Oracle PL/SQL Programming

下册

Oracle PL/SQL

程序设计（第5版）

[美] *Steven Feuerstein & Bill Pribyl* 著
张晓明 译

O'REILLY®

人民邮电出版社
POSTS & TELECOM PRESS

Oracle PL/SQL Programming

Oracle PL/SQL 程序设计（第5版）

[illegible]

O'REILLY

[illegible]

Oracle PL/SQL 程序设计（第 5 版）

（下册）

[美] Steven Feuerstein　Bill Pribyl　著

张晓明　译

人 民 邮 电 出 版 社

北 京

目录

（下册）

第 5 部分　构造 PL/SQL 应用程序

第 6 部分 高级 PL/SQL 主题

第 5 部分

构造 PL/SQL 应用程序

这一部分对前面的内容进行了综合。现在，你已经掌握了基础知识。你知道如何声明变量和使用变量。你俨然已经是一个错误处理和循环结构方面的专家了。是时候构建应用程序了——你可以构建块、创建过程、函数、包、触发器，这些在第 17 章到第 19 章中介绍。第 20 章讨论了对 PL/SQL 代码的管理，包括对程序的测试和调试以及管理依赖关系；这一章还对 Oracle 数据库 11g R2 版本引入的基于版本的重定义功能进行了简要的介绍，第 21 章，这是第 5 版新加的一章，主要关注的是如何可以通过若干工具和技术帮助你的 PL/SQL 程序达到最好的性能。第 22 章介绍了 PL/SQL 的 I/O 技术，从 DBMS_OUTPUT（输出内容到屏幕）和 UTL_FILE（文件的读写）到 UTL_MAIL（发送邮件）和 UTL_HTTP（从 Web 页面提取数据）。

第 17 章

过程、函数与参数

本书前面的内容中已经详细的介绍了 PL/SQL 语言的各种组件：包括游标、异常、循环、变量等。要想用 PL/SQL 编写应用程序，确实也需要知道所有这些组件，不过比这些内容更重要的是该如何把这些片段组装在一起，进而创建一个结构良好的、容易理解的、便于维护的程序。

交给我们的任务很少有简单直接的。而需要的解决办法也很少能不假思索就能写到纸上或者输入到键盘的。我们构建的系统通常是庞大并且复杂的，会有许多相互影响甚至是相互冲突的组件组成。而且，从最终用户的角度来说，他们想要得到东西是比之前更容易使用的、功能更强大的应用程序，这些应用程序的内部实现也相应的变得更加复杂了。

今天我们工作所面临的最大挑战来自于该如何简化我们的环境的复杂性。当面临一个复杂的问题时，感觉上是很恐怖的。我们该从那里入手呢？我们该如何快刀斩乱麻，从一堆需求和特性中找到出路呢？

我们人类毕竟赶不上大型的并行计算机。就算是最聪明的人也不可能同时处理 7 个任务（加减二）。我们需要把一个庞大的、吓人的项目分解成小的、更容易管理的组件，然后再进一步把这些组件分解成一个个好理解的单独程序。这样我们就知道该如何构建和测试这些程序了，最后再用这些组件构成一个完整的应用程序。

无论我们用的是“从顶到下的设计”（也叫做逐步分解，在“局部或者嵌模块”一节会有详细的介绍）还是某种其他的方法学，毋庸置疑的是，通过把程序模块化成一个个过程、函数或者对象类型，我们会得到一个高质量的容易管理的应用程序。

17.1 代码模块化

模块化是把一个大的代码块拆分成若干个小片段（模块）的过程，然后就可以在其他

模块中调用这些模块了。代码的模块化非常类似于数据的规范化，得到的好处是相同的，而且还有一些额外的优点。利用模块化，我们的代码可以：

重用性更好

通过把一大段代码或者整个程序分解成独立的“即插即用”的组件，我们经常发现，许多模块其实可以被当前应用中的其他程序使用。只要有良好的设计，这些工具程序甚至可以用于其他应用程序！

管理性更好

一个 1 000 行的程序或者 5 个单独的每个只有 200 行的并且互相调用的程序，你更愿意调试哪一个？在处理小任务时，我们的注意力会更加集中，也能处理的更好。我们可以在每个程序范围进行测试和调试（也叫做单元测试），然后再把这些独立的模块合并起来，进行一个更复杂的集成测试。

可读性更好

模块可以被命名，可以通过名字来描述其行为。通过程序接口隐藏起来的代码越多，程序行为就越容易阅读和理解。通过模块化，我们可以更专注于程序全局而不是一个个单独的可执行语句。我们甚至可以实现最隐蔽的软件：完全自我说明的代码。

更可靠的代码

按照模块化思路生成的代码的错误会更少。而且就算是有错误，也会因为它们已经被隔离在模块范围中而更容易被修正。而且，由于代码量更少并且可读性更好，我们维护起来也更容易。

只要你能够熟练地使用 PL/SQL 语言中的各种不同的循环、条件、游标结构（IF 语句、循环等），你其实就已经为构建应用系统做好准备了。不过，你还必须知道如何创建、整合 PL/SQL 的模块，这才真正可以构建应用程序。

PL/SQL 语言中提供了下面这些模块结构，你可以用不同的方法实现代码的模块化：

过程

过程是一个可以像可执行 PL/SQL 语句一样调用的程序，一个过程可以执行一个或者多个动作。我们可以通过参数列表向过程传递或者从过程传出信息。

函数

函数是一个通过 RETURN 语句返回数据的程序，使用起来就像是一个 PL/SQL 表达式。我们可以通过参数列表给函数传入参数。也可以通过参数列表传出参数，不过通常情况下这么做并不好。