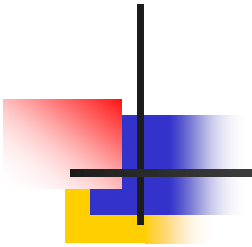
CHAPTER 3
Planning and Managing the Project

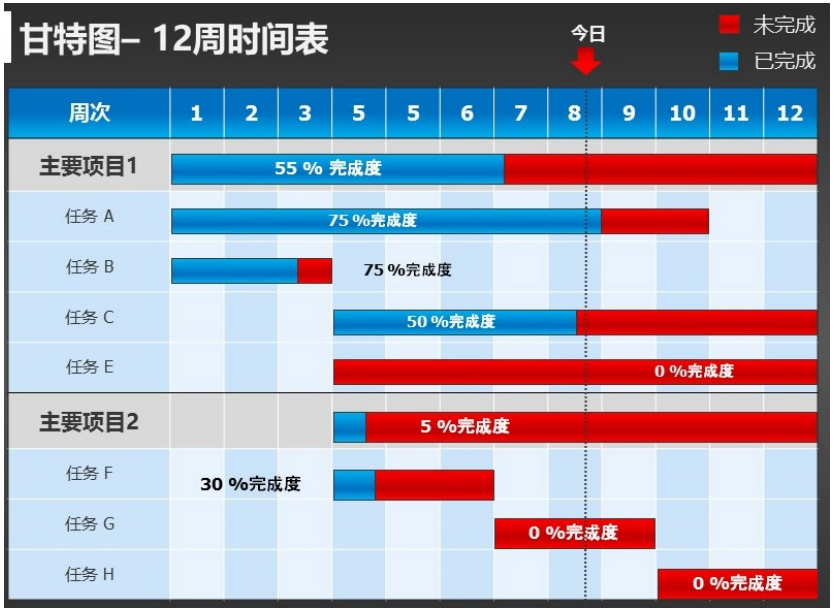
- Tracking project progress
- Project personnel and organization
- Effort and schedule estimation
- Risk management



- Project begins when customer approaches you to discuss a perceived need
- Usually, customers have several questions
  - Do you understand my **problem and needs** ?
  - Can you design a system that will **solve the problem or satisfy my needs** ?
  - **How long** will it take you to develop such a system ?
  - **How much** will it cost to have you develop such a system ?

3.1 Tracking Progress

- Answering the last 2 questions requires a well-thought project schedule



3.1.1 Project schedule

1. Project schedule 项目进度

   Describes the **development cycle** by enumerating the **phases or stages, and tasks or activities** to be done. The schedule also portrays the **interactions among the tasks** and **estimates the time each task or activity will take**.

Project schedule is a timeline from start to the end of the project, showing the activities that need to be performed: actual time to do work + weekends + holidays etc.

3.1.1 Project schedule

Processes to ensure the timely completion of the project

- Define activities
- Sequence Activities
- Estimate Activity Resources
- Estimate Activity Duration
- Develop Schedule
- Monitor and control Schedule

2. Activity 活动

It is a part of the project that takes place over a period of time;

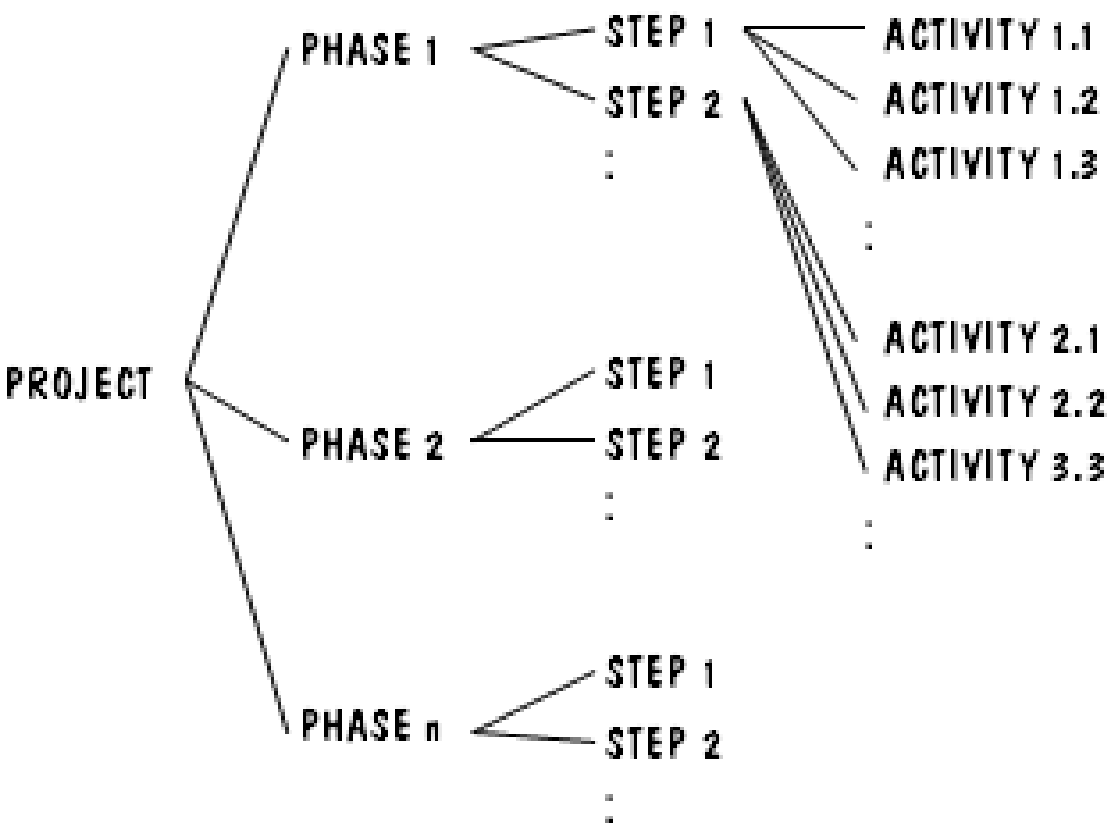
3. Milestone 里程碑

It is the completion of an activity – a particular point of time; is the end of a specially designated activity.

3.1.2 Project Schedule Construction

Development can be separated into succession of phases; Each phase is composed of steps; And each step can be further subdivided.

- See Fig 3.1, p.84.
- See Table 3.1, p.85, of phases, steps and activities of building a house.
- See Table 3.2, p.86, of milestones of Building a house.



3.1.3 Work Breakdown structure 工作分解结构 WBS

- The above analysis can lead to generating a work breakdown structure (WBS).
- WBS
  - 将所有工作任务分类的清单或表格
  - 是驱动项目的机器
  - The structure depicts the project as a set of discrete pieces of work.离散的工作单元

WBS示例：同学聚会晚宴项目

1. 参加者
   - 1.1获得同学名单
   - 1.2发邀请
   - 1.3确定参加者名单
2. 地点
   - 2.1访问可选的地点
   - 2.2与餐馆签约
   - 2.3定菜单
3. 活动
   - 3.1制定节目单
     - 3.1.1选定主持人
     - 3.1.2选定发言者
   - 3.2安排娱乐节目
     - 3.2.1选择表演者
     - 3.2.2与表演者签约
4. 材料
   - 4.1找出可选的印刷者
   - 4.2选定印刷者

……

WBS讨论

- 表达形式
  - 大纲、表格、图形等等
- 完成者
  - 项目经理、核心小组
  - 集体讨论、明确项目要求和必需的任务
- 任务的划分
  - 需要个人判断
  - 两个极端
    - 把发请柬分解为写信封和贴邮票
    - 推销员：期望、销售、退休
- WBS可复用

3.1.4 Activity Graph 活动图

But the structure gives no indication of the interdependence and concurrency of the work units. 无法表达工作单元之间的依赖和并行关系

1. Activity 活动

Each activity can be described with 4 parameters:

- Precursor 前驱活动集合
- Duration 持续时间
- Due-date 完成期限
- Endpoint 活动结束里程碑

2. Activity Graph 活动图

In an activity graph

- The nodes are the project milestones
- The lines linking the nodes represent the activities involved

Fig 3.2 is the activity graph for the work described in phase 2 of Table 3.1.

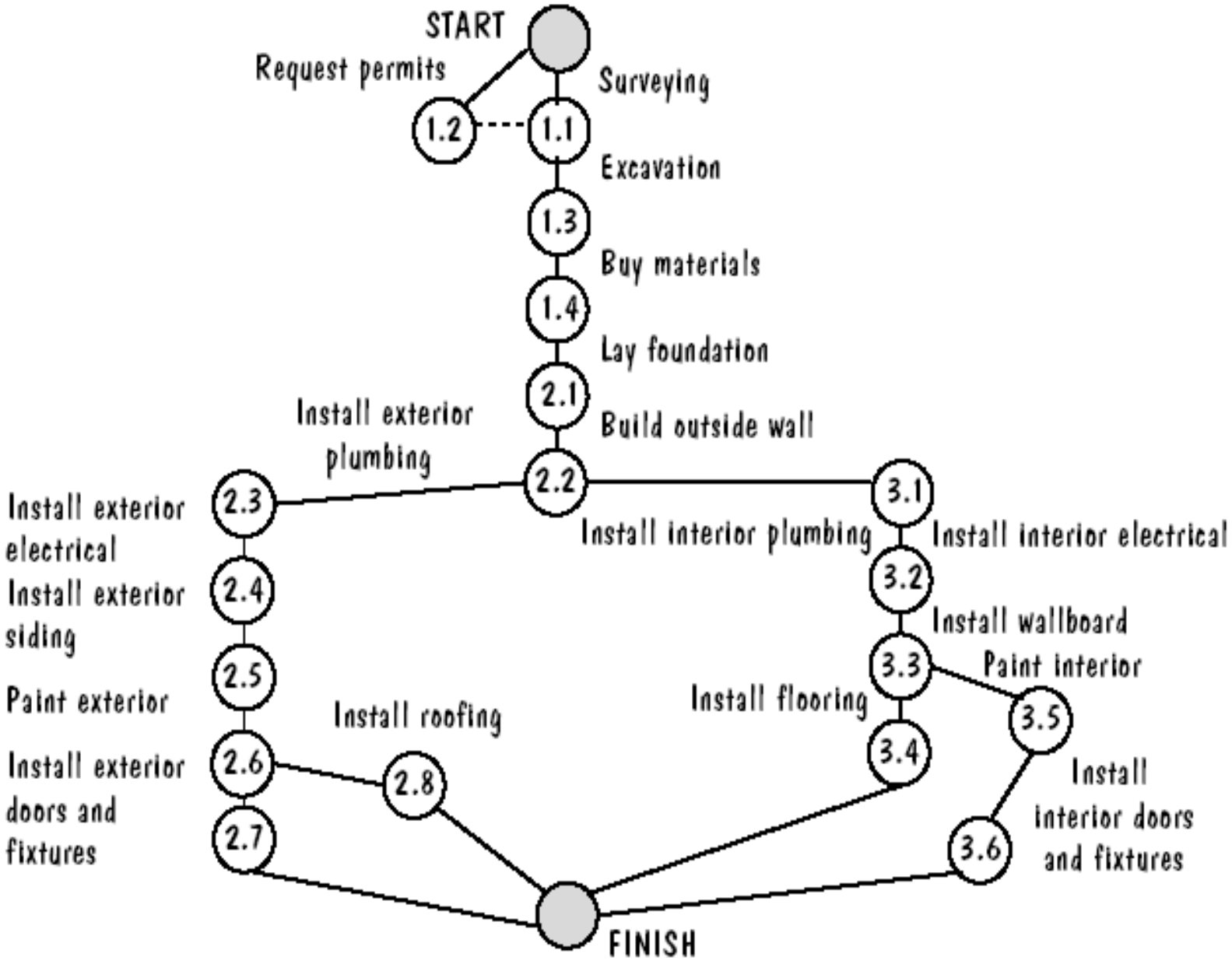


Figure 3.2 Activity graph for building a house

Characteristics of Activity Graph

- Precursors
- Dependency
- Concurrency
  - The parallel nature of tasks
- Dashed line link nodes that must all be completed to make afterward activities to start to work
- In an activity graph
  - The nodes are the project milestones 节点表示里程碑
  - The lines linking the nodes represent the activities involved 边表示活动

3.1.5 Completion estimation ---Critical Path Method (CPM) 关键路径方法

- **Activity Graphs made more useful**
  - By adding it information about the estimated time
    - It will take to complete each activity.
    - The corresponding edge of the graph is labeled with the estimated time.
    - See Fig 3.3, p.84
  - This graph tells us a lot about the project's schedule

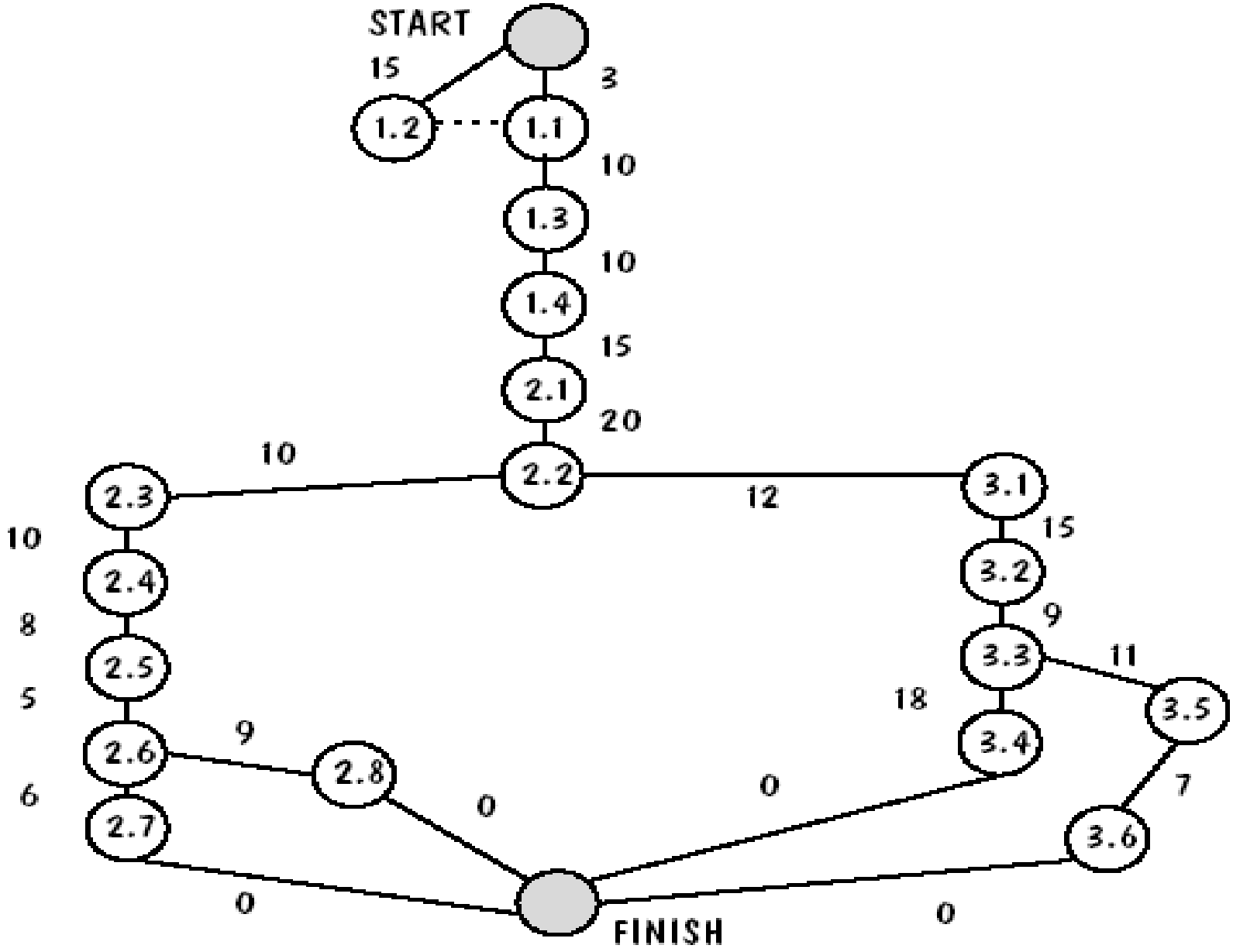


Figure 3.3 Activity graph with durations

关键路径方法（CPM）中的几个概念

- For each activity in the graph 2 times can be computed
- Real time or actual time for an activity （活动的）实际完成时间:
  - Is the estimated amount of time required for the activity to be completed.
- Available time （活动的）可用时间
  - Is the amount of time available in the schedule for the activity's completion.
  - This is the time allowable for the activity to take in order to make the project on schedule or without delaying the project.
- Slack time of float for an activity （活动的）空闲时间
  - Slack time = available time – real time

关键路径方法（CPM）中的几个概念

- Another way of looking at Slack time for an activity
  - Latest time is the time the activity may begin without delaying the project 最晚开始时间
  - Earliest time is the time the activity may begin as early as possible 最早开始时间
  - Slack time = latest time – earliest time 空闲时间

The algorithm of calculating the slack time

The earliest time: (forward pass)从起始节点开始，沿着路径向前递推计算每个活动的最早开始时间：max{前驱活动A_i的最早开始时间 + 活动A_i的持续时间}

We can work forward from the starting activities along the path to compute the earliest time of every activity.

The latest time: (backward pass)从终止节点开始，沿着路径反向递推计算每个活动的最晚开始时间：min{后继活动Ai的最晚开始时间} – 当前活动的持续时间

Then we can work backward along the path to compute the latest time of every activity..

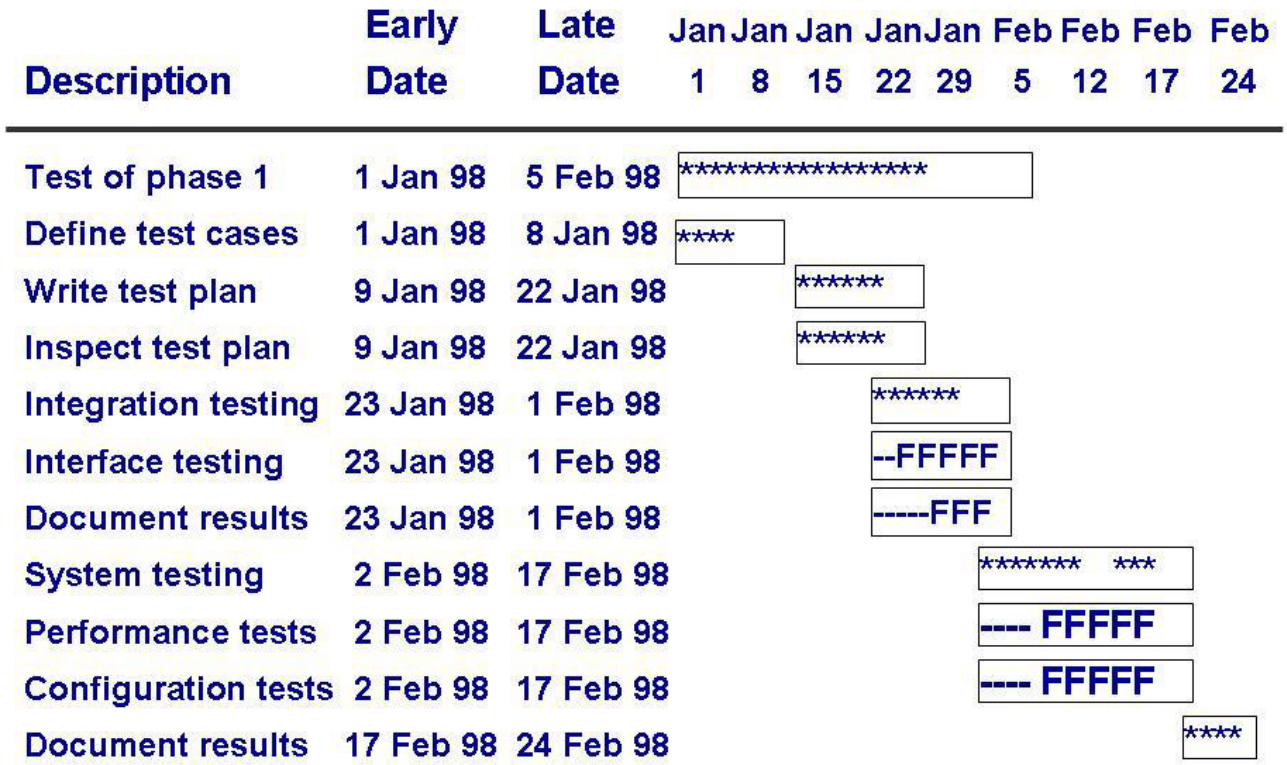
Critical Path 关键路径

The longest path has a slack of zero for each of its nodes because it is the path that determines whether the project is on schedule. This is called the Critical Path.

- Critical Path is the one for which the slack at every node is zero. That is, there is no margin for error when performing activities along a Critical Path.
- There may be more than one critical path.
- The late start pushes
  - all subsequent critical path activities forward, forcing them to be late, too.
  - all subsequent activities not on a critical path forward, forcing them to loss slack time.
- In this way, activity graph helps us to understand the impact of any schedule slippage.

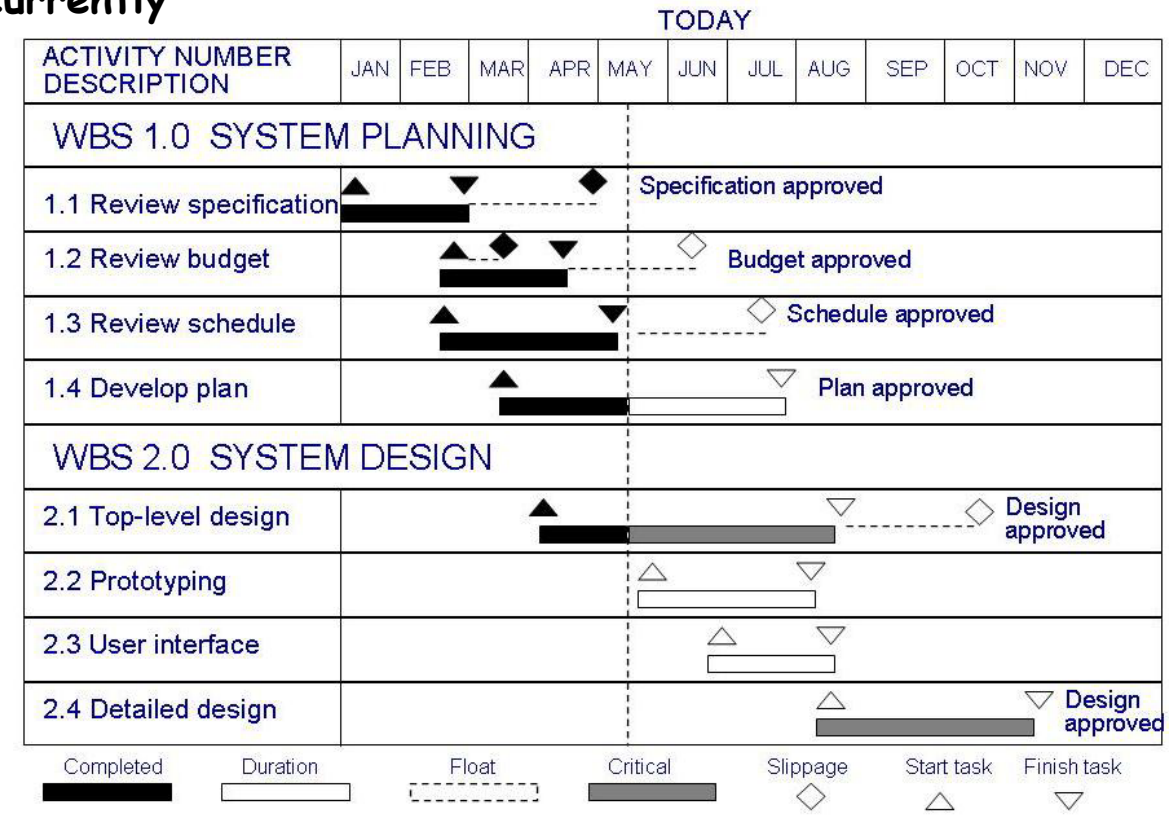
CPM bar chart

Horizontal bars represent the duration of each activity.
Bars composed of asterisks indicate the critical path.
Each activity has associated with it an expected value and a variance of duration.

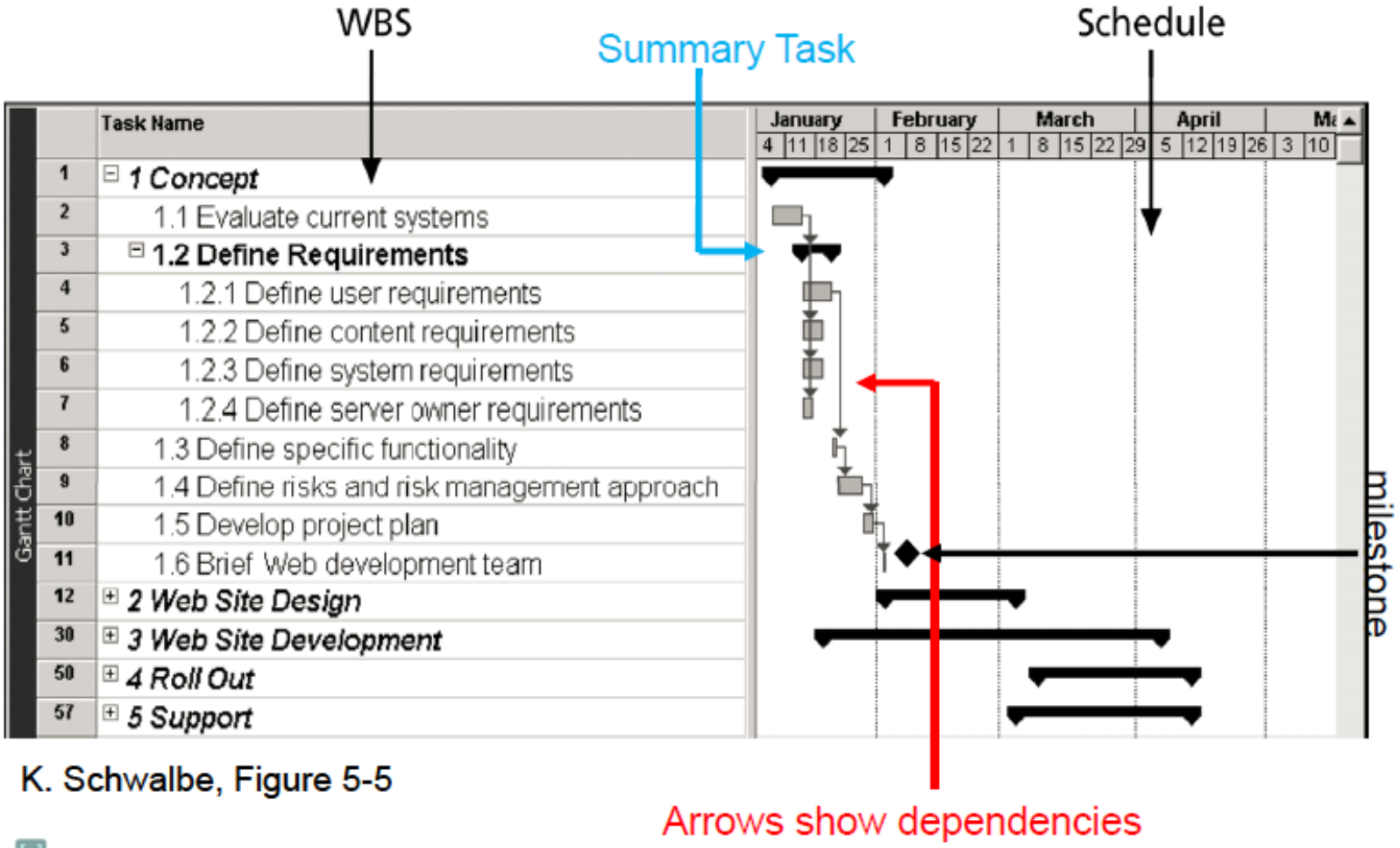


3.1.6 Tools to Track Progress

- Manual tools
- Spreadsheet tools
- Sophisticated tools with complex graphics
- E.g. Gantt Chart
  - Activities shown in parallel helps understand which activities can be performed concurrently



3.1.6 Tools to Track Progress



K. Schwalbe, Figure 5-5

3.2 Project Personnel 项目中的人员管理

团队是由若干人组成的一个群体，他们具有互补的技能，对一个共同目的、绩效目标及方法做出承诺并彼此负责。

3.2.1 Staff Roles and Characteristics
团队中的角色和人员特征

有效的项目管理集中于三个P上：

- 人员（people）
- 问题（problem）
- 过程（process）

- Players (项目参与者)
- Team leaders (项目负责人)
- Software team (软件项目组)

其顺序不是任意的。任何管理者如果忘记了软件工程是人的智力密集的劳动，他就永远不可能在项目管理上取得成功；任何管理者如果在项目开发早期没有支持有效的用户通信，他有可能为错误的问题构造一个不错的解决方案。最后，对过程不在意的管理者可能冒把有效的技术和工具插入到真空中的危险。

- Staff Roles 团队中的角色
  - Key project activities likely include
    - Personnel for requirements, system and program design, program implementation and testing, training, maintenance, quality assurance
  - The assignment of staff to tasks
    - It depends on project size, staff expertise and staff experience
    - Assigning different responsibilities to different set of people. E.g. it is useful for program designer to be different from system designer

Staff Characteristics 人员特征 /Choosing Personnel

- Ability to perform the work 工作能力
- Interest in the work 兴趣
- Experience
  - with similar applications 类似应用的经验
  - with similar tools, languages or techniques 工具,语言,技术
  - with similar development environment 类似开发环境的经验
- Training 培训情况
- Ability to communicate with others 交流能力
- Ability to share responsibility with others 承担责任的能力
- Management skills 管理技能

Characteristic differences

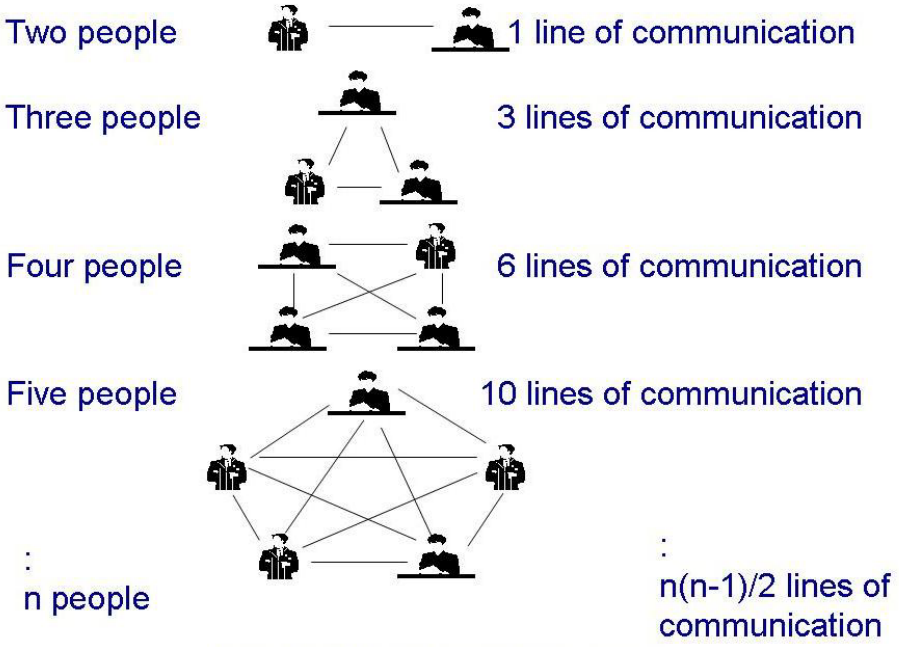
- These characteristics can affect an individual's **ability**
- Some are good at viewing "the big picture"
  - But may not enjoy focusing on detail
  - These people may be better suited to system design or testing than to program design or coding
- Sometimes, ability is related to comfort
  - Some feel more comfortable or confident in doing something
  - The feeling of comfort is important : people are usually more productive

Characteristic differences

- **Interest** in the work can also determine someone's ability or success on a project 兴趣
  - Though very good at doing something, some may be more interested in trying something new than repeating something /The novelty of the work 创新性工作兴趣
  - On the other hand, Some people prefer doing what they know and do best, rather than venturing int o new territory 已熟悉
  - It is important that whoever is chosen for a task be excited about performing it, no matter what the reason is.

Characteristic differences

- Individual ability in **communication** 沟通、交流
  - Project activity needs communication with one another, with users and customers
  - Project's progress is affected not only by the degree of communication, also by ability of individuals to communicate their ideas
  - How quickly the lines of communication can grow
    - With *n* people, there are *n(n-1)/2* communication lines



3.2.2 Work Styles 工作方式

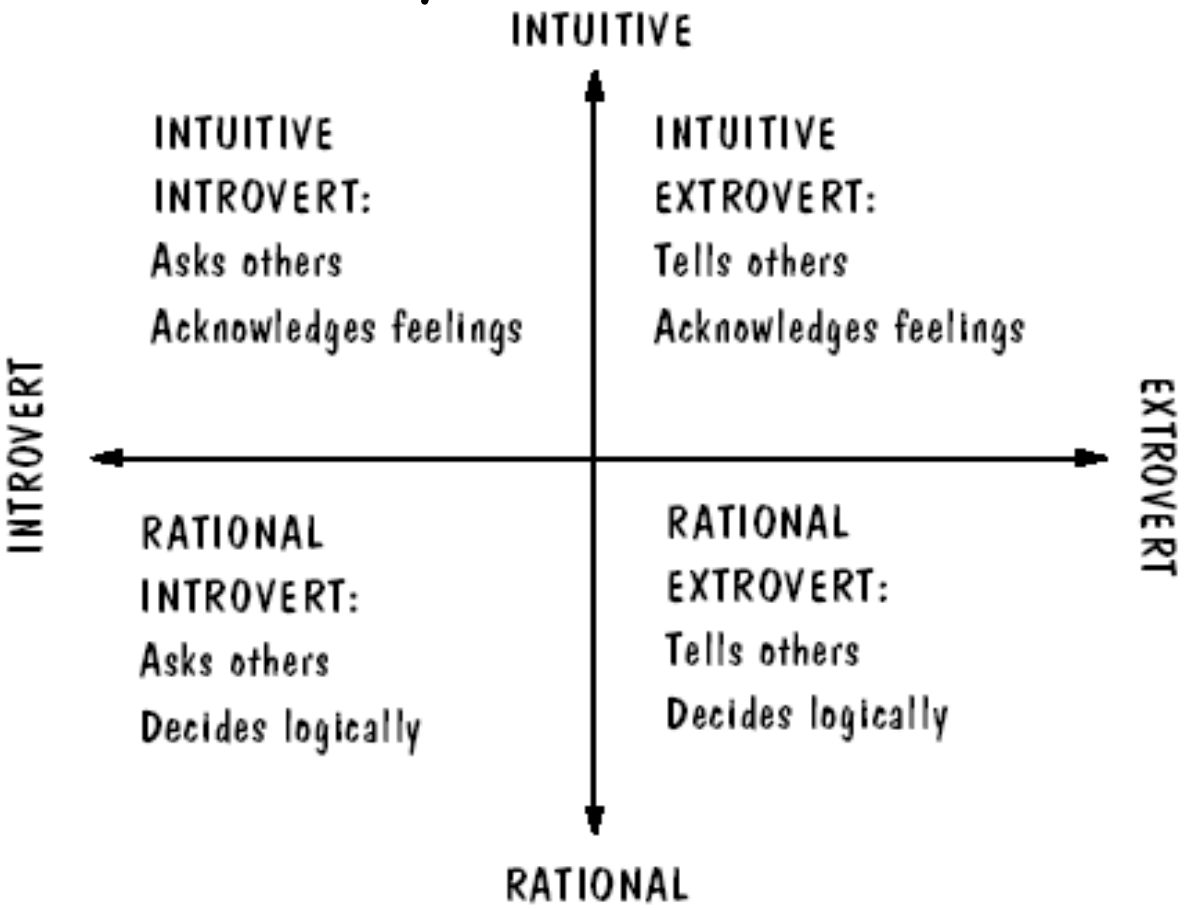
- Different people have different preferred styles 不同的工作方式
  - For interacting with others 交流、沟通方式
  - And understanding problems 理解问题的方式
  - Example
    - Some prefer to do detailed analysis of all possible information before making decisions 周密分析后作决定
    - Some may rely on "gut feeling" for most of their important decisions 凭借敏感的直觉作决定

Work Styles 工作方式

- **Extroverts** and **Introverts**：communication styles 从交流方式上划分：外向和内向
  - When communication with others, some people tell others their thoughts and others ask for suggestions from others for forming an opinion
  - The former is Extroverts
  - The latter is Introverts
- **Intuitive** and **rational**：decision styles 从决策方式上划分：直觉和理性
  - Some people base their decision on feelings and emotional reactions to a problem
  - Some people base their decision on carefully examining and considering all options

Work Styles 工作方式

- Horizontal axis: communication styles
- Vertical axis: decision styles



Working with Work styles

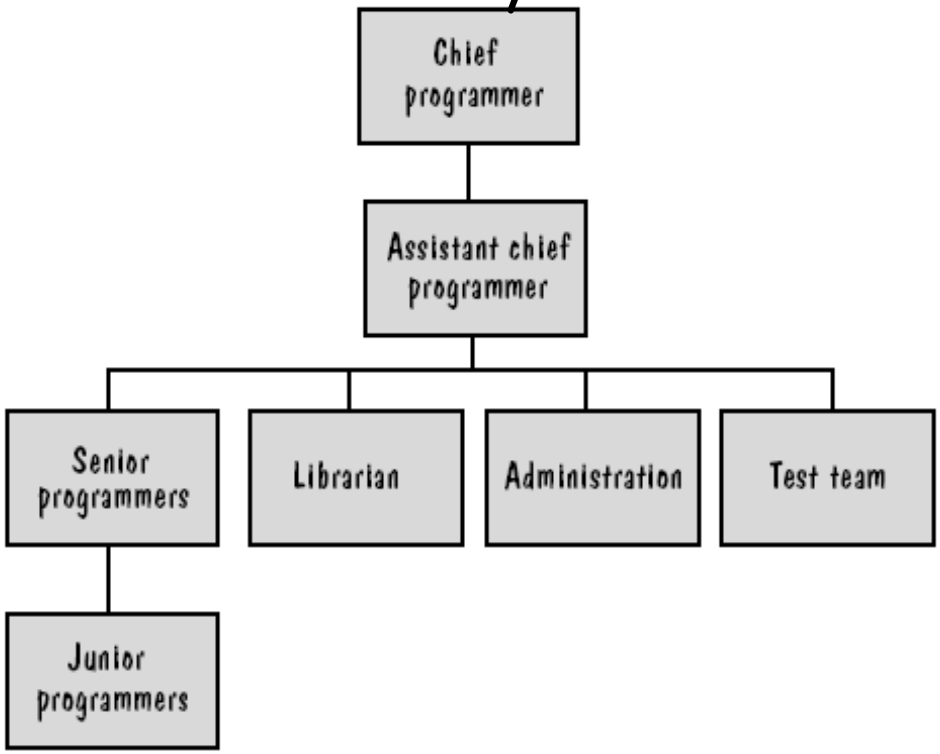
- **Communication** is critical to project success, and **work style** determine the communication style.
- Understanding work style can help you to be flexible in your approach to others
  - Work styles gives you information about the priorities of others
    - To introverts, give detailed presentation
    - To extroverts, include questions allow him to tell you what he wants and needs
    - To intuitive, take advantage of his creativity by soliciting new ideas from him
    - To rational, give facts or figures, rather than judgments or feelings
  - Work style affect interaction among customers, users and developers
  - Work style can also involve choice of workers for a given task

3.2.3 Project Organization 项目组织

- The choice of an appropriate structure depends on several things :选择项目组织结构需要考虑的因素
  - The background and work styles of team members
  - The number of people on the team
  - The management styles of the customer and developers
- Examples:
  - Chief programmer team: one person totally responsible for a system's design and development
  - Egoless approach: hold everyone equally responsible

Chief Programmer Team 核心程序员队伍

- An popular organizational structure
  - The chief programmer supervises all others, designs all problems, and assign the code development to the other team members.
  - Each team member must communicate often with chief, but not necessarily with other team members



Egoless Approach 平等、民主的程序员队伍

- Egoless approach
  - Every one in the team is equally responsible
  - This structure is democratic, all tem members vote on a decision, whether they concern design or testing tasks.
- The 2 described before represent extremes
  核心组织和平等组织是两个极端情况
  - There are many other ways

The Software Team

软件工程小组的组织方式：

- 民主分权式（Democratic Decentralized, DD）：这种软件工程小组没有固定的负责人。任务协调者是短期指定的，之后就由其他协调不同任务的人取代。问题和解决方法的确定是由小组讨论决策的。
- 控制分权式（Controlled Decentralized, CD）：这种软件工程小组有一个固定的负责人，他协调特定的任务及负责子任务的二级负责人关系。问题解决仍是一个群体活动，但解决方案的实现是有小组负责人在子组之间进行划分的。子组和个人间的通信是平行的，但也会发生沿着控制层产生的上下级的通信。
- 控制集权式（Controlled Centralized, CC）：顶层的问题解决和内部小组协调是由小组负责人管理的。负责人和小组成员之间的通信是上下级式的。

如何选择合适的组织方式

- Characteristics of projects and the suggested organization structure
- Different organizational structure can be combined where appropriate

Highly structured	Loosely structured
High certainty	Uncertainty
Repetition	New techniques or technology
Large projects	Small projects

3.3 Effort Estimation工作量估算

- A Crucial Aspect of Project

  成本估算是项目计划和管理的一个关键方面：

  帮助管理者建立合同；

  帮助管理者选择人员和团队组织形式。

  过高的成本估算：客户可能会放弃项目。

  过低的成本估算：软件开发机构会亏损，使项目失败。

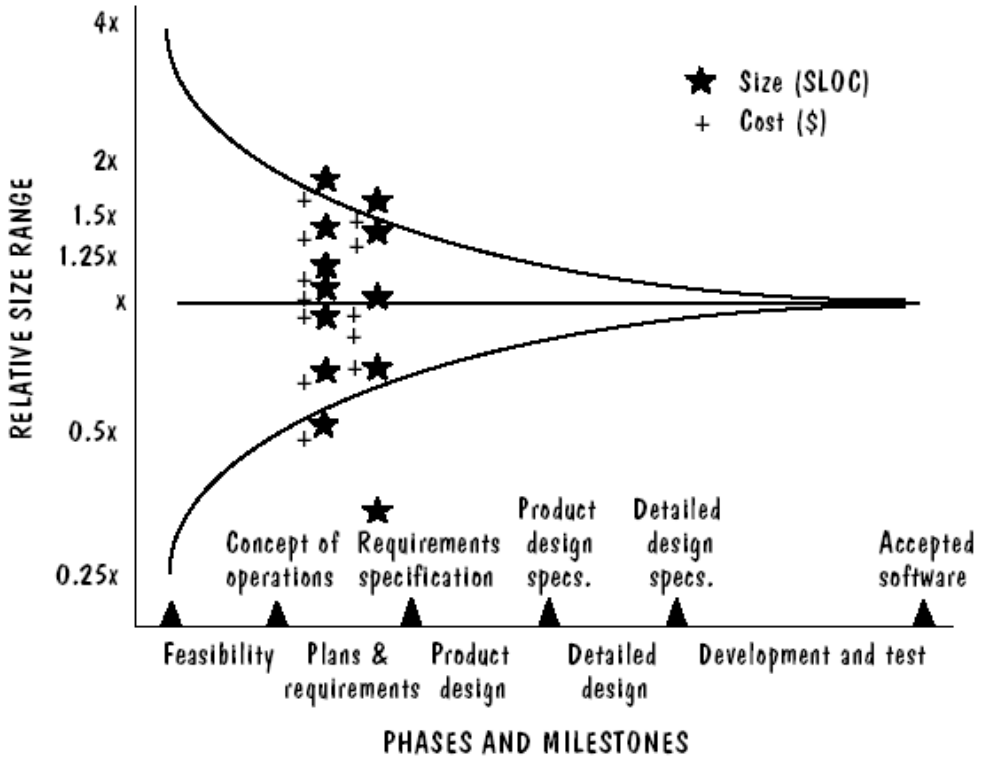
- Cost, schedule and effort estimation must be done as early as possible
  - Sometime the estimation should be done repeatedly throughout the life cycle to refine the estimation based on more completion information.

The Project budget 项目预算

- Type of costs
  - 显式成本：Facilities，staff, methods and tools
    - Facilities include hardware, space, furniture, air-conditioning, and all other items that provide the physical environment
    - This cost are well-understood and easy to estimate
  - 隐式成本：Purchasing software and tools 软件和工具
  - The biggest component of cost is effort 工作量（人-天）
    - We must determine how many of staff-days of effort will be required to complete the project
    - This is the greatest degree of uncertainty
    - When workers must communicate and consult with one another, the effort is increased by the time required for meetings, documentation and training.

Uncertainty's affect on size and cost estimation

- The earlier the estimates, the bigger the difference the estimate can differ from the eventual actual cost



- 软件成本及工作量估算永远不会是一门精确的科学；太多的变化--人员、技术、环境、策略影响了软件的最终成本及开发所需的工作量。

影响估算的因素

- 项目复杂性对计划中固有的不确定性产生重大影响
- 项目规模影响估算准确性，随着规模的增长，软件中各个元素之间的相互依赖性也迅速增加。项目规模的增长会对项目成本和进度产生几何级数的影响。
- 结构不确定性的程度也会对估算的风险产生影响。结构是指：需求能被确定的程度，功能能被分解的容易程度等
- 历史信息的可用程度。
- 为资源、成本及进度建立的定性估算中存在的不确定性。如果对项目范围理解很差或需求不断变化，不确定性及风险就会很高。
- 应该满足于事物的本性所能容许的精确度，不要去寻求绝对的准确。

3.3.1 Expert Judgment 专家判断法

1. 专家判断方法

基于类似项目经验的非形式化的成本估算方法。精确性依赖于估算者的能力、经验、客观性、预测能力等。

2. 常用方法

- Analogies 类比法
  - Asking several experts to make three predictions
  - A pessimistic one (x), An optimistic one (y), A most likely guess (z), the estimate is determined by *(x+4y+z)/6*
- Delphi technique
  - based on the average of "secret" expert judgments

3. Expert Judgment 的缺点

- Inaccuracies; It relies on expert's ability.

3.3.2 Algorithmic Methods算法估算

- Models have been created to express the relationship between effort and the factors that influence it. 工作量及其影响因素之间的关系模型
- The expression is described as equations
  - $E = (a + bS^c)\, m\,(\mathbf{X})$

    Where S : the estimated size of the system

    X : a vector of cost factors, x_i through x_n

    m : an adjustment multiplier based on these factors

    a, b, c are constants
  - Factors may cover as experience, size, application type, …

Some Algorithmic Methods

1. Walston and Felix (1977)

   Finding from IBM data from 60 projects and form

   $E = 5.25\, S^{0.91}$

   The basic equation is supplemented with a productivity index

   29 factors can affect the productivity (see Table 3.7)

   Each of the factors is

   - weighted by 1, if it increase the productivity
   - 0, if it has no effect
   - -1, if it decrease the productivity

   A weighted sum of the 29 factors was used to generate an Effort Estimation from the equation

Some Algorithmic Methods

1. Walston and Felix (1977)

1. Customer interface complexity	16. Use of design and code inspections
2. User participation in requirements definition	17. Use of top-down development
3. Customer-originated program design changes	18. Use of a chief programmer team
4. Customer experience with the application area	19. Overall complexity of code
5. Overall personnel experience	20. Complexity of application processing
6. Percentage of development programmers who participated in the design of functional specifications	21. Complexity of program flow
7. Previous experience with the operational computer	22. Overall constraints on program's design
8. Previous experience with the programming language	23. Design constraints on the program's main storage
9. Previous experience with applications of similar size and complexity	24. Design constraints on the program's timing
10. Ratio of average staff size to project duration (people per month)	25. Code for real-time or interactive operation or for execution under severe time constraints
11. Hardware under concurrent development	26. Percentage of code for delivery
12. Access to development computer open under special request	27. Code classified as nonmathematical application and input/output formatting programs
13. Access to development computer closed	28. Number of classes of items in the database per 1000 lines of code
14. Classified security environment for computer and at least 25% of programs and data	29. Number of pages of delivered documentation per 1000 lines of code
15. Use of structured programming	

Some Algorithmic Methods

2. Bailey and Basili (1981) meta-model

The model reflects your own organization's characteristics.

It used a database of 18 projects.

First, They minimized the standard error estimate and produced an equation

$E = 5.5 + 0.73\ S^{1.16}$

Then, adjust this initial estimate based on the ratio of error

R = actual effort E / predicated effort E'

R_{adj} = (p. 110)

Then, the initial effort is adjusted as E_{adj}

Finally, other factors that affect effort (see Table 3.8)

Problems with the Previous Algorithmic Methods 主要问题

Dependent on size as a key variable.

Estimates are required early, well before accurate size information is available.

So the models simply translate the effort estimation to a size estimation problem.

3. Boehm's estimation model: COCOMO II model (1990s)

- Reflect three major stages of any development将开发过程划分为3个主要阶段
  - Acknowledges that lines of code are impossible to know in the early stage
- At stage 1, plan and prototype stage计划和原型阶段
  - It estimate size in application points 通过应用点估算规模
  - This captures size in high level effort generators
    - Such as number of screens and reports,…
- At stage 2, the early design stage早期设计阶段
  - It Employs function points as a size measure 需求中的功能点
  - Estimate functionality captured in the requirements
  - This offer a richer system description than application points

COCOMO II model

- At stage 3, the post-architecture stage后体系结构阶段
  - Sizing can be done in terms of function points and lines of code, and other factors 功能点/代码行
  - This model also includes models of reuse
    - Taking into account maintenance and changes in requirements over time
- The calculation of the COCOMO II model
  - Basic equation : $E = b S^c m(X)$
  - m(X) is the cost driver information in Table 3.9

COCOMO II model

- At stage 1
  - **Application points** supply the size measure 应用点
  - Next you classify each application element as simple, medium or difficult as shown in Table 3.10 根据难度分类
  - The number to be used for simple, medium and difficult application points is a complexity weight fount in Table 3.11 获得复杂度权重
  - Then, you sum the weighted reports and screens to obtain a single application point number. 单个应用点的规模
    - If r percent of the object will be reused from previous then

      new application point = (application points) * (100-r)/100

COCOMO II model

- At stage 2
  - Based on a **function-point** calculation, the effort estimate is adjusted for degree of reuse, requirement change, and maintenance. 调整第1阶段的估算
- At stage 2 and 3
  - Cost drivers are adjustment factors expressed as effort multipliers based on rating your project experience from “extra low” to “extra high” (see Table 3.12)
    - Extra low : less than 3 months of experience
    - Very low :
    - ……
- Similarly, cost drivers of analyst capability and tool use categories are also used. (see Table 3.13)
- Tools are available

3.4 Risk Management

- Unwelcome events may occur during software development and maintenance.
  - Plans should be made to avoid these events.
  - If they are inevitable, minimize their negative consequence.
- Risk is an unwanted event that has negative consequences.风险
- Risk management is the way to understand and control the risks on their project. 风险管理
- Boehm's Top 10 Risk Items

3.4.1 Properties of Risk Event

1. Risk impact: a loss associated with the event 风险影响
   - Something negative happens to the project
   - Example: a loss of time, quality, money, control, understanding and so on.
   - Example:if requirements change dramatically after the design is done, then the project can suffer loss of control and understanding, i.e. time and money.
2. Risk probability: The likelihood that the event will occur 风险概率
   - It is estimated and measured from 0 to 1.
   - When the probability is 1, then the risk is called a problem, which means that it is certain to happen.

3. Risk control: a set of actions taken to reduce or eliminate a risk.风险控制

It describes the degree to which we can change the outcome.

We should determine what we can do to minimize or avoid the impact of the event.

4. Risk exposure 风险成本

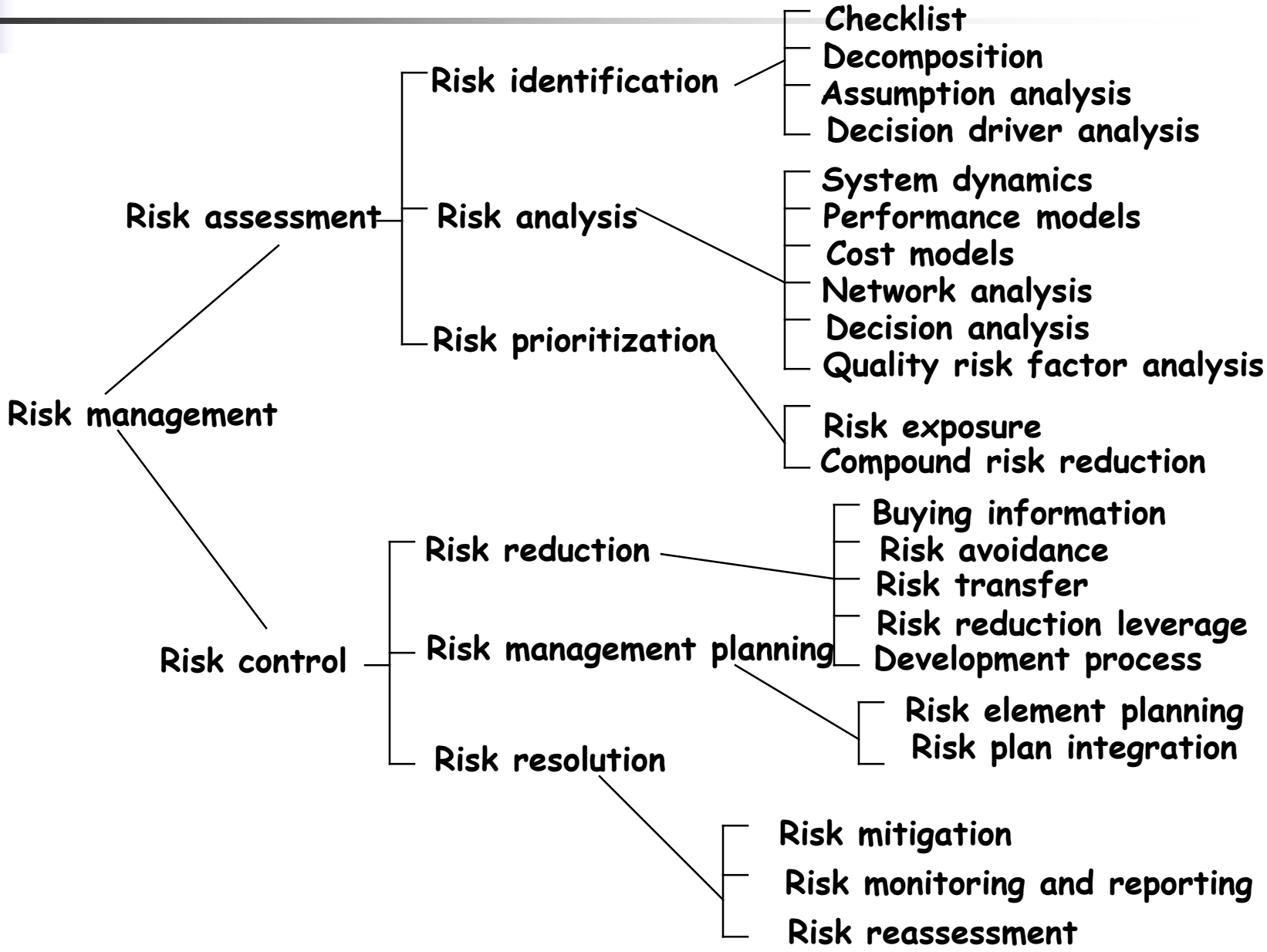
We can quantify the effects of a risk.

Risk exposure = risk impact × risk probability

Both Risk impact and Risk probability can change over time.

So, part of a manager's job is to track these changes over time and make plans for the risks.

3.4.2 Risk Management Activities



- Risk Assessment. 风险评估
  - (1) Identifying the risks 风险识别
    - Make a checklist of problems that may occur
    - By decomposing the process into small pieces and anticipate problems that may arise. Or,
    - By analyzing assumptions or decisions you are making about how the project will be done, who will do it, and with what resources. In each aspect there may involve some risks.
  - (2) Analyzing the risks 风险分析
    - Make sure you can understand as much as possible about when, why and where they might occur.
    - Many techniques can be used to enhance your understanding : system dynamics models, cost models, performance models,…

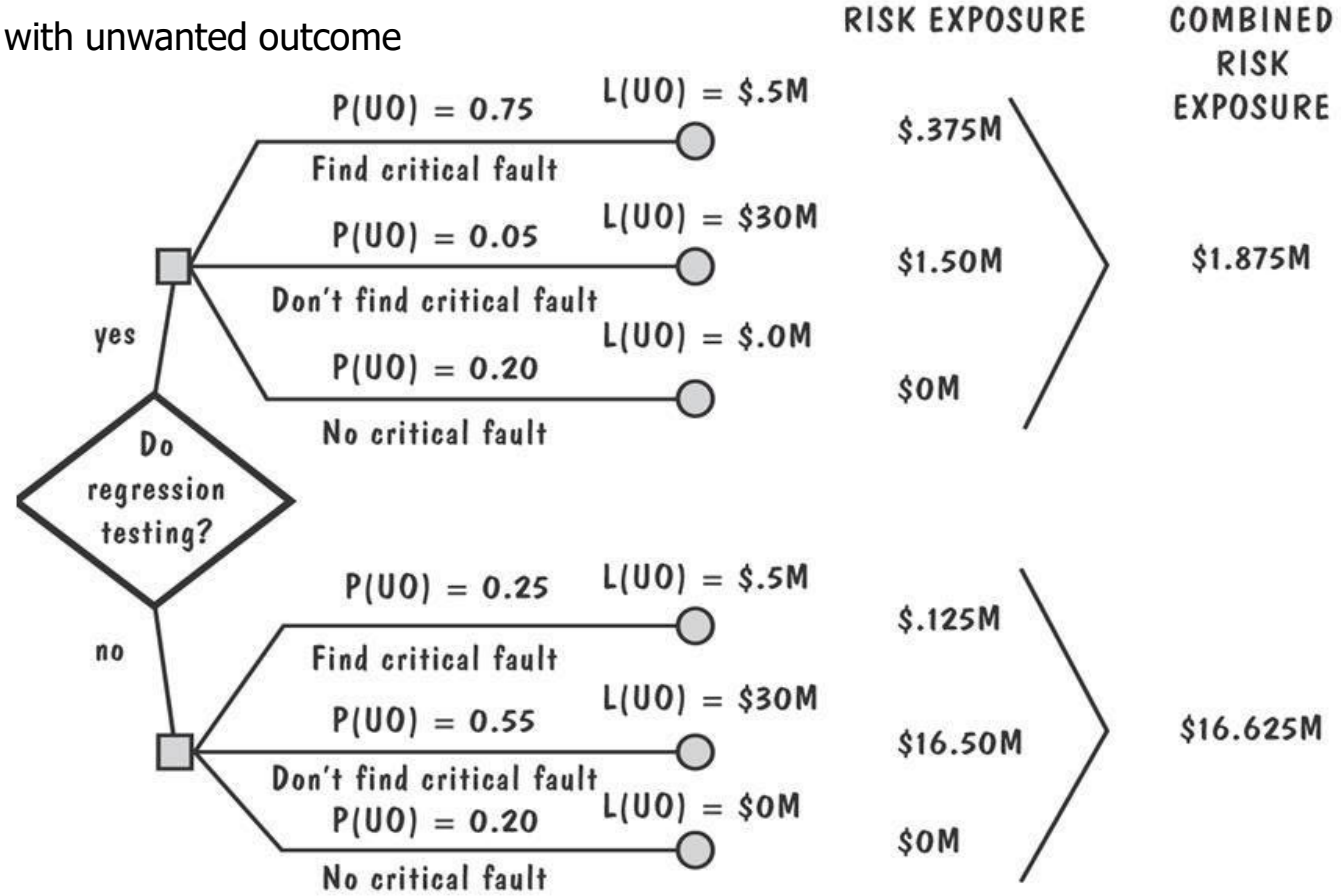
Risk Assessment 风险评估

- (3) Assigning priorities to each of the risks 划分优先级
  - Priorities are based on the risk exposure
    - Includes both the risk impact and the probability of risk occurrence
  - Priority scheme : see Fig 3.16, p.122
    - For each possible outcome, 2 quantities are estimated
    - P(UO) : Probability of an unwanted outcome
    - L(UO) : Loss associated with the unwanted outcome
  - Risk exposure calculation : to determine the risk Priority
  - see Fig 3.16, p.122
    - Regression testing : the assurance that existing functionality still works correctly.

Example of risk exposure calculation

PU: prob. of unwanted outcome

LU: lost assoc with unwanted outcome



- Risk control 风险控制
  - Risk control involves risk reduction, risk planning and risk resolution 风险控制活动包括风险降低、风险计划、风险解决方案
  - Three strategies for risk reduction
    - **Avoiding the risk**: change requirements for performance or functionality 避免风险
    - **Transferring the risk**: transfer to other system, or buy insurance 转移风险
    - **Assuming the risk**: accept and control it 承担风险
  - Cost of reducing risk
    - Risk leverage = (risk exposure before reduction – (risk exposure after reduction) / (cost of risk reduction)
  - Risk management plan : record of your decisions

3.5 The Project Plan

3.5.1 Purposes 项目计划的目的/作用

- To **communicate to customers** about the risk analysis and management, project cost estimates, schedule and organization
- **Customers** can refer to the plan for information about the activities in the development process, making it easy to follow the project's progress
- **Developers** can use the plan to confirm with the customer any assumptions about the project, especially about the cost and schedule

3.5.2 Contents 项目计划的内容

软件项目计划流程

开发问题描述 → 定义顶层设计 → 定义工作分解结构 → 建立初始时间表

开发问题描述

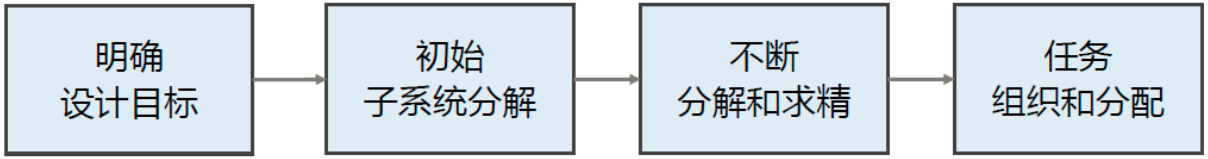
问题描述

问题描述是描述系统应该说明的问题、目标环境、客户交付和验收标准的简短文档。

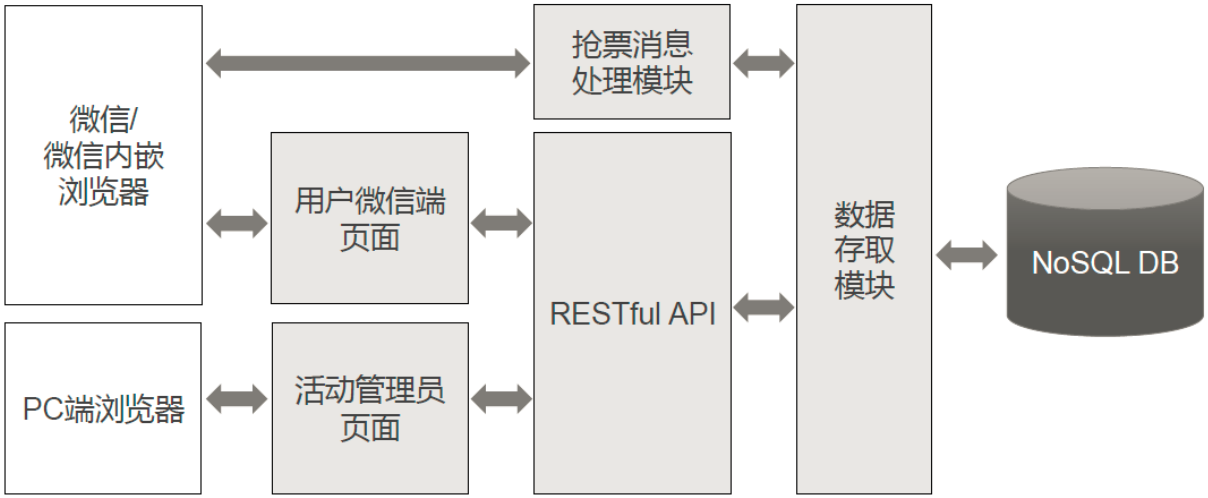
问题描述是对系统所表述问题的共同认识，通常是由项目团队和客户共同开发形成的，它定义了问题提出的背景、需要支持的功能和性能以及系统运行的目标环境等。

定义顶层设计

顶层设计描述了最初从系统到子系统的分解，它描述了系统的软件体系结构。

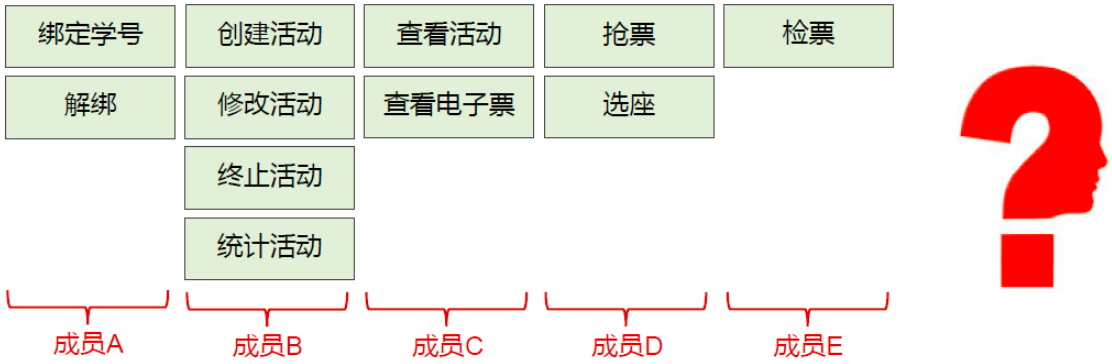


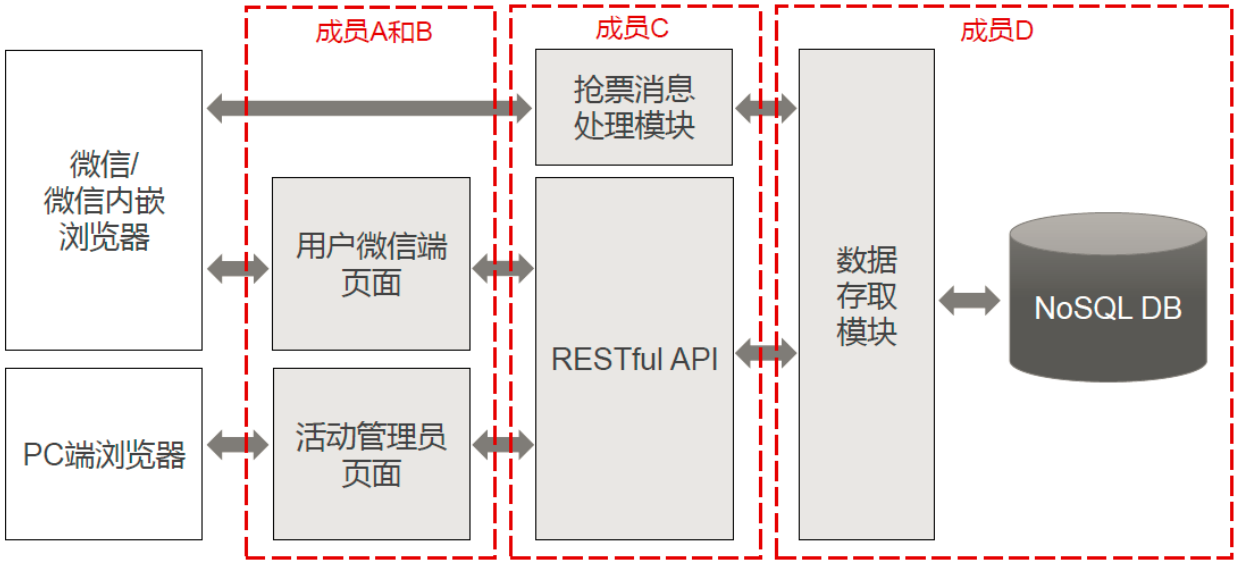
- 子系统分解应该是高层的，专注于功能，并且要保持稳定。
- 每一个子系统可以被分配给一个团队或一个人，由他负责其定义和实现。



定义项目工作分解

项目工作分解是将项目整体分解成较小的、易于管理和控制的若干子项目或工作单元，直到可交付成果定义的足够详细，足以支持项目将来的活动。





项目计划的内容

- Project scope 项目范围
- Project schedule 项目进程
- Project team organization 团队组织
- Technical description of system
- Project standards and procedures
- Quality assurance plan
- Configuration management plan
- Documentation plan
- Data management plan
- Resource management plan
- Test plan
- Training plan
- Security plan
- Risk management plan
- Maintenance plan

3.6 , 3.7, 3.8 Are Omitted

本章小结

1. 进度管理部分

- 项目进程 **Project schedule**
- 活动和里程碑 **Milestones and activities**
- 工作分解结构 **Work breakdown structure (WBS)**
- 活动及其属性 **Activity (precursor,duration,due date, endpoint)**
- 活动图 **Activity graph**
- 空闲时间 **Slack time = available time - real time = latest start time - earliest start time**

本章小结

- ◆关键路径 **Critical path**
- ◆关键路径方法 **Critical path method (CPM)**

2. 人员管理部分

- ◆关键活动 **Key activities requiring personnel**
- ◆人员特征 **Choosing personnel**
- ◆工作方式 **Work styles**
- ◆选择合适的组织 **Choosing project organization**

本章小结

3. 成本估算部分

◆专家判断方法的特征、问题

◆算法模型 $E = (a + bSc)\, m(X)$

◆**COCOMO II** 模型

4. 风险管理部分

◆风险影响 **Risk impact**

◆风险概率 **Risk probability**

◆风险控制 **Risk control**

◆风险成本 **Risk exposure = (risk probability) x (risk impact)**

本章小结

◆风险管理活动 **Risk management activities (Figure 3.15)**

◆风险降低的**3**种策略

5. 项目计划

（简单了解）

Review Exam

- **If a system is being developed where the customers are not sure of what they want, the requirements are often poorly defined. Which of the following would be an appropriate process model for this type of development?**

  a. prototyping

  b. waterfall

  c. V-model

  d. spiral

Review Exam

- **The project team developing a new system is experienced in the domain. Although the new project is fairly large, it is not expected to vary much from applications that have been developed by this team in the past. Which process model would be appropriate for this type of development?**

  a. prototyping
  b. waterfall
  c. V-model
  d. spiral

❖ 下列选项不属于瀑布模型的优点的是（　）。

A．可迫使开发人员采用规范的方法

B．严格的规定了每个阶段必须提交的文档

C．要求每个阶段交出的所有产品都必须经过质量保证小组的仔细验证

D．支持后期的变动

❖ 极限编程（XP）是一种轻量（敏捷）、高效、低风险、柔性、可预测、科学而且充满乐趣的软件开发方式，以下不属于它所倡导的十二个最佳实践的是___ (1)_____。其中____(2)_____是通过预防性、及时性的措施来提高系统的质量的重要实践活动，现在已涌现出了JUnit、NUnit、DUnit等一系列相应的工具来提供支持。

供选择的答案：

① A. 受控的需求文档　　B. 计划游戏　　C. 现场客户　　D. 隐喻

② A. 集体代码所有制　　B. 持续集成　　C. 测试先行　　D. 重构

Determine whether or not each of the following statements is describing a risk. Answer TRUE if the statement describes a risk, FALSE otherwise.

30. To catch defects early, requirements inspections have been incorporated into the process.

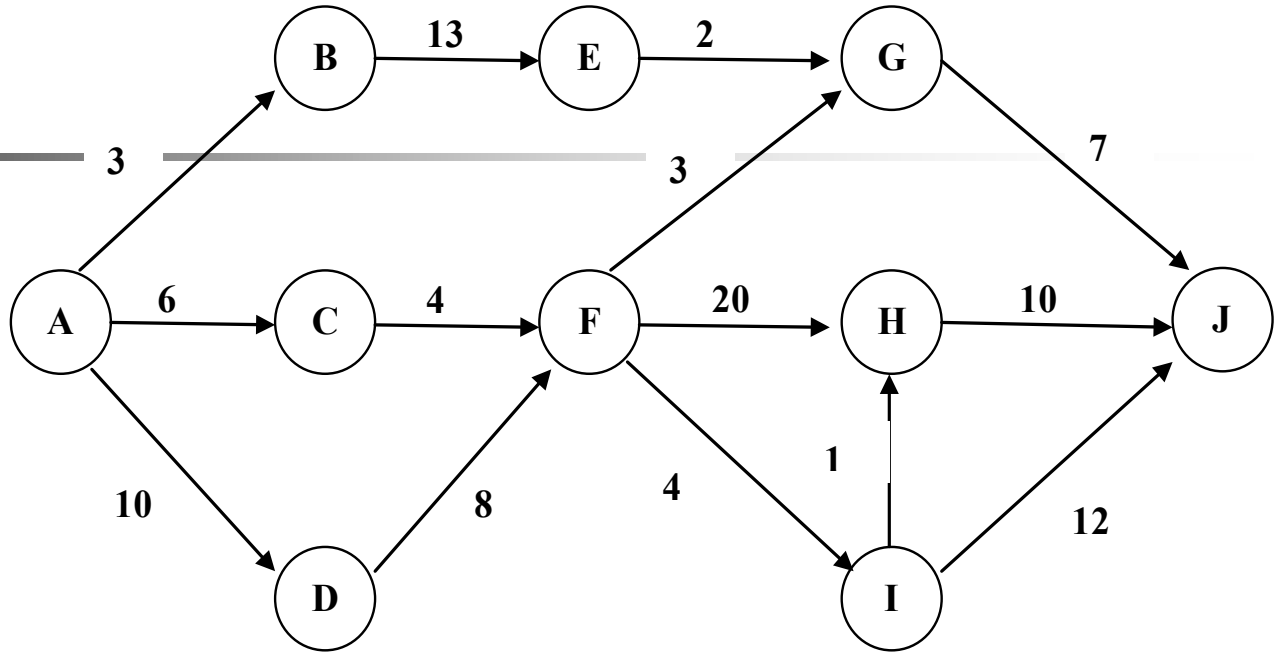
31. The customers are not clear about what they want. The requirements may be volatile.

32. The delivery of a subsystem being developed by another group may be delayed and cause the whole project schedule to slip.

33. The project team is inexperienced. A requirement may be misunderstood and designed incorrectly.

34. The development team is using a CASE tool for the first time on the design.

35. To aid the customer in identifying requirements, several prototypes are planned.



28. Which milestones are precursors to H?
    a. A
    b. B
    c. C
    d. A and B
    e. A and C
    f. All of the above

29. If there are seven team members assigned to a project team, how many potential lines of communication are there?
    a. 6
    b. 7
    c. 21
    d. 49

23. Which of the following is a critical path from milestone A to milestone J?
    a. ACFHJ
    b. ACFIHJ
    c. ABEGHJ
    d. ADFHJ

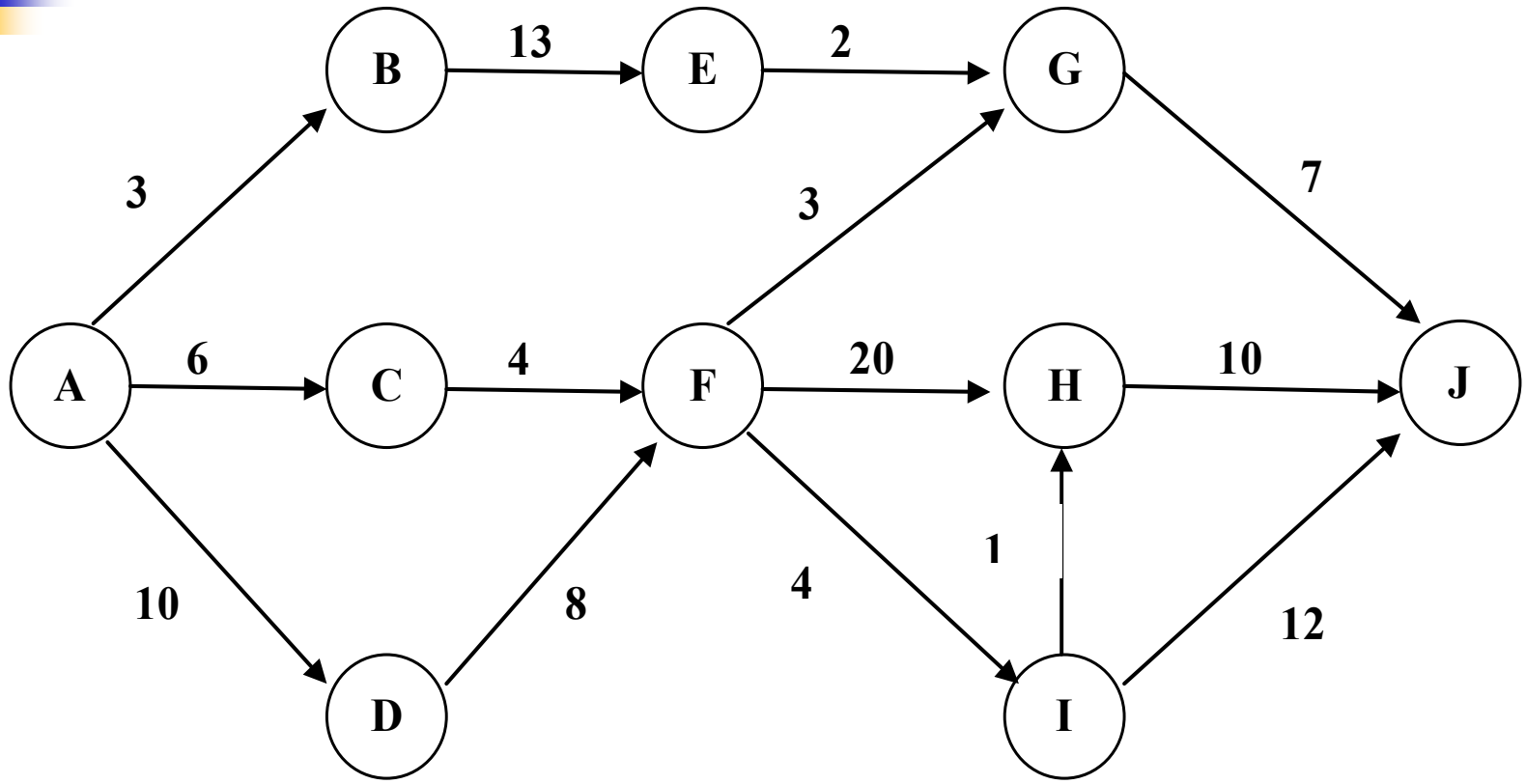
24. What is the slack time for the activity starting at milestone C?
    a. 7
    b. 8
    c. 15
    d. 20

25. What is the length of the critical path identified in question 23?
    a. 32
    b. 40
    c. 48
    d. 55

26. What is the latest start time for the activity starting at milestone E?
    a. 10
    b. 18
    c. 25
    d. 40

27. What is the earliest start time for the activities starting at milestone F?
    a. 11
    b. 19
    c. 33
    d. 37

本章课堂作业



标出每个活动的最早开始时间、最晚开始时间，并指出关键路径。