



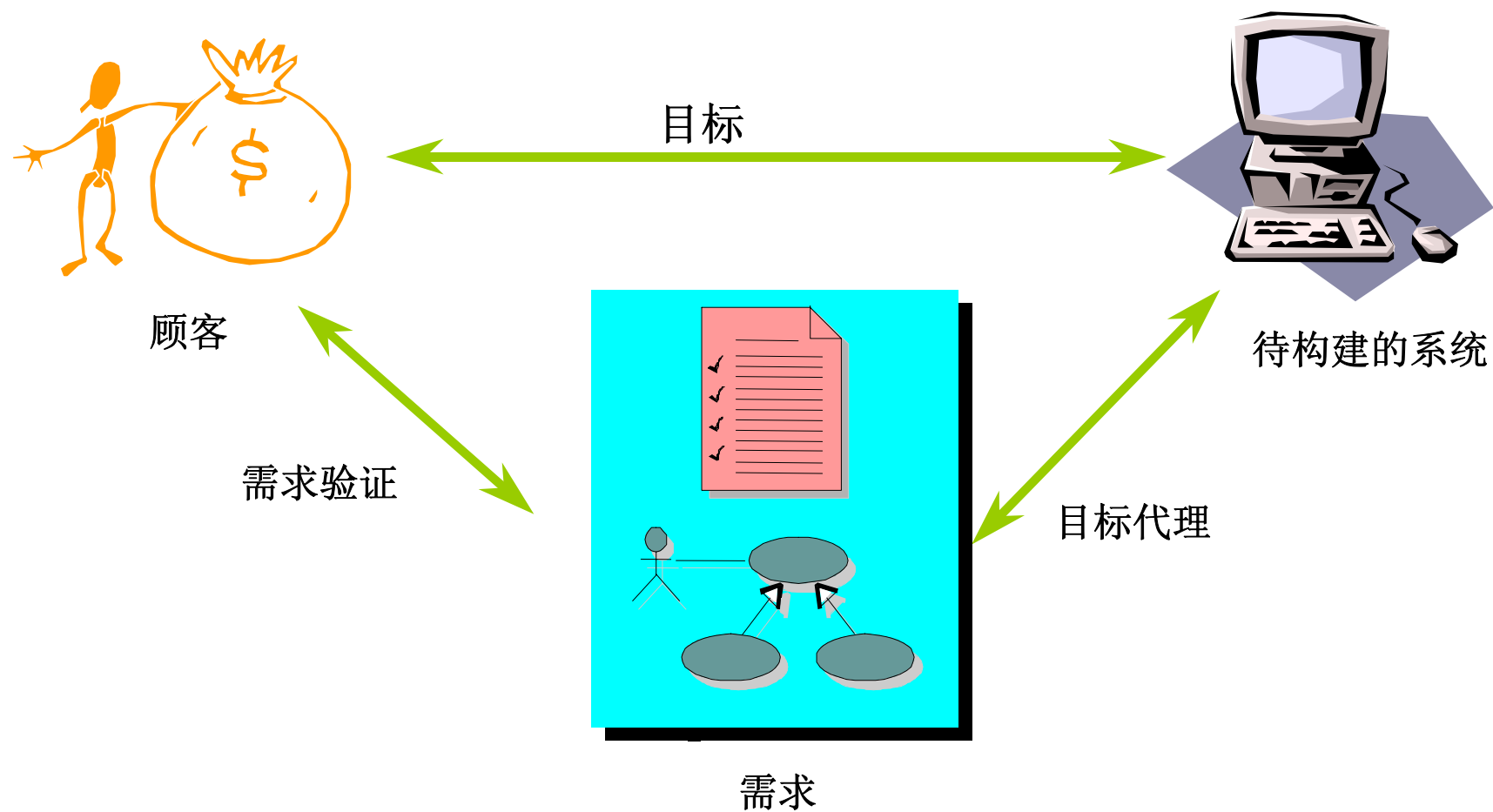
面向对象需求分析

需求规格说明书

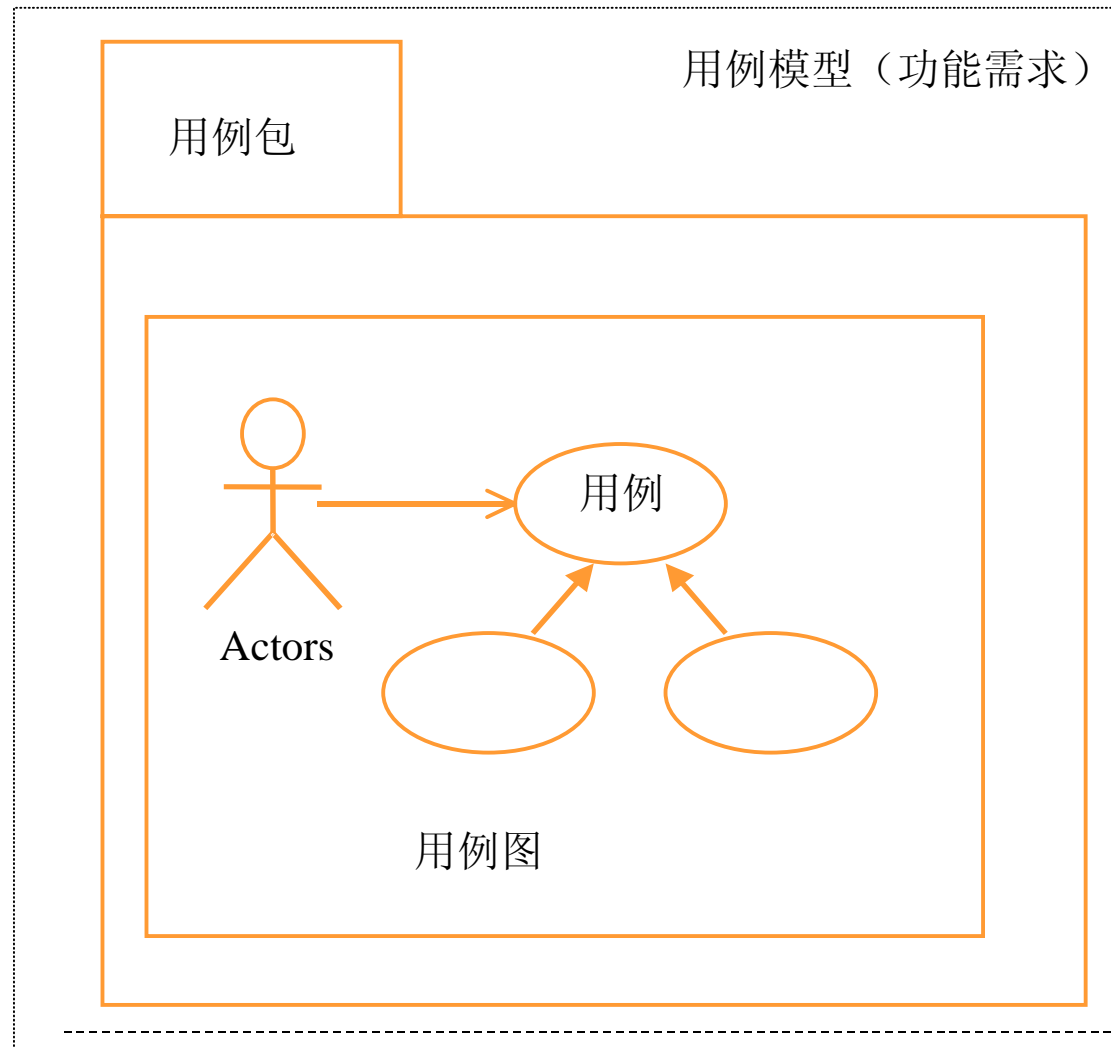


- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点

需求规格是一种契约



需求规格纵览



非功能需求



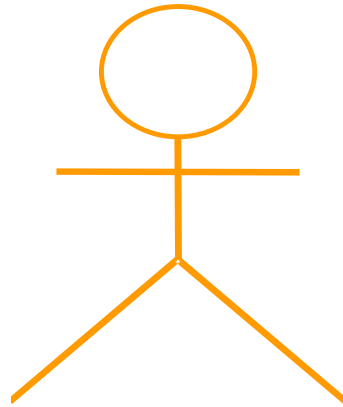
词汇表

需求规格说明书



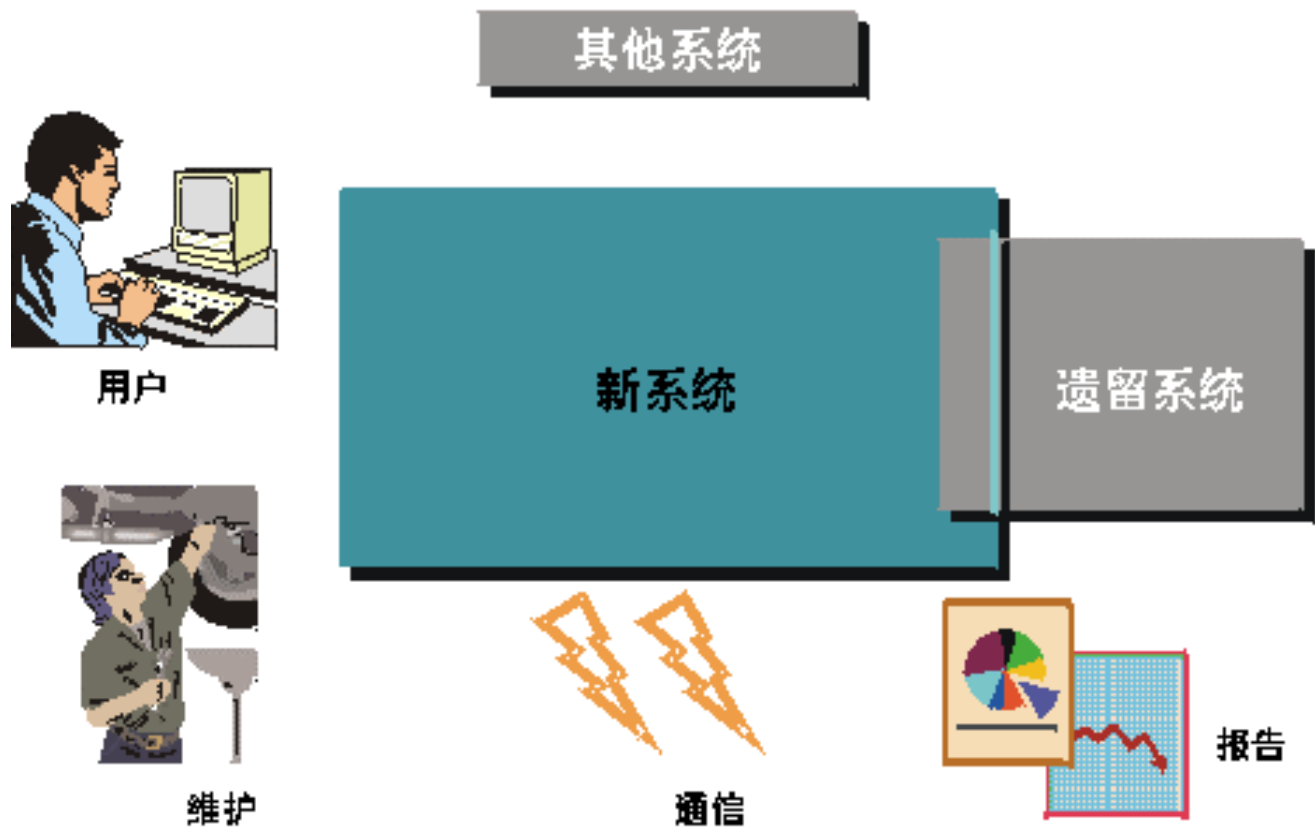
- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点

什么是Actor?

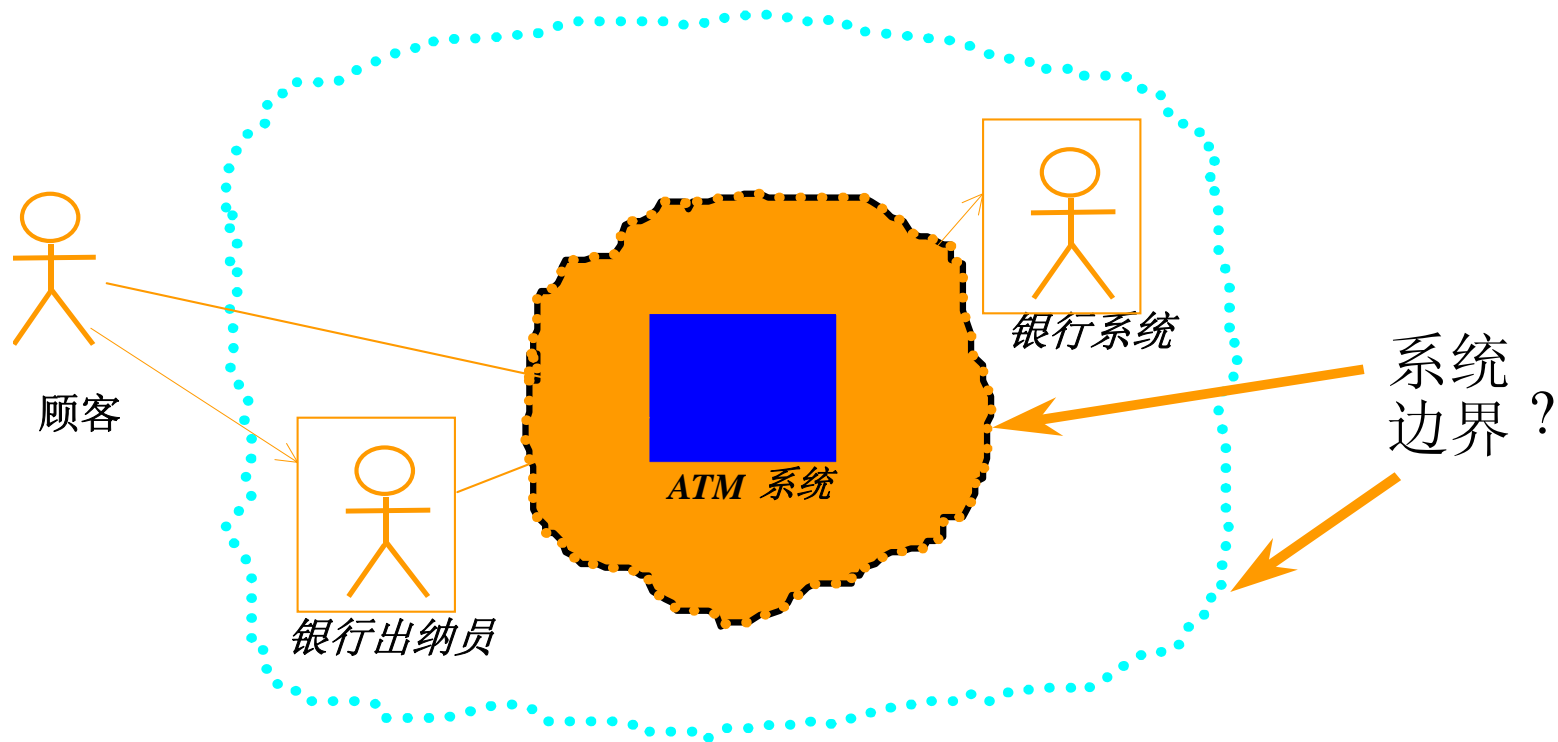


- **Actor**是一组紧密相关的、直接与系统实体发生作用的用户、系统等外部角色的统称。

系统的Actor有哪些？

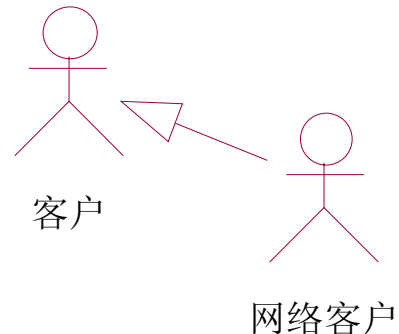


Actors 与系统的边界



查找主角意味着建立系统边界，这可以帮助您理解系统的目的和范围。只有那些直接与系统通信的人员才需要被当作主角。如果您将系统环境之外的角色也包括进来，那您就是在试图对使用系统的业务建模，而不是对系统本身建模。

Actor写作参考

名称	简要说明	关系
网络客户	<p>Actor所代表的对象。 简要描述其含义和特征：为什么需要Actor， Actor在系统中能获得哪些利益。</p> <p>例：网络客户是任何通过因特网访问“古典音乐”公司的电子商务系统，并进行浏览或订购商品的顾客。</p>	 <pre> graph LR A((客户)) B((网络客户)) B --> A </pre>

样例——Actor描述

双机备份系统的**Actor**:

- 1、系统管理员：双机异地容灾备份系统的管理者，要求有操作系统管理的相关经验。
- 2、应用进程：应用服务中的操作系统进程。

策略管理系统的**Actor**:

- 1、网管操作人员：创建任务，进行配置，以实现一些例行维护任务的无人管理。
- 2、应用程序员：使用策略管理系统提供的接口，注册自己提供的任务类型，并实现自己提供的任务的具体动作。

平台调试框架的**Actor**:

1. 调试用户--在系统运行期间，从人机交互界面接入的调试人员
2. 应用模块--应用此框架的具体模块，包括iMAP开发平台内部的各模块以及使用iMAP开发平台的产品应用模块

（以上Actor描述是否正确、完备了？）

样例——Actor描述

双机备份系统的Actor:

- 1、系统管理员：双机异地容灾备份系统的管理者，要求有操作系统管理的相关经验。
- 2、应用进程：应用服务中的操作系统进程，是双机备份系统控制启动的对象。

策略管理系统的Actor:

- 1、网管操作人员：创建任务，进行配置，以实现一些例行维护任务的无人管理。
- 2、应用程序员：使用策略管理系统提供的接口，注册自己提供的任务类型，并实现自己提供的任务的具体动作。

平台调试框架的Actor:

1. 调试用户--在系统运行期间，从人机交互界面接入，对系统进行调试的调试人员。
2. 应用模块--应用此框架的具体模块，包括iMAP开发平台内部的各模块以及使用iMAP开发平台的产品应用模块，响应、处理调试框架转发的调试命令。

（以上Actor描述是否完备了？）

总结——Actor

- Actor要从：(1)Actor的含义和特征；(2)为什么需要Actor，Actor在系统中能获得哪些利益这两方面来描述，通常容易忽略其中一方面。
- 软件平台的Actor要注意正确描述二次开发人员和二次开发应用模块。

需求规格说明书



- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点

包

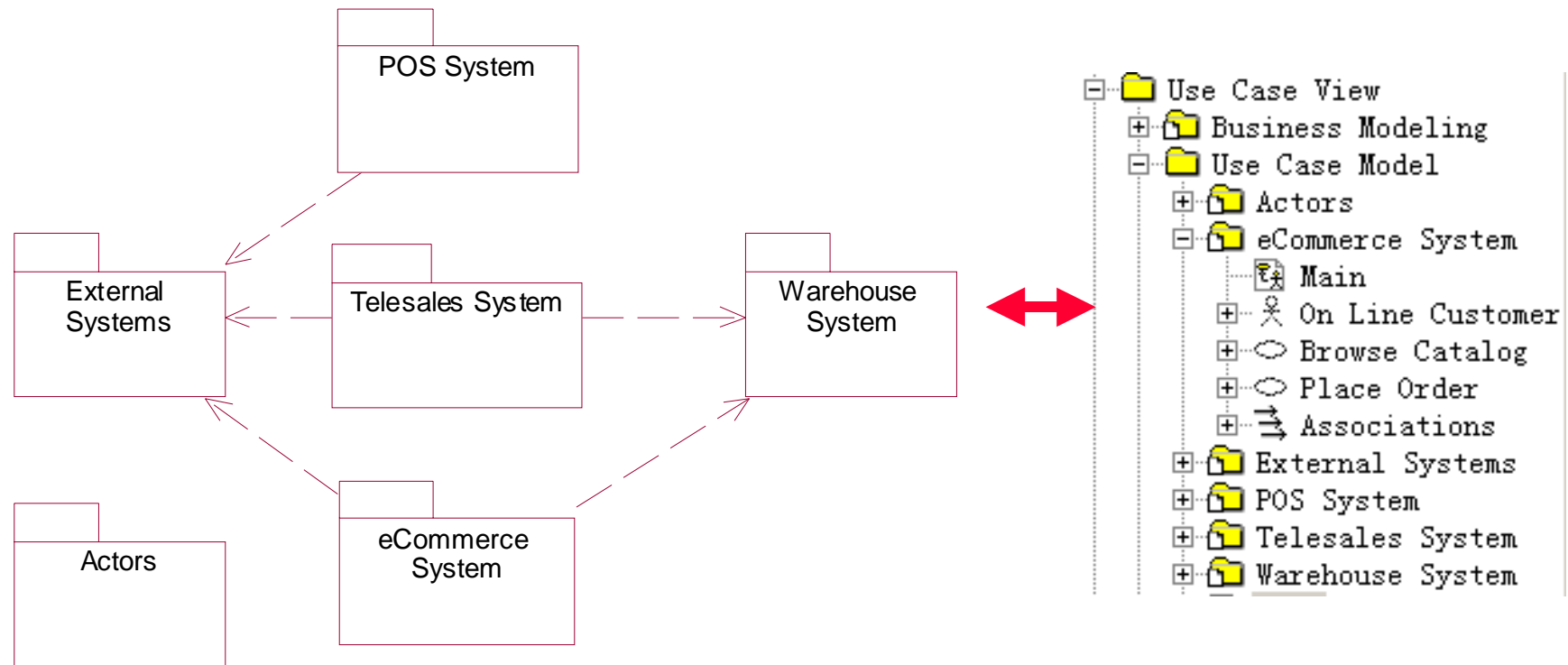
- 包是一种常用的将元素进行分组的机制。
- 包是一种能够容纳其他模型元素的模型元素。



OO 原则:
模块化

- 应用
 - 对模型进行组织
 - 配置管理的一个单位

例：包图与组织

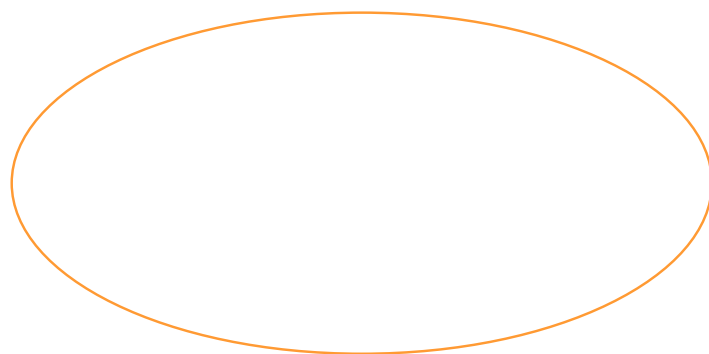


需求规格说明书

- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点



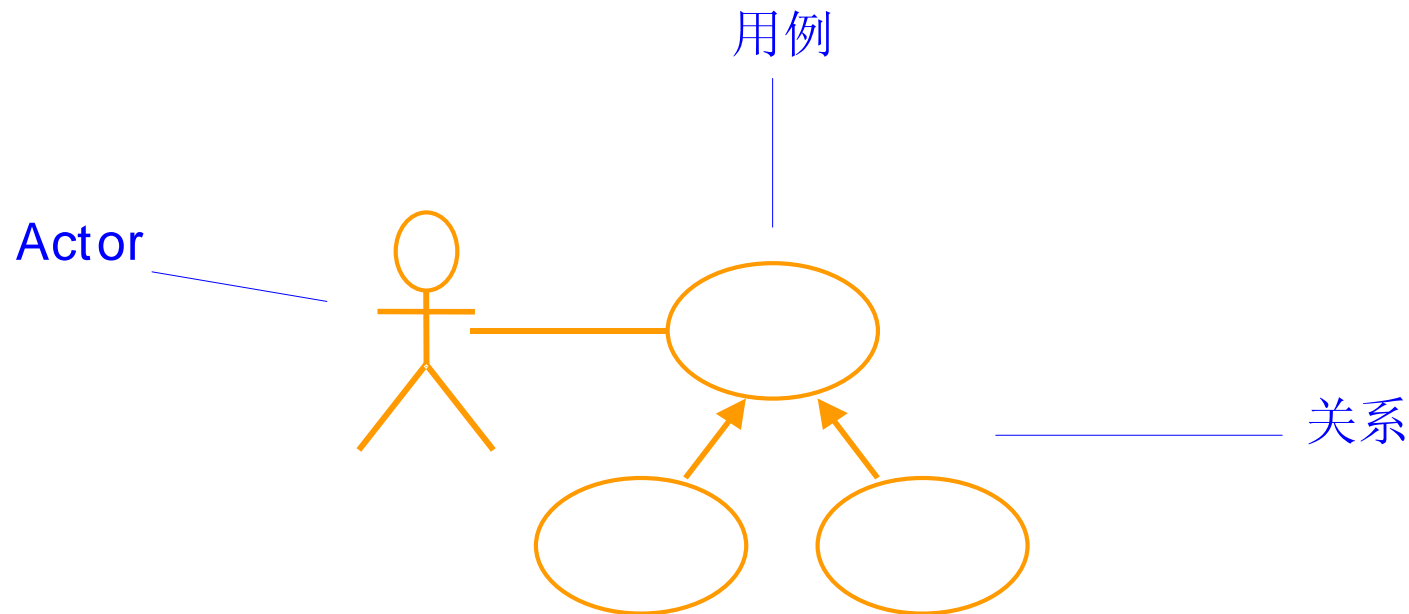
用例（Use Case）



用例（**Use Case**）是外部可见的一个系统功能单元，这些功能由系统、子系统或类等系统部件提供，并通过一系列系统部件与一个或多个**Actor**之间的交换消息所表达。

Use-Case 图

- 表示处于同一系统中的Actor和用例之间的关系图。

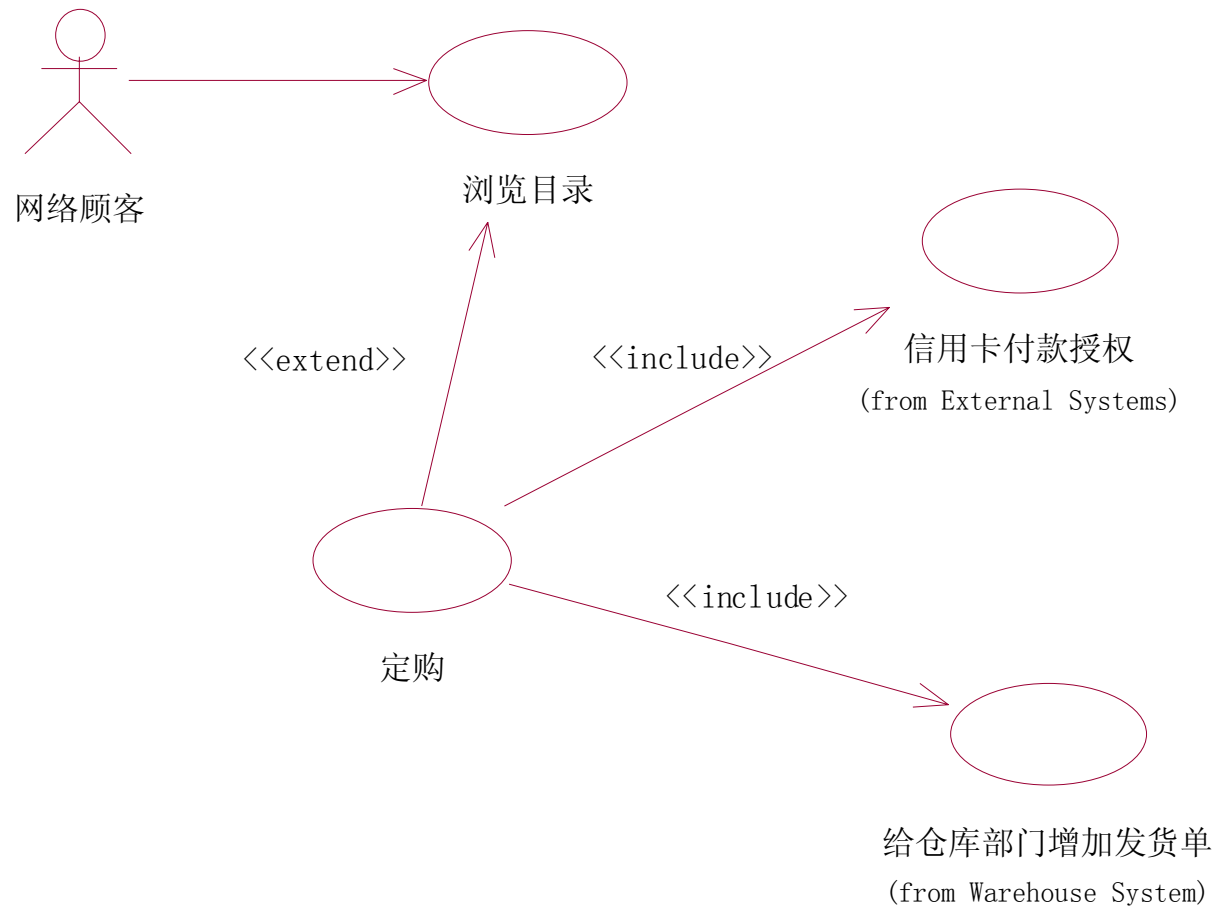


用例图的内容

- 哪些内容可以放在同一个用例图中？

- 属于同一个用例包的Actor。
- Actor及与之交互的所有用例。
- 处理相同信息的用例。
- 同一组Actor所使用的用例。
- 通常按一种顺序执行的用例。
- 属于同一个用例包的用例。
- 最重要的用例。这种用例图可作为模型的概要。
- 某一特定用例及其与Actor和其他用例的关系。

例：用例图



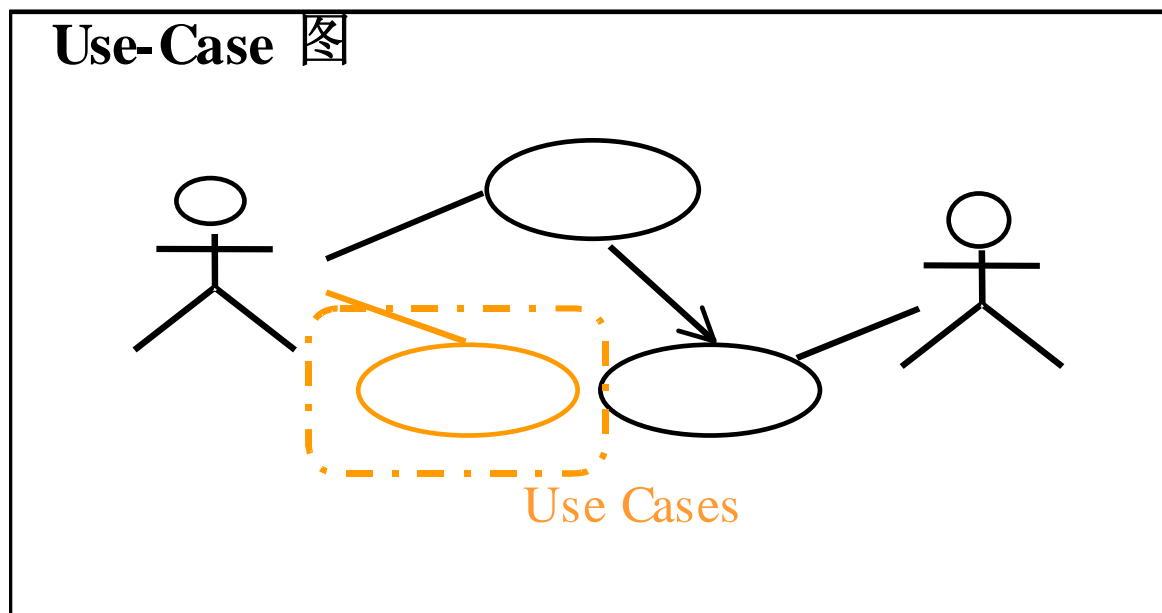
需求规格说明书

- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点



Use Case规格

- 名称
- 简要说明
- 前置条件
- 后置条件
- **Actors**
- 触发条件
- 基本事件流
- 备选事件流
- 特殊需求



Use Case规格范例

USE CASE 5 购买商品

简要说明 客户从我公司直接定购货物，并希望能够送货上门。

前置条件 无。

后置条件 成功：

客户获得商品，我们获得购物款

失败：

公司不运送货物，客户不付款。

Actor 客户，银行

触发条件 购物请求达到

基本事件流

- 1 客户提交购物申请
- 2 公司获取姓名，地址，商品名称，商品数量
- 3 公司为客户提供产品总价格，货期
- 4 客户签字确认
- 5 公司创建送货单

6 公司把货物和发票送给客户

7 客户付款

备选事件流

3a 如果公司该项商品数量不足

3a1. 要求客户修改定单

4a 如果客户直接用信用卡支付

4a1. 从信用卡扣款 (use case 44), step7不执行

7a 客户返回商品

7a1. 处理客户退货 (use case 105)

特殊需求

每个定单处理不超过5分钟

样例——简要说明

- 简要介绍该用例的目的、作用和背景。

例1：双机备份

Use Case名称：登录进入系统

简要说明：系统管理员在进行管理操作前必须登录进入系统。

改为：为了安全性目的，系统管理员在进行管理操作前必须登录进入系统，本用例提供登录进入系统的功能。

例2：调试框架

Use Case名称：设置变量的值

简要说明：用户可以使用此用例来设置当前模块的可以设置的变量的值，从而达到控制此模块运行行为的目的。不同模块具有不同的可以设置的变量。

例3：策略管理

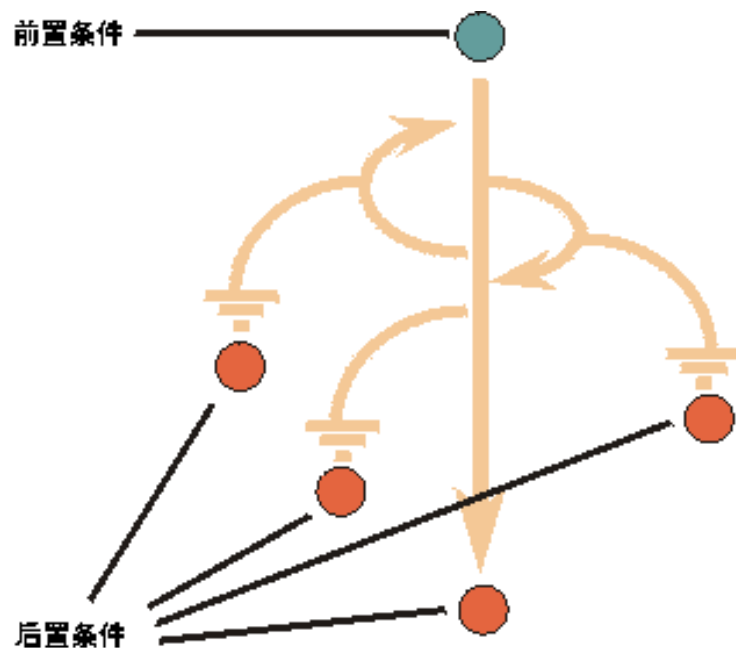
Use Case名称：查看任务的运行记录

简要说明：提供用户查看任务的运行记录的功能，包括目前存在的任务（当前任务）和已经删除的任务（历史任务）。

运行记录：是指该任务的每次运行的结果，如果一个任务已经运行了三次，则该任务有三条运行记录。

前置条件和后置条件

- 前置条件：开始用例前所必需的系统环境或状态。
- 后置条件：用例结束后系统可能具备的状态。



利用前置条件和后置条件的概念来阐明事件流如何开始和结束是一种非常有用的方法。然而，只有当用例的读者认为该方法可以增加价值时才应将其采用。



前置条件和后置条件写作要点

- ❑ 前置条件或后置条件所说明的状态应该是用户可以观察到的状态。
- ❑ 前置条件是对用例何时开始的约束。它并不是使用例开始的事件。
- ❑ 当同时使用后置条件和扩展关系时，应确保扩展用例不引入违反基本用例后置条件的分支流。
- ❑ 后置条件是一种对用例进行说明的有力工具。您可以首先定义用例应该实现什么，即后置条件。然后，可以说明如何达到这一条件（所需的事件流）。

！ 注意区分事件和状态

例：前置条件和后置条件

- 自动柜员机中“提取现金”用例的**前置条件**为：客户拥有一张个人专用卡，这张卡正好可以塞进读卡器，并且该卡已经分到一个 **PIN** 号，还向银行业务系统进行了登记。
- 自动柜员机中“提取现金”用例的**后置条件**为：当用例结束时，所有帐户和交易日志都已收支平衡，与银行业务系统的通信已重新初始化，并且银行卡已经返还给客户。

样例——前置条件和后置条件

样例1：策略管理

R.FUNC.SCHD.003 修改任务

简要说明

提供用户修改已创建任务的调度策略的功能。

前置条件

任务已经创建。

后置条件

成功后置条件

该任务的属性得到更新

失败后置条件

系统显示给用户具体的失败信息。

该任务属性回滚到修改前的状况。

样例——前置条件和后置条件

样例2：调试框架

3.2.4 R.FUNC.RTDBG.009 设置变量的值

1. 简要说明

用户可以使用此用例来设置当前模块的可以设置的变量的值，从而达到控制此模块运行行为的目的。不同模块具有不同的可以设置的变量。

2. 前置条件

用户已经从调试端口成功登录到被调试进程；

选择欲操作的模块为当前模块。

改为：用户已经选择了欲操作的模块为当前模块。

3. 后置条件

成功后置条件：

成功执行，处于等待接收下一条命令状态。

改为：当前模块的变量已改变，处于等待接收下一条命令状态。

失败后置条件：

提示错误信息，处于等待接收下一条命令状态。

改为：当前模块的变量未改变，处于等待接收下一条命令状态。（提示错误信息放到事件流中描述）

事件流---结构

□ 基本事件流

一条正常的，基本的路径。

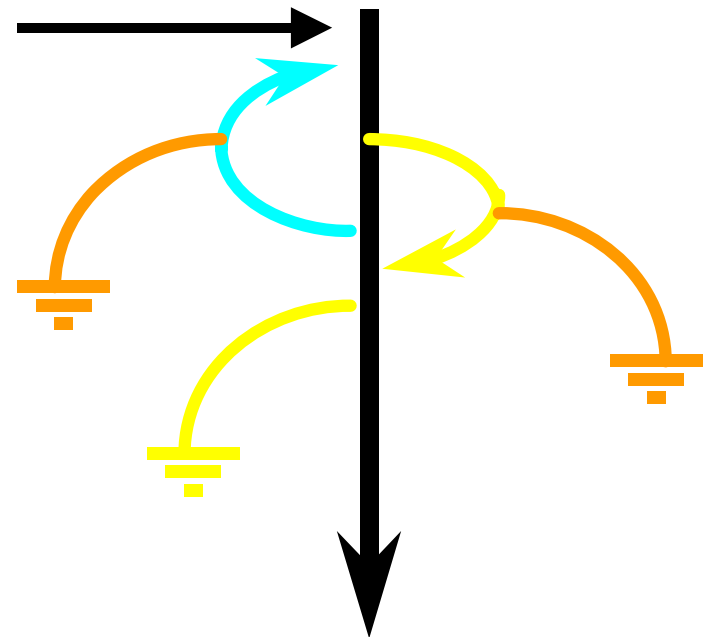
□ 备选事件流

➤ 正常行为的各种变形

➤ 临时情况

➤ 异常处理

基本事件流



事件流---内容

- 说明用例如何开始和结束
- 说明在Actor和用例之间交换的是什么数据
- 不要详细描述用户界面，除非要理解系统行为就必须详细了解用户界面。例如，如果事先知道应用程序将基于 Web，那么最好使用一组有限的 Web 专用术语。否则，您的读者就可能认为用例本身过于抽象。您可以在术语中包括这些词语：“导航”、“浏览”、“超链接”、“页”、“提交”和“浏览器”。若已实现低保真，则可将原型图附上，此描述会更加清晰。
- 说明事件流，而不只是功能。为了做到这一点，每个动作都应从“当主角...时”开始
- 只说明属于该用例的事件，而不是发生在其他用例中或系统外部的的事件
- 避免不明确的术语，如“例如”、“等等”和“信息”
- 详细说明事件流，即回答所有包含“什么”的问题。请记住，测试设计人员将使用此文本来确定测试用例。
- 在文字描述的基础上，也可以采用流程图来展示事件流。

事件流---风格

风格一：步骤型



1.1 用例开始

当主角操作员通知系统
创建一个评测订单时，
用例开始...

1.2 配置测评订单

系统让操作员主角选择
要评测的网络元素，然后...

1.3 初始化订单

操作员通知系统初始化
评测订单。系统将记录...

1.4. 用例结束

系统向该操作员确认已对
测评订单进行了初始化...

风格二：小说型

为了从网络元素中收集评测
数据，订货人可以创建订单。

系统将给该订单分配一个唯
一的名称，并指定评测持续
时间和评测开始时间和评测
重复频率的默认值。订货人
将能够编辑这些默认值。

订货人必须进一步指定适用
的评测功能、网络元素和评
测对象。订货人还可以在订
单上添加个人注释。

定义了必要的信息之后，单
并使用已定义的属性...

风格三：程序型

'管理订单'（用户身份）

REPEAT

<='显示管理订单菜单'

IF (=> '创建订单'（评测功能，
网络元素，评测对象）) **THEN**
系统查找唯一的名称、评测开
始时间

和持续时间的默认值。

<= '显示订单'（默认属性）

REPEAT

=> '编辑订单'（要更改的属性，
属性的新值）

<= '屏幕刷新'（新属性）

UNTIL（定义了所有的属性）

REPEAT...

样例——基本事件流和备选事件流

样例1：双机备份

系统管理员修改密码的用例事件流

Actor=系统管理员

Description 基本事件流描述

Step 步骤

1. 系统管理员点击管理控制台GUI上的“登录”按钮；
2. 控制台GUI弹出一个登录对话框；
3. 系统管理员在登录对话框中点击“修改密码”按钮；
4. 控制台GUI弹出修改密码对话框（包括三个输入栏：旧密码，新密码、确认新密码）
5. 系统管理员输入旧密码、新密码和确认新密码，点击“确认”按钮，如果此时系统管理员点击“取消”按钮，修改密码对话框会关闭，回到登录对话框；
6. GUI验证旧密码以及新密码和确认新密码是否一致，然后关闭修改密码对话框；
7. 系统管理员在登录对话框中输入新密码登录（过程见用例“登录进入系统”）

Extensions 备选事件流

Step 步骤

- 5a、输入的旧密码不对或者新密码和确认新密码不一致；
- 5a1、系统弹出出错对话框，向系统管理员提示出错。

增加：5b、系统管理员点击“取消”按钮，修改密码对话框关闭，回到步骤3。

样例——基本事件流和备选事件流

样例2：调试框架

应用模块调用平台接口写日志的用例事件流

4. Actors

应用模块

5. Trigger 触发条件

由应用模块通过API调用触发。

6. Description 基本事件流描述

Step 步骤

- 1、由应用模块调用，本用例开始；
- 2、应用模块传入信息级别、信息内容；
- 3、如果当前模块此级别信息屏幕输出开关为打开，那么在屏幕上显示此信息，然后继续；~~否则继续；~~改为：否则不显示，然后继续；
- 4、如果当前模块此级别信息日志记录开关为打开，那么在日志文件中记录此信息，然后继续；~~否则继续；~~改为：否则不记录，然后继续；

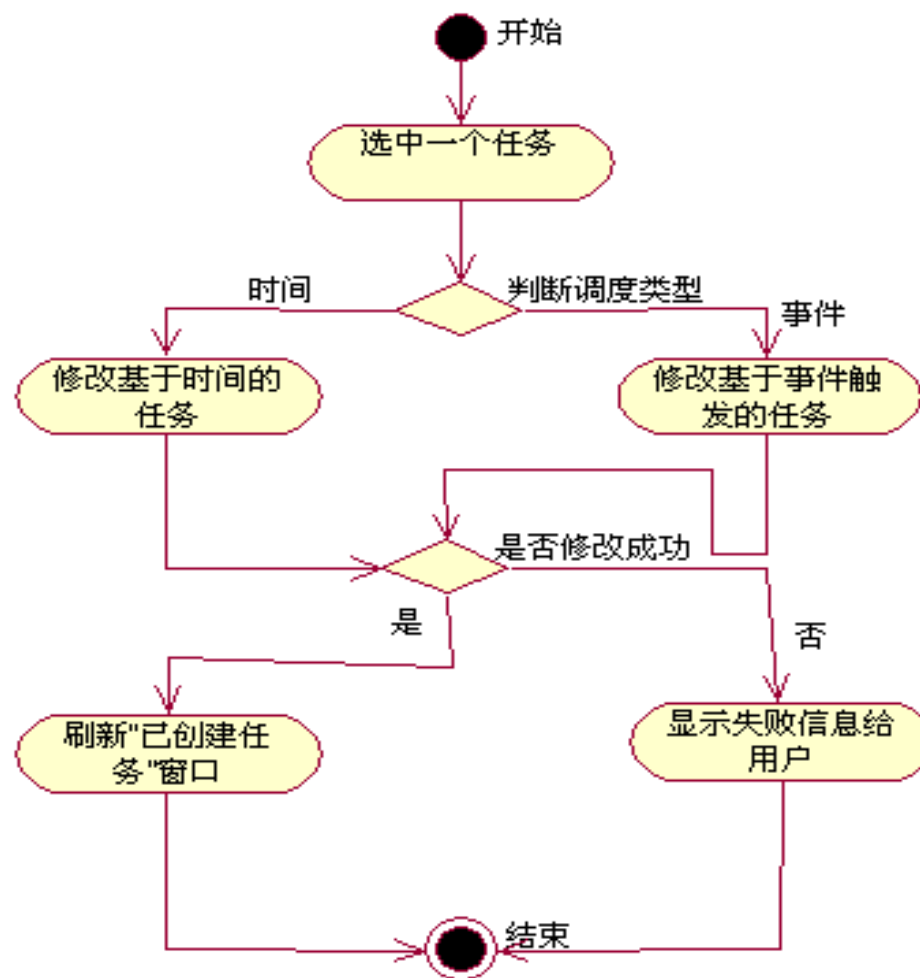
5、本用例结束

7. Extensions 备选事件流

无

样例——基本事件流和备选事件流

样例3：策略管理
用流程图描述事件流
修改任务用例



用例规格—其他

备选事件流：

- 1 客户打电话要求定购商品。
- 2 公司记录客户的姓名、地址、电话、需要的商品。
- 3 公司为客户提供商品的价格、发货日期信息。

3a 某项商品缺货：

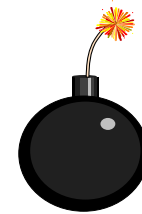
3a1. 要求客户重新考虑订单

触发条件：启动**use case**的动作或事件，如时间事件。

例：“定购商品”用例的触发条件是“订单到达”。

特殊需求：用例的非功能需求。

例：“定购商品”用例要求每个订单的处理时间不超过**5**分钟。



样例——特殊需求

样例：策略管理，修改任务用例的特殊需求

能够修改和不能修改的属性项标记如下

1、基于时间的任务的属性项：

任务名 （能修改）

任务操作类型（周期 或 定时）

开始时间 （能修改）

结束时间 （能修改）

调度执行设置参数 （能修改）

具体任务参数 （能修改）

备注（用户自己添加的用于帮助或提示性文字） （能修改）

任务触发类型 （不能修改）

2、基于事件的任务应当包括如下属性项：

任务名 （能修改）

调度执行设置参数 （能修改）

具体任务参数 （能修改）

目前状态（活动、挂起等） （能修改）

备注（用户自己添加的用于帮助或提示性文字） （能修改）

任务触发类型 （不能修改）

需求规格说明书

- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点



非功能需求

- 性能需求
- 接口需求
- 软件质量特性
- 其他需求

性能需求

静态的量化需求

- 支持的终端数目
- 支持的并发用户数目
- 需处理的文件和记录的数目
- 表和文件的大小

动态的量化需求

- 可包括正常和满负荷业务量条件下，某时间段（如一小时）内处理的事务和任务的数目以及数据量。

举例：

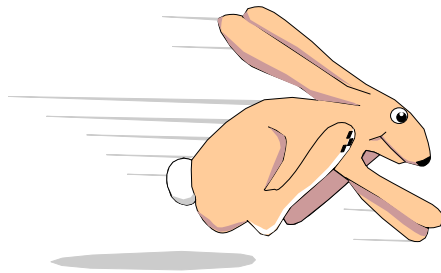
- 交易能很快完成，操作员不必等待。
- **95%的事务应在1秒内被处理。**



性能需求 & 用例特殊需求

性能需求是从系统角度看待软件和人机交互的静态以及动态量化需求。
而用例特殊需求是只关心用例本身的量化需求。

T2000必须支持最多8个终端。
--- 性能需求



从设备产生告警，到能够通过网管听到告警的声音和看到颜色提示的时间不能超过5秒。
--- 告警上报用例特殊需求

样例——性能需求

样例1：调试框架

1、支持同时使用终端数： ≤ 5 ，但是一个被调试模块同时只能有一个调试用户。

2、命令处理速度：人机交互满足即可，尽量不要占用太多系统资源来满足调试响应性能。

2、改为：命令响应时间不得超过5秒，占用系统内存不得超过5%，占用CPU时间不得超过5%。

样例2：策略管理

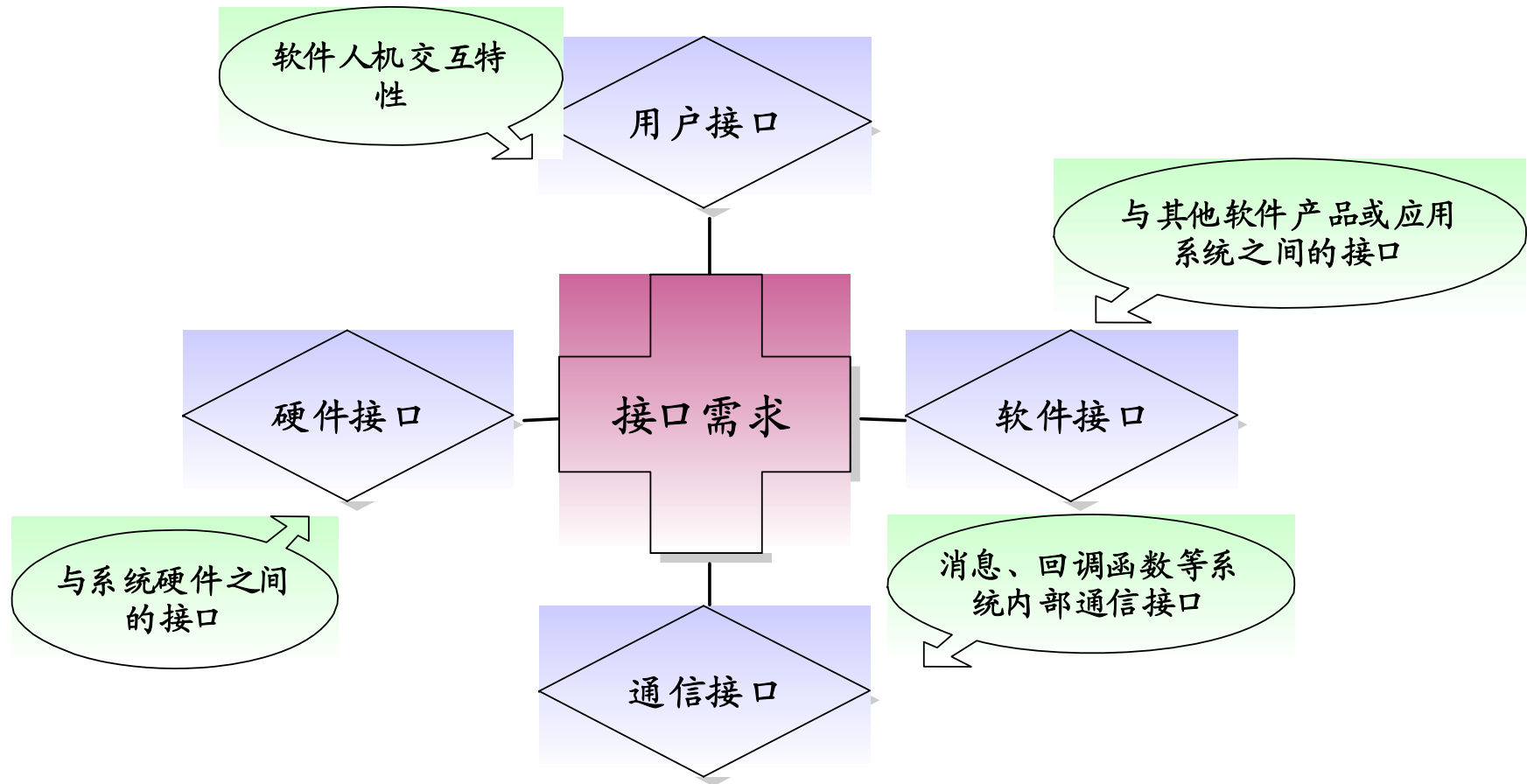
1、需要支持多终端对策略管理的并发操作能力

2、能支持100个任务同时运行

样例3：双机备份

1、切换间隔可以配置，但是不能小于6分钟。

接口需求



接口需求

例：（用户接口）如果系统用户通过一个显示终端进行操作，应描述要求的屏幕格式、页面布局以及报告或菜单的内容等。



样例——用户接口需求

样例：策略管理查看任务运行记录的用户接口
界面示意图

任务名称	完成时间	操作结果	操作结果描述
告警自动确认	2003-01-09 15:16...	成功	成功
告警自动确认	2003-01-10 15:16...	失败	找不到通讯端口

选中一行记录，点鼠标右键会出现如下菜单：“查看详细信息”，点击查看详细信息，会弹出具体的详细信息显示对话框。

界面元素说明

起始时间和结束时间是过滤条件，是表示查看这个时间段完成的运行记录；

任务名称过滤出所输入任务名称的运行记录，支持模糊查找，但不支持通配符；

如果任务名称为空，则过滤出所有在起始时间到结束时间段内的记录。

接口需求（续）

例2、（软件接口）

名称	版本	来源	目的	接口定义
TCL解析器	8.3.2	Share 代码	完成语法解析功能	详见XXX文档
MD5库	1.0	XXX项目组	实现MD5采样算法	详见XXX文档

例3、（通信接口）

本系统通过**FEP**和银行之间使用标准**8583**协议进行通信

样例——软件接口需求

样例，策略管理向其他**iMap**应用发送执行任务命令的接口

1、目的

策略管理系统本身并不执行任务的具体动作，它是按照配置好的调度策略触发任务的执行，本接口的功能就是向其他**iMap**应用发送命令，触发任务执行。

2、接口描述

INTERFACE BEGIN

JobID		: 整数
CmdCode	: 整数	
RecieverProcID		: 整数
RecieverDeskID		: 整数
RecieverModule		: 整数
RecieverCmdHandle		: 整数
TaskPara	: 字符串	
EventPara		: 字符串

INTERFACE END

JobID: 任务的编号，整个策略管理系统唯一。

TaskPara: 是client端用户设置的任务参数组成的一个字符串，组装和拆卸都由应用模块自己进行；

EventPara: 对于事件触发的任务，该项即为事件的**EventPara**，对于时间触发的任务，该项为**NULL**。

软件质量特性

系统的非功能需求，产品特性的一部分。

有效性	描述那些确保整个系统达到预定有效级别的因素（例如：恢复、重起、检查点等）。
安全性	防止软件遭到意外或恶意的侵入、使用、修改、破坏或泄密的需求
可维护性	此处主要是描述本软件对于后期维护上的一些特殊要求，主要是针对以后项目进入维护阶段时的一些考虑。
可移植性	度量把一个软件从一种运行环境转移到另一种运行环境中所花费的工作量
可靠性	软件无故障执行一段时间的概率, 包括容错性、可恢复性。
易用性	易懂、易学、易操作
可测性	描述系统能及时准确地确定其工作状态（可工作、不可工作、工作性能下降）并隔离其内部故障的一种设计特性。

样例——软件质量特性

样例1：策略管理

可靠性：

能够抵御其他应用的有意或无意的错误信息的攻击，不会coredown

当策略管理系统的某个或某些任务的调度和执行出现错误时，不会影响其他任务的调度和执行。

当出现故障时，系统重起能够恢复到上一次正常运行的状态。

样例2：双机备份

可靠性：

在出现软件错误的时候仍然能够打印出错误原因。

易用性：

易懂性：界面安排合理，描述信息清晰易懂。（难以验证）

易学性：用户学习操作不应该超过一个小时。

样例3：调试框架

可靠性

使用此动态调试框架应该保证应用模块的稳定运行，不应由此框架引入系统的不稳定性。
（难以验证）

易用性

提供友好的帮助信息，使得用户的操作较为简单、方便，尽量达到不需要查看其他文档。
（难以验证）

软件质量特性

软件质量特性主要是通过产品工程规范的方式进行统一定义，如：

- 可靠性规范
- 安全性规范
- 可移植性规范
- 可维护性规范
- 可测试性规范
-

需求规格说明书

- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点



词汇表

1Introduction 简介

1.1Purpose 目的

1.2Scope 范围

2Description概述

3Definition 定义

3.1<Term1 术语1>

建议词汇表作为单独的**CI**纳入配置管理，**SRS**文档中列出它的位置和文件名即可。

词汇表在需求分析及其规格描述中起到了统一概念的重要作用。

需求规格说明书

- 介绍
- Actor
- 包
- Use-Case 图
- Use-Case
- 非功能需求
- 词汇表
- 检查点



Use-Case的检查点

- ❑ 每个具体用例是否至少涉及一个Actor?
- ❑ 用例之间是否相互独立? 如果两个用例总是以同样的顺序被激活, 您可能需要将它们合并为一个用例。
- ❑ 用例之间是否避免存在非常相似的行为或事件流? 如果存在 - 而且您还希望它们的行为在将来也相似 - 您应该将它们合并为一个用例。这便于在将来引入变更。
- ❑ 是否明确定义了用例之间的包含关系, 如事件流的一部分被构建为另一个用例的模型, 应该让新用例使用旧的事件流; 或事件流的某一部分成为另一个用例的组成部分, 应该提取该分支流并让上述的用例使用它。
- ❑ 是否避免将一个用例的事件流插入另一个用例的事件流中? 如果是, 则利用与另一个用例的扩展关系来建立此模型。
- ❑ 用例的名称是否具有唯一性、直观性和自解释性, 这样以后就不至于将它们混淆?
- ❑ 是否明确说明用例的事件流开始及结束的方式和时间?
- ❑ 是否避免有过于复杂的用例? 如果要使用例模型易于理解, 您最好将复杂用例进行分解。
- ❑ 是否清楚地说明了用例与Actor之间的交互信息?

Actor的检查点

- ❑ 是否已经对系统环境中的所有角色都进行了说明和建模？
- ❑ 系统中是否有多个**Actor**担任相似的角色？如果有，可能需要将他们合并到一个主角中。
- ❑ 是否避免在一个用例中两个**Actor**都能担任同一角色？如果有，可能需要利用泛化关系来为他们的共享行为建立模型。
- ❑ 特定的**Actor**是否避免以几种（完全不同的）方式或目的使用系统？如果是这样，您也许应该有多个**Actor**。

包和用例图的检查点

- ❑ 是否已经标识出用例模型的所有用例？这些用例共同说明所有的必要行为
- ❑ 用例模型是否没有多余的行为？所有用例应可以被追溯到一个功能需求。
- ❑ 通过用例模型是否可以很容易地理解系统的行为？
- ❑ 包之间的交叉关系是否被尽量减少或去除？这便于避免各种冲突的发生。
- ❑ 包的划分是否便于理解、合乎情理？

案例——Actor

REQ-TC-SM-002 输入经纬度坐标

简要说明

用户在创建网元、子网或者修改网元、子网时可以输入经纬度坐标

前置条件

无

后置条件

成功后置条件

对象在指定的经纬度坐标处显示。

失败后置条件

无

Actors

用户

触发条件

无

基本事件流描述

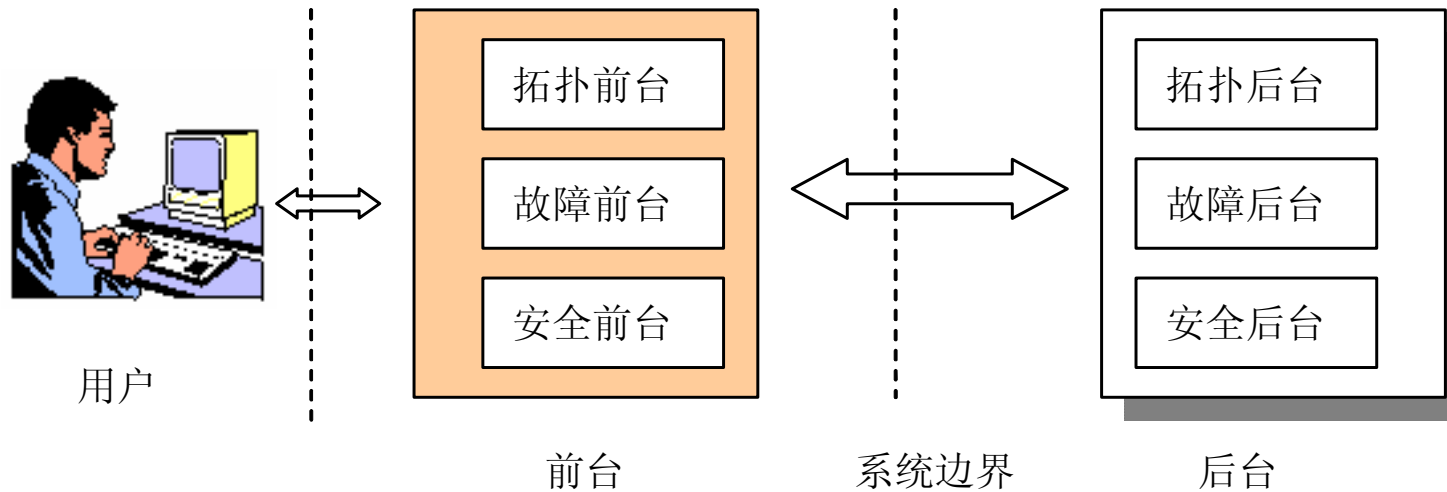
Step 步骤

- 1、用户执行创建、修改网元或者子网的操作
- 2、用户输入对象的经纬度坐标
- 3、下发命令到后台
- 4、拓扑前台收到事件，在指定的经纬度坐标处显示 对象

备选事件流

- 2a、如果输入的经纬度无效，则提示“请重新输入”
- 3a、如果命令发送失败，则前台保持现状

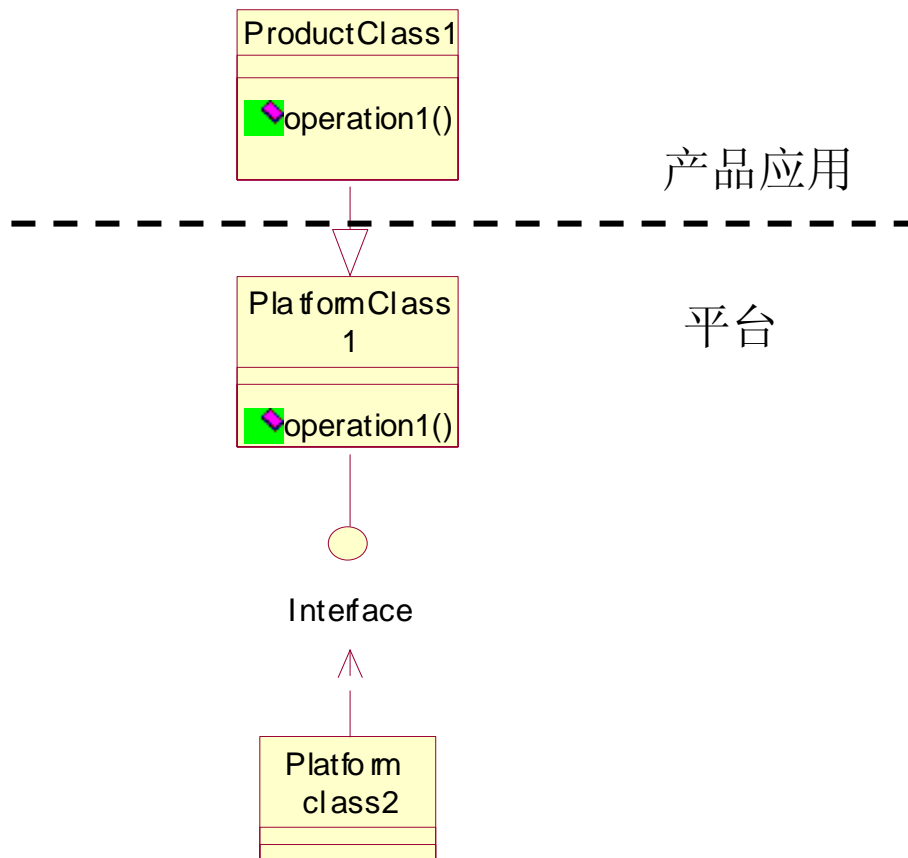
案例——网管系统前后台的Actor



- 在前台的需求分析当中，应该把用户和后台当作Actor；
- 在后台的需求分析中，应该把前台当作Actor；
- 但如果是前后台合在一起当作一个系统来分析，则只有用户是Actor。

案例——平台二次开发接口的需求描述

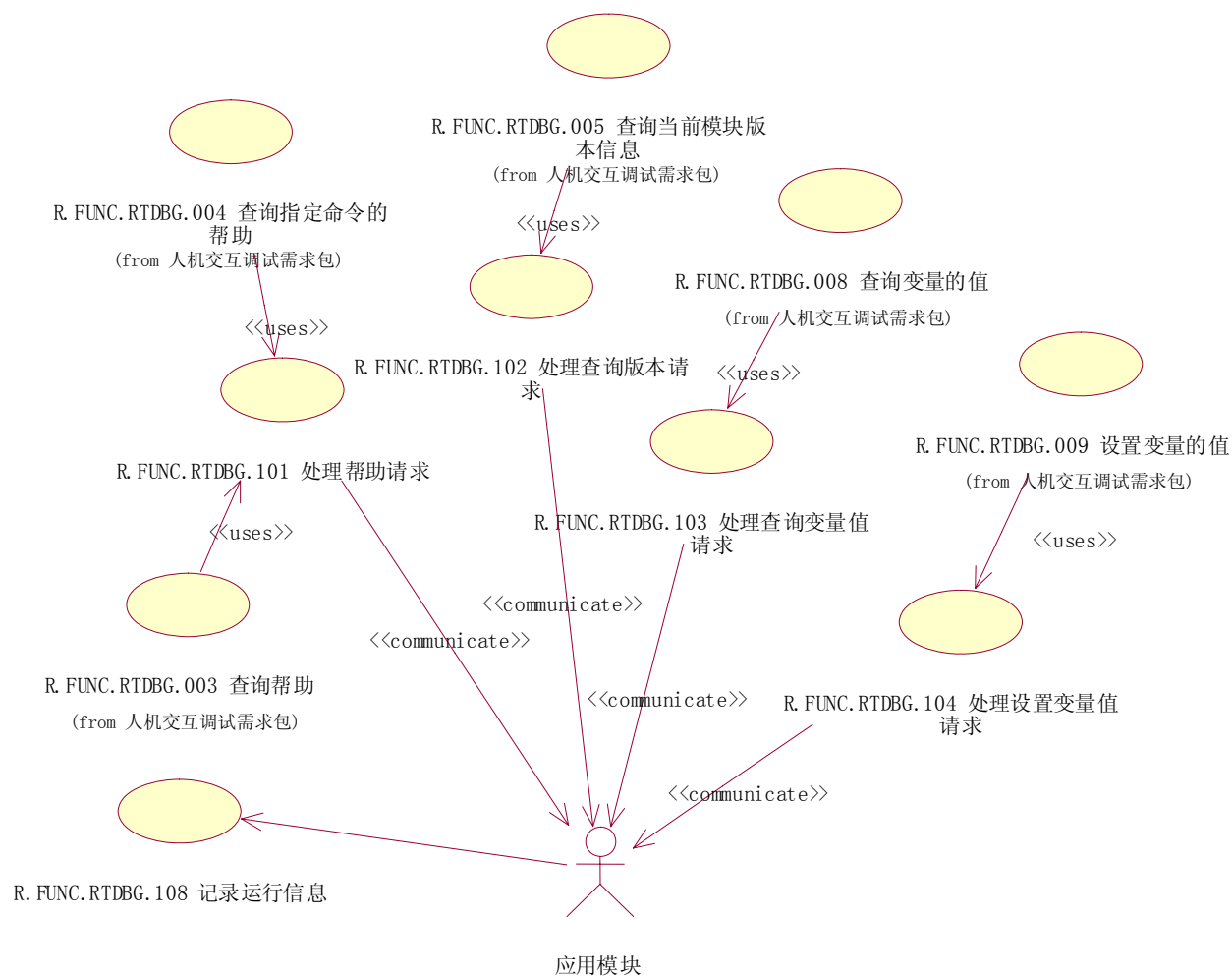
问题：平台的二次开发接口在**SRS**中如何描述？二次开发的**actor**如何定义？
经过分析，目前比较常见的平台二次开发接口是类继承方式：



案例——平台二次开发接口的需求描述

1. 可以在接口处描述需求，在**Use case**描述中，产品应用模块是**Actor**，平台是目标系统，应用模块是以平台的抽象类（或者代理）的形式出现，在此例中，**PlatformClass1**代表了应用模块。
2. 在**Use case**的事件流中，基于平台提供给应用模块的一个完整的接口功能，描述平台与应用模块的接口交互过程。
3. 所有平台二次开发接口都应该在**SRS**的**Use case**特殊需求或者集中在软件接口需求中说明具体的接口参数。
4. 在**Use case**特殊需求中，应说明抽象类的属性和方法的原型，作为此需求实现的约束。
5. 不要把二次开发的开发人员作为**Actor**、把编程活动作为事件流。

样例——平台二次开发接口的需求描述



样例——平台二次开发接口的需求描述

3.3.3 R.FUNC.RTDBG.102 处理查询版本请求

1. 简要说明

应用模块需要实现本模块的版本查询，即用户输入命令查询当前模块的版本信息时，通知应用模块需要处理此请求。

2. 前置条件

应用模块正确使用动态调试框架。

3. 后置条件

成功后置条件：

返回成功。

失败后置条件：

返回失败。

4. Actors

应用模块

5. 触发条件

由用例"R.FUNC.RTDBG.005 查询当前模块版本信息"触发。

6. 基本事件流描述

Step 步骤

1、由"R.FUNC.RTDBG.005 查询当前模块版本信息"用例使用此用例，本用例开始；

2、向应用模块查询版本信息；

3、如果查询成功，显示版本信息，否则执行备选事件流3a；

4、提示操作符，等待用户进一步操作，本用例成功结束。

7. 备选事件流

Step 步骤

3a 查询信息失败

3a1 输出错误提示，转向执行基本事件流4；



样例——平台二次开发接口的需求描述

用例：处理查询版本请求

.....

特殊需求：

本功能是平台提供给应用模块的二次开发接口，通过平台提供一个基类，应用模块继承基类并重载接口方法的方式来实现，基类的原型如下：

```
class PlatformClass1()
{
    public:
        operation1(); //查询版本信息接口方法
}
```

样例——平台二次开发接口的需求描述

REQ-TC-SM-003 二次开发者扩展查询条件

1、简要说明

公共拓扑前台不可能实现所有的查询条件，但是产品需要根据特殊的条件来进行查询，而公共拓扑没有这些信息，因此需要公共拓扑提供可以扩展的手段以便产品可以提供自己需要的查询条件。

2、前置条件

无

3、后置条件

成功后置条件

无

失败后置条件

无

4、Actors

二次开发者

5、触发条件

无

6、基本事件流描述

步骤

1、二次开发者根据公共拓扑提供的扩展接口提供自己的编程实现

7、备选事件流

无



Q&A

