

COCOMOII模型 -- 软件成本估算

1、COCOMOII 模型介绍

1981 年，Barry Boehm 在其经典著作《软件工程经济学》中，介绍了一种软件成本估算模型的层次体系，称为 COCOMO(Constructive Cost Model, 构造型成本模型)。这是 Barry Boehm 研究了加利福尼亚的 TRW 咨询公司的大量项目数据后，推导出的一个成本模型，他代表了软件成本估算综合经验的一个模型，为估算软件项目的成本和进度提供了一个良好定义的开放性的基础。1981 年提出的原始 COCOMO 模型，即 COCOMO81，在实际应用中取得了较好的效果，很好的匹配了采用瀑布模型的软件项目。

近二十年来，软件工程领域取得了巨大的进步，新的软件过程和软件生命周期模型层出不穷，面对由螺旋或进化开放模型创建软件以及通过商业产品应用组装能力开放软件的项目，COCOMO81 遇到了越来越到的困难。为了适应软件生命周期、技术、组件、工具、表示法及项目管理技术的进步，原始 COCOMO 的作者 Barry Boehm 对 COCOMO 做了调整和改进，提出了一个新的版本 - COCOMOII。

2、COCOMOII 模型分类

COCOMOII 模型中使用三个螺旋式的过程模型：应用组装模型，早期设计模型和后体系结构模型。

应用组装模型 (Application Composition)

应用组装模型是基于对象点的度量模型，它通过计算屏幕、报表、第三代语言 (3GL) 模块等对象点的数量来确定基本的规模，每个对象点都有权重，由一个三级的复杂性因子表示，将各个对象点的权值累加起来得到一个总体规模，然后再针对复用进行调整。

早期设计模型 (early design)

在项目开始后的一个阶段或者螺旋周期通常包括探索体系结构的可供选择方案或增量开放测量。为支持这一活动，COCOMOII 提出了一个早期设计模型，这一模型使用功能点和等价代码行估算规模。

后体系结构模型 (post architecture)

一旦项目进入开放阶段，就必然确定一个具体的生命周期体系结构，此时项目就能够为估算提供更多更准确的信息。

3、COCOMOII 模型估算方法

在利用 COCOMOII 模型进行软件成本估算过程中，首先采用软件规模估算方法对项目的规模进行估算。再应用五个比例因子，通过相关计算，将规模转化为工作量，并通过十七个成本驱动因子对工作量进行调整。最后，采用进度计算公式，计算出开发该项目所需要的进度以及人数。

3.1、软件规模估算方法

a、代码行分析法

软件产品的源代码的行数进行测量，是直接度量的方法，与使用的开发程序语言相依附。

b、德尔菲法

也被称为专家判断法，是依赖一个或多个专家的经验来做估算，其判断的信息包括经验法则，可用的资源，过去项目的开发数据，过去估算的反馈及过去类似项目的详细功能等。

c、类比分析法

由类似于已完成项目的实际成本为基础来估算新项目，估算新系统的每个主要组件就如同现存系统相似组件规模的百分比，将每个组件的估算规模加起来就是新系统的总规模。

d、价格胜算法

以争取项目合约为原则，因此以足够取得项目合约价格为基础所做的规模估算。

e、功能点分析法

是基于系统的一种规模估算方法。它是基于应用软件的外部、内部特性以及软件性能的一种间接的规模测量方法。通过软件产品进行评估，并根据明确定义的软件系统的功能特性来测量软件的规模。

3.2、工作量估算方法

后体系结构模型和早期设计模型采用相同的函数形式去估算软件项目开发所花费的工作量。以人月 (Person Month, PM) 表示的工作量值 PM_{NS} 。

$$PM_{NS} = A \times Size^E \times \prod_{i=1}^n EM_i$$

其中，Size 表示估算的规模，单位是源代码千行数 (KSLOC)。EM_i 为工作量乘数，EM_i的个数 n 的值对于后体系结构模型是 17，对早期设计模型是 7。指数 E 的计算公式如下。

$$E = B + 0.01 \times \sum_{j=1}^5 SF_j$$

参数 A，B，EM_i，SF_j的值是根据数据库中历史项目的实际参数和工作量的值进行校准而获得的。A 可校准，A=2.94，B 可校准，B=0.91。

SF 代表指数比例因子：

先例性 (PREC)：表示以前是否开发过类似项目。

开发灵活性 (FLEX)：表示软件性能与已经建立的需求和外部接口规范的一致程度。

体系结构 / 风险化解 (RESL): 通过风险管理衡量项目的风险及建立体系结构的工作量。

团队凝聚力 (RERM): 衡量项目相关人员的管理状况。

过程成熟度 (PMAT): 衡量项目过程的规范程度，主要围绕 SEI 的 CMM 而进行。

Driver	Very low	Low	Nominal	High	Very high	Extra high
PREC	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	5.07	4.05	3.04	2.03	1.01	0.00
RESL	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	7.80	6.24	4.68	3.12	1.56	0.00

工作量乘数 (Effort Multiplier, EM)。

1) 产品因子：用于说明你由正在开发产品的特征英气开发软件所需工作量的变化，COCOMOII 共有 5 个产品因子，其中复杂性对估算的工作量影响最大。

要求的软件可靠性 (RELY)、数据库规模 (DATA)、产品复杂性 (CPLX)、可服用开发 (RUSE)、匹配生命周期需求的稳定编制 (DOCU)

2) 平台因子：用于描述平台的一些约束。平台是包括复杂硬件和基础软件的目标机。

执行时间约束 (TIME)、主存储约束 (STOR)、平台易变性 (PVOL)。

3) 人员因子：除了产品规模之外，人员因子在确定开发软件产品所需的工作量中有最强的影响。人员因子是按照开发团队的能力和进行分级的，而不是针对个人。

分析员能力 (ACAP)、程序员能力 (PCAP)、人员联系性 (PCON)、应用经验 (APEX)、平台经验 (PLEX)、语言和工具经验 (LTEX)。

4) 项目因子：说明诸如现代软件工具的使用、开发组的地理位置和项目进度压缩等因素对工作量估算的影响。

软件工具的使用 (TOOL)、多点开发 (SITE)、要求的开发进度 (SCED)。

早期设计模型主要用在项目早期阶段中，由于此时对待开发产品的规模、目标平台性质、项目相关人员状态以及采用过程的想象说明等还不够了解，所以将后体系结构模型中的 17 个成本驱动因子综合成 7 个成本驱动因子。

早期设计模型成本驱动因子	后体系结构模型成本驱动因子
产品可靠性与复杂性 PCPX	RELY、DATA、CPLX、DOCU
可复用性开发 RUSE	RUSE
平台难度 PDIF	TIME、STOR、PVOL
人员能力 PERS	ACAP、PCAP、PCON
人员经验 PREX	APEX、PLEX、LTEX
设施 FCIL	TOOL、SITE
要求的开发进度 SCED	SCED

4、进度估算

在进行工作估算后，可利用工作量进行进度计算及人员调配。计算公式：

$$TDEV = [C \times (PM_{NS})^{(D+0.2 \times (E-B))} \times \frac{SCED}{100}]$$

其中，B，C，D 为可校准的常数，B 的值为 0.91，C 的值为 3.67，D 的值为 0.28，PMNS 为不考虑要求的开发进度工作量乘数 (SCED) 时的工作量，E 为根据项目比例影子之和所导出的工作量规模指数，

SCED% 为要求的开发进度工作量乘数等级量表中的开发进度压缩 / 扩展百分比。

5、本案例进度估算

开发团队正在开发一款与之前开发的不太相似的应用程序（PREC 是低的，分数为4.96），不太严格的软件工程文档要求（FLEX 是中等的，分数为 3.04），并且这些要求不太可能改变（RESL 是高的，分数为 2.83）。团队非常团结（TEAM 是高的，分数为 2.19），但是流程是非正式的（PMAT 是非常低的 分数为 7.80）。

$$\begin{aligned} SF &= 0.91 + 0.01 \times (4.96 + 3.04 + 2.83 + 2.19 + 7.80) = 0.91 + \\ &0.01 \times 20.82 \\ &= 0.91 + 0.2082 \\ &= 1.1182 \end{aligned}$$

如果系统包括 10kloc,

$$PM = 2.94 \times 10^{1.1182} = 38.5964$$

成本驱动因子如下：

PCPX (产品的可靠性和复杂度)	very high	1.91
PDIF (平台难度)	very high	1.81
PERS (人员的能力)	extra high	0.50
PREX (人员的经验)	nominal	1.00

总成本如下：

$$\begin{aligned} PM &= 38.5964 \times 1.91 \times 1.81 \times 0.50 = 66.7158 \\ TDEV &= 3.67 \times (66.7158)^{(0.28+0.2 \times (1.1182-0.91))} = 14.1713 \end{aligned}$$