

<<深入SQL Server 2008>>

图书基本信息

书名：<<深入SQL Server 2008>>

13位ISBN编号：9787115256195

10位ISBN编号：7115256195

出版时间：2011-7

出版时间：人民邮电

作者：(美)沃尔特斯|译者:任斌//刘芳芳

页数：664

译者：任 斌,刘芳芳 等

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>

<<深入SQL Server 2008>>

内容概要

本书涵盖了SQL Server 2008引入的新功能。

作者之一Robert E. Walters是微软公司SQL

Server开发团队的项目经理，在关系数据库方面拥有多年的相关经验，这让此书内容更具权威性。

与市面上的其他介绍SQL

Server的书籍相比，本书的突出特点是贴近实战且兼顾原理介绍，几乎每章都有可以直接借用的代码和范例，同时对于难以理解之处，也有其实现原理的介绍，使读者知其然，更知其所以然。

此外，本书介绍的SQL

Server

2008策略管理、高可用性、性能、安全等内容，均是企业级数据库必不可少的部分，阅读本书可以详细了解这些技术内幕。

本书面向SQL Server数据库管理员和数据库开发人员，旨在让读者对SQL

Server有更深入的理解和全新的认识，成为一个SQL Server数据库管理方面的行家里手。

<<深入SQL Server 2008>>

书籍目录

第一部分 SQL Server 概览

第1章 SQL Server 2008 概述

1.1 SQL Server 2008 愿景

1.1.1 企业数据平台

1.1.2 超关系数据

1.1.3 动态开发

1.1.4 深入的商业洞察力

1.2 SQL Server 2008 版本

1.3 服务器整合

1.4 小结

第2章 SQL Server 的安装和配置

2.1 SQL Server 安装要求

2.2 升级到SQL Server 2008

2.2.1 规划升级

2.2.2 使用升级顾问

2.2.3 执行升级

2.3 小结

第二部分 企业数据平台

第3章 策略管理

3.1 PM 解决的需求

3.2 PM 组件

3.2.1 管理目标

3.2.2 方面

3.2.3 条件

3.2.4 策略

3.3 示例策略

3.4 PM 管理

3.4.1 策略状态

3.4.2 PM 安全

3.5 小结

第4章 高可用性

4.1 HA 定义

4.2 数据库镜像

4.2.1 数据库镜像的工作方式

4.2.2 用T-SQL 管理数据库镜像

4.2.3 用Management Studio 管理数据库镜像

4.2.4 全文索引和镜像

4.2.5 Service Broker 和数据库镜像

4.2.6 客户端程序和数据库镜像

4.2.7 监控数据库镜像

4.2.8 数据库镜像的性能代价

4.2.9 数据库镜像的局限

4.3 数据库快照和镜像

4.3.1 数据库快照怎样工作

4.3.2 用T-SQL 管理快照

<<深入SQL Server 2008>>

- 4.3.3 在镜像上使用快照时的性能考虑
- 4.3.4 使用和监控数据库快照
- 4.3.5 数据库快照的限制
- 4.4 SQL Server 中的Windows 群集
- 4.5 SQL Server 复制
 - 4.5.1 快照复制
 - 4.5.2 合并复制
 - 4.5.3 事务复制
 - 4.5.4 复制和镜像
- 4.6 减少计划内的停机时间
 - 4.6.1 为运行中的系统添加CPU
 - 4.6.2 为运行中的系统增加内存
 - 4.6.3 执行在线索引操作
 - 4.6.4 为表和索引分区
- 4.7 小结
- 第5章 性能
 - 5.1 管理和监控资源
 - 5.1.1 数据收集器
 - 5.1.2 资源调控器
 - 5.2 优化存储
 - 5.2.1 备份压缩
 - 5.2.2 数据压缩
 - 5.3 提高查询性能
 - 5.3.1 计划指南支持
 - 5.3.2 稀疏列
 - 5.3.3 列集合
 - 5.4 小结
- 第6章 安全
 - 6.1 关闭数据库引擎特性
 - 6.1.1 远程连接
 - 6.1.2 专用管理员连接
 - 6.1.3 .NET Framework
 - 6.1.4 数据库邮件
 - 6.1.5 SQLMail
 - 6.1.6 Service Broker、HTTP 连接和数据库镜像
 - 6.1.7 Web 助手
 - 6.1.8 xp_cmdshell 扩展存储过程
 - 6.1.9 临时远程查询
 - 6.1.10 OLE 自动化扩展存储过程
 - 6.1.11 SMO 和DMO 扩展对象
 - 6.2 主体和安全对象
 - 6.2.1 主体
 - 6.2.2 安全对象
 - 6.3 权限
 - 6.3.1 权限类型
 - 6.3.2 权限管理
 - 6.4 代码访问安全性

<<深入SQL Server 2008>>

- 6.4.1 命令式和声明式CAS
- 6.4.2 在SQL Server 中使用CAS
- 6.5 SQL Server 2008 审核
 - 6.5.1 在何处写审核数据
 - 6.5.2 审核什么
 - 6.5.3 审核示例
 - 6.5.4 管理审核
- 6.6 小结
- 第7章 SQL Server 加密
 - 7.1 加密密钥
 - 7.1.1 服务主密钥
 - 7.1.2 数据库主密钥
 - 7.1.3 非对称密钥
 - 7.1.4 证书
 - 7.1.5 对称密钥
 - 7.2 透明数据加密
 - 7.2.1 启用TDE
 - 7.2.2 比较TDE 和列级加密
 - 7.3 可扩展密钥管理
 - 7.4 不用密钥加密
 - 7.5 散列和签名数据
 - 7.6 安全目录视图
 - 7.7 查询效率
 - 7.8 小结
- 第8章 自动化和监控
 - 8.1 SQL Server Agent
 - 8.1.1 安排代理作业日程
 - 8.1.2 执行代理作业的权限
 - 8.1.3 代理账户
 - 8.1.4 共享作业计划
 - 8.1.5 记录代理作业步骤的输出
 - 8.1.6 WMI 事件和代理警报
 - 8.1.7 代理性能计数器
 - 8.1.8 代理升级
 - 8.2 维护计划
 - 8.2.1 安排维护子计划的日程
 - 8.2.2 管理维护计划连接
 - 8.2.3 报告和记录维护计划
 - 8.2.4 定义维护计划任务
 - 8.3 SQLCMD
 - 8.3.1 连接到SQL Server
 - 8.3.2 传递变量
 - 8.3.3 使用专用管理员连接
 - 8.3.4 创建脚本
 - 8.4 SQL Server 的PowerShell
 - 8.4.1 PowerShell 简介
 - 8.4.2 使用SQL Server PowerShell

<<深入SQL Server 2008>>

8.5 数据库邮件

8.5.1 配置数据库邮件

8.5.2 发送邮件

8.6 SQL Profiler

8.6.1 关联到性能监控

8.6.2 显示计划

8.6.3 死锁可视化

8.7 扩展事件

8.7.1 扩展事件组件

8.7.2 扩展事件示例：探测死锁

8.8 小结

第9章 Service Broker

9.1 什么是Service Broker

9.1.1 Service Broker 构架

9.1.2 Service Broker 场景

9.2 创建Service Broker 应用

9.2.1 启用Service Broker

9.2.2 创建消息类型

9.2.3 创建协定

9.2.4 创建队列

9.2.5 创建服务

9.2.6 创建Service Broker 存储过程

9.2.7 一个简单的Service Broker示例

9.3 Service Broker 路由和安全

9.3.1 创建分布式的Service Broker应用

9.3.2 分布式Service Broker 的例子

9.4 消息优先级

9.5 使用SSBDiagnose 对Service Broker进行故障检测

9.6 小结

第10章 整合全文搜索

10.1 创建全文检索目录和索引

10.1.1 使用图形化工具创建全文目录和索引

10.1.2 使用T-SQL 创建全文目录和索引

10.2 使用iFTS 查询

10.2.1 FREETEXT 谓词搜索

10.2.2 CONTAINS 谓词搜索

10.2.3 FREETEXTTABLE 和CONTAINSTABLE函数搜索

10.3 管理词库文件

10.3.1 编辑词库文件

10.3.2 重新加载词库

10.4 使用非索引字表

10.5 搜索文档

10.5.1 为文档创建全文索引

10.5.2 查询文档

10.6 管理iFTS

10.7 小结

第三部分 SQL Server 开发

<<深入SQL Server 2008>>

第11章 SQL Server 2008 中的新数据类型

11.1 SQL Server 2008 中对空间数据的支持

11.1.1 GEOMETRY 类型

11.1.2 GEOGRAPH 类型

11.2 时间类型的改变

11.2.1 新的日期和时间数据类型

11.2.2 新的日期和时间系统函数

11.3 新的层次结构数据类型

11.4 文件流支持

11.4.1 启用文件流功能

11.4.2 文件流示例

11.5 小结

第12章 针对开发人员的T-SQL 改进

12.1 DML 特性

12.1.1 弃用旧式的外部连接

12.1.2 公共表表达式

12.1.3 TOP

12.1.4 扩展FROM 子句

12.1.5 OUTPUT

12.1.6 排序函数

12.1.7 EXCEPT 和INTERSECT

12.1.8 别名

12.1.9 MERGE

12.2 一般性的开发

12.2.1 错误处理

12.2.2 .WRITE 对UPDATE 语句的扩展

12.2.3 EXECUTE

12.2.4 代码安全上下文

12.2.5 .NET 声明

12.2.6 声明和设置变量

12.2.7 传递表值参数

12.3 小结

第13章 T-SQL 为DBA 所做的改进

13.1 锁的改进

13.2 元数据视图

13.2.1 兼容性视图

13.2.2 目录视图

13.2.3 动态管理视图和函数

13.3 SQL Server 性能监视器

13.4 DDL 触发器

13.4.1 创建和修改DDL 触发器

13.4.2 删除DDL 触发器

13.4.3 启用和禁用DDL 触发器

13.4.4 使用目录视图枚举DDL触发器

13.4.5 使用eventdata()函数编用DDL 触发器

13.5 索引及其性能增强

13.5.1 联机索引

<<深入SQL Server 2008>>

- 13.5.2 创建索引期间的锁控制
- 13.5.3 创建包含额外列的索引
- 13.5.4 修改索引
- 13.5.5 使用筛选索引
- 13.5.6 使用筛选统计信息
- 13.5.7 统计相关的日期时间列
- 13.5.8 为第三排序规则排序的性能改进
- 13.5.9 表和索引分区
- 13.5.10 使用索引视图
- 13.5.11 使用分区对齐索引视图
- 13.5.12 持久化计算列
- 13.6 快照
 - 13.6.1 SNAPSHOT 的隔离级别
 - 13.6.2 数据库快照
- 13.7 数据集成的改进
 - 13.7.1 验证数据库页
 - 13.7.2 使数据库进入应急状态
- 13.8 小结
- 第14章 .NET 集成
 - 14.1 SQL Server .NET 集成简介
 - 14.1.1 SQL Server 为什么托管CLR
 - 14.1.2 何时使用CLR 例程
 - 14.1.3 何时不使用CLR 例程
 - 14.1.4 SQL Server 如何托管.NET : 构架概览
 - 14.2 SQL Server .NET 编程模型
 - 14.2.1 SQL Server 托管的ADO.NET增强
 - 14.2.2 SQL Server 的.NET 命名空间概览
 - 14.3 编写CLR Stored Procedure
 - 14.3.1 开始一个Visual Studio 2008SQL Sever Project
 - 14.3.2 解剖Stored Procedure
 - 14.3.3 添加参数
 - 14.3.4 定义问题
 - 14.3.5 使用SqlPipe
 - 14.3.6 将所有这些组织在一起 : 编写Stored Procedure 体
 - 14.3.7 测试Stored Procedure
 - 14.3.8 调试Stored Procedure
 - 14.3.9 在CLR 例程中抛出异常
 - 14.4 部署CLR 例程
 - 14.5 小结
- 第15章 编写程序集
 - 15.1 CLR 用户自定义类型
 - 15.1.1 用户自定义类型的应用程序
 - 15.1.2 添加用户自定义类型到SQLServer 项目中
 - 15.1.3 User-Defined Type 的组成部分
 - 15.1.4 一个简单的例子 : PhoneNumber 类型
 - 15.1.5 另一个例子 : 在StringArray类型
 - 15.1.6 管理用户自定义类型

<<深入SQL Server 2008>>

15.2 CLR 用户自定义函数

- 15.2.1 添加用户自定义函数到Visual Studio 项目
- 15.2.2 Visual Studio 2008 User-Defined Function 模板
- 15.2.3 SqlFunction 特性
- 15.2.4 标量用户自定义函数
- 15.2.5 表值型用户自定义函数
- 15.2.6 管理CLR 用户自定义函数

15.3 CLR 用户自定义聚

- 15.3.1 添加用户自定义聚合到SQLServer 项目中
- 15.3.2 用户自定义聚合的组成部分

15.4 CLR 用户自定义触发器

- 15.4.1 向SQL Server 项目中添加CLR 用户自定义触发器
- 15.4.2 编写CLR 触发器
- 15.4.3 管理用户自定义触发器

15.5 管理程序集

15.6 小结

第16章 SQL Server 与XML

16.1 什么是XML

16.2 什么是XPath 和XMLDOM

- 16.2.1 XPath 语法
- 16.2.2 XPath 函数
- 16.2.3 XMLDOM: XML 文档对象模型
- 16.2.4

XPathDocument、XPathNavigator和XPathExpression类

16.3 XML 存储到数据库

- 16.3.1 支持SOAP 的SQL Server配置
- 16.3.2 OPENXML
- 16.3.3 使用注释XML 模式的XML视图
- 16.3.4 SQLXML Updategram
- 16.3.5 XML 大容量加载

16.4 从数据库取出XML

- 16.4.1 FOR XML
- 16.4.2 利用模板改善性能

16.5 使用XML 数据

- 16.5.1 “ any ” 类型验证
- 16.5.2 日期和时间支持
- 16.5.3 联合和列表类型

16.6 .NET 和COM中使用SQLXML编程

- 16.6.1 SQLXML 类
- 16.6.2 SQLXML 编码示例

16.7 小结

第17章 SQL Server XML 和XQuery支持

17.1 使用XML 数据类型

- 17.1.1 了解SQL Server 如何存储XML
- 17.1.2 创建XML 列
- 17.1.3 为模式创建设置权限
- 17.1.4 约束XML 列

<<深入SQL Server 2008>>

- 17.1.5 检查XML 数据类型限制
- 17.2 在XML 列中插入数据
 - 17.2.1 使用SSIS 插入XML 数据
 - 17.2.2 大容量加载XML
 - 17.2.3 编写自定义查询或应用
- 17.3 查询XML 数据
 - 17.3.1 XQuery 101
 - 17.3.2 基本XML 查询方法
 - 17.3.3 跨域查询
- 17.4 修改XML 数据
 - 17.4.1 插入元素
 - 17.4.2 删除元素
 - 17.4.3 改变节点值
 - 17.4.4 XML 修改的限制
- 17.5 建立XML 索引以提高性能
 - 17.5.1 了解XML 索引如何工作
 - 17.5.2 分析次级XML 索引
- 17.6 全文搜索和XML 数据类型
- 17.7 目录视图与XML
- 17.8 应用与XML
- 17.9 XML Web 服务支持
 - 17.9.1 创建端点
 - 17.9.2 使用高级Web 服务
 - 17.9.3 监视XML Web 服务的性能
- 17.10 小结
- 第18章 LINQ to SQL
 - 18.1 对象关系映射
 - 18.2 实体生成工具
 - 18.2.1 使用SQLMetal
 - 18.2.2 使用Visual Studio 的LINQ toSQL 类设计器
 - 18.2.3 分析生成的代码
 - 18.3 运行LINQ 查询
 - 18.3.1 Select 操作符
 - 18.3.2 Where 操作符
 - 18.3.3 Join 操作符
 - 18.3.4 OrderBy 操作符
 - 18.4 DataContext 类
 - 18.4.1 插入、更新和删除记录
 - 18.4.2 并发性冲突检测
 - 18.4.3 延迟查询
 - 18.4.4 延迟加载
 - 18.4.5 执行存储过程和用户自定义函数
 - 18.5 小结
- 第四部分 SQL Server 商业智能
 - 第19章 Report Services
 - 19.1 Reporting Service 组件
 - 19.1.1 Report Server Service

<<深入SQL Server 2008>>

- 19.1.2 元数据目录
- 19.1.3 BIDS 中的Report Designer
- 19.1.4 Report Designer Preview
- 19.1.5 SQL Server ManagementStudio 集成
- 19.1.6 Reporting Services 配置管理器
- 19.1.7 Reporting Services 安全性
- 19.2 创建一个基本报表
 - 19.2.1 启动设计器
 - 19.2.2 使用数据源和数据集
 - 19.2.3 报表布局与预览
 - 19.2.4 使用表达式
 - 19.2.5 部署报表
- 19.3 报表设计的高级功能
 - 19.3.1 多值参数
 - 19.3.2 选择日期值的DatePicker
 - 19.3.3 交互式排序
 - 19.3.4 Analysis Services 集成
 - 19.3.5 Reporting Services 数据源
 - 19.3.6 自定义报表项
- 19.4 Visual Studio 集成与ReportViewer控件
 - 19.4.1 使用WinForm 控件
 - 19.4.2 通过编程使用ReportViewer控件
 - 19.4.3 LocalReport 和ServerReport对象
- 19.5 SharePoint 集成
- 19.6 最终用户即席查询和报表
 - 19.6.1 报表生成器客户端
 - 19.6.2 报表模型和语义模型定义语言
- 19.7 报表呈现
 - 19.7.1 导出报表
 - 19.7.2 呈现大报表
- 19.8 数据区
 - 19.8.1 Tablix 数据区
 - 19.8.2 更新后的Chart 数据区
- 19.9 小结
- 第20章 分析服务
 - 20.1 SQL Server 2008 中Analysis Services的新特性
 - 20.1.1 设计工具的改进
 - 20.1.2 监视工具的改进
 - 20.1.3 运行时改进
 - 20.2 Analysis Services 基础知识
 - 20.2.1 体系结构
 - 20.2.2 开发环境
 - 20.2.3 Analysis Services 对象和概念
 - 20.3 OLAP、OLTP 和数据仓库
 - 20.3.1 OLAP 和OLTP 配置
 - 20.3.2 OLAP 概念
 - 20.4 Analysis Services 项目

<<深入SQL Server 2008>>

- 20.4.1 新建一个Analysis Services项目
- 20.4.2 定义数据源
- 20.4.3 编辑数据源
- 20.4.4 定义数据源视图
- 20.4.5 定义多维数据集
- 20.4.6 配置维度
- 20.4.7 部署项目
- 20.5 使用多维数据集
 - 20.5.1 查看多维数据集结构
 - 20.5.2 浏览多维数据集
 - 20.5.3 管理显示的数据
 - 20.5.4 执行计算
 - 20.5.5 使用KPI
- 20.6 使用Analysis Services 脚本语言
 - 20.6.1 创建一个包含命名查询的DSV
 - 20.6.2 查看ASSL
- 20.7 小结
- 第21章 集成服务
 - 21.1 SSIS 功能概述
 - 21.1.1 何时使用SSIS
 - 21.1.2 SSIS 的新特性
 - 21.2 SSIS 集成开发环境
 - 21.2.1 在Management Studio 中连接到SSIS
 - 21.2.2 在BIDS 中创建一个新的SSIS 项目
 - 21.3 SSIS 基础知识
 - 21.3.1 SSIS 设计器概述
 - 21.3.2 数据流示例
 - 21.3.3 事件处理程序设计界面
 - 21.4 控制流和数据流设计器任务
 - 21.4.1 控制流容器和任务
 - 21.4.2 数据流设计器任务
 - 21.5 变更数据捕获
 - 21.5.1 启用变更数据捕获
 - 21.5.2 用SSIS 提取变更数据
 - 21.6 日志
 - 21.7 动态包配置
 - 21.8 变量
 - 21.8.1 配置变量
 - 21.8.2 创建变量
 - 21.9 优先级约束
 - 21.10 检查点
 - 21.11 事务
 - 21.12 调试
 - 21.12.1 控制流和数据流可视化调试
 - 21.12.2 数据查看器
 - 21.12.3 断点
 - 21.12.4 其他调试窗口

21.13 SSIS 包处理

21.13.1 SSIS 包部署实用工具

21.13.2 迁移SQL Server 2000 DTS包

21.13.3 调度SSIS 包

21.14 小结

<<深入SQL Server 2008>>

媒体关注与评论

“ 本书名副其实，是一本深入SQLServer2008的好书。
书中介绍的SQLServer管理特性，正是我工作中需要用到的。
我会一直把它放在书架上的。

” &ldquo;作为一本深入介绍SQLServer2008的书，它帮我轻松实现从SQLServer2000和SQLServer2005到SQLServer2008的升级工作，使我能集中精力探索其他新功能。
&rdquo;

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>