

我们把极其复杂的研制对象称为系统，即由相互作用和相互依赖的若干组成部分结合成具有特定功能的有机整体，而且这个系统本身又是它所从属的一个更大系统的组成部分。

系统工程的目的是解决总体优化问题，从复杂问题的总体入手，认为总体大于各部分之和，各部分虽较劣但总体可以优化。

系统分析，旨在研究特定系统结构中各部分（各子系统）的相互作用，系统的对外接口与界面，以及该系统整体的行为、功能和局限，从而为系统未来的变迁与有关决策提供参考和依据。

分析是对问题和需求的调查研究，不是解决方案

功能模型。对系统的功能描述，包括用例和用例图等。

对象模型。对用例使用面向对象思想进行分析，比如抽象、封装、继承、聚合、关联等，把系统分解成互相协作的类，通过类图、包图来描述对象与对象以及对象内部关系，表明系统的静态特征。

动态模型。描述系统的动态行为。可以通过交互图、顺序图等来表示。

软件需求规格说明：

是客户和软件开发商（对于产品型的项目，这些角色可以由市场人员和开发部门担任）之间就软件产品达成的一致意见：软件产品做什么，软件产品不应该做什么。

它也为估算产品的成本、风险和进度提供一个现实的基础。

完整性

客户的所有要求都必须包含，软件开发人员能够从该文档中获取设计和实现软件功能的所有必要信息。

正确性

每项需求都必须准确表述。软件需求规格说明书需要用户的参与和确认，必须避免出现开发人员想当然的功能表述

可行性

每项需求都必须满足软硬件以及软件运行所在更高级系统的限制。在写作该文档时，要有软件开发人员的参与，避免出现技术上不可能实现的内容出现在文档中。

必要性

每项需求都必须来源于客户的真实需求，能够回溯到某个用例或用户的某个要求。

设计是使用分析的结果来描述系统如何实现的过程。

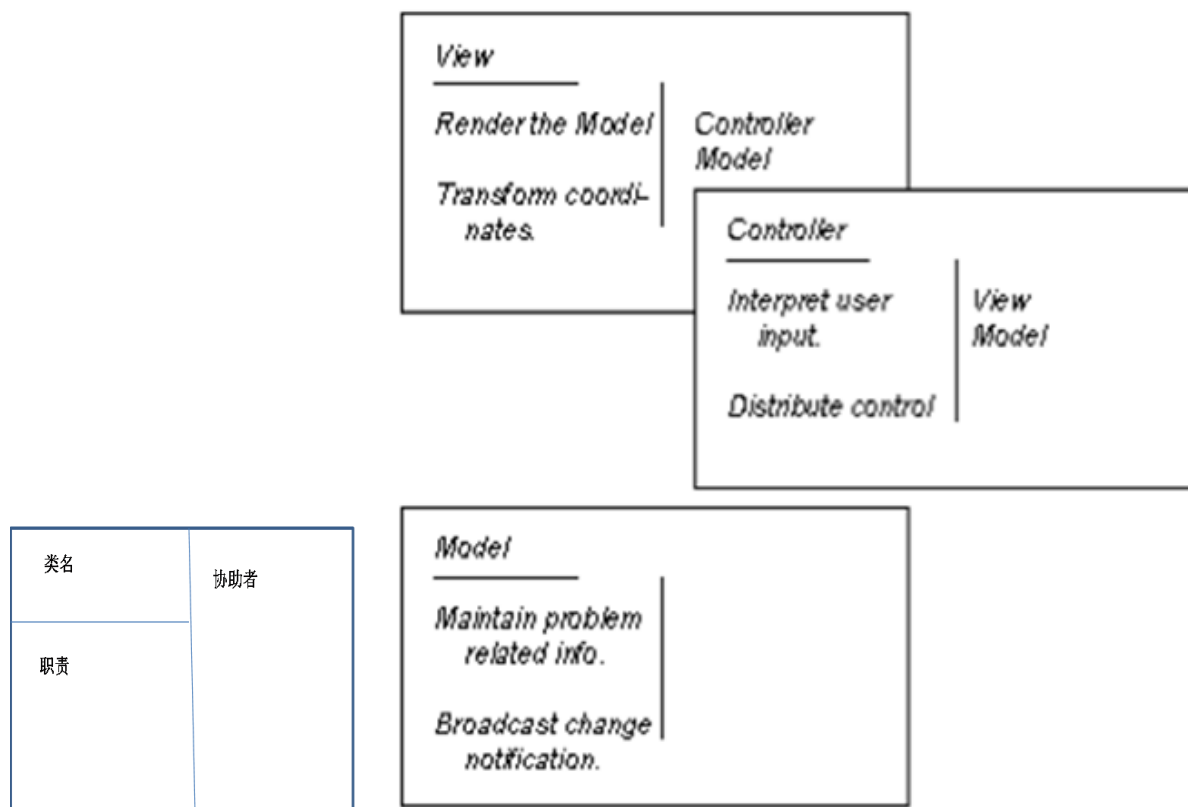
设计的结果是一组描述系统如何运转的逻辑说明，强调的是—个系统在概念上的满足需求的软硬件解决方案。

如果说分析是为了让软件开发设计人员清楚要做什么事情，那么设计则关注的是如何去把这件事情做得正确，而实现则是按照设计完成系统

设计被定义为“定义一个系统或组件的体系结构、组件、接口和其它特征的过程”和“这个过程的结果”。

作为过程看待时,软件设计是一种软件工程生命周期活动，在这个活动中分析软件需求，以产生一个将作为软件构造的基础的软件内部结构的描述。

更精确地说，软件设计（结果）必须描述软件体系结构（即,软件如何分解成组件并组织起来）和这些组件之间的接口，它必须在详细的层次上描述组件，以便能构造这些组件



比如上图中View和Controller有些重叠（表明它们之间有非常紧密的协助），View和Controller放在Model上面（它们是Model的表象）。

再例如，在表示一组继承关系时，开发者可以将这一组类放在一叠卡片中，最抽象的类可以放在最上面。

（）系统功能描述

- 1 系统说明：包括系统背景、系统用户和系统功能的简短介绍。
- 2 系统用例图：首先列举出系统可能的所有用户，再依次列出各用户的用例，并绘制成用例图。|在此过程中，请主要不要遗漏用例。
- 3 用例文本描述：针对2中的用例，依次写出用例文本描述。
 - 1.用例1：用例名称、参与者、正常流程、扩展流程、特殊需求。
 - 2.用例2：用例名称、参与者、正常流程、扩展流程、特殊需求。
 - 3.....
- 4 其它需注意事项：其它非功能性需求等。