

# Review-软件项目管理--进度计划

by: 2022211928 gwc

## 进度的定义

进度是对执行的活动和里程碑制定的**工作计划日期表**

## 进度计划的重要性

- 按时完成项目是项目经理**最大的挑战之一**
- **时间**是项目规划中**灵活性最小**的因素
- 进度问题是**项目冲突的主要原因**

## 项目进度计划制定过程

任务定义-->任务关系-->历时估算-->项目进度编排-->项目进度优化

### --任务

定义：为完成项目的各个交付成果所必须进行的诸项具体活动

### --任务关系

关联关系，顺序关系

**前置活动（任务） --> 后置活动（任务）**

PS: 关系依赖矩阵 ( $d_i$ 是 $d_j$ 的前置, 则关系依赖矩阵中 $d_{ij} = 1$ .)

### --历时估算

**内容：**估计任务、路径、项目的持续时间

**方法：**

- 定额估算法  
$$T = Q / (R * S)$$

T: 活动历时    Q: 任务工作量    R: 人力数量    S: 工作效率(贡献率)
- 经验导出模型 (**工作量->历时估算**)  
$$D = a * E^b$$

D: 进度(以月单位)    E: 工作量(以人月单位)    a: 2-4之间    b: 1/3左右: 依赖于项目的自然属性
- CPM(关键路径法估计)  
确定项目网络图  
每个任务有单一的历时估算  
确定网络图中任务的**逻辑关系**  
**关键路径是网络图中最长的路径**  
**关键路径可以确定项目完成时间**
- 类比估算
- 专家判断  
根据这些**专业知识**而做出的历时估算: 1.进度计划 2.有关估算经验 3.学科或应用知识

- 基于承诺的估计

内容：要求开发人员做出进度承诺，不进行中间的工作量（规模）估计

优点：有利于开发者对进度的关注，**充分发挥主观能动性**

## --项目进度管理图示

根据项目任务的执行排序、历时估算及所需资源等进行分析，制定计划

进度制定的主要工具： 1.甘特图 2.里程碑计划 3.网络图

## 网络图

网络图是活动排序的一个输出，展示项目中各个活动以及活动之间的逻辑关系

- PDM (Precedence Diagramming Method) 优先图法, 节点法 (单代号) 网络图

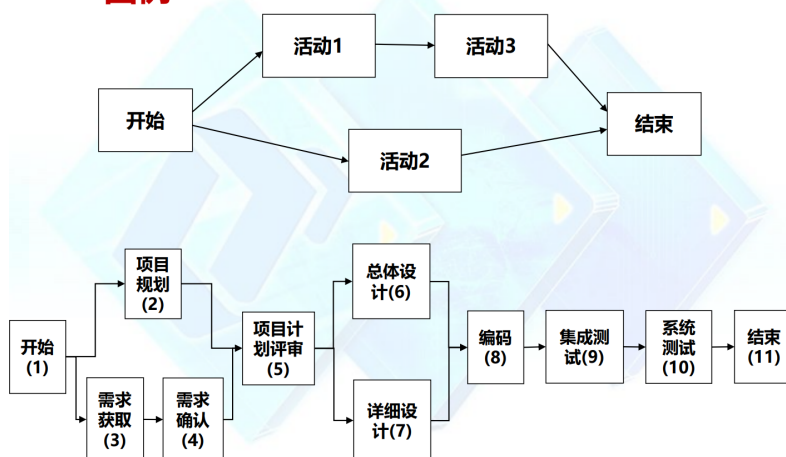
构成PDM网络图的基本元素是节点(Box)

节点(Box) 表示活动 (任务)

用箭线表示各活动(任务)之间的逻辑关系

可以方便的表示活动之间的各种逻辑关系

## PDM 图例



- **ADM (Arrow Diagramming Method) 箭线法 (双代号)网络图**

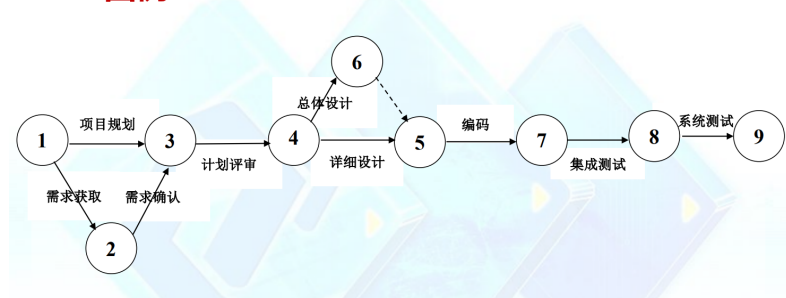
ADM也称为双代号项目网络图

在ADM网络图中, 箭线表示活动(任务)

两个代号唯一确定一个任务

代号表示前一任务的结束, 同时也表示后一任务的开始

## ADM 图例



## --项目进度计划编排

超前(Lead)与滞后(Lag) 关键路径法 时间压缩法 资源优化

## • 任务滞后 (lag)

A--lag=x--&gt;B 表示A完成后x天B开始

作用: 人员休整 &amp; 等待试用验证

## • 任务超前 (lead)

A--lead=x--&gt;B 表示A完成之前x天B开始

作用:

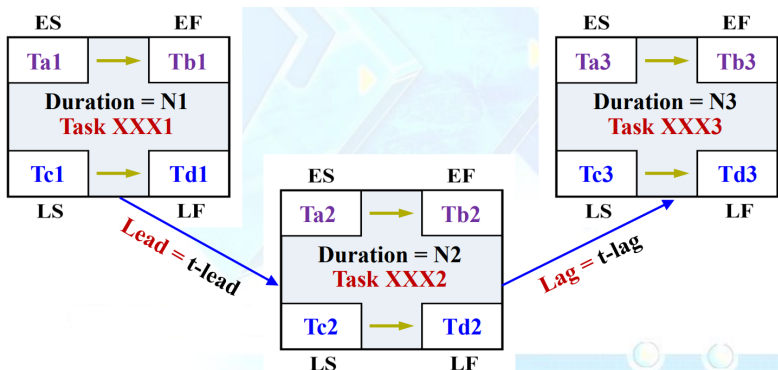
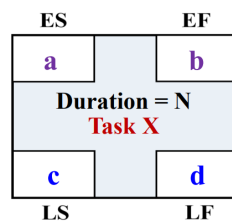
1. 解决任务的搭接
2. 对任务可以进行合理的拆分
3. 缩短项目工期

## 项目进度计划编排: 关键路径法(CPM, Critical Path Method)

**项目进度计划编排:** 针对项目开发中的所有活动进行合理的时间安排  
 其核心对象是**任务/活动**, 以及与其息息相关的任务间关系网络、任务历时估计、资源需求、相关责任人、其他假设条件等

## 关于“任务/活动”属性的基本概念

- 活动名称 (Task)、历时时间 (Duration)
- 最早开始时间 (Early Start, ES)
- 最晚开始时间 (Late Start, LS)
- 最早完成时间 (Early Finish, EF)
- 最晚完成时间 (Late Finish, LF)
- 总浮动时间 (Total Float, TF)
- 自由浮动时间 (Free Float, FF)
- 调整超前量 (lead)、调整滞后量 (lag)
- 前置活动 (Predecessor, p)、后置活动 (Successor, s)

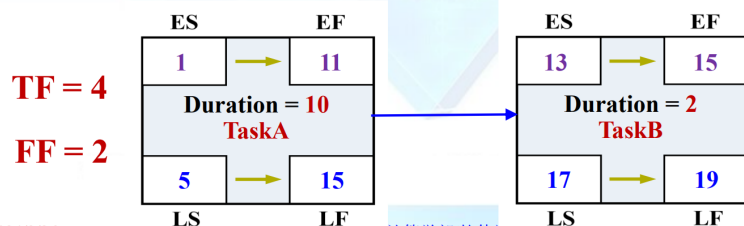


## 浮动时间(Float)

□ **浮动时间(Float)**: 是一个任务/活动的机动性时间, 它是一个任务/活动在不影响其它任务/活动或者项目完成的情况下可以延迟的时间量

□ **总浮动时间(Total Float, TF)**: 在不影响项目最早完成时间的前提下, 一个任务/活动可以延迟的时间

□ **自由浮动时间(Free Float, FF)**: 在不影响后置任务/活动最早开始时间的前提下, 一个任务/活动可以延迟的时间



## 关键路径 (Critical Path)

在一个项目中表达任务/活动关系的网络图 (PDM、ADM) 中:

- 网络图中时间最长的路径称为“关键路径”
- 关键路径是决定项目完成的最短时间
- 时间浮动为0 (Float=0) 的路径
- 关键路径上任何活动延迟, 都会导致整个项目完成时间的延迟
- 关键路径可能不止一条(长度一定一样!)



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.

对一个项目进度计划编排, 关键问题是要确定项目完成总时间长度

针对任务分解结果及活动网络图, 找出“关键路径”, 即抓住了进度管理的关键点

关键路径检索有2种方法: 正推法(Forward pass)和逆推法 (Backward pass)

### • 正推法

正推法(Forward pass): 按照时间顺序计算网络图中每个活动的最早开始时间ES和最早完成时间EF, 从而找到关键路径的方法

$$EF = ES + \text{Duration}$$

$$ES = EF(p) + \text{Lag} - \text{Lead} \quad (\text{Lag为前置节点p与本节点相关的滞后量, Lead为前置节点允许当前节点的超前量})$$

(当一个任务有多个前置任务时, 选择前置任务中最EF(p)+Lag-Lead结果最大值作为当前节点的ES)

根据EF最大值的节点为线索, 确定关键路径

### • 逆推法

逆推法(Forward pass): 按照时间顺序计算网络图中每个活动的最晚开始时间LS和最晚完成时间LF, 从而找到关键路径的方法

首先确定项目的结束时间，网络图中最后一个任务最晚完成时间是项目的结束时间，逐个对网络中每个任务节点，计算LS、LF

$$LS = LF - \text{Duration}$$

$$LF = LS(s) - \text{Lag} + \text{Lead} \quad (\text{Lag为后置节点s与本节点相关的滞后量, Lead为本节点 允许后置节点的超前量})$$

(当一个任务有多个后置任务时，选择后置任务中LS(s)-Lag+Lead结果最小值作为当前节点的LF)

遍历网络图，计算每个路径的所有任务的LS和LF。最后，确定关键路径，方法：**找到浮动时间Float为0的路径**