

# SYBASE SQL SERVER 11

## 管理员指南



[美] John Kirkwood 著  
希望图书创作室 译

 海洋出版社

# **SYBASE SQL SERVER 11**

## **管理员指南**

〔美〕 John Kirkwood 著  
希望图书创作室 译  
华卫文 审订

**海 洋 出 版 社**

1998 年 · 北京

## 内 容 简 介

本书较全面地介绍了 SQL Server System 11.0。全书分两部分。第一部分由第一章至第十三章构成,概述了 SQL Server System 11.0 的新特点,其中第一章至第十一章介绍了 System10 的特点,第十二章至第十三章介绍 System 11 的新特点。第二部分由附录 A—D 构成,概述了 SQL Server System 11.0 未曾变化的特点,其中附录 A 和附录 B 为系统表和命令/系统进程语法的完整清单。

欲得到本书技术支持的读者,请与(010)62562329,(010)62541992 联系,或传真至(010)62561057。

## 图书在版编目(CIP)数据

SYBASE SQL SERVER 11 管理员指南/(美)柯克伍德(Kirkwood,J.)  
著;希望图书创作室译. —北京:海洋出版社,1998. 10  
ISBN 7-5027-4593-9

I. S… II. ①柯… ②希… III. 服务器-数据库系统, SybaseSQLServer11-  
指南 N. TP311.13

中国版本图书馆 CIP 数据核字(98)第 18270 号

责任编辑:李 勤

海洋出版社出版发行

(100081 北京市海淀区大慧寺路 8 号)

北京 施园印刷厂印刷 新华书店发行所经销

1998 年 10 月第 1 版 1998 年 10 月第 1 次印刷

开本:787×1092 1/16 印张:25.75

字数:720 千字 印数:1—5000 册

定价:45.00 元

图字:01-98-1115 号

海洋版图书印、装错误可随时退换

12-2-11

## 前 言

很高兴有机会介绍 Sybase Press 出版的第一本书——John Kirkwood 所著的 Sybase SQL Server 11 管理员指南。此书及时地提供了最新的关系数据库管理系统 Sybase SQL Server 11 的全面介绍。SQL Server 11 明确地达到专业性、突破性和高品质的设计目标,它是一个强有力的、多侧面的软件,值得用一本厚书来概述从管理分散在各处的企业的数据到执行和调整任务要求苛刻的应用程序。

本书主要针对数据库的开发者和数据库的管理者,John Kirkwood 再一次以他的清晰、风度和精确展示了他解释 Sybase SQL Server 的能力。自从 1987 年以来,John 已经是一个 Sybase SQL Server 的熟练用户,一个很有口才的建议者和很有贡献的教师。John 将以他的热情和幽默以及作为骨干,按照用户在逻辑上可能遇到的问题,根据他在实践中所积累的经验来介绍 SQL Server 的特点。John 从来不会因他对我们的产品的长期的精通而使他挑选新特点的突出方面的能力黯然失色,他会告诉用户首先做什么,不该做什么。John 以他多年的经验,选择了那些最能补充标准 Sybase 文档的训练作为专题和示范。

的确,这本书对于 Sybase 用户通信来说,作为他积极参与国际 Sybase 用户小组和增加委员会(International Sybase User Group and its Enhancements Committee),这是 John 所做的贡献的又一个例证。在我们最好的客户提倡者之一——John Kirkwood 的这本书的前面,让我们说上这些话,我们感到很荣幸。

Sybase 公司总裁

# 导 言

本书全面介绍了 SQL Server System 11.0。本书由两部分构成：一部分描述自 4.9.2 版本以来的所有新特点；第二部分介绍 4.9.2 版本的主要特点。因此，如果用户已经了解了 SQL Server 是如何工作的，首先阅读第一部分，然后再浏览第二部分。如果用户是一个 SQL Server 的新用户，在阅读第一部分之前，请从第二部分开始，这部分解释了 SQL Server 的基本内容。阅读本书可能没有多大的困难，但是用户可能会发现缺乏一些逻辑性。

## 结构

第一部分由第一章至第十三章构成，概述了新的特点；第二部分由附录 A—D 构成，概述了没有变化的特点。第一章至第十一章介绍了 System 10 的特点，第十二章和第十三章介绍了 System 11 的新特点。附录 A 和附录 B 为系统表和命令/系统进程语法的完整清单。

## 各章内容

### 第一章 声明完整性

本章讨论 System 10 采用的新的声明完整性特点，它允许在创建表命令中定义域和参考完整性。每个新从句的语法和用法都要进行详细讨论。在一般介绍完整性之后，与 System 10 执行有关的完整性都用加亮外表表示，它们不会出现在声明的完整性从句之中。然后，以某些范例的形式给出所有创建表语法。本章以新声明从句和已存在的未声明的缺省完整性、规则和触发器的比较作为结束。

### 第二章 光标

本章描述 System 10 所支持的光标功能，用它可以连续存取从数据库的第一行到最后一行由 SQL 语句返回的一行串。本章解释了步入行串时使用循环和使用光标之间的不同，接下来描述 System 10 能够认可的光标类型。然后以基于 SQL Server 光标和基于 ct-library client 光标的示范详细介绍光标命令。最后讨论光标锁定。

### 第三章 数据类型

本章概述 System 10 和 System 11 可能获得的数据类型。

### 第四章 阈值管理器

本章描述 System 10 所采用的阈值管理。它可以在一个段的自由空间下降到一个指定量时，打开一个存储过程。因为要在段上定义阈值，所以在本章中要对段进行讨论。

### 第五章 稽查

本章描述 System 10 的稽查。介绍组件和安装之后，详细讨论各种稽查选项的设置和使用。

### 第六章 安全

本章描述 System 10 安全系统，其中包括设置适当的名称、ID 和能够注册到服务器上的保密字。一旦注册，就可以使用数据库。新的管理规则、命令的授权和废除以及用户访问数据库的特权均要进行讨论。最后描述在服务器之间配置远程访问。

## 第七章 Sybase System 10 的新特点

本章描述 Sybase 增加到 System 10 中的全部新特点。它们包括事务状态、反转触发器、触发器自身循环、按时间排序、联合密度以及定制出错消息。在本章中,还包括 SET 命令选项。

## 第八章 新的 ANSI 特点

本章描述了 System 10 增加的符合 ANSI 89 标准的全部新特点。就一般情况而言,它们可以在 4.9.2 版本之间进行交换,为了使用更简便,一般都把它们设置为缺省值。原则上的特点约束于事务的模式、锁定的隔离级别、复选项的考虑、空值的测试、过程的定义以及子查询的变更。

## 第九章 备份与恢复

本章讨论 System 10 的备份服务。备份服务可以控制数据的转储和存储,可以控制确保由于失败而全部恢复的事务日志。为了描述事务日志项,从介质失败和系统失败恢复时所采取的措施,我们先来讨论事务管理,之后详细描述能够采用的命令和过程。

## 第十章 优化器

本章描述 System 10 的优化器。因为编号的变化,为了突出重点题目,本书不是仅列出 System 10 特点的清单,而是论述全部项目。本章首先给出解释如何优化的变化项的定义。随后是两个详细优化的章节:一个是简单的对单表和联合进行优化的计划;另一个是对比较复杂的隐式语句的计划。

## 第十一章 出现问题怎么办? 问题的预料和解决

本章收集了配置、监视和解决问题的技术和有关 System 10 的工具。

## 第十二章 System 11 的新特点

本章描述 System 11 采用的新特点,它们包括数据隐藏和缓冲区管理、分割累积表、优化改进、放置规则修正、家务会晤、页面最大的行数以及锁定增进阈值。

## 第十三章 System 11 配置参数

本章描述 System 11 新的 sp—configure 配置参数。

## 附录 A 分配空间和创建数据库

本附录描述对磁盘空间的初始化、对数据库和日志进行定位等内容,其中包括磁盘镜像和使用缺省设备的内容。同时介绍数据库和日志的创建和扩充,如何判断所需规格并加以处理。

## 附录 B 主数据库与逻辑系统重建

本附录描述重建主数据库和完成一个完整系统重建的必要条件。

## 附录 C 存储过程和索引

本附录采用了详查 SQL Server B 树索引结构的方法:分簇和非分簇。本章概述了数据存储结构、索引结构以及涉及到插入/修正/删除的处理过程,而不采用索引的联合。本章还将讨论页面结构的其他类型——分配、分发和文本/图像。

## 附录 D 多用户考虑:一致性与存储器

本附录描述 SQL Server 接近多用户环境的重要外观,诸如协调和锁定以及存储器的分配等内容。

## 本书主旨

虽然我已经有了 4.9.2 版本这本书,但是本书所使用的材料并不是已有著作的改头换面地重复,它是根据我实际使用 System 10 得出的,最初用 System 11.1 工作不得不考虑如何使

用新的特点,随着我对出现的问题的研讨,我找到了解决方法。它不是软件缺陷的一个清单(有一点点),而是关于如何使用新特点的建议和我个人对它们优劣的见解。我总是说:你可能同意也可能不同意我的观点,介绍我希望它会引起对潜在问题的重视,提醒用户既是最简单的语句也要进行检查。

虽然我希望你欣赏这本书并认为它是实用的,但是如果你有任何评价——那怕是一点点你认为是不正确的地方,请与我联系,我将洗耳恭听。

最后,我要感谢 Jim Panttaja 先生对早期材料的收集,同时还要对给予技术评价和订正的 Sybase 的各位同仁表示感谢,他们是 Karen Paulsell, Scott McCargor, Marc Sugiyama, Brijesh Agarwal, Judy Bowman, Robert Garvey, dan Scanlan。

John Kirkwood

# 目 录

|                           |      |
|---------------------------|------|
| <b>第一章 声明完整性</b> .....    | (1)  |
| 1.1 基于图表的声明的完整性 .....     | (1)  |
| 1.2 完整性原理 .....           | (8)  |
| 1.3 创建表语法.....            | (10) |
| 1.4 与未声明完整性的对比.....       | (14) |
| 1.5 小结.....               | (15) |
| <b>第二章 光标</b> .....       | (16) |
| 2.1 光标是什么.....            | (16) |
| 2.2 SQL Server 光标类型 ..... | (18) |
| 2.3 服务器光标命令.....          | (19) |
| 2.4 修改与删除扩展.....          | (27) |
| 2.5 基于客户的光标命令.....        | (28) |
| 2.6 光标锁定.....             | (30) |
| 2.7 小结.....               | (32) |
| <b>第三章 数据类型</b> .....     | (33) |
| 3.1 新的数据类型.....           | (33) |
| 3.2 算法中的精度与标量.....        | (36) |
| 3.3 转换.....               | (38) |
| 3.4 改变系统表.....            | (38) |
| 3.5 改变函数.....             | (39) |
| 3.6 小结.....               | (39) |
| <b>第四章 阈值管理器</b> .....    | (40) |
| 4.1 简介.....               | (40) |
| 4.2 段.....                | (41) |
| 4.3 阈值.....               | (45) |
| 4.4 新函数.....              | (49) |
| 4.5 变更现存的过程.....          | (50) |
| 4.6 滞留值.....              | (50) |
| 4.7 系统表.....              | (52) |
| 4.8 日志段上的阈值替换.....        | (52) |
| 4.9 小结.....               | (53) |
| <b>第五章 稽查</b> .....       | (55) |
| 5.1 简介.....               | (55) |

|            |                                    |              |
|------------|------------------------------------|--------------|
| 5.2        | 安装.....                            | (56)         |
| 5.3        | 删除稽查.....                          | (58)         |
| 5.4        | 稽查级别.....                          | (58)         |
| 5.5        | 系统表.....                           | (65)         |
| 5.6        | 归档稽查数据.....                        | (66)         |
| 5.7        | 小结.....                            | (66)         |
| <b>第六章</b> | <b>安全 .....</b>                    | <b>(67)</b>  |
| 6.1        | 简介.....                            | (68)         |
| 6.2        | 用户的级别.....                         | (69)         |
| 6.3        | 注册.....                            | (70)         |
| 6.4        | 角色.....                            | (75)         |
| 6.5        | 用户.....                            | (77)         |
| 6.6        | 小组.....                            | (80)         |
| 6.7        | 系统表.....                           | (81)         |
| 6.8        | 特权.....                            | (85)         |
| 6.9        | 访问系统过程.....                        | (95)         |
| 6.10       | 系统表 .....                          | (96)         |
| 6.11       | 管理远程访问 .....                       | (97)         |
| 6.12       | 小结.....                            | (101)        |
| <b>第七章</b> | <b>Sybase System 10 的新特点 .....</b> | <b>(102)</b> |
| 7.1        | 安装 .....                           | (102)        |
| 7.2        | 事务状态 .....                         | (102)        |
| 7.3        | 返回触发器 .....                        | (103)        |
| 7.4        | 扩展错误信息 .....                       | (106)        |
| 7.5        | 网络连接系统表 .....                      | (107)        |
| 7.6        | 触发器自循环 .....                       | (107)        |
| 7.7        | 分类调谐 .....                         | (108)        |
| 7.8        | 联合的优化 .....                        | (110)        |
| 7.9        | SET 命令选项 .....                     | (112)        |
| 7.10       | 定制错误消息.....                        | (116)        |
| 7.11       | 小结.....                            | (117)        |
| <b>第八章</b> | <b>新的 ANSI 特性.....</b>             | <b>(118)</b> |
| 8.1        | 事务模式 .....                         | (118)        |
| 8.2        | 隔离级别 .....                         | (122)        |
| 8.3        | ANSI 兼容性 .....                     | (125)        |
| 8.4        | 规划 .....                           | (130)        |
| 8.5        | 子查询的变化 .....                       | (131)        |
| 8.6        | 小结 .....                           | (134)        |
| <b>第九章</b> | <b>备份和恢复.....</b>                  | <b>(136)</b> |

|             |                                |              |
|-------------|--------------------------------|--------------|
| 9.1         | 介绍 .....                       | (136)        |
| 9.2         | 安装 .....                       | (138)        |
| 9.3         | 事务管理 .....                     | (138)        |
| 9.4         | 恢复 .....                       | (142)        |
| 9.5         | 备份过程 .....                     | (146)        |
| 9.6         | 卷标控制要求 .....                   | (154)        |
| 9.7         | 恢复过程 .....                     | (155)        |
| 9.8         | 小结 .....                       | (158)        |
| <b>第十章</b>  | <b>优化器 .....</b>               | <b>(159)</b> |
| 10.1        | 介绍 .....                       | (159)        |
| 10.2        | 基本术语 .....                     | (160)        |
| 10.3        | 基本的优化计划 .....                  | (171)        |
| 10.4        | 嵌套查询 .....                     | (194)        |
| 10.5        | 实践与理论 .....                    | (203)        |
| 10.6        | 存储过程优化 .....                   | (206)        |
| 10.7        | 小结 .....                       | (209)        |
| <b>第十一章</b> | <b>出现问题怎么办? 问题的预料和解决 .....</b> | <b>(210)</b> |
| 11.1        | 服务器配置 .....                    | (210)        |
| 11.2        | 参数 .....                       | (212)        |
| 11.3        | 监视系统活动 .....                   | (220)        |
| 11.4        | 存储过程 .....                     | (223)        |
| 11.5        | DBCC .....                     | (225)        |
| 11.6        | 小结 .....                       | (245)        |
| <b>第十二章</b> | <b>System 11 新的特点 .....</b>    | <b>(246)</b> |
| 12.1        | 数据高速缓存管理 .....                 | (246)        |
| 12.2        | 分区表 .....                      | (255)        |
| 12.3        | 优化器 .....                      | (258)        |
| 12.4        | 修正 .....                       | (270)        |
| 12.5        | 家务会晤 .....                     | (273)        |
| 12.6        | 每页最大的行 .....                   | (273)        |
| 12.7        | 锁升级阈值 .....                    | (274)        |
| 12.8        | 小结 .....                       | (275)        |
| <b>第十三章</b> | <b>System 11 配置参数 .....</b>    | <b>(276)</b> |
| 13.1        | 介绍 .....                       | (276)        |
| 13.2        | 参数 .....                       | (277)        |
| 13.3        | 配置参数细节 .....                   | (282)        |
| 13.4        | 小结 .....                       | (294)        |
| <b>附录 A</b> | <b>分配空间和创建数据库 .....</b>        | <b>(295)</b> |
| A.1         | 简介 .....                       | (295)        |

|             |                            |              |
|-------------|----------------------------|--------------|
| A. 2        | 磁盘初始化 .....                | (296)        |
| A. 3        | 设备镜像 .....                 | (299)        |
| A. 4        | 缺省设备 .....                 | (302)        |
| A. 5        | 数据库的创建 .....               | (303)        |
| A. 6        | 数据尺寸的计算 .....              | (307)        |
| A. 7        | 索引尺寸的计算 .....              | (309)        |
| A. 8        | 自由空间 .....                 | (309)        |
| A. 9        | 日志尺寸的计算 .....              | (310)        |
| A. 10       | 监视空间的使用 .....              | (310)        |
| A. 11       | 扩展数据库 .....                | (312)        |
| A. 12       | 扩展日志 .....                 | (313)        |
| A. 13       | 数据库的删除 .....               | (313)        |
| A. 14       | 数据库的环境 .....               | (314)        |
| A. 15       | 系统表 .....                  | (315)        |
| A. 16       | 系统表中一些有用的提取 .....          | (322)        |
| A. 17       | 小结 .....                   | (323)        |
| <b>附录 B</b> | <b>主数据库与逻辑系统重建 .....</b>   | <b>(324)</b> |
| B. 1        | 重建主数据库 .....               | (324)        |
| B. 2        | Sybsystemprocs 的重建 .....   | (327)        |
| B. 3        | 逻辑系统重建 .....               | (328)        |
| B. 4        | 小结 .....                   | (333)        |
| <b>附录 C</b> | <b>存储结构与索引 .....</b>       | <b>(335)</b> |
| C. 1        | 索引 .....                   | (335)        |
| C. 2        | 页与记录布局 .....               | (340)        |
| C. 3        | 缓冲高速缓存器管理 .....            | (350)        |
| C. 4        | 索引处理 .....                 | (351)        |
| C. 5        | 页分离 .....                  | (356)        |
| C. 6        | 碎片 .....                   | (358)        |
| C. 7        | 小结 .....                   | (359)        |
| <b>附录 D</b> | <b>多用户考虑:一致性和存储器 .....</b> | <b>(360)</b> |
| D. 1        | 加锁 .....                   | (360)        |
| D. 2        | 服务器加锁 .....                | (360)        |
| D. 3        | 客户/服务器考虑 .....             | (366)        |
| D. 4        | 管理的考虑 .....                | (368)        |
| D. 5        | 计划考虑 .....                 | (369)        |
| D. 6        | 存储分配 .....                 | (372)        |
| D. 7        | 小结 .....                   | (375)        |
|             | SyBooks 安装指南 .....         | (377)        |

# 第一章 声明完整性

本章讨论 System 10 的新的声明完整性,这种特性允许在创建表命令中定义局部的和参考的完整性。开始先详细讨论新从句的语法和用法,然后讨论一般性的处理方法。关于 System 10 的执行方法,没有涉及到的声明完整性的从句用变亮办法显示。全部创建表的语法用某些范例给出。最后比较新声明和已经有的未声明的缺省完整性、规则和触发器相。

System 11 为了定义表增加了一致性的属性特征和每页的最大行数。

## 命令语法

alter table

create table

## 系统过程语法

sp\_addtype

sp\_addmessage

sp\_bindmsg

sp\_configure

sp\_dboption

sp\_helpconstraint

## 1.1 基于图表的声明的完整性

### 1.1.1 创建表的新从句

在定义表中已经引用了完整性约束。由于它们不涉及任何编程,所以这样的约束更新称为声明;换句话说,用户可以简便地对它们进行说明。它们可以作为缺省从句、约束从句、独特的从句、首选关键字从句、参考从句和核查从句出现在创建表命令中。所声明的这些完整性约束可以完全替代规则和缺省值。然而,那些外来的关键字参考从句宁可受范围的限制和缺乏广泛性,也不愿意使用用户可以在触发器中指定的参考完整性。由于这个原因,将不得不对触发器采用补充。依我的观点,首选关键字约束应该在创建表中进行定义,而规则和缺省值应该分离出来,也就是说,分别的项目应该使用创建规则和创建缺省值来定义。声明参考从句应该用于那些(现存的和受限制的)定义为参考的完整性,应该用于提供从句和无效的请求的触发器(其原因我们将在 1.4 节中进行详细讨论)。无论是否同意我的看法,都必须考虑这些情况并确保为了定义 System 10 中的完整性能够执行,要有相伴的标准。

总之,声明完整性涉及到了下面几个从句:

default        指定当一个记录在没有指派列时所需使用的列缺省值。

identity       指定某列包含了一个顺序号,而它是为了唯一标识各行,由服务器产生的顺序号。实际上,它不是完整性特征,只是新的 create table 语法将其包括在其

中。

constraint 指定列或表级的完整性约束,它可能含有下面的从句。

unique 为某列指定唯一的非空值。

primary key 指定可以编排表里面各行的唯一标识符(primary key)的一列或多列。

references 为单个列指定外来关键字参考的完整性。

foreign key... references...

为表级上的一个列或多个列指定外来关键字参考的完整性。check 给列或表级指定一个确定的规则。

虽然这里显得有点乱,最好是以新从句语法开始,我将在后面的 1.3 节的创建表中介绍。

### 1.1.2 缺省

```
DEFAULT { constant_expr | user | null }
```

注释:

---

constant\_expr 不可能参考任何的数据库对象,可能包括函数。

user 插入用户名字。

null 插入空值。

---

当一个列在插入状态下没有给出值时,这个语句将为这个列指定缺省的值。例如:

```
DEFAULT 'ZZZZ'
```

```
DEFAULT getdate()
```

```
DEFAULT user
```

这一列的数据类型必须与缺省值一致。对于 user 来说,数据类型必须是 char(30)或 varchar(30)。

在 identity 列上声明一个缺省值不会起作用。可识别的列总是由服务器指定,这个列具有顺序编号。根据我的看法,在这种情况下,最好能有一个出错消息,或者至少给出一种警告,因为你可能在某些地方出现了错误。要注意这种情况,仔细核查标识列。

### 1.1.3 一致性属性

```
IDENTITY | null | not null
```

它表明这个列具有了一致性属性,这种属性意味着它含有一个由服务器产生的唯一确定的行的顺序号(注意:对于这一列来说,可以替换 null | not null 说明符,当它不能为空时,是不无道理的)。

在一个表中只有一个列可以有一致性属性。这一列不能自动地赋与空值,不能进行修正。这列的数据类型必须是标准为 0 的数字(有关这种新数据类型的解释,参见第 3.1.1 小节中的内容)。对于一列来说,一致性的取值没有什么限制;用户可以简便地获得一个错误消息,不会有更多的行插入到表中的。因此要特别小心。因为数字型的数据类型的最大范围是 10 的 38 次方( $10^{38}-1$ ),所以它不会出现太多的问题。但是,如果用户试图用较小的精度来保存,那么用户可能在数据超量时锁死应用程序。

列值启始于 1,并且增量为 1。服务器使用时不会制做一个编号,但是在分配时会将一系列

的值分成连续的块的编号。这种块驻留在存储器中,它是为了在分配编号时改进服务器性能的。然而,如果出现了错误并且使得当前的编号块丢失的话,它允许在编号分配中留下间隔。基于从错误中进行恢复,当前块中可以使用的编号将不再出现,关于表的一致性取值将开始于下一个编号块。SQL Server 称这种方法为“预先加印算法”(pre-burned algorithm)。

如果某一列是按下面这样声明的:

```
a numeric(3, 0) identity
```

而服务器将它一致性值分配在 10 块内,如果服务器在分配 16 值时出现了错误,则 17-20 将不再使用,服务器恢复时,新的一块将从 21 开始。

同样,设置一致性列值的事务回转将按顺序留出间隔。考虑到回转,返回的一致性值将被丢弃,而下一个插入的值将使用下一个可能使用的一致性值。因此,如果返回的一个一致性值是 104,那么 104 将被丢弃,而下一个插入的,即使是同样的一个命令,也将使用 105。

由服务器获得的编号块的大小可以按下面形式配置:

```
sp_configure 'identity burning set factor',value
```

详细的解释参见第 11.2 节。

系统管理员可以通过下面一套 identity\_insert 选项,向一致性列插入外部值。

```
SET identity_insert table_name on
GO
INSERT INTO table_name (column_name)
VALUES (20)
GO
SET identity_insert table_name off
GO
```

当用户插入一个一致性字段值时,用户必须在插入命令中指定列的名称。当 set identity\_insert 没有出现在一致性列的情况下,用户不可能发出插入命令。

```
SET identity_insert orders on
GO
INSERT INTO orders
(order_key, cust_no, order_date)
VALUES (8000, 1234, getdate())
GO
INSERT INTO orders ( cust_no, order_date )
VALUES (1235, getdate())

Msg 585, level 16, state 1, Explicit value must be specified for
identity field in the table 'orders' when IDENTITY_INSERT IS ON.

SET identity_insert orders off
GO
```

如果 order\_key 是一个一致性列,那么当一个一致性值用手动方法插入时,可以产生间隔。当 identity\_insert 属性关闭时,服务器从最大(值)+1 开始。如果当前值是 1~243,并且手动插入的是 329,那么下一个一致性值将使用 330。如果当前值是 1~243,为了替换一个被删除的记录而手动插入是 189,那么下一个一致性值就要使用 244。

同一时间,只能有一张表可以将这个 identity\_insert 选项打开。用户可以通过全局变量 @@identity 显示最后插入的值。当使用 select into 创建表时,会创建下面这样一个一致性列:

```
SELECT id_key = identity(precision), col_a, col_b
INTO new_tab FROM old_tab
```

下面的 precision 决定了一致性值的最大范围,它表示的是 10 的次方数,既  $10^{\text{precision}}-1$ 。

在已存在的一致性列的表中使用 select into 时,只要用户一行接一行,按行的基数进行选择,并且不改变一致性值或者不止一次地包含他们,新的表列会继承原表的所有的属性。因此,如果用户进行下面的操作,新的列将没有一致性属性:

- 包含 UNION,联合,GROUP BY 或者选择的集合体。
- 不只一次地选择一致性的列(对此要小心,它不能根据两个一致性列产生错误,但是它处理这两个列不再具有一致性属性)。
- 在表达式中包含一致性列。

如果用户想要在创建新列时,将一个旧的一致性列具有同样的属性,那么由于在一个表中只能有一个一致性列可以选择,所以 select into 将失败。

为了随时参考一致性列,可以使用 syb\_identity 这个关键字替换一致性列的名称。当把这个名称用于内部代理时,这样可以免去泄露这个一致性列名称的需求。

```
SELECT max(syb_identity) FROM tab_name
```

```
-----
```

```
3457
```

新的数据库选项 auto identity 允许自动地对新的表加上一个 10 个数字的一致性列编号,而这个表可以在没有首选关键字、唯一性约束及一致性列的情况下创建。

```
* sp_dboption *
```

```
sp_dboption db_name, 'auto identity', true
```

这样的自动地生成的一致性列不能获得 select \*, 它只能在选择语句(大写命令)中,通过使用 SYB\_IDENTITY\_COL 列进行观察。

System 11 中增加的内容。

System 11.0 对一致性属性增加了某些控制。

为了使用光标和隔离级别 0(不干净的读)的过程,新的 System 11 数据库选项可以强制执行独特性。

```
sp_dboption db_name, 'identity in non unique index', true
```

它可以自动地将现有的一致性列包含在表索引之中,以便于所有的索引具有唯一性。注意:这张表必须具有能够应用的现存的一致性列。这种唯一的索引对于光标修正过程以及隔离级别 0 读来说,都是需要的。

新的 System 11 配置选项 identity grab size 指定了顺序的一致性值的编号,这些一致性值是为了插入处理表的特殊任务而保留的。

```
* sp_configure *
```

```
sp_configure 'identity grab size', number
```

这个语句可以为每个依表插入的过程保留一致性值的编号。如果在处理一张表中有两个过程,并且这个一致性强占尺寸设置为 20:

```
sp_configure 'identity grab size', 20
```

那么过程 1 将有 1~19 个保留的一致性值;过程 2 将有 20~39 个保留一致性值。如果过程 1 插入了 15 行,过程 2 插入了 8 行,那么这张表将含有 1~15 个一致性值和 20~28 个一致性值。如果过程 1 在插入 15 之后关闭了,则 16~19 的一致性值将丢失,并且其他的过程也不能再使用它们。

要小心,这种裂缝可能是普遍的。

#### 1.1.4 约束

```
CONSTRAINT constraint_name
{ {UNIQUE | PRIMARY KEY}
  [ CLUSTERED | NONCLUSTERED ]
  [ WITH {FILLFACTOR | MAX_ROWS_PER_PAGE}= X ]
  [on segment_name ]
  | REFERENCES table_name [ ( col_name ) ]
  | CHECK ( check_condition ) }
```

这个语句选取了 UNIQUE,从而指定了一个列的完整性约束。

##### UNIQUE

两行不可能有同样的非空值。空值是允许的,但是只有一行可以有空值。要保持一致,因为空值预示着它可能是一种不知道类型的值。如果这个值事先不知道,就不可能决定它是否与其他值相同。上述语句通过使用 alter table,给定约束,它会自动地创建一个唯一的索引。缺省时,这个索引是不分簇的。

##### PRIMARY KEY

两行不可能有同样的值,并且不允许出现空值。它会再一次自动地创建唯一的索引。缺省时是分簇的。很可怜,SQL Server 在遵守指定完整性规则时,对每个表所做的首选关键字的声明并不提供支持。然而为了适用前面的 System 10,不得不创建的表来解决所遗下的问题,这还是很容易理解的。这样可能遇到索引和触发器的复制问题,宁愿麻烦点,也不愿意去改变 System 10。

还有一个令人烦恼的问题,那就是 SQL Server 这一项的缺省索引是分簇的。如果仅是表的索引,那还很好。但是就普遍情况来说,对经常使用的单个记录存取的首选关键字进行分簇索引就显得有些浪费。要经常考虑到,一般情况需要忽视缺省索引选择,并且要使这种索引进行分簇。

除了缺省索引选择之外,这种声明的唯一性是同样的。然而即使从文件外观来说,这一点对声明表的主关键字也是很有用的。推荐大家在使用首选关键字时使用这个从句,在使用其他候选关键字时使用这种唯一性。通过缺省方法,将参考从句链接到主要关键字时,采用这种关键字从句是有益处的。

##### REFERENCES

它指明在这个列和命名表中的单个列之间使用的参考的完整性约束。如果指明了其他表中的列,它必须定义为唯一、定义为首选关键字或者定义在创建唯一索引之中。如果没有指定列,那么链会自动地链接到首选关键字,而这时参考从句中的表必须存在。SQL Server 使用这个参考从句核查输入的数据值是否存在于参考表的列中。因此,当向表增加一行时,任何的参

考列都将对其进行核查,看一看首选关键字是否已经存在。如果不存在,输入失败。同样,如果参考列被修改了,SQL Server 将核查新的已经存在的列值是否作为首选关键字出现。

#### CHECK

它是列必须遵守的规则。check\_condition 与规则条件有相同的格式。也就是说,除了不能包含子查询或参考表对象之外,它可以出现在任何从句可以出现的地方。

```
CHECK ( loc_code IN ( '01' , '02' )
```

```
OR loc_code LIKE '[ A-Z ][0-9] ')
```

check\_condition 周围的命令使用的括号要引起注意。

constraint\_name 是一个可选项。但是它指定的约束名字,在约束条件创建索引(唯一/首选关键字)时是很有用的。因为索引名字会设置为约束名字。如果用户不指定 constraint\_name,那么约束名字将按下面方式命名:

#### 参考/核查约束

---

```
tabname_colname_object_id    列约束;
```

```
tabname_object_id            表约束;
```

---

#### 唯一/首选关键字约束

---

```
tabname_colname_tabindid    这里的 tabindid 是表标识符:index_id。
```

---

为了限制这些缺省的名字的尺寸,表名字的前 10 个字符和列名字的前 5 个字符都要用到。

只有在使用转换表丢失约束时,为唯一性和首选关键字而创建的索引才会被丢失。

```
* alter table *
```

```
ALTER TABLE tab_name
```

```
DROP CONSTRAINT constrainr_name
```

这个语句是一个清楚地命名约束的良好的练习。同样,在不允许对包含其他列进行重定义时,在单个列的首选关键字上自动地创建索引相当令人麻烦。SQL 在首选关键字上增加列的修复将从包括查询的混合索引中得到好处。这样不得不增加约束的首选关键字索引。因为采用了单个列的索引,SQL 已创建的索引比 SQL 所需要的要多等原因,既使在恢复执行中减少增加,执行也将是困难的(有关索引概述的全部介绍参见查询优化一节)。

约束从句可以作为表级约束来声明。

```
CONSTRAINT constraint_name
```

```
{ UNIQUE | PRIMARY KEY }
```

```
[ CLUSTERED | NONCLUSTERED ]
```

```
(col_name [, col_name ]...)
```

```
[ WITH { FILLFACTOR | MAX_ROWS_PER_PAGE } = X ]
```

```
[ on segment_name ]
```

```
| FOREIGN KEY ( col_name [, col_name ]...)
```

```
REFERENCES tab_name [ ( col_name [, col_name ]... ) ]
```

```
| CHECK ( check_condition )
```

我宁愿将它用在表级上,而不是列级上,虽然 constraint\_name 仍是一个可选项。索引名最