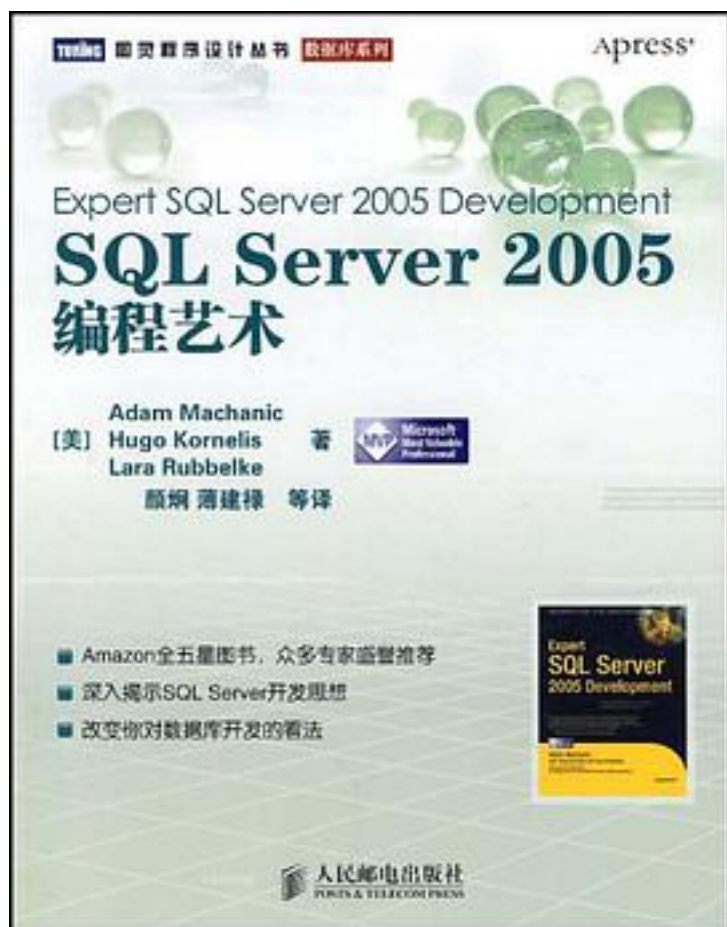


SQL Server 2005编程艺术



[SQL Server 2005编程艺术_下载链接1](#)

著者:[美] Adam Machanic

出版者:人民邮电出版社

出版时间:2008-7

装帧:

isbn:9787115179357

《SQL Server 2005编程艺术》是一部关于SQL Server 2005高级开发技术的权威著作，提供了详尽、全面的技术讨论，并给出了详细的应用实例。书中前4章着重介绍SQL Server所涉及的软件开发方法学、测试、异常处理和安全；第5章至第7章深入研究SQL

Server 独具的高级特性，主要介绍加密、SQLCLR 和动态SQL；第8 章至第11 章从数据库体系结构的角度分析问题，深入研究具体的设计和实现问题，包括应用程序并发、空间数据、时态数据和图，这是《SQL Server 2005编程艺术》最为复杂的内容，也是最有实际意义的内容。

作者介绍:

Adam Machanic，微软SQL Server MVP，著名的SQL Server专家。长期活跃于技术社区，是SQL Server Professional和TechNET等行业杂志的撰稿人，经常在各种技术会议上开设技术讲座、发表演讲。除本书外，他还是Pro SQL Server 2005的作者之一。

目录: 第1章 数据库领域的软件开发方法学

1.1 架构回顾

1.1.1 耦合、内聚和封装

1.1.2 接口

1.2 核心问题：集成数据库和面向对象系统

1.2.1 该何去何从

1.2.2 对象—关系失配

1.3 ORM：一个问题多多的解决方案

1.4 把数据库操作设计为API

1.5 全面权衡

1.5.1 可测试性

1.5.2 可维护性

1.5.3 安全性

1.5.4 性能

1.5.5 步履蹒跚的远期功能

1.6 小结

第2章 数据库例程测试

2.1 黑盒测试与白盒测试简介

2.1.1 单元测试和功能测试

2.1.2 单元测试框架

2.1.3 回归测试的重要性

2.2 实现数据库测试过程和测试例程的指导准则

2.2.1 为什么说软件测试很重要

2.2.2 哪种测试重要

2.2.3 需要进行多少测试

2.2.4 管理层是否愿意接受软件测试

2.3 性能测试和数据库系统事件探查

2.3.1 获取基准度量

2.3.2 使用跟踪器及事件探查器进行查看

2.3.3 评估性能计数器

2.3.4 宏观分析

2.3.5 粒度分析

2.3.6 修正问题：仅仅关注明显的错误是否已经足够

2.4 SQLQueryStress性能测试工具介绍

2.5 小结

第3章 错误和异常

3.1 异常和错误

3.2 SQL Server中异常的工作原理

3.2.1 语句级异常

3.2.2 批处理程序级异常

- 3.2.3 语法分析和作用域解析异常
- 3.2.4 连接和服务器级异常
- 3.2.5 XACT_ABORT设置
- 3.2.6 分析错误信息
- 3.2.7 SQL Server的RAISERROR函数
- 3.2.8 跟踪监控异常事件
- 3.3 异常处理
 - 3.3.1 为什么要在T-SQL中处理异常
 - 3.3.2 使用@@ERROR “处理” 异常
 - 3.3.3 SQL Server的TRY/CATCH语法
- 3.4 事务与异常
 - 3.4.1 事务退出的神话
 - 3.4.2 XACT_ABORT：神话（半）成真
 - 3.4.3 TRY/CATCH和被诅咒事务
- 3.5 小结
- 第4章 特权与授权
 - 4.1 最低特权原则
 - 4.1.1 在SQL Server中创建代理
 - 4.1.2 层级数据安全：洋葱模型
 - 4.2 使用Schema组织数据
 - 4.3 使用EXECUTE AS实现基本模拟
 - 4.4 所有权链
 - 4.5 非链式特权提升
 - 4.5.1 存储过程与EXECUTE AS
 - 4.5.2 使用证书对存储过程进行签名
 - 4.6 小结
- 第5章 加密
 - 5.1 要保护什么
 - 5.2 加密术语：我们需要知道的部分
 - 5.3 SQL Server 2005加密密钥体系
 - 5.3.1 服务主密钥
 - 5.3.2 数据库主密钥
 - 5.4 SQL Server 2005数据保护
 - 5.4.1 HashBytes ()
 - 5.4.2 非对称密钥加密和证书加密
 - 5.4.3 对称密钥加密
 - 5.4.4 EncryptByPassphrase
 - 5.5 隔离数据与DBA
 - 5.6 性能优化设计
 - 5.6.1 设计解决方案、定义问题
 - 5.6.2 搜索加密数据
 - 5.7 小结
- 第6章 SQLCLR：架构和设计注意事项
 - 6.1 填充SQL/CLR鸿沟：SqlTypes库
 - 6.2 包装代码以提高跨层重用性
 - 6.3 SQLCLR的安全性特征和可靠性特征
 - 6.3.1 代码安全性需求
 - 6.3.2 通过程序集引用有选择地提升特权
 - 6.3.3 授予交叉程序集特权
 - 6.4 利用SQLCLR加强Service Broker可扩展性功能
 - 6.5 扩展用户自定义聚合
 - 6.6 小结
- 第7章 动态T-SQL
 - 7.1 动态T-SQL与ad hoc T-SQL

7.2 为什么要使用动态SQL

7.2.1 编译与参数化

7.2.2 自动参数化

7.2.3 应用程序级参数化

7.2.4 参数化与高速缓存的性能暗示

7.3 支持可选参数

7.3.1 通过静态T-SQL实现可选参数

7.3.2 实现动态化：使用EXECUTE

7.3.3 SQL注入

7.3.4 sp_executesql：一个更好的EXECUTE

7.4 动态SQL在安全方面的考虑

7.4.1 被引用对象的权限

7.4.2 接口规则

7.5 小结

第8章 应用程序并发系统的设计

8.1 从业务角度观察：进程产生冲突时会发生什么情况

8.1.1 SQL Server隔离级别概述

8.1.2 并发控制与SQL Server提供的隔离级别

8.2 做最坏的打算：悲观的并发

8.2.1 在写入过程中使用悲观锁

8.2.2 应用程序锁：对悲观并发控制机制进行泛化

8.3 希望最好的情况发生：乐观的并发控制

8.4 包容冲突：多值并发控制

8.5 通过排队提高可扩展性

8.6 小结

第9章 空间数据处理

9.1 用纬度和经度表示地球空间数据

9.1.1 设置示例数据

9.1.2 计算两点之间的距离

9.1.3 从一个位置点移动到另一个位置点

9.1.4 邻近搜索

9.1.5 边界框

9.1.6 查找距离最近的邻近点

9.1.7 动态边界框

9.1.8 结论

9.2 使用分层三角形网格表示地球空间数据

9.2.1 HTM的简化描述

9.2.2 实现HtmlID

9.2.3 Spatial数据库提供的函数

9.2.4 结论

9.3 其他类型的空间数据

9.3.1 三维数据

9.3.2 天文数据

9.3.3 虚拟空间

9.3.4 将区域表达为多边形

9.4 小结

第10章 时态数据处理

10.1 表达的不仅仅是时间

10.2 SQL Server提供的日期/时间数据类型

10.2.1 日期输入格式

10.2.2 格式化输出日期

10.2.3 高效查询日期/时间列

10.2.4 日期/时间计算

10.3 使用日历表定义周期

10.4 设计时态数据存储方案及查询时态数据

10.4.1 处理时区

10.4.2 处理时间区间

10.4.3 为持续时间建立模型

10.4.4 管理双时态数据

10.5 小结

第11章 树、层次结构和图

11.1 术语：万物皆为图

11.2 基础知识：邻接表和图

11.2.1 为边加入约束

11.2.2 针对图的基本查询：给定一个结点，这个结点与其他哪些结点相连

11.2.3 遍历图

11.3 邻接表的层次结构

11.3.1 查询邻接表层次结构：基础知识

11.3.2 查找直接后代结点

11.3.3 在层次结构中向下级层次遍历

11.3.4 在层次结构中向上级层次遍历

11.3.5 插入新结点并重新定位子树

11.3.6 删除已有结点

11.3.7 为层次结构加入约束

11.4 持久保存具体化路径

11.4.1 查找下级结点

11.4.2 在层次结构中向上级层次导航

11.4.3 对具体化路径解决方案进行优化

11.4.4 插入结点

11.4.5 重新定位子树

11.4.6 删除结点

11.4.7 为层次结构加入约束

11.5 嵌套集模型

11.5.1 查找下级结点

11.5.2 在层次结构中向上级层次导航

11.5.3 插入结点

11.5.4 重新定位子树

11.5.5 删除结点

11.5.6 为层次结构加入约束

11.6 小结

• • • • • ([收起](#))

[SQL Server 2005编程艺术_下载链接1](#)

标签

软件

编程艺术

SQL_Server_2005

IT

C/C++

评论

没认真看。。。

[SQL Server 2005编程艺术_下载链接1](#)

书评

[SQL Server 2005编程艺术_下载链接1](#)