# 软件项目管理

13061147 文彬

#### 一．软件项目管理的基本要素：

1.资源

2.项目组织

3.需求和目标

.4.项目环境

二．软件项目管理的必要特征：  
1. 软件项目是一种逻辑实体，不是具体的物理实体，它具有抽象性

2. 软件本身是复杂的。它的复杂性源于应用领域实际问题的复杂性和应用软件技术的复杂性。

3.面向对象的实现方法。

4.软件的多次完善性。

5. 软件开发和软件项目管理工作涉及到很多社会因素，要受到机构政治、文化、决策体系和管理方式的限制。

6. 软件没有硬件的机械磨损和老化问题。然而，软件也存在退化问题，在软件生存期中，软件环境的变化将导致软件失效率的提高。

#### 三．用UML模型实现软件项目管理的基本要素和必要特征：

软件系统的不同模型可以捕获关于这个软件的应用领域、使用方法、度量手段和构造模式等方面的需求信息。在编写程序代码以前，软件系统的模型可以帮助软件开发人员方便地研究软件的多种构架和设计方案。软件系统的一类模型可以说明这个系统的外部行为和系统中对应于真实世界的有关信息，另一类模型可以展示系统中的类以及实现系统外部行为特性所需要的内部操作。实现这些行为有多种方法。利用软件系统的模型，可以获得类的声明、过程体、用户界面、数据库、合法使用的说明、配置草案以及与其他单位技术竞争情况的对比说明。软件系统用视图来组织信息：静态结构视图、状态机视图、交互视图、反映需求的视图等等。

第一类是用例视图(Use case View),强调从用户角度看到的或需要的系统功能,并指出各功能的操作者。包括用例图。

用例图用来描述系统功能。

第二类是静态视图(Static View),展现系统的静态或结构组成及特征。包括类图、对象图和包图。

类图描述系统中类的静态结构。不仅定义系统中的类,表示类之间的联系如关联、依赖、聚合等,也包括类的内部结构(类的属性和操作)。类图描述的是一种静态关系,在系统的整个生命周期都是有效的。

对象图是类图的实例,几乎使用与类图完全相同的标识。不同点在于对象图显示类的多个对象实例,而不是实际的类。

包由包或类组成,表示包与包之间的关系。包图用于描述系统的分层结构。

第三类是行为视图(Behavior View),描述系统的动态模型和组成对象间的交互关系。包括状态图和活动图。

状态图描述类的对象所有可能的状态以及事件发生时状态的转移条件。通常状态图是对类图的补充。在实用上并不需要为所有的类画状态图,仅为那些有多个状态其行为受外界环境的影响并且发生改变的类画状态图。

活动图描述满足用例要求所要进行的活动以及活动间的约束关系,有利于识别并行活动。

第四类是交互图(Interactive View),描述对象间的交互关系。包括顺序图和协作图。

顺序图显示对象之间的动态合作关系,它强调对象之间消息发送的顺序,同时显示对象之间的交互;

协作图描述对象间的协作关系,协作图跟顺序图相似,显示对象间的动态合作关系。除显示信息交换外,协作图还显示对象以及它们之间的关系。

如果强调时间和顺序,则使用顺序图;如果强调上下级关系,则选择协作图。

第五类是实现图(Implementation VIew)。包括构件图和部署图。

构件图描述代码部件的物理结构及各部件之间的依赖关系。一个部件可能是一个资源代码部件、一个二进制部件或一个可执行部件。包含逻辑类或实现类的有关信息。构件图有助于分析和理解部件之间的相互影响程度。

部署图定义系统中软硬件的物理体系结构。它可以显示实际的计算机和设备(用节点表示)以及它们之间的连接关系，也可显示连接的类型及部件之间的依赖性。在节点内部,放置可执行部件和对象以显示节点跟可执行软件单元的对应关系。