

# 第 29 章 大型、复杂项目和多项目管理

大型、复杂项目管理和多项目管理，都属于“组织级”的项目管理。按照 PMI 的定义，组织级项目管理是一种包括项目管理、大型项目管理和项目组合管理的系统管理体系。组织级项目管理是近年来项目管理领域最活跃的研究方向之一，其知识体系还在发展和完善当中。

根据《信息系统项目管理师考试大纲》，本章重点要求考生掌握以下几方面的知识：

- 大型、复杂项目和多项目管理的特征和分解。
- 大型、复杂项目和多项目的计划过程。
- 跟踪和控制管理。
- 范围管理。
- 资源管理。
- 协作管理。

## 29.1 大型、复杂项目管理的特征和分解

什么样的项目才能算是大型、复杂项目？从投资者的角度看，项目的投资额可以作为衡量的标准。从项目组织的角度看，项目团队的规模与参与单位的多少也应作为判别的依据。各种划分方式都有一定的片面性，界线也不太严格，我们不可能也没必要去研究“大型、复杂项目”的精确定义，但我们必须意识到大型、复杂项目管理的复杂性。

有专家从项目管理的角度给出了大型、复杂项目的定义，认为只有采用某种复杂管理模式的项目才能称为大型、复杂项目。我们认为，这是一种本末倒置的定义。实际上，一个项目的规模和复杂程度是客观存在的，正是由于项目本身的复杂性才导致了项目管理的复杂性，而不能说项目管理的复杂性导致了项目的复杂性。

### 29.1.1 大型、复杂项目管理的特征

一般说来，大型、复杂项目的管理具有以下两个特征：

(1) **分级管理与分工管理**。大型、复杂项目规模较大，目标构成复杂，项目经理很难直接管理到项目团队的每一个成员和项目的每一项目标，一般需要建立一个管理团

队，实行分级管理和分工管理。

大型、复杂项目的项目经理的职责更集中于管理职责，管理所体现的效益更直接地影响项目目标的实现。同时，大型、复杂项目多实行分级管理制，项目经理一般采用间接管理的方式。

**(2) 强化协调机制。**大型、复杂项目团队构成复杂。例如，“中国多民族文字电子出版系统”涉及到6个省、市、自治区、8个单位、10个民族的语言文字专家和计算机专家。这种复杂的团队构成会使团队之间的协作、沟通所需要的成本大幅度上升，如何建立有效的协调机制就成了整个项目成败的关键。

### 29.1.2 大型、复杂项目的分解

大型、复杂项目通常会分解成许多相对独立而又相互联系的中小型项目，一般称为子项目，每个子项目可以独立地开发、测试，子项目之间多为并行关系，但也可能存在串行关系。子项目有时还要进一步分解为“课题”，而“课题”还有可能再次分解为“子课题”。

项目所分解的层数称为项目分解的“深度”。如果深度过大，则应考虑简化分解层次。

同一层上的项目单元（子项目、课题、子课题）的最大值称为本层的“宽度”。宽度大意味着本层的复杂度高，需要控制和协调过多的下层项目单元，应该适当增加中间层。

但我们也应当注意，不应为了单纯追求深度和宽度的理想化而不顾项目的实际情况。

大型、复杂项目分解的总原则是，各个子项目的复杂程度之和应小于整个项目的复杂程度。项目分解时既要考虑到技术性因素，也要考虑到非技术性因素。

**(1) 项目分解的技术性因素。**软件设计中“高内聚”、“低耦合”的模块划分原则同样适用于大型、复杂项目的子项目划分。在这里，内聚是指一个子项目内各个目标之间彼此结合的紧密程度，耦合则是指不同子项目之间相互关联的程度。内聚与耦合是密切相关的，子项目内的高内聚往往意味着子项目间的低耦合。“高内聚”、“低耦合”的子项目划分，使项目中的子项目之间联系简单，发生在某一子项目的错误传播到整个项目的可能性就很小，研究、测试或维护任何一个子项目不需要对项目的其他子项目有很多了解。

**(2) 项目分解的非技术性因素。**大型、复杂项目往往是多方投资、多方参与、多方受益。项目分解时，还应考虑到资金来源、知识产权和利益分配等非技术性因素。

例如，“中国多民族文字电子出版系统”涉及蒙古文、藏文、维吾尔文、哈萨克文、柯尔克孜文、朝鲜文、彝文、傣文和傣文，可以看做一个大型、复杂项目。但因该项目涉及的文种、地区与合作单位太多，地方配套资金很难解决（任何地方都不会出钱支持与本地区无关的文种），项目的知识产权和利益分配问题也很复杂。

如果按照文种将这个大项目分解为“蒙古文电子出版系统”、“藏文电子出版系统”、“维吾尔、哈萨克、柯尔克孜文电子出版系统”、“朝鲜文电子出版系统”、“彝文电子出版系统”、“傣文电子出版系统”和“傣僳文电子出版系统”等一系列的子项目，虽然从技术角度看并非最佳分解法，但这样分解之后的子项目可以较容易地得到地方资金支持，知识产权和利益分配问题也可以简单地解决。

当然，并不是所有的多文种处理系统都可按文种进行分解。比如，“中国多民族文字身份证数据库系统”就不能按文种分解为子项目，因为该项目需要建立、管理的是一个能够支持多文种的、全国统一的身份证数据库。

### 29.1.3 大型、复杂项目管理的分解

根据项目的规模与具体特点，大型、复杂项目的管理可按照以下三种方式进行分解：

**(1) 按照子项目分解。**这是一种最常见也最容易理解的管理分解方式，各个子项目的负责人（子项目经理）直接承担子项目的管理工作，并向项目经理负责，项目经理则承担整个项目的规划、组织、指导，以及各个子项目之间的协调工作，并考核评定各个子项目经理的业绩。项目管理分解与项目分解的一致性有助于明确每一个管理者的职责与权限，子项目负责人既是子项目组的行政负责人，也是子项目的业务管理者，拥有子项目组内的人力资源管理、技术管理、质量管理、进度管理和资金管理权，对子项目的成败负完全责任。只要子项目的划分足够合理，子项目经理选择得当，项目经理就没有必要直接插手子项目组的内部管理，而应当专注于子项目的监控、子项目组之间的协调，以及项目组与外部组织的协调。但这种管理模式也有缺点，当子项目组的规模过大时容易出现管理失控现象，而当子项目组的数量过多时又增加了协调难度。

**(2) 按照管理职能分解。**这是一种“职能办公室”式的管理分解方式。项目经理领导着一个职业管理团队，这个团队中的每个成员（或小组）负责某一方面的管理工作，如质量管理、进度管理、资金管理、资源管理和文档管理等。由于有了这样一批专业助手，项目经理对整个项目的控制权得到增强，一般不会出现管理失控现象。然而，有一利必有一弊，这种管理方式的缺点也是十分明显的，很容易造成项目执行与项目管理的脱节，行政管理与职能管理的脱节，不同的职能管理之间的脱节。例如，当质量、进度与资金产生矛盾时，相关的管理人员常常会各执一词，令项目执行人员无所适从，大量的协调工作最终还是要落在项目经理身上。

**(3) 矩阵式分解。**这种管理分解形式实际上是将按照子项目分解和按照管理职能分解的两种形式综合起来，这样既有利于强化各个子项目经理的责任和权利，又有利于项目经理对整个项目的监督和控制，并且有利于管理工作的统一化、专业化和规范化。矩阵式管理成败的要点是如何清晰地划分子项目经理与职能管理人员（或部门）的管理责任与权限。对于子项目经理，既要大胆放权，又要通过职能管理方式加强监控。对于职能管理人员，一方面应要求他们严格监管，另一方面也应提醒他们不要干预子项目组的内部管理。总的来说，子项目经理侧重于子项目组的内部管理和主动性控制，职能管

理人员则应侧重于外部控制与检查监督。矩阵式管理模式非常适合于大型、复杂项目，但同时也加大了管理成本。

## 29.2 大型、复杂项目的计划过程

一般项目的计划过程内容主要包括：

- (1) **范围计划**。确定项目的产品和项目范围，回答项目“做什么”。
- (2) **质量计划**。确定项目实现其质量目标的方法。
- (3) **进度计划**。确定项目活动在时间上的分配。
- (4) **成本计划**。确定项目活动所需要的资源分配。

可以看出，一般项目的计划主要关注的是项目活动的计划。但是对大型、复杂项目来说，制订活动计划之前，必须先考虑项目的过程计划，也就是必须先确定用什么方法和过程来完成项目。需不需要为一个项目单独建立一整套过程体系？

软件工程方法学包含 3 个要素：方法、工具和过程。方法是指完成软件开发的各项任务的技术方法；工具是指为运用方法而提供的软件工程支撑环境；过程则是指为获得高质量的软件所需要完成的一系列活动和任务的框架。其中，“活动”是一个过程的组成元素。“任务”则是构成活动的基本元素，由若干个任务构成一项活动。

过程管理作为最近 20 年来在 IT 行业中最活跃、最有成效的研究领域，一直伴随着来自各个方面的认可和疑问。过程的根本目标和作用就在于：当我们遵循一个预定义的过程时，具有较高的可能性来实现预定的目标和结果。显然，这里有一个前提假设，就是我们所遵循的过程是已知且正确的。

国家质量监督检验检疫总局于 2001 年 11 月 2 日发布的国家标准《GB/T 8566—2001 信息技术软件生存周期过程》为软件生存周期过程建立了一个公共框架，它包括了在含有软件的系统、独立软件产品和软件服务（软件包括固件的软件部分）的获取期间，以及在软件产品的供应、开发、运作和维护期间需应用的过程、活动和任务。

该标准适用于系统和软件产品，以及服务的获取，还适用于软件产品和固件的软件部分的供应、开发、操作和维护，可在一个组织的内部或外部实施。该标准为系统和软件产品，以及服务的获取者编写，也是为软件产品的供方、开发者、操作者、维护者、管理者、质量保证管理者和使用者编写。该标准适用于供需双方，它覆盖从一项非正式协议直到法律约束的合同。若此双方来自同一组织时也可等同应用。该标准还可由单方作为自我改进工作采用。

该标准等同采用了国际标准“IDT ISO/IEC 12207-1995 Information technology-Software life cycle processes”，既适用于软件产品，也适用于信息系统（含有软件的系统）。

该标准所规定的软件及信息系统生存周期的过程、活动和任务如表 29-1 所示。

表 29-1 软件及信息系统生存周期的过程、活动和任务

	过程名	主要活动和任务描述
主要过程	获取过程 acquisition	定义、分析需求或委托供方进行需求分析然后认可；招准备；合同准备，以及验收
	供应过程 supply	评审需求；准备投标；签订合同；制订并实施项目计划；开展评审及评价；交付产品
	开发过程 development	系统需求分析；系统结构设计；软件需求分析；软件结构设计；软件详细设计；软件编码和测试；软件集成；软件合格测试；系统集成；系统合格测试；软件安装及软件验收支持
	运行过程 operation	制订并实施运行计划；运行测试；系统运行；对用户提供帮助和咨询
	维护过程 maintenance	问题和变更分析；实施变更；维护评审及维护验收；软件移植及软件退役
支持过程	文档编制过程 documentation	设计文档编制标准；确认文档输入数据的来源和适宜性；文档的评审及编辑；文档发布前的批准；文档的生产与提交、储存和控制；文档的维护
	配制管理过程 configuration management	配置标识；配置控制；记录配置状态；评价配置；发行管理与交付
	质量保证过程 quality assurance	软件产品的质量保证；软件过程的质量保证，以及按 ISO 9001 标准实施的质量体系保证
	验证过程 verification	合同、过程、需求、设计、编码、集成和文档等的验证
	确认过程 validation	为分析测试结果实施特定的测试；确认软件产品的用途；测试软件产品的适用性
	联合评审过程 joint review	实施项目管理评审（项目计划、进度、标准、指南等的评价）；技术评审（评审软件产品的完整性、标准符合性等）
	审计过程 audit	审核项目是否符合需求、计划、合同，以及规格说明和标准
	问题解决过程 problem resolution	分析和解决开发、运行、维护或其他过程中出现的问题，提出响应对策，使问题得到解决
组织过程	管理过程 management	制订计划；监控计划的实施；评价计划实施；涉及到有关过程的产品管理、项目管理和任务管理
	基础设施过程 infrastructure	为其他过程所需的硬件、软件、工具、技术、标准，以及开发、运行或维护所用的各种基础设施的建立和维护服务
	改进过程 improvement	对整个软件生存期过程进行评估、度量、控制和改进
	培训过程 train	制订培训计划；编写培训资料；培训计划的实施

如果说关于是否采用过程作为信息系统项目管理的基础在小型项目中还存在争议的话，对于大型、复杂项目来说则必须建立以过程为基础的管理体系。一个大型项目团

队,协作的效率要远远高于个体的效率,而过程正是体现在这一点上,过程作为一个项目团队内部共同认可的制度而存在,它主要起到约束各个相关方以一致的方式来实施项目。

虽然 GB/T 8566—2001 规定了软件及信息系统的通用过程,这一通用过程框架可用于各种不同类型的信息系统、各种不同的应用领域、各种不同的项目组织、各种不同的功能级别,以及各种不同的项目规模,但由于每个具体项目都有各自的特征,都有其各自不同的要求,所以为每个项目单独建立一套适合的过程是有益的。当然,建立项目过程也存在成本,需要平衡成本和收益。但对于一个大型、复杂项目来说,为项目单独建立一套适合的过程,其收益会远远大于建立过程本身的成本。

一般可采用裁减通用过程的方式建立每个具体的项目过程,裁减哪些、保留哪些、是否需要增加新的过程,取决于项目的规模与特征。

当确定了项目过程后,就需要制订项目计划。一个项目的计划是表述如何实现项目目标的具体过程,一般可从以下 4 个方面对项目目标进行定义。

- (1) **范围**——项目要完成的工作是什么。
- (2) **质量**——项目结果需要达到什么样的要求。
- (3) **进度**——项目完成所需要的时间。
- (4) **成本**——项目完成所需要的代价。

大型、复杂项目一般均可分解为若干个子项目,在制订大型、复杂项目的项目计划时,不仅应制订整个项目的范围、质量、进度和成本计划,还应确定每一个子项目的范围、质量、进度和成本要求,以及各个子项目之间的相互依赖、相互配合和相互约束关系,为每一个子项目的绩效测量和控制提供一个明确的基准线,使整个项目的实施和控制更易操作,责任分工更加明确。

## 29.3 大型、复杂项目的实施与控制过程

大型、复杂项目规模庞大,团队构成复杂,项目实施过程中的监督和控制尤为重要。控制过程的主要任务和目标是:获取项目的实施绩效,将项目实施的状态和结果与项目的基准计划进行比较,如果出现偏差及时进行调整和变更。

由于项目目标是范围、质量、进度和成本(资源)等几方面的集合,无论是基准计划还是实施绩效,都要从这几个方面来反映项目的特征。另外,对于大型、复杂项目,协作的作用特别突出,所以在控制过程中特别增加了协作管理的内容。

### 29.3.1 范围控制

对于大型、复杂项目,在项目实施过程中项目范围发生变化的事情是经常发生的,项目范围的变化称为项目范围变更。项目范围变更的主要原因包括:

**(1) 需求原因**——项目实施过程中，客户（用户）对项目、项目产品或服务的要求发生变化。

**(2) 技术原因**——项目实施过程中，研发人员发现或提出了制订范围计划时尚未出现的新技术或新方案，实行新技术或新方案可能会提高项目质量、加快项目进度或减少项目成本。

**(3) 组织原因**——项目实施过程中，项目实施组织本身发生变化。例如，某一合作伙伴退出，又找到了新的合作伙伴。

**(4) 外部原因**——项目执行过程中，外部环境发生变化。例如，国家发布了新的标准或规范。

**(5) 计划疏漏**——项目执行过程中，发现原来制订的项目范围计划存在错误或有遗漏。

项目范围的变更是不可避免的，项目范围控制的主要任务就是采用科学的策略和方法，对项目范围变更实施控制和管理，实现项目范围变更的规范化和程序化。未经控制的范围变更则被看做范围溢出。

除了范围变更控制之外，项目范围控制还有另一项重要任务——项目范围纠偏。

项目具体实施时有可能出现偏离项目范围计划的现象。范围控制人员应从项目绩效报告中提取关于范围绩效的信息，分析其是否偏离了范围管理计划，以及 WBS（工作分解结构）定义的项目范围基线。如果有偏离，应测量偏离的程度，分析偏离的原因，并提出纠正措施建议。

大型、复杂项目范围控制的工具和技术与一般项目范围控制的工具和技术是一致的，但我们应特别注意：大型、复杂项目的执行组织一般是由许多个子项目部组成的，如果项目范围变更的请求来自某一个子项目部，一定要仔细分析该项变更对于其他子项目的影响。经常会出现这样的情况：从局部的观点看，一项变更对于某一个子项目是有利的，可能会减少成本、加快进度；但因该项变更会引起其他子项目范围的巨大变动，从全局观点看，反而是得不偿失。由此可见，大型、复杂项目的范围变更应该特别谨慎，一定要减少项目中盲目的变更，降低变更的随意性。但我们也不能因噎而废食，该变的还是要变。对于因外部环境改变、客户需求改变、范围计划疏漏，以及其他无法拒绝的原因所引起的项目范围变更，我们必须勇敢地面对，并以科学、务实的态度积极处理变更。由于大型、复杂项目的综合性和整体性，局部的变更可能牵动整体的变化，需要在每个变更发生时触发项目管理计划的更新。有些项目范围的变更，直接关系到项目时间和成本的增加。对于客户定制型项目，如果项目范围的变更是由客户原因引起的，可与客户进行合同变更谈判，要求客户增加投入；如果范围变更是由项目组织自身原因引起的（比如，范围计划疏漏），就只能由项目组织自行承担损失；如果范围变更是由外部环境变化引起的（比如，国家发布了新标准），双方可通过友好协商的方式确定各自应承担的责任。对于不针对特定客户的通用产品型项目，任何变更所引起的成本增加都只能由项目组织（或投资者）自行承担。

### 29.3.2 质量控制

与一般项目相比，大型、复杂项目的质量问题更加突出。

(1) 大型、复杂项目组织团队庞大，人员构成复杂，对问题的不同认识和误解如不能及时消除必然影响项目质量。

(2) 大型、复杂项目研发周期长，研发人员频繁流动，骨干研发人员的流失，会使项目质量受到一定影响。

(3) 大型、复杂项目系统故障定位比较困难。比如，一个信息系统的性能问题可能是由计算机性能、网络性能、数据库性能、中间件性能和应用软件性能共同决定的，某一方面出现问题，就会影响整体的性能。

(4) 大型、复杂项目质量纠纷认定的难度大，各项目部、各合作方由于理解的差异，对质量问题的认定容易产生分歧。

(5) 大型、复杂项目可视性差，质量缺陷比较隐蔽，无法直接通过人的感观系统直观地判断一个信息系统的优劣，某些质量问题往往在特定条件下才会出现。

质量控制在大型、复杂项目的管理中占有特别重要的地位。确保大型、复杂项目的质量，是项目研发人员和项目管理人员的共同使命。

质量控制活动具体监控信息系统项目的进程和结果，以确定其是否符合质量计划中所规定的质量标准；分析产生质量问题的原因，并制订相应措施来消除导致不符合质量标准的因素，确保项目质量得以持续不断地改进。质量控制活动包括通过由内部或外部机构进行的监测管理，发现与质量标准的差异，消除成果或过程中不能满足质量要求的因素；还要审查质量标准，以确定可能达到的质量目标及为此需要支付的质量成本，并评价其费用效率，必要时可以修订质量标准或项目目标。

大型、复杂项目的质量控制手段主要包括：评审、测试和审计。

#### 1. 评审

根据 IEEE 1028 的定义，评审 (Review) 是对信息系统元素或者项目状态的一种评估手段，以确定其是否与计划的结果保持一致，并使其得到改进。

评审是信息系统工程质量控制的重要措施，一般是在主要的项目里程碑接近完成进行，可分为需求评审、总体设计评审、详细设计评审、验证和确认评审、功能检查、物理检查、综合检查和管理评审。

(1) **需求评审 (Requirements Review)**。在信息系统需求分析结束后必须进行系统需求评审，以确保在系统需求说明书中所规定的各项需求的合适性。

(2) **总体设计评审 (Preliminary Design Review)**。在信息系统总体设计结束后必须进行总体设计评审，以评价总体设计方案中所描述的系统总体设计在总体结构、外部接口、主要部件功能分配、全局数据结构，以及各主要部件之间的接口等方面的合适性。

(3) **详细设计评审 (Detailed Design Review)**。在信息系统详细设计结束后必须进

行详细设计评审，以评价详细设计方案中所描述的系统详细设计在每一个基本部件的功能、算法和过程描述等方面的合适性。

(4) **验证和确认评审 (Verification and Validation Review)**。在信息系统验证与确认计划完成后必须进行验证与确认评审，以评价验证与确认计划中所规定的验证与确认方法的合适性与完整性。

(5) **功能检查 (Functional Audit)**。在信息系统验收前，要对系统进行功能检查，以验证所开发的系统已经满足在系统需求说明书中规定的所有需求。

(6) **物理检查 (Physical Audit)**。在信息系统前，要对系统进行物理检查，以验证程序和文档已经一致并已做好了交付的准备。

(7) **综合检查 (Comprehensive Audit)**。在信息系统验收时，要允许用户或用户所委托的专家对所验收的系统进行设计抽样的综合检查，以验证代码和设计文档的一致性、接口规格说明的一致性（硬件和软件）、设计实现和功能需求的一致性、功能需求和测试描述的一致性。

(8) **管理评审 (Management Reviews)**。要对计划的执行情况定期（或按阶段）进行管理评审。

## 2. 测试

测试 (testing) 是信息系统质量控制的主要手段之一，也是在将系统交付给业主之前所必须完成的步骤。目前，信息系统的正确性证明尚未得到根本的解决，测试仍是发现系统错误（缺陷）的主要手段。测试的目的就是在信息系统正式移交业主之前，尽可能多地发现系统中的错误（缺陷）。

大型、复杂项目的测试过程包括测试计划、测试设计、测试执行和测试评估等阶段。下面给出 10 条有用的测试准则。

(1) 应该尽早地、不断地进行测试，把测试贯穿于开发过程的始终。

(2) 所有测试都应该能追溯到用户需求。从用户的角度看，最严重的错误是导致系统不能满足用户需求的那些错误。

(3) 大型、复杂项目的测试应该从中小项目的子项目开始，并逐步加大测试规模。

(4) 应该在测试之前就制订出测试计划。

(5) 根据 Pareto 原理，80% 的错误可能出现在 20% 的质量点，测试成功的关键是怎样找出这 20% 的质量点。

(6) 应该由独立于开发人员的第三方从事测试工作。

(7) 对非法和非预期的输入数据也要像合法的和预期的输入数据一样编写测试用例。

(8) 检查系统是否做了应该做的事仅是成功的一半，另一半是看系统是否做了不该做的事。

(9) 在规划测试时不要设想系统中不会查出错误。

(10) 测试只能证明系统中有错误，不能证明系统中没有错误。

### 3. 审计

借鉴传统的审计 (Audit) 手段, 客观评价信息系统及信息系统开发, 已成为信息系统工程质量控制的另一种行之有效的。最早的“审计”主要指“财务审计”。随着经济管理与科学技术的不断结合与日益渗透, 现代审计已经远远超出了仅对财务会计进行审查的狭窄范围, 不断向管理领域和技术领域渗透。信息系统审计可以说是财务审计、管理审计和技术审计相结合的一个典型。IEEE 计算机学会 (IEEE Computer Society) 职业实践委员会 (Professional Practices Committee) 发布的《软件工程知识体系指南 (2004 版)》(Guide to the Software Engineering Body of Knowledge 2004 Version), 已将审计列为质量控制的一项主要内容。

审计涉及信息系统的整个生命周期, 审计对象涵盖信息系统所有活动和中间产物, 并包括与系统实施有关的外部环境。

系统审计按照系统的生命周期分为计划审计、开发审计、执行审计、维护审计, 以及覆盖整个系统周期的管理审计。

- 计划审计主要面向信息系统的企划, 对信息系统的投资可行性, 系统规划与公司战略的相关性, 系统开发计划的可行性, 以及系统需求的完整性和正确性进行审核和验证。
- 开发审计对信息系统开发的各个阶段的相关人员的活动、信息、中间产物进行审核, 确认这些活动、信息和中间产物的规范性、有效性和对于信息系统目标的针对性。
- 执行审计确认与信息系统运行相关的数据、软硬件、安装环境等是否符合信息系统的运营要求, 同时对信息系统的功能、性能、易用度、可操作性等进行评估。
- 维护审计对信息系统的维护活动和维护结果实施审核和评价。发现在维护中可能出现的各种漏洞和信息系统维护中亟待改善的问题。
- 管理审计涉及文档管理、进度管理、人员管理、采购管理、风险管理等, 检查这些过程的规范性和有效性, 并提出改良建议。

#### 29.3.3 进度控制

项目进度控制是依据项目进度计划对项目的实际进度情况进行控制, 使项目能够按时完成。有效项目进度控制的关键是监控项目的实际进度, 及时、定期地将其与计划进度进行比较, 并立即采取必要的纠正措施。大型、复杂项目往往是由逐级分解的成千上万个相对独立的任务组成的, 这些任务可分为“关键任务”和“非关键任务”。前者的进度一旦拖后, 整个项目的完成日期就会拖后; 而后者的完成时间如果没有严重拖后, 一般不会影响整个项目的完成时间。大型、复杂项目进度控制的重点是关键任务的进度控制。

进度控制的内容包括：(a) 确定当前进度的状况；(b) 改变某些因素使进度朝有利方向改变；(c) 确定原有的进度是否已发生改变；(c) 当实际进度发生改变时加以控制。

大型、复杂项目进度计划中一般都设有项目的关键点和里程碑，进度控制的过程如下：按设定的关键点、里程碑检查和评审项目的进展情况；将实际进度与基准进度比较；对进度评审结果进行分析，以便确定项目进度变化情况和预测发展趋势，并制订相应的对策——进度纠偏或修改基准进度。

大型、复杂项目的进度控制必须与其他控制过程紧密结合。例如，当项目范围发生变化时，应尽量采取有效措施确保项目按原计划完成。但并不是在任何情况下都能随心所欲，很多情况下我们不得不变更项目进度计划。

大型、复杂项目进度控制常用的工具和技术包括：

### 1. 甘特图

甘特（Gantt）图是一种历史悠久、应用广泛的进度管理工具，它以水平线段表示子任务的工作阶段，线段的起点和终点分别对应着子任务的起始时间，线段长度指示完成该任务所需要的时间。

甘特图的优点是直观简明、易学易绘、可从图上清楚地标出子任务间的时间对比，但它也有三个主要缺点：不能显式地描绘各项任务彼此间的依赖关系；进度计划的关键部分不明确，难以判定哪些部分应当是主攻和主控的对象；计划中有潜力的部分，以及潜力的大小不明确，往往造成潜力的浪费。

当把一个软件项目分解成许多子任务，并且它们彼此间的依赖关系又比较复杂时，仅仅用甘特图作为安排进度的工具是不够的。

### 2. PERT 图与关键路径

PERT（Program Evaluation and Review Technique，计划评估与评审技术）图是一个由箭头（标志任务）和结点（标志事件）组成的有向图。

箭头之上所标的是完成该任务所需的时间，箭头之下所标的是在不影响整个工期的前提下完成该任务还有多少机动时间。完成任务时间为 0 的任务称为空任务，用虚线箭头标识。

每个事件都有一个事件号和出现该事件的最早时刻和最迟时刻。最早时刻表示在此时间之前从该事件出发的任务不可能开始；最迟时刻表示从该事件出发的任务必须在此时刻之前开始，否则整个工程就不能如期完成。

PERT 图不仅给出了每个任务的开始时间、结束时间和完成该任务所需的时间，还给出了任务之间的依赖关系，即哪些任务完成后才能开始另一些任务，以及如期完成整个工程的“关键路径”。

关键路径（Critical Path）是由一连串的任务所组成的链，它决定了完成该项目所需的时间。关键路径上的事件（关键事件）必须准时发生，组成关键路径的任务（关键任务）的实际持续时间不能超过估计的持续时间，否则工程就不能准时结束。

如果一条路径中所有任务的机动时间均为 0，这条路径显然为关键路径，但反之并不总是成立。

如果一条路径中所有事件的最早时刻和最迟时刻均相同，这条路径显然也为关键路径，但反之也并不总是成立。

软件项目的管理人员应该密切注视关键任务的进展情况。如果希望缩短工期，只有往关键任务中增加资源才会有效果。

### 3. 持续时间的压缩

持续时间压缩是对进度计划进行数学分析的一种特殊情况，即寻找在不改变项目范围的条件下，缩短项目持续时间的途径。持续时间压缩的技术有：

(1) **赶工**。对费用和进度进行权衡，确定如何使用费用增加最少而项目总工期压缩最大。一般来说，赶工总是会导致费用增加。

(2) **快速跟进**。将一般情况下的多项先后顺序实施的活动改为并行实施。快速跟进有可能导致平行活动的相互干扰、资源冲突甚至返工，一般会增加风险。

### 4. 资源调配

当项目的实际进度滞后于计划进度时，通常可通过资源调配的方式缩短项目的工期。例如，把稀缺的资源分配给关键线路上的活动；投入更多的资源以加速项目进度；派经验更丰富的人去完成或帮助完成项目工作。

### 5. 时差的应用

时差度量一个活动在项目进度计划中时间安排的可调整程度，为活动的最晚时间与最早时间之差。在不影响整个项目结束时间的前提下，活动允许调整的时间称为“总时差”；在不影响后续活动开始时间的前提下活动允许调整的时间称为“自由时差”。

在解决资源分配冲突时，需要考虑资源在不同活动中分配的优先级别，时差是资源调配的重要依据，可以将紧缺资源从时差较长的活动调到关键活动上去。

## 29.3.4 资源控制

项目资源，主要指项目所需的人、设备、材料、技术、信息等。资源控制，也称成本控制，是指项目组织为保证在变化的条件下实现其资源计划和预算成本，按照事先拟定的计划和标准，对项目实施过程中发生的各种实际成本与计划成本进行对比、检查、监督、引导和纠正，尽量使项目的实际成本控制在计划和预算范围内的管理过程。随着项目的进展，根据项目实际发生的成本，不断修正原先的成本估算和预算安排，并对项目的最终成本进行预测的工作，也属于项目成本控制的范畴。

大型、复杂项目规模大、时间长，项目成本的不确定因素较多。一旦项目成本失控，要在预算内完成项目是非常困难的。如果项目没有额外的资金支持，可能导致项目范围缩小、项目进度推迟，甚至会降低项目质量。为避免发生此类风险，应及时分析成本绩

效，尽早发现实际成本与计划成本的差异，以便在情况变坏之前能够采取纠正措施。另一方面，大型、复杂项目的范围计划与进度计划经常会变更，当项目范围与进度发生变化时，应尽量采取有效措施确保项目按原计划成本完成，但如果因为控制成本而有可能影响项目质量，我们将不得不变更项目资源计划和成本预算。

大型、复杂项目成本控制的内容包括：

- 对造成实际费用偏离基准费用计划的因素施加影响，以保证其朝着有利的方向发展；
- 确定基准费用计划是否必须变更和如何变更；
- 基准费用计划需要变更或已经变更时，对这种变更进行管理。

大型、复杂项目成本控制的的活动包括：

- 对照基准费用计划，找出实际费用与基准计划费用的偏差，做好费用分析和评估，并对费用偏差做出响应；
- 确保所有有关变更都准确地记录在基准费用计划中；
- 防止不正确、不适宜或未经核准的变更纳入基准费用计划中；
- 将核准的变更通知项目有关人员；
- 监控影响项目费用的内外部因素，以便预测项目费用的变化，必要时调整基准费用计划。

为了能在项目生命期内监测实际的支出和控制项目费用，应建立和使用资源管理系统。

大型、复杂项目成本控制的技术方法主要包括以下几种。

### 1. 费用分解结构（CBS）

将费用按照与工作分解结构（WBS）和组织分解结构（OBS）相适应的规则进行分解，并形成相应的、便于管理的账目分解结构（ABS）。ABS是组织单元为承担分项工作而对其费用加以管理的一种工具。分解的结果，可作为项目费用测定、衡量和控制的基准。

### 2. 挣值分析

挣值（Earned Value）表示已完成工作的计划费用或预算费用。

挣值分析是测量执行情况常用的方法，它综合了范围、费用和进度测量，帮助项目管理班子评价项目执行情况。挣值涉及计算任务和工作单元（或项目）的3个值：

- 计划费用值（Planned Value, PV）：在规定时间内，计划在某个任务和工作单元（或项目）上的预算费用；
- 实际费用值（Actual Costs, AC）：在规定时间内，在实际已经完成任务和工作单元（或项目）上的实际发生的直接与间接费用的总和；
- 挣费用值（Earned Value, EV）：在规定时间内，在已经完成活动和工作单元（或项目）上，按计划应当花费的预算费用。这个值往往用正在完成工作的预算费

用的一个实际完成百分比来计算，比如 30%，50%，70%，90%，100%以简化数据的收集。有的挣值分析仅用 0%或 100%（未完成或已完成）来计算，也有的用 0%或 50%或 100%（未开始或已开始或已完成）来计算。

这 3 个值的综合使用可以提供评价工作是否按照计划完成的尺度。常用的是费用偏差（ $CV=EV-AC$ ）、进度偏差（ $SV=EV-PV$ ）和费用绩效指数（ $CPI=EV/AC$ ）。CPI 被广泛用于预测完工时的项目费用。进度绩效指数（ $SPI=EV/PV$ ）被广泛用于预测项目完工日期。

如果进度偏差为正，表示项目进度提前；为 0，表示进度符合计划；为负，表示进度拖后。对于进度拖后的项目要及时采取适当措施。

如果费用偏差为正，表示项目费用在预算之内；为 0，表示项目费用符合预算；为负，表示费用超出预算。对于超支的项目要及时采取适当措施。

### 3. 类比估算法

类比估算法也称自上而下估算，即利用以前类似项目的实际费用作为估算当前项目费用的基本依据，是一种专家评定法，精度较低。如果被估算的项目与以前的项目是相似的，且进行估算的个人或团体具有所需要的专门知识，则这种方法也是可靠的。

### 4. 参数模型法

将项目特征（参数）用于数学模型，可以预测项目费用。模型可以是简单的，也可以是复杂的。如果开发模型的历史资料可靠，模型中所用参数可以被量化，模型可以按比例调整，则预测结果会相当精确。

### 5. 自下而上的估算

先估算项目各个活动的费用，然后按工作分解结构（WBS）的层次自下而上地汇总，估算出总费用。这种方法也可用于编制费用计划。只要项目各个活动的费用估计得准确，工作分解结构合理，用这种方法估算的结果和由此编制的费用计划一般比其他方法更精确。但是这种方法的估算工作量也是最大的。

### 6. 计算机估算软件

项目管理软件或办公自动化软件等工具可以加速费用估算和费用计划编制，并提供多种方案的费用比较和选择。

### 7. 费用变更识别和变更控制系统

费用变更识别和变更控制系统规定了改变基准费用计划应遵循的程序，包括文档、追踪系统、核准变更必需的批准层次等。

### 8. 完成项目所需成本估计

完成项目所需成本估计（EAC）是以项目执行的实际执行情况为基础，对整个项目成本的一个预测。最常见的 EAC 有以下几种：

(1)  $EAC = \text{实际已发生成本} + \text{对剩余的项目预算}$ （但一般用成本执行因子对原预算进行修正），在项目现在的偏差可视为将来偏差时，这种方法通常被利用。

(2)  $EAC = \text{实际已发生成本} + \text{对剩余项目的一个新估计值}$ 。当过去的执行情况表明先前的成本假设有根本缺陷或由于条件改变而不再适用新的情况时，这种方法最为常见。

(3)  $EAC = \text{实际已发生成本} + \text{剩余原预算}$ 。当现有偏差被认为是不正常的（由偶然因素引起）项目管理小组认为类似偏差不会发生时，用这种方法最为常见。

不同的工作可选用上述方法中的一种。

### 9. 总结教训

应记录下产生偏差的原因、采取纠正措施的理由和其他的成本控制方面教训，这样记录下来的教训便成为这个项目和执行组织其他项目历史数据库的一部分。

## 29.3.5 协作管理

大型、复杂项目组织团队庞大，参与单位众多，协作管理的效益尤为突出。一般来说，大型、复杂项目的协调管理可分为项目组织内部的协调和项目组织外部的协调。

### 1. 项目组织内部的协调

项目组织内部的协调是指一个项目组织内部各种关系的协调，如人际关系协调、组织关系协调和资源需求协调等。

**(1) 人际关系协调。**任何协调工作最终都表现为人与人之间的往来，而良好的人际关系可以使项目组成员易于沟通、相互信赖、相互支持，同时人际关系的渗透性和扩散性反过来也可以提高项目组织的效率。大型、复杂项目规模庞大，人际关系复杂，如何提高每个人的工作效率，在很大程度上取决于人际关系的协调程度。可以说，和谐的人际关系是项目成功的基础。

**(2) 组织关系协调。**大型、复杂项目的组织关系协调主要包括子项目组之间的协调和项目参与单位之间的协调，有时也包括项目组织内上下级之间的协调。在某些项目组织中，一个参与单位也就是一个子项目组；在另外一些项目组织中，一个单位的人员可能分散在几个子项目组中，几个单位的人员也可能集中在一个子项目组中。组织关系协调的目标是使这些子项目组（参与单位）都能按照整个项目的质量计划、进度计划和资源计划，积极主动地完成本组织的工作，使整个项目处于有序的良性状态。

组织关系协调可采用多种灵活多样的方式：口头谈话、书面文件、电子邮件、电话、短信、即时通信（如QQ、MSN等）、工作例会、电话会议、电视会议、可即时查询的项目信息数据库和项目管理软件等。

具体采用哪种方式取决于：

- 信息需求的即时性——项目的成功是取决于即时通知频繁更新的信息，还是通过定期发行的报告就已足够？

- 项目人员技能——项目参与人员是否能够掌握即时通信、电子邮件、项目信息数据库和项目管理软件等先进的通信方式？
- 项目组织的通信条件——项目组人员是否都有电脑？是否可以上网？项目组是否有专用的通信软件？
- 通信成本——是否有必要为了联系的方便而支出大额的资金？

应当注意，任何先进的通信方式都只能作为一种协调手段，组织关系协调能否成功的关键因素是每一个子项目组（参与单位）是否有局部服从全局的意识。项目组内的不同单位，可能会有不同的利益追求，应本着求大同存小异的原则，互谅互让，互利互惠，力争“双赢”和“多赢”。

**(3) 资源需求协调。**大型、复杂项目的资源需求协调是指在项目实施过程中，为实现内部资源的合理配置，达到内部资源的平衡，对人员需求、材料需求、硬件需求、软件需求和其他资源需求而进行的协调活动。

## 2. 项目组织外部的协调

项目组织外部的协调，以是否具有合同关系为界限，划分为具有合同因素的协调和不具有合同因素的协调。具有合同因素的协调主要包括项目组织与相关产品、服务提供商之间的关系协调。如果项目是接受某一客户（或投资者）的要求开发的，还应包括项目组织与客户（或投资者）之间的关系协调。

与合同因素协调相比，非合同因素协调所涉及的范围更广，可能遇到的问题更多，协调工作量更大、更复杂，而这些往往不是事先签好合同就可以进行约束的。

非合同因素协调工作主要涉及技术质量监督检测机构、专业测试与评估机构、行业许可和准入的管理机构、产品认证和管理体系认证机构、软件著作权登记和管理机构、社会服务机构（如投资咨询机构）、行政主管部门、行业协会、专业学会和其他社会团体等。这些机构与项目组织并无合同关系，但他们的作用不可低估，对项目建设的某些方面起着一定的控制、监督和支持的作用。只有将各方面的关系都协调好，才能保证实现项目的各项目标。

## 29.4 多项目管理

一个组织往往同时进行许多项目，虽然从项目目标和执行团队上看，这些项目可能是孤立的、无关联的（如果项目彼此关联，可视为一个大型项目），但我们应该意识到：

- 这些项目均服务于组织的战略规划和产品布局。
- 这些项目共享组织的资源，包括人力资源、技术资源、信息资源、物资资源和资金资源，资源的调配会在项目之间产生影响。
- 项目组之间可以彼此借鉴其成功经验和失败的教训，共同提高整个组织实施项目的能力。

由此可以看出,这些看似孤立的项目需要在组织层面上以某种方式进行统筹规划与管理,从而保证其产品布局的合理化和良性化,全面促进组织战略目标的实现,提高整个组织的项目管理能力。这就是多项目管理的根本目标。

多项目管理确保组织在项目选择、计划、实施,以及在处理项目之间的冲突时,以组织战略目标为导向,从而保证整个组织在大方向上的一致性。

### 29.4.1 不要把所有鸡蛋都放在一个篮子里

前面我们已经讨论了大型、复杂项目的管理。那么,能不能把一个组织内的所有项目合在一起作为一个大型、复杂项目管理呢?

表面上看,大型、复杂项目的管理和多项目管理有很多相似之处。大型、复杂项目肯定是由许多个中小型项目组成的;反之,如果把一个组织内的所有项目合在一起,一般也可看做是大型、复杂项目。大型、复杂项目管理中的分级管理与分工管理、资源管理和协作管理方法,同样适用于多项目管理。但大型、复杂项目管理和多项目管理,在管理依据、管理原则和管理目标方面,都是截然不同的。

大型、复杂项目的风险具有一定的整体性,尽管项目可以分解,项目管理也可以分解,但项目风险(尤其是市场方面的风险)并没有降低。如果一个组织把他们的大部分资金和技术力量都赌博似的放进一个庞大的、充满风险的项目中,一旦项目失败,对于整个组织可能是一场灭顶之灾。在中国,每年都有一大批IT企业因此而一蹶不振或者干脆破产关门。

古人早就告诫我们:不要把所有鸡蛋都放在一个篮子里!这一原则正是多项目管理的出发点和依据。举个形象的比喻,多项目管理要解决的问题就是:应该把鸡蛋放在几个篮子里?每个篮子里应该各放几个鸡蛋?

玩过股票的人都知道,只持有一种股票或同一类型的股票是非常危险的。理想的投资组合是同时持有几种不同类型的股票:长线股,短线股,高科技股,传统业股,蓝筹股,活跃股,绩优股,资产重组股,等等。至于持有哪几类,哪几种,每种各持有多少,投资机构都有专家专门研究。

更进一步,如果将所有资金均用于股票投资,这实际上也是把所有鸡蛋都放在一个篮子里。一旦遇到大盘巨跌,危巢之下岂有完卵!经历过最近几年股市暴跌的股票投资者想必都有切身体会。如果将资金分成几份,分别用于存款、购买债券、股票投资和其他投资,投资风险就会降低很多。

由于项目管理领域的多项目管理,最初就是来源于金融投资领域,所以有时又被称为项目投资组合管理(Project Portfolio Management)。

### 29.4.2 风险与收益的平衡

项目投资组合管理要求对组织内部的所有项目都进行风险评估和收益分析,并且随

着项目的进展，持续地跟踪项目的风险和收益变化，以掌握这些项目的状态。

根据项目的风险评估和收益分析结果，可将项目粗略地分为以下 4 类。

- A 类：低风险，高收益；
- B 类：低风险，低收益；
- C 类：高风险，高收益；
- D 类：高风险，低收益。

毫无疑问，对于高风险、低收益的项目（D 类），坚决不能立项。最理想的当然是低风险、高收益的项目（A 类），但遗憾的是，现实世界中的这类项目太少，更多的都是低风险、低收益（B 类）或者高风险、高收益（C 类）的项目。

任何组织，如果只在高风险、高收益的项目（C 类）上全力以赴，很可能会使组织陷入困境；但如果只在低风险、低收益的项目（B 类）上投资，组织就不可能得到大的发展。

对于一个具体的项目，可以在项目的风险和收益之间找到平衡点，做到风险最小化、效益最大化，这属于项目可行性分析和项目立项管理所研究的内容。而项目投资组合管理的着眼点则是在一个组织的所有项目的整体风险和整体收益之间找到平衡点，实现整个组织的风险最小化和收益最大化。当然，“整体”也是由一个个的“个体”构成的。如果每个项目都实现了“风险最小化、效益最大化”，整个组织自然也就实现了“风险最小化、效益最大化”。但我们在做一个具体项目的可行性研究时，往往首先考虑的“风险最小化”，在此前提之下，才会考虑“效益最大化”。实际上我们所追求的是“风险最小化前提之下的效益最大化”。而对于一个组织，我们不仅仅要进行每一个项目的风险评估和收益分析，还要进行整个组织的风险评估和收益分析。在确保组织的整体风险和整体收益平衡的原则下，可以组织开发少量的“高风险、高收益”类项目。至于哪一类项目应占多大比例，取决于组织的战略目标和战略方法。

### 29.4.3 目标与资源的平衡

组织的战略目标决定了组织发展的方向。但需要注意的是，组织的战略方法存在两种不同的倾向：目标导向和资源导向。前者关心的是组织的外部因素，即市场上可能存在哪些机会；后者则更关心组织的内部因素，即组织自身的核心优势。目标导向所发现的市场机会主要体现为所期望的收益因素，而资源导向所阐明的组织自身的优劣势则体现在可能出现的风险因素之中。最理想的状况当然是，目标导向和资源导向得出的结论完全匹配。但在现实中，目标导向和资源导向得出的结论往往并非完全匹配。

一种情况是，已经发现了市场的某一个机会，但组织自身在该领域并无资源优势。通俗地说，就是“有市场、无技术”。由于 IT 行业的市场变化和行业发展都很快，没有一家企业可以长期保持很强的技术优势，后来者居上的案例是屡见不鲜的。具体做法有几种：

- (1) 引进该领域的顶尖人才；

(2) 购买该领域的国内外先进技术（如专利或技术成果）；

(3) 与该领域的顶尖专家或机构联合研发。

另一种情况是，组织自身在某一领域具有很强的资源优势，但该领域的市场需求不大，也就是说“有技术、无市场”。对于这种情况，还应该进一步分析，市场需求不大的原因是什么。

(1) 如果市场已经饱和，要考虑用户有无升级的打算，准备向哪个方向升级，什么时间才会升级。如果确信用户在几年之后可能会大规模升级，可抽调少量技术力量，作为中长线项目先行研发。

(2) 如果市场还处在初创期，用户还在观望，要考虑什么时间市场才能进入高成长期。如果确信在几年之后会有大规模的市场需求，也可抽调部分技术力量，作为中长线项目先行研发。

(3) 如果可以确认自己研究的是“屠龙术”，根本就没有市场需求，那就要考虑改变研究方向了（国家资助的基础理论研究者除外）。

无论在哪种情况下，片面强调市场需求和片面强调资源优势都是不足取的，还是要懂一点“中庸之道”，在市场需求和资源优势之间找到一个平衡点。一方面，要尽量在组织优势领域的临近领域寻找市场机会。另一方面，也要让组织的资源优势向着有市场潜力的临近领域发展。

如果一个组织发现在自己的传统优势领域，市场需求正在逐渐萎缩，但还可以坚持几年的时间，理想的做法是：保留部分技术力量，继续从事原产品的维护和升级开发；同时抽掉部分技术力量，开发市场潜力更大的新产品。也就是说，部分项目是资源导向的，部分项目是目标导向的，这也是目标与资源相平衡的一种表现。