

<<SQL Server 2005 T-SQ>>

图书基本信息

书名：<<SQL Server 2005 T-SQL&.NET高级编程>>

13位ISBN编号：9787302173403

10位ISBN编号：7302173400

出版时间：2008-5

出版时间：清华大学出版社

作者：Dejan Sunderic

页数：574

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<SQL Server 2005 T-SQ>>

内容概要

为了开发出复杂的存储过程以检索、操作、更新和删除数据，本书介绍并描述了您所需掌握的关键概念、技术和最佳实践，以使您通过SQL Server自带的Transact—SQL和.NET CLR语言充分利用存储过程。

在本书中，您将学到如何将有效的Transact.-SQL存储过程结合到客户端与中间件代码中，以及如何生成将被编译到CLR存储过程中的CLR方法。

本书内容丰富、指导性强，是所有SQL Server 2005开发人员的必备参考。

本书主要内容

| | | |
|---------------------------------|------------------------------------|------------------------------|
| 使用批处理、脚本与事务来组合和执行T-SQL语句 | 创建用户自定义的、系统的、扩展的、临时的、全局临时的和远程的存储过程 | 使用ADO.NET实现数据库访问 |
| 用C#和Visual Basic.NET开发与管理存储过程 | 创建CLR用户自定义函数与触发器 | 实现可靠的调试、错误处理技术和安全度量 |
| 在类似于Visual SourceSafe的储存库中管理源代码 | 为Web搜索引擎创建存储过程 | 使用系统和扩展存储过程与SQL Server环境进行交互 |

作者简介

桑德里克，微软认证的解决方案专家（MCSD）和数据库管理员（MCDBA），是Trigon Blue, Inc. 的首席顾问，也是Toronto SQL Server User Group的总裁。他专门研究SQL Server和Windows平台上的数据库管理、部署及应用程序开发。Dejan曾经参与的项目涉及82C和B28电子商务、金融、

书籍目录

- 第1章 SQL Server 2005环境与工具 1.1 SQL Server 2005工具 1.1.1 SQL Server Configuration Manager
 1.1.2 SQL Server Management Studio 1.1.3 SQLCMD实用程序 1.1.4 SQL Server Profiler
 1.1.5 帮助文档和SQL Server Books Online 1.2 Management Studio中的基本操作 1.2.1 存储结构的定义
 1.2.2 Management Studio中存储过程的执行 1.2.3 编辑存储过程 1.2.4 创建存储过程
 1.2.5 在Query窗口中编辑存储过程 1.2.6 语法错误 1.2.7 查看和编辑表 1.2.8 修改表结构
- 第2章 存储过程设计的一些概念 2.1 存储过程的剖析 2.1.1 组成部分 2.1.2 功能 2.1.3 语法
 2.2 存储过程的类型 2.3 管理存储过程 2.3.1 列出存储过程 2.3.2 查看存储过程的代码
 2.3.3 重命名存储过程 2.3.4 删除存储过程 2.3.5 列出被依赖和依赖对象 2.4 存储过程在数据库应用程序开发中的作用
 2.4.1 增强数据完整性 2.4.2 复杂业务规则和约束的一致实现
 2.4.3 模块化设计 2.4.4 可维护性 2.4.5 降低的网络通信量 2.4.6 较快的执行 2.4.7 安全性的增强
- 第3章 基本Transact-SQL编程结构 3.1 T-SQL标识符 3.2 数据库对象限定符 3.3 数据类型
 3.3.1 字符串型 3.3.2 Unicode字符串型 3.3.3 日期和时间型 3.3.4 整型 3.3.5 近似数值型
 3.3.6 精确数值型 3.3.7 货币型 3.3.8 二进制型 3.3.9 特殊类型 3.3.10
 Transact-SQL用户自定义数据类型 3.4 变量 3.4.1 局部变量 3.4.2 全局变量 3.4.3 表变量
 3.5 流程控制语句 3.5.1 注释 3.5.2 语句块: Begin...End 3.5.3 条件执行: If语句 3.5.4
 循环: While语句 3.5.5 无条件执行: GoTo语句 3.5.6 调度执行: WaitFor语句 3.6 游标
 3.6.1 Transact-SQL游标 3.6.2 与游标相关的语句和函数 3.6.3 使用游标的问题 3.6.4 游标的正确用法
- 第4章 函数 4.1 函数的用法 4.1.1 用函数来选择和赋值 4.1.2 作为选择条件的一部分
 4.1.3 在表达式中使用函数 4.1.4 作为Check和Default约束 4.1.5 取代表 4.2 函数的类型
 4.2.1 标量函数 4.2.2 行集函数
- 第5章 复合Transact-SQL结构: 批处理、脚本和事务 5.1 批处理
 5.1.1 使用批处理 5.1.2 批处理与错误 5.1.3 DDL批处理 5.1.4 自给自足的内容 5.2
 脚本 5.3 事务 5.3.1 自动提交事务 5.3.2 显式事务 5.3.3 隐式事务 5.3.4 事务处理体系结构
 5.3.5 嵌套事务 5.3.6 命名事务 5.3.7 保存点 5.3.8 加锁 5.3.9 分布式事务
 5.3.10 与事务相关的典型问题
- 第6章 错误处理 6.1 Raiserror语句 6.2 使用错误处理 6.3 添加错误处理代码的原因
 6.4 基于@@Error的错误处理 6.5 Try-Catch语句 6.5.1 捕获的错误类型 6.5.2
 Catch块的函数 6.5.3 带显式事务的Try-Catch语句 6.5.4 死锁重试 6.5.5 Try-Catch语句嵌套
 6.5.6 错误处理体系结构: 使用Try-Catch语句 6.6 Xact_Abort选项的设置 6.6.1 错误处理体系结构:
 基于Set Xact_Abort On设置 6.6.2 错误处理体系结构: 设置Xact_Abort且事物嵌套为零
- 第7章 特殊存储过程类型 7.1 用户自定义存储过程 7.2 系统存储过程 7.3 CLR存储过程 7.4 扩展存储过程
 7.4.1 扩展存储过程的设计 7.4.2 注册扩展存储过程 7.5 临时存储过程 7.6 全局临时存储过程
 7.7 远程存储过程
- 第8章 视图 8.1 标准SQL视图的设计 8.1.1 语法 8.1.2 Enterprise Manager中的视图设计
 8.1.3 安全性 8.1.4 执行计划中的标准SQL视图 8.1.5 视图的限制 8.1.6 用视图编辑数据
 8.2 动态视图 8.3 临时视图--通用表表达式 8.3.1 非递归CTE的限制 8.3.2 递归CTE
 8.4 INFORMATION_SCHEMA视图 8.5 索引视图 8.5.1 索引视图限制 8.5.2 执行计划中的索引视图
 8.5.3 视图上的非群集索引 8.5.4 性能影响 8.6 分区视图 8.6.1 水平和垂直分区
 8.6.2 分布式分区视图 8.6.3 分布式分区视图的执行计划 8.6.4 可更新的分布式分区视图
 8.6.5 分布式系统的稳定性和性能 8.6.6 简单的联合服务器 8.7 使用SQL视图 8.7.1 导出和导入
 8.7.2 安全性实现 8.7.3 降低复杂性 8.7.4 性能提升
- 第9章 触发性 9.1 DML触发器
 9.1.1 After触发器的物理设计 9.1.2 插入和删除虚拟表 9.1.3 触发器激活条件 9.1.4 After触发器的完整语法
 9.1.5 处理多条记录上的修改 9.1.6 嵌套触发和递归触发 9.1.7 After触发器限制
 9.1.8 Instead-of触发器 9.1.9 视图上的触发器 9.1.10 DML触发器的执行次序 9.2 DDL触发器
 9.2.1 DDL触发器的作用域和事件 9.2.2 eventdata () 函数 9.2.3 DDL触发器的语法
 9.3 管理触发器 9.3.1 在Management Studio中管理DML触发器 9.3.2 在Management Studio中管理DDL触发器
 9.3.3 用Transact-SQL语句管理触发器 9.4 触发器的设计建议 9.4.1 尽可能快地退出触发器
 9.4.2 简化触发器的代码 9.4.3 按功能实现触发器 9.4.4 不要在触发器内使

用Select和Print 9.4.5 根本不要使用触发器 9.5 触发器中的事务管理 9.6 使用触发器 9.6.1 级联删除 9.6.2 聚集 9.6.3 增强不同服务器或数据库上的对象的模式完整性第10章 用户自定义函数 10.1 标量用户自定义函数的设计 10.1.1 副作用 10.1.2 内置函数的使用 10.1.3 加密 10.1.4 模式绑定 10.2 表-值用户自定义函数 10.3 在Management Studio中管理用户自定义函数第11章 SQL Server 2005中的.NET编程基础 11.1 非托管和托管代码 11.2 公共语言运行库 11.3 SQL Server 2005中的CLR集成 11.3.1 优缺点 11.3.2 支持和不支持的功能 11.3.3 所需最小空间 11.4 Visual Studio 2005中CLR数据库对象的开发、部署和使用 11.5 不使用Visual Studio 2005时CLR数据库对象的开发和部署 11.5.1 开发代码 11.5.2 编译程序 11.5.3 向SQL Server数据库中加载程序集 11.5.4 创建SQLCLR数据库对象 11.6 为程序集和CLR数据库对象编目录 11.6.1 为程序集编目录 11.6.2 为过程编目录 11.6.3 .NET Framework的版本第12章 CLR存储过程开发基础 12.1 CLR存储过程的结构 12.2 从CLR过程中访问数据 12.3 使用存储过程的基本操作 12.3.1 返回值 12.3.2 连接上下文 12.3.3 返回结果 12.3.4 返回自定义记录 12.3.5 CLR存储过程中的参数 12.4 带正规ADO.NET对象的操作 12.4.1 SqlCommand 12.4.2 SqlDataReader 12.5 管理LOB参数第13章 CLR函数与触发器 13.1 CLR函数 13.1.1 标量-值CLR函数 13.1.2 SqlFunction属性 13.1.3 表-值CLR函数 13.2 CLR触发器 13.2.1 支持的特性 13.2.2 DML触发器的创建 13.2.3 DDL触发器的创建第14章 CLR数据库对象高级主题 14.1 CLR用户自定义类型(UDT) 14.1.1 托管UDT的结构 14.1.2 不使用Visual Studio 2005部署CLR UDT 14.1.3 使用CLR用户自定义类型 14.1.4 跨数据库的用户自定义类型 14.2 是否使用UDT 14.3 用户自定义聚合CLR函数 14.4 CLR代码中的事务 14.4.1 使用TransactionScope类 14.4.2 事务选项 14.4.3 分布式事务 14.4.4 显式事务 14.4.5 新事务泛型的优点 14.5 是否设计CLR(设计与性能考虑) 14.6 警告第15章 高级存储过程编程 15.1 动态构造的查询 15.1.1 执行字符串 15.1.2 通过表单查询 15.1.3 数据脚本生成器 15.1.4 使用sp_executesql存储过程 15.1.5 安全隐患 15.2 使用timestamp值的乐观锁 15.2.1 timestamp数据类型 15.2.2 TSEqual()函数 15.3 全文本搜索和索引 15.4 嵌套存储过程 15.4.1 用临时表向嵌套存储过程传递记录集 15.4.2 用游标向嵌套存储过程传递记录集 15.4.3 如何处理存储过程的结果集 15.5 使用标识值 15.5.1 标准问题和答案 15.5.2 标识值与触发器 15.6 GUID 15.7 带Min()或Max()函数的循环 15.8 带sp_MSForEachTable和sp_MSForEachDb的循环 15.9 特性管理第16章 调试 16.1 什么是bug 16.2 调试过程 16.2.1 标识阶段 16.2.2 解决阶段 16.3 调试工具与技术 16.4 调试CLR数据库对象 16.5 简单的调试器 16.5.1 生产环境中的执行 16.5.2 嵌套存储过程 16.5.3 输出子句 16.6 SQL Profiler 16.7 典型错误 16.7.1 处理空值 16.7.2 从结果集中为变量赋值 16.7.3 没有记录受影响 16.7.4 错误的大小或数据类型 16.7.5 默认长度 16.7.6 触发器的回退 16.7.7 警告和较低优先级的错误 16.7.8 返回代码与Raiserror语句的使用 16.7.9 嵌套注释 16.7.10 延迟名称解析 16.7.11 游标 16.7.12 过分自信第17章 源代码管理 17.1 Microsoft Visual SourceSafe介绍 17.2 管理Visual SourceSafe数据库 17.3 在Visual Studio .NET中向Visual SourceSafe添加数据库对象 17.4 在Visual Studio .NET中管理Create脚本 17.5 Visual SourceSafe Explorer 17.5.1 历史 17.5.2 标签与版本 17.6 向Visual SourceSafe添加数据库对象:传统方法 17.6.1 用Script Wizard创建脚本 17.6.2 用TbDbScript创建脚本 17.6.3 用TbDir2Vss.vbs向Visual SourceSafe中添加脚本第18章 数据库部署 18.1 整个数据库的部署:传统方法 18.1.1 在Transact-SQL中附着与分离数据库 18.1.2 在Management Studio中附着和分离 18.1.3 备份与恢复 18.1.4 潜在的问题 18.2 单个对象的部署 18.2.1 部署脚本:传统方法 18.2.2 脚本化数据:传统方法 18.2.3 在Visual Studio .NET中脚本化数据 18.2.4 部署脚本:传统方法 18.2.5 在Visual Studio .NET中部署Create脚本 18.2.6 递增的构件:传统方法 18.2.7 Visual Studio .NET中的递增的构件第19章 安全 19.1 安全体系结构 19.1.1 身份验证与授权 19.1.2 主体 19.1.3 可保护对象 19.1.4 访问级别 19.1.5 角色 19.1.6 所有权链 19.1.7 交叉数据库所有者链 19.1.8 执行环境上下文的切换 19.2 实现安全性 19.2.1 身份验证模式的选择 19.2.2 管理登录名 19.2.3 授予数据库访问权限 19.2.4 赋予权限 19.3 登录名与用户名的同步 19.4 用存储过程、用户自定义函数和视图管理应用程序安全 19.5 用代理用户管理应用程序安全 19.6 用应用程序角色管理应用程序安全第20章 Web搜索引擎的

存储过程 20.1 环境特征 20.2 一个简单的解决方案 20.3 缺点 20.4 可用的解决方案 20.4.1 结果分割 20.4.2 快速查询 20.4.3 高级查询 20.4.4 新的解决方案：行版本第21章 SQL Server环境的交互作用 21.1 运行程序 21.2 运行Windows脚本文件 21.3 OLE Automation/COM对象的执行 21.4 数据类型转换 21.5 运行SQL Server 2005 Intergration Services (SSIS) 包 21.6 运行DTS包 21.7 在DTS包中实现循环 21.8 与NT注册表交互作用 21.8.1 xp_regread 21.8.2 xp_regwrite 21.9 作业 21.9.1 作业管理 21.9.2 Job Scheduler的替换方法 21.9.3 维护作业存储过程 21.9.4 操作员和警报 21.10 电子邮件 21.10.1 Database Mail 21.10.2 配置Database Mail 21.10.3 发送电子邮件 21.10.4 检查状态 21.11 将存储过程作为Web服务提供附录A 命名约定附录B 存储过程编译、存储和重用附录C SQL Server 2005中的数据类型

章节摘录

第1章 SQL Server 2005环境与工具 SQL Server 2005是一个功能全面而强大的数据库管理系统。您可能已对此系统的某些方面有一定的经验。但是在成为使用SQL Server存储过程的应用程序专家之前，我们先回过头来大体回顾一下，以便我们可以有相同的概念基础。

为了获得此概念基础，先浏览以下两个主题： ?SQL Server 2005工具概述 ?存储过程和表的设计快速浏览 编写本概述是为了使大家快速了解基础知识，然后深入学习复杂的存储过程，以便检索、操作、更新和删除数据，并解决各种业务问题。

本书假定您已经可以将SQL作为在SQL Server或其他数据库上使用的查询语言，而且假定您已理解了一般的数据库概念。

本概述的目的是为了定义术语，并在此基础上构建SQL Server环境中的编程知识体系。

如果您需要比本书更深入的概念知识，建议阅读《Microsoft SQLServer 2005初学者指南》（清华大学出版社引进并出版）这本书。

编辑推荐

《SQL Server 2005 T-SQL&.NET高级编程(第3版)》内容丰富、指导性强，是所有SQL Server 2005开发人员的必备参考。

掌握Transact-SQL和CLR存储过程编程，开发可扩展的Windows和基于Web的应用程序，实现调试技术与安全度量。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>