities.
- If the start and finish dates are not realistic, the project is unlikely to be finished as scheduled.
- The schedule development process must often be iterated (along with the processes that provide inputs, especially duration estimating and cost estimating) prior to determination of the project schedule.

Software Schedule Estimation

- Early and precise estimations are necessary and desirable by software clients
- However, early estimations are usually imprecise and prone to a high degree of uncertainty
- Estimates must be updated and revised whenever important influencing factors change
- If changes are not considered and actualized, the plan made from the original estimation will never be met

Software Schedule Estimation



Suggestion for Milestones for IT project estimation
 (Bundschuh & Dekkers, 2008)

Estimate is ...

- An assessment of the likely quantitative result
- An approximate judgment of the effort, cost & time to perform a specified piece of work
- A quotation for a piece of work that cannot be exceeded without notice given
- A document containing resources & consumption:
 - **What resources require?**
 - **Who will supply them?**
 - **When are they needed?**
 - **How long they are needed?**

Estimate is not ...

- Account strategy
- Pricing
- Investment marketing
- Client satisfaction
- Software or tools
- Finding the fastest way

Conclusion on Estimation

- Estimation contributes to making process improvements measurable
- Estimate as much as possible
- Estimation must be used cautiously
- Implement several techniques
- Most important: make use of historical data

Discussion #1

- Detail out four (4) items that you think we have to estimate when preparing for a software project.

Discussion #2

- If it takes 5 people 8 hours to build a wall, how many hours will it take for 10 people to build the same wall?

