

用例建模技巧

适合于更好的 UML 用例模型的技术

本文介绍了一些提高系统用例模型质量的技巧和技术。本文改编自 Object Primer 2nd Edition 的第 6 章。

[Scott W. Ambler](mailto:scott.ambler@ronin-intl.com) (scott.ambler@ronin-intl.com), 总裁, Ronin International

2001 年 1 月 04 日

内容



从参与者的角度并以主动语态编写用例。

应该以主动语态：“学生表明参加研习班意向”，而不是被动语态“研习班意向被学生表明”来编写用例。而且，应该从参与者的角度来编写用例。毕竟，用例的目的是理解用户如何对系统进行操作。

编写方案文本，而非功能需求。

用例描述的是对参与者来说有价值的一系列行动，而不是特性集。例如，“招收研习班的学生”用例描述的是学生如何与系统交互来参加研习班。它没有描述用户界面看上去是什么样子，或者它是如何工作的。有一些其它的模型来描述这些重要的信息，例如用户界面模型和增补规范。面向对象分析非常复杂，因此需要对它使用几种模型，并且应该适当地应用每一种模型。

用例只记载行为需求。

用例既不是类规范，也不是数据规范。这是应该由概念性模型捕捉的一种信息，在对象世界中，它是通过 UML 类模型建模的。您往往会引用概念性模型中描述的类，例如，“参加研习班”用例包括了“研习班”和“学生”等概念，它们都将由概念性模型描述。

不要忘记用户界面。

系统用例经常引用主用户界面 (UI) 元素，这些元素常常称为“边界”或“用户界面”项，例如 HTML 页面和报表。用例有时也引用一些次要的 UI 元素，例如按钮或数据输入字段，但这种级别的细节并不太常见。

创建用例模板。

用例包含了相当数量的信息，这些信息可以轻易地以常见格式记载。您应该考虑开发自己的模板（请参阅技巧“[记载用例](#)”）。

始终如一地组织用例图。

一般的做法是垂直地绘制继承 (inheritance) 和扩展 (extend) 关联，在父/基本用例下面绘制继承/扩展用例。同样，通常水平绘制包含 (include) 关联。请注意，这些是简单的经验法则 -- 只要始终遵循这些法则，产生的图将很容易理解。

不要忘记系统对参与者行动的响应。

用例既应该描述参与者是如何与系统交互的，也应该描述系统如何响应这些交互。例如，在“参加研习班”用例中，如果系统在学生表明他们希望参加研习班时没有做出响应，学生就会很沮丧地离开。

备选行动过程非常重要。

如果一切顺利，使用的将是基本行动过程 --但也不要忘记备选过程。引入备选过程是为了描述潜在的使用错误以及商业逻辑错误和异常。这些重要的信息对于驱动系统的设计来说很有必要，因此不要忘记在用例中对它们建模。

不要被 <<include>> 和 <<extend>> 关联所困扰。

我不是很确定到底发生了什么事，但我总是在想包含 (include) 和扩展 (extend) 关联，以及旧版本 UML 中使用 (uses) 和扩展 (extends) 关联的正确使用从来没有得到很好的描述。结果，用例建模小组往往在这些关联的正确应用上争论不休，在整个建模技术中一些有趣但次要的部分上浪费了惊人的时间。我曾在一个组织中工作，这家组织居然取缔了 <<include>> 和 <<extend>> 原型的使用，几个星期后，当意识到公司仍然需要这些概念时不得不撤消了这种极端的解决方案，而这时该组织对它们的正确使用还没有达成共识。

让用例带动用户文档。

用户文档的目的是描述如何使用系统。每个用例都描述了参与者通过使用系统所采取的一系列动作。简而言之，用例包含从中开始编写用户稳当的信息。例如，可以使用“参加研习班”用例作为基础来编写系统用户文档的“如何参加研习班”一节。

让用例带动演示。

软件开发过程中的一部分是向项目资金管理者通报工作成果，因此有时需要提供演示。因为用例是从用户的角度编写的，它们包含了演示中对资金管理者可能希望听到的事物的有价值的深刻见解。换句话说，用例通常包含制定演示稿所需的逻辑。

参考资料

- 您可以参阅本文在 developerWorks 全球站点上的 [英文原文](#)。
- [The Object Primer 2nd Edition](#)，由 Scott W. Ambler 著。New York: Cambridge University Press, 2001。
- [The Unified Process Inception Phase](#)，由 Scott W. Ambler 和 Larry L. Constantine 合著。Gilroy, CA: R&D Books, 2000。
- [The Unified Modeling Language Reference Manual](#)，由 James Rumbaugh、Grady Booch 和 Ivar Jacobson 合著。Reading, MA: Addison-Wesley Longman, Inc., 1999。

条评论

请 [登录](#) 或 [注册](#) 后发表评论。

添加评论:

注意：评论中不支持 HTML 语法

有新评论时提醒我

剩余 1000 字符