

第5章

项目进度估算



5.1 进度估算概述

一般来说,项目规模、成本和进度的估算基本上是同时进行的,项目的规模不仅决定成本,而且决定进度。规模越大,则成本越高、进度越慢。成本是从费用的角度对项目进行估算,而进度则是从时间的角度对项目进行估算。费用应理解为一个抽象概念,它可以是工时、材料或人员等。其实,成本和时间估算都是做计划的过程。在项目进行的过程中,会有更多新的变化,可能需要不断地调整估算。在项目的不同阶段会采用不同的估算方法,开始估算的结果可能误差比较大,随着项目的进行,会逐步地精确。

时间是一种特殊的资源,以其单向性、不可逆性、不可替代性、不可加性而有别于其他资源。如果项目的资金不够,则可以贷款或集资来借用别人的资金,但如果项目的时间不够,就无处可借。

项目经理应该定义所有的项目任务,识别出关键任务,跟踪关键任务的进展情况,同时及时发现拖延进度的情况。为此,项目经理必须制定一个足够详细的进度表,以便监督项目进度并控制整个项目。

进度计划是项目计划中最重要的部分之一。编制项目进度计划的主要过程如下:首先根据任务分解结构图进一步分解出主要的任务,确立任务之间的关联关系,然后估算出每个任务需要的资源、时间,最后编制出项目的进度计划。

交付期作为软件开发合同或者软件开发项目中的时间要素,是软件开发能否获得成功的重要判断标准之一。软件项目管理的主要目标是提升质量,降低成本,保证交付期,追求顾客满意。交付期意味着软件开发在时间上的限制,意味着软件开发的最终速度,也意味着满足交付期带来的预期收益和达到交付期需要付出的代价。目前,软件项目的进度是企业最重视的项目要素之一,原因有很多,例如,客户最关心的是进度,最明确的也是进度;进度是项目各要素中最容易度量的,在许多企业的领导人看来是最理想的管理考核指标。

为了编制进度计划,首先需要进行任务定义。任务分解结构图的每个工作包需要被划分成所需要的任务,每个被分配的任务都应该与一个工作包相关,通过任务定义这一过程可使项目目标体现出来。任务分解结构图是面向可提交物的,任务定义是面向任务的,是对任务分解结构图做进一步分解的结果,以便清楚应该完成的每个具体任务或者提交物应该执行的任务。

5.2 任务之间的关系

任务分解之后,接下来需要确定任务之间的关系。为了进一步制订切实可行的进度计划,必须对任务进行适当的顺序安排。它是通过分析所有的任务、项目范围、里程碑等信息来确定各个任务之间的关系。

5.2.1 任务之间的时间关系

项目各任务之间存在相互联系与相互依赖关系,根据这些关系安排各任务的先后顺序。任务排序过程包括确认并编制任务间的相关性。任务必须被正确地加以排序,以便今后制订现实的、可行的进度计划。排序可由 Project 软件执行或手工执行。对于小型项目,手工排序很方便;对于大型项目,在其早期(此时项目细节了解甚少)手工排序也是方便的。手工排序和计算机排序可以结合使用。任务之间的时间关系主要有 4 种情况,如图 5-1 所示。

其中:

结束→开始(Finish to Start,FS):表示 A 任务在 B 任务开始前结束。

开始→开始(Start to Start,SS):表示 A 任务开始,B 任务也可以开始,即 A、B 任务有相同的前置任务。

结束→结束(Finish to Finish,FF):表示 A 任务结束,B 任务也可以结束,即 A、B 任务有相同的后置任务。

开始→结束(Start to Finish,SF):表示 A 任务开始,B 任务应该结束。

结束→开始是最常见的关系,开始→结束关系(例如,A 任务是交接 B 任务的工作,这时就需要这种关系)极少使用。

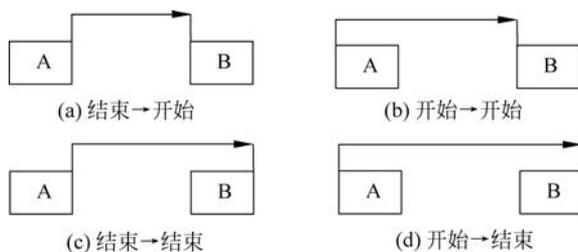


图 5-1 项目各任务之间的关系

5.2.2 任务之间的逻辑关系

任务之间的逻辑关系主要有以下几种。

1. 强制依赖关系

强制依赖关系是工作任务中固有的依赖关系,是一种不可违背的逻辑关系,又称硬逻辑关系或内在的相关性,它是由客观规律和物质条件的限制造成的。例如,需求分析一定要在软件设计之前完成,测试任务一定要在编码任务之后执行。

2. 软逻辑关系

软逻辑关系是由项目经理确定的项目任务之间的关系,是人为的、主观的,是一种根据主观意志去调整和确定的任务之间的关系。例如,安排计划的时候,哪个模块先做,哪个模块后做,哪个任务先做好一些,哪些任务同时做好一些,都可以由项目经理确定。

3. 外部依赖关系

外部依赖是项目任务与非项目任务之间的依赖关系,例如,环境测试依赖于外部提供的环境设备等。

5.3 进度安排的图示

软件项目进度安排的图示有很多,如甘特图、网络图、里程碑图、资源图等,下面分别进行介绍。

5.3.1 甘特图

甘特图(Gantt图)历史悠久,具有直观简明、容易学习、容易绘制等优点。甘特图可以显示任务的基本信息。使用甘特图能方便地查看任务的开始时间、结束时间、工期、资源信息。甘特图有两种表示方法,这两种方法都是将任务分解结构图中的任务排列在垂直轴,而时间作为水平轴。一种是棒状图,用于表示任务的起止时间,如图5-2所示。空心棒状图表示计划起止时间,实心棒状图表示实际起止时间。用棒状图表示任务进度时,一个任务需要占用两行的空间。另外一种表示甘特图的方式如图5-3所示,是用三角形表示特定日期,向上三角形表示开始时间,向下三角形表示结束时间,计划时间和实际时间分别用空心三角形和实心三角形表示。一个任务只需要占用一行的空间。

这两个图示说明同样的问题,从图5-2和图5-3中可以看出所有任务的起止时间都比计划推迟了,而且任务2的历时也比计划长很多。

利用甘特图可以很方便地制作项目计划和控制项目计划,由于其简单易用而且容易理解,因此被广泛地应用到项目管理中,尤其被软件项目管理所普遍使用。例如,图5-4是用Project软件生成的一个软件项目的甘特图。

从甘特图中可以很容易看出一个任务的开始时间和结束时间,但是甘特图的最大缺点是不能反映某一项任务的进度变化对整个项目的影 响,不能明显地表示各项任务彼此间的依赖关系,也不能明显地表示关键路径和关键任务。因此,在管理大型软件项目时,仅用甘特图是不够的,而网络图可以反映任务的起止日期变化对整个项目的影 响。

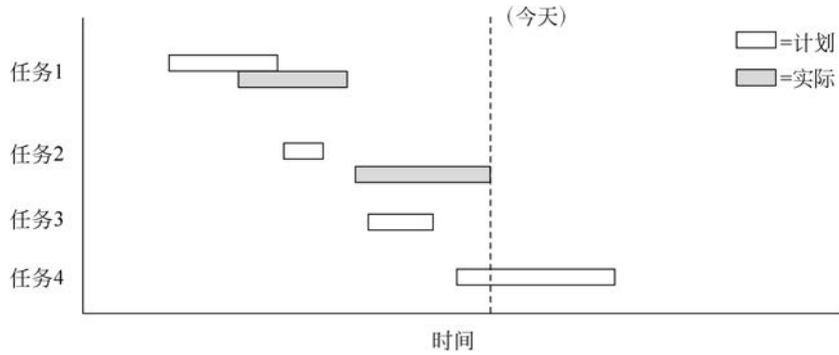


图 5-2 棒状甘特图

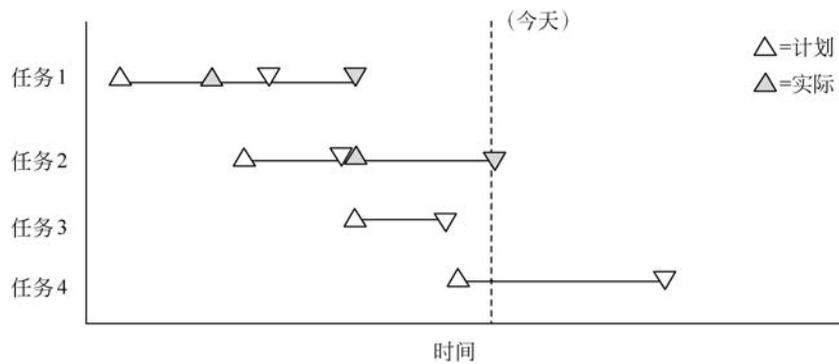


图 5-3 三角形甘特图

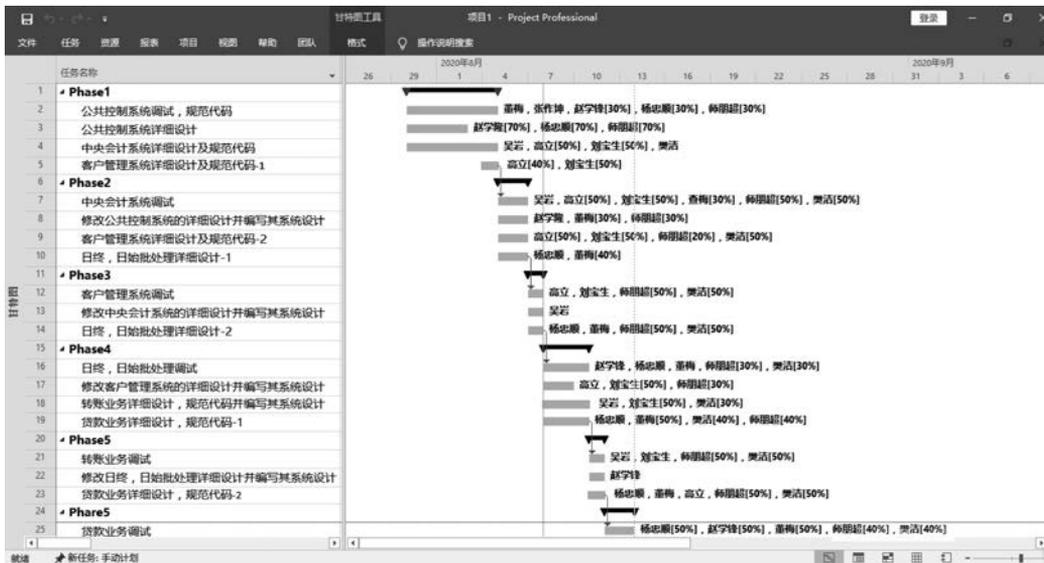


图 5-4 项目甘特图

5.3.2 网络图

网络图是任务排序的一个输出,用于展示项目中的各个任务及任务之间的逻辑关系,表明项目任务将以什么顺序进行。进行历时估计时,网络图可以表明项目将需要多长时间完成,当改变某项任务历时时,网络图表明项目历时将如何变化。

网络图不仅能描绘任务分解情况及每项作业的开始时间和结束时间,而且能清楚地表明各个作业彼此间的依赖关系。在网络图中容易识别出关键路径和关键任务。因此,网络图是制订进度计划的强有力的工具。通常联合使用甘特图和网络图这两种工具来制订和管理进度计划,使它们互相补充。

网络图是非常有效的进度表达方式。在网络图中可以将项目中的各个任务以及各个任务之间的逻辑关系表示出来,从左到右绘制出各个任务的时间关系图。网络图开始于一个任务、工作、活动或里程碑,结束于一个任务、工作、活动或里程碑,有些任务有前置任务或者后置任务。前置任务是在后置任务前进行的任务,后置任务是在前置任务后进行的任务,前置任务和后置任务表明项目中的任务将以什么顺序进行。常用的网络图有 PDM 网络图、ADM 网络图、CDM 网络图等。

1. PDM 网络图

PDM(Precedence Diagramming Method)网络图也称优先图,构成 PDM 网络图的基本特点是节点,节点表示任务,箭线表示各任务之间的逻辑关系。在图 5-5 所示的 PDM 网络图中,任务 1 是任务 3 的前置任务,任务 3 是任务 1 的后置任务。

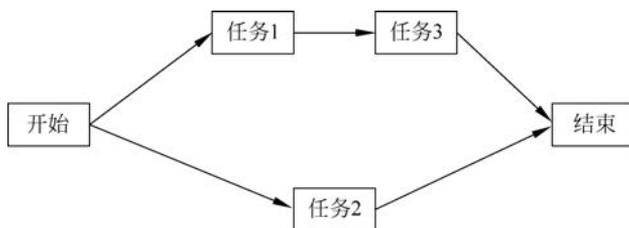


图 5-5 PDM 网络图的关系

PDM 网络图是目前比较流行的网络图,图 5-6 所示为一个软件项目的 PDM 网络图的实例。

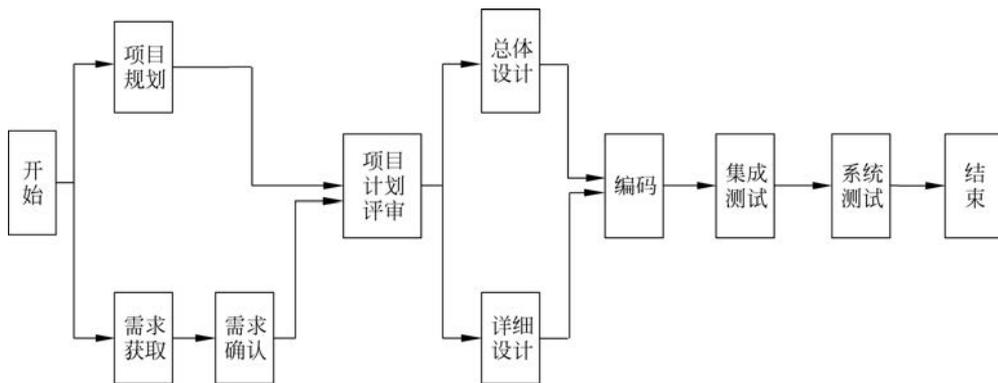


图 5-6 软件项目的 PDM 网络图

2. ADM 网络图

ADM(Arrow Diagramming Method)网络图也称为箭线法图或双代号图。在箭线法图中,箭线表示任务;节点表示前一个任务的结束,同时表示后一个任务的开始。将图 5-6 的项目改用 ADM 网络图表示,如图 5-7 所示。这里的双代号表示网络图中两个代号唯一确定一个任务,例如,代号 1 和代号 3 确定“项目规划”任务,代号 3 和代号 4 确定“项目计划评审”任务。

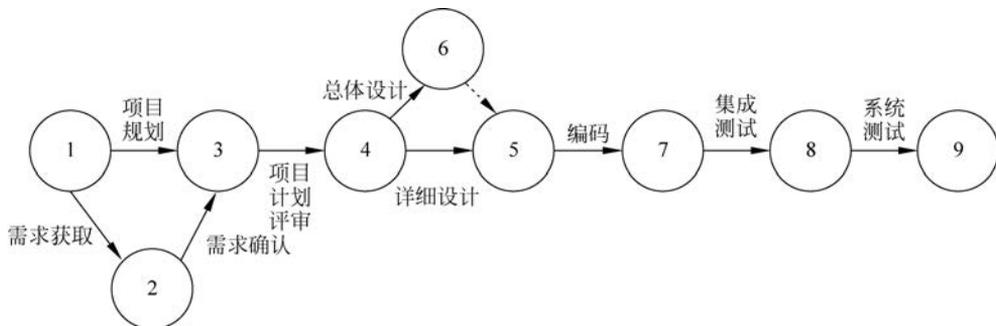


图 5-7 软件项目的 ADM 网络图

在 ADM 网络图中,有时为了表示逻辑关系,需要设置一个虚任务,虚任务不需要时间和资源,一般用虚箭线表示。图 5-8 中的任务 A 和任务 B 表示的 ADM 网络图是不正确的,因为在 ADM 网络图中,代号 1 和代号 2 只能确定一个任务(两个代号唯一确定一个任务),或者任务 A 或者任务 B。为了解决这个问题,需要引入虚任务(见图 5-9),为了表示任务 A、B 的逻辑关系需要引入代号 3,用虚线连接代号 2 到代号 3 的任务就是一个虚任务。它不是一个实际的任务,只是为了表达逻辑关系而引入的。图 5-7 中的代号 6 和代号 5 之间的虚线也代表一个虚任务。

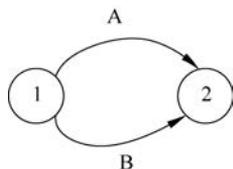


图 5-8 不正确的 ADM 图

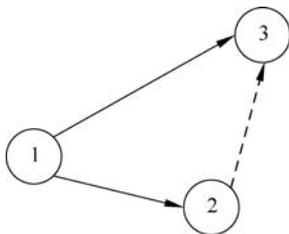


图 5-9 有虚任务的 ADM 图

3. CDM 网络图

CDM(Conditional Diagramming Method)网络图也称为条件箭线法图。它允许任务序列相互循环与反馈,如一个环(如某试验需重复多次)或条件分支(例如,一旦在检查中发现错误,就要修改设计),从而在绘制网络图的过程中会形成许多条件分支,而这在 PDM、ADM 中是不允许的。这种网络图在实际项目中使用很少。

5.3.3 里程碑图

里程碑图是由一系列的里程碑事件组成的,“里程碑事件”往往是一个时间要求为零的任务,即它并非一个要实实在在完成的任务,而是一个标志性的事件。例如,在软件开发项

目中的“测试”是一个子任务，“撰写测试报告”也是一个子任务，而“完成测试报告”可能就不能成为一个实实在在需要完成的子任务了，但在制订计划以及跟踪计划的时候，往往加上“完成测试报告”这一个子任务，其工期往往设置为“0 工作日”，目的在于检查这个时间点，这是“测试”整个任务结束的标志。



图 5-10 里程碑图

里程碑图显示项目进展中重大工作的完成情况。里程碑不同于任务，任务需要消耗资源并且需要花时间来完成，里程碑仅仅表示事件的标记，不消耗资源和时间。例如，图 5-10 是一个项目的里程碑图，从图中可以知道设计在 2013 年 4 月 10 日完成，测试在 2013 年 5 月 30 日完成。里程碑图表示了项目进展过程中几个重要的点，对项目干系人是非常重要的。

项目计划以里程碑为界限，将整个开发周期划分为若干阶段。根据里程碑的完成情况，适当地调整每一个较小阶段的任务量和完成的任务时间，这种方式非常有利于整个项目计划的动态调整。

对项目里程碑阶段点的设置必须符合实际，它必须有明确的内容并且通过努力能达到，要具有挑战性和可达性，只有这样才能在抵达里程碑时，使开发人员产生喜悦感和成就感，激发大家向下一个里程碑前进。实践表明，未达到项目里程碑的挫败感将严重地影响开发的效率，不能达到里程碑可能是里程碑的设置不切实际造成的。进度管理与控制其实就是确保项目里程碑的达成，因此里程碑的设置要尽量符合实际，并且不轻易改变里程碑的时间。

5.3.4 资源图

资源图可以用来显示项目进展过程中资源的分配情况，资源包括人力资源、设备资源等。图 5-11 就是一个人力资源随时间分布的资源图。

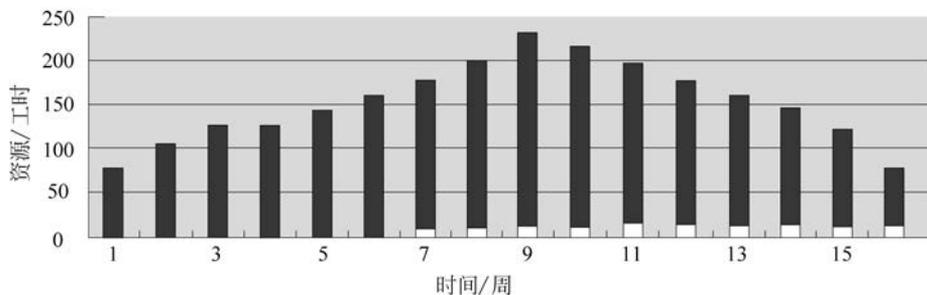


图 5-11 人力资源图

5.4 任务资源估计

在估计每个任务的历时之前，首先应该对每个任务需要的资源类型和数量有一定的考虑，这些资源包括人力资源、设备资源以及其他资源等。对于项目经理来说，应该回答下面

的问题。

(1) 对于特定的任务,它的难度如何?

(2) 是否有唯一的特性影响资源的分配?

(3) 企业以往类似项目的状况如何? 个人的成本如何?

(4) 企业现在是否有完成项目合适的资源(人、设备、资料等)? 企业的政策是否会影响这些资源的供给?

(5) 是否需要更多的资源来完成这个项目? 是否需要外包人员等?

为了准确估计任务需要的资源,项目的任务列表、任务属性、历史项目计划、企业的环境因素、企业的过程制度、可用资源状况等信息是必需的。

可以采用专家估算方法或者找有类似项目经验的人来辅助估算,也可以采用脑力风暴方法评估相关选项。由于人力资源是软件项目最主要的成本,因此在项目的早期,应该从不同渠道来获取相关的信息,当然这个结果也会进行修改和完善。

5.5 任务历时估计

定义了项目中的一系列任务和所有任务之间的先后关系,估计了需要的资源后,就需要估计任务的历时,即花费的时间。

任务历时估计就是估计任务的持续时间,它是项目计划的基础工作,直接影响整个项目所需的总时间。任务历时估计太长或太短对整个项目都是不利的。项目历时估计首先是对项目中的任务进行时间估计,然后确定项目的历时。任务时间估计指预计完成各任务所需时间长短,在项目团队中熟悉该任务特性的个人和小组,可对任务所需时间做出估计。

一般地,在历时估计的时候,应该考虑如下信息。

(1) 实际的工作时间:例如,一周工作几天,一天工作几个小时等。要充分考虑正常的工作时间,去掉节假日等。

(2) 项目的人员规模:一般规划项目时,应该按照人员完成时间来考虑,如多少人月、多少人天等,同时要考虑资源需求、资源质量和历史资料等。资源数量的多少也决定任务的历时估计,大多数任务所需时间由相关资源的数量所决定。例如,两人一起工作完成某设计任务只需一半的时间(相对一个人单独工作所需的时间),然而每日只能用半天进行工作的人通常至少需要两倍的时间完成某任务(相对一个人能整天工作所需的时间)。大多数任务所需时间与人和材料的能力(质量)有关。

(3) 生产率:根据人员的技能考虑完成项目的生产率,如 LOC/天等。

(4) 有效工作时间:在正常的工作时间内,去掉聊天、打电话、去卫生间、休息等时间后的有效工作时间。

(5) 连续工作时间:不被打断的持续工作时间。

(6) 人员级别:不同的人员,级别不同,生产率不同,成本也不同。对于同一任务,假设两个人均能全日进行工作,一个高级工程师所需时间少于初级工程师所需时间。资源质量也影响任务的估计,有关各类任务所需时间的历史资料是有用的。

(7) 历史项目:与这个项目有关的先前项目结果的记录,可以帮助进行时间估计。

在项目计划编制过程中,由于开发人员需要休息、吃饭、开会等,可能不会将所有的时间

放在项目开发工作上,而且这还不考虑开发人员的工作效率是否保持在一恒定水平上。其实,一天8小时工时制并不是花在项目上的时间就是8小时。在实际开发中,开发员工的时间利用率能够达到80%就已经很好了。

历时估计应该是有效工作时间加上额外时间(或者称为安全时间),历时估计的输出是各个任务的时间估计,即关于完成一个任务需多少时间的数量估计。下面介绍几种软件项目历时估计常用的估算方法。

5.5.1 定额估算法

定额估算法是比较基本的估算项目历时的方法,公式为

$$T = Q / (R \times S)$$

其中, T 为任务的持续时间; Q 为任务的规模(工作量); R 为人数; S 为效率(贡献率)。

定额估算法比较简单,而且容易计算。这种方法比较适合对单个任务的历时估算或者规模较小的项目,它有一定的局限性,没有考虑任务之间的关系。

例如,一个软件任务的规模估算是 $Q=6$ (人天),如果有2个开发人员,即 $R=2$ (人),而每个开发人员的效率是 $S=1$ (即正常情况下),则 $T=6/(2 \times 1)=3$ (天),即这个任务需要3天完成;如果 $S=1.5$,则 $T=6/(2 \times 1.5)=2$ (天),即这个任务需要2天完成。

5.5.2 经验导出模型

经验导出模型是根据大量项目数据统计分析而得出的模型。不同的项目数据导出的模型略有不同,整体的经验导出模型为

$$D = a \times E^b$$

其中, D 表示月进度; E 表示人月工作量; a 是2~4之间的参数; b 为1/3左右的参数。 a 、 b 是依赖于项目自然属性的参数。

例如,Walston-Felix模型的任务历时公式为 $D=2.4 \times E^{0.35}$,基本COCOMO模型的任务历时公式为 $D=2.5 \times E^b$ (b 是0.32~0.38之间的参数)。

【例 5-1】 一个33.3KLOC的软件开发项目,属于中等规模、半嵌入型的项目,试采用基本COCOMO模型估算项目历时。

根据公式 $D=2.5 \times E^b$ 以及已知条件(中等规模、半嵌入型),得 $b=0.35$,所以工作量为 $E=3.0 \times 33.3^{1.12} \approx 152$ (人月),历时估计为 $D=2.5 \times 152^{0.35} \approx 14.5$ (月)。

经验导出模型可以根据项目的具体情况选择合适的参数。例如,一个项目的规模估计是 $E=65$ 人月,如果模型中的参数 $a=3$, $b=1/3$,则 $D=3 \times 65^{1/3} \approx 12$ (月),即65人月的软件规模估计需要12个月完成。

5.5.3 工程评估评审技术

工程评估评审技术(Program Evaluation and Review Technique,PERT)最初发展于1958年,用来适应大型工程年代的需要。当估计历时存在不确定性时,可以采用PERT方

法,即估计具有一定的风险时采用这种方法。PERT 方法采用加权平均的算法进行历时估算。

$$\text{PERT 历时} = (O + 4M + P) / 6$$

其中: O 是任务完成的最小估算值,是最乐观值(optimistic time), P 是任务完成的最大估算值,是最悲观值(pessimistic time); M 是任务完成的最大可能估算值(most likely time)。

最乐观值是基于最好情况的估计,最悲观值是基于最差情况的估计,最大可能估算值是基于最大可能情况的估计或者基于最期望情况的估计。

在图 5-12 的网络图中,估计 A、B、C 任务的历时存在很大不确定性,故采用 PERT 方法估计任务历时,图 5-12 中标示了 A、B、C 任务的最乐观、最可能和最悲观的历时估计,根据 PERT 历时公式,计算各个任务的历时估计结果,见表 5-1。

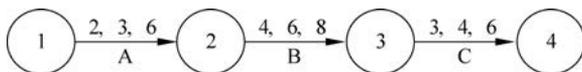


图 5-12 ADM 网络图

一条路径上的所有任务的历时估计之和便是这条路径的历时估计,其值称为路径长度。图 5-12 中的路径长度为 13.5,即这个项目总的的时间估计是 13.5,见表 5-1。

表 5-1 PERT 方法估计项目历时

任务	最乐观值	最可能值	最悲观值	PERT 估计值
A	2	3	6	3.33
B	4	6	8	6
C	3	4	6	4.17
项目历时				13.5

用 PERT 方法估计历时存在一定的风险,因此有必要进一步给出风险分析结果。为此引入了标准差和方差。

$$\text{标准差: } \delta = \frac{P - O}{6}$$

$$\text{方差: } \delta^2 = \left(\frac{P - O}{6} \right)^2$$

其中: O 是最乐观的估计; P 是最悲观的估计。

标准差和方差可以表示历时估计的可信度或者项目完成的概率。

我们需要估计网络图中一条路径的历时情况时,如果这条路径中每个任务的 PERT 历时估计分别为 E_1, E_2, \dots, E_n , 标准差分别为 $\delta_1, \delta_2, \dots, \delta_n$, 则这条路径的历时、方差、标准差分别为

$$E = E_1 + E_2 + \dots + E_n$$

$$\delta^2 = \delta_1^2 + \delta_2^2 + \dots + \delta_n^2$$

$$\delta = \sqrt{\delta_1^2 + \delta_2^2 + \dots + \delta_n^2}$$

图 5-12 中 A、B、C 任务的标准差和方差以及这条路径的标准差和方差见表 5-2。

表 5-2 任务与项目的标准差和方差

任务与项目	标准差	方差
A 任务	4/6	16/36
B 任务	4/6	16/36
C 任务	3/6	9/36
项目路径	1.07	41/36

根据概率理论,对于遵循正态概率分布的均值 E 而言, $E \pm \delta$ 的概率分布是 68.3%, $E \pm 2\delta$ 的概率分布是 95.5%, $E \pm 3\delta$ 的概率分布是 99.7%。

图 5-12 所示项目的 PERT 总历时估计是 13.5 天,标准差 $\delta=1.07$ 。这个项目总历时估计的概率见表 5-3。项目在 12.43 天到 14.57 天内完成的概率是 68.3%,项目在 11.36 天到 15.64 天内完成的概率是 95.5%,项目在 10.29 天到 16.71 天内完成的概率是 99.7%。

表 5-3 项目完成的概率分布

历时估计 $E=13.5, \delta=1.07$			
范围	概率	从	到
$E \pm \delta$	68.3%	12.43	14.57
$E \pm 2\delta$	95.5%	11.36	15.64
$E \pm 3\delta$	99.7%	10.29	16.71

【例 5-2】 图 5-12 所示项目在 14.57 天内完成的概率是多少?

由于 $14.57=13.5+1.07=E+\delta$,因此项目在 14.57 天内完成的概率是箭头 1 以左的概率(见图 5-13),很显然它等于箭头 2 以左的概率加上 68.3%/2,即 84.2%,所以项目在 14.57 天内完成的概率是 84.2%,接近于 85%。

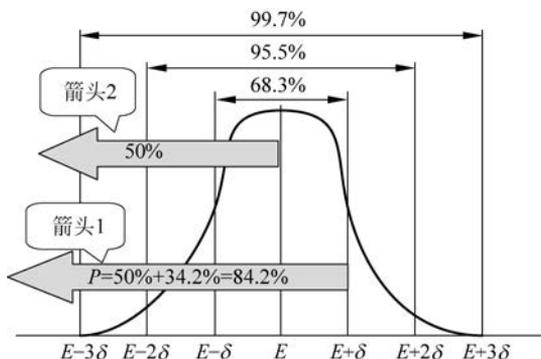


图 5-13 项目在 14.57 天内完成的概率

5.5.4 专家估计方法

估计项目所需时间经常是困难的,因为许多因素(如资源质量的高低、劳动生产率的不同)会影响项目所需时间。专家估计方法是通过专家依靠过去资料信息进行判断,以估算进

度的方法。如果找不到合适的专家,估计结果往往不可靠且具有较大风险。

5.5.5 类推估计方法

类推估计意味着利用一个先前类似任务的实际时间作为估计未来任务时间的基础,在项目早期,掌握项目信息不多的时候常使用这种方法。类推估计是专家判断的一种形式,以下情况的类推估计是可靠的:①先前任务和当前任务在本质上类似,而不仅仅是表面相似;②专家能力具备。对于软件项目,利用企业的历史数据进行历时估计是常见的方法。

5.5.6 模拟估计方法

模拟是用不同的假设条件试验一些情形,以便计算相应的时间。最常见的模拟估计方法是蒙特卡罗分析技术(Monte Carlo analysis)。在这种方法中,通过假设各任务所用时间的概率分布来计算整个项目完成所需时间的概率分布。让计算机多次进行一个项目的模拟,就可以得出一个可能结果的范围和每一个结果的概率。

5.5.7 基于承诺的进度估计方法

基于承诺的进度估计方法是从需求出发去安排进度,不进行中间的工作量(规模)估计,而是通过开发人员做出的进度承诺进行的进度估计,它本质上不是进度估算。其优点是有利于开发者对进度的关注,有利于开发者在接受承诺之后鼓舞士气;其缺点是开发人员对进度的估计存在一定的误差。

5.5.8 Jones 的一阶估计准则

Jones 的一阶估计准则是根据项目功能点的总和,从幂次表(见表 5-4)中选择合适的幂次将它升幂。例如,如果一个软件项目的功能点是 $FP=350$,承担这个项目的公司是平均水平的商业软件公司,则粗略的进度估算是 $350^{0.43} \approx 12$ (月)。

表 5-4 一阶幂次表

软件类型	最优级	平均	最差级
系统软件	0.43	0.45	0.48
商业软件	0.41	0.43	0.46
封装商品软件	0.39	0.42	0.45

5.6 进度计划编排方法

进度计划编排是决定项目任务的开始和结束日期的过程,若开始日期和结束日期是不现实的,项目则不可能按计划完成。编排进度计划时,如果资源分配没有被确定,决定项目

任务的开始日期和结束日期仍是初步的,资源分配可行性的确认应在项目计划编制完成前做好。其实,编制计划的时候,成本估计、时间估计、进度编制等过程常常交织在一起,这些过程反复多次,最后才能确定项目进度计划。

进度计划编排的输入有项目网络图、任务历时估计、资源需求、资源库描述(对于进度编制而言,有什么资源,在什么时候,以何种方法可供利用是必须知道的)、日历表、超前与滞后、约束和假设(例如,强制性日期、关键事件或里程碑事件,项目资助者、项目客户或其他项目相关人提出在某一特定日期前完成某些工作细目,一旦定下来,这些日期就很难被更改)等。

进度估算和进度编排常常是结合在一起进行的,采用的方法也是一致的。一般来说,项目进度编排的方法主要有关键路径法、时间压缩法等。

5.6.1 关键路径法

关键路径法(Critical Path Method,CPM)是根据指定的网络图逻辑关系进行的单一的历时估算。首先计算每一个任务的单一的、最早和最晚开始日期和完成日期,然后计算网络图中的最长路径,以便确定项目的完成时间估计。采用此方法可以配合进行进度的编制。关键路径法的关键是计算总时差,这样可决定哪一个任务有最小时间弹性,可以为更好地进行项目计划编制提供依据。CPM算法也在其他类型的数学分析中得到应用。

一个项目往往是由若干个相对独立的任务链条组成的,各链条之间的协作配合直接关系到整个项目的进度。

讲述关键路径进度编排方法前,先来了解一下有关进度编制的基本术语。

(1) 最早开始时间(Early Start,ES)。表示一项任务最早可以开始执行的时间。

(2) 最晚开始时间(Late Start,LS)。表示一项任务最晚可以开始执行的时间。

(3) 最早完成时间(Early Finish,EF)。表示一项任务最早可以完成的时间。

(4) 最晚完成时间(Late Finish,LF)。表示一项任务最晚可以完成的时间。

(5) 超前时间(lead)。表示两个任务的逻辑关系所允许的提前后置任务的时间,它是网络图中任务间的固定可提前时间。

(6) 滞后时间(lag)。表示两个任务的逻辑关系所允许的推迟后置任务的时间,是网络图中任务间的固定等待时间。举一个简单的例子:装修房子的时候,需要粉刷房子,刷油漆的后续任务是刷涂料,它们之间需要至少一段时间(一般是一天)的等待时间,等油漆变干后,再刷涂料,这个等待时间就是滞后时间。

(7) 浮动时间(float)。浮动时间体现了一个任务的机动性,它是一个任务在不影响项目完成的情况下可以延迟的时间量。

① 总浮动时间(Total Float,TF)。在不影响项目最早完成时间的前提下,本任务可以延迟的时间。 $TF=LS-ES$ 或者 $TF=LF-EF$ 。

② 自由浮动时间(Free Float,FF)。在不影响后置任务最早开始时间的前提下,本任务可以延迟的时间。 $FF=ES(s)-EF-lag$ [$ES(s)$ 是后置任务的最早开始时间, lag 是本任务与后置任务之间的滞后时间],即某任务的自由浮动时间等于它后置任务的ES减去它的EF,再减去它的lag。自由浮动时间是对总浮动时间的描述,表明总浮动时间的自由度。

(8) 关键路径。项目是由各个任务构成的,每个任务都有一个最早、最迟的开始时间和结束时间,如果一个任务的最早时间和最迟时间相同,则表示其为关键任务,一系列不同任务链条上的关键任务链接成为项目的关键路径。关键路径是整个项目的主要矛盾,是确保项目能否按时完成的关键。关键路径在网络图中的浮动时间为0,而且是网络图中的最长路径。关键路径上的任何任务延迟都会导致整个项目完成时间的延迟,它表明了完成项目的最短时间量。

下面以图 5-14 为例来进一步说明以上基本术语的含义(假设所有任务的历时以天为单位)。

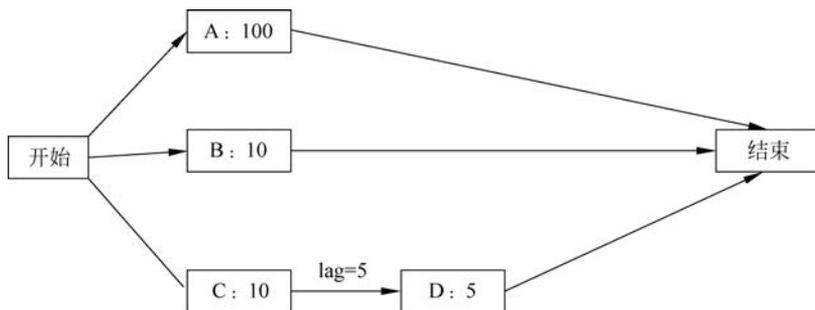


图 5-14 项目网络图

在图 5-14 中,A、B、C 是并行的关系,则项目的完成时间是 100。任务 A 的最早开始时间和最晚开始时间都为 0,最早结束时间和最晚结束时间都为 100,所以 $ES(A)=0$, $EF(A)=100$, $LF(A)=100$, $LS(A)=0$ 。

任务 B 的历时为 10 天,可以有一定的浮动时间,只要在任务 A 完成之前完成任务 B 就可以了,所以,任务 B 的最早开始时间是 0,最早结束时间是 10;而最晚结束时间是 100,最晚开始时间是 90,可知任务 B 有 90 天的浮动时间,这个浮动时间是总浮动时间。任务 B 的总浮动时间 $=90-0=100-10=90$ 。所以 $ES(B)=0$, $EF(B)=10$, $LF(B)=100$, $LS(B)=90$, $TF(B)=LS(B)-ES(B)=LF(B)-EF(B)=90$ 。

任务 C 是任务 D 的前置任务,任务 D 是任务 C 的后置任务,它们之间的 $lag=5$ 表示任务 C 完成后的 5 天开始执行任务 D。任务 C 的历时是 10 天,任务 D 的历时是 5 天,所以任务 C 和任务 D 的最早开始时间分别是 0 和 15,最早结束时间分别是 10 和 20。如果要保证任务 D 的最早开始时间不受影响,则任务 C 是不能自由浮动的,所以任务 C 的自由浮动时间为 0。任务 D 的最晚结束时间是 100,则任务 D 的最晚开始时间是 95,这样,任务 C 的最晚结束时间是 90,任务 C 的最晚开始时间是 80,所以 $ES(C)=0$, $EF(C)=10$, $ES(D)=15$, $EF(D)=20$, $LF(D)=100$, $LS(D)=95$, $LF(C)=90$, $LS(C)=80$, $TF(C)=LS(C)-ES(C)=LF(C)-EF(C)=80$, $FF(C)=ES(D)-EF(C)-lag=0$ 。

从图 5-14 看,路径 A 的浮动时间为 0 且是网络图中的最长路径,所以它是关键路径,表明了完成项目的最短时间。

【例 5-3】 项目的 ADM 网络图如图 5-15 所示,如何确定其中的关键路径(所有任务的历时以天为单位)?

(1) 从图 5-15 可以知道有两条路径: $A \rightarrow B \rightarrow C \rightarrow E$ 和 $A \rightarrow B \rightarrow D \rightarrow F$ 。

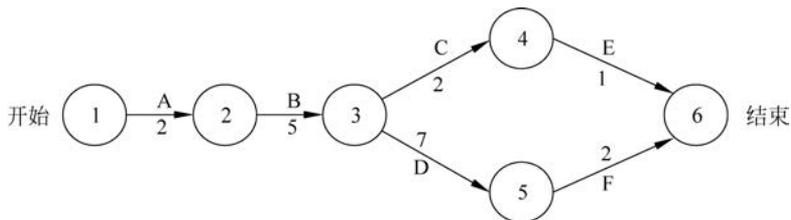


图 5-15 项目的 ADM 网络图

(2) $A \rightarrow B \rightarrow C \rightarrow E$ 的长度是 10, 有浮动时间; $A \rightarrow B \rightarrow D \rightarrow F$ 的长度是 16, 没有浮动时间。

(3) 最长而且没有浮动时间的路径 $A \rightarrow B \rightarrow D \rightarrow F$ 便是关键路径。

(4) 项目完成的最短时间是 16 天, 即关键路径的长度是 16 天。

图 5-16 代表网络图中的一个任务, 其中, 图中标识出任务的名称、任务的工期, 同时可以标识出任务的最早开始时间(ES)、最早完成时间(EF)、最晚开始时间(LS)以及最晚完成时间(LF)。项目路径中各个任务的最早开始时间、最早完成时间、最晚开始时间、最晚完成时间可以采用正推法和逆推法来确定。



图 5-16 任务图示

1. 正推法

在网络图中按照时间顺序计算各个任务的最早开始时间和最早完成时间的方法称为正推法。此方法的执行过程如下。

(1) 确定项目的开始时间。

(2) 从左到右, 从上到下进行任务编排。

(3) 计算每个任务的最早开始时间(ES)和最早完成时间(EF)。

① 网络图中第一个任务的最早开始时间是项目的开始时间。

② $ES + \text{Duration} = EF$: 任务的最早完成时间等于它的最早开始时间与任务的历时之和, 其中 Duration 是任务的历时。

③ $EF + \text{lag} = ES(s)$: 任务的最早完成时间加上(它与后置任务的)lag 等于后置任务的最早开始时间, 其中 $ES(s)$ 是后置任务的最早开始时间。

④ 当一个任务有多个前置任务时, 选择前置任务中最大的 EF 加上 lag(或者减去 lead) 作为其 ES。

图 5-17 所示网络图(假设所有任务的历时以天为单位)中项目的开始时间是 1, 如任务 A, 它的最早开始时间 $ES(A) = 1$, 任务历时 $\text{Duration} = 7$, 则任务 A 的最早完成时间是 $EF(A) = 1 + 7 = 8$ 。同理可以计算 $EF(B) = 1 + 3 = 4$; 任务 C 的最早开始时间 $ES(C) = EF(A) + 0 = 8$, 最早完成时间是 $EF(C) = 8 + 6 = 14$; 任务 G 的最早开始时间 $ES(G) = 14$, 最早完成时间 $EF(G) = 14 + 3 = 17$ 。同理 $ES(D) = 4$, $EF(D) = 7$; $ES(F) = 4$, $EF(F) = 6$ 。由于任务 E 有两个前置任务, 选择其中最大的最早完成时间(因为 $\text{lag} = 0$, $\text{lead} = 0$)作为其后置任务的最早开始时间, 所以 $ES(E) = 7$, $EF(E) = 10$; 任务 H 也有两个前置任务(任务 E 和任务 G), 选择其中最大的最早完成时间 17(因为 $\text{lag} = 0$, $\text{lead} = 0$), 作为任务 H 的最早开

始时间 $ES(H)=17, EF(H)=19$ 。这样,通过正推法确定了网络图中各个任务的最早开始时间和最早完成时间。

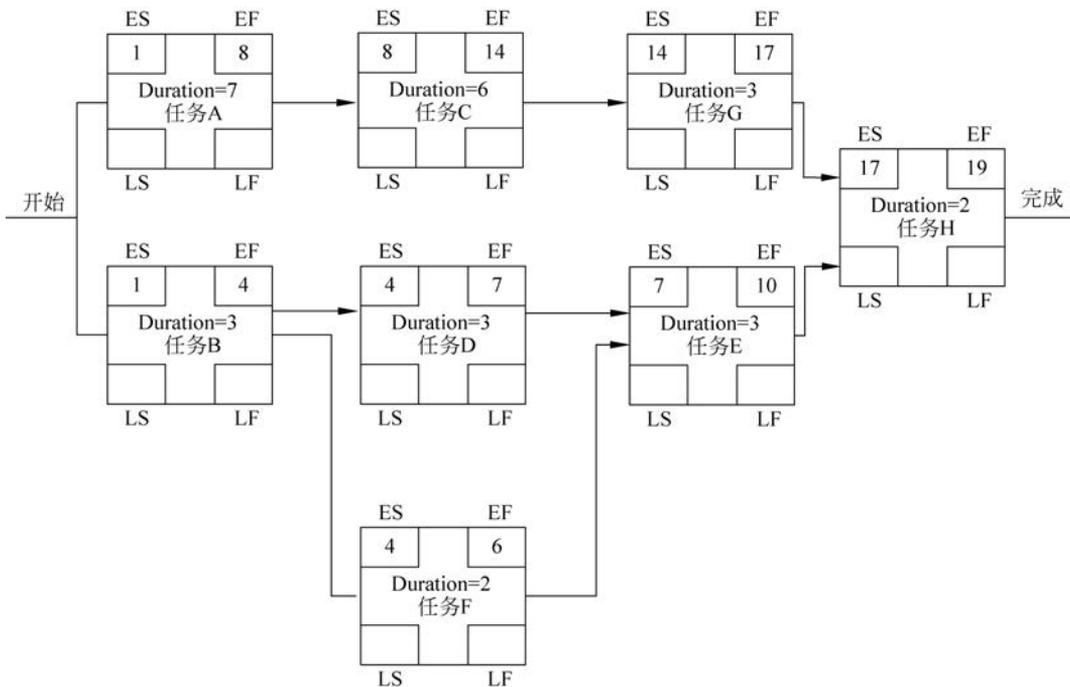


图 5-17 正推法示例

2. 逆推法

在网络图中按照逆时间顺序计算各个任务的最晚开始时间和最晚完成时间的方法,称为逆推法。此方法的执行过程如下。

(1) 确定项目的结束时间。

(2) 从右到左,从上到下进行任务编排。

(3) 计算每个任务的最晚开始时间(LS)和最晚完成时间(LF)。

① 网络图中最后一个任务的最晚完成时间是项目的结束时间。

② $LF - \text{Duration} = LS$: 一个任务的最晚开始时间等于它的最晚完成时间与历时之差。

③ $LS - \text{Lag} = LF(p)$: 一个任务的最晚开始与(它与前置任务的)lag之差等于它的前置任务的最晚完成时间 $LF(p)$,其中 $LF(p)$ 是其前置任务的最晚完成时间。

④ 当一个任务有多个后置任务时,选择其后置任务中最小LS减去lag(或者加上lead)作为其LF。

下面确定图 5-17 中各个任务的最晚开始时间和最晚完成时间。由于项目的结束时间是网络图中最后一个任务的最晚完成时间,对于图 5-17 所示网络图,这个项目的结束时间是 19,即 $LF(H)=19$,则 $LS(H)=19-2=17, LF(E)=17, LS(E)=17-3=14$ 。同理,任务 G、C、A、D、F 的最晚完成时间和最晚开始时间分别如下: $LF(G)=17, LS(G)=17-3=14$; $LF(C)=14, LS(C)=14-6=8$; $LF(A)=8, LS(A)=8-7=1$; $LF(D)=14, LS(D)=14-3=11$; $LF(F)=14, LS(F)=14-2=12$ 。任务 B 有两个后置任务,选择其中最小最晚

开始时间(因为 $lag=0, lead=0$)作为其最晚完成时间,所以将 11 作为任务 B 的最晚完成时间,即 $LF(B)=11, LS(B)=11-3=8$ 。另外,对于任务 F,它的自由浮动时间是 1($FF(F)=7-6$),而它的总浮动时间是 8($TF(F)=12-4=8$)。结果如图 5-18 所示,图中 A→C→G→H 的浮动时间为 0,而且是最长的路径,所以它是关键路径。关键路径长度是 19,所以项目的完成时间是 19 天,并且 A、C、G、H 都是关键任务。

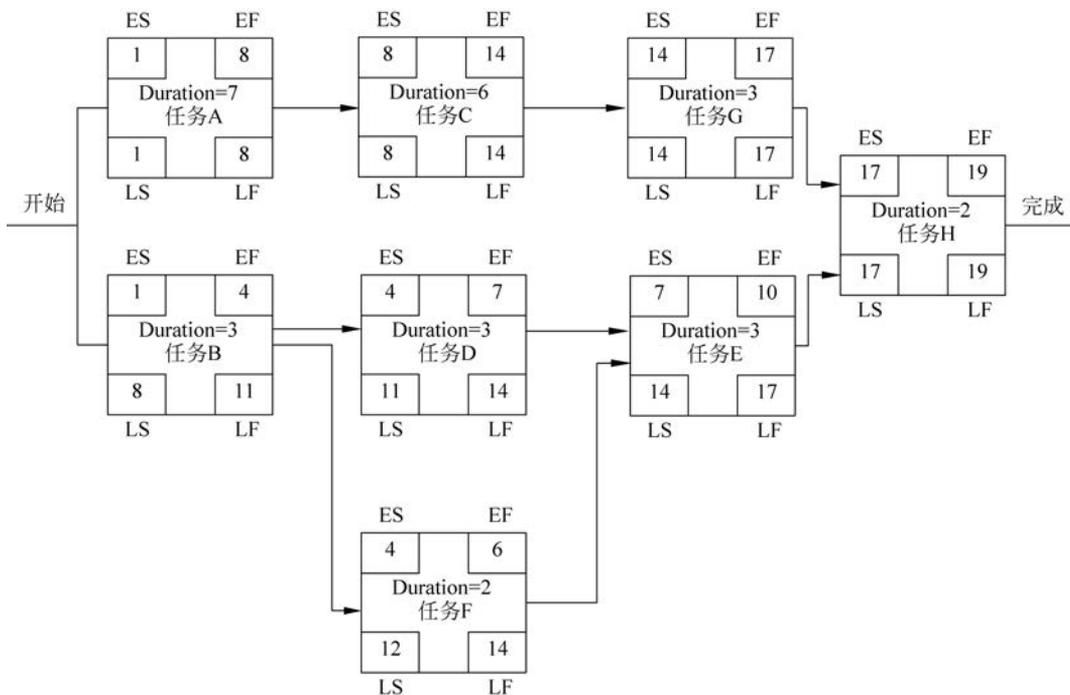


图 5-18 逆推法示例

图 5-18 所示网络图可以称为 CPM 网络图,如果采用 PERT 进行历时估计,则可以称为 PERT 网络图。PERT 网络与 CPM 网络是 20 世纪 50 年代末发展起来的两项重要的技术,其主要区别是 PERT 计算历时时存在一定的不确定性,采用的算法是加权平均 $(O+4M+P)/6$,CPM 计算历时的意见比较统一,采用的算法是最大可能值 M 。1956 年,美国杜邦公司首先在化学工业上使用了 CPM(关键路径法)进行计划编排,美国海军在建立北极星导弹时,采用了 Buzz Allen 提出的 PERT(计划评审法)技术。此后,这两种方法逐渐渗透到许多领域,为越来越多的人所采用,成为网络计划技术的主流。网络计划技术作为现代管理的方法,与传统的计划管理方法相比较具有明显优点,主要表现为以下几个方面。

(1) 利用网络图模型,明确表达各项工作的逻辑关系。按照网络计划方法,在制订项目计划时,首先必须清楚该项目内的全部工作和它们之间的相互关系,然后才能绘制网络图模型。

(2) 通过网络图时间参数计算,能够确定关键工作和关键线路。

(3) 掌握机动时间,进行资源合理分配。

(4) 运用计算机辅助手段,方便网络计划的调整与控制。

我国从 20 世纪 60 年代中期开始,在著名数学家华罗庚教授的倡导和亲自指导下,开始

试点应用网络计划,并根据“统筹兼顾,全面安排”的指导思想,将这种方法命名为“统筹方法”。网络计划技术从此在国内生产建设中卓有成效地推广开来。

为确保网络图的完整和安排的合理,可以进行如下检查。

- (1) 是否正确标识了关键路径?
- (2) 是否有哪个任务存在很大的浮动? 如果有,则需要重新规划。
- (3) 是否有不合理的空闲时间?
- (4) 关键路径上有什么风险?
- (5) 浮动有多大?
- (6) 哪些任务有哪种类型的浮动?
- (7) 工作可以在期望的时间内完成吗?
- (8) 提交物可以在规定的时间内完成吗?

关键路径法是理论上计算所有任务各自的最早和最晚开始时间与结束时间,但计算时并没有考虑资源限制,这样算出的时间可能并不是实际进度,而是表示所需的时间长短,在编排实际的进度时,应该考虑资源限制和其他约束条件,把任务安排在上述时间区间内,所以还需要如时间压缩、资源平衡等方法。

5.6.2 时间压缩法

时间压缩法是一种数学分析的方法,是在不改变项目范围前提下(例如,满足规定的时间或其他计划目标),寻找缩短项目时间途径的方法。应急法和平行作业法都是时间压缩法。

1. 应急法

应急法也称赶工,用于权衡成本和进度间的得失关系,以决定如何用最小增量成本达到大量的时间压缩。应急法并不总是产生一个可行的方案,且常常导致成本增加。

如果项目的工作方法和工具得当,就可以简单地通过增加人员和加班时间来缩短进度,进行进度压缩。在进行进度压缩时存在一定的进度压缩和费用增长的关系,很多人提出不同的方法来估算进度压缩与费用增长的关系,下面介绍其中两种方法。

1) 时间成本平衡方法

时间成本平衡方法是基于下面的假设提出的:

- (1) 每个任务存在一个正常进度和可压缩进度、一个正常成本和可压缩成本。
- (2) 通过增加资源,每个任务的历时可以从正常进度压缩到可压缩进度。
- (3) 每个任务无法在低于可压缩进度内完成。
- (4) 有足够需要的资源可以利用。

(5) 在“正常”与“可压缩”之间,进度压缩与成本的增长是成正比的,单位进度压缩成本 = (可压缩成本 - 正常成本) / (正常进度 - 可压缩进度)。

上述的线性关系方法是假设如果任务在可压缩进度内,进度压缩与成本的增长成正比,所以可通过计算任务的单位进度压缩成本来计算在压缩范围之内的进度压缩产生的压缩费用。

【例 5-4】 图 5-19 是一个项目的 PDM 网络图,在可压缩的范围内,假设 A、B、C、D 任

任务的进度压缩与成本增长呈线性正比关系。表 5-5 分别给出了 A、B、C、D 任务的正常进度、可压缩进度、正常成本、可压缩成本。从 PDM 网络图可知,目前项目的总工期为 18 周,如果将工期分别压缩到 17 周、16 周、15 周并且保证每个任务在可压缩的范围内,试分析应该压缩哪些任务,并计算压缩之后的总成本。

表 5-5 正常进度、可压缩进度、正常成本、可压缩成本

进度、成本	A	B	C	D
正常进度	7 周	9 周	10 周	8 周
正常成本	5 万元	8 万元	4 万元	3 万元
可压缩进度	5 周	6 周	9 周	6 周
可压缩成本	6.2 万元	11 万元	4.5 万元	4.2 万元

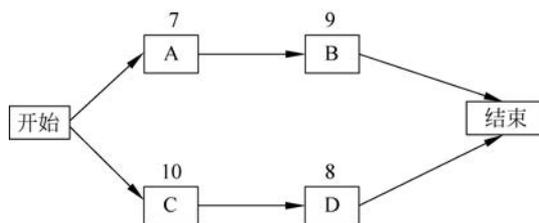


图 5-19 项目的 PDM 网络图

(1) 从 PDM 网络图可以看到,有“开始→A→B→结束”和“开始→C→D→结束”两个路径,前者的长度是 16 周,后者的长度是 18 周,所以“开始→C→D→结束”是关键路径,即项目完成的最短时间是 18 周。

(2) 如果将工期分别压缩到 17 周、16 周、15 周并且保证每个任务在可压缩的范围内,必须满足以下两个前提。

- ① A、B、C、D 任务必须在可压缩的范围内。
- ② 保证压缩之后的成本最小。

根据表 5-5 计算 A、B、C、D 任务单位进度压缩的成本,如表 5-6 所示。

表 5-6 每个任务的单位进度压缩成本

任务	A	B	C	D
单位进度压缩成本/(万元/周)	0.6	1	0.5	0.6

根据上述两个条件,首先看可以压缩哪些任务,然后选择压缩后成本增加最小的任务,如表 5-7 所示。

表 5-7 压缩后的项目成本

完成周期/周	压缩的任务	成本计算/万元	项目成本/万元
18		$5+8+4+3$	20
17	C	$20+0.5$	20.5
16	D	$20.5+0.6$	21.1
15	A、D	$21.1+0.6+0.6$	22.3

(3) 如果希望总工期压缩到 17 周,需要压缩关键路径“开始→C→D→结束”,可以压缩的任务有 C 和 D,根据表 5-6 知道压缩任务 C 的成本最小(压缩任务 C 一周增加 0.5 万元成本,压缩任务 D 一周增加 0.6 万元成本),故选择压缩任务 C 一周,压缩到 17 周后的总成本是 20.5 万元。

(4) 如果希望总工期压缩到 16 周,需要压缩关键路径“开始→C→D→结束”,可以压缩的任务还是 C 和 D,任务 C 压缩一周后,在可压缩范围内是不能再压缩的,否则压缩成本会非常高,这时应该选择压缩任务 D 一周,项目压缩到 16 周后的总成本是 21.1 万元。这时,项目网络图的两条路径的长度都是 16 周,即有两条关键路径。

(5) 如果希望总工期压缩到 15 周,应该压缩两条关键路径,即“开始→A→B→结束”和“开始→C→D→结束”两条路径都需要压缩。在 A、B 任务中应该选择压缩任务 A 一周(压缩任务 A 一周增加 0.6 万元成本,压缩任务 B 一周增加 1 万元成本),在 C、D 中选择压缩 D 一周(这样的压缩成本是最低的),所以,项目压缩到 15 周后的总成本是 22.3 万元。

2) 进度压缩因子方法

进度压缩与费用的上涨不是总能呈现正比的关系,当进度被压缩到“正常”范围之外,工作量就会急剧增加,费用会迅速上涨。而且,软件项目存在一个可能的最短进度,这个最短进度是不能突破的。在某些时候,增加更多的软件开发人员会减慢开发速度,而不是加快开发速度。例如,一个人 5 天写 1000 行程序,5 个人 1 天不一定能写 1000 行程序,40 个人 1 个小时不一定能写 1000 行程序。增加人员会存在更多的交流和管理的的时间。在软件项目中,不管怎样努力工作,无论怎么寻求创造性的解决办法,无论组织团队多大,都不能突破这个最短的进度点。

进度压缩因子方法是由 Charles Symons 提出来的,而且被认为是精确度比较高的一种方法。它的公式如下:

$$\text{进度压缩因子} = \text{期望进度} / \text{估算进度}$$

$$\text{压缩进度的工作量} = \text{估算工作量} / \text{进度压缩因子}$$

这个方法是首先估算初始的工作量和初始的进度,然后将估算与期望的进度相结合,计算进度压缩因子,以及压缩进度的工作量。例如,项目的初始估算进度是 12 个月,初始估算工作量 78 人月。如果期望压缩到 10 个月,则进度压缩因子 $=10/12 \approx 0.83$,压缩进度后的工作量 $=78/0.83 \approx 94$ (人月),即压缩进度增加的工作量是 16 人月。也就是说,进度缩短 17%,约增加 21%的工作量。

很多研究表明,进度压缩因子不应该小于 0.75,这说明一个任务最多压缩 25%是有意义的。

2. 平行作业法

平行作业法也称为快速跟进,是平行地做任务,这些任务通常要按前后顺序进行(例如,在设计完成前,就开始软件程序的编写)。例如,如图 5-20 所示项目在正常情况下,15 天内完成需求、设计。但是,如果需要在第 12 天内完成设计,则需要对项目的历时进行压缩。压缩方法有两种,一种是应急法,不改变任务之间的逻辑关系,将需求压缩到 8 天,设计压缩到 4 天,这样需求、设计可以在 12 天内完成。也可以采用另外一种方法,即调整任务需求和设计之间的逻辑关系,在需求没有完成前 3 天就开始设计,相当于需求任务与设计任务并行工作一段时间,或者说需求与设计任务之间的 lead=3,它解决任务的搭接,这样就压缩了项目的时间。但是,平行作业常导致返工和增加风险。

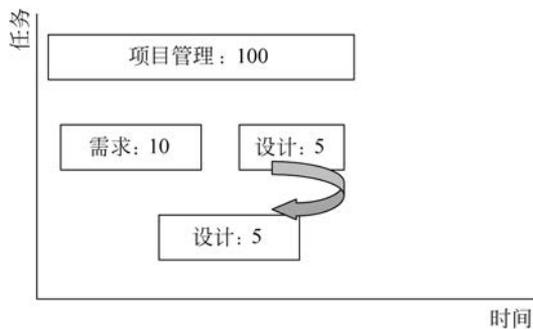


图 5-20 任务之间的快速跟进

5.6.3 资源平衡方法

为了成功地编制了一个项目进度计划,必须对项目中的任务分配资源,项目中的任务必须在一定的条件下人为操纵完成。要使用资源来完成项目中的任务,就必须将资源与任务联系起来。每项任务需要的资源包括人力资源、设备资源等。

资源平衡方法通过调整任务的时间来协调资源的冲突。这个方法的主要目的是形成平稳连续的资源需求,最有效地利用资源,使资源闲置的时间最小化,同时尽量避免超出资源能力。

关键路径法通常可以产生一个初始进度计划,而实施这个计划需要的资源可能比实际拥有的多。资源平衡法可在资源有约束条件下制订一个进度计划。例如,网络图 5-21 中有 A、B、C 3 个任务,A 需要 2 天 2 个开发人员完成,B 需要 5 天 4 个开发人员完成,C 需要 3 天 2 个开发人员完成。如果 3 个任务同时开始执行,如图 5-22 所示,一共需要 8 个开发人员,而资源高峰在项目开始的前 2 天,之后就会陆续有人出现空闲状态,资源利用率不合理。

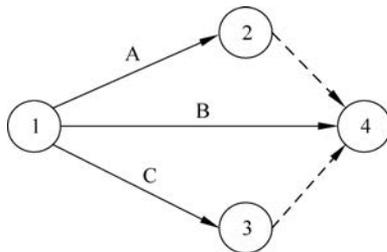


图 5-21 项目网络图

如果 C 任务利用浮动时间,使用最晚开始时间,即 A 任务完成之后再开始 C 任务,如图 5-23 所示,从项目开始到结束,一共需要 6 个开发人员,而且项目同样是 5 天内全部完成,但是资源利用率提高了,这就是一个资源平衡的例子。

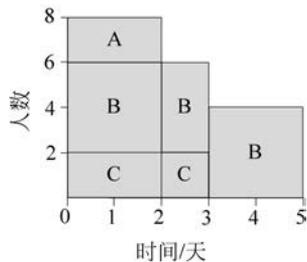


图 5-22 3 个任务同时开始的人员情况

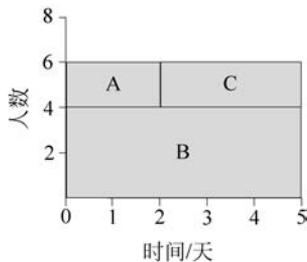


图 5-23 3 个任务不同时开始的人员情况

5.6.4 管理预留

管理预留是一项加在项目末端的人为任务,不是加在每一个任务间隔上,而是给项目增加一个储备时间。储备时间一般是项目中完成所有任务所需时间的 10%~15%。当一项任务超出了分配的时间,超出的部分可以使用关键路径末端的管理预留。

增加管理预留后,项目经理可以用百分比来查看项目的进展情况。例如,一个项目完成了 40%,却用去 65%的管理预留,如果余下的任务仍然保持这个趋势,那么这个项目就会陷入困境。

帕金森定律(Parkinson's Law)指出,工作总是拖延到它能够允许最迟完成的那一天。也就是说,如果工作允许拖延、推迟完成,往往这个工作总是推迟到它能够最迟完成的那一刻,很少有提前完成的。如果一项任务需要花费 10 小时完成,可能任务执行者自己知道只需要 6 小时就可以完成,但不可思议地花费了 10 小时。这样他们会不珍惜时间,可能会找一些其他事情来做或者简单地等待,直到预留时间花完,才开始正常地工作以期待将项目成功完成。

在任务的估算过程中,任务执行者常会受到夸大任务完成所估算时间的诱惑,不要受到这种诱惑的影响,应该总是反映任务完成所需要的准确时间。可以使用 PERT 方法中的最乐观值、最悲观值和最可能值来估算,但不要因为考虑了消极的因素(如出现错误、返工和任务延迟等)而增加每一项任务所需要的时间。所以,管理预留是将每一项任务的预留时间累加在一起,放在关键路径末端,而不要增加每一项任务时间。

5.6.5 Scrum 敏捷计划

Scrum 敏捷计划模型具有两层项目计划,基于远粗近细的原则和项目渐进明细的特点,通过将概要的项目整体规划和详细的近期迭代计划有机结合,帮助团队有效提高计划的准确度、资源管理能力和项目的按时交付能力。近期计划比较细化,长远计划比较粗糙。这两层计划是通过产品待办事项列表和 Sprint 待办事项列表体现的。

产品待办事项列表存在于产品的整个生命周期,它是产品的路线图。任何时候,产品待办事项列表都是团队依照优先排列顺序完成工作的唯一、最终的概括。一个产品只有一个产品待办事项列表,这意味着产品负责人必须纵观全局后做出优先级排列的决策,以体现利益相关人(包括团队)的意愿。表 5-8 就是一个产品待办事项列表,它相当于比较粗糙的远期计划。

表 5-8 产品待办事项列表

每个 Sprint 的新估算									
优先级	事 项	细节 (wiki 链接)	初始规模估算	1	2	3	4	5	6
1	作为买家,我想把书放入购物车 (见 wiki 页面用户界面草图)		5						
2	作为买家,我想从购物车中删除书		2						
3	提高交易处理性能(见 wiki 页面目标性能指标)		13						

续表

产品待办事项 列表事项	Sprint 中的任务	志愿者	初始工作 量估计	1	2	3	4	5	6
事务处理效率	合并 DCP 代码并完成分段测试		5						
	完成 pRank 的机器顺序		8						
	把 DCP 和读入需改为用 pRank http API		13						

习题

一、填空题

- _____决定了项目在给定的金钱关系和资源条件下完成项目所需的最短时间。
- _____是一种特殊的资源,以其单向性、不可重复性、不可替代性而有别于其他资源。
- 在 ADM 网络图中,箭线表示_____。
- _____和_____都是时间压缩法。
- 任务(活动)之间的排序依据主要有_____、_____、_____等。
- 工程评估评审技术采用加权平均的公式是_____,其中 O 是乐观值, P 是悲观值, M 是最可能值。

二、判断题

- 一个工作包可以通过多个活动完成。()
- 在项目进行过程中,关键路径是不变的。()
- 在 PDM 网络图中,箭线表示的是任务之间的逻辑关系,节点表示的是活动。()
- 项目各项活动之间不存在相互联系与相互依赖关系。()
- 在资源冲突问题中,过度分配也属于资源冲突。()
- 浮动是在不增加项目成本的条件下,一个活动可以延迟的时间量。()
- 在使用应急法压缩时间时,不一定要在关键路径上选择活动来进行压缩。()
- 时间是项目规划中灵活性最小的因素。()
- 外部依赖关系又称强制性依赖关系,指的是项目活动与非项目活动之间的依赖关系。()
- 当估算某活动时间,存在很大不确定性时应采用 CPM 估计。()

三、选择题

- 下面公式中不正确的是()。

A. $EF=ES+duration$	B. $LS=LF-duration$
C. $TF=LS-ES=LF-EF$	D. $EF=ES+lag$
- “软件编码完成之后,我才可以对它进行软件测试”,这句话说明了哪种依赖关系?()

A. 强制性依赖关系	B. 软逻辑关系	C. 外部依赖关系	D. 里程碑
------------	----------	-----------	--------

3. ()可以显示任务的基本信息,使用该类图能方便地查看任务的工期、开始时间、结束时间以及资源的信息。

- A. 甘特图 B. 网络图 C. 里程碑图 D. 资源图

4. ()是项目冲突的主要原因,尤其在项目后期。

- A. 优先级问题 B. 人力问题 C. 进度问题 D. 费用问题

5. 项目计划中灵活性最小的因素是()。

- A. 时间 B. 人工成本 C. 管理 D. 开发

6. ()不是编制进度的基本方法。

- A. 关键路径法 B. 时间压缩法 C. 系统图法 D. 资源平衡方法

7. 快速跟进是指()。

- A. 采用并行执行任务,加速项目进展 B. 用一个任务取代另外的任务
C. 如有可能,减少任务数量 D. 减轻项目风险

8. ()将延长项目的进度。

- A. lag B. lead C. 赶工 D. 快速跟进

9. ()可以决定进度的灵活性。

- A. PERT B. 总浮动 C. ADM D. 赶工

四、简答题

1. 对一个任务进行进度估算时,A是乐观者,估计用6天完成,B是悲观者,估计用24天完成,C是有经验者,认为最有可能用12天完成,那么这个任务的历时估算介于10天到16天的概率是多少?

2. 请将图5-24所示的PDM(优先图法)网络图改画为ADM(箭线法)网络图。

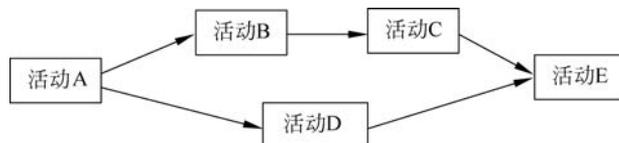


图 5-24 PDM 网络图