**用例技术族**

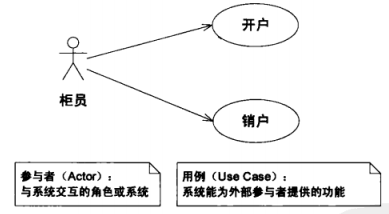
用例技术包括：用例图、用例简述、用例规约、用例实现

**用例图**

用例图的元素：参与者、用例

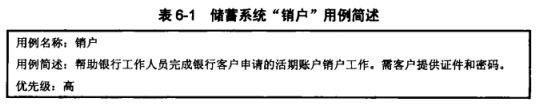
参与者：可以是系统，或用户

用例：用例是一个动词，是一个动作



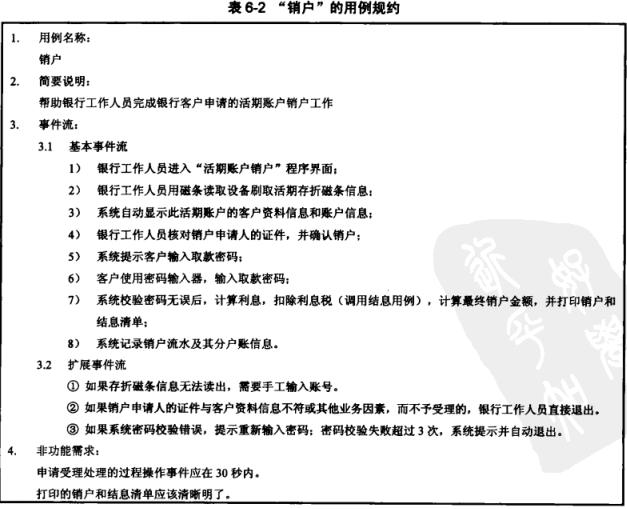
**用例简述**

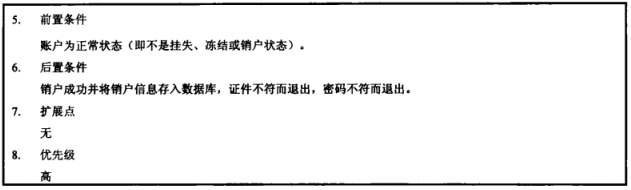
对用例简短的说明



**用例规约**

对用例详细的说明，包括流程的说明，如下：

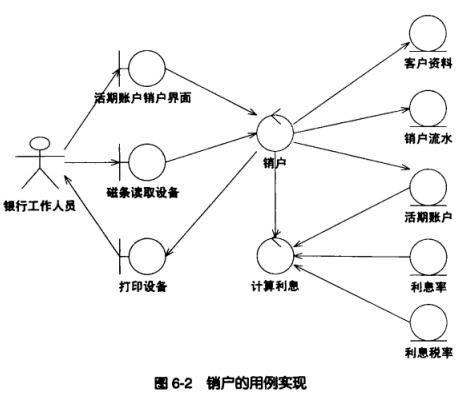




用例规约虽然详细，但不是每个用例都应该有用例规约，只有在难以理解的用例才需书写用例规约

**用例实现（不是代码的实现，是业务上的实现）**

用例实现=协作



**用例技术族应用场景**

**需求分析**

需求捕获：

用例图+用例简述可以用于需求捕获阶段

需求分析：

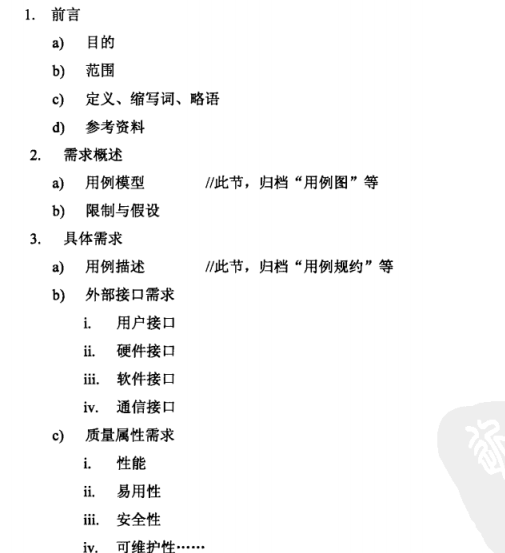
用例图+用例简述可以应用与不太复杂的系统的需求分析阶段，对于复杂的系统，一般使用用例规约

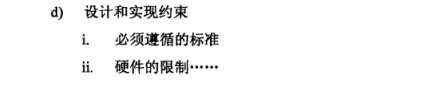
系统分析：

用例实现应用在初始设计中

**需求文档**

《软件需求规格说明书》





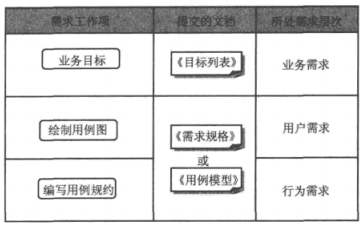
**需求变更**

用例图，用例简述在面对需求变更时比较稳定

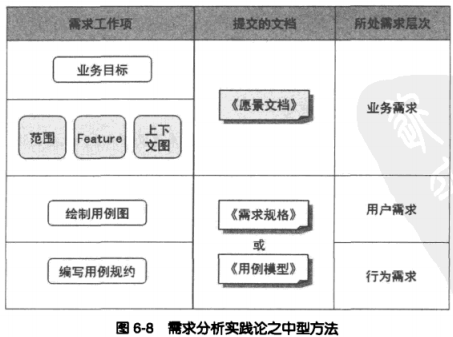
用例规约面对需求变更时不稳定，因为用例规约设计的比较细，所有用于规约一般在要定义功能时才进行细化

**只需用例建模就够的吗？**

**对于小型项目，我们的工作项**

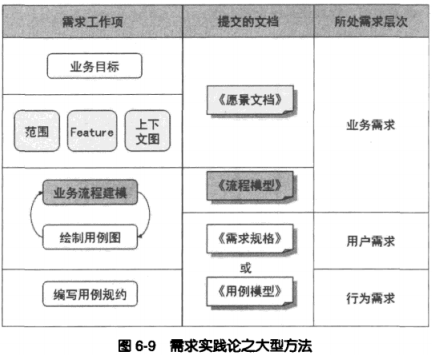


**对于中型项目**



**对于大型项目**

我们需要画流程模型，流程模型有助于我们发现用例



在流程模型中，需要我们系统辅助完成的，往往就是我们的用例，如下流程图示例

