

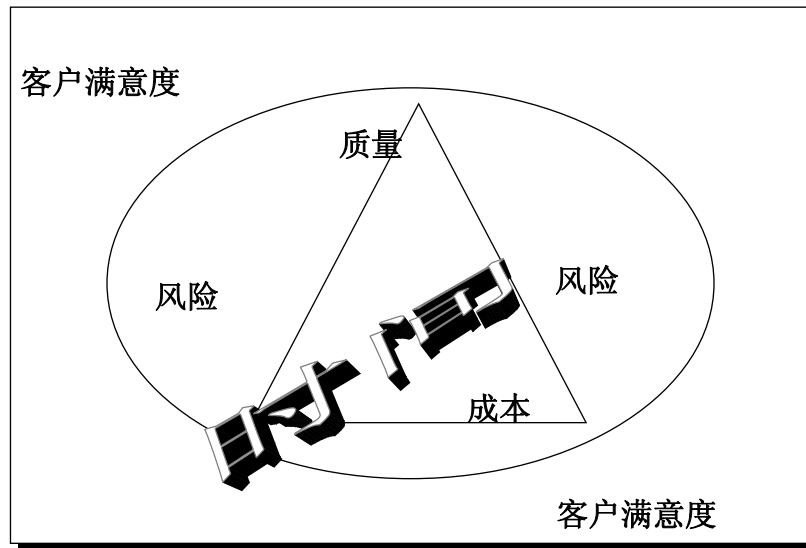
# 项目时间管理

周立新 博士

## 讲课提纲

1. 项目时间管理重要性
2. 项目时间管理
  - 活动定义
  - 活动排序
  - 活动资源估算
  - 活动历时估算
  - 进度规划
  - 进度控制
3. 有效进行时间管理

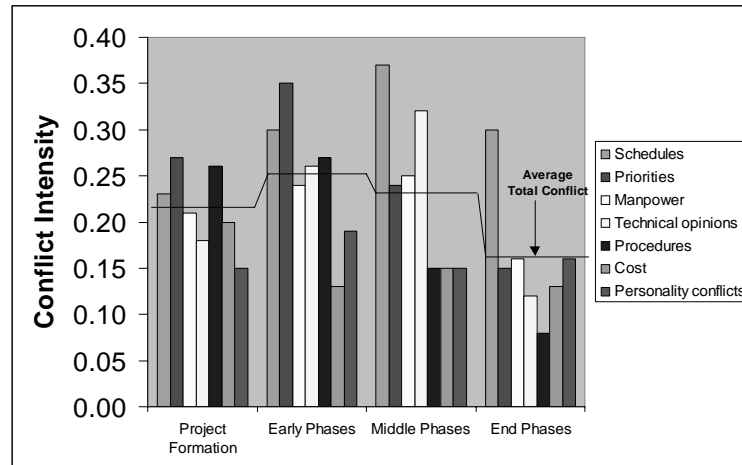
# 项目管理模板和时间



## 1. 时间在项目中为何如此重要？

- 在谈论项目时- 时间是所提及的一件重要的事情
- 大多数项目都有一个规定的交付日期, 这是因为:
  - 需求
  - 市场环境
- 进度控制是项目管理工作的一个主要的组成部分
- 时间易于测量, 缺乏弹性
- 进度问题是项目管理中最普遍的原因
- 一般只比较实际进度与计划进度, 忽略被批准的变更

Figure 5-1. Conflict Intensity Over the Life of a Project



时间是一种不可再生的  
资源

时间就是金钱

## 2. 项目时间管理

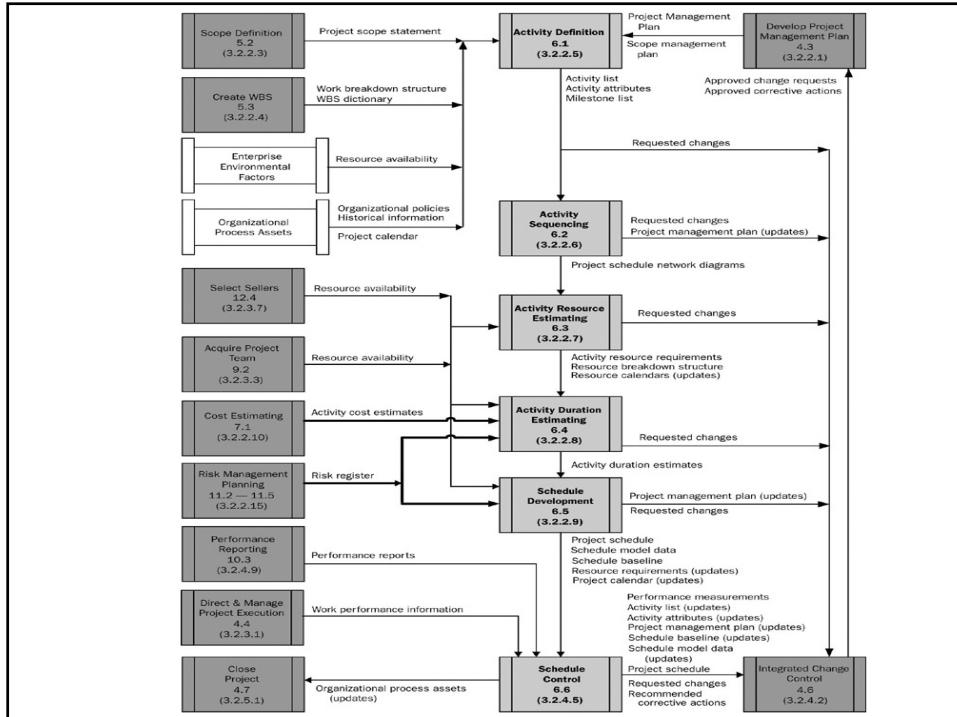
**Project Time Management includes the process required to accomplish timely completion of project.**

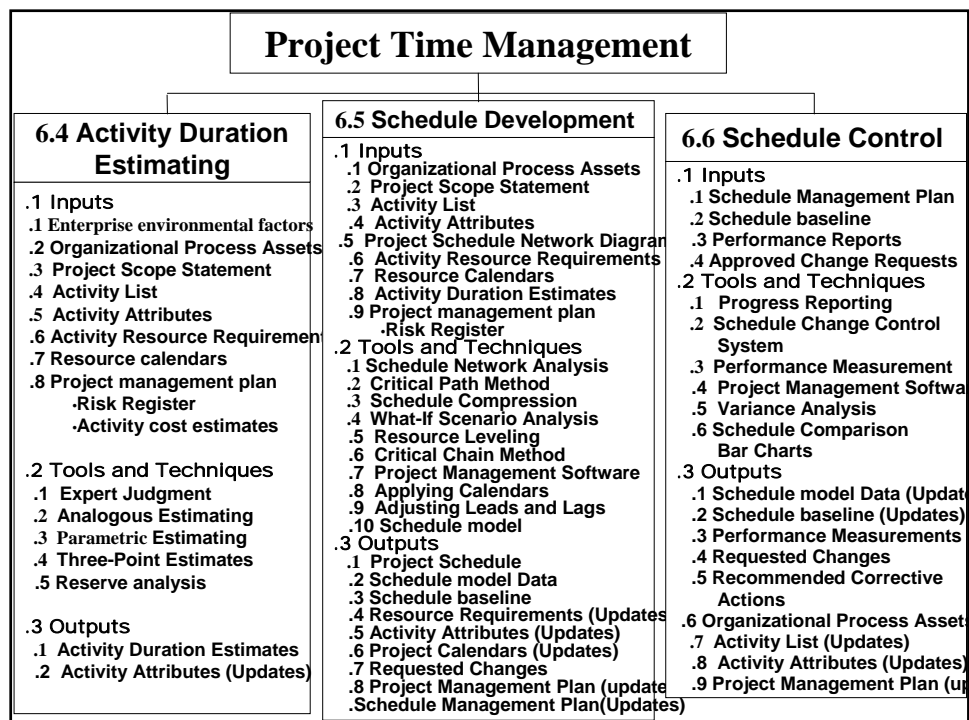
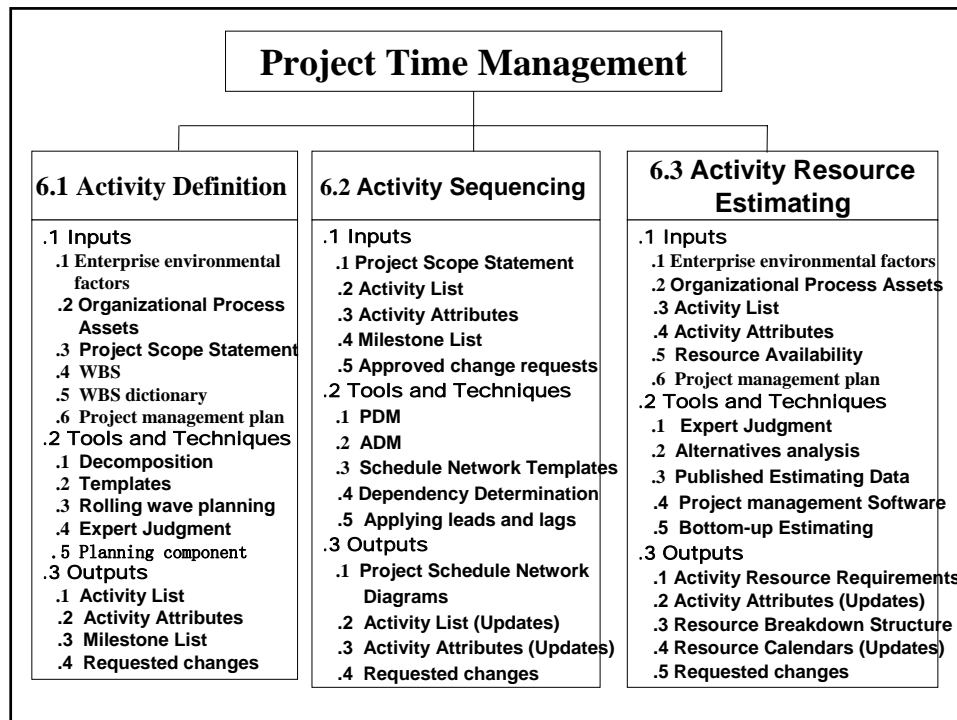
➤过程：

- 活动定义
- 活动排序
- 活动资源估算
- 活动历时估算
- 进度规划
- 进度控制

➤目标：

## 确保项目按时完成





## 2.1 活动定义

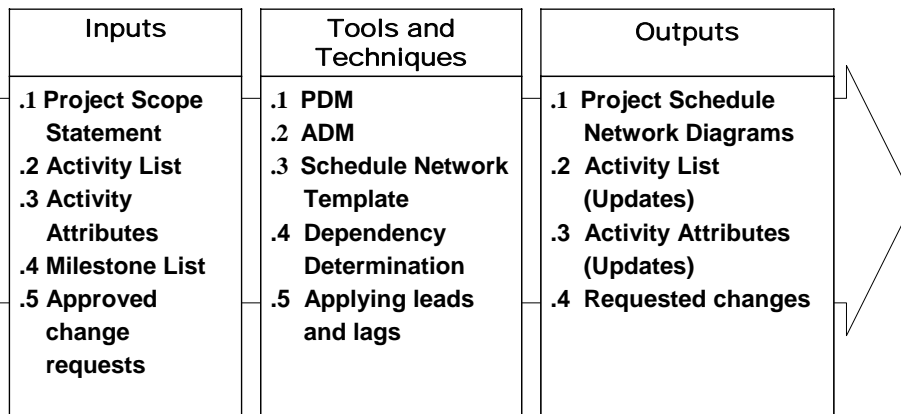
- 对工作分解结构（WBS）中规定的可交付成果或半成品的产生所必须进行的具体活动进行定义，并形成文档。

可交付成果	活 动
需求分析说明书	会晤用户
	研究现有系统
	明确用户需求
	撰写需求分析说明书
	需求评审

## 2.1 活动定义

Inputs	Tools and Techniques	Outputs
<ul style="list-style-type: none"><li>.1 Enterprise environmental factors</li><li>.2 Organizational Process Assets</li><li>.3 Project Scope Statement</li><li>.4 WBS</li><li>.5 WBS dictionary</li><li>.6 Project management plan</li></ul>	<ul style="list-style-type: none"><li>.1 Decomposition</li><li>.2 Templates</li><li>.3 Rolling wave planning</li><li>.4 Expert Judgment</li><li>.5 Planning component</li></ul>	<ul style="list-style-type: none"><li>.1 Activity List</li><li>.2 Activity Attributes</li><li>.3 Milestone List</li><li>.4 Requested changes</li></ul>

## 2.2 活动排序

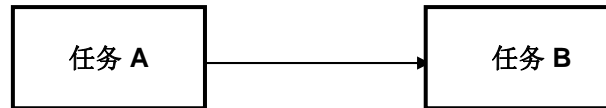


## 活动之间的逻辑关系

逻辑关系：

- 完成-开始 (FS, Finish - Start)
- 完成-完成 (FF, Finish - Finish)
- 开始-开始 (SS, Start - Start)
- 开始-完成 (SF, Start - Finish)
- 提前 (Lead) 与滞后 (Lag)

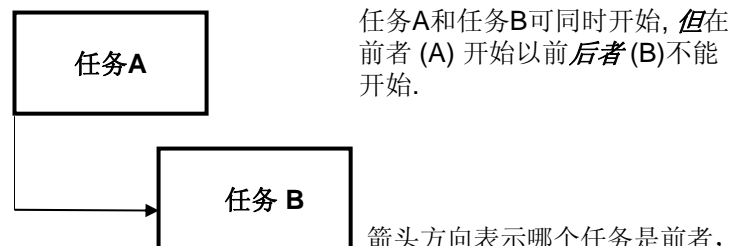
## 完成-开始 (FS, Finish - Start)



在活动任务B开始前，活动任务A必须完成

- 只有编码完成后才能进行测试

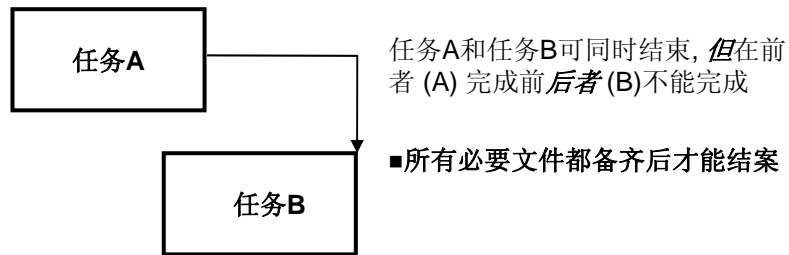
## 开始-开始 (SS, Start - Start)



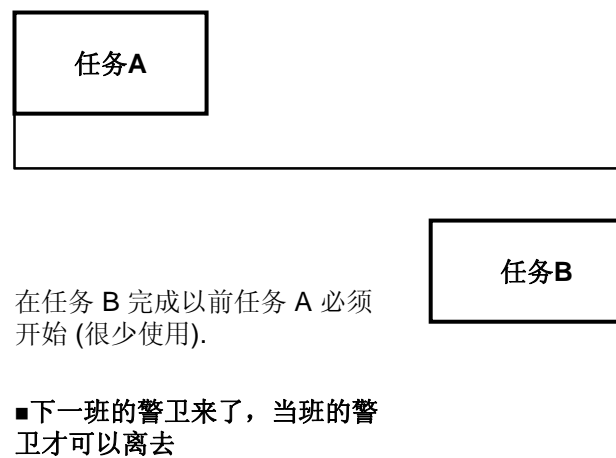
- 只有硬件安装开始后才开始软件安装



## 结束-结束 (FF, Finish - Finish)

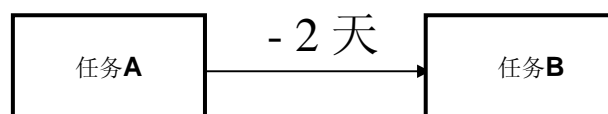


## 开始-结束 (SF, Start - Finish)



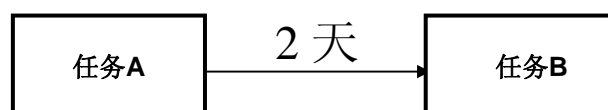
## 提前(Leading)

在任务 A 完成前2天，任务B必须开始



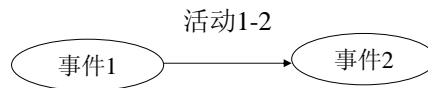
## 滞后(Lag)

任务 A 完成后2天，任务B 才能开始



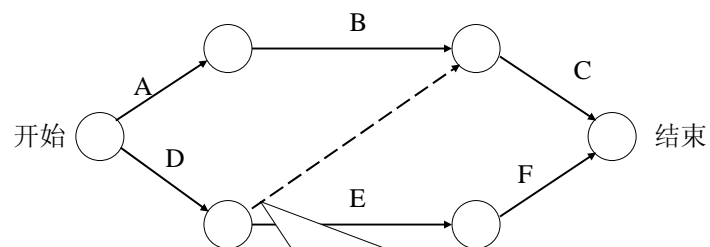
## 箭线图法（ADM）-双代号网络（AOA）

- 用箭线表示活动，用节点表示事件



- 只使用一种活动之间的逻辑关系：**FS**
- 作图要求：
  - 每一个事件必须有唯一的事件号；
  - 每一个活动必须用唯一的紧前事件和唯一的紧后事件描述；
  - 紧前事件编号要小于紧后事件编号；
- 使用虚活动

## 箭线图法（ADM）



### 虚活动

- 1) 虚活动没有历时，不需要资源
- 2) 箭线图网络表达活动关系的需要
- 3) 用带箭头的虚线表示

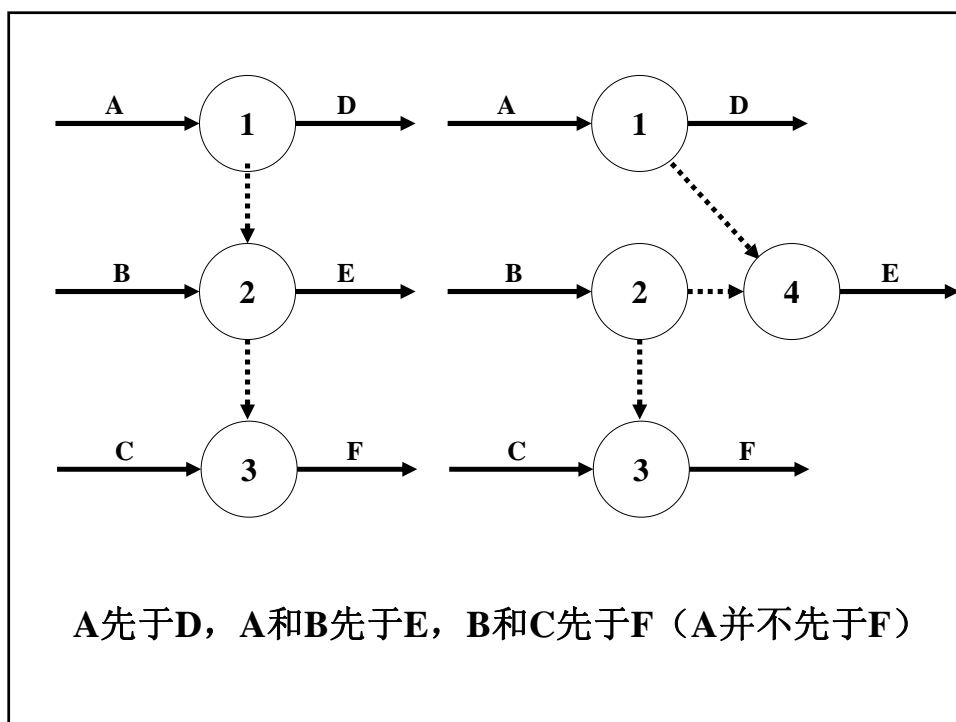
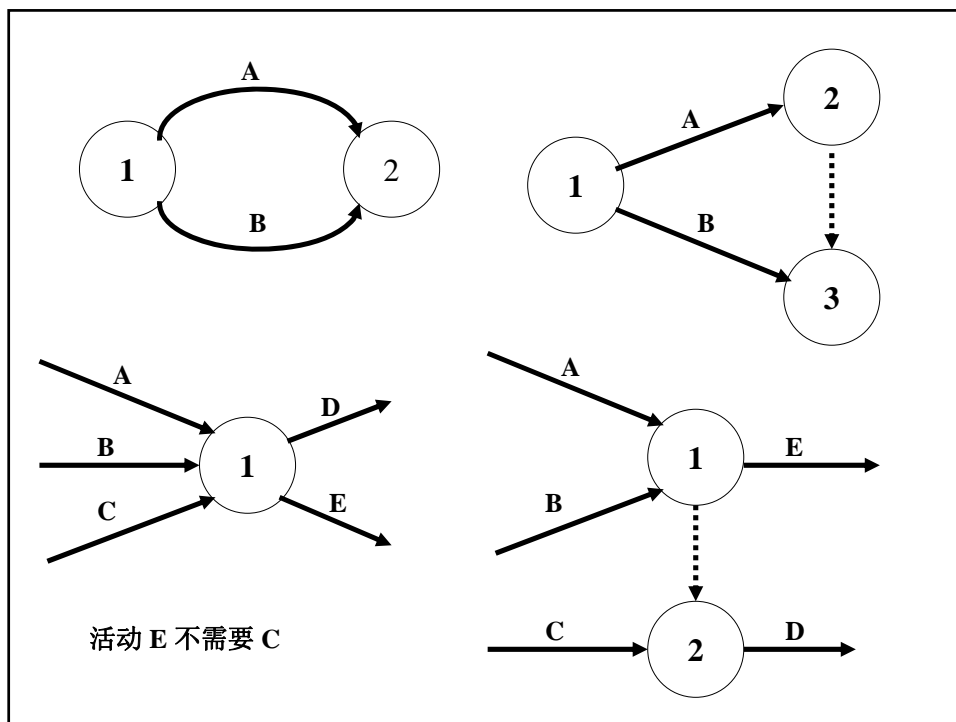
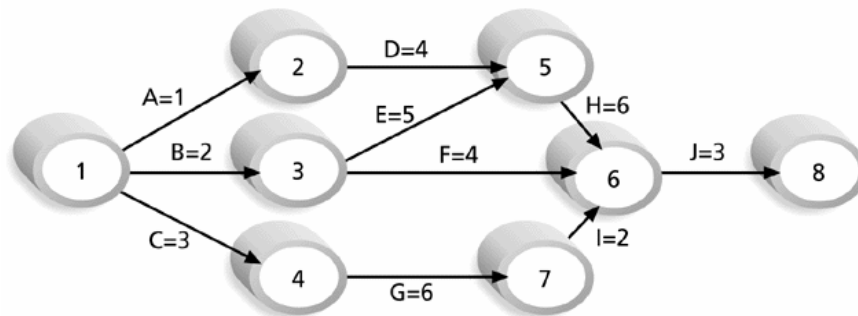


Figure 5-2. Sample Activity-on-Arrow (AOA) Network Diagram for Project X



Note: Assume all durations are in days; A=1 means Activity A has a duration of 1 day.

## 箭线图（ADM）或双代号网络（AOA）

### 练习

- 根据给定的条件，绘制箭线图
- 对每一项活动，确定：
  - 在该活动可以开始之前，那些活动必须完成；
  - 那些活动可以与该活动同时开始；
  - 那些活动只有在该活动完成之后开始；
- （找出路线最长的路径）

## 前导图法（PDM）-单代号网络（AON）

- 1) 用节点表示活动，用箭线表示活动之间的关系
- 2) 活动之间存在四种依存关系
 

结束——开始；      结束——结束  
 开始——开始；      开始——结束
- 3) 对活动增加了滞后、提前关系；
- 4) 没有虚活动
- 5) 大多数项目管理软件采用前导图法

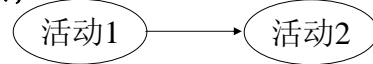
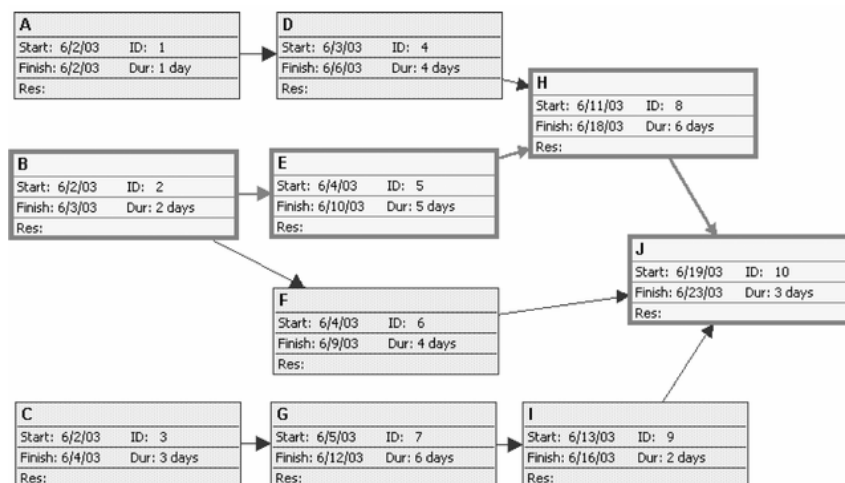
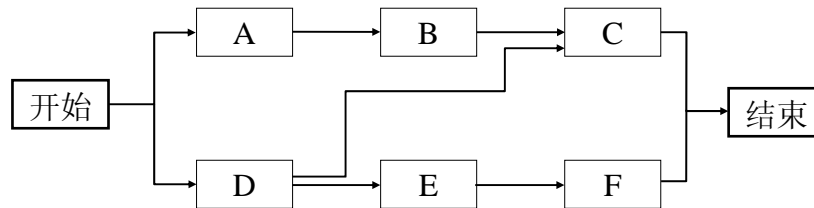


Figure 5-4. Sample Precedence Diagramming Method (PDM) Network Diagram for Project X



Project 98 file

## 前导图法（PDM）



## 前导图法（PDM） ——单代号网络（AON）练习

- 将练习所做的箭线图网络转换成前导图网络

## 2.3 活动资源估算

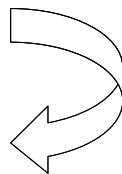
Inputs	Tools and Techniques	Outputs
<ul style="list-style-type: none"> <li>.1 Enterprise environmental factors</li> <li>.2 Organizational Process Assets</li> <li>.3 Activity List</li> <li>.4 Activity Attributes</li> <li>.5 Resource Availability</li> <li>.6 Project management plan</li> </ul>	<ul style="list-style-type: none"> <li>.1 Expert Judgment</li> <li>.2 Alternatives analysis</li> <li>.3 Published Estimating Data</li> <li>.4 Project management Software</li> <li>.5 Bottom-up Estimating</li> </ul>	<ul style="list-style-type: none"> <li>.1 Activity Resource Requirements</li> <li>.2 Activity Attributes (Updates)</li> <li>.3 Resource Breakdown Structure</li> <li>.4 Resource Calendars (Updates)</li> <li>.5 Requested changes</li> </ul>

## 资源计划编制



种类?  
数量?  
何时?

资源要满足需求并与  
实施进度相匹配





## 资源和资源平衡

### ❖资源的种类

- 劳动力 (Labour)
- 设备 (Equipment)
- 原材料 (Material)
- 其他 (Other): 指不符合上述分类的资源, 例如分包合同、租约等。

### ❖资源平衡技术

- 可储备资源与不可储备资源
- 平衡资源
- 稀缺的资源

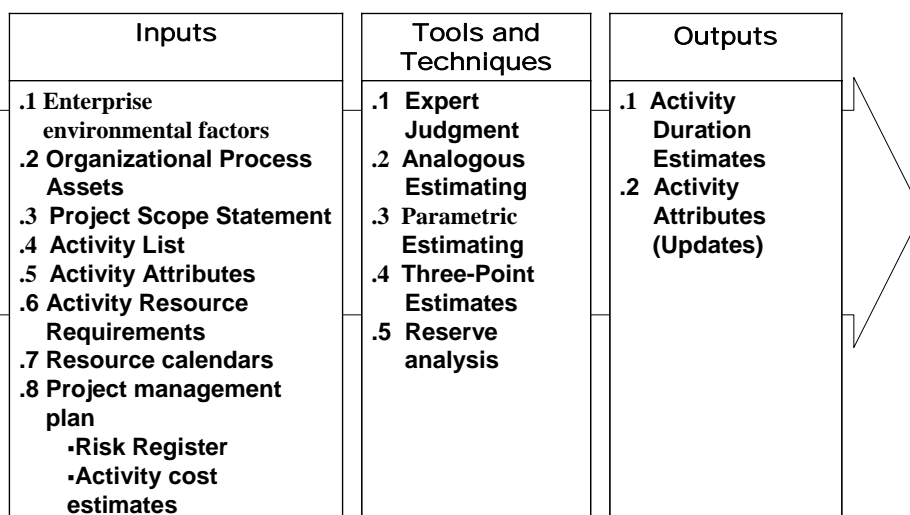
## 资源平衡的基本过程

- 建立网络, 将资源分配给各项活动。
- 首先按尽快原则安排活动——含有最迟原则或固定日期的活动除外。
- 建立资源需求直方图。
- 决定资源上限(可使用最大量), 找出资源冲突, 也就是说, 指出何时需要量超过可使用量。微软Project软件和其他项目管理出版物将这种情况称作资源过度配置 (Resource Over allocation)。
- 重新安排有松闲时间 (浮动时间) 的活动以减少资源冲突的数量。如果有可能, 可在它们的松闲时间限度内重新安排这些活动, 以避免更改项目的终止日期。

## 资源平衡的其他过程

- 增员或延长加班时间以增加资源。
- 延长项目工期，这将增强所有活动的浮动时间 (Float)，以增加重新进行进度安排的选择。
- 削减活动以减少资源需求。
- 改变网络的逻辑（例如，活动的顺序）。
- 限制项目范围。
- 将大项目细分。

## 2.4 活动历时估算



## 活动工期估计（1）

根据生产能力信息计算：

- 活动所需要的资源量=活动工作量/单位资源生产能力
- 活动工期=活动所需要的资源量/（单位时间可得资源量）

## 咨询专家

- 注意：
  - 需要最新的知识
  - 源于对项目的理解
  - 考虑新的技术/方法

## 通过类比 - 举例

- 与先前的项目类似 – 是其规模的150%，因此也要多花150%的时间 (less learning curve)
- 通常花3周时间准备一个合同
- 建设阶段大约花12个月

## 参数估计

- 来自于对所需数量的了解和正常的生产能力水平
- 例如：
  - 25 公里长的公路 – 每公里大约要1个月
  - 50,000线路 – 每月大约10,000 条线
  - 5架飞机-每月一架
  - 2天编写一个存储过程
- 记住要设定时间

FP  
COCOMO

## 历史信息

- 项目收尾阶段的一个任务即是收集信息
- 一些行业公布的图表：
  - 生产能力
  - 所需的时间
- 记住 – 即使你项目上的专职人员也有其他的组织任务

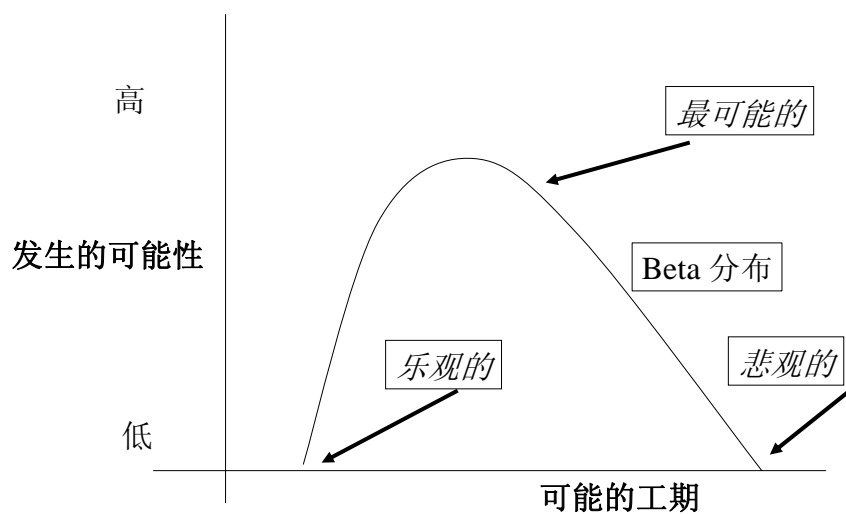
## 估计的准确性

每个行业在不同阶段  
都有不同的精确标准

## 计划评审法（PERT）

- 美国海军在1958年开发了计划评审技术PERT，北极星导弹系统项目
- PERT 方法对每个活动的工期有3种估计：
  - 1) 乐观的；
  - 2) 最可能的；
  - 3) 悲观的；并假定活动工期服从贝塔分布；

### PERT –活动工期



## PERT –活动工期计算

- 每个活动的平均工期  
= (乐观的+4(最可能的)+ 悲观的) /6

$$T_e = (O + 4M + P) / 6$$

- 每个活动工期的标准差 ( $\sigma$ )  
= (乐观的- 悲观的) /6
- 每个活动工期的方差 ( $\sigma^2$ )  
= ((乐观的- 悲观的) /6)<sup>2</sup>

活动A工时为3天并开始于星期一早晨，4号。后继活动B与A具有完成一开始依赖关系。完成一开始关系有3天的滞后，而且活动B工时为4天。星期日是非工作日。从这些数据可以得出什么结论？

- A. 两项活动总工时为8天
- B. 活动A开始到活动B完成之间的日历时间 (Calendar time) 是11天
- C. 活动B完成日期是星期三，13号
- D. 活动A开始与活动B完成之间日历时间为14天

## 2.5 进度规划

Inputs	Tools and Techniques	Outputs
.1 Organizational Process Assets .2 Project Scope Statement .3 Activity List .4 Activity Attributes .5 Project Schedule Network Diagrams .6 Activity Resource Requirements .7 Resource Calendars .8 Activity Duration Estimates .9 Project management plan •Risk Register	.1 Schedule Network Analysis .2 Critical Path Method .3 Schedule Compression .4 What-If Scenario Analysis .5 Resource Leveling .6 Critical Chain Method .7 Project Management Software .8 Applying Calendars .9 Adjusting Leads and Lags .10 Schedule model	.1 Project Schedule .2 Schedule model Data .3 Schedule baseline .4 Resource Requirements (Updates) .5 Activity Attributes (Updates) .6 Project Calendars (Updates) .7 Requested Changes .8 Project Management Plan (updates) •Schedule Management Plan(Updates)

### 制定进度计划（1）

- 确定项目的开始和结束时间
- 在不考虑资源约束的条件下，计算项目的工期，包括：
  - 1) 网络的关键路径
  - 2) 各个活动的自由浮动时间
- 计算资源需求计划



## 关键路径

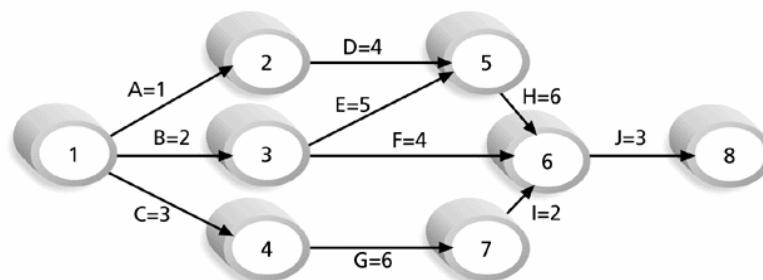
- 在ADM（双代号）网络图中，最长的一条路径称作关键路径，关键路径就是项目的工期。

计算练习题目（ADM）的关键路径！

- 在PDM（单代号）网络图中，活动的自由浮动时间等于零的一条路径就是关键路径，关键路径长度等于项目的工期。

– 杜邦公司在1957年提出关键路径法CPM

**Figure 5-8. Determining the Critical Path for Project X**



Note: Assume all durations are in days.

Path 1: A-D-H-J Length = 1+4+6+3 = 14 days

Path 2: B-E-H-J Length = 2+5+6+3 = 16 days

Path 3: B-F-J Length = 2+4+3 = 9 days

Path 4: C-G-I-J Length = 3+6+2+3 = 14 days

Since the critical path is the longest path through the network diagram, Path 2, B-E-H-J, is the critical path for Project X.

## 自由浮动 (时差)

- 定义 – 在没有延误任何随后活动的最早开始日期的情况下，一项活动可以被推迟的时间数量。
- Free Float,

## 总体浮动(时差)

- 定义 – 在没有延误项目最早结束日期的情况下，一项活动从最早开始日期被推迟的时间长度，称之为总体浮动时间。
- Total Float
- 总体浮动时间的计算：  
使用正推法计算最早开始日期 (ES)，最早结束日期(EF)，  
使用逆推法计算最迟开始日期(LS)，最迟结束日期(LF)，  
总体浮动=  $LS - ES$  或  $LF - EF$

## 正推法

- 正推法

- 从网络图左边开始，为每项任务制定最早开始和最早结束日期，进行到网络图结束（最右边）

- 最早开始日期 (ES)

- 基于网络逻辑和进度约束条件，一项活动开始的可能的最早时间。

- 最早结束日期(EF)

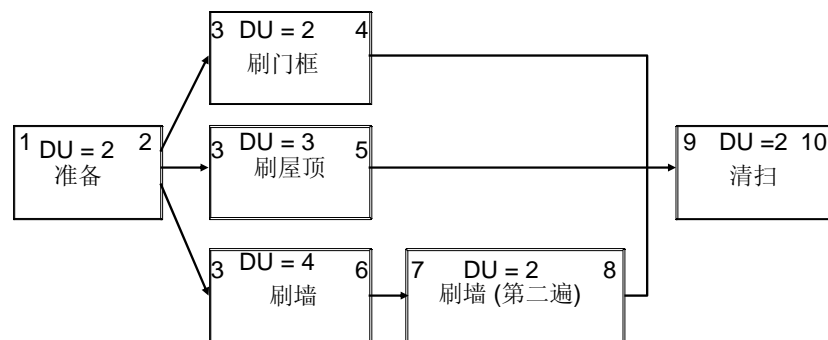
- 活动可完成的可能的最早时间

## 正推法计算

$$ES = \text{MAX}\{EF_i\} + 1$$

$$EF = ES + DU - 1$$

ES	工期	EF
LS	任务	LF
	浮动	



## 逆推法

- 逆推法

- 以项目完成日期为开始日期，使用结束时间和逆向工作，计算最迟开始和最迟结束日期

- 最迟开始日期 (LS)

- 在没有延误一项活动随后任务的情况下，可开始该任务的最迟时间。

- 最迟结束日期 (LF)

- 在没有延误一项活动之后任务的情况下，任务可被完成的最迟时间

## 逆推法计算

$$LF = \min\{LS_j\} - 1$$

$$LS = LF - DU + 1$$

ES	工期	EF
LS	任务	LF
	浮动	

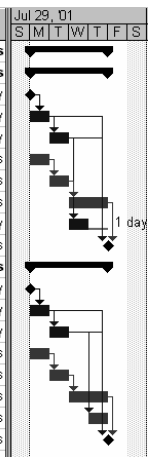


## 练习

计算练习题目中单代号网络的关键路径以及各个活动的自由浮动时间。

## 自由浮动演示

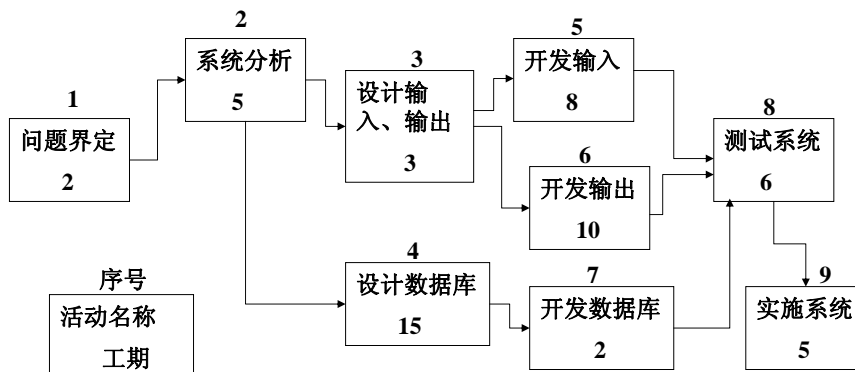
	Task Name	Start	Duration	Finish	Free Slack	Total Slack	Jul 29, '01						
							S	M	T	W	T	F	S
1	Float Demonstration	Jul 30 '01	4 days	Aug 2 '01	0 days	0 days							
2	All as soon as possible	Jul 30 '01	4 days	Aug 2 '01	0 days	0 days							
3	Start	Jul 30 '01	0 days	Jul 30 '01	0 days	1 day							
4	Task A	Jul 30 '01	1 day	Jul 30 '01	0 days	1 day							
5	Task B	Jul 31 '01	1 day	Jul 31 '01	0 days	1 day							
6	Task C	Jul 30 '01	1 day	Jul 30 '01	0 days	0 days							
7	Task D	Jul 31 '01	1 day	Jul 31 '01	0 days	0 days							
8	Task E	Aug 1 '01	2 days	Aug 2 '01	0 days	0 days							
9	Task F	Aug 1 '01	1 day	Aug 1 '01	1 day	1 day							
10	End	Aug 2 '01	0 days	Aug 2 '01	0 days	0 days							
11	Task F - as late as possible	Jul 30 '01	4 days	Aug 2 '01	0 days	0 days							
12	Start	Jul 30 '01	0 days	Jul 30 '01	0 days	1 day							
13	Task A	Jul 30 '01	1 day	Jul 30 '01	0 days	1 day							
14	Task B	Jul 31 '01	1 day	Jul 31 '01	0 days	1 day							
15	Task C	Jul 30 '01	1 day	Jul 30 '01	0 days	0 days							
16	Task D	Jul 31 '01	1 day	Jul 31 '01	0 days	0 days							
17	Task E	Aug 1 '01	2 days	Aug 2 '01	0 days	0 days							
18	Task F	Aug 2 '01	1 day	Aug 2 '01	0 days	0 days							
19	End	Aug 2 '01	0 days	Aug 2 '01	0 days	0 days							



任务	活动	工期估计(周)	最早		最迟		总时差
			开始时间	结束时间	开始时间	结束时间	
1	收集数据	3	0	3	-8	-5	-8
2	可行性分析	4	0	4	-9	-5	-9
3	准备问题界定报告	1	4	5	-5	-4	-9
4	与用户见面	5	5	10	-4	1	-9
5	研究现有系统	8	5	13	-2	6	-7
6	明确要求	5	10	15	1	6	-9
7	准备系统分析报告	1	15	16	6	7	-9
8	数据输入、输出	8	16	24	9	17	-7
9	数据处理和建立数据库	10	16	26	7	17	-9
10	分析评估	2	26	28	17	19	-9
11	准备系统设计报告	2	28	30	19	21	-9
12	开发软件	15	30	45	21	36	-9
13	开发硬件	10	30	40	26	36	-4
14	开发网络	6	30	36	30	36	0
15	准备开发报告	2	45	47	36	38	-9
16	软件测试	6	47	53	38	44	-9
17	硬件测试	4	47	51	40	44	-7
18	网络测试	4	47	51	40	44	-7
19	准备测试报告	1	53	54	44	45	-9
20	培训	4	54	58	45	49	-9
21	安装	2	54	56	47	49	-7
22	准备实施报告	1	58	59	49	50	-9

## 练习

计算下列各项活动的ES、EF、LS、LF以及时差，找出项目的关键路径，试问该项目能否在30周内完成？



## 尽快原则，最迟原则，和固定日期

- 从关键路径之外的活动中可获得一些自由浮动时间（float, also called slack）。
- 在这些活动完成过程中存在一个窗口时间（windows of times），这个窗口的极限是：
  - 尽快原则（As Soon As Possible, ASAP）
  - 最迟原则（As Late As Possible, ALAP）
- 大量的项目计划以ASAP为前提假设，但当ALAP与项目内容更相关的时候可以使用。
- 固定日期（Fixed Dates）是另一种选择。

## 制定进度计划（2）

第一次：项目工期很可能难以接受或者资源需求数量得不到满足或者变化很大。

### 1) 查明工期长的原因

- 先检查逻辑关系 – 再检查关键路径
- 然后考虑工期压缩技术

### 2) 检查资源需求状况

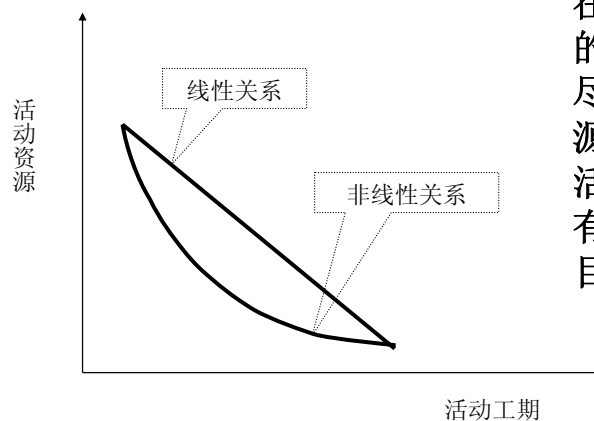
考虑使用资源平衡技术

反复几次，满足项目工期和资源约束条件！

## 缩短项目工期方法

- 在进度计划制订后，可能项目结束日期不满足要求，需要进一步缩短进度计划的项目工期。
- 可以采用赶工（Crashing, 通过增加资源）
- 或快速跟进（Fast tracking, 通过并行施工）等方法。

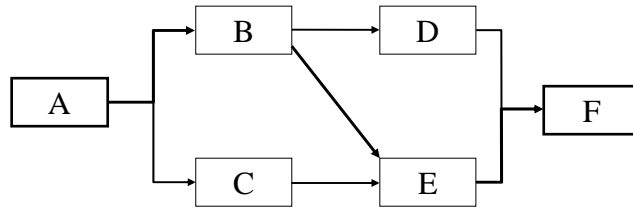
### 赶工（Crashing）——增加资源



在资源有限的条件下，尽可能把资源用到关键活动上，以有效压缩项目工期！

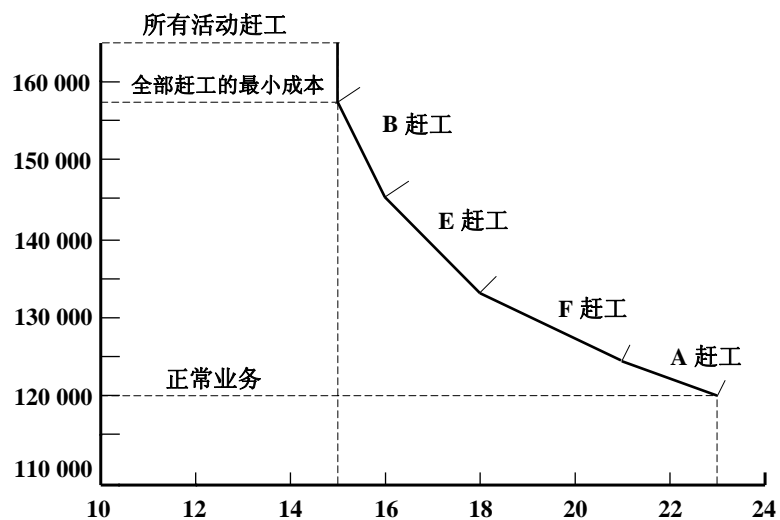


## “赶工”示例



活动	要求的时间/周		费用/美元		每周压缩成本/美元
	正常	压缩	正常	压缩	
A	4	2	10 000	14 000	2 000
B	6	5	30 000	42 500	12 500
C	2	1	8 000	9 500	1 500
D	2	1	12 000	18 000	6 000
E	7	5	40 000	52 000	6 000
F	6	3	20 000	29 000	3 000

## “赶工”示例



## 快速跟进（**Fast tracking**） 并行施工

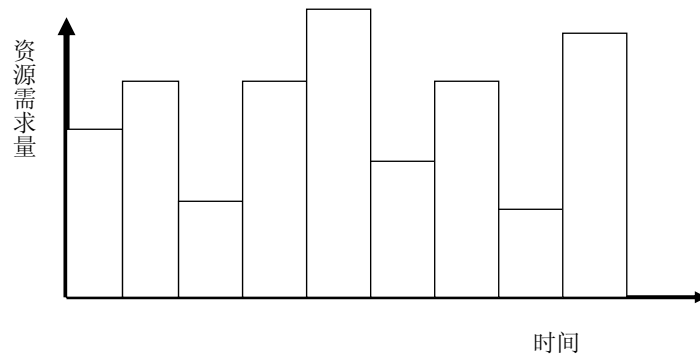
- 将一些一般顺序进行的任务改为并行实施，比如，软件项目设计完成之前开始写代码。
- 并行施工经常导致返工，一般要增加风险。

## 项目工期优化

- 项目工期不是越短越好！
- 工期压缩、并行施工需要支付代价。
- 资源的数量可能受到限制，不是无限的。
- 通过增加资源压缩工期有时是无效的。
- 一般地，考虑在资源约束下，将项目的工期、费用、收益进行综合考虑，选择科学的项目工期！

## 计算资源需求计划

- 根据每一个活动对资源的需求，计算各个单位时间内项目对资源的需求数量。



## 资源使用状况视图例

	资源名称	工时	详细信息	四	五	六
2	总监	2,176 工时	工时 成本 累计成本 累计工时	8工时 ¥100.00 ¥20,900.00 1,672工时	8工时 ¥100.00 ¥21,000.00 1,680工时	
	可行性分析	240 工时	工时 成本 累计成本 累计工时			
	总体设计	480 工时	工时 成本 累计成本 累计工时			
	软件开发	640 工时	工时 成本 累计成本 累计工时			
	工艺设计	312 工时	工时 成本 累计成本 累计工时			
	外型设计	240 工时	工时 成本 累计成本 累计工时			

## 资源平衡（Levelling）

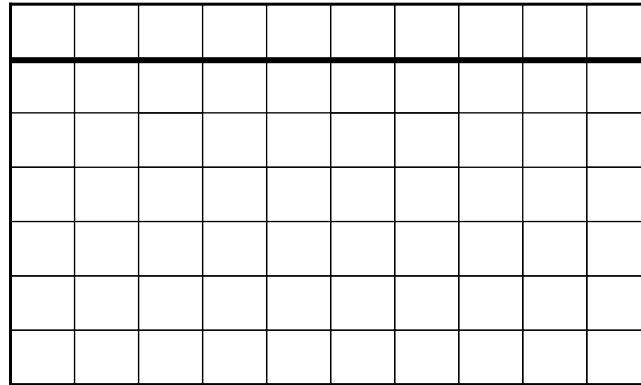
- 根据资源计划的分析，可能出现：
- 资源需求超过可得资源限制数量
- 或者资源需求量变化比较大，资源管理困难。

## 资源平衡（Levelling）

- 通过调整任务的工期或者次序，使对资源的需求尽可能在直方图上表现的平缓并不超过资源限量。
- 一般地，通过调整非关键路径的活动实现。  
向关键路径上要进度, 向非关键路径上要资源

# 资源平衡

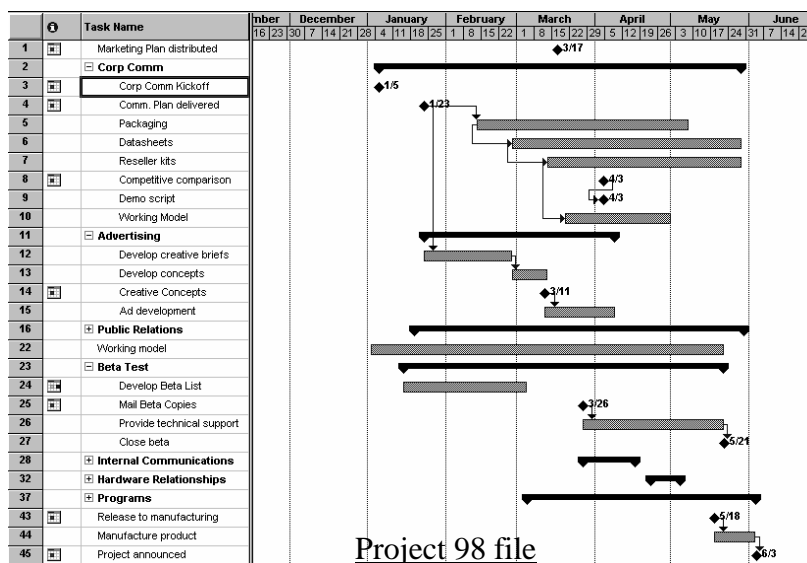
技术工人数量



可用资源数量

时间/月

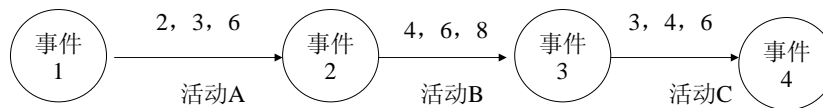
Figure 5-6. Gantt Chart for Software Launch Project



## PERT –活动工期的计算

- 项目的工期：是关键路线上每个活动平均工期的和
- 项目的方差：是关键路线上每个活动方差的和；
- 项目的标准差：是项目方差的平方根；

## PERT –活动工期计算举例



注：箭线下面的符号表示活动名称，上面的三个数字表示三种可能的活动时间

## PERT –活动工期计算举例

项 活动	O、M、P	平均工期	标准差	方差
A	2, 3, 6	3.33	4/6	16/36
B	4, 3, 8	6.00	4/6	16/36
C	3, 4, 6	4.17	3/6	9/36
估计的项目总工期		13.50	1.067	41/36

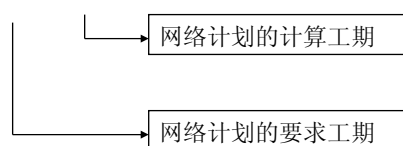
计算结果：项目的工期是13.50天，标准差是1.067天。

## PERT –活动工期计算举例

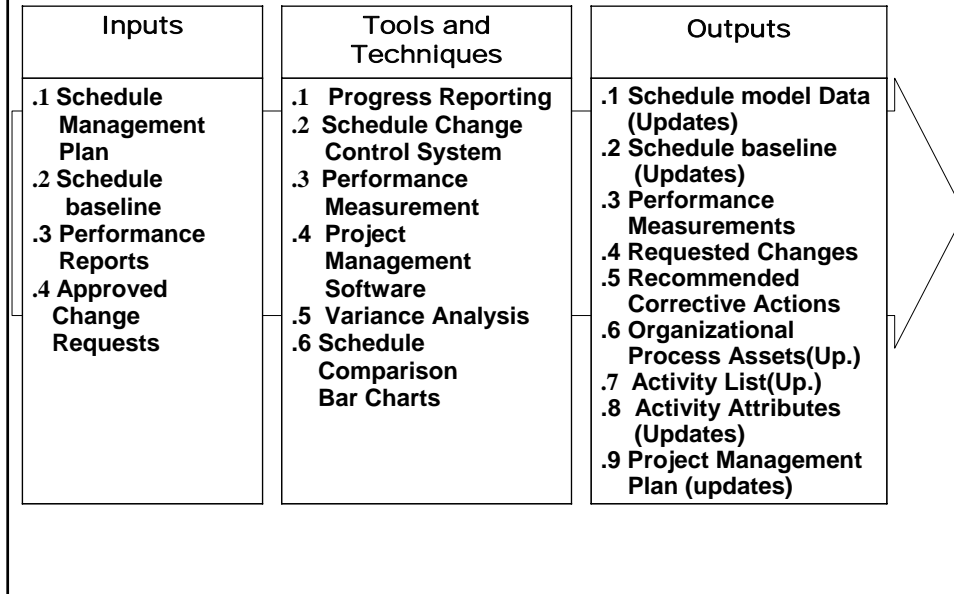
试问：项目在14.6天完成的概率大约有多大？

解法一：直接从标准差图中观察。

解法二：计算完成网络计划的可能性值，然后查表求解。



## 2.6 进度控制



### 进度控制

影响项目工期计划变化的因素进行事前控制

进度的度量和采取纠偏措施

对计划变更进行控制



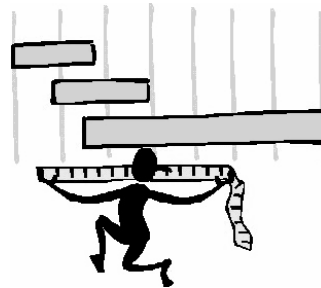
## 绩效度量

- 进度度量指标的复杂性
- 度量周期的选择

## 搜集数据并将数据转化为信息

- 搜集数据

- 沟通
  - 原因
  - 需求
  - 格式
- 衡量进度



- 将数据转化为信息

- 分析/
- 加工/
- 理解

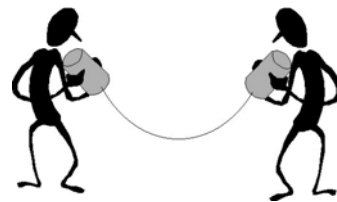
## 报告计划

- 为什么需要报告?
- 何时需要报告?
- 哪种报告?
- 向谁报告?
- 使用什么格式?
- 报告方式



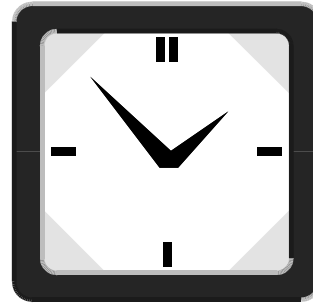
## 为何需要报告?

- 理解/追踪进度
- 监督和预测
- 识别变更等级需求
- 注意/事先警告
- 激发
- 记录
- 沟通



## 需要何时汇报?

- 日常报告
- 例外报告
- 特别分析报告



## 报告类型

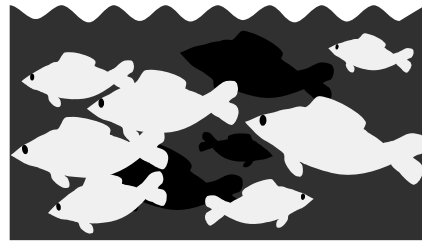
- 进度计划执行情况
- 项目关键点
- 项目执行状态
- 任务完成报告
- 重大突发事件
- 项目变更
- 项目进度
- 项目管理报告

参考讲义材料



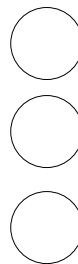
## 向谁报告？

- 项目经理
- 项目业主/客户
- 项目团队
- 视情况而定



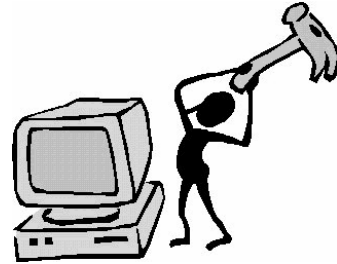
## 使用何种格式？

- 表格
- 横道图
- 文本
- 图表
- 红绿灯



## 报告方式

- 提交
  - 适当
- 散发
  - 硬拷贝
  - 电子
  - 及时
  - 被记录或被控制
- 储存



## 控制进度变更

- 进度计划的实际检查
- 处理人的问题
  - 授权——项目团队成员对他们的活动负责
  - 激励——罚 1 元钱
  - 纪律——Deadline
  - 谈判：
    1. 技术人员内向->外向（销售部门、高层）
    2. 年轻->年长
    3. 坚持立场

## 项目进度追回的方法

- 对近期内即将发生的活动进行控制
- 工期估计最长或预算估计最大的活动
- 加班或追加人员
- 更有经验的成员
- 外包或缩小范围
- 变更操作或施工方法
- 分解任务或重排任务
- 加强沟通与知识共享
- 实行变更控制与版本管理

## 3.项目成员的时间管理

- 每个周末，找出几个（2~5个）下周要完成的目标。
- 每天结束时，列出第二天要做的事情。
- 早晨的第一件事是看一下这个做事表，一整天都要看到这个表。
- 控制干扰。（电话、Email、随意的来访者）
- 学会说“不”。
- 有效利用等待时间。
- 尽量一次处理大部分文件工作。
- 周末，如果你完成了全部目标，就奖赏自己。

# 进度管理软件

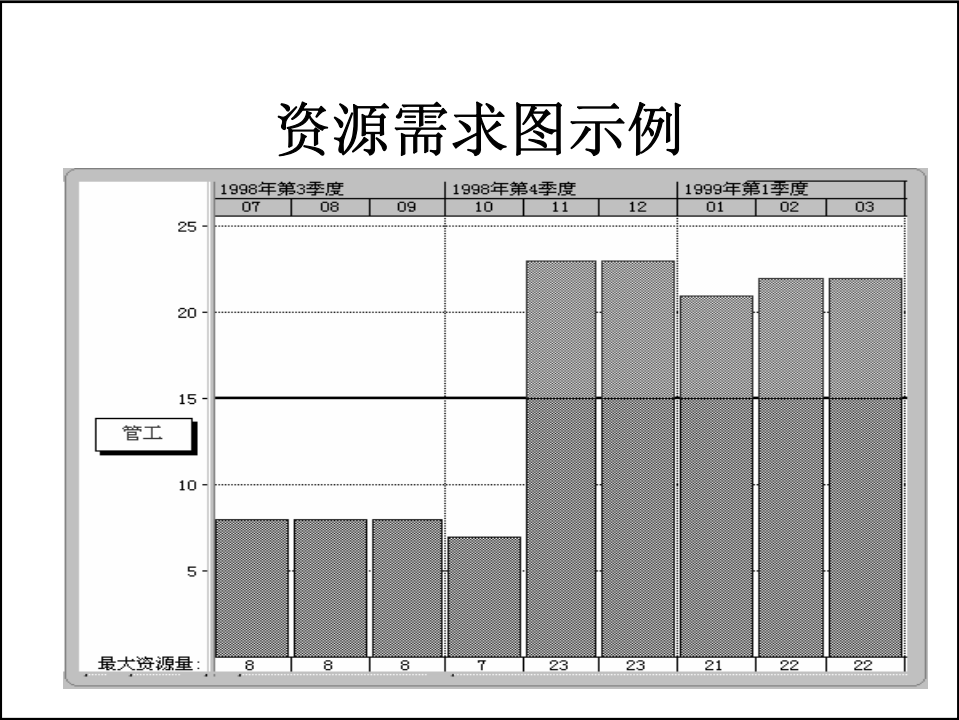
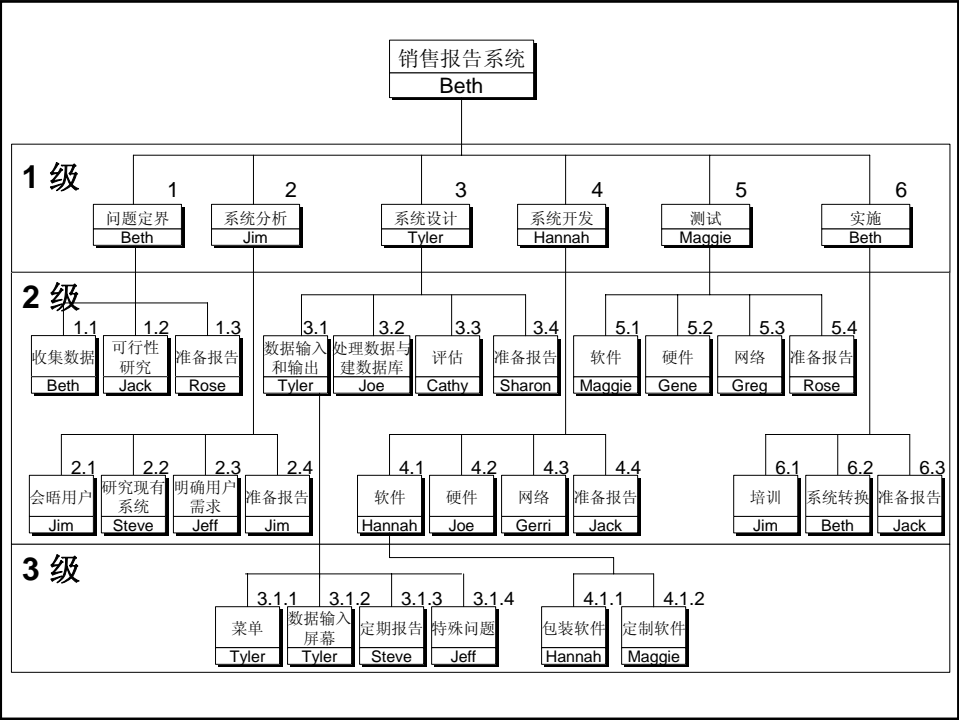
- 1 进度管理软件的发展
- 2 进度管理软件的主要功能
  - 项目的建立与进度计划
  - 时间进度分析网络处理
  - 资源分配处理和优化
  - 成本计算、分析、预测
  - 项目计划跟踪、更新管理
  - 多项目管理与多项目资源优化
  - 数据交换和开放的数据艰苦连接
  - 图表生成和输出
  - 编程和再开发
  - 支持计算机网络，实现应用集成功能

# 项目管理软件

- MS Project

大众化的项目管理软件，50%以上的项目管理者使用该软件
- P<sub>3</sub>（Primavera Project Planner）

专业级项目管理软件，83年第一版，主要基于CPM方法





## 其他几种网络计划表示方法

- 时标网络计划
- 搭接网络计划
- 平衡线网络计划
- 非肯定型网络计划

## 时标网络计划

- 时标网络计划：是以时间坐标为尺度绘制的网络图。箭线的长短表示工作的工期。
- 可以用于计算资源需求量。

## 搭接网络计划

搭接网络计划：

用标注相邻工作时间间隔来表示活动之间搭接关系的网络计划。

实际上，搭接网络计划是在单代号网络计划技术的基础上，增加了相邻活动之间的逻辑关系，不再是单一的结束—开始关系。

## 基线

## 里程碑

### 1. 基线（D、R、D）

- a. 业已经过正式审核与同意，可用作下一步开发的基础，并且只有通过正式的修改管理步骤方能加以修改的规格说明或产品。
- b. 在配置项目生存周期的某一特定时间内，正式指定或固定下来的配置标识文件和一组这样的文件。基线加上根据这些基线批准同意的改动构成了当前配置标识。对于配置管理，有以下三种基线：
  - 功能基线——最初通过的功能配置；
  - 分配基线——最初通过的分配的配置；
  - 产品基线——最初通过的或有条件地通过的产品配置。

### 2. 里程碑

项目有关人员或管理人员负责的在预定时间将发生的事件，用来标志工作进度。例如，正式的复审、规格说明的颁布、产品的交付。

——GB/T 11457-1995 软件工程术语