Lecture Slides for
Managing and Leading Software Projects

Chapter 4: Plans and Planning

developed by

Richard E. (Dick) Fairley, Ph.D.

to accompany the text

Managing and Leading Software Projects

published by Wiley, 2009

Chapter 4 Contents

- The Planning Process
- The CMMI-DEV-v1.2 Process Area for Project Planning
- Planning Agile Projects
- Balancing Agility and Discipline
- A Minimal Project Plan
- A Template for Software Project Management Plans
- Techniques for Preparing a Project Plan
- Incremental Planning

Additional Information (1)

- Appendix 4A to Chapter 4 provides an introduction to elements of the following frameworks, standards, and guidelines that are concerned with project plans and planning:
  - the SEI Capability Maturity Model® Integration CMMI-DEV-v1.2,
  - ISO/IEC and IEEE/EIA Standards 12207,
  - IEEE/EIA Standard 1058, and
  - the Project Management Body of Knowledge (PMBOK®).

Additional Information (2)

- Terms used in Chapter 4 and throughout the text are defined in Appendix A
- These presentation slides and other supporting material are available at the URL listed in the Preface to the textbook

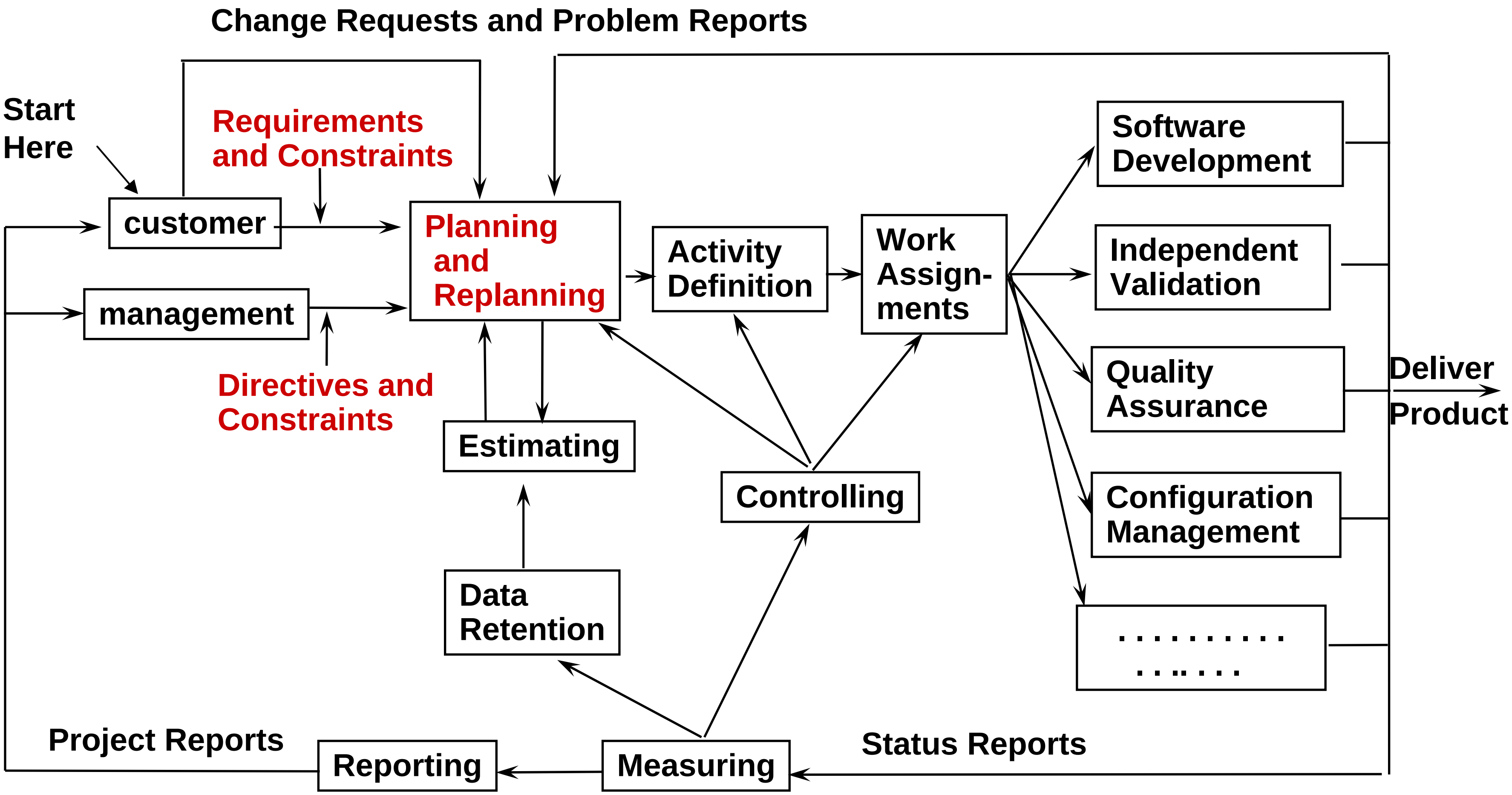
Objectives for Chapter 4

- After reading this chapter and completing the exercises, you should understand:
  - the planning process for software projects
  - the Project Planning process area of CMMI-DEV-v1.2
  - an approach to planning agile projects
  - a template for software project management plans (SPMPs)
  - tailoring the SPMP template
  - techniques for preparing a SPMP

Foundation Elements for Software Projects

Product Foundations	Explanation
Operational requirements	External, user view of the system
System requirements and architecture	Hardware, software, people elements
Design constraints	Predetermined design decisions
Software requirements	Internal technical specifications
Process Foundations	
Workflow model	Managerial work activities and work products
Development model	Technical work activities and work products
Contractual agreement	Statement of understanding between developer and acquirer
Project plan	The project roadmap

A Workflow Model for Software Projects



Requirements and Plans

- The requirements are the product specifications
- The project plan is the process specification
- They are separate, cross-referenced documents

- We can’t make a plan for making the product if we don’t know what product to make
- If we don’t know what product to make, we need a plan to figure out what product to make

Q: why should they be separate documents?

Why Is Planning Not Adequately Done?

- (Apparent) lack of time
- Lack of skills and tools
- Lack of organizational support
- Lack of information:
  1. insufficient understanding of the project
     - inadequate requirements analysis
     - novelty of the project
  2. insufficient historical data for planning
- Frequently heard excuses:
  - “why plan, when everything will change anyway?”
  - “excessive planning indicates a lack of confidence”
  - “I’m a doer, not a planner”

Why is a *Documented* Project Plan Important?

- To confirm and document breadth and depth of planning
- To assess consistency of cost, schedule, and resource estimates
- To determine the relationships among supporting plans
- To provide a vehicle for trade studies and tradeoff negotiations
- To communicate the plan, and changes to the plan, to the project stakeholders

Basic Elements of a Plan

- According to IEEE Standard 12207.1, every kind of plan, whether it is a project plan, a configuration management plan, a quality assurance plan, a training plan, or other kind of plan should contain the following information:
  - needs to be satisfied by executing the plan
  - success criteria for the planned work activities
  - work activities to be accomplished
  - schedule, budget, and resources
  - quality control measures
  - change procedures and tracking of project history
  - interfaces to relevant stakeholders
  - roles to be played
  - responsibilities and authorities
  - resource acquisition plan
  - skills acquisition plan, as needed

Planning Activities

- Tables 4.1a and 4.1b in Chapter 4 itemize
  - preplanning and planning activities for software projects

Specific Goals and Practices of the CMMI-DEV-v.12 Project Planning Process

- SG 1 Establish Estimates
  - SP 1.1 Estimate the Scope of the Project
  - SP 1.2 Establish Estimates of Work Product and Task Attributes
  - SP 1.3 Define Project Lifecycle
  - SP 1.4 Determine Estimates of Effort and Cost
- SG 2 Develop a Project Plan
  - SP 2.1 Establish the Budget and Schedule
  - SP 2.2 Identify Project Risks
  - SP 2.3 Plan for Data Management
  - SP 2.4 Plan for Project Resources
  - SP 2.5 Plan for Needed Knowledge and Skills
  - SP 2.6 Plan Stakeholder Involvement
  - SP 2.7 Establish the Project Plan
- SG 3 Obtain Commitment to the Plan
  - SP 3.1 Review Plans That Affect the Project
  - SP 3.2 Reconcile Work and Resource Levels
  - SP 3.3 Obtain Plan Commitment

Specific Goals and Practices of the CMMI-DEV-v1.2 Project Planning Process Area

SG 1 Establish Estimates	Specific Practices	Chapters of this text
SP 1.1-1	Estimate the Scope of the Project	Chapter 3
SP 1.2-1	Establish Estimates of Work Product and Task Attributes	Chapter 5
SP 1.3-1	Define Project Life Cycle	Chapter 2
SP 1.4-1	Determine Estimates of Effort and Cost	Chapter 6
SG 2 Develop a Project Plan		
SP 2.1-1	Establish the Budget and Schedule	Chapter 6
SP 2.2-1	Identify Project Risks	Chapter 9
SP 2.3-1	Plan for Data Management	Chapters 7 & 8
SP 2.4-1	Plan for Project Resources	Chapter 5
SP 2.5-1	Plan for Needed Knowledge and Skills	Chapter 5
SP 2.6-1	Plan Stakeholder Involvement	Chapter 2
SP 2.7-1	Establish the Project Plan	Chapter 4
SG 3 Obtain Commitment		
SP 3.1-1	Review Plans that Affect the Project	Chapter 4
SP 3.2-1	Reconcile Work and Resource Levels	Chapter 6
SP 3.3-1	Obtain Plan Commitment	Chapter 4

Project Planning SP1.3: Choosing a Development Model

- Development of a user-intensive system may require prototyping to clarify the operational requirements and to provide information for design of the user interface.
- Development of the software for an embedded system may require the participation of you and your technical leader on the system engineering team.
- Development of staged delivery of system capabilities based on stable requirements and a stable architecture may indicate that an incremental build strategy is appropriate.
- Development of a first-of-its-kind system may require an evolutionary development strategy.
- An agile process may be appropriate for development and on-going enhancement of a Web-based application or in cases where the requirements are evolving or changing rapidly.

Development models are presented in Chapter 2

Related Process Areas

- CMMI-DEV-v1.2 process areas related to project planning include:
  - requirements development,
  - requirements management,
  - risk management, and
  - the technical solution process areas.

Plan-driven Planning

- The 12207 standard and CMMI-DEV-v1.2 specify a “plan-driven” approach to planning and managing software projects
- The plan-driven approach to project planning is appropriate in two situations:
  1. when there is a formal contractual agreement between an acquirer and a supplier and/or:
  2. for large, complex projects internal to an organization.

Planning Agile Projects (1)

- An agile approach may be appropriate for small projects (say, 10 or fewer software developers)
  - o when formal contractual conditions do not apply
  - o and in cases where the requirements are evolving or changing on a continuing basis
    - and frequent delivery of evolving capabilities are to be delivered to users;
      - as for example, in an internet-based application.
- The agile development process is described in Section 2.6.3 of the text.

Planning Agile Projects (2)

- Planning an agile project involves:
  - Developing the product vision
  - Determining the project duration and level of effort
  - Obtaining commitment of a knowledgeable customer representative
  - Adopting a version of agile development
  - Establishing the development environment
  - Planning the frequency of iterations
  - Planning the frequency of deliveries
  - Establishing a design metaphor
  - Planning for on-going reviews by the stakeholders
  - Planning for periodic reviews of project status
  - Conducting an initial risk assessment and risk mitigation
  - Planning for on-going risk assessments and mitigation activities

A Minimal Project Plan

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

The 1058 Template

- Tables 4.4a, b, and c in Chapter 4 of the text present a template for Software Project Management Plans (SPMPs) based on IEEE Standard 1058
- An annotated version of IEEE Standard 1058 is presented in Appendix 4B to Chapter 4
  - an electronic copy of the annotated standard is available at the URL listed in the Preface to the textbook

Tailoring of Plans

- Small, simple projects have small, simple plans
- Large, complex projects have large, complex plans
- The effort devoted to planning should be determined as an issue of risk management
  - what is the risk if we do not do enough planning?
  - what is the risk if we spend too much time planning
    - and not enough time executing the plan?

templates and frameworks should be tailored to fit the needs of each project

Format of a Tailored Project Plan based on IEEE Std 1058

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

The Roles of Assumptions, Constraints, and Uncertainty

- An *assumption* is a statement of fact that is taken to be true
    - e.g., our best developers will be available to work on this project
    - e.g., the customer won't make "very many" changes to the requirements
- A *constraint* is an externally imposed condition
    - e.g., the system must delivered in six months
    - e.g., the system must run on both Intel and Motorola processors
- *Uncertainty* results from a lack of information
    - e.g., inadequate understanding of the requirements

false assumptions, excessive constraints, and uncertainty are major sources of risks and problems for software projects

Dealing With Uncertainty

- It is impossible, and undesirable even if possible, to make detailed plans for the entire project at the beginning of a software project
  - o because of what we don't know
  - o because what we know will change
- Therefore, we should adopt a "rolling wave" approach to planning
  - o each month we elaborate a detailed plan for the parts we understand
    - *at least* for the coming month
- The detailed plan is updated on a monthly basis and as event dictate

* updating of plans using the "rolling wave" technique is presented in Chapter 5 of the textbook

Level of Detail Criteria

- The level of detail in an *initial plan* should satisfy the following criteria:
  1. the scope of the plan includes all of the major work activities to be accomplished
  2. effort, schedule, and resources for each identified work activity can be estimated with confidence
  3. predecessor and successor activities for each work activity are specified and a schedule can be established
  4. complexities and risk factors are identified
  5. opportunities for reuse of existing components are identified

Planning Risk

- The risk to project success must be assessed for the items that are not planned in sufficient detail; for example:
  - What risks are incurred if you don’t have a process for managing changes to requirements?
  - What risks are incurred if you don’t have a process for assessing the impact of changes to requirements, cost, schedule, or technology?
  - What risks are incurred if you don’t have a schedule with objective milestones?
  - What risks are incurred if you don’t have a process for measuring effort and defects?
  - What risks are incurred if you don’t practice risk management?

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

Plans for Supporting Processes (1)

- The eight supporting processes included in the 12207 standards are:
  - Configuration Management
  - Verification and Validation
  - Documentation
  - Quality Assurance
  - Reviews and Audits
  - Problem Resolution
  - Subcontractor Management
  - Process Improvement

Kinds of Supporting Plans

- The nature of, and types of supporting processes required may vary from project to project
- However, the absence of
  - a configuration management plan,
  - verification and validation plan,
  - documentation plan,
  - quality assurance plan,
  - joint customer-developer review plan, or
  - problem resolution plan

  should be explicitly justified in any software project management plan that does not include them.

The Configuration Management Plan

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

The V&V Plan

- A Verification and Validation plan addresses these issues:
  - who will do verification and validation (V&V),
  - scope of activities that will be included,
  - methods, tools, and techniques that will be used,
  - the degree of independence between the development entities and the independent V&V entities of the project,
  - automated tools to be used for V&V, and
  - how interactions with an Independent V&V (IV&V) organization will be coordinated, if applicable.
- Verification planning should result in plans for activities such as traceability, milestone reviews, progress reviews, peer reviews, prototyping, simulation, and modeling.
- Validation planning should result in plans for activities such as testing, demonstration, analysis, and inspection.

The Documentation Plan

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

The Quality Assurance Plan

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

note: QA (SQA) is ***not*** testing

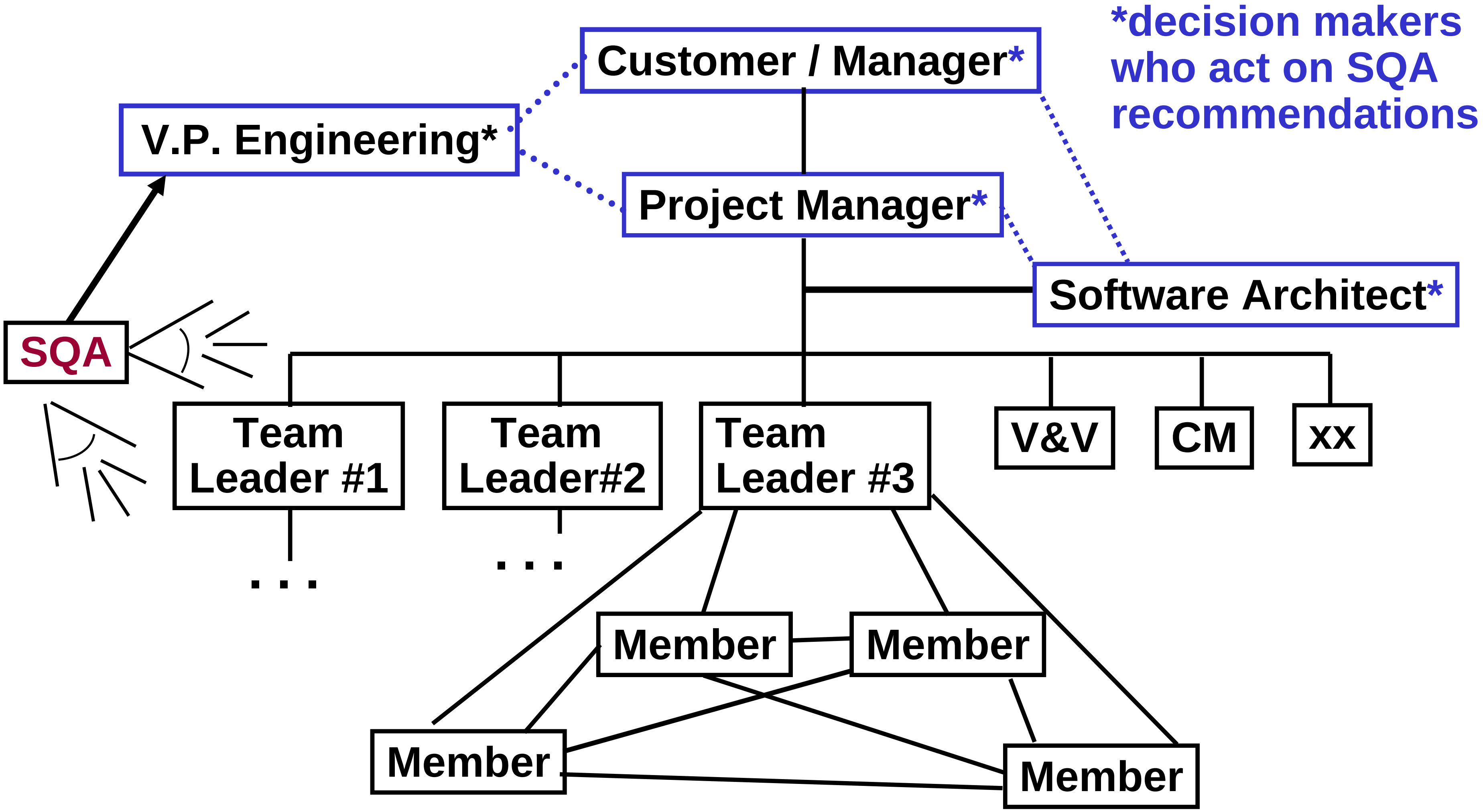
A Note on QA (1)

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

A Note on QA (2)

- The purpose of quality assurance is to assure adherence to standards, guidelines, practices, and procedure for both:
  - process and product
- Quality assurance of process and product is an oversight activity performed by individuals who have a manager and a reporting channel that are independent of the project manager
- Quality assurance must not be confused with V&V
  - V&V activities, like all other project activities are subject to QA audits and reviews

An Organizational Model
for Software Quality Assurance



*decision makers who act on SQA recommendations

The Reviews and Audits Plan

- A plan for Reviews and Audits documents:
  - the kinds of reviews and audits that will be conducted,
  - who will conduct them,
  - schedules, resources, methods, and procedures that will be used to conduct project reviews and project audits.
- This plan should include plans for joint customer-developer reviews, management reviews, developer peer reviews, quality assurance audits, and customer audits.
- Elements of this plan should be consistent with organizational policies, the project’s contractual agreement, and other contractual documents.

The Problem Resolution Plan

- According to the 12207 Standards a Problem Resolution Plan should indicate:
  - how problems in the work processes and work products will be reported, analyzed, prioritized, and resolved,
  - how problems will be tracked to closure,
  - the roles of organizational entities such as development, configuration management, the change control board, verification and validation, and quality assurance in problem resolution,
  - how the relationship between problem resolution and risk management will be managed, and
  - how effort devoted to problem reporting, analysis, and resolution will be separately reported so that rework can be tracked and needed process improvements identified.

The Subcontractor Management Plan (1)

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

The Subcontractor Management Plan (2)

- Plans for subcontractor management should include the items necessary to ensure successful completion of each subcontract.
- In particular plans for:
  - requirements management,
  - monitoring of technical progress,
  - schedule and budget reporting,
  - product acceptance criteria, and
  - risk management procedures

  should be included in each subcontractor plan.

The Process Improvement Plan

- A plan for Process Improvement documents:
  - the frequency of assessment to determine areas for improvement,
  - who will do the project assessments,
  - who will develop and implement specific improvement plans, and
  - who will implement improvement plans.
- The process improvement plan should be closely related to the risk management and problem resolution plans.

process improvements that will disrupt a project should not be initiated during the project

Plans for Supporting Processes (2)

- Plans for supporting processes should be developed to a level of detail consistent with the other sections of the plan.
- In particular the plan for each supporting process plan should include:
  - roles,
  - responsibilities,
  - authorities,
  - schedule,
  - budget,
  - resource requirements,
  - risk factors, and
  - work products.

Plans for Supporting Processes (3)

- Many of the supporting plans for software projects may have organizational policies and procedures for implementing the plans
  - and may be conducted as “business as usual”
  - or may be tailored to fit the needs of individual projects

small, simple projects have small, simple plans
large, complex projects have large, complex plans

Techniques for Preparing a Project Plan

- Techniques that can reduce the effort required to prepare a project plan include:
  - tailoring of standard templates
  - including predefined elements
  - using organizational support
  - leading a project planning team

The Main Points of Chapter 4 (1)

- Operational requirements, technical specifications, and process constraints provide the basis for project planning.
- A software project management plan is a baseline-controlled written document.
  - Appendix 4B to this text provides a template for developing software project management plans based on IEEE Standard 1058; an electronic copy is available at the URL listed in the Preface to the text.
- The comprehensive template for software project management plans presented in Tables 4.a,b&c can be, and should be, tailored to fit the needs of each project, as in the example of tailoring.
- Developing a software project management plan, like all software engineering processes, is best accomplished in an iterative manner. The initial version of the plan should be updated on a periodic basis and as events require.

The Main Points of Chapter 4 (2)

- The level of effort devoted to project planning, and the level of detail in a project plan, both initially and on-going, are determined by the risk factors created by not doing more.
  - small, simple projects have small, simple plans
  - large, complex projects have large, complex plans
- The level of detail in an initial project plan should satisfy the following criteria:
  - effort, schedule, and resources for each identified work activity can be estimated with confidence;
  - predecessor and successor activities for each work activity can be determined;
  - opportunities for reuse of existing components are identified; and
  - complexities and risk factors are identified.

The Main Points of Chapter 4 (3)

- Acceptable options for obtaining a balance among effort, schedule, and requirements in your project plan include:
  - descoping the requirements,
  - increasing the quantity of resources,
  - sing more productive resources,
  - extending the schedule, and
  - combinations of these options.

The Main Points of Chapter 4 (4)

- Unacceptable options for achieving a balance among effort, schedule, and requirements include descoping the plans for:
  - measurement and control,
  - peer reviews, and
  - verification and validation;
  - and planning for overtime effort.
- SEI, ISO, IEEE, and PMI provide frameworks, standards, and guidelines for project planning
  - see Appendix 4A to Chapter 4 of the textbook