

4. 软件项目范围计划 - 需求管理

1. 软件需求定义

需求是软件开发过程的基础，是软件项目建设的基石

需求是指用户对软件的功能和性能的要求，就是用户希望软件能做什么事情，完成什么样的功能，达到什么样的性能

需求的三个层次

- 业务需求
- 用户需求
- 功能需求

导出软件需求规格说明

特点

- 模糊性
- 不确定性
- 变化性
- 主观性

2. 软件需求管理过程

2.1. 需求获取

主要任务

和用户方的领导层、业务层人员进行沟通，了解现有的组织架构、业务流程、硬件环境、软件环境、现有的运行系统等具体情况和客观信息

2.2. 需求分析

需求分析是为最终用户所看到的系统建立一个概念模型，是对需求的抽象描述。

1. 任务

借助于当前系统的逻辑模型导出目标系统的逻辑模型，解决目标系统“做什么”的问题

2. 完成的标志

提交一份完整的软件需求规格说明书（SRS）

2.3. 需求规格编写

1. 目的

需求规格说明书是为了用户和开发者有共同的理解，使之成为整个开发工作的基础。

2. 意义

一个完整、规范的需求规格说明书是需求分析完成的标志

2.4. 需求验证

需求规格说明书作为输入、通过符号执行、模拟或快速原型等途径，评估

验证方面

- 需求正确性
- 需求一致性
- 需求的完整性
- 需求的可行性
- 需求的必要性
- 需求的可检验性
- 需求的可跟踪性

2.5. 需求变更

1. 意义

需求变更是软件项目的突出特点

2. 主要工作

- 1) 建立需求基线，每次变更并经过评审后，重新确定新的需求基线
- 2) 确定需求变更控制过程
- 3) 建立变更控制委员会（SCCB）
- 4) 进行需求变更影响分析
- 5) 跟踪所有受需求变更影响的产品
- 6) 建立需求基准和控制文档
- 7) 维护需求变更历史记录
- 8) 跟踪每项需求的状态
- 9) 衡量需求稳定性

3. 传统需求分析方法

3.1. 原型方法

- 1) 需求分析
- 2) 原型开发
- 3) 原型评价

3.2. 结构化分析法

主要是基于数据流建模方法

- 特点
 - 1) 面向数据流的方法
 - 2) 自顶向下逐步求精的方法

- 3) 根据软件内部数据传递、变换的关系进行分析的，帮助开发人员了解系统需要做什么，系统需要存储和使用那些数据，需要什么样的输入和输出，以及如何将这些功能结合在一起完成任务

- 技术

- 1) 数据流图（DFD）：描述软件系统逻辑模型的图形符号
 - 四种元素：过程、实体、数据流和数据存储
- 2) 数据字典（DD）：描述系统中设计的每个数据，是数据描述的集合
 - 数据项、数据流、数据文件等
- 3) 实体联系图（ERD）：描述系统需求要存储的数据信息
- 4) 系统流程图：表示操作顺序和信息流动过程的图表

3.3. 面向对象的用例分析法

1. 特点

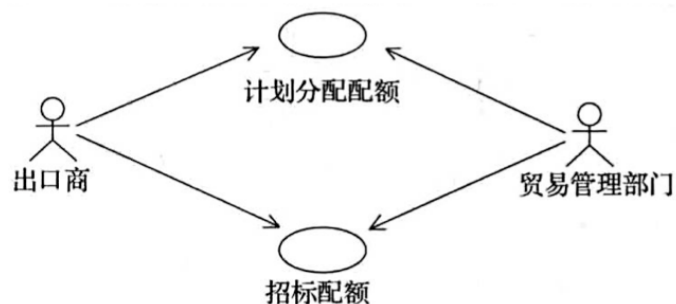
- 1) 基于面向对象的情景分析方法
- 2) 从用户角度出发考虑功能需求
- 3) 所有用例结合起来就是用例模型

2. 实现步骤

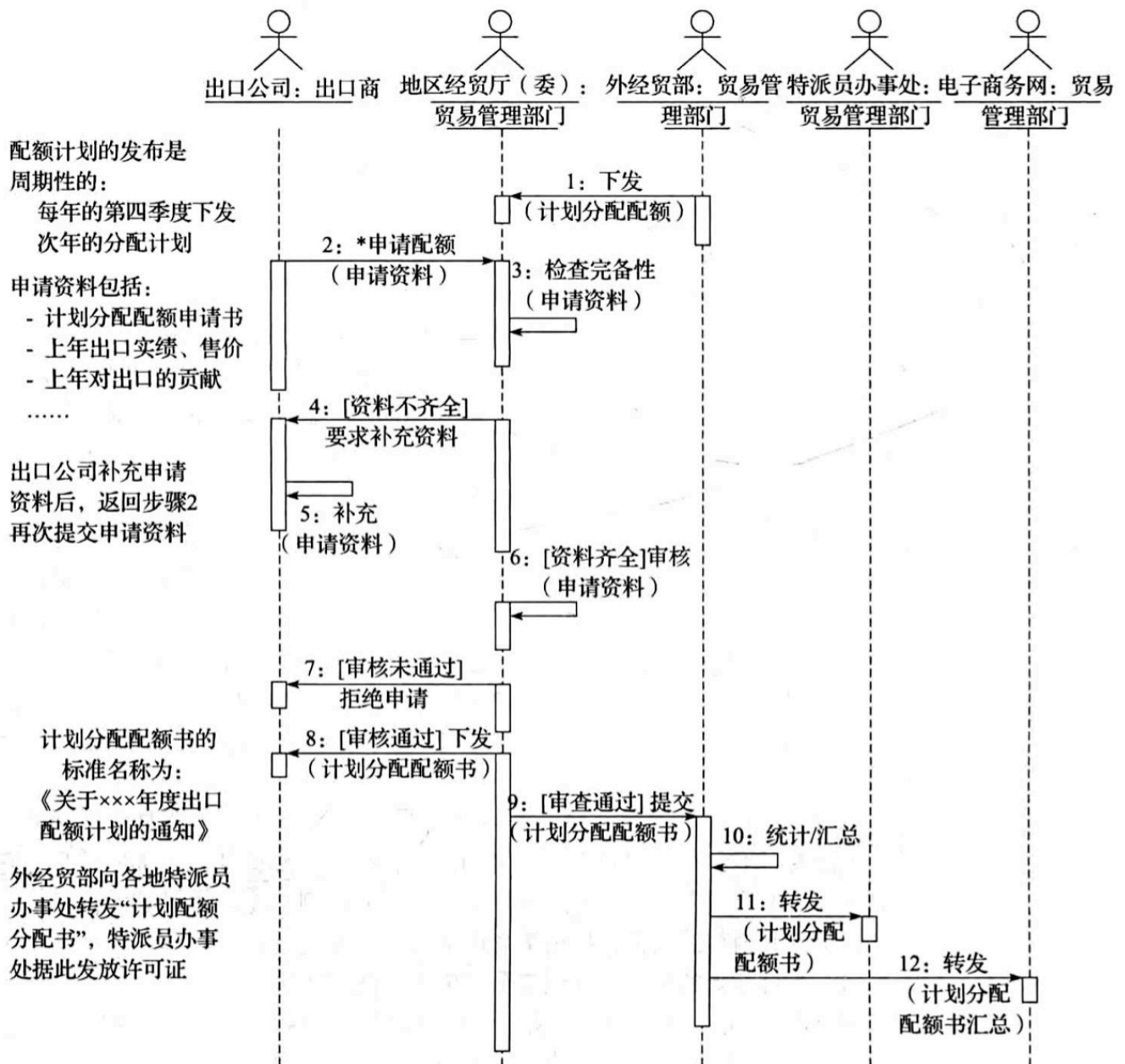
- 识别出系统的Actor
- 描述主要的Use case
- 实现用例视图
- 实现顺序视图，活动视图等

3. UML 需求视图

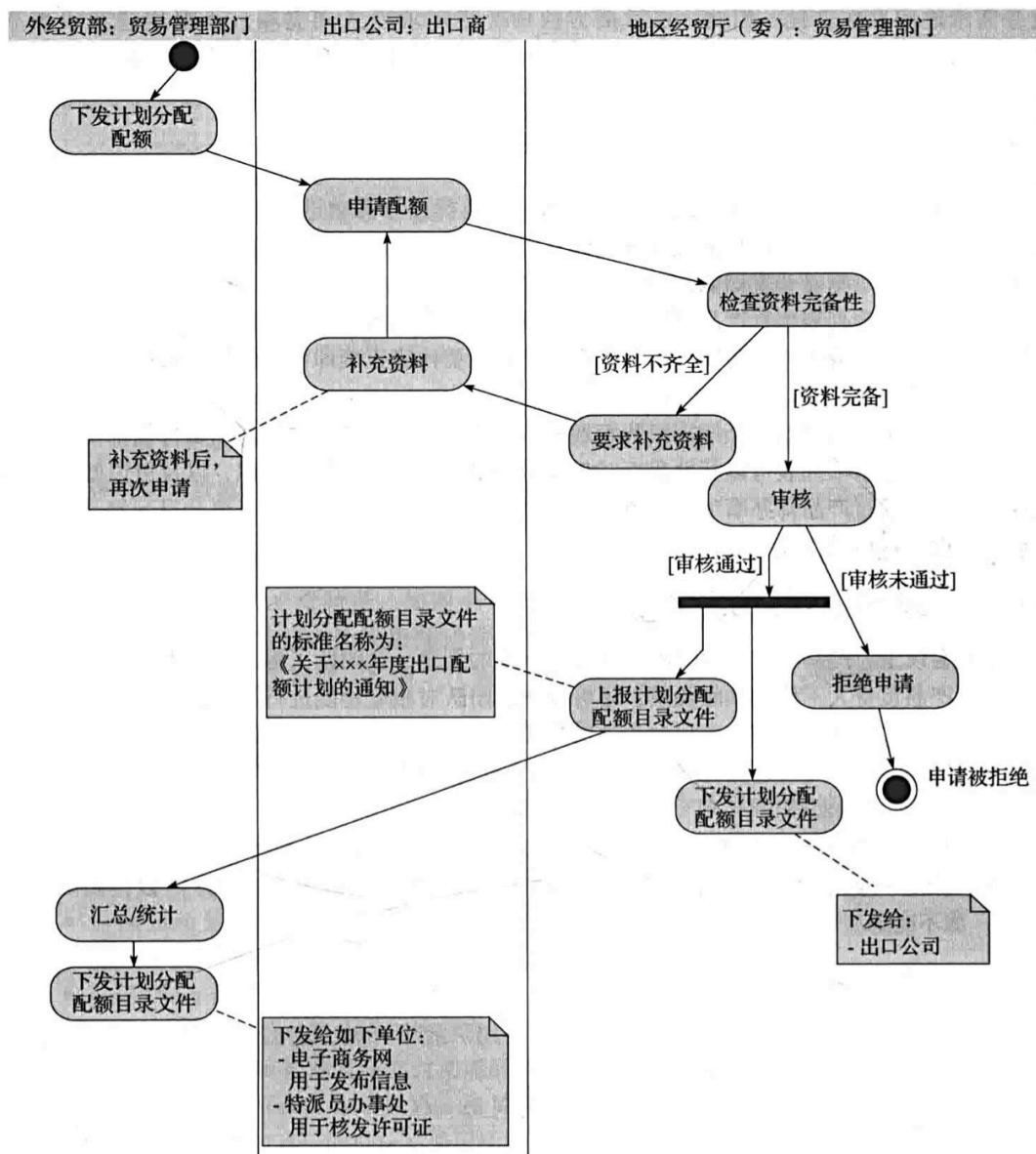
- 1) 用例视图



- 2) 顺序图



• 3) 活动图



3.4. 功能列表法

需求类别（功能/性能）	名称/标识	描述
特性 A	A. 1	
	
	A. n	
特性 B	B. 1	
	
	B. n	
特性 C	C. 1	
	
	C. n	

4. 敏捷项目需求分析

- 项目特征
 - 需求不断变化
 - 风险大、不确定性高
 - 项目开始时无法明确项目的范围
- 使用方法
 - 渐进明细
- 描述需求
 - 用户故事

4.1. 产品代办事项列表

- 从其中选择部分需求进行细化形成Sprint 订单
- 细化的过程就是编写用户故事的过程

4.2. 代办事项列表的细化

- 细化会议上完善用户故事

4.3. 用户故事

- 模版

```
As a <type of user>,  
I want < some goal >  
so that < some reason>.
```

- + 可以描述功能性和非功能性需求
- + 评价用户故事
- + 迭代