



华章原创精品

系统分析师 UML用例实战

邱郁惠 编著

- 知名UML技术专家邱郁惠老师新作
- 畅销书《系统分析师UML实务手册》姊妹篇
- 边做边学，实战演练
- 基础—案例—高级—练习，一步都不能少
- 情景式对话，再现建模全过程，令你身临其境



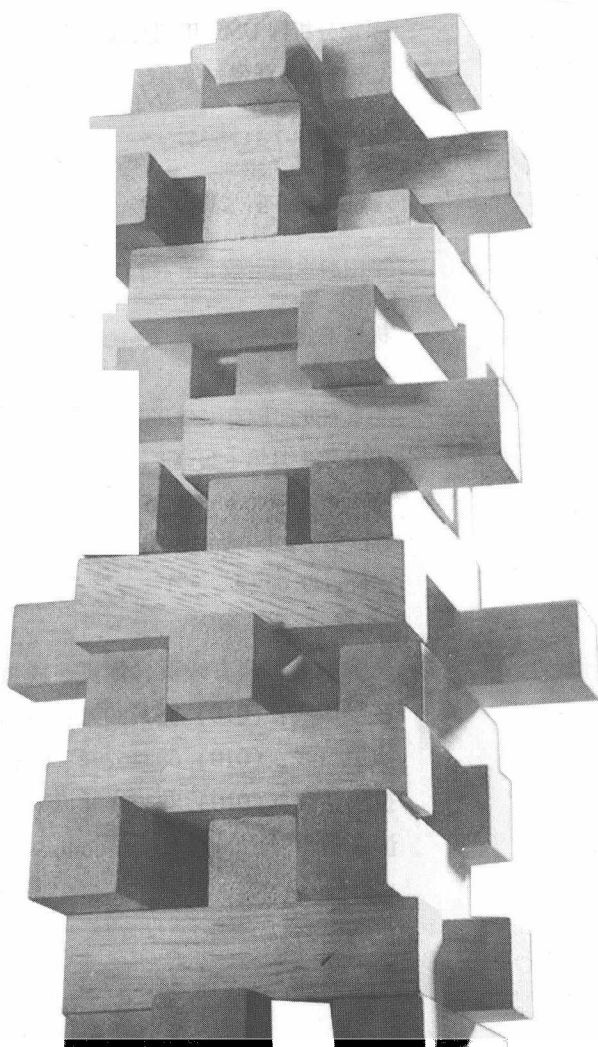
机械工业出版社
China Machine Press



华章原创精品

系统分析师 UML用例实战

邱郁惠 编著



机械工业出版社
China Machine Press

本书介绍如何通过用例掌握 UML。本书的案例基于 Wesley 和 Richard 两个角色叙述,从两人开始接到一个书店系统的项目,到动手建立用例模型,并且应用用例技术来估算工时,系统记述了 UML 用例的应用方法。

本书适用于 UML 用例技术初学者、系统分析师,同时也适合 UML/OOAD 项目经理阅读和学习。

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目 (CIP) 数据

系统分析师 UML 用例实战 / 邱郁惠编著. —北京: 机械工业出版社, 2010. 1
(原创精品系列)

ISBN 978-7-111-28576-2

I. 系… II. 邱… III. 面向对象语言, UML—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 188725 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 李东震

北京京北印刷有限公司印刷

2010 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 23.75 印张

标准书号: ISBN 978-7-111-28576-2

定价: 49.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991; 88361066

购书热线: (010) 68326294; 88379649; 68995259

投稿热线: (010) 88379604

读者信箱: hzjsj@hzbook.com

谨以本书献给我的先生和女儿

——他们是我生命中的天使

前 言

“许多概念必须在实践中学习，适时地给予机会指导”，这是我一开始写这本书时，想要呈现给读者的主旨。所以，在这本书中，我特别模拟了一个小型的两人团队的工作情况，有些诙谐地、有点趣味地，适时地指出什么时候该运用什么样的技术。在使用的过程中，又会发生什么样的问题。而这个问题，两人又用了什么方法解决。最后，两人小组运用了本书所讲述的技术，讨论了一个包含数个子系统的书店系统。

俗话说，万事开头难，确实如此，写每一章的基础知识小节是最难的。我总是得气定神闲地坐下来反问自己：如果我是一个读者，我没有太多时间听你（作者）说闲话，你只要先教我最必要的一招半式，等我先去闯荡江湖之后，发现不足了，再回头心甘情愿地听你长篇大论、唠唠叨叨。

遵循这样的构想，我在每一章的小节标题前都划分了“基础、案例、高级、练习”这4个主题。在我的构想中，读者先读完基础小节，学了一招半式；然后跟着两人小组进入案例小节，去闯“十八铜人阵”；挨了打，再回过头来，进入高级小节，耐心地多学些知识。最后，如果是学校学生，可以进入练习小节，为自己找个练习背景去闯荡闯荡。

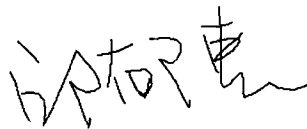
挑选书店系统作为案例，实在是因为它比较通俗，其实它也是一个最典型的电子商务系统，读者看到这个系统之后，很容易应用到自己手上的电子商务系统。而且，特别不同的是，大部分的 UML/OOAD 书籍不使用大型系统作为范例，但是我为了重现真实项目中所遇到的整合其他已有系统的情况，所以让这个书店系统规模变大，让它可以整合其他系统，这也更贴近真实的情况。

案例篇采用一个连贯的案例，由 Wesley 和 Richard 两人领衔演出（W: ，R: ）。两人从一开始接到一个书店系统的项目，在对项目范围认识模模糊糊的情况下，就开始动手建立使用用例模型，并且应用用例技术来估算工时。由于本书

的主题涵盖了估算工时的技术，所以除了一般的 IT 技术人员之外，UML/OOAD 项目经理也可以阅读本书。

我的第一本书《系统分析师 UML 实务手册》在机械工业出版社出版以后，在业界广受好评，而且还“阴错阳差”地成了一些大专院校的教材，销售“一鸣惊人”，也因此，有多位出版社的编辑希望我再写一本实用的学校教科书。

我没当过学校老师，不过却当了许多年的业界讲师。我通常把学员当成新书的假设读者，为他们编写教材，通过文字来传道、授业、解惑。只是没想到，这样的书也同样受到学校老师和学生的喜爱。令我受宠若惊。同样地，本书也是为业界与校园的读者而写，希望能够再一次深入浅出地讲述 UML/OOAD 的理论概念与实践经验。



E-mail: 271080@ gmail. com

UML Blog: <http://www.umltw.com>

作者简介

邱郁惠，中国台湾资深 UML 专家、讲师和畅销书作者。研究 OOAD、UML 和 MDA 十余年，从事过顾问、项目、教学及写作。创办了 UML Blog (<http://www.uml原因.com>) 推广 UML 技术，并且组织“UML 互助会”社区定期讲座、联络软件同好。除了定期举办社区活动、培训授课、参与项目外，其余时间她都在家里当专职作家并撰写博客。

邱老师在中国大陆出版了《系统分析师 UML 实务手册》、《C++ 程序员 UML 实务手册》、《SOC 设计 UML 实务手册》、《UML 那些事儿》等图书。她的作品以简洁明快、清晰实用的风格博得广大读者的好评，并因此荣获专业媒体和网站评选出的“2008 年优秀 IT 技术图书原创作者”奖。

目 录

前言

作者简介

第1章 绘制用例图 1

1.1 【基础】使用用例的时机 2

1.2 【基础】一睹用例的长相 3

1.3 【基础】绘制用例图 4

1.4 【案例】书店系统 9

1.5 【高级】系统内部启动的用例 37

1.6 【高级】UML 风格 40

1.7 【高级】用活动图来抓用例 58

1.8 【练习】动手做 61

第2章 编写用例叙述 67

2.1 【基础】用例叙述 68

2.2 【基础】主要流程与替代流程 71

2.3 【案例】书店系统 74

2.4 【高级】呈现风格 89

2.5 【高级】详细程度 91

2.6 【高级】站在巨人的肩膀上 92

2.7 【练习】动手做 106

第3章 建立用例关系 109

3.1 【基础】包含关系 110

3.2 【基础】扩展关系 114

3.3 【基础】泛化关系 117

3.4 【案例】书店系统 121

3.5 【高级】UML 风格 139

3.6 【高级】登录的问题 154

3.7 【高级】新增、读取、更新和删除
的问题 165

3.8 【练习】动手做 170

第4章 切分子系统用例 171

4.1 【基础】什么是“子系统
用例” 172

4.2 【基础】架构模式 173

4.3 【基础】切分子系统用例 176

4.4 【案例】书店系统 181

4.5 【高级】串起子系统用例 198

4.6 【高级】定期检查 201

4.7 【练习】动手做 212

第5章 估算工时 217

5.1 【基础】估算工时 218

5.2 【案例】书店系统 227

5.3 【高级】迭代式开发 247

5.4 【高级】事务模式 249

5.5 【练习】动手做 279

第6章 接下来可以怎么做 281

6.1 【基础】少不了三王一后 282

6.2 【案例】书店系统	283	7.3 打开或新增 UML 图	330
6.3 【高级】寻找类与对象	296	7.4 绘制用例图	331
6.4 【高级】面向对象分析	307	7.5 绘制类图	337
第 7 章 StarUML 操作说明	319	7.6 绘制序列图	342
7.1 StarUML 的特色	320	附录 A 书店系统	347
7.2 打开一个新的项目	328	附录 B 实用小工具	359

第 1 章 绘制用例图

- 1.1 【基础】 使用用例的时机
- 1.2 【基础】 一睹用例的长相
- 1.3 【基础】 绘制用例图
- 1.4 【案例】 书店系统
- 1.5 【高级】 系统内部启动的用例
- 1.6 【高级】 UML 风格
- 1.7 【高级】 用活动图来抓用例
- 1.8 【练习】 动手做

1.1 【基础】使用用例的时机

使用用例（use case）的时机，直白点讲，就是要回答“用例到底用在哪里”这个问题。如果将整个软件开发过程简单划分成分析（analysis）、设计（design）和实现（implementation）这3大步骤的话，用例主要用在分析阶段，如图1-1所示。也就是说，用例是一种系统分析技术。

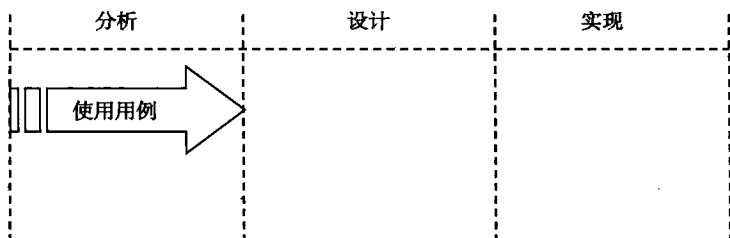


图1-1 在分析阶段使用用例

既然用例是一种系统分析技术，那我们免不了要了解一下，系统分析在做什么。针对前面提到的软件开发3大步骤，我们简单说明如下：

- 分析——是为了说明系统是什么（what），也就是搞清楚我们要开发一个什么样的系统。并说明这个系统会做哪些事。
- 设计——是为了说明系统如何工作（how），也就是说明系统应该如何一步一步地做到在分析阶段所承诺的事情。
- 实现——就是按照系统设计，真正地开始编写程序代码了。

现在，用例是开发UML/OOAD系统最常用的分析技术。系统分析师使用用例之后，主要会有两个结果：其一是用例图（use case diagram），其二是用例叙述（use case narrative）。前者纳入到了UML（Unified Modeling Language，统一建模语言）标准中，后者则没有。不过，在实际应用中，用例图与用例叙述两者缺一不可。

因为用例图纳入到了UML标准中，所以在使用上，它有规范可循。但是用例叙述没有纳入到UML标准，所以发展出多样化的叙述格式，这是在实际应用

中经常困扰系统分析师的问题。

1.2 【基础】一睹用例的长相

在开始动手绘制第一张用例图之前，先来看看用例图及用例叙述的最简单的模样，如图 1-2 与表 1-1 所示。这是个书店系统的范例，毋庸置疑，实际的项目，一定比这个范例复杂许多，但是任何复杂的事物，都是由简单的事物慢慢变复杂的，所以我们还是先从这个简单的事物学起吧！

或许你已经读过相关图书，知道图 1-2 中的人型图标叫做“参与者”（Actor），它代表系统外部的用户，而椭圆图标叫做“用例”（use case），代表系统内部对外提供的服务项目，方框图标则用来表示“系统”（system）。先知道这些就可以了，等会儿我们再来说明如何绘制这张图。

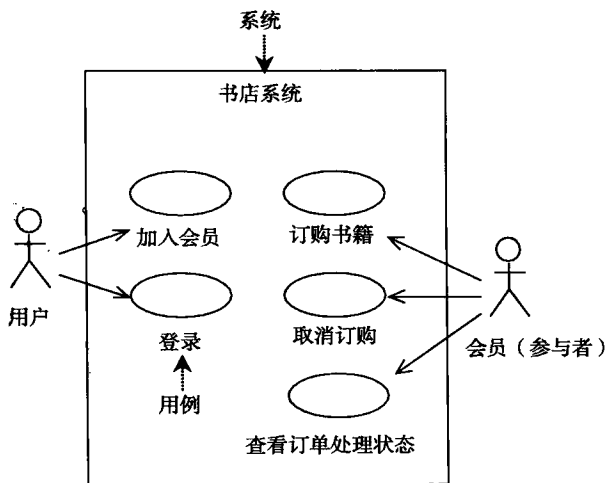


图 1-2 书店系统的用例图

系统分析师不是这么好当的，除了绘制用例图之外，还要针对每一个用例撰写叙述。所以，请你看表 1-1，这是订购书籍用例的文字叙述，里面主要记录了会员和书店系统交互的情况，说明会员如何一步一步地经历订购书籍的整个过程。

表 1-1 订购书籍的用例叙述

用例：订购书籍

事件流程：

1. 当会员选择订购书籍时，这个用例就会启动。
2. 会员输入他的姓名与地址。
3. 会员输入欲购买的书籍的书号。
4. 系统提供书籍说明与每一本书的价格。
5. 当会员输入欲订购的书号时，系统会持续保留订购的累计金额。
6. 会员输入信用卡付款信息。
7. 会员确认订购。
8. 系统核对信息，保存订购信息，并且把付款信息转交给会计系统。如果信息有任何不正确的地方，系统会提示会员修正该信息。
9. 当付款信息确认后，订购交易会标记为“已结账”，交易代号会回传给会员，而且这个用例即告终结。如果付款没有确认，系统会提示会员修正付款信息或取消。如果会员选择修正该信息，则回到上述的第 6 个步骤。如果会员选择取消，则该用例即告终结。

1.3 【基础】绘制用例图

先睹为快之后，我们回过头来介绍如何绘制用例图。

首先，让我们回到项目最开始的时候。想象一下，项目的一开始总是模模糊糊的，简单来说，就是不知道什么是我们该做的。什么又是我们不需要做的。如果能够搞清楚这一点，那一切就好办了。所以，抓一只鸡来，快速咚、咚、咚、咚，去头去尾去翅去爪，剩下的，就是我们需要负责的部分了，如图 1-3 所示。

确立系统边界，就是先划定一个系统范围，晚点再来推敲系统范围内该做的细节，这样才不会白费工夫。确立系统边界其实不难，就像剁鸡一样，需要的留下来，不需要的剁掉，所以系统分析师只要做好下面两件事情就可以了：

1. 找出位于系统外部的物，画小人。我们不需要开发位于系统外部的物，但是需要考虑建立接口（interface），让系统内外可以通过接口传递信息。就好像我们虽然不要鸡头，但是还是需要知道鸡头和鸡身连接的

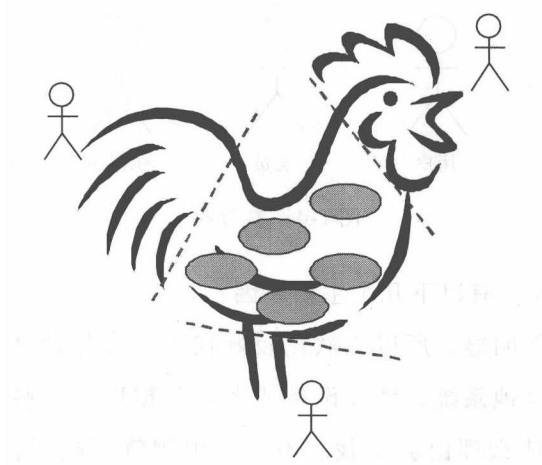


图 1-3 确立系统边界

关节在哪里！

2. 找出位于系统内部的事物，画椭圆。这些事物正是我们需要考虑开发的部分。

“确立系统边界”是句很抽象的话，而具体的做法就是：分内外，向系统内找用例，向系统外找参与者。

1.3.1 找参与者

参与者是位于系统外部的用户、联网的其他系统、硬件设备或者数据库，它们并不是系统的一部分，所以开发团队不需要构建这些参与者。不过，由于参与者会与系统互动、参与用例，以便获得特定的服务或结果，所以开发团队需要为它们设计接口，以便它们能够顺利地通过接口与系统交换信息。

请看图 1-4，虽然参与者的图标是人型图标，但不代表只有人类用户可以扮演参与者，位于系统之外并且会与系统交换信息的人或物（如与系统联网的系统、硬件设备、数据库等），都有可能成为系统的参与者。

到底是先找参与者好？还是先找用例好？这是许多学习用例的系统分析师会有的疑问。如果一定要有一个先后顺序的话，笔者建议还是先寻找参与者吧！



图 1-4 参与者

之所以给出这种建议，有以下几个主要原因：

- 参与者比用例明显，所以当然比较好找了。参与者是位于系统外部的用户、联网的其他系统、硬件设备或者是数据库，这些都是非常明确的人与物，没有什么理由会难找。不过，用例就不同了，它位于系统内部，可多可少、有大有小、能简能繁、或明或暗，所以寻找用例远比寻找参与者复杂得多。
- 参与者的个数远比用例的个数少，所以一下子就可以找齐全了，不像用例，似乎永远找不完。
- 找到一个参与者，就可以找到一堆用例，这比先找用例高效多了。（我记得曾经看过一个灭蟑的新闻提到：有专家表示，出现一只蟑螂代表背后至少还有超过 20 只。我想找到一个参与者的情况也相似吧，找到一个参与者代表背后至少还有 20 个用例！）
- 参与者是系统外部人物的代表，所以当然是要先找出参与者，才能够从参与者的角度去寻找用例。如果跳过参与者，直接寻找用例，就很容易陷入开发人员的角度，找出四不像的用例，爹（出钱的大老板）不疼、娘（真正的用户）不爱的，日后恐怕会很难验收吧！

那么，该如何下手寻找参与者呢？下列几个问题对寻找参与者很有帮助，可供参考：

1. 谁会来使用这个系统？
2. 谁会来安装这个系统？
3. 谁会来启动这个系统？

4. 谁会来维护这个系统？
5. 谁会来关闭这个系统？
6. 哪些系统会来使用这个系统？
7. 谁会从这个系统获取信息？
8. 谁会给这个系统提供信息？
9. 在预先设定的时间到达时，有什么事情会自动发生吗？
10. 哪些系统会与这个系统联网？
11. 是否有硬件设备会与这个系统联网？
12. 哪些数据库会与这个系统联网？
13. 公司内部有哪些人员会来使用这个系统？
14. 公司外部有哪些人会来使用这个系统？
15. 当特定的时间或事件发生时，这个系统需要自动通知什么人，或者是自动通知其他系统吗？

1.3.2 找用例

“用例是系统的一项行为，它能够生成参与者可评价的结果”。这句看似简单的描述，其实隐含着如下两个重点：

- 用例是系统的行为，所以它将包含着时间的概念，它会花费一段时间，具有一个执行过程。因此，当系统分析师描述用例时，要特别注意它所经历的一连串操作，而且这一连串的操作，正是参与者与系统之间的互动。
- 用例不仅能够，同时也必须，生成可以供参与者评价的服务或结果。也就是说，用例执行结束时，必须有一项明确的结果，并且参与者要能够接受这项结果。

如图 1-5 所示，在 UML 这种图形语言中，以椭圆图标代表用例，而且它与参与者之间会有关系线，代表着参与者将参与这个用例。当系统分析师绘制出如图 1-5 所示的用例图时，想传达的是，在系统外部有一个名为会员的参与者，

而且这个会员（参与者）会参与系统所提供的3个用例，这3个用例分别名为：订购书籍、取消订购和查看订单处理状态。

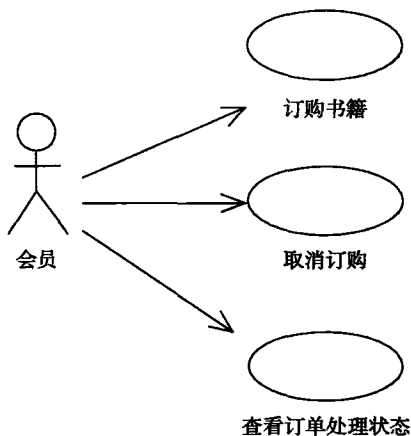


图 1-5 用例

找到参与者之后，再找用例就容易多了！我们可以针对每一个参与者寻找相关的用例。同样地，也可以问问自己下列问题，这对于寻找用例，帮助很大：

1. 参与者想要从这个系统获得什么样的功能？
2. 这个系统存储信息吗？哪些参与者将建立、读取、更新或删除这些信息？
3. 当系统内部状态有变化时，这个系统需要通知参与者吗？
4. 是否有什么外部事件是这个系统需要知道的？当这些外部事件发生时，哪些参与者会通知这个系统？
5. 这个系统需要定期执行什么操作吗？
6. 当发生了某些重要的外部事件时，这个系统需要自动执行什么操作吗？
7. 这个用例的名称够明确吗？是否能够从这个用例的名称，直接判断出它的结果？
8. 这个用例会有许多不同结果吗？还是这些结果，其实是在不同的时间点产生的？

前文曾提到，确定系统边界其实很简单，只要做两件事情就可以了：一是