软件工程概论
Software Engineering

刘伟

liuwei@xidian.edu.cn

88204608

CH3. Planning and Managing the Project

Content

- Project Management
- Tracking Progress
- Project Personnel
- Effort Estimation
- Risk Management
- The Project Plan

3.1 Project Management

What is a Project?

- “A project is a ***temporary endeavor*** undertaken to create a ***unique*** product, service, or result.” (This is the PMI ***Project Management Institute美国项目管理学会*** definition)

3.1 Project Management

What is a Project?

- 定义一：项目是完成某些特定指标的一次性任务。
- 定义二：项目是一项有待完成的专门任务，是在一定的组织机构内，在限定的资源条件下，在计划的时间里，按满足一定性能、质量与数量的要求去完成的一次性任务。

- 项目的特征
  - 唯一性
  - 一次性
  - 整体性
  - 多目标性
  - 寿命周期阶段性

3.1 Project Management

What is a Project?

- 一个*项目*无论大小、特点如何，一般包括下列要素：
  - (1)具体的结果（产品或结果）。
  - (2)明确的开始与结束日期（项目工作开始和结束日期）。
  - (3)既定的预算（包括人员、资金、设备和资料总额等）。
- 软件项目的产品与其它项目的产品相比的不同特征
  - 不可见性
  - 复杂性
  - 一致性
  - 灵活性

3.1 Project Management

Project Management Defined

- Project Management is the process of using proven tools and techniques to manage the scope, time and cost of a project.

3.1 Project Management

- Project Management
  - The application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations.

- 两种含义
  - **管理活动**：一种有意识地按照项目管理的特点和规律，对项目进行组织管理的活动
  - **管理学科**：以项目管理活动为研究对象的一门学科，它是探索项目活动科学组织管理的理论与方法。

3.1 Project Management

软件项目管理

- 软件项目管理是为了使软件项目能够按照预定的成本、进度、质量顺利完成，而对成本、人员、进度、质量、风险等进行分析和管理的活动。
- 软件项目管理的对象是软件工程项目，他所涉及的范围覆盖了整个软件工程过程。

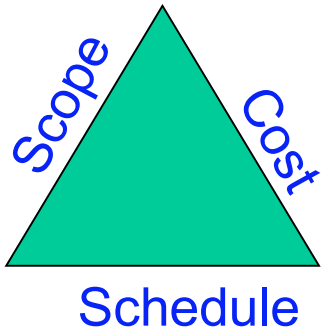
3.1 Project Management

管理的范围

- 有效的项目管理集中于三个*P* 上：
  - ***People***
    - 项目参与者
    - 项目负责人
    - 软件项目组
    - 协调和通讯
  - ***Problem***
    - 软件范围
    - 问题分解
  - ***Process***
    - 合并问题和过程
    - 过程分解

3.1 Project Manager - Responsibilities

- Planning the project
  - Technical activities
  - Project management activities
- Managing the triple constraint to the Project Sponsor's satisfaction
  - Cost
  - Scope
  - Schedule (Time)
  - Quality



- Organizing the project, which includes
  - Forming the project team
  - Setting up systems to document and communicate the project objectives and status
  - Establishing project plans and the processes by which the project will be controlled

3.2 Tracking project progress

- Questions from our customers
  - Do you understand my problem and my needs?
  - Can you design a system that will solve my 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?
- Require a well-thought-out project schedule

3.2 Tracking project progress

Project deliverables

- Documents
- Demonstrations of function
- Demonstrations of subsystems
- Demonstrations of accuracy
- Demonstrations of reliability, performance or security

3.2 Tracking project progress

Milestones and activities

- Activity: takes place over a period of time
- Milestone: completion of an activity -- a particular point in time

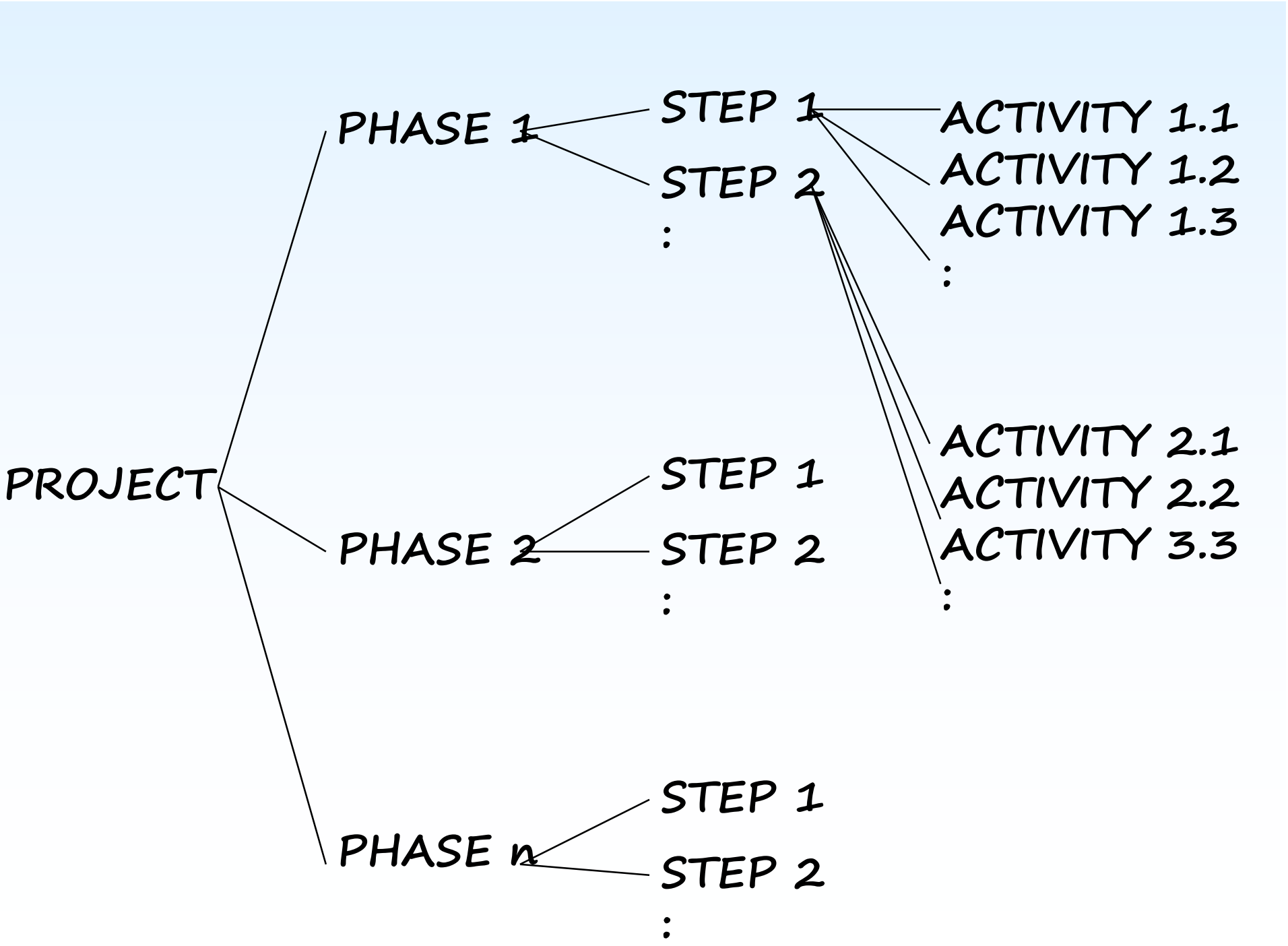

PROJECT
PHASE 1
PHASE 2
PHASE n
STEP 1
STEP 2
:
STEP 1
STEP 2
:
STEP 1
STEP 2
:
ACTIVITY 1.1
ACTIVITY 1.2
ACTIVITY 1.3
:
ACTIVITY 2.1
ACTIVITY 2.2
ACTIVITY 3.3
:

Table 3.1. Phases, steps and activities of building a house.

Phase 1: Landscaping the lot	**Phase 2: Building the house**
Step 1.1: Clearing and grubbing	*Step 2.1: Prepare the site*
Activity 1.1.1: Remove trees	Activity 2.1.1: Survey the land
Activity 1.1.2: Remove stumps	Activity 2.1.2: Request permits
Step 1.2: Seeding the turf	Activity 2.1.3: Excavate for the foundation
Activity 1.2.1: Aerate the soil	Activity 2.1.4: Buy materials
Activity 1.2.2: Disperse the seeds	*Step 2.2: Building the exterior*
Activity 1.2.3: Water and weed	Activity 2.2.1: Lay the foundation
Step 1.3: Planting shrubs and trees	Activity 2.2.2: Build the outside walls
Activity 1.3.1: Obtain shrubs and trees	Activity 2.2.3: Install exterior plumbing
Activity 1.3.2: Dig holes	Activity 2.2.4: Exterior electrical work
Activity 1.3.3: Plant shrubs and trees	Activity 2.2.5: Exterior siding
Activity 1.3.4: Anchor the trees and mulch around them	Activity 2.2.6: Paint the exterior
	Activity 2.2.7: Install doors and fixtures
	Activity 2.2.8: Install roof
	Step 2.3: Finishing the interior
	Activity 2.3.1: Install the interior plumbing
	Activity 2.3.2: Install interior electrical work
	Activity 2.3.3: Install wallboard
	Activity 2.3.4: Paint the interior
	Activity 2.3.5: Install floor covering
	Activity 2.3.6: Install doors and fixtures

3.2 Tracking project progress

Milestones

- Part of top-down approach
- Break large project into small problems, each of which can be estimated and planned
- A milestone is: ***An objectively identifiable point in a project***
- Good checkpoints are:
  - Clear, unambiguous, crisp, verifiable
  - Binary: done or not done

3.2 Tracking project progress

Milestones - Good Examples

- Design document reviewed
- Design document signed-off by management
- System software successfully passes integration test data suite
- Specification document approved by customer
- All risks determined at last process review addressed and resolved

3.2 Tracking project progress

Milestones - Bad Examples

- "Coding is 90% complete":
  - In terms of time, or loc?
  - How do you know?
- "Program is designed":
  - What's this mean?
  - In your head, or on paper?
  - Has it been reviewed, agreed upon?

Table 3.2. Milestones in building a house.

1.1. Survey complete
1.2. Permits issued
1.3. Excavation complete
1.4. Materials on hand
2.1. Foundation laid
2.2. Outside walls complete
2.3. Exterior plumbing complete
2.4. Exterior electrical work complete
2.5. Exterior siding complete
2.6. Exterior painting complete
2.7. Doors and fixtures mounted
2.8. Roof complete
3.1. Interior plumbing complete
3.2. Interior electrical work complete
3.3. Wallboard in place
3.4. Interior painting complete
3.5. Floor covering laid
3.6. Doors and fixtures mounted

3.2 Tracking project progress

Work breakdown structure (WBS)

- Define work breakdown structure (WBS)
  - Identify tasks and subtasks -- deliverables
  - Lowest element – stand alone work package

3.2 Tracking project progress

Work Breakdown and Activity Graph

- Describe each activity
  - Precursor: event or set of events that must occur in order for an activity to start
  - Duration: length of time needed to complete an activity
  - Due date: date by which an activity must be completed
  - End point : milestone or deliverable.

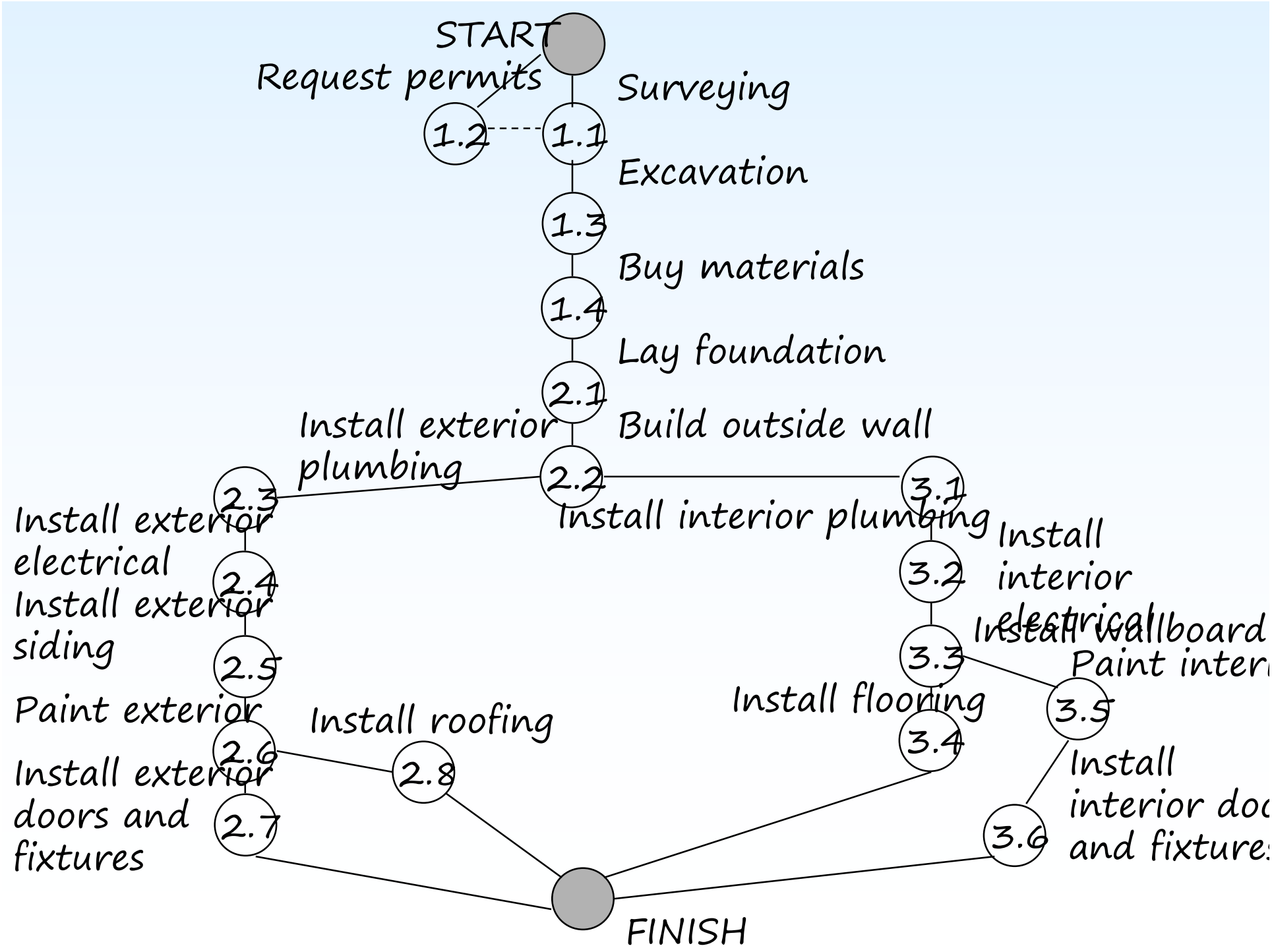
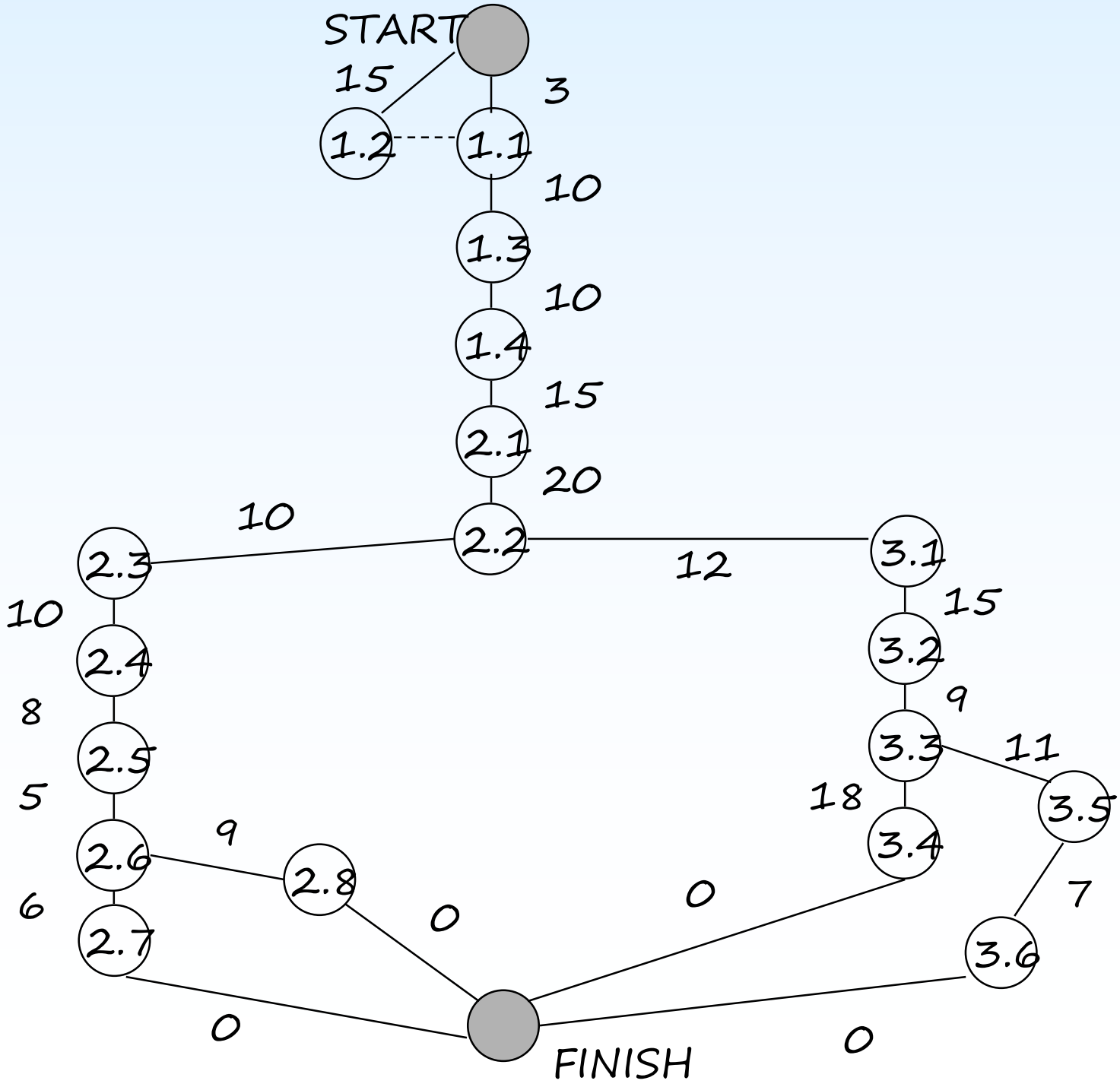

START
Request permits
Surveying
1.2
1.1
Excavation
1.3
Buy materials
1.4
Lay foundation
2.1
Build outside wall
Install exterior plumbing
2.2
2.3
3.1
Install interior plumbing
Install exterior electrical
2.4
3.2
Install interior
Install exterior siding
2.5
3.3
Paint exterior
Install roofing
Install flooring
3.5
2.6
2.8
3.4
Install exterior doors and fixtures
2.7
3.6
FINISH

Table 3.3. Activities and time estimates.

Activity	Time estimate (in days)
Step 1: Prepare the site	
Activity 1.1: Survey the land	3
Activity 1.2: Request permits	15
Activity 1.3: Excavate for the foundation	10
Activity 1.4: Buy materials	10
Step 2: Building the exterior	
Activity 2.1: Lay the foundation	15
Activity 2.2: Build the outside walls	20
Activity 2.3: Install exterior plumbing	10
Activity 2.4: Exterior electrical work	10
Activity 2.5: Exterior siding	8
Activity 2.6: Paint the exterior	5
Activity 2.7: Install doors and fixtures	6
Activity 2.8: Install roof	9
Step 3: Finishing the interior	
Activity 3.1: Install the interior plumbing	12
Activity 3.2: Install interior electrical work	15
Activity 3.3: Install wallboard	9
Activity 3.4: Paint the interior	18
Activity 3.5: Install floor covering	11
Activity 3.6: Install doors and fixtures	7


START
15
3
1.2
1.1
10
1.3
10
1.4
15
2.1
20
10
2.2
12
2.3
3.1
10
15
2.4
3.2
8
9
2.5
3.3
11
5
18
3.5
2.6
9
3.4
2.8
0
0
6
7
2.7
3.6
0
0
FINISH

3.2 Tracking project progress

Slack or float time

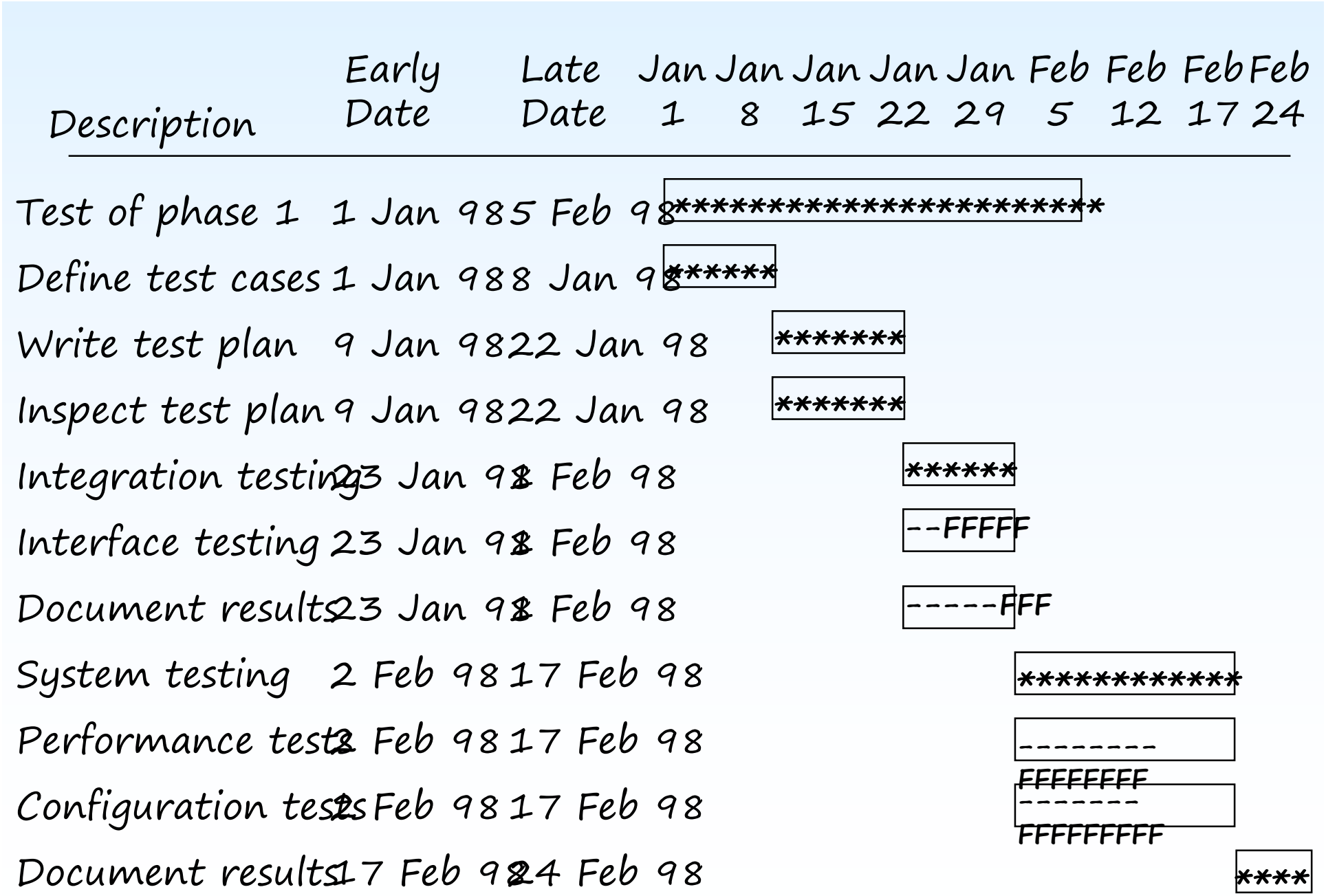
Slack time = available time - real time

= latest start time - earliest start time

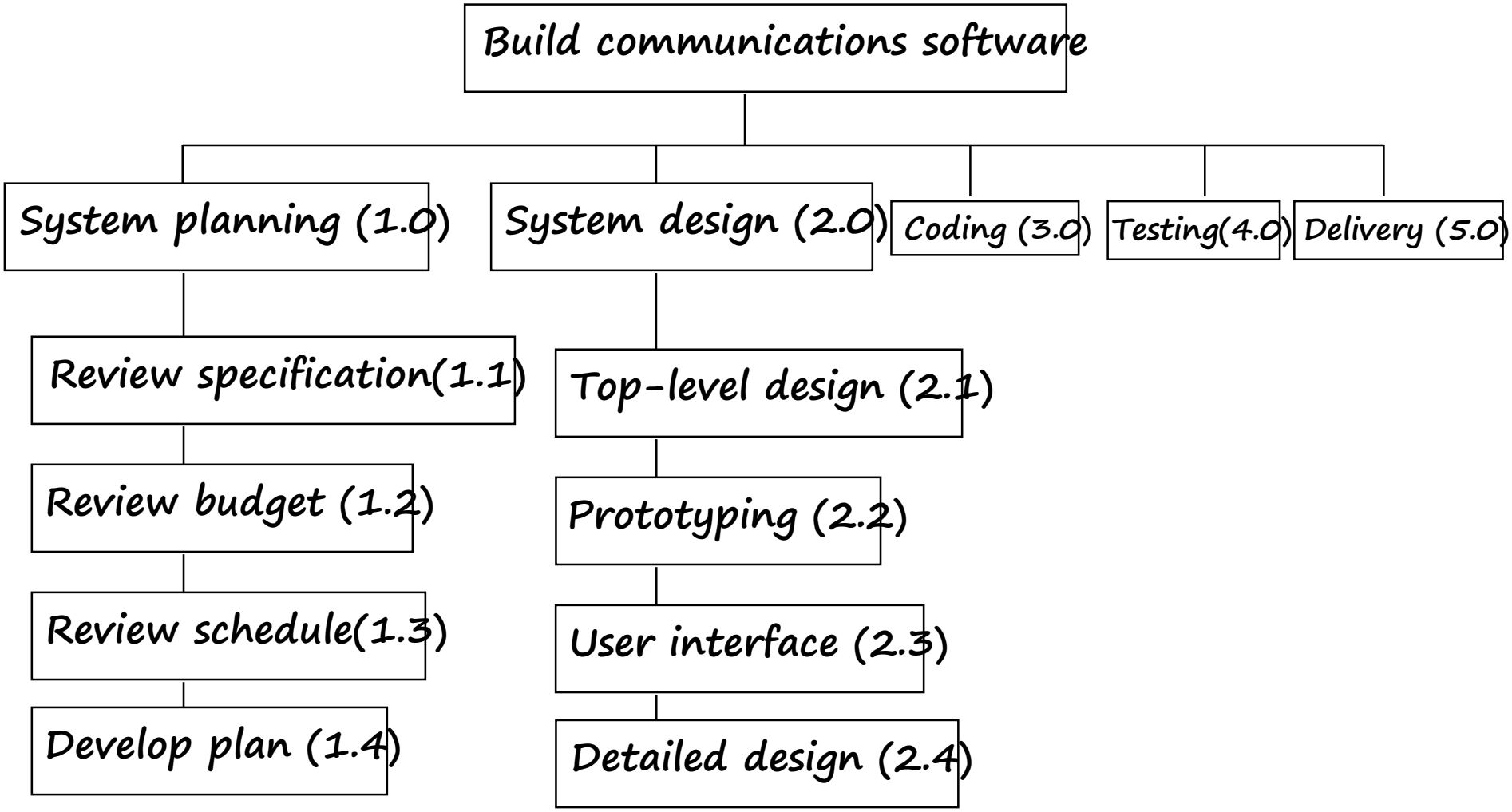
3.2 Tracking project progress

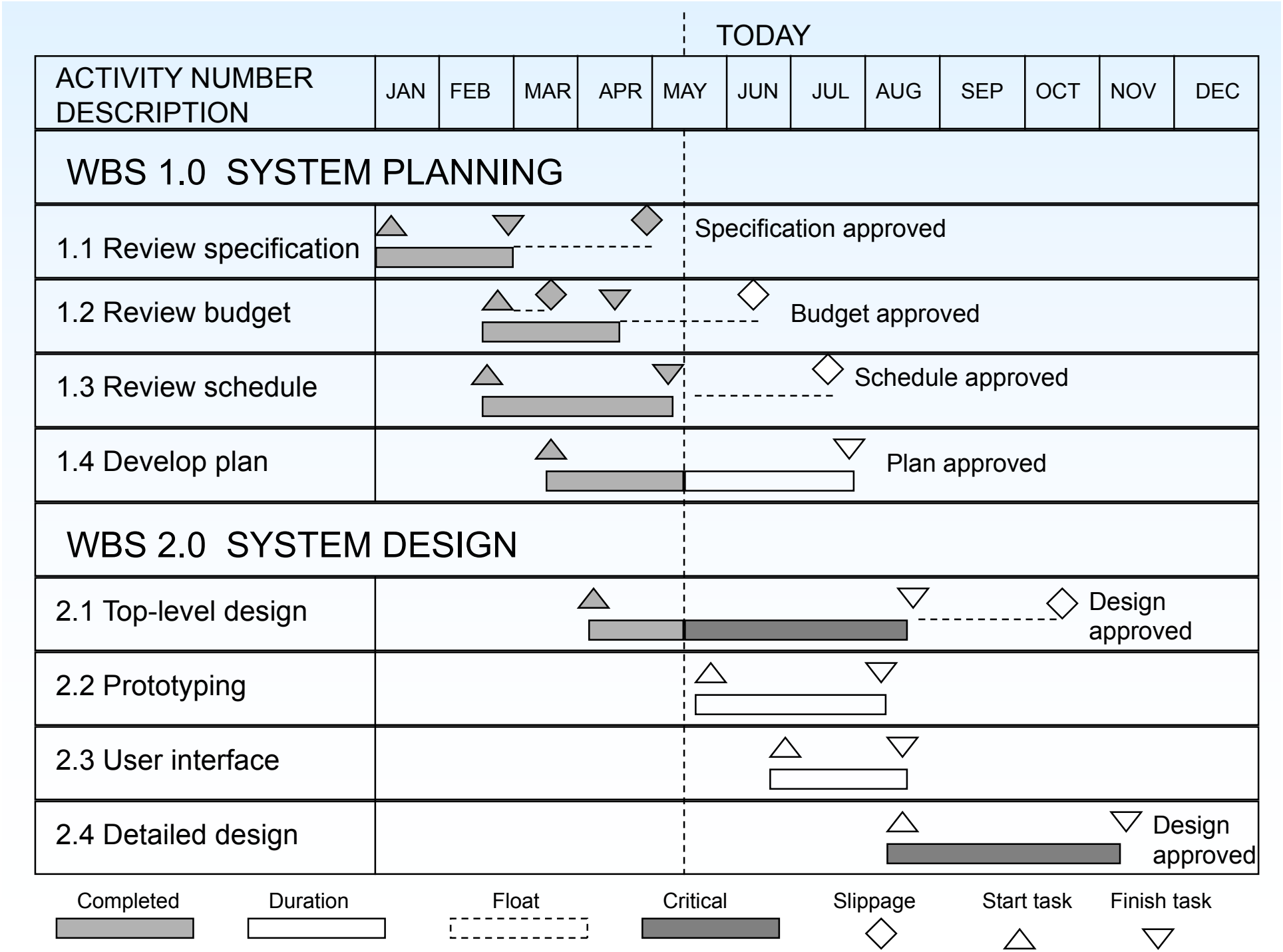
Table 3.4. Slack time for project activities.

Activity	Earliest start time	Latest start time	Slack
1.1	1	13	12
1.2	1	1	0
1.3	16	16	0
1.4	26	26	0
2.1	36	36	0
2.2	51	51	0
2.3	71	83	12
2.4	81	93	12
2.5	91	103	12
2.6	99	111	12
2.7	104	119	15
2.8	104	116	12
3.1	71	71	0
3.2	83	83	0
3.3	98	98	0
3.4	107	107	0
3.5	107	107	0
3.6	118	118	0
Finish	124	124	0


Description
Early Date
Late Date
Jan 1
Jan 8
Jan 15
Jan 22
Jan 29
Feb 5
Feb 12
Feb 17
Feb 24
Test of phase 1 1 Jan 98 5 Feb 98
Define test cases 1 Jan 98 8 Jan 98
Write test plan 9 Jan 98 22 Jan 98
Inspect test plan 9 Jan 98 22 Jan 98
Integration testing 23 Jan 98 1 Feb 98
Interface testing 23 Jan 98 1 Feb 98
Document results 23 Jan 98 1 Feb 98
System testing 2 Feb 98 17 Feb 98
Performance test 2 Feb 98 17 Feb 98
Configuration tests 2 Feb 98 17 Feb 98
Document results 17 Feb 98 24 Feb 98

3.2 Tracking project progress - Tools to Track Progress

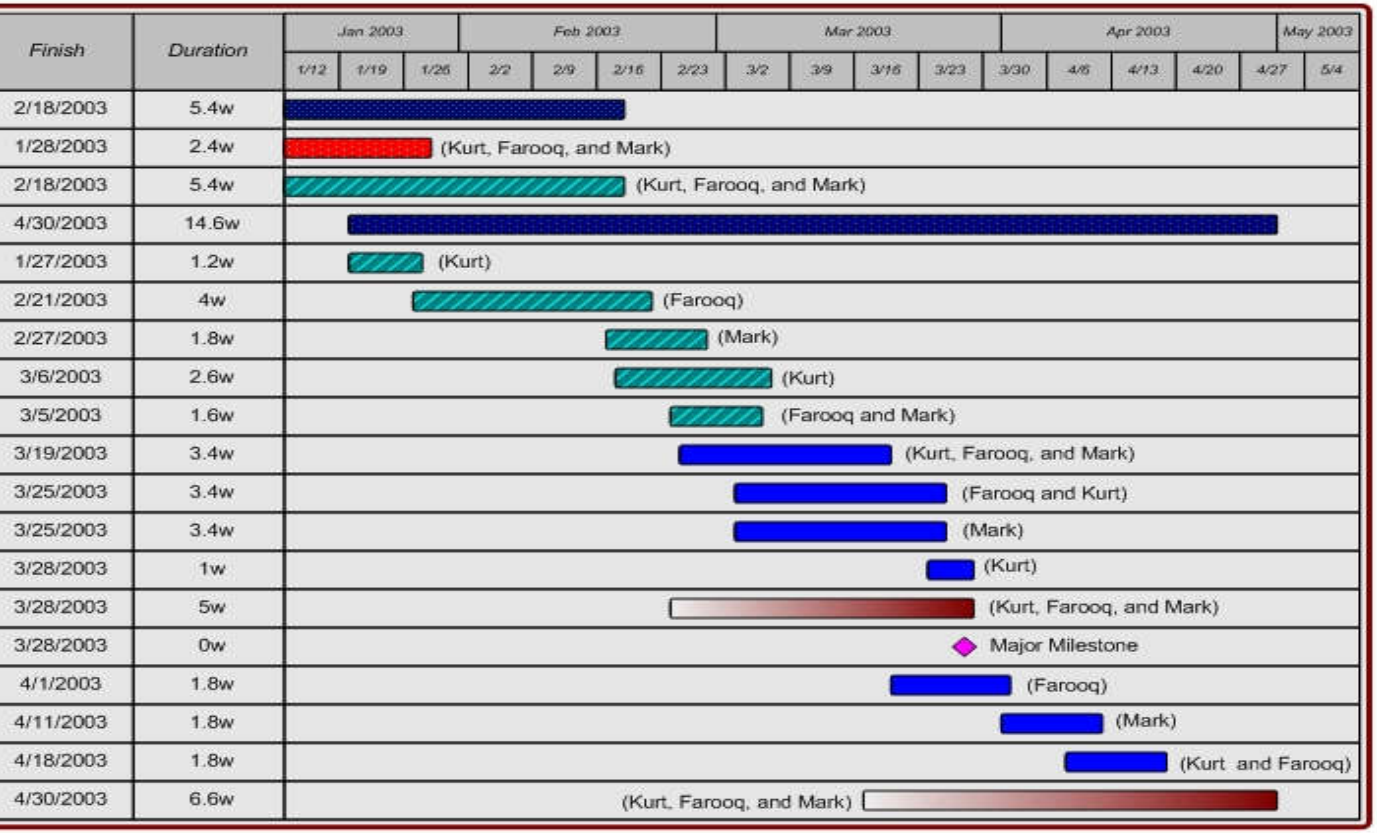


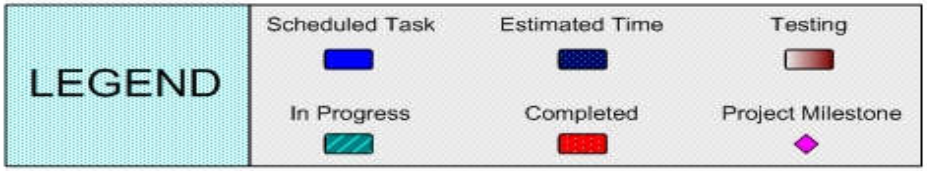

TODAY
ACTIVITY NUMBER
DESCRIPTION
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
WBS 1.0 SYSTEM PLANNING
1.1 Review specification
Specification approved
1.2 Review budget
Budget approved
1.3 Review schedule
Schedule approved
1.4 Develop plan
Plan approved
WBS 2.0 SYSTEM DESIGN
2.1 Top-level design
Design approved
2.2 Prototyping
2.3 User interface
2.4 Detailed design
Design approved
Completed
Duration
Float
Critical
Slippage
Start task
Finish task

3.2 Tracking project progress

Alien Professors Gantt Chart

ID	Task Name	Start	Finish	Duration
1	**Project Research**	1/13/2003	2/18/2003	5.4w
2	• Choose Project	1/13/2003	1/28/2003	2.4w
3	• Programming Games	1/13/2003	2/18/2003	5.4w
4	**Top-Level Design**	1/20/2003	4/30/2003	14.6w
5	• Splash Screen	1/20/2003	1/27/2003	1.2w
6	• Interface	1/27/2003	2/21/2003	4w
7	• Score Display	2/17/2003	2/27/2003	1.8w
8	• Design Level 1 Maze	2/18/2003	3/6/2003	2.6w
9	• Design Boundaries	2/24/2003	3/5/2003	1.6w
10	• Create Rules	2/25/2003	3/19/2003	3.4w
11	• Create Villains	3/3/2003	3/25/2003	3.4w
12	• Create a Student	3/3/2003	3/25/2003	3.4w
13	• Background Sounds	3/24/2003	3/28/2003	1w
14	• Test Level 1	2/24/2003	3/28/2003	5w
15	• **Completion of Level 1**	3/28/2003	3/28/2003	0w
16	• Design Level 2	3/20/2003	4/1/2003	1.8w
17	• Design Level 3	4/1/2003	4/11/2003	1.8w
18	• Design Level 4	4/8/2003	4/18/2003	1.8w
19	• Testing Game	3/17/2003	4/30/2003	6.6w





3.2 Tracking project progress - Building Gantt Charts

- Make list of tasks
- Assign each task tentative start and stop dates (or durations)
- Assign people responsible for each task
- List important milestones and their dates
- If there are more than about ~15, split project into main tasks & subtasks
- Make an overall Gantt chart for the main tasks with sub-Gantt charts for the subtasks
- Determine time resolution (day, week, month)

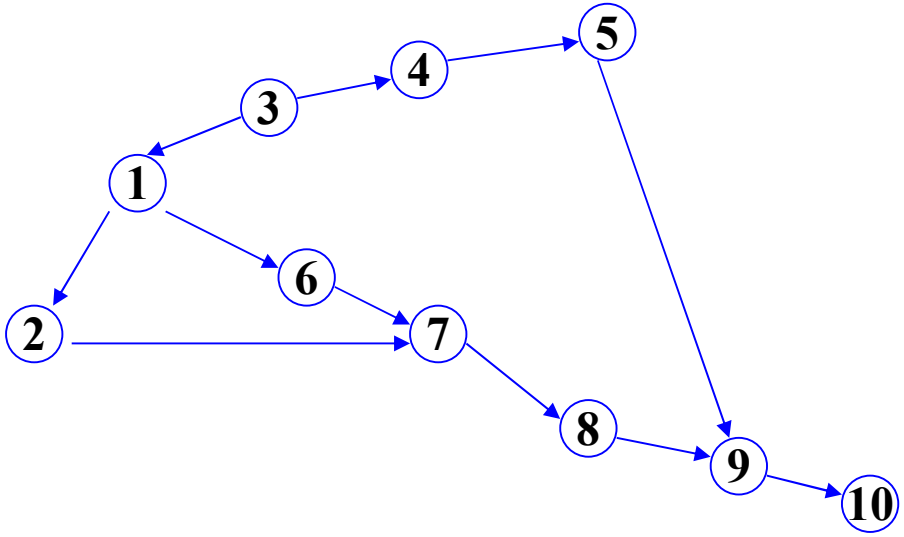
3.2 Tracking project progress - Ex: WBS - Buy A House

1	- Criteria	3	- Determine affordability	6	- Type of House
2	- Visit Locations	4	- Determine mortgage provider	7	- Find Real Estate Agent
		5	- Lockup mortgage commitment	8	- Look for House
				9	- Make Offer, P&S Agreem't
				10	- Closing
		TASK	PREDECESSOR		

Top Down Decomposition, Elemental Tasks

3.2 Tracking project progress - Network Diagrams

TASK	PREDECESSOR
1-criteria	3
2-visit loc.	1
3-affordability	
4-mortgage co	3
5-mortg lock	4
6-type of hse	1
7-real est agent	2, 6
8-look for hse	7
9-offer, P&S	5, 8
10-closing	9

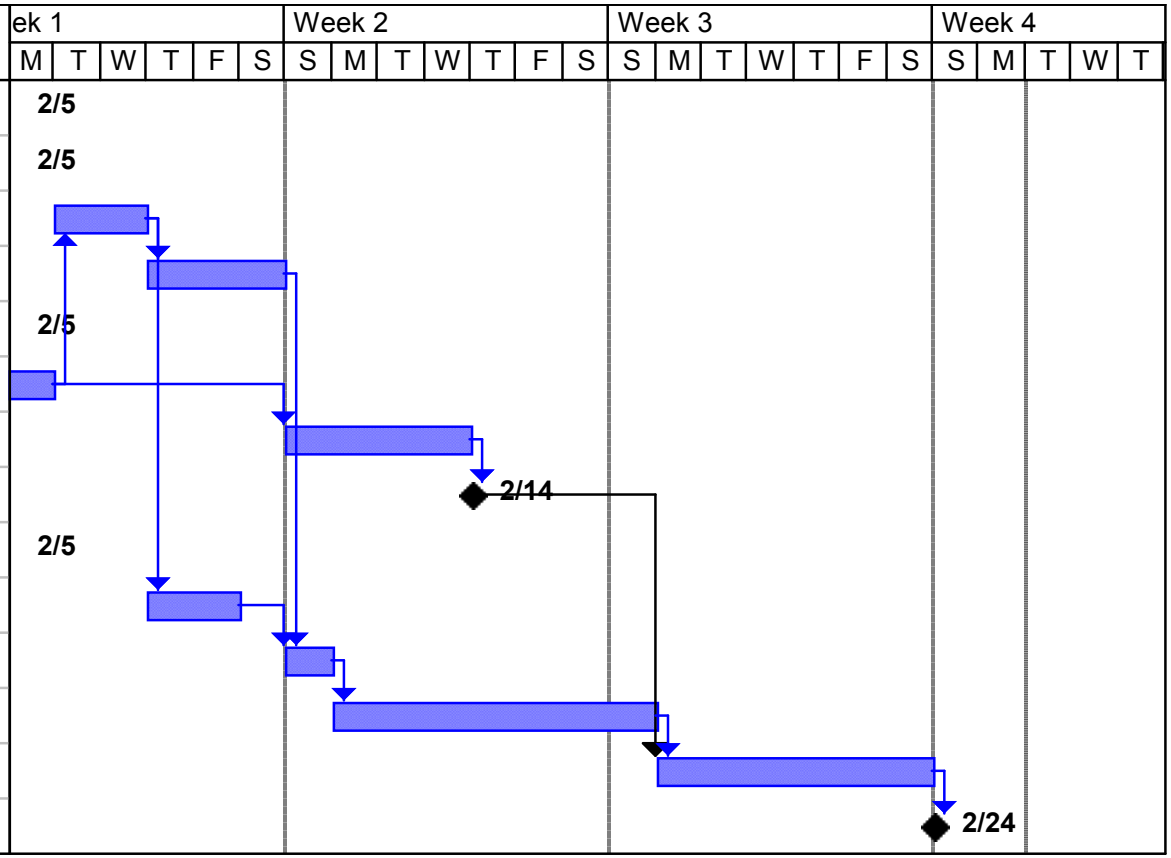


Scheduling Techniques

- **PERT – Program Evaluation and Review techniques**
- **CPM – Critical Path Method**

3.2 Tracking project progress - Buy a House

ID	Task Name
1	BUY HOUSE
2	LOCATION
3	Determine location criteria
4	Visit key locations
5	FINANCING
6	Determine affordability
7	Assess Mortagg Provider
8	Secure Mortagg Commitmt
9	FIND HOUSE
10	Type of House
11	Find RealEstate Agent
12	Look for House
13	Make Offer, P&S Agreement
14	Closing



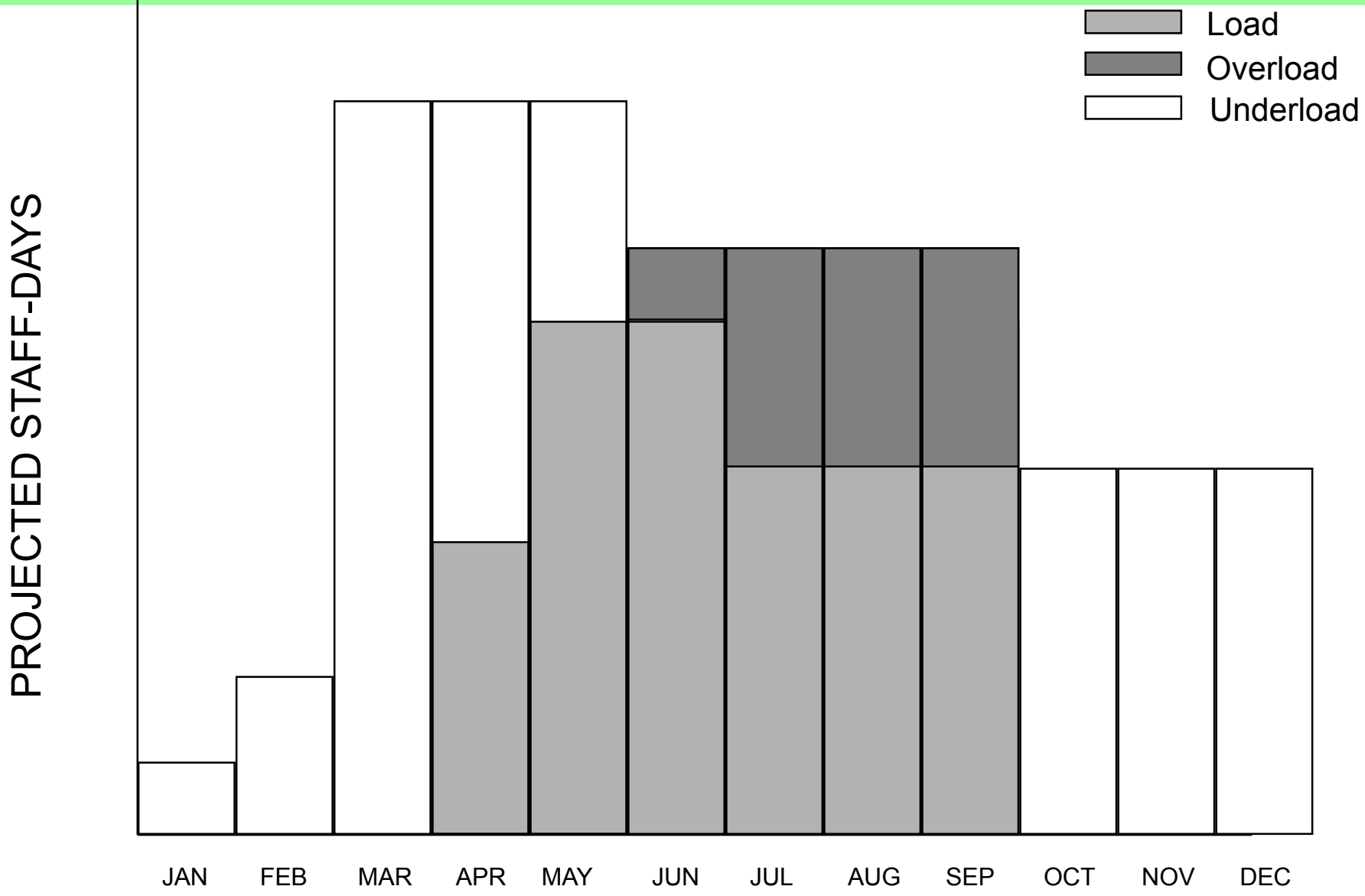
Simple Gantt Chart View

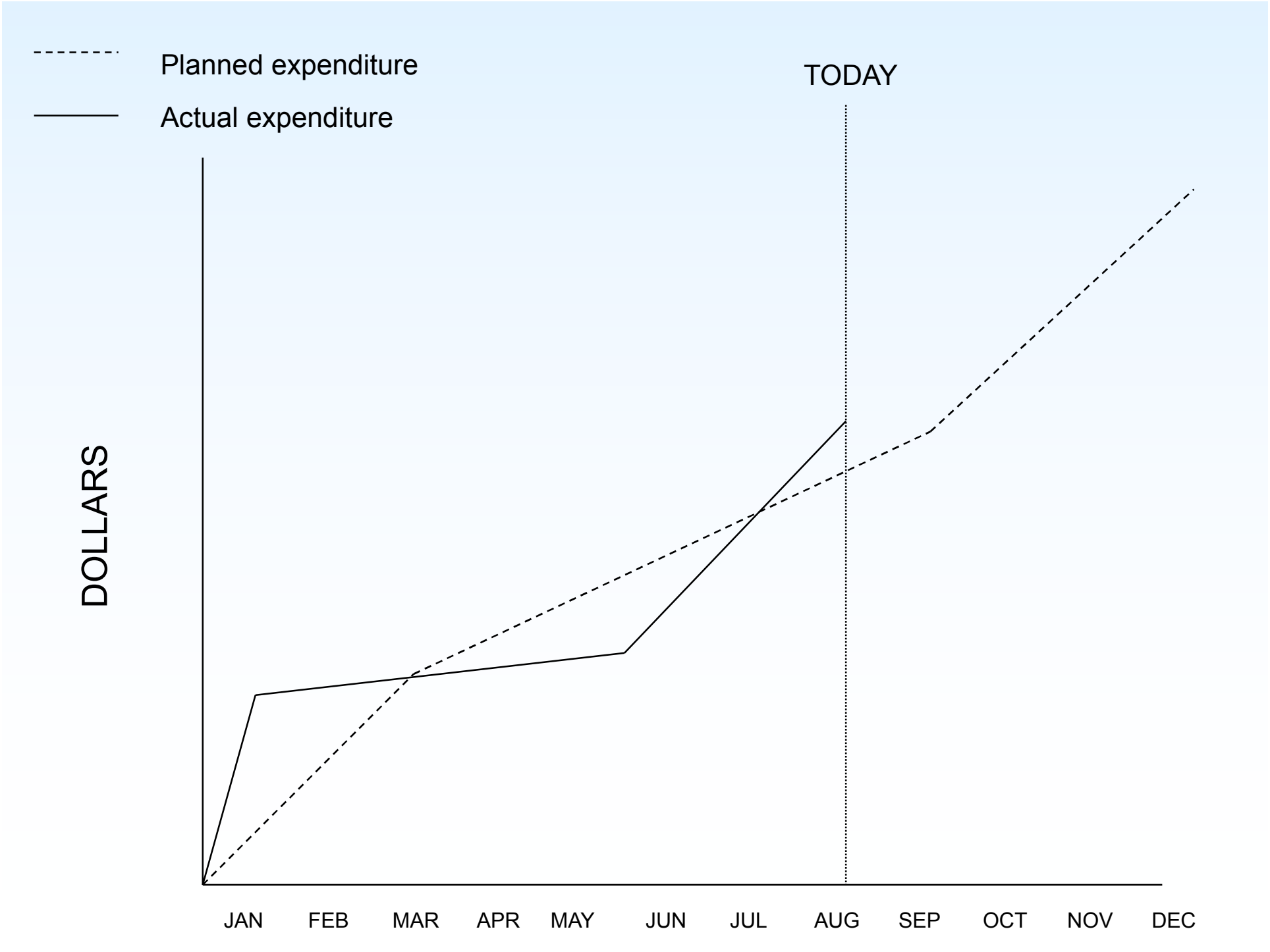
3.2 Tracking project progress

Tool Support For Gantt Charts

- Microsoft Project:
  - Data file standard: MPX Files (“Ms Project eXchange”)
- Visio:
  - Known As “Project Timeline” Diagram
  - Can Import/export Data In Mpx Format

3.2 Tracking project progress




Planned expenditure
Actual expenditure
TODAY
DOLLARS
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC

3.3 Project personnel

- To determine the project schedule and estimate （估计） the associated effort(工作量) and costs, we need to know :
  - Approximately how many people will be working on the project,
  - What tasks they will perform,
  - What abilities and experience they must have so they can do their jobs effectively.

3.3 Project personnel

- Key activities requiring personnel:
  - requirements analysis
  - system design
  - program design
  - program implementation
  - testing
  - training
  - maintenance
  - quality assurance

3.3 Project personnel

Choosing personnel

- ability to perform work
- interest in work
- experience with
  - similar applications
  - similar tools or languages
  - similar techniques
  - similar development environments
- training
- ability to communicate with others
- ability to share responsibility
- management skills

3.3 Project personnel

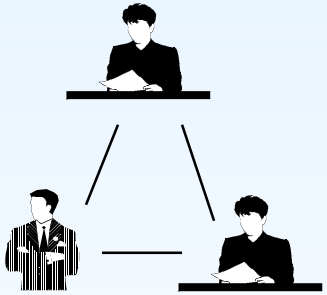
Choosing personnel

- On every software development or maintenance project, members of the development team communicate with on another, with users, and with the customer.
- Notice that the project's progress is affected not only by the degree of communication, but also by the ability of individuals to communicate their ideas.

Two people

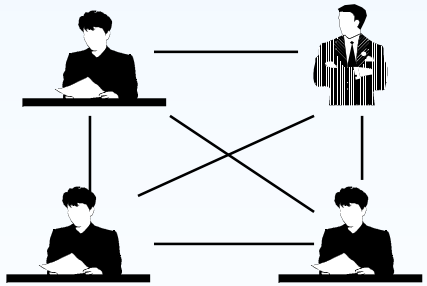
1 line of communicatio

Three people



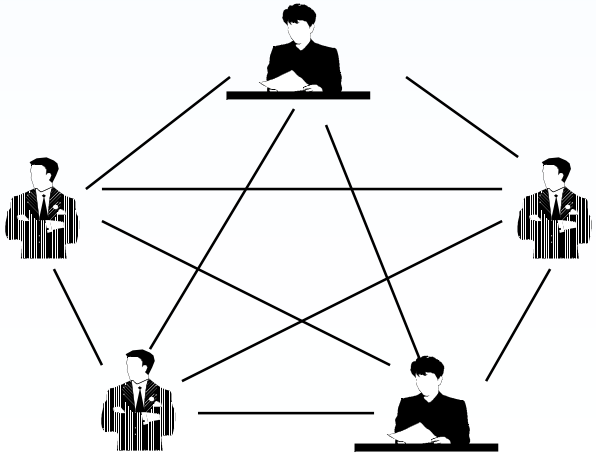
3 lines of communicatio

Four people



6 lines of communicatio

Five people



10 lines of communicatio

:

n people

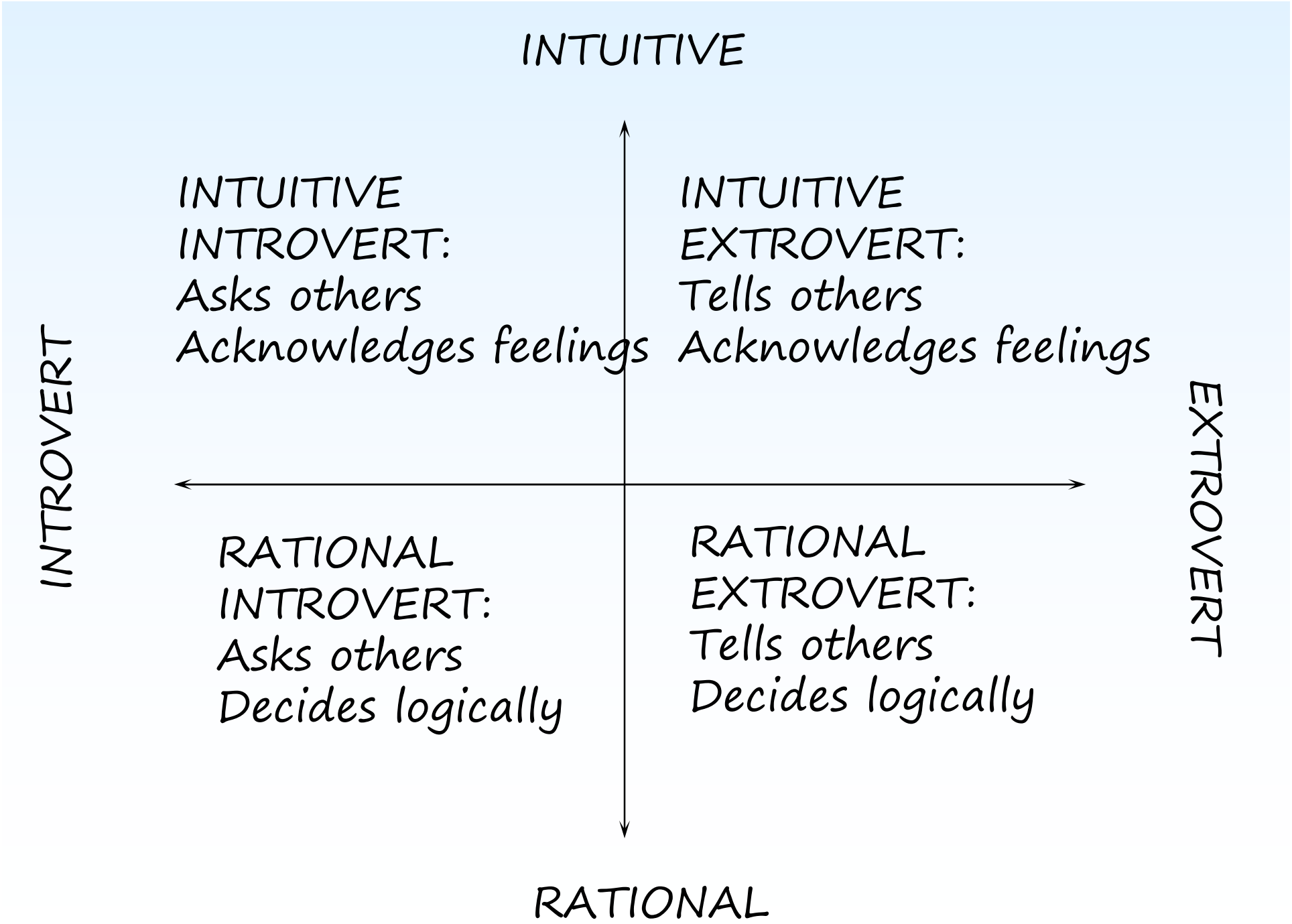
:

n(n-1)/2 lines o
communication

3.3 Project personnel

Work styles

- Extroverts（外向）： tell their thoughts
- Introverts（内向）： ask for suggestions
- Intuitives（直觉）： base decisions on feelings
- Rationals（理智）： base decisions on facts, options


INTUITIVE
INTUITIVE
INTROVERT:
Asks others
Acknowledges feelings
INTUITIVE
EXTROVERT:
Tells others
Acknowledges feelings
INTROVERT
EXTROVERT
RATIONAL
INTROVERT:
Asks others
Decides logically
RATIONAL
EXTROVERT:
Tells others
Decides logically
RATIONAL

3.3 Project personnel

Work styles

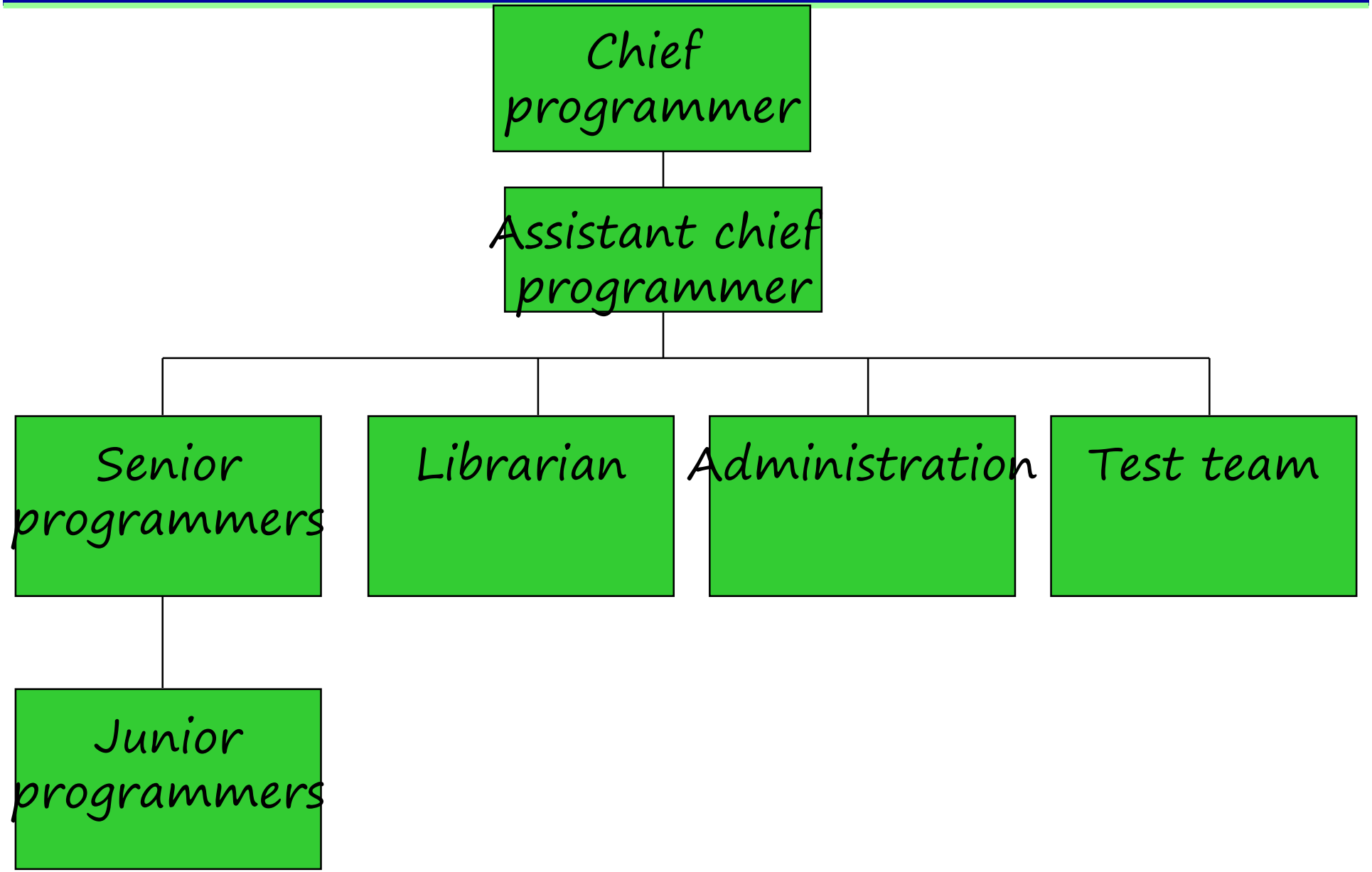
- Communication is critical to project success, and work style determines communication style.
- Understanding work styles can help you to be flexible in your approach to other project team members and to customers and users.

3.3 Project personnel

Project organization

- Depends on
  - backgrounds and work styles of team members
  - number of people on team
  - management styles of customers and developers
- Examples:
  - Chief programmer team
  - Egoless approach

3.3 Project personnel



3.3 Project personnel

Table 3.5. Comparison of organizational structures.

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

3.4 Effort estimation

- The causes (原因) of inaccurate estimates
  - Frequent requests for changes by user
  - Overlooked (忽视) tasks
  - User's lack of understanding of their own requirements
  - Insufficient analysis when developing an estimate
  - Lack of coordination (协调) of systems development, technical services, operations, data administration, and other functions during development
  - Lack of an adequate (适当的) method or guidelines for estimating

3.4 Effort estimation

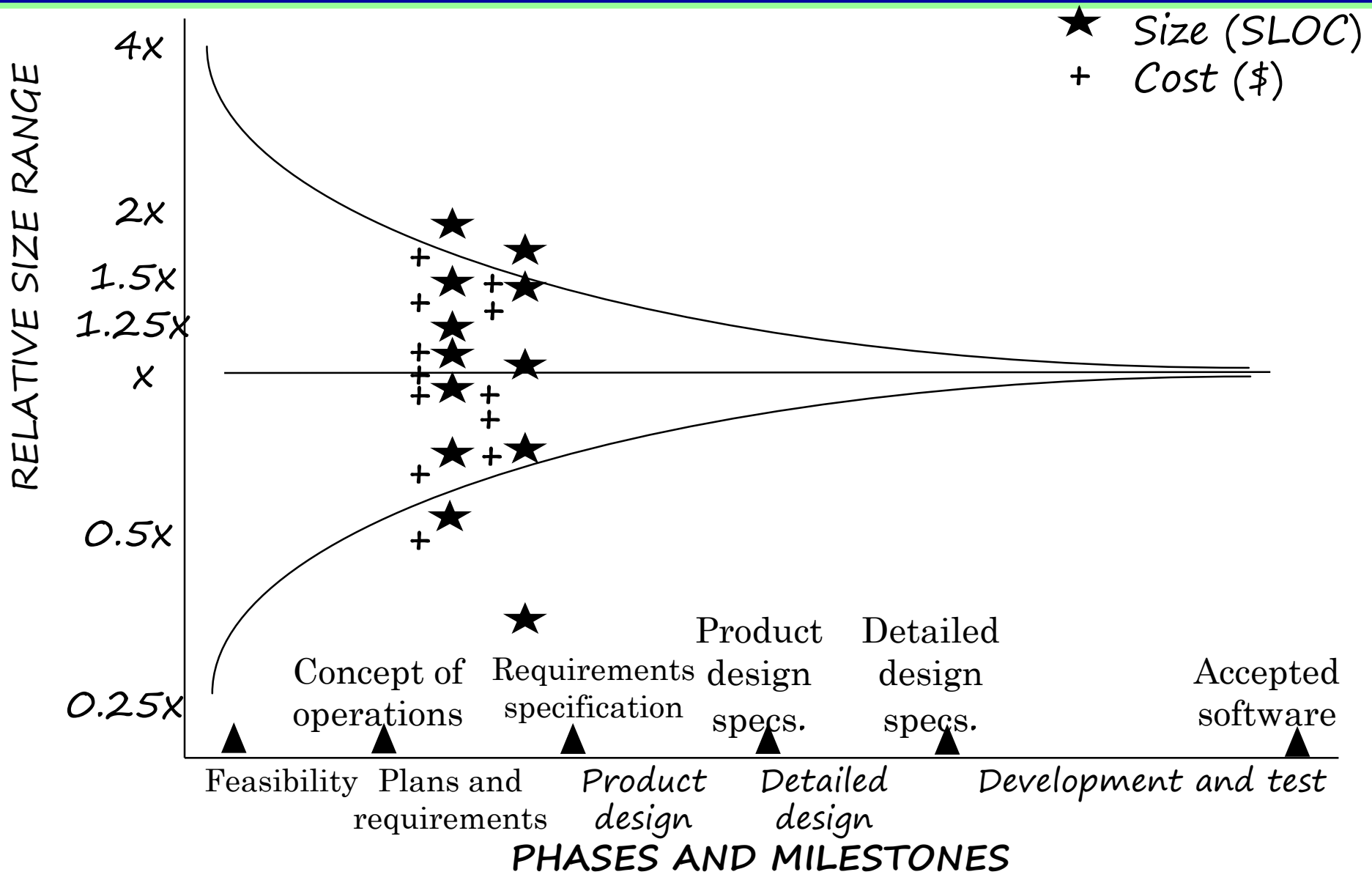
- The project budget pays for several types of costs: facilities (设施), staff (人员), methods and tools.
- There are sometimes hidden costs that are not apparent to the managers and developments.
- Other project costs involve purchasing software and tools to support development efforts.

Ex. Computer-Aided Software Engineering (CASE) tools.

3.4 Effort estimation

- For most projects, the biggest component of cost is effort.
- Effort is certainly the cost component with the greatest degree of uncertainty.
- Cost, schedule, and effort estimation must be done as early as possible during the project's lift cycle, since it affects resource allocation (分配) and project feasibility (可行性).

3.4 Effort estimation



3.4 Effort estimation

- Expert judgment
  - analogy
  - proportion
  - Delphi technique
  - Wolverton model
- Algorithmic methods: $E = (a + bS^c)\, m(\mathrm{X})$
  - Walston and Felix model: $E = 5.25S^{0.91}$
  - Bailey and Basili model: $E = 5.5 + 0.73S^{1.16}$

3.4 Effort estimation

Analogy (类推)

- Based on that we have already built a system much like the one proposed, then we can use the ***similarity*** as the basis for our estimates.
- Formalized: Ei=(x+4y+z)/6 (SLOC，Source Lines Of Code（源代码行 ））
  - X-pessimistic prediction (悲观估计)
  - Y-optimistic prediction (乐观估计)
  - Z-most likely guess (最可能的猜测)

3.4 Effort estimation

Wolverton mode (1974))

- Using ***cost matrix*** captures his experience with project cost at TRW.
- Fig. 3.6 Wolverton model Cost Matrix

3.4 Effort estimation

- In general, experiential models, by relying mostly on expert judgment, are subject to all its ***inaccuracies***.
- Projects that appear to be very similar can in fact be quite different.
- Expert judgment suffers not only from variability and subjectivity (主观因素) , but also from dependence on current data. So we must update the data.
- These methods are simplistic, neglecting to incorporate a large number of factors that can affect the effort needed on a project.

3.4 Effort estimation

- Algorithmic methods: $E = (a + bS^c)\, m(\mathrm{X})$
  - Walston and Felix model: $E = 5.25S^{0.91}$
  - Bailey and Basili model: $E = 5.5 + 0.73S^{1.16}$

- ***These model express the relationship between effort and the factors that influence it***.

3.4 Effort estimation

IBM模型 (Walston and Felix model)

- **E ＝ 5.2×$L^{0.91}$，L是源代码行数(以KLOC计)，E是工作量（以PM计）**
- **D ＝ 4.1×$L^{0.36}$，D是项目持续时间(以月计)**
- **S ＝ 0.54×$E^{0.6}$，S是人员需要量(以人计)**
- **DOC ＝ 49×$L^{1.01}$。DOC是文档数量(以页计)**
- 在此模型中，一般指一条机器指令为一行源代码。一个软件的源代码行数不包括程序注释、作业命令、调试程序在内。对于非机器指令编写的源程序，如汇编语言或高级语言程序，应转换成机器指令源代码行数来考虑。
- The basic equation was supplemented (补充) with a productivity index that reflected 29 factors that can affect productivity.

3.4 Effort estimation - Bailey-Basili technique

- Minimize standard error estimate to produce an equation such as:

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

- Adjust initial estimate based on the ratio of errors.

If R is the ratio between the actual effort, E, and the predicted effort, E', then the effort adjustment is defined as

$$ERadj = R - 1 \quad \text{if } R \geq 1$$

$$= 1 - 1/R \quad \text{if } R < 1$$

- Then adjust the initial effort estimate E:

$$Eadj = (1 + ERadj)E \quad \text{if } R \geq 1$$

$$= E/(1 + ERadj) \quad \text{if } R < 1$$

3.4 Effort estimation

COCOMO model: stages of development

- application composition:
  - prototyping to resolve high-risk user interface issues
  - size estimates in object points
- early design:
  - to explore alternative architectures and concepts
  - size estimates in function points
- post architecture:
  - development has begun
  - size estimates in lines of code

3.4 Effort estimation

Table 3.10. Application point complexity levels.

For Screens				*For Reports*			
	Number and source of data tables				*Number and source of data tables*		
Number of views contained	Total < 4 (<2 server, <3 client)	Total < 8 (2-3 server, 3-5 client)	Total 8+ (>3 server, >5 client)	*Number of sections contained*	Total < 4 (<2 server, <3 client)	Total < 8 (2-3 server, 3-5 client)	Total 8+ (>3 server, >5 client)
<3	simple	simple	medium	0 or 1	simple	simple	medium
3 - 7	simple	medium	difficult	2 or 3	simple	medium	difficult
8 +	medium	difficult	difficult	4 +	medium	difficult	difficult

Table 3.11. Complexity weights for application points.

Object type	*Simple*	*Medium*	*Difficult*
Screen	1	2	3
Report	2	5	8
3GL component	-	-	10

Table 3.12. Productivity estimate calculation.

Developers' experience and capability	Very low	Low	Nominal	High	Very high
CASE maturity and capability	Very low	Low	Nominal	High	Very high
Productivity factor	4	7	13	25	50

Table 3.13. Tool use categories.

Category	*Meaning*
Very low	Edit, code, debug
Low	Simple front-end, back-end CASE, little integration
Nominal	Basic life-cycle tools, moderately integrated
High	Strong, mature life-cycle tools, moderately integrated
Very high	Strong, mature, proactive life-cycle tools, well-integrated with processes, methods, reuse

3.4 Effort estimation

Other Models

面向LOC的估算模型

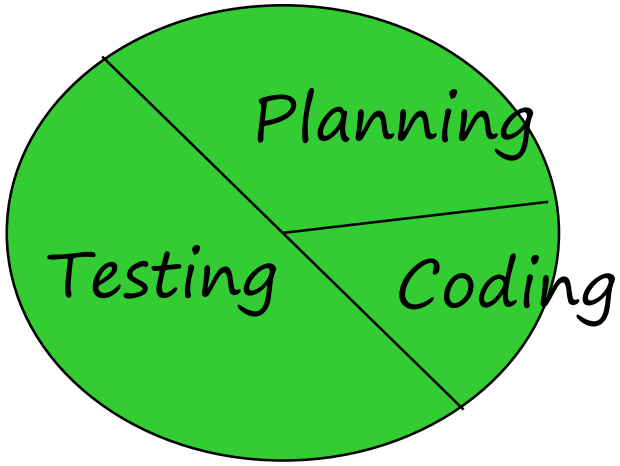
- E=5.2*(KLOC)^0.91 Walston-Felix模型
- E=5.5+0.73*(KLOC)^1.16 Bailey-Basili模型
- E=3.2*(KLOC)^1.05 Boehm模型
- E=5.288*(KLOC)^1.047 Doty模型

面向FP的估算模型

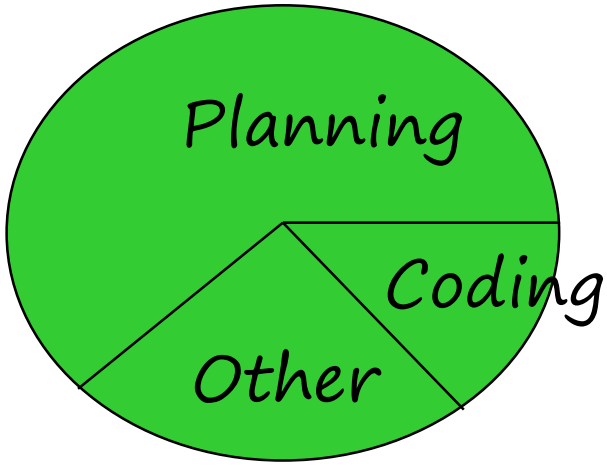
- E=-13.39+0.0545FP Albrecht 和Gaffney
- E=60.62*7.728*10^(-8)*FP^3 Kemerer
- E=585.7+5.12FP Maston、Barnett和Mellichamp

3.4 Effort estimation

- Even when estimation model produce reasonably accurate estimates, we must be able to understand which types of effort are needed during development.



Brooks

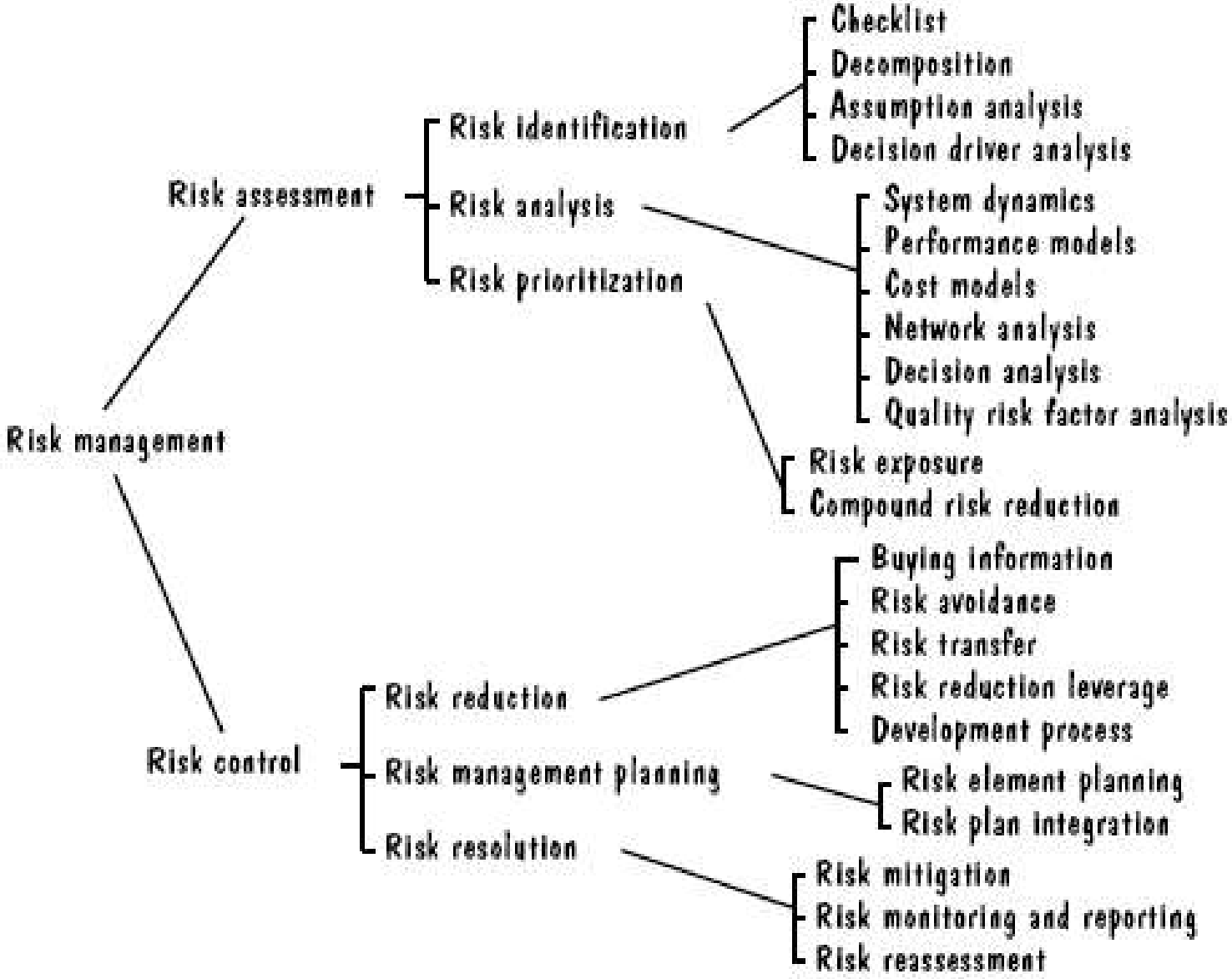


Yourdon

3.5 Risk management

- Risk management is concerned with ***identifying risks*** and drawing up ***plans to minimise their effect on a project.***
- A ***risk*** is a probability that some adverse circumstance will occur
  - Project risks affect schedule or resources;
  - Product risks affect the quality or performance of the software being developed;
  - Business risks affect the organisation developing or procuring the software.

3.5 Risk management - Steps



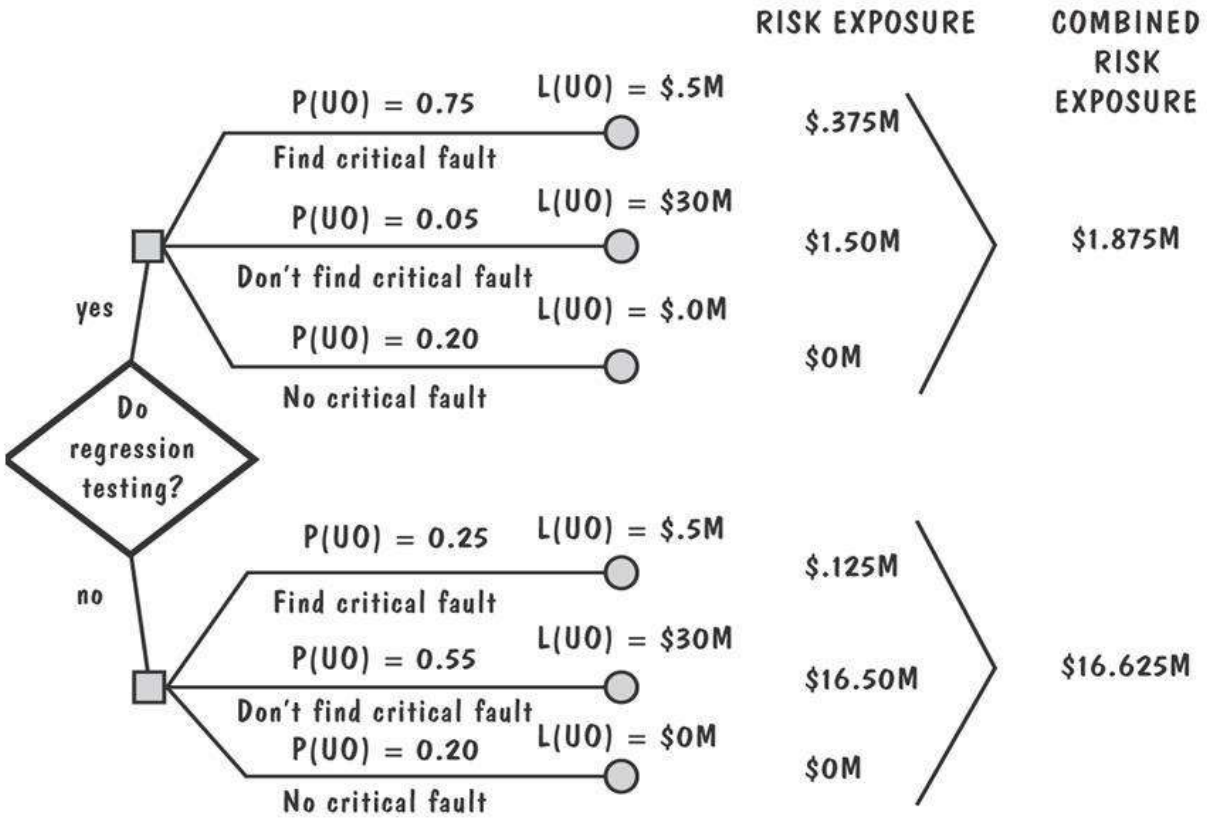
3.5 Risk management

- Risk impact（影响）：the loss associated with the event
- Risk probability（概率）：the likelihood（可能性）that the event will occur
- Risk control: the degree to which we can change the outcome（结果）

Risk exposure(成本) = (risk probability) x (risk impact)

3.5 Risk management

- Example of risk exposure calculation



3.5 Risk management - Software risks

Risk	Affects	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organisational management with different priorities.
Hardware unavailability	Project	Hardware that is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool under-performance	Product	CASE tools which support the project do not perform as anticipated
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

3.5 Risk management

Three strategies for risk reduction

- avoiding the risk: change requirements for performance or functionality
- transferring the risk: transfer to other system, or buy insurance
- assuming the risk: accept and control it

3.5 Risk management - Boehm's top ten risk items

- Personnel shortfalls
- Unrealistic schedules and budgets
- Developing the wrong functions
- Developing the wrong user interfaces
- Gold-plating
- Continuing stream of requirements changes
- Shortfalls in externally-performed tasks
- Shortfalls in externally-furnished components
- Real-time performance shortfalls
- Straining computer science capabilities

3.6 The Project Plan

Defines the work and how it is done.

It provides:

- definition of each major task
- an estimate of time and resources needed
- framework for mgmt. review and control
- benchmark to compare with actual performance

3.6 The Project Plan

Project plan structure

- Introduction.
- Project organisation.
- Risk analysis.
- Hardware and software resource requirements.
- Work breakdown.
- Project schedule.
- Monitoring and reporting mechanisms.

3.6 Project plan - contents

- project scope
- project schedule
- project team organization
- 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