

IBM[®] DB2[®] 通用数据库



发行说明

版本 7.2 / 版本 7.1 修订包 3

IBM[®] DB2[®] 通用数据库



发行说明

版本 7.2 / 版本 7.1 修订包 3

目录

欢迎使用 DB2 通用数据库版本 7! . . . xi

第1部分 特殊注释 1

特殊注释. 3

DB2 UDB 版本 7 的可访问性功能部件	3
键盘输入和导航	3
易用的显示功能部件	3
可选警报提示	3
与辅助性技术的兼容性	4
可访问文档	4
其他所需的 Solaris 补丁级别	4
在 DB2 版本 7 Solaris 版上受支持的 CPU	4
将节点添加至分区数据库时出现的问题	4
迁移期间的错误	6
在红旗 Linux 上的中文语言环境修正	6
如果未连接可更换驱动器的话, DB2 安装可能会挂起	6
日语和简体中文 Linux 环境中 DB2 Linux 版的附加语言环境设置	6
Microsoft Internet Explorer 的“控制中心”问题	6
Windows 环境中的信息目录管理器与 Sybase 的不兼容性	7
控制中心功能丧失	7
Netscape CD 未与 DB2 UDB 一起交付	7
XML 自述文件中的错误	7
在 Linux S/390 版上可能出现的数据损失	8
Windows 2000 上的 DB2 UDB	8

联机文档 (HTML、PDF 和 Search) 9

Windows 2000 操作系统上受支持的 Web 浏览器	9
在 Solaris 上搜索 DB2 联机信息	9
切换“NetQuestion OS/2 版”以使用 TCP/IP	9
试图启动 Netscape 时的错误消息	10
基于 UNIX 的系统上 Adobe Acrobat Reader 的配置需求	10
SQL Reference 是在一个 PDF 文件中提供的	11

第2部分 安装与配置 13

一般安装信息 15

下载所有受支持 DB2 客户机的安装软件包	15
在 Windows 2000 上安装 DB2	15
有关使用专用寄存器定义的视图的迁移问题	15
Windows 2000 上的 IPX/SPX 协议支持	15
在升级 DB2 的先前版本前, 停止 DB2 进程	15
在安装 DB2 之后运行 db2iupdt (如果已经安装了另一 DB2 产品的话)	16
设置要运行 DB2 控制中心的 Linux 环境	16
S/390 上的 DB2 通用数据库企业版和 DB2 Connect 企业版的 Linux 版	16

DB2 通用数据库扩充企业版 UNIX 版快速入门	17
HP-UX 的 shmseg 内核参数	17
迁移 IBM Visual Warehouse 控制数据库	17
访问仓库控制数据库	17

Data Links Manager 快速入门. 19

Dlfm 启动失败, 出现消息: “获取用于前缀的 afsfid 时出错”	19
对归档文件设置 Tivoli Storage Manager 类	19
DFS 客户机使能器的磁盘空间需求	19
在 AIX 上监控 Data Links File Manager 后端进程	19
安装和配置 DB2 Data Links Manager AIX 版: DCE-DFS 环境中的其他安装说明	19
“dlfm add_prefix” 命令失败	20
安装和配置 DB2 Data Links Manager AIX 版: 使用 db2setup 实用程序在 AIX 上安装 DB2 Data Links Manager	20
安装和配置 DB2 Data Links Manager AIX 版: DCE-DFS 后安装任务	20
安装和配置 DB2 Data Links Manager AIX 版: 使用 Smit 人工安装 DB2 Data Links Manager	21
安装和配置 DB2 Data Links DFS 客户机使能器	21
安装和配置 DB2 Data Links Manager Solaris 版	21
选择在 AIX 上的 DB2 Data Links Manager 的备份方法	22
选择在 Solaris 操作环境上的 DB2 Data Links Manager 的备份方法	23
选择在 Windows NT 上 DB2 Data Links Manager 的备份方法	24
在 AIX 上备份日志文件系统	25
Windows NT 上的 Data Links 中的管理员组特权	26
最小化 Data Links File System Filter (DLFF) 安装记录	26
安装后记录消息	27
使用 SMIT 卸载 DLFM 组件时可能会除去附加文件集	27
在开始/确定主机名之前	27
使用 Data Links File Manager: 在从 DB2 数据库中删除 DB2 Data Links Manager 之后进行清理	28
DLFM1001E (新的错误消息)	29
DLFM 设置配置文件选项	29
当在 AIX 上运行 Data Links/DFS 脚本 dmapp_prestart 时出错	29
Tivoli Space Manager 与 Data Links 集成	30
约束与限制	30
第 4 章 安装和配置 DB2 Data Links Manager AIX 版	31
普通安装说明	31

安装和配置补遗 33

第 5 章 在 UNIX 操作系统上安装 DB2 客户机	33
--	----

HP-UX 内核配置参数	33
第 12 章 运行您自己的应用程序	33
使用运行时客户机绑定数据库实用程序	33
使用 ODBC 对 DB2 的 UNIX 客户机访问	33
第 24 章 设置联合系统以访问多个数据源	33
联合系统	33
安装 DB2 Relational Connect	34
第 26 章 访问 Oracle 数据源	35
文档错误	36
访问 Sybase 数据源 (新章节)	37
将 Sybase 数据源添加至联合服务器	37
指定 Sybase 代码页	42
使用 ODBC 来访问 Microsoft SQL Server 数据源 (新章节)	42
将 Microsoft SQL Server 数据源添加至联合服务器	43
复查 Microsoft SQL Server 代码页	47
第3部分 管理	49
管理指南: 计划	53
第 8 章 物理数据库设计	53
分区键	53
设计节点组	53
第 9 章 设计分布式数据库	53
更新多个数据库	53
第 13 章 Windows NT 环境中的高可用性	53
在运行 DB2MCS 实用程序之前需要重新引导机器	53
第 14 章 Sun Cluster 2.2 上的 DB2 和高可用性	54
Solaris 上的 Veritas 支持	54
附录 B. 命名规则	55
有关长于 8 个字符的用户标识和模式名的注意事项	55
用户标识和口令	55
附录 D. 发行版之间的不兼容性	56
Windows NT DLFS 与 Norton 的实用程序不兼容	56
SET CONSTRAINTS 被替换为 SET INTEGRITY	56
附录 E. 国家语言支持	56
DB2 版本 7 的国家语言版本	56
DB2 管理服务器的语言环境设置	59
DB2 UDB 在 Windows 平台上支持 Baltic Rim 代码页 (MS-1257)	59
派生代码页值	59
国家代码和代码页支持	60
字符集	60
管理指南: 实现	61
添加或扩充 DMS 容器 (新进程)	61
第 1 章 使用 GUI 工具管理 DB2	61
第 3 章 创建数据库	61
创建表空间	61
创建序列	63
比较 IDENTITY 列和序列	64
创建索引、索引扩展项或索引规范	64
第 4 章 改变数据库	65
将容器添加至分区上的 SMS 表空间	65
改变标识列	65

改变序列	65
删除序列	66
切换表空间的状态	66
在 DMS 表空间中修改容器	66
第 5 章 控制数据库访问	66
序列特权	67
数据加密	67
第 8 章 恢复数据库	68
如何使用暂挂 I/O	68
增量备份和恢复	69
并行恢复	71
备份至命名管道	71
从分割映象备份	71
按需日志归档	73
日志镜像	74
Sun Solaris 和 HP 上的跨平台备份和恢复支持	74
DB2 Data Links Manager 注意事项 / 备份实用程序注意事项	74
DB2 Data Links Manager 注意事项 / 恢复和前滚实用程序注意事项	75
从脱机备份恢复数据库而不进行前滚	76
恢复数据库和表空间, 并前滚至日志结束	76
DB2 Data Links Manager 和恢复交互作用	77
需要协调的情况的检测	79
附录 C 数据库恢复的用户出口	80
附录 D 对多个数据库分区服务器发出命令	80
附录 I 高速节点间通信	81
使 DB2 能够运行 (使用 VI)	81
管理指南: 性能	83
第 3 章 应用程序注意事项	83
指定隔离级别	83
调整优化类	83
动态复合语句	84
第 4 章 环境注意事项	84
使用较大的索引键	84
第 5 章 系统目录统计信息	85
收集和使用分布统计信息	85
用于更新目录统计信息的规则	85
子元素统计信息	85
第 6 章 了解 SQL 编译器	87
复制的总结表	87
数据访问概念和优化	88
第 8 章 操作性能	89
管理数据库缓冲池	89
管理多个数据库缓冲池	90
第 9 章 使用管理器	90
第 13 章 配置 DB2	90
排序堆大小 (sortheap)	91
排序堆阈值 (sheapthres)	91
升级前锁定列表的最大百分比 (maxlocks)	91
配置 DB2/DB2 Data Links Manager / Data Links 访问令牌到期时间间隔 (dl_expint)	92
MIN_DEC_DIV_3 数据库配置参数	92
应用程序控制堆大小 (app_ctl_heap_sz)	93
数据库系统监控器堆大小 (mon_heap_sz)	93

最大活动应用程序数 (maxappls)	93
恢复范围和软检查点时间间隔 (softmax)	93
跟踪已修改页启用 (trackmod)	93
更改数据库日志路径 (newlogpath)	94
日志文件的位置 (logpath)	95
锁定列表的最大存储器 (locklist)	95
附录 A DB2 注册表和环境变量	95
新的或更改过的注册表变量表	95
附录 C SQL 说明工具	99
卫星管理指南和参考	101
将版本 7.2 DB2 个人版和 DB2 工作组版设置为卫星	101
先决条件	101
配置版本 7.2 系统以进行同步	103
在版本 6 企业版系统上安装修订包 2 或更高版本	104
升级版本 6 控制中心和卫星管理中心	110

Command Reference 111

db2batch - Benchmark Tool	111
db2cap (新命令)	111
db2cap - CLI/ODBC 静态软件包绑定工具	112
db2ckrst (新命令)	113
db2ckrst - 检查增量恢复映象序列	114
db2gncol (新命令)	115
db2gncol - 更新生成的列值	116
db2inidb - 初始化镜象数据库	117
db2look - DB2 统计信息抽取工具	117
db2updv7 - 将数据库更新为版本 7 当前修正级别	118
新命令行处理器选项 (-x, 不打印列标题)	119
DB2 CLP 的 True Type 字体需求	119
ADD DATALINKS MANAGER	119
ARCHIVE LOG (新命令)	119
归档日志	120
BACKUP DATABASE	121
语法图	121
DB2 Data Links Manager 注意事项	122
BIND	122
CALL	123
DROP DATALINKS MANAGER (新命令)	123
DROP DATALINKS MANAGER	124
EXPORT	127
GET DATABASE CONFIGURATION	127
GET ROUTINE (新命令)	128
GET ROUTINE	129
GET SNAPSHOT	129
IMPORT	129
LIST HISTORY	130
LOAD	131
PING (新命令)	132
PING	133
PUT ROUTINE (新命令)	134
PUT ROUTINE	135
RECONCILE	135
REORGANIZE TABLE	136
RESTORE DATABASE	136

语法	136
DB2 Data Links Manager Considerations	137
ROLLFORWARD DATABASE	137
CLP 返回码中的文档错误	137

Data Movement Utilities Guide and Reference 139

Chapter 2. Import	139
Using Import with Buffered Inserts	139
Chapter 3. Load	139
Pending States After a Load Operation	139
Load Restrictions and Limitations	139
totalreespace File Type Modifier	139
Chapter 4. AutoLoader	139
rexecd Required to Run Autoloader When Authentication Set to YES	139

Replication Guide and Reference 141

复制和非 IBM 服务器	141
在 Windows 2000 上进行复制	141
保存 SQL 文件时的已知错误	141
DB2 维护	141
Web 上的数据差异实用程序	141
Chapter 3. Data replication scenario	141
复制方案	141
Chapter 5. Planning for replication	142
表名和列名	142
DATALINK 复制	142
LOB 限制	143
计划复制	143
Chapter 6. Setting up your replication environment	143
随处更新先决条件	143
设置复制环境	143
Chapter 8. Problem Determination	144
Chapter 9. Capture and Apply for AS/400	145
Chapter 10. Capture and Apply for OS/390	145
Prerequisites for DB2 DataPropagator for OS/390	145
UNICODE and ASCII Encoding Schemes on OS/390	146
Chapter 11. Capture and Apply for UNIX platforms	147
在 UNIX 和 Windows 上为 Capture 和 Apply 设置环境变量	147
Chapter 14. Table Structures	147
Chapter 15. Capture and Apply Messages	148
Appendix A. Starting the Capture and Apply Programs from Within an Application	148

System Monitor Guide and Reference 149

db2ConvMonStream	149
----------------------------	-----

Troubleshooting Guide 151

在用户未登录的情况下在 Windows 95、Windows 98 和 Windows ME 上启动 DB2	151
Chapter 2. Troubleshooting the DB2 Universal Database Server	154

在 64 位的平台上使用“DB2 通用数据库” 155

第 5 章 配置 155
 LOCKLIST 155
 shmsys:shminfo_shmmax 155
第 6 章 限制 155

XML Extender 管理和编程 157

MQSeries 159

DB2 MQSeries 函数的安装和配置 159
 安装 MQSeries 159
 安装 MQSeries AMI 159
 启用 DB2 MQSeries 函数 160
MQSeries 消息传递样式 160
消息结构 161
MQSeries 函数概述 161
 限制 162
 错误码 162
使用方案 162
 基本消息传递 162
 发送消息 163
 检索消息 164
 应用程序间的连通性 165
enable_MQFunctions 168
 enable_MQFunctions 169
disable_MQFunctions 170
 disable_MQFunctions 171

第4部分 管理工具 173

控制中心. 175

管理 DB2 服务器 VSE 版和 VM 版服务器的能力 175
Java 1.2 支持“控制中心” 175
在 Windows 操作系统上使用联机帮助时发生“快捷方式无效”错误 176
OS/2 上的 Java 控制中心 176
试图在 Windows 操作系统上的日志中查看已完成作业时发生“拒绝访问文件”错误 176
多站点更新测试连接 176
 “DB2 OS/390 版”的“控制中心” 176
 对“控制中心 OS/390 版”的必要修正 178
 对“创建空间层”对话框的更改 178
 “DB2 控制中心”的故障诊断信息 178
 基于 UNIX 的系统上的控制中心故障诊断 178
 OS/2 上可能的 Infopops 问题 178
 jdk11_path 配置参数的帮助 179
 在使用“脚本中心”或“日志”时出现的 Solaris 系统错误 (SQL10012N) 179
 DPREPL.DFT 文件的帮助 179
 启动多个控制中心小应用程序 179
 关于作为小应用程序运行的“控制中心”的联机帮助 179
 以小应用程序方式运行“控制中心”(Windows 95) 180
 使用大型查询结果 180

信息中心. 181

在 Windows 操作系统上出现“快捷方式无效”错误 181
当 Netscape 已打开时, 打开 Netscape Navigator 中的外部 Web 链接(基于 UNIX 的系统) 181
启动“信息中心”时的问题 181

向导 183

在创建数据库向导中设置范围大小 183
“MQSeries 辅助程序”向导 183
“OLE DB 辅助程序”向导 183

第5部分 商业智能 185

商业智能教程. 189

修正商业智能教程 189

数据仓库中心管理指南 191

故障诊断 191
将 Excel 设置为仓库源 191
定义和运行进程 191
 “导出元数据”对话框 191
为“提交 OS/390 JCL 作业流”(VWPMVS) 程序定义值 191
对『数据入库样本』附录的更改 191
数据仓库中心消息 192
创建轮廓并在“DB2 OLAP 集成服务器”中装入数据 192
将 Classic Connect 配合“数据仓库中心”使用 192
 “数据仓库中心”环境结构 192
使用“反转变换器” 193
用 DB2 版本 7 仓库代理程序来访问 DB2 版本 5 数据 193
 迁移 DB2 版本 5 服务器 193
 更改代理程序配置 193
IBM ERwin 元数据抽取程序 194
 内容 194
 软件需求 195
 程序文件 195
 创建标记语言文件 196
 将标记语言文件导入数据仓库中心 197
 将标记语言文件导入信息目录管理器 198
 故障诊断 198
 ERwin 与“DB2 数据仓库中心”之间的映射 199
清理“数据仓库中心”中的名称和地址 202
编写 Trillium Batch System JCL 文件 206
 在 UNIX 环境变量 Windows 上编写 Trillium Batch System 脚本文件 206
 定义 Trillium Batch System 步骤 206
 使用 Trillium Batch System 用户定义程序 206
 错误处理 207
MQSeries 与“数据仓库中心”集成 209
 为 MQSeries 消息创建视图 209
 导入 MQSeries 消息和 XML 元数据 210
Microsoft OLE DB 和“数据事务服务”支持 212
 为 OLE DB 表函数创建视图 213

为 DTS 软件包创建视图	213	将步骤信息打印至文件	231
将增量提交与替换配合使用	213	Data Warehouse Center Application	
组件跟踪数据文件名	213	Integration Guide	233
在 AIX 和 Solaris 操作环境下, Sybase 源所需要的		附加的元数据模板	233
开放式客户机	214	Commit.tag	233
已校正的样本条目	214	ForeignKey.tag	234
第 3 章 设置仓库源	214	ForeignKeyAdditional.tag	236
将 Microsoft Access 中的“备忘录”字段映射至		PrimaryKey.tag	237
仓库源.	214	PrimaryKeyAdditional.tag	239
第 10 章 维护仓库数据库	215	数据仓库中心联机帮助	241
将表链接至 DB2 UDB RUNSTATS 程序的步骤		对复制定义表或视图	241
子类型.	215	将 Essbase VWP 与 AS/400 代理程序一起运行	241
缺省仓库控制数据库	215	使用“发布数据仓库中心元数据”窗口和“相关特	
“仓库控制数据库管理”窗口	215	性”窗口	241
更改活动的仓库控制数据库	215	外键	241
创建和初始化仓库控制数据库	216	复制笔记本	241
创建版本化的 SQL 步骤.	216	导入标记语言	241
在“过程模型”窗口中更改源和目标	216	用于添加数据的链接	241
为“数据仓库中心”对象添加描述.	216	导入表.	241
运行样本内容	216	对 RUNSTATS 和 REORGANIZE TABLE 联机帮助	
编辑“创建 DDL SQL”语句	217	的校正.	242
迁移 Visual Warehouse 商业视图	217	“通知”页 (“数据仓库特性”笔记本和“调度”笔	
生成目标表和主键	217	记本).	242
使用 Merant ODBC 驱动程序	217	“代理点”笔记本中的“代理模块”字段	242
新的 ODBC 驱动程序	218	DB2 OLAP Starter Kit	243
在 OS/2 数据库中定义仓库源或目标	218	OLAP 服务器 Web 站点.	243
监控仓库控制数据库的状态	218	受支持的操作系统服务级别	243
将“SQL 辅助程序”与 TBC_MD 样本数据库配合		在 UNIX 上完成 DB2 OLAP Starter Kit 设置	243
使用	218	为 OLAP Starter Kit 配置 ODBC.	244
使用 FormatDate 函数	218	在 UNIX 系统上配置数据源	244
更改语言设置	218	在 UNIX 系统上配置“OLAP 元数据目录”	245
使用“生成键表”变换器.	219	在 Windows 系统上配置数据源	246
维护与数据库的连接	219	在 Windows 系统上配置“OLAP 元数据目录”	247
设置远程“数据仓库中心”客户机.	219	在配置数据源之后	248
定义“DB2 VM 版”仓库源	219	从“OLAP Starter Kit 桌面”进行登录	248
定义“DB2 VM 版”或“DB2 VSE 版”目标表	219	Starter Kit 登录示例	249
启用定界标识符支持	220	人工创建和配置 OLAP Starter Kit 的样本数据库	249
Data Joiner 错误指示绑定问题	220	将应用程序迁移至 OLAP Starter Kit 版本 7.2.	250
用“数据仓库中心”设置和运行复制	220	已知问题和限制.	250
故障诊断提示	222	缺少 OLAP Spreadsheet Add-in EQD 文件.	252
访问源和目标	223	Information Catalog Manager	
受支持的非 IBM 数据库源的补充内容	224	Administration Guide	253
在“数据仓库中心”中人工创建数据源	225	信息目录管理器初始化实用程序	253
使用“公共数据仓库元数据交换”(CWMI)导入和导		许可证发放问题.	253
出元数据	226	安装问题.	253
介绍	226	使用 DB2 版本 7 信息目录管理器来访问 DB2 版本	
导入元数据	226	5 信息目录	254
运行导入实用程序之后更新元数据.	227	设置信息目录	255
导出元数据	228	与其他产品交换元数据	255
OS/390 Runstats 实用程序步骤	229	使用 flgnxoln 命令来交换元数据	255
OS/390 装入实用程序步骤	229	使用 MDISDGC 命令来交换元数据	255
公共仓库元模型 (CWMI) XML 支持	230	调用程序	255
进程造型程序	230		
模式造型程序	231		
必填字段	231		
数据仓库中心启动板增强功能	231		

Information Catalog Manager Programming Guide and Reference . 257

信息目录管理器原因码 257

信息目录管理器用户指南 259

信息目录管理器: 联机消息 261

消息 FLG0260E. 261
消息 FLG0051E. 261
消息 FLG0003E. 261
消息 FLG0372E. 261
消息 FLG0615E. 261

信息目录管理器: 联机帮助 263

用于 Web 的“信息目录管理器” 263

DB2 仓库管理器安装指南 265

仓库变换器的软件需求 265
SAP R/3 的连接器的 265
 安装先决条件 265
用于 Web 的连接器的 265
 安装先决条件 265

Query Patroller Administration Guide 267

DB2 Query Patroller 客户机是一个独立组件 267
使用 dqpmigrate 从 DB2 Query Patroller 的版本 6
迁移 267
Enabling Query Management. 267
控制表的表空间的位置 267
dqptest 命令的新参数. 268
iwm_cmd 命令的新参数 268
新的注册表变量: DQP_RECOVERY_INTERVAL
启动 Query Administrator 269
用户管理 269
创建作业队列 269
使用“命令行接口” 269
 “查询使能器”注释 269
DB2 Query Patroller Tracker 可能会返回空白列页
Query Patroller 和复制工具 270
Appendix B. Troubleshooting DB2 Query Patroller
Clients 270

第6部分 应用程序开发 271

Administrative API Reference 275

db2ArchiveLog (新 API) 275
 db2ArchiveLog 276
db2ConvMonStream. 278
db2DatabasePing (新 API) 278
 db2DatabasePing - Ping 数据库 279
db2HistData 280
db2HistoryOpenScan 281
db2XaGetInfo (新 API) 282
 db2XaGetInfo - 获取资源管理器的信息 283
db2XaListIndTrans (取代 sqlxphqr 的新 API) 283

 db2XaListIndTrans - 列示不确定事务 284
db2GetSnapshot - 获取快照 287
忘记日志记录 287
sqlaintp - 获取错误消息 287
sqlbctcq - 关闭表空间容器查询 287
sqlubkp - 备份数据库 287
sqlureot - 重组表 288
sqlurestore - 恢复数据库 288
关于“AIX 扩展共享内存支持”(EXTSHM)的文档
错误 288
SQLFUPD 288
 锁定列表 288
SQLEDBDESC 289
SQLFUPD 文档错误 289

应用程序构建指南 291

第 1 章 介绍 291
 受支持软件 291
 样本程序 293
第 3 章 构建 DB2 应用程序的一般信息 293
 构建文件、Makefile 和错误检查实用程序 293
第 4 章 构建 Java 小应用程序和应用程序 293
 设置环境 293
第 5 章 构建 SQL 过程 295
 设置 SQL 过程环境 295
 设置编译器环境变量 296
 定制编译命令 297
 保留中间文件 300
 备份与恢复 300
 创建 SQL 过程 301
 调用存储过程 301
 分布编译型 SQL 过程 301
第 7 章 构建 HP-UX 应用程序 302
 HP-UX C. 302
 HP-UX C++. 302
第 9 章 构建 OS/2 应用程序 302
 VisualAge C++ OS/2 版的版本 4.0 302
第 10 章 构建 PTX 应用程序 303
 ptx/C++. 303
第 12 章 构建 Solaris 应用程序 303
 SPARCCompiler C++. 303
第 13 章 为 Windows 32 位操作系统构建应用程序 303
 VisualAge C++ 版本 4.0 303

Application Development Guide 305

Chapter 2. Coding a DB2 Application. 305
 对 Microsoft Visual C++ 激活 IBM DB2 通用数
 据库项目和工具加载件 305
Chapter 6. Common DB2 Application Techniques 305
 生成序列值 305
Chapter 7. Stored Procedures 307
 DECIMAL 类型在 Linux Java 例程中失效 307
 在递归存储过程中使用游标 308
 编写 OLE 自动存储过程 308
Chapter 12. Working with Complex Objects:
User-Defined Structured Types 308

将结构化类型属性插入到列中	308	启用新函数和过程	335
Chapter 13. Using Large Objects (LOBs).	309	标量函数	335
联合数据库系统中的大对象 (LOB) 支持	309	表函数	357
调整系统	310	过程	360
Part 5. DB2 Programming Considerations.	310	Chapter 5. Queries	362
IBM DB2 OLE DB 提供器	310	select-statement 语法图	362
Chapter 20. C 和 C++ 编程.	311	select-statement/fetch-first-clause	363
存储过程、函数和方法的 C 和 C++ 类型	311	Chapter 6. SQL Statements	363
Chapter 21. Programming in Java	312	现在支持更新分区键	363
PARAMETER STYLE JAVA 过程和函数中的		用于 Unicode 数据库的较大索引键	364
Java 方法签名	312	ALTER SEQUENCE	364
连接至 JDBC 小应用程序服务器	312	ALTER TABLE	367
Appendix B. Sample Programs	313	复合 SQL (嵌入式)	370
CLI Guide and Reference 315		复合语句 (动态)	370
使用运行时客户机绑定数据库实用程序	315	CREATE FUNCTION (源或模板)	373
在 CLI 应用程序中使用静态 SQL	315	CREATE FUNCTION (SQL 标量、表或行)	374
创建静态 JDBC/ODBC/CLI 简要表的限制	315	CREATE METHOD	375
ADT 变换	315	CREATE SEQUENCE	376
Chapter 3. Using Advanced Features	316	CREATE TRIGGER	380
编写多线程应用程序	316	CREATE WRAPPER	383
可滚动游标	317	DECLARE CURSOR	383
使用复合 SQL	318	DELETE	383
使用存储过程	318	DROP	384
Chapter 4. Configuring CLI/ODBC and Running		GRANT (序列特权)	385
Sample Applications	319	INSERT	386
配置关键字	319	SELECT INTO	387
Chapter 5. DB2 CLI Functions	319	SET ENCRYPTION PASSWORD	388
SQLBindFileToParam - 将 LOB 文件参考绑定至		SET transition-variable	389
LOB 参数	319	UPDATE	392
SQLNextResult - 将下一个结果集与另一语句句柄		Chapter 7. SQL Procedures 现在名为 Chapter 7.	
相关联	319	SQL Control Statements	393
Appendix D. Extended Scalar Functions	321	SQL 过程语句	393
日期和时间函数	321	FOR	395
Appendix K. Using the DB2 CLI/ODBC/JDBC Trace		Compound Statement 更改为 Compound Statement	
Facility	322	(Procedure)	397
消息参考 323		RETURN	398
获取消息和 SQLSTATE 帮助	323	SIGNAL	400
DB2 Connect 中的 SQLCODE 重新映射更改	323	Appendix A. SQL Limits	403
新消息和已更改的消息	323	Appendix D. Catalog Views	403
调用层接口 (CLI) 消息	323	SYSCAT.SEQUENCES	403
DB2 消息	323	DB2 存储过程构建器 405	
DBI 消息	324	Java 1.2 支持 DB2 存储过程构建器	405
数据仓库中心 (DWC) 消息	324	DB2 存储过程的远程调试	405
SQL 消息	325	在 Windows、OS/2 或 UNIX 平台上构建 SQL 过程	406
校正过的 SQLSTATES	327	在 Solaris 平台上使用 DB2 存储过程构建器	406
SQL Reference 329		已知问题和限制	406
SQL Reference 是在一个 PDF 文件中提供的	329	将 DB2 存储过程构建器与繁体中文语言环境配合使	
Chapter 3. Language Elements	329	用	406
Naming Conventions and Implicit Object Name		UNIX (AIX、Sun Solaris 和 Linux) 安装和存储过	
Qualifications	329	程构建器	407
DATALINK Assignments	329	在 OS/390 上构建 SQL 存储过程	407
表达式	329	调试 SQL 存储过程	407
Chapter 4. Functions	335	导出 Java 存储过程	407
		在 OS/390 上插入存储过程	407

在 workstation 服务器上为“SQL 存储过程”设置构建选项	408
自动刷新 WLM 地址空间以便在 OS/390 上构建存储过程	408
在 OS/390 上开发 Java 存储过程	408
为 MQ Series 和 OLE DB 构建 DB2 表用户定义函数 (UDF)	408
Unicode 更新	409
介绍	409
DB2 Unicode 数据库和应用程序	409
文档更新	411
SQL Reference	412
Chapter 3 Language Elements	412
第 4 章 功能	415
CLI Guide and Reference	416
Chapter 3. Using Advanced Features	416
附录 C DB2 CLI 和 ODBC	420
Data Movement Utilities Guide and Reference	421
附录 C 导出 / 导入 / 装入实用程序文件格式	421

第7部分 连接至主机系统 423

Connectivity Supplement 425

在 VM 环境中设置应用程序服务器	425
CLI/ODBC/JDBC 配置 PATCH1 和 PATCH2 设置	425

第8部分 一般信息 429

一般信息 431

DB2 通用数据库商业智能快速导览	431
DB2 Everywhere 现在更名为 DB2 Everyplace	431
需要鼠标	431
试图从“DB2 运行时客户机”进行绑定导致“绑定文件找不到”错误	432
Search Discovery	432
HP-UX 11 的内存窗口	432
发生 dlfm client_conf 故障时的用户操作	433
偶尔出现“复制守护程序”在 dlfm 停止时不能停止的情况	433
卸载 DB2 DFS 客户机使能器	433
Windows NT 上的客户机认证	433
自动装入程序可能会在创建子进程期间挂起	434
DATALINK 恢复	434
在“IBM 通信服务器 Windows NT 版”(CS/NT)中定义用户标识和口令	434
节点定义	434
联合系统限制	435
DataJoiner 的限制	435
希伯莱语的“信息目录管理器 Windows NT 版”	435

Microsoft SNA Server 和 SNA 多站点更新(两阶段提交)支持	436
DB2 的 SNA SPM 在引导 Windows 后无法启动	436
DB2 管理服务器的语言环境设置	436
快捷方式不起作用	437
Windows NT 和 Windows 2000 上的 DB2 的服务帐户需求	437
版本 6 中创建的 Query Patroller 用户的 EXECUTE 特权丢失	437
Query Patroller 限制	437
需要提交将在“数据仓库中心”(DWC)中使用的所有用户定义程序	438
用于数据仓库中心命令行导出的新选项	438
备份服务 API (XBSA)	438
OS/390 代理程序	439
安装概述	439
安装详细信息	440
设置附加代理程序功能	442
使用触发器程序 (XTClient) 调度仓库步骤	443
变换器	444
访问 DB2 系列之外的数据库	447
运行“DB2 OS/390 版”实用程序	449
复制	450
代理程序日志记录	451
在 Windows NT 上进行客户机方高速缓存	452
企业版 UNIX CD-ROM 上的试用产品	452
DB2 Connect 企业版 UNIX CD-ROM 上的试用产品	453
删除 Data Links Manager	453
使用 SMIT 卸载 DLFM 组件时可能会除去附加文件集	453
在 Windows 2000 上使用 CLP 时发生错误	454
SQL1035N	454
SQL 辅助程序的增强功能	454
在 Linux 上对 DB2 的 Gnome 和 KDE 桌面集成	454
在 Windows 2000 Terminal Server, Administration Mode 下运行 DB2	454
关于备份和恢复命令的联机帮助	454
“仓库管理器”应为“DB2 仓库管理器”	455

第9部分 附加信息 457

附加信息 459

DB2 通用数据库和 DB2 Connect 联机支持	459
DB2 Magazine	459

第10部分 附录 461

附录. 声明 463

注册商标	465
------	-----

索引 467

欢迎使用 DB2 通用数据库版本 7！

注：将字体设置为等高等宽以便更好地查看这些“发行说明”。

“DB2 通用数据库和 DB2 Connect 支持”站点会定期更新。有关最新的信息，请参考 <http://www.ibm.com/software/data/db2/udb/winos2unix/support>。

此文件包含有关下列产品的信息，这些信息在打印 DB2 手册时并未提供：

- IBM DB2 通用数据库个人版的版本 7.2
- IBM DB2 通用数据库工作组版的版本 7.2
- IBM DB2 通用数据库企业版的版本 7.2
- IBM DB2 Data Links Manager 版本 7.2
- IBM DB2 通用数据库扩充企业版的版本 7.2
- IBM DB2 Query Patroller 版本 7.2
- IBM DB2 个人开发者修订版的版本 7.2
- IBM DB2 通用开发者修订版的版本 7.2
- IBM DB2 数据仓库管理器版本 7.2
- IBM DB2 Relational Connect 版本 7.2

为下列产品提供了单独的发行说明文件（安装为 READCON.TXT）：

- IBM DB2 个人版的版本 7.2
- IBM DB2 Connect 企业版的版本 7.2

《新增内容》一书包含了版本 7.2 的某些主要 DB2 增强功能的概述。如果没有《新增内容》一书的 7.2 版，则可以从 <http://www.ibm.com/software/data/db2/udb/winos2unix/support> 来查看和下载它。

注：页面左边的修订栏 (i) 指示自第一次发布《发行说明》之后，已在该级别上添加了一行或作出了修改。

第1部分 特殊注释

特殊注释

DB2 UDB 版本 7 的可访问性功能部件

DB2 UDB 产品系列中包括一些功能部件，它们可使残障人士更易使用这些产品。这些功能部件包括：

- 使键盘输入和导航更加容易的功能部件
- 增强显示特性的功能部件
- 声音和可视警报提示的选项
- 与辅助性技术的兼容性
- 与操作系统的可访问性功能部件 的兼容性
- 可访问的文档格式

键盘输入和导航

键盘输入

只使用键盘即可操作“DB2 控制中心”。菜单项和控件提供了一些访问键，使用户可以激活控件或直接从键盘选择菜单项。这些键都是自带说明的，在它们所出现的控件或菜单上，这些键都标有下划线。

键盘焦点

在基于 UNIX 的系统上，键盘焦点的位置是突出显示的，指示窗口的哪个区域是活动的，以及用户在何处击键才会有效。

易用的显示功能部件

“DB2 控制中心”中有一些功能部件增强了用户界面的功能，并改进了易用性以供视力较弱的用户使用。这些易用性的增强包括了对高对比度的设置以及可定制字体特性的支持。

高对比度方式

“控制中心”界面支持由操作系统提供的高对比度方式选项。此功能部件对需要背景色与前景色反差度较高的用户提供了帮助。

字体设置

“控制中心”界面允许用户选择菜单和对话框中文本的颜色、大小和字体。

与颜色无关

用户不需要辨别颜色就可使用此产品的所有功能。

可选警报提示

用户可选择是通过声音提示还是可视提示来接收警报。

与辅助性技术的兼容性

“DB2 控制中心”界面与诸如 Via Voice 之类的屏幕阅读应用程序兼容。处于应用程序方式时，“控制中心”界面具有这些易用的应用程序所必需的特性，使盲人用户可得到屏幕上的信息。

可访问文档

DB2 产品系列的文档是以 HTML 格式提供的。因此允许用户根据他们浏览器中的显示首选项设置来查看文档。它还允许使用屏幕阅读程序和其他辅助性技术。

其他所需的 Solaris 补丁级别

除 DB2 (UNIX 版)《快速入门》手册中列示的补丁外，“DB2 通用数据库版本 7 Solaris 版的版本 2.6”还需要补丁 106285-02 或更高版本。

在 DB2 版本 7 Solaris 版上受支持的 CPU

UltraSparc 之前的 CPU 版本不受支持。

将节点添加至分区数据库时出现的问题

将节点添加至一个分区数据库（该数据库具有一个或多个系统临时表空间，其页大小不同于缺省页大小（4 KB））时，您可能会遇到错误消息“SQL6073N “添加节点”操作失败”和一个 SQLCODE。发生这种情况的原因是创建该节点时仅存在页大小为 4 KB 的 IBMDEFAULTBP 缓冲池。

例如，可使用 **db2start** 命令，将节点添加至当前分区数据库：

```
DB2START NODENUM 2 ADDNODE HOSTNAME newhost PORT 2
```

如果分区数据库具有页大小为缺省页大小的系统临时表空间，会返回以下消息：

```
SQL6075W “启动数据库管理器”操作成功地添加了该节点。  
该节点在停止并再次启动所有节点之前将处于不活动状态。
```

但是，如果分区数据库的系统临时表空间的页大小不是缺省值页大小，会返回以下消息：

```
SQL6073N “添加节点”操作失败。SQLCODE = "<-902>"
```

在类似的示例中，可在用新节点描述人工更新 `db2nodes.cfg` 文件后使用 `ADD NODE` 命令。在对具有系统临时表空间（其页大小为缺省页大小）的分区数据库编辑该文件和运行 `ADD NODE` 命令之后，将返回以下消息：

```
DB20000I ADD NODE 命令成功完成。
```

但是，如果分区数据库的系统临时表空间的页大小不是缺省值页大小，会返回以下消息：

```
SQL6073N “添加节点”操作失败。SQLCODE = "<-902>"
```

防止上述问题的一个方法是在发出 **db2start** 或 `ADD NODE` 命令之前运行：

```
DB2SET DB2_HIDDENBP=16
```

此注册表变量使 DB2 能够分配每个池为 16 页（所使用的页大小与缺省页大小不同）的隐藏缓冲池。因此 ADD NODE 操作能够成功完成。

避免这些问题的另一方法是在 ADD NODE 或 **db2start** 命令上指定 WITHOUT TABLESPACES 子句。在进行此操作之后，必须使用 CREATE BUFFERPOOL 语句来创建缓冲池，并使用 ALTER TABLESPACE 语句来将系统临时表空间与缓冲池相关联。

在将节点添加至这样的节点组时，该节点组的一个或多个表空间的页大小不同于缺省页大小（4 KB），可能会遇到以下错误消息：“SQL0647N 缓冲池 "" 当前处于不活动状态。”。发生这种情况的原因是尚未对这些表空间激活在新节点上创建的非缺省页大小缓冲池。

例如，可使用 ALTER NODEGROUP 语句，将节点添加至节点组：

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2)
```

如果节点组具有页大小为缺省页大小的表空间，会返回以下消息：

```
SQL1759W 需要重新分布节点组以更改节点组 "<ng1>" 中对象的数据定位，
          以包括某些添加的节点或排除某些删除的节点。
```

但是，如果节点组具有页大小不为缺省页大小的表空间，则返回的消息为：

```
SQL0647N 缓冲池 "" 当前处于不活动状态。
```

避免此问题的一个方法是对每种页大小创建一些缓冲池，然后在发出 ALTER NODEGROUP 语句之前重新连接数据库：

```
DB2START
CONNECT TO mpp1
CREATE BUFFERPOOL bp1 SIZE 1000 PAGESIZE 8192
CONNECT RESET
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2)
```

防止该问题出现的第二种方法是在发出 **db2start** 命令、CONNECT 和 ALTER NODEGROUP 语句之前运行：

```
DB2SET DB2_HIDDENBP=16
```

在 ALTER TABLESPACE 语句用来将表空间添加至节点时，可能会出现另一个问题。例如：

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

这一系列命令和语句会生成错误消息 SQL0647N（而非期望的消息 SQL1759W）。

要正确地完成此更改，应在执行 ALTER NODEGROUP... WITHOUT TABLESPACES 语句之后重新连接数据库。

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
CONNECT RESET
CONNECT TO mpp1
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

防止该问题的另一种方法是在发出 **db2start** 命令、CONNECT、ALTER NODEGROUP 和 ALTER TABLESPACE 语句之前运行:

```
DB2SET DB2_HIDDENBP=16
```

迁移期间的错误

在迁移期间，即使迁移成功，db2diag.log 文件中也会出现错误条目（数据库未迁移）。这些错误条目可被忽略。

在红旗 Linux 上的中文语言环境修正

如果正在使用“简体中文版红旗 Linux 服务器版本 1.1”，则与开发红旗产品的公司联系以获得简体中文语言环境修正。如果没有版本 1.1 的“简体中文”语言环境修正，则 DB2 将不能识别“简体中文”的代码页是 1386。

如果未连接可更换驱动器的话，DB2 安装可能会挂起

在 DB2 安装期间，如果计算机的可更换驱动器未连接，则安装可能会在选择安装类型之后挂起。要解决此问题，运行安装程序，并指定 -a 选项:

```
setup.exe -a
```

日语和简体中文 Linux 环境中 DB2 Linux 版的附加语言环境设置

如果想要在日语或简体中文 Linux 系统上使用 Java GUI 工具（如“控制中心”），则附加语言环境设置是必需的。如果没有此设置的话，就不能正确显示日语或中文字符（汉字）。请将以下设置包括在您的用户简要表中，或在每次调用“控制中心”之前从命令行运行它。

对于日语系统:

```
export LC_ALL=ja_JP
```

对于简体中文系统:

```
export LC_ALL=zh_CN
```

Microsoft Internet Explorer 的“控制中心”问题

有一个问题是由 Internet Explorer (IE) 安全性选项设置造成的。“控制中心”使用未带符号的 jar，因此，对系统信息的访问被安全性管理器禁用。

要排除此问题，重新配置 IE 安全性选项，如下所示:

1. 在**查看**菜单 (IE4) 或**工具**菜单 (IE5) 上选择 **Internet** 选项。
2. 在“安全”页面上，选择**可信站点**。
3. 单击**添加站点...**。
4. 将“控制中心”Web 服务器添加至可信站点列表。如果“控制中心”Web 服务器在同一域中，则只添加该 Web 服务器（而不带域名）会非常有用。例如:

```
http://ccWebServer.ccWebServerDomain  
http://ccWebServer
```

5. 单击**确定**。
6. 单击**设置...**。

7. 向下滚动至 **Java** → **Java** 权限并选择自定义。
8. 单击 **Java 自定义设置...**。
9. 选择“编辑权限”页面。
10. 向下滚动至“未签名的内容 → 运行未签名的内容 → 附加的未签名权限 → 系统信息”，并选择启用。
11. 在每个打开的窗口上单击**确定**。

Windows 环境中的信息目录管理器与 Sybase 的不兼容性

在带有 Sybase Open Client 的 Windows NT 或 Windows 2000 机器上安装“信息目录管理器 (ICM) 版本 7.1”会导致错误，且“Sybase 实用程序”会停止运行。会出现类似如下的错误消息：

无法初始化 LIBTCL.DLL。请确保 SYBASE 环境变量设置正确。

通过从“Windows 环境”参数中除去环境参数 LC_ALL 来避免发生此情况。LC_ALL 是语言环境类别参数。语言环境类别是明显常量，本地化例程用它们来指定要使用程序的语言环境信息的哪一部分。语言环境指的是可对其定制程序的某些方面的位置（或国家）。例如，与语言环境相关的方面包括日期的格式化和货币值的显示格式。LC_ALL 会影响所有特定于语言环境的行为（所有类别）。

如果除去 LC_ALL 环境参数，ICM 就可与 Sybase 在 Windows NT 平台上共存，而下列设施不再有效：

- 信息目录用户
- 信息目录管理员
- 信息目录管理器

控制中心功能丧失

通过对 DB2 服务器应用修订包 2，就不会导致低级别的“控制中心”客户机出现问题。但是，在 DB2 版本 7.2 中，低级别的“控制中心”客户机几乎失去了所有功能。在这种情况下，低级别指的是“修订包 6”之前的任何版本 6 客户机，以及“修订包 2”之前的任何版本 7 客户机。版本 5 客户机不受影响。

建议的修正是升级所有受影响的客户机。必须将版本 6 客户机升级为“修订包 6”或更新版本，而将版本 7 客户机升级为“修订包 2”或更新版本。

Netscape CD 未与 DB2 UDB 一起交付

Netscape CD 不再与 DB2 UDB 一起交付。Netscape 产品可以从 <http://www.netscape.com> 处获得。

XML 自述文件中的错误

在 DB2 XML Extender 版本 7.1 的 README.TXT 文件的“注意事项”中作了如下描述：

3. DB2 UDB 的缺省版本是 DB2 UDB 版本 7.1。如果想要在 AIX 和 Solaris 上使用 DB2 UDB 版本 6.1, 则应确保您正在运行的是 DB2 UDB V6.1 实例和 DB2 UDB V6.1 库。

这是不正确的。只有 **DB2 版本 7.1 和 7.2** 才支持 **DB2 XML Extender**。

文件 readme.aix、readme.nt 和 readme.sun 列示了下列产品的软件需求:

- 具有 FP1_U465423 或更高版本的 DB2 UDB 6.1 (AIX)
- 安装了“修订包 3”的 DB2 通用数据库版本 6.1 或更高版本 (NT)
- 具有“修订包 FP1_U465424”或更高版本的 DB2 UDB 版本 6.1 (Sun)

这是不正确的。 **DB2 XML Extender** 要求 **DB2 版本 7.1 或 7.2**。

在 Linux S/390 版上可能出现的数据损失

当在具有 2.2 系列内核的 Linux S/390 版上使用 DB2 时, 应该将 Linux 机器上可用的 RAM 量限制为小于 1 GB。将 RAM 限制为 1 GB 将避免因 Linux 内核故障而导致 DB2 中可能的数据损失。

这只会影响 Linux S/390 版上的 DB2, 而不会影响 Intel 上的 Linux。

http://www10.software.ibm.com/developerworks/opensource/linux390/alpha_src.html 处提供了内核补丁, 安装它之后就可以使用超过 1 GB 的 RAM。

Windows 2000 上的 DB2 UDB

在这些“发行说明”中, 在提及 Windows NT 时也包括 Windows 2000, 除非另有所指。

联机文档 (HTML、PDF 和 Search)

Windows 2000 操作系统上受支持的 Web 浏览器

我们建议您在 Windows 2000 上使用 Microsoft Internet Explorer。

如果使用的是 Netscape, 请注意以下几点:

- 在 Windows 2000 上使用 Netscape, DB2 联机信息搜索可能会花很长时间才能完成。Netscape 将使用所有可用的 CPU 资源, 且可能会运行不稳定。在等待搜索结果最终返回时, 我们建议您在提交搜索后, 单击另一窗口, 更改鼠标焦点。搜索结果会在一段合理的时间范围内返回。
- 您可能注意到, 在请求帮助时, 它会正确显示在 Netscape 浏览器窗口中, 但如果您让浏览器窗口保持打开状态, 并在稍后从“控制中心”的另一部分请求帮助, 则浏览器中的内容不会更改。如果关闭该浏览器窗口并再次请求帮助, 这次会出现正确的帮助信息。通过遵循第10页的『试图启动 Netscape 时的错误消息』中的步骤, 可能可以修正该问题。通过在请求“控制中心”的其他帮助前, 关闭该浏览器窗口, 也可解决该问题。
- 在请求“控制中心”的帮助或“信息中心”的主题时, 可能会看到错误消息。要修正该问题, 应遵循第10页的『试图启动 Netscape 时的错误消息』中的步骤。

在 Solaris 上搜索 DB2 联机信息

如果在 Solaris 上搜索 DB2 联机信息时有任何问题, 应检查 /etc/system 中的系统内核参数。以下是 DB2 搜索系统 NetQuestion 所需的最小内核参数:

```
semsys:seminfo_semmni 256
semsys:seminfo_semmmap 258
semsys:seminfo_semmns 512
semsys:seminfo_semmnu 512
semsys:seminfo_semmsl 50
shmsys:shminfo_shmmax 6291456
shmsys:shminfo_shmseg 16
shmsys:shminfo_shmmni 300
```

要设置内核参数, 在 /etc/system 末尾添加一行, 如下所示:

```
set <semaphore_name> = value
```

必须重新引导系统以使所有新的或更改过的值生效。

切换“NetQuestion OS/2 版”以使用 TCP/IP

有关将 NetQuestion 切换为在 OS/2 系统上使用 TCP/IP 的指导是不完整的。这些指导中所提及的 *.cfg 文件的位置是 NetQuestion 安装目录的数据子目录。您可通过输入下列其中一个命令来确定 NetQuestion 安装目录:

```
echo %IMNINSTSRV% //对于 SBCS 安装
echo %IMQINSTSRV% //对于 DBCS 安装
```

试图启动 Netscape 时的错误消息

如果在试图启动 Netscape 时遇到以下错误消息:

找不到文件 <文件路径> (或其组件之一)。
进行检查以确保路径和文件名正确,且所有必需的库可用。

无法打开 "D:\Program Files\SQLLIB\CC\..\doc\html\db2help\XXXXX.htm"

应遵循以下步骤,在 Windows NT、95、或 98 上校正此问题(有关应在 Windows 2000 上采取的措施,参见以下内容):

1. 从“开始”菜单中,选择“程序”→“Windows 资源管理器”。“Windows 资源管理器”打开。
2. 在“Windows 资源管理器”中,选择“查看”→“选项”。“选项笔记本”打开。
3. 单击**文件类型**选项卡。“文件类型”页打开。
4. 在**已注册的文件类型**字段中突出显示 Netscape Hypertext Document,然后单击**编辑**。“编辑文件类型”窗口打开。
5. 在**操作**字段中突出显示“打开”。
6. 单击**编辑**按钮。“编辑这种类型的操作”窗口打开。
7. 取消对**使用 DDE**复选框的选择。
8. 在**用来执行操作的应用程序**字段中,确保“%1”出现在字符串(包括引号,且在第一个引号之前有一个空格)的最末尾。

如果在 Windows 2000 上遇到该错误消息,应遵循以下步骤:

1. 从**开始**菜单中,选择 **Windows 资源管理器**。“Windows 资源管理器”打开。
2. 在“Windows 资源管理器”中,选择**工具**→**文件夹选项**。“文件夹选项笔记本”打开。
3. 单击**文件类型**选项卡。
4. 在“文件类型”页,已注册的文件类型字段中,突出显示: **HTML Netscape Hypertext Document** 然后单击**高级**。“编辑文件类型”窗口打开。
5. 在**操作**字段中突出显示“打开”。
6. 单击**编辑**按钮。“编辑这种类型的操作”窗口打开。
7. 取消对**使用 DDE**复选框的选择。
8. 在**用来执行操作的应用程序**字段中,确保“%1”出现在字符串(包括引号,且在第一个引号之前有一个空格)的最末尾。
9. 单击**确定**。
10. 对 **HTML Netscape Hypertext Document** 和 **SHTML Netscape Hypertext Document** 文件类型重复步骤 4 至 8。

基于 UNIX 的系统上 Adobe Acrobat Reader 的配置需求

在基于 UNIX 的平台上只提供了 Acrobat Reader 的英语版本,所以在试图打开非英语版本的其他语言环境的 PDF 文件时可能会返回错误。这些错误指示 PDF 文件存在字体访问或抽取问题,但实际上是因为英语版 Acrobat Reader 不能在 UNIX 非英语语言环境中正确工作。

要查看这样的 PDF 文件，可通过在启动英语版 Acrobat Reader 之前执行下列步骤之一，切换至英语语言环境：

- 通过在启动脚本文件中的 `#!/bin/sh` 语句后面添加以下行，编辑 Acrobat Reader 的启动脚本：

```
LANG=C;export LANG
```

此方法将确保 Acrobat Reader 被其他应用程序（如 Netscape Navigator）或应用程序帮助菜单启动时能正确运行。

- 在命令提示处输入 `LANG=C`，以将 Acrobat Reader 的应用程序环境设置为英语。

要获取进一步的信息，请与 Adobe Systems (<http://www.Adobe.com>) 联系。

SQL Reference 是在一个 PDF 文件中提供的

每本书中的『使用 DB2 资料库』附录都指示，提供了 PDF 格式的 SQL Reference（分为单独的两卷）。这是不正确的。

尽管打印书籍以两卷的形式出现，且相应的两个书号是正确的，但是仅有一个 PDF 文件，它包含有两卷。该 PDF 文件名为 `db2s0x70`。

第2部分 安装与配置

一般安装信息	15	DLFM 设置配置文件选项	29
下载所有受支持 DB2 客户机的安装软件包	15	当在 AIX 上运行 Data Links/DFS 脚本	
在 Windows 2000 上安装 DB2	15	dmapp_prestart 时出错	29
有关使用专用寄存器定义的视图的迁移问题	15	Tivoli Space Manager 与 Data Links 集成	30
Windows 2000 上的 IPX/SPX 协议支持	15	约束与限制	30
在升级 DB2 的先前版本前, 停止 DB2 进程	15	第 4 章 安装和配置 DB2 Data Links Manager AIX	
在安装 DB2 之后运行 db2iupdt (如果已经安装了另		版	31
一 DB2 产品的话)	16	普通安装说明	31
设置要运行 DB2 控制中心的 Linux 环境	16	从 DB2 File Manager 版本 5.2 迁移至 DB2	
S/390 上的 DB2 通用数据库企业版和 DB2 Connect		Data Links Manager 版本 7	31
企业版的 Linux 版	16	安装和配置补遗	33
DB2 通用数据库扩充企业版 UNIX 版快速入门	17	第 5 章 在 UNIX 操作系统上安装 DB2 客户机	33
HP-UX 的 shmseg 内核参数	17	HP-UX 内核配置参数	33
迁移 IBM Visual Warehouse 控制数据库	17	第 12 章 运行您自己的应用程序	33
访问仓库控制数据库	17	使用运行时客户机绑定数据库实用程序	33
		使用 ODBC 对 DB2 的 UNIX 客户机访问	33
Data Links Manager 快速入门	19	第 24 章 设置联合系统以访问多个数据源	33
Dlfm 启动失败, 出现消息: “获取用于前缀的 afsfid		联合系统	33
时出错”	19	限制	34
对归档文件设置 Tivoli Storage Manager 类	19	安装 DB2 Relational Connect	34
DFS 客户机使能器的磁盘空间需求	19	在 Windows NT 服务器上安装 DB2 Relational	
在 AIX 上监控 Data Links File Manager 后端进程	19	Connect	34
安装和配置 DB2 Data Links Manager AIX 版:		在 AIX、Linux 和 Solaris 操作环境服务器上安	
DCE-DFS 环境中的其他安装说明	19	装 DB2 Relational Connect	35
“dlfm add_prefix” 命令失败	20	第 26 章 访问 Oracle 数据源	35
安装和配置 DB2 Data Links Manager AIX 版: 使用		文档错误	36
db2setup 实用程序在 AIX 上安装 DB2 Data Links		访问 Sybase 数据源 (新章节)	37
Manager	20	将 Sybase 数据源添加至联合服务器	37
安装和配置 DB2 Data Links Manager AIX 版:		步骤 1: 设置环境变量并更新简要表注册表	37
DCE-DFS 后安装任务	20	步骤 2: 将 DB2 链接至 Sybase 客户机软件	
安装和配置 DB2 Data Links Manager AIX 版: 使用		(仅 AIX 和 Solaris)	38
Smit 人工安装 DB2 Data Links Manager	21	步骤 3: 回收 DB2 实例	38
安装和配置 DB2 Data Links DFS 客户机使能器	21	步骤 4: 创建并设置接口文件	39
安装和配置 DB2 Data Links Manager Solaris 版	21	步骤 5: 创建封装器	39
选择在 AIX 上的 DB2 Data Links Manager 的备份		步骤 6: 可选: 设置 DB2_DJ_COMM 环境变量	39
方法	22	步骤 7: 创建服务器	40
选择在 Solaris 操作环境上的 DB2 Data Links		可选: 步骤 8: 设置 CONNECTSTRING 服务	
Manager 的备份方法	23	器选项	40
选择在 Windows NT 上 DB2 Data Links Manager 的		步骤 9: 创建用户映射	41
备份方法	24	步骤 10: 为表和视图创建别名	42
在 AIX 上备份日志文件系统	25	指定 Sybase 代码页	42
Windows NT 上的 Data Links 中的管理员组特权	26	使用 ODBC 来访问 Microsoft SQL Server 数据源	
最小化 Data Links File System Filter (DLFF) 安装的		(新章节)	42
记录	26	将 Microsoft SQL Server 数据源添加至联合服务器	
安装后记录消息	27	步骤 1: 设置环境变量 (仅 AIX)	43
使用 SMIT 卸载 DLFM 组件时可能会除去附加文件		步骤 2: 运行 shell 脚本 (仅 AIX)	44
集	27	步骤 3: 可选: 设置 DB2_DJ_COMM 环境变量	44
在开始 / 确定主机名之前	27	步骤 4: 回收 DB2 实例 (仅 AIX)	44
使用 Data Links File Manager: 在从 DB2 数据库中		步骤 5: 创建封装器	45
删除 DB2 Data Links Manager 之后进行清理	28	步骤 6: 创建服务器	45
DLFM1001E (新的错误消息)	29		

	步骤 7: 创建用户映射	46
	步骤 8: 为表和视图创建别名	46
	步骤 9: 可选: 获取 ODBC 跟踪	47
	复查 Microsoft SQL Server 代码页	47

一般安装信息

下载所有受支持 DB2 客户机的安装软件包

要下载所有受支持 DB2 客户机（包括版本 7 以前的所有客户机）的安装软件包，连接至 IBM DB2 Client Application Enabler Pack Web 站点，网址为 <http://www.ibm.com/software/data/db2/db2tech/clientpak.html>

在 Windows 2000 上安装 DB2

在 Windows 2000 上，如果是在先前版本的 DB2 上进行安装或是重新安装当前版本，则要确保所有 DB2 服务的恢复选项都被设置为“不采取任何操作”。

有关使用专用寄存器定义的视图的迁移问题

如果专用寄存器 USER 或 CURRENT SCHEMA 用来定义视图列的话，则视图将在数据库迁移之后变得不可用。例如：

```
create view v1 (c1) as values user
```

在版本 5 中，USER 和 CURRENT SCHEMA 为数据类型 CHAR(8)，但自版本 6 开始，它们已被定义为 VARCHAR(128)。在此示例中，如果视图是在版本 5 中创建的，则列 c1 的数据类型为 CHAR，且在数据库迁移之后它仍为 CHAR。如果是在迁移之后使用该视图，它将在运行时进行编译，但会失败，原因是数据类型不匹配。

解决方案是删除该视图，然后重新创建它。在删除该视图之前，通过查询 SYSCAT.VIEWS 目录视图来捕捉用来创建该视图的语法。例如：

```
select text from syscat.views where viewname='<>'
```

Windows 2000 上的 IPX/SPX 协议支持

此信息是指《快速入门》中的『安装规划』一章中称为『可能的客户机与服务器的连接方案』的一节。

所发布的协议支持表不完全正确。不支持使用 IPX/SPX 与任何基于 OS/2 或 UNIX 的服务器连接的 Windows 2000 客户机。使用 IPX/SPX 与 Windows 2000 服务器连接的任何基于 OS/2 或 UNIX 的客户机也不受支持。

在升级 DB2 的先前版本前，停止 DB2 进程

此信息是指《DB2 Windows 版快速入门》一书中的迁移信息。

如果正在升级运行于 Windows 机器上的先前版本的 DB2，则安装程序会提供一个警告，该警告包含将 DB2 DLL 保存在内存中的进程的列表。此时，您可选择人工停止出现在该列表中的进程，或您可让安装程序自动停止这些进程。建议您在安装之前人工停止所有的 DB2 进程，以避免数据丢失。确保 DB2 进程未在运行的最好方法是

通过“Windows 服务”屏面查看系统的进程。在“Windows 服务”屏面中，确保没有任何 DB2 服务、OLAP 服务或“数据仓库中心服务”在运行。

注：一次仅可有一个版本的 DB2 运行于 Windows 平台上。例如，不能让 DB2 版本 7 和 DB2 版本 6 运行于同一 Windows 机器上。如果在安装有 DB2 版本 6 的机器上安装 DB2 版本 7，安装程序将在安装期间删除 DB2 版本 6。参考适当的《快速入门》手册以了解有关从先前版本的 DB2 进行迁移的详情。

在安装 DB2 之后运行 db2iupdt (如果已经安装了另一 DB2 产品的话)

在《快速入门》安装文档中已经提供了以下信息。

如果在基于 UNIX 的系统上安装 DB2 UDB 版本 7，而先前已经安装了一个 DB2 产品，您将需要运行 **db2iupdt** 命令以更新您打算将其与本产品的新功能部件配合使用的那些实例。在运行此命令之前，某些功能部件将不可用。

设置要运行 DB2 控制中心的 Linux 环境

此信息应该包括在《快速入门》中的『安装 DB2 控制中心』一章中。

在脱离 Linux 上的 DB2 安装程序并返回至终端窗口之后，输入下列命令以设置运行“DB2 控制中心”的正确环境：

```
su -l <instance name>
export JAVA_HOME=/usr/jdk118
export DISPLAY=<your machine name>:0
```

然后，打开另一终端窗口，并输入：

```
su root
xhost +<your machine name>
```

关闭该终端窗口并返回至您以实例所有者标识登录的终端，并输入以下命令：

```
db2cc
```

来启动“控制中心”。

S/390 上的 DB2 通用数据库企业版和 DB2 Connect 企业版的 Linux 版

现在，“DB2 通用数据库企业版”和“DB2 Connect 企业版”可供 S/390 上的 Linux 使用。在 S/390 机器上安装 Linux 之前，应了解软件和硬件需求：

硬件

S/390 9672 Generation 5 或更高版本，Multiprise 3000。

软件

- SuSE Linux v7.0 S/390 版或者 Turbolinux Server 6 zSeries 版和 S/390 版
- 内核级别 2.2.16，带有用于 S/390 的补丁（如下所示）
- glibc 2.1.3
- libstdc++ 6.1

下列补丁是 S/390 上的 Linux 所需的：

- 此时不需要任何补丁。

要获取最新的更新信息，可访问 Web 站点 <http://www.software.ibm.com/data/db2/linux>。

注：

1. 仅 32 位基于 Intel 的 Linux 和 S/390 上的 Linux 才是受支持的。
2. 在“DB2 版本 7”中，下列各项在 Linux/390 上不可用：
 - DB2 UDB 扩充企业版
 - DB2 Extender
 - Data Links Manager
 - DB2 管理客户机
 - 更改口令支持
 - LDAP 支持

DB2 通用数据库扩充企业版 UNIX 版快速入门

『第 5 章 在 Linux 上安装和配置 DB2 通用数据库』应该指出，Linux EEE 群集中的每个物理节点都必须具有相同的内核、glibc 和 libstdc++ 级别。

可以从以下 Web 站点来下载“DB2 EEE Linux 版”的测试版：
<http://www6.software.ibm.com/dl/db2udbd1/db2udbd1-p>

HP-UX 的 shmseg 内核参数

《快速入门》一书中提供的关于更新 HP-UX 内核配置参数的信息是不正确的。应该忽略 HP-UX 的 shmseg 内核参数的建议值。

而应该使用缺省的 HP-UX 值 (120)。

迁移 IBM Visual Warehouse 控制数据库

*DB2 通用数据库快速入门 Windows 版*提供了有关在 Windows NT 和 Windows 2000 上进行“DB2 通用数据库版本 7”的典型安装期间如何迁移活动的仓库控制数据库的信息。如果要迁移多个数据仓库控制数据库，必须使用“数据仓库控制数据库管理”窗口来迁移其他数据库。一次仅可有一个仓库控制数据库是活动的。如果迁移的上一个数据库并非您打算在下一次登录“数据仓库中心”时使用的数据库，则必须使用“仓库控制数据库管理”窗口来注册您打算使用的数据库。

访问仓库控制数据库

在 Windows NT 上进行的 DB2 版本 7 典型安装中，DB2 版本 7 仓库控制数据库是随仓库服务器创建的。如果您有 Visual Warehouse 仓库控制数据库，则必须将包含该仓库控制数据库的 DB2 服务器升级为 DB2 版本 7，才能迁移仓库控制数据库中的元数据以供“DB2 版本 7 数据仓库中心”使用。必须将想要继续使用的所有仓库控制数据库迁移至版本 7。活动仓库控制数据库中的元数据会在 DB2 版本 7 安装过程期间迁移为版本 7。要迁移所有其他仓库控制数据库中的元数据，使用“仓库控制数据库迁移”

实用程序，您可以通过在 Windows NT 上选择**开始** → **程序** → **IBM DB2** → **仓库控制数据库管理**来启动该实用程序。有关迁移仓库控制数据库的信息，参见 *DB2 通用数据库 Windows 版快速入门*。

Data Links Manager 快速入门

Dlfm 启动失败, 出现消息: “获取用于前缀的 afsfid 时出错”

对于运行于 DCE-DFS 环境中的 Data Links Manager, 如果 dlfm 启动因以下错误而失败, 则与“IBM 服务机构”联系:

获取用于前缀的 afsfid 时出错

当使用 "dlfm add_prefix" 注册至 Data Links Manager 的 DFS 文件集被删除时, 可能会出现该错误。

对归档文件设置 Tivoli Storage Manager 类

要指定哪个 TSM 管理类将用于归档文件, 将 DLFM_TSM_MGMTCLASS DB2 注册表条目设置为适当的管理类名。

DFS 客户机使能器的磁盘空间需求

“DFS 客户机使能器”是一个可选组件, 可在“DB2 通用数据库”客户机或服务器安装期间选择它。不能在未安装“DB2 通用数据库”客户机或服务器产品的情况下安装“DFS 客户机使能器”, 即使“DFS 客户机使能器”自行运行而不需要 DB2 UDB 客户机或服务器亦如此。除“DFS 客户机使能器”代码所需的 2 MB 磁盘空间之外, 您还应再留出 40 MB (如果是将“DFS 客户机使能器”作为“DB2 运行时客户机”安装的一部分来安装的话)。如果将“DFS 客户机使能器”作为“DB2 管理客户机”或 DB2 服务器安装的一部分来安装, 将需要更多的磁盘空间。有关“DB2 通用数据库”产品的磁盘空间需求的详情, 参考 *DB2 (UNIX 版)《快速入门》* 手册。

在 AIX 上监控 Data Links File Manager 后端进程

已对 `dlfm see` 命令的输出作了更改。当发出了此命令以在 AIX 上监控 Data Links File Manager 后端进程时, 所返回的输出将类似如下内容:

PID	PPID	PGID	RUNAME	UNAME	ETIME	DAEMON NAME
17500	60182	40838	dlfm	root	12:18	dlfm_copyd_(dlfm)
41228	60182	40838	dlfm	root	12:18	dlfm_chownd_(dlfm)
49006	60182	40838	dlfm	root	12:18	dlfm_upcall_(dlfm)
51972	60182	40838	dlfm	root	12:18	dlfm_gcd_(dlfm)
66850	60182	40838	dlfm	root	12:18	dlfm_retrieved_(dlfm)
67216	60182	40838	dlfm	dlfm	12:18	dlfm_delgrp_(dlfm)
60182	1	40838	dlfm	dlfm	12:18	dlfmd_(dlfm)

DLFM SEE request was successful.

括在括号中的名称是 dlfm 实例的名称, 在此例中为 "dlfm"。

安装和配置 DB2 Data Links Manager AIX 版: DCE-DFS 环境中的其他安装说明

在『安装先决条件』一节中, 应添加以下新信息:

还必须安装 DFS 3.1 的电子修正或 PTF set 1 (在它成为可用时)。
电子修正可从以下网址处获取：
http://www.transarc.com/Support/dfs/Data Links/efix_dfs31_main_page.html

而且：

必须先运行 dfs 客户机，才能安装 Data Links Manager。
使用 db2setup 或 smitty。

在『键表文件』一节中有一个错误，应校正为如下内容：

包含负责人和口令信息的键表文件应称为 dataLink.ktb 且

以下示例中使用正确名称：dataLink.ktb。『键表文件』一节应移到“DCE-DFS 后安装任务”下面，因为在创建 DLMADMIN 实例之前不会创建此文件。

在『Data Links File Manager 服务器和客户机』一节中，应注意必须先安装 Data Links Manager 服务器，而后才安装任何 Data Links Manager 客户机。

应添加一个新节『备份目录』：

如果备份方法是用于本地文件系统的，则这必须是 DFS 文件系统目录。
确保此 DFS 文件集是 DFS 管理员创建的。这不应该是 DMLFS 文件集。

“dlfm add_prefix” 命令失败

对于运行于 DCE/DFS 环境中的 Data Links Manager，**dlfm add_prefix** 命令可能会失败，且返回码为 -2061 (备份失败)。如果发生这种情况，执行下列步骤：

1. 通过发出 **dlfm stop** 命令来停止 Data Links Manager 守护程序。
2. 通过发出 **dlfm stopdbm** 命令来停止 DB2 进程。
3. 通过发出 **dce_login root** 命令来获取 dce root 用户凭证。
4. 通过发出 **dlfm startdbm** 命令来启动 DB2 进程。
5. 通过发出 **dlfm add_prefix** 命令来向 Data Links Manager 注册文件集。
6. 通过发出 **dlfm start** 命令来启动 Data Links Manager 守护程序进程。

安装和配置 DB2 Data Links Manager AIX 版：使用 db2setup 实用程序 在 AIX 上安装 DB2 Data Links Manager

在『创建的 DB2 数据库 DLFM_DB』一节中，DLFM_DB 并非在 DCE_DFS 环境中创建的。这必须作为后安装步骤完成。

在『为 DMAPP 进行 DCE-DFS 预启动注册』一节中，步骤 2 应更改为以下内容：

2. 命令被添加至 /opt/dcelocal/tcl/user_cmd.tcl
以便确保在启动 DFS 时将启动 DMAPP。

安装和配置 DB2 Data Links Manager AIX 版：DCE-DFS 后安装任务

应添加以下一个新节“完成 Data Links Manager 安装”：

在 Data Links Manager 服务器上，必须执行下列步骤来完成安装：

1. 按『安装和配置 DB2 Data Links Manager AIX 版』一章的『DCE-DFS 环境中的附加安装说明』一节的“键表文件”中所述来创建键表文件。

2. 作为 root 用户，输入下列命令来启动 DMAPP:

```
stop.dfs all
start.dfs all
```

3. 使用 dce root 用户凭证来运行 "dlfm setup"，如下所示:
 - a. 作为 Data Links Manager 管理员 DLMADMIN 登录。
 - b. 作为 root 用户，发出 dce_login。
 - c. 发出命令: dlfm setup。

在 Data Links Manager 客户机上，必须执行下列步骤来完成安装:

1. 按『安装和配置 DB2 Data Links Manager AIX 版』一章的『DCE-DFS 环境中的附加安装说明』一节的“键表文件”中所述来创建键表文件。
2. 作为 root 用户，输入下列命令来启动 DMAPP:

```
stop.dfs all
start.dfs all
```

安装和配置 DB2 Data Links Manager AIX 版: 使用 Smit 人工安装 DB2 Data Links Manager

在『SMIT 后安装任务』一节中，修改步骤 7 来指示必须在发出 "dlfm setup" 之前先发出 "dce_login root" 命令。不需要步骤 11。此步骤是在步骤 6 (dlfm server_conf) 或步骤 8 (dlfm client_conf) 完成时自动执行的。还要除去步骤 12 (dlfm start)。要完成安装，执行下列步骤:

1. 按『安装和配置 DB2 Data Links Manager AIX 版』一章的『DCE-DFS 环境中的附加安装说明』一节的“键表文件”中所述来创建键表文件。
2. 作为 root 用户，输入下列命令来启动 DMAPP:

```
stop.dfs all
start.dfs all
```

安装和配置 DB2 Data Links DFS 客户机使能器

在『配置 DFS 客户机使能器』一节中，将下列信息添加至步骤 2:

通常，执行 "secval" 命令将会完成配置。但是，可能还需要重新引导机器。

若在访问 READ PERMISSION DB 文件时遇到问题，则重新引导刚刚安装了“DB2 DFS 客户机使能器”的机器。

安装和配置 DB2 Data Links Manager Solaris 版

在安装“DB2 Data Links Manager Solaris 版”之后必须执行下列操作:

1. 将下列三行添加至 /etc/system 文件:

```
set dlfsdrv:glob_mod_pri=0x100800
set dlfsdrv:glob_mesg_pri=0xff
set dlfsdrv:ConfigDlfsUId=UID
```

其中，UID 表示 id dlfm 的用户标识。

2. 重新引导机器以激活更改。

选择在 AIX 上的 DB2 Data Links Manager 的备份方法

除“磁盘复制”和 XBSA 之外，还可使用 Tivoli Storage Manager(TSM) 来备份驻留在 Data Links 服务器上的文件。

要将 Tivoli Storage Manager 作归档服务器:

1. 在 Data Links 服务器上安装 Tivoli Storage Manager。有关详情，参考 Tivoli Storage Manager 产品文档。
2. 向 Tivoli Storage Manager 服务器注册 Data Links 服务器客户机应用程序。有关详情，参考 Tivoli Storage Manager 产品文档。
3. 将下列环境变量添加至“Data Links Manager 管理员”的 db2profile 或 db2cshrc 脚本文件:

```
(对于 Bash、 Bourne 或 Korn shell)
export DSMI_DIR=/usr/tivoli/tsm/client/api/bin
export DSMI_CONFIG=$HOME/tsm/dsm.opt
export DSMI_LOG=$HOME/dldump
export PATH=$PATH:$DSMI_DIR
```

```
(对于 C shell)
setenv DSMI_DIR /usr/tivoli/tsm/client/api/bin
setenv DSMI_CONFIG ${HOME}/tsm/dsm.opt
setenv DSMI_LOG ${HOME}/dldump
setenv PATH=${PATH}:$DSMI_DIR
```

4. 确保 dsm.sys TSM 系统选项文件位于 \$DSMI_DIR 目录中。
5. 确保 dsm.opt TSM 用户选项文件位于 *INSTHOME*/tsm 目录之中，其中 *INSTHOME* 是“Data Links Manager 管理员”的主目录。
6. 在 /usr/tivoli/tsm/client/api/bin/dsm.sysTivoli Storage Manager 系统选项文件中，将 *PASSWORDACCESS* 选项设置为 generate。
7. 在首次启动 Data Links File Manager 之前，向 generate 选项注册 TSM 口令。这样，您将不必在 Data Links File Manager 启动与 TSM 服务器的连接时提供口令。有关详情，参考 TSM 产品文档。
8. 将 *DLFM_BACKUP_TARGET* 注册表变量设置为 TSM。在这种情况下，将被忽略 *DLFM_BACKUP_DIR_NAME* 注册表变量的值。这将激活 Tivoli Storage Manager 备份选项。

注:

- a. 如果在运行时更改了 TSM 与磁盘之间的 *DLFM_BACKUP_TARGET* 注册表变量设置，您应该知道归档文件未被移至新指定的归档位置。例如，如果在启动 Data Links File Manager 时 *DLFM_BACKUP_TARGET* 注册表值设置为 TSM，且注册表值更改为磁盘位置，则所有新归档的文件将存储在磁盘上的新位置。先前归档至 TSM 的文件将不会移至新的磁盘位置。
 - b. 要覆盖缺省 TSM 管理类，可使用一个新注册表变量 *DLFM_TSM_MGMTCLASS*。如果未设置此注册表变量，则将使用缺省 TSM 管理类。
9. 通过输入 **dlfm stop** 命令来停止 Data Links File Manager。
 10. 通过输入 **dlfm start** 命令来启动 Data Links File Manager。

选择在 Solaris 操作环境上的 DB2 Data Links Manager 的备份方法

除“磁盘复制”和 XBSA 之外，还可使用 Tivoli Storage Manager(TSM) 来备份驻留在 Data Links 服务器上的文件。

要将 Tivoli Storage Manager 作归档服务器:

1. 在 Data Links 服务器上安装 Tivoli Storage Manager。有关详情，参考 Tivoli Storage Manager 产品文档。
2. 向 Tivoli Storage Manager 服务器注册 Data Links 服务器客户机应用程序。有关详情，参考 Tivoli Storage Manager 产品文档。
3. 将下列环境变量添加至“Data Links Manager 管理员”的 db2profile 或 db2cshrc 脚本文件:

```
(对于 Bash、 Bourne 或 Korn shell)
export DSMI_DIR=/opt/tivoli/tsm/client/api/bin
export DSMI_CONFIG=$HOME/tsm/dsm.opt
export DSMI_LOG=$HOME/dldump
export PATH=$PATH:/opt/tivoli/tsm/client/api/bin
```

```
(对于 C shell)
setenv DSMI_DIR /opt/tivoli/tsm/client/api/bin
setenv DSMI_CONFIG ${HOME}/tsm/dsm.opt
setenv DSMI_LOG ${HOME}/dldump
setenv PATH=${PATH}:/opt/tivoli/tsm/client/api/bin
```

4. 确保 dsm.sys TSM 系统选项文件位于 /opt/tivoli/tsm/client/api/bin 目录中。
5. 确保 dsm.opt TSM 用户选项文件位于 *INSTHOME*/tsm 目录之中，其中 *INSTHOME* 是“Data Links Manager 管理员”的主目录。
6. 在 /opt/tivoli/tsm/client/api/bin/dsm.sys Tivoli Storage Manager 系统选项文件中将 *PASSWORDACCESS* 选项设置为 generate。
7. 在首次启动 Data Links File Manager 之前，向 generate 选项注册 TSM 口令。这样，您将不必在 Data Links File Manager 启动与 TSM 服务器的连接时提供口令。有关详情，参考 TSM 产品文档。
8. 将 *DLFM_BACKUP_TARGET* 注册表变量设置为 TSM。在这种情况下，将被忽略 *DLFM_BACKUP_DIR_NAME* 注册表变量的值。这将激活 Tivoli Storage Manager 备份选项。

注:

- a. 如果在运行时更改了 TSM 与磁盘之间的 *DLFM_BACKUP_TARGET* 注册表变量设置，您应该知道归档文件未被移至新指定的归档位置。例如，如果在启动 Data Links File Manager 时 *DLFM_BACKUP_TARGET* 注册表值设置为 TSM，且注册表值更改为磁盘位置，则所有新归档的文件将存储在磁盘上的新位置。先前归档至 TSM 的文件将不会移至新的磁盘位置。
 - b. 要覆盖缺省 TSM 管理类，可使用一个新注册表变量 *DLFM_TSM_MGMTCLASS*。如果未设置此注册表变量，则将使用缺省 TSM 管理类。
9. 通过输入 **dlfm stop** 命令来停止 Data Links File Manager。
 10. 通过输入 **dlfm start** 命令来启动 Data Links File Manager。

选择在 Windows NT 上 DB2 Data Links Manager 的备份方法

每当 DATALINK 值被插入到带有为进行恢复而定义的 DATALINK 列的表中时，会调度 Data Links 服务器上的相应 DATALINK 文件来备份至归档服务器。目前，“磁盘复制”（缺省方法）和 Tivoli Storage Manager 是将文件备份至归档服务器所支持的两个选项。DB2 Data Links Manager Windows NT 版的将来发行版将支持其他供应商的备份媒体和软件。

磁盘复制（缺省方法）

在 DB2 服务器上输入 **backup** 命令时，将确保数据库中的链接文件被备份至 Data Links 服务器上由 `DLFM_BACKUP_DIR_NAME` 环境变量所指定的目录。此变量的缺省值是 `c:\dlfmbackup`，其中 `c:\` 表示 Data Links Manager 备份安装驱动器。

要将此变量设置为 `c:\dlfmbackup`，输入以下命令：

```
db2set -g DLFM_BACKUP_DIR_NAME=c:\dlfmbackup
```

由 `DLFM_BACKUP_DIR_NAME` 环境变量指定的位置一定不能位于使用 Data Links Filesystem Filter 的文件系统上，且您对备份文件指定的目录上有足够的所需空间。

而且，通过输入以下命令来确保 `DLFM_BACKUP_TARGET` 变量被设置为 LOCAL：

```
db2set -g DLFM_BACKUP_TARGET=LOCAL
```

在设置或更改这些变量之后，使用 **dlfm stop** 和 **dlfm start** 命令停止并重新启动 Data Links File Manager。

Tivoli Storage Manager

要将 Tivoli Storage Manager 作归档服务器：

1. 在 Data Links 服务器上安装 Tivoli Storage Manager。有关详情，参考 Tivoli Storage Manager 产品文档。
2. 向 Tivoli Storage Manager 服务器注册 Data Links 服务器客户机应用程序。有关详情，参考 Tivoli Storage Manager 产品文档。
3. 单击**开始**并选择**设置** → **控制面板** → **系统**。“系统属性”窗口打开。选择**环境**选项卡并输入下列环境变量和相应值：

变量	值
DSMI_DIR	c:\tsm\baclient
DSMI_CONFIG	c:\tsm\baclient\dsm.opt
DSMI_LOG	c:\tsm\dlldump

4. 确保 `dsm.sys` TSM 系统选项文件位于 `c:\tsm\baclient` 目录之中。
5. 确保 `dsm.opt` TSM 用户选项文件位于 `c:\tsm\baclient` 目录之中。
6. 在 `c:\tsm\baclient\dsm.sys` Tivoli Storage Manager 系统选项文件中，将 `PASSWORDACCESS` 选项设置为 `generate`。
7. 在首次启动 Data Links File Manager 之前，向 `generate` 选项注册 TSM 口令。这样，您将不必在 Data Links File Manager 启动与 TSM 服务器的连接时提供口令。有关详情，参考 TSM 产品文档。

8. 使用以下命令将 `DLFM_BACKUP_TARGET` 环境变量设置为 `TSM`:

```
db2set -g DLFM_BACKUP_TARGET=TSM
```

在这种情况下，`DLFM_BACKUP_DIR_NAME` 环境变量的值将被忽略。这将激活 Tivoli Storage Manager 备份选项。

注:

- a. 如果在运行时更改了 `TSM` 和 `LOCAL` 之间的 `DLFM_BACKUP_TARGET` 环境变量设置，则您应该知道归档文件未被移至新指定的归档位置。例如，如果在 `DLFM_BACKUP_TARGET` 环境变量设置为 `TSM` 时启动 Data Links File Manager，并将它的值更改为 `LOCAL`，则所有新归档的文件都将存储在磁盘的新位置上。先前归档至 `TSM` 的文件将不会移至新的磁盘位置。
 - b. 要覆盖缺省 `TSM` 管理类，可使用一个新环境变量 `DLFM_TSM_MGMTCLASS`。如果未设置此变量，则将使用缺省 `TSM` 管理类。
9. 通过输入 `dlfm stop` 命令来停止 Data Links File Manager。
 10. 通过输入 `dlfm start` 命令来启动 Data Links File Manager。

在 AIX 上备份日志文件系统

本书指出必须停止 Data Links Manager，且应对该文件系统进行脱机备份。建议需要较高可用性的用户使用以下方法，该方法不需要停止 Data Links Manager。

1. 访问 CLI 源文件 `quiesce.c` 和 shell 脚本 `online.sh`。这些文件位于 `/samples/dlfm` 目录中。
2. 编译 `quiesce.c`:

```
xlc -o quiesce -L$HOME/sqlllib/lib -I$HOME/sqlllib/include -c quiesce.c
```
3. 作为 `root` 用户，在具有 `DLFS` 文件系统的节点上运行该脚本。

shell 脚本 `online.sh` 假定对于向 DataLink Manager 注册的每个数据库，在 Data Link Manager 上都相应有一个目录条目。它还假定 `/etc/filesystems` 有完整的 `DLFS` 文件系统条目。shell 脚本执行下列操作:

- 停顿向 Data Links Manager 注册的数据库中的所有表。这将停止任何新的活动。
- 卸载文件系统并将该文件系统作为只读文件系统重新进行安装。
- 执行文件系统备份。
- 卸载文件系统并将该文件系统作为读写文件系统重新进行安装。
- 对 `DB2` 表进行复位；即，让它们脱离停顿状态。

必须对脚本做如下修改，以适合于您的环境:

1. 选择备份命令并将其放入到脚本的 `do_backup` 功能中。
2. 在脚本内设置下列环境变量:
 - `DLFM_INST`: 将此项设置为 `DLFM` 实例名。
 - `PATH_OF_EXEC`: 将此项设置为 `"quiesce"` 可执行文件所在的路径。

调用该脚本如下:

```
online.sh <filesystem_name>
```

Windows NT 上的 Data Links 中的管理员组特权

在 Windows NT 上，对于大多数功能，属于管理员组的用户对使用 DataLinks 链接的文件所具有的特权相当于 UNIX 上的 root 用户。下表对两者进行了比较。

操作	Unix (Root 用户)	Windows NT (管理员)
重命名	是	是
在不需要令牌的情况下访问文件	是	是
删除	是	否 (参见下面的注释)
更新	是	否 (参见下面的注释)

注: NTFS 不接受对只读文件执行这些操作。管理员用户可以通过对文件启用写许可权来成功地执行这些操作。

最小化 Data Links File System Filter (DLFF) 安装的记录

可以通过更改 dlfs_cfg 文件来使 Data Links File System Filter (DLFF) 安装的记录最小化。dlfs_cfg 文件被传送到 stload 例程，以装入驱动程序和配置参数。该文件位于 /usr/lpp/db2_07_01/cfg/ 目录中。通过符号链接，还可以在 /etc 目录中找到该文件。dlfs_cfg 文件具有如下格式:

```
d 'driver-name' 'vfs number' 'dlfm id' 'global message priority'  
'global module priority' - 0 1
```

其中:

d d 参数指定要装入的驱动程序。

driver-name

driver-name 是要装入的驱动程序的全路径。例如，“DB2 版本 7”的全路径是 /usr/lpp/db2_07_01/bin/dlfsdrv。驱动程序的名称为 dlfsdrv。

vfs number

这是 /etc/vfs 中的 DLFS 的 vfs 项。

dlfm id

这是 Data Links Manager 管理员的用户标识。

global message priority

这是全局消息优先级

global module priority

这是全局模块优先级

0 1 0 1 是用于为此驱动程序创建非克隆节点的子式数字。节点名是通过将子式数字追加至克隆的驱动程序节点名后创建的。最多可给定五个子式数字 (0-4)。

现实示例可能如下所示:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,255,-1 - 0 1
```

所记录的消息取决于全局消息优先级和全局模块优先级的设置。要最小化记录，可以更改全局消息优先级的值。

有四个消息优先级值可用:


```
#define LOG_EMERGENCY    0x01
#define LOG_TRACING      0x02
#define LOG_ERROR        0x04
#define LOG_TROUBLESHOOT 0x08
```

DLFF 中的大部分消息都将 LOG_TROUBLESHOOT 作为消息优先级。以下是一些备用配置示例:

如果您确实需要紧急情况消息和错误消息,则在 dlfs_cfg 配置文件中将全局消息优先级设置为 5 (1+4):

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,5,-1 - 0 1
```

若只需要错误消息,则将全局消息优先级设置为 4:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,4,-1 - 0 1
```

若不需要对 DLFS 进行记录,则将全局消息优先级设置为 0:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,0,-1 - 0 1
```

安装后记录消息

安装后若需要记录紧急情况、错误和故障诊断消息,则必须修改 dlfs_cfg 文件。dlfs_cfg 文件位于 /usr/lpp/db2_07_01/cfg 目录中。全局消息优先级必须设置为 255 (最大优先级)或 13 (8+4+1)。将优先级设置为 13 (8+4+1) 将记录紧急情况、错误和故障诊断消息。

设置全局消息优先级之后,卸载 DLFS 过滤器文件系统,并重新装入 dlfsdrv 驱动程序,以便在装入时设置新的优先级值。在重新装入 dlfsdrv 驱动程序之后,必须重新安装 DLFS 过滤器文件系统。

注: 对于 dlfsdrv 驱动程序的任何后续装入,在 dlfs_cfg 文件再次更改之前将保留对 dlfs_cfg 的设置。

使用 SMIT 卸载 DLFM 组件时可能会除去附加文件集

在从安装有 Data Links Manager 的 AIX 机器中卸载 DB2 (版本 5、6 或 7) 之前,应遵循下列步骤:

1. 作为 root 用户,使用以下命令创建 /etc/vfs 的副本:

```
cp -p /etc/vfs /etc/vfs.bak
```
2. 卸载 DB2。
3. 作为 root 用户,将 /etc/vfs 替换为在步骤 1 中创建的备份副本:

```
cp -p /etc/vfs.bak /etc/vfs
```

在开始 / 确定主机名之前

必须确定每个 DB2 服务器和 Data Links 服务器的名称。您将需要知道这些主机名才能验证安装。当连接至 DB2 Data Links File Manager 时,DB2 UDB 服务器将在内部将下列信息发送至 DLFM:

- 数据库名
- 实例名
- 主机名

然后，DLFM 将此信息与它的内部表进行比较，以确定是否应该允许连接。仅当已使用 `dlfm add_db` 命令来将此数据库名、实例名和主机名组合向 DLFM 注册时，DLFM 才允许连接。`dlfm add_db` 命令中使用的主机名必须与 DB2 UDB 服务器在内部发送的主机名完全相同。

按如下所示使用所获得的精确主机名：

1. 在 DB2 服务器上输入主机名命令。例如，此命令可能会返回 `db2server`。
2. 根据您使用的平台不同，执行下列操作之一：

- 在 AIX 上，输入 `host db2server` 命令，其中 `db2server` 是在前一步骤中获得的名称。此命令应返回类似于以下内容的输出：

```
db2server.services.com 为 9.11.302.341, 别名: db2server
```

- 在 Windows NT 上，输入 `nslookup db2server` 命令，其中 `db2server` 是在前一步骤获得的名称。此命令应返回类似于以下内容的输出：

```
Server: dnsserv.services.com
Address: 9.21.14.135
Name: db2server.services.com
Address: 9.21.51.178
```

- 在 Solaris 上，输入 `cat /etc/hosts | grep 'hostname'`。如果在 `/etc/hosts` 中指定了主机名但无域名，则会返回类似于以下内容的输出信息：

```
9.112.98.167 db2server loghost
```

如果指定主机名时带有域名，则该命令将返回类似于以下内容的输出信息：

```
9.112.98.167 db2server.services.com loghost
```

当使用 `dlfm add_db` 命令来注册 DB2 UDB 数据库时，对主机名使用 `db2server.services.com`。如果在 `dlfm add_db` 命令中使用了其他任何别名，则 DB2 服务器与 DLFM 的内部连接将失败。

Data Links 服务器是使用 DB2 “使用节点主机名端口 `port_number` 来为数据库 `database_alias` 添加 Data Links Manager” 命令来注册至 DB2 数据库的。

主机名是 Data Links 服务器的名称。在此命令中可以使用 Data Links 服务器的任何有效别名。引用此 Data Links 服务器的 DATALINK 值必须在 URL 值中指定主机名；即，当为 DATALINK 列指定 URL 值时，必须使用与在 “添加 Data Links Manager” 命令中所使用的名称完全一样的名称。使用不同的别名将导致 SQL 语句失败。

使用 Data Links File Manager: 在从 DB2 数据库中删除 DB2 Data Links Manager 之后进行清理

当使用 `DROP DATALINKS MANAGER` 命令从数据库中删除 DB2 Data Links Manager 时，该命令本身不会清理关于 DB2 Data Links Manager 的相应信息。用户可以显式地启动链接至数据库的任何文件的解链操作，以及启动备份信息的无用信息收集。可以使用 `dlfm drop_dlm` 命令来完成此任务。此命令将启动对特定数据库所有信息的异步删除操作。必须运行 DB2 Data Links Manager，此命令才能成功。仅在删除了 DB2 Data Links Manager 之后才可使用此命令，这一点相当重要；否则，将丢失有关 DB2 Data Links Manager 的重要信息，且不能恢复。

要启动特定数据库的解链处理和备份信息的无用信息收集：

1. 向系统登录为 DB2 Data Links Manager Administrator。

2. 发出以下命令:

```
dlfm drop_dlm database instance hostname
```

其中:

database 是远程 DB2 UDB 数据库的名称;
instance 是数据库驻留其下的实例;
hostname 是数据库驻留于其上的 DB2 UDB 服务器的主机名。

3. 注销。

有关显示应在其中使用此命令的上下文的完整使用方案, 参见 *Command Reference*。

已为此命令创建了新的错误码 (参见 『DLFM1001E (新的错误消息)』)。

DLFM1001E (新的错误消息)

DLFM1001E: 进行 drop_dlm 处理时出错。

原因:

Data Links Manager 无法对指定的数据库启动解链和无用信息收集处理。这可能是由于下列任何原因造成的:

- Data Links Manager 未在运行。
- 在命令中指定了数据库、实例和主机名的无效组合。
- 在 Data Links Manager 的其中一个组件服务中有故障。

操作:

执行下列步骤:

1. 确保 Data Links Manager 正在运行。启动 Data Links Manager (如果它尚未在运行的话)。
2. 确保数据库、实例和主机名的组合标识已注册的数据库。可以在 Data Links Manager 上使用 “dlfm 列示已注册的数据库” 命令来执行此操作。
3. 如果错误仍然存在, 则参考 db2diag.log 文件中的信息, 以便了解是否任何组件服务 (例如, “连接管理服务”、“事务管理服务”等等) 已失败。记下 db2diag.log 中的错误码, 并执行对于该错误码而建议的正确操作。

DLFM 设置配置文件选项

已除去 dlfm setup dlfm.cfg 选项。应忽略在文档中对它的任何引用。

当在 AIX 上运行 Data Links/DFS 脚本 dmapp_prestart 时出错

如果当您运行 Data Links/DFS 脚本 dmapp_prestart 时, 命令
/usr/sbin/cfgdmepl -a "/usr/lib/drivers/dmlfs.ext"

失败且返回码为 1, 则应安装 DFS 3.1 ptfset1 以修正 cfgdmepl。

Tivoli Space Manager 与 Data Links 集成

DB2 Data Links Manager 现在将能够利用 Tivoli Space Manager 的功能。Tivoli Space Manager Hierarchical Storage Manager (HSM) 客户机程序会自动将合格的文件迁移至存储器，以便在本地文件系统中维护特定级别的空闲空间。在访问迁移的文件时，它会自动重新调用这些文件，并允许用户迁移和重新调用特定文件。

对于具有需要定期移至第三级存储器的大型文件的文件系统的客户来说，此新功能部件很有用（在此情况下需要定期管理文件系统的空间）。对于许多客户，Tivoli Space Manager 当前提供了多种方法来管理它们的第三级存储器。Tivoli Space Manager 的新的 DB2 Data Links Manager 支持可以更灵活地管理 DATALINK 文件的空间。不必在 DB2 Data Links Manager 文件系统中为可能存储在文件系统中的所有文件预先分配足够的存储空间，Tivoli Space Manager 允许在一段时间之后调整由 Data Links 管理的文件系统的分配，以免在正常使用期间无意中填满了文件系统。

将 Data Links 和 HSM 支持添加至文件系统

当向“层次结构存储器管理”（HSM）注册文件系统时，首先向 HSM 注册它，然后向 Data Links File Manager 注册它。

1. 使用命令 **"dsmmigfs add /fs"** 来向 HSM 注册。
2. 使用命令 **"dlfmfsmd /fs"** 来向 DLM 注册。

文件系统的 Data Links 支持在 HSM 文件系统的 `/etc/filesystems` 的一节中是通过下列条目来反映的：

```
vfs = dlfs
mount = false
options = rw,Basefs=fsm
nodename = -
```

将 Data Links 支持添加至现存的 HSM 文件系统

使用命令 **"dlfmfsmd /fs"** 来向 DLM 注册。

将 HSM 支持添加至现存的 Data Links 文件系统

1. 使用命令 **"dsmmigfs add /fs"** 来向 HSM 注册。
2. 使用命令 **"dlfmfsmd /fs"** 来向 DLM 注册。

从 Data Links-HSM 文件系统中除去 Data Links 支持

使用命令 **"dlfmfsmd -j /fs"** 来除去 Data Links 支持。

从 Data Links-HSM 文件系统中除去 HSM 支持

1. 使用命令 **"dsmmigfs remove /fs"** 来除去 HSM 支持。
2. 使用 **"dlfmfsmd -j /fs"** 命令来除去 Data Links 支持。
3. 使用命令 **"dlfmfsmd /fs"** 来向 DLM 注册。

从 Data Links-HSM 文件系统中除去 Data Links 和 HSM 支持

1. 使用命令 **"dsmmigfs remove /fs"** 来除去 HSM 支持。
2. 使用命令 **"dlfmfsmd -j /fs"** 来除去 Data Links 支持。

约束与限制

此功能当前仅在 AIX 上才受支持。

只能由 **root** 用户来执行选择性迁移 (**dsmmigrate**) 和重新调用 **FC** (读许可权 **DB**) 链接的文件。

只能在“读许可权 **DB**”文件是“DataLink Manager 管理员” (**dlfm**) 的情况下由文件所有者来执行选择性迁移。要访问这样的文件，主机数据库端需要令牌。唯一不需要令牌的用户是“**root**”用户。“**root**”用户较容易执行选择性迁移和重新调用“读许可权 **DB**”文件。**dlfm** 用户只有第一次才能使用有效令牌来迁移 **FC** 文件。(在重新调用之后) 试图第二次进行迁移，但是该操作将失败，将出现错误消息“**ANS1028S** 内部程序错误。请与服务代表联系”。由非 **root** 用户在 **FC** 文件上运行 **dsmmigrate** 将失败。此限制是次要的，因为通常是管理员访问文件服务器上的文件。

stat 和 **statfs** 系统调用将把 **Vfs** 类型作为 **fsm** 而不是 **dlfs** 来显示，尽管 **dlfs** 是在 **fsm** 基础上安装的。

以上行为是针对 **dsmrecaIld** 守护程序的正常功能的，它在文件系统上执行 **statfs** 来检查它的 **Vfs** 类型是不是 **fsm**。

如果具有最小 **inode** 号的文件是通过 **FC** (读许可权 **DB**) 链接的，则命令“**dsmls**”不会显示任何输出

dsmls 命令类似于 **ls** 命令，并列示由 **TSM** 管理的文件。不需要用户操作

第 4 章 安装和配置 DB2 Data Links Manager AIX 版

普通安装说明

从 DB2 File Manager 版本 5.2 迁移至 DB2 Data Links Manager 版本 7

第 3 步中的信息是不正确的。步骤 3 应为如下内容:

"3. 作为 **DLFM** 管理员，运行 **/usr/lpp/db2_07_01/adm/db2dlmmg** 命令。

安装和配置补遗

第 5 章 在 UNIX 操作系统上安装 DB2 客户机

HP-UX 内核配置参数

关于设置 HP-UX 内核参数的建议中说明应该将 `msgmbn` 和 `msgmax` 设置为 65535 或更高，这是不正确的。必须将这两个参数都精确地设置为 65535。

第 12 章 运行您自己的应用程序

使用运行时客户机绑定数据库实用程序

在可将数据库实用程序（导入、导出、重组、命令行处理器）和 DB2 CLI 绑定文件与每个数据库配合使用之前，不能使用“DB2 运行时客户机”来将这些实用程序和文件与该数据库绑定。必须转而使用“DB2 管理客户机”或“DB2 应用程序开发客户机”。

必须将这些数据库实用程序和 DB2 CLI 绑定文件与每个数据库绑定，这些程序和文件才能与该数据库配合使用。在网络环境中，如果使用运行于不同操作系统之上的多个客户机，或这些客户机处于不同的 DB2 版本或服务级别，则必须对每个操作系统和 DB2 版本组合都绑定这些实用程序一次。

使用 ODBC 对 DB2 的 UNIX 客户机访问

第 12 章『运行您自己的应用程序』指出如果是使用 ODBC 客户机应用程序或 ODBC SDK 来安装“ODBC 驱动程序管理器”，则需要更新 `odbcinst.ini`。这并不完全正确。如果是安装“Merant ODBC 驱动程序管理器”产品，则不必更新 `odbcinst.ini`。

第 24 章 设置联合系统以访问多个数据源

联合系统

DB2 联合系统是一种特殊类型的分布式数据库管理系统 (DBMS)。联合系统允许您查询和检索位于其他 DBMS（例如 Oracle、Sybase 和 Microsoft SQL Server）上的数据。SQL 语句可以在单个语句中引用多个 DBMS 或者个别数据库。例如，可以连接位于“DB2 通用数据库”表、Oracle 表和 Sybase 视图中的数据。受支持的 DBMS 包括 Oracle、Sybase、Microsoft SQL Server（用于 Windows NT）以及“DB2 通用数据库”系列的成员（例如 DB2 OS/390 版、DB2 AS/4000 版和 DB2 Windows 版）。

DB2 联合系统由具有 DB2 实例（将充当联合数据库的数据库）的服务器和一个或多个数据源组成。联合数据库包含用来标识数据源及其特征的目录条目。数据源由 DBMS 和数据组成。“DB2 通用数据库”具有一些协议（称为封装器），可以用来访问这些数据源。封装器是联合服务器用来与数据源进行通信以及从数据源中检索数据的机制。使用别名来表示位于数据源中的表和视图。应用程序连接至于联合数据库，就象连接至任何其他 DB2 数据库一样。您使用的封装器取决于运行“DB2 通用数据库”的平台。

在设置联合系统之后，就可以访问数据源中的信息，尽管它在一个大型数据库中。用户和应用程序将请求发送给一个联合数据库，它会检索数据源中的数据。

DB2 联合系统的运行会受到一些限制。在 DB2 版本 7 中，分布式查询仅限于只读操作。另外，不能对别名执行实用程序操作

(LOAD、REORG、REORGCHK、IMPORT、RUNSTATS 等等)。然而，可以使用联通设施来通过使用与数据源相关联的 SQL 方言来将 DDL 和 DML 语句直接提交给 DBMS。

限制

在“修订包 3”中未提供版本 7.2 的新封装器（例如 Linux 和 Solaris 上的 Oracle、AIX 和 Solaris 上的 Sybase 以及 NT 和 AIX 上的 Microsoft SQL Server）；您必须购买 DB2 Relational Connect 版本 7.2。

安装 DB2 Relational Connect

本节提供了有关在将用作联合系统服务器的服务器上安装 DB2 Relational Connect 的指导。

在 Windows NT 服务器上安装 DB2 Relational Connect

在 Windows NT 联合服务器上安装 DB2 Relational Connect 之前：

- 确保已在联合服务器上安装“DB2 通用数据库企业版”或“DB2 通用数据库扩充企业版”。如果打算在分布式请求中包括 DB2 系列数据库，则当您安装“DB2 通用数据库”时必须已经选择了 **DB2 的分布式连接** 数据源选项。要验证是否实现了此选项，检查 **FEDERATED** 参数是否被设置为 **YES**。可通过发出 **GET DATABASE MANAGER CONFIGURATION** 命令来检查此设置，该命令将显示所有参数及其当前设置。

- 确保已经在联合服务器上为数据源（例如 Sybase Open Client）安装了客户机软件。

1. 利用您创建的用来执行安装的用户帐户来登录至系统。
2. 关闭正在运行的任何程序，以便安装程序可以根据需要来更新文件。
3. 调用安装程序。可以自动或者人工调用安装程序。如果安装程序未能自动启动，或者如果您想以其他语言来运行安装程序，则人工调用安装程序。

- 要自动调用安装程序，将 DB2 Relational Connect CD 插入驱动器。自动运行功能部件将自动启动安装程序。确定了系统语言，并启动了该语言的安装程序。

- 要人工调用安装程序：

- a. 单击 **开始**，并选择 **运行** 选项。
- b. 在打开字段中，输入以下命令：

```
x:\setup /i language
```

其中：

x: 表示 CD-ROM 驱动器。

language

表示您使用的语言的国家代码（例如 EN 表示英语）。

- c. 单击 **确定**。

安装启动板打开。

4. 单击 **安装** 以开始安装过程。

5. 遵循安装程序中的提示。

当程序完成时，DB2 Relational Connect 将与其他 DB2 产品一起安装在您的安装目录中。

在 AIX、Linux 和 Solaris 操作环境服务器上安装 DB2 Relational Connect

在 AIX、Linux 和 Solaris 操作环境联合服务器上安装 DB2 Relational Connect 之前：

- 确保已在联合服务器上安装“DB2 通用数据库企业版”或“DB2 通用数据库扩充企业版”。如果打算在分布式请求中包括 DB2 系列数据库，则当您安装“DB2 通用数据库”时必须已经选择了 **DB2 的分布式连接数据源**选项。要验证是否实现了此选项，检查 **FEDERATED** 参数是否被设置为 **YES**。可通过发出 **GET DATABASE MANAGER CONFIGURATION** 命令来检查此设置，该命令将显示所有参数及其当前设置。
- 确保已经在联合服务器上为数据源（例如 Sybase Open Client）安装了客户机软件。

要在 AIX、Linux 和 Solaris 操作环境服务器上安装 DB2 Relational Connect，使用 db2setup 实用程序：

1. 作为具有 root 用户权限的用户登录。
2. 插入并安装 DB2 产品 CD-ROM。有关如何安装 CD-ROM 的信息，参见 *AIX 快速入门*。
3. 通过输入 **cd lcdrom** 命令来切换至 CD-ROM 所安装在的目录，其中 *cdrom* 是产品 CD-ROM 的安装点。
4. 输入 **./db2setup** 命令。几分钟后，“DB2 安装实用程序”窗口将打开。
5. 选择**安装**。“安装 DB2 V7”窗口打开。
6. 导航至客户机的 DB2 Relational Connect 产品，例如，Relational Connect Sybase 版，并按空格键来选择它。当选择了某个选项时，该选项旁将出现一个星号。
7. 选择**确定**。“创建 DB2 服务”窗口打开。
8. 可以选择创建 DB2 实例。选择**确定**。“总结报告安装”窗口打开。系统自动安装了两个项目：Oracle 的分布式连接和 DB2 Relational Connect 的“产品签名”。必须要有“产品签名”才能与 Sybase 数据源连接。
9. 选择**继续**。出现一个窗口，指示您的最终选择是停止 Relational Connect 安装。选择**确定**以便继续安装。可能要花几分钟才能完成安装。
10. 当出现一个通知，指示已成功地完成安装时，选择**确定**。“总结报告”窗口打开，该窗口指示每次安装的选项是成功还是失败；再次选择**确定**。

当完成安装时，DB2 Relational Connect 将与其他 DB2 产品一起安装在目录中。

在 AIX 上，此目录为 `/usr/lpp/db2_07_01` 目录。在 Solaris 上，此目录为 `/opt/IBMdb2/V7.1` 目录。在 Linux 上，此目录为 `/usr/IBMdb2/V7.1` 目录。

第 26 章 访问 Oracle 数据源

除了支持 AIX 和 Windows NT 上的封装器之外，“DB2 通用数据库”现在还支持 Linux 和 Solaris 操作环境上的 Oracle 封装器。此支持仅限于 Oracle 版本 8。要访问这些平台的封装器，需要插入 V7.2 DB2 Relational Connect CD，并选择“DB2 Relational Connect Oracle 版”数据源。

一旦安装了 DB2 Relational Connect，就可以将 Oracle 数据源添加至联合服务器中：

1. 在 DB2 联合服务器上安装和配置 Oracle 客户机软件。
2. 通过修改 db2dj.ini 文件并发出 **db2set** 命令来设置数据源环境变量。
3. 对于在 UNIX 平台上运行的 DB2 联合服务器，运行 djxlink 脚本以将 Oracle SQL*Net 或 Net8 库链接编辑至 DB2 联合服务器。
4. 确保更新了 SQL*Net 或 Net8 tnsnames.ora 文件。
5. 回收 DB2 实例。
6. 创建封装器。
7. 可选：设置 DB2_DJ_COMM 环境变量。
8. 创建服务器。
9. 创建用户映射。
10. 为表和视图创建别名。

第 26 章中提供了有关这些步骤（包括设置环境变量）的详细指导。“设置联合系统以访问 Oracle 数据源”在 *DB2 安装与配置补遗* 中。

文档错误

『将 Oracle 数据源添加至联合系统』一节存在下列错误：

- 过程中缺少了某个步骤。正确步骤是：
 1. 使用由 Oracle 提供的文档来在 DB2 联合服务器上安装和配置 Oracle 客户机软件。
 2. 通过修改 db2dj.ini 文件并发出 **db2set** 命令来设置数据源环境变量。**db2set** 命令使用您的设置来更新 DB2 简要表注册表。

在第 26 章中提供了有关设置环境变量的详细指导。设置联合系统以访问 *DB2 安装与配置补遗* 的 Oracle 数据源。
 3. 对于在 UNIX 平台上运行的 DB2 联合服务器，运行 djxlink 脚本以将 Oracle SQL*Net 或 Net8 库链接编辑至 DB2 联合服务器。根据使用的平台，djxlink 脚本位于：
 - 在 AIX 上，位于 /usr/lpp/db2_07_01/bin
 - 在 Solaris 上，位于 /opt/IBMdb2/V7.1/bin
 - 在 Linux 上，位于 /usr/IBMdb2/V7.1/bin仅当在 DB2 联合服务器上安装了 Oracle 的客户机软件之后，才应运行 djxlink 脚本。
- 文档指示要设置：

```
DB2_DJ_INI = sqllib/cfg/db2dj.ini
```

这是不正确的，它应该设置为以下内容：

```
DB2_DJ_INI = $INSTHOME/sqllib/cfg/db2dj.ini
```

访问 Sybase 数据源（新章节）

在将 Sybase 数据源添加至联合服务器之前，需要在 DB2 联合服务器上安装和配置 Sybase Open Client 软件。有关如何安装 Open Client 软件的特定详细信息，参见随 Sybase 数据库软件提供的文档中的安装过程。作为安装的一部分，确保包括了目录存储过程和 Sybase Open Client 库。

要设置联合服务器以访问存储在 Sybase 数据源上的数据，您需要：

1. 安装 DB2 Relational Connect 版本 7.2。参见第34页的『安装 DB2 Relational Connect』。
2. 将 Sybase 数据源添加至联合服务器。
3. 指定 Sybase 代码页。

本章讨论步骤 2 和步骤 3。

本章中的指导适用于 Windows NT、AIX 和 Solaris 操作环境。特定于平台的区别在出现这些平台时作了注释。

将 Sybase 数据源添加至联合服务器

要将 Sybase 数据源添加至联合服务器，需要：

1. 设置环境变量并更新简要表注册表。
2. 将 DB2 链接至 Sybase 客户机软件（仅 AIX 和 Solaris）
3. 回收 DB2 实例。
4. 创建并设置接口文件。
5. 创建封装器。
6. 可选：设置 DB2_DJ_COMM 环境变量。
7. 创建服务器。
8. 可选：设置 CONNECTSTRING 服务器选项
9. 创建用户映射。
10. 为表和视图创建别名。

本节中详细说明了这些步骤。

步骤 1：设置环境变量并更新简要表注册表

通过修改 db2dj.ini 文件并发出 **db2set** 命令来设置数据源环境变量。db2dj.ini 文件包含关于安装在联合服务器上的 Sybase 客户机软件的配置信息。**db2set** 命令使用您的设置来更新 DB2 简要表注册表。

在分区数据库系统中，可对特定实例中的所有节点使用单个 db2dj.ini 文件，也可以对特定实例中的一个或多个节点使用唯一的 db2dj.ini 文件。对于每个实例，非分区数据库系统只能有一个 db2dj.ini 文件。

要设置环境变量：

1. 编辑位于 sqllib/cfg 中的 db2dj.ini 文件，并设置以下环境变量：

```
SYBASE=<sybase home directory>
```

其中 <sybase home directory> 是安装 Sybase 客户机的目录。

2. 利用 Sybase 环境变量来更新 DB2 实例的 .profile 文件。可以通过发出以下命令来完成此操作:

```
export PATH="$SYBASE/bin:$PATH"
export SYBASE="<sybase home directory>"
```

其中 <sybase home directory> 是安装 Sybase 客户机的目录。

3. 通过输入以下命令来执行 DB2 实例 .profile:

```
..profile
```

4. 发出 **db2set** 命令来使用您的更改来更新 DB2 简要表注册表。**db2set** 命令的语法取决于数据库系统的结构。仅当在下列任何数据库系统结构中使用 db2dj.ini 文件时, 才需要执行此步骤。

如果在非分区数据库系统中使用 db2dj.ini 文件, 或者您想只对当前节点应用 db2dj.ini 文件, 则发出:

```
db2set DB2_DJ_INI = sqllib/cfg/db2dj.ini
```

如果在分区数据库系统中使用 db2dj.ini 文件, 并且想要将 db2dj.ini 文件中的值应用于此实例内的所有节点, 则发出:

```
db2set -g DB2_DJ_INI = sqllib/cfg/db2dj.ini
```

如果在分区数据库系统中使用 db2dj.ini 文件, 并且想要将 db2dj.ini 文件中的值应用于特定节点, 则发出:

```
db2set -i INSTANCEX 3 DB2_DJ_INI = sqllib/cfg/node3.ini
```

其中:

INSTANCEX

是实例的名称。

3 是 db2nodes.cfg 文件中列示的节点号。

node3.ini

是 db2dj.ini 文件的已修改并且重命名的版本。

步骤 2: 将 DB2 链接至 Sybase 客户机软件 (仅 AIX 和 Solaris)

要允许访问 Sybase 数据源, DB2 联合服务器必须编辑为链接至客户机库。链接 - 编辑过程将为每个数据源创建一个封装器, 联合服务器将与封装器进行通信。当运行 djsxlink 脚本时, 就创建了封装器库。要发出 djsxlink 脚本, 输入:

```
djsxlink
```

步骤 3: 回收 DB2 实例

为确保在程序中设置了环境变量, 应回收 DB2 实例。回收实例时, 刷新 DB2 实例以接受您所作的更改。

发出下列命令来回收 DB2 实例:

在“DB2 Windows NT 版”服务器上:

```
NET STOP instance_name
NET START instance_name
```

在 DB2 AIX 版和 Solaris 版服务器上:

```
db2stop
db2start
```

步骤 4: 创建并设置接口文件

要创建并设置接口文件，必须创建该文件并使其可供访问。

1. 使用由 Sybase 提供的实用程序来创建包含想要访问的所有 Sybase Open Server 的数据的接口文件。有关使用此实用程序的详情，参见来自 Sybase 公司的安装文档。

Windows NT 通常将此文件称为 `sql.ini`。将您刚创建的文件从 `sql.ini` 重命名为 `interfaces` 以便为该文件在所有平台上命名。如果您选择不将 `sql.ini` 重命名为 `interfaces`，则必须使用 `IFILE` 参数或步骤 8 中说明的 `CONNECTSTRING` 选项。

在 AIX 和 Solaris 系统上，此文件被命名为 `<instance home>/sqllib/interfaces`。

2. 使 DB2 可以访问接口文件。

在“DB2 Windows NT 版”服务器上:

将该文件放置在 DB2 实例的 `%DB2PATH%` 目录中。

在 DB2 AIX 版和 Solaris 版服务器上:

将该文件放置在 DB2 实例的 `$HOME/sqllib` 目录中。使用 `ln` 命令来从 DB2 实例的 `$HOME/sqllib` 目录链接至该文件。例如:

```
ln -s -f /home/sybase/interfaces /home/db2djin1/sqllib
```

步骤 5: 创建封装器

使用 `CREATE WRAPPER` 语句来指定将用来访问 Sybase 数据源的封装器。封装器是联合系统用来与数据源进行通信以及从数据源中检索数据的机制。DB2 中包括用于 Sybase、`CTLIB` 和 `DBLIB` 的两个封装器。以下示例显示了 `CREATE WRAPPER` 语句:

```
CREATE WRAPPER CTLIB
```

其中，`CTLIB` 是与 Sybase Open Client 软件配合使用的缺省封装器名。`CTLIB` 封装器可以在 Windows NT、AIX 和 Solaris 服务器上使用。

可以用您选择的名称来替代缺省封装器名。但是，如果这样做的话，则还必须在 `CREATE WRAPPER` 语句中包括 `LIBRARY` 参数以及联合服务器的封装器库的名称。有关封装器库名的详情，参见 *DB2 SQL Reference* 中的 `CREATE WRAPPER` 语句。

步骤 6: 可选: 设置 DB2_DJ_COMM 环境变量

当访问 Sybase 数据源时，要提高性能，应设置 `DB2_DJ_COMM` 环境变量。此变量确定初始化联合服务器时是否装入封装器。将 `DB2_DJ_COMM` 环境变量设置为包括与您在先前步骤中指定的封装器相对应的封装器库；例如:

在“DB2 Windows NT 版”服务器上:

```
db2set DB2_DJ_COMM='ctlib.dll'
```

在“DB2 AIX 版”服务器上:

```
db2set DB2_DJ_COMM='libctlib.a'
```

在“DB2 Solaris 版”服务器上:

```
db2set DB2_DJ_COMM='libctlib.so'
```

必须确保等号 (=) 两边都没有空格。

有关封装器库名的详情，参考 *DB2 SQL Reference*。有关 `DB2_DJ_COMM` 环境变量的信息，参考管理指南。

步骤 7: 创建服务器

使用 CREATE SERVER 语句来定义您想访问其数据源的每个 Sybase 服务器; 例如:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
OPTIONS (NODE 'sybnode', DBNAME 'sybdb')
```

其中:

SYBSERVER

是为 Sybase 服务器指定的名称。此名称必须是唯一的。

SYBASE

是正在配置要访问的数据源的类型。Sybase 是唯一受支持的数据源。

12.0 是您正在访问的 Sybase 的版本。受支持的版本为 10.0、11.0、11.1、11.5、11.9 和 12.0。

CTLIB 是您在 CREATE WRAPPER 语句中指定的封装器名。

'sybnode'

是 SYBSERVER 所驻留在的节点的名称。从接口文件中获取节点值。此值是区分大小写的。

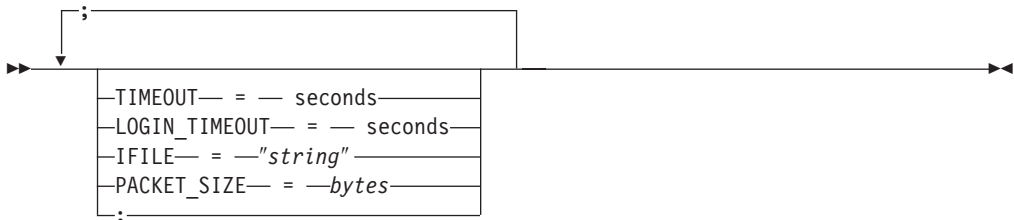
尽管节点名被指定为一个选项, 但是它是 Sybase 数据源必需的。有关其他选项的信息, 参见 *DB2 SQL Reference*。

'sybdb' 是想要访问的 Sybase 数据库的名称。

可选: 步骤 8: 设置 CONNECTSTRING 服务器选项

指定超时阈值、接口文件的路径和名称以及接口文件的信息包大小。Sybase Open Client 使用超时阈值来中断运行了太长时间的查询和响应。可以通过使用 CREATE SERVER OPTION DDL 语句的 CONNECTSTRING 选项来在 DB2 中设置这些阈值。使用 CONNECTSTRING 选项来指定:

- SQL 查询的超时持续时间。
- 登录响应的超时持续时间。
- 接口文件的路径和名称。
- 信息包大小。



TIMEOUT

指定“DB2 通用数据库”为任何 SQL 语句而等待来自 Sybase Open Client 的响应的秒数。seconds 的值是“DB2 通用数据库”的整数范围内的一个正整数。您指定的超时值取决于您正在使用的封装器。Windows NT、AIX 和 Solaris 服务器都能够利用 DBLIB 封装器。DBLIB 封装器的缺省值为 0。在 Windows NT、AIX 和 Solaris 服务器上, DBLIB 的缺省值将导致“DB2 通用数据库”无限期地等待响应。

LOGIN_TIMEOUT

指定“DB2 通用数据库”等待来自 Sybase Open Client 的对登录请求的响应的秒数。缺省值与 TIMEOUT 的缺省值相同。

IFILE

指定 Sybase Open Client 接口文件的路径和名称。必须用双引号 (") 将 *string* 中所标识的路径括起来。在 Windows NT 服务器上，缺省值为 %DB2PATH%。在 AIX 和 Solaris 服务器上，缺省值为“DB2 通用数据库”实例的主目录中的 sqllib/interfaces。

PACKET_SIZE

指定接口文件的信息包大小（以字节计）。如果数据源不支持所指定的信息包大小，则连接将失败。当每个记录都非常大时（例如，当在大型表中插入行时），通过增大信息包大小可以明显地提高性能。字节大小为数值。有关详情，参见 Sybase 参考手册。

示例:

在 Windows NT 服务器上，要将超时值设置为 60 秒，并将接口文件设置为 C:\etc\interfaces，使用:

```
CREATE SERVER OPTION connectstring FOR SERVER sybase1
SETTING 'TIMEOUT=60;LOGIN_TIMEOUT=5;IFILE="C:\etc\interfaces";'
```

在 AIX 和 Solaris 服务器上，要将超时值设置为 60 秒，并将接口文件设置为 /etc/interfaces，使用:

```
CREATE SERVER OPTION connectstring FOR SERVER sybase1
SETTING 'TIMEOUT=60;PACKET_SIZE=4096;IFILE="/etc/interfaces";'
```

步骤 9: 创建用户映射

如果联合服务器上的用户标识和口令与 Sybase 数据源上的用户标识或口令不同，则使用 CREATE USER MAPPING 语句来将本地用户标识映射为在 Sybase 数据源中定义的用户标识和口令；例如:

```
CREATE USER MAPPING FOR DB2USER SERVER SYBSERVER
OPTIONS ( REMOTE_AUTHID 'sybuser', REMOTE_PASSWORD 'dayllte')
```

其中:

DB2USER

是您正在映射至 Sybase 数据源中定义的用户标识的本地用户标识。

SYBSERVER

是在 CREATE SERVER 语句中定义的 Sybase 数据源的名称。

'sybuser'

是您正在将 DB2USER 映射至的 Sybase 数据源的用户标识。此值是区分大小写的。

'dayllte'

是与 'sybuser' 相关联的口令。此值是区分大小写的。

有关其他选项的详情，参见 *DB2 SQL Reference*。

步骤 10: 为表和视图创建别名

对位于 Sybase 数据源中的每个视图或表指定别名。当您查询 Sybase 数据源时将使用这些别名。Sybase 别名是区分大小写的。将模式名和表名都用双引号 (") 括起来。以下示例显示了 CREATE NICKNAME 语句:

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europe"
```

其中:

SYBSALES

是 Sybase 表或视图的唯一别名。

SYBSERVER."salesdata"."europe"

三部分组成的标识符, 遵循以下格式:

data_source_name."remote_schema_name"."remote_table_name"

对您想为其创建别名的每个表或视图重复此步骤。创建别名时, DB2 将使用连接来查询数据源目录。此查询将测试您与数据源的连接。如果连接不工作, 则将接收到错误消息。

有关 CREATE NICKNAME 语句的详情, 参见 *DB2 SQL Reference*。有关一般别名和验证数据类型映射的详情, 参见 *DB2 管理指南*。

指定 Sybase 代码页

仅当 DB2 联合服务器和 Sybase 服务器正在运行不同的代码页时, 才需要执行此步骤。使用与 DB2 使用的代码集相同的代码集的数据源不需要转换。下表提供了常见的“国家语言支持”(NLS) 代码页的等效 Sybase 选项。必须将 Sybase 数据源配置为对应于这些等效选项, 或者客户机代码必须能够检测不匹配的项, 并将它标志为错误, 或者使用它自己的语义来映射数据。如果找不到从源代码页至目标代码页的转换表, 则 DB2 会发出错误消息。有关详情, 参考 Sybase 文档。

表 1. Sybase 代码页选项

代码页	等效 Sybase 选项
850	cp850
897	sjis
819	iso_1
912	iso_2
1089	iso_6
813	iso_7
916	iso_8
920	iso_9

使用 ODBC 来访问 Microsoft SQL Server 数据源 (新章节)

在将 Microsoft SQL Server 数据源添加至 DB2 联合服务器之前, 需要在联合服务器上安装并配置 ODBC 驱动程序。有关如何安装 ODBC 驱动程序的特定详细信息, 参见随 ODBC 驱动程序提供的文档中的安装过程。

要设置联合服务器以访问存储在 Microsoft SQL Server 数据源上的数据, 需要:

1. 在联合服务器上安装并配置 ODBC 驱动程序。有关如何安装 ODBC 驱动程序的特定详细信息，参见随 ODBC 驱动程序提供的文档中的安装过程。

在“DB2 Windows NT 版”服务器上:

使用 ODBC 设备管理器来配置系统 DSN。

在“DB2 AIX 版”服务器上:

安装由 MERANT 提供的库的线程版本，将 MERANT 库目录指定为 LIBPATH 中的第一个条目，并设置 .odbc.ini 文件。在主目录中创建 .odbc.ini 文件。

2. 安装 DB2 Relational Connect 版本 7.2。参见第34页的『安装 DB2 Relational Connect』。
3. 将 Microsoft SQL Server 数据源添加至联合服务器。
4. 指定 Microsoft SQL Server 代码页。

本章讨论步骤 3 和步骤 4。

本章中的指导适用于 Windows NT 和 AIX 平台。特定于平台的区别在出现这些平台时作了注释。

将 Microsoft SQL Server 数据源添加至联合服务器

在安装 ODBC 驱动程序和 DB2 Relational Connect 之后，使用下列步骤将 Microsoft SQL Server 数据源添加至联合服务器:

1. 设置环境变量（仅 AIX）。
2. 运行 shell 脚本（仅 AIX）。
3. 可选: 设置 DB2_DJ_COMM 环境变量。
4. 回收 DB2 实例（仅 AIX）。
5. 创建封装器。
6. 创建服务器。
7. 创建用户映射。
8. 为表和视图创建别名。
9. 可选: 获取 ODBC 跟踪。

下列各节中详细说明了这些步骤。

步骤 1: 设置环境变量（仅 AIX）

通过修改 db2dj.ini 文件并发出 **db2set** 命令来设置数据源环境变量。db2dj.ini 文件包含用来连接至 Microsoft SQL Server 数据源的配置信息。**db2set** 命令使用您的设置来更新 DB2 简要表注册表。

在分区数据库系统中，可对特定实例中的所有节点使用单个 db2dj.ini 文件，也可以对特定实例中的一个或多个节点使用唯一的 db2dj.ini 文件。对于每个实例，非分区数据库系统只能有一个 db2dj.ini 文件。

要设置环境变量:

1. 编辑位于 \$HOME/sql/lib/cfg/ 中的 db2dj.ini 文件，并设置下列环境变量:

```
ODBCINI=$HOME/.odbc.ini
DJX_ODBC_LIBRARY_PATH=<path to the Merant driver>/lib
LIBPATH=<path to the Merant driver>/lib
DB2ENVLIST=LIBPATH
```

发出 **db2set** 命令来使用您的更改来更新 DB2 简要表注册表。 **db2set** 的语法取决于数据库系统结构:

- 如果在非分区数据库系统中使用 db2dj.ini 文件, 或者如果在分区数据库系统中使用 db2dj.ini 文件, 并且想只对当前节点应用 db2dj.ini 文件中的值, 则发出以下命令:

```
db2set DB2_DJ_INI=<path to ini file>/db2dj.ini
```

- 如果在分区数据库系统中使用 db2dj.ini 文件, 并且想要将 db2dj.ini 文件中的值应用于此实例内的所有节点, 则发出以下命令:

```
db2set -g DB2_DJ_INI=<path to ini file>/db2dj.ini
```

- 如果在分区数据库系统中使用 db2dj.ini 文件, 并且想要将 db2dj.ini 文件中的值应用于特定节点, 则发出以下命令:

```
db2set -i INSTANCEX 3 DB2_DJ_INI=$HOME/sqllib/cfg/node3.ini
```

其中:

INSTANCEX

是实例的名称。

3 是 db2nodes.cfg 文件中列示的节点号。

node3.ini

是 db2dj.ini 文件的已修改并且重命名的版本。

步骤 2: 运行 shell 脚本 (仅 AIX)

djxlink.sh shell 脚本将客户机库链接至封装器库。要运行 shell 脚本:

```
djxlink
```

步骤 3: 可选: 设置 DB2_DJ_COMM 环境变量

如果您发现花了过长的时间来访问 Microsoft SQL Server 数据源, 则可以将 DB2_DJ_COMM 环境变量设置为在初始化联合系统时装入封装器, 而不是在尝试访问数据源时才装入封装器, 以提高性能。将 DB2_DJ_COMM 环境变量设置为包括与您步骤 5 中指定的封装器相对应的封装器库。例如:

在“DB2 Windows NT 版”服务器上:

```
db2set DB2_DJ_COMM=djxmssql13.dll
```

在“DB2 AIX 版”服务器上:

```
db2set DB2_DJ_COMM=libmssql13.a
```

必须确保等号 (=) 两边都没有空格。

有关封装器库名的详情, 参见 *DB2 SQL Reference*。

步骤 4: 回收 DB2 实例 (仅 AIX)

为确保在程序中设置了环境变量, 应回收 DB2 实例。回收实例时, 刷新 DB2 实例以接受您所作的更改。通过发出下列命令来回收 DB2 实例:

```
db2stop
db2start
```

步骤 5: 创建封装器

“DB2 通用数据库”具有两种不同的协议（称为封装器），可以用来访问 Microsoft SQL Server 数据源。封装器是联合服务器用来与数据源进行通信以及从数据源中检索数据的机制。您使用的封装器取决于运行“DB2 通用数据库”的平台。使用表2来作为选择适当的封装器的指南。

表 2. ODBC 驱动程序

ODBC 驱动程序	平台	封装器名
ODBC 3.0（或更高级别）驱动程序	Windows NT	DJXMSSQL3
MERANT DataDirect Connect ODBC 3.6 驱动程序	AIX	MSSQLODBC3

使用 CREATE WRAPPER 语句来指定将用来访问 Microsoft SQL Server 数据源的封装器。以下示例显示了 CREATE WRAPPER 语句:

```
CREATE WRAPPER DJXMSSQL3
```

其中, DJXMSSQL3 是在“DB2 Windows NT 版”服务器（使用 ODBC 3.0 驱动程序）上使用的缺省封装器名称。如果您具有“DB2 AIX 版”服务器, 则将指定 MSSQLODBC3 封装器名。

可以用您选择的名称来替代缺省封装器名。但是, 如果这样做的话, 则必须在 CREATE WRAPPER 语句中包括 LIBRARY 参数以及联合服务器平台的封装器库的名称。例如:

在“DB2 Windows NT 版”服务器上:

```
CREATE WRAPPER wrapper_name LIBRARY 'djxmssql3.dll'
```

其中, *wrapper_name* 是您想给予封装器的名称, 而 'djxmssql3.dll' 是库名。

在“DB2 AIX 版”服务器上:

```
CREATE WRAPPER wrapper_name LIBRARY 'libmssql3.a'
```

其中, *wrapper_name* 是您想给予封装器的名称, 而 'libdjxmssql.a' 是库名。

有关封装器库名的详情, 参见 *DB2 SQL Reference* 中的 CREATE WRAPPER 语句。

步骤 6: 创建服务器

使用 CREATE SERVER 语句来定义您想连接的每个 Microsoft SQL Server 数据源。例如:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER djxmssql3  
OPTIONS (NODE 'sqlnode', DBNAME 'database_name')
```

其中:

sqlserver

是为 Microsoft SQL Server 服务器指定的名称。此名称必须是唯一的。

MSSQLSERVER

是正在配置要访问的数据源的类型。

7.0 是您正在访问的 Microsoft SQL Server 的版本。“DB2 通用数据库”支持 Microsoft SQL Server 的版本 6.5 和 7.0。

DJXMSSQL3

是您正在 `CREATE WRAPPER` 语句中定义的封装器名。

'sqlnode'

是引用 Microsoft SQL Server 的系统 DSN 名。此值是区分大小写的。是您正在访问的 Microsoft SQL Server 的版本。“DB2 通用数据库”支持 Microsoft SQL Server 的版本 6.5 和 7.0。

尽管节点名（系统 DNS 名）被指定为 `CREATE SERVER` 语句中的一个选项，但是它是 Microsoft SQL Server 数据源所必需的。

有关可配合 `CREATE WRAPPER` 语句使用的其他选项，参见 *DB2 SQL Reference*。

'database_name'

是您正在连接的数据库的名称。

尽管数据库的名称被指定为 `CREATE SERVER` 语句中的一个选项，但是它是 Microsoft SQL Server 数据源所必需的。

步骤 7: 创建用户映射

如果联合服务器上的用户标识和口令与 Microsoft SQL Server 数据源上的用户标识或口令不同，则使用 `CREATE USER MAPPING` 语句来将本地用户标识映射为在 Microsoft SQL Server 数据源中定义的用户标识和口令；例如：

```
CREATE USER MAPPING FOR db2user SERVER server_name
OPTIONS (REMOTE_AUTHID 'mssqluser', REMOTE_PASSWORD 'dayllite')
```

其中：

db2user

是您正在映射至在 Microsoft SQL Server 数据源中定义的用户标识的本地用户标识。

server_name

是在 `CREATE SERVER` 语句中定义的服务器的名称。

'mssqluser'

是您正在将 *db2user* 映射至的 Microsoft SQL Server 数据源的用户标识。此值是区分大小写的。

'dayllite'

是与 *'mssqluser'* 相关联的口令。此值是区分大小写的。

有关可配合 `CREATE USER MAPPING` 语句使用的其他选项，参见 *DB2 SQL Reference*。

步骤 8: 为表和视图创建别名

对于您想访问的 Microsoft SQL Server 数据源中的每个视图或表指定别名。当您查询 Microsoft SQL Server 数据源时将使用这些别名。使用 `CREATE NICKNAME` 语句来指定别名。别名是区分大小写的。以下示例显示了 `CREATE NICKNAME` 语句：

```
CREATE NICKNAME mssqlsales FOR server_name.salesdata.europe
```

其中：

mssqlsales

是 Microsoft SQL Server 表或视图的唯一别名。

`server_name.salesdata.europe`

三部分组成的标识符，遵循以下格式：

`data_source_server_name.remote_schema_name.remote_table_name`

建议对别名的 `remote_schema_name` 和 `remote_table_name` 部分用双引号引起来。

当您创建别名时，DB2 试图访问数据源目录表（在 Microsoft SQL Server 中将它们称为系统表）。这会测试与数据源的连接。如果连接失败，则将接收到错误消息。

对您想为其创建别名的所有数据库表和视图重复此步骤。

有关 CREATE NICKNAME 语句的详情，参见 *DB2 SQL Reference*。有关一般别名和验证数据类型映射的详情，参见 *DB2 管理指南*。

步骤 9: 可选: 获取 ODBC 跟踪

如果在访问数据源时遇到了问题，则可以获取 ODBC 跟踪信息以分析和解决这些问题。为了确保 ODBC 跟踪正常工作，应使用由“ODBC 数据源管理员”提供的跟踪工具。激活跟踪将影响系统的性能，因此，一旦解决了问题，就应该关闭跟踪。

复查 Microsoft SQL Server 代码页

Microsoft SQL Server 支持 DB2 UDB 所支持的许多常见“国家语言支持” (NLS) 代码页。使用与 DB2 使用的代码集相同的代码集的数据源不需要转换。表 3 中列示了“DB2 通用数据库”和 Microsoft SQL Server 都支持的代码页。

表 3. DB2 UDB 和 Microsoft SQL Server 代码页选项

代码页	受支持的语言
1252	ISO 字符集
850	多种语言
437	美国英语
874	泰国语
932	日语
936	简体中文
949	韩语
950	繁体中文
1250	中欧语
1251	斯拉夫语
1253	希腊语
1254	土耳其语
1255	希伯来语
1256	阿拉伯语

当 DB2 联合服务器和 Microsoft SQL Server 正在运行不同的“国家语言支持” (NLS) 代码页时，必须将 Microsoft SQL Server 数据源配置为对应于这些等效选项，或者客户机代码必须能够检测不匹配项，并将它标志为错误，或者使用它自己的语义来映射数据。如果找不到从源代码页至目标代码页的转换表，则 DB2 会发出错误消息。有关详情，参考 Microsoft SQL Server 文档。

第3部分 管理

管理指南: 计划	53	备份至命名管道	71
第 8 章 物理数据库设计	53	从分割映像备份	71
分区键	53	按需日志归档	73
设计节点组	53	日志镜像	74
第 9 章 设计分布式数据库	53	Sun Solaris 和 HP 上的跨平台备份和恢复支持	74
更新多个数据库	53	DB2 Data Links Manager 注意事项 / 备份实用程序注意事项	74
第 13 章 Windows NT 环境中的高可用性	53	DB2 Data Links Manager 注意事项 / 恢复和前滚实用程序注意事项	75
在运行 DB2MCS 实用程序之前需要重新引导机器	53	从脱机备份恢复数据库而不进行前滚	76
第 14 章 Sun Cluster 2.2 上的 DB2 和高可用性	54	恢复数据库和表空间, 并前滚至日志结束	76
Solaris 上的 Veritas 支持	54	DB2 Data Links Manager 和恢复交互作用	77
附录 B. 命名规则	55	需要协调的情况的检测	79
有关关于 8 个字符的用户标识和模式名的注意事项	55	附录 C 数据库恢复的用户出口	80
用户标识和口令	55	附录 D 对多个数据库分区服务器发出命令	80
附录 D. 发行版之间的不兼容性	56	附录 I 高速节点间通信	81
Windows NT DLFS 与 Norton 的实用程序不兼容	56	使 DB2 能够运行 (使用 VI)	81
SET CONSTRAINTS 被替换为 SET INTEGRITY	56	管理指南: 性能	83
附录 E. 国家语言支持	56	第 3 章 应用程序注意事项	83
DB2 版本 7 的国家语言版本	56	指定隔离级别	83
控制中心和文档文件集	58	调整优化类	83
DB2 管理服务器的语言环境设置	59	动态复合语句	84
DB2 UDB 在 Windows 平台上支持 Baltic Rim 代码页 (MS-1257)	59	第 4 章 环境注意事项	84
派生代码页值	59	使用较大的索引键	84
国家代码和代码页支持	60	第 5 章 系统目录统计信息	85
字符集	60	收集和分布统计信息	85
管理指南: 实现	61	用于更新目录统计信息的规则	85
添加或扩充 DMS 容器 (新进程)	61	子元素统计信息	85
第 1 章 使用 GUI 工具管理 DB2	61	第 6 章 了解 SQL 编译器	87
第 3 章 创建数据库	61	复制的总结表	87
创建表空间	61	数据访问概念和优化	88
在 Linux 上使用原始 I/O	61	第 8 章 操作性能	89
创建序列	63	管理数据库缓冲池	89
比较 IDENTITY 列和序列	64	管理多个数据库缓冲池	90
创建索引、索引扩展项或索引规范	64	第 9 章 使用管理器	90
第 4 章 改变数据库	65	第 13 章 配置 DB2	90
将容器添加至分区上的 SMS 表空间	65	排序堆大小 (sortheap)	91
改变标识列	65	排序堆阈值 (sheapthres)	91
改变序列	65	升级前锁定列表的最大百分比 (maxlocks)	91
删除序列	66	配置 DB2/DB2 Data Links Manager / Data Links 访问令牌到期时间间隔 (dl_expint)	92
切换表空间的状态	66	MIN_DEC_DIV_3 数据库配置参数	92
在 DMS 表空间中修改容器	66	应用程序控制堆大小 (app_ctl_heap_sz)	93
第 5 章 控制数据库访问	66	数据库系统监控器堆大小 (mon_heap_sz)	93
序列特权	67	最大活动应用程序数 (maxappls)	93
数据加密	67	恢复范围和软检查点时间间隔 (softmax)	93
第 8 章 恢复数据库	68	跟踪已修改页启用 (trackmod)	93
如何使用暂挂 I/O	68	更改数据库日志路径 (newlogpath)	94
增量备份和恢复	69	日志文件的位置 (logpath)	95
从增量备份映像恢复	70	锁定列表的最大存储器 (locklist)	95
并行恢复	71		

附录 A DB2 注册表和环境变量	95	CLP 返回码中的文档错误	137
新的或更改过的注册表变量表	95	Data Movement Utilities Guide and Reference	139
附录 C SQL 说明工具	99	Chapter 2. Import	139
卫星管理指南和参考	101	Using Import with Buffered Inserts	139
将版本 7.2 DB2 个人版和 DB2 工作组版设置为卫星	101	Chapter 3. Load	139
先决条件	101	Pending States After a Load Operation	139
安装说明	102	Load Restrictions and Limitations	139
配置版本 7.2 系统以进行同步	103	totalreespace File Type Modifier	139
在版本 6 企业版系统上安装修订包 2 或更高版本	104	Chapter 4. AutoLoader	139
升级版本 6 DB2 企业版以用作 DB2 控制服务器	104	rexecd Required to Run Autoloader When Authentication Set to YES	139
升级版本 6 控制中心和卫星管理中心	110	Replication Guide and Reference	141
Command Reference	111	复制和非 IBM 服务器	141
db2batch - Benchmark Tool	111	在 Windows 2000 上进行复制	141
db2cap (新命令)	111	保存 SQL 文件时的已知错误	141
db2cap - CLI/ODBC 静态软件包绑定工具	112	DB2 维护	141
db2ckrst (新命令)	113	Web 上的数据差异实用程序	141
db2ckrst - 检查增量恢复映像序列	114	Chapter 3. Data replication scenario	141
db2gncol (新命令)	115	复制方案	141
db2gncol - 更新生成的列值	116	Chapter 5. Planning for replication	142
db2inidb - 初始化镜像数据库	117	表名和列名	142
db2look - DB2 统计信息抽取工具	117	DATALINK 复制	142
db2updv7 - 将数据库更新为版本 7 当前修正级别	118	LOB 限制	143
新命令行处理器选项 (-x, 不打印列标题)	119	计划复制	143
DB2 CLP 的 True Type 字体需求	119	Chapter 6. Setting up your replication environment	143
ADD DATALINKS MANAGER	119	随处更新先决条件	143
ARCHIVE LOG (新命令)	119	设置复制环境	143
归档日志	120	Chapter 8. Problem Determination	144
BACKUP DATABASE	121	Chapter 9. Capture and Apply for AS/400	145
语法图	121	Chapter 10. Capture and Apply for OS/390	145
DB2 Data Links Manager 注意事项	122	Prerequisites for DB2 DataPropagator for OS/390	145
BIND	122	UNICODE and ASCII Encoding Schemes on OS/390	146
CALL	123	Choosing an Encoding Scheme	146
DROP DATALINKS MANAGER (新命令)	123	Setting Encoding Schemes	146
DROP DATALINKS MANAGER	124	Chapter 11. Capture and Apply for UNIX platforms	147
EXPORT	127	在 UNIX 和 Windows 上为 Capture 和 Apply 设置环境变量	147
GET DATABASE CONFIGURATION	127	Chapter 14. Table Structures	147
GET ROUTINE (新命令)	128	Chapter 15. Capture and Apply Messages	148
GET ROUTINE	129	Appendix A. Starting the Capture and Apply Programs from Within an Application	148
GET SNAPSHOT	129	System Monitor Guide and Reference	149
IMPORT	129	db2ConvMonStream	149
LIST HISTORY	130	Troubleshooting Guide	151
LOAD	131	在用户未登录的情况下在 Windows 95、Windows 98 和 Windows ME 上启动 DB2	151
PING (新命令)	132	Chapter 2. Troubleshooting the DB2 Universal Database Server	154
PING	133	在 64 位的平台上使用“DB2 通用数据库”	155
PUT ROUTINE (新命令)	134	第 5 章 配置	155
PUT ROUTINE	135	LOCKLIST	155
RECONCILE	135		
REORGANIZE TABLE	136		
RESTORE DATABASE	136		
语法	136		
DB2 Data Links Manager Considerations	137		
ROLLFORWARD DATABASE	137		

shmsys:shminfo_shmmax	155
第 6 章 限制	155
XML Extender 管理和编程	157
MQSeries	159
DB2 MQSeries 函数的安装和配置	159
安装 MQSeries	159
安装 MQSeries AMI	159
启用 DB2 MQSeries 函数	160
MQSeries 消息传递样式	160
消息结构	161
MQSeries 函数概述	161
限制	162
错误码	162
使用方案	162
基本消息传递	162
发送消息	163
检索消息	164
应用程序间的连通性	165
请求 / 答复通信	165
发布 / 预订	166
enable_MQFunctions	168
enable_MQFunctions	169
disable_MQFunctions	170
disable_MQFunctions	171

管理指南: 计划

第 8 章 物理数据库设计

分区键

在『设计节点组』一节的『节点组设计注意事项』小节中，“分区键”部分中的以下文本中说明了在定义仅当 `DB2_UPDATE_PART_KEY=ON` 才应删除分区键时应考虑的事项:

注: 如果 `DB2_UPDATE_PART_KEY=OFF`, 则限制仍然适用。

注: 在修订包 3 和更新版本中, 缺省值将为 `ON`。

- 不能对表中的某行更新分区键列值。
- 只能删除或插入分区键列值。

设计节点组

在标题为『设计节点组』的一节中, 题为『节点组设计注意事项』这一小节和题为『复制的总结表』这一小节忽略了第二段的最后一句:

只对用 `REFRESH DEFERRED` 选项定义的总结表才可指定 `REPLICATED` 关键字。

第 9 章 设计分布式数据库

更新多个数据库

在『更新多个数据库』一节中, 安装步骤列表不够精确。步骤 4 现在为如下所示:

对应用程序进行预编译, 以指定类型 2 连接 (即, 在 `PRECOMPILE PROGRAM` 命令上指定 `CONNECT 2`) 和一阶段提交 (即, 在 `PRECOMPILE PROGRAM` 命令上指定 `SYNCPOINT ONEPHASE`), 正如 `Application Development Guide` 中所描述的那样。

应更改为:

对应用程序进行预编译, 以指定类型 2 连接 (即, 在 `PRECOMPILE PROGRAM` 命令上指定 `CONNECT 2`) 和两阶段提交 (即, 在 `PRECOMPILE PROGRAM` 命令上指定 `SYNCPOINT TWOPHASE`), 正如 `Application Development Guide` 中所描述的那样。

第 13 章 Windows NT 环境中的高可用性

在运行 DB2MSCS 实用程序之前需要重新引导机器

DB2MSCS 实用程序用来执行启用 DB2 所需的设置, 以便在 Microsoft Cluster Service 环境中获取“故障恢复”支持。要使 DB2MSCS 实用程序运行成功, Cluster Service 必须能够定位资源 DLL 和 `db2wolf.dll` (它驻留在 `%ProgramFiles%\SQLLIB\bin` 目录中)。

“DB2 UDB 版本 7 安装程序”将 PATH 系统环境变量设置为指向 %ProgramFiles%\SQLLIB\bin 目录。但是，如果是在 Windows 2000 操作系统上运行的话，就不需要在安装后重新引导系统。

如果想要运行 DB2MSCS 实用程序，必须重新引导机器以便对 Cluster Service 更新 PATH 环境变量。

第 14 章 Sun Cluster 2.2 上的 DB2 和高可用性

如果满足下列条件，则 DB2 Connect 在 Sun Cluster 2.2 上是受支持的：

- 主机协议为 TCP/IP（而非 SNA）
- 未使用两阶段提交。如果用户将 SPM 日志配置为置于共享磁盘上（此操作可通过 *spm_log_path* 数据库管理器配置参数完成），且故障恢复机器具有完全相同的 TCP/IP 配置（相同的主机名、IP 地址，等等），则此限制将会放宽。

Solaris 上的 Veritas 支持

DB2 现在支持 Veritas，后者在 Solaris 上为“DB2 高可用性”提供了群集支持。

描述 联机、脱机，以及监控 DB2 UDB 实例。

入口点

联机 使用 db2start 来启动实例。

脱机 使用 db2stop 来停止实例。

监控 确定指定的 DB2 实例是否已启动。使用适当的过程监控和（可选）数据库监控。

清理 除去 DB2 实例资源。

属性	类型	定义
probeDatabase	string	要监控的数据库
instanceOwner	string	实例所有者名称
instanceHome	string	实例所有者的主目录
probeTable	string	probeDatabase 中要监控的表
monitorLevel	integer	1 表示过程监控，2 表示数据库监控
nodeNumber	integer	要启动的实例节点号（取消设置时为 EE）

类型定义

```
type DB2UDB (  
    static int CleanTimeout = 240  
    static int MonitorTimeout = 30  
    static int OfflineTimeout = 240  
    static int OnlineRetryLimit = 2  
    static int OnlineTimeout = 120  
    static int OnlineWaitLimit = 1  
    static int RestartLimit = 3  
    static int ToleranceLimit = 1  
    static str ArgList[] =  
{ probeDatabase, instanceOwner, instanceHome, probeTable, monitorLevel, nodeNumber }  
    NameRule = resource.db2udb  
    str probeDatabase  
    str instanceOwner
```

```
    str instanceHome
    str probeTable
    int monitorLevel
    int nodeNumber
)
```

样本配置

```
DB2UDB db2_resource_n0 (
    probeDatabase = sample
    probeTable = vcstable
    instanceOwner = stevera
    instanceHome = "/export/home/stevera"
    monitorLevel = 2
)
```

安装

- 创建目录 /opt/VRTSvcs/bin/DB2UDB。
- 将联机、脱机、监控、清理、DB2UDBAgent 文件复制到 /opt/VRTSvcs/bin/DB2UDB 中，并确保它们被标记为可执行。
- 将文件 db2udb.type.cf 复制到 /etc/VRTSvcs/conf/config 中。
- 停止群集（例如，hastop -all）。
- 将 include db2udb.type.cf 这一行添加到文件 main.cf 中的 include types.cf 这一行后面
- 用 /opt/VRTSvcs/bin/hacf -verify /etc/VRTSvcs/conf/config 验证群集配置是有效的

现在，您可以开始创建控制 DB2 实例所必需的 DB2 资源了。

附录 B. 命名规则

有关长于 8 个字符的用户标识和模式名的注意事项

- Windows 32 位平台上的 DB2 版本 7 产品支持长度最多为 30 个字符的用户标识。但是，由于 Windows NT 和 Windows 2000 的本机支持，对用户标识的实际限制是 20 个字符。
- 如果用户标识和口令是显式指定的，则 DB2 版本 7 支持非 Windows 32 位客户机使用长度超过 8 个字符的用户标识连接至 Windows NT 和 Windows 2000。这排除了使用“客户机”或 DCE 认证的连接。
- 所有平台上的 DCE 认证仍然有用户标识只能为 8 个字符的限制。
- 在 SQLCA 中，从成功的连接（CONNECT 或 ATTACH）返回的认证标识被截断为 8 个字符。SQLWARN 字段在发生截断时会包含警告。有关详情，参考 *SQL Reference* 中对 CONNECT 语句的描述。
- 由命令行处理器（CLP）从成功的连接（CONNECT 或 ATTACH）返回的认证标识被截断为 8 个字符。省略号 (...) 被迫加至认证标识，以指示截断。
- DB2 版本 7 支持长度最多为 30 个字节的模式名，但以下情况除外：
 - 不能复制模式名长度超过 18 个字节的表。
 - 用户定义类型（UDT）的模式名长度不能超过 8 个字节。

用户标识和口令

在标题为『用户标识和口令』这一节内，将对“A 到 Z”的引用更改为：

单字节大写和小写的拉丁字母 (A 到 Z、a 到 z)。对其他字母和字符的支持取决于所使用的代码页。有关代码页支持的详情, 参见标题为“国家语言支持 (NLS)”的附录。

附录 D. 发行版之间的不兼容性

Windows NT DLFS 与 Norton 的实用程序不兼容

Windows NT Data Links File System 与“Norton 实用程序”不兼容。从由 DLFS 控制的驱动器删除文件时, 会出现内核异常: 错误 0x1E (未能处理的内核方式异常)。该异常为 0xC0000005 (访问违例)。此访问违例发生的原因是“Norton 实用程序”驱动程序是在装入 DLFS 过滤器驱动程序后装入的。

临时解决方法是先装入“Norton 实用程序”驱动程序, 再装入 DLFS 驱动程序。可通过将 DLFS 驱动程序启动更改为人工来完成此解决方法。单击**开始**并选择**设置** → **控制面板** → **设备** → **DLFS**, 并将其设置为人工。

可创建批处理文件, 将其添加至启动文件夹, 用来在系统启动时装入 DLFS 驱动程序和“DLFS 服务”。该批处理文件的内容如下所示:

```
net start dlfsd
net start "dlfs service"
```

将此批处理文件命名为 `start_dlfs.bat`, 并将其复制到 `C:\WINNT\Profiles\Administrator\Start Menu\Programs\Startup` 目录中。

只有管理员才有权装入 DLFS 过滤器驱动程序和 DLFS 服务。

SET CONSTRAINTS 被替换为 SET INTEGRITY

SET CONSTRAINTS 语句被替换为 SET INTEGRITY 语句。为向下兼容起见, 两个语句在 DB2 UDB V7 中都是可接受的。

附录 E. 国家语言支持

DB2 版本 7 的国家语言版本

DB2 版本 7 具有下列语言版本: 英语、法语、德语、意大利语、西班牙语、巴西葡萄牙语、日语、韩语、简体中文、繁体中文、丹麦语、芬兰语、挪威语、瑞典语、捷克语、荷兰语、匈牙利语、波兰语、土耳其语、俄语、保加利亚语和斯洛文尼亚语。

在基于 UNIX 的平台上, 可安装几种不同语言版本的 DB2 产品消息和库。DB2 安装实用程序把消息目录文件集放置到给定平台的最常用语言环境目录中, 如下列表所示。表4 提供了关于 AIX、HP-UX 和 Solaris 的信息。第57页的表5 提供了关于 Linux、Linux/390、SGI 和 Dynix 的信息。

表 4. AIX、HP-UX、Solaris

操作系统	AIX		HP-UX		Solaris	
语言	语言环境	Cde Pg	语言环境	Cde Pg	语言环境	Cde Pg
法语	fr_FR	850	fr_FR.iso88591	1051	fr	
	Fr_FR		fr_FR.roman8			

表 4. AIX、HP-UX、Solaris (续)

德语	de_DE	850	de_DE.iso88591	1051	de	
	De_DE		de_DE.roman8			
意大利语	it_IT	850	it_IT.iso88591	1051	it	
	It_IT		it_IT.roman8			
西班牙语	es_ES	850	es_ES.iso88591	1051		
	Es_ES		es_ES.roman8			
巴西葡萄牙语	pt_BR				pt_BR	
日语	ja_JP	954	ja_JP.eucJP	954	ja	954
	Ja_JP	932				
韩国语	ko_KR	970	ko_KR.eucKR	970	ko	970
简体中文	zh_CN	1383	zh_CN.hp15CN	1383	zh	1383
	Zh_CN.GBK	1386				
繁体中文	zh_TW	964	zh_TW.eucTW	964	zh_TW	964
	Zh_TW	950	zh_TW.big5	950	zh_TW.BIG5	950
丹麦语	da_DK	850	da_DK.iso88591	1051	da	
	Da_DK		da_DK.roman8			
芬兰语	fi_FI	850	fi_FI.iso88591	1051	fi	
	Fi_FI		fi_FI.roman8			
挪威语	no_NO	850	no_NO.iso88591	1051	no	
	No_NO		no_NO.roman8			
瑞典语	sv_SE	850	sv_SE.iso88591	1051	sv	
	Sv_SE		sv_SE.roman8			
捷克语	cs_CZ	912				
匈牙利语	hu_HU	912				
波兰语	pl_PL	912				
荷兰语	nl_NL	850				
	Nl_NL					
土耳其语	tr_TR	920				
俄语	ru_RU	915				
保加利亚语	bg_BG	915	bg_BG.iso88595	915		
斯洛文尼亚语	sl_SI	912	sl_SI.iso88592	912	sl_SI	912

表 5. Linux、Linux/390、SGI、Dynix

操作系统	Linux		Linux/390		SGI		Dynix	
语言	语言环境	C d e Pg	语言环境	C d e Pg	语言环境	C d e Pg	语言环境	C d e Pg
法语	fr		fr				fr	

表 5. Linux、Linux/390、SGI、Dynix (续)

德语	de		de				de	
意大利语								
西班牙语								
巴西葡萄牙语								
日语	ja_JP.ujis	954	ja_JP.ujis	954			ja_JP.EUC	954
韩国语	ko	970	ko	970	ko_KO.euc	970		
简体中文	zh_zh_CN.GBK	1386	zh_zh_CN.GBK	1386				
繁体中文	zh_TW.Big5	950	zh_TW.Big5	950				
丹麦语								
芬兰语								
挪威语								
瑞典语								
捷克语								
匈牙利语								
波兰语								
荷兰语							nl	
土耳其语								
俄语								
保加利亚语								
斯洛文尼亚语								

如果您的系统使用的代码页与以上提供的相同，但语言环境名不同，可以创建一个与相应消息目录的链接，这样仍可以看到翻译了的消息。

例如，如果 AIX 机器缺省语言环境为 ja_JP.IBM-eucJP，而 ja_JP.IBM-eucJP 的代码页为 954，则可通过发出以下命令创建从 /usr/lpp/db2_07_01/msg/ja_JP.IBM-eucJP 至 /usr/lpp/db2_07_01/msg/ja_JP 的链接：

In -s /usr/lpp/db2_07_01/msg/ja_JP /usr/lpp/db2_07_01/msg/ja_JP.IBM-eucJP

执行完此命令后，出现的所有 DB2 消息都是日语版本的。

控制中心和文档文件集

控制中心、控制中心帮助和文档文件集都放在目标工作站上的以下目录中：

- DB2 AIX 版：
 - /usr/lpp/db2_07_01/cc/%L
 - /usr/lpp/db2_07_01/java/%L
 - /usr/lpp/db2_07_01/doc/%L
 - /usr/lpp/db2_07_01/qp/\$L
 - /usr/lpp/db2_07_01/spb/%L
- DB2 HP-UX 版：
 - /opt/IBMd2/V7.1/cc/%L
 - /opt/IBMd2/V7.1/java/%L

- /opt/IBMd2/V7.1/doc/%L
- DB2 Linux 版:
 - /usr/IBMd2/V7.1/cc/%L
 - /usr/IBMd2/V7.1/java/%L
 - /usr/IBMd2/V7.1/doc/%L
- DB2 Solaris 版:
 - /opt/IBMd2/V7.1/cc/%L
 - /usr/IBMd2/V7.1/java/%L
 - /opt/IBMd2/V7.1/doc/%L

“控制中心”文件集是以 Unicode 代码页的形式提供的。文档和控制中心帮助文件集是浏览器可识别的代码页。如果系统使用的语言环境与所提供的不同，通过创建与相应语言目录的链接，仍可以运行控制中心的已翻译版本和查看已翻译版本的帮助信息。

例如，如果您的 AIX 机器的缺省语言环境为 ja_JP.IBM-eucJP，则您可通过发出下列命令来创建从 /usr/lpp/db2_07_01/cc/ja_JP.IBM-eucJP 至 /usr/lpp/db2_07_01/cc/ja_JP 的链接和从 /usr/lpp/db2_07_01/doc/ja_JP.IBM-eucJP 至 /usr/lpp/db2_07_01/doc/ja_JP 的链接：

- **In -s /usr/lpp/db2_07_01/cc/ja_JP /usr/lpp/db2_07_01/cc/ja_JP.IBM-eucJP**
- **In -s /usr/lpp/db2_07_01/doc/ja_JP /usr/lpp/db2_07_01/doc/ja_JP.IBM-eucJP**

执行完此命令后，出现的所有“控制中心”和帮助文本都是日语版本的。

注：不支持“Web 控制中心”在 Linux/390 或 NUMA-Q 上以本地方式运行。从客户机工作站，可以使用“Web 控制中心”来管理这些平台上的数据库。

DB2 管理服务器的语言环境设置

请确保“DB2 管理服务器”实例的语言环境与 DB2 实例的语言环境兼容。否则，DB2 实例不能与“DB2 管理服务器”通信。

如果未在“DB2 管理服务器”的用户简要表中设置 LANG 环境变量，则将以缺省系统语言环境来启动“DB2 管理服务器”。如果未定义缺省系统语言环境，则将以代码页 819 来启动“DB2 管理服务器”。如果 DB2 实例使用其中一个 DBCS 语言环境，且以代码页 819 启动“DB2 管理服务器”，则该实例将不能够与“DB2 管理服务器”通信。

“DB2 管理服务器”的语言环境必须与 DB2 实例的语言环境兼容。例如，在简体中文 Linux 系统上，应在“DB2 管理服务器”的用户简要表中设置“LANG=zh_CN”。

DB2 UDB 在 Windows 平台上支持 Baltic Rim 代码页 (MS-1257)

在 Windows 32 位操作系统上，DB2 UDB 支持 Baltic Rim 代码页 (MS-1257)。此代码页用于拉脱维亚语、立陶宛语和爱沙尼亚语。

派生代码页值

在标题为『派生代码页值』的一节中，将第一段：

但是，不需要设置 DB2CODEPAGE 注册表变量，因为 DB2 将从操作系统确定适当的代码页值。

更改为：通常，不需要设置 DB2CODEPAGE 注册表变量，因为 DB2 会自动从操作系统派生代码页信息。

国家代码和代码页支持

在标题为『国家代码和代码页支持』的一节中，向表中添加以下信息：

代码页	组	代码集	国家 Tr.	代码	语言环境	操作系统	国家名
943	D-1	IBM-943	JP	81	ja_JP.PCK	Sun	日本

字符集

在标题为『字符集』一节的『标识符的字符集』小节中，将第一段中的最后两个句子替换为下列内容：

在 NLS 环境中要小心使用特殊字符 #、@ 和 \$，因为它们未包括在 NLS 主机 (EBCDIC) 不变量字符集中。还可以使用扩充字符集中的字符，这取决于所使用的代码页。如果您正在一个多代码页环境中使用数据库，则必须确保所有代码页都支持您计划使用的扩充字符集中的任何元素。

管理指南: 实现

添加或扩充 DMS 容器 (新进程)

DMS 容器 (文件容器和原始设备容器) 的添加 (在创建表空间期间或之后) 或扩充已通过预取器以并行方式完成。要通过对容器并行执行这些创建 / 调整大小操作来获取增长, 可增加系统中运行的预取器的数目。未以并行方式执行的唯一进程就是记录这些操作, 如果是创建容器, 则该进程应是对容器的标记。

注: 如果预取器的数目等于被添加的容器的数目, 则 CREATE TABLESPACE / ALTER TABLESPACE (涉及将新容器添加至现存表空间) 的并行执行不会增加表空间的大小。

第 1 章 使用 GUI 工具管理 DB2

在『警报中心』一节中, 除去该节中最后两句。

在『性能监控器』一节中, 从小节『监控时间点的性能』的“定义性能变量”列表中除去第二个加重项目。

而且, 在同一小节的稍后部分, 该节中的最后几段应重写为如下所示:

对于每一个时间点, 都会监控各种性能变量。可从任何“快照监控器”窗口的帮助菜单中获取“性能变量参考帮助”, 它会提供所有性能变量的描述。这些变量被分成几种类别。存在有下列类别:

- 实例: 代理程序、连接、排序
- 数据库: 锁定和死锁、缓冲池和 I/O、连接、排序、SQL 语句活动
- 表: 表
- 表空间: 缓冲池和 I/O
- 数据库连接: 缓冲池和 I/O、锁定和死锁、排序、SQL 语句活动

有关如何生成快照的详情, 参见联机帮助。

在同一节中, 除去小节『警报中心中出现对象时所需的操作』中的最后一句

第 3 章 创建数据库

创建表空间

在 Linux 上使用原始 I/O

Linux 有一个原始设备节点池, 必须将其绑定至块设备, 才能对其执行原始 I/O。有一个原始设备控制器可充当块设备绑定信息的原始中央库。绑定是使用名为原始的实用程序执行的, 它通常是由 Linux 发行人提供的。

在 Linux 上设置原始 I/O 之前, 需要下列各项:

- 一个或多个空闲 IDE 或 SCSI 磁盘分区
- Linux 内核 2.4.0 或更新版本（但是，某些 Linux 分布式系统在 2.2 内核上提供原始 I/O）。
- 名为 /dev/rawctl 或 /dev/raw 的原始设备控制器。如果没有的话，创建一个符号链接：

```
# ln -s /dev/your_raw_dev_ctrl /dev/rawctl
```
- 原始实用程序，通常是随 Linux 分布式系统提供的
- DB2 版本 7.1 修订包 3 或更新版本

注：当前支持原始 I/O 的分布式系统，原始设备节点的命名是不同的：

分布式系统	原始设备节点	原始设备控制器
RedHat 6.2	/dev/raw/raw1 to 255	/dev/rawctl
SuSE 7.0	/dev/raw1 to 63	/dev/raw

DB2 支持上述任一种原始设备控制器，及原始设备节点的大部分其他名称。
Linux/390 上的 DB2 不支持原始设备。

要在 Linux 上配置原始 I/O:

在本示例中，要使用的原始分区为 /dev/sda5。它不应包含任何重要数据。

步骤 1. 计算此分区中的 4096 字节页的数目，必要时进行下舍入。例如：

```
# fdisk /dev/sda
Command (m for help): p

Disk /dev/sda: 255 heads, 63 sectors, 1106 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot   Start    End  Blocks  Id System
  /dev/sda1             1    523   420097   83  Linux
  /dev/sda2             524   1106  4682947+   5  Extended
  /dev/sda5             524   1106   4682947   83  Linux

Command (m for help): q
#
```

/dev/sda5 中的页数量为

```
num_pages = floor( ((1106-524+1)*16065*512)/4096 )
num_pages = 11170736
```

步骤 2. 将未使用的原始设备节点绑定至此分区。这需要在每次重新引导机器时完成，并需要 root 用户访问权。使用 raw -a 来查看已在使用中的原始设备节点：

```
# raw /dev/raw/raw1 /dev/sda5
/dev/raw/raw1: bound to major 8, minor 5
```

步骤 3. 设置对原始设备控制器和磁盘分区的的全局读许可权。设置对原始设备的全局读写许可权：

```
# chmod a+r /dev/rawctl
# chmod a+r /dev/sdb1
# chmod a+rw /dev/raw/raw1
```

步骤 4. 在 DB2 中创建表空间，指定原始设备，而不是磁盘分区。例如：

```
CREATE TABLESPACE dms1
MANAGED BY DATABASE
USING (DEVICE '/dev/raw/raw1' 11170736)
```

对于受 DB2 支持的所有其他页大小，原始设备上的表空间也是受支持的。

创建序列

在『在新表上定义标识列』一节后添加『创建序列』一节：

序列是一个数据库对象，它允许自动生成值。在理论上，序列非常适合于生成唯一键值的任务。应用程序可使用序列来避免可能因在数据库之外生成唯一计数器而产生的并行度和性能的问题。

与标识列属性不同，序列不会受特定表列的约束，也不会被绑定至唯一表列，且只能通过该表列来访问它。

可创建或改变序列，它以下列其中一种方式来生成值：

- 单向递增或递减而没有绑定
- 单向递增或递减至用户定义限制并停止
- 单向递增或递减至用户定义限制并返回至起始处再次开始

下面是创建序列对象的示例：

```
CREATE SEQUENCE order_seq
  START WITH 1
  INCREMENT BY 1
  NOMAXVALUE
  NOCYCLE
  CACHE 24
```

在此示例中，序列被称为 `order_seq`。它将从 1 开始，每次递增 1，且没有上限。应不会循环回起始处并从 1 重新开始，因为没有指定上限。与 `CACHE` 参数相关联的数指定数据库管理器预先分配内保留在内存中的序列值的最大数目。

生成的序列号具有下列特性：

- 这些值可以是任何精确的数字数据类型，且带有零标度。这样的数据类型包括：`SMALLINT`、`BIGINT`、`INTEGER` 和 `DECIMAL`。
- 可使用任何指定的整数增量来区分连续值。缺省增量为 1。
- 计数器值是可恢复的。需要恢复时，可通过日志来重建计数器值。
- 可对值进行高速缓存以提高性能。对值进行预分配并存储至高速缓存可减少对序列生成值时日志的同步 I/O。如果系统故障，则所有未提交的高速缓存的值决不会被使用，并视为已丢失。对 `CACHE` 指定的值是可能丢失的序列值的最大数目。

如果包含一个或多个序列的数据库被恢复至先前时间点，则这可能会导致对某些序列生成重复的值。要避免可能的重复值，不应将带有序列的数据库恢复至先前时间点。

序列仅在单节点数据库中受支持。

有两个表达式可与序列配合使用。

`PREVVAL` 表达式会对当前会话内的先前语句返回指定序列最新生成的值。

`NEXTVAL` 表达式会返回指定序列的下一个值。如果 `NEXTVAL` 表达式指定了序列的名称，则会生成新的序列号。但是，如果有多个 `NEXTVAL` 表达式实例在查询内指定相同序列名，则序列的计数器对每一行结果只递增一次。

可通过引用序列号，并对第一个表使用 NEXTVAL 表达式，对所有其他表使用 PREVVAL 表达式，以将相同序列号用作两个不同表中的唯一键值。

例如：

```
INSERT INTO order (orderno, custno)
VALUES (NEXTVAL FOR order_seq, 123456);
INSERT INTO line_item (orderno, partno, quantity)
VALUES (PREVVAL FOR order_seq, 987654, 1)
```

可在下列位置使用 NEXTVAL 或 PREVVAL 表达式：

- INSERT 语句，VALUES 子句
- SELECT 语句，SELECT 列表
- SET 赋值语句
- UPDATE 语句，SET 子句
- VALUES 或 VALUES INTO 语句

比较 IDENTITY 列和序列

在标题为『创建序列』的新一节后添加以下一节：

虽然 IDENTITY 列与序列之间有相似之处，但它们还是有差异的。在设计数据库和应用程序时可利用每一项的特征。

标识列具有下列特征：

- 仅当创建表时，标识列才能被定义为该表的一部分。一旦创建了表，就不能改变它来添加标识列。（但是，可能会改变现存标识列特征。）
- 标识列可自动为单个表生成值。
- 如果标识列被定义为 GENERATED ALWAYS，则使用的值将始终是由数据库管理器生成的。在修改表的内容期间，不允许应用程序提供它们自己的值。

序列对象具有下列特征：

- 序列对象是不受任何一个表约束的数据库对象。
- 序列对象生成的顺序值可用在任何 SQL 语句中。
- 因为任何应用程序都可使用序列对象，所以有两个表达式可用来控制指定序列中的下一个值及执行该语句之前生成的值的检索。PREVVAL 表达式会对当前会话内的先前语句返回指定序列最新生成的值。NEXTVAL 表达式会返回指定序列的下一个值。使用这些表达式允许您在几个表内的几个 SQL 语句中使用同一值。

虽然这些并非这两项的所有特征，但这些特征将帮助您根据数据库设计和使用该数据库的应用程序来确定要使用哪一个。

创建索引、索引扩展项或索引规范

在标题为『创建索引、索引扩展项或索引规范』的一节中，在以以下语句开头的段落中添加下列注释：“作为索引键的一部分的任何列被限制为 255 字节。”

注：DB2_INDEX_2BYTEVARLEN 注册表变量可用来允许长度超过 255 字节的列被指定为索引键的一部分。

第 4 章 改变数据库

在『改变表空间』一节中，将添加下列新的部分：

将容器添加至分区上的 SMS 表空间

可将容器添加至当前没有任何容器的分区（或节点）上的 SMS 表空间。

表空间的内容会在所有的容器间重新进行平衡。在再平衡期间，对表空间的访问不会受到限制。如果需要添加多个容器，应一次全部添加。

要使用命令行将容器添加至 SMS 表，输入以下内容：

```
ALTER TABLESPACE <名称>
  ADD ('<路径>')
  ON NODE (<分区号>)
```

按编号指定的分区，以及分区范围内的每个分区（或节点），必须存在于对其定义了表空间的节点组之中。分区号可能只会显式出现，或刚好在一个语句的 *on-nodes-clause* 范围内出现。

下例显示如何在基于 UNIX 的操作系统上将新容器添加至：分区号为 3 且由表空间 "plans" 使用的节点组：

```
ALTER TABLESPACE plans
  ADD ('/dev/rhdisk0')
  ON NODE (3)
```

在标题为『更改表属性』的一节后添加下列几节：

改变标识列

使用 ALTER TABLE 语句来修改现存标识列的属性。有关此语句（包括其语法）的详情，参考 *SQL Reference*。

有几个方法可用来修改标识列，以使其具有序列的某些特征。

下列任务对于 ALTER TABLE 和标识列都是唯一的：

- RESTART 会将与标识列相关联的序列复位至隐式或显式指定的值，以作为最初创建该标识列时的起始值。
- RESTART WITH <numeric-constant> 会将与标识列相关联的序列复位至精确的数字常量值。该数字常量可以是小数点右边没有非零位的任何正值或负值，该值也可以指定给标识列。

改变序列

使用 ALTER SEQUENCE 语句来修改现存序列的属性。有关此语句（包括其语法）的详情，参考 *SQL Reference*。

可修改的序列属性包括：

- 更改未来值之间的增量
- 建立新的最小值或最大值
- 更改高速缓存序列号的数目
- 更改序列是否循环

- 更改序列号是否必须以请求的次序生成
- 重新启动序列

有两个任务未作为创建序列的一部分。它们是：

- **RESTART**。将序列复位为隐式或显式指定的值，以作为创建序列时的起始值。
- **RESTART WITH** 数字常量。将序列复位为精确的数字常量值。该数字常量可以是小数点右边没有非零位的任何正值或负值。

重新启动序列或更改为 **CYCLE** 后，有可能会生成重复的序列号。仅将来的序列号才会受 **ALTER SEQUENCE** 语句的影响。

序列的数据类型不能更改。而是，必须删除当前序列，然后创建指定新数据类型的新序列。

改变序列时，DB2 未使用的所有高速缓存序列值都会丢失。

删除序列

要删除序列，使用 **DROP** 语句。有关此语句（包括其语法）的详情，参考 *SQL Reference*。

可使用下列语句来删除特定序列：

```
DROP SEQUENCE sequence_name
```

其中 `sequence_name` 是要删除的序列的名称，并包括确切标识现存序列的隐式或显式模式名。

针对 **IDENTITY** 列由系统创建的序列不能使用 **DROP SEQUENCE** 语句来删除。

一旦删除了序列，该序列的所有特权也会被删除。

切换表空间的状态

如果与处于 **OFFLINE** 状态的表空间相关联的容器已变成可访问，则 **ALTER TABLESPACE** 语句的 **SWITCH ONLINE** 子句可用于将该表空间置为 **ONLINE** 状态。该表空间被置为 **ONLINE** 状态，而数据库的其余表空间仍处于打开状态且在使用中。

使用此子句的一个替代方法是断开所有应用程序与数据库的连接，然后，再次将这些应用程序连接至该数据库。这会使表空间的状态从 **OFFLINE** 变为 **ONLINE**。

要使用命令行来将表空间切换为 **ONLINE** 状态，输入：

```
ALTER TABLESPACE <名称>  
SWITCH ONLINE
```

在 DMS 表空间中修改容器

DMS 表空间现在是以并行方式创建和调整大小的，这提供了性能优势。并行度相当于预取器的数目加 1。

第 5 章 控制数据库访问

在『索引特权』一节后添加以下一节：

序列特权

序列的创建程序会自动接收 USAGE 特权。USAGE 特权是对序列使用 NEXTVAL 和 PREVVVAL 表达式所必需的。要允许其他用户使用 NEXTVAL 和 PREVVVAL 表达式，必须将序列特权授予 PUBLIC。这将允许所有用户对指定序列使用这些表达式。

在『使用审计设施监控对数据库的访问』一节后添加以下一节：

数据加密

安全性计划的一部分可能涉及加密数据。为此，可使用加密和解密内置函数：ENCRYPT、DECRYPT_BIN、DECRYPT_CHAR 和 GETHINT。有关这些函数（包括其语法）的详情，参考《发行说明》的 SQL Reference 一节。

ENCRYPT 函数使用基于口令的加密方法对数据进行加密。这些函数还允许您封装口令提示。口令提示会嵌入加密数据中。一旦加密，解密数据的唯一方法就是使用正确的口令。选择使用这些函数的开发者应计划对忘记口令和不可用数据的管理。

ENCRYPT 函数的结果与第一个自变量为同一数据类型。

只有 VARCHAR 才会被加密。

结果的声明长度为下列其中之一：

- 如果指定了可选提示参数，则数据自变量的长度加 42。
- 如果未指定可选提示参数，则数据自变量长度加 10。

DECRYPT_BIN 和 DECRYPT_CHAR 函数使用基于口令的解密方法来对数据库进行解密。

DECRYPT_BIN 和 DECRYPT_CHAR 函数的结果与第一个自变量为同一数据类型。

结果的声明长度为原始数据的长度。

GETHINT 函数返回封装的口令提示。口令提示是一个短语，用来帮助数据所有者记起口令。例如，单词“大海”可用作提示以记起口令“太平洋”。

用来加密数据的口令是以下列其中一种方式确定的：

- 口令自变量。口令是调用 ENCRYPT 函数时显式传送的字符串。数据是使用给定口令进行加密和解密的。
- 专用寄存器口令。SET ENCRYPTION PASSWORD 语句对口令值进行加密，并将加密口令发送至数据库管理器以存储至专用寄存器。不带口令参数调用 ENCRYPT、DECRYPT_BIN 和 DECRYPT_CHAR 函数时会使用 ENCRYPTION PASSWORD 专用寄存器中的值。

专用寄存器的初始值或缺省值是空字符串。

口令的有效长度为 6 到 127 个字符之间（包括 6 和 127 个字符）。提示的有效长度为 0 到 32 个字符之间（包括 0 和 32 个字符）。

如果 ENCRYPTION PASSWORD 专用寄存器是通过客户机设置的，则口令会在客户机上进行加密，发送至数据库服务器，然后进行解密。要确保口令不可读，还应在数

数据库服务器上对它进行再加密。DECRYPT_BIN 和 DECRYPT_CHAR 函数必须对专用寄存器进行解密，然后才能使用。在 ENCRYPTION PASSWORD 中找到的值也是不可读的。网关安全性不受支持。

第 8 章 恢复数据库

如何使用暂挂 I/O

在第 8 章『恢复数据库』中，添加有关使用暂挂 I/O 功能的以下新节：

注：以下有关 db2inidb 实用程序的信息将替代《版本 7.2 新增内容》一书中的信息。

db2inidb 是随 DB2 交付的一个新工具，可用来执行应急恢复，使数据库处于前滚暂挂状态。

通过提供联机分割镜象处理（即分割镜象而不关闭数据库）的完全实现，暂挂的 I/O 支持系统的持续可用性。如果客户不能负担对大型数据库的脱机或联机备份，可通过使用暂挂的 I/O 和分割镜象从镜象映象执行备份或系统复制。

根据镜象存储器设备的方式，db2inidb 的使用会有所变化。以下使用方式假定在存储器系统中整个数据库的镜象方式是一致的。

在多节点环境中，必须在每个分区上运行 db2inidb 工具，才能从任何一个分区使用分割映象。可同时在所有分区上运行 db2inidb 工具。

1. 建立克隆数据库

此处的目的是克隆主数据库，以用于只读目的。以下过程描述了建立克隆数据库的可能方式：

- a. 输入以下命令以在主系统上暂挂 I/O:

```
db2 set write suspend for database
```

- b. 使用操作系统级别命令从主数据库分割镜象。

- c. 输入以下命令以在主系统上继续执行 I/O 操作:

```
db2 set write resume for database
```

运行该命令之后，主系统上的数据库应返回正常状态。

- d. 从另一机器连接至镜象数据库。

- e. 输入以下命令来启动数据库实例:

```
db2start
```

- f. 输入以下命令启动 DB2 应急恢复:

```
db2inidb database_name AS SNAPSHOT
```

注：此命令将回滚分割时处于暂挂状态的事务所作的更改。

还可对脱机备份使用此过程，但如果在主系统上进行了恢复，则此备份不能用来前滚，因为日志链不匹配。

2. 将分割镜象用作备用数据库

因为镜象（备用）数据库会通过日志不断前滚，所以会经常从主系统取装主数据库创建的新日志。以下过程描述分割镜象如何可用作备用数据库：

- a. 在主数据库上暂挂 I/O 写操作。
 - b. 从主系统中分割镜像。
 - c. 在主数据库上继续进行 I/O 写操作，以使主数据库返回至正常处理。
 - d. 将镜像数据库连接至另一实例。
 - e. 将镜像置于前滚暂挂状态，并对镜像执行前滚操作。运行 `db2inidb` 工具 (`db2inidb <db_alias> as standby`) 以除去暂挂写状态，并将镜像数据库置于前滚暂挂状态。
 - f. 通过设置出口程序从主系统中检索日志文件来复制日志，以确保最新的日志可用于此镜像数据库。
 - g. 将数据库前滚至日志末尾。
 - h. 返回至步骤 f 并重复此过程，直至主数据库关闭。
3. 将分割镜像用作备份映象
- 以下过程描述如何将镜像系统用作备份映象以恢复主系统:
- a. 使用操作系统命令复制主系统顶部的镜像数据和日志。
 - b. 输入以下命令来启动数据库实例:


```
db2start
```
 - c. 运行以下命令以将镜像数据库置于前滚暂挂状态，并除去暂挂写状态。


```
db2inidb database_alias AS MIRROR
```
 - d. 将数据库前滚至日志末尾。

增量备份和恢复

在第 8 章『恢复数据库』中，添加有关增量备份和恢复的以下新节:

随着数据库的大小，特别是仓库的大小持续扩展至太字节甚至拍字节，备份和恢复这些数据库所需的时间和硬件资源也会大幅度增长。在处理大型数据库时，数据库和表空间的完全备份并非总是最好的办法，因为这种数据库的多个副本的存储需求是非常大的。考虑以下问题:

- 如果仓库中有少量的数据发生了更改，应不必备份整个数据库。
- 将表空间追加至现存数据库，然后只备份表空间也是很危险的，因为已备份表空间外的数据也会更改。

现在，DB2 支持增量备份和恢复（但不支持长字段或大对象数据）。增量备份是一个备份映象，只包含自上次备份后进行了更新的页面。除已更新的数据和索引页外，每个增量备份映象还包含通常存储在完全备份映象中的所有初始数据库元数据（如数据库配置、表空间定义、数据库历史等等）。

受支持的两种增量备份是:

- **增量。** 增量备份映象是自最近一次成功进行完全备份操作后发生更改的所有数据库数据的副本。它又称为累积备份映象，因为随时间进行的一系列增量备份中的每一个都会有上次增量备份映象的内容。增量备份映象的前一次备份映象始终是同一对象的最近一次成功的完全备份。
- **Delta。** delta 备份映象或增量 delta 备份映象是自对该表空间的最近一次成功备份（完全、增量或 delta）后发生更改的所有数据库数据的副本。它又称为差分或非累积备份映象。delta 备份映象的前一次备份映象是在该 delta 备份映象中包含每个表空间副本的最近一次成功备份。

增量和 delta 备份映象之间的差别主要是在对随时间不断更改的对象进行连续备份时的行为。每个连续的增量映象包含前一次增量映象的完整内容，及自上次备份后发生更改或新建的任何数据。Delta 备份映象只包含自生成上次映象后后发生更改的页面。

在联机和脱机方式的操作中，会执行数据库和表空间的增量备份。在计划备份策略时应特别仔细，因为组合数据库和表空间增量备份暗示数据库备份（或多个表空间的表空间备份）的上一次备份对于单个映象并非必要，但可能是不同时期进行的唯一一组前数据库和表空间备份。

要将数据库或表空间重建为一致状态，恢复过程必须从要恢复的整个对象（数据库或表空间）的一致映象开始，然后，必须以下述次序应用每个适当的增量备份映象（参见『恢复方法』一节）。

要启用数据库更新的跟踪，DB2 应支持新的数据库配置参数 TRACKMOD，它可为下列两个接受值之一：

- NO（否）。此配置不允许增量备份。不会以任何方式跟踪或记录数据库页面更新。
- YES（是）。此配置允许增量备份。启用更新跟踪时，更改会在第一次成功连接至实例中的任何数据库时生效。必须进行完全数据库备份，才能执行增量备份。

现存数据库的缺省 TRACKMOD 设置为 NO；对于新建数据库则为 YES。

对于 SMS 和 DMS 表空间，跟踪密度应为表空间级别。

尽管很小，但数据库更新的跟踪还是会影影响更新或插入数据的事务的运行性能。

从增量备份映象恢复

从增量备份映象进行的恢复操作总是由下列步骤组成：

1. 标识增量目标映象。DBA 首先必须确定要恢复的最终映象，并向 DB2 恢复实用程序请求增量恢复操作。此映象又称增量恢复的目标映象，因为它将是要恢复的最后一个映象。对此映象发出增量恢复命令可能会导致使用此目标映象的配置和表空间定义创建新的数据库。增量目标映象是在 RESTORE DATABASE 命令中使用 TAKEN AT 参数指定的。
2. 恢复最新的完全数据库或表空间映象以建立可对其应用每个后续增量备份映象的基线。
3. 在存储在“步骤 2”中的基线映象的顶部，以生成每个所需的完全或表空间增量备份映象的次序来恢复它们。
4. 重复“步骤 3”直到再次读取“步骤 1”中的目标映象为止。在完整的增量恢复操作中，目标映象会被访问两次。在第一次访问时，只会从映象读取初始数据；而不会读取用户数据。只有在第二次访问时才会读取并处理完整映象。

必须对增量恢复操作的目标映象访问两次，才能确保一开始使用正确的历史、数据库配置和数据库的表空间定义来配置该数据库，该数据库将会在恢复操作期间创建。如果进行了初始完全数据库备份映象后就删除了表空间，则该映象的表空间数据将从备份映象中读取，但在增量恢复处理期间会被忽略。

例如：

1. <ts> 执行的 db2 恢复数据库样本增量

其中：

<ts> 指是要恢复的最后一个增量备份映象

2. <ts1> 执行的 db2 恢复数据库样本增量

其中:

<ts1> 指的是初始完全数据库 (或表空间) 映像

3. <tsX> 执行的 db2 恢复数据库样本增量

其中:

<tsX> 指的是创建序列中的每一个增量备份映像

4. 重复 “步骤 3”, 恢复每个增量备份映像直至映像 <ts> (包括该映像)

如果正在尝试数据库恢复操作, 且已经生成了表空间增量备份映像, 则必须以表空间映像的备份时间戳记的时间次序来恢复这些映像。

并行恢复

DB2 现在使用多个代理程序来执行应急恢复和数据库前滚恢复。您可能期望在进行这些操作期间会有更好的性能, 特别是在对称多处理机 (SMP) 机器上; 在数据库恢复期间使用多个代理程序可利用 SMP 机器上提供的额外 CPU。

此增强功能引入的新代理程序类型为 db2agnsc。DB2 会根据机器上的 CPU 数量来选择要用于数据库恢复的代理程序的数量。对于 SMP 机器, 使用的代理程序的数量为 CPU 的数量加 1。在只有一个 CPU 的机器上, 使用三个代理程序可更有效地读取日志、处理日志记录及预取数据页面。

DB2 将日志记录分布至这些代理程序, 以便能以并行方式重复应用它们 (如果适当的话)。日志记录是以并行方式在页面级别处理的 (同一数据页面上的日志记录由同一代理程序处理), 因此, 性能会增强, 即使是在一个表上执行所有工作也是如此。

备份至命名管道

现在支持将数据库备份至基于 UNIX 的系统上的本地命名管道 (及从其进行数据库恢复)。命名管道的作者和读者必须在同一机器上。该管道必须存在, 且位于本地文件系统上。因为命名管道被视作本地设备, 所以不需要指定目标作为命名管道。以下是 AIX 示例:

1. 创建命名管道:
`mkfifo /u/dbuser/mypipe`
2. 将此管道用作数据库备份操作的目标:
`db2 backup db sample to /u/dbuser/mypipe`
3. 恢复数据库:
`db2 restore db sample into mynewdb from /u/dbuser/mypipe`

从分割映像备份

DB2 现在支持在数据库的分割镜像副本上进行完全脱机数据库备份。联机备份不受支持, 也不是必需的, 因为处于前滚暂挂状态的数据库是不可用的。如果恢复了分割镜像备份映像, 就必须对它进行前滚, 因为发生分割时可能会有活动的事务。

注: 对于 DB2 版本 7.1 修订包 3 和 DB2 版本 7.2, 此支持被限制为只包含 DMS 表空间的数据库。如果尝试在分割后备份数据库, 且该数据库包含所有 SMS 表空间, 备份将会失败。

一旦对数据库进行了分割，就必须使用 **db2inidb** 实用程序才能指定下列其中一个选项：

- 快照。这会启动应急恢复，并使数据库保持一致。新的日志链启动，而数据库将不能通过原始数据库中的任何日志进行前滚。对于任何操作（包括备份），该数据库都是可用的。
- 备用。这会将数据库置于前滚暂挂状态。应急恢复是不会执行的，而数据库仍然会不一致。
- 镜象。这会导致数据库的镜象副本替换原始数据库。数据库被置于前滚暂挂状态，而 **WRITE SUSPEND** 状态会被关闭。应急恢复是不会执行的，而数据库仍然会不一致。

以下是一些使用方案：

- 进行数据库克隆。

此处的目标是对可使用的主要数据库进行只读克隆，例如，创建一个报告。为此，遵循以下步骤：

1. 暂挂主系统上的 I/O:

```
db2 set write suspend for database
```

2. 分割镜象。使用操作系统级别命令以从主数据库分割镜象。

3. 恢复主系统上的 I/O:

```
db2 set write resume for database
```

主系统上的数据库现在应返回至正常状态。

4. 将数据库的分割镜象安装至另一主机。

5. 启动实例:

```
db2start
```

6. 启动 DB2 应急恢复:

```
db2inidb <db_name> as snapshot
```

还可对脱机备份使用此过程，但如果在主系统上进行了恢复，则此备份不能用来前滚，因为日志链不匹配。

- 将分割镜象用作备用数据库。

此处的目标是镜象（备用）数据库通过日志持续前滚，即使一直在从主系统取装主数据库创建的新日志。要将分割镜象用作备用数据库，遵循下列步骤：

1. 暂挂主系统上的 I/O:

```
db2 set write suspend for database
```

2. 分割镜象。使用操作系统级别命令以从主数据库分割镜象。

3. 恢复主系统上的 I/O:

```
db2 set write resume for database
```

主系统上的数据库现在应返回至正常状态。

4. 将数据库的分割镜象安装至另一主机。

5. 除去暂挂写状态，并将镜象数据库置于前滚暂挂状态。

```
db2inidb <db_name> as standby
```

6. 复制日志。设置用户出口程序来从主系统的归档位置检索日志文件，以便最新的日志可用于此镜像数据库。

7. 将镜像前滚至日志末尾。

```
db2 rollforward db <db_name> to end of logs
```

8. 从“步骤 6”开始重复该过程直到主数据库关闭为止。

- 使用分割镜像来恢复主系统。

以下过程描述如何将镜像系统用作备份映象以恢复主系统:

1. 复制完成。使用操作系统命令复制主系统顶部的镜像数据和日志。

2. 启动实例:

```
db2start
```

3. 将恢复的镜像置于前滚暂挂状态，并将镜像前滚至日志结束:

```
db2inidb <dbname> as mirror
```

- 进行备份而不执行应急恢复。

对分割镜像执行脱机备份而不执行应急恢复表示您可在主系统的顶部恢复此备份映象。为此，遵循以下步骤:

1. 暂挂主系统上的 I/O:

```
db2 set write suspend for database
```

2. 分割镜像。使用操作系统级别命令以从主数据库分割镜像。

3. 恢复主系统上的 I/O:

```
db2 set write resume for database
```

主系统上的数据库现在应返回至正常状态。

4. 将数据库的分割镜像安装至另一主机。

5. 启动实例:

```
db2start
```

6. 将镜像数据库置于前滚暂挂状态:

```
db2inidb <db_name> as standby
```

7. 调用数据库备份操作:

```
db2 backup database <db_name>
```

这会产生隐式数据库连接，但不会启动 DB2 应急恢复。

按需日志归档

DB2 现在支持随时关闭（而且，如果启用了用户出口选项的话，还有归档）可恢复数据库的归档日志。这允许您收集一组完整的直至已知点的日志文件，然后使用这些日志文件来更新备用数据库。

注: 按需日志归档不能保证立即归档日志文件；它会截断日志文件并发出归档请求，但它容易因用户出口程序而延迟

可通过调用新的 DB2 ARCHIVE LOG 命令或调用新的 **db2ArchiveLog** API 来启动按需日志归档。

日志镜像

在第 8 章『恢复数据库』中，添加有关使用暂挂 I/O 功能的以下新节：

DB2 现在支持数据库级别的日志镜像。镜像日志文件有助于保护数据库，以免：

- 活动日志的意外删除
- 由硬件故障导致的数据毁坏

如果担心活动日志可能会损坏（因为磁盘崩溃），应考虑使用新的 DB2 注册表变量 DB2_NEWLOGPATH2 来对数据库指定辅助路径，以管理活动日志的副本，镜像存储这些日志的卷。

DB2_NEWLOGPATH2 注册表变量允许数据库将日志文件的完全相同的另一副本写至另一路径。建议将辅助日志路径放在物理上不同的磁盘上（最好是还处于不同的磁盘控制器上的磁盘）。这样的话，磁盘控制器就不会出现单点故障。

注：因为 Windows NT 和 OS/2 不允许使用任意路径名“安装”设备，所以不可能（在这些平台上）在另一设备上指定辅助路径指定。

可启用 DB2_NEWLOGPATH2（设置为 1）或禁用它（设置为 0）。缺省值为零。如果此变量被设置为 1，则辅助路径名为 LOGPATH 变量的当前值与字符 2 的并置。例如，在 SMP 环境中，如果 LOGPATH 为 /u/dbuser/sqllogdir/logpath，则辅助的日志路径将为 /u/dbuser/sqllogdir/logpath2。在 MPP 环境中，如果 LOGPATH 为 /u/dbuser/sqllogdir/logpath，DB2 会将节点指示符追加至该路径并将 /u/dbuser/sqllogdir/logpath/NODE0000 用作主日志路径。这样的话，辅助日志路径将为 /u/dbuser/sqllogdir/logpath2/NODE0000。

如果先启用了 DB2_NEWLOGPATH2，则在下一次数据库启动时完成当前日志文件之前不会实际上使用它。这类似于当前使用 NEWLOGPATH 的方式。

如果在写至主或辅助日志路径时出错，则数据库会将失效路径标为“坏”，将一条消息写至 db2diag.log 文件，并将后续日志记录写至余下仅有的“好”的日志路径。在完成当前日志文件之前，DB2 将不会再次尝试使用“坏”的路径。如果 DB2 需要打开下一个日志文件，它将验证此路径是否有效，如果有效的话，将开始使用它。如果此路径无效，在第一次访问下一个日志文件之前，DB2 将不会再次使用该路径。系统不会试图将日志路径同步，但 DB2 会保存有关发生的访问错误的信息，以便在归档日志文件时使用正确的路径。如果写至余下的“好”的路径时发生故障，数据库会异常结束。

Sun Solaris 和 HP 上的跨平台备份和恢复支持

现在支持 Sun Solaris 和 HP 之间的跨平台备份和恢复支持。如果在系统之间传输备份映像，则必须以二进制方式传输它。在目标系统上，用来创建数据库的代码页 / 区域必须与创建原始数据库的系统上的代码页 / 区域相同。

DB2 Data Links Manager 注意事项 / 备份实用程序注意事项

将本节中的第二段替换为：

链接文件时，Data Links 服务器会调度这些文件以在不同时间复制至归档服务器（如 ADSM）或磁盘。备份实用程序运行时，DB2 会确保调度以进行复制的所有文件已完成复制。备份处理开始时，DB2 会联系在 DB2 配置文件中指定的所有 Data Links 服务器。如果 Data Links 服务器有一个或多个已链接文件，且该服务器未在运

行，或是在备份操作期间停止运行，则备份将不会包含完整的 DATALINK 信息。备份操作将成功完成。所有重要备份必须成功完成，Data Links 服务器才能被再次标记为可供数据库使用。如果在 Data Links 服务器上仍有两倍于 num_db_backups（参见下文）的值的重要备份在等待完成的情况下启动备份，备份操作将会失败。该 Data Links 服务器必须重新启动，并允许先完成重要备份，再完成其他备份。

DB2 Data Links Manager 注意事项 / 恢复和前滚实用程序注意事项

将以以下语句开头的段落：

在恢复数据库或表空间且未指定
WITHOUT DATALINK 的情况下，...

以及

在恢复数据库或表空间且指定了 WITHOUT DATALINK 选项的情况下...

替换为：

在恢复数据库或表空间时，
必须满足下列条件，才能成功执行恢复操作：

- o 如果备份文件中记录的任何 Data Links 服务器未在运行，
恢复操作仍然会成功完成。

受缺少的 Data Links 服务器影响的带有 DATALINK 列信息的表将在恢复操作（或前滚操作，如果使用的話）完成后置于数据链路协调暂挂状态。此恢复处理必须成功完成，Data Links 服务器才能被再次标记为可供数据库使用。

- o 如果备份文件中记录的任何 Data Links 服务器在恢复操作期间停止运行，恢复操作将会失败，但可在 Data Links 服务器关闭的情况下重新启动恢复操作（参见下文）。
- o 如果任何 Data Links 服务器上的先前数据库恢复操作仍未完成，
则后续数据库或表空间恢复操作将会失败，直到这些 Data Links 服务器重新启动，
未完成的恢复操作才会完成。
- o 有关记录在备份文件中的所有 DATALINK 列的信息必须存在于适当的 Data Links 服务器的注册表中。

如果有关 DATALINK 列的信息未记录在注册表中，则带有缺少的 DATALINK 列信息的表会在恢复操作（或前滚操作，如果使用的話）完成后置于数据链路协调不可能状态。

如果备份未记录在注册表中，这可能意味着提供的备份文件比 num_db_backups 的值要早，且已成为“已收集的无用单元”。这意味着此较早备份中的归档文件已被除去，且不能恢复。带有 DATALINK 列的所有表都将置于数据链路协调暂挂状态。

如果备份未记录在注册表中，这可能意味着备份处理因 Data Links 服务器未在运行而尚未完成。带有 DATALINK 列的所有表都将置于数据链路协调暂挂状态。重新启动 Data Links 服务器时，会先完成备份处理，才进行恢复处理。

该表仍然可供用户使用，但 DATALINK 列中的值可能不能正确地引用这些文件（例如，可能找不到与 DATALINK 列的值相匹配的文件）。如果不想要此行为发生，则通过发出 "SET CONSTRAINTS for tablename TO DATALINK RECONCILE PENDING" 语句将该表置于检查暂挂状态。

完成恢复操作后，如果表处于数据链路协调不可能状态，可以采取“从 Datalink_Reconcile_Not_Possible 状态中除去表”中建议的方式之一来修正 DATALINK 列数据。

第一段底部的注释仍然不变。

将下文添加至本节的末尾：

强烈建议对 datalink.cfg 文件进行归档以覆盖某些未使用的恢复操作，因为数据库备份映象中的 datalink.cfg 文件只会反映自备份时间开始的 datalink.cfg 文件。具有最新的 datalink.cfg 文件是覆盖所有恢复操作所必需的。因此，每当调用 ADD DATALINKS MANAGER 或 DROP DATALINKS MANAGER 命令后就必须备份 datalink.cfg 文件。这将有助于检索最新的 datalink.cfg 文件（如果磁盘上没有最新的 datalink.cfg

文件的话)。
如果磁盘上未提供最新的 `datalink.cfg` 文件, 则将现存的 `datalink.cfg` 文件 (该文件是从备份映像恢复的) 替换为运行前滚操作之前归档的最新 `datalink.cfg` 文件。在恢复数据库之后进行此操作。

从脱机备份恢复数据库而不进行前滚

您仅能在数据库级别而不是表空间级别进行恢复操作而不进行前滚。要恢复数据库而不进行前滚, 可恢复非可恢复数据库 (即使用循环记录的数据库), 或在 `RESTORE DATABASE` 命令上指定 `WITHOUT ROLLING FORWARD` 参数。

如果将恢复实用程序与 `WITHOUT DATALINK` 选项配合使用, 则带有 `DATALINK` 列的所有表都会置于数据链路协调暂挂 (DRP) 状态, 且在恢复操作期间不会使用 `Data Links` 服务器执行任何协调。

如果未使用 `WITHOUT DATALINK` 选项, 且备份文件中记录的 `Data Links` 服务器不再是对数据库定义的 (即, 已使用 `DROP DATALINKS MANAGER` 命令删除了该服务器), 则包含引用已删除的 `Data Links` 服务器的 `DATALINK` 数据的表会被恢复实用程序置于 `DRP` 状态。

如果未使用 `WITHOUT DATALINK` 选项, 所有 `Data Links` 服务器都可用, 且有关 `DATALINK` 列的所有信息完整地记录在注册表中, 则对于记录在备份文件中的每个 `Data Links` 服务器, 会发生下列情况:

- 用于数据库恢复操作的备份映像完成后链接的所有文件被标记为断开链接 (因为它们未记录在正在链接的备份映像中)。
- 备份映像完成后断开链接但进行备份映像前处于链接状态的所有文件被标记为已链接 (因为它们被记录在正在链接的备份映像中)。如果该文件随即链接至另一数据库中的另一个表, 则恢复的表会被置于数据链路协调暂挂状态。

注: 如果用于数据库恢复操作的备份映像是在至少有一个 `Data Links` 服务器未在运行的情况下进行的, 则不能执行上述操作, 因为备份中的 `DATALINK` 信息不完整。如果用于数据库恢复操作的备份映像是在进行了前滚或未进行前滚的数据库恢复操作后进行的, 也不能执行上述操作。在这两种情况下, 带有 `DATALINK` 列的所有表会置于数据链路协调暂挂状态, 且在恢复操作期间不会使用 `Data Links` 服务器执行任何协调。

恢复数据库和表空间, 并前滚至日志结束

如果恢复了数据库或表空间, 并将其前滚至日志结束 (表示提供了所有日志), 则不需要进行协调检查, 除非至少备份文件中记录的其中一个 `Data Links` 服务器在恢复操作期间未在运行。如果不确定是否为前滚操作提供了所有日志, 或考虑可能需要协调 `DATALINK` 值, 则执行下列操作:

1. 对涉及的表发出 `SQL` 语句:

```
SET CONSTRAINTS FOR tablename TO DATALINK RECONCILE PENDING
```

这会将表置于数据链路协调暂挂状态和检查暂挂状态。

2. 如果不要表处于检查暂挂状态, 则发出以下 `SQL` 语句:

```
SET CONSTRAINTS FOR tablename IMMEDIATE CHECKED
```

这会使表脱离检查暂挂状态, 但会让该表仍保留为数据链路协调暂挂状态。必须使用协调实用程序才能使该表脱离此状态。

备份文件包含的 DATALINK 数据可能会引用已从数据库中删除的 DB2 Data Links Manager（即进行备份时 DB2 Data Links Manager 已注册至该数据库）。对于包含至少一个带有引用已删除的 DB2 Data Links Manager 的 DATALINK 数据的表的每个正在前滚的表空间，所有的表都会被前滚实用程序置于 DRP 状态。

DB2 Data Links Manager 和恢复交互作用

下表显示可执行的不同类型的恢复，恢复和前滚处理期间发生的 DB2 Data Links Manager 处理，及恢复完成后是否需要运行“协调”实用程序：

恢复类型	恢复期间的 DB2 Data Links Manager 处理	前滚期间的 DB2 Data Links Manager 处理	协调
非可恢复数据库 (<i>logretain=NO</i>)			
完整备份的数据库恢复，所有 Data Links 服务器启动	执行快速协调	N/A	可选择运行（如果检测到文件链接问题的话）
使用 WITHOUT DATALINK 选项的数据库恢复	表置于 <i>Datalink_Reconcile_Pending</i> 状态	N/A	必需
完整备份的数据库恢复，至少一个 Data Links 服务器关闭	仅对表空间中没有指向关闭的 Data Links 服务器的链接的表执行快速协调，其他表置于 <i>Datalink_Reconcile_Pending</i> 状态	NA	对表空间中带有指向关闭的 Data Links 服务器的链接的表是必需的
不完整备份的数据库恢复，所有 Data Links 服务器启动	未执行快速协调，带有 DATALINK 列的所有表置于 <i>Datalink_Reconcile_Pending</i> 状态	NA	必需
可恢复数据库 (<i>logretain=YES</i>)			
使用 WITHOUT ROLLING FORWARD 选项的数据库恢复，使用完整备份，所有 Data Links 服务器启动	执行快速协调	N/A	可选
使用 WITHOUT ROLLING FORWARD 和 WITHOUT DATALINK 选项的数据库恢复，使用完整或不完整备份，Data Links 服务器启动或关闭	表置于 <i>Datalink_Reconcile_Pending</i> 状态	N/A	必需
使用 WITHOUT ROLLING FORWARD 选项的数据库恢复，使用完整备份，至少一个 Data Links 服务器关闭	仅对表空间中没有指向关闭的 Data Links 服务器的链接的表执行快速协调，其他表置于 <i>Datalink_Reconcile_Pending</i> 状态	N/A	对表空间中带有指向关闭的 Data Links 服务器的链接的表是必需的
使用 WITHOUT ROLLING FORWARD 选项的数据库恢复，使用不完整备份，Data Links 服务器启动或关闭	未执行快速协调，带有 DATALINK 列的所有表置于 <i>Datalink_Reconcile_Pending</i> 状态	N/A	必需

恢复类型	恢复期间的 DB2 Data Links Manager 处理	前滚期间的 DB2 Data Links Manager 处理	协调
数据库恢复并前滚至日志结束，使用完整备份，所有 Data Links 服务器启动	没有任何操作	没有任何操作	可选
数据库恢复并前滚至日志结束，使用完整备份，前滚处理期间至少一个 Data Links 服务器关闭	没有任何操作	没有任何操作	可选
数据库恢复并前滚至日志结束，使用完整或不完整备份，恢复期间任何 Data Links 服务器关闭	没有任何操作	带有 DATALINK 列的所有表置于 Datalink_Reconcile_Pending 状态	对所有带有 DATALINK 列的所有表是必需的
数据库恢复并前滚至日志结束，使用不完整备份，恢复期间所有 Data Links 服务器启动	没有任何操作	没有任何操作	可选
数据库恢复并前滚至日志结束，使用完整或不完整备份，所有 Data Links 服务器启动，任何 Data Links 服务器上的备份未知	没有任何操作	表空间中带有指向备份未知的 Data Links 服务器的链接的所有表置于 Datalink_Reconcile_Pending 状态	必需
表空间恢复并前滚至日志结束，使用完整备份，所有 Data Links 服务器启动	没有任何操作	没有任何操作	可选
表空间恢复并前滚至日志结束，使用完整备份，前滚处理期间至少一个 Data Links 服务器关闭	没有任何操作	没有任何操作	可选
表空间恢复并前滚至日志结束，使用完整或不完整备份，恢复处理期间任何 Data Links 服务器关闭	没有任何操作	表空间中带有指向关闭的任何 Data Links 服务器的链接的所有表置于 Datalink_Reconcile_Pending 状态	对表空间中带有指向关闭的任何 Data Links 服务器的链接的表是必需的
表空间恢复并前滚至日志结束，使用不完整备份，所有 Data Links 服务器启动	没有任何操作	没有任何操作	可选
数据库恢复并前滚至时间点，使用完整或不完整备份，恢复和 / 或前滚处理期间 Data Links 服务器启动或关闭	没有任何操作	表置于 Datalink_Reconcile_Pending 状态	必需
表空间恢复并前滚至时间点，使用完整或不完整备份，恢复和 / 或前滚处理期间 Data Links 服务器启动或关闭	没有任何操作	表置于 Datalink_Reconcile_Pending 状态	必需

恢复类型	恢复期间的 DB2 Data Links Manager 处理	前滚期间的 DB2 Data Links Manager 处理	协调
数据库恢复为另一数据库名、别名、主机名或实例，而没有任何前滚 ¹	表置于 <i>Datalink_Reconcile_Not_Possible</i> 状态	N/A	可选，但处于 <i>Datalink_Reconcile_Not_Possible</i> 状态的表必须人工修正
数据库恢复为另一数据库名、别名、主机名或实例，并进行前滚	没有任何操作	表置于 <i>Datalink_Reconcile_Not_Possible</i> 状态	可选，但处于 <i>Datalink_Reconcile_Not_Possible</i> 状态的表必须人工修正
从不可用的备份（已在 Data Links 服务器上对映象作了无用单元收集）进行数据库恢复，而不进行前滚 ¹ ，带有或不带 WITHOUT DATALINK 选项	表置于 <i>Datalink_Reconcile_Pending</i> 状态	没有任何操作	必需
从不可用的备份（已在 Data Links 服务器上对映象作了无用单元收集）进行数据库恢复，并进行前滚，带有或不带 WITHOUT DATALINK 选项	没有任何操作	表置于 <i>Datalink_Reconcile_Pending</i> 状态	必需
从不可用的备份（已在 Data Links 服务器上对映象作了无用单元收集）进行表空间恢复，并进行前滚	没有任何操作	表置于 <i>Datalink_Reconcile_Pending</i> 状态	必需

注：

1. 使用脱机备份和 WITHOUT ROLLING FORWARD 选项的恢复，（*logretain* 为开）或使用脱机备份的恢复（*logretain* 为关）。
2. 完整备份是在所有所需的 Data Links 服务器都在运行时进行的备份。不完整备份是在至少一个所需 Data Links 服务器未在运行时进行的备份
3. 如果用于数据库恢复操作的备份映象是在进行了前滚或未进行前滚的数据库恢复操作后进行的，则不能执行快速协调处理。这样的话，带有 DATALINK 列的所有表都会置于 *Datalink_Reconcile_Pending* 状态。

需要协调的情况的检测

以下是您可能需要运行协调实用程序的某些情况：

- 整个数据库会恢复并前滚至时间点。因为整个数据库被前滚至已提交事务，不会有任何表处于检查暂挂状态（因为参考约束或检查约束）。数据库中的所有数据被置于一致状态。但是，DATALINK 列可能不会与 DB2 Data Links Manager 中的元数据同步，并需要协调。

在这种情况下，带有 DATALINK 数据的表将已经处于 DRP 状态。应对每个表调用协调实用程序。

- 运行 DB2 Data Links Manager 的特定 Data Links 服务器会丢失对其元数据的跟踪。可能会因为不同的原因而发生这种情况。例如：
 - Data Links 服务器是冷启动的。

– Data Links 服务器元数据被恢复至返回级别状态。

在某些情况下（如在 SQL UPDATE 和 DELETE 期间），DB2 可能能够检测到 Data Links 服务器中的元数据的问题。在这些情况下，SQL 语句将会失效。通过使用 SET CONSTRAINTS 语句来将该表置于 DRP 状态，然后对该表运行协调实用程序。

- 文件系统不可用（例如，因为磁盘崩溃）且不能恢复至当前状态。在此情况下，文件可能会丢失。
- DB2 Data Links Manager 会从数据库中删除，而有一些 DATALINK FILE LINK CONTROL 值引用了该 DB2 Data Links Manager。您应对这种表运行协调实用程序。

附录 C 数据库恢复的用户出口

在『归档与检索注意事项』一节中，以下段落所述的内容不再正确，应从列表除去：

如果远程客户机与 DB2 服务器的连接断开，用户出口可能会中断。即，在通过用户出口处理日志归档时，其他以 SNA 方式连接的客户机之一死机或断电会导致向服务器发送一个信号（SIGUSR1）。该服务器将该信号传送给导致中断的用户出口。可修改用户出口程序以检查中断并继续处理。

『错误处理』一节有一个“注意事项”列表，应将“注意事项 3”的内容替换为以下信息：

- 用户出口程序请求被挂起五分钟。在此期间，将忽略所有的请求，包括导致返回码的日志文件请求。

在处理请求的五分钟挂起之后，会处理下一个请求。如果在处理此请求时未发生任何错误，则会继续处理新的用户出口程序请求，而 DB2 将对先前无法归档日志文件或被挂起的日志文件重新发出归档请求。如果在重试期间生成的返回码长度超过 8，则请求会再挂起五分钟。这样的五分钟挂起会继续下去，直到问题被校正或停止并重新启动数据库为止。

一旦所有的应用程序都断开了与数据库的连接，且数据库被重新打开，DB2 就将对上次使用数据库时可能未成功归档的任何日志文件发出归档请求。

如果用户出口程序未能归档日志文件，则您的磁盘可能已为日志文件所填充，而性能可能会因要对这些日志文件进行额外的格式化工作而下降。一旦磁盘已满，数据库管理器将不会再接受更改数据库的应用程序请求。

如果调用用户出口程序来检索日志文件，则前滚恢复被挂起但未停止，除非在 ROLLFORWARD DATABASE 实用程序中指定了停止操作。如果未指定停止操作，则可校正问题并继续进行恢复。

附录 D 对多个数据库分区服务器发出命令

在『指定要运行的命令』这一节的底部，添加下列内容：

在运行任何 korn-shell shell-script（包含要从后台的标准输入文件中读取的逻辑），则应该显式地将标准输入文件重定向至这样的源，在该源中，进程可以读取而不会在终端上停止（SIGTTIN 消息）。要重定向标准输入文件，可以按以下格式来运行脚本：

```
shell_script </dev/null &
```

如果未提供任何输入的话。

以类似的方式，当在后台运行 db2_all 时，应该总是指定 </dev/null。例如：

```
db2_all ";run_this_command" </dev/null &
```

通过执行此操作，就可以重定向标准输入文件，且不会在终端上停止。

如果不关心远程命令的输出时，此方法的一种替代方法是在 `db2_all` 前缀中使用 "daemonize" 选项：

```
db2_all ";daemonize_this_command" &
```

附录 I 高速节点间通信

下面一节已更新：

使 DB2 能够运行（使用 VI）

详细的安装信息可在 *DB2 Enterprise - Extended Edition for Windows Quick Beginnings* 中找到。

完成 DB2 的安装（如 *DB2 Enterprise - Extended Edition for Windows Quick Beginnings* 中所述）之后，设置下列 DB2 注册表变量，并在实例中的每台数据库分区服务器上执行下列任务：

- 设置 `DB2_VI_ENABLE=ON`

使用 `db2set` 命令来修改注册表变量的值。使用 `db2_all` 命令来在实例中的所有数据库分区服务器上运行 `db2set` 命令。必须使用作为“管理”组成员之一的用户帐户登录才能运行 `db2_all` 命令。

在以下示例中，“;”字符放在双引号内以允许请求在实例中的所有分区服务器上并行运行：

```
db2_all ";db2set DB2_VI_ENABLE=ON"
```

有关 `db2_all` 命令的详情，参见《管理指南》：实现中的“对多个数据库分区服务器发出命令”。

- 设置 `DB2_VI_DEVICE=nic0`

例如：

```
db2_all ";db2set DB2_VI_DEVICE=nic0"
```

注：对于 Synfinity Interconnect，此变量应设置为 `DB2_VI_DEVICE=VINIC`。设备名 (VINIC) 必须大写。

- 设置 `DB2_VI_VIPL=vipl.dll`

例如：

```
db2_all ";db2set DB2_VI_VIPL=vipl.dll"
```

注：示例中使用的值是注册表变量的缺省值。有关注册表变量的详情，参见《管理指南》：性能。

- 在 MPP 实例上输入 `db2start`。
- 复查 `db2diag.log` 文件。对于声明“VI 已启用”的每个分区，都应相应有一条信息。
- “快速通信管理器” (FCM) 配置参数可能需要更新。如果遇到了由于涉及 FCM 的资源约束而造成的问题，应提高 FCM 配置参数的值。如果是在 FCM 配置参数的值已经增加的另一高速互连环境中出现问题，则可能需要降低这些值。而且，在 Windows NT 上，您可能需要设置 `DB2NTMEMSIZE` 注册表变量以覆盖 DB2 缺省值。参考《管理指南》：性能以了解有关注册表变量的详情。

第 3 章 应用程序注意事项

指定隔离级别

在标题为『指定隔离级别』的一节中，添加下列段落：

除了在准备或绑定应用程序时设置软件包级隔离级别之外，还可以设置语句级隔离级别。语句级隔离级别是使用 **WITH** 子句指定的。

下列 SQL 语句支持语句级别的隔离：

- **SELECT** 语句
- **SELECT INTO**
- **Searched DELETE**
- **INSERT**
- **Searched UPDATE**
- **DECLARE CURSOR**

有一些与使用语句级别的隔离相关联的条件：

- **WITH** 子句不能用在子查询上
- **WITH UR** 选项只适用于只读操作。如果它在其他情况中使用，则语句自动从 “UR” 更改为 “CS”。
- 语句的缺省隔离级别是在其中绑定语句的软件包的隔离级别。
- 语句级隔离级别将覆盖为语句出现在其中的软件包指定的隔离级别。

调整优化类

在标题为『调整优化类』的一节中，将当前级别 2 信息替换为下列内容：

- 2 -** 此类指导优化器使用一种优化度，它显著地改进了第 1 类的性能，同时对于复杂查询，又使编译成本明显低于第 3 类及第 3 类以上的类。特别是：
- 利用了所有可用的统计信息，包括频率和分位数非均匀分布统计信息。
 - 应用了所有查询重写规则（包括查询总结表的路由），但是针对计算强度大情况的规则除外，因为它仅在很少情况下适用。
 - 使用了 **Greedy** 连接枚举。
 - 考虑了许多种访问方法，包括列表预取和总结表路由。
 - 考虑了星型连接方案（如果适用的话）。

优化类 2 非常类似于类 5，只不过它使用的是 **Greedy** 连接枚举而不是“动态编程”。在使用 **Greedy** 连接枚举算法的所有优化类中，此类的优化级别最高。它考虑的复杂查询替代项少得多，因此，它比类 3 及更高的类需要更少的编译时间。因此，对于决策支持或联机分析处理 (OLAP) 环境中很复杂的查询，建议使用此类。在这样的情况下，不会很频繁地执行相同的查询，因此，其访问方案不太可能保留在高速缓存中，直到下一次出现该查询为止。

动态复合语句

应该将下面这一新节添加到标题为『复合 SQL』的那一节后面:

动态复合语句将其他 SQL 语句编组成可执行块。在动态复合语句中, 可以声明 SQL 变量, 声明与 SQLSTATE 相关联的条件, 并且具有一个或多个 SQL 过程语句。如果在动态复合语句中出现错误, 则先前的所有 SQL 语句都会被回滚, 并且不会处理动态复合语句中的其余 SQL 语句。

可以将动态复合语句嵌入触发器、SQL 函数、SQL 方法, 或者通过使用动态 SQL 语句来发出该语句。可以动态地准备此可执行语句。不需要特权就可以调用该语句, 但是与该语句相关联的授权标识必须具有必需的特权才能在复合语句内调用嵌入式 SQL 语句。

变量存在于变量声明的子语句中。条件存在于子语句中(这视条件声明的 SQLSTATE 值而定)。动态复合语句被 DB2 作为单个语句来编译。此语句可以有效地用于涉及很少控制流逻辑但涉及很多数据流的简短脚本。对于具有嵌套复杂控制流的较大型构造, 应该考虑使用 SQL 过程。

有几个控制流逻辑语句可以用于动态复合语句中。这些语句包括: FOR 语句、IF 语句、ITERATE 语句和 WHILE 语句。可以在 *SQL Reference* 中找到有关这些语句以及其他受支持语句的详细信息。

第 4 章 环境注意事项

在标题为『使用索引顾问程序』的一节后面, 添加下面一节:

使用较大的索引键

可允许将长度大于 255 个字节的列指定为索引键的一部分。DB2_INDEX_2BYTEVARLEN 注册表变量允许使用 2 个字节而不是 1 个字节来存储索引键的长度。

有几个 SQL 语句会受到对注册表变量所作更改的影响。这些语句是:

- CREATE TABLE。具有变量键部分的主键、外键和唯一键可以具有大于 255 个字节的大小。
- CREATE INDEX。具有变量键部分的所有索引(包括唯一索引及包括列)可以具有大于 255 个字节的大小。
- ALTER TABLE。具有变量键部分的主键、外键和唯一键可以具有大于 255 个字节的大小。具有变量键部分的所有索引(包括唯一索引及包括列)可以具有大于 255 个字节的大小。

取消了 255 个字节的外键限制, 而不管注册表变量的值是多大。与外键相对应的主键的真实情况将强制使用任何限制。

要将现存的索引转换为使用更大的索引键: 删除索引, 将 DB2_INDEX_2BYTEVARLEN 注册表变量设置为 ON, 然后(使用更大的列来)重新创建索引)。

有关 SQL 语句(包括语法描述)的详情, 参考 *SQL Reference*。

第 5 章 系统目录统计信息

收集和使用分布统计信息

在『对等式谓词的影响示例』小节中，讨论了谓词 $C \leq 10$ 。错误率显示为 -86%。这是不正确的。该段结尾处的句子应为：

假定使用统一的数据分发，且使用公式 (1)，则满足该谓词的行数估计为 1，错误率为 -87.5%。

在『对等式谓词的影响示例』小节中，讨论了谓词 $C > 8.5 \text{ AND } C \leq 10$ 。使用线性插值估计的 r_2 值必须更改为如下内容：

$$\begin{aligned} r_2 & *= \frac{10 - 8.5}{100 - 8.5} \times (\text{具有 } > 8.5 \text{ 且 } \leq 100.0 \text{ 的值的行数}) \\ r_2 & *= \frac{10 - 8.5}{100 - 8.5} \times (10 - 7) \\ r_2 & *= \frac{1.5}{91.5} \times (3) \\ r_2 & *= 0 \end{aligned}$$

此新示例后面的段落还必须修改为如下内容：

最终估计为 $r_1 + r_2 = 7$ ，
错误率仅为 -12.5%。

用于更新目录统计信息的规则

在标题为『用于更新列统计信息的规则』的一节中，应该将第一个列表项中的最后一个公告的列表项替换为下列内容：

每当对应列中具有 3 个以上的相异值时，HIGH2KEY 就必须大于 LOW2KEY。在一列中具有 3 个或以下的相异值的情况下，HIGH2KEY 可以等于 LOW2KEY。

子元素统计信息

在“修订包 1”中提供了一个选项，用来收集和使用子元素统计信息。这些是在数据结构呈由空格定界的一系列子字段或子元素形式时，有关列中这些数据的内容的统计信息。

例如，假设有一个数据库包含表 DOCUMENTS，该表中的每行描述一个文档，并假设在 DOCUMENTS 中有一列 KEYWORDS，包含有关此文档的适当关键字列表，以便于进行文本检索。KEYWORDS 中的值可能如下所示：

```
'database simulation analytical business intelligence'  
'simulation model fruitfly reproduction temperature'  
'forestry spruce soil erosion rainfall'  
'forest temperature soil precipitation fire'
```

在此示例中，每一列的值由 5 个子元素组成，每个子元素为一个单词（关键字），且用一个空格与其他子元素隔开。

对于在这种列上指定 LIKE 谓词的查询，使用 % 匹配所有字符：

```
SELECT .... FROM DOCUMENTS WHERE KEYWORDS LIKE '%simulation%'
```

对于优化器而言，了解有关该列的子元素结构的某些基本统计信息是非常有利的，以下是一些基本统计信息：

SUB_COUNT

子元素的平均数。

SUB_DELIM_LENGTH

用来隔开每个子元素的每个定界符的平均长度，在此上下文中，定界符是一个或多个连续的空格字符。

在 KEYWORDS 列示例中，SUB_COUNT 为 5，而 SUB_DELIM_LENGTH 为 1，因为每个定界符都是单个空格字符。

在“修订包 1”中，系统管理员通过扩展 DB2_LIKE_VARCHAR 注册表变量来控制这些统计信息的收集和使用。此注册表变量会影响 DB2 UDB 优化器处理该格式的谓词的方式：

```
COLUMN LIKE '%xxxxxx'
```

其中 xxxxxx 是任意字符串；即，其搜索值以 % 字符开头的任意 LIKE 谓词。（它可以或不以 % 字符结尾）。在下文提到这些谓词时它们被称为“通配 LIKE 谓词”。对于所有谓词，优化器必须估计有多少行匹配该谓词。对于通配 LIKE 谓词，优化器会假定匹配的 COLUMN 的结构为一系列元素并置在一起组成整列，并根据字符串长度（排除前导和结尾的 % 字符）估计每个元素的长度。新语法为：

```
db2set DB2_LIKE_VARCHAR=[Y|N|S|num1][,Y|N|num2]
```

其中

- 第一项（逗号之前）意义如下所示，但仅限于没有实际子元素统计信息的列
 - S 按在 DB2 版本 2 中使用的方式来使用算法。
 - N 使用定长子元素算法。
 - Y （缺省）将变长子元素算法与算法参数的缺省值配合使用。
 - num1 使用变长子元素算法，并将 num1 用作算法参数。
- 第二项（跟在逗号之后）表示：
 - N （缺省）不要收集或使用子元素统计信息。
 - Y 收集子元素统计信息。在列带有实际子元素统计信息的情况下将使用这些统计信息的变长子元素算法与该算法参数的缺省值配合使用。
 - num2 收集子元素统计信息。在列带有实际子元素统计信息的情况下将使用这些统计信息的变长子元素算法与作为该算法参数的 num2 配合使用。

如果 DB2_LIKE_VARCHAR 的值仅包含第一项，则不会收集任何子元素统计信息，而先前收集的所有子元素统计信息将被忽略。指定的值会影响优化器以与以前相同的方法计算通配 LIKE 谓词的选择性的方式；即：

- 如果值为 S，则优化器会使用与在 DB2 版本 2 中使用的同一算法，该算法未假定子元素模型。
- 如果值为 N，则优化器会使用假定子元素模型的算法，并假定 COLUMN 为定长，即使它被定义为变长。
- 如果值为 Y（缺省值）或浮点常量，则优化器会使用假定子元素模型的算法，并认为 COLUMN 为变长（如果这样定义的话）。它还会根据查询本身而不是数据来推导子元素统计信息。此算法包含一个参数（“算法参数”），该参数用来指定该元素比用 % 字符括起来的字符串要长多少。
- 如果值为 Y，则优化器会对算法参数使用缺省值 1.9。

- 如果值为浮点常量，则优化器会对算法参数使用指定值。此常量必须在范围 0 至 6.2 之内。

如果 DB2_LIKE_VARCHAR 的值包含两项，且第二项为 Y 或浮点常量，则有关类型为 CHAR、VARCHAR、GRAPHIC 或 VARGRAPHIC 的单字节字符集字符串列的子元素统计信息是在 RUNSTATS 操作期间收集的，并在涉及通配 LIKE 谓词的查询编译期间使用。优化器使用假定子元素模型的算法、SUB_COUNT 和 SUB_DELIM_LENGTH 统计信息，以及算法参数来计算谓词的选择性。算法参数是以与指定推理算法的相同方式来指定的，即：

- 如果值为 Y，则优化器会对算法参数使用缺省值 1.9。
- 如果值为浮点常量，则优化器会对算法参数使用指定值。此常量必须在范围 0 至 6.2 之内。

如果优化器在编译期间发现未收集涉及查询的列的子元素统计信息，它将使用“推理”子元素算法；即，在只指定了 DB2_LIKE_VARCHAR 的第一项时使用的算法。因此，为使优化器可使用子元素统计信息，必须同时在 RUNSTATS 和编译期间设置 DB2_LIKE_VARCHAR 的第二项。

可通过查询 SYSIBM.SYSCOLUMNS 来查询子元素统计信息的值。例如：

```
select substr(NAME,1,16), SUB_COUNT, SUB_DELIM_LENGTH
from sysibm.syscolumns where tname = 'DOCUMENTS'
```

SUB_COUNT 和 SUB_DELIM_LENGTH 列未出现在 SYSSTAT.COLUMNS 统计信息视图中，因此不能进行更新。

注：如果使用此选项的话，RUNSTATS 运行的时间可能较长。例如，如果未使用 DETAILED 和 DISTRIBUTION 选项的话，在带有 5 个字符列的表上，RUNSTATS 运行的时间可能会延长 15% 到 40%。如果指定了 DETAILED 或 DISTRIBUTION 选项，则额外开销的百分比会少一些，即使额外开销的绝对量相同。如果考虑使用此选项，则应估计此额外开销对查询性能的改进所带来的影响。

第 6 章 了解 SQL 编译器

以下部分需要更改：

复制的总结表

以下信息将替换或添加至本节中已有的现存信息：

复制的总结表可用来对连接的并置提供帮助。例如，如果有一个星型模式，其中有一个很大的事实表，它分布在 20 个节点上，则事实表与维表之间的连接在这些表被并置的情况下最有效率。

如果将所有表放在同一节点组中，则至多能有一个维表被正确地分区来进行并置连接。所有其他维表将不能用于并置连接，原因是事实表上的连接列未能与事实表的分区键相对应。

例如，可将表 FACT (C1, C2, C3, ...) 在 C1 上进行分区；将表 DIM1 (C1, dim1a, dim1b, ...) 在 C1 上进行分区；并将表 DIM2 (C2, dim2a, dim2b, ...) 在 C2 上进行分区；以此类推。

从此示例中，可看到 FACT 与 DIM1 之间的连接是非常理想的，原因是谓词 DIM1.C1 = FACT.C1 将被并置。这两个表都是在列 C1 上进行分区的。

FACT 与 DIM2 之间带有谓词 WHERE DIM2.C2 = FACT.C2 的连接不能并置，原因是 FACT 是在列 C1 而不是在列 C2 上进行分区的。

在这种情况下，最好是在事实表的节点组中复制 DIM2。这样，就可在每个分区上以本地方式进行连接。

注：此处讨论的复制总结表涉及数据库间复制。数据库间复制涉及不同数据库中和不同操作系统上的预订、控制表和数据。如果对数据库间的复制有兴趣，参考 *Replication Guide and Reference* 以了解详情。

在创建复制的总结表时，源表可以是单节点节点组表或多节点节点组表。在大多数情况下，表很小，可放在单节点节点组中。可通过几种方式来限制要复制的数据：仅指定部分表列、通过谓词来限制行数或在创建复制的总结表时同时使用前述两种方法。

注：不需要数据捕捉选项就可使复制的总结表生效。

还可在多节点节点组中创建复制的总结表。该节点组就是放置大型表的节点组。在这种情况下，源表的副本是在节点组的所有分区上创建的。在此环境中，可更好地以本地方式（而不是将源表广播至所有分区）来完成大型事实表与维表之间的连接。

复制表上的索引不是自动创建的。索引被创建且可能与源表中标识的那些索引有所不同。

注：不能对复制的表创建唯一索引（或施行任何约束）。这样将会避免源表上不存在的约束违例。这些约束是不允许的，即使在源表上存在相同的约束。

在使用 REFRESH 语句之后，应对复制的表运行 RUNSTATS，就象对任何其他表所做的那样。

可直接在查询内引用复制的表。但是，不能将 NODENUMBER() 谓词与复制的表一起使用来查看特定分区上的表数据。

要查看是否使用了创建的复制总结表（给定引用了源表的查询），可使用 EXPLAIN 功能。首先，应确保 EXPLAIN 表存在。然后，对您感兴趣的 SELECT 语句创建一个说明方案。最后，使用 db2exfmt 实用程序来格式化 EXPLAIN 输出。

由优化器选择的访问方案可能使用也可能不使用复制的总结表，这要视需要连接的信息而定。如果优化器确定将原始源表广播至节点组中的其他分区会更经济的话，将不会使用复制的总结表。

数据访问概念和优化

『索引扫描概念』下的『多重索引访问』一节已有更改。

在本节末尾的注释前面添加以下信息：

要在扫描多个索引时实现动态位图的性能优点，可能需要更改排序堆大小 (*sortheap*) 数据库配置参数的值，以及排序堆阈值 (*sheapthres*) 数据库管理器配置参数。

在访问方案中使用动态位图时，附加排序堆空间是必需的。如果 *sheapthres* 设置为比较接近 *sortheap*（即，每个并行查询少两到三倍），则带有多重索引访问的动态位图必须使用较优化器预期要少得多的内存。

解决方案是增加 *sheapthres* 的值（相对于 *sortheap*）。

『谓词术语』下的『星形连接的搜索策略』一节已有更改。

在本节末尾添加以下信息：

所创建并作为“星形连接”技术一部分使用的动态位图使用排序堆内存。参见《管理指南》：性能手册中的第 13 章『配置 DB2』以获取有关“排序堆大小”(*sortheap*) 数据库配置参数的详情。

第 8 章 操作性能

管理数据库缓冲池

在标题为『管理数据库缓冲池』的一节中，将下列信息添加到以“创建缓冲池时，缺省情况下的页大小为 4 KB。”开头的那一段的后面：

当使用 Windows 2000 时，小于 DB2 和操作系统大小的缓冲池大小（它最大可为 64 GB）是受支持的。（这里假定 DB2 是系统上的主要产品。）可通过 Microsoft Address Windowing Extensions (AWE) 获得此支持。

尽管可以将 AWE 与任何大小的缓冲池配合使用，但是，如果您需要在更大的缓冲池上使用 AWE，则还建议使用其他 Windows 产品。“Windows 2000 高级服务器”支持多达 8 GB 的内存。“Windows 2000 数据中心服务器”支持多达 64 GB 的内存。

必须正确地配置 DB2 和 Windows 2000 才能支持 AWE 缓冲池。数据库中必须存在将利用 AWE 的缓冲池。

要分配 3 GB 的用户空间，可使用 /3GB Windows 2000 引导选项。这允许使用更大的 AWE 窗口大小。要启用通过 AWE 内存接口对超过 4 GB 内存的访问，使用 /PAE Windows 2000 引导选项。要验证您是否选择了正确的引导选项，在“控制面板”下面，选择“系统”，然后选择“启动和故障恢复”。从下拉列表中，您可以看到可用的引导选项。如果选择了您想要的引导选项（/3GB 或 /PAE），则可以继续进行设置 AWE 支持的下一任务。如果选择不了您想要的选项，则必须将该选项添加到系统驱动器上的 boot.ini 文件中。boot.ini 文件包含了启动操作系统时要执行的操作的列表。在现存参数列表末尾添加 /3GB 或 /PAE 或者这两者（用空格隔开）。一旦您保存了此更改过的文件，就可以按以上所述那样验证和选择正确的引导选项。

还必须修改 Windows 2000 以便使“lock pages in memory”-right 与在其下安装了 DB2 的用户相关联。要设置“lock pages in memory”-right，可在您作为安装了 DB2 的用户一登录至 Windows 2000 时，就在 Windows 2000 上的“开始”菜单下面选择“管理工具”文件夹，然后选择“本地安全策略”程序。在本地策略下面，可以选择“lock pages in memory”-right 的用户权利指定。

DB2 需要设置 DB2_AWE 注册表变量。要正确地设置此注册表变量，您将需要知道您希望允许 AWE 支持的缓冲池的缓冲池标识。还需要知道要分配的物理页数及地址窗口页数。要分配的物理页数应该小于可用的总物理页数。选择的实际数目将取决于工作

环境。例如，如果您具有这样一个环境，在系统上只使用了 DB2 和数据库应用程序，则可以选择将少于物理页总大小的 1/2 GB 到 1 GB 作为配合 DB2_AWE 变量使用的值。如果您具有这样一个环境，在该环境中其他非数据库应用程序正在使用系统，则您将需要增大从总大小中减去的值，以便允许更多的物理页用于其他那些应用程序。DB2_AWE 注册表变量中所使用的数值是用在 AWE 支持中以及供 DB2 使用的物理页数。地址窗口页的上限是 1.5 GB，如果选择了 Windows 2000 引导选项 /3GB 的话，则为 2.5 GB。

有关设置 DB2 注册表变量 DB2_AWE 的信息，参见本节中后面的“附录 A DB2 注册表和环境变量”中新的和更改过的注册表变量表。

管理多个数据库缓冲池

在标题为『管理多个数据库缓冲池』的小节中，在以“当使用数据库设计时，您可能已经确定具有 8 KB 页大小的表是最好的。”开头的那一段后面添加下面段落：

当使用 Windows 2000 时，可以使用 DB2_AWE 注册表变量来覆盖目录和配置文件中的缓冲池大小设置。使用此注册表变量时允许最大约为 64 GB 的缓冲池大小。

在上述同一节中，将刚好在注释前面的段落替换为下列内容：

允许数据库管理器以最小值启动的原因是允许您连接至数据库。然后，您可以重新配置缓冲池大小或执行其他重要任务，其目的是用正确的缓冲池大小重新启动数据库。不要考虑以这种状态来运行数据库更长的时间。

在标题为『重组目录和用户表』的一节中，可以将以“REORG 实用程序允许您指定临时表空间...”开头的那一段中的最后一句（具有简短列表）替换为：

使用同一表空间来重组表将更快，但是将产生更大的记录，从而必须有足够的空间用于重组的表。如果您指定临时表空间，则通常建议您指定 SMS 临时表空间。不建议指定 DMS 临时表空间，因为在使用此类型的表空间的过程中，只能同时进行一个 REORG 操作。

在标题为『扩充内存』的一节中，将下面这一段添加到此节中的第三段后面：

当使用 DB2_AWE 注册表变量分配 Windows 2000 Address Windowing Extensions (AWE) 缓冲池时，不能使用扩充存储器高速缓存。

第 9 章 使用管理器

在标题为『创建管理器配置文件』的一节中，应该将调度操作讨论后面第一段中的第一句替换为：

如果多个规则都适用于一个应用程序，则应用所有规则。根据所设置的规则和限制，与最先遇到的规则限制相关联的操作就是首先要应用的操作。

第 13 章 配置 DB2

以下参数需要更改：

排序堆大小 (sortheap)

『建议』一节已有更改。此处的信息现应为:

使用排序堆时, 应考虑下列事项:

- 适当的索引可将排序堆的使用减至最少。
- 散列连接缓冲区和动态位图 (用于索引 “与” (AND) 和 “星形连接”) 使用排序堆内存。在使用这些技术时增加此参数的大小。
- 当经常需要进行大型排序时, 增加此参数的大小。
- ... (其余项未发生更改)

排序堆阈值 (sheapthres)

此参数描述中的倒数第二段已有更改。该段落现应为:

使用排序堆的操作的示例包括: 排序、动态位图 (用于索引 “与” (AND) 和 “星形连接”) 以及表处于内存中的那些操作。

以下信息将添加至此参数的描述:

在从单节点环境移至多节点环境时不应增加此参数的值。一旦调整了单节点 (在 DB2 EE 中) 环境中数据库和数据库管理器配置参数, 则在大多数情况下, 相同的值也适用于多节点 (在 DB2 EEE 中) 环境。

“排序堆阈值” 参数是一个数据库管理器配置参数, 它在整个 DB2 实例中都适用。在不同节点或分区上将此参数设置为不同的值的唯一方法是创建多个 DB2 实例。这将需要管理不同节点组上的不同 DB2 数据库。这样的安排会抵消分区数据库环境的许多优点。

升级前锁定列表的最大百分比 (maxlocks)

下列更改涉及 “升级前锁定列表的最大百分比 (maxlocks)” 数据库配置参数的 “建议” 部分。

建议: 以下公式允许将 *maxlocks* 设置为允许应用程序保存锁定的平均数目的两倍:

$$\text{maxlocks} = 2 * 100 / \text{maxappls}$$

其中 2 用来获取对平均数目的翻倍, 而 100 表示允许的最大百分比值。如果只有几个应用程序并行运行, 可将以下公式用作第一个公式的替代项:

$$\text{maxlocks} = 2 * 100 / (\text{并行运行的应用程序的平均数目})$$

设置 *maxlocks* 时的注意事项之一就是将其与锁定列表 (*locklist*) 的大小配合使用。在锁定升级之前, 应用程序可持有的锁定数目的实际限制为:

$$\text{maxlocks} * \text{locklist} * 4096 / (100 * 36)$$

其中 4096 是页面的字节数, 100 是 *maxlocks* 所允许的最大百分比值, 而 36 是每个锁定的字节数。如果您知道每个应用程序需要 1000 个锁定, 且不需要进行锁定升级, 则应选择此公式中 *maxlocks* 和 *locklist* 的值, 以使结果大于 1000。(对 *maxlocks* 使用 10, 而对 *locklist* 使用 100, 此公式的结果就会大于所需的 1000。)

如果 `maxlocks` 设置得太低，当仍有足够的锁定空间可供其他并行应用程序使用时，就会进行锁定升级。如果 `maxlocks` 设置得太高，少数几个应用程序可能会消耗大部分的锁定空间，而其他应用程序将不得不执行锁定升级。在此情况下的锁定升级需要会导致并行度降低。

可使用数据库系统监控器来帮助您跟踪和调整此配置参数。

配置 DB2/DB2 Data Links Manager / Data Links 访问令牌到期时间间隔 (`dl_expint`)

与文档相反，如果 `dl_expint` 设置为 `"-1"`，则访问控制令牌就会到期。此问题的解决方法是将 `dl_expint` 设置为其最大值 31536000（秒）。这对应于一年的到期时间，对于所有应用程序都应该都是比较充足的。

MIN_DEC_DIV_3 数据库配置参数

`MIN_DEC_DIV_3` 数据库配置参数的附加部分是作为启用对 SQL 中的十进制除法的小数位计算的更改的快速方法提供的。`MIN_DEC_DIV_3` 可设置为 `YES` 或 `NO`。`MIN_DEC_DIV_3` 的缺省值是 `NO`。

`MIN_DEC_DIV_3` 数据库配置参数会更改涉及除法的十进制算法操作的结果小数位。如果该值为 `NO`，则小数位取为 `31-p+s'`。参考 *SQL Reference* 的第 3 章『SQL 中的十进制算法』以获取详情。如果设置为 `YES`，则小数位取为 `MAX(3, 31-p+s')`。这会导致十进制除法结果的小数位至少为 3，精度总是为 31。

更改此数据库配置参数可能会导致更改现存数据库的应用程序。如果更改此数据库配置参数会影响十进制除法的结果小数位，就会发生这种情况。以下列出了可能影响应用程序的一些可能情况。在带有现存数据库的数据库服务器上更改 `MIN_DEC_DIV_3` 之前应考虑到这些情况。

- 如果其中一个视图列的结果小数位已经更改，则在数据库配置参数更改后引用视图（该视图是使用一个设置在环境中定义的）时，该视图可能会失效，`SQLCODE` 为 344。但是，如果对象名（第一个令牌）是一个视图，则消息 `SQL0344N` 涉及递归公共表表达式，您需要删除并重新创建该视图以避免此错误。
- 静态软件包在该软件包被重新绑定之前不会更改行为，不管是隐式地还是显式地。例如，将该值从 `NO` 更改为 `YES` 后，在重新绑定之前，附加的小数位可能不会包括在结果中。对于所有更改过的静态软件包，显式的重新绑定命令可用来强制重新绑定。
- 涉及十进制除法的检查约束可能会限制先前已接受的某些值。这种行现在违反了约束，但在涉及检查约束行的其中一列更新或带有 **IMMEDIATE CHECKED** 选项的 **SET INTEGRITY** 命令被处理之前，这些行不会被检测到。要强制检查这种约束，执行 **ALTER TABLE** 命令以删除该检查约束，然后执行 **ALTER TABLE** 命令以再次添加该约束。

注：“DB2 版本 7”还具有下列限制：

1. 命令 **GET DB CFG FOR DBNAME** 将不会显示 `MIN_DEC_DIV_3` 设置。确定当前设置的最好方法是观察十进制除法结果的负作用。例如，考虑以下语句：

```
VALUES (DEC(1,31,0)/DEC(1,31,5))
```

如果此语句返回 sqlcode SQL0419N, 则数据库不具有 MIN_DEC_DIV_3 支持或该支持被设置为 OFF。如果该语句返回 1.000, 则表明 MIN_DEC_DIV_3 被设置为 ON。

2. 当您运行以下命令时, MIN_DEC_DIV_3 不会出现在配置关键字列表中: ?
UPDATE DB CFG

应用程序控制堆大小 (app_ctl_heap_sz)

此参数的文本现在应该为:

对于分区数据库和启用了内部并行性 (intra_parallel=ON) 的非分区数据库, 这是为应用程序控制堆分配的共享内存区大小。对于禁用了内部并行性 (intra_parallel=OFF) 的非分区数据库, 这是将为堆分配的最大专用内存。每个分区的每个连接都有一个应用程序控制堆。

应用程序控制堆主要对于在为同一请求工作的代理程序之间共享信息是必需的, 并且, 在分区数据库环境中, 对于存储表示 SQL 语句的可执行部分, 应用程序控制堆也是必需的。当以小于或等于 1 的并行度运行查询时, 对于非分区数据库, 使用此堆是最小用法。

此堆还用来存储已声明临时表的描述符信息。尚未显式删除的所有已声明临时表的描述符信息被保存在此堆的内存中, 并且在删除已声明临时表之前不能删除这些信息。

“建议”部分保持不变。

数据库系统监控器堆大小 (mon_heap_sz)

具有本地客户机和远程客户机的 OS/2 和 Windows NT 数据库服务器以及具有本地客户机的“卫星”数据库服务器的缺省值已经从 24 更改为 32。范围仍然未变。

最大活动应用程序数 (maxappls)

所有平台的范围上限已经从 64 000 更改为 60 000。缺省值未变。

恢复范围和软检查点时间间隔 (softmax)

测量单位被更改为一个主日志文件大小的百分比。

跟踪已修改页启用 (trackmod)

配置类型: 数据库

参数类型: 可配置

缺省值 [范围]: Off [On; Off]

当此参数被设置为 ON 时, 数据库管理器将跟踪自从执行最新完全备份以来数据库中的哪些页已更改。这允许备份实用程序确定哪些页应该包括在增量式备份中, 而不必个别检查每一页。对于 SMS 表空间, 此跟踪的粒度是表空间级别。对于 DMS 表空间, 粒度是在数据和索引页的范围级别, 以及其他页类型的表空间级别。在将此参数设置为 ON 之后, 必须进行完全数据库备份, 以便具有一个基线, 根据它可以执行增量式备份。

更改数据库日志路径 (newlogpath)

配置类型: 数据库

参数类型: 可配置

缺省值 [范围]: Null [任何有效路径或设备]

相关参数: 日志文件的位置 (logpath); 数据库是一致的 (database_consistent)

此参数允许您指定一个可达 242 个字节的字符串来更改存储日志文件的位置。该字符串可以指向路径名或者原始设备。如果字符串指向路径名，则它必须是全限定路径名，而不是相对路径名。

注: 在分区数据库环境中，节点号被自动追加至该路径。完成此操作是为了在多个逻辑节点配置中维护路径的唯一性。

要指定设备，应指定操作系统标识为设备的字符串。例如，在 Windows NT 上，
\\.\d: or \\.\PhysicalDisk5

注: 必须安装了带有服务包 3 的 Windows NT 版本 4.0 才能将日志写入设备中。

在基于 UNIX 的平台上，
/dev/rdblog8

注: 只能在 AIX、Windows 2000、Windows NT、Solaris、HP-UX、NUMA-Q 和 Linux 平台上指定设备。

新设置不会成为 logpath 的值，直到下面两种情况都发生为止:

- 数据库处于一致状态，如 database_consistent 参数所示。
- 所有用户都与数据库断开连接。

当与数据库进行第一次新连接时，数据库管理器将把日志移到由 logpath 所指定的新位置。

在旧的日志路径中可能有日志文件。这些日志文件可能尚未归档。可能需要人工归档这些日志文件。另外，如果您正在对此数据库运行复制，则在日志路径更改之前，复制可能仍然需要日志文件。如果在“用户出口启用” (userexit) 数据库配置参数被设置为“是”的情况下配置数据库，并且如果所有日志文件已经被 DB2 自动归档或者由您自己人工归档，则 DB2 将能够检索日志文件以完成复制过程。否则，您可以将文件从旧日志路径复制到新日志路径中。

建议: 理想情况是，日志文件处于没有高 I/O 的物理磁盘上。例如，避免将日志与操作系统或大容量数据库放在同一磁盘上。这将使得记录活动很有效率而同时开销又最小，例如，等待 I/O。

可以使用数据库系统监控器来跟踪与数据库记录相关的 I/O 数。

有关详情，参考 System Monitor Guide and Reference 中的下列监控元素描述:

- log_reads (读取的日志页数)
- log_writes (写入的日志页数)

上述数据元素将返回与数据库记录相关的 I/O 活动量。可以使用操作系统监控工具来收集关于其他磁盘 I/O 活动的信息，然后比较这两种类型的 I/O 活动。

日志文件的位置 (logpath)

配置类型: 数据库

参数类型: 信息性

相关参数: 更改数据库日志路径 (newlogpath)

此参数包含用于记录目的的当前路径。不能直接更改此参数，因为它是在对 newlogpath 参数的更改生效之后由数据库管理员设置的。当创建数据库时，它的恢复日志文件是在包含数据库的目录的子目录中创建的。缺省值是在为数据库创建的目录下名为 SQLOGDIR 的子目录。

锁定列表的最大存储器 (locklist)

最大值从 60 000 增大为 524 288。

附录 A DB2 注册表和环境变量

下列注册表变量是新增的或需要更改:

新的或更改过的注册表变量表

表 6. 注册表变量

变量名 描述	操作系统	值
DB2MAXFSCRSEARCH	全部	缺省值 = 5 值: -1, 1 至 33554
指定要在将记录添加至表时搜索的空闲空间控制记录数。缺省情况是搜索五个空闲空间控制记录。修改此值可获得插入速度与空间再利用之间的平衡。使用较大的值可优化空间再利用率。使用较小的值可优化插入速度。将值设置为 -1 会强制数据库管理器搜索所有的空闲空间控制记录。		
DLFM_TSM_MGMTCLASS	AIX、Windows NT、Solaris	缺省值: 缺省 TSM 管理类 值: 任何有效的 TSM 管理类
指定要使用哪个 TSM 管理类来归档和检索链接的文件。如果未对此变量设置任何值，则使用缺省 TSM 管理类。		
DB2_CORRELATED_PREDICATES	全部	缺省值 = YES 值: YES 或 NO
此变量的缺省值为 YES。如果连接中的相关列存在唯一索引，且此注册表变量为 YES，则优化器将尝试检测并补偿连接谓词的相关性。如果此注册表变量为 YES，则优化器将使用唯一索引统计信息的 KEYCARD 信息来检测相关情况，并动态调整相关谓词的组合选择性，从而获得对连接大小和成本的更精确估计。		
DB2_VI_DEVICE	Windows NT	缺省值 = null 值: nic0 或 VINIC

表 6. 注册表变量 (续)

变量名	操作系统	值
描述		
指定与“网络接口卡”(NIC)相关联的设备或“虚拟接口提供者实例”的符号名。每个“独立硬件供应商”(IHV)都会有他们自己的NIC。每台Windows NT机器仅允许有一个NIC;同一物理机器上的多个逻辑节点将共享同一NIC。符号设备名“VINIC”必须是大写,且仅可与Synfinity Interconnect配合使用。所有其他当前受支持的实现方法都将“nic0”用作符号设备名。		
DB2_SELECTIVITY	ALL	缺省值 = NO 值: YES 或 NO
此注册表变量控制可使用SELECTIVITY子句的位置。查看 <i>SQL Reference</i> 、语言元素、搜索条件以获取有关SELECTIVITY子句的完整详细信息。		
当此注册表变量被设置为YES时,可在谓词作为基本谓词(其中至少有一个表达式包含主变量)时指定SELECTIVITY。		
DB2_UPDATE_PART_KEY	ALL	缺省值 = YES 值: YES 或 NO
对于修订包3和更新版本,缺省值为YES。此注册表变量指定是否执行分区键的更新。		
DB2_BLOCK_ON_LOG_DISK_FULL	ALL	缺省值 = NO 值: YES 或 NO
可以设置此DB2注册表变量,以防止当DB2不能在活动日志路径中创建新的日志文件时产生“磁盘已满”错误。		
然而,DB2每隔5分钟就会尝试创建日志文件,直到成功为止。在每次尝试之后,DB2就会向db2diag.log文件写入一条消息。可确认应用程序因磁盘已满而挂起的唯一方法就是监控db2diag.log文件。		
在成功地创建了日志文件之前,试图更新表数据的任何用户应用程序都将不能提交事务。只读查询可能不会受到直接影响;但是,如果查询需要访问被更新请求锁定的数据,或者是访问由更新应用程序在缓冲池中修正的数据页,则只读查询也会显得被挂起一样。		
DB2_INDEX_2BYTEVARLEN	全部	缺省值 = NO 值: YES 或 NO
此注册表变量允许将长度大于255个字节的列指定为索引键的一部分。在将此注册表变量置为YES之前已经创建的索引将仍具有255键限制。在将此注册表变量置为YES之后创建的索引将以两字节索引的形式运作,即使注册表变量再次变成NO亦如此。		
若更改此注册表变量,则有几个SQL语句将受到影响,包括CREATE TABLE、CREATE INDEX和ALTER TABLE。有关这些语句的详情,参考对 <i>SQL Reference</i> 说明的更改。		
DB2_FORCE_FCM_BP	AIX	缺省值 = NO 值: YES 或 NO
指定从何处分配“快速通信管理器”(FCM)资源。这些资源可以从数据库管理器共享内存段或者从单独的内存段分配。若同一机器上有多个逻辑节点,则应该使用此注册表变量。在启用了对称多重处理(SMP)的分区数据库系统上,此注册表变量的设置不会影响进行通信的方式。在此情况下,通信总是通过共享内存来进行的。但是,它不会影响DB2将使用的共享内存段的数目。		
DB2_AWE	Windows 2000	缺省值 = Null 值: <entry>[; <entry>;...] 其中, <entry>=<缓冲池标识>、<物理页数>、<地址窗口数>

表 6. 注册表变量 (续)

变量名	操作系统	值
描述		
允许 Windows 2000 上的 DB2 UDB 分配最多使用 64 GB 内存的缓冲池。必须正确地配置 Windows 2000 才能支持 Address Windowing Extensions (AWE) 缓冲池。这包括将 "lock pages in memory"-right 与 Windows 2000 上的用户相关联，并在 DB2 上设置此注册表变量。设置此变量时，您需要知道将用于 AWE 支持的缓冲池标识。还需要确定要分配的物理页数和地址窗口数。		
有关确定要分配的物理页数和地址窗口数的信息，参见『第 8 章 操作性能』中的『管理数据库缓冲池』一节。 注： 如果启用了 AWE 支持，则扩充存储器 (ESTORE) 不能用于数据库中的任何缓冲池。在 SYSIBM.SYSBUFFERPOOLS 中必须已经存在此变量所引用的缓冲池。		
DB2_STPROC_LOCKUP_FIRST	全部	缺省值 = NO 值: YES 或 NO
此注册表变量已经从 DB2_DARI_LOOKUP_ALL 进行了重命名。		
DB2MEMDISCLAIM	AIX	缺省值 = YES 值: YES 或 NO
在 AIX 上，DB2 进程所使用的内存可能具有某些相关联的调页空间。此调页空间可能会一直保留，即使已经释放了相关联的内存。调页空间的保留取决于 AIX 系统的可调整虚拟内存管理分配策略。此注册表变量将控制 DB2 代理程序是否显式地请求 AIX 取消所保留调页空间与已释放内存的关联。		
设置 "YES" 将导致更小的调页空间需求，且可能在调页过程中存在较少的磁盘活动。设置 "NO" 将导致更大的调页空间需求，且可能在调页过程中存在较多的磁盘活动。在某些情况下，例如，如果有大量的调页空间，且如果实际内存太多以致于从不发生调页，则设置 NO 将只提供很小的性能改进。		
DB2MEMMAXFREE	全部	缺省值 = 8 388 608 个字节 值: 0 到 $2^{32}-1$ 个字节
此注册表变量控制由 DB2 进程保留的最大未使用内存量 (以字节计)。		
DB2_ANTIJOIN	全部	在 EEE 环境中，缺省值 = NO 在非 EEE 环境中，缺省值 = YES 值: YES 或 NO
对于 "DB2 通用数据库 EEE" 环境: 当指定 YES 时，优化器将寻找机会将 NOT EXISTS 子查询转换成反连接，DB2 可以更有效地处理这种反连接情况。对于非 EEE 环境: 当指定 NO 时，优化器将限制将 NOT EXISTS 子查询转换成反连接的机会。		
NEWLOGPATH2	UNIX	缺省值 = NO 值: YES 或 NO
此参数允许您指定是否应该使用辅助路径来实现双重记录。将使用的路径是通过将字符 '2' 追加到当前值 'LOGPATH' 而生成的。		
DB2DOMAINLIST	Windows NT	缺省值 = Null 值: 一个或多个有效的 Windows NT 域 (相互之间用逗号隔开)
定义一个或多个 Windows NT 域。只有属于这些域的用户连接或附件请求才会被接受。		
此注册表变量应仅在纯 Windows NT 域环境 (而与此同时 DB2 服务器和客户机则运行 "DB2 通用数据库" 版本 7.1 (或更新版本)) 下使用。		

DB2_LIKE_VARCHAR	全部	缺省值 = Y, N
值: Y、N、S, 以及 0 到 6.2 之间的浮点常量		
控制子元素统计信息的收集和使用。这些是在数据结构呈由空格定界的一系列子字段或子元素形式时, 有关列中这些数据的内容的统计信息。		
此注册表变量将影响优化器如何处理表单的谓词:		
<pre>COLUMN LIKE '%xxxxx%'</pre>		
其中, xxxxxx 是任何字符串。		
显示如何使用此注册表变量的语法是:		
<pre>db2set DB2_LIKE_VARCHAR=[Y N S num1] [,Y N S num2]</pre>		
其中		
<ul style="list-style-type: none"> • 在逗号前面的术语（亦即谓词右边的那些术语）表示下列含义，但是仅适用于没有实际子元素统计信息的那些列： <ul style="list-style-type: none"> – S – 优化器将根据用 % 字符括起来的字符串的长度来估计一系列元素（它们并置在一起而形成一列）中每个元素的长度。 – Y – 这是缺省值。对算法参数使用缺省值 1.9。将变长子元素算法与算法参数配合使用。 – N – 使用定长子元素算法。 – num1 – 将作为算法参数的 num1 的值与变长子元素算法配合使用。 • 逗号后面的术语的含义如下： <ul style="list-style-type: none"> – N – 这是缺省值。不收集和使用子元素统计信息。 – Y – 收集子元素统计信息。对于具有实际子元素统计信息的列，对算法参数使用变长子元素算法，该算法将所收集的统计信息与缺省值 1.9 一起使用。 – num2 – 收集子元素统计信息。对于具有实际子元素统计信息的列，使用变长子元素算法，该算法将所收集的统计信息与值 num2 一起作为算法参数。 		
DB2_PINNED_BP	AIX, HP-UX	缺省值 = NO
值: YES 或 NO		
在某些 AIX 操作系统上, 此变量用来保存与主内存中的数据库相关联的数据库全局内存 (包括缓冲池)。将此数据库全局内存保存在系统主内存中将会使得数据库性能更加一致。		
例如, 如果缓冲池被交换出了系统主内存, 则数据库性能将明显降低。通过使系统内存中存在缓冲池来减少磁盘 I/O 将提高数据库性能。如果有其他应用程序需要更多主内存, 则您将想使数据库全局内存被交换出主内存, 这取决于系统主内存需求。		
当在 64 位环境中使用 HP-UX 时, 除了修改此注册表变量之外, 还必须授予 DB2 实例组以 MLOCK 特权。这是通过让具有 root 用户访问权的用户执行下列操作来完成的:		
<ol style="list-style-type: none"> 1. 将 DB2 实例组添加至 /etc/privgroup 文件。例如, 如果 DB2 实例组属于 db2iadm1 组, 则必须将下面这一行添加到 /etc/privgroup 文件中: db2iadm1 MLOCK 2. 发出以下命令: setprivgrp -f /etc/privgroup 		
DB2_RR_TO_RS	全部	缺省值 = NO
值: YES 或 NO		

下一个键锁定通过自动锁定所有 INSERT 和 DELETE 语句的下一个键及高于 SELECT 语句的结果集的下一个键值来保证“可重复读”(RR)隔离级别。对于改变索引的键部件的 UPDATE 语句，原始索引键被删除，而将插入新的键值。对键插入和键删除都执行了下一个键锁定。下一个键锁定是保证 ANSI 和 SQL92 标准 RR 所必需的，并且它是 DB2 缺省值。

如果您的应用程序似乎要停止或挂起，则应该检查应用程序的快照信息。如果是下一个键锁定有问题，则可以根据两个条件来将 DB2_RR_TO_RS 注册表变量设置为开。如果没有任何应用程序依赖“可重复读”(RR)行为，并且，如果进行扫描以便跳过未提交的删除是可行的，则可以打开 DB2_RR_TO_RS。跳过的行为会影响 RR、“读稳定性”(RS)和“游标稳定性”(CS)隔离级别。(对“未提交的读”(UR)隔离级别没有行锁定。)

当 DB2_RR_TO_RS 打开时，不能保证 RR 行为对用户表进行扫描，因为索引键插入和删除期间未执行下一个键锁定。目录表不会受此选项的影响。

行为中的其他更改是打开 DB2_RR_TO_RS 时，扫描将跳过已删除但是未提交的行，尽管该行可能已经对扫描进行了限定。

附录 C SQL 说明工具

『运行 db2expln 和 dynexpln』一节应将最后一段替换为以下内容：

要运行 db2expln，必须对系统目录视图具有 SELECT 特权，且对 db2expln 软件包具有 EXECUTE 权限。要运行 dynexpln，必须对数据库具有 BINDADD 权限，且用来连接数据库的模式必须存在，或您必须对数据库具有 EXPLICIT_SCHEMA 权限，且必须具有正在解释的 SQL 语句所需的任何特权。(注意，如果您具有 SYSADM 或 DBADM 权限，则您将自动拥有上述所有权限级别。)

卫星管理指南和参考

将版本 7.2 DB2 个人版和 DB2 工作组版设置为卫星

接下来的几节描述如何设置基于 Windows 的版本 7.2 的“DB2 个人版”和“DB2 工作组版”系统，以便它们可在卫星环境中用作全功能的卫星。有关接下来的信息中使用的术语和概念，参考《卫星管理指南和参考》。可在以下 URL 处找到本书：

http://www-4.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v6pubs.d2w/en_main

有关补充《卫星管理指南和参考》中的信息的技术注释，参考以下 URL：

<http://www-4.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/browse.d2w/report?type=tech5udb&tech5udb=Y>

先决条件

要将“DB2 个人版”或“DB2 工作组版”设置为卫星，需要下列各项：

1. DB2 控制服务器

DB2 控制服务器是在 Windows NT 或 AIX 上运行的“DB2 企业版”系统，且安装有“控制服务器”组件。使用的“DB2 企业版”系统必须是带有“修订包 2”或更高版本的版本 6，或处于任何“修订包”级别的版本 7。

- 如果具有想要用作 DB2 控制服务器的“版本 6 企业版”系统，则参见第104页的『在版本 6 企业版系统上安装修订包 2 或更高版本』。
- 如果正在使用版本 7，且未安装“控制服务器”组件，则安装此组件，重新安装已安装的所有“修订包”，然后创建 DB2 控制服务器实例和卫星控制数据库。参考《卫星管理指南和参考》以获取有关创建这些对象的指导。

注：如果在 Windows NT 上安装“版本 7.2 企业版”系统以用作 DB2 控制服务器，且想要执行响应文件安装，则参考技术注释 *DB2 控制服务器响应文件关键字*，以获取有关要在响应文件中指定的关键字的信息。

2. DB2 控制服务器实例和卫星控制数据库

通常，DB2 控制服务器实例称为 DB2CTLSV，而卫星控制数据库称为 SATCTLDB。DB2 控制服务器实例和卫星控制数据库在“企业版”系统上，而且，它们是您在安装 DB2 和“控制服务器”组件时在 Windows NT 上自动创建的。如果在 AIX 上安装 DB2，则参见《卫星管理指南和参考》以获取有关创建 DB2 控制服务器实例和卫星控制数据库的信息。

3. 卫星管理中心

卫星管理中心是一组 GUI 工具，可用于设置和管理卫星环境。从控制中心访问这组工具。有关卫星管理中心和卫星环境的详情，参见《卫星管理指南和参考》和卫星管理中心提供的联机帮助。如果正在运行“版本 6 控制中心”，参见第110页的『升级版本 6 控制中心和卫星管理中心』。

如果还未使用卫星管理中心来设置卫星环境，也未创建在卫星管理中心中表示新卫星的对象，则应在安装卫星之前完成这些操作。有关详情，参见《卫星管理指南和参考》中有关如何设置和测试卫星环境的描述。

4. 想要用作卫星的版本 7.2 个人版或工作组版系统。

安装说明

在安装“DB2 个人版”或“DB2 工作组版”时，不需要选择任何特殊组件就可以启用要同步的任何一个系统。如果打算执行响应文件安装，则参见『执行响应文件安装』以获取在安装版本 7.2 系统时应指定的关键字。如果正在版本 7.2 系统上执行交互式安装，则参见第103页的『配置版本 7.2 系统以进行同步』以获取在完成 DB2 的安装后版本 7.2 上必须设置的值，以使系统同步。

执行响应文件安装：如果是执行版本 7.2 的“DB2 个人版”或“DB2 工作组版”的响应文件安装，可在响应文件中设置下列关键字。

如果决定在响应文件安装期间不指定这些关键字中的一个或多个关键字，则参见第103页的『配置版本 7.2 系统以进行同步』以获取在安装 DB2 后必须执行的其他步骤，以使版本 7.2 系统同步。如果想要更改在响应文件安装期间指定的任何值，还可使用本节中的指导。

db2.db2satelliteid

在系统上设置卫星标识。

注：如果未指定此关键字，则卫星标识被自动设置为用来安装 DB2 的用户标识。如果想要将此用户标识用作卫星标识，则不必对此关键字指定值。

db2.db2satelliteappver

在系统上设置应用程序版本。

注：如果未指定此关键字，则卫星上的应用程序版本被自动指定为 V1R0M00。如果想要将此值用作应用程序版本，则不必对此关键字指定值。

db2.satctldb_username

设置要用于系统的，用来连接至卫星控制数据库的用户名。

db2.satctldb_password

设置在使用用户名连接至卫星控制数据库时，该用户名传送至 DB2 控制服务器的口令。

完成响应文件安装后，版本 7.2 系统就可以同步了。应在卫星上发出 **db2sync -t** 命令以验证在卫星上指定的值是否正确，且卫星是否能连接至卫星控制数据库。

有关执行响应文件安装的详情，参考《卫星管理指南和参考》。

注：

1. 在版本 7 中，用户标识和口令是在 Windows NT 和 Windows 2000 上创建所有服务必需的。这些用户标识和口令是在响应文件中按关键字对指定的。响应文件中发现的第一个关键字对会成为所有服务的缺省用户标识和口令，除非通过对该服务指定特定关键字对来为该服务重设关键字。

在版本 6 中，可在“DB2 卫星版”的响应文件安装期间指定 **admin.userid** 和 **admin.password** 关键字以指定将由“远程命令服务”使用的用户标识和口令。对于版本 7.2 的“个人版”和“工作组版”，如果指定这些关键字，则它们将用于版本 7.2 系统上的 DB2DAS00 实例。对于 DB2 版本 7.2 系统，“远程命令服务”将使用由系统上的 DB2 实例使用的用户标识和口令。如果未指定 **db2.userid** 和 **db2.password** 的值，则应用上述缺省规则。

2. 在版本 6 中，在安装“DB2 卫星版”时可能使用响应文件安装创建的数据库。不能在打算用作卫星的版本 7.2 的“个人版”或“工作组版”系统上进行响应文件安装时创建数据库。下列关键字（在《卫星管理指南和参考》中作了描述）是不受支持的：

- **db2.userdb_name**
- **db2.userdb_recoverable**
- **db2.userdb_rep_src**

配置版本 7.2 系统以进行同步

如果以交互方式安装版本 7.2 系统，则在安装 DB2 之后必须在“DB2 个人版”或“DB2 工作组版”系统上设置几个值，系统才能同步。

注：还可在系统上使用执行操作系统脚本以在卫星上设置所有的值，卫星用来连接至卫星控制数据库的用户标识和口令除外（参见步骤4）。

1. 通过使用 **db2set** 命令来设置卫星标识。

如果以交互方式安装“DB2 个人版”或“DB2 工作组版”，则卫星标识被自动设置为用来安装 DB2 的用户标识。如果想要将此用户标识用作卫星标识，则不必执行此步骤。有关设置卫星标识的信息，参见《卫星管理指南和参考》。

2. 使用 **db2sync -s** 命令在卫星上设置应用程序版本。

如果以交互方式安装“DB2 个人版”或“DB2 工作组版”，则卫星上的应用程序版本被自动设置为 V1R0M00。如果想要将此值用作应用程序版本，则不必执行此步骤。

可在卫星上使用 **db2sync -g** 命令以查看应用程序版本的当前设置。如果想要更改此值，则发出 **db2sync -s** 命令。系统会提示您对应用程序版本输入新值。有关设置应用程序版本的详情，参见《卫星管理指南和参考》。

3. 在卫星上发出**编目节点和编目数据库**命令，以在卫星上对 DB2 控制服务器实例和卫星控制数据库 SATCTLDB 进行编目。

还可在卫星上使用 **db2sync -t** 命令来以测试方式打开 DB2 Synchronizer 应用程序。如果在发出命令时还未在卫星上编目 SATCTLDB 数据库，则“编目控制数据库”窗口打开。可使用“编目控制数据库”窗口中提供的 DB2 discovery 功能部件来编目 DB2 控制服务器和 SATCTLDB 数据库，可在此窗口中输入主机名和服务器名。系统还会提示您指定卫星将用来连接至卫星控制数据库的用户标识和口令，如步骤4中所述。

注：在以交互方式安装了版本 7.2 的“DB2 个人版”或“DB2 工作组版”后，DB2 Synchronizer 不会以测试方式自动启动（就象对版本 6 DB2 卫星版一样）。

4. 在卫星上发出 **db2sync -t** 命令，以：

- 指定卫星将用来连接至卫星控制数据库的用户标识和口令

如果还未在卫星上存储同步凭证，则“连接至控制数据库”窗口打开。必须使用此窗口来指定卫星将用来连接至卫星控制数据库的用户标识和口令。

- 验证在卫星上设置的值是否正确
- 验证卫星是否能连接至卫星控制数据库

完成这些配置任务后，版本 7.2 系统就可以同步了。

在版本 6 企业版系统上安装修订包 2 或更高版本

接下来的几节描述您必须执行的任务，以在 Windows NT 或 AIX 上升级“版本 6 企业版”系统，以使用作 DB2 控制服务器。如果正在使用版本 6 控制中心，则还应执行第110页的『升级版本 6 控制中心和卫星管理中心』中的步骤以验证是否有正确级别的控制中心和卫星管理中心来管理卫星环境。

升级版本 6 DB2 企业版以用作 DB2 控制服务器

要将“版本 6 DB2 企业版”系统用作 DB2 控制服务器，必须将其与“控制服务器”组件安装在一起，且“DB2 企业版”应为“修订包 2”服务级别或更高级别。根据是否安装了 DB2 控制服务器组件及“DB2 企业版”的服务级别，您将必须执行下列其中一个任务：

- 将 DB2 控制服务器组件安装至现存“DB2 企业版 V6.1”系统并安装“修订包 2”或更高版本。然后在系统上更新卫星控制数据库 (SATCTLDB)。
- 将已安装的 DB2 控制服务器升级至“修订包 2”级别或更高级别。

使用下面的信息标识需要执行前面两个任务中的哪一个，以及适用于您的情况的步骤。下面是将要执行的步骤的总结。

1. 首先，访问“DB2 企业版”安装的当前状态。您将确定是否安装了“控制服务器”组件，以及 DB2 的服务级别。
2. 接着，根据您获得的状态信息，确定需要执行的操作。
3. 然后，执行升级“DB2 企业版”所需的步骤。

DB2 控制服务器只能在 DB2 企业版的 Windows NT 和 AIX 版上运行。继续遵循适用于您的平台的指导：

- 『在 Windows NT 上升级 DB2 企业版』
- 第107页的『在 AIX 上升级 DB2 企业版』

在 Windows NT 上升级 DB2 企业版： 使用以下几节中的信息来确定“版本 6 DB2 企业版”系统的当前服务级别，以及将系统更新至“修订包 2”服务级别或更高级别需要执行的步骤。您将需要执行下面一节或多节中的步骤：

- 『在 Windows NT 上评估 DB2 企业版』
- 第105页的『确定需要完成的操作』
- 第105页的『在 Windows NT 上安装控制服务器组件』
- 第106页的『在 Windows NT 上安装修订包 2 或更高版本』
- 第106页的『在 Windows NT 上升级 SATCTLDB』

在 Windows NT 上评估 DB2 企业版： 如果在 Windows NT 上安装了“DB2 企业版”，则执行下列步骤：

1. 检查是否安装了“控制服务器”组件。使用“注册表编辑器”来显示已安装的组件的列表：
 - a. 在命令提示符处输入 regedit。
 - b. 在 HKEY_LOCAL_MACHINE\SOFTWARE\IBM\DB2\Components 注册表键下，检查是否列出了“控制服务器”。如果未列出它的话，则表明未安装控制服务器。
2. 确定“DB2 企业版”的服务级别。从命令提示符处发出 db2level 命令。使用下表来解释输出：

db2level 输出中的键字段的值			DB2 系统为:
发行版	级别	信息标记	
SQL06010	01010104	db2_v6, n990616	版本 6.1 基本
SQL06010	01020104	DB2 V6.1.0.1, n990824, WR21136	版本 6.1, 修订包 1
SQL06010	01030104	DB2 V6.1.0.6, s991030, WR21163 或 DB2 V6.1.0.9, s000101, WR21173	版本 6.1, 修订包 2

注: 如果级别高于 01030104, 则表明系统的“修订包”级别比“修订包 2”级别要高。

3. 记下找到的信息, 在『确定需要完成的操作』处继续。

确定需要完成的操作: 使用您收集的信息查找下表中适用于您的情况的行, 并遵循准备“DB2 企业版”系统以支持“修订包 2”级别或更高级别的 DB2 控制服务器所需的步骤。

该表下面的几节提供了有关执行所需步骤的指导。考虑在执行每一步时核对一下。只执行那些适用于您的情况的步骤。

安装的控制服务器组件	DB2 企业版系统的服务级别	准备“DB2 企业版”系统所需的步骤
否	版本 6.1 基本或版本 6.1, 修订包 1, 或版本 6.1, 修订包 2 或更高级别	执行下列步骤: 1. 『在 Windows NT 上安装控制服务器组件』 2. 第106页的『在 Windows NT 上安装修订包 2 或更高版本』 3. 第106页的『在 Windows NT 上升级 SATCTLDB』
是	版本 6.1 基本, 或版本 6.1, 修订包 1	执行下列步骤: 1. 第106页的『在 Windows NT 上安装修订包 2 或更高版本』 2. 第106页的『在 Windows NT 上升级 SATCTLDB』
是	版本 6.1, 修订包 2 或更高级别	执行以下步骤: 1. 第106页的『在 Windows NT 上升级 SATCTLDB』

在 Windows NT 上安装控制服务器组件: 要在 Windows NT 上安装“控制服务器”组件:

1. 确保系统上的所有数据库活动完成后才继续进行。
2. 将“DB2 通用数据库企业版 6.1 CD”插入 CD 驱动器。
如果安装程序未自动启动, 在 CD 的根目录中运行 `setup` 命令以启动安装进程。
3. 提示时关闭使用 DB2 的所有进程。
4. 在“欢迎”窗口上, 选择下一步。
5. 在“选择产品”窗口上, 确保选择了“DB2 企业版”。
6. 在“选择安装类型”面板上, 单击**定制**。
7. 在“选择组件”面板上, 确保选择了“控制服务器”组件, 并单击**下一步**。

注：如果选择还未安装在系统上的其他组件，则还应安装这些组件。不能改变安装 DB2 的驱动器或目录。

8. 在“配置 DB2 服务”面板上，可修改协议值和“控制服务器”实例的启动选项，或采用缺省值。修改缺省值并单击下一步，或单击下一步以使用缺省值。
9. 在“开始复制文件”窗口中单击下一步以开始安装进程。
10. 文件复制进程完成时，就可选择重新引导系统。应立刻重新引导。对系统所作的有关“控制服务器”的更改在系统重新引导之前不会生效。

如果安装进程完成且已经重新引导了系统，则在想要在系统本地使用控制中心和卫星管理中心的情况下，必须在 DB2 实例中编目作为“控制服务器”安装的一部分创建的卫星控制数据库 (SATCTLDB)。要编目 SATCTLDB 数据库：

1. 通过选择开始 > 程序 > **DB2 Windows NT 版** > 命令窗口来打开“DB2 命令窗口”
2. 确保您已经在 DB2 实例中。
发出 set 命令并检查 db2instance 的值。如果值不是 db2，则发出以下命令：

```
set db2instance=db2
```
3. 输入以下命令以编目 db2ctlsv 实例：

```
db2 catalog local node db2ctlsv instance db2ctlsv
```
4. 输入以下命令以编目 SATCTLDB 数据库

```
db2 catalog database satctldb at node db2ctlsv
```
5. 输入以下命令以提交编目操作：

```
db2 terminate
```
6. 关闭“DB2 命令窗口”。

在 Windows NT 上安装修订包 2 或更高版本：要将 Windows NT 上的现存“版本 6 DB2 企业版”系统升级至“修订包 2”或更高版本，则：

- 从 Web 上下载“DB2 企业版 Windows NT 版 V6.1”的最新“修订包”及其附带的自述文件。可遵循以下 URL 中的指导来下载“修订包”：

<http://www-4.ibm.com/software/data/db2/db2tech/version61.html>

遵循 readme.txt 文件中的指导来安装“修订包”。

- 使用处于“修订包 2”级别或更高级别的“DB2 通用数据库版本 6.1 修订包 Windows NT 版 CD”，并遵循 CD 上 WINNT95 目录中的 readme.txt 文件中的指导来完成安装。

在 Windows NT 上升级 SATCTLDB：要在 Windows NT 上升级 SATCTLDB 数据库

1. 确定 SATCTLDB 数据库的级别：
 - a. 使用对 Windows NT 系统具有本地管理权限的用户标识登录。
 - b. 选择开始 > 程序 > **DB2 Windows NT 版** > 命令窗口来打开“DB2 命令窗口”。
 - c. 输入以下命令以连接至 SATCTLDB

```
db2 connect to satctldb
```
 - d. 发出以下查询以确定数据库中是否存在触发器 I_BATCHSTEP_TRGSCR：

```
db2 select name from sysibm.systriggers where name='I_BATCHSTEP_TRGSCR'
```

记下返回的行数。

- e. 输入以下命令以关闭与数据库的连接:

```
db2 connect reset
```

如果步骤第106页的1d返回一行，则表明数据库处于正确级别。在此情况中，跳过步骤2，并在步骤3处继续。如果返回零 (0) 行，则表明数据库的级别不正确，必须升级（如步骤2中所述）才能执行步骤3。

2. 要升级 SATCTLDB 数据库，执行下列步骤。在“DB2 命令窗口”中输入所有命令:

- a. 切换至目录 <db2path>\misc，其中 <db2path> 是安装驱动器和路径，例如 c:\sql1lib。

- b. 确保您在 db2ct1sv 实例中。

发出 set 命令并检查 db2instance 的值。如果值不是 db2ct1sv，则发出以下命令:

```
set db2instance=db2ct1sv
```

- c. 输入以下命令以删除 SATCTLDB 数据库:

```
db2 drop database satctldb
```

- d. 输入以下命令创建新的 SATCTLDB 数据库:

```
db2 -tf satctldb.dd1 -z satctldb.log
```

- e. 发出以下命令:

```
db2 terminate
```

3. 将 db2satcs.dll 存储过程绑定至 SATCTLDB 数据库。执行下列步骤:

- a. 输入以下命令以连接至 SATCTLDB 数据库

```
db2 connect to satctldb
```

- b. 切换至目录 <db2path>\bnd，其中 <db2path> 是安装驱动器和路径，例如 c:\sql1lib。

- c. 发出绑定命令，如下所示:

```
db2 bind db2satcs.bnd
```

4. 输入以下命令以关闭与数据库的连接:

```
db2 connect reset
```

5. 关闭“DB2 命令窗口”。

在 AIX 上升级 DB2 企业版: 使用以下几节中的信息来确定“版本 6 DB2 企业版”系统的当前服务级别，以及将系统更新至“修订包 2”服务级别或更高级别需要执行的步骤。您将需要执行下面一节或多节中的步骤:

- 『在 AIX 上评估 DB2 企业版』
- 第108页的『确定需要完成的操作』
- 第108页的『在 AIX 上安装“控制服务器”组件』
- 第109页的『在 AIX 上安装修订包 2 或更高版本』
- 第109页的『在 AIX 上升级 SATCTLDB 数据库』

在 AIX 上评估 DB2 企业版: 如果在 AIX 上安装了“版本 6 DB2 企业版”，则执行下列步骤:

1. 检查是否安装了“控制服务器”组件。输入以下命令:

```
ls1pp -l | grep db2_06_01.ctsr
```

如果未返回任何数据，则表明未安装“控制服务器”组件。

2. 确定“DB2 企业版”的服务级别。作为 DB2 实例所有者登录，并发出 db2level 命令。使用下表来解释输出：

db2level 输出中的关键字段的值			DB2 系统为:
发行版	级别	信息标记	
SQL06010	01010104	db2_v6, n990616	版本 6.1 基本
SQL06010	01020104	DB2 V6.1.0.1, n990824, U465423	版本 6.1, 修订包 1
SQL06010	01030104	DB2 V6.1.0.6, s991030, U468276 或 DB2 V6.1.0.9, s000101, U469453	版本 6.1, 修订包 2

注：如果级别高于 01030104，则表明系统的“修订包”级别比“修订包 2”级别要高。

3. 记下找到的信息，在『确定需要完成的操作』处继续。

确定需要完成的操作：使用您收集的信息查找下表中适用于您的情况的行，并遵循准备“版本 6 DB2 企业版”系统以支持“修订包 2”级别的 DB2 控制服务器所需的步骤。

该表下面的几节提供了有关执行所需步骤的指导。考虑在执行每一步时核对一下。只执行那些适用于您的情况的步骤。

安装的控制服务器组件	DB2 企业版系统的服务级别	准备“DB2 企业版”系统所需的步骤
否	版本 6.1 基本或版本 6.1, 修订包 1, 或版本 6.1, 修订包 2 或更高级别	执行下列步骤： 1. 『在 AIX 上安装“控制服务器”组件』 2. 第109页的『在 AIX 上安装修订包 2 或更高版本』 3. 第 109 页的『在 AIX 上升级 SATCTLDB 数据库』
是	版本 6.1 基本, 或版本 6.1, 修订包 1	执行下列步骤： 1. 第109页的『在 AIX 上安装修订包 2 或更高版本』 2. 第 109 页的『在 AIX 上升级 SATCTLDB 数据库』
是	版本 6.1, 修订包 2 或更高级别	执行以下步骤： 1. 第 109 页的『在 AIX 上升级 SATCTLDB 数据库』

在 AIX 上安装“控制服务器”组件：要在 AIX 上安装“控制服务器”组件

1. 作为具有 root 用户权限的用户登录。
2. 将“DB2 通用数据库企业版 6.1 CD”插入 CD 驱动器。
3. 切换至安装有该 CD 的目录，例如 cd/cdrom。
4. 输入以下命令以启动 DB2 安装程序：

```
./db2setup
```

5. “DB2 安装程序” 窗口打开时，使用 Tab 键选择**安装**选项，并按 Enter 键。
6. 定位“企业版”行并使用 Tab 键来选择它旁边的**定制**选项。按 Enter 键。
7. 选择“DB2 控制服务器”组件，按“确定”，并按 Enter 键。
8. 遵循对其余窗口的指导以完成“DB2 控制服务器”组件的安装。

安装进程完成时，创建 DB2CTLSV 实例和 SATCTLDB 数据库。要执行这些任务，遵循《卫星管理指南和参考》中第 13 章的『在 AIX 上设置 DB2 控制服务器』中的详细指导。

在 AIX 上安装修订包 2 或更高版本：要将 AIX 上现存的“DB2 企业版”系统升级至“修订包 2”或更高版本，则：

- 从 Web 上下载“DB2 企业版 AIX 版 V6.1”的最新“修订包”及其附带的 Fixpak.Readme。可遵循以下 URL 中的指导来下载“修订包”：

<http://www-4.ibm.com/software/data/db2/db2tech/version61.html>

遵循 Fixpak.Readme 文件中的指导来安装“修订包”。

- 使用处于“修订包 2”级别或更高级别的“DB2 通用数据库版本 6.1 修订包 AIX 版 CD”，并遵循 CD 上自述文件目录中的指导来完成安装。

确保您已经按 FixPak.Readme 文件中的指导运行了 db2iupdt 命令，更新了 DB2CTLSV 实例。

在 AIX 上升级 SATCTLDB 数据库：要在 AIX 上升级 SATCTLDB 数据库：

1. 确定 SATCTLDB 数据库的级别：

- a. 作为 db2ctlsv 登录。
- b. 确保已经启动了数据库服务器。如果未启动服务器，则发出 db2start 命令。
- c. 输入以下命令以连接至 SATCTLDB 数据库：

```
db2 connect to satctldb
```

- d. 发出以下查询以确定数据库中是否存在触发器 I_BATCHSTEP_TRGSCR：

```
db2 "select name from sysibm.systriggers where name='I_BATCHSTEP_TRGSCR'"
```

记下返回的行数。

- e. 输入以下命令以关闭与数据库的连接：

```
db2 connect reset
```

如果步骤 1d 返回一行，则表明数据库处于正确级别。在此情况下，跳过步骤 2，并在步骤第 110 页的 3 处继续。如果返回零 (0) 行，则表明数据库的级别不正确，必须升级（如步骤 2 中所述）才能执行步骤第 110 页的 3。

2. 要将 SATCTLDB 数据库升级至“修订包 2”级别，则执行下列步骤。在“DB2 命令窗口”中输入所有命令：

- a. 切换至 \$HOME/sql1lib/misc 目录。
- b. 输入以下命令以删除 SATCTLDB 数据库：

```
db2 drop database satctldb
```

- c. 输入以下命令创建新的 SATCTLDB 数据库：

```
db2 -tf satctldb.dd1 -z $HOME/satctldb.log
```

- d. 发出以下命令：

```
db2 terminate
```

3. 将 db2satcs.dll 存储过程绑定至 SATCTLDB 数据库。执行下列步骤:

a. 输入以下命令以连接至 SATCTLDB 数据库

```
db2 connect to satctldb
```

b. 切换至目录 \$HOME/sql1lib/bnd。

c. 发出绑定命令，如下所示:

```
db2 bind db2satcs.bnd
```

4. 输入以下命令以关闭与数据库的连接:

```
db2 connect reset
```

升级版本 6 控制中心和卫星管理中心

要将版本 6 控制中心和卫星管理中心与已升级至“修订包 2”或更高版本的版本 6 DB2 控制服务器和卫星控制数据库 (SATCTLDB) 配合使用，这些工具必须也升级至“修订包 2”或更高版本。

如果控制中心和卫星管理中心是在 DB2 控制服务器所在的系统上运行，则它们会在“DB2 企业版”系统升级至“修订包 2”时进行升级。但是，如果在另一系统上运行这些工具，则必须将此系统升级至“修订包 2”级别或更高级别。

要将此系统升级至“修订包 2”或更高版本:

- 从 Web 上下载位于 V6.1 级别的产品最新“修订包”及其附带的自述文件。可遵循以下 URL 中的指导来下载“修订包”:

```
http://www-4.ibm.com/software/data/db2/db2tech/version61.html
```

遵循自述文件中的指定来安装“修订包”。

- 对正在运行的位于“修订包 2”级别或更高级别的操作系统使用“DB2 通用数据库版本 6.1 修订包 CD”，并遵循自述文件中的指导来完成安装。

Command Reference

db2batch - Benchmark Tool

PERF_DETAIL 参数的描述中的最后一句应为:

A value greater than 1 is only valid on DB2 Version 2 and DB2 UDB servers,
and is not currently supported on host machines.

db2cap (新命令)

db2cap - CLI/ODBC 静态软件包绑定工具

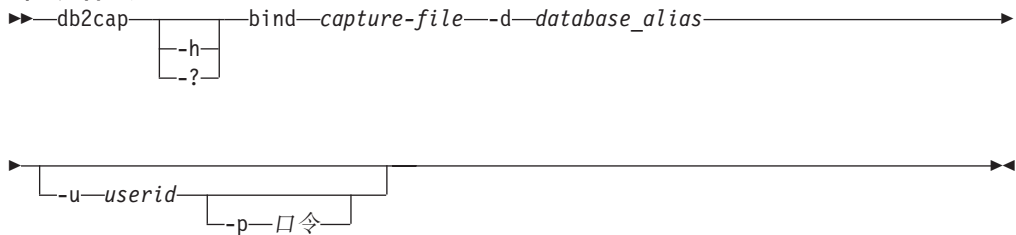
绑定捕捉文件以生成一个或多个静态软件包。捕捉文件是在创建 CLI/ODBC/JDBC 应用程序的静态简要表会话期间生成的，且包含在应用程序运行期间捕捉到的 SQL 语句。此实用程序会处理捕捉文件，以使 CLI/ODBC/JDBC 驱动程序可以用它来执行应用程序的静态 SQL。

有关在 CLI/ODBC/JDBC 应用程序中如何使用静态 SQL 的详情，参见 *CLI Guide and Reference* 中的“创建静态简要表”功能部件。

权限

- 对由记录在捕捉文件中的 SQL 语句引用的任何数据库对象的访问特权。
- 设置绑定选项（如 OWNER 和 QUALIFIER）的足够权限（如果这些选项不同于用来调用 **db2cap** 命令的连接标识的话）。
- BINDADD 权限（如果是第一次绑定软件包的话）；否则，需要 BIND 权限。

命令语法



命令参数

-h/-? 显示命令语法的帮助文本。

bind capture-file

绑定捕捉文件中的语句并创建一个或多个软件包。

-d database_alias

对将包含一个或多个软件包的数据库指定数据库别名。

-u userid

指定要用来连接至数据源的用户标识。

注：如果未指定用户标识，则从系统获取可信赖授权标识。

-p password

指定要用来连接至数据源的口令。

使用说明

在 UNIX 平台上，该命令必须以小写形式输入，但在 Windows 操作系统和 OS/2 上可以小写或大写形式输入。

此实用程序支持一定数量的用户指定绑定选项，这些选项可在捕捉文件中找到。由于性能和安全性原因，可使用文本编辑器来检查和编辑该文件以更改这些选项。

SQLERROR(CONTINUE) 和 VALIDATE(RUN) 绑定选项可用来创建软件包。

在使用此实用程序来创建软件包时，必须禁止创建静态简要表。

创建的软件包数目取决于捕捉文件中记录的 SQL 语句所使用的隔离级别。软件包名最多可以包括捕捉文件的软件包关键字的前七个字符和下列其中一个单个字符前缀：

- 0 - 未提交的读取 (UR)
- 1 - 游标稳定性 (CS)
- 2 - 读稳定性 (RS)
- 3 - 可重复读 (RR)
- 4 - 未提交 (NC)

要获得有关软件包的特定信息，用户可：

- 使用捕捉文件中的 COLLECTION 和 PACKAGE 关键字来查询适当的 SYSIBM 目录表。
- 查看捕捉文件。

db2ckrst (新命令)

db2ckrst - 检查增量恢复映象序列

查询数据库历史并生成增量恢复所需的备份映象的时间戳记列表。还会生成人工增量恢复的简化恢复语法。

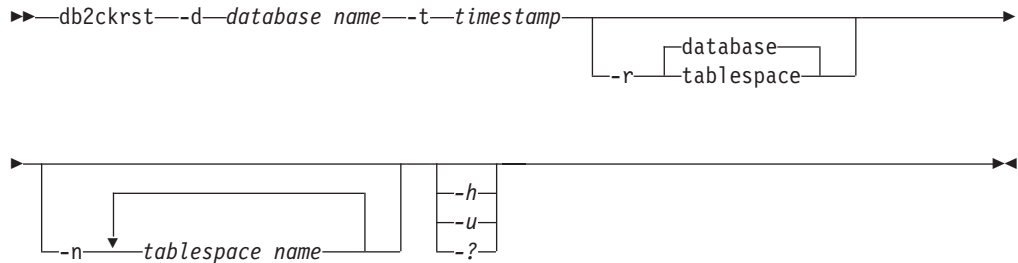
权限

无

必需的连接

无

命令语法



命令参数

-d database name

指定将要恢复的数据库的别名。

-t timestamp

指定将要进行增量恢复的备份映象的时间戳记。

-r 指定将要执行的恢复的类型。缺省值为数据库。

注：如果选择了表空间，且未给定任何表空间名，则实用程序会检查指定映象的历史条目，并使用列出的表空间名来执行恢复。

-n tablespace name

指定将要恢复的一个或多个表空间的名称。

注：如果选择了数据库恢复类型，且指定了表空间名称列表，则实用程序将会作为使用给定表空间名称的表空间恢复继续进行。

-h/-u/-?

显示帮助信息。指定此选项时，会忽略所有其他选项，而只显示帮助信息。

示例

```
db2ckrst -d mr -t 20001015193455 -r database
db2ckrst -d mr -t 20001015193455 -r tablespace
db2ckrst -d mr -t 20001015193455 -r tablespace -n tbspl tbsp2
```

```
> db2 backup db mr
```

```
Backup successful. The timestamp for this backup image is : 20001016001426
```

```
> db2 backup db mr incremental
```

```
Backup successful. The timestamp for this backup image is : 20001016001445
```

```

> db2ckrst -d mr -t 20001016001445

Suggested restore order of images using timestamp 20001016001445 for database mr.
=====
db2 restore db mr incremental taken at 20001016001445
db2 restore db mr incremental taken at 20001016001426
db2 restore db mr incremental taken at 20001016001445
=====

> db2ckrst -d mr -t 20001016001445 -r tablespace -n userspace1
Suggested restore order of images using timestamp 20001016001445 for database mr.
=====
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001445
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001426
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001445
=====

```

使用说明

数据库历史必须存在，才能使用此实用程序。如果数据库历史不存在，则在 RESTORE 命令中指定 HISTORY FILE 选项，然后才使用此实用程序。

如果使用了 PRUNE HISTORY 命令的 FORCE 选项，有可能会删除从最新的完整数据库备份映象进行恢复所需的条目。PRUNE HISTORY 命令的缺省操作会阻止删除所需的条目。建议不要使用 PRUNE HISTORY 命令的 FORCE 选项。

建议保存好备份记录，并将此实用程序用作指南。

db2gncol (新命令)

db2gncol - 更新生成的列值

更新处于检查暂挂方式且日志空间有限的表中的生成列。此工具用于在表（该表具有由表达式生成的列）上准备 SET INTEGRITY 语句。

权限

下列其中一项:

- *sysadm*
- *dbadm*

命令语法

```
▶▶ db2gncol -d database -s schema_name -t table_name -c commit_count ▶▶  
  
┌-u userid -p 口令┐ ┌-h┐  
└──────────────────┘ └──┘
```

命令参数

-d *database*

指定该表所在的数据库的别名。

-s *schema_name*

指定该表的模式名。模式名是区分大小写的。

-t *table_name*

指定要对其计算由表达式生成的新列值的表。表名是区分大小写的。

-c *commit_count*

指定在各提交间更新的行数。此参数会影响生成列值所需的日志空间的大小。

-u *userid*

指定具有系统管理员或数据库管理员特权的用户标识。如果忽略此选项，则采用当前用户。

-p *password*

对所指定的用户标识指定口令。

-h

显示帮助信息。指定此选项时，会忽略所有其他选项，而只显示帮助信息。

使用说明

如果表很大且存在下列情况的话，可能需要使用此工具而不是使用 SET INTEGRITY 语句上的 FORCE GENERATED 选项:

- 在改变了生成列的生成表达式之后，所有的列值都必须重新生成。
- 在生成列中使用的外部 UDF 已更改，导致许多列值也更改。
- 生成列已被添加至表。
- 执行了大型的装入或装入追加，但未提供生成列的值。
- 由于长期运行的并行事务或由于表大小而造成日志空间过小。

此工具将重新生成根据表达式创建的所有列值。在更新表时，会执行间歇性的提交以避免用完所有的日志空间。一旦运行了 **db2gncol**，就可使用 SET INTEGRITY 语句来使表脱离检查暂挂方式。

db2inidb - 初始化镜像数据库

在分割镜像环境中，此命令用来初始化用于不同目的的镜像数据库。

权限 必须是下列其中一项：

- *sysadm*
- *sysctrl*
- *sysmaint*

必需的连接

无

命令语法



命令参数

数据库别名

指定要初始化的数据库的别名。

SNAPSHOT

指定镜像数据库将作为主数据库的克隆来进行初始化。此数据库是只读的。

STANDBY

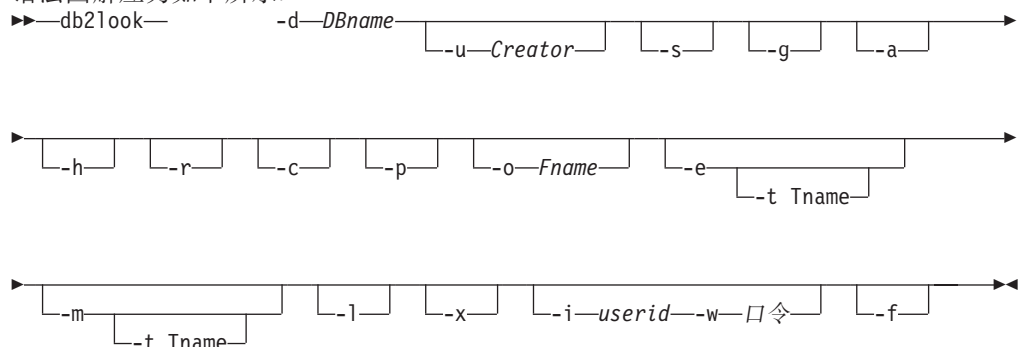
指定数据库将被置于前滚暂挂状态。可取装主数据库中的新日志，并应用于备用数据库。然后，可以使用备用数据库来代替主数据库（如果主数据库关闭的话）。

MIRROR

指定镜像数据库将用作备份映象，它可用来恢复主数据库。

db2look - DB2 统计信息抽取工具

语法图解应为如下所示：



已在 `-c` 参数后添加了 `-td x` 参数。其定义如下所示：

Specifies the statement delimiter for SQL statements generated by db2look. If this option is not specified, the default is the semicolon ';'. It is recommended that this option be used if the -e option is specified. In this case, the extracted objects may contain triggers or SQL routines.

还会添加以下示例:

Generate the DDL statements for objects created by all users in the database DEPARTMENT. The db2look output is sent to file db2look.sql:

```
db2look -d department -a -e -td % -o db2look.sql
db2 -td% -f db2look.sql
```

db2updv7 - 将数据库更新为版本 7 当前修正级别

此命令会以下列方式更新数据库中的系统目录，以支持当前“修订包”：

- 允许使用新内置功能（ABS、DECRYPT_BIN、DECRYPT_CHAR、ENCRYPT、GETHINT、MULTIPLY_ALT 和 ROUND）。
- 启用新内置过程（GET_ROUTINE_SAR 和 PUT_ROUTINE_SAR）的使用。
- 在 Windows 和 OS/2 数据库上添加或应用对 WEEK_ISO 和 DAYOFWEEK_ISO 功能的校正。
- 应用对从版本 2 至迁移至版本 6 的表的表包装描述符的校正。
- 创建视图 SYSCAT.SEQUENCES。

权限 *sysadm*

必需的连接

数据库。此命令自动建立与指定数据库的连接。

命令语法

```
db2updv7 -d database_name [-u userid -p 口令] [-h]
```

命令参数

-d database-name

指定要更新的数据库的名称。

-u userid

指定用户标识。

-p password

指定用户的口令。

-h

显示帮助信息。指定此选项时，会忽略所有其他选项，而只显示帮助信息。

示例 安装“修订包”之后，通过发出以下命令更新样本数据库中的系统目录:

```
db2updv7 -d sample
```


使用说明

此工具只能在运行至少安装有“修订包 2”的 DB2 版本 7.1 或 7.2 的数据库上使用。如果对该命令发出了多次，则不会报告任何错误，且每个目录更新只应用一次。

要启用新的内置功能，应用程序都必须与此数据库断开连接，且必须释放该数据库（如果它曾经是活动的话）。

新命令行处理器选项（-x，不打印列标题）

新选项 -x 告知命令行处理器返回数据而不带任何标题，包括列名。此命令选项的缺省设置为 OFF。

DB2 CLP 的 True Type 字体需求

要在 DB2 命令行处理器 (CLP) 窗口中正确地显示单字节 (SBCS) 的国家语言字符，将字体更改为 True Type。

ADD DATALINKS MANAGER

此命令所需的权限级别为下列其中一项：

- *sysadm*
- *sysctrl*
- *sysmaint*

还应添加下列使用说明：

仅当所有应用程序已与数据库断开连接之后，此命令才会生效。正在添加的 DB2 Data Links Manager 必须配置完整，且对此命令的运行必须成功。还必须使用 `d1fm add_db` 命令在 DB2 Data Links Manager 上注册该数据库。可添加至数据库的 DB2 Data Links Manager 的最大数目为 16。

ARCHIVE LOG（新命令）

归档日志

关闭并截断可恢复数据库的活动日志文件。如果启用了用户出口，则发出归档请求。

权限

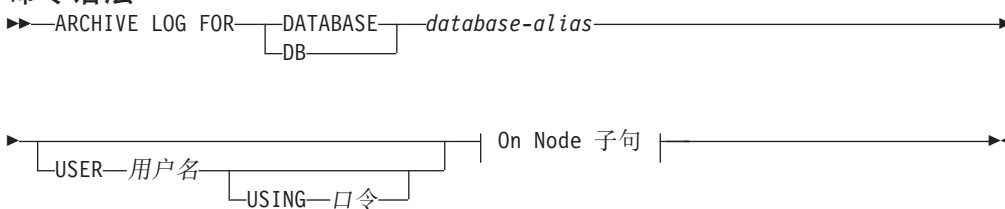
下列其中一项:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必需的连接

此命令自动建立与指定数据库的连接。如果连接已经存在，则会返回错误。

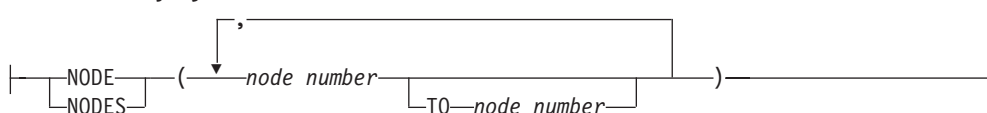
命令语法



On Node 子句:



Node List 子句:



命令参数

DATABASE database-alias

指定要归档其活动日志的数据库的别名。

USER username

标识用来尝试连接的用户名。

USING password

指定用来认证用户名的口令。

ON ALL NODES

指定应对 `db2nodes.cfg` 文件中的所有节点发出该命令。如果未指定节点子句，则这是缺省值。

EXCEPT

指定应对 db2nodes.cfg 文件中的所有节点（节点列表中指定的节点除外）发出该命令。

ON NODE/ON NODES

指定应对节点组上的指定数据库归档日志。

node number

在节点列表中指定节点号。

TO node number

在指定应对其归档日志的节点的范围时使用。从指定的第一个节点号至（包括）第二个节点号的所有节点都包括在节点列表中。

使用说明

可以使用此命令来收集一组完整的直至已知点的日志文件。然后使用这些日志文件来更新备用数据库。

仅当没有与指定数据库的数据库连接时，才能执行此功能。这会阻止用户使用未提交的事务执行该命令。同样的，ARCHIVE LOG 命令将不会强制提交用户的不完整事务。如果与指定数据库的数据库连接已经存在且执行了此命令，则该命令将终止并返回错误。如果执行此命令时另一应用程序的事务正在使用指定数据库，性能将会稍微降低，因为该命令会将日志缓冲区刷新至磁盘。尝试将日志记录写至缓冲区的任何其他事务只能等待，直到刷新完成为止。

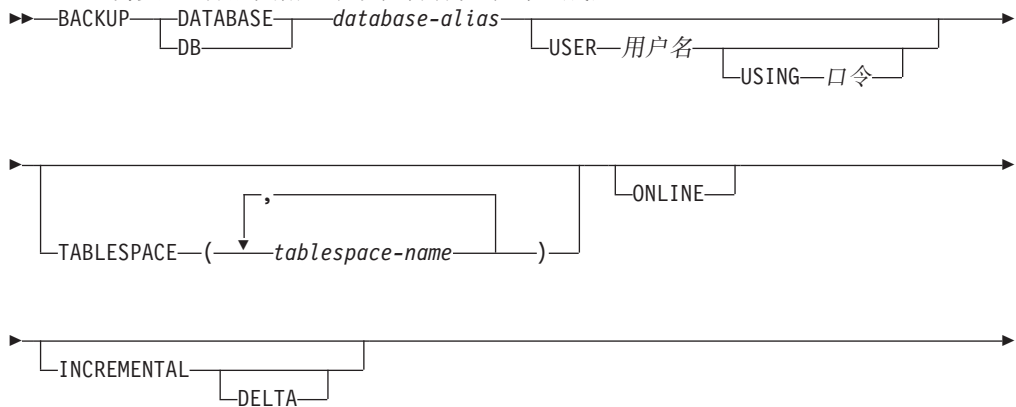
如果是在 MPP 环境中使用的话，可能会使用节点子句来指定节点子集。如果未指定节点子句，此命令的缺省行为是关闭所有节点上的活动日志并对其进行归档。

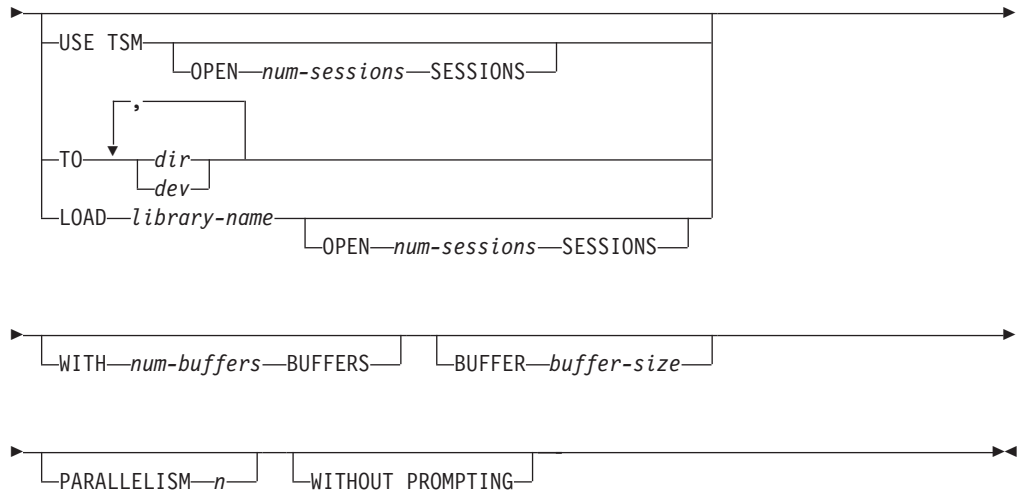
使用此命令将导致数据库丢失一部分 LSN 空间，从而会加速耗尽有效的 LSN。

BACKUP DATABASE

语法图

将会更新 BACKUP DATABASE 的语法图，以反映新的 INCREMENTAL 参数和可选 DELTA 自变量。单独指定 INCREMENTAL 选项将导致生成累积备份映象。可选 DELTA 自变量可用来指定非累积备份映象的生成。





DB2 Data Links Manager 注意事项

如果为数据库配置了一个或多个 Data Links 服务器，则备份操作将成功进行，甚至在 Data Links 服务器不可用的情况下也会如此。如果 Data Links 服务器重新启动，将会先在 Data Links 服务器上完成备份处理，然后该服务器才再次对数据库可用。

注：如果仍在等待不可用的 Data Links 服务器的备份是保留在数据库历史文件中的备份（数据库配置参数 num_db_backups）的两倍，备份操作将失败。

BIND

应修改 DB2 的命令语法以显示联合参数，如下所示：

```
FEDERATED---NO---
      '-YES--'
```

FEDERATED

指定软件包中的静态 SQL 语句是引用别名还是引用联合视图。如果未指定此选项，且软件包中的静态 SQL 语句引用了别名或联合视图，则会返回警告并创建一个软件包。

NO 软件包的静态 SQL 语句名中未引用别名或联合视图。如果在此软件包的准备或绑定期间在静态 SQL 语句中遇到了别名或联合视图，会返回错误，且不会创建该软件包。

YES 可在软件包的静态 SQL 语句名中引用别名或联合视图。如果该软件包的准备或绑定期间在静态 SQL 语句中未遇到任何别名或联合视图，则不会返回任何错误，且会创建该软件包。

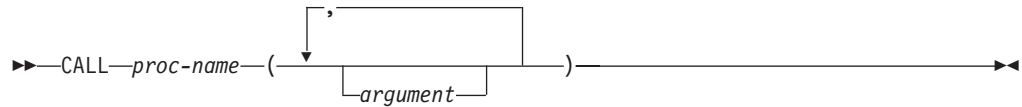
注：在版本 7 修订包 2 中，在未对 FEDERATED 选项指定值的情况下预编译源文件或绑定绑定文件时，服务器生成了 SQL1179W 警告消息。当源文件或绑定文件包括静态 SQL 对别名的引用时，会生成同一消息。有两个例外情况：

- 对于“修订包”版本早于“版本 7 修订包 2”的客户机或下级客户机，sqlprep() API 不会在此消息文件中报告此 SQL1179W 警告。在此情况下，“命令行处理器”PRECOMPILE 命令也不会输出该警告。

- 对于“修订包”版本早于“版本 7 修订包 2”的客户机或下级客户机，sqlabndx API 不会在消息文件中报告此 SQL1179W 警告。但是，消息文件也会错误地包括 SQL0092N 消息，指示未创建任何软件包。这是不正确的，因为实际上创建了该软件包。“命令行处理器”BIND 命令返回相同的错误警告。

CALL

CALL 命令的语法应以如下形式出现：



argument 参数的描述已经更改为：

Specifies one or more arguments for the stored procedure. All input and output arguments must be specified in the order defined by the procedure. Output arguments are specified using the "?" character. For example, a stored procedure foo with one integer input parameter and one output parameter would be invoked as "call foo (4, ?)".

注：

1. 当从操作系统提示符处调用此实用程序时，可能需要对该命令进行定界，如下所示：

```
"call DEPT_MEDIAN (51)"
```

还可使用单引号 (')。

2. 被调用的存储过程的名称在数据库中必须唯一。
3. 必须对存储过程进行编目。如果调用了未编目的过程，将返回 DB21036 错误消息。
4. 按照存储过程定义，如果未在命令行上指定足够的参数，或是命令行参数未以正确的次序（输入、输出）出现，则将返回 DB21101E 消息。
5. 结果列最多可为 1023 个字符。
6. LOBS 和二进制数据（FOR BIT DATA、VARBINARY、LONGVARBINARY、GRAPHIC、VARGAPHIC 或 LONGVARGRAPHIC）不受支持。
7. CALL 支持结果集。
8. 如果调用的 SP 带有类型不受支持的 OUTPUT 变量，则 CALL 会失败，而返回消息 DB21036。
9. CALL 的 INPUT 参数的最大长度为 1024。

DROP DATALINKS MANAGER (新命令)

DROP DATALINKS MANAGER

从指定数据库的已注册 DB2 Data Links Manager 列表中删除 DB2 Data Links Manager。

权限

下列其中一项:

- *sysadm*
- *sysctrl*
- *sysmaint*

命令语法

```
▶▶ DROP DATALINKS MANAGER FOR DATABASE dbname USING 名称 ▶▶  
                                  └───┬───┘  
                                  DB
```

命令参数

DATABASE *dbname*

指定数据库名称。

USING *name*

指定 DB2 Data Links Manager 服务器的名称 (如 LIST DATALINKS MANAGER 命令中所示)。

示例

示例 1

当某些数据库表与 micky.almaden.ibm.com Dropping 存在链接时, 从驻留在主机 bramha.almaden.ibm.com 上的实例验证中的数据库 TEST 中删除 DB2 Data Links Manager micky.almaden.ibm.com。在删除 DB2 Data Links Manager 时, 采取下列步骤是非常重要的。

1. 对数据库 TEST 进行数据库备份。
2. 如果有指定向 micky.almaden.ibm.com 的任何链接, 断开这些链接:
 - a. 使用属于 SYSADM_GROUP 的用户标识进行登录, 并获取与数据库 TEST 的专用方式连接。

```
connect to test in exclusive mode
```

确保这是使用该用户标识与测试的唯一连接。这将停止创建任何新的链接。

- b. 获取所有 FILE LINK CONTROL DATALINK 列及在数据库中包含这些列的表的列表。

```
select tabname, colname from syscat.columns where substr(dl_features, 2, 1) = 'F'
```

- c. 对于列表中的每个 FILE LINK CONTROL DATALINK 列, 发出 SQL SELECT 以确定与 micky.almaden.ibm.com 的链接是否存在。例如, 对于表 t 中的 DATALINK 列 c, SELECT 语句将为:

```
select count(*) from t where dlurlserver(t.c) = 'MICKY.ALMADEN.IBM.COM'
```

- d. 对于包含这种链接的每个 FILE LINK CONTROL DATALINK 列, 发出 SQL UPDATE 以断开作为与 micky.almaden.ibm.com 的链接的值的链接。例如, 对于表 t 中的 DATALINK 列 c, UPDATE 语句应为:

```
update t set t.c = null where dlurlserver(t.c) = 'MICKY.ALMADEN.IBM.COM'
```

如果 t.c 不可空，则可使用以下语句代替：

```
update t set t.c = dlvalue('') where dlurlserver(t.c) = 'MICKY.ALMA DEN.IBM.COM'
```

e. 提交此 SQL UPDATE:

```
commit
```

3. 发出 DROP DATALINKS MANAGER 命令:

```
drop datalinks manager for db test using node micky.almaden.ibm.com
```

4. 终止专用方式连接以使更改生效，并允许与数据库的其他连接:

```
terminate
```

5. 对 micky.almaden.ibm.com 上的 TEST 启动备份信息的断开链接处理和无用单元收集。作为“DB2 Data Links Manager 管理员”，对 micky.almaden.ibm.com 发出以下命令:

```
dlfm drop_dlm test validate bramha.almaden.ibm.com
```

这将断开仍链接至数据库 TEST 的所有文件的链接，以防用户在调用步骤 3 之前未能断开它们的链接。如果 micky.almaden.ibm.com 具有先前链接至数据库 TEST 的文件的备份信息（例如，归档文件、元数据），则此命令将启动该信息的无用单元收集。实际的断开链接和无用单元收集将在不同时间执行。

示例 2

在已经从数据库 TEST 中删除了 DB2 Data Links Manager 的情况下，删除作为称为 micky.almaden.ibm.com 的 DB2 Data Links Manager 上的文件的链接的 DATALINK 值。如果在删除 micky.almaden.ibm.com 时未遵循“示例 1”，则这可能是必需的。对于这种 DATALINK 值 (SQL0368)，SQL DELETE、SELECT 和 UPDATE 语句将不会成功。用户必须对包含这种 DATALINK 值的每个表执行调整操作。作为与 micky.almaden.ibm.com 的链接的每个 DATALINK 值都将更新为 NULL 或零长度的 DATALINK 值。包含这种值的任何行都将被插入异常表中（如果指定了异常表的话）。但是，DATALINK 值将不会包括前缀名。原始 DATALINK 值中的前缀名对于系统将不再是可获取的，因为 micky.almaden.ibm.com 已被删除。例如，如果原始 DATALINK 值为 'http://host.com/dlfs/x/y/a.b'，且 '/dlfs' 为前缀名，则异常表中的 DATALINK 值将包含 'http://host.com/x/y/a.b'。这些 DATALINK 值引用的文件在 DB2 Data Links Manager 上将继续处于链接状态。可在 micky.almaden.ibm.com 上发出 dlfm drop_dlm 命令，以启动这些文件的断开链接处理。如果 micky.almaden.ibm.com 具有先前链接至数据库 TEST 的文件的备份信息（例如，归档文件、元数据），则此命令将启动该信息的无用单元收集。实际的断开链接和无用单元收集将在不同时间执行。

示例 3

数据库 TEST 的 DB2 Data Links Manager micky.almaden.ibm.com 多种形式。此方案演示 DB2 Data Links Manager 在删除后可重新注册，然后会被视作全新的 DB2 Data Links Manager。下列步骤只是说明可行方案。如果遵循了“示例 1”中的步骤来删除 micky.almaden.ibm.com（建议），则指向 micky.almaden.ibm.com 的较旧形式的链接将不存在；即，将下面的步骤 7 中将不会看到错误 SQL0368。

1. 将 micky.almaden.ibm.com 注册至数据库 TEST:

```
add datalinks manager for db test using node micky.almaden.ibm.com port 14578
```

2. 创建指向 micky.almaden.ibm.com 上的文件的链接:


```
connect to test
create table t(c1 int, c2 datalink linktype url file link control mode db2options)
insert into t values(1, dlvalue('file://micky.almaden.ibm.com/pictures/yosemite.jpg'))
commit
terminate
```

3. 从数据库 TEST 中删除 micky.almaden.ibm.com:

```
drop datalinks manager for db test using micky.almaden.ibm.com
```

4. 选择 DATALINK 值:

```
connect to test
select * from t
terminate
```

该用户将看到:

```
SQL0368 The DB2 Data Links Manager "MICKY.ALMADEN.IBM.COM"
is not registered to the database. SQLSTATE=55022.
```

5. 再次将 micky.almaden.ibm.com 注册至数据库 TEST:

```
add datalinks manager for db test using node micky.almaden.ibm.com port 14578
```

6. 插入其他 DATALINK 值:

```
connect to test
insert into t values(2, dlvalue('file://micky.almaden.ibm.com/pictures/tahoe.jpg'))
commit
```

7. 选择 DATALINK 值:

```
select c2 from t where c1 = 2
```

是成功的, 因为选择的值是指向 micky.almaden.ibm.com 的当前注册形式的链接。

```
select c2 from t where c1 = 1
```

返回:

```
SQL0368 The DB2 Data Links Manager "MICKY.ALMADEN.IBM.COM"
is not registered to the database. SQLSTATE=55022.
```

因为选择的值是指向在上面步骤 3 中已删除的 micky.almaden.ibm.com 的形式的链接。

使用说明

DROP DATALINKS MANAGER 命令的影响不会回滚。在使用 DROP DATALINKS MANAGER 命令时遵循“步骤 1”中的步骤是非常重要的。

仅当所有应用程序已与数据库断开连接之后, 此命令才会生效。

成功完成该命令时, 会通知用户 (DB210201I) 未对 DB2 Data Links Manager 进行任何处理。删除 DB2 Data Links Manager 之前, 用户必须确保数据库没有指向 DB2 Data Links Manager 上的文件的任何链接。如果删除 DB2 Data Links Manager 后数据库中的链接仍然存在, 则用户必须运行调整实用程序以除去它们。调整实用程序将把这些链接设置为 NULL (如果 DATALINK 列可空的话) 或设置为零长度 DATALINK 值。

对应于数据库和已删除 DB2 Data Links Manager 之间的链接的文件仍处于链接状态。即, 不能对它们进行类似读、写、重命名、删除、更改许可权或更改所有权之类的操作。

此命令将不会对 DB2 Data Links Manager 上的未链接文件的归档副本进行无用单元收集。但是，用户可在 DB2 Data Links Manager 上使用 `dlfm drop_dlm` 命令显式启动断开链接处理和无用单元收集。

建议在删除 DB2 Data Links Manager 之前进行数据库备份。此外，确保所有复制预订已复制了涉及此 DB2 Data Links Manager 的所有更改。

如果从数据库删除 DB2 Data Links Manager 之前进行了备份，且备份映像被用来在删除 DB2 Data Links Manager 后进行恢复，则恢复或前滚处理可能会将某些表置于数据链路调整暂挂 (DRP) 状态。

EXPORT

在 “DB2 Data Links Manager Considerations” 一节中，过程的步骤 3（该步骤确保复制的是表和由 DATALINK 列引用的对应文件的一致副本以供输出）应为：

```
3. Run the dlfm_export utility at each Data Links server. Input to the
dlfm_export utility is the control file name, which is generated by the
export utility. This produces a tar (or equivalent) archive of the files listed
within the control file. For Distributed File Systems (DFS), the dlfm_export
utility will get the DCE network root credentials before archiving the files
listed in the control file. dlfm_export does not capture the ACLs information
of the files that are archived.
```

在同一节中，“Successful execution of EXPORT results in the generation of the following files” 后的公告牌应作如下修改：

第一个公告牌中的第二句应为：

```
A DATALINK column value in this file has the same format
as that used by the import and load utilities.
```

第二个公告牌中的第一句应为：

```
Control files server_name, which are generated for
each Data Links server. (On the Windows NT operating system,
a single control file, ctrlfile.lst, is used by all Data Links
servers. For DFS, there is one control file for each cell.)
```

以下句子应添加至 “表 5” 前面的段落：

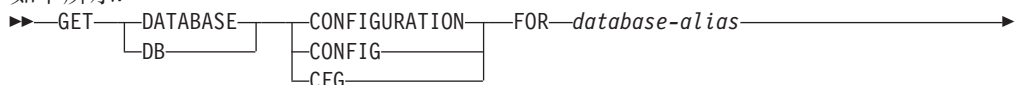
```
For more information about dlfm_export, refer to the "Data Movement
Utilities Guide and Reference" under "Using Export to move DB2 Data
Links Manager Data".
```

GET DATABASE CONFIGURATION

DL_TIME_DROP 配置参数的描述应更改为如下内容：

```
Applies to DB2 Data Links Manager only. This parameter specifies
the number of days files would be retained on an archive server
(such as a TSM server) after a DROP DATABASE command is issued.
```

将对 GET DATABASE CONFIGURATION 命令添加新参数 TRACKMOD。语法将如下所示：





将添加参数描述，如下所示：

TRACKMOD

指示 DB2 是否应跟踪数据库中的已修改页面，以允许进行增量备份。

OFF 指定不应跟踪已更改页面。对于在版本 7.1，修订包 3 之前创建的数据库，这是缺省情况。

ON 指定应跟踪已更改页面。如果设置了此参数，会进行数据库的增量备份。对于使用版本 7.1，修订包 3 或更新版本创建的数据库，这是缺省情况。

GET ROUTINE (新命令)

GET ROUTINE

检索指定 SQL 例程的例程 “SQL 归档” (SAR) 文件。

权限

dbadm

必需的连接

数据库。如果启用隐式连接，则会建立与缺省数据库的连接。

命令语法

```
▶▶ GET ROUTINE INTO file_name FROM SPECIFIC PROCEDURE routine_name ▶▶
```

命令参数

INTO file-name

对存储例程 “SQL 归档” (SAR) 的文件进行命名。

FROM 指示要检索的例程的规范的起始处。

SPECIFIC

指定的例程名被给定为特定名称。

PROCEDURE

该例程为 SQL 过程。

routine-name

过程的名称。如果指定了 SPECIFIC，则它是过程的特定名称。如果未用模式名限定该名称，则会将 CURRENT SCHEMA 用作例程的模式名。 *routine-name* 必须是被定义为 SQL 过程的现存过程。

示例

```
GET ROUTINE INTO procs/proc1.sar FROM PROCEDURE myappl.proc1;
```

GET SNAPSHOT

FCM FOR ALL NODES 参数的描述应为如下所示:

Provides Fast Communication Manager (FCM) statistics between the node against which the GET SNAPSHOT command was issued and the other nodes in the EEE instance.

IMPORT

在 “DB2 Data Links Manager Considerations” 一节中，以下句子应添加至步骤 3:

```
For Distributed File Systems (DFS), update the cell name information in the URLs (of the DATALINK columns) from the exported data for the SQL table, if required.
```

以下句子应添加至步骤 4:

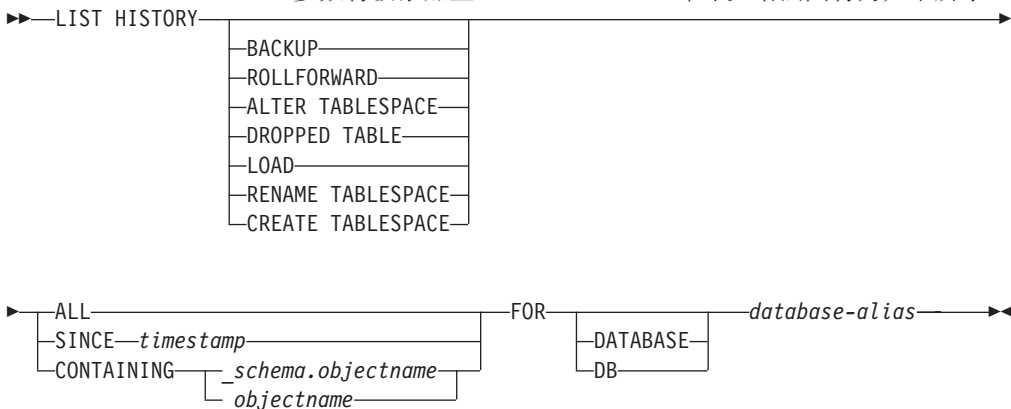
```
For DFS, define the cells at the target configuration in the DB2 Data Links Manager configuration file.
```

步骤 4 后面的段落应为:

When the import utility runs against the target database, files referred to by DATALINK column data are linked on the appropriate Data Links servers.

LIST HISTORY

CREATE TABLESPACE 参数将被添加至 LIST HISTORY 命令。语法图将为如下所示:



将添加参数描述, 如下所示:

CREATE TABLESPACE

列出所有 CREATE TABLESPACE 和 DROP TABLESPACE 操作。

Usage Notes 将更新为:

下列符号将添加至报告的 Operation 一节:

- A - 创建表空间
- O - 删除表空间
- U - 卸装

报告的 Type 一节中的符号将被重组为:

- 备份类型
 - F - 脱机
 - N - 联机
 - I - 增量脱机
 - O - 增量联机
 - D - Delta 脱机
 - E - Delta 联机
- 前滚类型
 - E - 日志结束
 - P - 时间点
- 装入类型
 - I - 插入

- R - 替换
- 改变表空间类型
 - C - 添加容器
 - R - 再平衡
- 停顿类型
 - S - 停顿共享
 - U - 停顿更新
 - X - 停顿专用
 - Z - 停顿复位

LOAD

在 "DB2 Data Links Manager Considerations" 一节中, 将以下句子添加至过程的步骤 1, 如果数据是被装入到带有 DATALINK 列 (该列是使用 FILE LINK CONTROL 定义的) 的表中的话, 则该步骤要在调用装入实用程序之前执行:

For Distributed File Systems (DFS), ensure that the DB2 Data Links Managers within the target cell are registered.

以下句子应添加至步骤 5:

For DFS, register the cells at the target configuration referred to by DATALINK data (to be loaded) in the DB2 Data Links Manager configuration file.

在 "Representation of DATALINK Information in an Input File" 一节中, `urlname` 的参数描述后的第一个注释应为:

Currently "http", "file", "unc", and "dfs" are permitted as a schema name.

第二个注释的第一句应为:

The prefix (schema, host, and port) of the URL name is optional. For DFS, the prefix refers to the schema cellname filespace-junction portion.

在定界 ASCII (DEL) 文件格式和非定界 ASCII (ASC) 文件格式的 DATALINK 数据示例中, 应除去第三个示例。

应除去列装入或导入规范被假定为 `DL_URL_DEFAULT_PREFIX` 的 DATALINK 数据示例, 并将这些示例替换为如下内容:

以下是列装入或导入规范被假定为 `DL_URL_REPLACE_PREFIX` 的 DATALINK 数据示例 ("http://qso"):

- `http://www.almaden.ibm.com/mrep/intro.mpeg`
这是使用下列部件存储的样本 URL:
 - 模式 = http
 - 服务器 = qso
 - 路径 = /mrep/intro.mpeg
 - 注释 = NULL 字符串
- `/u/me/myfile.ps`

这是使用下列组件进行存储的:

- 模式 = http
- 服务器 = qso
- 路径 = /u/me/myfile.ps
- 注释 = NULL 字符串

PING (新命令)

PING

测试使用 DB2 Connect 来建立连接的客户机与数据库服务器之间的基本连接的网络响应时间。

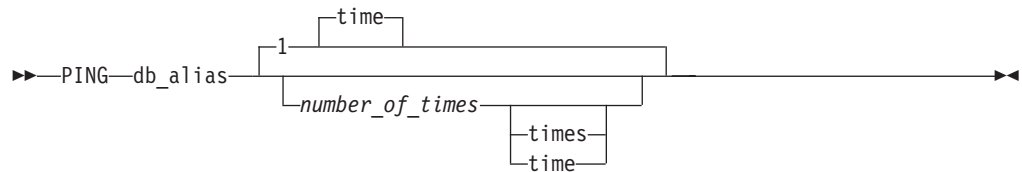
权限

无

必需的连接

数据库

命令语法



命令参数

db_alias

对 ping 操作发送至的 DRDA 服务器上的数据库指定数据库别名。

注：此参数尽管是必要的，但是目前不需要使用。它被留作将来使用。可以指定任何有效的数据库别名。

number of times

指定此测试的重复次数。该值必须在 1 和 32767 之间，包括 1 和 32767。缺省值是 1。将对每次重复返回一次记时。

示例

要测试连接至主机数据库服务器 `hostdb` 一次所需要的网络响应时间：

```
db2 ping hostdb 1  
或：  
db2 ping hostdb
```

该命令将显示类似以下的输出：

```
Elapsed time: 7221 microseconds
```

要测试连接至主机数据库服务器 `hostdb` 五次所需要的网络响应时间：

```
db2 ping hostdb 5  
或：  
db2 ping hostdb 5 times
```

该命令将显示类似以下的输出：

```
Elapsed time: 8412 microseconds  
Elapsed time: 11876 microseconds  
Elapsed time: 7789 microseconds  
Elapsed time: 10124 microseconds  
Elapsed time: 10988 microseconds
```

使用说明

数据库连接必须存在，才能调用此命令，否则，将发生错误。

返回的所用时间表示通过 DB2 Connect 建立的客户机与 DRDA 服务器数据库之间的连接所花的时间。

PUT ROUTINE (新命令)

PUT ROUTINE

使用指定例程“SQL 归档”(SAR) 文件来定义数据库中的例程。

权限

dbadm

必需的连接

数据库。如果启用隐式连接，则会建立与缺省数据库的连接。

命令语法

```
▶▶—PUT ROUTINE—FROM—file-name—OWNER—new-owner—USE REGISTERS
```

命令参数

FROM *file-name*

对存储例程“SQL 归档”(SAR) 的文件进行命名。

OWNER *new-owner*

为例程指定将用于例程的权限检查的新权限名。*new_owner* 必须对要定义的例程具有必需的特权。如果未指定 OWNER 子句，会使用一开始定义该例程的权限名。

USE REGISTERS

指示 CURRENT SCHEMA 和 CURRENT PATH 专用寄存器用来定义该例程。如果未指定此子句，则缺省模式和 SQL 路径的设置即在一开始定义该例程时使用的设置。CURRENT SCHEMA 在例程定义（包括例程名）中被用作未限定对象名的模式名，而 CURRENT PATH 则用来解析例程定义中的未限定例程和数据类型。

示例

```
PUT ROUTINE FROM procs/proc1.sar;
```

使用说明

不能以给定模式并行安装多个过程。

RECONCILE

命令描述中将添加以下使用说明：

During reconciliation, attempts are made to link files which exist according to table data, but which do not exist according to Data Links Manager metadata, if no other conflict exists. A required DB2 Data Links Manager is one which has a referenced DATALINK value in the table. Reconcile tolerates unavailability of a required DB2 Data Links Manager as well as those that are configured to the database but are not part of the table data.

如果未指定异常表，则异常报告文件 (*filename.exp*) 将包含主机名、文件名、列标识和不能对其重建文件引用的每个 DATALINK 列值的原因码。如果因为已使用 DROP DATALINKS MANAGER 命令从数据库中删除了 DB2 Data Links Manager 本身而不

能重建文件引用，则在异常报告中报告的文件名不是全文件名；即，它丢失了前缀部分。在其 DB2 Data Links Manager 已被删除或不可用的 DATALINK 值的异常表中，DATALINK 值中的文件名不是全文件名。前缀部分已丢失。例如，如果原始 DATALINK 值为 'http://host.com/dlfs/x/y/a.b'，则在异常表中报告的值将为 'http://host.com/x/y/a.b'；即，不包括前缀名 'dlfs'。这种情况下的异常报告文件将为 'x/y/a.b'；即不包括前缀名 'dlfs'。

调整过程结束时，仅当在所有所需的 DB2 Data Links Manager 上完成调整处理时，该表才会脱离数据链路调整暂挂 (DRP) 状态。如果调整处理在任何一个所需的 DB2 Data Links Manager 上暂挂（因为它们不可用），该表将保持（或被置于）DRP 状态。

以下内容应添加至可能的违例列表：

```
00010-DB2 Data Links Manager referenced by the DATALINK value
has been dropped from the database using the DROP DATALINKS
MANAGER command. In this case, the corresponding DATALINK value
in the exception table will not contain the prefix name.
For example, if the original DATALINK value was
'http://host.com/dlfs/prfx/x/y/a.b', and '/dlfs/prfx' is
the prefix name, the exception table will contain
'http://host.com/x/y/a.b'.
```

REORGANIZE TABLE

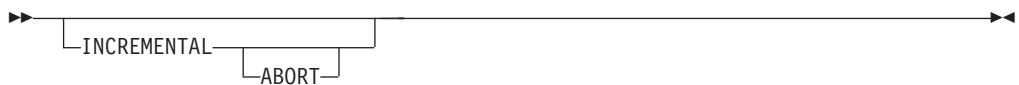
以下语句将添加至 Usage Notes:

REORGANIZE TABLE cannot use an index that is based on an index extension.

RESTORE DATABASE

语法

以下选项将被添加至 RESTORE DATABASE 命令语法的 TABLESPACE/TABLESPACE ONLINE/HISTORY FILE 选项后：



将添加参数描述，如下所示：

INCREMENTAL

Results in a manual cumulative restore of the database. The user will issue each of the restore commands.

ABORT

This parameter should be used to terminate an incremental restore before successful completion.

还将添加下列示例：

以下是可恢复数据库的样本周增量备份策略。完全备份被安排为每周进行一次，每天增加一点，周三增加一个增量：

```
(Sun) backup db kdr use adsm
(Mon) backup db kdr online incremental delta use adsm
(Tue) backup db kdr online incremental delta use adsm
(Wed) backup db kdr online incremental use adsm
(Thu) backup db kdr online incremental delta use adsm
(Fri) backup db kdr online incremental delta use adsm
(Sat) backup db kdr online incremental use adsm
```

要对周五早晨以前创建的映象进行人工数据库恢复，发出以下命令：

```
restore db kdr incremental taken at (Thu)
restore db kdr incremental taken at (Sun)
restore db kdr incremental taken at (Wed)
restore db kdr incremental taken at (Thu)
```

注：不管所恢复的映象是数据库映象还是表空间映象，格式为 `db2 restore db <name>` 的任何 `RESTORE` 命令都将执行完全数据库恢复。格式为 `db2 restore db <name> tablespace` 的任何 `RESTORE` 命令都将对在映象中找到的表空间执行表空间恢复。提供了表空间列表的任何 `RESTORE` 命令将对显式列示的任何表空间执行恢复。

DB2 Data Links Manager Considerations

“DB2 Data Links Manager Considerations” 一节中的第二段应替换为：

```
If one or more Data Links servers are configured for the database,
the restore operation will succeed, even if a Data Links server is not available.
When the Data Links server restarts, restore processing will be completed
on that Data Links server before it becomes available to the database again.
NOTE: If a database restore operation is still waiting for an unavailable
Data Links server, any subsequent database or table space restore operations
will fail.
```

ROLLFORWARD DATABASE

“DB2 Data Links Manager Considerations” 一节中的第二段应替换为：

```
If one or more Data Links servers are configured for the database,
the rollforward operation will succeed, even if a Data Links server is not available.
When the Data Links server restarts, rollforward processing will be completed
on that Data Links server before it becomes available to the database again.
```

CLP 返回码中的文档错误

在第 2 章的 `Command Line Processor Return Codes` 一节中，第二段应为如下所示：

For example, the following Bourne shell script executes the `GET DATABASE MANAGER CONFIGURATION` command, then inspects the CLP return code:

```
db2 get database manager configuration
if [ "$?" = "0" ]
then echo "OK!"
fi
```

Data Movement Utilities Guide and Reference

Chapter 2. Import

Using Import with Buffered Inserts

本节末尾的注释应为:

注: In all environments except EEE, the buffered inserts feature is disabled during import operations in which the INSERT_UPDATE parameter is specified.

Chapter 3. Load

Pending States After a Load Operation

本节最后一段中的头两句已经更改为如下内容:

The fourth possible state associated with the load process (check pending state) pertains to referential and check constraints, DATALINKS constraints, AST constraints, or generated column constraints. For example, if an existing table is a parent table containing a primary key referenced by a foreign key in a dependent table, replacing data in the parent table places both tables (not the table space) in check pending state.

Load Restrictions and Limitations

下列限制适用于生成的列和装入实用程序:

- 不能装入唯一索引中有生成列的表, 除非生成列是索引的“包括”列或使用了 `generatedoverride` 文件类型修饰符。如果使用了此修饰符, 则应在输入数据文件中提供该列的所有值。
- 不能装入分区键中有生成列的表, 除非使用了 `generatedoverride` 文件类型修饰符。如果使用了此修饰符, 则应在输入数据文件中提供该列的所有值。

`totalreespace` File Type Modifier

`totalreespace` 文件类型修饰符 (LOAD) 已被修改为接受 0 到 2 147 483 647 之间的值。

Chapter 4. AutoLoader

`rexecd` Required to Run Autoloader When Authentication Set to YES

在 `Autoloader Options` 一节中, 将把下列注释添加到“AUTHENTICATION 和 PASSWORD 参数”描述中:

| In a Linux environment, if you are running the autoloader with the authentication option
| set to YES, rexecd must be enabled on all machines. If rexecd is not enabled the following
| error message will be generated:

```
| openbreeze.torolab.ibm.com: Connection refused  
| SQL6554N An error occurred when attempting to remotely execute a process.
```

| The following error messages will be generated in the db2diag.log file:

```
| 2000-10-11-13.04.16.832852 Instance:svtdbm Node:000  
| PID:19612(db2at1d) Appid:  
| oper_system_services sqloRemoteExec Probe:31
```

Replication Guide and Reference

复制和非 IBM 服务器

必须使用 DataJoiner 版本 2 或更新版本来将数据复制至非 IBM 服务器（如 Informix、Microsoft SQL Server、Oracle、Sybase 和 Sybase SQL Anywhere），或从这些服务器复制数据。不能将关系连接功能用于此类型的复制，原因是 DB2 Relational Connect 版本 7 没有更新功能。另外，必须使用 DJRA（DataJoiner 复制管理）来为所有现存版本的 DB2 和 DataJoiner 管理所有平台（AS/400、OS/2、OS/390、UNIX 和 Windows）上的这些多机种复制。

在 Windows 2000 上进行复制

DB2 DataPropagator 版本 7 与 Windows 2000 操作系统兼容。

保存 SQL 文件时的已知错误

如果使用“DB2 Connect 个人版”中的“控制中心”，则不能保存 SQL 文件。如果您试图保存 SQL 文件，则会获得错误消息，指出“数据库管理服务器”（DAS）不活动，这时实际上 DAS 不可用，因为它不是随 DB2 Connect PE 一起交付的。

DB2 维护

建议为在复制环境中使用的各种 DB2 产品安装最新的维护。

Web 上的数据差异实用程序

可以从 Web 上的以下网址处下载“数据差异实用程序”（DDU）：
<ftp://ftp.software.ibm.com/ps/products/datapropagator/fixes/>。DDU 是一个样本实用程序，可以用来对同一文件的两个版本进行比较，并生成一个输出文件，显示这两个文件之间的差异。有关详情，参见随样本实用程序提供的自述文件。

Chapter 3. Data replication scenario

复制方案

查看 DataPropagator Web 站点 (<http://www.ibm.com/software/data/dpropr/>) 的 Library 页，以了解另一异机种数据复制方案。遵循该方案中的步骤，将 AIX 上的 Oracle 数据库中的复制源表中的更改复制到 DB2 Windows NT 版上的数据库的目标表中。该方案使用“DB2 DataJoiner 复制管理”（DJRA）工具、Capture 触发器、Apply 程序和 DB2 DataJoiner。

在该书的第 44 页上的步骤 6 中有关创建口令文件的指导应为如下所示：

Step 6: Create a password file

Because the Apply program needs to connect to the source server, you must create a password file for user authentication. Make sure that the user ID that will run the Apply program can read the password file.

To create a password file:

1. From a Windows NT command prompt window, change to the C:\scripts directory.
2. Create a new file in this directory called DEPTQUAL.PWD. You can create this file using any text editor, such as Notepad. The naming convention for the password file is *applyqual.pwd*; where *applyqual* is a case-sensitive string that *must* match the case and value of the Apply qualifier used when you created the subscription set. For this scenario, the Apply qualifier is DEPTQUAL.

注: The filenaming convention from Version 5 of DB2 DataPropagator is also supported.

3. The contents of the password file has the following format:

```
SERVER=server USER=userid PWD=password
```

Where:

server

The name of the source, target, or control server, exactly as it appears in the subscription set table. For this scenario, these names are SAMPLE and COPYDB.

userid

The user ID that you plan to use to administer that particular database. This value is case-sensitive for Windows NT and UNIX operating systems.

password

The password that is associated with that user ID. This value is case-sensitive for Windows NT and UNIX operating systems.

Do not put blank lines or comment lines in this file. Add only the server-name, user ID, and password information.

4. The contents of the password file should look similar to:

```
SERVER=SAMPLE USER=subina PWD=subpw  
SERVER=COPYDB USER=subina PWD=subpw
```

有关 DB2 认证和安全性的详情, 参考 *IBM DB2 管理指南*。

Chapter 5. Planning for replication

表名和列名

复制不支持带有空格的表名和列名。

DATALINK 复制

在 Solaris 上, DATALINK 复制是作为版本 7.1 修订包 1 中的一部分来提供的。它需要在源和目标 DATALINK 文件系统中运行的 FTP 守护程序, 它还支持使用 MDTM (modtime) 命令, 该命令会显示给定文件的最新修改时间。若您正在使用 Solaris 操作系统的版本 2.6, 或者不包括对 MDTM 的 FTP 支持的任何其他版本, 则需要其他软件, 如 WU-FTPD。

不能在 AS/400 上的 DB2 数据库与其他平台上的 DB2 数据库之间复制 DATALINK 列。

在 AS/400 平台上，不支持复制 DATALINK 值的“注释”属性。

如果运行的是 AIX 4.2，则在运行缺省用户出口程序 (ASNDLCOPY) 之前，必须安装 APAR IY03101 (AIX 4210-06 RECOMMENDED MAINTENANCE FOR AIX 4.2.1) 的 PTF。此 PTF 包含了对 PTF 守护程序中“modtime/MDTM”命令的 Y2K 修正。要验证修正，检查从“modtime <文件>”命令中返回的上一次修正时间，其中 <文件> 是在 2000 年 1 月 1 日之后修改的文件。

如果目标表是外部 CCD 表，则 DB2 DataPropagator 会调用 ASNDLCOPY 例程以复制 DATALINK 文件。有关如何使用 ASNDLCOPY 和 ASNDLCOPYD 程序的最新消息，参见每个程序源代码的前言部分。下列限制适用：

- 内部 CCD 表可包含 DATALINK 指示符，但不能包含 DATALINK 值。
- 压缩的外部 CCD 表可包含 DATALINK 值。
- 未压缩的 CCD 目标表不能包含任何 DATALINK 列。
- 如果源服务器和目标服务器为同一服务器，则预订集一定不能包含带有 DATALINK 列的任何成员。

LOB 限制

压缩的内部 CCD 表不能包含对 LOB 列或 LOB 指示符的引用。

计划复制

第 65 页的“连通性”应包括以下事实：

```
If the Apply program cannot connect to the control server,  
the Apply program terminates.
```

在对 AS/400 使用数据分块时，必须确保要在该时间间隔期间复制的数据总量不超过“4,000,000 行”，而不是书中第 69 页所说的“4 MB”。

Chapter 6. Setting up your replication environment

随处更新先决条件

如果想要设置带有冲突检测且在预订集中有超过 150 个预订集成员的随处更新复制，则必须运行以下 DDL 以在控制服务器上创建 ASN.IBMSNAP_COMPENSATE 表：

```
CREATE TABLE ASN.IBMSNAP_COMPENSATE (  
    APPLY_QUAL char(18) NOT NULL,  
    MEMBER SMALLINT,  
    INTENTSEQ CHAR(10) FOR BIT DATA,  
    OPERATION CHAR(1));
```

设置复制环境

第 95 页“Customizing CD table, index, and tablespace names”中声明 DPREPL.DFT 文件在 \sqllib\bin 目录或者 \sqllib\java 目录中。这是不正确的，DPREPL.DFT 应该是在 \sqllib\cc 目录中。

在第 128 页上，保留限制描述应该声明仅当 Capture 热启动或者当您使用 Capture 剪除命令时，才使用保留限制来剪除行。如果利用自动剪除选项来启动 Capture，则 Capture 将不使用保留限制来剪除行。

Chapter 8. Problem Determination

“复制分析器”在 Windows 32 位系统和 AIX 上运行。要在 AIX 上运行“分析器”，应确保在 PATH 环境变量中，sqlllib/bin 目录在 /usr/local/bin 之前出现，以避免与 /usr/local/bin/analyze 发生冲突。

“复制分析器”还有两个附加的可选关键字 CT 和 AT。

CT=*n* 仅显示在 Capture 跟踪表中出现不到 *n* 天的那些项目。此关键字是可选的。如果未指定此关键字，则缺省值为 7 天。

AT=*n* 仅显示在 Apply 跟踪表中出现不到 *n* 天的那些项目。此关键字是可选的。如果未指定此关键字，则缺省值为 7 天。

示例:

```
analyze mydb1 mydb2 f=mydirectory ct=4 at=2 deepcheck q=applyqual1
```

对于“复制分析器”，更新了以下关键字信息:

deepcheck

指定“分析器”执行更为彻底的分析，包括下列信息：CD 和 UOW 表调整信息、DB2 OS/390 版表空间分区和压缩详细信息、有关预订键的目标索引的分析、预订时间限制和预订集 SQL 语句错误。分析会涵盖所有服务器。此关键字是可选的。

lightcheck

指定下列信息应从报告中排除：即下列项中的所有列详细信息应被排除：ASN.IBMSNAP_SUBS_COLS 表、预订错误、不规则或忽略的内容，以及不正确或效率低的索引。信息的减少会节省资源，并使生成的 HTML 输出文件更小一些。此关键字是可选的，且它与 deepcheck 关键字互斥。

“分析器”工具是在 PTF 中提供的，供在 AS/400 平台上进行复制之用。这些工具收集有关复制环境的信息，并生成一个 HTML 文件，该文件可发送至“IBM 服务代表”以帮助确定问题所在。要获取 AS/400 工具，下载适当的 PTF（例如，对于产品 5769DP2，必须下载 PTF SF61798 或其最新的替代产品）。

将以下问题和解决方案添加至“Troubleshooting”一节:

Problem: The Apply program loops without replicating changes; the Apply trail table shows STATUS=2.

The subscription set includes multiple source tables. To improve the handling of hotspots for one source table in the set, an internal CCD table is defined for that source table, but in a different subscription set. Updates are made to the source table but the Apply process that populates the internal CCD table runs asynchronously (for example, the Apply program might not be started or an event not triggered, and so on). The Apply program that replicates updates from the source table to the target table loops because it is waiting for the internal CCD table to be updated.

To stop the looping, start the Apply program (or trigger the event that causes replication) for the internal CCD table. The Apply program will populate the internal CCD table and allow the looping Apply program to process changes from

all source tables.

A similar situation could occur for a subscription set that contains source tables with internal CCD tables that are populated by multiple Apply programs.

Chapter 9. Capture and Apply for AS/400

在第 178 页上, "A note on work management" 应为如下内容:

You can alter the default definitions or provide your own definitions. If you create your own subsystem description, you must name the subsystem QZSDPR and create it in a library other than QDPR. See "OS/400 Work Management V4R3", SC41-5306 for more information about changing these definitions.

将以下内容添加至第 178 页的 "Verifying and customizing your installation of DB2 DataPropagator for AS/400":

If you have problems with lock contention due to high volume of transactions, you can increase the default wait timeout value from 30 to 120. You can change the job every time the Capture job starts or you can use the following procedure to change the default wait timeout value for all jobs running in your subsystem:

1. Issue the following command to create a new class object by duplicating QGPL/QBATCH:

```
CRTDUPOBJ OBJ(QBATCH) FROMLIB(QGPL) OBJTYPE(*CLS) TOLIB(QDPR) NEWOBJ(QZSDPR)
```

2. Change the wait timeout value for the newly created class (for example, to 300 seconds):

```
CHGCLS CLS(QDPR/QZSDPR) DFTWAIT(300)
```

3. Update the routing entry in subsystem description QDPR/QZSDPR to use the newly created class:

```
CHGRTGE SBS(QDPR/QZSDPR) SEQNBR(9999) CLS(QDPR/QZSDPR)
```

在第 194 页上, "Using the delete journal receiver exit routine" 应该包括下面这一句:

If you remove the registration for the delete journal receiver exit routine, make sure that all the journals used for source tables have DLTRCV(*NO).

在第 195 页, ADDEXITPGM 命令参数应为:

```
ADDEXITPGM EXITPNT(QIBM_QJO_DLT_JRNRCV)
          FORMAT(DRCV0100)
          PGM(QDPR/QZSDPREP)
          PGMNBR(*LOW)
          CRTEXITPNT(*NO)
          PGM(65535 10 QSYS)
```

Chapter 10. Capture and Apply for OS/390

在第 10 章中, 更新了下列段落:

Prerequisites for DB2 DataPropagator for OS/390

You must have DB2 for OS/390 Version 5, DB2 for OS/390 Version 6, or DB2 for OS/390 Version 7 to run DB2 DataPropagator for OS/390 Version 7 (V7).

UNICODE and ASCII Encoding Schemes on OS/390

DB2 DataPropagator for OS/390 V7 supports UNICODE and ASCII encoding schemes. To exploit the new encoding schemes, you must have DB2 for OS/390 V7 and you must manually create or convert your DB2 DataPropagator source, target, and control tables as described in the following sections. However, your existing replication environment will work with DB2 DataPropagator for OS/390 V7 even if you do not modify any encoding schemes.

Choosing an Encoding Scheme

If your source, CD, and target tables use the same encoding scheme, you can minimize the need for data conversions in your replication environment. When you choose encoding schemes for the tables, follow the single CCSID rule: Character data in a table space can be encoded in ASCII, UNICODE, or EBCDIC. All tables within a table space must use the same encoding scheme. The encoding scheme of all the tables in an SQL statement must be the same. Also, all tables that you use in views and joins must use the same encoding scheme.

If you do not follow the single CCSID rule, DB2 will detect the violation and return SQLCODE -873 during bind or execution. Which tables should be ASCII or UNICODE depends on your client/server configuration. Specifically, follow these rules when you choose encoding schemes for the tables:

- Source or target tables on DB2 for OS/390 can be EBCDIC, ASCII, or UNICODE. They can be copied from or to tables that have the same or different encoding scheme in any supported DBMS (DB2 family, or non-DB2 with DataJoiner).
- On a DB2 for OS/390 source server, all CD, UOW, register, and prune control tables on the same server must use the same encoding scheme. To ensure this consistency, always specify the encoding scheme explicitly.
- All the control tables (ASN.IBMSNAP_SUBS_xxxx) on the same control server must use the same encoding scheme.
- Other control tables can use any encoding scheme; however, it is recommended that the ASN.IBMSNAP_CRITSEC table remain EBCDIC.

Setting Encoding Schemes

To specify the proper encoding scheme for tables, modify the SQL that is used to generate the tables:

- Create new source and target tables with the proper encoding scheme, or change the encoding schemes of the existing target and source tables. It is recommended that you stop the Capture and Apply programs before you change the encoding scheme of existing tables, and afterwards that you could start the Capture program and restart the Apply program. To change the encoding scheme of existing tables:
 1. Use the Reorg utility to copy the existing table.
 2. Drop the existing table.
 3. Re-create the table specifying the new encoding scheme.
 4. Use the Load utility to load the old data into the new table.

See the *DB2 Universal Database for OS/390 Utility Guide and Reference* for more information on the Load and Reorg utilities.

- Create new control tables with the proper encoding scheme or modify the encoding scheme for existing ones.

DPCNTLMVS is shipped with DB2 for OS/390 in `sql1lib\samples\repl` and it contains several CREATE TABLE statements that create the control tables. For those tables that need to be ASCII or UNICODE (for example, ASN.IBMSNAP_REGISTER and ASN.IBMSNAP_PRUNCNTL), add the CCSID ASCII or CCSID UNICODE keyword, as shown in the following example.

```
CREATE TABLE ASN.IBMSNAP_PRUNCNTL (
  TARGET_SERVER      CHAR( 18)          NOT NULL,
  TARGET_OWNER      CHAR( 18)          NOT NULL,
  TARGET_TABLE      CHAR( 18)          NOT NULL,
  SYNCHTIME         TIMESTAMP,
  SYNCHPOINT        CHAR( 10)          FOR BIT DATA,
  SOURCE_OWNER      CHAR( 18)          NOT NULL,
  SOURCE_TABLE      CHAR( 18)          NOT NULL,
  SOURCE_VIEW_QUAL  SMALLINT           NOT NULL,
  APPLY_QUAL        CHAR( 18)          NOT NULL,
  SET_NAME          CHAR( 18)          NOT NULL,
  CNTL_SERVER       CHAR( 18)          NOT NULL,
  TARGET_STRUCTURE  SMALLINT           NOT NULL,
  CNTL_ALIAS        CHAR( 8)
) CCSID UNICODE
  DATA CAPTURE CHANGES
  IN TSSNAP02;
```

To modify existing control tables and CD tables, use the Reorg and Load utilities.

- When you create new replication sources or subscription sets, modify the SQL file generated by the administration tool to specify the proper encoding scheme. The SQL has several CREATE TABLE statements that are used to create the CD and target tables for the replication source and subscription set, respectively. Add the keyword CCSID ASCII or CCSID UNICODE where appropriate. 例如:

```
CREATE TABLE user1.cdtable1 (
  employee_name  varchar,
  employee_age   decimal
) CCSID UNICODE;
```

DB2 UDB OS/390 版 SQL Reference 中包含有关 CCSID 的详情。

Chapter 11. Capture and Apply for UNIX platforms

在 UNIX 和 Windows 上为 Capture 和 Apply 设置环境变量

如果使用代码页而不是缺省代码页值创建源数据库，则应将 DB2CODEPAGE 环境变量设置为该代码页。有关在设置 DB2CODEPAGE 之前派生代码页值的信息，参见 *DB2 管理指南*。Capture 必须与它正在捕捉其数据的数据库运行于同一代码页中。DB2 从 Capture 正运行于其中的活动环境中派生 Capture 代码页。如果未设置 DB2CODEPAGE，则 DB2 将从操作系统中派生代码页值。如果在创建数据库时使用了缺省代码页，则从操作系统中派生的值对于 Capture 来说是正确的。

Chapter 14. Table Structures

在第 339 页上，将以下句子追加至值 "2" 的 STATUS 列描述:

If you use internal CCD tables and you repeatedly get a value of "2" in the status column of the Apply trail table, go to "Chapter 8: Problem Determination" and refer to "Problem: The Apply program loops without replicating changes, the Apply trail table shows STATUS=2".

Chapter 15. Capture and Apply Messages

消息 ASN0017E 应为:

ASN0017E

The Capture program encountered a severe internal error and could not issue the correct error message. The routine name is "routine". The return code is "return_code".

应添加消息 ASN1027S:

ASN1027S

There are too many large object (LOB) columns specified. The error code is "<error_code>".

Explanation: Too many large object (BLOB, CLOB, or DBCLOB) columns are specified for a subscription set member. The maximum number of columns allowed is 10.

User response: Remove the excess large object columns from the subscription set member.

消息 ASN1048E 应为如下内容:

ASN1048E

The execution of an Apply cycle failed. See the Apply trail table for full details: "<text>"

Explanation: An Apply cycle failed. In the message, "<text>" identifies the "<target_server>", "<target_owner, target_table, stmt_number>", and "<cntl_server>".

User response: Check the APPERRM fields in the audit trail table to determine why the Apply cycle failed.

Appendix A. Starting the Capture and Apply Programs from Within an Application

在本书的第 399 页上, 在启动 Capture 和 Apply 程序的“样本”例程的注释中出现了一些错误; 但样本中的代码是正确的。样本中较后面的部分涉及 Apply 参数, 尽管注释指示它涉及 Capture 参数。

可在下列目录中获得 Apply 和 Capture API 的样本, 以及它们各自的 makefile。

对于 NT - sql1lib\samples\repl
对于 UNIX - sql1lib/samples/repl

System Monitor Guide and Reference

db2ConvMonStream

在 Usage Notes 中，快照变量数据流类型 SQLM_ELM_SUBSECTION 的结构应为 sqlm_subsection。

Troubleshooting Guide

在用户未登录的情况下在 Windows 95、Windows 98 和 Windows ME 上启动 DB2

要使 **db2start** 命令在 Windows 95、Windows 98 或 Windows Millennium Edition (ME) 环境中成功执行，您必须：

- 使用“Windows 登录”窗口或“Microsoft 联网登录”窗口进行登录
- 发出 **db2logon** 命令（参见注释第152页的1以了解有关 **db2logon** 命令的信息）。

此外，在登录期间指定的或是对 **db2logon** 命令指定的用户标识必须满足 DB2 的需求（参见注释第153页的2）。

当 **db2start** 命令启动时，它首先会查看用户是否已经登录。如果用户已经登录，则 **db2start** 命令将使用该用户标识。若用户未登录，则 **db2start** 命令将检查是否已经运行了 **db2logon** 命令，若已经运行了该命令，则 **db2start** 命令会使用为 **db2logon** 命令所指定的用户标识。如果 **db2start** 命令不能找到有效的用户标识，则命令会终止。

在 Windows 95、Windows 98 和 Windows ME 上安装 DB2 通用数据库 版本 7 期间，缺省情况下，当引导系统时，安装软件会为运行 **db2start** 命令的“启动”文件夹添加快捷方式（有关详情，参见注释 第152页的1）。如果系统的用户既未登录又未发出 **db2logon** 命令，则 **db2start** 命令将会终止。

如果您或您的用户未正常登录 Windows 或网络，则可通过从批处理文件运行命令以隐藏在运行 **db2start** 命令之前发出 **db2logon** 命令的需求，如下所示：

1. 创建一个批处理文件，其内容是发出 **db2logon** 命令，然后再发出 **db2start.exe** 命令。例如：

```
@echo off
db2logon db2local /p:password
db2start
cls
exit
```

2. 将该批处理文件命名为 **db2start.bat**，并将其存储在安装有 DB2 的驱动器和路径下的 **/bin** 目录中。将该批处理文件存储在此位置以确保操作系统可找到指向该批处理文件的路径。

安装有 DB2 的驱动器和路径存储在 DB2 注册表变量 **DB2PATH** 中。要查找安装有 DB2 的驱动器和路径，发出以下命令：

```
db2set -g db2path
```

假定 **db2set** 命令返回值 **c:\sql1lib**。在这种情况下，应将该批处理文件存储在如下目录中：

```
c:\sql1lib\bin\db2start.bat
```

3. 要在引导系统时启动 DB2，应从“启动”文件夹中的快捷方式运行批处理文件。您有两个选择：

- 将 DB2 安装程序创建的快捷方式修改为运行批处理文件而不是运行 **db2start.exe**。在前一示例中，现在快捷方式将运行 **db2start.bat** 批处理文件。

由 DB2 安装程序创建的快捷方式称为 **DB2 - DB2.lnk**，在大多数系统上，它都位于 `c:\WINDOWS\Start Menu\Programs\Start\DB2 - DB2.lnk` 中。

- 添加您自己的快捷方式以运行批处理文件，并删除由 DB2 安装程序添加的快捷方式。使用以下命令来删除 DB2 快捷方式：

```
del "C:\WINDOWS\Start Menu\Programs\Startup\DB2 - DB2.lnk"
```

如果决定使用您自己的快捷方式，应为快捷方式设置退出时关闭属性。如果未设置此属性，则即使在成功完成了 **db2start** 命令之后，DOS 命令提示符仍会留在任务栏中。要避免在 **db2start** 进程期间打开 DOS 窗口，可创建此快捷方式（以及它所运行的 DOS 窗口）集以将运行任务减至最少。

注：作为在系统引导期间启动 DB2 的备用方法，可在运行使用 DB2 的任何应用程序之前启动 DB2。参见注释第153页的5以了解详情。

如果使用批处理文件在运行 **db2start** 命令之前发出 **db2logon** 命令，且您的用户偶尔会进行登录，则 **db2start** 命令将继续工作，唯一的差别就是 DB2 将使用已登录的用户的用户标识。参见注释1以了解其他详细信息。

注：

1. **db2logon** 命令可模拟用户登录。**db2logon** 命令的格式为：

```
db2logon userid /p:password
```

对该命令指定的用户标识必须满足 DB2 命名需求（参见注释第153页的2以了解详情）。如果发出命令而无用户标识和口令，则一个窗口将会打开，提示用户输入用户标识和口令。如果提供的唯一参数是用户标识，则不会提示用户输入口令；在某些情况下会需要口令，如下面所述。

仅当用户未使用“Windows 登录”窗口或“Microsoft 联网登录”窗口时，才使用由 **db2logon** 命令设置的用户标识和口令值。若用户已经登录，并且已经发出了 **db2logon** 命令，则 **db2logon** 命令的用户标识用于所有 DB2 操作，但在 **db2logon** 命令上指定的口令会被忽略

当用户未使用“Windows 登录”窗口或“Microsoft 联网登录”窗口登录时，通过 **db2logon** 命令提供的用户标识和口令使用如下：

- **db2start** 命令会在启动时使用用户标识，而不需要口令。
- 在缺少某些操作（如创建表）的高级限定符的情况下，用户标识将用作高级限定符。例如：
 - a. 如果发出以下命令：`db2logon db2local`
 - b. 然后发出以下命令：`create table tab1`则以高级限定符创建的表将为 `db2local.tab1`。

应使用与表和其他对象的模式名完全相同的用户标识。

- 如果系统充当某服务器的客户机，且用户发出了不带用户标识和口令的 **CONNECT** 语句（例如 `CONNECT TO TEST`），并且认证被设置为服务器，则 **db2logon** 命令中的用户标识和口令将用于验证远程服务器上的用户。如果用户用显式的用户标识和口令进行连接（例如 `CONNECT TO TEST USER userID USING password`），则将使用对 **CONNECT** 语句指定的值。

2. 在版本 7 中，用来登录或为 **db2logon** 命令指定的用户标识必须满足下列 DB2 要求：
 - 它可以是下列任何一项：USERS、ADMINS、GUESTS、PUBLIC、LOCAL 或 *SQL Reference* 中列示的任何 SQL 保留字。
 - 它不能以 SQL、SYS 或 IBM 开头
 - 字符可包括：
 - A 到 Z（Windows 95、Windows 98 和 Windows ME 支持区分大小写的用户标识）
 - 0 至 9
 - @、# 或 \$
3. 在定制的交互式安装期间，或若正在执行响应文件安装且指定了 DB2.AUTOSTART=NO 选项的话，您可避免在“启动”文件夹中创建 **db2start** 快捷方式。如果使用这些选项，则“启动”文件夹中将不会有任何 **db2start** 快捷方式，且您必须添加自己的快捷方式以运行 **db2start.bat** 文件。
4. 在 **Windows 98** 和 **Windows ME** 上，提供了一个选项，您可用它来指定总是在 Windows 98 或 Windows ME 启动时进行登录的用户标识。在这种情况下，将不会出现“Windows 登录”窗口。如果您使用此选项，则如果用户标识满足 DB2 需求，就会登录用户，且 **db2start** 命令将成功（有关详细信息，参见注释 2）。如果不使用此选项，则总是有一个登录窗口显示给用户。如果用户取消了此窗口而未进行登录，则 **db2start** 命令将会失败，除非先前从批处理文件发出或调用了 **db2logon** 命令，如上所述。
5. 如果在系统引导期间未启动 DB2，则可由应用程序来启动 DB2。可将 **db2start.bat** 文件作为使用 DB2 的应用程序初始化的一部分来运行。通过使用这种方法，则仅当将要使用 DB2 的应用程序启动时才会启动 DB2。用户退出该应用程序时，可发出 **db2stop** 命令来停止 DB2。如果不在系统引导期间启动 DB2 的话，您的业务应用程序可以此方式来启动 DB2。

要使用 DB2 Synchronizer 应用程序或从应用程序中调用同步 API，必须启动 DB2（如果被下载以供执行的脚本包含要对本地实例或本地数据库执行的命令的话）。这些命令可处于数据库脚本、实例脚本中或嵌入在操作系统（OS）脚本中。如果 OS 脚本不包含使用实例或数据库的“命令行处理器”命令或 DB2 API，则它可在不启动 DB2 的情况下运行。因为在同步进程中，可能很难预先知道脚本中哪些命令将会进行，所以应在同步开始之前正常启动 DB2。

如果正在从应用程序中调用 **db2sync** 命令或同步 API，则应在应用程序初始化期间启动 DB2。如果您的用户将在“DB2 Windows 版”文件夹中使用 DB2 Synchronizer 快捷方式来启动同步，则必须修改“DB2 同步”快捷方式以运行 **db2sync.bat** 文件。批处理文件应包含以下命令以确保在同步开始之前 DB2 已在运行：

```
@echo off
db2start.bat
db2sync.exe
db2stop.exe
cls
exit
```

在此示例中，假定 **db2start.bat** 文件调用 **db2logon** 和 **db2start** 命令，如上所述。

如果决定在应用程序启动时启动 DB2，则确保 DB2 安装未将快捷方式添加至“启动”文件夹来启动 DB2。参见注释第153页的3以了解详细信息。

Chapter 2. Troubleshooting the DB2 Universal Database Server

在“Locking and Deadlocks”一节的“Applications Slow or Appear to Hang”的小节中，将“Lock waits or deadlocks are not caused by next key locking”下的描述更改为以下内容：

Next key locking guarantees Repeatable Read (RR) isolation level by automatically locking the next key for all INSERT and DELETE statements and the next higher key value above the result set for SELECT statements. For UPDATE statements that alter key parts of an index, the original index key is deleted and the new key value is inserted. Next key locking is done on both the key insertion and key deletion. It is required to guarantee ANSI and SQL92 standard RR, and is the DB2 default.

Examine snapshot information for the application. If the problem appears to be with next key locking, you can set the DB2_RR_TO_RS option on if none of your applications rely on Repeatable Read (RR) behavior and it is acceptable for scans to skip over uncommitted deletes.

When DB2_RR_TO_RS is on, RR behavior cannot be guaranteed for scans on user tables because next key locking is not done during index key insertion and deletion. Catalog tables are not affected by this option.

The other change in behavior is that with DB2_RR_TO_RS on, scans will skip over rows that have been deleted but not committed, even though the row may have qualified for the scan.

For example, consider the scenario where transaction A deletes the row with column1=10 and transaction B does a scan where column1>8 and column1<12.

With DB2_RR_TO_RS off, transaction B will wait for transaction A to commit or rollback. If it rolls back, the row with column1=10 will be included in the result set of transaction B's query.

With DB2_RR_TO_RS on, transaction B will not wait for transaction A to commit or rollback. It will immediately receive query results that do not include the deleted row.

Do not use this option if you require ANSI and SQL92 standard RR or if you do not want scans to skip uncommitted deletes.

在 64 位的平台上使用 “DB2 通用数据库”

第 5 章 配置

LOCKLIST

应该将以下信息添加到 “表 2” 中。

Parameter	Previous Upper Limit	Current Upper Limit
LOCKLIST	60000	524288

shmsys:shminfo_shmmax

64 位 Solaris 操作系统上的 DB2 用户应该在必要时增大 /etc/system 中的 “shmsys:shminfo_shmmax” 的值，以便能够分配大量的数据库共享内存集。DB2 (UNIX 版) 《快速入门》一书建议将该参数设置为 “机器中 90% 的物理 RAM (以字节计)”。此建议对 64 位实现方法也有效。

然而，在 DB2 (UNIX 版) 《快速入门》一书中，下列建议有问题：对于具有多于 4 GB RAM 的 32 位系统（在 Solaris 操作系统中，最多可以有 64 GB），若用户将 shmmax 值设置为大于 4 GB 的数，并且正在使用 32 位内核，则内核仅查找较低的 32 位数，有时会导致 shmmax 的值相当小。

第 6 章 限制

当前在 64 位操作系统上没有 LDAP 支持。

不能在同一路径上创建 32 位和 64 位数据库。例如，若 <somepath> 上存在 32 位数据库，则发出：

```
db2 create db <somepdb> on <somepath>
```

若从 64 位实例中发出此命令，则会失败，并出现信息 “SQL10004C 访问数据库目录时发生 I/O 错误。”

XML Extender 管理和编程

可以在 DB2 XML Web 站点上找到 IBM DB2 XML Extender 的“发行说明”，网址为：<http://www-4.ibm.com/software/data/db2/extenders/xmlext/library.html>

MQSeries

本节描述可如何使用 DB2 和 MQSeries 来构造用来进行消息传递和数据库访问的应用程序。本节的重点是可选择在 “DB2 通用数据库版本 7.2” 中启用的一组函数，类似于 “用户定义函数” (UDF)。使用这些基本函数，就可以支持大范围的应用程序，从简单事件通知到数据入库。

有关数据入库应用程序的详情，参见第209页的『MQSeries 与 “数据仓库中心” 集成』

DB2 MQSeries 函数的安装和配置

本节描述如何配置 DB2 环境以使用 “DB2 MQSeries 函数”。成功完成下列过程后，就可以在 SQL 中使用 “DB2 MQSeries 函数”。这些函数的描述可在《发行说明》的 SQL Reference 一节中找到。附加信息（包含最新的文档、提示和技巧）可在以下网址找到：<http://www.ibm.com/software/data/integration/MQSeries>。

配置和启用 “DB2 MQSeries 函数” 的基本过程是：

1. 安装 MQSeries。
2. 安装 MQSeries AMI。
3. 启用和配置 “DB2 MQSeries 函数”。

此外，要利用 “DB2 MQSeries 函数” 提供的发布 / 预订功能，还必须安装 “MQSeries 集成器” 或 “MQSeries 发布 / 预订函数”。有关 “MQSeries 集成器” 的信息，可在以下网址找到：<http://www.ibm.com/software/ts/mqseries/integrator>。有关 “MQSeries 发布 / 预订” 功能部件的信息，可在类别 3 中的 <http://www.ibm.com/software/ts/mqseries/txppacs> 中找到。

安装 MQSeries

第一步是确保 DB2 服务器上安装了 MQSeries 版本 5.2。如果已安装了此版本的 MQSeries，则跳至下一步 “安装 MQSeries AMI”。DB2 版本 7.2 包括要与 DB2 配合使用的 MQSeries 服务器。有关安装 MQSeries 或升级现存 MQSeries 安装的特定于平台的指导可在以下网址处的特定于平台的 “快速入门” 中找到：

<http://www.ibm.com/software/ts/mqseries/library/manuals>。在进行安装过程时，一定要设置缺省队列管理器。

安装 MQSeries AMI

下一步就是安装 “MQSeries 应用程序消息传递接口” (AMI)。这是对让管理和编程任务完全分离的 MQSeries 编程接口的扩展。“DB2 MQSeries 函数” 需要安装此接口。如果已在 DB2 服务器上安装了 MQSeries AMI，则跳至下一步 “启用 DB2 MQSeries 函数”。如果未安装 MQSeries AMI，则可通过随 DB2 7.2 提供的安装软件包或通过从 MQSeries Support Pacs Web 站点（网址为 <http://www.ibm.com/software/ts/mqseries/txppacs>）下载 AMI 的副本来完成此操作。AMI 可在 “第 3 类 - 产品扩展功能” 中找到。为方便起见，我们在提供 DB2 的同时也提供了 MQSeries AMI 的副本。此文件在 `sql1lib/cfg` 目录中。该文件的名称是与操作系统有关的：

AIX 版本 4.3 和更高版本	maOf_ax.tar.Z
HP-UX	maOf_hp.tar.Z
Solaris 操作环境	maOf_sol7.tar.Z
Windows 32 位	maOf_nt.zip

遵循 AMI 自述文件（包含在压缩安装映像中）中所概述的正常 AMI 安装过程。

启用 DB2 MQSeries 函数

在此步骤中，您将对“DB2 MQSeries 函数”配置和启用数据库。**enable_MQFunctions** 实用程序是一个非常灵活的命令，它首先会检查是否设置了正确的 MQSeries 环境，然后为 DB2 MQSeries 函数创建缺省配置，启用带有这些函数的指定数据库，并确认配置是否生效。

1. 对于 Windows NT 或 Windows 2000，转至步骤 5。
2. 在 UNIX 上设置组：如果正在 UNIX 上启用这些函数，则必须首先将 DB2 实例所有者（通常是 db2inst1）和与受防护 UDF（通常是 db2fenc1）相关联的用户标识添加至 MQSeries 组 mqm。这对于 DB2 函数访问 MQSeries 是必需的。
3. 在 UNIX 上设置“DB2 环境变量”：将 AMT_DATA_PATH 环境变量添加至 DB2 理解的列表。可编辑文件 \$INSTHOME/sqllib/profile.env，将 AMT_DATA_PATH 添加至 DB2ENVLIST。还可使用 **db2set** 命令。
4. 在 UNIX 上，重新启动数据库实例：要使环境变量更改生效，必须重新启动数据库实例。
5. 将目录切换至 \$INSTHOME/sqllib/cfg（对于 UNIX）或 %DB2PATH%/cfg（在 Windows 上）。
6. 运行命令 **enable_MQFunctions** 以对“DB2 MQSeries 函数”配置和启用数据库。参考第168页的『enable_MQFunctions』以获取此命令的完整描述。下面给出了一些常见示例。成功完成后，将启用指定数据库并测试配置。
7. 要使用“命令行处理器”测试这些函数，在连接至启用的数据库后发出下列命令：

```
values DB2MQ.MQSEND('a test')
values DB2MQ.MQRECEIVE()
```

第一个语句会将消息“a test”发送至 DB2MQ_DEFAULT_Q 队列，而第二个语句将接收回该消息。

注：作为运行 **enable_MQFunctions** 的结果，将会建立缺省 MQSeries 环境。将创建 MQSeries 队列管理器 DB2MQ_DEFAULT_MQM 和缺省队列 DB2_DEFAULT_Q。如果 AMT_DATA_PATH 所指向的目录中不存在文件 amt.xml、amthost.xml 和 amt.dtd 的话，将会创建这些文件。如果 amthost.xml 文件不存在，且不包含对 connectionDB2MQ 的定义，则会对该文件添加一行，以指出适当的信息。原始文件的副本将另存为 DB2MQSAVE.amthost.xml。

MQSeries 消息传递样式

DB2 MQSeries 函数支持三种消息传递模型：数据报、发布 / 预订 (p/s) 和请求 / 答复 (r/r)。

作为数据报发送的消息会被发送至单个目的地，且没有答复。在 p/s 模型中，一个或多个发布方将一条消息发送至发布服务，而该发布服务将该消息分发给一个或多个订户。请求 / 答复类似于数据报，但发送方将接收响应。

消息结构

MQSeries 自身不会要求或支持它传送的消息的任何特定结构。

其他产品（如“MQSeries 集成器”（MQSI））不支持以 C 或 Cobol 又或是以 XML 字符串组成的消息。MQSI 中的结构化消息是由消息库定义的。XML 消息通常是自述消息结构，也可能是通过库管理的。消息也可能是未组织的，需要用户代码来分析或构造消息内容。这种消息通常是准结构化的，即，它们使用字节位置或固定定界符来隔开消息内的各个字段。“MQSeries 辅助程序向导”提供了对这种准结构化的消息的支持。对 XML 消息的支持是通过 DB2 XML Extender 的某些新增功能部件提供的。

MQSeries 函数概述

DB2 UDB 版本 7.2 附带提供了一组 MQSeries 函数，允许 SQL 语句包括消息传递操作。这表示此支持可用于以任何受支持的语言（例如 C、Java 和使用任何数据库接口的 SQL）编写的应用程序。下面显示的所有示例都是以 SQL 编写的。可以任何标准方式通过其他编程语言来使用此 SQL。上述所有 MQSeries 消息传递样式都是受支持的。有关 MQSeries 函数的详情，参见《发行说明》中的 SQL Reference 一节。

在基本配置中，MQSeries 服务器位于带有 DB2 的数据库服务器的机器上。MQSeries 函数被安装至 DB2，并提供对 MQSeries 服务器的访问。DB2 客户机可能在 DB2 服务器可访问到的任何机器上。多台客户机可通过数据库并行访问 MQSeries 函数。通过提供的这些函数，DB2 客户机可执行 SQL 语句内的消息传递操作。这些消息传递操作允许 DB2 应用程序之间进行通信，或与其他 MQSeries 应用程序通信。

enable_MQFunctions 命令用来对 MQSeries 函数启用 DB2 数据库。它将自动建立简单的缺省配置，以便客户机应用程序只将其与管理操作配合使用。有关描述，参见 enable_MQFunctions 和 disable_MQFunctions。缺省配置使得应用程序程序员能够迅速入门，且有较为简单的接口来进行开发。可根据需要增加配置其他功能。

示例 1: 要使用缺省配置发送简单的消息，SQL 语句应为:

```
VALUES DB2MQ.MQSEND('simple message')
```

这会将消息 simple message 发送至缺省配置指定的 MQSeries 队列管理器和队列。

MQSeries 的“应用程序消息传递接口”（AMI）使得消息传递操作和指示应如何执行这些操作的定义完全分离。这些定义被保存在外部库文件中，且使用“AMI 管理”工具进行管理。这使得 AMI 应用程序易于开发和维护。随 DB2 提供的 MQSeries 函数是以 AMI MQSeries 接口为基础的。AMI 支持使用外部配置文件（称为“AMI 库”）来存储配置信息。缺省配置包括为与 DB2 配合使用而配置的“MQSeries AMI 库”。

MQSeries AMI 中的两个关键概念服务点和策略被引入 DB2 MQSeries 函数。服务点是一个逻辑端点，可在其中发送或接收消息。在 AMI 库中，每个服务点都是使用 MQSeries 队列名和队列管理器定义的。策略定义应该用于给定消息传递操作的服务质量

选项。关键服务质量包括消息优先级和持久性。系统提供了缺省服务点和策略定义，开发者可使用它们来更进一步地简化他们的应用程序。可按如下所示重写示例 1 以显式指定缺省服务点和策略名：

示例 2:

```
VALUES DB2MQ.MQSEND('DB2.DEFAULT.SERVICE', 'DB2.DEFAULT.POLICY', 'simple message')
```

在队列和应用程序所驻留的服务器上，队列可供一个或多个应用程序使用。在许多配置中，可定义多个队列以支持不同的应用程序和目的。因此，在提出 MQSeries 请求时定义不同服务点是非常重要的。这在以下示例中作了演示：

示例 3:

```
VALUES DB2MQ.MQSEND('ODS_Input', 'simple message')
```

注：在此示例中，未定义策略，因此将使用缺省策略。

限制

MQSeries 允许将消息操作和数据库操作组合到单个工作单元中作为原子事务。最初在 UNIX 和 Windows 上，MQSeries 不支持此功能部件。

在使用发送或接收函数时，消息的最大长度是 4000 个字符。在使用 MQPublish 发布消息时，消息的最大长度也是 4000 个字符。

错误码

“MQSeries 函数”返回的返回码可在《MQSeries 应用程序消息传递接口手册》的『附录 B』中找到。

使用方案

可在各种方案中使用“MQSeries 函数”。本节将复查一些较常见的方案，包括“基本消息传递”、“应用程序连通性”和“数据发布”。

基本消息传递

在所有数据库应用程序连接至同一 DB2 服务器时，会使用以“MQSeries DB2 函数”进行消息传递的最基本形式。客户机对于数据库服务器可以是本地的，也可以分布在网络环境中。

在简单方案中，“客户机 A”调用 MQSEND 函数将用户定义字符串发送至缺省服务位置。然后，会在数据库服务器上的 DB2 内执行 MQSeries 函数。稍后，“客户机 B”会调用 MQRECEIVE 函数来除去缺省服务定义的队列的顶部的消息，并将其返回至客户机。DB2 会再一次地执行 MQSeries 函数以进行此工作。

数据库客户机可以多种方式使用简单消息传递。消息传递的某些常见用法包括：

- 数据收集 -- 从一个或多个可能的不同信息源中以消息形式接收信息。信息源可以是商业应用程序（如 SAP），也可以是室内开发的应用程序。可从队列中接收这种数据，并存储在数据库表中以进行进一步的处理或分析。

- 工作负荷分发 -- 工作请求被发送至由同一应用程序的多个实例共享的队列。在实例准备执行某些工作时，它从包含要执行的工作请求的队列顶部接收到一条消息。使用此技术，多个实例可共享由单个人池请求队列所表示的工作负荷。
- 应用程序信号发送 -- 几个进程协作时，消息通常用来协调它们的工作。这些消息可能包含执行工作的命令或请求。通常，这种信息发送是单向的；即，启动消息的一方不会得到答复。参见第165页的『请求 / 答复通信』以获取详情。
- 应用程序通知 -- 通知类似于信息发送，因为数据是从启动器发送出的，且没有响应。但是，通常通知包含有关已发生的商业事件的数据。第166页的『发布 / 预订』是通知的较高级的形式。

以下方案扩展上述简单方案以合并远程消息传递。即，在“机器 A”和“机器 B”之间发送消息。步骤的顺序如下所示：

1. “DB2 客户机”执行 MQSEND 调用，指定已定义为在“机器 B”上表示远程队列的目标服务。
2. MQSeries DB2 函数执行实际的 MQSeries 工作以发送消息。“机器 A”上的 MQSeries 服务器接受该消息并保证它会将该消息发送至服务点定义和“机器 A”的当前配置定义的目的地。服务器确定这是“机器 B”上的队列。然后，它会尝试将该消息发送至“机器 B”上的 MQSeries 服务器，在需要时显式重试。
3. “机器 B”上的 MQSeries 服务器从“机器 A”上的服务器接受该消息，并将其放在“机器 B”上的目标队列中。
4. “机器 B”上的 MQSeries 客户机请求队列顶部的消息。

发送消息

使用 MQSEND，DB2 用户或开发者选择要发送什么数据，要发送到哪里，以及何时发送。在业界，这通常称之为“发送并遗忘”，表示发送者只根据 MQSeries 的承诺发送协议发送消息，以确保该消息到达目的地。以下示例演示了此过程。

示例 4: 要将用户定义字符串发送至带有策略 highPriority 的服务点 myplace:

```
VALUES DB2MQ.MQSEND('myplace','highPriority','test')
```

此处的策略 highPriority 指的是在“AMI 库”中定义的策略，它将 MQSeries 优先级设置为最高级别，并可能会调整其他服务质量，如持久性，等等。

消息内容可能由 SQL 和用户指定数据的任何合法组合组成。这包括嵌套函数、运算符和强制转型。例如，假定有一个表 EMPLOYEE，带有 VARCHAR 列 LASTNAME、FIRSTNAME 和 DEPARTMENT，要对 DEPARTMENT 5LGA 中的每个雇员发送包含此信息的消息，应执行下列操作：

示例 5:

```
SELECT DB2MQ.MQSEND(LASTNAME || ' ' || FIRSTNAME || ' ' || DEPARTMENT)
FROM EMPLOYEE
WHERE DEPARTMENT = '5LGA'
```

如果此表还包含整数 AGE 列，还应包括下列各项：

示例 6:

```
SELECT DB2MQ.MQSEND(LASTNAME || ' ' || FIRSTNAME || ' ' || DEPARTMENT || ' ' || char(AGE))
FROM EMPLOYEE
WHERE DEPARTMENT = '5LGA'
```

最后，以下示例显示如何使用任何有效的 SQL 表达式派生消息内容。假定有一个辅助表 DEPT 包含 varchar 列 DEPT_NO 和 DEPT_NAME，可发送包含雇员 LASTNAME 和 DEPT_NAME 的消息：

示例 7:

```
SELECT DB2MQ.MQSEND(e.LASTNAME || ' ' || d.DEPTNAME) FROM EMPLOYEE e, DEPT d
WHERE e.DEPARTMENT = d.DEPTNAME
```

检索消息

“MQSeries DB2 函数”允许接收或读取消息。读取和接收的区别在于读取会在队列顶部返回该消息而不会从队列中除去它，而接收操作会从队列中除去该消息。使用接收操作检索的消息只能被检索一次，而使用读取操作检索的消息允许同一消息被检索多次。以下示例演示了此过程：

示例 8:

```
VALUES DB2MQ.MQREAD()
```

此示例在由缺省服务使用缺省服务质量策略定义的队列顶部返回包含该消息的 VARCHAR 字符串。非常重要的一点就是要注意，如果没有任何消息可供读取，将返回空值。此操作不会更改队列。

示例 9:

```
VALUES DB2MQ.MQRECEIVE('Employee_Changes')
```

上述示例显示如何从由 Employee_Changes 服务使用缺省策略定义的队列顶部除去消息。

DB2 有一个非常强大的功能部件能够通过用户定义（或 DB2 提供的）函数生成表。可使用此表函数功能部件来允许队列的内容具体化为一个 DB2 表。以下示例演示了此过程的最简单形式：

示例 10:

```
SELECT t.* FROM table ( DB2MQ.MQREADALL() ) t
```

此查询返回由缺省服务定义的队列中的所有消息和有关这些消息的元数据组成的表。虽然返回的表结构的完整定义是在“附录”中定义的，但第一列仍然会返回该消息的内容和包含元数据的其余列。要想只返回这些消息，示例应重写为：

示例 11:

```
SELECT t.MSG FROM table (DB2MQ.MQREADALL() ) t
```

表函数返回的表与从数据库直接检索的表完全相同。这表示可以各种各样的方式来使用此表。例如，可将一个表的内容与另一个表连在一起或计算队列中消息的数目：

示例 12:

```
SELECT t.MSG, e.LASTNAME
FROM table (DB2MQ.MQREADALL() ) t, EMPLOYEE e
WHERE t.MSG = e.LASTNAME
```

示例 13:

```
SELECT COUNT(*) FROM table (DB2MQ.MQREADALL() ) t
```

还可隐藏表的源是根据表函数创建视图的队列的事实。例如，以下示例根据服务 NEW_EMPLOYEES 引用的队列创建称为 NEW_EMP 的视图：

示例 14:

```
CREATE VIEW NEW_EMP (msg) AS
  SELECT t.msg FROM table (DB2MQ.MQREADALL()) t
```

在此示例中，视图是使用包含整条消息的单个列定义的。如果只是简单组织了消息（例如包含两个固定长度的字段），它会直接使用 DB2 内置函数来将该消息分为两列。例如，如果知道发送至特定列的消息总是包含一个 18 字符的姓，且后跟 18 字符的名，则可定义一个视图，将每个字段作为单独的列来包含，如下所示：

示例 15:

```
CREATE VIEW NEW_EMP2 AS
  SELECT left(t.msg,18) AS LNAME, right(t.msg,18) AS FNAME
  FROM table(DB2MQ.MQREADALL()) t
```

“DB2 存储过程构建器”的新功能部件“MQSeries 辅助程序向导”可用来创建新的 DB2 表函数和视图，它们会将定界消息结构映射为列。

最后，将一条或多条消息存储到数据库中通常是最好的。这可通过使用 SQL 全权处理并存储消息内容来完成。此过程的可能的最简单示例为：

示例 16:

```
INSERT INTO MESSAGES
  SELECT t.msg FROM table (DB2MQ.MQRECEIVEALL()) t
```

假定有一个表 MESSAGES，带有一列 VARCHAR(2000)，上述语句会将这些消息从缺省服务队列插入表。可利用此技术来应付各种各样的情况。

应用程序间的连通性

应用程序集成是许多解决方案中的一个常见元素。无论是将购买的应用程序集成到现存基本设施中，还是仅将新开发的应用程序集成到现存环境中，我们通常会面对的任务是将不同集合的子系统连接到一起以作为一个整体来运作。MQSeries 通常被视作集成应用程序的基本工具。在大部分硬件、软件和语言环境中都可使用 MQSeries，它提供了非常不同的集合的应用程序互连的方式。

本节将讨论某些应用程序集成方案以及它们如何与 DB2 配合使用。因为主题非常广泛，所以“应用程序集成”的完善处理超出了此工作的范围之外。因此，重点只放在两个简单的主题上：“请求 / 答复通信”，以及“MQSeries 集成器和发布 / 预订”。

请求 / 答复通信

对于一个应用程序请求另一个应用程序的服务，“请求 / 答复” (R/R) 通信方法是非常常见的技术。完成此任务的一个方法就是让请求者向服务供应商发送一条消息，请求执行一些工作。一旦工作完成，供应商可能会决定将结果（或只是完成的确认信息）发送回请求者。但使用上述的基本消息传递技术，与将发送方的请求与服务供应商的响应连接在一起并无区别。除非请求者要等到答复才继续进行，否则必须使用一些机制来将每个答复与其请求相关联。MQSeries 并不强制开发者创建这些机制，它提供了相关标识，允许交换中的消息相关。

虽然可以多种方法来使用此机制，但最简单的方法是让请求者以使用如下所示的已知相关标识符来标记消息：

示例 17:

```
DB2MQ.MQSEND ('myRequester','myPolicy','SendStatus:cust1','Req1')
```

此语句将最终参数 Req1 添加至以上的 MQSEND 语句，以指示请求的相关标识。

要接收对此特定请求的答复，使用相应的 MQRECEIVE 语句来有选择地检索指定服务定义的，与相关标识相匹配的第一条消息，如下所示：

示例 18:

```
DB2MQ.MQRECEIVE('myReceiver','myPolicy','Req1')
```

如果响应请求的应用程序正忙，且请求者在答复送出之前发出了上述 MQRECEIVE，则表明找不到匹配此相关标识的任何消息。

要接收服务请求和相关标识，使用类似如下的语句：

示例 19:

```
SELECT msg, correlid FROM table (DB2MQ.MQRECEIVEALL('aServiceProvider','myPolicy',1)) t
```

这会返回来自服务 aServiceProvider 的第一个请求的消息和相关标识符。

一旦执行了服务，它就会将答复消息发送至 aRequester 所描述的队列。其间，服务请求者可能正进行其他工作。实际上，并不能保证会在设置时间内响应服务请求。这种应用程序级别超时必须由开发者管理；请求者必须进行轮询以检测是否有答复。

这种与时间无关的异步处理的优点就在于请求者和服务供应商在执行时是完全独立的。它可用来配合仅间歇连接应用程序的环境以及在聚集多个请求或答复之后才进行处理的多个面向批处理的环境。通常在数据仓库环境中使用这种聚合以定期更新数据仓库或操作性数据仓库。

发布 / 预订

简单数据发布：应用程序集成的另一常见方案是让一个应用程序通知其他应用程序所关心的事件。可通过将一条消息发送至由另一应用程序监控的队列很容易地完成此操作。消息的内容可以是用户定义字符串，或者，可由数据库列组成。通常，只需要使用 MQSEND 函数发送简单的消息。如果需要将这种消息并行发送至多个收件人，可使用 MQSeries AMI 的“分发列表”设施。

分发列表是使用“AMI 管理”工具定义的。“分发列表”包含个别服务的列表。发送至分发列表的消息被转发至列表内定义的每个服务。这在已知几个服务始终对每条消息都有兴趣的情况下特别有用。以下示例显示了将消息发送至分发列表 interestedParties 的过程：

示例 20:

```
DB2MQ.MQSEND('interestedParties','information of general interest');
```

如果需要对特定服务应接收的消息进行更多的控制，则需要“发布 / 预订”功能。

“发布 / 预订”系统通常会提供可伸缩的安全的环境，许多订户都可在其中进行注册

以从多个发布者接收消息。要支持此功能，可将 MQPublish 接口与“MQSeries 集成器”或“MQSeries 发布/预订”设施配合使用。

MQPublish 允许用户选择指定要与消息相关联的主题。主题允许订户更明确地指定要接受的消息。步骤序列如下所示：

1. MQSeries 管理员配置“MQSeries 集成器”发布/预订功能。
2. 感兴趣的应用程序预订至 MQSI 配置定义的预订点，可选择对其感兴趣的主体。每个订户选择相关主题，还可利用“MQSeries 集成器 V2”的基于内容的预订技术。很重要的一点就是要注意到，队列（由服务名表示）定义订户。
3. DB2 应用程序将消息发布至服务点“天气”。该消息指示天气是“雨加雪”（带有主题 Austin），并从而通知感兴趣的订户 Austin 的天气是“雨加雪”。
4. 实际发布消息的机制是由 DB2 提供的 MQSeries 函数处理的。使用服务“天气”将该消息发送至“MQSeries 集成器”。
5. MQSI 从“天气”服务接受该消息，执行 MQSI 配置定义的任何处理，并确定它满足哪些预订。然后，MQSI 将该消息转发至它满足其标准的订户队列。
6. 已预订至“天气”服务且注册为对 Austin 有兴趣的应用程序将在其接收服务中接收消息雨加雪。

要使用所有缺省值和空主题发布此数据，应使用以下语句：

示例 21:

```
SELECT DB2MQ.MQPUBLISH(LASTNAME || ' ' || FIRSTNAME || ' ' || DEPARTMENT || ' ' || char(AGE))
FROM EMPLOYEE
WHERE DEPARTMENT = '5LGA'
```

完整指定所有参数，并将该消息简化为只包含 LASTNAME，该语句将类似如下所示：

示例 22:

```
SELECT DB2MQ.MQPUBLISH('HR_INFO_PUB', 'SPECIAL_POLICY', LASTNAME,
'ALL_EMP:5LGA', 'MANAGER')
FROM EMPLOYEE
WHERE DEPARTMENT = '5LGA'
```

此语句将使用 SPECIAL_POLICY 服务将消息发布至 HR_INFO_PUB 发布服务。这些消息指示发送方是 MANAGER 主题。主题字符串演示多个主题，可指定使用 ':' 将它们连接起来。在此示例中，使用两个主题允许订户向 ALL_EMP 注册或仅向 5LGA 注册以接收这些消息。

要接收发布的消息，必须首先注册对包含给定主题的消息的兴趣，并指示应将消息发送至的订户服务的名称。很重要的一点就是要注意到，AMI 订户服务定义代理服务和接收方服务。代理服务是订户与“发布/预订”代理进行通信的方式，而接收方服务是要将与预订请求相匹配的消息发送至的位置。以下语句注册对主题 ALL_EMP 的兴趣。

示例 23:

```
DB2MQ.MQSUBSCRIBE('aSubscriber', 'ALL_EMP')
```

一旦应用程序已预订，使用主题 ALL_EMP 发布的消息将转发至由订户服务定义的接收方服务。应用程序可有多个并行预订。要获取满足预订的消息，可使用任何标准消息检索函数。例如，如果订户服务 aSubscriber 将接收方服务定义为 aSubscriberReceiver，则以下语句将以非破坏的形式读取第一条消息：

示例 24:

```
DB2MQ.MQREAD('aSubscriberReceiver')
```

要确定消息和在其中发布这些消息的主题，应使用其中一个表函数。以下语句将从 aSubscriberReceiver 接收头五条消息，并显示消息和主题:

示例 25:

```
SELECT t.msg, t.topic FROM table (DB2MQ.MQRECEIVEALL('aSubscriberReceiver',5)) t
```

要读取带有主题 ALL_EMP 的所有消息，可利用 SQL 的功能来发出:

示例 26:

```
SELECT t.msg FROM table (DB2MQ.MQREADALL('aSubscriberReceiver')) t
WHERE t.topic = 'ALL_EMP'
```

注: 很重要的一点就是要认识到，如果 MQRECEIVEALL 在使用时带有约束，则会影响整个队列，而不仅仅是那些使用主题 ALL_EMP 发布的消息。这是因为先执行了表函数，再才应用约束。

如果不再对预订至特定主题感兴趣，则必须使用如下语句显式取消预订:

示例 27:

```
DB2MQ.MQUNSUBSCRIBE('aSubscriber', 'ALL_EMP')
```

一旦发出了此语句，发布 / 预订代理就不再发送与此预订相匹配的消息。

自动发布: 数据库消息传递的另一重要技术就是自动发布。在 DB2 内使用触发器设施，可作为触发器调用的一部分来自动发布消息。虽然对于自动数据发布还存在其他技术，但基于触发器的方法允许管理员或开发者在构造消息内容方面有极大的自由，且在触发器操作方面有着很高的灵活性。就象使用任何触发器一样，必须注意执行的频率和成本。下列示例演示了如何将触发器与“MQSeries DB2 函数”配合使用。

以下示例显示每次雇用一个新员工时发布一条消息是很容易的。预订至 HR_INFO_PUB 服务且注册了对 NEW_EMP 的兴趣的任何用户或应用程序都将接收包含每个新员工的日期、名称和部门的消息。

示例 28:

```
CREATE TRIGGER new_employee AFTER INSERT ON employee REFERENCING NEW AS n
FOR EACH ROW MODE DB2SQL
VALUES DB2MQ.MQPUBLISH('HR_INFO_PUB&', 'NEW_EMP',
current date || ' ' || LASTNAME || ' ' || DEPARTMENT)
```

enable_MQFunctions

enable_MQFunctions

对指定数据库启用 DB2 MQSeries 函数，并验证是否正确执行了 DB2 MQSeries 函数。如果未正确安装和配置 MQSeries 和 MQSeries AMI，命令将会失败。

权限

下列其中一项:

- *sysadm*
- *dbadm*
- 在数据库上为 `IMPLICIT_SCHEMA`（如果函数的隐式或显式模式不存在的话）
- 对模式具有 `CREATEIN` 特权（如果模式名 `DB2MQ` 存在的话）

命令语法

```
▶▶ enable_MQFunctions -n database -u userid -p password [force]
[noValidate]
```

命令参数

-n database

指定要启用的数据库的名称。

-u userid

指定要用来连接至数据库的用户标识。

-p password

指定用户标识的口令。

-force 指定重新安装期间遇到的警告应被忽略。

-noValidate

指定将不验证 DB2 MQSeries 函数。

示例

在以下示例中，将创建 `DB2MQ` 函数。用户连接至数据库 `SAMPLE`。使用缺省模式 `DB2MQ`。

```
enable_MQFunctions -n sample -u user1 -p password1
```

使用说明

DB2 MQ 函数按此命令自动创建的模式 `DB2MQ` 运行。

在执行此命令之前:

- 确保已安装了 MQ 和 AMI，则 MQSeries 的版本为 5.2 或更高版本。
- 确保已经定义了环境变量 `$AMT_DATA_PATH`。
- 将目录切换至 `DB2PATH` 的子目录 `cfg`

在 UNIX 上:

- 使用 `db2set` 以将 `AMT_DATA_PATH` 添加至 `DB2ENVLIST`。

- 确保与 UDF 执行相关联的用户帐户是 mqm 组的成员。
- 确保将调用此命令的用户是 mqm 组的成员。

注: MQSeries 5.2 不支持 AIX 4.2。

disable_MQFunctions

disable_MQFunctions

对指定数据库禁用 DB2 MQSeries 函数。

权限

下列其中一项:

- *sysadm*
- *dbadm*
- 在数据库上为 **IMPLICIT_SCHEMA** (如果函数的隐式或显式模式不存在的话)
- 对模式具有 **CREATEIN** 特权 (如果模式名 **DB2MQ** 存在的话)

命令语法

►► disable_MQFunctions -n *database* -u *userid* -p *password* ◀◀

命令参数

-n database

指定数据库的名称。

-u userid

指定要用来连接至数据库的用户标识。

-p password

指定用户标识的口令。

示例

在以下示例中, 对数据库 **SAMPLE** 禁用 **DB2MQ** 函数。

```
disable_MQFunctions -n sample -u user1 -p password1
```


第4部分 管理工具

控制中心	175
管理 DB2 服务器 VSE 版和 VM 版服务器的能力	175
Java 1.2 支持“控制中心”	175
在 Windows 操作系统上使用联机帮助时发生“快捷方式无效”错误	176
OS/2 上的 Java 控制中心	176
试图在 Windows 操作系统上的日志中查看已完成作业时发生“拒绝访问文件”错误	176
多站点更新测试连接	176
“DB2 OS/390 版”的“控制中心”	176
对“控制中心 OS/390 版”的必要修正	178
对“创建空间层”对话框的更改	178
“DB2 控制中心”的故障诊断信息	178
基于 UNIX 的系统上的控制中心故障诊断	178
OS/2 上可能的 Infopops 问题	178
jdk11_path 配置参数的帮助	179
在使用“脚本中心”或“日志”时出现的 Solaris 系统错误 (SQL10012N)	179
DPREPL.DFT 文件的帮助	179
启动多个控制中心小应用程序	179
关于作为小应用程序运行的“控制中心”的联机帮助	179
以小应用程序方式运行“控制中心”(Windows 95)	180
使用大型查询结果	180
信息中心	181
在 Windows 操作系统上出现“快捷方式无效”错误	181
当 Netscape 已打开时, 打开 Netscape Navigator 中的外部 Web 链接 (基于 UNIX 的系统)	181
启动“信息中心”时的问题	181
向导	183
在创建数据库向导中设置范围大小	183
“MQSeries 辅助程序”向导	183
“OLE DB 辅助程序”向导	183

控制中心

管理 DB2 服务器 VSE 版和 VM 版服务器的能力

“DB2 通用数据库版本 7 控制中心”已经增强了它对 DB2 服务器 VSE 版和 VM 版数据库的支持。所有“DB2 服务器 VSE 版和 VM 版”数据库对象都可通过“控制中心”查看。它还支持 CREATE INDEX、REORGANIZE INDEX 和 UPDATE STATISTICS 语句以及 REBIND 命令。REORGANIZE INDEX 和 REBIND 需要一个在 DB2 服务器 VSE 版和 VM 版主机上运行的存储过程。此存储过程是由“DB2 服务器 VSE 版和 VM 版”的“控制中心 VSE 版和 VM 版”功能部件提供的。

完全集成“控制中心”允许用户管理 DB2，而不管 DB2 服务器运行于何种平台之上。

“DB2 服务器 VSE 版和 VM 版”对象与“DB2 通用数据库”对象一起显示在“控制中心”主窗口上。管理这些对象的对应操作和实用程序是通过选择该对象来调用的。例如，用户可列示特定数据库的索引，选择其中一个索引，并对其进行重组。用户还可列示数据库的表并运行更新统计，或将表定义为复制源。

有关配置“控制中心”以执行“DB2 服务器 VSE 版和 VM 版”对象的管理任务的信息，参考《DB2 Connect 用户指南》或《安装和配置补遗》。

Java 1.2 支持“控制中心”

“控制中心”使用 Java 1.2 中的双向支持来支持双向语言（如阿拉伯语和希伯来语）。仅对 Windows NT 平台提供了此支持。

必须为“控制中心”安装 Java 1.2 以对其进行重组和使用：

1. JDK 1.2.2 是在 DB2 UDB CD 上的 DB2\bid\NT 目录下提供的。ibm-inst-n122p-win32-x86.exe 是安装程序，而 ibm-jdk-n122p-win32-x86.exe 是 JDK 分布式系统。将这两个文件复制至硬盘驱动器上的临时目录，然后从该位置运行安装程序。
2. 在 <DB2PATH>\java\Java12（其中 <DB2PATH> 是 DB2 的安装路径）下安装它。
3. 在 JDK/JRE 安装提示时不要选择 JDK/JRE 作为系统 VM。

在成功安装 Java 1.2 之后，以正常方式启动“控制中心”将使用 Java 1.2。

要停止使用 Java 1.2，可从 <DB2PATH>\java\Java12 卸载 JDK/JRE，或就将 <DB2PATH>\java\Java12 子目录重命名为别的目录名。

注：不要混淆 <DB2PATH>\java\Java12 与 <DB2PATH>\Java12。<DB2PATH>\Java12 是 DB2 安装的一部分，且包括对 Java 1.2 的 JDBC 支持。

在 Windows 操作系统上使用联机帮助时发生“快捷方式无效”错误

在使用“控制中心”联机帮助时，您可能会遇到类似“快捷方式无效”之类的错误。如果最近安装了新的 Web 浏览器或新版本的 Web 浏览器，确保 HTML 与 HTM 文档与正确的浏览器相关联。参见“Windows 帮助”主题“要更改打开文件时启动哪个程序”

OS/2 上的 Java 控制中心

“控制中心”必须安装在 HPFS 格式的驱动器上。

试图在 Windows 操作系统上的日志中查看已完成作业时发生“拒绝访问文件”错误

在“DB2 通用数据库 Windows NT 版”上，试图打开“日志”以查看在“脚本中心”中创建的作业的详细信息时发生“拒绝访问文件”错误。作业状态显示为完成。当在“脚本中心”中创建的作业包含 START 命令时，会发生此情况。要避免发生此情况，同时在批处理文件和作业本身中使用 START/WAIT 而不是 START。

多站点更新测试连接

“版本 7 控制中心”中的“多站点更新测试连接”功能受到目标实例版本的限制。要使“远程”测试连接功能运行，目标实例必须至少为版本 7。要在版本 6 中运行“多站点更新测试连接”功能，必须在目标实例上以本地方式启动“控制中心”并从该处运行它。

“DB2 OS/390 版”的“控制中心”

“DB2 UDB 控制中心 OS/390 版”使您可以管理特许 IBM DB2 实用程序的使用。实用程序功能（作为可单独订购的“DB2 UDB OS/390 版”功能部件的组件）必须已在您的环境中获得许可证并已安装，才能由“DB2 控制中心”进行管理。

“CC390”数据库（它是在配置 DB2 OS/390 版子系统时使用“控制中心”定义的）用于“控制中心”的内部支持。不要修改此数据库。

尽管“控制中心”目录或“信息中心任务”信息中并未专门提及 DB2 OS/390 版的版本 7.1，但是该文档确实支持 DB2 OS/390 版的版本 7.1 功能。许多特定于 DB2 OS/390 版的版本 6 的功能还与 DB2 OS/390 版的版本 7.1 有关，而目录中特定于 DB2 OS/390 版的版本 7.1 的某些功能没有指明版本。如果已经在“控制中心”上配置了 DB2 OS/390 版的版本 7.1 子系统，您就具有对该版本所有文档的访问权。

要从“DB2 OS/390 版的控制中心”访问和使用“生成 DDL”功能，必须安装“生成 DDL”功能：

- 对于版本 5，用 DB2 OS/390 版的版本 5 安装 DB2Admin 2.0。
- 对于版本 6，安装将作为“DB2 OS/390 版的版本 6”的 DB2 Admin 功能部件的 PTF 提供的小型编程增强功能。
- 对于版本 7.1，“生成 DDL”功能是“DB2 OS/390 版的版本 7.1”中另外定价的 DB2 Admin 功能部件的一部分。

可以从“控制中心”中访问“存储过程构建器”，但是当您启动“DB2 UDB 控制中心”时必须已经安装了“存储过程构建器”。它是“DB2 应用程序开发客户机”的一部分。

要直接在工作站上编目 DB2 OS/390 版子系统，选择使用“客户机配置辅助程序”工具。

1. 在“源”页面上，指定**人工配置与数据库的连接**单选按钮。
2. 在“协议”页面上，填入适当的通信信息。
3. 在“数据库”页面上，在**数据库名字段**中指定子系统名。
4. 在“节点选项”页面上，选择**配置节点选项（可选）**复选框。
5. 在**操作系统**字段中，从列表中选择 MVS/ESA、OS/390。
6. 单击**完成**以完成配置。

要通过网关机器来编目“DB2 OS/390 版”子系统，在网关机器中遵循以上的步骤 1-6，然后：

1. 在客户机上，启动“控制中心”。
2. 用鼠标右键单击“系统”文件夹，并选择**添加**。
3. 在“添加系统”对话框中，在**系统名字段**中输入网关机器名。
4. 在**远程实例**字段中输入 DB2DAS00。
5. 对于 TCP/IP 协议，在“协议”参数中，在**主机名字段**中输入网关机器的主机名。
6. 在**服务名字段**中输入 523。
7. 单击**确定**以添加系统。您现在就应该看到在**系统**文件夹中添加的网关机器名。
8. 展开网关机器名。
9. 用鼠标右键单击“实例”文件夹，并选择**添加**。
10. 在“添加实例”对话框中，单击**刷新**以列示网关机器上可用的实例。如果网关机器为 Windows NT 系统，则可能会在实例 DB2 下编目“DB2 OS/390 版”子系统。
11. 选择该实例。会为此实例自动填入协议参数。
12. 单击**确定**以添加实例。
13. 打开**实例**文件夹以查看您刚才添加的实例。
14. 展开该实例。
15. 用鼠标右键单击**数据库**文件夹，并选择**添加**。
16. 单击**刷新**以显示网关机器上的本地数据库。若您正在“添加数据库”对话框中添加 DB2 子系统，则在“数据库名”字段中输入子系统名。选项：输入子系统（或数据库）的本地别名。
17. 单击**确定**。

现在已在“控制中心”中成功添加了子系统。打开数据库时，应可看到所显示的 DB2 OS/390 版子系统。

“控制中心 390”一节的第一段内容为：

“DB2 UDB 控制中心 OS/390 版”允许您管理特许 IBM DB2 实用程序的使用情况。作为“DB2 UDB OS/390 版”的可单独订购的功能部件的元素的实用程序功能必须是特许的，且在由“DB2 控制中心”来管理该功能之前已经在环境中安装了这些功能。

本节的内容现在应该为：

“DB2 控制中心 OS/390 版”允许您管理特许 IBM DB2 实用程序的使用情况。作为可单独订购的产品的元素的实用程序功能必须是特许的，且已经安装在环境中以便由“DB2 控制中心”进行管理。

对“控制中心 OS/390 版”的必要修正

必须对“DB2 OS/390 版的版本 5”和“DB2 OS/390 版的版本 6”的 390 Enablement 功能部件应用 APAR PQ36382，才能使用版本 7 的“DB2 UDB 控制中心”来管理这些子系统。如果没有此修正，就不能使用版本 7 的“DB2 UDB 控制中心”来对这些子系统运行实用程序。

APAR 应该应用于下列 FMID:

DB2 OS/390 版的版本 5 390 启用: FMID JDB551D
DB2 for OS/390 Version 6 390 启用: FMID JDB661D

对“创建空间层”对话框的更改

"<<" 和 ">>" 按钮已从“创建空间层”对话框除去。

“DB2 控制中心”的故障诊断信息

在“快速入门”一书的『控制中心安装和配置』一章中，标题为『故障诊断信息』的一节告诉您：如果在将“控制中心”作为小应用程序运行出现问题，则在命令窗口中不会设置客户机浏览器的 CLASSPATH。本节还说明应从同一命令窗口启动浏览器。但是，未提供用于启动浏览器的命令。要启动 Internet Explorer，输入 start iexplore 并按 **Enter** 键。要启动 Netscape，输入 start netscape 并按 **Enter** 键。这些命令假定浏览器在路径中。如果浏览器不在路径中的话，将其添加至路径或切换至浏览器的安装目录，并重新发出 **start** 命令。

基于 UNIX 的系统上的控制中心故障诊断

如果无法在基于 UNIX 的系统上启动“控制中心”，则将 JAVA_HOME 环境变量设置为指向 Java 分布式系统:

- 如果 java 是安装在 /usr/jdk118 下的，则将 JAVA_HOME 设置为 /usr/jdk118。
- 对于 sh、ksh 或 bash shell:

```
export JAVA_HOME=/usr/jdk118.
```

- 对于 csh 或 tcsh shell:

```
setenv JAVA_HOME /usr/jdk118
```

OS/2 上可能的 Infopops 问题

如果正在 OS/2 上运行“控制中心”，所使用的屏幕大小为 1024x768 且为 256 色，并启用了 Workplace Shell Palette Awareness，则超出当前窗口边界之外的 infopops 可能会显示成黑色背景上的黑色文本。要修正此问题，可将显示设置更改为超过 256 色，或禁用 Workplace Shell Palette Awareness。

jdk11_path 配置参数的帮助

在“控制中心”帮助中，“Java Development Kit 1.1 安装路径” (jdk11_path) 配置参数的描述中适用于小标题下的一行丢失了。适用于下的完整列表为：

- 带有本地和远程客户机的数据库服务器
- 客户机
- 带有本地客户机的数据库服务器
- 带有本地和远程客户机的分区数据库服务器
- 带有本地客户机的卫星数据库服务器

在使用“脚本中心”或“日志”时出现的 Solaris 系统错误 (SQL10012N)

如果从“脚本中心”或“日志”选择了 Solaris 系统，则可能遇到以下错误：

```
SQL10012N - 当装入指定的库  
"/udbprod/db2as/sqllib/function/unfenced/db2scdar!ScheduleInfoOpenScan"  
时接收到意外的操作系统错误。SQLSTATE=42724。
```

这是由 Solaris 运行时链接器中的错误造成的。要校正此问题，应用以下补丁：

105490-06 (107733 使 105490 作废) Solaris 2.6 版

DPREPL.DFT 文件的帮助

在“控制中心”中，“工具设置”笔记本的“复制”页帮助中的步骤 5d 为：

将文件保存到“控制中心”的工作目录（例如，SQLLIB\BIN）中，以便系统可以将它用作缺省文件。

步骤 5d 应为：

将文件保存到“控制中心”的工作目录（SQLLIB\CC）中，以便系统可以将它用作缺省文件。

启动多个控制中心小应用程序

不能在同一机器上同时启动多个“控制中心”小应用程序。此限制适用于在所有受支持的浏览器上运行的“控制中心”。

关于作为小应用程序运行的“控制中心”的联机帮助

当“控制中心”作为小应用程序来运行时，F1 键只有在具有信息弹出窗口的窗口和记事本中才起作用。

可按下 F1 键，下列组件中出现信息弹出窗口：

- DB2 通用数据库 OS/390 版
- 向导

在“控制中心”组件的余下组件中，按 F1 键不会激活任何帮助。要显示其他组件的帮助，请使用“帮助”按钮或“帮助”下拉菜单。

以小应用程序方式运行“控制中心”(Windows 95)

如果指定了无效的用户标识和口令，则对打开“脚本中心”的尝试可能失败。确保在登录“控制中心”时输入了有效的用户标识和口令。

使用大型查询结果

用户很容易生成返回多行的查询。但是用户不容易预测实际上可能返回了多少行。对于可能潜在地返回上千（上万）行的查询，可能有两个问题：

1. 可能要花很长时间来检索结果。
2. 可能需要大量的客户机内存用来保存结果。

为了便于完成此过程，DB2将大型结果集分为几个部分。它将一次只检索和显示一个部分的查询结果。

因此：

1. 显示时间将缩短，因为在查看查询的第一部分时也在检索其余部分。
2. 将减少对客户机的内存需求，因为在任何给定时间都只有查询结果的一个部分存储在客户机上。

要控制内存中的查询结果行数：

1. 打开“工具设置”笔记本的“一般”页。
2. 在**最大大小**一节中，选择：
 - **样本内容**来限制“样本内容”窗口中显示的结果行数。在输入字段中指定结果集的部分大小（行数）。
 - **命令中心**来限制“命令中心”的“查询结果”页上所显示的结果行数。在输入字段中指定结果集的部分大小（行数）。

当使用“样本内容”窗口或者“命令中心”的“查询结果”页上的查询结果时，**内存中的行数字段**指示内存中为查询保存的行数。此数目决不能大于**最大大小**设置。单击下一个来检索结果集的下一部分。当下一个不活动时，就表示已经到达了结果集末尾。

信息中心

在 Windows 操作系统上出现“快捷方式无效”错误

在使用“信息中心”时，您可能会遇到类似“快捷方式无效”之类的错误。如果最近安装了新的 Web 浏览器或新版本的 Web 浏览器，确保 HTML 与 HTM 文档与正确的浏览器相关联。参见“Windows 帮助”主题“要更改打开文件时启动哪个程序”

当 Netscape 已打开时，打开 Netscape Navigator 中的外部 Web 链接（基于 UNIX 的系统）

如果 Netscape Navigator 已打开，且正在显示一个本地 DB2 HTML 文档或一个外部 Web 站点，则试图从“信息中心”打开一个外部 Web 站点会导致 Netscape 错误。该错误指出“Netscape 找不到名为 <外部站点> 的文件或目录。”

要解决此问题，应在打开外部 Web 站点之前，关闭打开的 Netscape 浏览器。Netscape 将重新启动并带出该外部 Web 站点。

应注意，在 Netscape 已打开时打开本地 DB2 HTML 文档不会出现此错误。

启动“信息中心”时的问题

在某些系统上，如果使用“开始”菜单、“第一步”或 **db2ic** 命令来调用“信息中心”，启动它可能会很慢。如果遇到此问题，应启动“控制中心”，然后选择帮助 —> 信息中心。

向导

在创建数据库向导中设置范围大小

使用“创建数据库向导”，可对新数据库的“用户表空间”（而不是那些“目录”或“临时表”）设置“范围大小”和“预取大小”参数。仅当对“向导”的“用户表”页上的“用户表空间”指定了至少一个容器时，才将启用此功能部件。

“MQSeries 辅助程序”向导

DB2 版本 7.2 提供了新的“MQSeries 辅助程序”向导。此向导创建一个表函数，该函数使用“DB2 MQSeries 函数”来从 MQSeries 队列读取，它是版本 7.2 中的新增项。该向导可以将每条 MQSeries 消息视作一个定界字符串或者定长列字符串，这取决于您的规范。所创建的表函数根据您的规范来对字符串进行语法分析，并将每条 MQSeries 消息作为表函数的一行来返回。该向导还允许您在表函数的顶部创建视图，以及预览 MQSeries 消息和表函数结果。可以从“存储过程构建器”或者“数据仓库中心”来启动此向导。

此向导的需求是：

- MQSeries 版本 5.2
- MQSeries 应用程序消息传递接口 (AMI)
- DB2 MQSeries 函数

有关这些需求的详细信息，参见第159页的『MQSeries』。

要获取样本和“MQSeries 辅助程序”向导教程，转至教程部分，网址为：
<http://www.ibm.com/software/data/db2/udb/ide>

“OLE DB 辅助程序”向导

此向导帮助您创建表函数，该表函数从支持 Microsoft OLE DB 标准的另一个数据库提供者读取数据。可以选择使用 OLE DB 表函数读取的数据创建 DB2 表，并可为 OLE DB 表函数创建视图。可以从“存储过程构建器”或者“数据仓库中心”来启动此向导。

此向导的需求是：

- OLE DB 提供者（例如 Oracle、Microsoft SQL Server）
- OLE DB 支持函数

要获取样本和“OLE DB 辅助程序”向导教程，转至教程部分，网址为：
<http://www.ibm.com/software/data/db2/udb/ide>

第5部分 商业智能

商业智能教程	189	为 MQSeries 消息创建视图	209
修正商业智能教程	189	需求	209
		限制	209
		为 MQSeries 消息创建视图	209
数据仓库中心管理指南	191	导入 MQSeries 消息和 XML 元数据	210
故障诊断	191	需求	210
将 Excel 设置为仓库源	191	限制	210
定义和运行进程	191	导入 MQSeries 消息和 XML 元数据	210
“导出元数据”对话框	191	使用 MQSeries 用户定义程序	211
为“提交 OS/390 JCL 作业流”(VWPMVS)程序定 义值	191	错误返回码	211
对『数据入库样本』附录的更改	191	错误日志文件	212
数据仓库中心消息	192	Microsoft OLE DB 和“数据事务服务”支持	212
创建轮廓并在“DB2 OLAP 集成服务器”中装入数 据	192	为 OLE DB 表函数创建视图	213
将 Classic Connect 配合“数据仓库中心”使用	192	为 DTS 软件包创建视图	213
“数据仓库中心”环境结构	192	将增量提交与替换配合使用	213
使用“反转变换器”	193	组件跟踪数据文件名	213
用 DB2 版本 7 仓库代理程序来访问 DB2 版本 5 数据	193	在 AIX 和 Solaris 操作环境下, Sybase 源所需要的 开放式客户机	214
迁移 DB2 版本 5 服务器	193	已校正的样本条目	214
更改代理程序配置	193	第 3 章 设置仓库源	214
UNIX 仓库代理程序	193	将 Microsoft Access 中的“备忘录”字段映射至 仓库源	214
Microsoft Windows NT、Windows 2000 和 OS/2 仓库代理程序	193	第 10 章 维护仓库数据库	215
IBM ERwin 元数据抽取程序	194	将表链接至 DB2 UDB RUNSTATS 程序的步骤 子类型	215
内容	194	缺省仓库控制数据库	215
软件需求	195	“仓库控制数据库管理”窗口	215
程序文件	195	更改活动的仓库控制数据库	215
创建标记语言文件	196	创建和初始化仓库控制数据库	216
将标记语言文件导入数据仓库中心	197	创建版本化的 SQL 步骤	216
将标记语言文件导入信息目录管理器	198	在“过程模型”窗口中更改源和目标	216
故障诊断	198	为“数据仓库中心”对象添加描述	216
ERwin 与“DB2 数据仓库中心”之间的映射	199	运行样本内容	216
ERwin 与“信息目录管理器”之间的映射	200	编辑“创建 DDL SQL”语句	217
清理“数据仓库中心”中的名称和地址	202	迁移 Visual Warehouse 商业视图	217
需求	202	生成目标表和主键	217
Trillium Software System 组件	203	使用 Merant ODBC 驱动程序	217
将 Trillium Batch System 与“数据仓库中心” 配合使用	203	新的 ODBC 驱动程序	218
导入 Trillium 元数据	204	在 OS/2 数据库中定义仓库源或目标	218
映射元数据	205	监控仓库控制数据库的状态	218
限制	205	将“SQL 辅助程序”与 TBC_MD 样本数据库配合 使用	218
编写 Trillium Batch System JCL 文件	206	使用 FormatDate 函数	218
在 UNIX 环境变量 Windows 上编写 Trillium Batch System 脚本文件	206	更改语言设置	218
定义 Trillium Batch System 步骤	206	使用“生成键表”变换器	219
使用 Trillium Batch System 用户定义程序	206	维护与数据库的连接	219
错误处理	207	设置远程“数据仓库中心”客户机	219
错误返回码	208	定义“DB2 VM 版”仓库源	219
日志文件	209	定义“DB2 VM 版”或“DB2 VSE 版”目标表	219
MQSeries 与“数据仓库中心”集成	209	启用定界标识符支持	220
		Data Joiner 错误指示绑定问题	220

用“数据仓库中心”设置和运行复制	220
故障诊断提示	222
访问源和目标	223
受支持的非 IBM 数据库源的补充内容	224
在“数据仓库中心”中人工创建数据源	225
使用“公共数据仓库元数据交换”(CWMI)导入和导出元数据	226
介绍	226
导入元数据	226
运行导入实用程序之后更新元数据	227
导出元数据	228
OS/390 Runstats 实用程序步骤	229
OS/390 装入实用程序步骤	229
公共仓库元模型 (CWM) XML 支持	230
进程造型程序	230
模式造型程序	231
必填字段	231
数据仓库中心启动板增强功能	231
将步骤信息打印至文件	231
Data Warehouse Center Application Integration Guide	233
附加的元数据模板	233
Commit.tag	233
标记	234
示例值	234
ForeignKey.tag	234
标记	234
示例值	235
ForeignKeyAdditional.tag	236
标记	236
示例值	237
PrimaryKey.tag	237
标记	237
示例值	238
PrimaryKeyAdditional.tag	239
标记	239
示例值	240
数据仓库中心联机帮助	241
对复制定义表或视图	241
将 Essbase VWP 与 AS/400 代理程序一起运行	241
使用“发布数据仓库中心元数据”窗口和“相关性”窗口	241
外键	241
复制笔记本	241
导入标记语言	241
用于添加数据的链接	241
导入表	241
对 RUNSTATS 和 REORGANIZE TABLE 联机帮助	242
的校正	242
“通知”页 (“数据仓库特性”笔记本和“调度”笔记本)	242
“代理点”笔记本中的“代理模块”字段	242
DB2 OLAP Starter Kit	243
OLAP 服务器 Web 站点	243

受支持的操作系统服务级别	243
在 UNIX 上完成 DB2 OLAP Starter Kit 设置	243
为 OLAP Starter Kit 配置 ODBC	244
在 UNIX 系统上配置数据源	244
配置 ODBC 环境变量	244
编辑 odbc.ini 文件	244
将数据源添加至 odbc.ini 文件	245
DB2 的 ODBC 设置的示例	245
Oracle 的 ODBC 设置的示例	245
在 UNIX 系统上配置“OLAP 元数据目录”	245
在 Windows 系统上配置数据源	246
在 Windows 系统上配置“OLAP 元数据目录”	247
在配置数据源之后	248
从“OLAP Starter Kit 桌面”进行登录	248
Starter Kit 登录示例	249
人工创建和配置 OLAP Starter Kit 的样本数据库	249
将应用程序迁移至 OLAP Starter Kit 版本 7.2	250
已知问题和限制	250
缺少 OLAP Spreadsheet Add-in EQD 文件	252
Information Catalog Manager Administration Guide	253
信息目录管理器初始化实用程序	253
许可证发放问题	253
安装问题	253
使用 DB2 版本 7 信息目录管理器来访问 DB2 版本 5 信息目录	254
设置信息目录	255
与其他产品交换元数据	255
使用 flgnxoln 命令来交换元数据	255
使用 MDISDGC 命令来交换元数据	255
调用程序	255
Information Catalog Manager Programming Guide and Reference	257
信息目录管理器原因码	257
信息目录管理器用户指南	259
信息目录管理器: 联机消息	261
消息 FLG0260E	261
消息 FLG0051E	261
消息 FLG0003E	261
消息 FLG0372E	261
消息 FLG0615E	261
信息目录管理器: 联机帮助	263
用于 Web 的“信息目录管理器”	263
DB2 仓库管理器安装指南	265
仓库变换器的软件需求	265
SAP R/3 的连接器的	265
安装先决条件	265
用于 Web 的连接器的	265
安装先决条件	265
Query Patroller Administration Guide	267

DB2 Query Patroller 客户机是一个独立组件	267
使用 dqpmigrate 从 DB2 Query Patroller 的版本 6	
迁移	267
Enabling Query Management.	267
控制表的表空间的位置	267
dqptest 命令的新参数.	268
iwm_cmd 命令的新参数	268
新的注册表变量: DQP_RECOVERY_INTERVAL	268
启动 Query Administrator	269
用户管理	269
创建作业队列	269
使用“命令行接口”	269
“查询使能器”注释	269
DB2 Query Patroller Tracker 可能会返回空白列页	270
Query Patroller 和复制工具	270
Appendix B. Troubleshooting DB2 Query Patroller	
Clients	270

商业智能教程

修正商业智能教程

“修订包 2”中包括修正“商业智能教程”和“数据仓库中心样本”数据库，它们用来校正版本 7.1 中存在的各种问题。为应用修正“数据仓库中心样本”数据库，必须执行下列操作：

若尚未安装样本数据库，则使用“第一步骤”启动板来创建新的样本数据库。单击**开始**，并选择**程序 → IBM DB2 → 第一步骤**。

若先前已安装了样本数据库，则删除样本数据库 DWCTBC、TBC_MD 和 TBC。若已经将您想保存的任何数据添加到了样本数据库中，则在删除它们之前应进行备份。要删除这三个样本数据库：

1. 要打开“DB2 命令窗口”，打开**开始**，并选择**程序 → IBM DB2 → 命令窗口**。
2. 在“DB2 命令窗口”中，输入下列三个命令，在输入每个命令之后按 Enter 键：

```
db2 drop database dwctbc
db2 drop database tbc_md
db2 drop database tbc
```
3. 关闭“DB2 命令窗口”。
4. 使用“第一步骤”启动板来创建新的样本数据库。单击**开始**，并选择**程序 → IBM DB2 → 第一步骤**。

数据仓库中心管理指南

故障诊断

“数据仓库中心”故障诊断信息已移至“DB2 故障诊断指南”。

将 Excel 设置为仓库源

在第 3 章『设置仓库源』中的『在 Windows NT 中设置非 DB2 数据库仓库源』一节中，与 Microsoft Excel 有关的一节中缺少一个步骤。新步骤在下面显示为“第 3 步”。

如果您正在使用 Microsoft Excel 95/97 ODBC 驱动程序来访问 Excel 电子表格，则需要为电子表格中的每个工作表创建一个命名的表。要为每个工作表创建命名的表：

1. 选择您想要的列和行。
2. 单击“Excel —> 插入 —> 名称 —> 定义”。
3. 确保“定义名称”窗口的“引用”字段包含您在“第 1 步”中已经选择的单元。如果不包含，则单击“引用”字段右边较远处的图标以包括您已选择的所有单元。
4. 对所标记的数据输入名称（或者使用缺省名称）。
5. 单击“确定”。

定义和运行进程

在第 5 章『定义和运行进程』的『从“数据仓库中心”外部来启动步骤』一节中，应指出：若启动具有双字节名称的步骤，则仓库服务器工作站和代理点上需要 JDK 1.1.8 或更新版本。

“导出元数据”对话框

在第 12 章的『导出和导入数据仓库中心元数据』这一节中的『将元数据导出至标记语言文件』这一小节中，步骤 5 应该为如下所示：

如果不想导出您正在导出的与进程相关的调度信息，则应清除包括调度复选框。

为“提交 OS/390 JCL 作业流” (VWPMVS) 程序定义值

在第 180 页上的『为“提交 OS/390 JCL 作业流” (VWPMVS) 程序定义值』一节中，步骤 8 指出您必须在 JES 文件所在的目录中定义 .netrc 文件。从而，程序会创建 .netrc 文件。如果该文件不存在，程序会在主目录中创建该文件，如果主目录中已经存在 .netrc 文件，则程序会重命名现存文件并创建一个新文件。在程序完成处理时，它会删除所创建的新 .netrc 文件，并将原来的文件重命名为 .netrc。

对『数据入库样本』附录的更改

- 在『数据入库样本』附录的『查看和修改样本元数据』一节中，GEOGRAPHIES 表应包括在源表列表中。

- 在『数据入库样本』附录的『提升步骤』一节中，在将步骤提升为生产方式的过程中，以下语句是不正确的，原因是目标表是在将步骤提升为测试方式时创建的：
“数据仓库中心”开始创建目标表，并显示进度窗口。

数据仓库中心消息

在 Microsoft Windows NT 和 Windows 2000 上，“数据仓库中心”将事件记录至系统事件日志。“事件标识”对应于“数据仓库中心”消息号。有关“数据仓库中心”消息的信息，参考《消息参考》。

创建轮廓并在“DB2 OLAP 集成服务器”中装入数据

第 315 页上的图 20 中的示例有错误。下列命令是正确的：

```
"C:\IS\bin\olapicmd" < "C:\IS\Batch\my_script.script" >  
"C:\IS\Batch\my_script.log"
```

如果路径中的目录名包含空格（如 Program Files），则需要将 "C:\IS\bin\olapicmd" 用双引号括起来。

将 Classic Connect 配合“数据仓库中心”使用

- 在附录 F『将 Classic Connect 配合“数据仓库中心”使用』中，第 388 页上的『安装 CROSS ACCESS ODBC 驱动程序』一节已替换为以下信息：

通过对“DB2 数据仓库管理器版本 7”执行定制安装，并选择“Classic Connect 驱动程序”组件来安装 CROSS ACCESS ODBC 驱动程序。该驱动程序未作为“DB2 仓库管理器”的典型安装的部分安装。

CROSS ACCESS ODBC 驱动程序将安装在 SQLLIB 目录的 ODBC32 子目录中。在完成安装之后，必须将该驱动程序的路径（例如，C:\Program Files\SQLLIB\ODBC32）人工添加至 PATH 系统环境变量。若已经安装了另一版本的 CROSS ACCESS ODBC 驱动程序，则将 ... \SQLLIB\ODBC32\ 路径放在另一版本的路径前面。操作系统将使用包含 CROSS ACCESS ODBC 驱动程序的路径中的第一个目录。

- 应该将以下过程添加到附录 F『将 Classic Connect 配合“数据仓库中心”使用』中：
安装“Classic Connect ODBC 驱动程序”：
 1. 将“仓库管理器” CD-ROM 插入 CD-ROM 驱动器中。启动板打开。
 2. 在启动板中单击**安装**。
 3. 在“选择产品”窗口中，确保选择了 **DB2 数据管理器**复选框，然后单击**下一步**。
 4. 在“选择安装类型”窗口中，选择**定制**，然后单击**下一步**。
 5. 在“选择组件”窗口中，选择 **Classic Connect 驱动程序**和**仓库代理程序**，清除所有其他复选框，然后单击**下一步**。
 6. 在“开始复制文件”窗口中，复查您的选择。若您想更改所作的任何选择，单击**上一步**以返回到可更改选择的窗口。单击**下一步**以开始安装。

“数据仓库中心”环境结构

在 401 页上的附录 G『数据仓库中心”环境结构』中，表中有一个不正确的条目。C:\Program Files\SQLLIB\ODBC32 未被添加至 PATH 环境变量。对 PATH 环境变量的唯一更新是 C:\Program Files\SQLLIB\BIN。

使用“反转变换器”

本书说明“反转变换器”可以根据参数来创建目标表，但是它忽略了生成的目标表将没有期望的输出列，而必须在目标表中显式地创建它。

用 DB2 版本 7 仓库代理程序来访问 DB2 版本 5 数据

由于 DB2 版本 7 仓库代理程序是由 DB2 版本 7 安装过程配置的，所以它将支持对 DB2 版本 6 和 DB2 版本 7 数据的访问。如果需要访问 DB2 版本 5 数据，必须采用下列两个方法之一：

- 将 DB2 版本 5 服务器迁移至 DB2 版本 6 或 DB2 版本 7。
- 在适当的操作系统上，修改代理程序配置以允许访问 DB2 版本 5 数据。

DB2 版本 7 仓库代理程序不支持对 DB2 版本 2 或所有其他先前版本数据的访问。

迁移 DB2 版本 5 服务器

有关迁移 DB2 版本 5 服务器的信息，参见适用于您的操作系统的 *DB2 通用数据库快速入门*。

更改代理程序配置

以下信息描述了如何对每个操作系统更改代理程序配置。在将 DB2 服务器迁移为 DB2 版本 6 或更新版本时，除去您对配置所作的更改。

UNIX 仓库代理程序

要设置 UNIX 仓库代理程序以使用 CLI 或 ODBC 访问方法从 DB2 版本 5 访问数据：

1. 安装 DB2 版本 6 运行时客户机。可通过选择从以下 URL 下载的客户机以获得运行时客户机：

<http://www.ibm.com/software/data/db2/udb/support>

2. 更新仓库代理程序配置文件，以使 DB2INSTANCE 环境变量指向 DB2 版本 6 实例。
3. 在此 DB2 版本 6 实例中编目仓库代理程序将要访问的所有数据库。
4. 通过使用代理程序守护程序进程标识发出终止命令来停止代理程序守护程序进程。然后，该代理程序守护程序将自动重新启动。您需要有 root 用户权限才能终止该进程。

Microsoft Windows NT、Windows 2000 和 OS/2 仓库代理程序

要设置 Microsoft NT、Windows 2000 或 OS/2 仓库代理程序以从 DB2 版本 5 访问数据：

1. 在不同于安装有 DB2 版本 7 仓库代理程序的工作站上安装“DB2 Connect 企业版版本 6”。

“DB2 通用数据库企业版”和“DB2 通用数据库扩充企业版”包括“DB2 Connect 企业版”。如果安装了这些 DB2 产品中任一种产品的版本 6，则不必单另安装 DB2 Connect。

限制：不能在同一 Windows NT 或 OS/2 工作站上安装多个版本的 DB2。可在另一 Windows NT 工作站或在 OS/2 或 UNIX 工作站上安装 DB2 Connect。

2. 配置仓库代理程序和 DB2 Connect 版本 6 以访问 DB2 版本 5 数据。有关详情，参见 *DB2 Connect 用户指南*。下列步骤是所需步骤的概述：
 - a. 在 DB2 版本 5 系统上，使用“DB2 命令行处理器”来编目仓库代理程序将要访问的版本 5 数据库。
 - b. 在 DB2 Connect 系统上，使用“DB2 命令行处理器”来编目：
 - DB2 版本 5 系统的 TCP/IP 节点
 - DB2 版本 5 系统的数据库
 - DB2 版本 5 系统的 DCS 项目
 - c. 在仓库代理程序工作站上，使用“DB2 命令行处理器”来编目：
 - DB2 Connect 系统的 TCP/IP 节点
 - DB2 Connect 系统的数据库

有关编目数据库的信息，参见 *DB2 通用数据库安装和配置补遗*。

3. 在仓库代理程序工作站上，将 DB2 CLI 软件包与要通过 DB2 Connect 访问的每个数据库进行绑定。

下列 DB2 命令给出了与 v5database 进行绑定的示例，一个假定的 DB2 版本 5 数据库。使用“DB2 命令行处理器”来发出下列命令。db2cli.lst 和 db2ajgrt 位于 \sqllib\bnd 目录中。

```
db2 connect to v5database user userid using password
db2 bind db2ajgrt.bnd
db2 bind @db2cli.lst blocking all grant public
```

其中，*userid* 是 v5 数据库的用户标识，而 *password* 是用户标识的口令。

在 db2cli.lst 与 DB2 版本 5 数据库进行绑定时出错。会出现此错误的原因是大对象 (LOB) 在此配置中不受支持。此错误将不会影响仓库代理程序对 DB2 版本 5 数据库的访问。

“DB2 通用数据库版本 5”的“修订包 14”（是在 2000 年 6 月提供的）是通过 DB2 Connect 访问“DB2 版本 5”所必需的。参考该“修订包”中的 APAR 号 JR14507。

IBM ERwin 元数据抽取程序

内容

- 软件需求
- 程序文件
- 创建标记语言文件
- 将标记语言文件导入数据仓库中心
- 将标记语言文件导入信息目录管理器
- 故障诊断
- ERwin 与“数据仓库中心”之间的映射

ERwin 与 “信息目录管理器” 之间的映射

本节描述如何使用 “IBM ERwin 元数据抽取程序” 来从 ER1 文件中抽取元数据，并创建 “DB2 数据仓库中心” 或 “信息目录管理器” (DataGuide) 标记语言文件。

元数据抽取程序抽取存储在输入 ERI 文件中的所有物理对象（如数据库、表和列），并将元数据模型写至 “数据仓库中心” 或 “信息目录管理器” 标记语言文件。还会抽取和创建 “信息目录管理器” 的逻辑模型（它由实体和属性组成），并创建对象之间的所有相关关系标记，如数据库与表之间以及表与实体之间的关系标记。对于没有数据库的表，会创建一个称为 DATABASE 的缺省数据库。对于没有模式的表，将使用缺省模式 USERID。对于模型名，将使用 ER1 文件名。有关 ER1 属性至 “数据仓库中心” 或 “信息目录管理器” 标记的映射的详情，参见『ERwin 与 “DB2 数据仓库中心” 之间的映射』和『ERwin 与 “信息目录管理器” 之间的映射』。

元数据抽取程序支持具有关系数据库的所有 ER1 模型，包括 DB2、Informix、Oracle、Sybase、ODBC 数据源和 Microsoft SQL Server。

软件需求

必须满足下列软件需求才能运行抽取程序：

- Windows NT 4.0 或更新版本
- 带有 Service Pack 3 Build 466 的 ERwin 3.5.2

必须满足下列软件需求才能导入 ERwin 标记语言文件：

对于数据仓库中心：

IBM DB2 通用数据库版本 7.2

对于信息目录管理器：

IBM DB2 仓库管理器 7.2

模板标记语言文件 (.tag) 必须在 VWS_TEMPLATES 环境变量所指向的目录中。

类型标记语言文件 (.typ) 必须在 DGWPATH 环境变量所指向的目录中。

程序文件

元数据抽取程序安装在 IBM DB2 目录的 sqllib\bin 子目录中。该程序将在目录中安装下列文件：

flgerwin.exe

主迁移程序

erwext.dll

标记语言文件生成器 DLL

cdmerwsn.dll

ERwin API 封装器类 DLL

要启动抽取程序，从命令提示符处发出 flgerwin 命令。

创建标记语言文件

要创建“数据仓库中心”或“信息目录管理器”标记语言文件，运行 `flgerwin.exe` 程序并提供两个主要参数。第一个参数是要从其中抽取元数据的 ER1 文件。第二个参数是输出标记语言文件的名称。缺省情况下，抽取程序会将 MERGE 参数添加至“数据仓库中心”标记语言文件。

命令语法是：

```
flgerwin inputFile.er1 outputFile.tag [-dwc] [-icm] [-m] [-u] [-a] [-d]
```

如果想创建星型模式，则命令的语法应为：

```
flgerwin inputFile.er1 outputFile.tag [-dwc] [-starschema]
```

-dwc 创建“数据仓库中心”标记语言文件。可用于 `-dwc` 的可选参数是 `-m` 和 `-starschema`。

-icm 创建“信息目录管理器”标记语言文件。可用于 `-icm` 的可选参数是 `-m`、`-u`、`-a` 和 `-d`。

-starschema

创建 ERwin 模型星型标记语言文件。

-m 指定对对象的操作是 MERGE。

-u 指定对对象的操作是 UPDATE。

-a 指定对对象的操作是 ADD。

-d 指定对对象的操作是 DELETE。

元数据抽取程序使用元数据而不是数据。在完成 ERwin 标记语言文件导入之后，在使用目标表之前，必须使口令与用户标识相匹配。要将元数据与现存数据库数据合并：

在特性 → 数据库 → 用户标识下更改“数据仓库中心”用户标识和口令，以便与合并后的数据库用户标识和口令相匹配。

借助元数据抽取程序，可以将标记语言文件作为目标来导入。在新导入的元数据中，尚未填充表。可以逻辑或物理表示法来查看这些表，然后构建仓库步骤来填充从 ERwin 导入的表定义。

输入 ER1 文件必须处于可写状态。在运行元数据抽取程序之后，ER1 文件就成为只读文件。要将该文件更改为读/写方式，使用类似以下示例的命令：

```
attrib -r erwinsimplemode.er1
```

其中，`erwinsimplemode.er1` 是 ERwin 平面文件的名称。

如果 ER1 文件正用于当前的 ERwin 会话，或者检测到某些错误情况，则元数据抽取程序会将该 ER1 文件保存为只读状态。如果 ER1 文件处于只读状态，则您可能会接收到异常的程序终止错误消息。元数据抽取程序将显示它当前正在处理的表名。当元数据抽取程序完成处理时，您将接收到信息性消息。

当您正在通过自动连接维表与事实表来创建星型模式时，可能要花很长时间来处理，这取决于您使用的表的数目。在处理期间，自动连接的线是绿色的。保存时，这些线会变成黑色的。

使用自动生成的约束名来确保约束名是唯一的。

在处理期间，可能会接收到消息“发现重复列。将不会抽取该列”。这是一条信息性消息，不会影响抽取程序的成功完成。当外键的物理名与当前所处理的表中某列的物理名相同时，则会显示此消息。

将标记语言文件导入数据仓库中心

可以利用两种方法的其中任何一种来将标记语言文件导入“数据仓库中心”。可以使用“数据仓库中心”或者命令行来导入。

要使用“数据仓库中心”来导入标记语言文件：

1. 单击开始 → 程序 → IBM DB2 → 控制中心。“DB2 控制中心”打开。
2. 打开“数据仓库中心”并进行登录。
3. 用鼠标右键单击仓库。“导入”窗口打开。
4. 单击导入元数据 → ERwin。导入元数据窗口打开。
5. 在输入文件字段中，输入输入标记语言文件的名称，并单击确定。
6. 选择抽取星型模式复选框来将 ERwin 星型模式元数据模型定义为仓库模式。

在导入完成之后，可以单击查看 → 刷新来查看新步骤。

要使用命令行来导入标记语言文件，输入以下命令：

```
iwh2imp2 tag-filename log-pathname  
target-control-db userid password
```

tag-filename

标记语言文件的全路径和文件名。

log-pathname

日志文件的全路径名。

target-control-db

要导入的目标数据库的名称。

userid 用来访问控制数据库的用户标识。

password

用来访问控制数据库的口令。

要更改 DB2 数据库定义以便使它成为“数据仓库中心”中的源，可以更改标记语言文件：

- 对于您想作为源的每个数据库，将 ISWH 标记从 ISWH(Y) 更改为 ISWH(N)。
- 对于您想作为源的每个数据库，将关系标记从 **:RELTYPE.TYPE(LINK) SOURCTYPE(SCGTARIR) TARGETYPE(DATABASE)** 更改为 **:RELTYPE.TYPE(LINK) SOURCTYPE(SCGSR CIR) TARGETYPE(DATABASE)**。

当导入标记语言文件时，您可能会接收到以下消息：

消息: DWC13238E 在标记语言文件中，由
“DBNAME(____) OWNER(____) TABLE(____) COLUMNS(____)” 标识的类型为
“COLUMN” 的对象被定义了两次。

这是一条信息性消息，但是您的导入还是成功地完成了。如果实体的外键具有相同的名称，或是实体的名称相似的列被截断，或者在其他类似的情况下，您就会接收到此消息。检查模型中是否有重复的列名，并进行适当的调整。

将标记语言文件导入信息目录管理器

可以用两种方法来将标记语言文件导入“信息目录管理器”。可以使用“信息目录管理器”或者使用命令行。

要使用“信息目录管理器”来导入标记语言文件：

1. 单击开始 → 程序 → DB2 → 信息目录管理器。
2. 单击目录 → 导入。“导入”窗口打开。
3. 单击查找来搜索标记语言文件，然后单击导入。

在导入完成之后，可双击主题图标，将打开一个窗口，显示已导入的所有模型和数据库。

要使用命令接口来导入标记语言文件，输入以下命令：

```
DGUIDE /USERID userid /PASSWORD password  
/DGNAME dgname /IMPORT filename /LOGFILE  
filename /ADMIN /RESTART (B|C)
```

/USERID

用来访问控制数据库的用户标识。

/PASSWORD

此用户标识的口令。

/DGNAME

“信息目录管理器”的名称。

/IMPORT

标记语言文件的全路径和文件名。

/LOGFILE

日志文件的全路径名。

/ADMIN

指示您正在作为管理员来登录。

/RESTART

指示将从标记语言文件开头开始导入（选项 B）或者从最后一个提交点开始导入（选项 C，这是缺省值）。

故障诊断

如果接收到错误消息，则查看此处的消息，并利用您可以执行的操作来解决该错误。

丢失 **ER1** 输入文件或标记输出文件。

元数据抽取程序需要两个具有特定次序的参数。第一个参数是 **ER1** 文件的名称，第二个参数是标记输出文件的名称。如果指定现存标记语言文件的名称，则该文件将被覆盖。

Windows 系统异常程序终止。

输入 ER1 文件可能处于只读状态。如果当您保存 ER1 文件时出现问题，且元数据抽取程序使文件处于只读方式，就可能发生这种情况。在命令 shell 中发出命令

```
attrib -r inputFile.er1
```

来将 ER1 文件的状态更改为读 / 写。

标记语言文件 ... 无法打开。

检查是否存在任何系统问题从而无法在当前驱动器上创建或打开文件。

找不到指向临时文件的路径。

未设置环境变量 VWS_TEMPLATES。检查是否安装了“数据仓库中心”。

找不到指向类型文件的路径。

未设置环境变量 DGWPATH。检查是否安装了“数据仓库中心”。

不受支持的服务器版本: ...

您正试图从其中抽取数据的输入 ER1 文件被存储在程序不支持的目标服务器上。启动 ERwin，打开 ER1 文件，然后单击服务器 → 目标服务器和适当的版本 [参见“软件需求”]。保存 ER1 文件。

未知 ERwAPI 错误。

已经发生了 ERwin API 错误，程序无法获得关于该错误的详情。确保安装了 ERwin 3.5.2。必须注册 ERwin API。

要注册 ERwin API，在安装了 ERwin 程序文件的目录中运行以下命令：**regsvr32 er2api32.dll**。将出现消息“er2api32.dll 中的 DllRegisterServer 已成功”。可以从“数据仓库中心”来启动抽取程序，或者通过从命令 shell 发出 flgerwin 命令来启动抽取程序。

抽取程序错误: ...

检查错误消息并执行适当的操作。很可能这是一个内部抽取程序错误，需要向 IBM 代表报告该问题。

未知抽取程序错误。

发生了未知错误。很可能这是一个内部错误，需要向 IBM 代表报告该问题。

抽取程序由于错误而终止。

发生了错误，使得抽取程序无法完成。请参考其他错误消息来解决问题或者与 IBM 代表联系。

ERwin 与“DB2 数据仓库中心”之间的映射

本节说明主要的 ERwin 对象属性如何与“数据仓库中心”标记对应:

数据库 - WarehouseDatabase.tag 或 SourceDatabase.tag

ERwin	命令行标记	数据仓库中心
图表名	NAME	仓库源或仓库目标的名称
图表作者	RESPNSBL	联系人
数据库名	DBNAME	数据库名
数据库版本	DBTYPE	数据库类型
图表描述	SHRTDESC	描述

表 - Table.tag

ERwin	命令行标记	数据仓库中心
表名	NAME	表名
表名	TABLES	表名
数据库名	DBNAME	n/a
表所有者	OWNER	表模式
表注释	SHRTDESC	描述

列 - Column.tag

ERwin	命令行标记	数据仓库中心
列名	NAME	列名
数据类型	NATIVEDT	数据类型
长度	LENGTH	长度
小数位	SCALE	小数位
空选项	NULLABLE	允许空值（复选框）
位置	POSNO	n/a
主键	KEYPOSNO	n/a
数据库名	DBNAME	n/a
表所有者	OWNER	n/a
表名	TABLES	n/a
列注释	SHRTDESC	描述

ERwin 与“信息目录管理器”之间的映射

本节说明主要的 ERwin 对象属性如何与“信息目录管理器”标记对应:

数据库 - Database.tag

ERwin	命令行标记	“信息目录管理器”接口
图表名	NAME	数据库名
图表作者	RESPNSBL	数据库所有者
数据库名	DBNAME	数据库名
数据库版本	DBTYPE	数据库类型
图表描述	SHRTDESC	简要描述

表 - TableOrView.tag

ERwin	命令行标记	“信息目录管理器”接口
表名	NAME	表名
表名	TABLES	表名
数据库名	DBNAME	数据库名
表所有者	OWNER	表所有者
表注释	SHRTDESC	简要描述
ERwin API	TABLVIEW	定义表示视图

列 - ColumnOrField.tag

ERwin	命令行标记	“信息目录管理器”接口
列名	NAME	列名
数据类型	DATATYPE	列的数据类型
长度	LENGTH	列的长度
小数位	SCALE	列的小数位
空选项	NULLS	列可空吗 (?)
位置	POSNO	列位置
主键	KEYPOSNO	主键中的列位置
ERwin API	ISKEY	列是键的一部分吗 (?)
ERwin API	UNIKEY	列是唯一键吗 (?)
数据库名	DBNAME	数据库名
表所有者	OWNER	表所有者
表名	TABLES	表名
列注释	SHRTDESC	简要描述
ERwin	ISTEXT	是数据文本吗 (?)
ERwin API	IDSRES	数据的解析率

模型 - Model.tag

ERwin	命令行标记	“信息目录管理器”接口
ER1 文件名	NAME	模型名
图表作者	RESPNSBL	有关详情...
图表描述	SHRTDESC	简要描述

实体 - Entity.tag

ERwin	命令行标记	“信息目录管理器”接口
实体名	NAME	实体名
注意事项	SHRTDESC	简要描述
定义	LONGDESC	长描述
实体所有者	RESPNSBL	有关详情...

属性 - Attribute.tag

ERwin	命令行标记	“信息目录管理器”接口
属性名	NAME	属性名
注意事项	SHRTDESC	简要描述
定义	LONGDESC	长描述
数据类型	DATATYPE	成员的数据类型
长度	LENGTH	成员的长度

清理“数据仓库中心”中的名称和地址

使用“数据仓库中心”和 Trillium Software System 来清理名称和地址数据。Trillium Software System 是一种名称和地址清理产品，它可以重新格式化、标准化和验证名称和地址数据。可以通过从用户定义程序启动 Trillium Batch System 程序来在“数据仓库中心”中使用 Trillium Software System。当您从 Trillium Batch System 脚本或 JCL 导入元数据时，用户定义程序就被添加到“仓库”树中。

“数据仓库中心”已经提供了与 Vality and Evolutionary Technologies 公司的工具的集成。

需求

- 必须在仓库代理点或远程主机上安装 Trillium Software System。
- 在 UNIX 和 Windows 平台上，必须将指向 Trillium Software System 的 bin 目录的路径添加到系统环境变量 PATH 中，才能使代理程序的进程运行 Trillium Batch System 程序。在 UNIX 上，必须通过在启动 vwdaemon 进程之前在 IWH.environment 文件中添加 PATH 变量来完成此操作。
- 用户必须具有 Trillium 软件的操作知识。

下表显示了软件需求。

操作系统	必需的软件
UNIX	Trillium Software System 版本 4.0 “数据仓库管理器”版本 7.2 仓库代理程序
Windows NT 和 Windows 2000	Trillium Software System 版本 4.0 “数据仓库管理器”版本 7.2 仓库代理程序 要进行远程访问，主机上必须安装了 ftpd 和 rexecd 守护程序。
OS/390	在远程 OS/390 主机上安装了 Trillium Software System 版本 4.0 在 UNIX、Windows NT 上安装了“数据仓库管理器”版本 7.2 仓库代理程序 必须安装 TCP/IP 3.2 或更高版本 只支持 OS/390 操作系统作为远程主机

Trillium Software System 组件

Trillium Software System 由四个主要组件组成：变换器、语法分析器、地理编码器和匹配器。将组件作为一组功能来执行名称和地址清理操作。可以从 Trillium Batch System（这是一个用户定义程序）来运行组件。

变换器 使用变换器来将源数据标准化，并将源数据转换成指定的输出格式。

语法分析器

使用语法分析器来解释名称和地址源数据，并创建关于源数据的元数据。

地理编码器

使用地理编码器来将源数据与邮政服务数据进行比较，以便提供任何丢失的信息，例如，送信人或 ZIP+4 代码。地理编码器还会执行美国户口普查资料的匹配操作。

匹配器 使用匹配器来比较相似的名称和地址，以便标识重复的记录。可以通过使用匹配器来将一条记录与一组记录进行比较来执行参考匹配。

将 Trillium Batch System 与“数据仓库中心”配合使用

在“数据仓库中心”中，可以导入 Trillium Batch System 元数据和创建用户定义程序步骤。此步骤在本地仓库代理点或者远程仓库代理点上调用 Trillium Batch System 脚本。在“数据仓库中心”中，Trillium Batch System 脚本是具有源文件和目标文件的步骤。源文件是用于第一个 Trillium Batch System 命令的输入数据文件。目标文件是由脚本中的最后一个 Trillium 命令创建的输出数据文件。然后，可以将该步骤复制到另一个进程中，以便与其他步骤配合使用。

下图显示了 Trillium Batch System 输入和输出文件与“数据仓库中心”中的源文件和目标文件之间的关系。

```
REM 正在运行变换器
pfcondrv -parmfile c:\tril40\us_proj\parms\pfcondrv.par
REM 正在运行语法分析器
pfprsdrv -parmfile c:\tril40\us_proj\parms\pfprsdrv.par
REM 正在运行匹配器
cfmatdrv -parmfile c:\tril40\us_proj\parms\pfmatdrv.par
```

图 1. 样本 Trillium 脚本文件

```
INP_FNAME01 c:\tril40\us_proj\data\convinp
INP_DDL01 c:\tril40\us_proj\dict\input.ddl
```

图 2. pfcondrv.par 文件的内容

```
OUT_DDNAME c:\tril40\us_proj\data\maout
DDL_OUT_FNAME c:\tril40\us_proj\dict\parseout.ddl
```

图 3. pfmatdrv.par 文件的内容

c:\Tril40\us_proj\data\convinp (源文件) ->
Trillium Batch System 步骤 -> c:\tril40\us_proj\data\maout (目标文件)

图 4. Trillium Batch System 步骤定义

导入 Trillium 元数据

要将 Trillium 元数据导入“数据仓库中心”：

1. 创建 Trillium Batch System 脚本或 JCL。可以使用任何脚本或 JCL 编写工具来创建脚本或 JCL 文件。
2. 用鼠标右键单击**仓库**，并单击**导入元数据 —> Trillium** 来打开 Trillium Batch System 窗口。
3. 在**脚本或 JCL** 字段中，输入您想运行的 Trillium Batch System 脚本或 JCL 文件的名称。
4. 在**输入文件**字段中，输入第一次在指定脚本或 JCL 文件中运行的 Trillium Batch System 程序的输入数据文件的名称。
5. 在**输入 DDL** 字段中，输入用来描述输入数据文件的输入 DDL 文件的名称。在仓库代理点上必须存在此文件。
6. 在**输出文件**字段中，输入在脚本或 JCL 文件中的最后一个 Trillium Batch System 程序的输出数据文件的名称。
7. 在**输出 DDL** 字段中，输入用来描述输出数据文件的输出 DDL 文件的名称。在仓库代理点上必须存在此文件。
8. 可选：在**输出错误文件**字段中，输入您想使用的输出错误文件的名称。此错误文件捕捉 Trillium Batch System 程序中的运行时错误。这些错误被记录在 stderr 日志中。对于本地主机，如果在此处不指定名称，则会创建缺省的输出错误文件。有关输出错误文件的详情，参见主题“错误处理”。
9. 单击**连接**选项卡。
10. 如果您正在导入的 Trillium 元数据在仓库代理点上，则单击**本地主机**。
如果您正在导入的 Trillium 元数据不在仓库代理点上，则单击**远程主机**，并指定远程主机。稍后参见本节中的主题“指定远程主机”。
11. 单击**确定**以导入 Trillium 元数据并关闭笔记本。
12. 如果脚本或 JCL 不在缺省代理点运行，则在“特性”笔记本中指定用于 Trillium Batch System 步骤的仓库代理点。

当导入操作完成时，下列仓库对象便被添加到“仓库”树中。

- Trillium Batch System.*scriptName* 模板，其中 *scriptName* 是脚本或 JCL 文件的名称。
- Trillium Batch System 进程。
- 运行用户定义程序的 Trillium Batch System 步骤。
- 导入元数据时所指定的仓库文件源和仓库文件目标。文件源和文件目标是固定文件。
- Trillium Batch System 程序组。

指定远程主机

要指定远程主机：

1. 单击**远程主机**，并输入包含您正在导入的元数据的远程系统的 TCP/IP 主机名。如果选择“远程”主机，则目标文件是作为本地文件来创建的，因为远程目标文件是不受支持的。可以添加 FTP 步骤来将远程文件传送至指定的本地目标文件。
2. 在**远程操作系统**列表中，单击您正在访问的远程主机的操作系统。
3. 在**远程用户标识**字段中，输入您正在访问的远程主机的用户标识。
4. 在**口令选项**列表中，选择您想用于您正在访问的远程主机的口令选项：

不需要口令

指定访问远程主机上的元数据不需要口令。

检索口令

指定将从用户定义程序中检索口令。

在**口令程序**字段中，输入将检索口令的口令程序的名称。程序必须驻留在仓库代理点上，并将口令写到输出文件的第一行上。

在**程序参数**字段中，输入口令程序的参数。第一个参数必须是要将口令写至的输出文件。

稍后输入口令

指定将在稍后输入口令。

在“特性”笔记本中输入口令，以运行 Trillium Batch System 程序。

映射元数据

要为源文件和目标文件创建元数据，Trillium 应读取 Trillium DDL 文件。DDL 文件被转换成下列“数据仓库中心”数据类型：

UNIX、Windows NT 操作系统的 DDL 数据类型	仓库数据类型
ASCII CHARACTER	CHARACTER(<i>n</i>)
ASCII NUMERIC	
EBCDIC CHARACTER	
EBCDIC NUMERIC	
其他类型	NUMERIC
注：仅当 Trillium Software System 在 OS/390 操作系统上运行时，才支持 EBCDIC CHARACTER 和 EBCDIC NUMERIC 数据类型。	

变量 *n* 是字符串中的字符数。

限制

可以通过“数据仓库中心”中的 Trillium DDL 和导入元数据操作来指定输入和输出 DDL 文件中的重叠字段。然而，在“数据仓库中心”中，相应的仓库源和仓库目标文件不能配合 SQL 步骤或“样本”内容使用。因为导入元数据操作会忽略分散在整个记录中的重叠字段，所以仍然可以指定这些字段，但是它们将不会被用作生成的源文件和目标文件的列。

如果指定了错误文件，则脚本的名称中不能包含任何空白。

编写 Trillium Batch System JCL 文件

如果您正在编写 Trillium Batch System JCL 文件，则必须满足下列需求。

- 作业名必须是用户标识再加上一个字符。
- 必须将作业按路径发送至被挂起的输出类。
- 运行 Trillium Batch System 程序的每个作业步骤必须包括用来定义永久数据集的 `SYSTEM DD` 语句。数据集中包含来自 Trillium Batch System 程序的错误。在提交该 JCL 之前将自动删除此数据集。有关错误处理和报告的详情，参见主题“错误处理”。

当脚本或 JCL 在远程主机上运行时，必须指定输出错误文件；否则，将不会捕捉错误消息并返回至“数据仓库中心”。在 UNIX 或 Windows 上，捕捉错误的最简单方法就是编写另一个脚本来调用 Trillium Batch System 脚本，并将标准错误用管道输送至输出文件中。

```
//SYSTEM DD UNIT=&UNIT,  
//          DISP=(MOD,CATLG,KEEP),  
//          SPACE=(400,(20,20),,ROUND),  
//          DSN=&PROJPREF.&TRILVER.&PROJECT.STDERR;
```

图 5. 包括了 `SYSTEM DD` 语句的作业步骤的示例

在 UNIX 环境变量 Windows 上编写 Trillium Batch System 脚本文件

如果 Trillium Batch System 脚本或参数文件中包含输入文件的相对路径，则用户必须将脚本文件开头的 `cd` 语句放到脚本文件的目录中。

定义 Trillium Batch System 步骤

在定义 Trillium Batch System 步骤之前，必须导入您想在进程中使用的 Trillium 元数据。要将 Trillium Batch System 添加至进程：

1. 打开进程造型程序中的进程。
2. 单击调色板上的 **Trillium Batch System** 图标。
3. 单击 **Trillium Batch System 程序** → *programName*，其中，*programName* 是您想使用的 Trillium Batch System 程序的名称。
4. 单击想要步骤出现在画布上的位置。
5. 完成 *DB2 通用数据库帮助* 中的主题“定义运行用户定义程序的步骤”中的步骤。

使用 Trillium Batch System 用户定义程序

Trillium Batch System 用户定义程序是与“DB2 数据仓库中心版本 7.2”的 Windows NT 版和 UNIX 版一起提供的。在导入 Trillium 元数据时创建的 Trillium Batch System 步骤将运行 Trillium Batch System 用户定义程序。用户定义程序将调用 Trillium Batch System 脚本或 JCL。下表包含了用于 Trillium Batch System 脚本或 JCL 的参数：

参数	值
----	---

远程主机	<ul style="list-style-type: none"> • localhost 是缺省值。如果在仓库代理点上安装了 Trillium Batch System, 则使用此值。 • 如果 Trillium Batch System 是安装在远程操作系统上的, 则使用远程主机的名称。
脚本或 JCL	脚本或 JCL 的名称
远程操作系统	<p>远程主机上的操作系统的名称。如果远程主机参数的值为 localhost, 则会忽略此参数。有效值为:</p> <ul style="list-style-type: none"> • 用于 OS/390 操作系统的 MVS • 用于 AIX、SUN Solaris、HP-UX 和 NUMA/Q 操作系统的 UNIX • 用于 Windows NT 或 2000 操作系统的 WIN
远程用户标识	有权执行远程命令的用户标识。如果 RemoteHostName 的值为 localhost, 则会忽略此参数。
口令选项	<p>获取口令的方法。有效值为:</p> <p>ENTERPASSWORD 如果口令是在下一个参数中传送的, 则使用此值。</p> <p>PASSWORDNOTREQUIRED 如果不需要口令, 则使用此值。</p> <p>GETPASSWORD 如果程序名是在下一个参数中传送的, 则使用此值。</p> <p>限制:</p> <ul style="list-style-type: none"> • 程序必须驻留在代理点上, 将口令写到输出文件的第一行上并返回 0 (如果它运行成功的话)。 • “口令”参数的值必须是口令程序的名称。 • “程序”参数的值必须是用双引号引起来的字符串。 • 字符串中的第一个参数必须是将把口令写入的输出文件的名称。
口令	有效值为口令或口令程序名。口令程序对于仓库代理点必须是本地的。
程序参数	口令程序的参数。
输出错误文件	输出错误文件的名称。
注: 此表中的所有参数的数据类型都是 CHARACTER。	

错误处理

在 Windows NT 和 UNIX 操作系统上, Trillium Batch System 程序将错误消息写入标准错误 (stderr) 文件中, 而在 OS/390 操作系统上, 该程序将错误消息写入 SYSTEMR 数据集中。

在 Windows NT 或 UNIX 操作系统上，要捕捉 Trillium Batch System 程序中的错误，必须将标准输出重定向至输出错误文件。

在 OS/390 操作系统上，要捕捉 Trillium Batch System 程序中的错误，JCL 中必须包括 SYSTEM DD 语句。

如果在“导入元数据”窗口中指定输出错误文件名，则必须将标准错误输出重定向或者存储至错误文件。“数据仓库中心”将读取该文件，并报告包含字符串 ERROR 的所有行以作为错误消息。所有 Trillium Batch System 程序错误消息都包含字符串 ERROR。

如果在仓库代理点上运行的脚本或 JCL 中未指定输出错误文件，则“数据仓库中心”将自动创建一个文件名，并将标准错误输出重定向至该文件。如果遇到了错误，则不会删除错误文件。错误文件被存储在由环境变量 VWS_LOGGING 指定的目录中。文件名为 tbsudp-*date*-*time*.err，其中，*date* 是创建文件时的系统日期，*time* 是创建文件时的系统时间。以下文件名显示了输出错误文件名的格式：

tbsudp-021501-155606.err

错误返回码

错误号	描述
0	成功
4	警告。或者是无法擦除口令文件，或者是 Trillium Batch System 用户定义程序在访问临时文件时发生了内部错误。 请检查口令文件的状态，或者检查在由环境变量 VWS_LOGGING 指定的目录下创建的所有临时文件的状态。
8	参数的数目或值不正确。阅读日志文件或文档以便了解正确的语法。
12	当 Trillium Batch System 用户定义程序通过 FTP 来与远程主机连接时发生了问题。检查 FTP 连接或主机名、用户标识和口令。
16	Trillium Batch System 用户定义程序不能创建日志或内部文件。 检查用户是否具有正确的权限，并检查磁盘是否已满。
20	不能执行 OS/390 JCL，或者当 Trillium Batch System 用户定义程序通过 FTP 删除或获取 OS/390 中的文件时出错。 检查 JESLogFile 以便找出原因。
48	找不到环境变量 VWS_LOGGING，或者不能创建日志文件。有关详情，请检查日志文件。
56	不能执行 Windows NT 或 UNIX 脚本，或者当 Trillium Batch System 用户定义程序与远程主机连接时出错。检查连接或主机名、用户标识和口令。

500	脚本或 JCL 文件返回了错误，或者虽然它没有返回错误，但是错误文件中包含数据。有关详情，请检查日志文件。在 OS/390 上，还要检查 JESLogFile。
-----	--

日志文件

当 Trillium Batch System 用户定义程序运行时，“数据仓库中心”将所有诊断信息存储在日志文件中。日志文件的名称为 `tbsudp-date-time.log`，其中，`date` 是创建文件时的系统日期，`time` 是创建文件时的系统时间。日志文件是在由代理点上的环境变量 `VWS_LOGGING` 指定的目录中创建的。如果成功地运行了 Trillium Batch System 用户定义程序，则会删除日志文件。

MQSeries 与“数据仓库中心”集成

“数据仓库中心”现在允许您以 DB2 数据库视图的形式来访问 MQSeries 消息队列中的数据。提供了一个向导来创建表函数或 DB2 视图，可以通过它来访问数据。每条 MQSeries 消息都被视作定界字符串，它是根据您的规范来进行语法分析的，并返回结果行。另外，可将 XML 文档的 MQSeries 消息作为仓库源来访问。通过使用“数据仓库中心”，可以从 MQSeries 消息队列和“DB2 XML Extender 文档访问定义”(DAD) 文件中导入元数据。

为 MQSeries 消息创建视图

需求

DB2 通用数据库版本 7.2。

DB2 仓库管理器版本 7.2

MQSeries 支持。有关 MQSeries 需求的详情，请参考第159页的『MQSeries』。有关设置仓库源的信息，参见用户定义函数的设置部分。

限制

- 当对仓库源数据库进行编目时，数据库别名是在代理机器上编目的。然而，当创建 MQSeries 和 XML 视图时，“数据仓库中心”假定在客户机上也定义了数据库别名，并且将试图使用仓库源数据库用户标识和口令来与它连接。如果成功的话，就将调用该向导，并且您就可以创建视图。如果不成功，则会显示警告消息，并且必须在向导中编目或者选择另一个数据库别名。

- 要了解 MQ 消息的最大长度，请参考《发行说明》的 SQL Reference 一节。

为 MQSeries 消息创建视图

要为 MQSeries 消息创建视图：

1. 从“数据仓库中心”窗口中，展开仓库源树。
2. 展开要包含视图的仓库源。
3. 用鼠标右键单击视图文件夹，并单击为 **MQSeries 消息创建...**

MQSeries 向导打开。当完成该向导时，就在“数据仓库中心”中创建了新视图。选择了视图，就访问到 MQSeries 队列，并且根据向导中的规范来将每条消息作为定界字符串进行了语法分析。

导入 MQSeries 消息和 XML 元数据

需求

DB2 通用数据库版本 7.2。

DB2 XML Extender 版本 7.2。

MQSeries 支持。有关 MQSeries 需求的详情，请参考第159页的『MQSeries』。有关设置仓库源的信息，参见用户定义函数的设置部分。

限制

如果目标表中存在主键或外键，则导入将失败。在导入之前，必须在“数据仓库”中人工删除这些键的定义。

导入 MQSeries 消息和 XML 元数据

要将 MQSeries 元数据导入“数据仓库中心”：

1. 准备仓库目标数据库：
 - 必须定义仓库目标并注册和启用变换器。
 - 必须对 DB2 XML Extender 启用仓库目标。有关详情，参考“DB2 XML Extender 版本 7.2 发行说明”。
 - 创建“XML Extender 数据访问定义”(DAD) 文件来告诉“数据仓库中心”如何将 XML 文档的内容映射至仓库表。通过对数据库使用 DAD 文件来启用 XML 集合。有关详情，参考“DB2 XML Extender 版本 7.2 发行说明”。
2. 用鼠标右键单击**仓库**，并单击**导入元数据** → **MQSeries** 来打开**导入元数据**窗口。
3. 在 **AMI 服务**字段中，输入将消息发送至或者从其中检索消息的服务点。
4. 在 **AMI 策略**字段中，输入消息传递系统将用来执行操作的策略。
5. 在 **DAD 文件**字段中，输入 DB2 XML Extender DAD 文件的名称，或者通过单击省略号 (...) 来搜索要选择的文件。此文件必须是本地的。
6. 在 **仓库目标**字段中，从组合框列表中选择将在其中运行步骤的仓库目标的名称。必须已经定义了仓库目标。
7. 在 **模式**字段中，输入模式名，以限定 DAD 文件中没有限定符的表名。缺省模式被当作您先前选择的仓库目标的登录用户标识。
8. 选择**目标选项**：
 - 如果想要该步骤在运行时替换目标表内容，则单击**替换表内容**单选按钮。
 - 如果想要该步骤在运行时追加至目标表内容，则单击**追加表内容**单选按钮。
9. 单击**确定**。
 - “导入元数据”窗口关闭。

当导入操作完成时，下列仓库对象便被添加到“仓库”树中。

- 称为 MQSeries 和 XML 的主题区。
- 称为 MQSeries 和 XML 的进程。

- 称为 MQSeries 和 XML 的用户定义程序。
- 在 DAD 文件中描述了所有仓库目标表的定义。
- <ServiceName>.<DAD 文件库名 >.<仓库目标名 > 步骤。
- <ServiceName>.<DAD 文件库名 > 程序模板。

如果仓库目标代理点与本地机器不同，则必须更改步骤参数：

1. 用鼠标右键单击该步骤，并选择**特性**。单击特性笔记本中的**参数**选项卡。
2. 将 DAD 文件参数的名称更改为远程仓库目标代理点上的 DAD 文件的名称。
3. 确保“处理选项”选项卡中的“代理点”包含想要的代理点。

使用 MQSeries 用户定义程序

MQSeries 和 XML 存储过程被称为 **MQXMLXF**，并且是与“DB2 数据仓库中心版本 7.2” Windows NT 版和 UNIX 版一起提供的。在导入 MQSeries 和 XML 元数据时所创建的步骤将运行该存储过程。下表描述了该存储过程的参数：

参数	值
MQSeries ServiceName	是将消息发送至或者从其中检索消息的服务点的名称。
MQSeries PolicyName	是消息传递系统将用来执行操作的策略的名称。
DAD 文件名	DB2 XML Extender DAD 文件的名称
TargetTableList	用逗号隔开的步骤的目标表的列表
选项	REPLACE 或 APPEND
RUN ID	步骤版本号（用于日志记录）
注：此表中的所有参数的数据类型都是 CHARACTER。	

如果选项的值为 REPLACE，则存储过程将删除目标表中的所有行。存储过程还会调用 DB2 XML Extender 存储过程来填充所有现存 MQSeries 消息的目标表。

错误返回码

当运行步骤时，存储过程可能返回错误码 SQLCODE -443 和 SQLSTATE 38600。要诊断该错误，参见下表以获取可能的诊断文本。

错误号	描述
AMIRC=xxxx;<日志文件名>	xxxx 是来自 AMI 层的返回码。有关详情，参考 MQSeries 文档。<日志文件名> 指示日志文件的位置。
XMLRC=xxxx;<日志文件名>	xxxx 是来自 DB2 XML Extender 的返回码。有关返回码的描述，参考 DB2 XML Extender 文档。<日志文件名> 指示日志文件的位置。
SQLCODE=xxxx;<日志文件名>	xxxx 是执行 SQL 请求时返回的非零 SQLCODE。<日志文件名> 指示日志文件的位置。

有关所有错误的详情，参考日志文件。

错误日志文件

当 MQXMLXF 运行时，“数据仓库中心”将所有诊断信息存储在日志文件中。日志文件的名称为 mqxf<nnnnnnnn>.log，其中，<nnnnnnnn> 是传送给存储过程的运行标识。

“数据仓库中心”将在由 VWS_LOGGING 环境变量指示的目录中创建该文件。如果未定义此环境，则将在临时目录中创建日志文件。

为了在 Unix 平台上的存储过程中可以看到 VWS_LOGGING 环境变量，应该在执行 **db2start** 命令之前使用 **db2set** 命令来将 VWS_LOGGING 添加至 DB2ENVLIST 环境变量。下图是一个示例环境命令。

```
db2set DB2ENVLIST="AMT_DATA_PATH VWS_LOGGING"
```

图 6. 环境变量命令示例

如果成功地运行了该步骤，则会删除日志文件。

Microsoft OLE DB 和“数据事务服务”支持

“数据仓库中心”现在允许您以 DB2 数据库视图的形式来访问 OLE DB 提供器中的数据。可以使用随“数据仓库中心”提供的“OLE DB 辅助程序”向导来创建 DB2 OLE DB 表函数或 DB2 视图，可以通过它来访问数据。

“Microsoft 数据变换服务”（DTS）允许您在 OLE DB 源和目标之间导入、导出和转换数据，以便构建数据仓库和数据超市。DTS 是与 Microsoft SQL Server 一起安装的。所有 DTS 任务都存储在 DTS 软件包中，可以通过对“DTS 软件包”使用 Microsoft OLE DB Provider 来运行和访问该软件包。因为可以从 DTS 访问作为 OLE DB 源的软件包，所以还可使用为 OLE DB 数据源创建视图的方法来使用“OLE DB 辅助程序”向导来为 DTS 软件包创建视图。在运行时访问视图时，将执行 DTS 软件包，并且 DTS 软件包中的任务的目标表就成为所创建的视图。

在“数据仓库中心”中创建视图之后，就可以象使用其他任何视图一样使用该视图。例如，在 SQL 步骤中可以将 DB2 表与 OLE DB 源进行连接。当您在 SQL 步骤中使用已创建的视图时，就会调用 DTS 提供器并运行 DTS 软件包。

软件需求:

- DB2 通用数据库 Windows NT 版的版本 7.2 作为仓库目标数据库
- DB2 仓库管理器版本 7.2
- 如果仓库目标数据库是在版本 7.2 之前创建的，则在安装 DB2 UDB Windows NT 版的版本 7.2 之后必须运行 db2updv7 命令。
- 对仓库源数据库进行编目时，数据库别名是在仓库代理点上编目的。然而，启动向导时，“数据仓库中心”假定在客户机工作站上还定义了数据库别名，并且将试图使用仓库源数据库用户标识和口令来与它进行连接。如果连接成功的话，就将启动该向导，并且可以创建视图。如果连接不成功，则会显示警告消息，并且必须在向导中编目或者选择另一个数据库别名。
- 要在 DST 软件包中标识特定的表，必须在创建目标表的 DataPumpTask 的“工作流特性”窗口的“选项”选项卡中选择 **DSO 行集提供器**复选框。如果打开多个 DSO 行集提供器属性，则只使用所选的第一个步骤的结果。如果选择了视图，将返回其目标表的行集，而忽略您在后续步骤中创建的所有其他行集。

- 当对向导输入表名时，使用步骤名，它显示在任务的“ workflow 特性”笔记本的“选项”页上。
- DTS 软件包连接字符串与 `dtsrun` 命令具有相同的语法。

为 OLE DB 表函数创建视图

要为 OLE DB 表函数创建视图：

1. 从“数据仓库中心”窗口中，展开仓库源树。
2. 展开要包含视图的仓库源。
3. 用鼠标右键单击视图文件夹，并单击为 **OLE DB 表函数创建**。

“OLE DB 辅助程序”向导打开。该向导将指导您完成在仓库源数据库中创建新视图的任务。

为 DTS 软件包创建视图

要为 DTS 软件包创建视图：

1. 从“数据仓库中心”窗口中，展开仓库源树。
2. 展开要包含视图的仓库源。
3. 用鼠标右键单击视图文件夹，并单击 **DTS 软件包的 Microsoft OLE DB Provider**。

“OLE DB 辅助程序”向导打开。该向导将指导您完成在仓库源数据库中创建新视图的任务。

有关 DTS 的详情，参见 Microsoft Platform SDK 2000 文档，该文档中包括关于如何构建向导与 DTS 提供器相连所需要的提供器字符串的详细说明。

将增量提交与替换配合使用

在填充类型为“替换”的步骤中，仅当插入新数据时才使用增量提交。在单个提交作用域中将删除旧数据。如果需要删除数据而不生成日志记录，则在运行具有追加填充类型的 SQL 步骤之前应运行装入空文件的步骤。

组件跟踪数据文件名

“数据仓库中心”在 Windows NT 上编写这些文件：

AGNTnnnn.Log 包含跟踪信息。*nnnn* 是仓库代理程序的数字进程标识，该标识可以是 4 位或 5 位，这取决于操作系统。

AGNTnnnn.Set 包含代理程序的环境设置。*nnnn* 是仓库代理程序的数字进程标识，该标识可以是 4 位或 5 位，这取决于操作系统。

缺省目录为 `x:\program files\sqllib\logging`，其中 *x* 是安装了 DB2 的驱动器。

在 AIX 和 Solaris 操作环境上, Sybase 源所需要的开放式客户机

在第 3 章『设置仓库源 AIX』中的“表 3 AIX 和 Solaris 操作环境上受支持的数据源的连通性需求”和“表 4 Solaris 操作环境上受支持的数据源的连通性需求”中, 在“如何连接:”列中, Sybase 的条目应该包含附加步骤。附加步骤就是下面所显示的步骤 3。

3. 安装“开放式客户机”

注意, 连接至 Windows NT 或 Windows 2000 平台上的 Sybase 源时需要“开放式客户机”。

已校正的样本条目

在数据仓库中心管理指南的『第 3 章 设置仓库源』中的图 6、8、10 和 11 中包含指向“驱动程序”属性的不正确路径。下列路径才是正确的。

图 6

```
Driver=/home/db2_07_01/3.6/odbc/lib/ivinf12.so
```

图 8

```
Driver=/home/db2_07_01/3.6/odbc/lib/ivsyb1112.so
```

图 10

```
Driver=/home/db2_07_01/3.6/lib/ivor814.so
```

图 11

```
Driver=/home/db2_07_01/3.6/odbc/lib/ivmsss14.so
```

第 3 章 设置仓库源

将 Microsoft Access 中的“备忘录”字段映射至仓库源

Microsoft Access 数据库的“备忘录”字段在“数据仓库中心”源中被表示为数据类型 LONG VARCHAR, 其列大小超过了 1 GB。为了支持实际的系统配置, “数据仓库中心”将截断超过 128 KB 的值。为了避免在仓库源中截断“备忘录”字段值, 当您在某个步骤中使用表之前, 将接收“备忘录”字段数据的列的数据类型从 LONG VARCHAR 更改为 CLOB。如果您没有更改该列的数据类型, 则大于 128 KB 的任何值都将被截断。

OS/390 和 OS/400 需要对 CLOB 数据类型的 DRDA 支持。从 DB2 版本 6 开始, OS/390 就支持 CLOB 数据类型。从版本 4、发行版 4、DB 修订包 4 或更新版本 (PTF SF99104) 开始, OS/400 就支持 CLOB 数据类型了。对于 OS/400, 日期标为 1999 年 2 月的安装磁盘“版本 4、发行版 4”还包含对 CLOB 数据类型的支持。

第 10 章 维护仓库数据库

将表链接至 DB2 UDB RUNSTATS 程序的步骤子类型

RUNSTATS 程序的步骤子类型对仓库目标进行读写操作。在为步骤定义值之前，在“进程模型”窗口中将目标与步骤子类型链接。

缺省仓库控制数据库

在 Windows NT 或 Windows 2000 上进行典型 DB2 安装期间，如果未在 Windows NT 注册表中标识任何活动的数据库控制数据库，DB2 会创建并初始化“数据仓库中心”的缺省数据库控制数据库。初始化是一个过程，在该过程中，“数据仓库中心”将创建存储“数据仓库中心”元数据所必需的控制表。

缺省数据库控制数据库被命名为 DWCTRLDB。在登录时，缺省情况下“数据仓库中心”会将 DWCTRLDB 指定为数据库控制数据库。要查看将要使用的数据库控制数据库的名称，单击“数据仓库中心登录”窗口上的“高级”按钮。

“仓库控制数据库管理”窗口

“数据仓库控制数据库管理”窗口是在 Windows NT 或 Windows 2000 上进行典型 DB2 安装时安装的。还可使用此窗口来切换活动的数据库控制数据库，创建和初始化新的数据库控制数据库，以及迁移已与 IBM Visual Warehouse 配合使用的数据库控制数据库。以下各节讨论了每一个这样的活动。

在使用“数据仓库控制数据库管理”窗口之前停止数据库服务器。

更改活动的仓库控制数据库

如果想要使用除活动数据库控制数据库之外的数据库控制数据库，可使用“数据仓库控制数据库管理”窗口来将该数据库注册为活动的控制数据库。如果在登录“数据仓库中心”时指定了不同于活动数据库控制数据库的名称，将会接收到一个错误，指出指定的数据库与由数据库服务器指定的数据库不匹配。

要注册该数据库：

1. 单击**开始** → **程序** → **IBM DB2** → **数据仓库控制数据库管理**。
2. 在**新控制数据库**字段中，输入想要使用的控制数据库的名称。
3. 在**模式**字段中，输入用于数据库的模式名。
4. 在**用户标识**字段中，输入访问数据库所需的用户标识的名称。
5. 在**口令**字段中，输入该用户标识的口令名。
6. 在**验证口令**字段中，再次输入该口令。
7. 单击**确定**。

窗口仍处于打开状态。**消息**字段显示一些消息，指示注册过程的状态。

8. 在该过程完成之后，关闭该窗口。

创建和初始化仓库控制数据库

如果想要创建非缺省的数据仓库控制数据库，则可在安装过程中创建它或是安装之后使用“数据仓库控制数据库管理”窗口来创建它。可使用安装过程来在与数据仓库服务器所在的同一工作站上创建数据库或在另一工作站上创建数据库。

要更改在安装期间创建的数据仓库控制数据库的名称，必须执行定制安装并在“定义本地数据仓库控制数据库”窗口上更改该名称。安装过程将使用您指定的名称来创建数据库，初始化该数据库以与“数据仓库中心”配合使用，并将该数据库注册为活动的数据仓库控制数据库。

要在安装期间在不同于安装有数据仓库服务器的工作站上创建数据仓库控制数据库，可在定制安装期间选择“数据仓库本地控制数据库”。安装过程将会创建该数据库。安装之后，必须遵循第215页的『更改活动的仓库控制数据库』中的步骤来在数据仓库服务器工作站上使用“数据仓库控制数据库管理”窗口。指定您在安装期间指定的数据库名。将初始化该数据库以与“数据仓库中心”配合使用，并将其注册为活动的数据仓库控制数据库。

要在安装过程之后创建和初始化数据仓库控制数据库，在数据仓库服务器工作站上使用“数据仓库控制数据库管理”窗口。如果新的数据仓库控制数据库不在数据仓库服务器工作站上，则必须首先创建该数据库并在数据仓库服务器工作站上对其进行编目。然后遵循第215页的『更改活动的仓库控制数据库』中的步骤。指定您在安装期间指定的数据库名。

在登录“数据仓库中心”时，单击**高级按钮**并输入活动数据仓库控制数据库的名称。

创建版本化的 SQL 步骤

在创建版本化的 SQL 步骤时，根据使用情况，您可能想要考虑对版本列创建非唯一索引以加快删除版本的速度。只应对大型数据仓库表考虑此问题，因为在插入的行数目较少时会影响插入性能。

在“过程模型”窗口中更改源和目标

在“过程模型”窗口中，如果更改了源或目标，则所作的更改会立即自动保存。如果进行任何其他更改（如添加步骤），则必须显式地保存该更改以使该更改成为永久性更改。要保存更改，单击“过程 → 保存”。

为“数据仓库中心”对象添加描述

在“数据仓库中心”的笔记本的“描述”字段中，可指定最多 254 个字符。此最大数目取代了在联机帮助中指定的最大长度。

运行样本内容

- 不能在平面文件源上成功运行使用 AS/400 代理程序的“采样内容”请求。尽管您可创建平面文件源并尝试使用 AS/400 代理程序发出 sampleContent 请求，但该请求将会失败。

- 您可能在过程造型程序中的数据仓库目标上运行“采样内容”时收到错误消息。此错误与过程中的数据仓库源、数据仓库目标和步骤的公共代理点的可用性有关。步骤的可用代理点列表是从数据仓库源代理点、数据仓库目标代理点以及可用于此特定步骤（这些步骤是在代理点特性笔记本的最后一页中选择的）的代理点的公共代理点获得的。例如，您想要查看运行 FTP Put 程序 (VWPRCPY) 的过程的“采样内容”。必须对代理点定义中的代理点选择在过程中使用的步骤。在对“目标”文件运行“采样内容”时，通常会使用所选列表上的第一个代理点。但是，数据库维护操作可能会影响所列示的代理点的次序。如果所选择的代理点未驻留在源或目标文件所在的系统上，采样内容将会失败。

编辑“创建 DDL SQL”语句

在尝试对处于开发方式的步骤编辑目标表的“创建 DDL SQL”语句时，您会看到以下令人误解的消息：“对“创建 DDL SQL”语句的任何更改将不会反映在表定义上或实际物理表上。要继续吗？”

该更改将反映在实际的物理表中。忽略该消息并继续更改“创建 DDL”语句。

对于处于开发方式的步骤，此消息的校正版本应为如下内容：“对“创建 DDL SQL”语句的任何更改将不会在表定义中反映出来。要继续吗？”

对于处于测试或生产方式的步骤，该消息是正确的。“数据仓库中心”将不会更改在将步骤提升为测试方式时创建的物理目标表。

迁移 Visual Warehouse 商业视图

如果想要将 Visual Warehouse 元数据同步商业视图迁移至“数据仓库中心”，则在迁移数据仓库控制数据库之前将商业视图提升为生产状态。如果商业视图处于生产状态，则它们的调度会迁移至“数据仓库中心”。如果商业视图并非处于生产状态，则它们将在处于测试状态时迁移，但它们的调度不会迁移。不能将已迁移的步骤提升为生产状态。必须在“数据仓库中心”中再次创建同步步骤，并删除已迁移的步骤。

生成目标表和主键

虽然“数据仓库中心”生成了步骤的目标表，但它不会生成目标表的主键。某些变换器（如“滑动平均值”）将生成的表用作源表，并且要求源表具有主键。在将生成的表与变换器配合使用之前，通过在“DB2 控制中心”中用鼠标右键单击该表，并单击**改变**来定义该表的主键。

使用 Merant ODBC 驱动程序

要使用 Merant ODBC 驱动器来访问 Windows NT 上的“Microsoft SQL 服务器”，验证系统路径是否包含 `sqllib\odbc32` 目录。

新的 ODBC 驱动程序

如果您将使用已经链接的“数据仓库中心” AIX 或 Sun 代理程序来访问 Merant ODBC 源并且还将访问 DB2 数据库，则按如下所示来更改 .odbc.ini 文件的“DB2 源”一节中 “Driver=” 属性的值：

AIX: 驱动程序名为 /usr/lpp/db2_07_01/lib/db2_36.o

AIX 的样本 ODBC 源条目：

```
[SAMPLE] Driver=/usr/lpp/db2_07_01/lib/db2_36.o
Description=DB2 ODBC Database
Database=SAMPLE
```

Sun: 驱动程序名为 /opt/IBMdbs2/V7.1/lib/libdb2_36.so

Sun 的样本 ODBC 源条目：

```
[SAMPLE] Driver=/opt/IBMdbs2/V7.1/lib/libdb2_36.so
Description=DB2 ODBC Database
Database=SAMPLE
```

在 OS/2 数据库中定义仓库源或目标

在对 OS/2 数据库定义数据仓库源或数据仓库目标时，用大写字母输入该数据库的名称。

监控仓库控制数据库的状态

“DB2 控制中心”或“命令行处理器”可能会指示数据仓库控制数据库处于不一致状态。应该会出现此状态，原因是它指示数据仓库服务器未将其初始启动消息提交至数据仓库记录器。

将“SQL 辅助程序”与 TBC_MD 样本数据库配合使用

在包含在 TBC_MD 数据库中的数据入库样本中，不能使用“SQL 辅助程序”来更改“选择方案” SQL 步骤中的 SQL，原因是该 SQL 是在由“SQL 辅助程序”生成它之后才编辑的。

使用 FormatDate 函数

要使用 FormatDate 函数，在 SQL 步骤的“特性”笔记本“SQL 语句”页上单击**构建 SQL**。

FormatDate 函数的输出的数据类型为 varchar(255)。不能通过在“函数参数 - FormatDate”页上的类别列表中选择“日期”、“时间”或“日期/时间”来更改数据类型。

更改语言设置

在 AIX 和“Solaris 操作环境”中，安装过程设置要对信息目录发布的语言，并将它导出至“OLAP 集成服务器”。如果想要以不同于在安装期间设置的语言使用这些函数，通过在一行上输入以下命令来创建以下软链接：

在 AIX 上

```
/usr/bin/ln -sf /usr/lpp/db2_07_01/msg/locale/flgnxolv.str
/usr/lpp/db2_07_01/bin/flgnxolv.str
```

locale 以 *xx_yy* 格式出现的语言的语言环境名称

在 Solaris 操作环境上

```
/usr/bin/ln -sf /opt/IBDb2/V7.1/msg/locale/flgnxolv.str
/opt/IBDb2/V7.1/bin/flgnxolv.str
```

locale 以 *xx_yy* 格式出现的语言的语言环境名称

使用“生成键表”变换器

使用“生成键表”变换器的**更新键列中的值**选项时，变换器仅更新表中那些没有键值的行。（即，这些值为空）。在其他行插入到表中时，这些键值为空，直到再次运行变换器为止。

要避免此问题，使用下列方法：

- 在变换器的初次运行之后，使用**替换所有值**选项来重新创建所有行的键。

维护与数据库的连接

在管理本地或远程数据库的 DB2 服务器停止并重新启动时，数据仓库服务器不会维护与这些数据库的连接。如果停止并重新启动 DB2 的话，则还会停止并重新启动数据仓库服务。

设置远程“数据仓库中心”客户机

如果是安装“DB2 管理客户机”和“数据入库工具”以在不同于包含数据仓库服务器的工作站上设置“数据仓库中心”管理客户机，则必须添加 TCP/IP 端口号，数据仓库服务器工作站会在该端口号上监听客户机工作站的 *services* 文件。将一项添加到 *services* 文件中，如下所示：

```
vwkernel          11000/tcp
```

定义“DB2 VM 版”仓库源

在对 DB2 VM 版数据库（它是通过 DRDA 网关访问的）定义数据仓库源时，则对 CLOB 和 BLOB 数据类型的使用有一些限制：

- 不能使用“采样内容”功能来查看 CLOB 和 BLOB 数据类型的数据。
- 不能将 CLOB 和 BLOB 数据类型的列与 SQL 步骤配合使用。

此限制是对 DB2 VM 版的版本 5.2 服务器的已知限制，它不允许使用 DRDA 来将 LOB 对象传送至 DB2 版本 7 客户机。

定义“DB2 VM 版”或“DB2 VSE 版”目标表

在“数据仓库中心”中定义 DB2 VM 版或 DB2 VSE 版目标表时，不要选择**授予公用**复选框。“数据仓库中心”生成的 GRANT 命令语法在 DB2 VM 版和 DB2 VSE 版上是不受支持的。

启用定界标识符支持

要在 Windows NT 上启用对 Sybase 和“Microsoft SQL 服务器”的定界标识符支持，在“ODBC 驱动程序安装”笔记本的“高级”页中选择启用带引号的标识符复选框。

要在 UNIX 上对启用对 Sybase 的定界标识符支持，在 .odbc.ini 文件中编辑要包括连接属性 EQI=1 的 Sybase 数据源。

Data Joiner 错误指示绑定问题

将 DataJoiner 与 DB2 版本 7.1 + 修订包 2 或更新版本配合使用的客户可能会遇到指示绑定问题的错误。

例如，当将 DataJoiner 源与“数据仓库中心”版本 7 代理程序配合使用时，您可能会遇到类似以下的错误：

```
DWC07356E An agent's processing of a command of type "importTableNames" failed for
          edition "0" of step "?".
SQL0001N Binding or precompilation did not complete successfully.
SQL0001N Package "NULLID.SQLL6D05" was not found.  SQLSTATE=51002 RC = 7356  RC2 = 8600
```

要校正该问题，应将下列各行添加到 db2cli.ini 文件中：

```
[COMMON]
DYNAMIC=1
```

在 UNIX 系统上，db2cli.ini 文件位于 .../sqllib/cfg 目录中。在 Windows NT 上，db2cli.ini 文件位于 .../sqllib 目录中。

用“数据仓库中心”设置和运行复制

1. 用“数据仓库中心”设置和运行复制要求“复制控制”表同时存在于“数据仓库控制”数据库和“数据仓库目标”数据库之上。

复制要求“复制控制”表同时存在于“控制”和“目标”数据库上。“复制控制”表可在 ASN 模式中找到，它们全都以 IBMSNAP 开头。“复制控制”表是在通过“控制中心”定义“复制源”时在数据库上自动创建的（如果这些“控制”表尚未存在的话）。注意，“控制”表还必须存在于“目标数据库”上。要获取在目标数据库上创建的一组“控制”表，可使用“控制中心”创建一个“复制源”，然后除去该“复制源”，而只留下“控制”表。或者，可使用 DJRA（Data Joiner Replication Administration）产品来仅定义控制表。

2. 安装和使用 DJRA

如果想要或是需要使用 DJRA 来定义控制表的话，将需要首先安装它。DJRA 是作为 DB2 的部分交付的。要安装 DJRA，转至 d:\sqllib\djra 目录（其中安装有 DB2），并单击 djra.exe 软件包。这将会在您的系统上安装 DJRA。要在此之后访问 DJRA，在 Windows NT 上，在开始菜单中单击“DB2 Windows NT 版”选项，然后选择“复制”，接着选择“复制管理工具”。DJRA 接口与通常的 NT 应用程序有些不同。对于它执行的每个功能，它都会创建一组要运行的 SQL，但不会执行它。用户必须人工保存生成的 SQL，然后选择“执行 SQL”功能来运行该 SQL。

3. 进行设置以运行 Capture 和 Apply

对于您正在测试的系统，参见 Replication Guide and Reference Manual 以获取有关配置系统来运行 Capture 和 Apply 程序的指导。必须在其中将使用 Capture 和

Apply 程序的每个数据库上对这些程序进行绑定。注意，您不需要创建口令文件。
“数据仓库中心”将为“复制预订”自动创建一个口令文件。

4. 在“控制中心”中定义“复制源”

使用“控制中心”来定义“复制源”。“数据仓库中心”支持五种类型的复制：用户副本、时间点、基聚合、更改聚合和分级表（CCD 表）。“用户副本”、“时间点”和“压缩分级表”类型要求复制源表具有主键。其他复制类型则不需要如此。在选择要定义为“复制源”的输入表时记住这一点。“复制源”实际上是原始源表和创建的 CD（更改数据）表的定义，该 CD 表用来在数据更改被移至目标表之前保存这些更改。在“控制中心”中定义“复制源”时，一个记录被写至 ASN.IBMSNAP_REGISTER，以定义源及其 CD 表。CD 表是同时创建的，但一开始它未包含任何数据。在定义“复制源”时，可选择仅包括后映象列还是包括前映象列和后映象列两者。这些选择是通过“控制中心复制源”界面的复选框来作出的。然后，对前后映射列的选择将转换成新 CD 表中创建的列。在 CD 表中，后映象列具有与其原始源表列名相同的名称。前映象列也有同样的名称，但在列名前有一个 'X'，以确保该名称是唯一的。

5. 将“复制源”导入到“数据仓库中心”中

一旦在“控制中心”中创建了“复制源”，就可将其导入到“数据仓库中心”中。在导入源时，一定要单击内容为“可复制的表”的复选框。这会使“数据仓库中心”查看 ASN.IBMSNAP_REGISTER 表中的记录以了解哪些表已被定义为“复制源”。

6. 在“数据仓库中心”中定义“复制步骤”

在过程造型者上，选择下列五个“复制”类型的其中一个：基聚合、更改聚合、时间点、分级表或用户副本。如果想要定义基聚合或更改聚合复制类型，参见下面有关如何在“数据仓库中心”中设置“基聚合”或“更改聚合”复制的部分。为“复制”类型选择适当的“复制源”。如上所述，复制类型“用户副本”、“时间点”和“压缩分级表”要求输入源具有主键。将“复制源”连接至“复制步骤”。打开“复制步骤”上的特性。转至“参数”选项卡。选择想要的列。选择该复选框以创建一个目标表。选择“数据仓库”目标。转至“处理选项”并填入参数。按“确定”。

7. 启动 Capture 程序

在 DOS 窗口中，输入：ASNCCP 源数据库 COLD PRUNE

COLD 参数指示“COLD 启动”且将删除 CD 表中所有的现存数据。PRUNE 参数使 Capture 程序维护 IBMSNAP_PRUNCNTL 表。让 Capture 程序继续运行。要退出该程序时，可在其 DOS 窗口中按 Ctrl-Break 键来停止它。注意，您需要在启动 Apply 程序之前启动 Capture 程序。

8. 将复制步骤提升为测试

返回到“数据仓库中心”中，对于定义的“复制步骤”，将该步骤提升为“测试”方式。这会导致“复制预订”信息被写至“复制控制”表。您将会看到被添加至 IBMSNAP_SUBS_SET、IBMSNAP_SUBS_MEMBR、IBMSNAP_SUBS_COLS 和 IBMSNAP_SUBS_EVENT 以支持预订的记录。还将在目标数据库中创建目标表。如果复制类型是用户副本、时间点或压缩分级表，则目标表上需要有主键。转至“控制中心”以创建“主键”。注意，某些复制目标表还需要各种列的唯一索引。“数据仓库中心”中存在用来在创建表时创建这些唯一索引的代码，所以您不必自己创建它们。注意，尽管在“控制中心”中定义了主键且该列的唯一索引已经存在，您在创建该主键时还是会遇到警告消息。忽略此警告消息。

9. 将复制步骤提升为生产

在“提升为生产”期间未对复制预订作任何更改。严格说来，这是一个“数据仓库中心”操作，象任何其他步骤一样。

10. 运行复制步骤

在“复制步骤”已提升为“测试”方式之后，可运行该步骤。在对源表作出任何更改之前，进行初始运行。转至“工作中”(WIP)部分，并选择“复制步骤”。运行它。运行该步骤时，IBMSNAP_SUBS_EVENT表中的事件记录被更新，而IBMSNAP_SUBS_SET中的预订记录被置为活动状态。该预订应立即运行。该预订运行时，“代理程序”会调用Apply程序以处理活动的预订。如果在此以后更新了原始源表，则更改过的数据将会移到CD表中。如果在这之后运行复制步骤，从而再次运行Apply程序，则更改过的数据将会从CD表移至目标表。

11. 将复制步骤降为测试

在“降为测试”期间未进行任何复制预订更改。严格说来，这是一个“数据仓库中心”操作，象任何其他步骤一样。

12. 将复制步骤降为开发

在将“复制步骤”降为开发方式时，会从“复制控制”表中除去预订信息。对于该特定预订，在“降为开发”操作结束之后，“复制控制”表中将不会留有任何记录。此时，还将删除目标表。CD表仍然存在，原因是它属于“复制源”的定义。

13. 如何在“数据仓库中心”中设置“基聚合复制”或“更改聚合复制”。

- 输入表。选择可与GROUP BY语句配合使用的输入表。在本示例中，将使用包含列SALES、REGION和DISTRICT的“输入”表。
- 复制步骤。选择“基聚合”或“更改聚合”。打开“步骤”特性。
 - Apply程序运行时，它需要执行一个类似于SELECT SUM(SALES)、REGION、DISTRICT GROUP BY REGION和DISTRICT的SELECT语句。因此，在所选择的输出列中，您将需要选择REGION、DISTRICT和一个计算列SUM(SALES)。使用“添加计算列”按钮。对于本示例，在“表达式”字段中输入SUM(SALES)。保存它。
 - Where子句。有一个“复制”要求，即，在设置仅需要一个GROUP BY子句的“复制步骤”时，您还必须提供一个DUMMY where子句，如1=1。不要将词“WHERE”包括在WHERE子句中。因此，在“基聚合”的“数据仓库中心GUI”中，只有一个WHERE子句输入字段。对于本示例，在此字段中输入：1=1 GROUP BY REGION, DISTRICT。对于“更改聚合”，有一个WHERE子句和一个GROUP BY输入字段：在WHERE子句字段中输入：1=1，而在GROUP BY字段中输入：GROUP BY REGION, DISTRICT
 - 如对其他类型的“复制”所做的那样设置其余步骤特性。按“确定”以保存该步骤并创建目标表对象。
- 打开目标表对象。现在，需要将计算列表达式的输出列重命名为有效列名，且需要对该列指定有效数据类型。保存目标表对象。
- 对“复制步骤”运行“提升为测试”。将创建目标表。它不需要主键。
- 如运行任何其他“复制步骤”那样运行该步骤。

故障诊断提示

- 要打开对“Apply程序”的跟踪，在“数据仓库特性”屏幕中将“代理程序跟踪”值设置为4。在“代理程序跟踪”为4时，“代理程序”会对Apply打开完全跟踪。

如果未在 CD 表中看到任何数据，则很可能是 Capture 程序未启动或是未更新原始源表来创建某些更改数据。

- 联机帮助中缺少“调度”笔记本的“通知”页的邮件服务器字段。
- 邮件服务器需要支持 ESMTP，以使“数据仓库中心”通知工作。在“打开“进行中的操作”窗口”帮助中，单击“数据仓库 —> 进行中的操作”而不是“数据仓库中心 —> 进行中的操作”。

访问源和目标

下表列出了“数据仓库中心”支持的源和目标的版本和发行版级别。

表 7. 受支持的 IBM 数据仓库源的版本和发行版级别

源	版本 / 发行版
IMS	5.1
DB2 通用数据库 Windows NT 版	5.2 - 7.1
DB2 通用数据库扩充企业版	5.2 - 7.1
DB2 通用数据库 OS/2 版	5.2 - 7.1
DB2 通用数据库 AS/400 版	3.7 - 4.5
DB2 通用数据库 AIX 版	5.2 - 7.1
DB2 通用数据库 Solaris 操作系统版	5.2 - 7.1
DB2 通用数据库 OS/390 版	4.1 - 7.1
DB2 DataJoiner	2.1.1
DB2 VM 版	5.3.4 或更新版本
DB2 VSE 版	7.1

源	Windows NT	AIX
Informix	7.2.2 - 8.2.1	7.2.4 - 9.2.0
Oracle	7.3.2 - 8.1.5	8.1.5
Microsoft SQL 服务器	7.0	
Microsoft Excel	97	
Microsoft Access	97	
Sybase	11.5	11.9.2

表 8. 受支持的 IBM 数据仓库目标的版本和发行版级别

目标	版本 / 发行版
DB2 通用数据库 Windows NT 版	6 - 7
DB2 通用数据库扩充企业版	6 - 7
DB2 通用数据库 OS/2 版	6 - 7
DB2 通用数据库 AS/400 版	3.1-4.5
DB2 通用数据库 AIX 版	6 - 7
DB2 通用数据库 Solaris 操作系统版	6 - 7
DB2 通用数据库 OS/390 版	4.1 - 7
DB2 DataJoiner	2.1.1

表 8. 受支持的 IBM 数据仓库目标的版本和发行版级别 (续)

目标	版本 / 发行版
DB2 DataJoiner/Oracle	8
DB2 VM 版	3.4 - 5.3.4
DB2 VSE 版	3.2, 7.1
CA/400	3.1.2

受支持的非 IBM 数据库源的补充内容

下表包含受支持的非 IBM 数据库源的补充内容:

数据库	操作系统	数据库客户机需求
Informix	AIX	Informix-Connect 和 ESQL/C 版本 9.1.4 或更新版本
Informix	Solaris 操作环境	Informix-Connect 和 ESQL/C 版本 9.1.3 或更新版本
Informix	Windows NT	Informix-Connect Windows 版平台 2.x 或 Informix-Client Software Developer's Kit Windows 版平台 2.x
Oracle 7	AIX	Oracle7 SQL*Net 和 Oracle7 SQL*Net 共享库 (由 genclntsh 脚本构建)
Oracle 7	Solaris 操作环境	Oracle7 SQL*Net 和 Oracle7 SQL*Net 共享库 (由 genclntsh 脚本构建)
Oracle 7	Windows NT	当前版本的 SQL*Net 的适当 DLL, 及 OCIW32.DLL。例如, SQL*Net 2.3 需要 ORA73.DLL、CORE35.DLL、NLSRTL32.DLL、CORE350.DLL 或 OCIW32.DLL。
Oracle 8	AIX	Oracle8 Net8 和 Oracle8 SQL*Net 共享库 (由 genclntsh8 脚本构建)
Oracle 8	Solaris 操作环境	Oracle8 Net8 和 Oracle8 SQL*Net 共享库 (由 genclntsh8 脚本构建)

数据库	操作系统	数据库客户机需求
Oracle 8	Windows NT	要访问级别为版本 8.0.3 或更新版本的远程 Oracle8 数据库服务器，安装“Oracle Net8 客户机”版本 7.3.4.x、8.0.4 或更新版本。 在 Intel 系统上，将“Oracle Net8 客户机”的适当 DLL（如 Ora804.DLL、PLS804.DLL 和 OCI.DLL）安装在路径上。
Sybase	AIX	在非 DCE 环境（ ibsyb15 ODBC 驱动程序 ）中： libct 库 在 DCE 环境（ ibsyb1115 ODBC 驱动程序 ）中： Sybase 11.1 客户机库 libct_r
Sybase	Solaris 操作环境	在非 DCE 环境（ ibsyb15 ODBC 驱动程序 ）中： libct 库 在 DCE 环境（ ibsyb1115 ODBC 驱动程序 ）中： Sybase 11.1 客户机库 libct_r
Sybase	Windows NT	Sybase Open Client-Library 10.0.4 或更新版本，以及适当的 Sybase Net-Library。

在“数据仓库中心”中人工创建数据源

如果数据源是通过使用 Relational Connect 和“Create Nickname”语句创建的，在有关在“数据仓库中心”中导入表的功能中，数据源将不可用。要将数据源用作源表或目标表，执行下列步骤：

1. 定义源 / 目标，但不导入任何表。
2. 在“数据数据仓库中心”的主窗口中，展开“数据仓库源 / 目标”树，并用右键单击“表”以获取期望的源 / 目标。
3. 单击**定义**。
4. 使用打开的记事本来定义数据源，并确保为每个数据源都定义了列。

有关详情，参见“信息中心”中的“定义数据仓库源表”或“定义数据仓库目标表”。

使用“公共数据仓库元数据交换”(CWMI)导入和导出元数据

介绍

除了对标记语言文件的现存支持之外，“数据仓库中心”现在可以在符合“公共数据仓库元模型”(CWM)标准的XML文件之间导入和导出元数据。导入和导出这些符合CWM的XML文件被称为公共数据仓库元数据交换(CWMI)。

可从下列“数据仓库中心”对象中导入和导出元数据:

- 数据仓库源
- 数据仓库目标
- 主题区, 包括进程、源、目标和步骤
- 用户定义程序

CWMI导入和导出实用程序当前不支持某些种类的元数据, 包括: 调度、数据仓库模式、快捷方式步骤、级联关系、用户和组。

“数据仓库中心”创建了包含导入和导出进程的结果的日志文件。通常, 日志文件是在 `x:\program files\sqllib\logging` 目录中创建的, (其中, `x:` 是安装有 DB2 的驱动器), 或是在指定为 `VWS_LOGGING` 环境变量的目录中创建的。日志文件是纯文本; 可以使用任何文本编辑器进行查看。

导入元数据

可以从“数据仓库中心”或者从命令行中导入元数据。

通过导入过程创建的新对象可指定给“数据仓库中心”的缺省安全组。有关详情, 参见这些“发行说明”中的“导入之后更新安全性”。

若您正在导入关于某个步骤的元数据, 则可以有多文件与该步骤相关联。关于该步骤的元数据存储在XML文件中但是有时步骤会带有存储为BLOB的关联数据。BLOB元数据与XML文件具有相同的名称, 但是它保存在对扩展名进行了编号的独立文件中。当您导入时, 所有相关步骤文件必须在同一目录中。

在步骤处于测试或生产方式时更新步骤

必须在步骤处于开发方式时, “数据仓库中心”才能更新该步骤的元数据。若步骤处于测试或生产方式, 则在导入元数据之前, 应使步骤降为开发方式:

1. 登录“数据仓库中心”。
2. 用鼠标右键单击您想降级的步骤, 并单击**方式**。
3. 单击**开发**。

该步骤现在处于开发方式。在导入元数据之后, 将该步骤更改回测试或生产方式。

从“数据仓库中心”中导入数据

可以从“数据仓库中心”中导入元数据:

1. 登录“数据仓库中心”。
2. 在左窗格中, 单击**数据仓库**。
3. 单击**选中** —> **导入元数据** —> **交换文件...**

4. 在“导入元数据”窗口中，指定包含您想导入的元数据的文件名。可以输入文件名或者进行浏览以查找该文件。
 - 若您知道文件的位置，则输入您想导入的文件的全限定路径和文件名。一定要包括 .xml 文件扩展名，以指定您想以 XML 格式来导入元数据。
 - 要进行浏览以查找该文件：
 - a. 单击省略号 (...) 按钮。
 - b. 在“文件”窗口中，将**文件类型**更改为 XML。
 - c. 转至正确的目录，并选择您想导入的文件。

注：该文件必须具有扩展名 .xml。

 - d. 单击**确定**。
5. 在**导入元数据**窗口中，单击**确定**以完成操作。“数据仓库中心”导入该文件时会显示**进度**窗口。

使用命令行导入元数据

还可以使用命令行来导入元数据。以下是导入命令语法：

```
CWMImport XML_file dwcControlDB dwcUserId dwcPW [PREFIX = DWCtbschema]
```

<i>XML_file</i>	您想导入的 XML 文件的全限定路径和文件名（包括驱动器和目录）。此参数是必需的。
<i>dwcControlDB</i>	想要将元数据导入至的数据仓库控制数据库的名称。此参数是必需的。
<i>dwcUserId</i>	用来登录数据仓库控制数据库的用户标识。此参数是必需的。
<i>dwcPW</i>	用来登录数据仓库控制数据库的用户口令。此参数是必需的。
<i>[PREFIX=DWCtbschema]</i>	“数据仓库中心”系统表的数据库模式名，有时又称为表前缀。若没有为 PREFIX= 指定任何值，则缺省模式名为 IWH。此参数是可选的。

运行导入实用程序之后更新元数据

导入之后更新安全性

为安全起见，“数据仓库中心”不会导入或导出口令。需要根据需要来更新新对象的口令。有关导入注意事项的更多详细信息，参见**数据仓库中心管理指南**的第 12 章“导出和导入数据仓库中心元数据”。

导入元数据时，所有对象都被指定给缺省安全性组。可以更改对该对象具有访问权的组：

1. 登录“数据仓库中心”。
2. 用鼠标右键单击包含您想要更改的对象的文件夹。
3. 单击**特性**，然后单击**安全性**选项卡。
4. 从所选的**数据仓库组**列表中除去组，或从**可用数据仓库组**列表中添加组。
5. 单击**确定**。

导出元数据

可以从“数据仓库中心”或者从命令行中导出元数据。

某些步骤带有存储为 BLOB 的元数据。BLOB 元数据被导出至与步骤的 XML 文件具有相同文件名，但带有编号扩展名（.1、.2 等等）的单独文件。

从“数据仓库中心”中导出数据

可以从“数据仓库中心”中导出元数据：

1. 登录“数据仓库中心”。
 2. 在左窗格中，单击**数据仓库**。
 3. 单击**已选择** —> **导出元数据** —> **交换文件**。
 4. 在“导出元数据”窗口中，指定将包含所导出的元数据的文件名。可以输入文件名或者进行浏览以查找该文件：
 - 若您知道想要使用的全限定路径和文件名，则在**文件名**输入字段中输入它。一定要包括 .xml 文件扩展名，以指定您想以 XML 格式来导出元数据。
 - 要进行浏览以查找该文件：
 - a. 单击省略号 (...) 按钮。
 - b. 在文件窗口中，将**文件类型**更改为 XML。
 - c. 转至正确的目录，并选择您想要包含导出的元数据的文件。
- 注：**您选择的任何现存文件都将被导出的元数据覆盖。
- d. 单击**确定**。
 5. 当“导出元数据”窗口显示正确的文件名时，单击**可用对象**列表中您想导出其元数据的对象。
 6. 单击 > 符号来将所选对象从**可用对象**列表移至**所选的对象**列表。重复以上步骤，直到想要导出的所有对象都列示在**所选的对象**列表中为止。
 7. 单击**确定**。

“数据仓库中心”创建输入文件，该文件包含有关选择导出的“数据仓库中心”对象的信息，然后导出有关这些对象的元数据。“数据仓库中心”导出元数据时会显示进度窗口。

导出进程完成时，您将接收到有关导出进程的资料性信息。返回码 0 指示导出成功。还可以查看日志文件以获取更详细的信息。

使用命令行导出元数据

必须首先创建输入文件，才能从命令行导出元数据。输入文件是带有扩展名 .INP 的文本文件，并且它根据想要导出的对象类型列示所有对象。在从“数据仓库中心”中导出时，会自动创建输入文件，但是从命令行导出时，您必须先创建输入文件。可以用任何文本编辑器创建输入文件。输入所有要出现在“数据仓库中心”中的对象名。确保您是在读/写目录中创建该文件的。当运行导出实用程序时，“数据仓库中心”将 XML 文件写入与输入文件所在的目录中。

以下是样本输入文件：

<PROC>
Tutorial Fact Table Process
<IR>
Tutorial file source
Tutorial target
<UDP>
New Program group

在 <PROC>（进程）一节中，列示您想导出的所有进程。在 <IR>（信息资源）一节中，列示您想导出的所有数据仓库源和目标。“数据仓库中心”会自动包括与这些源和目标相关联的表和列。在 <UDP>（用户定义程序）一节，列示想要导出的所有程序组。

要导出元数据，在 DOS 命令提示符处输入以下命令：

```
CWMExport INPcontrol_file dwcControlDB dwcUserID dwcPW [PREFIX=DWCtbschema]
```

INPcontrol_file

包含您想导出的对象的 .INP 文件的全限定路径和文件名（包括驱动器 and 目录）。此参数是必需的。

dwcControlDB

想要从其中导出的数据仓库控制数据库的名称。此参数是必需的。

dwcUserID

用来登录数据仓库控制数据库的用户标识。此参数是必需的。

dwcPW

用来登录数据仓库控制数据库的用户口令。此参数是必需的。

[PREFIX=DWCtbschema]

“数据仓库中心”系统表的数据库模式名，有时又称为表前缀。若没有为 PREFIX= 指定任何值，则缺省值为 IWH。此参数是可选的。

OS/390 Runstats 实用程序步骤

当在 DWC 中定义 OS/390 Runstats 实用程序步骤时，用户应该了解关于“步骤特性”对话框的“参数”选项卡的下列信息。对于表空间字段，用户应该以大写字母输入名称。如果表空间不在数据库 DSND04 中，则需要由包含该表空间的数据库来限定表空间名。例如，输入 SAMPLE.EMPLOYEE。帮助信息当前没有提及此字段。

OS/390 装入实用程序步骤

当在 DWC 中定义 OS/390 装入实用程序步骤时，用户应该了解关于“步骤特性”对话框的“参数”选项卡的下列信息。为了使装入实用程序工作，用户需要始终选择“高级”按钮。否则，不会生成装入语句的 INTO 子句，并且装入操作在运行时将失败。

另外，修订包 3 中包括一个修正，用来去掉括起装入数据集名称的双引号。没有此修正，装入就不会工作。

公共仓库元模型 (CWM) XML 支持

版本 7.2 CWM 工具箱使用“Java 开发工具箱”(JDK) 1.2.2 或 1.3。

现在可以导入和导出下列 CWM XML 对象:

其他进程中的快捷方式步骤

当您导出一个进程时, 如果该进程中包含与另一个进程中的步骤(快捷方式)有关系的步骤, 则这两个进程都会被导出, 并维持步骤之间的关系。

有条件的级联关系

现在可以导入和导出步骤之间的不同级联关系, 包括 CHILD、SUCCESS、FAILURE 和 UNCONDITIONAL。

将仓库源作为视图对象

当导出时, 现在可以将仓库源定义为视图对象。视图对象与表对象的处理方式相同。

SQLDataType 用于列和字段

现在可以对列和字段使用 SQLDataType。

同一个表的多个相关名

在导入或导出期间, 可以具有多个相关名, 对于同一个表, 每个相关名都有它自己的列映射。

新的 SAP 和“WebSphere 站点分析器”(WSA) 源支持

由于添加了新的源支持标记, 因此现在可以导出保存在仓库中的 SAP 和 WSA 信息。

进程造型程序

可以调整进程造型程序调色板的大小以便适合您的屏幕。调色板上的图标将自动重新定位至多列调色板。

当单击调色板图标时, 您将在调色板对象上看到一个标题。

现在可以通过选择**查看**菜单中的**显示商业名**选项来按照表和文件对象的商业名来查看表和文件对象。还可以通过**缩放比例**选项来调整百分比设置并使进程视图更小或更大。

如果在调色板中对象发生了重叠, 则可以单击该对象, 以便将它们拖到屏幕顶部。另外, 对象名现在被折成多行, 以便节省调色板的空间。

现在可以使用“删除”键来除去对象。当保存进程时同时就保存了对表所作的更改。选择行为不是自动的。如果想要从仓库源或目标以及进程中除去表、文件或视图, 如果该对象在仓库源中, 则可以单击右键, 并选择**从源中除去**操作, 如果该对象在仓库目标中, 则选择**从目标中除去**操作。

游标现在显示调色板选择状态。另外, 状态行显示游标所在的对象的名称。

模式造型程序

现在可以在模式造型程序中使表最小化和最大化。当使表最小化时，它会变成一个图标。为了获得更高的视觉精度，现在可以创建星型模式布局。


必填字段

现在，“数据仓库中心”会在必需字段上显示红色边界。红色边界提醒您输入定义“数据仓库中心”对象所需要的一些必填信息，例如，数据库名、用户标识或口令。当输入必需的信息时，红色边界就会消失。

数据仓库中心启动板增强功能

当从启动板中创建“数据仓库中心”对象时，导航树将展开，以显示新对象的位置。

将步骤信息打印至文件

现在可以将关于步骤的信息（例如，主题区、源表名和目标表名）打印至文本文件。要将步骤信息打印至文件，在进程造型程序中用鼠标右键单击步骤图标，单击**打印**  **打印至文件**，并指定想要将信息打印至的文件的名称。

Data Warehouse Center Application Integration Guide

在 Chapter 5. Metadata Templates 中，表 16 描述了“列”标记。该手册中的信息应该陈述 “*ColumnPositionNumber” 应该从 “1” 开始。该手册中不正确地将 “0” 作为起始字符。

如果在相应的 SourceDataBase.tag 中为 DatabaseType 标记指定的类型是 ISV_IR_FFLLan，则以后在第 5 章的表 42 中必须要有 TableTypeIfFile 标记。如果未指定该标记，则将检测到错误。

在 Chapter 6. Data Warehouse Center metadata 中，应将 POSNO 列对象特性的描述更改为：

An index, starting with 1, of the column
or field in the row of the table or file.

在 Chapter 8. Information Catalog Manager object types 中，可在其中找到 .TYP 文件（这些文件包括用于定义对象类型的标记语言）的目录已更改为 \SQLLIB\DGWIN\TYPES。

附加的元数据模板

在 Chapter 5. Metadata Templates 中，应该包括下列元数据模板。

表 9. 随“数据仓库中心”提供的新元数据模板

模板	描述	参见:
commit.tag		『 Commit.tag 』
Foreignkey.tag		第234页的 『 ForeignKey.tag 』
Foreignkeyadditional.tag		第236页的 『 ForeignKeyAdditional.tag 』
Primarykey.tag		第237页的 『 PrimaryKey.tag 』
Primarykeyadditional.tag		第239页的 『 PrimaryKeyAdditional.tag 』

Commit.tag

当您使用大型标记语言文件时，可以使用此模板来改进性能。可以在此处描述的任何模板组之间插入提交模板。不能在同一个组内的模板之间插入提交模板。例如，在 AgentSite.tag 和 VWPGroup.tag 之间插入提交模板是有效的，但是，在 VWPPProgramTemplate.tag 和 VWPPProgramTemplateParameter.tag 之间插入提交标记则是无效的。如果不正确地使用了提交模板，则导入时可能会报告错误。

- AgentSite.tag
- VWPGroup.tag
- VWPPProgramTemplate.tag, VWPPProgramTemplateParameter.tag
- SourceDatabase.tag
- WarehouseDatabase.tag
- Table.tag, Column.tag

- SubjectArea.tag
- Process.tag
- Step.tag, StepInputTable.tag, StepOutputTable.tag, StepVWPOutputTable.tag, StepVWPPProgramInstance.tag, VWPPProgramInstanceParameter.tag
- StepCascade.tag
- StarSchema.tag, StarSchemaInputTable.tag
- PrimaryKey.tag, PrimaryKeyAdditional.tag
- ForeignKey.tag, ForeignKeyAdditional.tag

提交模板的使用是可选的。

标记

表10提供了关于模板中的每个标记的信息。

表 10. Commit.tag 标记

标记	描述	允许值
关系参数		
*CurrentCheckPointID++	一个索引，从 0 开始，每当在标记中被替代一次，该值就会增加。	一个数值。
	此标记是必需的。	

示例值

表11提供了每个标记的示例值，用来说明可以为每个标记提供的元数据的种类。

表 11. Commit.tag 标记的示例值

标记	示例值
*CurrentCheckPointID++	1

ForeignKey.tag

使用此模板来定义对表的外键约束。ForeignKey.tag 模板定义要对其定义约束的表和列的关系。此模板还定义引用的主键的表和列之间的关系。在使用 ForeignKey.tag 模板之前，必须定义主键约束（使用 PrimaryKey.tag 模板），并定义想要对其定义外键约束的表和列（使用 Table.tag 和 Column.tag 模板）。

标记

表12提供了关于模板中的每个标记的信息。

表 12. ForeignKey.tag 标记

标记	描述	允许值
实体参数		
*ConstraintName	约束的名称。	文本字符串，其长度最多为 80 个字节。
	在表或字段中该名称必须是唯一的。	
	此标记是必需的。	

表 12. ForeignKey.tag 标记 (续)

标记	描述	允许值
*ForeignColumnName	对其定义外键约束的列的名称。	文本字符串，其长度最多为 254 个字节。
*ForeignKeyID	用来唯一地标识外键的键。该键必须在标记语言文件中的其他所有键中是唯一的。 提示： 在增加键值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
*MapID	一个随机数，它在交换文件的所有其他键中是唯一的。 提示： 在增加此标记的值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
*PrimaryColumnName	所引用的列的列名。	文本字符串，其长度最多为 80 个字节。
*ReferencedPrimaryKeyID	用来唯一地标识主键的键。该键必须在标记语言文件中的其他所有键中是唯一的。 提示： 在增加键值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
关系参数		
*DatabaseName	仓库源或仓库目标的商业名。 此标记是必需的。	文本字符串，其长度最多为 40 个字节。
*ForeignTablePhysicalName	物理表的数据库定义名称，该物理表中包含引用了其他表中的键的外键。	文本字符串，其长度最多为 254 个字节。
*PrimaryTablePhysicalName	物理表的数据库定义名称，该物理表中包含被外键引用的键。	文本字符串，其长度最多为 80 个字节。
*PrimaryTableOwner	包含了被引用的主键列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 128 个字节。
*ForeignTableOwner	包含了外键约束列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 128 个字节。

示例值

第236页的表13提供了每个标记的示例值，用来说明可以为每个标记提供的元数据的种类。

表 13. ForeignKey.tag 标记的示例值

标记	示例值
*ConstraintName	部门
*DatabaseName	金融仓库
*ForeignColumnName	国家代码
*ForeignKeyID	07011
*ForeignTablePhysicalName	地理
*MapID	02568
*PrimaryColumnName	状态码
*Referenced PrimaryKeyID	名称
*PrimaryTablePhysicalName	城市
*PrimaryTableOwner	DB2ADMIN
*ForeignTableOwner	IWH

ForeignKeyAdditional.tag

使用此模板来定义复合外键。在使用 ForeignKeyAdditional.tag 模板之前，必须对第一列定义约束（使用 ForeignKey.tag 模板）。然后，可以使用此模板来为想要对其添加列的每一列来添加列。

标记

表14提供了关于模板中的每个标记的信息。

表 14. ForeignKeyAdditional.tag 标记

标记	描述	允许值
实体参数		
*ForeignKeyColumnName	对其定义外键约束的列的名称。	文本字符串，其长度最多为 80 个字节。
*ForeignKeyID	用来唯一地标识外键的键。该键必须在标记语言文件中的其他所有键中是唯一的。 提示： 在增加键值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
*MapID	一个随机数，它在交换文件的所有其他键中是唯一的。 提示： 在增加此标记的值之前，必须完成 ForeignKeyAdditional.tag 模板的处理。 此标记是必需的。	一个数值。
*MapSeqNo	一个数字，表示作为复合键的一部分添加至外键约束的每个附加列。	从 2 开始的唯一、递增、连续的数目。
*PrimaryColumnName	所引用的列的列名。	文本字符串，其长度最多为 80 个字节。
关系参数		

表 14. *ForeignKeyAdditional.tag* 标记 (续)

标记	描述	允许值
<i>*DatabaseName</i>	仓库源或仓库目标的商业名。 此标记是必需的。	文本字符串，其长度最多为 40 个字节。
<i>*ForeignTablePhysicalName</i>	物理表的数据库定义名称，该物理表中包含引用了其他表中的键的外键。	文本字符串，其长度最多为 80 个字节。
<i>*PrimaryTablePhysicalName</i>	物理表的数据库定义名称，该物理表中包含被外键引用的键。	文本字符串，其长度最多为 80 个字节。
<i>*PrimaryTableOwner</i>	包含了被引用的主键列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 128 个字节。
<i>*ForeignTableOwner</i>	包含了外键约束列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 128 个字节。

示例值

表15提供了每个标记的示例值，用来说明可以为每个标记提供的元数据的种类。

表 15. *ForeignKeyAdditional.tag* 标记的示例值

标记	示例值
<i>*DatabaseName</i>	金融仓库
<i>*ForeignColumnName</i>	国家代码
<i>*ForeignKeyID</i>	07011
<i>*ForeignTablePhysicalName</i>	地理
<i>*MapID</i>	22578
<i>*MapSeqNo</i>	2
<i>*PrimaryColumnName</i>	状态码
<i>*PrimaryTablePhysicalName</i>	城市
<i>*PrimaryTableOwner</i>	DB2ADMIN
<i>*ForeignTableOwner</i>	IWH

PrimaryKey.tag

使用此模板来定义对表的主键约束。该模板还定义了要对其定义约束的表和列的关系。在使用 *PrimaryKey.tag* 模板之前，必须定义想要对其定义主键约束的表和列（使用 *Table.tag* 和 *Column.tag* 模板）。

标记

第238页的表16提供了关于模板中的每个标记的信息。

表 16. PrimaryKey.tag 标记

标记	描述	允许值
实体参数		
*ColumnName	列或字段的名称。 在表或字段中该名称必须是唯一的。 此标记是必需的。	文本字符串，其长度最多为 80 个字节。
*MapID	一个随机数，它在交换文件的所有其他键中是唯一的。 提示： 在增加此标记的值之前，必须完成 PrimaryKey.tag 模板的处理。 此标记是必需的。	一个数值。
*PrimaryKeyID	用来唯一地标识主键的键。该键必须在标记语言文件中的其他所有键中是唯一的。 提示： 在增加键值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
关系参数		
*DatabaseName	仓库源或仓库目标的商业名。 此标记是必需的。	文本字符串，其长度最多为 40 个字节。
*TableOwner	包含列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 128 个字节。
*TablePhysicalName	包含了为数据库管理器或文件系统定义的列的表或文件的物理名。 此标记是必需的。	文本字符串，其长度最多为 80 个字节。

示例值

表17提供了每个标记的示例值，用来说明可以为每个标记提供的元数据的种类。

表 17. PrimaryKey.tag 标记的示例值

标记	示例值
*ColumnName	国家代码
*DatabaseName	金融仓库
*MapID	54627
*PrimaryKeyID	74622
*TableOwner	DB2ADMIN
*TablePhysicalName	地理

PrimaryKeyAdditional.tag

使用此模板来定义复合主键。在使用 PrimaryKeyAdditional.tag 模板之前，必须使用 PrimaryKey.tag 模板来对第一列定义约束。然后，可以使用此模板来添加任何附加列。该模板还使附加主键与使用 PrimaryKey.tag 定义的第一个主键相关联。

标记

表18提供了关于模板中的每个标记的信息。

表 18. PrimaryKeyAdditional.tag 标记

标记	描述	允许值
实体参数		
*ColumnName	列或字段的名称。 在表或字段中该名称必须是唯一的。 此标记是必需的。	文本字符串，其长度最多为 80 个字节。
*FirstPrimaryKeyID	用来唯一地标识主键的键。该键必须在标记语言文件中的其他所有键中是唯一的。 提示： 在增加键值之前，必须完成 ForeignKey.tag 模板的处理。 此标记是必需的。	一个数值。
*MapID	一个随机数，它在交换文件的所有其他键中是唯一的。 提示： 在增加此标记的值之前，必须完成 PrimaryKeyAdditional.tag 模板的处理。 此标记是必需的。	一个数值。
*MapSeqNo	一个数字，表示作为复合键的一部分添加至主键约束的每个附加列。	从 2 开始的唯一、递增、连续的数目。
关系参数		
*DatabaseName	仓库源或仓库目标的商业名。 此标记是必需的。	文本字符串，其长度最多为 40 个字节。
*TableOwner	包含列的表的所有者、高级限定符、集合或模式。 此标记是必需的。	文本字符串，其长度最多为 15 个字节。
*TablePhysicalName	包含了为数据库管理器或文件系统定义的列的表或文件的物理名。 此标记是必需的。	文本字符串，其长度最多为 80 个字节。

示例值

表19提供了每个标记的示例值，用来说明可以为每个标记提供的元数据的种类。

表 19. *PrimaryKeyAdditional.tag* 标记的示例值

标记	示例值
<i>*ColumnName</i>	国家代码
<i>*DatabaseName</i>	金融仓库
<i>*MapID</i>	99542
<i>*MapSeqNo</i>	2
<i>*FirstPrimaryKeyID</i>	07801
<i>*TableOwner</i>	DB2ADMIN
<i>*TablePhysicalName</i>	地理

数据仓库中心联机帮助

对复制定义表或视图

必须在使用“DB2 控制中心”对复制定义表或视图后，该表或视图才可用作“数据仓库中心”中的复制源。

将 Essbase VWP 与 AS/400 代理程序一起运行

在与 AS/400 代理程序一起运行 Essbase VWP 时，ARBORLIB 和 ARBORPATH 需要设置为 *sys 环境变量。要设置这些变量，用户标识必须具有 *jobctl 权限。这些环境变量需要指向安装有 Essbase 的库。

使用“发布数据仓库中心元数据”窗口和“相关特性”窗口

在任务帮助的步骤 10 中，有一个示例指出如果指定了限制值 1（限制树中对象的级别）并发布了一个进程，则仅发布和显示该进程中的一个步骤。此示例并非在所有情况下都是正确的。在步骤 8 的第二个发布项目上，第一个语句不正确。它应为“单击列级别以生成信息目录源列和目标列之间的转换对象。”

外键

联机帮助中任何出现“外键”的地方都应为“数据仓库外键”。

复制笔记本

联机帮助中任何出现“定义复制笔记本”的地方都应为“复制步骤笔记本”。

导入标记语言

导入标记语言联机帮助：在显示常见导入错误的发布列表中，列表中的一项为“正在导入未正确导出的标记语言文件”。此项不适用于常见输入错误的列表。

用于添加数据的链接

在联机帮助的“添加数据”主题中，指向“将源表添加至进程”和“将目标表添加至进程”主题的链接被破坏。您可在帮助索引中找到这些主题。

导入表

帮助主题“将源表和视图导入数据仓库源”和“将目标表导入数据仓库目标”中包含有关通配符的信息不正确。该句为：

例如，XYZ* 将以这些字符开头的模式返回表和视图。

应为：

例如，XYZ% 将以这些字符开头的模式返回表和视图。

对 RUNSTATS 和 REORGANIZE TABLE 联机帮助的校正

这些实用程序的联机帮助指出您想要对其运行统计或是要对其进行重组的表必须同时作为源和目标链接。但是，因为步骤会写至源，所以您只需要从源链接至步骤。

“通知”页（“数据仓库特性”笔记本和“调度”笔记本）

在“数据仓库特性”笔记本的“通知”页上，语句：

“发送方”输入字段被初始化为 <当前用户的登录标识>。

应更改为：

“发送方”输入字段被初始化为 <当前登录用户电子邮件地址>。

在“调度”笔记本的“通知”页上，发送方将被初始化为在“数据仓库特性”笔记本中设置的内容。如果未设置任何内容，则它被初始化为当前登录用户电子邮件地址。如果登录用户没有相关联的电子邮件地址，则发送方将设置为登录用户标识。

“代理点”笔记本中的“代理模块”字段

“代理点”笔记本中的“代理模块”字段提供了数据仓库代理程序守护程序生成数据仓库代理程序时运行的程序的名称。不要更改该字段中名称，除非 IBM 指示您这样做。

DB2 OLAP Starter Kit

在某些操作系统平台上，IBM DB2 OLAP Starter Kit 7.2 添加了对 Oracle、MS-SQL、Sybase 和 Informix 关系数据库管理系统 (RDBMS) 的支持。版本 7.2 中包含用于所有受支持的 RDBMS (包括 DB2) 的脚本和工具。但是有一些限制；有关详情，参见第250页的『已知问题和限制』。

“DB2 通用数据库版本 7.2”的 DB2 OLAP Starter Kit 的服务级别相当于 Hyperion Essbase 6.1 的补丁 2 加上 Hyperion Integration Server 2.0 的补丁 2。

OLAP 服务器 Web 站点

要获取关于 DB2 OLAP Starter Kit 的最新安装和用法提示，咨询“DB2 OLAP 服务器”Web 站点的 Library 页面：

<http://www.ibm.com/software/data/db2/db2olap/library.html>

受支持的操作系统服务级别

版本 7.2 的 OLAP Starter Kit 的服务器组件支持下列操作系统和服务级别：

- 具有 SP 5 的 Windows NT 4.0 服务器和 Windows 2000
- AIX 版本 4.3.3 或更高版本
- Solaris 操作系统版本 2.6、7 和 8 (Sun OS 5.6、5.7 或 5.8)

在 Windows 95、Windows 98、Windows NT 4.0 SP5 和 Windows 2000 上运行的客户机组件。

在 UNIX 上完成 DB2 OLAP Starter Kit 设置

DB2 OLAP Starter Kit 安装遵循 UNIX 版的“DB2 通用数据库”安装的基本过程。安装程序会将产品文件放置在系统目录中：（对于 AIX: /usr/lpp/db2_07_01；对于 Solaris: /opt/IBMDB2/V7.1）。

然后，在创建实例阶段，在实例用户的主目录的 sqllib 下创建两个了 DB2 OLAP 目录 (essbase 和 is)。一台机器一次仅可运行一个 OLAP 服务器实例。要完成安装，用户必须人工设置 is/bin 目录，以使它不是指向系统中 is/bin 目录的链接。它应链接至实例主目录内的可写目录。

要完成 Solaris 版的安装，使用实例标识进行登录，切换至 sqllib/is 目录，然后输入以下内容：

```
rm bin
mkdir bin
cd bin
ln -s /opt/IBMDB2/V7.1/is/bin/ismesg.mdb ismesg.mdb
ln -s /opt/IBMDB2/V7.1/is/bin/olapicmd olapicmd
ln -s /opt/IBMDB2/V7.1/is/bin/olapisvr olapisvr
ln -s /opt/IBMDB2/V7.1/is/bin/essbase.mdb essbase.mdb
ln -s /opt/IBMDB2/V7.1/is/bin/libolapams.so libolapams.so
```

为 OLAP Starter Kit 配置 ODBC

要使“OLAP 集成服务器”与关系数据源和“OLAP 元数据目录”之间进行“开放式数据库连接”(ODBC)连接, IBM DB2 OLAP Starter Kit 7.2 需要具有 ODBC.ini 文件。

- 在 Windows 系统上, 此文件位于 HKEY_LOCAL_MACHINE/SOFTWARE/ODBC 下面的“注册表”中。使用“ODBC 数据源管理器”来存储关于如何连接至关系数据源的信息。
- 在 UNIX 系统上, 安装程序将创建模型 odbc.ini 文件。要存储关于如何连接至关系数据源的信息, 使用首选编辑器来编辑该文件。

ODBC.ini 文件在 ODBC 软件包中, 并且包括在 Microsoft Office 软件中。有关安装 ODBC 驱动程序或“ODBC 管理器”的应用程序的详情, 请访问以下 Web 站点:
<http://support.microsoft.com/support/kb/articles/Q113/1/08.asp>。

对于 AIX 机器上的 Oracle 用户: 要为 Oracle 配置 ODBC, 必须更新 ODBC.ini 文件以指向 MERANT 3.6 驱动程序。

在版本 7.2 中, OLAP Starter Kit 负责管理与关系数据源和与“OLAP 元数据目录”的 ODBC 连接。为了应用这些 ODBC 连接, OLAP Starter Kit 在 Windows NT 4.0、Windows 2000、AIX 和 Solaris 上使用 ODBC 驱动程序。

- “DB2 通用数据库版本 6”数据库客户机: 在 Windows NT 4.0 SP5 或 Windows 2000、AIX 4.3.3 和 Solaris 操作系统 2.6、7 或 8 (Sun OS 5.6、5.7 或 5.8) 上为 DB2 版本 6 ODBC 驱动程序。
- “DB2 通用数据库 7.1”数据库客户机: 在 Windows NT 4.0 SP5 或 Windows 2000、AIX 4.3.3 和 Solaris 操作系统 2.6、7 或 8 (Sun OS 5.6、5.7 或 5.8) 上为 DB2 版本 7 ODBC 驱动程序。
- Oracle 8.04 和 8i SQL*Net 8.0 数据库客户机: 在 Windows NT 4.0 SP5 或 Windows 2000、AIX 4.3.3、Solaris 操作系统 2.6、7 或 8 (Sun OS 5.6、5.7 或 5.8) 上为 MERANT 3.6 ODBC 驱动程序。
- MS SQL Server 6.5.201 (不需要数据库客户机): 在 Windows NT 4.0 SP5 或 Windows 2000 上为 MS SQL Server 6.5 ODBC 驱动程序。
- MS SQL Server 7.0 (不需要数据库客户机): 在 Windows NT 4.0 SP5 或 Windows 2000 上为 MS SQL Server 7.0 ODBC 驱动程序。

在 UNIX 系统上配置数据源

在 AIX 和 Solaris 上, 必须人工设置 ODBC 的环境变量并编辑 odbc.ini 文件以配置关系数据源和“OLAP 元数据目录”。如果添加了新驱动程序或数据源, 或者更改了驱动程序或数据源, 则一定要编辑 odbc.ini 文件。

配置 ODBC 环境变量

在 UNIX 系统上, 必须设置环境变量以便允许访问 ODBC 核心组件。在 Starter Kit 主目录中提供了用来设置必需的变量的 is.sh 和 is.csh shell 脚本。必须在运行这些脚本的其中之一以后才能使用 ODBC 来连接至数据源。对于您用来运行 OLAP Starter Kit 的用户名, 应该在登录脚本中包括这些脚本。

编辑 odbc.ini 文件

要在 odbc.ini 文件中配置数据源, 必须为 ODBC 数据源添加名称和描述, 并在为数据源名创建的单独一节中提供 ODBC 驱动程序路径、文件名和其他驱动程序设置。安装程序将在 ISHOME 目录中安装样本 odbc.ini 文件。该文件包含类属 ODBC 连接

和受支持的 ODBC 驱动程序的配置信息。将该文件作为映射用于关系数据源和 “OLAP 元数据目录” 的 ODBC 驱动程序的起始点。

如果您使用不同于 `odbc.ini` 文件的文件，则必须将 `ODBCINI` 环境变量设置为您使用的文件的名称。

将数据源添加至 `odbc.ini` 文件

1. 在运行 OLAP Starter Kit 服务器的系统上，通过使用诸如 `vi` 的文本编辑器来打开 `odbc.ini` 文件。
2. 查找以『ODBC 数据源』开头的那一节，并添加具有数据源名和描述的新的一行，例如：`mydata=data source for analysis`。为了尽量避免混淆，数据源名应该与 RDBMS 中的数据库的名称相匹配。
3. 通过用方括号将新数据源的名称括起来创建新的一行（例如：`[mydata]`），以将新的一节添加到文件中。
4. 在数据源名后面的各行上，添加此数据源所需要的 ODBC 驱动程序的全路径和文件名，并添加其他任何必需的 ODBC 驱动程序信息。将下列各节中所显示的示例作为映射至 RDBMS 上的数据源的指南。确保 ODBC 驱动程序文件确实存在于您为 `Driver=` 设置指定的位置中。
5. 当编辑完 `odbc.ini` 之后，保存该文件并退出文本编辑器。

DB2 的 ODBC 设置的示例

以下示例显示了可以使用 IBM DB2 本机 ODBC 驱动程序来编辑 `odbc.ini`，以便与 AIX 上的 “DB2 通用数据库版本 6.1” 上的关系数据源 `db2data` 进行连接。在 `vi` 编辑器中，使用 `$ODBCINI` 命令来编辑 `odbc.ini` 并插入下列语句：

```
[ODBC Data Sources]
db2data=DB2 Source Data on AIX
...
[db2data]
Driver=/home/db2inst1/sqllib/lib/db2.o
Description=DB2 Data Source - AIX, native
```

Oracle 的 ODBC 设置的示例

以下示例显示了可以使用 MERANT 版本 3.6 ODBC 驱动程序来编辑 `odbc.ini`，以便与 Oracle 版本 8（在 Solaris 上）上的关系数据源 `oradata` 进行连接。在此示例中，`LogonID` 和 `Password` 被 OLAP Starter Kit 用户名和口令中使用的实际值覆盖。

```
[ODBC Data Sources]
oradata=Oracle8 Source Data on Solaris
...
[myoracle] Driver=
/export/home/users/dkendric/is200/odbc/lib/ARor815.so
Description=my oracle source
```

在 UNIX 系统上配置 “OLAP 元数据目录”

在 AIX 和 Solaris 上配置 “OLAP 元数据目录” 类似于配置数据源。对于 “OLAP 元数据目录” 数据库，将数据源名和一节添加到 `odbc.ini` 文件中，正如第244页的『编辑 `odbc.ini` 文件』中所述。不需要执行其他更改。

必须在受支持的 RDBMS 中创建 “OLAP 元数据目录” 数据库才能将它作为 ODBC 数据源来配置。

以下示例显示了如何使用本机 ODBC 驱动程序来编辑 odbc.ini，以便与 DB2 版本 6.1（在 Solaris 上）上的“OLAP 元数据目录”TBC_MD 进行连接：

```
[ODBC Data Sources]
ocd6a5a=db2 v6
...
[ocd6a5a]
Driver=/home/db2inst1/sqllib/lib/db2.0
Description=db2
```

在 Windows 系统上配置数据源

要在 Windows NT 或 Windows 2000 系统上配置关系数据源，必须启动“ODBC 管理器”，然后创建与您将用于创建 OLAP 模型和元轮廓的数据源的连接。从 Windows “控制面板”中运行“ODBC 管理器”实用程序。以下示例将创建一个 DB2 数据源；其他 RDBMS 的对话框将是不同的。

要利用“ODBC 管理器”来配置关系数据源，完成下列步骤：

1. 在 Windows 桌面上，打开“控制面板”窗口。
2. 在“控制面板”窗口中，执行下列其中一个步骤：
 - a. 在 Windows NT 上，双击 **ODBC** 图标以打开“ODBC 数据源管理器”对话框。
 - b. 在 Windows 2000 上，双击**管理工具**图标，然后双击**数据源 (ODBC)** 图标以打开“ODBC 数据源管理器”对话框。
3. 在“ODBC 数据源管理器”对话框中，单击**系统 DSN** 选项卡。
4. 单击**添加**以便打开“创建新的数据源”对话框。
5. 在“ODBC 管理器”的“创建新的数据源”对话框的驱动程序列表框中，选择适当的驱动程序（例如 IBM DB2 ODBC 驱动程序），并单击**完成**以便打开“ODBC IBM DB2 驱动程序 - 添加”对话框。
6. 在“ODBC IBM DB2 驱动程序 - 添加”对话框中，在**数据库别名**下拉列表中，选择关系源数据的数据库的名称（例如，样本应用程序中的 TBC）。
7. 在**描述**文本框中输入可选描述，指示如何使用此驱动程序，然后单击**添加**。例如，输入下列单词来描述“我的业务”数据库：

客户、产品、市场

可以输入下列单词来描述样本应用程序数据库：

样本关系数据库

当您从“OLAP Starter Kit 桌面”中进行连接时，这些描述可以帮助您标识可供选择的数据源。

8. 单击“确定”，以便返回至“ODBC 数据源管理器”对话框。输入的数据源名以及映射至该数据源的驱动程序显示在**系统 DSN** 选项卡上的**系统数据源**列表框中。

要编辑数据源的配置信息：

1. 选择数据源名，并单击**配置**以打开“ODBC IBM DB2 - 添加”对话框。
2. 校正您想更改的任何信息。
3. 单击两次**确定**退出。

在 Windows 系统上配置“OLAP 元数据目录”

要在 Windows NT 或 Windows 2000 上配置“OLAP 元数据目录”，启动“ODBC 管理器”，然后创建与包含“OLAP 元数据目录”数据库的数据源的连接。

以下示例将创建一个 DB2 数据源；其他 RDBMS 的对话框将是不同的。要为“OLAP 元数据目录”创建数据源，应完成下列步骤：

1. 在桌面上，打开“控制面板”窗口。
2. 在“控制面板”窗口中，执行下列其中一个步骤：
 - a. 在 Windows NT 上，双击 **ODBC** 图标以打开“ODBC 数据源管理器”对话框。
 - b. 在 Windows 2000 上，双击**管理工具**图标，然后双击**数据源 (ODBC)** 图标以打开“ODBC 数据源管理器”对话框。
3. 在“ODBC 数据源管理器”对话框中，单击**系统 DSN** 选项卡。
4. 单击**添加**以便打开“创建新的数据源”对话框。
5. 在“ODBC 管理器”的“创建新的数据源”对话框的驱动程序列表框中，选择适当的驱动程序（例如 IBM DB2 ODBC 驱动程序），并单击**完成**以便打开“ODBC IBM DB2 驱动程序 - 添加”对话框。
6. 在“ODBC IBM DB2 驱动程序 - 添加”对话框中，在**数据库别名**下拉列表中，选择“OLAP 元数据目录”的数据库的名称（例如，样本应用程序中的 TBC_MD）。所选的数据库的名称将自动显示在**数据源名**文本框中。
7. 如果想要更改数据源的名称，选择**数据源名**文本框中所显示的名称，输入新名称以指示您如何使用此驱动程序，然后单击**添加**。例如，可以输入以下名称来指示您正在使用驱动程序来与第一个“OLAP 元数据目录”进行连接：

OLAP Catalog first

应输入以下名称来指示您正在与样本应用程序“OLAP 元数据目录”数据库进行连接：

TBC_MD

8. 在**描述**文本框中，输入描述用来指示如何使用此驱动程序。例如，可以输入下列单词来描述“OLAP 元数据目录”：

My first models and metaoutlines

可以输入下列单词来描述样本应用程序“OLAP 元数据目录”数据库：

样本模型和元轮廓

当您从“OLAP Starter Kit 桌面”来与“OLAP 元数据目录”进行连接时，这些描述可以帮助您标识您想选择的目录。

9. 单击“确定”，以便返回至“ODBC 数据源管理器”对话框。输入的数据源名以及映射至该数据源的驱动程序显示在**系统 DSN** 选项卡上的**系统数据源**列表框中。

要编辑数据源的配置信息：

1. 选择数据源名，并单击**配置**以打开“ODBC IBM DB2 - 添加”对话框。
2. 校正您想更改的任何信息。
3. 单击两次**确定**退出。

在配置数据源之后

在配置关系数据源和“OLAP 元数据目录”之后，可以从 OLAP Starter Kit 中与它们进行连接。然后，可以创建、修改和保存 OLAP 模型和元轮廓。

在调用 SQL Server 数据库期间，SQL Server ODBC 驱动程序可能会超时。当数据库不忙时再试。增大驱动程序超时期间可能会避免此问题。有关详情，参见关于您正在使用的驱动程序的 ODBC 文档。

有关 ODBC 连接问题和解决方案的详情，参见 *OLAP Integration Server System Administrator's Guide*。

从“OLAP Starter Kit 桌面”进行登录

要使用“OLAP Starter Kit 桌面”来创建 OLAP 模型和元轮廓，必须将客户机软件连接至两个服务器组件：“DB2 OLAP 集成服务器”和“DB2 OLAP 服务器”。登录对话框提示您输入要使“桌面”连接至这两个服务器所必需的信息。在对话框的左边，输入有关“DB2 OLAP 集成服务器”的信息。在对话框右边，输入有关“DB2 OLAP 服务器”的信息。

要连接至“DB2 OLAP 集成服务器”：

- 服务器：输入“集成服务器”的主机名或 IP 地址。如果已经在桌面所在的同一工作站上安装了“集成服务器”，则典型值为“localhost”或“127.0.0.1”。
- OLAP 元数据目录：在连接至“OLAP 集成服务器”时，还必须指定“元数据目录”。
“OLAP 集成服务器”会将有关您创建的 OLAP 模型和元轮廓的信息存储在称为“元数据目录”的关系数据库中。必须对 ODBC 注册此关系数据库。目录数据库包含一组特殊的关系表，“OLAP 集成服务器”可识别这些表。在登录对话框上，您可指定“集成服务器”，然后展开“OLAP 元数据目录”字段的下拉菜单，以查看“OLAP 集成服务器”识别的 ODBC 数据源名称列表。选择包含元数据目录表的 ODBC 数据库。
- 用户名和口令：“OLAP 集成服务器”将使用在此屏面上指定的用户名和口令连接至“元数据目录”。这是一个存在于服务器（并非客户机，除非服务器和客户机运行于同一机器上）之上的登录帐户。用户名必须是创建“OLAP 元数据目录”的用户。否则，“OLAP 集成服务器”在目录数据库中找不到关系表，原因是表模式名不同。

“DB2 OLAP 服务器”信息是可选的，所以“登录”对话框右边的输入字段可能会留为空白。但是，“桌面”和“管理器”中的某些操作可能需要您连接至“DB2 OLAP 服务器”。如果将这些字段留为空白，则在“集成服务器”需要连接至“DB2 OLAP 服务器”以完成您请求的操作时，“桌面”将再次显示“登录”对话框。建议总是在“登录”对话框上填充“DB2 OLAP 服务器”字段。

要连接至“DB2 OLAP 服务器”：

- 服务器：输入“DB2 OLAP 服务器”的主机名或 IP 地址。如果正在运行 OLAP Starter Kit，则“OLAP 服务器”与“集成服务器”的主机名或 IP 地址相同。如果“集成服务器”与“OLAP 服务器”安装在不同主机上，则输入在“OLAP 集成服务器”上定义的主机名或 IP 地址。
- 用户名和口令：“OLAP 集成服务器”将使用在此屏面上指定的用户名和口令连接至“DB2 OLAP 服务器”。必须已经对“DB2 OLAP 服务器”定义了此用户名和口令。“OLAP 服务器”管理它自己的用户名和口令，与主机操作系统分开进行。

Starter Kit 登录示例

以下示例假定您创建了“OLAP 样本”，且选择了 *db2admin* 作为 OLAP Starter Kit 安装期间的管理员用户标识并将 *password* 作为管理员口令。

- 对于“OLAP 集成服务器”：“服务器”为 *localhost*，“OLAP 元数据目录”为 *TBC_MD*，“用户名”为 *db2admin*，“口令”为 *password*
- 对于“DB2 OLAP 服务器”：“服务器”为 *localhost*，“用户名”为 *db2admin*

人工创建和配置 OLAP Starter Kit 的样本数据库

这些样本数据库是在安装 OLAP Starter Kit 时自动创建的。下列指导说明如何人工设置“目录”和“样本”数据库（如果必要的话）。

1. 在 Windows 中，通过单击开始 → 程序 → **DB2 Windows NT 版** → 命令窗口来打开“命令中心”窗口。
2. 创建产品目录数据库：
 - a. 输入 `db2 create db OLAP_CAT`
 - b. 输入 `db2 connect to OLAP_CAT`
3. 在数据库中创建表：
 - a. 定位至 `\SQLLIB\IS\ocscript\ocdb2.sql`
 - b. 输入 `db2 -tf ocdb2.sql`
4. 创建样本源数据库：
 - a. 输入 `db2 connect reset`
 - b. 输入 `db2 create db TBC`
 - c. 输入 `db2 connect to TBC`
5. 在数据库中创建表：
 - a. 定位至 `\SQLLIB\IS\samples\`
 - b. 将 `tbcdb2.sql` 复制至 `\SQLLIB\samples\db2sampl\tbc`
 - c. 将 `lddb2.sql` 复制至 `\SQLLIB\samples\db2sampl\tbc`
 - d. 定位至 `\SQLLIB\samples\db2sampl\tbc`
 - e. 输入 `db2 -tf tbcdb2.sql`
 - f. 输入 `db2 -vf lddb2.sql` 以将样本源数据装入到表中。
6. 创建样本目录数据库：
 - a. 输入 `db2 connect reset`
 - b. 输入 `db2 create db TBC_MD`
 - c. 输入 `db2 connect to TBC_MD`
7. 在数据库中创建表：
 - a. 定位至 `\SQLLIB\IS\samples\tbc_md`
 - b. 将 `ocdb2.sql` 复制至 `\SQLLIB\samples\db2sampl\tbcmd`
 - c. 将 `lcdb2.sql` 复制至 `\SQLLIB\samples\db2sampl\tbcmd`
 - d. 定位至 `\SQLLIB\samples\db2sampl\tbcmd`
 - e. 输入 `db2 -tf ocdb2.sql`
 - f. 输入 `db2 -vf lcdb2.sql` 以将样本元数据装入到表中。

8. 配置 TBC_MD、TBC 和 OLAP_CAT 的 ODBC:
 - a. 通过单击开始 → 设置 → 控制面板来打开 NT 控制面板
 - b. 从列表选择 ODBC (或 ODBC 数据源)。
 - c. 选择“系统 DSM”选项卡。
 - d. 单击添加。创建新数据源窗口打开。
 - e. 从列表选择 IBM DB2 ODBC DRIVER。
 - f. 单击完成。“ODBC IBM D2 驱动程序 - 添加”窗口打开。
 - g. 在数据源名字段中输入数据源 (OLAP_CAT) 的名称。
 - h. 在数据库别名字段中输入别名, 或单击向下箭头并从列表选择 OLAP_CAT。
 - i. 单击“确定”。
 - j. 对 TBC_MD 和 TBC 数据库重复这些步骤。

将应用程序迁移至 OLAP Starter Kit 版本 7.2

安装程序不会重新安装 OLAP Starter Kit 样本应用程序、数据库和数据文件。现存应用程序和数据库不会受到任何影响。然而, 最好是在安装之前备份应用程序和数据库。

当打开应用程序时, 它们会被自动迁移到版本 7.2。

已知问题和限制

本节列示了 DB2 OLAP Starter Kit 的已知限制。

Informix RDBMS 与 Windows 平台上的 Merant 驱动程序兼容

为了使 Windows 平台的 Merant 驱动程序使用 Informix RDBMS, 必须将下列两个条目添加至 PATH 语句:

- C:\Informix
- C:\Informix\bin

这两个条目都必须在 PATH 的开头。

OLAP 模型和相关联的元轮廓中的维之间可能存在的不一致性

在某些情况下, 可在元轮廓中创建一个维, 它在 OLAP 模型中没有相应维。在下列方案中就可能发生这种情况:

1. 创建新的 OLAP 模型并保存它。
2. 根据模型来创建元轮廓, 但是不保存元轮廓。
3. 返回至 OLAP 模型, 并删除其中一个元轮廓维所基于的维。
4. 返回至元轮廓, 保存它, 关闭它, 然后重新打开它。元轮廓将包含一个维, 该维在 OLAP 模型中没有相对应的维。

OLAP Starter Kit 不能区分以此方式创建的维与元轮廓中用户定义的维之间的不一致。因此, 不一致的维将显示在元轮廓中, 但是, 元轮廓认为它是用户定义的维, 因为 OLAP 模型中不存在相对应的维。

在 Windows 2000 平台上, TMP 的环境变量设置导致成员和数据装入失败

由于 Windows 2000 与 Windows NT 之间的缺省系统和 TMP 的环境变量设置的差异, 当 OLAP Starter Kit 在 Windows 2000 平台上运行时, 成员和数

据装入将失败。所产生的错误消息告诉用户未能创建临时文件。在 Windows 2000 上，可以通过执行下列步骤来解除此限制：

1. 创建一个称为 C:\TEMP 的目录
2. 将系统和用户的环境变量 TMP 都设置为 TMP=C:\TEMP

ODBC 的安装不会替换现存的 Merant 驱动程序

现存的 3.6 Merant ODBC 驱动程序将不会被此安装更新。如果您从 OLAP Starter Kit 版本 7.1 修订包 2 或更早的版本进行升级，则应该继续使用先前安装的 ODBC 驱动程序

在 UNIX 平台上使用 Merant Informix ODBC 驱动程序

要在 UNIX 平台上使用 Merant Informix ODBC 驱动程序，必须执行下列其中一个操作：

- 在启动 Starter Kit 之前，将 LANG 环境变量设置为 "en_US"。例如，对于 korn shell，输入：

```
export LANG='en_US'
```

每次启动 OLAP Starter Kit 时就设置此变量。

- 如果您的 LANG 环境变量已经设置为另一值，则在安装之后进行以下符号链接：

```
ln -s $ISHOME/locale/en_US $ISHOME/locale/$LANG
```

混合 OLAP 客户机和服务器的服务级别

IBM 建议您使 DB2 OLAP Starter Kit 的客户机和服务器组件处于相同的版本和修订包级别。但是，在某些情况下，可能可以混合客户机和服务器组件的不同服务级别：

使用一个版本的但处于不同服务级别的客户机和服务器

IBM 不支持也不建议将较新的客户机与较旧的服务器配合使用。然而，可能可以将较旧的客户机与较新的服务器配合使用，尽管 IBM 不支持这种方式。这样做可能会遇到一些问题。例如：

- 来自服务器的消息可能不正确。可以通过将客户机上的 message.MDB 文件升级，使它与服务器上的级别相匹配，从而解决此问题。
- 新服务器功能部件不起作用。当您试图使用新功能部件时，客户机和 / 或服务器可能会失败。
- 客户机可能没有与服务器正确连接。

将一个版本内的多个服务器与单个客户机配合使用

如果需要将客户机与不同机器或操作系统上的几个 OLAP 服务器连接，IBM 建议让它们都处于同一版本和服务级别。客户机至少应该处于最低的服务器级别。如果遇到了问题，则可能需要使用不同的客户机来与适当的主机匹配，或者将所有客户机和服务器都升级到同一服务级别。

混合不同版本中的客户机和服务器

IBM 不支持将版本 7.1 的 OLAP Starter Kit 客户机和服务器与版本 7.2 的客户机和服务器一起使用。当 IBM OLAP 产品被升级到新的版本级别时，通常会有一些网络更新和数据格式更改，这要求客户机和服务器处于同一版本级别。

将 IBM 产品 (DB2 OLAP Starter Kit) 与 Hyperion 产品 (Hyperion Essbase 和 Hyperion Integration Server) 混合

IBM 不支持将 IBM 公司的 OLAP 客户机和服务器与 Hyperion Solutions 的 OLAP 客户机和服务器混合在一起使用。尽管在某些情况下将这些组件混合在一起可以工作，但是，功能部件上的某些差异可能会导致问题。

缺少 OLAP Spreadsheet Add-in EQD 文件

在 DB2 OLAP Starter Kit 中，Spreadsheet add-in 中有一个组件称为“查询设计程序”(EQD)。EQD 的联机帮助菜单包括一个称为**教程**的按钮，但它不显示任何内容。EQD 教程中应显示的内容是 *OLAP Spreadsheet Add-in User's Guide for Excel* 和 *OLAP Spreadsheet Add-in User's Guide for 1-2-3* 的第二章中的一小部分。EQD 教程中的所有信息在“信息中心”中是在这些书籍的 HTML 版本中提供的，这些书籍的 PDF 版本也提供了所有这些信息。

Information Catalog Manager Administration Guide

信息目录管理器初始化实用程序

借助“初始化信息目录管理器”(ICM)实用程序,现在可以使用以下命令来将 SQL 语句追加到 CREATE TABLE 语句的末尾:

```
CREATEIC \DBTYPE dbtype \DGNAME dgname \USERID userid \PASSWORD password
\KA1 userid \TABOPT "directory:tabopt.file"
```

可以从安装了 DB2 的目录中指定 CREATEIC 实用程序中的 TABOPT 关键字。TABOPT 关键字后面的值是具有全路径的 *tabopt.file* 文件名。如果目录名包含空白,则用引号将该名称引起来。*tabopt.file* 文件的内容必须包含要追加至 CREATE TABLE 语句的信息。可以使用下面的任何 SQL 语句来写入此 *tabopt.file* 文件。ICM 实用程序将读取此文件,然后将它追加至 CREATE TABLE 语句。

表 20. SQL 语句

IN MYTABLESPACE	利用它在 MYTABLESPACE 中的数据来创建表
DATA CAPTURE CHANGES	创建表,并且以扩展格式记录 SQL 更改
IN ACCOUNTING INDEX IN ACCOUNT_IDX	利用它在 ACCOUNTING 中的数据以及它在 ACCOUNT_IDX 中的索引来创建表

内容文件的最大大小为 1000 个单字节字符。

此新功能仅在 Windows 和 UNIX 系统上可用。

许可证发放问题

如果您接收到以下消息:

```
FLG0083E: 您没有“IBM 信息目录管理器初始化”实用程序的有效许可证。请与当地的软件经销商或 IBM 市场营销代表联系。
```

您必须购买“DB2 仓库管理器”或“IBM DB2 OLAP 服务器”并安装“信息目录管理器”组件,该组件包括“信息目录初始化”实用程序。

安装问题

如果安装了“DB2 仓库管理器”或“IBM DB2 OLAP 服务器”,并在同一工作站上安装了另一“信息目录管理器管理员”组件(使用“DB2 通用数据库 CD-ROM”),则您可能已经覆盖了“信息目录初始化”实用程序。在这种情况下,从 `\sqllib\bin` 目录查找文件 `createic.bak` 和 `flgnmocr.bak`,并分别将它们重命名为 `createic.exe` 和 `flgnmocr.exe`。

如果从“DB2 通用数据库”安装其他“信息目录管理器”组件,则这些组件必须位于不同于安装有“数据仓库管理器”的工作站上。有关详情,参见《DB2 数据仓库管理器安装指南》中的『第 3 章 安装信息目录管理器组件』。

使用 DB2 版本 7 信息目录管理器来访问 DB2 版本 5 信息目录

“DB2 版本 7 信息目录管理器”子组件是由 DB2 版本 7 安装过程配置的，它支持对存储在 DB2 版本 6 和 DB2 版本 7 数据库中的信息目录的访问。可修改这些子组件的配置以访问存储在 DB2 版本 5 数据库中的信息目录。“DB2 版本 7 信息目录管理器”子组件不支持对 DB2 版本 2 或所有其他先前版本中的数据的访问。

要设置“信息目录管理员”、“信息目录用户”和“信息目录初始化实用程序”以访问存储在 DB2 版本 5 数据库中的信息目录：

1. 在没有安装“DB2 版本 7 信息目录管理器”的工作站上安装“DB2 Connect 企业版版本 6”。

“DB2 通用数据库企业版”和“DB2 通用数据库扩充企业版”包括“DB2 Connect 企业版”。如果安装了这些 DB2 产品中任一种产品的版本 6，则不必单另安装 DB2 Connect。

限制：不能在同一 Windows NT 或 OS/2 工作站上安装多个版本的 DB2。可在另一 Windows NT 工作站或在 OS/2 或 UNIX 工作站上安装 DB2 Connect。

2. 配置“信息目录管理器”和 DB2 Connect 版本 6 以对 DB2 版本 5 数据进行访问。有关详情，参见 *DB2 Connect 用户指南*。下列步骤是所需步骤的概述：
 - a. 在 DB2 版本 5 系统上，使用“DB2 命令行处理器”来编目“信息目录管理器”将要访问的版本 5 数据库。
 - b. 在 DB2 Connect 系统上，使用“DB2 命令行处理器”来编目：
 - DB2 版本 5 系统的 TCP/IP 节点
 - DB2 版本 5 系统的数据库
 - DB2 版本 5 系统的 DCS 项目
 - c. 在带有“信息目录管理器”的工作站上，使用“DB2 命令行处理器”来编目：
 - DB2 Connect 系统的 TCP/IP 节点
 - DB2 Connect 系统的数据库

有关编目数据库的信息，参见 *DB2 通用数据库安装和配置补遗*。

3. 在带有“信息目录管理器”的数据仓库上，将 DB2 CLI 软件包与将要通过 DB2 Connect 访问的每个数据库进行绑定。

下列 DB2 命令给出了与 v5database 进行绑定的示例，一个假定的 DB2 版本 5 数据库。使用“DB2 命令行处理器”来发出下列命令。db2cli.lst 和 db2ajgrt 位于 \sqllib\bnd 目录中。

```
db2 connect to v5database user userid using password
db2 bind db2ajgrt.bnd
db2 bind @db2cli.lst blocking all grant public
```

其中，*userid* 是 v5database 的用户标识，而 *password* 是该用户标识的口令。

在 db2cli.lst 与 DB2 版本 5 数据库进行绑定时出错。会出现此错误的原因是大对象 (LOB) 在此配置中不受支持。此错误将不会影响数据仓库代理程序对 DB2 版本 5 数据库的访问。

“DB2 通用数据库版本 5”的“修订包 14”（是在 2000 年 6 月提供的）是通过 DB2 Connect 访问“DB2 版本 5”所必需的。参考该“修订包”中的 APAR 号 JR14507。

设置信息目录

Chapter 1 "Setting up an information catalog" 第一节中的步骤 2 中说到:

```
When you install either the DB2 Warehouse Manager  
or the DB2 OLAP Server, a default information catalog  
is created on DB2 Universal Database for Windows NT.
```

该语句不正确。必须定义新的信息目录。参见 "Creating the Information Catalog" 一节以了解详情。

与其他产品交换元数据

在 Chapter 6 "Exchanging metadata with other products" 的 "Identifying OLAP objects to publish" 一节中, 在第二段有一句的内容如下:

```
When you publish DB2 OLAP Integration Server metadata, a linked relationship  
is created between an information catalog "dimensions within a  
multi-dimensional database" object type and a table object  
in the OLAP Integration Server.
```

该语句应为:

```
When you publish DB2 OLAP Integration Server metadata, a linked relationship  
is created between an information catalog "dimensions within a  
multi-dimensional database object and a table object".
```

此语句还出现在 Appendix C, "Metadata mappings" 的 "Metadata mappings between the Information Catalog Manager and OLAP Server" 一节中。

使用 flgnxoln 命令来交换元数据

在 Chapter 6 "Exchanging Metadata" 中, 有一节的标题是 "Identifying OLAP objects to publish"。在此节的末尾有一个使用 flgnxoln 命令来将 OLAP 服务器元数据发布至信息目录的示例。该示例错误地将 db2olap.ctl 和 db2olap.ff 文件的目录显示为 x:\Program Files\sqllib\logging。目录名应为 x:\Program Files\sqllib\exchange, 如第 87 页上所述。

使用 MDISDGC 命令来交换元数据

Chapter 6. Exchanging metadata with other products: "Converting MDIS-conforming metadata into a tag language file", 第 97 页。不能从 MS-DOS 命令提示符发出 MDISDGC 命令。必须从 DB2 命令窗口发出 MDISDGC 命令。"Converting a tag language file into MDIS-conforming metadata" 一节的第一句还提到必须从 MS-DOS 命令提示符发出 DGMDISC 命令。必须从 DB2 命令窗口发出 DGMDISC 命令。

调用程序

Information Catalog Administration Guide 中的某些示例显示了包含目录名 Program Files 的一些命令。当您调用包含作为路径名一部分的 Program Files 的程序时, 必须用双引号将程序调用引起来。例如, 附录 B "Predefined Information Catalog Manager object types" 的 "Initializing your information catalog with the predefined object types" 一节中包含一个示例。若使用本节中的示例, 则当您从 DOS 提示符运行它时将接收到错误。以下示例是正确的:


```
"X:Program Files\SQLLIB\SAMPLES\SAMPDATA\DGWDEMO"  
/T userid password dgname
```

Information Catalog Manager Programming Guide and Reference

信息目录管理器原因码

在附录 D: Information Catalog Manager 原因码中, 某些文本可能会在以下原因码的列的最右边被截断: 31014、32727、32728、32729、32730、32735、32736、32737、33000、37507、37511 和 39206。若文本被截断, 则查看本书的 HTML 版本以查看完整列。

信息目录管理器用户指南

在第 2 章中，有一节是『注册服务器节点和远程信息目录』。本节列示了在使用“信息目录管理器”注册远程信息目录之前可通过“DB2 控制中心”完成的步骤。本节的最后一段提到：在通过“DB2 控制中心”完成一系列步骤之后（添加系统、添加实例和添加数据库），必须关闭“控制中心”才能打开“信息目录管理器”。该信息是错误的。不需要关闭“控制中心”就可以打开“信息目录管理器”。

同样的校正措施也适用于联机帮助任务“注册服务器节点和远程信息目录”，以及“注册服务器节点和信息目录”窗口的联机帮助。

信息目录管理器: 联机消息

消息 FLG0260E

消息说明的第二句应为:

该错误导致失败的信息目录回滚。
信息目录处于不稳定状态, 但未发生任何更改。

消息 FLG0051E

消息说明中的第二个公告牌应为:

信息目录包含了过多的对象或对象类型。

管理员响应应为:

使用导入功能从当前信息目录删除某些对象或对象类型。

消息 FLG0003E

消息说明应为:

信息目录必须在使用之前进行注册。
信息目录可能未正确注册。

消息 FLG0372E

消息说明的第一句应为:

对象的 ATTACHMENT-IND 值被忽略, 原因是该对象是“附件”对象。

消息 FLG0615E

该消息的第二句应为:

“信息目录管理器”遇到了意外的数据库错误,
或者在当前目录或路径中找不到绑定文件。

信息目录管理器: 联机帮助

“信息目录”窗口: “已选择”菜单“打开”项的联机帮助为“打开已选择对象”, 这是不正确的。它应为“打开“定义搜索”窗口”。

用于 Web 的“信息目录管理器”

在使用位于 DB2 UDB OS/390 版系统上的信息目录时, 区分大小写搜索不可用。对于简单搜索和高级搜索都是这样。联机帮助不解释以下情况: 对于简单搜索, 对 DB2 UDB OS/390 版信息目录的所有搜索都是区分大小写的。此外, 所有分组类别对象都是可展开的, 即使没有任何底层对象。

DB2 仓库管理器安装指南

仓库变换器的软件需求

必须在您计划使用仓库变换器的数据库上安装 “Java 开发者工具箱” (JDK) 版本 1.1.8 或更新版本。

SAP R/3 的连接器的

当将列从 SAP R/3 商业对象的字段映射至 DB2 表时，生成的一些列名可能会长于 30 个字符。在此情况下，生成的列名将只反映 SAP 字段名的前 30 个字符。如果生成的名称不是您想要的名称，则可以使用该表的 “特性” 笔记本来更改它。

安装先决条件

设置 RFC_INI 环境变量。例如：Set RFC_INI=c:\rfcapl.ini。在设置此变量之后，必须重新引导机器。

用于 Web 的连接器的

如果在运行 “用于 Web 的连接器” 时出现问题，则 IBM 服务机构可能会请求您对 “连接器” 发送跟踪。

要对 “用于 Web 的连接器” 启用跟踪，应将 “仓库中心” 代理程序跟踪设置为大于 0 的级别。跟踪文件被称为 WSApiid.log，其中 pid 是代理程序的 Windows 进程标识。跟踪文件是在 \sqllib\logging 目录中创建的。

安装先决条件

安装 “Java 运行时环境” (JRE) 或者 “Java 虚拟机” (JVM) 版本 1.2.2 或更新版本，并使它作为缺省值。要使 JRE 的版本成为缺省值，将 1.2.2 JRE 的路径添加到系统 PATH 变量中 (例如 C:\JDKs\IBM\java12\bin;)。在更改缺省 JRE 之后，必须重新引导机器。如果您没有安装 Java，则可以从 “数据仓库连接器” 安装 CD 中安装它。

Query Patroller Administration Guide

DB2 Query Patroller 客户机是一个独立组件

DB2 Query Patroller 客户机是一个单独的组件，它不是 DB2 管理客户机的的组件。这意味着它不是在“DB2 管理客户机”安装期间安装的，就象“Query Patroller 安装指南”中所指示的那样。而 Query Patroller 客户机必须单独安装。

Query Patroller 客户机和 Query Patroller 服务器的版本和级别必须相同。

使用 dqpmigrate 从 DB2 Query Patroller 的版本 6 迁移

如果“版本 7 Query Patroller 服务器”是在“版本 6 Query Patroller 服务器”上覆盖安装的，必须使用 dqpmigrate 命令。对于“修订包 2”或更新版本，不需要人工运行 dqpmigrate，因为“修订包”的安装会运行此命令。如果不使用此命令，在 V6 中定义的现存用户对在版本 7 中添加的几个新存储过程没有 EXECUTE 特权。

注： dqpmigrate.bnd 在 sqllib/bnd 目录中，而 dqpmigrate.exe 在 sqllib/bin 目录中。

要人工使用 **dqpmigrate** 授予 EXECUTE 特权，在安装“修订包”之后执行下列步骤：

1. 通过输入以下命令，将 /sqllib/bnd/dqpmigrate.bnd 软件包文件绑定至安装有 Query Patroller 服务器的数据库：

```
db2 bind dqpmigrate.bnd
```

2. 通过输入以下命令来执行 **dqpmigrate**：

```
dqpmigrate dbalias userid passwd
```

Enabling Query Management

在“Enabling Query Management”下的“Getting Started”一章中，文本应为：

```
You must be the owner of the data base, or you must have SYSADM,  
SYSCTRL, or SYSMANT authority to set database configuration parameters.
```

控制表的表空间的位置

在 Chapter 1 System Overview 的 DB2 Query Patroller Control Tables 中，在该节的第一段末尾将添加下列文本：

The table space for the DB2 Query Patroller control tables must reside in a single-node nodegroup, or DB2 Query Patroller will not function properly.

dqptest 命令的新参数

在 Chapter 2 Getting Started 的 Starting and Stopping DB2 Query Patroller 中，在最后一段后面将添加下列文本：

New Parameters for the **dqptest** command:

RESTART parameter:

Allows the user to replace the host name and/or the node type of the specified node in the dqpnodes.cfg file. DB2 Query Patroller will be started on this node.

注： Before running the DQPSTART command with the RESTART parameter, ensure the following:

1. DB2 Query Patroller is already stopped on the host that is going to be replaced.
2. DB2 Query Patroller is not already running on the new host.

The syntax is as follows:

```
dqptest nodenum node_num restart hostname server | agent | none
```

ADDNODE parameter:

Allows the user to add a new node to the dqpnodes.cfg file. DB2 Query Patroller will be started on this node after the new node entry is added to the dqpnodes.cfg file. The syntax is as follows:

```
dqptest nodenum node_num addnode hostname server | agent | none
```

DROPNODE parameter:

Allows the user to drop a node from the dqnodes.cfg file. DB2 Query Patroller will be stopped on this node before the node entry is dropped from the dqpnodes.cfg file. The syntax is as follows:

```
dqptest nodenum node_num dropnode
```

iwm_cmd 命令的新参数

已经将新的 **-v** 参数添加到 **iwm_cmd** 命令中，以允许用户恢复在指定节点上运行的作业的状态。只允许恢复不活动节点上的作业。节点发生故障且某些作业在该节点上运行或者被取消时，应该发出此命令。将重新提交处于“正在运行”状态的作业，并设置回到“已排队”状态。处于“正在取消”的作业将被设置为“已取消”状态。

部分语法为如下所示：

```
▶▶ iwm_cmd -u user_id -p password -v node_id_to_recover
```

node_id_to_recover

指定将在它上面恢复作业的节点。

新的注册表变量：DQP_RECOVERY_INTERVAL

有一个称为 DQP_RECOVERY_INTERVAL 的新注册表变量，用来设置 iwm_scheduler 搜索恢复文件的时间间隔（以分钟计）。缺省值为 60 分钟。

启动 Query Administrator

在 "Using QueryAdministrator to Administer DB2 Query Patroller" 一章中，提供了用于在 Windows 上从“开始”菜单启动 QueryAdministrator 的指导。第一步提供了以下文本：

```
If you are using Windows, you can select DB2  
Query Patroller -> QueryAdministrator  
from the IBM DB2 program group.
```

该文本应为：

```
DB2 Query Patroller -> QueryAdmin.
```

用户管理

在 "Using QueryAdministrator to Administer DB2 Query Patroller" 一章的 "User Administration" 一节中，“最长所用时间”参数指示如果值被设置为 0 或 -1，则查询将始终运行完成状态。此参数不能设置为负值。该文本应指示如果值被设置为 0，则查询将始终运行完成状态。

Max Queries 参数指定 DB2 Query Patroller 将同步运行的最大作业数。Max Queries 必须是 0 到 32767 范围内的整数。

创建作业队列

在 "Using QueryAdministrator to Administer DB2 Query Patroller" 一章的 "Job Queue Administration" 一节中，“创建作业队列”步骤中的屏幕捕捉应在第二步之后显示。一旦在 QueryAdministrator 工具的 Job Queue Administration 页上单击了新建按钮，有关新建 Job Queue 窗口的信息就会打开。

出现 Job Queue 页或 Job Queue 选项卡的地方应分别为 Job Queue Administration 页和 Job Queue Administration 选项卡。

使用“命令行接口”

要让对 DB2 Query Patroller 系统具有“用户”权限的用户提交查询并创建结果表，用户可能需要对数据库具有 CREATETAB 权限。如果未设置 DQP_RES_TBLSPC 简要表变量或是 DQP_RES_TBLSPC 简要表变量被设置为缺省表空间名，则用户不需要数据库的 CREATETAB 权限。在这种情况下，将会成功地创建结果表，原因是用户具有在缺省表空间中创建表的权限。

“查询使能器”注释

- 在用使用键组游标的第三方查询工具时，查询将不会被截断。为使“查询使能器”截断这些查询，必须修改 db2cli.ini 文件以包括：

```
[common]  
DisableKeySetCursor=1
```

- 对于 AIX 客户机，请确保未设置环境变量 LIBPATH。随 JDK 交付的库 libXext.a 与 /usr/lib/X11 子目录中的库不兼容。这会使“查询使能器 GUI”出现问题。

DB2 Query Patroller Tracker 可能会返回空白列页

修订包 3 包括对 DB2 Query Patroller Tracker 的修正。Tracker 现在将正确地报告没有命中任何列的查询。这样的查询的一个示例为 "SELECT COUNT(*) FROM ..."。由于此种查询不会命中表中的任何列，所以 Tracker 将为列页提供一个空白页。此空白列页不是一个错误。

Query Patroller 和复制工具

Query Patroller 版本 7 将拦截复制工具（asnapply、asnccp、djra 和 analyze）的查询，并导致这些工具失灵。一种解决方案是在运行这些工具时禁用动态查询管理。

Appendix B. Troubleshooting DB2 Query Patroller Clients

在 Appendix B, Troubleshooting DB2 Query Patroller Clients, section: Common Query Enabler Problems, problem #2 一节中，应该将第一个公告牌的文本更改为：

Ensure that the path setting includes jre.

第6部分 应用程序开发

Administrative API Reference	275
db2ArchiveLog (新 API)	275
db2ArchiveLog	276
db2ConvMonStream.	278
db2DatabasePing (新 API)	278
db2DatabasePing - Ping 数据库	279
db2HistData	280
db2HistoryOpenScan	281
db2XaGetInfo (新 API)	282
db2XaGetInfo - 获取资源管理器的信息	283
db2XaListIndTrans (取代 sqlxphqr 的新 API)	283
db2XaListIndTrans - 列示不确定事务	284
db2GetSnapshot - 获取快照	287
忘记日志记录	287
sqlaintp - 获取错误消息	287
sqlbctcq - 关闭表空间容器查询	287
sqlubkp - 备份数据库	287
sqlureot - 重组表	288
sqlurestore - 恢复数据库	288
关于“AIX 扩展共享内存支持”(EXTSHM)的文档 错误	288
SQLFUPD	288
锁定列表	288
SQLEDBDESC	289
SQLFUPD 文档错误	289
应用程序构建指南	291
第 1 章 介绍	291
受支持软件	291
样本程序	293
第 3 章 构建 DB2 应用程序的一般信息	293
构建文件、Makefile 和错误检查实用程序	293
第 4 章 构建 Java 小应用程序和应用程序	293
设置环境	293
OS/2 上的 JDK 级别	295
HP-UX 上的 Java2	295
第 5 章 构建 SQL 过程	295
设置 SQL 过程环境	295
设置编译器环境变量	296
定制编译命令	297
保留中间文件	300
备份与恢复	300
创建 SQL 过程	301
调用存储过程	301
分布编译型 SQL 过程	301
第 7 章 构建 HP-UX 应用程序	302
HP-UX C	302
HP-UX C++	302
第 9 章 构建 OS/2 应用程序	302
VisualAge C++ OS/2 版的版本 4.0	302
第 10 章 构建 PTX 应用程序	303
ptx/C++	303
第 12 章 构建 Solaris 应用程序	303
SPARCCompiler C++	303
第 13 章 为 Windows 32 位操作系统构建应用程序	303
VisualAge C++ 版本 4.0	303
Application Development Guide	305
Chapter 2. Coding a DB2 Application.	305
对 Microsoft Visual C++ 激活 IBM DB2 通用数 据库项目和工具加载件	305
Chapter 6. Common DB2 Application Techniques	305
生成序列值	305
控制序列行为	306
用序列对象提高性能	307
比较序列对象和身份列	307
Chapter 7. Stored Procedures	307
DECIMAL 类型在 Linux Java 例程中失效	307
在递归存储过程中使用游标	308
编写 OLE 自动存储过程	308
Chapter 12. Working with Complex Objects: User-Defined Structured Types	308
将结构化类型属性插入到列中	308
Chapter 13. Using Large Objects (LOBs).	309
联合数据库系统中的大对象 (LOB) 支持	309
DB2 如何检索 LOB	309
应用程序可以如何使用 LOB 定位器	309
对 LOB 的限制	310
LOB 与非 LOB 数据类型之间的映射	310
调整系统	310
Part 5. DB2 Programming Considerations.	310
IBM DB2 OLE DB 提供器	310
Chapter 20. C 和 C++ 编程.	311
存储过程、函数和方法的 C 和 C++ 类型	311
Chapter 21. Programming in Java	312
PARAMETER STYLE JAVA 过程和函数中的 Java 方法签名	312
连接至 JDBC 小应用程序服务器	312
Appendix B. Sample Programs	313
CLI Guide and Reference	315
使用运行时客户机绑定数据库实用程序	315
在 CLI 应用程序中使用静态 SQL	315
创建静态 JDBC/ODBC/CLI 简要表的限制	315
ADT 变换	315
Chapter 3. Using Advanced Features	316
编写多线程应用程序	316
可滚动游标	317
Server-side Scrollable Cursor Support for OS/390.	317
使用复合 SQL	318
使用存储过程	318

在 CLI 中编写存储过程	318	MQSEND	346
CLI 存储过程和自动绑定	318	MQSUBSCRIBE	348
Chapter 4. Configuring CLI/ODBC and Running		MQUNSUBSCRIBE	349
Sample Applications	319	MULTIPLY_ALT	350
配置关键字	319	REC2XML	351
Chapter 5. DB2 CLI Functions	319	ROUND	355
SQLBindFileToParam - 将 LOB 文件参考绑定至		WEEK_ISO	356
LOB 参数	319	表函数	357
SQLNextResult - 将下一个结果集与另一语句句柄		MQREADALL	357
相关联	319	MQRECEIVEALL	358
用途	319	过程	360
语法	319	GET_ROUTINE_SAR	361
函数自变量	319	PUT_ROUTINE_SAR	361
用法	319	Chapter 5. Queries	362
返回码	320	select-statement 语法图	362
诊断	320	select-statement/fetch-first-clause	363
限制	321	Chapter 6. SQL Statements	363
引用	321	现在支持更新分区键	363
Appendix D. Extended Scalar Functions	321	语句: ALTER TABLE	363
日期和时间函数	321	语句: CREATE TABLE	364
Appendix K. Using the DB2 CLI/ODBC/JDBC Trace		语句: DECLARE GLOBAL TEMPORARY	
Facility.	322	TABLE PARTITIONING KEY	
消息参考	323	(column-name,...)	364
获取消息和 SQLSTATE 帮助	323	语句: UPDATE	364
DB2 Connect 中的 SQLCODE 重新映射更改	323	用于 Unicode 数据库的较大索引键	364
新消息和已更改的消息	323	ALTER TABLE	364
调用层接口 (CLI) 消息	323	CREATE INDEX	364
DB2 消息	323	CREATE TABLE	364
DBI 消息	324	ALTER SEQUENCE	364
数据仓库中心 (DWC) 消息	324	ALTER SEQUENCE	365
SQL 消息	325	ALTER TABLE	367
校正过的 SQLSTATES	327	复合 SQL (嵌入式)	370
SQL Reference	329	复合语句 (动态)	370
SQL Reference 是在一个 PDF 文件中提供的	329	复合语句 (动态)	371
Chapter 3. Language Elements	329	CREATE FUNCTION (源或模板)	373
Naming Conventions and Implicit Object Name		CREATE FUNCTION (SQL 标量、表或行)	374
Qualifications	329	CREATE METHOD	375
DATALINK Assignments	329	CREATE METHOD	376
表达式	329	CREATE SEQUENCE	376
语法图	329	CREATE SEQUENCE	377
OLAP 函数	330	CREATE TRIGGER	380
序列参考	332	CREATE TRIGGER	381
Chapter 4. Functions	335	CREATE WRAPPER	383
启用新函数和过程	335	DECLARE CURSOR	383
标量函数	335	DELETE	383
ABS 或 ABSVAL	335	DROP	384
DECRYPT_BIN 和 DECRYPT_CHAR	335	GRANT (序列特权)	385
ENCRYPT	336	GRANT (序列特权)	386
GETHINT	338	INSERT	386
IDENTITY_VAL_LOCAL	339	SELECT INTO	387
LCASE 和 UCASE (Unicode)	343	SET ENCRYPTION PASSWORD	388
MQPUBLISH	343	SET ENCRYPTION PASSWORD	389
MQREAD	344	SET transition-variable	389
MQRECEIVE	345	SET Variable	390
		UPDATE	392

Chapter 7. SQL Procedures 现在名为 Chapter 7.	附录 C DB2 CLI 和 ODBC	420
SQL Control Statements	ODBC Unicode 应用程序	420
SQL 过程语句	Data Movement Utilities Guide and Reference	421
SQL Procedure Statement	附录 C 导出 / 导入 / 装入实用程序文件格式	421
FOR		
FOR		
Compound Statement 更改为 Compound Statement (Procedure)		397
RETURN		398
RETURN		399
SIGNAL		400
SIGNAL		401
Appendix A. SQL Limits		403
Appendix D. Catalog Views		403
SYSCAT.SEQUENCES		403
DB2 存储过程构建器.		405
Java 1.2 支持 DB2 存储过程构建器		405
DB2 存储过程的远程调试		405
在 Windows、OS/2 或 UNIX 平台上构建 SQL 过程		406
在 Solaris 平台上使用 DB2 存储过程构建器		406
已知问题和限制.		406
将 DB2 存储过程构建器与繁体中文语言环境配合使 用		406
UNIX (AIX、Sun Solaris 和 Linux) 安装和存储过 程构建器		407
在 OS/390 上构建 SQL 存储过程.		407
调试 SQL 存储过程		407
导出 Java 存储过程		407
在 OS/390 上插入存储过程.		407
在 workstation 服务器上为“SQL 存储过程”设置构建选 项		408
自动刷新 WLM 地址空间以便在 OS/390 上构建存 储过程.		408
在 OS/390 上开发 Java 存储过程.		408
为 MQ Series 和 OLE DB 构建 DB2 表用户定义函 数 (UDF).		408
Unicode 更新.		409
介绍		409
DB2 Unicode 数据库和应用程序		409
文档更新		411
SQL Reference		412
Chapter 3 Language Elements		412
数据类型提升		412
在数据类型之间强制转型.		412
指定和比较		413
用于结果数据类型的规则.		414
用于字符串转换的规则		415
表达式.		415
谓词		415
第 4 章 功能		415
标量函数		415
CLI Guide and Reference.		416
Chapter 3. Using Advanced Features		416
编写 DB2 CLI Unicode 应用程序.		416

Administrative API Reference

| **db2ArchiveLog (新 API)**

db2ArchiveLog

关闭并截断可恢复数据库的活动日志文件。如果启用了用户出口，则发出归档请求。

权限

下列其中一项:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必需的连接

此 API 自动建立与指定数据库的连接。如果与指定数据库的连接已经存在，则 API 将返回错误。

注意事项

db2ApiDf.h

C API 语法

```
/* File: db2ApiDf.h */
/* API: Archive Active Log */
SQL_API_RC SQL_API_FN
db2ArchiveLog (
    db2UInt32 version,
    void *pDB2ArchiveLogStruct,
    struct sqlca * pSqlca);

typedef struct
{
    char          *piDatabaseAlias;
    char          *piUserName;
    char          *piPassword;
    db2UInt16     iAllNodeFlag;
    db2UInt16     iNumNodes;
    SQL_PDB_NODE_TYPE *piNodeList;
    db2UInt32     iOptions;
}
```

类属 API 语法

```

/* File: db2ApiDf.h */
/* API: Archive Active Log */
SQL_API_RC SQL_API_FN
db2gArchiveLog (
    db2UInt32 version,
    void *pDB2ArchiveLogStruct,
    struct sqlca * pSqlca);

typedef struct
{
    db2UInt32          iAliasLen;
    db2UInt32          iUserNameLen;
    db2UInt32          iPasswordLen;
    char               *piDatabaseAlias;
    char               *piUserName;
    char               *piPassword;
    db2UInt16          iAllNodeFlag;
    db2UInt16          iNumNodes;
    SQL_PDB_NODE_TYPE *piNodeList;
    db2UInt32          iOptions;
}

```

API 参数

版本 输入。指定作为第二个参数 *pDB2ArchiveLogStruct* 传送的变量的版本和发行版级别。

pDB2ArchiveLogStruct

输入。指向 *db2ArchiveLogStruct* 结构的指针。

pSqlca

输出。指向 *sqlca* 结构的指针。

iAliasLen

输入。一个 4 字节不带符号的整数，表示数据库别名的长度（以字节计）。

iUserNameLen

一个 4 字节不带符号的整数，表示用户名的长度（以字节计）。如果未使用任何用户名，则设置为零。

iPasswordLen

输入。一个 4 字节不带符号的整数，表示口令的长度（以字节计）。如果未使用任何口令，则设置为零。

piDatabaseAlias

输入。一个字符串，包含要归档的归档日志的数据库的数据库别名（在系统数据库目录中作了编目）。

piUserName

输入。一个字符串，包含要在尝试连接时使用的用户名。

piPassword

输入。一个字符串，包含要在尝试连接时使用的口令。

iAllNodeFlag

仅 MPP。输入。一个标志，指示是否应该将操作应用于 *db2nodes.cfg* 文件中列示的所有节点。有效值为：

db2ArchiveLog - 归档活动日志

DB2ARCHIVELOG_NODE_LIST

应用于在 *piNodeList* 中传送的节点列表中的节点。

DB2ARCHIVELOG_ALL_NODES

应用于所有节点。*piNodeList* 应该为 NULL。这是缺省值。

DB2ARCHIVELOG_ALL_EXCEPT

应用于所有节点，但是在 *piNodeList* 中传送的节点列表中的节点除外。

iNumNodes

仅 MPP。输入。指定 *piNodeList* 数组中的节点数。

piNodeList

仅 MPP。输入。指向要对其应用归档日志操作的节点号数组的指针。

iOptions

输入。留作将来使用。

db2ConvMonStream

在 Usage Notes 中，快照变量数据流类型 `SQLM_ELM_SUBSECTION` 的结构应为 `sqlm_subsection`。

db2DatabasePing (新 API)

db2DatabasePing - Ping 数据库

测试客户机与数据库服务器之间的基本连接的网络响应时间。如果主机数据库服务器是直接通过 DB2 Connect 或通过网关访问的，则应用程序可使用此 API。

权限

无

必需的连接

数据库

注意事项

db2ApiDf.h

C API 语法

```

/* File: db2ApiDf.h */
/* API: Ping Database */
/* ... */

SQL_API_RC SQL_API_FN
db2DatabasePing (
    db2Uint32    versionNumber,
    void        *pParmStruct,
    struct sqlca *pSqlca);
/* ... */

typedef SQL_STRUCTURE db2DatabasePingStruct
{
    char        iDbAlias[SQL_ALIAS_SZ + 1];
    db2Uint16   iNumIterations;
    db2Uint32   *poElapsedTime;
}

```

类属 API 语法

```

/* File: db2ApiDf.h */
/* API: Ping Database */
/* ... */
SQL_API_RC SQL_API_FN
db2gDatabasePing (
    db2Uint32    versionNumber,
    void        *pParmStruct,
    struct sqlca *pSqlca);
/* ... */

typedef SQL_STRUCTURE db2gDatabasePingStruct
{
    db2Uint16   iDbAliasLength;
    char        iDbAlias[SQL_ALIAS_SZ];
    db2Uint16   iNumIterations;
    db2Uint32   *poElapsedTime;
}

```

API 参数

versionNumber

输入。应用程序正在使用的“DB2 通用数据库”或 DB2 Connect 产品的版本和发行版。

db2ArchiveLog - 归档活动日志

注：应对 DB2 版本 7.1 或更高版本使用常量 db2Version710 或更高值。

iDbAliasLength

输入。数据库别名的长度。

注：目前不使用此参数。它被留作将来使用。

iDbAlias

输入。数据库别名。

注：目前不使用此参数。它被留作将来使用。

iNumIterations

输入。测试请求迭代数。该值必须在 1 和 32767 之间，包括 1 和 32767。

poElapsedTime

输出。指向 32 位整数数组的指针，其中元素数等于 iNumIterations。该数组中的每个元素都将包含一个测试请求迭代的所用时间（以微秒计）。

注：应用程序负责在调用此 API 之前为此数组分配内存。

pSqlca

输出。指向 *sqlca* 结构的指针。有关此结构的详情，参见 *Administrative API Reference*。

使用说明

数据库连接必须存在，才能调用此 API，否则将会产生错误。

此函数还可使用 PING 命令来调用。有关此命令的描述，参见 *Command Reference*。

db2HistData

下列条目应添加至表 11。db2HistData 结构中的字段：

字段名	数据类型	描述
oOperation	char	参见表 12。
oOptype	char	参见表 13。

应在表 11 后添加下表。

表 12。db2HistData 结构中的 oOperation 的有效事件值

值	描述	C 定义	COBOL/FORTRAN 定义
A	添加表空间	DB2HISTORY_OP_ADD_TABLESPACE	DB2HIST_OP_ADD_TABLESPACE
B	备份	DB2HISTORY_OP_BACKUP	DB2HIST_OP_BACKUP
C	装入副本	DB2HISTORY_OP_LOAD_COPY	DB2HIST_OP_LOAD_COPY
D	已删除的表	DB2HISTORY_OP_DROPPED_TABLE	DB2HIST_OP_DROPPED_TABLE
F	前滚	DB2HISTORY_OP_ROLLFWD	DB2HIST_OP_ROLLFWD
G	重组表	DB2HISTORY_OP_REORG	DB2HIST_OP_REORG
L	装入	DB2HISTORY_OP_LOAD	DB2HIST_OP_LOAD
N	重命名表空间	DB2HISTORY_OP_REN_TABLESPACE	DB2HIST_OP_REN_TABLESPACE

值	描述	C 定义	COBOL/FORTRAN 定义
O	删除表空间	DB2HISTORY_OP_DROP_TABLESPACE	DB2HIST_OP_DROP_TABLESPACE
Q	停顿	DB2HISTORY_OP_QUIESCE	DB2HIST_OP_QUIESCE
R	恢复	DB2HISTORY_OP_RESTORE	DB2HIST_OP_RESTORE
S	运行统计	DB2HISTORY_OP_RUNSTATS	DB2HIST_OP_RUNSTATS
T	改变表空间	DB2HISTORY_OP_ALT_TABLESPACE	DB2HIST_OP_ALT_TBS
U	卸装	DB2HISTORY_OP_UNLOAD	DB2HIST_OP_UNLOAD

还将添加下表。

表 13. 有效 oOptype 值 db2HistData 结构

oOperation	oOptype	描述	C/COBOL/FORTRAN 定义
B	F	脱机	DB2HISTORY_OPTYPE_OFFLINE
	N	联机	DB2HISTORY_OPTYPE_ONLINE
	I	增量脱机	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	增量联机	DB2HISTORY_OPTYPE_INCR_ONLINE
	D	Delta 脱机	DB2HISTORY_OPTYPE_DELTA_OFFLINE
	E	Delta 联机	DB2HISTORY_OPTYPE_DELTA_ONLIN
F	E	日志结束	DB2HISTORY_OPTYPE_EOL
	P	时间点	DB2HISTORY_OPTYPE_PIT
L	I	插入	DB2HISTORY_OPTYPE_INSERT
	R	替换	DB2HISTORY_OPTYPE_REPLACE
Q	S	停顿共享	DB2HISTORY_OPTYPE_SHARE
	U	停顿更新	DB2HISTORY_OPTYPE_UPDATE
	X	停顿专用	DB2HISTORY_OPTYPE_EXCL
	Z	停顿复位	DB2HISTORY_OPTYPE_RESET
R	F	脱机	DB2HISTORY_OPTYPE_OFFLINE
	N	联机	DB2HISTORY_OPTYPE_ONLINE
	I	增量脱机	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	增量联机	DB2HISTORY_OPTYPE_INCR_ONLINE
T	C	添加容器	DB2HISTORY_OPTYPE_ADD_CONT
	R	再平衡	DB2HISTORY_OPTYPE_REB

db2HistoryOpenScan

以下值将被添加至 **iCallerAction** 参数。

DB2HISTORY_LIST_CRT_TABLESPACE

只选择传送其他过滤器的 CREATE TABLESPACE 和 DROP TABLESPACE 记录。

db2XaGetInfo (新 API)

db2XaGetInfo - 获取资源管理器的信息

一旦进行了 `xa_open` 调用，就会抽取特定资源管理器的信息。

权限

无

必需的连接

数据库

注意事项

`sqlxa.h`

C API 语法

```

/* File: sqlxa.h */
/* API: Get Information for Resource Manager */
/* ... */
SQL_API_RC SQL_API_FN
db2XaGetInfo (
    db2Uint32 versionNumber,
    void * pParmStruct,
    struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaGetInfoStruct
{
    db2int32 iRmid;
    struct sqlca oLastSqlca;
} db2XaGetInfoStruct;

```

API 参数

versionNumber

输入。指定作为第二个参数 `pParmStruct` 传送的结构的版本和发行版级别。

pParmStruct

输入。指向 `db2XaGetInfoStruct` 结构的指针。

pSqlca

输出。指向 `sqlca` 结构的指针。有关此结构的详情，参见 *Administrative API Reference*。

iRmid 输入。指定需要信息的资源管理器。

oLastSqlca

输出。包含上一个 XA API 调用的 `sqlca`。

注：仅可检索上一个失败的 XA API 产生的 `sqlca`。

db2XaListIndTrans (取代 `sqlxphqr` 的新 API)

db2XaListIndTrans - 列示不确定事务

为当前连接的数据库提供了所有不确定事务的列表。

作用域

此 API 仅影响在其上发出该 API 的节点。

权限

下列其中一项:

- *sysadm*
- *dbadm*

必需的连接

数据库

注意事项

db2ApiDf.h

C API 语法

```
/* File: db2ApiDf.h */
/* API: List Indoubt Transactions */
/* ... */
SQL_API_RC SQL_API_FN
db2XaListIndTrans (
    db2UInt32 versionNumber,
    void * pParmStruct,
    struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaListIndTransStruct
{
    db2XaRecoverStruct * piIndoubtData;
    db2UInt32            iIndoubtDataLen;
    db2UInt32            oNumIndoubtsReturned;
    db2UInt32            oNumIndoubtsTotal;
    db2UInt32            oReqBufferLen;
} db2XaListIndTransStruct;

typedef SQL_STRUCTURE db2XaRecoverStruct
{
    sqluint32            timestamp;
    SQLXA_XID            xid;
    char                 dbalias[SQLXA_DBNAME_SZ];
    char                 applid[SQLXA_APPLID_SZ];
    char                 sequence_no[SQLXA_SEQ_SZ];
    char                 auth_id[SQL_USERID_SZ];
    char                 log_full;
    char                 connected;
    char                 indoubt_status;
    char                 originator;
    char                 reserved[8];
} db2XaRecoverStruct;
```

API 参数

versionNumber

输入。指定作为第二个参数 *pParmStruct* 传送的结构的版本和发行版级别。

pParmStruct

输入。指向 *db2XaListIndTransStruct* 结构的指针。

pSqlca

输出。指向 *sqlca* 结构的指针。有关此结构的详情，参见 *Administrative API Reference*。

piIndoubtData

输入。指向将在其中返回不确定数据的应用程序提供的缓冲区的指针。不确定数据为 *db2XaRecoverStruct* 格式。应用程序可通过使用 *db2XaRecoverStruct* 结构的大小，并从此参数提供的地址开始，遍历不确定事务的列表。

如果值为 NULL，DB2 将计算所需缓冲区的大小，并以 *oReqBufferLen* 的形式返回此值。*oNumIndoubtsTotal* 将包含不确定事务的总数。该应用程序可分配所需的缓冲区大小并再次发出该 API。

oNumIndoubtsReturned

输出。在由 *pIndoubtData* 指定的缓冲区中返回的不确定事务记录数。

oNumIndoubtsTotal

输出。调用 API 时可用的不确定事务记录总数。如果 *piIndoubtData* 缓冲区太小，难以容纳所有记录，则 *oNumIndoubtsTotal* 将大于 *oNumIndoubtsReturned* 的总数。应用程序可重新发出 API，以获取所有记录。

注：此数目可能会视不同的 API 调用而有所不同，这是由于自动的或试探性的不确定事务再同步，或是其他事务的状态变成不确定状态而造成的。

oReqBufferLen

输出。在 API 调用时保存所有不确定事务记录所需的缓冲区长度。应用程序可通过调用 API 并将 *pIndoubtData* 设置为 NULL 来用此值确定所需的缓冲区大小。然后，可用此值来分配所需缓冲区，并且将 *pIndoubtData* 设置为所分配缓冲区的地址来发出 API。

注：所需缓冲区大小可能会视各 API 调用的不同而有所不同，这是由于自动的或试探性的不确定事务再同步，或是由于其他事务的状态变成不确定状态而造成的。应用程序可分配更大的缓冲区来达到此目的。

timestamp

输出。指定事务变成不确定状态的时间。

xid

输出。指定由事务管理器分配的用以唯一地标识全局事务的 XA 标识符。

dbalias

输出。指定不确定事务所在的数据库的别名。

applid

输出。指定数据库管理器对此事务分配的应用程序标识符。

sequence_no

输出。指定由数据库管理器分配的作为 *applid* 的扩展的序列号。

auth_id

输出。指定运行事务的用户的授权标识。

log_full

输出。指示此事务是否导致了日志已满状态。有效值为：

SQLXA_TRUE

此不确定事务导致了日志已满状态。

SQLXA_FALSE

此不确定事务未导致日志已满状态。

connected

输出。指示是否连接了应用程序。有效值为:

SQLXA_TRUE

事务正在进行正常的同步点处理，且正在等待两阶段提交的第二阶段。

SQLXA_FALSE

由于较早前发生的故障，事务处于不确定状态，现在正等待与事务管理器的再同步。

indoubt_status

输出。指示此不确定事务的状态。有效值为:

SQLXA_TS_PREP

事务已准备就绪。连接的参数可用来确定事务是否正等待正常提交处理的第二阶段，或是否发生了错误，需要与事务管理器的再同步。

SQLXA_TS_HCOM

已试探性提交事务。

SQLXA_TS_HROL

已试探性回滚事务。

SQLXA_TS_MACK

未从分区数据库中的节点进行事务提交确认。

SQLXA_TS_END

事务在此数据库中结束。稍后此事务可能被重新激活、提交或回滚。还可能是事务管理器遇到了错误，事务将不能完成。如果出现这种情况，则此事务需要试探性操作，原因是它可能会挂起锁定并阻止其他应用程序访问数据。

使用说明

在设置了与数据库或分区数据库协调节点的当前连接之后，典型的应用程序将执行下列步骤:

1. 调用 **db2XaListIndTrans**，而 *piIndoubtData* 被设置为 NULL。这将在 *oReqBufferLen* 和 *oNumIndoubtsTotal* 中返回值。
2. 使用 *oReqBufferLen* 中的返回值来分配缓冲区。此缓冲区可能不够大（如果有其他不确定事务存在的话），原因是为获取 *oReqBufferLen* 而对此 API 进行了初始调用。该应用程序可提供大于 *oReqBufferLen* 的缓冲区。
3. 确定是否获得了所有的不确定事务记录。可通过将 *oNumIndoubtsReturned* 与 *oNumIndoubtTotal* 进行比较来完成此任务。如果 *oNumIndoubtsTotal* 大于 *oNumIndoubtsReturned*，应用程序可以重复上述步骤。

另见

Administrative API Reference 中的 "sqlxhfrg - Forget Transaction Status", "sqlxphcm - Commit an Indoubt Transaction", and "sqlxphrl - Roll Back an Indoubt Transaction" 。

db2GetSnapshot – 获取快照

db2GetSnapshot API 的语法应为如下所示:

```
int db2GetSnapshot( unsigned char version;
db2GetSnapshotData *data,
struct sqlca *sqlca);
```

数据中描述的参数如下所示:

```
typedef struct db2GetSnapshotData{
    sqlma *piSqlmaData;
    sqlm_collected *poCollectedData
    void *poBuffer;
    db2uint32 iVersion;
    db2int32 iBufferSize;
    db2uint8 iStoreResult;
    db2uint16 iNodeNumber;
    db2uint32 *poOutputFormat;
}db2GetSnapshotData;
```

忘记日志记录

将把以下信息添加到 MPP Subordinator Prepare 一节后面的附录 F 中。

This log record is written after a rollback of indoubt transactions or after a commit of two-phase commit. The log record is written to mark the end of the transaction and releases any log resources held. In order for the transaction to be forgotten, it must be in a heuristically completed state.

表 21. 忘记日志记录结构

描述	类型	偏移量 (字节数)
日志头	LogManagerLogRecordHeader	0(20)
时间	sqluint64	20(8)
总长度: 28 个字节		

sqlaintp – 获取错误消息

以下使用说明将被添加至此 API 的描述:

In a multi-threaded application, sqlaintp must be attached to a valid context; otherwise, the message text for SQLCODE -1445 cannot be obtained.

sqlbctcq – 关闭表空间容器查询

对于此 API, 装入并非有效的“权限”级别。

sqlubkp – 备份数据库

对于 **BackupType** 参数, SQLUB_FULL 值将被替换为 SQLUB_DB。将会对数据库中的所有表空间进行备份。

db2ArchiveLog - 归档活动日志

要支持新的增量备份功能，还应添加 `SQLUB_INCREMENTAL` 和 `SQLUB_DELTA` 参数。增量备份映象是自上次成功地进行完全备份之后发生了更改的所有数据库数据的副本。`delta` 备份映象是自上次成功地进行任何类型的备份之后发生更改的所有数据库数据的副本。

sqlureot - 重组表

以下语句将添加至 Usage Notes:

```
REORGANIZE TABLE cannot use an index that is based on an index extension.
```

sqlurestore - 恢复数据库

对于 **RestoreType** 参数，`SQLUD_FULL` 值将被替换为 `SQLUD_DB`。将会对数据库中的所有表空间进行恢复。这将会脱机运行。

要支持新的增量恢复功能，还应添加 `SQLUD_INCREMENTAL` 参数。

增量备份映象是自上次成功地进行完全备份之后发生了更改的所有数据库数据的副本。

关于“AIX 扩展共享内存支持”(EXTSHM) 的文档错误

在 "Appendix E. Threaded Applications with Concurrent Access" 中，注释 2 现在应为:

2. By default, AIX does not permit 32-bit applications to attach to more than 11 shared memory segments per process, of which a maximum of 10 can be used for local DB2 connections.

要将 EXTSHM 与 DB2 配合使用，执行下列操作:

在客户机会话中:

```
export EXTSHM=ON
```

当启动 DB2 时:

```
export EXTSHM=ON
db2set DB2ENVLIST=EXTSHM
db2start
```

在 EEE 上，还要将下列几行添加至 `sqllib/db2profile`:

```
EXTSHM=ON
export EXTSHM
```

SQLFUPD

锁定列表

令牌名称已从 `SQLF_DBTN_LOCKLIST` 更改为 `SQLF_DBTN_LOCK_LIST`。锁定列表参数已从 `SMALLINT` 更改为 64 位不带符号的 `INTEGER`。还应对“可更新数据库配置参数”表添加以下附加项。

参数名	令牌	令牌值	数据类型
锁定列表	SQLF_DBTN_LOCK_LIS	704	UInt64

此参数的新的最大值为 524 288。

SQLLEDBDESC

将会对 SQLDBCSS（在 sqlenv 中作了定义）的有效值列表添加两个值。它们是：

SQL_CS_SYSTEM_NLSCHAR

对字符类型使用 NLS 版本的比较例程的系统的整理顺序。

SQL_CS_USER_NLSCHAR

对字符类型使用 NLS 版本的比较例程的用户的整理顺序。

SQLFUPD 文档错误

在 "Chapter 3. Data Structures" 的 "Table 53. Updatable Database Configuration Parameters" 中，dbheap 的令牌值被错误地列示为 701。正确的值为 58。

应用程序构建指南

第 1 章 介绍

受支持软件

注: PHP。 现在, 可以使用 PHP 来通过基于 Web 的应用程序访问 DB2。PHP 是一种服务器端的、嵌入 HTML、跨平台的脚本语言。它支持使用统一 ODBC 访问方法进行的 DB2 访问, 在该方法中, 用户级别的 PHP 使用 ODBC 调用来与 DB2 通信。与标准 ODBC 不同, 借助统一 ODBC 方法可以直接与 DB2 CLI 层通信, 而不需要通过 ODBC 层。有关将 PHP 与 DB2 配合使用的详情, 搜索 DB2 支持站点, 网址为: www.ibm.com/software/data/db2/udb/winunix/support。

AIX

C 和 C++ 编译器的列示版本应为如下所示:

IBM C and C++ Compilers for AIX Version 3.6.6 (对于 64 位, 为版本 3.6.6.3)
IBM C for AIX 4.4
IBM VisualAge C++ Version 4.0

注: 请从以下网址下载这些编译器的最新可用修订包:

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

列示的 Micro Focus COBOL 编译器版本应为如下所示:

AIX 4.2.1

Micro Focus COBOL Version 4.0.20 (PRN 12.03 或更新版本)
Micro Focus COBOL Version 4.1.10 (PRN 13.04 或更新版本)

AIX 4.3

Micro Focus COBOL Server Express Version 1.0

注: 有关 AIX 4.3 上的 Micro Focus COBOL 存储过程和 UDF 的 DB2 支持的信息, 参见“DB2 应用程序开发” Web 页面:

<http://www.ibm.com/software/data/db2/udb/ad>

要使用“IBM XL Fortran AIX 版的版本 5.1.0”编译器构建 64 位应用程序, 在编译和链接步骤中使用“-q64”选项。注意, 64 位应用程序在此编译器的较早版本上不受支持。

HP-UX

列示的 C++ 编译器版本应为如下所示:

HP aC++, Version A.03.25

注: HP 不支持使用旧的和新的编译器编译的对象间的二进制兼容性, 所以, 这将强制重新编译为在 HP-UX 上访问 DB2 而构建的任何 C++ 应用程序。还必须构建 C++ 应用程序以使用此新编译器处理异常。

这是 aCC 转换指南的 URL: <http://www.hp.com/esy/lang/cpp/tguide>。C++ 不兼容部分位于:

<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.1.2>
<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.3.3>

C 与 C++ 部分位于:

<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.3.3.1>

即使 C 和 aCC 是兼容的, 在使用两个不同的对象类型时, 包含 "main" 的对象必须用 aCC 进行编译, 而最终的可执行文件必须与 aCC 链接。

Linux

DB2 Linux 版支持以下 REXX 版本:

Object REXX Interpreter for Linux Version 2.1

Linux/390

"DB2 Linux/390 版" 仅支持 Java、C 和 C++。

OS/2

列示的 C/C++ 编译器版本应为如下所示:

IBM VisualAge C++ for OS/2 Version 3.6.5 and Version 4.0

注: 请从以下网址下载这些编译器的最新可用修订包:

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

有关对这些 VisualAge C++ 编译器的将来服务支持的限制, 请参见位于以下站点的消息部分:

<http://www-4.ibm.com/software/ad/vacpp/>

Solaris

列示的 Micro Focus COBOL 编译器版本应为:

Micro Focus COBOL Server Express Version 1.0

Windows 32 位操作系统

列示的 IBM VisualAge C++ 编译器版本应为如下所示:

IBM VisualAge C++ for Windows Versions 3.6.5 and 4.0

注: 请从以下网址下载这些编译器的最新可用修订包:

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

有关对这些 VisualAge C++ 编译器的将来服务支持的限制, 请参见位于以下站点的消息部分:

<http://www-4.ibm.com/software/ad/vacpp/>

列示的 Micro Focus COBOL 编译器版本应为如下所示:

样本程序

以下内容应添加至『对象链接和嵌入样本』一节:

salarycltvc

一个 Visual C++ DB2 CLI 样本, 它调用 Visual Basic 存储过程 salarysrv。

SALSVADO

一个样本 OLE 自动存储过程 (SALSVADO) 和一个 SALCLADO 客户机 (SALCLADO), 是用 32 位 Visual Basic 和 ADO 实现的, 用来计算表 staff2 中的平均薪水。

以下内容应添加至『日志管理用户出口样本』一节:

AIX 上使用级别 3.1.6 和更高级别的 ADSM API 客户机的应用程序必须使用 xlc_r 或 xlC_r 而不是 xlc 或 xlC 编译器调用来构建, 即使这些应用程序是单线程的。这就确保了库线程是安全的。这适用于“日志管理用户出口样本”db2uext2.cadsm。

如果应用程序是用非线性安全库编译的, 可应用 fixtest IC21925E 或与应用程序供应商联系。fixtest 是在 index.storsys.ibm.com 匿名 ftp 服务器上提供的。这会使 ADSM API 级别回复为 3.1.3。

第 3 章 构建 DB2 应用程序的一般信息

构建文件、Makefile 和错误检查实用程序

表 16 中的 bldevm 项应为:

bldevm

事件监控器样本程序 evm (仅在 AIX、OS/2 和 Windows 32 位操作系统上可用)。

表 17 应包括下列项:

bldmevm

事件监控器样本程序 evm 及 Microsoft Visual C++ 编译器。

bldvevm

事件监控器样本程序 evm 及 VisualAge C++ 编译器。

第 4 章 构建 Java 小应用程序和应用程序

设置环境

如果正在受支持的平台上使用 IBM JDK 1.1.8 来构建 SQLJ 程序, 则 JDK 构建日期必须是 1999 年 11 月 24 日 (或之后)。否则, 可能会在编译期间出现 JNI panic 错误。

如果正在受支持的平台上使用 IBM JDK 1.2.2 来构建 SQLJ 程序, 则 JDK 构建日期必须是 2000 年 4 月 17 日 (或之后)。否则, 可能会在编译期间出现 Java 类型无效错误。

对于 **AIX**、**HP-UX**、**Linux** 和 **Solaris** 这些小节，将有关 JDBC 2.0 的信息替换为下列内容：

将 JDBC 2.0 驱动程序与 Java 应用程序配合使用

JDBC 1.22 驱动程序仍为所有操作系统上的缺省驱动程序。要利用 JDBC 2.0 的新功能部件，必须安装 JDK 1.2 支持。在执行利用 JDBC 2.0 新功能部件的应用程序之前，必须通过从 `sqllib/java12` 目录发出 `usejdbc2` 命令来设置环境。如果想要应用程序始终使用 JDBC 2.0 驱动程序，考虑将以下行添加至登录简要表（如 `.profile`）或 shell 初始化脚本（如 `.bashrc`、`.cshrc` 或 `.kshrc`）：

```
. sqllib/java12/usejdbc2
```

确保此命令放在要运行 `db2profile` 的命令后面，因为 `usejdbc2` 应在 `db2profile` 之后运行。

要切换回 JDBC 1.22 驱动程序，从 `sqllib/java12` 目录执行以下命令：

```
. usejdbc1
```

将 JDBC 2.0 驱动程序与 Java 存储过程和 UDF 配合使用

要将 JDBC 2.0 驱动程序与 Java 存储过程和 UDF 配合使用，必须对实例的受防护用户标识设置环境。缺省受防护用户标识为 `db2fenc1`。要对受防护用户标识设置环境，执行下列步骤：

1. 将以下行添加至受防护用户标识简要表（如 `.profile`）或受防护用户标识 shell 初始化脚本（如 `.bashrc`、`.cshrc` 或 `.kshrc`）：

```
. sqllib/java12/usejdbc2
```

2. 从 CLP 发出以下命令：

```
db2set DB2_USE_JDK12=1
```

要切换回 Java UDF 和存储过程的 JDBC 1.22 驱动程序支持，执行下列步骤：

1. 从受防护用户标识简要表（如 `.profile`）或受防护用户标识 shell 初始化脚本（如 `.bashrc`、`.cshrc` 或 `.kshrc`）除去以下行：

```
. sqllib/java12/usejdbc2
```

2. 从 CLP 发出以下命令：

```
db2set DB2_USE_JDK12=
```

如果想要应用程序始终使用 JDBC 2.0 驱动程序，可将以下行添加至登录简要表（如 `.profile`）或 shell 初始化脚本（如 `.bashrc`、`.cshrc` 或 `.kshrc`）：

```
. sqllib/java12/usejdbc2
```

确保此命令放在要运行 `db2profile` 的命令后面，因为 `usejdbc2` 应在 `db2profile` 之后运行。

HP-UX

Java 存储过程和用户定义函数在具有 JDK 1.1 的“DB2 HP-UX 版”上不受支持。

Silicon Graphics IRIX

在使用 -o32 对象类型构建 SQLJ 应用程序，且将 Java JIT 编译器与 JDK 1.2.2 配合使用时，如果 SQLJ 译码器由于分段错误而发生故障，则尝试用以下命令关闭 JIT 编译器：

```
export JAVA_COMPILER=NONE
```

JDK 1.2.2 是在 Silicon Graphics IRIX 上构建 Java SQLJ 程序所必需的。

Windows 32 位操作系统

将 JDBC 2.0 驱动程序与 Java 存储过程和 UDF 配合使用

要将 JDBC 2.0 驱动程序与 Java 存储过程和 UDF 配合使用，必须通过执行下列步骤来设置环境：

1. 在 sqllib\java12 目录中发出以下命令：

```
usejdbc2
```

2. 从 CLP 发出以下命令：

```
db2set DB2_USE_JDK12=1
```

要切换回 Java UDF 和存储过程的 JDBC 1.22 驱动程序支持，执行下列步骤：

1. 在 sqllib\java12 目录中发出以下命令：

```
usejdbc2
```

2. 从 CLP 发出以下命令：

```
db2set DB2_USE_JDK12=
```

OS/2 上的 JDK 级别

当运行 1999 年 9 月之前发行的 JDK 1.1.8 版本时，某些信息将不会在 OS/2 上显示。确保您具有最新的 JDK 版本 1.1.8。

HP-UX 上的 Java2

要运行 Java2 存储过程，必须将共享库路径更改为如下所示：

```
export SHLIB_PATH=$JAVADIR/jre/lib/PA_RISC:$JAVADIR/  
jre/lib/PA_RISC/classic:$HOME/sqllib/lib:/usr/lib:$SHLIB_PATH
```

\$JAVADIR 是 Java2 SDK 所在的位置。

第 5 章 构建 SQL 过程

设置 SQL 过程环境

除“安装”中用于设置 DB2 环境的那些指导之外，还有下列这些指导。

要获取 SQL 过程支持，必须在服务器上安装“应用程序开发客户机”。有关安装“应用程序开发客户机”的信息，参考用于您所使用的平台的《快速入门》一书。对于平台上受 DB2 支持的 C 和 C++ 编译器，参见“受平台支持的软件”。

注：在 OS/2 FAT 文件系统上，您只能对“SQL 过程”使用不超过 8 个字符的模式名。您必须对超过 8 个字符的模式名使用 HPFS 文件系统。

编译器配置由两部分组成：设置编译器的环境变量和定义编译命令。环境变量提供了指向编译器的二进制文件、库和包含文件的路径。编译命令是 DB2 将用来编译为 SQL 过程生成的 C 文件的完整命令。

设置编译器环境变量

对于在 OS/2、Windows 和基于 UNIX 的操作系统上配置该环境，有一些不同的规则，如下所述。在某些情况下，不需要任何配置；而在另一些情况下，必须将 DB2_SQLROUTINE_COMPILER_PATH DB2 注册表变量设置为指向用来适当设置环境变量的可执行脚本。

注：可以使用 **db2set** 命令或者从“存储过程构建器”使用“SQL 存储过程构建选项”对话框来设置此 DB2 注册表变量的值。通过使用“SQL 存储过程构建选项”对话框，就不再需要物理地访问数据库服务器，也不再需要重新启动数据库服务器就可以使更改生效。

在 OS/2 上：

对于 IBM VisualAge C++ OS/2 版的版本 3.6:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcxxo\bin\setenv.cmd"
```

对于 IBM VisualAge C++ OS/2 版的版本 4:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcpp40\bin\setenv.cmd"
```

注：对于这些命令，假定已经在 C: 驱动器上安装了 C++ 编译器。如果必要的话，切换驱动器或路径，以反映系统上 C++ 编译器的位置。

在 Windows 32 位操作系统上，如果编译器的环境变量被设置为 SYSTEM 变量，则不需要进行配置。否则，按如下所示来设置 DB2_SQLROUTINE_COMPILER_PATH DB2 注册表变量：

对于 Microsoft Visual C++ 版本 5.0:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\devstudio\vc\bin\vcvars32.bat"
```

对于 Microsoft Visual C++ 版本 6.0:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\Micros~1\vc98\bin\vcvars32.bat"
```

对于 IBM VisualAge C++ Windows 版的版本 3.6:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcxxw\bin\setenv.bat"
```

对于 IBM VisualAge C++ Windows 版的版本 4:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcppw40\bin\setenv.bat"
```

注：对于这些命令，假定已经在 C: 驱动器上安装了 C++ 编译器。如果必要的话，切换驱动器或路径，以反映系统上 C++ 编译器的位置。

在基于 UNIX 的操作系统上，DB2 将在您第一次编译存储过程时生成一个可执行脚本文件 \$HOME/sql/lib/function/routine/sr_cpath（它包含编译器环境变量的缺省值）。如果缺省值不适用于编译器，可对此文件进行编辑。或者，可将 DB2_SQLROUTINE_COMPILER_PATH DB2 注册表变量设置为包含用来指定预期设置（参见上述示例）的另一可执行脚本的全路径名。

定制编译命令

“应用程序开发客户机”的安装提供了一个缺省编译命令，它可用于每个平台上受支持的编译器中的至少一个：

```
AIX: IBM C Set++ for AIX Version 3.6.6
Solaris: SPARCCompiler C++ Versions 4.2 and 5.0
HP-UX: HP-UX C++ Version A.12.00
Linux: GNU/Linux g++ Version egcs-2.90.27 980315 (egcs-1.0.2 release)
PTX: ptx/C++ Version 5.2
OS/2: IBM VisualAge C++ for OS/2 Version 3
Windows NT and Windows 2000: Microsoft Visual C++ Versions 5.0 and 6.0
```

要使用其他编译器，或定制缺省命令，必须使用如下命令来设置 `DB2_SQLROUTINE_COMPILE_COMMAND` DB2 注册表变量：

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=compilation_command
```

其中 `compilation_command` 是 C 或 C++ 编译命令，包括创建存储过程所需的选项和参数。

在编译命令中，使用关键字 `SQLROUTINE_FILENAME` 来替换生成的 `SQC`、`C`、`PDB`、`DEF`、`EXP`、消息日志和共享库文件的文件名。仅对 AIX 使用关键字 `SQLROUTINE_ENTRY` 来替换入口点名。

注：可以使用 `db2set` 命令或者从“存储过程构建器”中使用“SQL 存储过程构建选项”对话框来设置此 DB2 注册表变量的值。通过使用“SQL 存储过程构建选项”对话框，就不再需要物理地访问数据库服务器，也不再需要重新启动数据库服务器就可以使更改生效。

以下是在受支持服务器平台上 C 或 C++ 编译器的 `DB2_SQLROUTINE_COMPILE_COMMAND` 的缺省值。

AIX

要使用 IBM C AIX 版的版本 3.6.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=xlc -H512 -T512 \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c -bE:SQLROUTINE_FILENAME.exp \
-e SQLROUTINE_ENTRY -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -lc -ldb2
```

要使用 IBM C Set++ AIX 版的版本 3.6.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=xlc -H512 -T512 \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c -bE:SQLROUTINE_FILENAME.exp \
-e SQLROUTINE_ENTRY -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -lc -ldb2
```

如果未设置 `DB2_SQLROUTINE_COMPILE_COMMAND` DB2 注册表变量，则这是缺省编译命令。

注：要在 AIX 上编译 64 位 SQL 过程，将 `-q64` 选项添加至上述命令。

要使用 IBM VisualAge C++ AIX 版的版本 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

如果在 `vacbld` 命令后面未指定配置文件，DB2 将在首次尝试创建任何 SQL 过程时创建以下缺省配置文件：

```
$HOME/sql1lib/function/routine/sqlproc.icc
```

如果想要使用您自己的配置文件，可在对 DB2_SQLROUTINE_COMPILE_COMMAND 设置 DB2 注册表值时指定您自己的配置文件：

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld
%DB2PATH%/function/sqlproc.icc"
```

HP-UX

要使用 “HP C 编译器版本 A.11.00.03” ：

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc +DAportable +ul -Aa +z \
-I$HOME/sql1lib/include -c SQLROUTINE_FILENAME.c; \
ld -b -o SQLROUTINE_FILENAME SQLROUTINE_FILENAME.o \
-L$HOME/sql1lib/lib -ldb2
```

要使用 HP-UX C++ 版本 A.12.00:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=CC +DAportable +a1 +z -ext \
-I$HOME/sql1lib/include -c SQLROUTINE_FILENAME.c; \
ld -b -o SQLROUTINE_FILENAME SQLROUTINE_FILENAME.o \
-L$HOME/sql1lib/lib -ldb2
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量，则这是缺省编译命令。

Linux

要使用 GNU/Linux gcc 版本 2.7.2.3:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \
-shared -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -ldb2
```

要使用 GNU/Linux g++ 版本 egcs-2.90.27 980315 (egcs-1.0.2 发行版)：

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=g++ \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \
-shared -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -ldb2
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量，则这是缺省编译命令。

PTX

要使用 ptx/C 版本 4.5:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc -KPIC \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \
-G -o SQLROUTINE_FILENAME.so -L$HOME/sql1lib/lib -ldb2 ; \
cp SQLROUTINE_FILENAME.so SQLROUTINE_FILENAME
```

要使用 ptx/C++ 版本 5.2:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=c++ -KPIC \
-D_RWSTD_COMPILE_INSTANTIATE=0 \
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \
-G -o SQLROUTINE_FILENAME.so -L$HOME/sql1lib/lib -ldb2 ; \
cp SQLROUTINE_FILENAME.so SQLROUTINE_FILENAME
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量，则这是缺省编译命令。

OS/2

要使用 IBM VisualAge C++ OS/2 版的版本 3:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="icc -Ge- -Gm+ -W2  
-I%DB2PATH%\include SQLROUTINE_FILENAME.c  
/B\"/NOFREE /NOI /ST:64000\" SQLROUTINE_FILENAME.def  
%DB2PATH%\lib\db2api.lib"
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量, 则这是缺省编译命令。

要使用 IBM VisualAge C++ OS/2 版的版本 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

如果在 vacbld 命令后面未指定配置文件, DB2 将在首次尝试创建任何 SQL 过程时创建以下缺省配置文件:

```
%DB2PATH%\function\routine\sqlproc.icc
```

如果想要使用您自己的配置文件, 可在对 DB2_SQLROUTINE_COMPILE_COMMAND 设置 DB2 注册表值时指定您自己的配置文件:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld  
%DB2PATH%\function\sqlproc.icc"
```

Solaris

要使用 SPARCCompiler C 版本 4.2 和 5.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc -xarch=v8plusa -Kpic \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib \  
-R$HOME/sql1lib/lib -ldb2
```

要使用 SPARCCompiler C++ 版本 4.2 和 5.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=CC -xarch=v8plusa -Kpic \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib \  
-R$HOME/sql1lib/lib -ldb2
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量, 则这是缺省编译命令。

注:

1. 编译器选项 -xarch=v8plusa 已被添加至缺省编译器命令。有关添加此选项的原因的详细信息, 参见第303页的『第 12 章 构建 Solaris 应用程序』。
2. 要在 Solaris 上编译 64 位 SQL 过程, 取消 -xarch=v8plusa 选项, 而将 -xarch=v9 选项添加至上述命令。

Windows NT 和 Windows 2000

注: SQL 过程在 Windows 98 或 Windows 95 上不受支持。

要使用 Microsoft Visual C++ 版本 5.0 和 6.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=c1 -Od -W2 /TC -D_X86_=1  
-I%DB2PATH%\include SQLROUTINE_FILENAME.c /link -dll  
-def:SQLROUTINE_FILENAME.def /out:SQLROUTINE_FILENAME.dll  
%DB2PATH%\lib\db2api.lib
```

如果未设置 DB2_SQLROUTINE_COMPILE_COMMAND DB2 注册表变量, 则这是缺省编译命令。

要使用 IBM VisualAge C++ Windows 版的版本 3.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="ilib /GI
SQLROUTINE_FILENAME.def &icc -Ti -Ge- -Gm+ -W2 -I%DB2PATH%\include SQLROUTINE_FILENAME.c
/B\"/ST:64000 /PM:VIO /DLL\" SQLROUTINE_FILENAME.exp
%DB2PATH%\lib\db2api.lib"
```

要使用 IBM VisualAge C++ Windows 版的版本 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

如果在 vacbld 命令后面未指定配置文件, DB2 将在首次尝试创建任何 SQL 过程时创建以下缺省配置文件:

```
%DB2PATH%\function\routine\sqlproc.icc
```

如果想要使用您自己的配置文件, 可在对 DB2_SQLROUTINE_COMPILE_COMMAND 设置 DB2 注册表值时指定您自己的配置文件:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld
%DB2PATH%\function\sqlproc.icc"
```

要返回至缺省编译器选项, 使用以下命令来将

DB2_SQLROUTINE_COMPILE_COMMAND 的 DB2 注册表值设置为空:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=
```

保留中间文件

您必须人工删除在创建 SQL 过程失败时可能留下来的中间文件。这些文件在下列目录中:

UNIX \$DB2PATH/function/routine/sqlproc/\$DATABASE/\$SCHEMA/tmp

其中, \$DB2PATH 表示在其中创建了实例的目录, \$DATABASE 表示数据库名, 而 \$SCHEMA 表示以其创建 SQL 过程的模式名。

OS/2 和 Windows

```
%DB2PATH%\function\routine\sqlproc\%DATABASE%\%SCHEMA%\tmp
```

其中 %DB2PATH% 表示创建实例的目录, %DATABASE% 表示数据库名, %SCHEMA% 表示用其创建 SQL 过程的模式名。

备份与恢复

如果创建了 SQL 过程, 生成的共享库/DLL 还会保存在目录表中 (如果生成的共享库/DLL 小于 2 MB 的话)。如果备份并恢复了数据库, 则生成的共享库/DLL 小于 2 MB 的任何 SQL 过程都将使用目录表中保存的版本进行备份并恢复。如果 SQL 过程带有的生成共享库/DLL 大于 2 MB, 则确保您还使用数据库备份与恢复对文件系统进行了备份与恢复。如果没有的话, 您将必须使用 syscat.procedures 目录表中的源来人工重新创建 SQL 过程的共享库/DLL。

注: 在数据库恢复期, 属于正在恢复的数据库的文件系统上的所有 SQL 过程可执行文件将被除去。如果索引创建配置参数 (indexrec) 被设置为 RESTART, 则将从目录

表中抽取所有的 SQL 过程可执行文件，并在下一次连接时放回到文件系统上。否则，将会在第一次执行 SQL 过程时抽取 SQL 可执行文件。可执行文件将会放回到以下目录中：

UNIX \$DB2PATH/function/routine/sqlproc/\$DATABASE

其中，\$DB2PATH 表示在其中创建了实例的目录，而 \$DATABASE 表示以其创建 SQL 过程的数据库名。

OS/2 和 Windows

%DB2PATH%\function\routine\sqlproc%\DATABASE%

其中 %DB2PATH% 表示创建实例的目录，而 %DATABASE% 表示用来创建 SQL 过程的数据库名。

创建 SQL 过程

将数据库管理器配置参数 KEEPDI 设置为 'NO' 以开发 SQL 过程。如果每当执行 SQL 过程时都会装入该过程，则可能会在删除存储过程并以同一名称重新创建该存储过程时出现问题，原因是不能刷新库且不能从文件系统删除可执行文件。还将在尝试回滚更改或删除数据库时出现问题，原因是不能删除可执行文件。

参见《应用程序构建指南》的第 2 章『设置』中的“更新数据库管理器配置文件”，以了解有关设置 KEEPDI 参数的详情。

注：SQL 过程不支持下列参数数据类型：

- LONG VARGRAPHIC
- 二进制大对象 (BLOB)
- 字符大对象 (CLOB)
- 双字节字符大对象 (DBCLOB)

调用存储过程

“使用 CALL 命令”中的第一段应为：

要使用调用命令，必须输入存储过程名和所有 IN 或 INOUT 参数，并将 '?' 作为每个 OUT 参数的占位符。有关 CALL 命令语法的详细信息，参见第123页的『CALL』。

分布编译型 SQL 过程

注：要在 DB2 服务器之间分布编译型 SQL 过程，必须对充当编译型 SQL 过程的源或目的地的每个 DB2 服务器执行下列步骤：

步骤 1. 安装修订包 3

步骤 2. 发出 **db2updv7** 命令来使 DB2 能够抽取和安装编译型 SQL 过程：

```
db2updv7 -d database_name
```

定义 SQL 过程时，它被转换为 C 程序、预编译、与目标数据库绑定、编译并链接以创建共享库。编译和链接步骤需要 C 或 C++ 编译器在数据库服务器上可用。但是，一旦定义了 SQL 过程，就可以将它以编译的表单形式分布给在同一平台上运行的 DB2 数据库，而不需要访问 C 或 C++ 编译器。DB2 允许用户从一个数据库以编译的表单形式抽取 SQL 过程，并将它们以编译的表单形式安装到另一个数据库中。

DB2 提供了命令行接口和编程接口来执行抽取和安装操作。命令行接口由两个 CLP 命令 GET ROUTINE 和 PUT ROUTINE 组成。编程接口由两个内置存储过程 GET_ROUTINE_SAR 和 PUT_ROUTINE_SAR 组成。有关命令行接口的详情，参考 Command Reference。有关编程接口的详情，参考 SQL Reference。

要将编译型 SQL 过程从一个数据库服务器分布给另一个数据库服务器，应执行下列步骤：

- 步骤 1. 开发应用程序，包括定义作为应用程序一部分的 SQL 过程。
- 步骤 2. 在测试这些过程之后，将每个过程的编译的版本抽取到另一个文件中。有关详情，参考 Command Reference 中的 GET ROUTINE 命令或者 SQL Reference 中的 GET_ROUTINE_SAR 存储过程。
- 步骤 3. 通过发出 PUT ROUTINE 命令或通过调用 PUT_ROUTINE_SAR 存储过程，并使用由 2 创建的文件，来在每个服务器上安装每个过程的编译的版本。每个数据库服务器必须具有相同的操作系统和 DB2 级别。

第 7 章 构建 HP-UX 应用程序

HP-UX C

在“多线程应用程序”中，已经用不同的编译选项修正了 bldmt 脚本文件。新版本在 sqllib/samples/c 目录中。

HP-UX C++

在构建脚本中，对于 HP aC++ 编译器，C++ 编译器变量 CC 已替换为 aCC。修正过的构建脚本在 sqllib/samples/cpp 目录中。

“+u1”编译选项将用来与 aCC 编译器配合使用构建存储过程和 UDF。此选项允许未标记的数据访问。未用此选项更新随 DB2 HP-UX 版交付的样本构建脚本 bldsrv 和 bldudf 以及样本 makefile。将对它们进行修正，以在使用之前添加此选项。以下是 bldsrv 和 bldudf 脚本的新编译步骤：

```
aCC +DAportable +u1 -Aa +z -ext -I$DB2PATH/include -c $1.C
```

在“多线程应用程序”中，已经用不同的编译选项修正了 bldmt 脚本文件。新版本在 sqllib/samples/cpp 目录中。

第 9 章 构建 OS/2 应用程序

VisualAge C++ OS/2 版的版本 4.0

对于 OS/2 和 Windows，使用本节中所描述的 set 命令而不是 export 命令。例如，set CLI=tbinfo。

在『DB2 CLI 应用程序』的『构建和运行嵌入式 SQL 应用程序』小节中，对于 OS/2 和 Windows，必须使用 cliapi.icc 文件而不是 cli.icc 文件，原因是嵌入式 SQL 应用程序需要 cliapi.icc 链接的 db2api.lib 库。

第 10 章 构建 PTX 应用程序

ptx/C++

库需要与 `-shared` 选项链接，以构建存储过程和用户定义函数。在 `sqllib/samples` 目录中，`makefile`、构建脚本 `blsrv` 和 `bludf` 已更新为包括此选项，如 `blsrv` 中的下列链接步骤中所示：

```
c++ -shared -G -o $1 $1.o -L$DB2PATH/lib -ldb2
```

第 12 章 构建 Solaris 应用程序

SPARCCompiler C++

在 Solaris 上执行“C/C++ 应用程序”和运行“SQL 过程”的问题

在使用 Sun WorkShop Compiler C/C++ 时，如果可执行文件中遇到问题，接收到类似如下的错误：

1. 在行 1 处有语法错误：意外地出现了 ‘(’
2. ksh: <应用程序名>: 不能执行（其中应用程序名是编译过的可执行文件名）

则您遇到的问题可能是，编译器在与 `libdb2.so` 进行链接时不生成有效的可执行文件。修正此问题的一个建议是将以下编辑器选项添加至编辑和链接命令：

```
-xarch=v8plusa
```

例如，在编辑样本应用程序 `dynamic.sqc` 时：

```
embprep dynamic sample
embprep utilemb sample
cc -c utilemb.c -xarch=v8plusa -I/export/home/db2inst1/sqllib/include
cc -o dynamic dynamic.c utilemb.o -xarch=v8plusa -I/export/home/db2inst1/sqllib/include \
-L/export/home/db2inst1/sqllib/lib -R/export/home/db2inst1/sqllib/lib -l db2
```

注：

1. 如果在 Solaris 上使用“SQL 过程”，且您通过 `DB2_SQLROUTINE_COMPILE_COMMAND` 简要表变量来使用您自己的编译字符串，则请确保您包括有上面给定的编译器选项。缺省编辑器命令包括此选项：

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="cc -# -Kpic -xarch=v8plusa -I$HOME/sqllib/include \
SQLROUTINE_FILENAME.c -G -o SQLROUTINE_FILENAME -L$HOME/sqllib/lib -R$HOME/sqllib/lib -l
```

2. 要在 Solaris 上编译 64 位 SQL 过程，取消 `-xarch=v8plusa` 选项，而将 `-xarch=v9` 选项添加至上述命令。

第 13 章 为 Windows 32 位操作系统构建应用程序

VisualAge C++ 版本 4.0

对于 OS/2 和 Windows，使用本节中所描述的 `set` 命令而不是 `export` 命令。例如，`set CLI=tbinfo`。

在『DB2 CLI 应用程序』的『构建和运行嵌入式 SQL 应用程序』小节中，对于 OS/2 和 Windows，必须使用 `cliapi.icc` 文件而不是 `cli.icc` 文件，原因是嵌入式 SQL 应用程序需要 `cliapi.icc` 链接的 `db2api.lib` 库。

Application Development Guide

Chapter 2. Coding a DB2 Application

对 Microsoft Visual C++ 激活 IBM DB2 通用数据库项目和工具加载件

在运行 **db2vccmd** 命令（步骤 1）之前，请确保您已经使用当前登录标识启动并停止了 Visual C++ 至少一次。第一次运行 Visual C++ 时，会为用户标识创建一个简要表，这就是由 **db2vccmd** 命令更新的内容。如果未立即启动 Visual C++，且试图运行 **db2vccmd**，则可能会见到类似如下的错误：

```
"Registering DB2 Project add-in ...Failed! (rc = 2)"
```

Chapter 6. Common DB2 Application Techniques

生成序列值

生成序列值是一个常见的数据库应用程序开发问题。该问题的最佳解决方案是在 SQL 中使用序列对象和序列表达式。每个序列对象是具有唯一命名的数据库对象，只能通过序列表达式才能访问它。有两个序列表达式：PREVVAL 表达式和 NEXTVAL 表达式。PREVVAL 表达式将返回先前语句的指定序列的最新生成值。NEXTVAL 序列表达式增大了序列对象的值，并返回序列对象的新值。

要创建序列对象，发出 CREATE SEQUENCE 语句。例如，要使用缺省属性创建一个称为 id_values 的序列对象，可发出以下语句：

```
CREATE SEQUENCE id_values
```

要显示序列对象的当前值，使用 PREVVAL 表达式发出 VALUES 语句：

```
VALUES PREVVAL FOR id_values
```

```
1
-----
1
```

选择了 1 个记录。

您可以反复检索序列对象的当前值，并且在您发出 NEXTVAL 表达式之前该序列对象返回的值不会更改。在以下示例中，PREVVAL 表达式将返回值 1，直到 NEXTVAL 表达式增大了序列对象的值为止：

```
VALUES PREVVAL FOR id_values
```

```
1
-----
1
```

```
1 record(s) selected.
VALUES PREVVAL FOR id_values
```

```
1
-----
```

```

1
1 record(s) selected.
VALUES NEXTVAL FOR id_values

1
-----
                2
1 record(s) selected.
VALUES PREVVAl FOR id_values

1
-----
                2
1 record(s) selected.

```

要用序列对象的下一个值来更新一列的值，应在 UPDATE 语句中包括 NEXTVAL 表达式，如下所示：

```

UPDATE staff
   SET id = NEXTVAL FOR id_values
   WHERE id = 350

```

要用序列对象的下一个值向表中插入新行，应在 INSERT 语句中包括 NEXTVAL 表达式，如下所示：

```

INSERT INTO staff (id, name, dept, job)
   VALUES (NEXTVAL FOR id_values, 'Kandil', 51, 'Mgr')

```

有关 PREVVAl 和 NEXTVAL 表达式的详情，参考 *SQL Reference*。

控制序列行为

可以调整序列对象的行为以满足您的应用程序的需要。当您发出 CREATE SEQUENCE 语句来创建新的序列对象，以及当您为现存序列对象发出 ALTER SEQUENCE 语句时，您就更改了序列对象的属性。以下是您可以指定的一些序列对象属性：

数据类型

CREATE SEQUENCE 语句的 AS 子句指定序列对象的数字数据类型。如 *SQL Reference* 的“SQL 限制”附录中所指定的那样，数据类型将确定序列对象的最小可能值和最大可能值。不能更改序列对象的数据类型；相反，必须以新数据类型通过发出 DROP SEQUENCE 语句及发出 CREATE SEQUENCE 语句来删除序列对象。

起始值 CREATE SEQUENCE 语句的 START WITH 子句设置序列对象的初始值。ALTER SEQUENCE 语句的 RESTART WITH 子句将序列对象的值复位为指定值。

最小值 MINVALUE 子句设置序列对象的最小值。

最大值 MAXVALUE 子句设置序列对象的最大值。

增量值 INCREMENT BY 子句设置每个 NEXTVAL 表达式添加至序列对象的值。要减小序列对象的值，应指定负值。

序列循环

CYCLE 子句使达到其最大值或最小值的序列对象值在下面的 NEXTVAL 表达式中返回到其起始值。

例如，要创建一个称为 `id_values` 的序列对象，其起始值为 0、最大值为 1000、每个 `NEXTVAL` 表达式的增量为 2，并且当达到最大值时返回到其起始值，则应发出以下语句：

```
CREATE SEQUENCE id_values
  START WITH 0
  INCREMENT BY 2
  MAXVALUE 1000
  CYCLE
```

有关 `CREATE SEQUENCE` 和 `ALTER SEQUENCE` 语句的详情，参考 *SQL Reference*。

用序列对象提高性能

与身份列相似，使用序列对象来生成值通常会提高应用程序的性能（与备用方法相比较而言）。序列对象的备用方法是创建用来存储当前值的单列表，并用触发器或者在应用程序的控制下增大该值。在应用程序并行访问单列表的分布式环境中，强制对表进行序列化访问所需要的锁定可能会严重影响性能。

序列对象避免了与单列表方法相关联的锁定问题，并且可以将序列值高速缓存到内存中，从而缩短 DB2 响应时间。要使使用序列对象的应用程序的性能最佳，应确保序列对象将适当数量的序列值高速缓存。`CREATE SEQUENCE` 和 `ALTER SEQUENCE` 语句的 `CACHE` 子句指定 DB2 生成并存储在内存中的序列值的最大数目。

如果序列对象必须按次序生成值而不会因系统故障或数据库释放在该次序中产生缝隙，则在 `CREATE SEQUENCE` 语句中使用 `ORDER` 和 `NO CACHE` 子句。`NO CACHE` 子句可以保证所生成的值中不出现缝隙，但代价是某些应用程序性能会有所下降，因为每当它生成新值时，它就会强制序列对象写入数据库日志。

比较序列对象和身份列

尽管序列对象和身份列对 DB2 应用程序起的作用似乎类似，但还是有一些重要区别：

- 身份列将为单个表中的列自动生成值。序列对象将生成可以在任何 SQL 语句中使用的序列值。
- 身份列将生成保证是唯一的值。将 `CYCLE` 子句包括在 `CREATE SEQUENCE` 或 `ALTER SEQUENCE` 语句中，就可以使序列对象生成重复值。

Chapter 7. Stored Procedures

DECIMAL 类型在 Linux Java 例程中失效

发生此问题的原因是 IBM Developer Kit for Java 未创建其库（在 `/usr/lib` 目录中）的链接。DB2 例程的安全性模型不允许它们访问标准系统库之外的库。要在 Linux 上的 Java 例程中启用 `DECIMAL` 支持，执行下列步骤：

1. 通过以 `root` 用户权限发出以下命令，创建从 IBM Developer Kit for Java 库至 `/usr/lib/` 的符号链接：

对于 IBM Developer Kit for Java 1.1.8:

```
ln -sf /usr/jdk118/lib/linux/native_threads/* /usr/lib/
```

对于 IBM Developer Kit for Java 1.3:

```
ln -sf /opt/IBMJava2-13/jre/bin/*.so /usr/lib/
```

2. 发出 `ldconfig` 命令来更新系统范围内的库的列表。

在递归存储过程中使用游标

要在使用以嵌入式 SQL 编写的“SQL 过程”或存储过程时避免错误，在发出递归 CALL 语句之前关闭所有打开的游标。

例如，假定存储过程 MYPROC 包含下列代码段：

```
OPEN c1;
CALL MYPROC();
CLOSE c1;
```

DB2 在 MYPROC 被调用时返回错误，因为游标 c1 在 MYPROC 发出递归 CALL 语句时仍处于打开状态。DB2 返回的特定错误要视 MYPROC 对游标执行的操作而定。

要成功地调用 MYPROC，重写 MYPROC 以在发出嵌入式 CALL 语句之前关闭所有打开的游标，如下例所示：

```
OPEN c1;
CLOSE c1;
CALL MYPROC();
```

在发出嵌入式 CALL 语句之前关闭所有打开的游标以避免错误。

编写 OLE 自动存储过程

以下段落中的最后一句在“Writing OLE automation Stored Procedures”一节下的第二段中丢失：

```
After you code an OLE automation object, you must register the
methods of the object as stored procedures using the CREATE
PROCEDURE statement. To register an OLE automation stored
procedure, issue a CREATE PROCEDURE statement with the LANGUAGE
OLE clause. The external name consists of the OLE progID identifying
the OLE automation object and the method name separated
by ! (exclamation mark). The OLE automation object needs to be
implemented as an in-process server (.DLL).
```

Chapter 12. Working with Complex Objects: User-Defined Structured Types

将结构化类型属性插入到列中

以下规则适用于嵌入式静态 SQL 语句：要将用户定义结构化类型的属性插入到其类型与该属性类型相同的列中，可将表示类型的实例的主变量用括号括起来，并将双点运算符和属性名追加至右括号后：例如，考虑以下情况：

- PERSON_T 是一个包括类型为 VARCHAR(30) 的属性 NAME 的结构化类型。
- T1 是包括类型为 VARCHAR(30) 的列 C1 的表。
- personhv 是在编程语言中声明类型为 PERSON_T 的主变量。

将 NAME 属性插入到列 C1 中的正确语法为：

```
EXEC SQL INSERT INTO T1 (C1) VALUES ( (:personhv)..NAME)
```

Chapter 13. Using Large Objects (LOBs)

联合数据库系统中的大对象 (LOB) 支持

DB2 支持三种类型的大对象 (LOB): 字符大对象 (CLOB)、双字节字符大对象 (DBCLOB) 和二进制大对象 (BLOB)。有关 DB2 LOB 支持的一般信息, 参见下列 DB2 书籍:

- *DB2 Application Development Guide*
- *DB2 SQL Reference*
- *DB2 管理指南: 计划*

在联合数据库系统中, 可访问和处理远程数据源中的 LOB。因为 LOB 可能非常大, 所以从远程数据源传送 LOB 可能要花一些时间。DB2 联合数据库试图使从数据源传送 LOB 的工作量最小化, 并且试图将请求的 LOB 数据直接从数据源传送至请求应用程序, 而不需要在 DB2 中使 LOB 具体化。

本节讨论:

- DB2 如何检索 LOB
- 应用程序可以如何使用 LOB 定位器
- 对 LOB 的限制
- LOB 与非 LOB 数据类型之间的映射
- 调整系统

DB2 如何检索 LOB

DB2 联合系统使用两种机制来检索 LOB: LOB 流动和 LOB 具体化。

LOB 流动: 在 LOB 流动中, LOB 数据是分阶段检索的。DB2 对完全完成的查询的结果集中的数据使用 LOB 流动。例如, 考虑以下查询:

```
SELECT empname, picture FROM orc_emp_table WHERE empno = '01192345'
```

其中, *picture* 表示 LOB 列, *orc_emp_table* 表示用来引用包含职员数据的 Oracle 表的别名。如果 DB2 查询处理器决定在 Oracle 数据源中运行整个查询, 它会将 *picture* 列标记为流动。在执行时, 如果 DB2 注意到 LOB 被标记为流动, 则它将从数据源中分阶段检索 LOB。然后, DB2 会将数据传送至应用程序内存空间。

LOB 具体化: 在 LOB 具体化中, 远程 LOB 数据是由 DB2 检索的, 并在本地存储到联合服务器中。DB2 在下列情况下将使用 LOB 具体化:

- LOB 列不能延迟或流动。
- 必须在本地对 LOB 应用某种功能才能传送数据。当 DB2 为在远程数据源不可用的功能进行补偿时就会发生此情况。例如, Microsoft SQL Server 就不会为 LOB 列提供 SUBSTR 功能。要进行补偿, DB2 就会在本地具体化 LOB 列, 并对检索到的 LOB 应用 DB2 SUBSTR 功能。

应用程序可以如何使用 LOB 定位器

应用程序可以向存储在远程数据源中的 LOB 请求 LOB 定位器。LOB 定位器是存储在主变量中的 4 字节值, 程序可以使用它来表示在数据库系统中挂起的 LOB 值 (或 LOB 表达式)。通过使用 LOB 定位器, 程序就可以将 LOB 值当作是存储在常规主变量中的

LOB 值来进行处理。使用 LOB 定位器时的区别在于不需要将 LOB 值从服务器传送到应用程序（可能也不需要从应用程序传回服务器）。有关 LOB 定位器的附加信息，参见 *DB2 Application Development Guide*。

DB2 可以从远程数据源中检索 LOB，将它们存储在 DB2 中，然后对所存储的 LOB 发出 LOB 定位器。在下列情况下将释放 LOB 定位器：

- 应用程序发出了 "FREE LOCATOR" SQL 语句。
- 应用程序发出了 COMMIT 语句。
- 重新启动了 DB2。

对 LOB 的限制

当使用和检索 LOB 时，应当考虑：

- DB2 无法将远程 LOB 绑定至文件参考变量。
- LOB 在联通方式中不受支持。

LOB 与非 LOB 数据类型之间的映射

有时可在数据源中将 DB2 LOB 数据类型映射为非 LOB 数据类型。当需要在数据源中具有 DB2 LOB 类型的列与其匹配列之间创建映射时，建议您使用 LOB 数据类型来作为匹配列（如果完全有可能的话）。

要创建映射，使用创建类型映射 DDL 语句。例如：

```
CREATE TYPE MAPPING my_oracle_lob FROM sysibm.clob TO SERVER TYPE oracle TYPE long
```

其中：

my_oracle_lob

是类型映射的名称。

sysibm.clob

是 DB2 CLOB 数据类型。

oracle 是您正在连接至的服务器的类型。

long 是 Oracle 数据类型匹配项。

调整系统

如果检索远程 LOB 的应用程序返回一条错误消息，指出没有足够的系统资源来处理该语句，则应增大数据库配置文件中应用程序堆大小参数 APPLHEAPSZ 的值。例如：

```
DB2 UPDATE DB CFG FOR EMPLOYEE USING APPLHEAPSZ 512
```

其中，*EMPLOYEE* 是您正在调整的数据库的名称，512 是应用程序堆大小参数的值。

Part 5. DB2 Programming Considerations

IBM DB2 OLE DB 提供器

安装“IBM DB2 版本 7.1 修订包 1”或更新版本将校正导致 DB2 发生下列错误的情况：

测试连接失败，因为在初始化提供器时出错。“IBM OLE DB 提供器”此时不可用。请参考自述文件以了解详情。

有关对 DB2 使用“IBM OLE DB 提供器”的详情，请参考 <http://www.ibm.com/software/data/db2/udb/ad/v71/oledb.html>。

Chapter 20. C 和 C++ 编程

下表补充说明第 7 章“Stored Procedures”、第 15 章“Writing User-Defined Functions and Methods”及第 20 章“Programming in C and C++”中包括的信息。该表列示了存储过程、UDF 和方法的 SQL 数据类型与 C 数据类型之间的受支持映射。

存储过程、函数和方法的 C 和 C++ 类型

表 22. 映射至 C/C++ 的 SQL 数据类型声明

SQL 列类型	C/C++ 数据类型	SQL 列类型描述
SMALLINT (500 或 501)	sqlint16	16 位带符号整数
INTEGER (496 或 497)	sqlint32	32 位带符号整数
BIGINT (492 或 493)	sqlint64	64 位带符号整数
REAL (480 或 481)	float	单精度浮点
DOUBLE (480 或 481)	double	双精度浮点
DECIMAL(<i>p,s</i>) (484 或 485)	不受支持。	要传送十进制值，将该参数定义为可从 DECIMAL 转换的数据类型（例如 CHAR 或 DOUBLE），并显式地将自变量转换为此类型。
CHAR(<i>n</i>) (452 或 453)	char[<i>n+1</i>], 其中 <i>n</i> 足够大, 可保存数据 1<= <i>n</i> <=254	定长, 零终止字符串
CHAR(<i>n</i>) FOR BIT DATA (452 或 453)	char[<i>n+1</i>], 其中 <i>n</i> 足够大, 可保存数据 1<= <i>n</i> <=254	定长字符串
VARCHAR(<i>n</i>) (448 或 449) (460 或 461)	char[<i>n+1</i>], 其中 <i>n</i> 足够大, 可保存数据 1<= <i>n</i> <=32 672	零终止变长字符串
VARCHAR(<i>n</i>) FOR BIT DATA (448 或 449)	struct { sqluint16 length; char[<i>n</i>] }	非零终止变长字符串
	1<= <i>n</i> <=32 672	
LONG VARCHAR (456 或 457)	struct { sqluint16 length; char[<i>n</i>] }	非零终止变长字符串
	32 673<= <i>n</i> <=32 700	
CLOB(<i>n</i>) (408 或 409)	struct { sqluint32 length; char data[<i>n</i>]; }	带有 4 字节字符串长度指示符的非零终止变长字符串
	1<= <i>n</i> <=2 147 483 647	

表 22. 映射至 C/C++ 的 SQL 数据类型声明 (续)

SQL 列类型	C/C++ 数据类型	SQL 列类型描述
BLOB(<i>n</i>) (404 或 405)	struct { sqluint32 length; char data[<i>n</i>]; }	带有 4 字节字符串长度指示符的非零终止变长二进制字符串
	1<= <i>n</i> <=2 147 483 647	
DATE (384 或 385)	char[11]	零终止字符格式
TIME (388 或 389)	char[9]	零终止字符格式
TIMESTAMP (392 或 393)	char[27]	零终止字符格式
注: 仅当用 WCHARTYPE NOCONVERT 选项进行预编译时, 下列数据类型在 DBCS 或 EUC 环境中才是可用的。		
GRAPHIC(<i>n</i>) (468 或 469)	sqldbchar[<i>n</i> +1] 其中 <i>n</i> 足够大, 可保存数 据 1<= <i>n</i> <=127	定长, 零终止双字节字符串
VARGRAPHIC(<i>n</i>) (400 或 401)	sqldbchar[<i>n</i> +1] 其中 <i>n</i> 足够大, 可保存数 据 1<= <i>n</i> <=16 336	非零终止, 变长双字节字符串
LONG VARGRAPHIC (472 或 473)	struct { sqluint16 length; sqldbchar[<i>n</i>] }	非零终止, 变长双字节字符串
	16 337<= <i>n</i> <=16 350	
DBCLOB(<i>n</i>) (412 或 413)	struct { sqluint32 length; sqldbchar data[<i>n</i>]; }	带有 4 字节字符串长度指示符的非零终止变长字符串
	1<= <i>n</i> <=1 073 741 823	

Chapter 21. Programming in Java

PARAMETER STYLE JAVA 过程和函数中的 Java 方法签名

如果在 CREATE PROCEDURE 或 CREATE FUNCTION 语句的 EXTERNAL NAME 子句中的 Java 方法名之后指定了 Java 方法签名, 则 Java 方法签名必须与在过程或函数名之后指定的签名的缺省 Java 类型映射相对应。例如, SQL 类型 INTEGER 的缺省 Java 映射是 "int", 而不是 "java.lang.Integer"。

连接至 JDBC 小应用程序服务器

Java 小应用程序所使用的 db2java.zip 文件一定要与 JDBC 小应用程序服务器位于同一“修订包”级别。正常情况下, db2java.zip 是从运行 JDBC 小应用程序服务器的“Web 服务器”中装入的, 如本书的“图 22”中所示。这确保了两者的匹配。但是, 如果配置为 Java 小应用程序从另一位置装入 db2java.zip, 则这两者就不是匹配的。对于“修订包 2”之前的版本, 这可能会导致意外的故障。至于“修订包 2”, 在连接时会严格强制两个文件间的“修订包”级别的匹配。如果检测到不匹配, 连接会被拒绝, 且客户机会接收到下列其中一个异常:

- 如果 db2java.zip 位于 “修订包 2” 或更新版本:

```
COM.ibm.db2.jdbc.DB2Exception: [IBM][JDBC Driver]
CLI0621E JDBC 服务器配置不受支持。
```

- 如果 db2java.zip 为 “修订包 2” 之前的版本:

```
COM.ibm.db2.jdbc.DB2Exception: [IBM][JDBC Driver]
CLI0601E 语句句柄无效或语句已关闭。
SQLSTATE=S1000
```

如果不匹配, JDBC 小应用程序服务器会在 jdbcerr.log 文件中记录下列其中一条消息:

- 如果 JDBC 小应用程序服务器位于 “修订包 2” 或更新版本:

```
jdbcFSQLConnect: “JDBC 小应用程序服务器” 与客户机 (db2java.zip)
版本不匹配。无法继续进行连接。 , einfo= -111
```

- 如果 JDBC 小应用程序服务器为 “修订包 2” 之前的版本:

```
jdbcServiceConnection(): 接收到无效请求。 , einfo= 0
```

Appendix B. Sample Programs

以下内容应添加至 “Object Linking and Embedding Samples” 一节:

```
salarycltvc A Visual C++ DB2 CLI sample that calls the
Visual Basic stored procedure, salarysrv.
```

```
SALSVADO A sample OLE automation stored procedure (SALSVADO) and a
SALCLADO client (SALCLADO), implemented in 32-bit Visual Basic and ADO,
that calculates the median salary in table staff2.
```

CLI Guide and Reference

使用运行时客户机绑定数据库实用程序

在可将数据库实用程序（导入、导出、重组、命令行处理器）和 DB2 CLI 绑定文件与每个数据库配合使用之前，不能使用“DB2 运行时客户机”来将这些实用程序和文件与该数据库绑定。必须转而使用“DB2 管理客户机”或“DB2 应用程序开发客户机”。

必须将这些数据库实用程序和 DB2 CLI 绑定文件与每个数据库绑定，这些程序和文件才能与该数据库配合使用。在网络环境中，如果使用运行于不同操作系统之上的多个客户机，或这些客户机处于不同的 DB2 版本或服务级别，则必须对每个操作系统和 DB2 版本组合都绑定这些实用程序一次。

在 CLI 应用程序中使用静态 SQL

有关在 CLI 应用程序中使用静态 SQL 的详情，参见网址如下的 Web 页面：
<http://www.ibm.com/software/data/db2/udb/staticcli/>

创建静态 JDBC/ODBC/CLI 简要表的限制

创建 JDBC/ODBC/CLI 静态简要表当前主要用于简明应用程序。它不适用于带有许多功能组件以及在执行期间要进行复杂程序逻辑运算的复杂应用程序。

SQL 语句必须已成功执行，才能捕捉到创建简要表对话中。在语句匹配对话中，不匹配的动态语句将继续作为动态 JDBC/ODBC/CLI 调用来执行。

SQL 语句必须与被捕捉和绑定为语句匹配的有效候选语句的每个字符都相同。空格也是有意义的：例如，“COL = 1”会被认为与“COL=1”不同。使用参数标记取代文字，可增大匹配的命中率。

执行预先绑定了静态 SQL 语句的应用程序时，控制动态语句行为的动态寄存器对转换为静态方式的语句将没有影响。

如果应用程序对后续 DML 语句中引用的对象发出 DDL 语句，将会在捕捉文件中找到全部这些语句。JDBC/ODBC/CLI 静态创建简要表绑定工具将尝试绑定它们。对于支持 VALIDATE(RUN) 绑定选项的 DBMS，绑定尝试将会成功，但对于不支持 VALIDATE(RUN) 绑定选项的 DBMS，绑定尝试将会失败。在这种情况下，应用程序不应使用创建静态简要表。

数据库管理员可能会根据应用程序特定的要求来编辑捕捉文件，以添加、更改或删除 SQL 语句。

ADT 变换

以下内容会取代本书中的现存信息。

- There is a new descriptor type (smallint) SQL_DESC_USER_DEFINED_TYPE_CODE, with values:

```
SQL_TYPE_BASE 0 (this is not a USER_DEFINED_TYPE)
SQL_TYPE_DISTINCT 1
SQL_TYPE_STRUCTURED 2
This value can be queried with either SQLColAttribute
or SQLGetDescField (IRD only).
```

The following attributes are added to obtain the actual type names:

```
SQL_DESC_REFERENCE_TYPE
SQL_DESC_STRUCTURED_TYPE
SQL_DESC_USER_TYPE
```

The above values can be queried using SQLColAttribute
or SQLGetDescField (IRD only).

- Add `SQL_DESC_BASE_TYPE` in case the application needs it. For example, the application may not recognize the structured type, but intends to fetch or insert it, and let other code deal with the details.
- Add a new connection attribute called `SQL_ATTR_TRANSFORM_GROUP` to allow an application to set the transform group (rather than use the SQL "SET CURRENT DEFAULT TRANSFORM GROUP" statement).
- Add a new statement/connection attribute called `SQL_ATTR_RETURN_USER_DEFINED_TYPES` that can be set or queried using `SQLSetConnectAttr`, which causes CLI to return the value `SQL_DESC_USER_DEFINED_TYPE_CODE` as a valid SQL type. This attribute is required before using any of the transforms.
 - By default, the attribute is off, and causes the base type information to be returned as the SQL type.
 - When enabled, `SQL_DESC_USER_DEFINED_TYPE_CODE` will be returned as the `SQL_TYPE`. The application is expected to check for `SQL_DESC_USER_DEFINED_TYPE_CODE`, and then to retrieve the appropriate type name. This will be available to `SQLColAttribute`, `SQLDescribeCol`, and `SQLGetDescField`.
- The `SQLBindParameter` does not give an error when you bind `SQL_C_DEFAULT`, because there is no code to allow `SQLBindParameter` to specify the type `SQL_USER_DEFINED_TYPE`. The standard default C types will be used, based on the base SQL type flowed to the server. For example:

```
sqlrc = SQLBindParameter (hstmt, 2, SQL_PARAM_INPUT, SQL_C_CHAR, SQL_VARCHAR, 30,
                          0, &c2, 30, NULL);
```

Chapter 3. Using Advanced Features

编写多线程应用程序

应将以下内容添加至 "Multi-Threaded Mixed Applications" 一节:

注: It is recommended that you do not use the default stack size, but instead increase the stack size to at least 256 000. DB2 requires a minimum stack size of 256 000 when calling a DB2 function. You must ensure therefore, that you allocate a total stack size that is large enough for both your application and the minimum requirements for a DB2 function call.

可滚动游标

应该将以下信息添加到 "Scrollable Cursors" 一节中:

Server-side Scrollable Cursor Support for OS/390

The UDB client for the Unix, Windows, and OS/2 platforms supports updatable server-side scrollable cursors when run against OS/390 Version 7 databases. To access an OS/390 scrollable cursor on a three-tier environment, the client and the gateway must be running DB2 UDB Version 7.1, FixPak 3 or later.

有两个可以访问可滚动游标的应用程序启用接口: ODBC 和 JDBC。JDBC 接口只能访问静态的可滚动游标, 而 ODBC 接口可以访问静态的和键集驱动的服务器端可滚动游标。

游标属性: 下表列示了 ODBC 中 OS/390 版本 7 游标的缺省属性。

表 23. ODBC 中 OS/390 游标的缺省属性

游标类型	游标灵敏度	游标可更新	游标并行性	游标可滚动
仅转发 ^a	未指定	不可更新	只读并行性	不可滚动
静态	不敏感	不可更新	只读并行性	可滚动
键集驱动的	敏感的	可更新	值并行性	可滚动

a “仅转发”是在没有 FOR UPDATE 子句情况下的可滚动游标的缺省行为。对“仅转发”游标指定 FOR UPDATE 将创建一个可更新、锁定并行性、不可滚动的游标。

受支持的取装方向: 所有 ODBC 取装方向都是通过 SQLFetchScroll 或 SQLExtendedFetch 接口来支持的。

更新键集驱动的游标: 键集驱动的游标是可更新游标。CLI 驱动程序将 FOR UPDATE 子句追加至查询, 但以下两种情况除外: 查询是作为 SELECT ... FOR READ ONLY 查询发出的, 或者 FOR UPDATE 子句已经存在。在“DB2 OS/390 版”中实现的键集驱动游标是值并行性游标。值并行性游标将导致有利锁定, 在此情况下, 锁定不会被挂起, 直到尝试了更新或删除操作为止。当尝试了更新或删除操作时, 数据库服务器将把应用程序检索到的先前值与基本表中的当前值进行比较。如果这些值相匹配, 则表示更新或删除成功。如果这些值不匹配, 则表示操作失败。如果发生了故障, 则应用程序应该再次查询这些值, 并重新发出更新或删除操作 (如果它仍适用的话)。

应用程序可以用两种方法来更新键集驱动游标:

- 发出 UPDATE WHERE CURRENT OF "<cursor name>" 或 DELETE WHERE CURRENT OF "<cursor name>" (发出时将 SQLPrepare() 与 SQLExecute() 或 SQLExecDirect() 配合使用)。
- 使用 SQLSetPos() 或 SQLBulkOperations() 来对结果集更新、删除或添加行。

注: 通过 SQLSetPos() 或 SQLBulkOperations() 而添加到结果集中的行被插入到服务器上的表中, 但是不会添加到服务器的结果集中。因此, 这些行是不可更新的, 并且它们对其他事务所作的更改不敏感。然而, 所插入行将作为结果集的一部分出现, 因为它们是在客户机上高速缓存的。任何应用于所插入行的触发器对于应用程序来说就好象尚未应用它们一样。要使所插入行可更新、敏感, 且要查看适用触发器的结果, 应用程序必须再次发出查询来重新生成结果集。

在可滚动游标支持之前创建的应用程序的故障诊断： 因为可滚动游标支持是新的，所以使用“UDB OS/390 版”或者“UDB Unix 版、Windows 版和 OS/2 版”前发行版的某些 ODBC 应用程序可能会遇到行为或性能方面的更改。发生这种情况的原因是：在可滚动游标受支持之前，请求了可滚动游标的应用程序将接收到“仅转发”游标。要在可滚动游标支持之前恢复应用程序的先前行为，应在 db2cli.ini 文件中设置下列配置关键字：

表 24. 配置关键字值在可滚动游标支持之前恢复应用程序行为

配置关键字设置	描述
PATCH2=6	返回一条消息，陈述（键集驱动的和静态的）可滚动游标都不受支持。CLI 将自动使对可滚动游标的任何请求降级为“仅转发”游标。
DisableKeysetCursor=1	禁用服务器端和客户机端键集驱动的可滚动游标。这可以在请求了键集驱动的游标时用来强制 CLI 驱动程序为应用程序提供静态游标。
UseServerKeysetCursor=0	禁止（使用客户机端键集驱动游标库的应用程序的）服务器端键集驱动游标仿真键集驱动游标。仅当遇到服务器端键集驱动游标问题时才使用此选项，因为客户机端游标将导致大量的开销，且通常比服务器端游标的性能差。

使用复合 SQL

本书中缺少以下注释：

Any SQL statement that can be prepared dynamically, other than a query, can be executed as a statement inside a compound statement.

Note: Inside Atomic Compound SQL, savepoint, release savepoint, and rollback to savepoint SQL statements are also disallowed. Conversely, Atomic Compound SQL is disallowed in savepoint.

使用存储过程

在 CLI 中编写存储过程

以下是未说明的对 CLI 存储过程的限制：

如果在调用多个 CLI 存储过程，则应用程序必须关闭一个存储过程中的打开的游标，才能调用下一个存储过程。更具体地说，必须关闭第一组打开的游标，下一个存储过程才能尝试打开游标。

CLI 存储过程和自动绑定

本书中补充了下列信息：

通常，在 CLI/ODBC 应用程序第一次对数据库执行 SQL 时，CLI/ODBC 驱动程序将会自动绑定 CLI 软件包（假定用户具有适当的特权或权限）。CLI 软件包的自动绑定不能在存储过程中进行，因此，如果应用程序最开始做的操作是调用 CLI 存储过程的话，自动绑定操作将不会进行。在运行 CLI 应用程序以对另一 DB2 数据库调用 CLI 存储过程时，您必须使用此命令对 CLI 软件包进行一次绑定：

UNIX

```
db2 bind <BNDPATH>/@db2cli.lst blocking all
```

Windows 和 OS/2

```
db2bind "%DB2PATH%\bnd\@db2cli.lst" blocking
```

建议使用的方法是总是在创建数据库时绑定这些软件包，以避免在运行时进行自动绑定。如果用户没有特权，或者如果另一应用程序试图在同一时间进行自动绑定，则自动绑定将会失败。

Chapter 4. Configuring CLI/ODBC and Running Sample Applications

配置关键字

忽略 CURRENTFUNCTIONPATH 关键字中的最后一段。正确的信息如下所示：

This keyword is used as part of the process for resolving unqualified function and stored procedure references that may have been defined in a schema name other than the current user's schema. The order of the schema names determines the order in which the function and procedure names will be resolved. For more information on function and procedure resolution, refer to the SQL Reference.

Chapter 5. DB2 CLI Functions

SQLBindFileToParam - 将 LOB 文件参考绑定至 LOB 参数

SQLBindFileToParam() CLI 函数中的最后一个参数 IndicatorValue 当前被描述为 "output (deferred)"。它应为 "input (deferred)"。

SQLNextResult - 将下一个结果集与另一语句句柄相关联

应将以下文本添加至 Chapter 5, "DB2 CLI Functions"：

用途

规范：DB2 CLI 7.x

语法

```
SQLRETURN SQLNextResult (SQLHSTMT StatementHandle1  
                          SQLHSTMT StatementHandle2);
```

函数自变量

表 25. *SQLNextResult* 自变量

数据类型	自变量	使用	描述
SQLHSTMT	<i>StatementHandle</i>	输入	语句句柄。
SQLHSTMT	<i>StatementHandle</i>	输入	语句句柄。

用法

存储过程通过在退出后留下一个或多个游标返回多个结果集。第一个结果集总是通过使用调用该存储过程的语句句柄进行访问的。如果返回了多个结果集，可使用 *SQLMoreResults()* 或 *SQLNextResult()* 来描述和取装该结果集。

SQLMoreResults() 用来关闭第一个结果集的游标，并允许处理下一个结果集，而 SQLNextResult() 会将下一个结果集移至 *StatementHandle2*，且未关闭 *StatementHandle1* 上的游标。如果没有任何结果集要取装的话，两个函数都会返回 SQL_NO_DATA_FOUND。

使用 SQLNextResult() 允许结果集被传输至其他语句句柄后以任意次序处理这些结果集。允许同时调用 SQLMoreResults() 和 SQLNextResult()，直到 *StatementHandle1* 上没有其他游标（打开的结果集）为止。

如果 SQLNextResult() 返回了 SQL_SUCCESS，下一个结果集就不再与 *StatementHandle1* 相关联。而是，下一个结果集会与 *StatementHandle2* 相关联，就好像对 SQLExecDirect() 的调用成功执行了 *StatementHandle2* 的查询。因此，该游标可使用 SQLNumResultSets()、SQLDescribeCol() 或 SQLColAttribute() 来描述。

调用了 SQLNextResult() 之后，现在与 *StatementHandle2* 相关联的结果集会从余下的结果集链中除去，且不能在 SQLNextResult() 或 SQLMoreResults() 中再次使用。这意味着对于 'n' 个结果集 SQLNextResult() 最多只能成功地调用 'n-1' 次。

如果调用 SQLFreeStmt() 时带有 SQL_CLOSE 选项，或调用 SQLFreeHandle() 时 *HandleType* 被设置为 SQL_HANDLE_STMT，此语句句柄上的所有暂挂结果集都会被废弃。

如果 *StatementHandle2* 带有打开的游标，或 *StatementHandle1* 和 *StatementHandle2* 不在同一连接上，SQLNextResult() 会返回 SQL_ERROR。如果返回任何错误或警告，SQLError() 必须始终是在 *StatementHandle1* 上调用的。

注：SQLMoreResults() 还使用了参数化查询，该查询带有输入参数值数组，这些参数值是使用 SQLParamOptions() 和 SQLBindParameter() 指定的。但是，SQLNextResult() 不支持该查询。

返回码

- SQL_SUCCESS
- SQL_SUCCESS_WITH_INFO
- SQL_STILL_EXECUTING
- SQL_ERROR
- SQL_INVALID_HANDLE
- SQL_NO_DATA_FOUND

诊断

表 26. SQLNextResult SQLSTATE

SQLSTATE	描述	解释
40003 08S01	通信链路故障。	应用程序与数据源间的通信链路失效，函数未完成。
58004	意外的系统故障。	不可恢复的系统错误。
HY001	内存分配故障。	DB2 CLI 无法分配支持函数的执行和完成所需的内存。

表 26. *SQLNextResult SQLSTATE* (续)

SQLSTATE	描述	解释
HY010	函数序列错误。	函数是在执行数据 (SQLParamData()、SQLPutData()) 操作时调用的。 <i>StatementHandle2</i> 有一个打开的游标与其相关联。 该函数是在执行 BEGIN COMPOUND 和 END COMPOUND SQL 操作时调用的。
HY013	意外的内存处理错误。	DB2 CLI 无法访问支持执行和完成该函数所需的内存。
HYT00	超时到期。	超时周期到期, 数据源未返回结果集。超时仅在非多任务系统 (如 Windows 3.1 和 Macintosh 系统 7) 上才受支持。超时周期可使用 SQLSetConnectAttr() 的 SQL_ATTR_QUERY_TIMEOUT 属性来进行设置。

限制

只有 SQLMoreResults() 才能用于参数化查询。

引用

- 第 535 页中的 "SQLMoreResults - Determine If There Are More Result Sets"
- 第 120 页中的 "Returning Result Sets from Stored Procedures"

Appendix D. Extended Scalar Functions

日期和时间函数

下列函数是附录 D "Extended Scalar Functions" 的 "Date and Time Functions" 一节中所缺少的函数:

DAYOFWEEK_ISO(*date_exp*)

在 *date_exp* 中以范围为 1 到 7 内的整数值返回一个星期中的某天, 其中 1 表示星期一。注意此函数与 DAYOFWEEK() 函数之间的差异, 在后者中 1 表示星期日。

WEEK_ISO(*date_exp*)

在 *date_exp* 中以范围为 1 到 53 内的整数值返回一年中的某个星期。第一个星期被定义为要包含星期四的一年中的首个星期。因此, Week1 相当于包含 1 月 4 日的第一个星期, 因为星期一被视作一个星期中的第一天。

注意, WEEK_ISO() 与 WEEK() 的当前定义不同, 后者最多可返回值 54。对于 WEEK() 函数, Week 1 是包含第一个星期六的星期。这相当于包含 1 月 1 日的星期, 即使该星期只包含一天。

DAYOFWEEK_ISO() 和 WEEK_ISO() 在版本 7 中创建的数据库中是自动提供的。如果数据库是在版本 7 之前创建的, 则可能未提供这些函数。要使 DAYOFWEEK_ISO() 和 WEEK_ISO() 函数在这种数据库中可用, 使用 **db2updb** 系统命令。有关 **db2updb** 的详情, 参见《发行说明》中的 "Command Reference" 一节。

Appendix K. Using the DB2 CLI/ODBC/JDBC Trace Facility

此附录中的各节已更新。查看 *Troubleshooting Guide* 中的 "Traces" 一章以获取有关此跟踪设施的最新信息。

消息参考

获取消息和 SQLSTATE 帮助

命令行处理器中可用的帮助包含关于《消息参考》中不可用的消息和 SQLSTATE 值的新帮助和更新过的帮助。

要显示命令行处理器中的消息帮助，在操作系统命令提示符处输入以下命令：

```
db2 "? XXXnnnnn"
```

其中，XXX 表示消息前缀，nnnnn 表示消息号。

例如，db2 "? SQL30081" 将显示关于 SQL30081 消息的帮助。

要显示命令行处理器中的 SQLSTATE 文本，在操作系统命令提示符处输入以下命令：

```
db2 "? XXXXX"
```

其中，XXXXX 表示 SQLSTATE 值。

例如，db2 "? 428F1" 将显示 SQLSTATE 428F1 的文本。

DB2 Connect 中的 SQLCODE 重新映射更改

在版本 7.2 中，DB2 Connect 的缺省 SQLCODE 重新映射已经更改。当主机数据库返回 SQLCODE 值 -567 时，DB2 Connect 现在会在将 SQLCODE 值返回给 DB2 客户机之前将 SQLCODE 值重新映射为 -551。

新消息和已更改的消息

以下列表中包含自对 DB2 版本 7.1 发布消息参考以来已经更改的消息的消息号。如果在使用 DB2 时接收到其中一条消息，则您还将接收到正确的更新过的消息；但是，该消息将不会对应于消息参考中的信息。

调用层接口 (CLI) 消息

CLI0645E

CLI0646E

CLI0647E

DB2 消息

DB21086I

DB210060E

DB210061E

DB210062E

DB210113E
DB210114E
DB210115E
DB210116E
DB210117E
DB210118E
DB210120E
DB210121E
DB210200I
DB210201I

DBI 消息

DBI1172E
DBI1793W
DBI1794E
DBI1795E
DBI1796W
DBI1797I

数据仓库中心 (DWC) 消息

DWC0000I
DWC03504E
DWC08900E
DWC08901E
DWC08902E
DWC08903E
DWC08904E
DWC08907C
DWC08908C
DWC08909C
DWC08910E
DWC08911E
DWC08912E
DWC08913E
DWC08914E
DWC08915E
DWC08917E
DWC08919I
DWC08930E
DWC08931E

DWC08932E
DWC08933E
DWC08934E
DWC08935E
DWC08936W
DWC08937I
DWC08938I
DWC08939I
DWC08940I
DWC08941I
DWC08960I
DWC08961I
DWC08962I
DWC08963I
DWC08964I
DWC08965I
DWC08966E
DWC08967E
DWC08968E
DWC13239E
DWC13300E
DWC13301E
DWC13302E
DWC13304E
DWC13603E
DWC13700E
DWC13701E
DWC13702E
DWC13703E
DWC13705E
DWC13706E
DWC13707E

SQL 消息

SQL0017N
SQL0056N
SQL0057N
SQL0058N
SQL0097N
SQL0224N

SQL0225N
SQL0227N
SQL0228N
SQL0231W
SQL0243N
SQL0244N
SQL0270N
SQL0301N
SQL0303N
SQL0336N
SQL0348N
SQL0349N
SQL0357N
SQL0358N
SQL0368N
SQL0408N
SQL0423N
SQL0590N
SQL0670N
SQL0845N
SQL0846N
SQL1179W
SQL1186N
SQL1550N
SQL1551N
SQL1552N
SQL1553N
SQL1704N
SQL2077W
SQL2078N
SQL2417N
SQL2426N
SQL2571N
SQL2572N
SQL2573N
SQL2574N
SQL2575N
SQL2576N
SQL4942N
SQL5012N

SQL6583N
SQL20005N
SQL20117N
SQL20121N
SQL20133N
SQL20134N
SQL20135N
SQL20143N
SQL20144N
SQL20145N
SQL20146N
SQL20147N
SQL20148N
SQL20153N
SQL21000N

校正过的 SQLSTATES

表 27.

42630	SQLSTATE 或 SQLCODE 变量在此上下文中无效。
42631	必须在 SQL 函数的 RETURN 语句上指定表达式。
42632	在 SQL 函数或方法中必须有 RETURN 语句。
428F2	必须在 SQL 过程的 RETURN 语句上指定整数表达式。
560B7	对于多行 INSERT, NEXTVAL 序列表达式的用法对每一行都必须是相同的。

SQL Reference

SQL Reference 是在一个 PDF 文件中提供的

每本书中的『使用 DB2 资料库』附录都指示，提供了 PDF 格式的 SQL Reference（分为单独的两卷）。这是不正确的。

尽管打印书籍以两卷的形式出现，且相应的两个书号是正确的，但是仅有一个 PDF 文件，它包含有两卷。该 PDF 文件名为 db2s0x70。

Chapter 3. Language Elements

Naming Conventions and Implicit Object Name Qualifications

将以下注释添加至第 3 章中的这一节：

The following names, when used in the context of SQL Procedures, are restricted to the characters allowed in an ordinary identifier, even if the names are delimited:

- condition-name
- label
- parameter-name
- procedure-name
- SQL-variable-name
- statement-name

DATALINK Assignments

本节中的一个段落已更改为：

Note that the size of a URL parameter or function result is the same on both input or output and is bound by the length of the DATALINK column. However, in some cases the URL value returned has an access token attached. In situations where this is possible, the output location must have sufficient storage space for the access token and the length of the DATALINK column. Hence, the actual length of the comment and URL in its fully expanded form provided on input should be restricted to accommodate the output storage space. If the restricted length is exceeded, this error is raised.

表达式

语法图

语法图已更改：



operator:



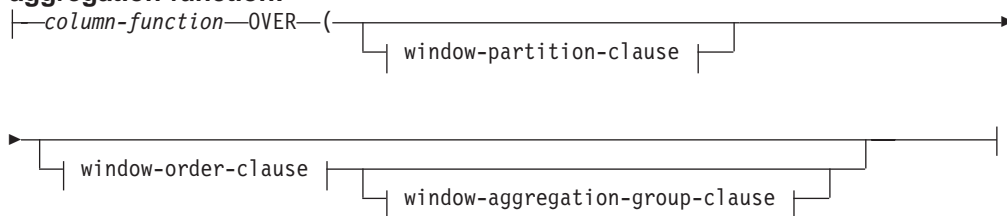
注:

1 || 可用作 CONCAT 的同义词。

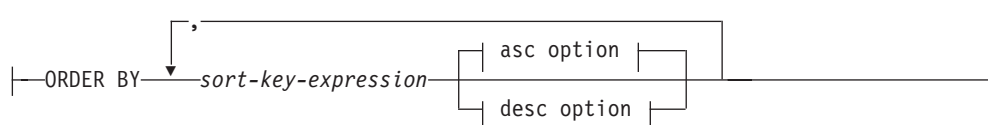
OLAP 函数

以下内容表示对第 3 章中的 "Expressions" 下 "OLAP Functions" 节的校正。

aggregation-function:



window-order-clause:



asc option:



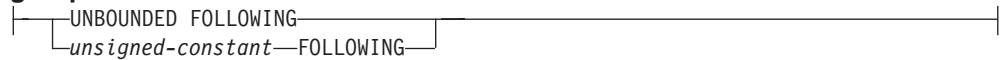
desc option:



window-aggregation-group-clause:



group-end:



在 window-order-clause 描述中:

NULLS FIRST

窗口的排序为: 在排序次序中空值置于所有非空值之前。

NULLS LAST

窗口的排序为: 在排序次序中空值置于所有非空值之后。

在 window-aggregation-group-clause 描述中:

window-aggregation-group-clause

行 R 的聚合组是一组行, 这些行是在 R 的分区中行排序中相对于 R 定义的。此子句指定该聚合组。如果未指定此子句, 则缺省值与 RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW 相同, 并提供了累积的聚合结果。

ROWS

指示聚合组是通过对行进行计数而定义的。

RANGE

指示聚合组是按从排序键的偏移量定义的。

group-start

指定聚合组的起始点。该聚合组以当前行结尾。group-start 子句的规范等同于格式为 "BETWEEN group-start AND CURRENT ROW" 的 group-between 子句。

group-between

根据 ROWS 或 RANGE 来指定聚合组的起点和终点。

group-end

指定聚合组的终点。该聚合组从当前行开始。group-end 子句的规范等同于格式为 "BETWEEN CURRENT ROW AND group-end" 的 group-between 子句。

UNBOUNDED PRECEDING

包括当前行之前的整个分区。这可使用 ROWS 或 RANGE 来指定。这还可通过 window-order-clause 中的多个 sort-key-expression 指定。

UNBOUNDED FOLLOWING

包括当前行之后的整个分区。这可使用 ROWS 或 RANGE 来指定。这还可通过 window-order-clause 中的多个 sort-key-expression 指定。

CURRENT ROW

根据当前行指定聚合组的起点或终点。如果指定了 ROWS，则当前行为聚合组边界。如果指定了 RANGE，则聚合组边界包括 *sort-key-expressions* 的值与当前行的值相同的行的集合。此子句不能在 *group-bound2* 中指定（如果 *group-bound1* 指定了 *value* FOLLOWING 的话）。

value PRECEDING

指定当前行之前的行的范围或数目。如果指定了 ROWS，则 *value* 为一个指示行数的正整数。如果指定了 RANGE，则 *value* 的数据类型必须与 window-order-clause 的 sort-key-expression 类型相类似。仅可有一个 sort-key-expression，且 sort-key-expression 的数据类型必须允许使用减法。不能在 *group-bound2* 中指定此子句（如果 *group-bound1* 为 CURRENT ROW 或 *value* FOLLOWING 的话）。

value FOLLOWING

指定当前行之后的行的范围或数目。如果指定了 ROWS，则 *value* 为一个指示行数的正整数。如果指定了 RANGE，则 *value* 的数据类型必须与 window-order-clause 的 sort-key-expression 类型相类似。仅可有一个 sort-key-expression，且 sort-key-expression 的数据类型必须允许使用加法。

序列参考

应当将以下信息添加至 Expressions 节的末尾（在 "Subtype Treatment" 之后）。

sequence-reference:

```
| nextval-expression |
| prevval-expression |
```

nextval-expression:

```
| NEXTVAL FOR sequence-name
```

prevval-expression:

```
| PREVVAL FOR sequence-name
```

NEXTVAL FOR *sequence-name*

NEXTVAL 表达式返回 *sequence-name* 指定的序列的下一个值。

PREVVAL FOR *sequence-name*

A PREVVAL expression returns the most recently generated value for the specified sequence for a previous statement within the current session. This value can be repeatedly referenced using PREVVAL expressions specifying the name of the sequence.

There may be multiple instances of PREVVAL expressions specifying the same sequence name within a single statement and they all return the same value.

A PREVVAL expression can only be used if a NEXTVAL expression specifying the same sequence name has already been referenced in the current user session (in the current or a previous transaction) (SQLSTATE 51035).

注:

- 当 NEXTVAL 表达式指定序列的名称时，会生成新的序列号。然而，如果一个查询中有多个指定同一序列名的 NEXTVAL 表达式的实例，则序列的计数器只对结果的每一行增加一次。
- 可以使用指定了序列名的 PREVVAL 表达式来重复地引用最新为该序列生成的值。在单一语句中，可能有多个指定了同一序列名的 PREVVAL 表达式的实例。
- 可以将同一个序列号用作两个不同的表中的唯一键值，方法是对第一行使用 NEXTVAL 表达式来引用该序列号（这生成序列值），并对其他行使用 PREVVAL 表达式来引用该序列号（PREVVAL 的这个实例引用先前语句中的 NEXTVAL 表达式生成的序列值），如下所示：

```
INSERT INTO order(orderno, custno)
VALUES (NEXTVAL FOR order_seq, 123456);
```

```
INSERT INTO line_item (orderno, partno, quantity)
VALUES (PREVVAL FOR order_seq, 987654, 1);
```

- 可以指定 NEXTVAL 和 PREVVAL 表达式的示例包括：
 - select-statement 或 SELECT INTO 语句：在 select-clause 中，只要该语句不包含 DISTINCT 关键字、GROUP BY 子句、ORDER BY 子句、UNION 关键字、INTERSECT 关键字或 EXCEPT 关键字
 - INSERT 语句：在 VALUES 子句内
 - INSERT 语句：在全查询的 select-clause 中
 - UPDATE 语句：在 SET 子句中的表达式的全查询的 select-clause 中（搜索型或定位型 UPDATE 语句）
 - VALUES INTO 语句：在表达式的全查询的 select-clause 中
- 不能指定 NEXTVAL 和 PREVVAL 表达式的示例包括 (SQLSTATE 428F9):
 - 全外连接的连接条件
 - CREATE TABLE 或 ALTER TABLE 语句中的列的 DEFAULT 值
 - CREATE TABLE 或 ALTER TABLE 语句中的已生成的列定义
 - CHECK 约束的条件
 - CREATE TRIGGER 语句
 - CREATE VIEW 语句
 - CREATE METHOD 语句
 - CREATE FUNCTION 语句。
- 另外，不能在下列各项中指定 NEXTVAL 表达式 (SQLSTATE 428F9):
 - CASE 表达式
 - 聚合函数的参数列表
 - 子查询
 - 包含 DISTINCT 运算符的 SELECT 语句

- 连接的连接条件
 - SELECT 语句的 GROUP BY 子句
 - 使用 UNION、INTERSECT 或 EXCEPT 集合运算符与另一 SELECT 语句组合在一起的 SELECT 语句
 - 嵌套的表表达式
 - 表函数的参数列表
 - SELECT、DELETE 或 UPDATE 语句的 WHERE 子句
 - ORDER BY 子句
 - CALL 语句的参数列表。
- 当为序列生成一个值时，便消耗了那个值，下次需要值时，将生成新的值。即使包含 NEXTVAL 表达式的语句失败，情况亦如此。
 - 如果 INSERT 语句在列的 VALUES 列表中包含 NEXTVAL 表达式，并且 INSERT 的执行期间发生了一些错误（这可能是生成下一个序列值出问题，也可能是另一个列的值有问题），则插入失败，为序列生成的值被认为是已消耗。在某些情况下，重新发出同一 INSERT 语句可能会成功。
例如，请考虑作为这种情况的结果发生的错误：对存在唯一索引的列使用了 NEXTVAL，而生成的序列值已存在于该索引中。为序列生成的下一个值有可能是不存在于该索引中的值，因此后续的 INSERT 将成功。
 - 如果在为序列生成值时超出了序列的最大值（或者，对于降序序列，超出了最小值），并且不允许循环，则会出错 (SQLSTATE 23522)。在这种情况下，用户可以改变 (ALTER) 该序列以扩展可接受的值的范围，或对该序列启用循环，或者删除 (DROP) 该序列并使用另一种具有更大的值范围的数据类型创建 (CREATE) 一个新序列。
例如，一个序列定义为具有数据类型 SMALLINT，并且，这个序列最终耗尽了可用于赋值的值。要将该序列重新定义为 INTEGER，您需要删除并使用新定义重新创建该序列。
 - 对游标的选择语句中的 PREVVAL 的引用涉及在打开游标之前对指定序列生成的值。然而，关闭该游标可能会影响 PREVVAL 对后续语句中的指定序列所返回的值，甚至会影响同一语句（如果重新打开游标的话）。当游标的选择语句包括对同一序列名的 NEXTVAL 的引用时，可能就会发生这种情况。

示例： 这些示例假定有一个名为 "order" 的表，且名为 "order_seq" 的序列是这样创建的：

```
CREATE SEQUENCE order_seq
  START WITH 1
  INCREMENT BY 1
  NOMAXVALUE
  NOCYCLE
  CACHE 24
```

- 下面是如何使用 NEXTVAL 表达式为上面创建的序列生成 "order_seq" 序列号的一些示例：

```
INSERT INTO order(orderno, custno)
  VALUES (NEXTVAL FOR order_seq, 123456);
```

或者，

```
UPDATE order
  SET orderno = NEXTVAL FOR order_seq
  WHERE custno = 123456;
```

或者,

```
VALUES NEXTVAL FOR order_seq INTO :hv_seq;
```

Chapter 4. Functions

启用新函数和过程

在“版本 7 修订包”中提供了新的 SQL 内置标量函数。有关这些新函数的描述, 参考 *SQL Reference* 更新。当数据库服务器代码升级为新的服务级别时, 不会在每个数据库上自动启用新函数。要启用这些新函数, 系统管理员必须发出命令 **db2updv7**, 以指定服务器上的每个数据库。此函数在数据库生成一个条目, 以确保在执行此命令之前所创建的数据库对象使用可与新函数签名匹配的现存函数签名。

有关启用 MQSeries 函数 (在 MQDB2 模式中定义的那些函数) 的信息, 参见 第 159 页的『MQSeries』。

标量函数

ABS 或 ABSVAL

► ABS (expression) ◄
└─ ABSVAL ─┘

模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

注: ABS (或 ABSVAL) 函数的 SYSFUN 版本仍然可用。

返回自变量的绝对值。

自变量是返回任何内置数字数据类型的值的表达式。

函数的结果与自变量具有相同的数据类型和长度属性。若自变量可空, 或者配置数据库时将 DFT_SQLMATHWARN 设置为“是”, 则结果可空; 若自变量为空, 则结果为空值。

例如:

```
ABS(-51234)
```

返回 INTEGER, 其值为 51234。

DECRYPT_BIN 和 DECRYPT_CHAR

► DECRYPT_BIN (expression) ◄
└─ DECRYPT_CHAR ─┘

► (—encrypted-data—, —password-string-expression—) ◄

模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

DECRYPT_BIN 和 DECRYPT_CHAR 函数返回一个值，这个值是对 *encrypted-data* 进行解密的结果。用于解密的口令是 *password-string-expression* 值或 ENCRYPTION PASSWORD 值（使用 SET ENCRYPTION PASSWORD 语句指定）。DECRYPT_BIN 和 DECRYPT_CHAR 函数只能对使用 ENCRYPT 函数加密的值进行解密 (SQLSTATE 428FE)。

encrypted-data

一个表达式，此表达式返回 CHAR FOR BIT DATA 或 VARCHAR FOR BIT DATA 值，这个值是已使用 ENCRYPT 函数进行加密的完整、已加密数据字符串。

password-string-expression

一个表达式，此表达式返回至少 6 个字节且不超过 127 个字节的 CHAR 或 VARCHAR 值 (SQLSTATE 428FC)。这应该是以前用来对数据进行加密的口令，否则解密将导致错误 (SQLSTATE 428FD)。如果口令自变量的值是空，或者未提供，则将使用 ENCRYPTION PASSWORD 值（必须对会话设置这个值）对口令进行解密 (SQLSTATE 51039)。

DECRYPT_BIN 函数的结果是 VARCHAR FOR BIT DATA。DECRYPT_CHAR 函数的结果是 VARCHAR。即使 *encrypted-data* 包含暗示，此函数也不返回该暗示。结果的长度属性是 *encrypted-data* 的数据类型的长度属性减去 8 个字节。此函数返回的值的实际长度将与被加密的原始字符串的长度相匹配。如果 *encrypted-data* 包括已加密字符串后面的字节，则此参数不返回这些字节。如果第一个自变量可空，则结果可空；如果第一个字变量为空，则结果是空值。

如果数据是在另一个系统上加密的，且加密时使用的代码页与执行解密所使用的代码页不同，则可能是在将已加密的值转换至数据库代码页时发生了扩充。在这样的情况下，应该将 *encrypted-data* 值强制转换为字节数更大的 VARCHAR 字符串。

另请参见『ENCRYPT』和第338页的『GETHINT』，以了解有关使用此函数的其他信息。

示例:

示例 1: 本示例使用 ENCRYPTION PASSWORD 值来存放加密口令。

```
SET ENCRYPTION PASSWORD = 'Ben123';
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832');
SELECT DECRYPT_CHAR(SSN)
FROM SSN;
```

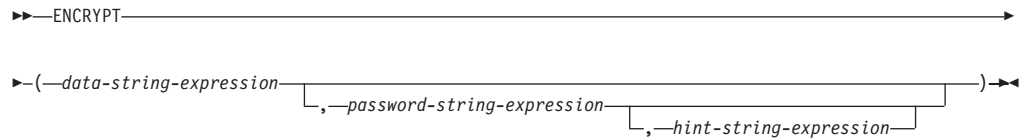
返回的值是 '289-46-8832'。

示例 2: 本示例显式地传送加密口令。

```
SELECT DECRYPT_CHAR(SSN, 'Ben123')
FROM SSN;
```

返回的值是 '289-46-8832'。

ENCRYPT



模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

ENCRYPT 函数返回一个值，这个值是对 *data-string-expression* 进行加密的结果。用于加密的口令是 *password-string-expression* 值或 ENCRYPTION PASSWORD 值（使用 SET ENCRYPTION PASSWORD 语句指定的值）。

data-string-expression

一个表达式，此表达式返回要加密的 CHAR 或 VARCHAR 值。在没有指定 *hint-string-expression* 自变量的情况下，*data-string-expression* 的数据类型的长度属性限制为 32663，当指定了 *hint-string-expression* 自变量时，限制为 32631 (SQLSTATE 42815)。

password-string-expression

一个表达式，此表达式返回至少 6 个字节且不超过 127 个字节的 CHAR 或 VARCHAR 值 (SQLSTATE 428FC)。这个值表示用来对 *data-string-expression* 进行加密的口令。如果口令自变量的值是空，或者未提供，则将使用 ENCRYPTION PASSWORD 值（必须对会话设置这个值）对数据进行加密 (SQLSTATE 51039)。

hint-string-expression

一个表达式，此表达式返回限长 32 个字节的 CHAR 或 VARCHAR 值，这个值有助于数据所有者记忆口令（例如，将 'Ocean' 作为记忆 'Pacific' 的暗示）。如果给出了暗示值，则将暗示嵌入到结果中，并可以使用 GETHINT 函数检索此暗示。如果此自变量为空，或者未提供，则不在结果中嵌入任何暗示。

函数的结果数据类型是 VARCHAR FOR BIT DATA。

结果的长度属性是：

- 当指定了可选的暗示参数时，是未加密数据的长度属性 + 8 个字节 + 到下一个 8 字节边界的字节数 + 作为暗示长度的 32 个字节。
- 未指定暗示参数时，是未加密数据的长度属性 + 8 个字节 + 到下一个 8 字节边界的字节数。

如果第一个自变量可空，则结果可空；如果第一个自变量为空，则结果是空值。

注意，加密结果的长度大于 *data-string-expression* 值。因此，在指定已加密的值时，请确保目标已被声明为具有足够的大小，以包含整个已加密值。

注意事项：

- **加密算法：**使用的内部加密算法是具有填充的 RC2 块密码，128 位的密钥是使用 MD2 消息digest从口令派生的。

- **加密口令和数据:** 用户负责执行口令管理。一旦对数据作了加密, 就只能使用对其进行加密时使用的口令来对其进行解密 (SQLSTATE 428FD)。使用 CHAR 变量设置口令时千万要小心, 这是因为可能会对某些变量填充空白。加密结果可能包含空终止符和其他不可打印字符。

•

表列定义: 在定义要包含已加密数据的列和类型时, 务必按如下方式计算长度属性。对于不带暗示的已加密数据:

未加密数据的最大长度 + 8 个字节 + 到下一个 8 字节边界的字节数 = 已加密数据列长度。

对于嵌入了暗示的已加密数据:

未加密数据的最大长度 + 8 个字节 + 到下一个 8 字节边界的字节数 + 用于暗示长度的 32 个字节 = 已加密数据列长度。

指定给或强制转换为比建议的数据长度短的长度都可能会导致将来解密失败和丢失数据。空白是有效的已加密数据值, 当存储在太短的列中时, 可以将空白截断。

样本列长度计算

未加密数据的最大长度	6 个字节
8 个字节	8 个字节
到下一个 8 字节边界的字节数	2 个字节

已加密数据列长度	16 个字节
未加密数据的最大长度	32 个字节
8 个字节	8 个字节
到下一个 8 字节边界的字节数	8 个字节

已加密数据列长度	48 个字节

- **已加密数据的管理:** 只能在支持与 ENCRYPT 函数相对应的解密函数的服务器上对已加密数据进行解密。因而, 带有已加密数据的列的复制只能对支持 DECRYPT_BIN 或 DECRYPT_CHAR 函数的服务器进行。

另请参见第335页的『DECRYPT_BIN 和 DECRYPT_CHAR』和『GETHINT』, 以了解有关使用此函数的其他信息。

示例:

示例 1: 本示例使用 ENCRYPTION PASSWORD 值来存放加密口令。

```
SET ENCRYPTION PASSWORD = 'Ben123';
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832');
```

示例 2: 本示例显式地传送加密口令。

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832','Ben123',');
```

示例 3: 存储暗示 'Ocean', 以帮助用户记忆 'Pacific' 的加密口令。

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832','Pacific','Ocean');
```

GETHINT

►—GETHINT—(—encrypted-data—)—————►

模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

如果 `GETHINT` 函数在 *encrypted-data* 中找到口令暗示，则此函数将返回该暗示。口令暗示是一个短语，它可以帮助数据所有者记忆口令（例如，将 'Ocean' 作为记忆 'Pacific' 的暗示）。

encrypted-data

一个表达式，此表达式返回 `CHAR FOR BIT DATA` 或 `VARCHAR FOR BIT DATA` 值，这个值是已使用 `ENCRYPT` 函数进行加密的完整、已加密数据字符串 (`SQLSTATE 428FE`)。

此函数的结果为 `VARCHAR(32)`。结果可空；如果 `ENCRYPT` 函数未将暗示参数添加至 *encrypted-data*，或者第一个自变量为空，则结果为空值。

另请参见第335页的『`DECRYPT_BIN` 和 `DECRYPT_CHAR`』和第336页的『`ENCRYPT`』，以了解有关使用此函数的其他信息。

示例:

在此示例中，存储了暗示 'Ocean' 以帮助用户记忆加密口令 'Pacific'。

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832', 'Pacific','Ocean');
SELECT GETHINT(SSN)
FROM EMP;
```

返回的值是 'Ocean'。

IDENTITY_VAL_LOCAL

►—IDENTITY_VAL_LOCAL—(—)—————►

模式为 `SYSIBM`。

此过程是版本 7.1 的修订包 2 中的新增函数。

`IDENTITY_VAL_LOCAL` 函数是一个不确定函数，它返回对一个身份列最近赋予的值，其中，赋值作为使用 `VALUES` 子句的单行 `INSERT` 语句的结果发生。此函数没有任何输入参数。

无论相应身份列的实际数据类型是什么，结果都是 `DECIMAL(31,0)`。

此函数返回的值是赋给最近的单行 `INSERT` 语句中标识的表的身份列的值。该 `INSERT` 语句必须使用 `VALUES` 子句，且必须对包含身份列的表执行。并且，该 `INSERT` 语句必须在相同的级别发出¹。（即，该值可在指定它的级别上以本地方式获得，直到它被下一个指定值替换为止。）

赋给的值或者是用户提供的值（如果身份列定义为 `GENERATED BY DEFAULT` 的话），或者是由 `DB2` 生成的身份值。

在下列情况下，此函数返回一个空值：

1. 每次调用触发器、函数或存储过程时，都会启动新级别

- 如果尚未在当前处理级别对包含身份列的表发出带有 VALUES 子句的单行 INSERT 语句。
- 如果在最近的执行了赋值的 INSERT 语句后发生了工作单元的 COMMIT 或 ROLLBACK²。

此函数的结果不受下列各项的影响:

- 对不带身份列的表执行的带有 VALUES 子句的单行 INSERT 语句。
- 带有 VALUES 子句的多行 INSERT 语句。
- 带有全查询的 INSERT 语句。
- ROLLBACK TO SAVEPOINT 语句。

注意事项:

- INSERT 语句 VALUES 子句中的表达式是在对指定 INSERT 语句的目标列之前进行求值的。因此, 在 INSERT 语句的 VALUES 子句中调用 IDENTITY_VAL_LOCAL 函数将使用前一 INSERT 语句对身份列赋予的最后一个值。如果没有在与 IDENTITY_VAL_LOCAL 相同的级别中执行的、用于包含身份列的表的带有 VALUES 子句的前一行 INSERT 语句, 则此函数返回空值。
- 对于对其定义了触发器的表的身份列值, 可在触发器内通过引用该身份列的触发器事务变量来确定。
- 从插入触发器的触发条件中调用 IDENTITY_VAL_LOCAL 函数的结果是空值。
- 对一个表有可能存在多个插入前或插入后触发器。在这种情况下, 每个触发器都单独地进行处理, 一个被触发操作所指定的身份值不可用于其他使用 IDENTITY_VAL_LOCAL 函数的被触发操作。即使在同一级别上概念性地定义多个被触发操作, 情况亦如此。
- 通常不建议在插入前触发器的主体中使用 IDENTITY_VAL_LOCAL 函数。从插入前触发器的被触发操作中调用 IDENTITY_VAL_LOCAL 函数的结果是空值。不能通过在插入前触发器的被触发操作中调用 IDENTITY_VAL_LOCAL 函数来获取对其定义了触发器的表的身份列的值。然而, 通过引用身份列的触发器过渡变量, 可以在被触发操作中获取身份列的值。
- 在插入后触发器的被触发操作中调用 IDENTITY_VAL_LOCAL 函数的结果³是赋给在同一被触发操作中调用的最近单行 INSERT 语句中标识的表的身份列的值, 这个 INSERT 语句带有 VALUES 子句, 用于包含身份列的表。在调用 IDENTITY_VAL_LOCAL 函数之前, 如果没有在同一被触发操作中执行用于包含身份列的表的带有 VALUES 子句的单行 INSERT 语句, 则此函数返回空值。
- 因为 IDENTITY_VAL_LOCAL 函数的结果不确定, 所以在一个游标的 SELECT 语句中调用 IDENTITY_VAL_LOCAL 函数的结果对每个 FETCH 语句都可能有所变化。
- 指定值就是实际对身份列指定的值(即, 将在后续 SELECT 语句上返回的值)。此值不必一定是在 INSERT 语句的 VALUES 子句中提供的值, 或是由 DB2 生成的值。指定值可以是插入前触发器的主体中的 SET transition variable 语句对与身份列相关的触发器过渡变量指定的值。
- 在带有 VALUES 子句的单行 INSERT 向带有身份列的表中的插入操作失败之后, 函数返回的值是不可预测的。如果函数是在 INSERT 失败之前调用的, 则该值可能是从

2. 除非关闭自动提交, 否则, 当在单独的语句中调用此函数时, 在每个语句之后自动执行提交的接口将返回空值

3. 这适用于 FOR EACH ROW 和 FOR EACH STATEMENT 插入后触发器。

该函数返回的值，或者，如果 INSERT 成功的话，它有可能是已指定的值。返回的实际值取决于失败的时间，因此，它是不可预测的。

示例:

示例 1: 将变量 IVAR 设置为对 EMPLOYEE 表中的身份列指定的值。如果这是第一次对 EMPLOYEE 表的插入，则 IVAR 的值将为 1。

```
CREATE TABLE EMPLOYEE
(EMPNO  INTEGER GENERATED ALWAYS AS IDENTITY,
 NAME CHAR(30),
 SALARY DECIMAL(5,2),
 DEPTNO SMALLINT)
```

示例 2: INSERT 语句中调用的 IDENTITY_VAL_LOCAL 函数返回与用于带有身份列的表的前一个带有 VALUES 子句的单行 INSERT 语句相关联的值。对于此示例，假定有两个表 T1 和 T2。T1 和 T2 都具有名为 C1 的身份列。DB2 为 T1 表中的 C1 列从 1 开始按顺序生成值，并为 T2 表中的 C1 列从 10 开始按顺序生成值。

```
CREATE TABLE T1
(C1 INTEGER GENERATED ALWAYS AS IDENTITY,
 C2 INTEGER),
CREATE TABLE T2
(C1 DECIMAL(15,0) GENERATED BY DEFAULT AS IDENTITY
 (START WITH 10),
 C2 INTEGER),
INSERT INTO T1 (C2) VALUES (5),
INSERT INTO T1 (C2) VALUES (6),
SELECT * FROM T1
```

结果是:

C1	C2
1	5
2	6

现在，对变量 IVAR 声明此函数:

```
VALUES IDENTITY_VAL_LOCAL() INTO :IVAR
```

此时，IDENTITY_VAL_LOCAL 函数将在 IVAR 中返回值 2，因为这就是 DB2 最近指定的值。以下 INSERT 语句将单一行插入到 T2 中，其中，C2 列从 IDENTITY_VAL_LOCAL 函数获取值 2。

```
INSERT INTO T2 (C2) VALUES (IDENTITY_VAL_LOCAL());
SELECT * FROM T2
WHERE C1 = DECIMAL(IDENTITY_VAL_LOCAL(),15,0)
```

返回的结果是:

C1	C2
10.	2

在这个插入操作后调用 IDENTITY_VAL_LOCAL 函数将生成值 10，这是 DB2 为 T2 的 C1 列生成的值。

在涉及触发器的嵌套环境中，使用 IDENTITY_VAL_LOCAL 函数可检索在特定级别指定的身份值，即使可能已在更低的级别指定了身份值。假定有三个表

EMPLOYEE、EMP_ACT 和 ACCT_LOG。EMPLOYEE 上定义了一个插入后触发器，此触发器生成对 EMP_ACT 和 ACCT_LOG 表的附加插入操作。

```
CREATE TABLE EMPLOYEE
(EMPNO SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 1000),
 NAME CHAR(30),
 SALARY DECIMAL(5,2),
 DEPTNO SMALLINT);

CREATE TABLE EMP_ACT
(ACNT_NUM SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 1),
 EMPNO SMALLINT);

CREATE TABLE ACCT_LOG
(ID SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 100),
 ACNT_NUM SMALLINT,
 EMPNO SMALLINT);

CREATE TRIGGER NEW_HIRE
AFTER INSERT ON EMPLOYEE
REFERENCING NEW AS NEW_EMP
FOR EACH ROW MODE DB2SQL
BEGIN ATOMIC
  INSERT INTO EMP_ACT (EMPNO)
  VALUES (NEW_EMP.EMPNO);
  INSERT INTO ACCT_LOG (ACNT_NUM EMPNO)
  VALUES (IDENTITY_VAL_LOCAL(), NEW_EMP.EMPNO);
END
```

触发的第一个 INSERT 语句将一行插入到 EMP_ACT 表中。这个 INSERT 语句使用 EMPLOYEE 表的 EMPNO 列的触发器过渡变量来指示要将 EMPLOYEE 表的 EMPNO 列的身份值复制至 EMP_ACT 表的 EMPNO 列。不能使用 IDENTITY_VAL_LOCAL 函数来获取对 EMPLOYEE 表的 EMPNO 列指定的值。这是因为尚未在这个嵌套级别中发出 INSERT 语句，同样，如果在 INSERT 的 VALUES 子句中对 EMP_ACT 调用 IDENTITY_VAL_LOCAL 函数，此函数将返回空值。这个用于 EMP_ACT 表的 INSERT 语句还导致为 ACNT_NUM 列生成新的身份列值。

触发的第二个 INSERT 语句将一行插入到 ACCT_LOG 表中。这个语句调用 IDENTITY_VAL_LOCAL 函数，以指示要将在被触发操作中的前一 INSERT 语句中对 EMP_ACT 表的 ACNT_NUM 列指定的身份值复制至 ACCT_LOG 表的 ACNT_NUM 列。对 EMPNO 列指定与 EMPLOYEE 表的 EMPNO 列相同的值。

从执行调用的应用程序（即，对 EMPLOYEE 发出 INSERT 的级别）中，将变量 IVAR 设置为最初的 INSERT 语句对 EMPLOYEE 表的 EMPNO 列指定的值。

```
INSERT INTO EMPLOYEE (NAME, SALARY, DEPTNO)
VALUES ('Rupert', 989.99, 50);
```

在处理最初的 INSERT 语句和所有的被触发操作之后，三个表的内容是：

```
SELECT EMPNO, SUBSTR(NAME,10) AS NAME, SALARY, DEPTNO
FROM EMPLOYEE;
```

EMPNO	NAME	SALARY	DEPTNO
1000	Rupert	989.99	50

```
SELECT ACNT_NUM, EMPNO
FROM EMP_ACT;
```

ACNT_NUM	EMPNO
1	1000

```

1          1000
SELECT * FROM ACCT_LOG;

ID          ACNT_NUM  EMPNO
-----
100          1          1000

```

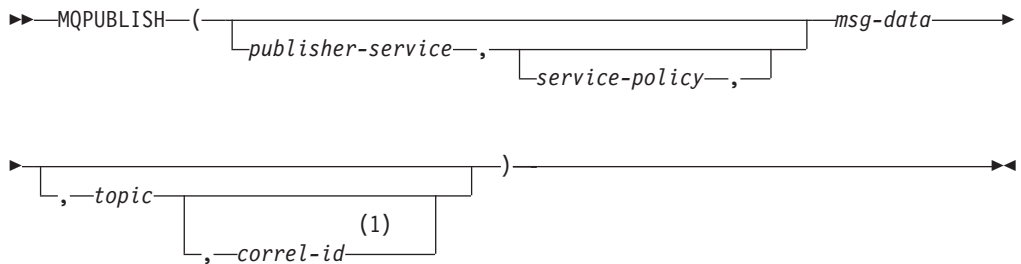
IDENTITY_VAL_LOCAL 函数的结果是在同一嵌套级别对身份列最近指定的值。在处理最初的 INSERT 语句以及所有被触发操作之后，IDENTITY_VAL_LOCAL 函数返回值 1000，因为这是对 EMPLOYEE 表的 EMPNO 列指定的值。接着的 VALUES 语句导致将 IVAR 设置为 1000。对 EMP_ACT 表执行的插入（在对 EMPLOYEE 表的插入之后发生，并且在较低的嵌套级别发生）对 IDENTITY_VAL_LOCAL 函数的本次调用的返回值没有影响。

```
VALUES IDENTITY_VAL_LOCAL() INTO :IVAR;
```

LCASE 和 UCASE (Unicode)

在 Unicode 数据库中，Unicode 字符的整个指令表都是大写（或小写），这取决于这些字符的 Unicode 特性。ASCII 字符的倍宽版本以及罗马数字现在都正确地转换为大写或小写字母。

MQPUBLISH



注:

1 不能指定 *correl-id*，除非先前定义了 *service* 和 *policy*。

模式为 MQDB2。

MQPUBLISH 函数将数据发布给 MQSeries。此函数要求安装“MQSeries 发布 / 预订”或“MQSeries 集成器”。请参考 www.ibm.com/software/MQSeries 以获取更详细的信息。

MQPUBLISH 函数将 *msg-data* 中包含的数据发布给在 *publisher-service* 中指定的 MQSeries 发布方，并使用由 *service-policy* 定义的服务质量策略。可以指定消息的可选主题，还可以指定可选的用户定义消息相关标识符。如果成功的话，该函数将返回值 '1'，如果失败，则返回 '0'。

publisher-service

是包含要将消息发送到的逻辑 MQSeries 目的地的字符串。如果指定了该字符串，则 *publisher-service* 必须是指 AMT.XML 库文件中定义的发布方“服务点”。服务点是一个逻辑端点，可以在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息，参见 MQSeries Application Messaging Interface。如果未指定 *publisher-service*，则将使用 DB2.DEFAULT.PUBLISHER。*publisher-service* 的最大大小为 48 个字符。

service-policy

一个字符串，包含在处理此消息时要使用的“MQSeries AMI 服务策略”。如果指定了该字符串，则 *service-policy* 必须是指在 AMT.XML 库文件中定义的“策略”。

“服务策略”定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 *service-policy*，则将使用缺省值 DB2.DEFAULT.POLICY。*service-policy* 的最大大小为 48 个字符。

msg-data

一个字符串表达式，包含要通过 MQSeries 来发送的数据。最大大小为 4000 个字符。

topic

一个字符串表达式，包含消息发布的主题。如果未指定主题，则没有任何主题将与该消息相关联。*topic* 的最大大小为 40 个字符。在一个字符串中可以指定多个主题（最长可达 40 个字符）。必须用冒号将每个主题隔开。例如，“t1:t2:the third topic”指示消息与所有三个主题 t1、t2 和 “the third topic” 相关联。

correl-id

一个可选的字符串表达式，包含要与此消息相关联的相关标识符。*correl-id* 通常是在请求和答复方案中指定的，以将请求与答复相关联。如果未指定该表达式，则不会向消息添加相关标识。*correl-id* 的最大大小为 24 个字符。

示例

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来将字符串 “Testing 123” 发布至缺省发布方服务 (DB2.DEFAULT.PUBLISHER)。没有为消息指定相关标识符或主题。

```
VALUES MQPUBLISH('Testing 123')
```

示例 2: 此示例将字符串 “Testing 345” 发布至主题 “TESTS” 下的发布方服务 “MYPUBLISHER”。使用了缺省策略，但是未指定相关标识符。

```
VALUES MQPUBLISH('MYPUBLISHER','Testing 345','TESTS')
```

示例 3: 此示例使用具有相关标识符 “TEST1” 的策略 “MYPOLICY” 来将字符串 “Testing 678” 发布至发布方服务 “MYPUBLISHER”。消息是使用主题 “TESTS” 来发布的。

```
VALUES MQPUBLISH('MYPUBLISHER','MYPOLICY','Testing 678','TESTS','TEST1')
```

示例 4: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来将字符串 “Testing 901” 发布主题 “TESTS” 下的发布方服务 “MYPUBLISHER”，且没有任何相关标识符。

```
VALUES MQPUBLISH('Testing 901','TESTS')
```

如果成功的话，所有示例都将返回值 ‘1’。

MQREAD

```
MQREAD—(—  
           |  
           |—receive-service—|  
           |  
           |—, —service-policy—|  
           |  
           )—
```

模式为 MQDB2。

MQREAD 函数通过使用在 *service-policy* 中定义的服务质量策略来从由 *receive-service* 指定的 MQSeries 位置处返回消息。执行此操作不会从与 *receive-service* 相关联的队列中除去消息，而是在队列开头返回消息。返回值是包含该消息的 VARCHAR(4000)。如果没有要返回的消息，则将返回 NULL。

receive-service

是一个字符串，包含要从其中接收消息的逻辑 MQSeries 目的地。如果指定了该字符串，则 *receive-service* 必须是指定在 AMT.XML 库文件中定义的“服务点”。服务点是一个逻辑端点，可以在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息，参见 MQSeries Application Messaging Interface。如果未指定 *receive-service*，则将使用 DB2.DEFAULT.SERVICE。*receive-service* 的最大大小为 48 个字符。

service-policy

一个字符串，包含在处理此消息时所使用的“MQSeries AMI 服务策略”。如果指定了该字符串，则 *service-policy* 必须是指定在 AMT.XML 库文件中定义的“策略”。

“服务策略”定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 *service-policy*，则将使用缺省值 DB2.DEFAULT.POLICY。*service-policy* 的最大大小为 48 个字符。

示例:

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来读取缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列开头的消息。

VALUES MQREAD()

示例 2: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来读取服务 "MYSERVICE" 指定的队列开头的消息。

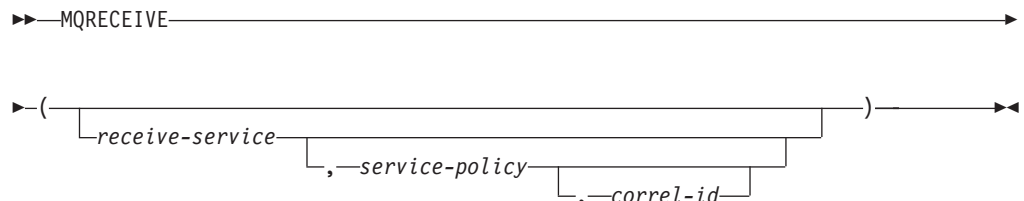
VALUES MQREAD('MYSERVICE')

示例 3: 此示例使用策略 "MYPOLICY" 来读取服务 "MYSERVICE" 指定的队列开头的消息。

VALUES MQREAD('MYSERVICE', 'MYPOLICY')

如果成功的话，所有这些示例都会将消息的内容作为 VARCHAR(4000) 返回。如果没有任何消息，则将返回 NULL。

MQRECEIVE



模式为 MQDB2。

MQRECEIVE 函数通过使用服务质量策略 *service-policy* 来从由 *receive-service* 指定的 MQSeries 位置处返回消息。执行此操作时将从与 *receive-service* 相关联的队列中除去消

息。如果指定了 *correl-id*，则将返回具有匹配的相关标识符的第一条消息。如果未指定 *correl-id*，则将返回队列开头的消息。返回值是包含该消息的 VARCHAR(4000)。如果没有要返回的消息，则将返回 NULL。

receive-service

是一个字符串，包含从其中接收消息的逻辑 MQSeries 目的地。如果指定了该字符串，则 *receive-service* 必须是指 AMT.XML 库文件中定义的“服务点”。服务点是一个逻辑端点，可以在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息，参见 MQSeries Application Messaging Interface。如果未指定 *receive-service*，则使用 DB2.DEFAULT.SERVICE。*receive-service* 的最大大小为 48 个字符。

service-policy

一个字符串，包含在处理此消息时要使用的“MQSeries AMI 服务策略”。如果指定了该字符串，则 *service-policy* 必须是指 AMT XML 库文件中定义的“策略”⁴。如果未指定 *service-policy*，则使用缺省值 DB2.DEFAULT.POLICY。*service-policy* 的最大大小为 48 个字符。

correl-id

一个字符串，包含要与此消息相关联的可选相关标识符。*correl-id* 通常是在请求和答复方案中指定的，以将请求与答复相关联。如果未指定该字符串，则将不会指定相关标识。*correl-id* 的最大大小为 24 个字符。

示例:

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列开头的消息。

VALUES MQRECEIVE()

示例 2: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收服务 "MYSERVICE" 指定的队列开头的消息。

VALUES MQRECEIVE('MYSERVICE')

示例 3: 此示例使用策略 "MYPOLICY" 来接收服务 "MYSERVICE" 指定的队列开头的消息。

VALUES MQRECEIVE('MYSERVICE', 'MYPOLICY')

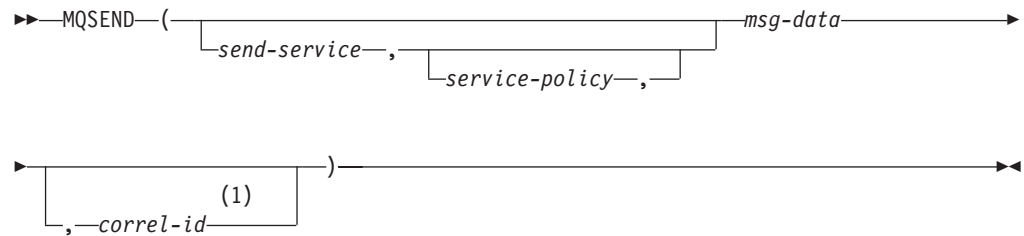
示例 4: 此示例使用策略 "MYPOLICY" 来从服务 "MYSERVICE" 指定的队列开头接收相关标识与 '1234' 相匹配的第一条消息。

VALUES MQRECEIVE('MYSERVICE', 'MYPOLICY', '1234')

如果成功的话，所有这些示例都会将消息的内容作为 VARCHAR(4000) 返回。如果没有任何消息，则将返回 NULL。

MQSEND

4. “服务策略”定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。



注:

1 不能指定 *correl-id*, 除非先前定义了 *service* 和 *policy*。

模式为 MQDB2。

MQSEND 函数使用 *service-policy* 定义的服务质量策略来将 *msg-data* 中包含的数据发送至 *send-service* 指定的 MQSeries 位置。可用 *correl-id* 来指定可选的用户定义消息相关标识符。如果成功的话, 该函数将返回值 '1', 如果失败, 则返回 '0'。

msg-data

一个字符串表达式, 包含要通过 MQSeries 来发送的数据。最大大小为 4000 个字符。

send-service

是包含要将消息发送至的逻辑 MQSeries 目的地的字符串。如果指定了该字符串, 则 *send-service* 是指在 AMT.XML 库文件中定义的服务点。服务点是一个逻辑端点, 可在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息, 参见 MQSeries Application Messaging Interface 手册。如果未指定 *send-service*, 则使用值 DB2.DEFAULT.SERVICE。 *send-service* 的最大大小为 48 个字符。

service-policy

一个字符串, 包含在处理此消息时所使用的“MQSeries AMI 服务策略”。如果指定了该字符串, 则 *service-policy* 必须是指定在 AMT XML 库文件中定义的服务策略。“服务策略”定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息, 参见 MQSeries Application Messaging Interface 手册。如果未指定 *service-policy*, 则将使用缺省值 DB2.DEFAULT.POLICY。 *service-policy* 的最大大小为 48 个字符。

correl-id

一个可选的字符串, 包含与此消息相关联的相关标识符。 *correl-id* 通常是在请求和答复方案中指定的, 以将请求与答复相关联。如果未指定该字符串, 则将不会发送相关标识。 *correl-id* 的最大大小为 24 个字符。

示例:

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来将字符串 "Testing 123" 发送至缺省服务 (DB2.DEFAULT.SERVICE), 且不带相关标识符。

```
VALUES MQSEND('Testing 123')
```

示例 2: 此示例使用策略 "MYPOLICY" 来将字符串 "Testing 345" 发送至服务 "MYSERVICE", 且不带相关标识符。

```
VALUES MQSEND('MYSERVICE','MYPOLICY','Testing 345')
```

示例 3: 此示例使用策略 "MYPOLICY" 来将字符串 "Testing 678" 发送至服务 "MYSERVICE", 且不带相关标识符 "TEST3"。

```
VALUES MQSEND('MYSERVICE','MYPOLICY','Testing 678','TEST3')
```

示例 4: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来将字符串 "Testing 901" 发送至服务 "MYSERVICE", 且不带相关标识符。

```
VALUES MQSEND('MYSERVICE','Testing 901')
```

如果成功的话, 所有示例都将返回标量值 '1'。

MQSUBSCRIBE

```
MQSUBSCRIBE( ( subscriber-service, service-policy, topic ) )
```

模式为 MQDB2。

MQSUBSCRIBE 函数用来注册是否对针对指定主题发布的 MQSeries 消息有兴趣。*subscriber-service* 指定与指定主题相匹配的消息的逻辑目的地。与 *topic* 相匹配的消息将被放置在 *subscriber-service* 定义的队列上, 并且可以通过对 MQREAD、MQRECEIVE、MQREADALL 或 MQRECEIVEALL 的后续调用来读取或接收这些消息。此函数要求安装和配置基于 MQSeries 的发布和预订系统, 例如 “MQSeries 集成器” 或 “MQSeries 发布 / 预订”。有关更详细的信息, 参见 www.ibm.com/software/MQSeries。

如果成功的话, 该函数将返回值 '1', 如果失败, 则返回 '0'。如果成功地执行此函数, 则会导致发布和预订服务器将与该主题相匹配的消息转发至 *subscriber-service* 定义的服务点。

subscriber-service

一个字符串, 包含将与 *topic* 匹配的消息发送至的逻辑 MQSeries 预订点。如果指定了该字符串, 则 *subscriber-service* 必须是指 AMT.XML 库文件中定义的 “订户服务点”。服务点定义包括 “MQSeries 队列管理器和队列” 的名称。有关更详细的信息, 参见 MQSeries Application Messaging Interface 手册。如果未指定 *subscriber-service*, 则将转而使用 DB2.DEFAULT.SUBSCRIBER。*subscriber-service* 的最大大小为 48 个字符。

service-policy

一个字符串, 包含在处理消息时要使用的 “MQSeries AMI 服务策略”。如果指定了该字符串, 则 *service-policy* 必须是指 AMT.XML 库文件中定义的 “策略”。“服务策略” 定义要应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息, 参见 MQSeries Application Messaging Interface 手册。如果未指定 *service-policy*, 则将转而使用缺省值 DB2.DEFAULT.POLICY。*service-policy* 的最大大小为 48 个字符。

topic

一个字符串, 用来定义要接收的消息的类型。此预订将只接收对指定主题所发布的消息。多个预订可以共存。*topic* 的最大大小为 40 个字符。在一个字符串中可以指

定多个主题（最长可达 40 个字符）。必须用冒号将每个主题隔开。例如，“t1:t2:the third topic” 指示消息与所有三个主题 t1、t2 和 “the third topic” 相关联。

示例:

示例 1: 此示例注册是否对包含主题 “Weather” 的消息感兴趣。缺省 subscriber-service (DB2.DEFAULT.SUBSCRIBER) 被注册为订户，而缺省 service-policy (DB2.DEFAULT.POLICY) 指定服务质量。

```
VALUES MQSUBSCRIBE('Weather')
```

示例 2: 此示例演示一个订户注册对包含 “Stocks” 的消息感兴趣。订户使用策略 “BASIC-POLICY” 注册为 “PORTFOLIO-UPDATES”。

```
VALUES MQSUBSCRIBE('PORTFOLIO-UPDATES','BASIC-POLICY','Stocks')
```

如果成功的话，所有示例都将返回标量值 '1'。

MQUNSUBSCRIBE

```
MQUNSUBSCRIBE( ( subscriber-service, service-policy, topic ) )
```

模式为 MQDB2。

MQUNSUBSCRIBE 函数用来取消注册现存消息预订。subscriber-service、service-policy 和 topic 用来标识取消了哪个预订。此函数要求安装和配置基于 MQSeries 的发布和预订系统，例如 “MQSeries 集成器” 或 “MQSeries 发布 / 预订”。有关更详细的信息，参见 www.ibm.com/software/MQSeries。

如果成功的话，该函数将返回值 '1'，如果失败，则返回 '0'。成功执行此函数的结果是：发布和预订服务器将除去由给定的参数所定义的预订。具有指定 topic 的消息将不再发送至 subscriber-service 定义的逻辑目的地。

subscriber-service

如果指定了该字符串，则 subscriber-service 必须是指 AMT.XML 库文件中定义的 “订户服务点”。服务点定义包括 “MQSeries 队列管理器和队列” 的名称。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 subscriber-service，则使用 DB2.DEFAULT.SUBSCRIBER 值。subscriber-service 的最大大小为 48 个字符。

service-policy

如果指定了该字符串，则 service-policy 必须是指 AMT.XML 库文件中定义的 “策略”。“服务策略” 定义要应用于此消息传递操作的一组服务质量选项。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 service-policy，则将使用缺省值 DB2.DEFAULT.POLICY。service-policy 的最大大小为 48 个字符。

topic

一个字符串，指定未接收到的消息的主题。topic 的最大大小为 40 个字符。在一个

字符串中可以指定多个主题（最长可达 40 个字符）。必须用冒号将每个主题隔开。
例如，“t1:t2:the third topic” 指示消息与所有三个主题 t1、t2 和 “the third topic” 相关联。

示例:

示例 1: 此示例取消对包含主题 “Weather” 的消息的兴趣。缺省 subscriber-service (DB2.DEFAULT.SUBSCRIBER) 被注册为非订户，而缺省 service-policy (DB2.DEFAULT.POLICY) 指定服务质量。

```
VALUES MQUNSUBSCRIBE('Weather')
```

示例 2: 此示例演示一个订户取消对包含 “Stocks” 的消息的兴趣。订户使用策略 “BASIC-POLICY” 注册为 “PORTFOLIO-UPDATES”。

```
VALUES MQUNSUBSCRIBE('PORTFOLIO-UPDATES','BASIC-POLICY','Stocks')
```

如果成功的话，这些示例将返回标量值 '1'，如果不成功，则将返回标量值 '0'。

MULTIPLY_ALT

►—MULTIPLY_ALT—(*exact_numeric_expression*, *exact_numeric_expression*)—►

模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

MULTIPLY_ALT 标量函数以十进制值的形式返回两个自变量的乘积。它是作为乘法运算符的替代项提供的，特别是自变量的精度和超过 31 时。

自变量可以是任何内置精确数字数据类型（DECIMAL、BIGINT、INTEGER 或 SMALLINT）。

此函数的结果为 DECIMAL。确定结果的精度和小数位的方法如下：使用符号 p 和 s 来表示第一个自变量的精度和小数位，使用符号 p' 和 s' 来表示第二个自变量的精度和小数位。

- 精度是 $\text{MIN}(31, p + p')$
- 小数位是：
 - 0，若两个自变量的小数位都为 0 的话
 - $\text{MIN}(31, s+s')$ ，若 $p+p'$ 小于或等于 31 的话
 - $\text{MAX}(\text{MIN}(3, s+s'), 31-(p-s+p'-s'))$ ，若 $p+p'$ 大于 31 的话。

若至少有一个自变量可以为空，或者配置数据库时将 DFT_SQLMATHWARN 设置为 “是”，则结果可以为空；其中一个若自变量为空，则结果为空值。

当在至少需要 3 个小数位，以及在精度的总和超过 31 的情况下执行十进制算术运算时，相对于乘法运算符来说，MULTIPLY_ALT 函数是更好的选择。在这些情况下，由于是在内部进行计算，因此避免了溢出。然后，最终结果被指定为使用截断的结果数据类型，其中需要匹配小数位。注意，当小数位为 3 时，最终结果仍然可能溢出。

以下是使用 MULTIPLY_ALT 和乘法运算符时的结果类型的样本比较。

自变量 1 的类型	自变量 2 的类型	使用 MULTIPLY_ALT 时的结果	使用乘法时的结果
DECIMAL(31,3)	DECIMAL(15,8)	DECIMAL(31,3)	DECIMAL(31,11)
DECIMAL(26,23)	DECIMAL(10,1)	DECIMAL(31,19)	DECIMAL(31,24)
DECIMAL(18,17)	DECIMAL(20,19)	DECIMAL(31,29)	DECIMAL(31,31)
DECIMAL(16,3)	DECIMAL(17,8)	DECIMAL(31,9)	DECIMAL(31,11)
DECIMAL(26,5)	DECIMAL(11,0)	DECIMAL(31,3)	DECIMAL(31,5)
DECIMAL(21,1)	DECIMAL(15,1)	DECIMAL(31,2)	DECIMAL(31,2)

示例:

两个值相乘，其中，第一个自变量的数据类型是 DECIMAL(26,3)，第二个自变量的数据类型是 DECIMAL(9,8)。结果的数据类型是 DECIMAL(31,7)。

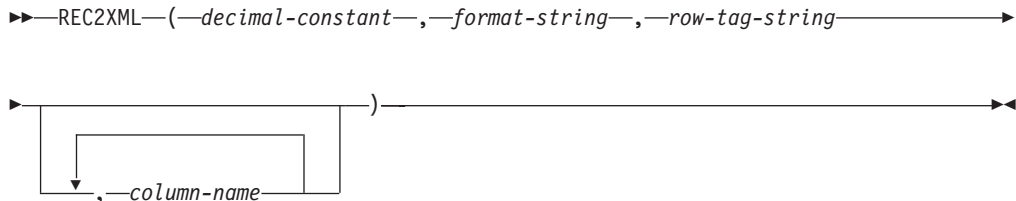
```

值 multiply_alt(98765432109876543210987.654,5.43210987)
1
-----
536504678578875294857887.5277415

```

注意，这两个数的完整乘积应为 536504678578875294857887.52774154498，但是最后 4 位被截断，以便与结果数据类型的小数位相匹配。若对相同的值使用乘法运算符，则会导致运算溢出，因为结果数据类型是 DECIMAL(31,11) 且结果值的小数点左边保留 24 位，而结果数据类型只支持 20 位。

REC2XML



模式为 SYSIBM。

REC2XML 函数将返回利用 XML 标记格式化、并且包含列名和列值的字符串。

decimal-constant

用来替换列值字符的扩展因子。十进制值必须大于 0.0 且小于或等于 6.0 (SQLSTATE 42820)。

decimal-constant 值被用来计算函数的结果。对于具有字符数据类型的每一列，在将该列添加到结果长度中之前，应将该列的长度属性乘以此扩展因子。

要指定不进行扩展，则使用值 1.0。指定小于 1.0 的值将缩短所计算的结果长度。如果结果字符串的实际长度大于函数的计算出的结果长度，则会出现错误 (SQLSTATE 22001)。

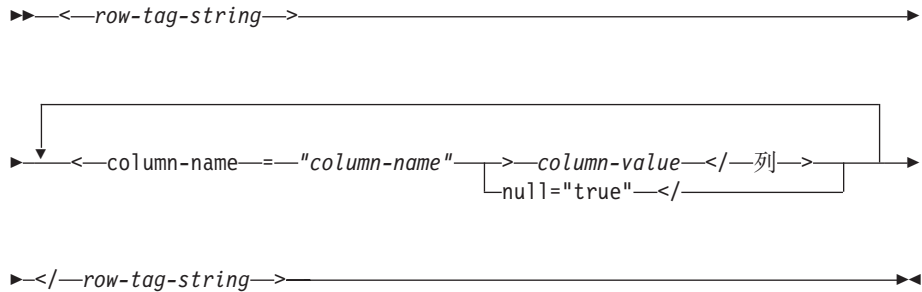
format-string

字符串常量，它指定在执行期间函数将使用哪种格式。

format-string 是区分大小写的，因此，必须用大写来指定下列值才能识别。

COLATTVAL 或 COLATTVAL_XML

这些格式将返回将列作为属性值的字符串。



列名可能是也可能不是有效的 XML 属性值。对于作为无效 XML 属性值的那些列名，对列名执行字符替换，然后才将其包括在结果字符串中。

列值可能是也可能不是有效的 XML 元素值。如果指定了 *format-string* COLATTVAL，对于作为无效 XML 元素值的那些列值，对列值执行字符替换，然后才将其包括在结果字符串中。如果指定了 *format-string* COLATTVAL_XML，则不会对列值执行字符替换（注意，仍然对列名执行字符替换）。

row-tag-string

字符串常量，指定用于每一行的标记。

如果指定了空字符串，则会采用 'row' 的值。

如果指定了具有一个或多个空白字符的字符串，则结果字符串中将不会出现开始 *row-tag-string* 或结束 *row-tag-string*（包括尖括号定界符）。

column-name

表列的限定名或非限定名。该列必须具有下列数据类型之一（SQLSTATE 42815）：

- 数字（SMALLINT、INTEGER、BIGINT、DECIMAL、NUMERIC、REAL、DOUBLE）
- 字符串（CHAR、VARCHAR）⁵
- 日期时间（DATE、TIME、TIMESTAMP）
- 基于以上其中一个类型的用户定义类型

不能多次指定同一列名（SQLSTATE 42734）。

函数的结果是 VARCHAR。最大长度为 32672 个字节（SQLSTATE 54006）。

考虑以下调用：

REC2XML (dc, fs, rt, c₁, c₂, ..., c_n)

如果 **fs** 的值为 "COLATTVAL" 或 "COLATTVAL_XML"

结果与以下表达式相同：

```
'<' CONCAT rt CONCAT '>' CONCAT y1 CONCAT y2 CONCAT ... CONCAT yn
CONCAT '</' CONCAT rt CONCAT '>'
```

其中 y_n 相当于：

5. 不允许具有子类型 BIT DATA 的字符串。

```
'<column name="" CONCAT xvcn CONCAT vn
```

而 v_n 相当于:

```
'">' CONCAT rn CONCAT '</column>'
```

如果该列不为空值, 并且

```
'" null="true"/>'
```

如果列值为空值。

xvc_n 相当于列名 c_n 的字符串表示法, 其中, 表29中出现的任何字符被替换为相应的表示法。这样可以确保所得到的字符串是有效的 XML 属性或者元素值标记。

r_n 相当于表28中所指示的字符串表示法。

结果列值:

根据该列的数据类型以及指定的实际 *format-string*, 在将表中的列值并置到结果字符串中之前可能会转换表中的列值。下表显示了对列值执行的转换。

表 28. 列值字符串结果

数据类型 c_n	r_n
CHAR, VARCHAR	该值为字符串。如果 <i>format-string</i> 不是以字符 "_XML" 结尾, 则 c_n 中的每个字符都会按表29中指示的相应替换表示法进行替换。长度属性为: $dc * c_n$ 的长度属性。
SMALLINT, INTEGER, BIGINT, DECIMAL, NUMERIC, REAL, BINARY, CHAR(c_n))	值为 $CHAR(c_n)$ 。长度属性为结果长度 $CHAR(c_n)$ 。十进制字符始终是句点字符。
DATE	值为 $CHAR(c_n, ISO)$ 。长度属性为结果长度 $CHAR(c_n, ISO)$ 。
TIME	值为 $CHAR(c_n, JIS)$ 。长度属性为结果长度 $CHAR(c_n, JIS)$ 。
TIMESTAMP	值为 $CHAR(c_n)$ 。长度属性为结果长度 $CHAR(c_n)$ 。

字符替换:

根据为 *format-string* 指定的值, 将替换列名和列值中的某些字符。以确保列名组成有效的 XML 属性值, 列值组成有效的 XML 元素值。

表 29. XML 属性值和元素值的字符替换

<	被替换为 <
>	被替换为 >
"	被替换为 "
&	被替换为 &
'	被替换为 '

示例:

- 使用 DEPARTMENT 表，将部门 'D01' 的部门表行 (DEPTNAME 和 LOCATION 列除外) 格式化为有效 XML 的字符串。由于数据中不包含需要替换的任何字符，所以扩展因子将为 1.0 (不扩展)。还要注意，在此行中，MGRNO 值为空值。

```
SELECT REC2XML (1.0, 'COLATTVAL', '', DEPTNO, MGRNO, ADMRDEPT)
FROM DEPARTMENT
WHERE DEPTNO = 'D01'
```

此示例将返回以下 VARCHAR(117) 字符串:

```
<row>
  <column name="DEPTNO">D01</column>
  <column name="MGRNO" null="true"/>
  <column name="ADMRDEPT">A00</column>
</row>
```

注: REC2XML 在输出中不会插入换行字符。对以上示例输出进行了格式化以便于阅读。

- 5 日制大学调度表将名称为 '&43<FIE' 的类引入称为 CL_SCHED 的表，并使 CLASS_CODE 列具有新格式。通过使用 REC2XML 函数，此示例将使用新的类数据来格式化 XML 字符串，但是类结束时间除外。

扩展因子为 1.0 的 REC2XML 调用 (参见下文) 的长度属性为 128 (11 表示 '<row>' 和 '</row>' 开销、21 表示列名长度、75 表示 '<column name='、'>'、'</column>' 和双引号长度、7 表示 CLASS_CODE 数据长度、6 表示 DAY 数据长度、8 表示 STARTING 数据长度)。由于将替换 '&' 和 '<' 字符，所以扩展 1.0 将是不够的。该函数的长度属性将需要支持增加 7 到 14 个字符以用于新格式的 CLASS_CODE 数据。

然而，众所周知，DAY 值将决不会超过 1 位数，因此，决不会用到计算到长度中的余下 5 位。因此，扩展只需要处理增量 2。由于 CLASS_CODE 是自变量列表中唯一的字符串列，所以，这是扩展因子唯一适用于的列值。要使长度增加 2，将需要扩展因子 9/7 (大约等于 1.2857)。将使用扩展因子 1.3。

```
SELECT REC2XML (1.3, 'COLATTVAL', 'record', CLASS_CODE, DAY, STARTING)
FROM CL_SCHED
WHERE CLASS_CODE = '&43<FIE'
```

返回以下字符串:

```
<record>
  <column name="CLASS_CODE">&43<FIE</column>
  <column name="DAY">5</column>
  <column name="STARTING">06:45:00</column>
</record>
```

注: REC2XML 在输出中不会插入换行字符。对以上示例输出进行了格式化以便于阅读。

- 此示例显示在列名中替换的字符。

```
SELECT REC2XML (1.3, 'COLATTVAL', '', Class, "time<noon")
FROM (SELECT Class_code, Starting
      FROM Cl_sched
      WHERE Starting < '12:00:00')
AS Early (Class, "time<noon")
```

此查询将返回以下字符串:

```

<row>
  <column name="CLASS">&43</column>
  <column name="time<noon>06:45:00</column>
</row>

```

注: REC2XML 在输出中不会插入换行字符。对以上示例输出进行了格式化以便于阅读。

- 假定已经向 EMP_RESUME 表中添加了新行，将简历作为有效 XML 的字符串存储起来。因为使用了 COLATTVAL_XML *format-string*，所以将不会执行字符替换。没有任何简历的长度超过 3500 个字符。可以使用以下查询来从 EMP_RESUME 表中选择 XML 版本的简历，并将它格式化为 XML 文档段。

```

SELECT REC2XML (1.0, 'COLATTVAL_XML', '', EMPNO, RESUME_XML)
  FROM (SELECT EMPNO, CAST(RESUME as VARCHAR(3500)) as RESUME_XML
        FROM EMP_RESUME
        WHERE RESUME_FORMAT = 'XML')
AS EMP_RESUME_XML

```

ROUND

►—ROUND—(*expression1*, *expression2*)—◄

模式为 SYSIBM。

此函数是版本 7.1 的修订包 2 中的新增函数。

注: ROUND 函数的 SYSFUN 版本仍然可用。

如果 *expression2* 为正数，则 ROUND 函数将 *expression1* 四舍五入为小数点右边的 *expression2* 位，如果 *expression2* 为零或负数，则 ROUND 函数将 *expression1* 四舍五入为小数点左边的 *expression2* 位。

如果 *expression1* 为正数，则值 5 四舍五入为下一个更大的正数。例如，ROUND(3.5,0) = 4。如果 *expression1* 为负数，则值 5 四舍五入为下一个更小的负数。例如，ROUND(-3.5,0) = -4。

expression1

是返回任何内置数字数据类型的值的表达式。

expression2

返回较小或较大整数的表达式。如果 *expression2* 的值不是负数，则它指定四舍五入为小数分隔符右边的位数。如果 *expression2* 的值为负数，则它指定四舍五入为小数分隔符左边的 *expression2* 位的绝对值。

如果 *expression2* 不是负数，则 *expression1* 四舍五入为小数点右边的 *expression2* 位数的绝对值。如果 *expression2* 的值大于 *expression1* 的小数位，则该值保持不变，除非结果值的精度要增大 1。例如，ROUND(748.58,5) = 748.58，其中，现在精度为 6，而小数位仍然为 2。

如果 *expression2* 为负数，则 *expression1* 四舍五入为小数点左边的 *expression2*+1 位数的绝对值。

如果负数 *expression2* 的绝对值大于小数点左边的位数，则结果为 0。例如，ROUND(748.58,-4) = 0。

如果 *expression1* 为 DECIMAL 或 NUMERIC，并且精度小于 31，则结果的数据类型和长度属性与第一个自变量的数据类型和长度属性相同，但是精度要增加 1。例如，数据类型为 DECIMAL(5,2) 的自变量的结果为 DECIMAL(6,2)。数据类型为 DECIMAL(31,2) 的自变量的结果为 DECIMAL(31,2)。小数位与第一个自变量的小数位是相同的。

若自变量可以为空，或者配置数据库时将 DFT_SQLMATHWARN 设置为“是”，则结果可以为空。若其中一个自变量为空，则结果为空值。

示例： 将数值 873.726 分别四舍五入为 2、1、0、-1、-2、-3 和 -4 个小数位。

```
VALUES (ROUND(873.726, 2),
        ROUND(873.726, 1),
        ROUND(873.726, 0),
        ROUND(873.726, -1),
        ROUND(873.726, -2),
        ROUND(873.726, -3),
        ROUND(873.726, -4) )
```

此示例的结果为:

1	2	3	4	5	6	7
873.730	873.700	874.000	870.000	900.000	1000.000	0.000

计算正数和负数。

```
VALUES (ROUND(3.5, 0),
        ROUND(3.1, 0),
        ROUND(-3.1, 0),
        ROUND(-3.5, 0) )
```

此示例的结果为:

1	2	3	4
4.0	3.0	-3.0	-4.0

WEEK_ISO

将此函数的描述更改为以下内容:

模式为 SYSFUN。

以范围为 1 到 53 内的整数值返回一年中的某个星期这一自变量。每周都是从星期一开始，并且一周始终包括 7 天。第一个星期被定义为一年中包含星期四的首个星期，它相当于包含 1 月 4 日的首个星期。因此，在一年开头最多可以有 3 天属于前一年的最后一个星期。反过来，在一年末尾最多可以有 3 天属于下一年的第一个星期。

自变量必须是日期、时间戳记或者既不是 CLOB 也不是 LONG VARCHAR 的日期或时间戳记的有效字符串表示法。

该函数的结果为 INTEGER。结果可以为空值；如果自变量为空值，则结果为空值。

示例:

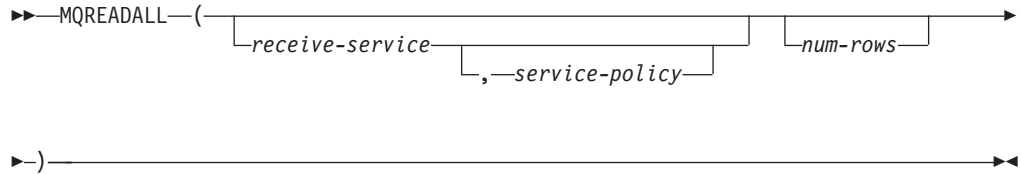
以下列表显示了结果 WEEK_ISO 和 DAYOFWEEK_ISO 的示例。

DATE	WEEK_ISO	DAYOFWEEK_ISO
1997-12-28	52	7

1997-12-31	1	3
1998-01-01	1	4
1999-01-01	53	5
1999-01-04	1	1
1999-12-31	52	5
2000-01-01	52	6
2000-01-03	1	1

表函数

MQREADALL



模式为 MQDB2。

`MQREADALL` 函数通过使用服务质量策略 `service-policy` 来从 `receive-service` 指定的 MQSeries 位置处返回包含消息和消息元数据的表。执行此操作不会从与 `receive-service` 相关联的队列中除去消息。

如果指定了 `num-rows`，则将返回最多为 `num-rows` 的消息。如果未指定 `num-rows`，则将返回所有可用的消息。返回的表包含下列各列：

- `MSG - VARCHAR(4000)` 列，包含 MQSeries 消息的内容。
- `CORRELID - VARCHAR(24)` 列，用来保存用于关联消息的相关标识。
- `TOPIC - VARCHAR(40)` 列，用来保存发布了消息的主题（如果有的话）。
- `QNAME - VARCHAR(48)` 列，用来保存接收到消息的队列名。
- `MSGID - CHAR(24)` 列，用来保存为此消息指定的 MQSeries 唯一标识符。
- `MSGFORMAT - VARCHAR(8)` 列，用来保存 MQSeries 定义的消息的格式。通常字符串具有 `MQSTR` 格式。

receive-service

是一个字符串，包含从其中读取消息的逻辑 MQSeries 目的地。如果指定了该字符串，则 `receive-service` 必须是指定在 `AMT.XML` 库文件中定义的服务点。服务点是一个逻辑端点，可以在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息，参见 `MQSeries Application Messaging Interface`。如果未指定 `receive-service`，则将使用 `DB2.DEFAULT.SERVICE`。`receive-service` 的最大大小为 48 个字符。

service-policy

一个字符串，包含在处理此消息时所使用的“MQSeries AMI 服务策略”。如果指定了该字符串，则 `service-policy` 是指定在 `AMT XML` 库文件中定义的“策略”。服务策略定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息，参见 `MQSeries Application Messaging Interface` 手册。如果未指定 `service-policy`，则将使用缺省值 `DB2.DEFAULT.POLICY`。`service-policy` 的最大大小为 48 个字符。

num-rows

一个正整数，包含要由该函数返回的最大消息数。

示例:

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列中的所有消息。消息和所有元数据都是作为表返回的。

```
SELECT *  
FROM table (MQREADALL()) T
```

示例 2: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收服务 MYSERVICE 指定的队列开头的所有消息。只返回 MSG 和 CORRELID 列。

```
SELECT T.MSG, T.CORRELID  
FROM table (MQREADALL('MYSERVICE')) T
```

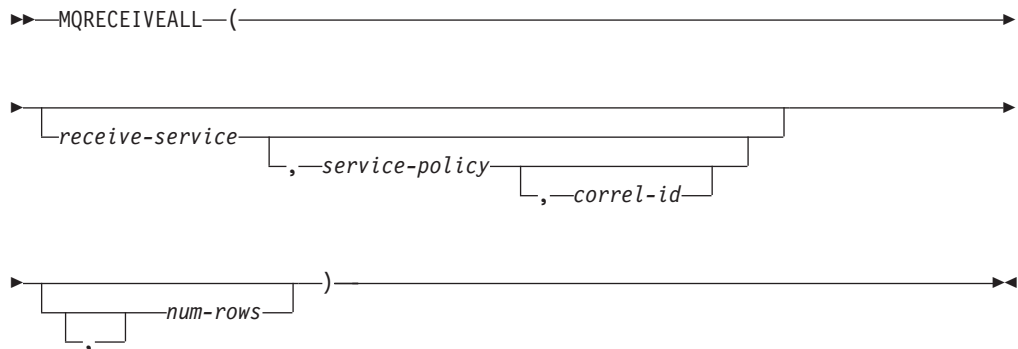
示例 3: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来读取缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列开头。只返回 CORRELID 为 '1234' 的消息。将返回所有列。

```
SELECT *  
FROM table (MQREADALL()) T  
WHERE T.CORRELID = '1234'
```

示例 4: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列的前 10 条消息。将返回所有列。

```
SELECT *  
FROM table (MQREADALL(10)) T
```

MQRECEIVEALL



模式为 MQDB2。

MQRECEIVEALL 函数通过使用服务质量策略 *service-policy* 来从 *receive-service* 指定的 MQSeries 位置处返回包含消息和消息元数据的表。执行此操作时将从与 *receive-service* 相关联的队列中除去消息。

如果指定了 *correl-id*，则将只返回具有匹配的相关标识符的那些消息。如果未指定 *correl-id*，则将返回队列开头的消息。

如果指定了 *num-rows*，则将返回最多为 *num-rows* 的消息。如果未指定 *num-rows*，则返回所有可用的消息。返回的表包含下列各列:

- MSG - VARCHAR(4000) 列，包含 MQSeries 消息的内容。
- CORRELID - VARCHAR(24) 列，用来保存用于关联消息的相关标识。
- TOPIC - VARCHAR(40) 列，用来保存发布了消息的主题（如果有的话）。
- QNAME - VARCHAR(48) 列，用来保存接收到消息的队列名。
- MSGID - CHAR(24) 列，用来保存为此消息指定的 MQSeries 唯一标识符。
- MSGFORMAT - VARCHAR(8) 列，用来保存 MQSeries 定义的消息的格式。通常字符串具有 MQSTR 格式。

receive-service

是一个字符串，包含从其中接收消息的逻辑 MQSeries 目的地。如果指定了该字符串，则 *receive-service* 必须是指在 AMT.XML 库文件中定义的服务点。服务点是一个逻辑端点，可以在其中发送或接收消息。服务点定义包括“MQSeries 队列管理器和队列”的名称。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 *receive-service*，则将使用 DB2.DEFAULT.SERVICE。*receive-service* 的最大大小为 48 个字符。

service-policy

一个字符串，包含在处理此消息时所使用的“MQSeries AMI 服务策略”。如果指定了该字符串，则 *service-policy* 是指在 AMT XML 库文件中定义的“策略”。服务策略定义应该应用于此消息传递操作的一组服务质量选项。这些选项包括消息优先级和消息持久性。有关更详细的信息，参见 MQSeries Application Messaging Interface 手册。如果未指定 *service-policy*，则将使用缺省值 DB2.DEFAULT.POLICY。*service-policy* 的最大大小为 48 个字符。

correl-id

一个可选的字符串，包含与此消息相关联的相关标识符。*correl-id* 通常是在请求和答复方案中指定的，以将请求与答复相关联。如果未指定该字符串，则不会指定相关标识。*correl-id* 的最大大小为 24 个字符。

num-rows

一个正整数，包含要由该函数返回的最大消息数。

示例:

示例 1: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列中的所有消息。消息和所有元数据都是作为表返回的。

```
SELECT *
FROM table (MQRECEIVEALL()) T
```

示例 2: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收服务 MYSERVICE 指定的队列开头的所有消息。只返回 MSG 和 CORRELID 列。

```
SELECT T.MSG, T.CORRELID
FROM table (MQRECEIVEALL('MYSERVICE')) T
```

示例 3: 此示例使用策略 "MYPOLICY" 来接收服务 "MYSERVICE" 指定的队列开头的消息。只返回 CORRELID 为 '1234' 的消息。只返回 MSG 和 CORRELID 列。

```
SELECT T.MSG, T.CORRELID
FROM table (MQRECEIVEALL('MYSERVICE','MYPOLICY','1234')) T
```


示例 4: 此示例使用缺省策略 (DB2.DEFAULT.POLICY) 来接收缺省服务 (DB2.DEFAULT.SERVICE) 指定的队列的前 10 条消息。将返回所有列。

```
SELECT *  
FROM table (MQRECEIVEALL(10)) T
```

过程

GET_ROUTINE_SAR

►► GET_ROUTINE_SAR (—*sarblob*—, —*type*—, —*routine_name_string*—) ►►

模式为 SYSFUN。

此过程是版本 7.1 的修订包 2 中的新增函数。

GET_ROUTINE_SAR 过程检索在另一数据库服务器（该服务器在同一操作系统上运行同一级别）中安装同一例程所需的信息。该信息被检索到单个 BLOB 字符串（该字符串表示一个 SQL 归档文件）中。GET_ROUTINE_SAR 过程的调用者必须具有 DBADM 权限。

sarblob

类型为 BLOB(3M) 的输出自变量，包含例程 SAR 文件的内容。

type

类型为 CHAR(2) 的输入自变量，此自变量使用下列其中一个值指定例程类型：

- **P**，表示过程。
- **SP**，表示过程的特定名称。

routine_name_string

类型为 VARCHAR(257) 的输入自变量，用来指定例程的限定名称。如果未指定模式名，则当处理例程时，缺省值是 CURRENT SCHEMA。

注：*routine_name_string* 不能包含双引号字符 (")。

例程的限定名称用来确定要检索哪个例程。找到的例程必须是 SQL 例程，否则会出错 (SQLSTATE 428F7)。当不使用特定名称时，可能会生成多个例程，并发生错误 (SQLSTATE 42725)。如果发生这种情况，则该例程的指定名称必须用来获取该例程。

SAR 文件必须包括服务器上可能不存在的绑定文件。如果绑定文件找不到，并且没有存储在 SAR 文件中，则会出错 (SQLSTATE 55045)。

PUT_ROUTINE_SAR

►► PUT_ROUTINE_SAR (—*sarblob*—, —*new_owner*—, —*use_register_flag*—) ►►

模式为 SYSFUN。

此过程是版本 7.1 的修订包 2 中的新增函数。

PUT_ROUTINE_SAR 过程用来传送在服务器上创建 SQL 例程所需的文件，并定义该例程。PUT_ROUTINE_SAR 过程的调用者必须具有 DBADM 权限。

sarblob

类型为 BLOB(3M) 的输入自变量，包含例程 SAR 文件的内容。

new_owner

类型为 VARCHAR(128) 的输入自变量，包含用于例程的权限检查的权限名称。*new-owner* 必须对要定义的例程具有必需的特权。如果未指定 *new-owner*，则使用原始例程定义者的授权名称。

use_register_flag

类型为 INTEGER 的输入自变量，用来指示是否使用 CURRENT SCHEMA 和 CURRENT PATH 专用寄存器来定义该例程。如果未使用这两个专用寄存器，则缺省模式和 SQL 路径的设置即在一开始定义该例程时使用的设置。可能的 *use-register-flag* 值包括：

0 不要使用当前环境的专用寄存器

1 使用 CURRENT SCHEMA 和 CURRENT PATH 专用寄存器。

如果该值为 1，则对例程定义（包括例程名）中的未限定对象名使用 CURRENT SCHEMA，而 CURRENT PATH 则用来解析例程定义中的未限定例程和数据类型。

如果未指定 *use-registers-flag*，则行为与指定了 0 值相同。

将会检查 *sarblob* 中包含的标识信息，以确认输入适合于环境，否则会出错 (SQLSTATE 55046)。然后，PUT_ROUTINE_SAR 过程使用 *sarblob* 的内容来定义服务器上的例程。

sarblob 自变量的内容会被抽取到构成 SQL 归档文件的独立文件中。共享库和绑定文件会写入临时目录中的文件。设置了环境，因此例程定义语句处理知道不需要进行编译和链接，且可以获得共享库和绑定文件的设置。然后，会使用 DDL 文件的内容动态执行该例程定义语句。

注：不能以给定模式并行安装多个过程。

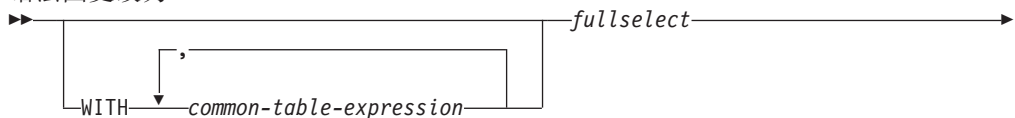
此语句的处理可能会导致在使用其他接口执行例程定义语句时出现的错误。在例程定义处理期间，会指出共享库和绑定文件的存在，而预编译、编译和链接步骤会被跳过。绑定文件是在进行绑定处理时使用的，且两个文件的内容都会被复制至 SQL 例程的常用目录。

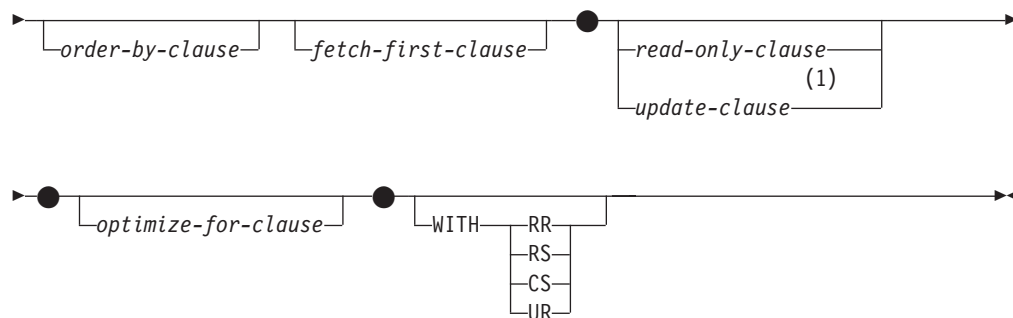
注：如果 GET ROUTINE 或 PUT ROUTINE 操作（或它们的相应过程）未能成功地执行，则始终会返回错误 (SQLSTATE 38000)，并随同返回提供关于故障原因的信息的诊断文本。例如，如果向 GET ROUTINE 提供的过程名未标识 SQL 过程，则会返回诊断 "100, 02000" 文本，其中，"100" 和 "02000" 分别是 SQLCODE 和 SQLSTATE，它们标识问题的原因。此示例中的 SQLCODE 和 SQLSTATE 指示对给定过程名指定的行在目录表中找不到。

Chapter 5. Queries

select-statement 语法图

语法图更改为：





注:

1 不能在同一个 select-statement 中同时指定 update-clause 和 order-by-clause.

将下面这一段添加至语法图下面的描述:

可选的 WITH 子句指定执行选择语句的隔离级别。

- RR - 可重复的读取
- RS - 读取稳定性
- CS - 游标稳定性
- UR - 未提交的读取

语句的缺省隔离级别是在其中绑定语句的软件包的隔离级别。

select-statement/fetch-first-clause

取装首个子句描述中的最后一段:

Specification of the fetch-first-clause in a select-statement makes the cursor not deletable (read-only). This clause cannot be specified with the FOR UPDATE clause.

是不正确的, 应该除去。

Chapter 6. SQL Statements

现在支持更新分区键

现在支持更新分区键。仅当 DB2_UPDATE_PART_KEY=ON 时, 才应该删除第 6 章中各个语句中的以下文本:

注: If DB2_UPDATE_PART_KEY=OFF, then the restrictions still apply.

语句: ALTER TABLE

规则

- 不能更新表中的分区键列 (SQLSTATE 42997)。
- 当关系是用 ON DELETE SET NULL 来定义的时候, 分区键的可空列不能作为外键列包括 (SQLSTATE 42997)。

语句: CREATE TABLE

规则

- 不能更新表中的分区键列 (SQLSTATE 42997)。
- 当关系是用 ON DELETE SET NULL 来定义的时候, 分区键的可空列不能作为外键列包括 (SQLSTATE 42997)。

语句: DECLARE GLOBAL TEMPORARY TABLE PARTITIONING KEY (column-name,...)

注: 不能更新分区键列 (SQLSTATE 42997)。

语句: UPDATE

脚注

- 108 分区键列是不可更新的 (SQLSTATE 42997)。必须删除和插入数据行以更改分区键中的列。

用于 Unicode 数据库的较大型索引键

ALTER TABLE

可以将注册表变量 DB2_INDEX_2BYTEVARLEN 为 ON 时定义的作为任何索引 (包括主索引和唯一索引) 的一部分的变长列的长度改变为大于 255 个字节的长度。外键涉及变长列这一事实不再会导致不能将该列的长度改变为大于 255 个字节, 无论该注册表变量是如何设置的均如此。然而, 不能将长度大于 255 个字节的数据库插入到表中, 除非相应主键中列的长度大于 255 个字节 (如果主键是在该注册表变量为 ON 时创建的, 则这是不可能的)。

CREATE INDEX

如果注册表变量 DB2_INDEX_2BYTEVARLEN 为 ON, 则可以对长度大于 255 个字节的变长列定义索引。

CREATE TABLE

如果注册表变量 DB2_INDEX_2BYTEVARLEN 为 ON, 则带有变量键部件的主键和唯一键的大小可以大于 255。可以对长度大于 255 个字节的变长列定义外键。

ALTER SEQUENCE

ALTER SEQUENCE

ALTER SEQUENCE 语句通过执行下列各项修改序列的属性:

- 重新启动序列
- 更改将来序列值之间的增量
- 设置新的最小或最大值
- 更改高速缓存的序列号的数目
- 更改序列是否可以循环
- 更改序列号是否必须按请求次序生成

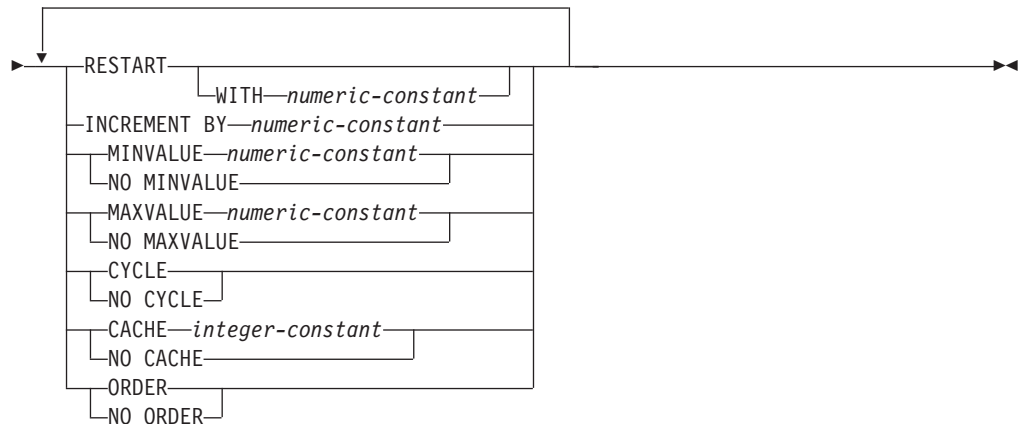
调用: 可以将此语句嵌入在应用程序中,也可以通过使用动态 SQL 语句发出此语句。这是一条可以动态准备的可执行语句。然而,如果应用了绑定选项 DYNAMICRULES BIND,则不能动态准备此语句 (SQLSTATE 42509)。

权限: 此语句的授权标识所拥有的特权必须至少包括下列其中一项:

- 序列的定义者
- 用于隐式或显式指定的模式的 ALTERIN 特权
- SYSADM 或 DBADM 权限

语法:

▶▶ ALTER SEQUENCE *sequence-name* ▶▶



描述:

sequence-name

标识特定的序列。名称与隐式或显式模式名的组合必须标识当前服务器上的现存序列。如果显式或隐式指定的模式中不存在具有此名称的序列,则会发生错误 (SQLSTATE 42704)。

RESTART

重新启动序列。如果未指定 *numeric-constant*,则序列从以显式或隐式方式作为最初创建该序列的 CREATE SEQUENCE 语句上的起始值指定的值重新启动。

WITH *numeric-constant*

使用指定的值重新启动序列。这个值可以是任何可以对此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，但小数点右边不能有非零位 (SQLSTATE 42894)。

INCREMENT BY

指定序列的连续值之间的间隔。这个值可以是任何可以对此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，不超出大整数常量的值 (SQLSTATE 42815)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)。

如果这个值是负数，则值序列是降序的。如果这个值是正数，则值序列是升序的。如果这个值是 0 或大于 MINVALUE 和 MAXVALUE 定义的范围，则只生成一个值，但视该序列为升序序列。

MINVALUE 或 NO MINVALUE

指定最小值，降序序列在遇到这个值时将循环或停止生成值，升序序列在遇到最大值后会循环至这个值。

MINVALUE *numeric-constant*

指定作为最小值的数字常量。这个值可以是任何可以对此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)，但此值必须小于或等于最大值 (SQLSTATE 42815)。

NO MINVALUE

对于升序序列，这个值是 START WITH 值，或者，如果未指定 START WITH 则为 1。对于降序序列，这个值是与该序列相关联的数据类型的最小值。这是缺省值。

MAXVALUE 或 NO MAXVALUE

指定最大值，升序序列在遇到这个值时将循环或停止生成值，降序序列在遇到最小值后会循环至这个值。

MAXVALUE *numeric-constant*

指定作为最大值的数字常量。这个值可以是任何可以对此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 428FA)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)，但此值必须大于或等于最小值 (SQLSTATE 42815)。

NO MAXVALUE

对于升序序列，这个值是与该序列相关联的数据类型的最大值。对于降序序列，这个值是 START WITH 值，或者，如果未指定 START WITH 则为 -1。这是缺省值。

CYCLE 或 NO CYCLE

指定序列在遇到它的最大值或最小值后是否应当继续生成值。到达序列边界的情况有两种，一种情况是下一个值刚好位于边界条件上，另一种情况是越过序列边界，在这种情况下，下一个值将通过回绕至 START WITH 值确定（如果允许循环的话）。

CYCLE

指定在遇到最大值或最小值之后继续为此序列生成值。如果使用此选项，则升序序列在遇到它的最大值之后会生成它的最小值；而降序序列在遇到它的最小值之后会生成它的最大值。序列的最大值和最小值确定了用于循环的范围。

当 CYCLE 生效时，可以为序列生成重复的值。

NO CYCLE

指定在遇到序列的最大值或最小值之后，便不再为序列生成值。这是缺省值。

CACHE 或 NO CACHE

指定是否将一些预先分配的值放在内存中，以加快访问速度。这是一个性能和调整选项。

CACHE *integer-constant*

指定预先分配并存放在内存中的序列值的最大数目。预先分配值并将这些值存储在高速缓存中可以减少在为序列生成值时对日志执行的同步 I/O。

如果发生系统故障，则所有已存入高速缓存但尚未在已提交的语句中使用过的序列值都会丢失（即，永远不再使用它们）。对 CACHE 选项指定的值便是在发生系统故障时可能会丢失的序列值的最大数目。

最小值是 2 (SQLSTATE 42815)。缺省值是 CACHE 20。

NO CACHE

指定不预先分配序列的值。这确保即使发生系统故障、关机或数据库释放也不会丢失值。如果指定此选项，则不将序列的值存储在高速缓存中。在这种情况下，每一个对新序列值的请求都会导致对日志执行同步 I/O 操作。

NO ORDER 或 ORDER

指定是否必须按请求次序生成序列号。

ORDER

指定按请求次序生成序列号。

NO ORDER

指定无需按请求次序生成序列号。这是缺省值。

在重新启动序列或更改为 CYCLE 之后，序列号便有可能是序列先前生成的序列号的重复值。

注意事项:

- ALTER SEQUENCE 语句只影响将来的序列号。
- 不能更改序列的数据类型。而是，您需删除并重新创建该序列，并为新序列指定期望的数据类型。
- 改变序列时，所有已存入高速缓存的值都会丢失。

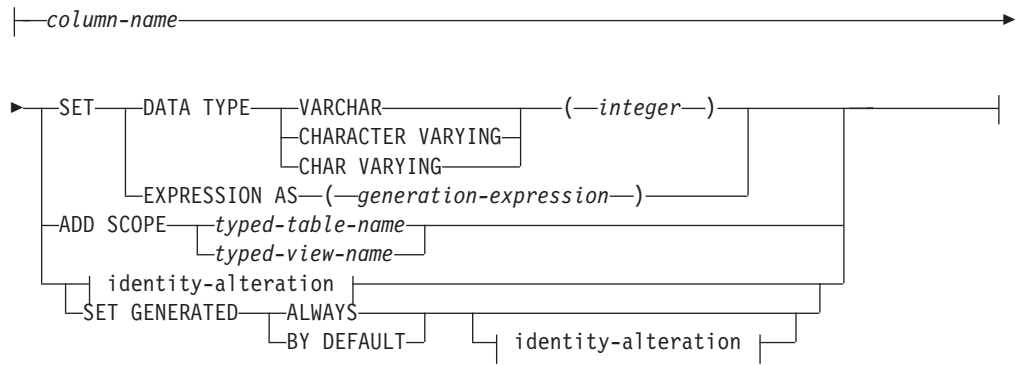
示例: 示例 1: 指定不带数值的 RESTART 的可能原因是将序列复位至 START WITH 值。在此示例中，目标是生成 1 至表中行数的数字，然后将这些数字插入到使用临时表添加至该表的列中。另一种使用是获取结果，在这些结果中，所有结果行都已编号:

```
ALTER SEQUENCE org_seq
RESTART
SELECT NEXTVAL for org_seq, org.*
FROM org
```

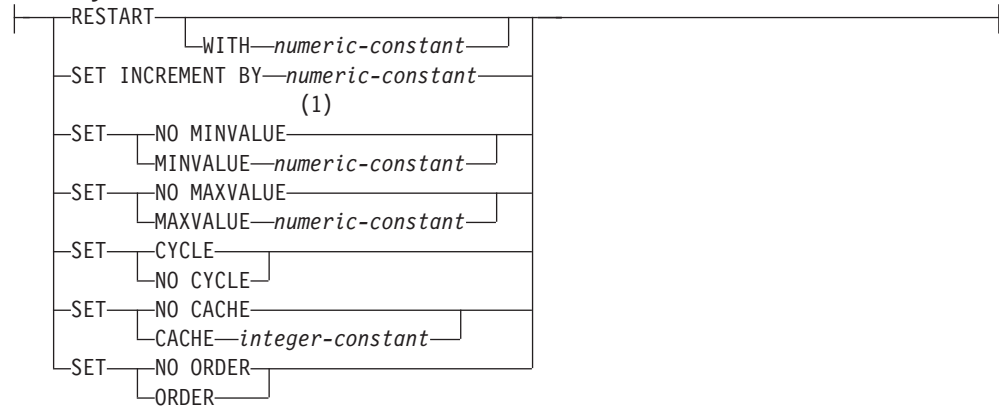
ALTER TABLE

对语法段的更改:

column-alteration:



identity-alteration:



注:

- 1 可以指定这些参数（不带空格）：
NOMINVALUE、NOMAXVALUE、NOCYCLE、NOCACHE 和 NOORDER。这些
单字版本全都是双字版本的可接受的备用项。

添加下列参数:

SET GENERATED

指定是始终为列生成值还是仅当需要缺省值时才生成值。

ALWAYS

在表中插入或更新行时，始终为列生成值。该列必须已定义为已生成列 (SQLSTATE 42837)。

BY DEFAULT

在将行插入表中时，除非已指定一个值，否则将为列生成值。该列必须已定义为已生成列 (SQLSTATE 42837)。

RESTART 或 RESTART WITH numeric-constant

将与身份列相关联的序列的状态复位。如果未指定 **WITH numeric-constant**，则身份列的序列从最初创建该身份列时作为起始值隐式或显示指定的值重新启动。*numeric-constant* 是一个精确的数字常量，它可以是任何可以对此列指定的正数或负数值 (SQLSTATE 42820)，但小数点右边不能有非零位 (SQLSTATE 42894)。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。*numeric-constant* 将被用作该列中的下一个值。

SET INCREMENT BY *numeric-constant*

指定身份列的连续值之间的间隔。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。这个值是什么可以对此列指定的正数或负数值 (SQLSTATE 42820)，并且不超出大整数常量的值 (SQLSTATE 42815)，但小数点右边不能有非零位 (SQLSTATE 42894)。

如果这个值是负数，则这个身份列的值的序列是降序的。如果这个值是正数，则这个身份列的值的序列是升序的。如果这个值是 0，或大于由 **MINVALUE** 和 **MAXVALUE** 定义的范围，则 DB2 将只生成一个值，但视该序列为升序序列。

SET MINVALUE *numeric-constant* 或 NO MINVALUE

指定最小值，降序身份列在遇到这个值时将循环或停止生成值，或指定升序身份列在遇到最大值后将循环所至的值。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。

MINVALUE *numeric-constant*

指定最小数字常量值。这个值可以是任何可以对此列指定的正数值或负数值 (SQLSTATE 42820)，并且，在小数点右边没有非零位 (SQLSTATE 42894)，但此值必须小于最大值 (SQLSTATE 42815)。

NO MINVALUE

对于升序序列，这个值是 **START WITH** 值，或者，如果未指定 **START WITH** 则为 1。对于降序序列，这个值是该列的数据类型的最小值。

SET MAXVALUE *numeric-constant* or NO MAXVALUE

指定最大值，升序身份列在遇到这个值时将循环或停止生成值，或指定降序身份列在遇到最小值后将循环所至的值。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。

MAXVALUE *numeric-constant*

指定作为最大值的数字常量。这个值可以是任何可以对此列指定的正数值或负数值 (SQLSTATE 42820)，并且，在小数点右边没有非零位 (SQLSTATE 42894)，但此值必须大于最小值 (SQLSTATE 42815)。

NO MAXVALUE

对于升序序列，这个值是该列的数据类型的最大值。对于降序序列，这个值是 **START WITH** 值，或者，如果未指定 **START WITH** 则为 -1。

SET CYCLE 或 NO CYCLE

指定此身份列在生成最大值或最小值后是否应当继续生成值。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。

CYCLE

指定在遇到最大值或最小值之后继续为此列生成值。如果使用此选项，则升序身份列在遇到最大值之后会生成它的最小值；而降序序列在遇到最小值之后会生成它的最大值。身份列的最大值和最小值确定了用于循环的范围。

当 **CYCLE** 生效时，可以为身份列生成重复的值。虽然并不要求这样做，但如果您期望唯一的值，使用身份列定义的单列唯一索引可以确保唯一性。如果这样的身份列上存在唯一索引，但生成了不唯一的值，则会出错 (SQLSTATE 23505)。

NO CYCLE

指定在遇到最大值或最小值之后，便不再为身份列生成值。

SET CACHE *integer-constant* 或 NO CACHE

指定是否将一些预先分配的值放在内存中，以加快访问速度。这是一个性能和调整选项。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。

CACHE *integer-constant*

指定预先分配多少个身份序列值并将它们存放在内存中。预先分配值并将这些值存储在高速缓存中可以减少在为身份列生成值时对日志执行的同步 I/O。

如果身份列需要新值，而高速缓存中已没有未使用过的值，则值的分配要求等待对日志的 I/O。然而，当身份列需要新值并且高速缓存中有未使用过的值时，因为不必对日志执行 I/O 操作，所以该身份值的分配的速度就会快得多。

当数据库管理器停止时（例如数据库释放、系统故障或关机），所有已存入高速缓存但尚未在已提交的语句中使用过的序列值都会丢失（即，永远不再使用它们）。对 **CACHE** 选项指定的值便是在发生系统故障时可能会丢失的身份列值的最大数目。

最小值是 2 (SQLSTATE 42615)。

NO CACHE

指定不预先分配身份列的值。

如果指定此选项，则不将身份列的值存储在高速缓存中。在这种情况下，每一个对新身份值的请求都会导致对日志执行同步 I/O 操作。

SET ORDER 或 NO ORDER

指定是否必须按请求次序生成身份列值。必须已经用 **IDENTITY** 属性定义了该列 (SQLSTATE 42837)。

ORDER

指定按请求次序生成身份列值。

NO ORDER

指定无需按请求次序生成身份列值。

复合 SQL（嵌入式）

在 **ATOMIC** 复合 SQL 语句中，不允许已准备的 **COMMIT** 语句。

复合语句（动态）

复合语句（动态）

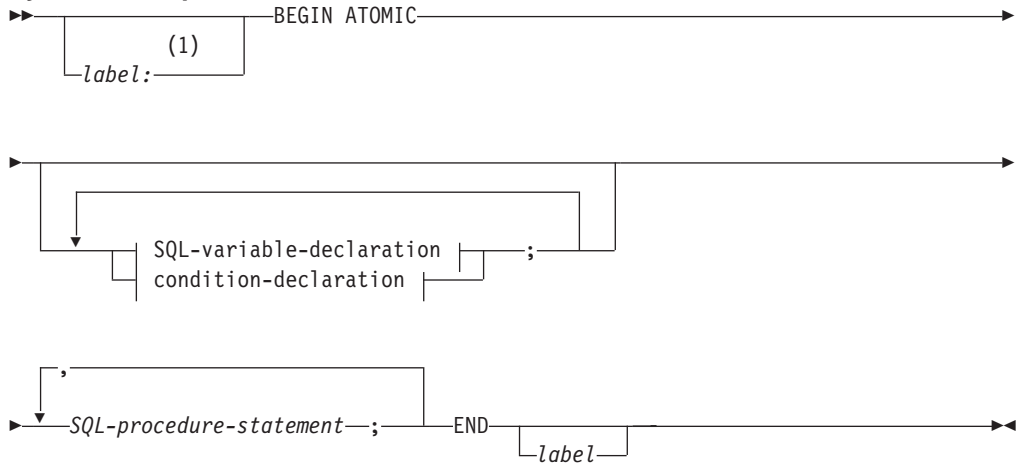
复合语句将其他语句分组到一起，形成可执行块。您可以在以动态方式准备的原子复合语句中声明 SQL 变量。

调用： 可以将此语句嵌入在触发器、“SQL 函数”或“SQL 方法”中，也可以通过使用动态 SQL 语句发出此语句。这是一条可以动态准备的可执行语句。

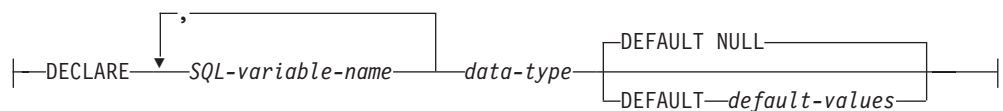
权限： 调用动态复合语句不需要特权。然而，复合语句的授权标识必须拥有调用嵌入在该复合语句中的 SQL 语句所必需的特权。

语法：

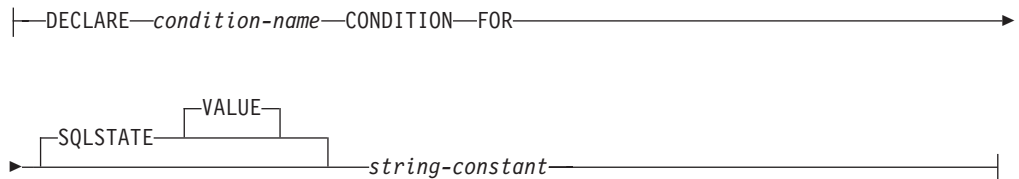
dynamic-compound-statement



SQL-variable-declaration:



condition-declaration:



注：

1 仅当该语句位于函数、方法或触发器定义中时，才能指定标号。

描述：

label

定义代码块的标号。如果指定的是开始标号，则它可用来限定动态复合语句中声明的 SQL 变量，并可以在 LEAVE 语句上指定。如果指定的是结束标号，则它必须与开始标号相同。

ATOMIC

ATOMIC 指示: 如果复合语句中发生错误, 则该复合语句中的所有 SQL 语句都将被回滚, 并且不处理该复合语句中的任何其余 SQL 语句。

SQL-procedure-statement

SQL-control-statement 的以下列表可以在动态复合语句内使用:

- FOR 语句
- GET DIAGNOSTICS 语句
- IF 语句
- ITERATE 语句
- LEAVE 语句
- SIGNAL 语句
- WHILE 语句

可以发出的 SQL 语句包括:

- 全查询⁶
- 搜索型 UPDATE
- 搜索型 DELETE
- INSERT
- SET variable 语句

SQL-variable-declaration

声明动态复合语句的局部变量。

SQL-variable-name

定义局部变量的名称。DB2 将所有 SQL 变量名都转换为大写。此名称不能:

- 与同一复合语句中的另一 SQL 变量相同。
- 与参数名相同。
- 与列名相同。

如果 SQL 语句包含与 SQL 变量和列引用同名的标识符, 则 DB2 将该标识符解释成列。

data-type

指定变量的数据类型。

DEFAULT *default-values* 或 NULL

定义 SQL 变量的缺省值。变量是在动态复合语句被调用时初始化的。如果未指定缺省值, 则变量初始化为 NULL。

condition-declaration

声明条件名和相应的 SQLSTATE 值。

condition-name

指定条件的名称。条件名在过程主体中必须是唯一的, 且只能在声明它的复合语句内引用它。

6. 全查询可以前置 `common-table-expression`

FOR SQLSTATE *string-constant*

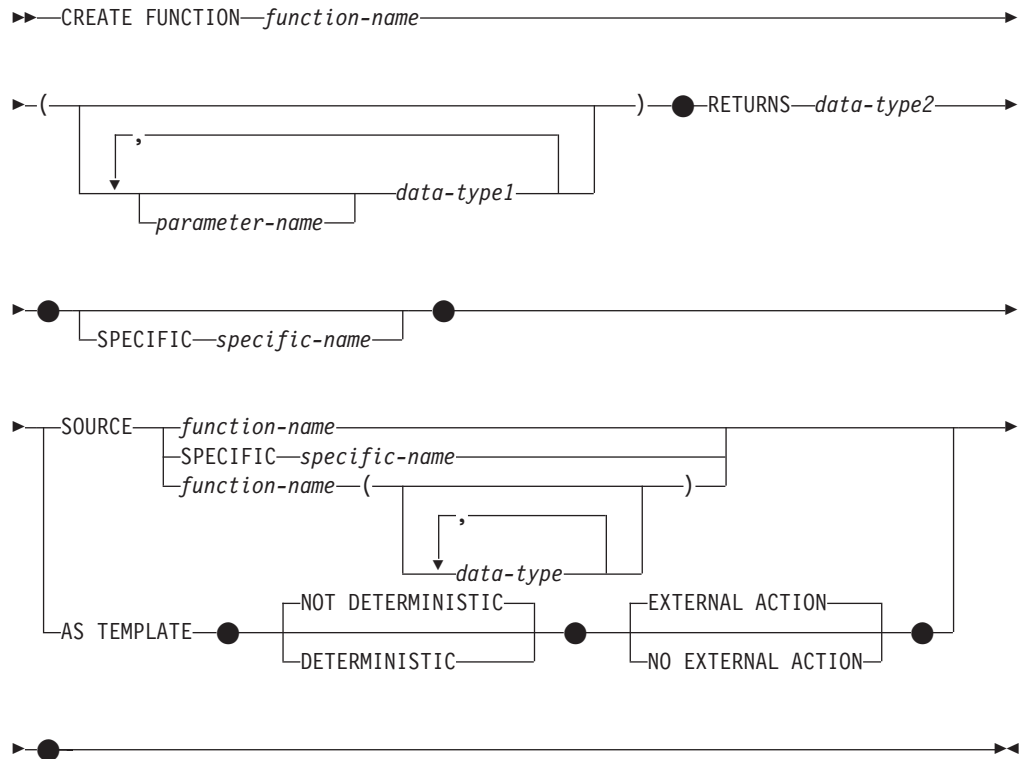
指定与条件相关联的 SQLSTATE。必须将 *string-constant* 指定为 5 个字符，括在单引号中，并且不能是 '00000'。

注意事项:

- DB2 将动态复合语句编译成单一语句。此语句对于涉及很少控制流逻辑但却涉及大量数据流的脚本而言非常有效。对于带有嵌套的复杂控制流的较大型构造，更好的选择是使用 SQL 过程。

CREATE FUNCTION (源或模板)

语法图更改为:



向 "Description" 节添加以下内容:

DETERMINISTIC or **NOT DETERMINISTIC**

This optional clause specifies whether the function always returns the same results for given argument values (**DETERMINISTIC**) or whether the function depends on some state values that affect the results (**NOT DETERMINISTIC**). That is, a **DETERMINISTIC** function must always return the same table from successive invocations with identical inputs. Optimizations taking advantage of the fact that identical inputs always produce the same results are prevented by specifying **NOT DETERMINISTIC**.

NOT DETERMINISTIC must be explicitly or implicitly specified if the body of the function accesses a special register or calls another non-deterministic function (SQLSTATE 428C2).

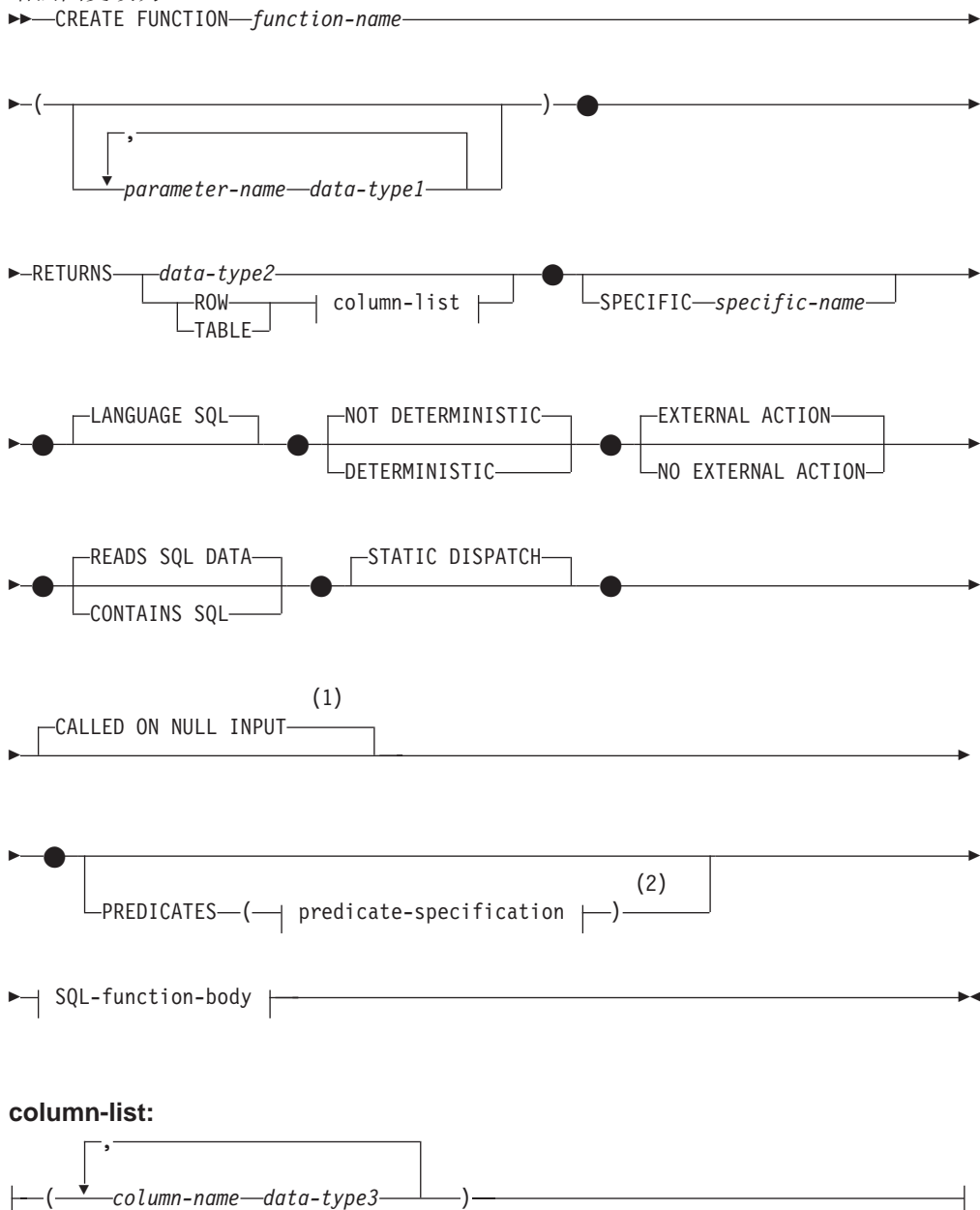
NO EXTERNAL ACTION or EXTERNAL ACTION

This optional clause specifies whether or not the function takes some action that changes the state of an object not managed by the database manager. By specifying NO EXTERNAL ACTION, the system can use certain optimizations that assume functions have no external impacts.

EXTERNAL ACTION must be explicitly or implicitly specified if the body of the function calls another function that has an external action (SQLSTATE 428C2).

CREATE FUNCTION (SQL 标量、表或行)

语法图更改为:



SQL-function-body:

├── RETURN 语句
└── dynamic-compound-statement

注:

- 1 可以指定 NULL CALL 来代替 CALLED ON NULL INPUT
- 2 仅当 RETURNS 指定了标量结果 (data-type2) 时才有效

更改下列参数:

LANGUAGE SQL

指定此函数是使用 SQL 编写的。

此参数节替换『RETURN 表达式、NULL、WITH common-table-expression、全查询』参数节。

SQL-function-body

指定函数的主体。可以在 SQL-function-body 中引用参数名。可以使用函数名对参数名加以限定以避免模糊引用。

如果 SQL-function-body 是动态复合语句，则它必须包含至少一个 RETURN 语句，且必须在调用此函数时执行 RETURN 语句 (SQLSTATE 42632)。如果此函数是一个表或行函数，则它只能包含一个 RETURN 语句，并且该语句必须是动态复合中的最后一个语句 (SQLSTATE 429BD)。

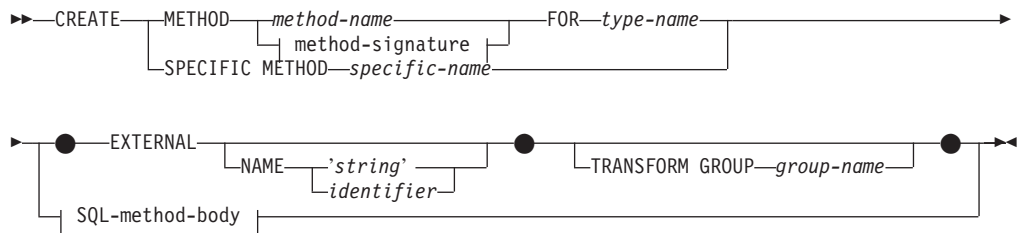
请参见第371页的『复合语句 (动态)』和第399页的『RETURN』，以获取其他详细信息。

CREATE METHOD

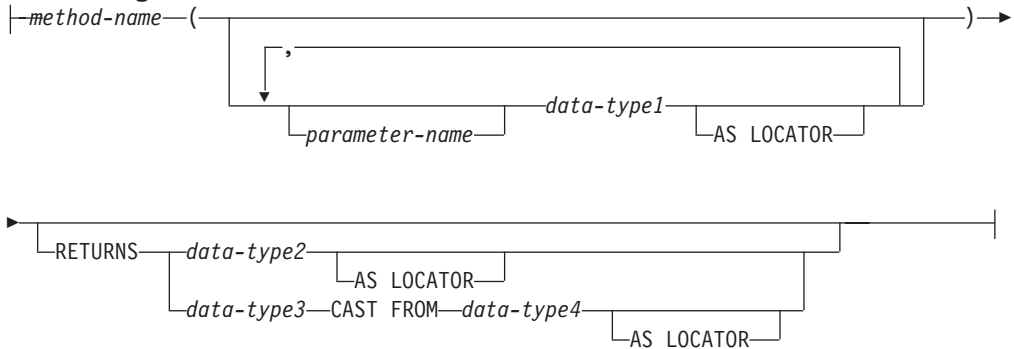
语法图更改为:

CREATE METHOD

语法:



method-signature:



SQL-method-body:



下列参数替换 "RETURN scalar-expression or NULL" 节:

SQL-method-body

The SQL-method-body defines the how the method is implemented if the method specification in CREATE TYPE is LANGUAGE SQL.

The SQL-method-body must comply with the following parts of the method specification:

- DETERMINISTIC or NOT DETERMINISTIC (SQLSTATE 428C2)
- EXTERNAL ACTION or NO EXTERNAL ACTION (SQLSTATE 428C2)
- CONTAINS SQL or READS SQL DATA (SQLSTATE 42985)

Parameter names can be referenced in the SQL-method-body. The subject of the method is passed to the method implementation as an implicit first parameter named SELF.

请参见第371页的『复合语句 (动态)』和第399页的『RETURN』, 以获取其他详细信息。

CREATE SEQUENCE

CREATE SEQUENCE

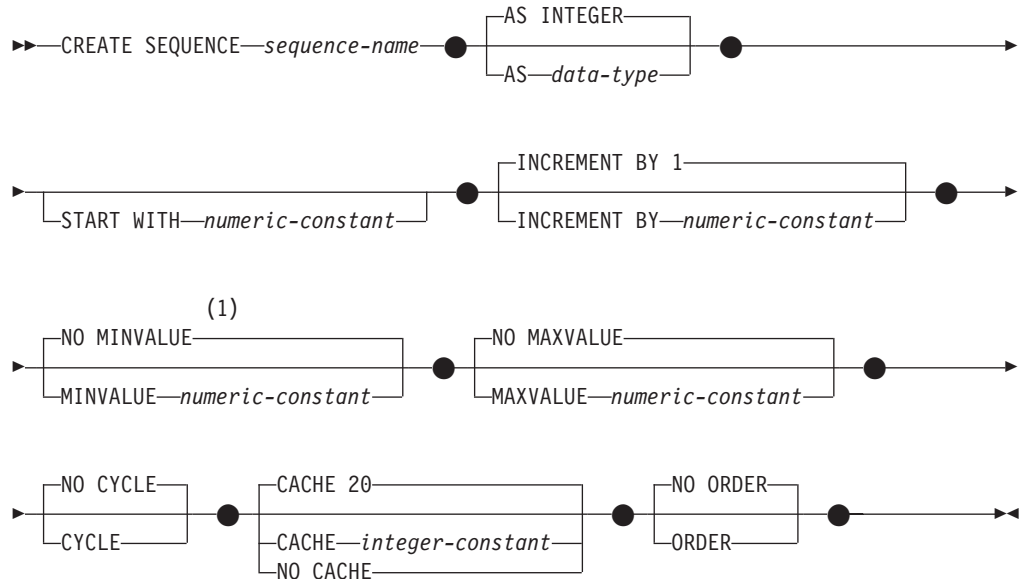
CREATE SEQUENCE 语句在应用程序服务器上创建一个序列。

调用: 可以将此语句嵌入在应用程序中,也可以通过使用动态 SQL 语句发出此语句。这是一条可以动态准备的可执行语句。然而,如果应用了绑定选项 DYNAMICRULES BIND,则不能动态准备此语句 (SQLSTATE 42509)。

权限: 此语句的授权标识所拥有的特权必须至少包括下列其中一项:

- 用于隐式或显式指定的模式的 CREATEIN 特权
- SYSADM 或 DBADM 权限

语法:



注:

- 1 可以指定这些参数 (不带空格): NOMINVALUE、NOMAXVALUE、NOCYCLE、NOCACHE 和 NOORDER。这些单字版本全都是双字版本的可接受的备用项。

描述:

sequence-name

对序列命名。名称与隐式或显式模式名的组合一定不能标识当前服务器上的现存序列 (SQLSTATE 42710)。

序列名的未限定格式是 SQL 标识符。限定格式是后跟句点和 SQL 标识符的限定符。限定符是模式名。

如果序列名是使用模式名显式限定的,则模式名不能以 'SYS' 开头,否则会出错 (SQLSTATE 42939)。

AS data-type

指定要用于序列值的数据类型。数据类型可以是任何不带小数位的精确数字类型

(SMALLINT、INTEGER、BIGINT 或 DECIMAL)，也可以是其源类型为不带小数位的精确数字类型的用户定义单值类型 (SQLSTATE 42815)。缺省值是 INTEGER。

START WITH *numeric-constant*

为序列指定第一个值。这个值可以是任何可以对与此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，但小数点右边不能有非零位 (SQLSTATE 428FA)。对于升序序列，缺省值是 MINVALUE，对于降序序列，是 MAXVALUE。

这个值不必是序列在遇到序列的最大值或最小值后要循环至的值。可以使用 START WITH 子句来在用于循环的范围外部启动一个序列。用于循环的范围由 MINVALUE 和 MAXVALUE 定义。

INCREMENT BY *numeric-constant*

指定序列的连续值之间的间隔。这个值可以是任何可以对与此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，不超出大整数常量的值 (SQLSTATE 42815)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)。

如果这个值是负数，则值序列是降序的。如果这个值是正数，则值序列是升序的。如果这个值是 0 或大于 MINVALUE 和 MAXVALUE 定义的范围，则只生成一个值，但视该序列为升序序列。缺省值是 1。

MINVALUE 或 NO MINVALUE

指定最小值，降序序列在遇到这个值时将循环或停止生成值，升序序列在遇到最大值后会循环至这个值。

MINVALUE *numeric-constant*

指定作为最小值的数字常量。这个值可以是任何可以对与此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 42820)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)，但此值必须小于或等于最大值 (SQLSTATE 42815)。

NO MINVALUE

对于升序序列，这个值是 START WITH 值，或者，如果未指定 START WITH 则为 1。对于降序序列，这个值是与该序列相关联的数据类型的最小值。这是缺省值。

MAXVALUE 或 NO MAXVALUE

指定最大值，升序序列在遇到这个值时将循环或停止生成值，降序序列在遇到最小值后会循环至这个值。

MAXVALUE *numeric-constant*

指定作为最大值的数字常量。这个值可以是任何可以对与此序列相关联的数据类型的列指定的正数值或负数值 (SQLSTATE 428FA)，并且，在小数点右边没有非零位 (SQLSTATE 428FA)，但此值必须大于或等于最小值 (SQLSTATE 42815)。

NO MAXVALUE

对于升序序列，这个值是与该序列相关联的数据类型的最大值。对于降序序列，这个值是 START WITH 值，或者，如果未指定 START WITH 则为 -1。这是缺省值。

CYCLE 或 NO CYCLE

指定序列在遇到它的最大值或最小值后是否应当继续生成值。到达序列边界的情况有两种，一种情况是下一个值刚好位于边界上，另一种情况是越过序列边界。

CYCLE

指定在遇到最大值或最小值之后继续为此序列生成值。如果使用此选项，则升序序列在遇到它的最大值之后会生成它的最小值；而降序序列在遇到它的最小值之后会生成它的最大值。序列的最大值和最小值确定了用于循环的范围。

当 CYCLE 生效时，可以为序列生成重复的值。

NO CYCLE

指定在遇到序列的最大值或最小值之后，便不再为序列生成值。这是缺省值。

CACHE 或 NO CACHE

指定是否将一些预先分配的值放在内存中，以加快访问速度。这是一个性能和调整选项。

CACHE *integer-constant*

指定预先分配并存放在内存中的序列值的最大数目。预先分配值并将这些值存储在高速缓存中可以减少在为序列生成值时对日志执行的同步 I/O。

如果发生系统故障，则所有已存入高速缓存但尚未在已提交的语句中使用过的序列值都会丢失（即，永远不再使用它们）。对 CACHE 选项指定的值便是在发生系统故障时可能会丢失的序列值的最大数目。

最小值是 2 (SQLSTATE 42815)。缺省值是 CACHE 20。

NO CACHE

指定不预先分配序列的值。这确保即使发生系统故障、关机或数据库释放也不会丢失值。如果指定此选项，则不将序列的值存储在高速缓存中。在这种情况下，每一个对新序列值的请求都会导致对日志执行同步 I/O 操作。

NO ORDER 或 ORDER

指定是否必须按请求次序生成序列号。

ORDER

指定按请求次序生成序列号。

NO ORDER

指定无需按请求次序生成序列号。这是缺省值。

注意事项:

- 可以定义常量序列，即，始终返回常量值的序列。这可以通过对 MINVALUE 或 MAXVALUE 指定相同的值，或通过将 INCREMENT 值指定为零做到。在任一种情况下，为了允许 NEXTVAL 多次生成同一个值，必须指定 CYCLE。可以将常量序列用作数字全局变量。可以使用 ALTER SEQUENCE 来调整将要为常量序列生成的值。
- 通过使用 ALTER SEQUENCE 语句，可以以人工方式让序列循环。如果以隐式或显式方式指定了 NO CYCLE，则可以使用 ALTER SEQUENCE 语句来重新启动或扩充序列，以便在遇到序列的最大值或最小值之后继续生成值。
- 将序列号存入高速缓存意味着可以将一系列序列号存放在内存中，以加快访问速度。当应用程序访问可以从高速缓存分配下一个序列号的序列时，序列号分配便可以快速地发生。然而，如果应用程序访问不能从高速缓存分配下一个序列号的序列，则序列号分配可能会要求必须等待对永久存储器执行的 I/O 操作。在为 CACHE 选择值时，您应当在性能与应用程序需求之间进行权衡。
- 所有者对新序列具有 ALTER 和 USAGE 特权。所有者只能授予 USAGE 特权，且只能将该特权授予 PUBLIC。

- 还支持下列语法：NOMINVALUE、NOMAXVALUE、NOCYCLE、NOCACHE 和 NOORDER。

示例： 示例 1: 创建名为 org_seq 的序列:

```
CREATE SEQUENCE org_seq
START WITH 1
INCREMENT BY 1
NO MAXVALUE
NO CYCLE
CACHE 24
```

CREATE TRIGGER

CREATE TRIGGER

语法:

▶ CREATE TRIGGER *trigger-name* NO CASCADE BEFORE
AFTER

▶ INSERT
DELETE
UPDATE ON *table-name*
OF *column-name*

▶ REFERENCING (1) (2)
OLD AS *correlation-name*
NEW AS *correlation-name*
OLD_TABLE AS *identifier*
NEW_TABLE AS *identifier*

▶ FOR EACH ROW (3) MODE DB2SQL | *triggered-action* |
FOR EACH STATEMENT

triggered-action:

| WHEN (*search-condition*) | *SQL-procedure-statement* |

注:

- 1 OLD 和 NEW 每个只能指定一次。
- 2 OLD_TABLE 和 NEW_TABLE 每个只能指定一次，且只能对 AFTER 触发器指定。
- 3 不能对 BEFORE 触发器指定 FOR EACH STATEMENT。

将 "triggered-action" 的描述替换为:

triggered-action

Specifies the action to be performed when a trigger is activated. A triggered-action is composed of an *SQL-procedure-statement* and an optional condition for the execution of the *SQL-procedure-statement*.

WHEN (*search-condition*)

指定一个条件，该条件为真、假或未知。*search-condition* 提供了用于确定是否应执行已触发的特定操作的能力。

仅当指定的搜索条件求值为真时，才会执行相关联的操作。如果省略了 WHEN 子句，则始终执行相关联的 *SQL-procedure* 语句。

SQL-procedure-statement

SQL-procedure-statement 可以包含动态复合语句或第371页的『复合语句（动态）』中列示的任何 SQL 控制语句。

如果触发器是 BEFORE 触发器，则 *SQL-procedure-statement* 还可以包含全查询或 SET variable 语句 (SQLSTATE 42987)。

如果触发器是 AFTER 触发器，则 *SQL-procedure-statement* 还可以包含下列其中一项 (SQLSTATE 42987):

- INSERT SQL 语句
- 搜索型 UPDATE SQL 语句
- 搜索型 DELETE SQL 语句
- SET variable 语句
- 全查询⁷

SQL-procedure-statement 不能引用未定义的过渡变量 (SQLSTATE 42703) 或已声明的临时表 (SQLSTATE 42995)。

BEFORE 触发器中的 *SQL-procedure-statement* 不能引用使用 REFRESH IMMEDIATE 定义的总结表 (SQLSTATE 42997)。

BEFORE 触发器中的 *SQL-procedure-statement* 不能在新的过渡变量中引用除身份列之外的已生成列 (SQLSTATE 42989)。

Notes 节更改为:

- The result of a fullselect specified in the *SQL-procedure-statement* is not available inside or outside of the trigger.
- **Inoperative triggers:** An *inoperative trigger* is a trigger that is no longer available and is therefore never activated. A trigger becomes inoperative if:
 - A privilege that the creator of the trigger is required to have for the trigger to execute is revoked.
 - An object such as a table, view or alias, upon which the triggered action is dependent, is dropped.
 - A view, upon which the triggered action is dependent, becomes inoperative.
 - An alias that is the subject table of the trigger is dropped.

In practical terms, an inoperative trigger is one in which a trigger definition has been dropped as a result of cascading rules for DROP or REVOKE statements. For example, when an view is dropped, any trigger with an *SQL-procedure-statement* defined using that view is made inoperative.

When a trigger is made inoperative, all packages with statements performing operations that were activating the trigger will be marked invalid. When the package is rebound (explicitly or implicitly) the **inoperative trigger is completely ignored**. Similarly, applications with dynamic SQL statements performing operations that were activating the trigger will also completely ignore any inoperative triggers.

7. 全查询的前面可以带有 common-table-expression.

The trigger name can still be specified in the DROP TRIGGER and COMMENT ON TRIGGER statements.

An inoperative trigger may be recreated by issuing a CREATE TRIGGER statement using the definition text of the inoperative trigger. This trigger definition text is stored in the TEXT column of SYSCAT.TRIGGERS. Note that there is no need to explicitly drop the inoperative trigger in order to recreate it. Issuing a CREATE TRIGGER statement with the same *trigger-name* as an inoperative trigger will cause that inoperative trigger to be replaced with a warning (SQLSTATE 01595).

Inoperative triggers are indicated by an X in the VALID column of the SYSCAT.TRIGGERS catalog view.

- **Errors executing triggers:** Errors that occur during the execution of triggered SQL statements are returned using SQLSTATE 09000 unless the error is considered severe. If the error is severe, the severe error SQLSTATE is returned. The SQLERRMC field of the SQLCA for non-severe error will include the trigger name, SQLCODE, SQLSTATE and as many tokens as will fit from the tokens of the failure.

The *SQL-procedure-statement* could include a SIGNAL SQLSTATE statement or contain a RAISE_ERROR function. In both these cases, the SQLSTATE returned is the one specified in the SIGNAL SQLSTATE statement or the RAISE_ERROR condition.

CREATE WRAPPER

Linux 使用库 LIBDRDA.S0 和 LIBSQLNET.S0, 而不是 LIBDRDA.A 和 LIBSQLNET.A.

DECLARE CURSOR

在 "DECLARE CURSOR" 语句中, 在接近 Notes 一节末尾的地方, 应该将以下语句从:

An ambiguous cursor is considered read-only if the BLOCKING bind option is ALL, otherwise it is considered deletable.

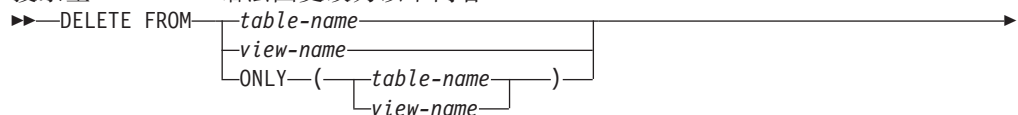
更改为:

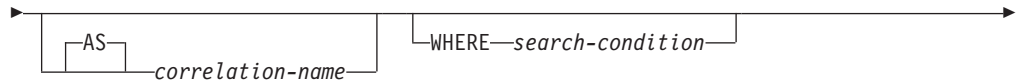
An ambiguous cursor is considered read-only if the BLOCKING bind option is ALL; otherwise, it is considered updatable.

所作的更改是将单词 "deletable" 更改为单词 "updatable".

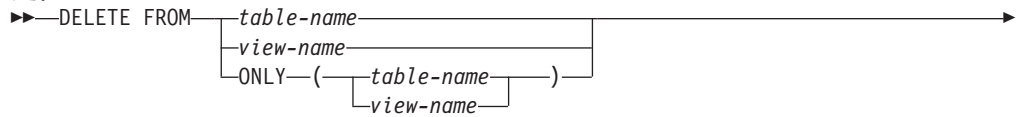
DELETE

搜索型 DELETE 语法图更改为以下内容:





定位型 DELETE:



向描述节添加以下内容:

WITH

Specifies the isolation level used when locating the rows to be deleted.

RR

Repeatable Read

RS

Read Stability

CS

Cursor Stability

UR

Uncommitted Read

The default isolation level of the statement is the isolation level of the package in which the statement is bound.

DROP

添加下列选项:



添加下列参数:

SEQUENCE *sequence-name* RESTRICT

标识要删除的特定序列。 *sequence-name* (与隐式或显式模式名一起) 必须标识当前服务器上的现存序列。 如果显式或隐式指定的模式中不存在具有此名称的序列, 则会发生错误 (SQLSTATE 42704)。

RESTRICT 键强制这样一个规则：如果表列的定义引用一个序列（通过 IDENTITY 列），则不删除该序列。

注：

- 不能使用 DROP sequence 命令删除系统为 IDENTITY 列创建的序列。
- 删除序列时，还将删除该序列上的所有特权。

需要更新显示对象相互具有的相关性的表（表 27），如下所示：

新行：DROP SEQUENCE 新行 "DROP SEQUENCE" 与列 "PACKAGE" 的交叉点除的条目将是 "A"。这个新行中的其他条目将是 "-"

GRANT (序列特权)

GRANT (序列特权)

GRANT 语句的这种格式授予对用户定义序列的特权。

调用: 可以将此语句嵌入在应用程序中,也可以通过使用动态 SQL 语句发出此语句。这是一条可以动态准备的可执行语句。然而,如果应用了绑定选项 DYNAMICRULES BIND,则不能动态准备此语句 (SQLSTATE 42509)。

权限: 此语句的授权标识所拥有的特权必须至少包括下列其中一项:

- 序列的所有者
- SYSADM 或 DBADM 权限

语法:

►► GRANT USAGE ON SEQUENCE *sequence-name* TO PUBLIC

描述:

USAGE

授予对序列的 USAGE 特权。

ON SEQUENCE *sequence-name*

标识要授予对其的 USAGE 特权的序列。*sequence-name* (包括隐式或显式模式限定符)必须唯一地标识当前服务器上的现存序列。如果指定的模式中不存在具有此名称的序列,则会发生错误 (SQLSTATE 42704)。

TO PUBLIC

将 USAGE 特权授予所有用户。

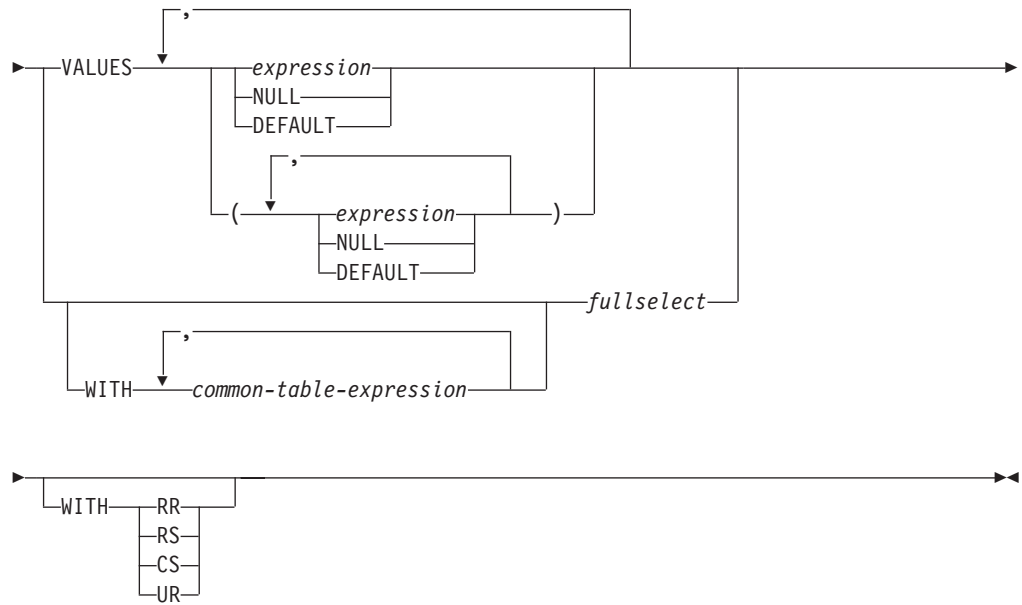
示例: 示例 1: 将对名为 MYNUM 的序列的特权授予任何用户

```
GRANT USAGE ON SEQUENCE MYNUM TO PUBLIC
```

INSERT

语法图更改为:

►► INSERT INTO *table-name* | *view-name* (*column-name*)



向描述节添加以下内容:

WITH

Specifies the isolation level at which the fullselect is executed.

RR

Repeatable Read

RS

Read Stability

CS

Cursor Stability

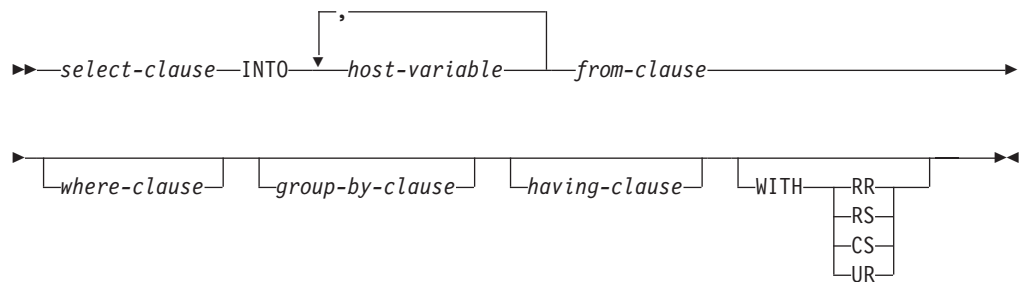
UR

Uncommitted Read

The default isolation level of the statement is the isolation level of the package in which the statement is bound.

SELECT INTO

语法图更改为:



向描述节添加以下内容:

WITH

Specifies the isolation level at which the SELECT INTO statement is executed.

RR

Repeatable Read

RS

Read Stability

CS

Cursor Stability

UR

Uncommitted Read

The default isolation level of the statement is the isolation level of the package in which the statement is bound.

SET ENCRYPTION PASSWORD

SET ENCRYPTION PASSWORD

SET ENCRYPTION PASSWORD 语句设置加密和解密函数将要使用的口令。此口令与 DB2 认证无关，它只用于数据加密。

此语句不受事务控制。

调用： 此语句可以嵌入在应用程序中，也可以以交互方式发出。这是一条可以动态准备的可执行语句。

权限： 执行此语句无需权限。

语法：

```
►► SET ENCRYPTION PASSWORD   host-variable | string-constant ►►
```

描述： ENCRYPT、DECRYPT_BIN 和 DECRYPT_CHAR 内置函数可以将 ENCRYPTION PASSWORD 用于基于口令的加密。长度必须在 6 与 127 之间（包括 6 和 127）。因为不会将任何字符转换为大写字符，所以所有字符都必须以期望的大小写指定。

host-variable

类型为 CHAR 或 VARCHAR 的变量。host-variable 的内容的长度必须在 6 与 127 之间（包括 6 和 127）(SQLSTATE 428FC)。不能将其设置为空。因为不会将任何字符转换为大写字符，所以所有字符都必须以期望的大小写指定。

string-constant

字符串常量。长度必须在 6 与 127 之间（包括 6 和 127）(SQLSTATE 428FC)。

规则：

- 初始的 ENCRYPTION PASSWORD 值是空字符串 (‘’)
- 将使用正常的 DB2 机制将 host-variable 或 string-constant 传送至数据库服务器。

注意事项：

- 请参见第 336 页的『ENCRYPT』和第 335 页的『DECRYPT_BIN 和 DECRYPT_CHAR』，以了解有关使用此语句的其他信息。

示例： 示例 1：以下语句设置 ENCRYPTION PASSWORD。

```
SET ENCRYPTION PASSWORD = 'bubbalu'
```

SET transition-variable

本节更改为以下内容：

SET Variable

The SET Variable statement assigns values to local variables or to new transition variables. It is under transaction control.

调用: This statement can only be used as an SQL statement in either a dynamic compound statement, trigger, SQL function or SQL method.

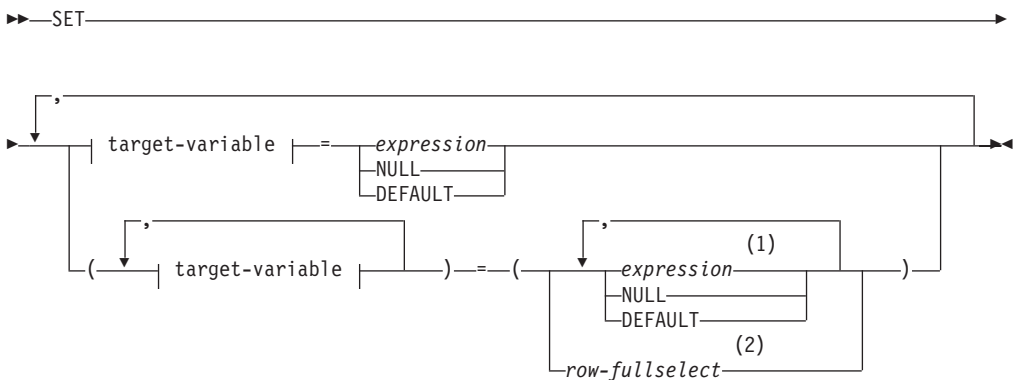
权限: To reference a transition variable, the privileges held by the authorization ID of the trigger creator must include at least one of the following:

- UPDATE of the columns referenced on the left hand side of the assignment and SELECT for any columns referenced on the right hand side.
- CONTROL privilege on the table (subject table of the trigger)
- SYSADM or DBADM authority.

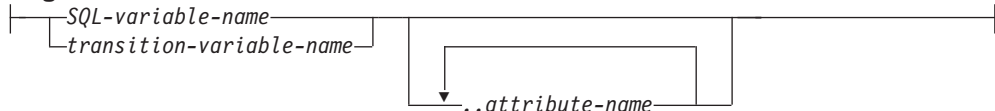
To execute this statement with a *row-fullselect* as the right hand side of the assignment, the privileges held by the authorization ID of either the trigger definer or the dynamic compound statement owner must also include at least one of the following, for each table or view referenced:

- SELECT privilege
- CONTROL privilege
- SYSADM or DBADM.

语法:



target-variable:



注:

- 1 表达式、NULL 和 DEFAULT 的数目必须与 *target-variable* 的数目相匹配。
- 2 选择列表中的列数必须与 *target-variable* 的数目相匹配。

描述:

target-variable

标识赋值的目标变量。一定不能多次指定表示同一变量的 *target-variable* (SQLSTATE 42701)。

SQL-variable-name

标识作为赋值目标的 SQL 变量。在使用 SQL 变量之前，必须先声明。可以在动态复合语句中定义 SQL 变量。

transition-variable-name

标识要在过渡行中更新的列。*transition-variable-name* 必须标识触发器的主体表中的一列，可以选择通过标识新值的相关名进行限定 (SQLSTATE 42703)。

..attribute name

指定设置的结构化类型的属性（称为属性指定）。指定的 *SQL-variable-name* 或 *transition-variable-name* 必须使用用户定义的结构化类型定义 (SQLSTATE 428DP)。 *attribute-name* 必须是结构化类型的属性 (SQLSTATE 42703)。不涉及 *..attribute name* 子句的指定称为常规指定。

expression

指定 *target-variable* 的新值。表达式是具有 SQL Reference 的第 2 章中描述的任何类型的任何表达式。表达式不能包括列函数，除非它出现在标量全查询中 (SQLSTATE 42903)。在 CREATE TRIGGER 语句的上下文中，*expression* 可以包含对 OLD 和 NEW 过渡变量的引用，必须由 *correlation-name* 限定为指定过渡变量 (SQLSTATE 42702)。

NULL

指定空值，只能对可空列指定 (SQLSTATE 23502)。NULL 不能是属性指定中的值 (SQLSTATE 429B9)，除非将其明确地强制转换为属性的数据类型。

DEFAULT

指定应使用的缺省值。

如果 *target-variable* 是一列，则插入的值取决于该列在表中是如何定义的。

- 如果该列是使用 WITH DEFAULT 子句定义的，则将该值设置为对该列定义的缺省值。
- 如果该列是使用 IDENTITY 子句定义的，则值由数据库管理器生成。
- 如果定义该列时未指定 WITH DEFAULT 子句、IDENTITY 子句或 NOT NULL 子句，则该值为 NULL。
- 如果该列是使用 NOT NULL 子句定义的，而没有使用 IDENTITY 子句，或未使用 WITH DEFAULT 子句，或使用了 DEFAULT NULL，则不能对该列指定 DEFAULT 键 (SQLSTATE 23502)。

如果 *target-variable* 是一个 SQL 变量，则插入的值是在变量声明中指定的或暗指的缺省值。

row-fullselect

一个全查询，它返回带有与对赋值指定的 *target-variable* 的数目相对应的列数的单一行。值被赋给每一个相对应的 *target-variable*。如果 *row-fullselect* 的结果是没有行，则指定空值。在 CREATE TRIGGER 语句的上下文中，*row-fullselect* 可能包含对 OLD 和 NEW 过渡变量的引用，这些过渡变量必须由它们的 *correlation-name* 限定，以指定要使用的过渡变量 (SQLSTATE 42702)。如果结果中有多行，则会返回错误 (SQLSTATE 21000)。

规则:

- 要从表达式、NULL 和 DEFAULT 或 *row-fullselect* 赋给的值数目必须与对赋值指定的 *target-variable* 的数目相匹配 (SQLSTATE 42802)。
- SET Variable 语句不能在一个语句中对 SQL 变量和过渡变量进行赋值 (SQLSTATE 42997)。
- 值根据 SQL Reference 的第 2 章中描述的赋值规则被赋给 *target-variable*。
如果在 BEFORE UPDATE 触发器中使用该语句, 且注册表变量 DB2_UPDATE_PART_KEY=OFF, 则作为 *target-variable* 指定的 *transition-variable* 不能是分区键列 (SQLSTATE 42997)。

注意事项:

- 如果包括多个赋值, 则在执行赋值之前, 对所有 *expression* 和 *row-fullselect* 进行求值。因此, 对表达式或行全查询中的 *target-variable* 的引用始终是单一 SET 语句中的任何赋值之前的 *target-variable* 值。
- 当更新作为相异类型定义的身份列时, 整个计算在源类型中执行, 在将值实际赋给该列之前, 将结果强制转换为相异类型。⁸
- 要让 DB2 在 SET 语句上未身份列生成一个值, 请使用 DEFAULT 键:

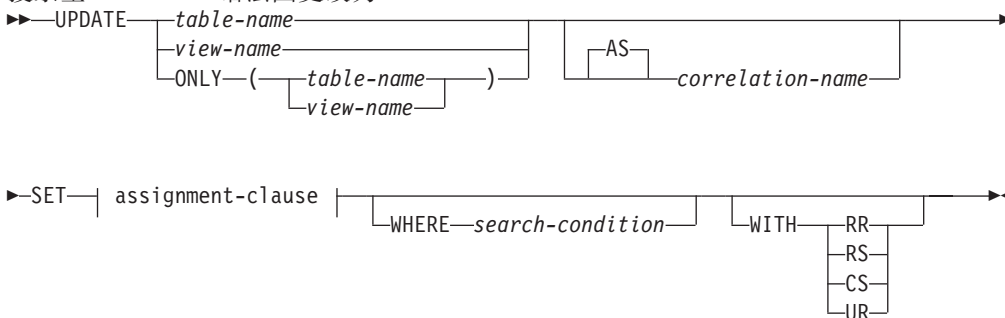
```
SET NEW.EMPNO = DEFAULT
```

在这个示例中, NEW.EMPNO 定义为身份列, 用来更新此列的值由 DB2 生成。

此语句的示例保持不变。

UPDATE

搜索型 UPDATE 语法图更改为:



向描述节添加以下内容:

WITH

Specifies the isolation level at which the UPDATE statement is executed.

RR

Repeatable Read

RS

Read Stability

8. 在计算之前, 不将先前的值强制转换为源类型。

CS

Cursor Stability

UR

Uncommitted Read

The default isolation level of the statement is the isolation level of the package in which the statement is bound.

Chapter 7. SQL Procedures 现在名为 Chapter 7. SQL Control Statements

控制语句是 SQL 语句，控制语句允许以与使用结构化编程语言编写程序相类似的方式使用 SQL。可以在例程、触发器或动态复合语句的主体中使用 SQL 控制语句。

本章包含受支持的 SQL 控制语句以及 SQL-procedure-statement 的语法和描述。

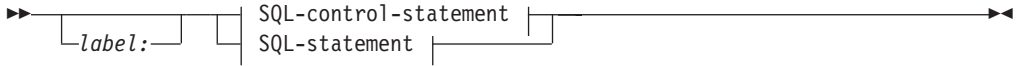
SQL 过程语句

“SQL 过程语句”的信息更改为:

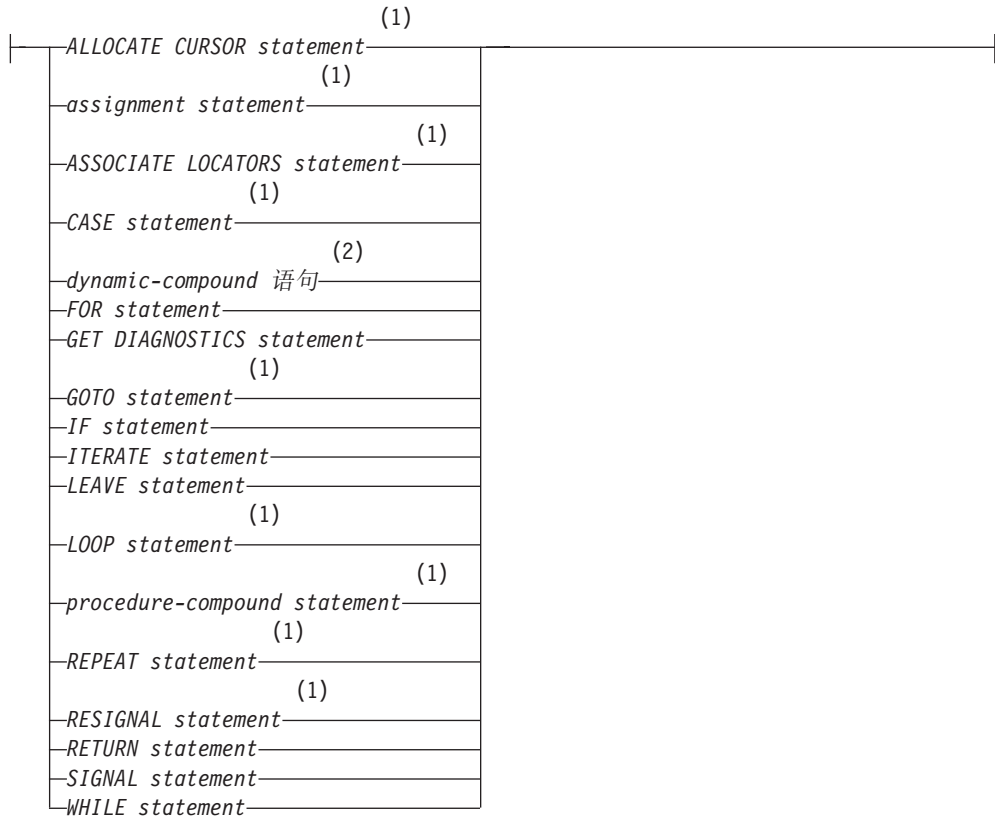
SQL Procedure Statement

This chapter contains syntax diagrams, semantic descriptions, rules, and examples of the use of the statements that constitute the procedure body of an SQL routine, trigger, or dynamic compound statement.

语法:



SQL-control-statement:



注:

- 1 此语句只有在“SQL 过程”的作用域内才受支持。
- 2 此语句只有在触发器、SQL 函数或 SQL 方法中才受支持。它必须是最外面的语句。

描述:

label:

指定 SQL 过程语句的标号。标号在 SQL 过程语句的列表（包括嵌套在该列表内的任何复合语句）中必须是唯一的。注意，不嵌套的复合语句可以使用同一标号。在许多 SQL 控制语句中，可以存在 SQL 过程语句的列表。

在触发器、SQL 函数或方法或动态复合语句的上下文中，只有动态复合语句、FOR 语句和 WHILE 语句可以带有标号。

SQL-statement

在 SQL 过程中，可以包含所有可执行的 SQL 语句，但下列各项除外：

- CONNECT
- CREATE 任何除索引、表或视图之外的对象
- DESCRIBE
- DISCONNECT
- DROP 任何除索引、表或视图之外的对象
- FLUSH EVENT MONITOR
- REFRESH TABLE
- RELEASE (仅连接)
- RENAME TABLE
- RENAME TABLESPACE
- REVOKE
- SET CONNECTION
- SET INTEGRITY

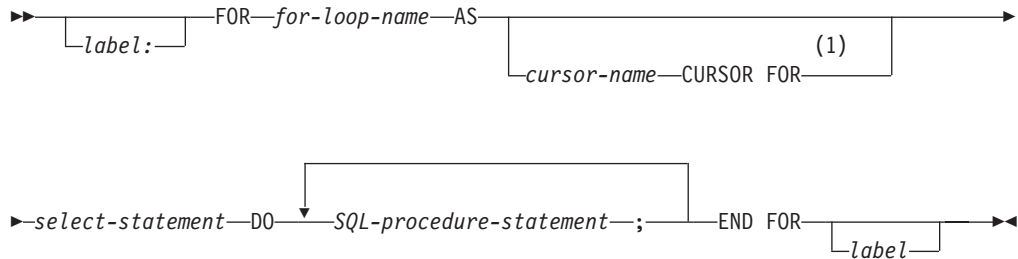
注：可以将 CALL 语句包括在 SQL 过程主体中，但这些 CALL 语句只能调用另一个 SQL 过程或 C 过程。SQL 过程主体中的 CALL 语句不能调用其他类型的存储过程。

FOR

FOR

FOR 语句对表的每一行执行语句或语句组。

语法:



注:

1 此选项只能在“SQL 过程”的上下文中使用。

描述:

label

指定 FOR 语句的标号。如果指定的是开始标号，则该标号可以在 LEAVE 和 ITERATE 语句中使用。如果指定的是结束标号，则它必须与开始标号相同。

for-loop-name

指定为了实现 FOR 语句而生成的隐式复合语句的标号。它遵循复合语句的标号规则，但它不能与 FOR 语句中的 ITERATE 或 LEAVE 语句配合使用。*for-loop-name* 用来限定指定的 *select-statement* 所返回的列名。

cursor-name

命名游标，该游标用来从 SELECT 语句的结果表中选择行。如果未指定，则 DB2 将生成唯一的游标名。

select-statement

指定游标的 SELECT 语句。选择列表中的所有列都必须具有名称，任何两列都不能同名。

在触发器、函数、方法或动态复合语句中，*select-statement* 必须只包含全查询和可选的公共表表达式。

SQL-procedure-statement

指定要对表的每一行调用的语句。

规则:

- 选择列表必须由唯一的列名组成，在创建该过程时，选择列表中指定的表必须存在，或者，它必须是在前一 SQL 过程语句中创建的表。
- for-statement 中指定的游标不能在该 for-statement 外部引用，且不能在 OPEN、FETCH 或 CLOSE 语句中指定。

示例: 在下例中，for-statement 用来对整个 employee 表执行迭代。对于该表中的每一行，SQL 变量 fullname 设置为雇员的姓，后跟逗号、名、空白和中央首字母。fullname 的每个值都被插入到 tnames 表中。

```
BEGIN
  DECLARE fullname CHAR(40);
  FOR v1 AS
```

```

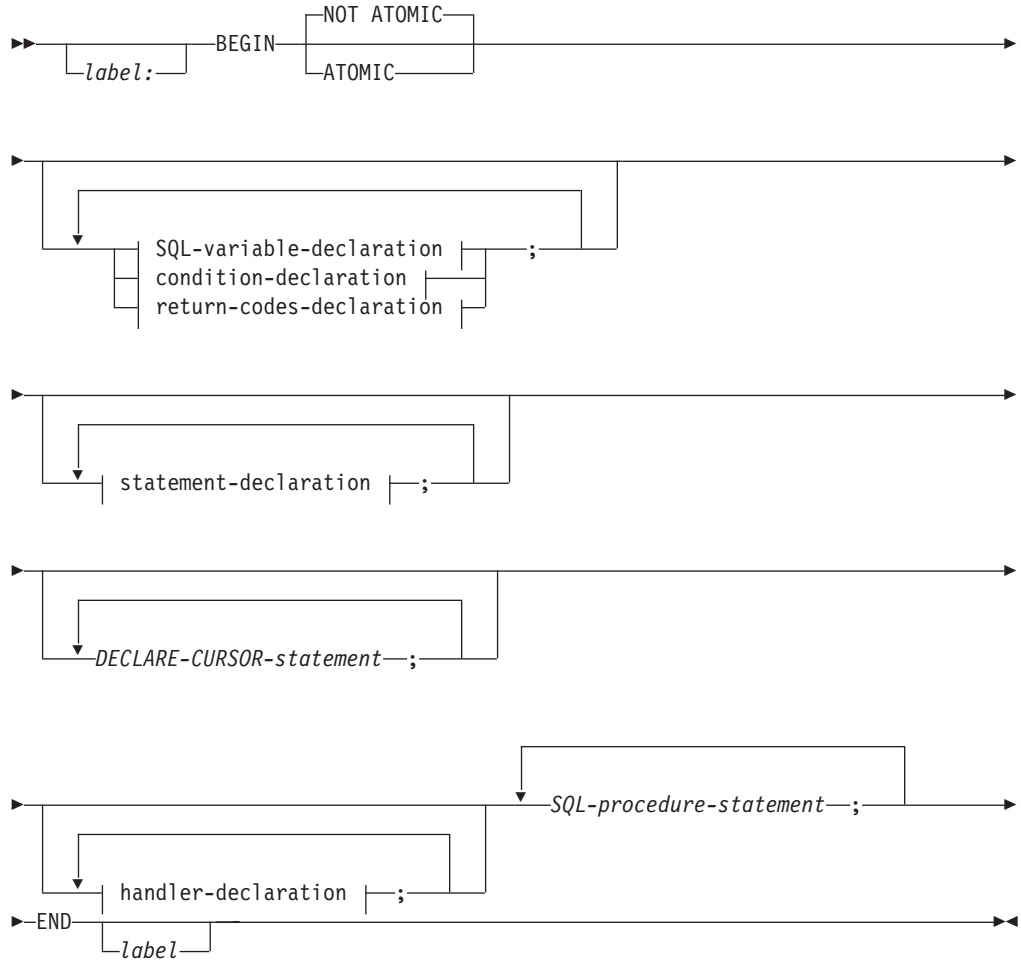
SELECT firstme, midinit, lastname FROM employee
DO
SET fullname = lastname || ',' || firstme || ' ' || midinit;
INSERT INTO tnames VALUE (fullname);
END FOR
END

```

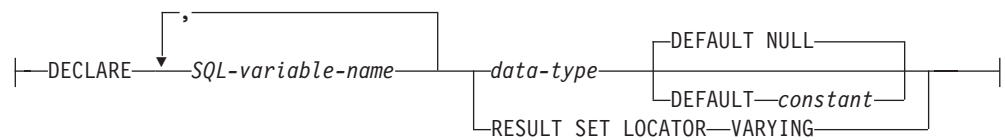
Compound Statement 更改为 Compound Statement (Procedure)

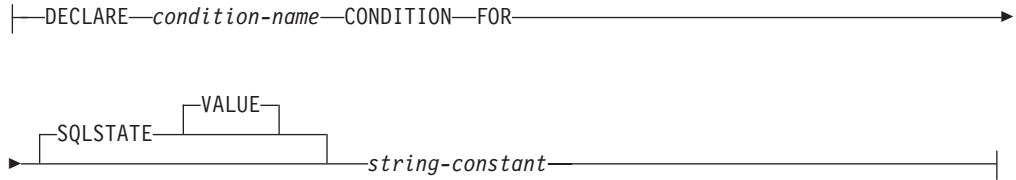
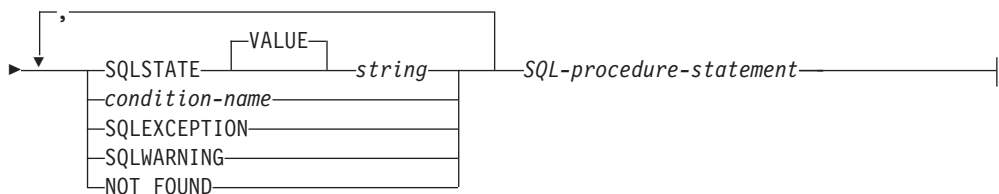
过程复合语句将其他语句分组到 SQL 过程中。可以在复合语句中声明 SQL 变量、游标和条件处理程序。

语法图现在带有标题: 过程复合语句。



SQL-variable-declaration:



condition-declaration:**statement-declaration:****return-codes-declaration:****handler-declaration:****statement-declaration**

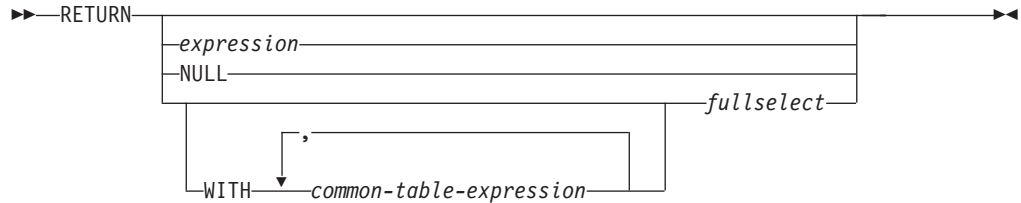
statement-declaration 声明仅限于复合语句使用的一个或多个名称的列表。在同一复合语句内，两个语句的名称不能相同。

RETURN

RETURN

RETURN 语句用来从例程返回。对于 SQL 函数或方法，它返回函数或方法的结果。对于 SQL 过程，它有选择性地返回整数状态值。

语法:



描述:

expression

指定要从例程返回的值:

- 如果该例程是函数或方法，则必须指定 *expression*、NULL 或 *fullselect* 之一 (SQLSTATE 42630)，结果的数据类型必须可指定给例程的 RETURNS 类型 (SQLSTATE 42866)。
- 不能对表函数指定标量表达式 (标量全查询除外)。
- 如果该例程是过程，则 *expression* 的数据类型必须是 INTEGER (SQLSTATE 428E2)。过程不能返回 NULL 或全查询。

NULL

指定函数或方法返回 RETURNS 子句中定义的数据类型的空值。不能对过程中的 RETURN 指定 NULL。

WITH *common-table-expression*

定义要与后面的全查询配合使用的公共表表达式。

fullselect

指定要对函数返回的行。全查询中的列数必须与函数结果中的列数相匹配 (SQLSTATE 42811)。另外，全查询的静态列类型必须能够赋给函数结果的已声明列类型 (使用对列赋值的规则) (SQLSTATE 42866)。

不能对过程中的 RETURN 指定 *fullselect*。

如果该例程是标量函数或方法，则全查询必须返回一个列 (SQLSTATE 42823)，且最多只返回一行 (SQLSTATE 21000)。

如果该例程是行函数，则它必须也只能返回一行 (SQLSTATE 21505)。

如果该例程是表函数，则它可以返回零行或多行 (带有一列或多列)。

规则:

- SQL 函数或方法的执行必须以 RETURN 结束 (SQLSTATE 42632)。
- 在使用 *dynamic-compound-statement* 的 SQL 表或行函数中，只有位于复合语句末尾的 RETURN 语句才是允许的 (SQLSTATE 429BD)。

注意事项:

- 当从过程返回一个值时，调用程序可以使用下列各项访问该值:

- 当该 SQL 过程是从另一个 SQL 过程中调用的时，可以使用 GET DIAGNOSTICS 语句来检索 RETURN_STATUS
- 对 CLI 应用程序中的转义子句 CALL 语法 (?=CALL...) 中的返回值参数标记绑定的参数
- 当 SQLCODE 不小于零时，通过检索 SQLERRD[0] 的值，直接从（处理该 SQL 过程的 CALL 返回的）SQLCA 访问（当 SQLCODE 小于零时，假设 -1 值）。

示例： 使用 RETURN 语句来从 SQL 存储过程返回，并返回状态值零（如果成功）和 -200（如果不成功）。

```
BEGIN
...
  GOTO FAIL
...
  SUCCESS: RETURN 0
  FAIL: RETURN -200
END
```

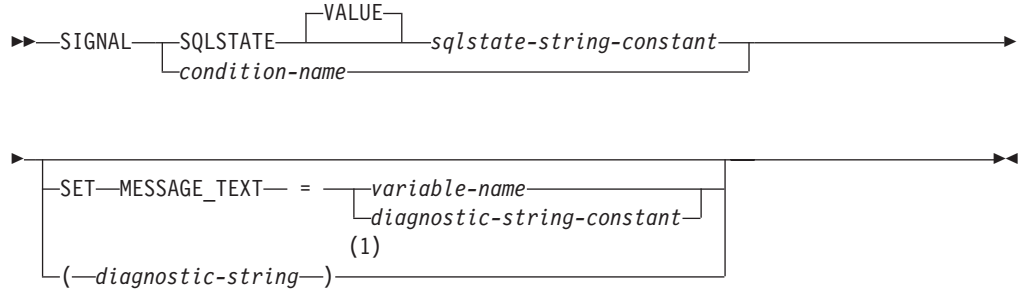
SIGNAL

不再使用 SIGNAL SQLSTATE Statement，而是采用这种用法。

SIGNAL

SIGNAL 语句用来指示错误或警告状态。它导致使用指定的 SQLSTATE 返回错误或警告以及可选的消息文本。

语法:



注:

1 此选项只在 CREATE TRIGGER 的作用域内提供，以与 DB2 的旧版本兼容。

描述:

SQLSTATE VALUE *sqlstate-string-constant*

指定的字符串常量表示 SQLSTATE。它必须是刚好包含 5 个字符的字符串常量，并遵循 SQLSTATE 的规则:

- 每个字符都必须来自数字集 ('0' 至 '9') 或非重音大写字母 ('A' 至 'Z')。
- SQLSTATE 类 (前两个字符) 不能是 '00'，因为这表示成功完成。

在动态复合语句、触发器、SQL 函数或 SQL 方法的上下文中，必须应用下列规则:

- SQLSTATE 类 (前两个字符) 不能是 '01' 或 '02'，因为它们不是错误类。
- 如果 SQLSTATE 类以数字 '0' 至 '6' 或字母 'A' 至 'H' 开头，则子类 (后三个字符) 必须以范围 'T' 至 'Z' 中的字母开头。
- 如果 SQLSTATE 类以数字 '7'、'8'、'9' 或字母 'T' 至 'Z' 开头，则子类可以是 '0' 至 '9' 或 'A' 至 'Z' 中的任何一个。

如果 SQLSTATE 不符合这些规则，则会发生错误 (SQLSTATE 428B3)。

condition-name

指定条件的名称。条件名在过程中必须是唯一的，且只能在声明它的复合语句内引用它。

SET MESSAGE_TEXT=

指定描述错误或警告的字符串。此字符串在 SQLCA 的 SQLERRMC 字段中返回。如果实际的字符串长于 70 个字节，则它会被截断，而不会有警告。仅当还指定了 SQLSTATE 或 condition-name 时，才能指定此子句 (SQLSTATE 42601)。

variable-name

标识必须在复合语句内声明的 SQL 变量。必须将该 SQL 变量定义为 CHAR 或 VARCHAR 数据类型。

diagnostic-string-constant

指定包含消息文本的字符串常量。

diagnostic-string

类型为 CHAR 或 VARCHAR 的表达式，它返回限长 70 个字节的字符串，以描述错误状态。如果此字符串长于 70 个字节，则它将被截断。此选项只在 CREATE TRIGGER 的作用域内提供，以与 DB2 的旧版本兼容。不建议进行常规使用。

注意事项:

- 如果发出 SIGNAL 语句，则指定的 SQLCODE 是:
 - +438, 如果 SQLSTATE 以 '01' 或 '02' 开头
 - 438, 别的情况
- 如果 SQLSTATE 或状态指示发出了异常（除 '01' 或 '02' 之外的 SQLSTATE）：
 - 如果信号语句所在的复合语句（或外部复合语句）中存在处理程序，且该复合语句包含用于指定的 SQLSTATE、condition-name 或 SQLEXCEPTION 的处理程序，则处理该异常，将控制传送至处理程序；
 - 如果不能处理该异常，则立即将控制返回值该复合语句的末尾。
- 如果 SQLSTATE 或状态指示发出了警告（SQLSTATE 类为 '01'）或“找不到”状态（SQLSTATE 类为 '02'）：
 - 如果信号语句所在的复合语句（或外部复合语句）中存在处理程序，且该复合语句包含用于指定的 SQLSTATE、condition-name、SQLWARNING（如果 SQLSTATE 类是 '01'）或 NOT FOUND（如果 SQLSTATE 类是 '02'）的处理程序，则处理该警告或“找不到”状态，将控制传送至处理程序；
 - 如果不能处理警告，则处理对下一语句继续。
- SQLSTATE 值由双字符的类码值组成，后跟三字符的子类码值。类码值表示成功和不成功的执行状态的类。

可以在 SIGNAL 语句中使用任何的有效 SQLSTATE 值。然而，建议程序员根据为应用程序保留的范围定义新的 SQLSTATE。这可以预防无意中使用了将来发行版的数据库管理器定义的 SQLSTATE 值。

- 可以定义以字符 '7' 至 '9' 或 'I' 至 'Z' 开头的 SQLSTATE 类。在这些类中，可以定义任何子类。
- 以字符 '0' 至 '6' 或 'A' 至 'H' 开头的 SQLSTATE 类是为数据库管理器保留的。在这些类中，以字符 '0' 至 'H' 开头的子类是为数据库管理器保留的。可以定义以字符 'I' 至 'Z' 开头的子类。

示例: 这是一个订购系统的 SQL 过程，当应用程序不知道客户号时，此过程发出应用程序错误。ORDERS 表包括 CUSTOMER 表的外键，这要求必须存在 CUSTNO 才能插入订单。

```
CREATE PROCEDURE SUBMIT_ORDER
(IN ONUM INTEGER, IN CNUM INTEGER,
 IN PNUM INTEGER, IN QNUM INTEGER)
SPECIFIC SUBMIT_ORDER
MODIFIES SQL DATA
LANGUAGE SQL
BEGIN
  DECLARE EXIT HANDLER FOR SQLSTATE VALUE '23503'
    SIGNAL SQLSTATE '75002'
      SET MESSAGE TEXT = 'Customer number is not known';
  INSERT INTO ORDERS (ORDERNO, CUSTNO, PARTNO, QUANTITY)
    VALUES (ONUM, CNUM, PNUM, QNUM);
END
```

Appendix A. SQL Limits

对表 33 “Database Manager Limits” 进行了更改。通过将注册表变量 DB2_INDEX_2BYTEVARLEN 设置为 ON，最长的变量索引键部件（以字节计）现在可以大于 255。

Appendix D. Catalog Views

已添加新的目录视图：

SYSCAT.SEQUENCES

视图 SYSCAT.SEQUENCES 会自动地为使用修订包 3 或更新版本创建的数据库生成。对于在修订包 3 之前创建的数据库，请运行 **db2updv7** 命令以将该视图添加至数据库。请参阅《发行说明》中的 Command Reference 更新以了解详细信息。

作为对 SQL 数据定义语句、环境例程和特定实用程序的响应，此目录视图会在正常操作期间更新。可通过正常的 SQL 查询设施获得目录视图中的数据。根据列所描述的对象类型，它们具有一致的名称。

表 30. SYSCAT.SEQUENCES 目录视图中的列

列名	数据类型	可空	描述
SEQSHEMA	VARCHAR(128)		序列的模式。
SEQNAME	VARCHAR(128)		序列名（由 DB2 为身份列生成）。
DEFINER	VARCHAR(128)		序列的定義者。
OWNER	VARCHAR(128)		序列的所有者。
SEQID	INTEGER		序列的内部标识。
SEQTYPE	CHAR(1)		序列类型 S - 正规序列
INCREMENT	DECIMAL(31,0)		增量值。
START	DECIMAL(31,0)		起始值。
MAXVALUE	DECIMAL(31,0)	是	最大值。
MINVALUE	DECIMAL(31,0)		最小值。
CYCLE	CHAR(1)		在遇到边界后是否发生循环： Y - 将发生循环 N - 不会发生循环
CACHE	INTEGER		要在内存进行预先分配以加快访问速度的序列值的数目。0 指示不预先分配值。
ORDER	CHAR(1)		序列号是否必须按请求次序生成： Y - 序列号必须按请求次序生成 N - 序列号不必按请求次序生成
DATATYPEID	INTEGER		对于内置类型，是内置类型的内部标识。对于单值类型，是单值类型的内部标识。
SOURCETYPEID	INTEGER		对于内置类型，值为 0。对于单值类型，这是作为单值类型的源类型的内置类型的内部标识。
CREATE_TIME	TIMESTAMP		序列的创建时间。

表 30. SYSCAT.SEQUENCES 目录视图中的列 (续)

列名	数据类型	可空	描述
ALTER_TIME	TIMESTAMP		对此序列执行上一 ALTER SEQUENCE 语句的时间。
PRECISION	SMALLINT		对具有小数或数字类型的序列定义的精度。值包括：5（表示 SMALLINT）、10（表示 INTEGER）和 19（表示 BIGINT）。
ORIGIN	CHAR(1)		序列起始地 U - 用户生成的序列 S - 系统生成的序列
REMARKS	VARCHAR(254)	是	用户提供的注释，或者是空的。

DB2 存储过程构建器

Java 1.2 支持 DB2 存储过程构建器

“DB2 存储过程构建器”支持使用 Java 1.2 功能来构建 Java 存储过程。此外，“存储过程构建器”通过使用 Java 1.2 中的双向支持来支持双向语言（如阿拉伯语和希伯来语）。

仅对 Windows NT 平台提供了此支持。

要使“存储过程构建器”识别和使用 Java 1.2 功能，必须安装 Java 1.2。

要安装 Java 1.2:

1. JDK 1.2.2 是在 DB2 UDB CD 上的 DB2\bid\NT 目录下提供的。
ibm-inst-n122p-win32-x86.exe 是安装程序，而 ibm-jdk-n122p-win32-x86.exe 是 JDK 分布式系统。将这两个文件复制至硬盘驱动器上的临时目录，然后从该位置运行安装程序。
2. 在 <DB2PATH>\java\Java12（其中 <DB2PATH> 是 DB2 的安装路径）下安装它。
3. 在 JDK/JRE 安装提示时不要选择 JDK/JRE 作为系统 VM。

在成功安装了 Java 1.2 之后，以正常方式启动“存储过程构建器”。

要使用 JDK 1.2 支持执行“Java 存储过程”，使用以下命令来将数据库服务器环境变量 DB2_USE_JDK12 设置为 TRUE:

```
DB2SET DB2_USE_JDK12=TRUE
```

同时，将 JDK11_PATH 设置为指向安装有 Java 1.2 支持的目录。通过使用以下命令来设置此路径:

```
DB2 UPDATE DBM CFG USING JDK11_PATH
```

要停止使用 Java 1.2，可从 <DB2PATH>\java\Java12，卸载 JDK/JRE，或只是重命名 <DB2PATH>\java\Java12 子目录。

要点：不要将 <DB2PATH>\java\Java12 与 <DB2PATH>\Java12 混淆。
<DB2PATH>\Java12 是 DB2 安装的一部分，它包括对 Java 1.2 的 JDBC 支持。

DB2 存储过程的远程调试

要在 UNIX 和 Windows 平台上使用 Java 和 C 存储过程的远程调试功能，需要安装 IBM Distributed Debugger。IBM Distributed Debugger 包括在“Visual Age for Java 专业版 CD”上。调试器客户机仅在 Windows 平台上运行。受支持的服务器平台包括：Windows、AIX 和 Solaris。使用“存储过程构建器”内置 SQL 调试功能来调试 Windows 和 UNIX 平台的本地和远程“SQL 存储过程”。对 OS/2 平台的支持此时不可用。有关“DB2 OS/390 版存储过程构建器”的详情，转至以下 Web 站点：
<http://www-4.ibm.com/software/data/db2/os390/spb/exciting>

要在 OS/390 平台上调试 SQL 过程，还必须具有 “IBM C/C++ Productivity Tools OS/390 R1 版” 产品。有关 “IBM C/C++ Productivity Tools OS/390 R1 版” 的详情，转至以下 Web 站点：<http://www.ibm.com/software/ad/c390/pt/>

在 Windows、OS/2 或 UNIX 平台上构建 SQL 过程

必须先配置服务器以供 “SQL 过程” 使用，才能使用 “存储过程构建器” 以在 Windows、OS/2 或 UNIX 数据库上成功地构建 “SQL 过程”。有关如何为 “SQL 过程” 配置服务器的信息，参见 第293页的『第 4 章 构建 Java 小应用程序和应用程序』。

在 Solaris 平台上使用 DB2 存储过程构建器

要在 Solaris 平台上使用 “DB2 存储过程构建器”：

1. 下载并安装 JDK 1.1.8。可从 JavaSoft web 站点下载 JDK 1.1.8。
2. 将环境变量 JAVA_HOME 设置为安装有 JDK 的位置。
3. 将 DB2 JDK11_PATH 设置为安装有 JDK 的目录。

要设置 DB2 JDK11_PATH 使用以下命令：

```
DB2 UPDATE DBM CFG USING JDK11_PATH.
```

已知问题和限制

- 目前，“SQL 过程” 在 Windows 98 上不受支持。
- 对于 Java 存储过程，JAR 标识、类名和方法名不能包含非 ASCII 字符。
- 在 AS/400 上，必须将下列 V4R4 PTF 应用于 OS/400 V4R4：
 - SF59674
 - SF59878
- 从数据库恢复存储过程时，带有字符子类型 FOR MIXED DATA 或 FOR SBCS DATA 的存储过程参数并未以源代码的形式显示在编辑器窗格中。
- 目前，从数据库检索 Java 源代码时会有问题。在进行检索时，代码中的注释会以折叠形式出现。这将影响在非 ASCII 代码页中工作的 “DB2 存储过程构建器” 用户，及其客户机和服务器在不同代码页上的用户。

将 DB2 存储过程构建器与繁体中文语言环境配合使用

在将 “Java 开发工具箱” 或 “Java 运行时 1.1.8” 与繁体中文语言环境配合使用时会出现问题。“存储过程构建器” 程序的图形部分（包括菜单、编辑器文本、消息等等）将不能正确地显示。解决方案是更改文件 `font.properties.zh_TW`，该文件在下列所有或其中一个目录中：

```
sqllib/java/jdk/lib  
sqllib/java/jre/lib
```

将：

```
monospaced.0=\u7d30\u660e\u9ad4,CHINESEBIG5_CHARSET,NEED_CONVERTED
```

更改为：

```
monospaced.0=Courier New,ANSI_CHARSET
```

UNIX (AIX、Sun Solaris 和 Linux) 安装和存储过程构建器

对于 Sun Solaris 安装，如果正将不同于安装在 AIX 上的“Java 开发工具箱”或“Java 运行时”与 UDB 配合使用，则必须将环境变量 JAVA_HOME 设置为安装有 Java 的路径（即，设置为包含有 /bin 和 /lib 目录的目录）。“存储过程构建器”在 Linux 上不受支持，但可在受支持的平台上使用它来在“DB2 UDB Linux 版”系统上构建和运行存储过程。

受支持的平台包括 AIX、Solaris 和 NT 客户机以及 AIX、Solaris、NT、Linux、OS/2、HP-UX 和 NUMA-Q 服务器。

在 OS/390 上构建 SQL 存储过程

“DB2 存储过程构建器”支持在“DB2 UDB OS390 版”的版本 7 服务器上构建 SQL 存储过程。

调试 SQL 存储过程

现在已经将在 Windows 和 UNIX 平台上调试 SQL 存储过程的功能直接集成到“DB2 存储过程构建器”中。当调试不受保护的（可信的）“SQL 过程”时，可以将 KEEPDIAG 数据库管理器配置选项设置为 YES 或 NO；但是，当调试受保护的（不可信的）“SQL 过程”时，必须将该选项设置为 YES（缺省值）。有关使用集成调试器的附加信息，参见“存储过程构建器”帮助。

导出 Java 存储过程

“DB2 存储过程构建器”现在支持导出 Java 存储过程。要导出 Java 存储过程：

1. 用鼠标右键单击存储过程文件夹，并单击“导出 Java 存储过程”，以便打开“导出 Java 存储过程”窗口。
2. 选择您想导出的存储过程，并将它们移到“所选的存储过程”这一列中。
3. 选择您的首选选项，然后单击“确定”。

在 OS/390 上插入存储过程

对于在 OS/390 上运行的“DB2 存储过程构建器”版本 5 和更新版本，如果您使用该向导来插入存储过程且指示没有 WLM 环境选项，则生成的代码中包含以下文本：NO WLM ENVIRONMENT。此行代码将导致存储过程如预期的那样在 SPAS 地址空间中运行。此修正解决了“DB2 存储过程构建器”版本 6 和更高版本上存在的问题。

在修正之后所生成的代码如下所示：

```
CREATE PROCEDURE SYSPROC.Proc2 ( )
  RESULT SETS 1
  LANGUAGE SQL
  MODIFIES SQL DATA
  COLLID TEST
  NO WLM ENVIRONMENT
  ASUTIME NO LIMIT
  RUN OPTIONS 'NOTEST(ALL,*,VADTCP&9.112.14.91:*)'
-----
-- SQL Stored Procedure
-----
```



```
P1: BEGIN
  -- Declare cursor
  DECLARE cursor1 CURSOR WITH RETURN FOR
    SELECT * FROM SYSIBM.SYSPROCEDURES;

  -- Cursor left open for client application
  OPEN cursor1;

END P1
```

在 workstation 服务器上为“SQL 存储过程”设置构建选项

通过在 UNIX 和 Windows 平台上使用“DB2 存储过程构建器”，可以为所有 SQL 存储过程设置构建选项。这些构建选项包括下列编译器和预编译器 DB2 注册表变量：

- DB2_SQLROUTINE_PREOPTS
- DB2_SQLROUTINE_COMPILER_PATH
- DB2_SQLROUTINE_COMPILE_COMMAND
- DB2_SQLROUTINE_KEEP_FILES

尽管可以使用 db2set 命令来设置这些注册表变量，但是，通过使用“存储过程构建器”，就不再需要物理地访问数据库服务器来发出命令，或者先停止然后再重新启动服务器来使更改生效。

要打开“SQL 存储过程构建选项”窗口，用鼠标右键单击项目视图中的数据库连接，然后单击“SQL 存储过程构建选项”。有关设置这些选项的详情，参见“DB2 存储过程”帮助。

自动刷新 WLM 地址空间以便在 OS/390 上构建存储过程

当您在 OS/390 上成功地构建了将在 WLM 中运行的存储过程之后，“DB2 存储过程构建器”将自动刷新 WLM 地址空间。

在 OS/390 上开发 Java 存储过程

“DB2 存储过程构建器”支持在“DB2 UDB OS/390 版的版本 6”和更高版本上开发 Java 存储过程。可以创建新的 Java 存储过程或者更改现存的存储过程。

为 MQ Series 和 OLE DB 构建 DB2 表用户定义函数 (UDF)

“DB2 存储过程构建器”提供了向导来帮助您在 MQSeries 和 OLE DB 创建表 UDF。可以使用“创建 OLE DB 表 UDF”向导来访问 OLE DB 数据提供者。该向导创建 OLE 表 UDF，并可选择创建表视图。可以使用“创建 MQSeries 表 UDF”向导来创建具有选项表视图的表 UDF 用来访问 MQSeries 消息，并对采用表格形式的数据进行语法分析。

Unicode 更新

介绍

Unicode 标准是编写的字符和文本的通用字符编码方案。Unicode 是字符的多字节表示法。它定义对多语言文本进行编码所采用的一致方式，这允许在国际间进行文本数据交换，同时也创建了软件全球化的基础。

Unicode 提供了以下两种编码方案。缺省编码方案是 UTF-16，它采用 16 位编码格式。UCS-2 是 UTF-16 的子集，它使用两个字节来表示一个字符。UCS-2 通常被认为是能够表示所有现存单字节和双字节代码页中的所有必需字符的通用代码页。UCS-2 在 IBM 公司被注册为代码页 1200。

另一 Unicode 编码格式是 UTF-8，它是面向字节的，并且已经被设计为更易于配合现存的基于 ASCII 的系统使用。UTF-8 使用变化的字节数（通常为 1-3，有时为 4）来存储每个字符。不可变的 ASCII 字符被作为单个字节存储下来。所有其他字符都是使用多字节来存储的。通常，UTF-8 数据可被未对多字节代码页指定的代码视作扩充 ASCII 数据。UTF-8 在 IBM 公司被注册为代码页 1208。

非常重要的一点是，当在本地代码页、UCS-2 和 UTF-8 之间转换数据时，应用程序应考虑这些数据的需求。例如，在 UCS-2 中，20 个字符刚好需要 40 个字节，而在 UTF-8 中，有时需要 20 到 60 个字节，这取决于原始代码页和所使用的字符。

DB2 Unicode 数据库和应用程序

使用 UTF-8 代码集创建的“DB2 通用数据库 Unix 版、Windows 版或 OS/2 版”可以用来存储 UCS-2 和 UTF-8 格式的数据。这种数据库被称为 Unicode 数据库。SQL CHAR 数据是使用 UTF-8 编码的，而 SQL GRAPHIC 数据是使用 UCS-2 编码的。这相当于将“单字节” (SBCS) 和“多字节” (MBCS) 代码集存储在 CHAR 列中，而将“双字节” (DBCS) 代码集存储在 GRAPHIC 列中。

应用程序的代码页可能与 DB2 用来存储数据的代码页不匹配。在非 Unicode 数据库中，当代码页不相同，数据库管理器会对在客户机和服务器之间传送的字符和图形（纯 DBCS）数据进行转换。在 Unicode 数据库中，客户机代码页与 UTF-8 之间的字符数据转换是由数据库管理器自动执行的，但是所有图形 (UCS-2) 数据在传送时在客户机和服务器之间没有执行任何转换。

数据库管理器执行的代码页转换

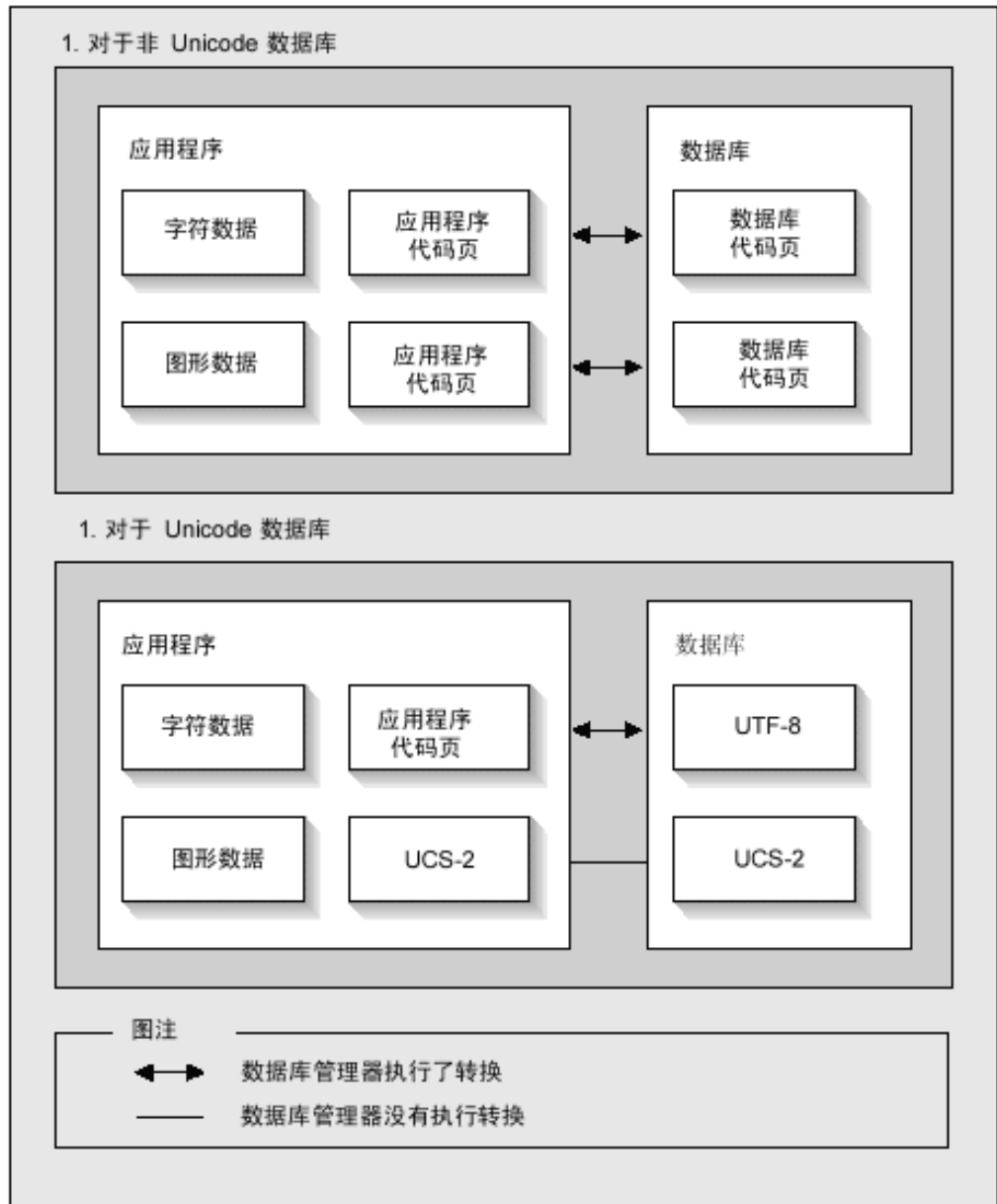


图 7. 数据库管理器执行的代码页转换

注:

1. 当连接至 Unicode 数据库时，如果应用程序设置为 `DB2CODEPAGE=1208`，且本地代码页为 UTF-8，则不需要执行代码页转换。
2. 当连接至 Unicode 数据库时，CLI 应用程序还可以将字符数据作为图形数据接收，及将图形数据作为字符数据接收。

应用程序可以指定 UTF-8 代码页，指示它将发送和接收采用 UCS-2 格式的所有图形数据以及采用 UTF-8 的字符数据。此应用程序代码页仅用于 Unicode 数据库时才是受支持的。

使用 Unicode 时应考虑的其他事项:

1. 数据库代码页是在创建数据库时确定的，并且缺省情况下，它的值是根据操作系统语言环境（或代码页）来确定的。可以使用 `CODESET` 和 `TERRITORY` 关键字来显式创建 Unicode DB2 数据库。例如：

```
CREATE DATABASE unidb USING CODESET UTF-8 TERRITORY US
```

2. 应用程序代码页也被缺省为本地代码页，但是它可以被 UTF-8 以以下两种方式的其中一种进行覆盖：

- 用以下命令来将应用程序代码页设置为 UTF-8 (1208)：

```
db2set DB2CODEPAGE=1208
```

- 对于 CLI/ODBC 应用程序，通过调用 `SQLSetConnectAttr()` 并将 `SQL_ATTR_ANSI_APP` 设置为 `SQL_AA_FALSE` 进行。缺省设置为 `SQL_AA_TRUE`。

3. `GRAPHIC` 列中的数据将对每个 Unicode 字符采用刚好两个字节，而 `CHAR` 列中的数据将对每个 Unicode 字符采用 1 到 3 个字节。在字符的 SQL 限制方面，`GRAPHIC` 列中的字符的 SQL 限制通常是 `CHAR` 列中的字符的 SQL 限制的一半，但是字节数方面是相等的。`CHAR` 列的最大字符长度为 254。`GRAPHIC` 列的最大字符长度为 127。有关详情，参见 *SQL Reference* 的“功能”这一章中的 `MAX`。

4. 图形文字与带有 G 前缀的字符文字是不同的。例如：

```
SELECT * FROM mytable WHERE mychar = 'utf-8 data' AND mygraphic = G'ucs-2 data'
```

注：Unicode 数据库不需要 G 前缀。

有关详情及更新支持，参见第418页的『Unicode 数据库中的文字』。

5. 对 CLI/ODBC 和 JDBC 应用程序的支持与对嵌入式应用程序的支持有所不同。有关特定于 CLI/ODBC 支持的信息，参见第416页的『CLI Guide and Reference』。
6. 平台间的 UCS-2 数据的字节排序可能有所不同。在平台内部，DB2 使用大尾数法格式。

文档更新

在将 Unicode 配合“DB2 版本 7.1”使用时，此文档更新了下列信息：

- *SQL Reference*:

Chapter 3 Language Elements

Chapter 4 Functions

- *CLI Guide and Reference*:

Chapter 3. Using Advanced Features

Appendix C. DB2 CLI and ODBC

- *Data Movement Utilities Guide and Reference*的附录 C『导出 / 导入 / 装入实用程序文件格式』

有关将 Unicode 配合 DB2 使用的详情，参考《管理指南》的附录 J 国家语言支持 (NLS)：“DB2 UDB 中的 Unicode/UCS-2 和 UTF-8 支持”。

Chapter 3 Language Elements

数据类型提升

在本节中，表 5 显示每个数据类型的优先权列表。请注意：

- 对于 Unicode 数据库，下列各项被认为是数据类型：
 - CHAR 和 GRAPHIC
 - VARCHAR 和 VARGRAPHIC
 - LONG VARCHAR 和 LONG VARGRAPHIC
 - CLOB 和 DBCLOB
- 在 Unicode 数据库中，可以创建这样的函数：函数签名中的唯一区别在于等效的 CHAR 和 GRAPHIC 数据类型之间的区别。例如，foo(CHAR(8)) and foo(GRAPHIC(8))。我们强烈建议您不要定义这样的重复函数，因为迁移到未来的发行版时将要求在继续迁移之前删除其中一个函数。

若确实存在这样的重复函数，则选择取消哪个函数是由两遍算法来确定的。第一遍尝试使用用于解析非 Unicode 数据库中的函数的算法来查找匹配项。若找不到匹配项，则会根据 CHAR 和 GRAPHIC 字符串的以列提升优先权来执行第二遍算法：

GRAPHIC->CHAR->VARGRAPHIC->VARCHAR->LONG VARGRAPHIC->LONG VARCHAR->DBCLOB->CLOB

在数据类型之间强制转型

以下项已经添加至列表，并指出“涉及单值类型的下列强制转型是受支持的”：

- 对于 Unicode 数据库，从 VARCHAR 或 VARGRAPHIC 强制转型为其源数据类型为 CHAR 或 GRAPHIC 的单值类型 DT。

以下是对表 6 受支持的内置数据类型之间的强制转型的更新。只包括了该表中受影响的行。

表 31. 受支持的内置数据类型之间的强制转型

目标数据类型 →	CH A R	V A R C H A R	L O N G V A R C H A R	CL O B	G R A P H I C	V A R G R A P H I C	L O N G V A R G R A P H I C	D B C L O B	C L O B
源数据类型 ↓									
CHAR	Y	Y	Y	Y	Y ¹	Y ¹	-	-	
VARCHAR	Y	Y	Y	Y	Y ¹	Y ¹	-	-	
LONGVARCHAR	Y	Y	Y	Y	-	-	Y ¹	Y ¹	

表 31. 受支持的内置数据类型之间的强制转型 (续)

目标数据类型 →	CH A R	V A R C H A R	L O N G V A R C H A R	CL O B	G R A P H I C	V A R C H A R	L O N G V A R C H A R	D B C L O B
源数据类型 ↓								
CLOB	Y	Y	Y	Y	-	-	-	Y ¹
GRAPHIC	Y ¹	Y ¹	-	-	Y	Y	Y	Y
VARGRAPHIC	Y ¹	Y ¹	-	-	Y	Y	Y	Y
LONGVARGRAPHIC	-	-	Y ¹	Y ¹	Y	Y	Y	Y
DBCLOB	-	-	-	Y ¹	Y	Y	Y	Y

¹ 只有 Unicode 数据库才支持强制转型。

指定和比较

涉及到字符和图形数据的指定和比较仅当其中一个字符串为文字时才受支持。对于函数解析，图形文字和字符文字都将与字符和图形函数参数相匹配。

以下是对表 7. 用于指定和比较的数据类型的更新。只包括了受影响的表行和新脚注 ⁶：

操作数	二进制 整数	十进制数	浮点	字符串	图形字 符串	日期	时间	时间戳 记	二进制 字符串	UDT
字符串	否	否	否	是	是 ⁶	¹	¹	¹	否 ³	²
图形字符串	否	否	否	是 ⁶	是	否	否	否	否	²

⁶ 仅对于 Unicode 数据库才是受支持的。

字符串分配:

存储器指定

这一小节的最后一段被修改为如下所示:

当为定长列指定了字符串，而字符串长度小于目标的长度属性时，则使用必需数目的单字节、双字节或 UCS-2² 空格来填充该字符串的右边。填充字符通常是空格，即使对于用 FOR BIT DATA 属性定义的列。

检索指定

这一小节的第三段被修改为如下所示:

当为定长变量指定了字符串，而字符串长度小于目标的长度属性时，则用必需数目的单字节、双字节或 UCS-2² 空格来填充该字符串的右边。填充字符通常是空格，即使对于用 FOR BIT DATA 属性定义的字符串。

² UCS-2 定义了几个具有不同特性的 SPACE 字符。对于 Unicode 数据库，数据库管理器总是将 x'0020' 处的 ASCII SPACE 用作 UCS-2 空格。对于 EUC 数据库，位置 x'3000' 处的 IDEOGRAPHIC SPACE 被用于填充 GRAPHIC 字符串。

字符串指定的转换规则

已经将下图添加至这一小节的结尾：

对于 Unicode 数据库，可为图形列指定字符串，也可为字符列指定图形字符串。

用于图形字符串指定的 DBCS 注意事项

这一小节的第一段已被修改为如下所示：

图形字符串指定是按照类似于处理字符串的指定方式来处理的。对于非 Unicode 数据库，图形字符串数据类型只与其他图形字符串数据类型兼容，而决不与数字、字符串或日期时间数据类型兼容。对于 Unicode 数据库，图形字符串数据类型与字符串数据类型是兼容的。

字符串比较：

用于比较的转换规则

这一小节已被修改为如下所示：

当两个字符串进行比较时，其中一个字符串会首先转换为另一字符串的编码方案和 / 或代码页（如果需要的话）。有关详细信息，参见 *SQL Reference* 中的 Chapter 3 Language Elements 的『用于字符串转换的规则』一节。

用于结果数据类型的规则

Unicode 数据库中的字符和图形字符串：这是插入到“图形字符串”小节后面的新的小节。

在 Unicode 数据库中，字符串和图形字符串是兼容的。

若一个操作数是...	而另一个操作数是...	结果的数据类型是...
GRAPHIC(x)	CHAR(y) 或 GRAPHIC(y)	GRAPHIC(z) 其中 z = max(x,y)
VARGRAPHIC(x)	CHAR(y) 或 VARCHAR(y)	VARGRAPHIC(z) 其中 z = max(x,y)
VARCHAR(x)	GRAPHIC(y) 或 VARGRAPHIC(y)	VARGRAPHIC(z) 其中 z = max(x,y)
LONG VARGRAPHIC	CHAR(y) 或 VARCHAR(y)	LONG VARGRAPHIC 或 LONG VARCHAR
LONG VARCHAR	GRAPHIC(y) 或 VARGRAPHIC(y)	LONG VARGRAPHIC 或 LONG VARCHAR
DBCLOB(x)	CHAR(y) 或 VARCHAR(y)	DBCLOB(z) 其中 z = max(x,y) 或 CLOB(y)

若一个操作数是...	而另一个操作数是...	结果的数据类型是...
DBCLOB(x)	LONG VARCHAR	DBCLOB(z) 其中 $z = \max(x, 16350)$
CLOB(x)	GRAPHIC(y) VARGRAPHIC(y)	或 DBCLOB(z) 其中 $z = \max(x, y)$
CLOB(x)	LONG VARGRAPHIC	DBCLOB(z) 其中 $z = \max(x, 16350)$

用于字符串转换的规则

第三点已添加至本节的以下列表:

对于每一对代码页, 结果是由下列规则的结果应用程序确定的:

- 若代码页相等, 则结果就是该代码页。
- 若其中一个代码页是 BIT DATA (代码页 0), 则结果代码页是 BIT DATA。
- 在 Unicode 数据库中, 若一个代码页采用与另一个代码页的编码方案不相同的编码方案来表示数据, 则结果为 UCS-2 到 UTF-8 (即, 图形数据类型到字符数据类型)。¹
- 否则, 就由 *SQL Reference* 中的 Chapter 3 Language Elements 的『用于字符串转换的规则』一节中的“表 8”来确定结果代码页。该表中的“第一”这一项表示选择第一个操作数的代码页, “第二”这一项表示选择了第二个操作数的代码页。

¹ 在非 Unicode 数据库中, 不同编码方案之间的转换不受支持。

表达式

已经添加了以下内容:

在 Unicode 数据库中, 接受字符或图形字符串的表达式将接受支持转换的所有字符串类型。

对于串联运算符: 已经将以下内容添加至这一小节的结尾:

在 Unicode 数据库中, 同时涉及字符串操作数和图形字符串操作数的串联会首先将字符操作数转换为图形操作数。注意, 在非 Unicode 数据库中, 串联不能同时涉及到字符操作数和图形操作数。

谓词

以下项已添加至由语句“下列规则适用于所有类型的谓词”所指示的列表:

- 在 Unicode 数据库中, 接受字符或图形字符串的所有谓词将接受支持转换的所有字符串类型。

第 4 章 功能

标量函数

已经将以下语句添加至这一小节的结尾:

在 Unicode 数据库中, 接受字符或图形字符串的所有标量函数都将接受支持转换的所有字符串类型。

Chapter 3. Using Advanced Features

以下是本章中新增的一节:

编写 DB2 CLI Unicode 应用程序

对“DB2 CLI Unicode 应用程序”的支持有两个主要方面:

1. 添加一组可以接受 Unicode 字符串自变量替换 ANSI 字符串自变量的函数。
2. 添加新的 C 和 SQL 数据类型, 以将数据作为 ANSI 或 Unicode 数据描述。下列各节提供了有关这两个方面的详情。要考虑 Unicode 应用程序, 应用程序必须将 SQL_ATTR_ANSI_APP 连接属性设置为 SQL_AA_FALSE 之后才能进行连接。这将确保 CLI 将作为 Unicode 客户机来连接, 而对于 CHAR 数据, 所有 Unicode 数据将采用 UTF-8 格式发送, 对于 GRAPHIC 数据则采用 UCS-2 格式进行发送。

Unicode 函数: 以下是同时支持 Unicode (W) 和 ANSI (A) 版本的 ODBC API 函数的列表 (对于 Unicode, 函数名中将有一个 W):

SQLBrowseConnect	SQLForeignKeys	SQLPrimaryKeys
SQLColAttribute	SQLGetConnectAttr	SQLProcedureColumns
SQLColAttributes	SQLGetConnectOption	SQLProcedures
SQLColumnPrivileges	SQLGetCursorName	SQLSetConnectAttr
SQLColumns	SQLGetDescField	SQLSetConnectOption
SQLConnect	SQLGetDescRec	SQLSetCursorName
SQLDataSources	SQLGetDiagField	SQLSetDescField
SQLDescribeCol	SQLGetDiagRec	SQLSetStmtAttr
SQLDriverConnect	SQLGetInfo	SQLSpecialColumns
SQLDrivers	SQLGetStmtAttr	SQLStatistics
SQLError	SQLNativeSQL	SQLTablePrivileges
SQLExecDirect	SQLPrepare	SQLTables

总是返回或采用字符串长度自变量的 Unicode 函数被作为字符计数来传送。对于返回服务器数据的长度信息的函数, 显示大小和精度是以字符数描述的。当长度 (数据的传送大小) 可能涉及字符串或非字符串数据时, 长度是以八位字节长度来描述的。例如, SQLGetInfoW 将仍然采用字节计数形式的长度, 但是 SQLExecDirectW 将使用字符计数。CLI 将以 Unicode 或 ANSI 形式返回结果集, 这要视应用程序的绑定情况而定。若应用程序绑定至 SQL_C_CHAR, 则驱动程序会将 SQL_WCHAR 数据转换为 SQL_CHAR。驱动程序管理器将 SQL_C_WCHAR 映射为 ANSI 驱动程序的 SQL_C_CHAR, 但它不会对 Unicode 驱动程序进行任何映射。

新的数据类型和有效转换: CLI 或 ODBC 定义了两个新的数据类型, 即, SQL_C_WCHAR 和 SQL_WCHAR。SQL_C_WCHAR 指示 C 缓冲区包含 UCS-2 数据。SQL_WCHAR 指示特定列或参数标记包含 Unicode 数据。对于“DB2 Unicode 服务器”, 图形列将被描述为 SQL_WCHAR。允许在 SQL_C_WCHAR 与 SQL_CHAR、SQL_VARCHAR、SQL_LONGVARCHAR 和 SQL_CLOB 之间进行转换, 以及在图形数据类型之间进行转换。

表 32. 受支持的数据转换

SQL 数据类型	SQL_C_CHAR	SQL_C_WCHAR	SQL_C_LONG	SQL_C_SHORT	SQL_C_TINYINT	SQL_C_FLOAT	SQL_C_DOUBLE	SQL_C_TYPE_DATE	SQL_C_TYPE_TIME	SQL_C_TYPE_TIMESTAMP	SQL_C_BINARY	SQL_C_BIT	SQL_C_DBCHAR	SQL_C_CLOB	SQL_C_BLOB	SQL_C_DBCLOB	SQL_C_BIGINT	SQL_C_NUMERIC
BLOB	X	X									D				X			
CHAR	D	X	X	X	X	X	X	X	X	X	X	X					X	X
CLOB	D	X									X			X				
DATE	X	X						D		X								
DBCLOB		X									X		D			X		
DECIMAL	D	X	X	X	X	X	X				X	X					X	X
DOUBLE	X	X	X	X	X	X	D					X					X	X
FLOAT	X	X	X	X	X	X	D					X					X	X
GRAPHIC (非 Unicode)	X	X											D					
GRAPHIC(Unicode)	X	X	X	X	X	X	X	X	X	X	X	X	D				X	
INTEGER	X	X	D	X	X	X	X					X					X	X
LONG VARCHAR	D	X									X							
LONG VARGRAPHIC (非 Unicode)	X	X									X		D					
LONG VARGRAPHIC(Unicode)	X	X									X		D					
NUMERIC	D	X	X	X	X	X	X					X						X
REAL	X	X	X	X	X	D	X					X						X
SMALLINT	X	X	X	D	X	X	X					X					X	X
BIGINT	X	X	X	X	X	X	X				X	X					D	X
TIME	X	X							D	X								
TIMESTAMP	X	X						X	X	D								
VARCHAR	D	X	X	X	X	X	X	X	X	X	X	X					X	X
VARGRAPHIC (非 Unicode)	X	X											D					
VARGRAPHIC(Unicode)	X	X	X	X	X	X	X	X	X	X	X	X	D				X	

注:

D 转换是受支持的。这是 SQL 数据类型的缺省转换。

X 所有 IBM DBMS 都支持该转换。

空白 没有任何 IBM DBMS 支持该转换。

- 数据未转换为 LOB 定位器类型，反而是定位器表示数据值，参考使用大对象以获取详情。
- SQL_C_NUMERIC 只有在 32 位 Windows 操作系统上才可用。

过时的关键字/补丁值: 在 Unicode 应用程序受支持之前，可以通过一系列 `cli.ini` 文件关键字（例如 `GRAPHIC=1, 2 或 3`, `Patch2=7` 等等）来使编写用来处理单字节字符数据的应用程序可处理双字节图形数据。这些解决方法将图形数据视作字符数据，同时还会影响所报告的数据长度。

Unicode 应用程序不再需要并且事实上不应使用这些关键字，否则，可能会导致严重的负面影响。若不知道特定应用程序是否是 Unicode 应用程序，我们建议您不要尝试会影响图形数据处理的任何关键字。

Unicode 数据库中的文字: 在非 Unicode 数据库中，`LONG VARCHAR` 和 `LONG VARCHAR` 列中的数据不能进行比较。`GRAPHIC/VARGRAPHIC` 和 `CHAR/VARCHAR` 列中的数据只能使用显式的强制转型函数来进行相互比较或指定，因为隐式的代码页转换不受支持。这包括 `GRAPHIC/VARGRAPHIC` 和 `CHAR/VARCHAR` 文字，其中 `GRAPHIC/VARGRAPHIC` 文字与 `CHAR/VARCHAR` 文字的区别就在于前缀 `G`。

对于 Unicode 数据库，不需要在 `GRAPHIC/VARGRAPHIC` 和 `CHAR/VARCHAR` 文字之间进行强制转型。而且，`GRAPHIC/VARGRAPHIC` 文字前不需要前缀 `G`。如果至少其中一个自变量是文字，则会进行隐式转换。这允许在使用 `SQLPrepareW()` 或 `SQLExecDirect()` 的语句中使用带有前缀 `G` 或者不带前缀 `G` 的文字。`LONG VARCHAR` 的文字仍必须带有前缀 `G`。

有关详情，请参见 *SQL Reference* 的 Chapter 3 Language Elements 中的『在数据类型之间强制转型』。

新的 CLI 配置关键字: 已添加下列三个关键字，以避免 Unicode 应用程序连接至数据库时带来的额外开销。

1. DisableUnicode

关键字描述:

禁用对 Unicode 的基本支持。

db2cli.ini 关键字语法:

`DisableUnicode = 0 | 1`

缺省设置:

0 (false)

DB2 CLI/ODBC 设置标签:

不能使用“CLI/ODBC 设置”笔记本设置此关键字。必须直接修改 `db2cli.ini` 文件才能使用此关键字。

使用说明:

在启用了 Unicode 支持，且由 Unicode 应用程序调用时，CLI 将尽可能地使用最佳的客户机代码页尝试连接至数据库，以确保没有因代码页转换带来的不必要的数据损失。这可能会导致连接时间的增加（因为代码页进行了交换），或者可能导致客户机上进行添加此支持之前不会发生的代码页转换。

将此关键字设置为 True 将导致所有 Unicode 数据在发送至服务器之前先被转换为应用程序的本地代码页。这会导致不能以本地代码页表示的所有数据丢失。

2. ConnectCodepage

关键字描述:

指定要在连接至数据源时使用的特定代码页，以避免额外的连接开销。

db2cli.ini 关键字语法:

ConnectCodepage = 0 | 1 <任何有效 db2 代码页>

缺省设置:

0

DB2 CLI/ODBC 设置标签:

不能使用“CLI/ODBC 设置”笔记本设置此关键字。必须直接修改 db2cli.ini 文件才能使用此关键字。

使用说明:

非 Unicode 应用程序总是使用应用程序的本地代码页或 DB2Codepage 环境设置来连接至该数据库。缺省情况下，CLI 将确保 Unicode 应用程序使用 UTF-8 和 UCS-2 代码页连接至 Unicode 数据库，而使用非 Unicode 数据库的代码页连接至该数据库。这确保没有因代码页转换造成的不必要的损失。

此关键字允许用户在连接至非 Unicode 数据库时指定数据库的代码页，以避免在连接时带来任何额外的开销。

指定值 1 会导致 SQLDriverConnect() 以输出连接字符串的形式返回正确的值，以便可在将来的 SQLDriverConnect() 调用上使用该值。

3. UnicodeServer

关键字描述:

指示数据源是“Unicode 服务器”。等效于设置 ConnectCodepage=1208。

db2cli.ini 关键字语法:

UnicodeServer = 0 | 1

缺省设置:

0

DB2 CLI/ODBC 设置标签:

不能使用“CLI/ODBC 设置”笔记本设置此关键字。必须直接修改 db2cli.ini 文件才能使用此关键字。

使用说明:

此关键字等效于 ConnectCodepage=1208，添加它仅出于方便起见。设置此关键字以避免在连接至“DB2 OS/390 版的版本 7”或更高版本时带来的额外连接开销。不需要对“DB2 Windows 版”、“DB2 Unix 版”或“DB2 OS/2 版”数据库设置此关键字，因为不需要进行任何额外的处理。

附录 C DB2 CLI 和 ODBC

以下是添加至本附录的新一节:

ODBC Unicode 应用程序

Unicode ODBC 应用程序主要采用 UCS-2 格式来发送和检索字符数据。它通过调用 Unicode 版本的 ODBC 函数 ('W' 前缀) 并指示 Unicode 数据类型来完成此操作。应用程序不会显式地指定本地代码页。该应用程序还可以调用 ANSI 函数并传送本地的代码页字符串。

例如, 应用程序可以调用 `SQLConnectW()`, 并将 DSN、“用户标识”和“口令”作为 Unicode 自变量来传送。接着, 它可以调用 `SQLExecDirectW()`, 并以 Unicode SQL 语句字符串的形式进行传送, 然后绑定 ANSI 本地代码页缓冲区 (`SQL_C_CHAR`) 与 Unicode 缓冲区 (`SQL_C_WCHAR`) 的组合。数据库数据类型可以是本地代码页或 UCS-2 和 UTF-8。

如果 CLI 应用程序调用 `SQLConnectW` 或调用 `SQLSetConnectAttr`, 并将 `SQL_ATTR_ANSI_APP` 设置为 `SQL_AA_FALSE`, 则该应用程序被认为是 Unicode 应用程序。这意味着所有 CHAR 数据都是以 UTF-8 格式从数据库中发送和检索的。然后, 该应用程序可以将 CHAR 数据取装到本地代码页中的 `SQL_C_CHAR` 缓冲区中(这可能会丢失数据), 或以 UCS-2 格式取装到 `SQL_C_WCHAR` 缓冲区中, 而不会丢失任何数据。

若应用程序未执行以上两个调用中的任何一个, 则 CHAR 数据会在服务器上被转换为应用程序本地代码页。这意味着将 CHAR 数据取装到 `SQL_C_WCHAR` 中可能会导致数据丢失。

若 `DB2CODEPAGE` 实例变量被设置为代码页 1208 (UTF-8) (使用 `db2set`), 则应用程序将接收采用 UTF-8 格式的所有 CHAR 数据, 因为这现在是本地代码页。应用程序还必须确保所有 CHAR 输入数据也采用 UTF-8 格式。ODBC 还假设所有 `SQL_C_WCHAR` 数据都采用本机结尾格式。CLI 将对 `SQL_C_WCHAR` 执行任何必需的字节逆转。

ODBC Unicode 应用程序与非 Unicode 应用程序: “DB2 通用数据库”的此发行版中包含 `SQLConnectW()` API。Unicode 驱动程序必须导出 `SQLConnectW`, 以便被驱动程序管理器识别为 Unicode 驱动程序。注意, 许多 ODBC 应用程序 (如 Microsoft Access 和 Visual Basic) 都调用 `SQLConnectW()`, 这一点很重要。在“DB2 通用数据库”的前发行版中, DB2 CLI 不支持此 API, 因此, 也不会被 ODBC 驱动程序管理器识别为 Unicode 驱动程序。这导致 ODBC 驱动程序管理器将所有 Unicode 数据转换为应用程序的本地代码页。借助添加的 `SQLConnectW()` 函数支持, 现在这些应用程序将作为 Unicode 应用程序来连接, 且 DB2 CLI 将会对所有必需的数据转换进行管理。

DB2 CLI 现在接受 Unicode API (带有前缀 "W") 和常规 ANSI API。ODBC 定义一组前缀为 "A" 的函数, 但驱动程序管理器不会将带有前缀 "A" 的 ANSI 函数传送到驱动程序。而是, 它会将这些函数转换为不带前缀的 ANSI 函数调用, 然后将它们传送到驱动程序。

调用 `SQLConnectW()` API 的 ODBC 应用程序被视作 Unicode 应用程序。由于 ODBC 驱动程序管理器总是会调用 `SQLConnectW()` API, 而不管调用的是哪个版本的应用程序, 所以, ODBC 引入了 `SQL_ATTR_ANSI_APP` 连接属性以通知驱动程序该应用程序

| 应视作 ANSI 应用程序还是 UNICODE 应用程序。若 SQL_ATTR_ANSI_APP 未设置
| 为 SQL_AA_FALSE，则 DB2 CLI 会在将 Unicode 数据发送至服务器之前转换为本地
| 代码页。

| **Data Movement Utilities Guide and Reference**

| **附录 C 导出 / 导入 / 装入实用程序文件格式**

| 已经将下列更新添加至此附录：

| 当导出、导入和装入实用程序配与连接至非 Unicode 数据库的 Unicode 客户机配合使
| 用时，这些实用程序是不受支持的。仅当 Unicode 客户机连接至 Unicode 数据库时，
| Unicode 客户机文件才受支持。

第7部分 连接至主机系统

Connectivity Supplement	425
在 VM 环境中设置应用程序服务器	425
CLI/ODBC/JDBC 配置 PATCH1 和 PATCH2 设置	425

Connectivity Supplement

在 VM 环境中设置应用程序服务器

在 "Provide Network Information" 一节的 "Defining the Application Server" 小节中，在第一个（且是唯一的）句子后面添加以下句子：

```
The RDB_NAME is provided on the SQLSTART EXEC as the DBNAME parameter.
```

CLI/ODBC/JDBC 配置 PATCH1 和 PATCH2 设置

CLI/ODBC/JDBC 驱动程序可通过“客户机配置辅助程序”或“ODBC 驱动程序管理器”来进行配置（如果它安装在系统上的话），或者通过人工编辑 db2cli.ini 文件来进行配置。有关详情，参见《安装和配置补遗》或 *CLI Guide and Reference*。

可通过 db2cli.ini 文件，或通过 SQLDriverConnect() 或 SQLBrowseConnect() CLI API，为 PATCH1 和 PATCH2 关键字指定值，来修改 DB2 CLI/ODBC 驱动程序的缺省行为。

将用户要设置的所有关键字加在一起指定 PATCH1 关键字。例如，如果指定了补丁 1、2 和 8，则 PATCH1 将具有值 11。以下是每个关键字值以及它对驱动程序的影响的描述：

- 1 - 这会使驱动程序搜索 "count(exp)"，并将其替换为 "count(distinct exp)"。这是必需的，因为 DB2 的某些版本支持 "count(exp)" 语法，而该语法是由某些 ODBC 应用程序生成的。这是服务器不支持 "count(exp)" 语法时 Microsoft 应用程序所需的。
- 2 - SQLGetTypeInfo() 函数中对 LITERAL_PREFIX 或 LITERAL_SUFFIX 列返回 SQL_NULL_DATA 时，某些 ODBC 应用程序被捕获。这会强制驱动程序反而返回空的字符串。这是 Impromptu 2.0 所需的。
- 4 - 如果时间戳记的时间和分数部分被设置为零的话，这会强制驱动程序将输入时间戳记数据视作日期数据，这是 Microsoft Access 所需的。
- 8 - 这会强制驱动程序将输入时间戳记数据视作时间数据（如果时间戳记的日期部分为 1899-12-30 的话）。这是 Microsoft Access 所需要的。
- 16 - 未使用。
- 32 - 这会强制驱动程序不返回有关 SQL_LONGVARCHAR、SQL_LONGVARBINARY 和 SQL_LONGVARGRAPHIC 列的信息。对于应用程序，它表现得好像不支持长字段一样。这是 Lotus 123 所需的。
- 64 - 这会强制驱动程序使终止图形输出字符串为 NULL。这是双字节环境中的 Microsoft Access 所需要的。
- 128 - 这会强制驱动程序让查询 "SELECT Config, nValue FROM MSysConf" 转至服务器。驱动程序当前返回一个错误，相关联的 SQLSTATE 值为 S0002（找不到表）。如果用户已经在数据库中创建了此配置表，且想要应用程序访问它，则这是必需的。
- 256 - 这会强制驱动程序在 SQLStatistics() 调用中首先返回主键列，驱动程序当前返回了根据索引名排序的索引，这就是标准的 ODBC 行为。
- 512 - 这会强制驱动程序在 SQLGetFunctions() 中对 SQL_API_SQLTABLEPRIVILEGES 和 SQL_API_SQLCOLUMNPRIVILEGES

返回 FALSE。

- 1024 - 这会强制驱动程序在 SQLExecute() 或 SQLExecDirect() 中返回 SQL_SUCCESS 而不是 SQL_NO_DATA_FOUND, 如果执行的 UPDATE 或 DELETE 语句未影响任何行的话。这是 Visual Basic 应用程序所需的。
- 2048 - 未使用。
- 4096 - 这会强制驱动程序在处于自动提交方式时关闭游标之后不发出 COMMIT。
- 8192 - 这会强制驱动程序在调用存储过程后返回额外的结果集。此结果集是组成该存储过程的输出值的一行结果集。可通过 Powerbuild 应用程序进行访问。
- 32768 - 这会强制驱动程序使 Microsoft Query 应用程序使用 DB2 MVS 同义词。
- 65536 - 这会强制驱动程序在实际上是图形文字的字符文字前人工插入 "G"。在双字节环境中工作时, 总是会提供此补丁。
- 131072 - 这会强制驱动程序在时间戳记列作为唯一索引的一部分时, 将该列反而描述为 CHAR(26) 列。这是 Microsoft 应用程序所需的。
- 262144 - 这会强制驱动程序使用伪目录表 db2cli.procedures 代替 SYSCAT.PROCEDURES 和 SYSCAT.PROCPARMS 表。
- 524288 - 这会强制驱动程序在对 DB2/400 V3.x 系统执行系统表查询时使用 SYSTEM_TABLE_SCHEMA 而不是 TABLE_SCHEMA。这会提高性能。
- 1048576 - 这会强制驱动程序将 SQLPutData() 中的零长度字符串视作 SQL_NULL_DATA。

PATCH2 关键字与 PATCH1 关键字不同。对于 PATCH2, 是使用逗号分隔符来指定多个补丁。例如, 如果指定了补丁 1、4 和 5, 则 PATCH2 将具有值 "1,4,5"。以下是每个关键字值以及它对驱动程序的影响的描述:

- 1 - 这会强制驱动程序将 CALL 语句中的存储过程名转换为大写。
- 2 - 未使用。
- 3 - 它强制驱动程序把模式调用的所有自变量都转换为大写。
- 4 - 这会强制驱动程序对模式调用 (即 SQLColumns()、SQLProcedureColumns() 等等) 返回类似版本 2.1.2 的结果集, 而不是类似版本 5 的结果集。
- 5 - 这会强制驱动程序不对输入 VARCHAR 列的处理进行优化, 其中指向数据的指针和指向长度的指针在内存中是连续的。
- 6 - 这会强制驱动程序返回消息“可滚动游标不受支持”。如果 DB2 客户机为版本 5 且服务器为“DB2 UDB 版本 5”的话, 这是 Visual Basic 程序所需的。
- 7 - 这会强制驱动程序将所有 GRAPHIC 列数据类型映射至 CHAR 列数据类型。这是双字节环境所需要的。
- 8 - 它强制驱动程序忽略模式调用中的目录搜索自变量。
- 9 - 不要提交游标的“提前关闭”
- 10 - 未使用
- 11 - 报告支持目录名, (VB 存储过程)
- 12 - 从模式调用自变量中除去双引号, (Visual Interdev)
- 13 - 不要将 db2cli.ini 中的关键字追加到输出连接字符串

- 14 - 忽略 SQLProcedures() 和 SQLProcedureColumns() 中的模式名
- 15 - 在字符输出中始终使用句点作为小数分隔符
- 16 - 对于每次打开, 强制返回描述信息
- 17 - 不要在描述信息中返回列名
- 18 - 试图用参数标记替换文字
- 19 - 目前 DB2 MVS V4.1 不支持 ODBC 语法, 该语法允许 ON 子句和 Outer 连接子句中出現括号。打开此 PATCH2 将导致 IBM DB2 ODBC 驱动程序在 outer 连接子句中处于 ODBC 转义序列中时去掉括号。仅当对 DB2 MVS 4.1 运行时, 才应使用此 PATCH2。
- 20 - 目前, MVS 上的 DB2 不支持将两个操作数作为参数标记的 BETWEEN 谓词 (表达式为 ? BETWEEN ?)。打开此补丁将导致 “IBM ODBC 驱动程序” 将该谓词重写为 (表达式 >= ? 和表达式 <= ?)。
- 21 - 只将存储过程的所有 OUTPUT 参数设置为 SQL_NULL_DATA
- 22 - 此 PATCH2 会导致 IBM ODBC 驱动程序将 OUTER 连接报告为不受支持。这是针对在使用 outer 连接语句时生成 SELECT DISTINCT col1 或 ORDER BY col1 (其中 col1 的长度大于 254 个字符) 的应用程序而言的, 并会导致 DB2 UDB 返回错误 (因为 DB2 UDB 不支持使用长度超过 254 个字节的列)
- 23 - 不要优化与 cbColDef=0 绑定的参数输入
- 24 - 将 “时间” 值映射为 “字符” 的访问解决方法
- 25 - 小数列的访问解决方法 - 除去 char 表示法中的结尾零
- 26 - 不要将 sqlcode 464 返回至应用程序 - 指示结果集被返回
- 27 - 强制 SQLTables 使用 TABLETYPE 关键字值, 即使应用程序指定了有效值
- 28 - 将实型列描述为双精度型列
- 29 - 小数列的 ADO 解决方法 - 除去值 x 的前导零
其中 $1 > x > -1$ (仅某些 MDAC 版本需要)
- 30 - 禁用 “存储过程” 高速缓存优化
- 31 - 在 SQLStatistics 调用上报告别名统计信息
- 32 - 覆盖 sqlcode -727 原因码 4 处理
- 33 - 在转换为 char 时返回 ISO 版本的时间戳记 (与 ODBC 版本相对照)
- 34 - 将 CHAR FOR BIT DATA 列报告为 CHAR
- 35 - 在请求 SQL_DESC_BASE_TABLE_NAME 时报告无效的 TABLENAME - ADO 只读优化
- 36 - 保留
- 37 - 保留

第8部分 一般信息

一般信息	431	在 Windows 2000 上使用 CLP 时发生错误	
DB2 通用数据库商业智能快速导航	431	SQL1035N	454
DB2 Everywhere 现在更名为 DB2 Everyplace	431	SQL 辅助程序的增强功能	454
需要鼠标	431	在 Linux 上对 DB2 的 Gnome 和 KDE 桌面集成	454
试图从“DB2 运行时客户机”进行绑定导致“绑定文件找不到”错误	432	在 Windows 2000 Terminal Server, Administration Mode 下运行 DB2	454
Search Discovery	432	关于备份和恢复命令的联机帮助	454
HP-UX 11 的内存窗口	432	“仓库管理器”应为“DB2 仓库管理器”	455
发生 dlfm client_conf 故障时的用户操作	433		
偶尔出现“复制守护程序”在 dlfm 停止时不能停止的情况	433		
卸载 DB2 DFS 客户机使能器	433		
Windows NT 上的客户机认证	433		
自动装入程序可能会在创建子进程期间挂起	434		
DATALINK 恢复	434		
在“IBM 通信服务器 Windows NT 版”(CS/NT)中定义用户标识和口令	434		
节点定义	434		
联合系统限制	435		
DataJoiner 的限制	435		
希伯莱语的“信息目录管理器 Windows NT 版”Microsoft SNA Server 和 SNA 多站点更新(两阶段提交)支持	436		
DB2 的 SNA SPM 在引导 Windows 后无法启动	436		
DB2 管理服务器的语言环境设置	436		
快捷方式不起作用	437		
Windows NT 和 Windows 2000 上的 DB2 的服务帐户需求	437		
版本 6 中创建的 Query Patroller 用户的 EXECUTE 特权丢失	437		
Query Patroller 限制	437		
需要提交将在“数据仓库中心”(DWC)中使用的所			
有用户定义程序	438		
用于数据仓库中心命令行导出的新选项	438		
备份服务 API (XBSA)	438		
OS/390 代理程序	439		
安装概述	439		
安装详细信息	440		
设置附加代理程序功能	442		
使用触发器程序 (XTClient) 调度仓库步骤	443		
变换器	444		
访问 DB2 系列之外的数据库	447		
运行“DB2 OS/390 版”实用程序	449		
复制	450		
代理程序日志记录	451		
在 Windows NT 上进行客户机方高速缓存	452		
企业版 UNIX CD-ROM 上的试用产品	452		
DB2 Connect 企业版 UNIX CD-ROM 上的试用产品	453		
删除 Data Links Manager	453		
使用 SMIT 卸载 DLFM 组件时可能会除去附加文件			
集	453		

一般信息

DB2 通用数据库商业智能快速导览

“快速导览”在“DB2 Linux 版或 Linux/390 版”上不可用。

已优化快速导览，以小系统字体运行。在 OS/2 上，您可能必须调整 Web 浏览器的字体大小以便正确地查看快速导览。可参考 Web 浏览器的帮助，获取关于调整字体大小的信息。要正确地查看快速导览（仅针对 SBCS），建议您使用 8 磅 Helv 字体。对于日本和韩国客户，建议您使用 8 磅 Mincho 字体。设置字体首选项时，确保在“首选项”窗口的“字体”页中选择“使用缺省字体，覆盖文档指定的字体”选项。

在某些情况下，快速导览可能会在第二个浏览器窗口后面启动。要校正此问题，应关闭快速导览，然后遵循第10页的『试图启动 Netscape 时的错误消息』中的步骤。

启动快速导览时，可能会看到类似以下内容的 JavaScript 错误：

```
file:/C:/Program Files/SQLLIB/doc/html/db2qt/index4e.htm, line 65:
```

```
Window is not defined.
```

此 JavaScript 错误会阻碍快速导览启动页 index4e.htm 在快速导览启动后自动关闭。可通过关闭其中显示有 index4e.htm 的浏览器窗口来关闭快速导览启动页。

在『新增内容』一节的“数据管理”主题中提到，在版本 7.1 中支持“按需记录归档支持”。事实并非如此。它还提到：

```
日志文件的大小已从 4GB 增长为 32GB。
```

此句应为：

```
总的活动日志空间已从 4GB 增长为 32GB。
```

描述 DB2 Data Links Manager 的一节包含如下内容：

```
而且，它现在支持使用 Veritas XBSA 接口，  
以利用 NetBackup 进行备份与恢复。
```

此句应为：

```
而且，它现在支持 XBSA 接口进行文件归档和恢复。支持 XBSA  
接口的存储管理器包括 Legato NetWorker 和 Veritas NetBackup。
```

DB2 Everywhere 现在更名为 DB2 Everyplace

DB2 Everywhere 的名称已经更改为 DB2 Everyplace。

需要鼠标

对于除 Windows 之外的所有平台，都需要鼠标才能使用这些工具。

试图从“DB2 运行时客户机”进行绑定导致“绑定文件找不到”错误

由于“DB2 运行时客户机”不具有完整的绑定文件集，所以不能从“DB2 运行时客户机”完成 GUI 工具的绑定，该绑定仅可从“DB2 管理客户机”完成。

Search Discovery

Search discovery 仅在广播媒体上受支持。例如，search discovery 将不能通过 ATM 适配器进行工作。但是，此限制不适用于 known discovery。

HP-UX 11 的内存窗口

内存窗口是供大型 HP 64 位机器上的用户使用的，这些用户希望利用超过 1.75 GB 的共享内存来运行 32 位应用程序。如果您正在运行 64 位版本的 DB2，则不需要内存窗口。内存窗口使得每个进程或进程组都有单独的 1 GB 共享内存可用。这允许一个实例拥有它自己的 1 GB 共享内存，以及 0.75 GB 全局共享内存。如果用户希望利用这一点，他们可运行多个实例，每个实例在各自的窗口中运行。以下是使用内存窗口的先决条件：

- DB2 EE 环境

- 补丁程序：扩充软件 12/98 和 PHKL_17795。
- 必须为实例设置 \$DB2INSTANCE 变量。
- 对于想要在内存窗口下运行的每个 DB2 实例，在 /etc/services.window 文件中都必须有相应的一项。例如：

```
db2instance1 50
db2instance2 60
```

注意：在名称和标识之间只能有一个空格。

- 您希望在服务器上运行的以及需要多条语句的任何 DB2 命令都必须使用 TCP/IP 回送方法运行。这是因为当内存窗口处理完第一条语句时 shell 将终止。“DB2 服务”知道如何完成此操作。
- 想要对正运行于内存窗口中的实例运行的任何 DB2 命令都必须以 db2win（位于 sqlllib/bin 中）开头。例如：

```
db2win db2start
db2win db2stop
```

- 在内存窗口外部运行的所有 DB2 命令（内存窗口正在运行时在外部运行的命令除外）都应返回 1042。例如：

```
db2win db2start <== OK
db2 connect to db <==SQL1042
db2stop <==SQL1042
db2win db2stop <== OK
```

- DB2 EEE 环境

- 补丁程序：扩充软件 12/98 和 PHKL_17795。
- 必须为实例设置 \$DB2INSTANCE 变量。
- DB2_ENABLE_MEM_WINDOWS 注册表变量必须设置为 TRUE。
- 对于想要在内存窗口下运行的每个实例的每一逻辑节点，在 /etc/services.window 文件中都必须有相应的一项。每一项的首个字段都应该是与端口号并置的实例名。例如：

```

=== $HOME/sqllib/db2nodes.cfg for db2instance1 ===
5 host1 0
7 host1 1
9 host2 0

=== $HOME/sqllib/db2nodes.cfg for db2instance2 ===
1 host1 0
2 host2 0
3 host2 1

=== /etc/services.window on host1 ===
db2instance10 50
db2instance11 55
db2instance20 60

=== /etc/services.window on host2 ===
db2instance10 30
db2instance20 32
db2instance21 34

```

- 不可让所有的 DB2 命令都以 db2win 开头， db2win 仅在 EE 环境中使用。

发生 dlfm client_conf 故障时的用户操作

在 DLFM 客户机上，如果 dlfm client_conf 由于某种原因失败，则可能是由于 DB2 目录中的 "stale" 项造成的。解决方案是发出下列命令：

```

db2 uncatalog db <dbname>
db2 uncatalog node <node alias>
db2 terminate

```

然后再次尝试 dlfm client_conf。

偶尔出现“复制守护程序”在 dlfm 停止时不能停止的情况

极少会出现用户发出 dlfm 停止命令时或发生异常停机时 dlfm_copyd（复制守护程序）不能停止的情况。如果发生这种情况，则在尝试重新启动 dlfm 之前发出 dlfm 停机命令。

卸载 DB2 DFS 客户机使能器

在卸载“DB2 DFS 客户机使能器”之前，Root 用户应确保无任何 DFS 文件处于使用状态，且在 DFS 文件空间中，没有任何用户的 shell 处于打开状态。作为 root 用户，发出以下命令：

```
stop.dfs dfs_cl
```

检查是否未再安装 /...:

```
mount | grep -i dfs
```

如果未进行此操作，而卸载了“DB2 DFS 客户机使能器”，则机器将需要重新引导。

Windows NT 上的客户机认证

引入一个新的 DB2 注册表变量 DB2DOMAINLIST 来完善 Windows NT 环境中的现有客户机认证机制。在 DB2 Windows NT 版服务器上使用该变量来定义一个或多个 Windows NT 域。只有来自属于此列表中所定义域的用户连接请求才会被接受。

此注册表变量应仅在纯 Windows NT 域环境下使用，而 DB2 服务器和客户机在版本 7（或更高版本）中运行。

有关设置此注册表变量的信息，参考《管理指南》：性能中的『DB2 注册表和环境变量』一节。

自动装入程序可能会在创建子进程期间挂起

AIX 4.3.3 包含对 libc 问题的修正，该问题会导致“自动装入程序”在创建子进程期间挂起。“自动装入程序”是一个多线程程序。其中一个线程脱离另一进程。从一个子进程再创建子进程会导致在子进程中创建父进程内存的映象。

可能会出现这样一种情况，由 libc.a 用以管理多个线程（这些线程从同一进程内的堆分配内存）的锁定已被不包含有子进程的线程挂起。既然不包含有子进程的线程将不存在于子进程中，则始终不会在子进程中释放此锁定，这使得父进程在将来某些时候被挂起。

DATALINK 恢复

在数据库恢复之后进行的任何脱机备份的恢复，不管有没有前滚，都将不会涉及快速协调处理。在这种情况下，其 DATALINK 列处于文件链路控制之下的所有表都将被置于“数据链路协调暂挂”（DRP）状态。

在“IBM 通信服务器 Windows NT 版”（CS/NT）中定义用户标识和口令

如果将 APPC 用作远程 DB2 客户机的通信协议以连接至 DB2 服务器，且将 CS/NT 用作 SNA 产品，则应确保在 CS/NT 配置文件中正确设置了下列关键字。此文件通常可在 x:\ibmcs\private 目录中找到。

节点定义

TG_SECURITY_BEHAVIOR

此参数允许用户确定节点处理 ATTACH 中存在的安全性信息的方式（若未为安全性配置 TP 的话）

IGNORE_IF_NOT_DEFINED

此参数允许用户确定 ATTACH 中是否存在安全性参数，且若未为安全性配置 TP，则忽略这些参数。

如果使用 IGNORE_IF_NOT_DEFINED，则您不必在 CS/NT 中定义用户标识和口令。

VERIFY_EVEN_IF_NOT_DEFINED

此参数允许用户确定 ATTACH 中是否存在安全性参数并验证它们，即使并配置 TP 以确保安全性。这是缺省值。

如果使用 VERIFY_EVEN_IF_NOT_DEFINED，则必须在 CS/NT 中定义用户标识和口令。

要定义 CS/NT 用户标识和口令，执行下列步骤：

1. 开始 → 程序 → IBM 通信服务器 → SNA 节点配置。 “欢迎使用通信服务器配置”窗口打开。

2. 选择想要修改的配置文件。单击**下一步**。“选择配置方案”窗口打开。
3. 突出显示 **CPI-C**、**APPC** 或 **5250 模拟**。单击**完成**。“通信服务器 SNA 节点”窗口打开。
4. 单击 **CPI-C** 和 **APPC** 旁的 **[+]**。
5. 单击 **“LU6.2 安全性”** 旁的 **[+]**。
6. 用鼠标右键单击 **“用户口令”** 并选择**创建**。“定义用户标识和口令”窗口打开。
7. 填入用户标识和口令。单击**确定**。单击**完成**以接受更改。

联合系统限制

以下是适用于联合系统的一些限制:

- 在涉及别名的查询中, Oracle 数据类型 **NCHAR**、**NVARCHAR2**、**NCLOB** 和 **BFILE** 不受支持。
- **“创建服务器选项”**、**“改变服务器选项”**、和 **“删除服务器选项”** 命令从 **“控制中心”** 不受支持。要使用这些命令中的任何命令, 必须使用命令行处理器 (CLP)。
- 对于涉及别名的查询, **DB2 UDB** 不总是遵守 **DFT_SQLMATHWARN** 数据库配置选项。相反地, **DB2 UDB** 会直接从远程数据资源返回算术错误或警告, 而不管 **DFT_SQLMATHWARN** 设置如何。
- **CREATE SERVER OPTION** 语句不允许 **COLSEQ** 服务器选项对带有不区分大小写的比较顺序的数据源设置为 **'I'**。
- 指定了无效的选项时, **ALTER NICKNAME** 语句会返回 **SQL0901N**。
- 对于 Oracle、Microsoft SQL Server 和 Sybase 数据源, 数字数据类型不能映射为 **DB2** 的 **BIGINT** 数据库类型。缺省情况下, Oracle 的数字 (p,s) 数据类型 (其中 $10 \leq p \leq 18$ 、 $s = 0$) 映射为 **DB2** 的 **DECIMAL** 数据类型。

DataJoiner 的限制

在联合环境中发出的分布式请求只限于只读操作。

希伯莱语的“信息目录管理器 Windows NT 版”

“信息目录管理器”组件具有希伯莱语版, 它是在“**DB2 仓库管理器 Windows NT 版**”CD 上提供的。

希伯莱语翻译版本是在一个称为 **IL_ICM.ZIP**, 且位于“**DB2 仓库管理器 Windows NT 版**”CD 上 **DB2\IL** 目录中的 **zip** 文件中提供的。

要安装“信息目录管理器”的希伯莱语翻译版本, 首先在支持希伯莱语的 **Windows NT** 版本上安装“**DB2 仓库管理器 Windows NT 版**”的英语版和所有先决条件。

在安装了“**DB2 仓库管理器 Windows NT 版**”之后, 从 **DB2\IL** 将 **IL_ICM.ZIP** 文件解压到在其中安装了“**DB2 仓库管理器 Windows NT 版**”的目录中。确保为解压程序提供了正确的选项, 以在 **zip** 文件中创建目录结构。

在解压文件之后, 全局环境变量 **LC_ALL** 必须从 **En_US** 更改为 **Iw_IL**。要更改设置:

1. 打开 **Windows NT “控制面板”** 并双击**系统**图标。
2. 在**系统特性**窗口中, 单击**环境**选项卡, 然后在**系统变量**部分中定位 **LC_ALL**。

3. 单击该变量以在值编辑框中显示该值。将值从 **En_US** 更改为 **lv_IL**。
4. 单击**设置**按钮。
5. 关闭**系统特性窗口**和**控制面板**。

现在应已安装好了“信息目录管理器”的希伯来语版本。

Microsoft SNA Server 和 SNA 多站点更新（两阶段提交）支持

如果 Microsoft SNA Server 是正在使用的 SNA 产品，则主机和 AS/400 应用程序不能使用 SNA 两阶段提交来访问 DB2 UDB 服务器。指示这是受支持的所有 DB2 UDB 出版物都是不正确的。需要 IBM 通信服务器 Windows NT 版的版本 5.02 或更高版本。

注：使用 DB2 UDB Windows 版访问主机和 AS/400 数据库服务器的应用程序可使用 SNA 两阶段提交（通过使用“Microsoft SNA Server 版本 4 服务包 3”或更高版本）。

DB2 的 SNA SPM 在引导 Windows 后无法启动

如果您使用的是 Microsoft SNA Server 版本 4 SP3 或更高版本，请验证 DB2 的 SNA SPM 在重新引导后启动正常。检查 `\sqlllib\<instance name>\db2diag.log` 文件中与以下内容类似的项：

```
2000-04-20-13.18.19.958000 Instance:DB2 Node:000
PID:291(db2syscs.exe) TID:316 Appid:none
common_communication sqlccspmconnmgr_APPC_init Probe:19
SPM0453C Sync point manager did not start because Microsoft SNA Server has not
been started.
```

```
2000-04-20-13.18.23.033000 Instance:DB2 Node:000
PID:291(db2syscs.exe) TID:302 Appid:none
common_communication sqlccsna_start_listen Probe:14
DIA3001E "SNA SPM" protocol support was not successfully started.
```

```
2000-04-20-13.18.23.603000 Instance:DB2 Node:000
PID:291(db2syscs.exe) TID:316 Appid:none
common_communication sqlccspmconnmgr_listener Probe:6
DIA3103E Error encountered in APPC protocol support. APPC verb "APPC(DISPLAY 1
BYTE)". Primary rc was "F004". Secondary rc was "00000000".
```

如果 `db2diag.log` 中有这样的项，且时间戳记与最新的引导时间相匹配，必须：

1. 调用 `db2stop`。
2. 启动 `SnaServer` 服务（如果尚未启动的话）。
3. 调用 `db2start`。

再次检查 `db2diag.log` 文件，验证该项已不再附加在该文件中。

DB2 管理服务器的语言环境设置

请确保“DB2 管理服务器”实例的语言环境与 DB2 实例的语言环境兼容。否则，DB2 实例不能与“DB2 管理服务器”通信。

如果未在“DB2 管理服务器”的用户简要表中设置 `LANG` 环境变量，则将以缺省系统语言环境来启动“DB2 管理服务器”。如果未定义缺省系统语言环境，则将以代码页 819 来启动“DB2 管理服务器”。如果 DB2 实例使用其中一个 DBCS 语言环境，且以代码

页 819 启动“DB2 管理服务器”，则该实例将不能够与“DB2 管理服务器”通信。
“DB2 管理服务器”的语言环境必须与 DB2 实例的语言环境兼容。例如，在简体中文 Linux 系统上，应在“DB2 管理服务器”的用户简要表中设置“LANG=zh_CN”。

快捷方式不起作用

在某些语言版本中，对于基于 UNIX 的系统和 OS/2 上的“控制中心”，某些键盘快捷方式不起作用。请使用鼠标来选择选项。

Windows NT 和 Windows 2000 上的 DB2 的服务帐户需求

在 DB2 Windows NT 版或 Windows 2000 版的安装期间，安装程序会创建几个 Windows 服务，并为每个服务指定一个服务帐户。要正常运行 DB2，安装程序应将下列用户权利授予与该 DB2 服务相关联的服务帐户：

- 充当操作系统的一部分
- 创建标记对象
- 增加份额
- 作为服务登录
- 替换进程级别令牌。

如果想要对 DB2 服务使用另一服务帐户，必须将这些用户权利授予该服务帐户。

除这些用户权利之外，服务帐户还必须对安装有 DB2 产品的目录具有写访问权。

“DB2 管理服务器”服务（DB2DAS00 服务）的服务帐户还必须具有启动和停止其他 DB2 服务的权限（即，服务帐户必须属于“电源用户”组），并对由其管理的所有 DB2 实例具有 DB2 SYSADM 权限。

版本 6 中创建的 Query Patroller 用户的 EXECUTE 特权丢失

因为已在 Query Patroller 版本 7 中添加了某些新的存储过程（IWM.DQPGROUP、IWM.DQPVALUR、IWM.DQPCALCT 和 IWM.DQPINJOB），所以 Query Patroller 版本 6 中创建的现存用户不会挂起对那些包的 EXECUTE 特权。用来自动校正此问题的应用程序已添加至“修订包 1”。

在尝试使用 DQP Query Admin 来修改 DQP 用户信息时，请不要试图从用户列表除去现存用户。

Query Patroller 限制

由于 JVM（Java 虚拟机）平台限制，Query Enabler 在 HP-UX 和 NUMA-Q 上不受支持。此外，Query Patroller Tracker 在 NUMA-Q 上也不受支持。若需要所有 Query Patroller 客户机工具，则建议使用另一平台（例如，Windows NT）来对 HP-UX 或 NUMA-Q 服务器运行这些工具。

需要提交将在“数据仓库中心”(DWC)中使用的所有用户定义程序

如果要在“数据仓库中心”(DWC)中将由“DB2 存储过程构建器”构建的存储过程作用用户定义程序, 必须将以下语句插入到存储过程中的 `con.close()`; 语句的前面:

```
con.commit();
```

如果未插入此语句, 则存储过程所作的更改将会在从 DWC 运行存储过程时发生回滚。

对于 DWC 中的所有用户定义程序, 需要显式提交所有包含的 DB2 功能, 以使这些更改在数据库中生效; 即, 您必须将 `COMMIT` 语句添加至用户定义程序。

用于数据仓库中心命令行导出的新选项

标记文件的命令行导出有一个新选项 `/B`。此选项在“数据仓库中心”界面上不可用。

`iwh2exp2` 命令的新语法为:

```
iwh2exp2 filename.INP dbname userid password [PREFIX=table_schema] [/S] [/R] [/B]
```

其中

- filename.INP 是 INP 文件的全路径名
- dbname 是“数据仓库中心”控制数据库名
- userid 是用来登录数据库的用户标识
- password 是用来登录数据库的口令
- 可选参数有:
 - PREFIX=table_schema: 控制数据库表的表模式 (缺省值为 IWH)
 - /S: 用选定步骤导出调度
 - /R: 不要用选定步骤导出数据仓库源
 - /B: 不要用选定步骤导出生效步骤

注: 如果指定了 `/R` 或 `/B`, 则数据仓库源或生效步骤在导入生成的标记文件时必须已经存在, 否则会返回错误。

备份服务 API (XBSA)

英国的“开放组”已经将“备份服务 API”(XBSA) 定义为需要数据存储管理的应用程序或设施之间的开放式应用编程接口, 以便进行备份或归档。这在“Open Group Technical Standard System Management: Backup Services API (XBSA)”, 文档号 C425 (ISBN: 1-85912-056-3) 中作了说明。

为支持备份服务, 创建了两个新的 DB2 注册表变量, 且这两个注册表变量目前在 AIX、HP、Solaris 和 Windows NT 上是受支持的:

DB2_VENDOR_INI

指向包含所有特定于供应商的环境设置的文件。在数据库管理器启动时会使用该值。

DB2_XBSA_LIBRARY

指向供应商提供的 XBSA 库。在 AIX 上, 如果共享对象未命名为 `shr.o`, 则该设置必须包括它。HP、Solaris 和 Windows NT 不需要共享内存名。例如, 要使用 Legato's NetWorker Business Suite Module for DB2, 注册表变量必须按如下设置:

```
db2set DB2_XBSA_LIBRARY="/usr/lib/libxdb2.a(bsashr10.o)"
```

可在 `BACKUP DATABASE` 或 `ESTORE DATABASE` 命令中调用 XBSA 接口。例如:

```
db2 backup db sample use XBSA
db2 restore db sample use XBSA
```

OS/390 代理程序

本文档中有哪些内容？安装 OS/390 及其功能部件

本文档中提供了关于如何安装 OS/390 代理程序的指导以及有关其功能部件的信息。参见『安装概述』以便快速复习安装过程，参见『安装详细信息』以便获取更详细的指导。有关代理程序的功能部件的信息，参见『设置附加代理程序功能』、『变换器』以及『访问 DB2 系列之外的数据库』。

概述

“DB2 仓库中心”包括 OS/390 代理程序。可使用该代理程序在“DB2 通用数据库 OS/390 版”与其他数据库（包括其他平台上的 DB2 数据库和非 DB2 数据库）之间进行通信。该代理程序还可与使用 ODBC 连接的受支持的数据源进行通信。该代理程序在“OS/390 UNIX 系统服务”上运行。它要求 OS/390 V2R6 或更新版本，且它与“DB2 OS/390 版”的版本 5、版本 6 和版本 7 兼容。

OS/390 代理程序支持下列任务：

- 将数据从源 DB2 数据库复制至目标 DB2 数据库。
- 对表或文件的内容进行采样
- 运行用户定义程序
- 通过 IBM DataJoiner 来访问非 DB2 数据库
- 通过 Cross Access Classic Connect 来访问 VSAM 或 IMS 数据
- 运行“DB2 通用数据库 OS/390 版”实用程序
- 对 IBM Data Propagator 运行 apply 作业

安装概述

下列步骤概述了安装过程。『安装详细信息』一节提供了关于这些步骤的更多详细信息。

1. 通过“DB2 通用数据库 OS/390 版”磁带来安装 OS/390 代理程序。
2. 更新简要表文件中的环境变量。
3. 设置连接：
 - 在内核与代理程序守护程序之间。
 - 在代理程序与它将访问的数据库之间。
4. 在本地绑定 CLI 以及将其绑定至任何远程数据库。
5. 设置 ODBC 初始化文件。
6. 设置权限以便用户：
 - 可以运行代理程序守护程序。
 - 对计划 DSNAOCLI 具有执行权限。
 - 对记录和 ODBC 跟踪目录具有读写权限（如果需要的话）。
7. 启动代理程序守护程序。

安装详细信息

安装 OS/390 代理程序

OS/390 代理程序包括在“DB2 通用数据库 OS/390 版的版本 7”磁带中。查看随该磁带提供的“程序目录”，以获取有关安装 OS/390 代理程序的详情。

在安装 OS/390 代理程序之前，必须对 DB2 子系统应用 APAR PQ36585 或 PQ36586。

更新简要表文件中的环境变量

这些变量将代理程序指向各个 DB2 库、输出目录等等。

以下示例显示了样本 .profile 文件的内容。.profile 文件定义了环境变量，且该文件必须在启动代理程序守护程序的用户的主目录中：

```
export VWS_LOGGING=/usr/lpp/DWC/logs
export VWP_LOG=/usr/lpp/DWC/vwp.log
export VWS_TEMPLATES=/usr/lpp/DWC/
export DSNAOINI=/usr/lpp/DWC/dsnaoini
export LIBPATH=/usr/lpp/DWC/:$LIBPATH
export PATH=/usr/lpp/DWC/:$PATH
export STEPLIB=DSN710.SDSNEXIT:DSN710.SDSNLOAD
```

设置连接

要设置内核和守护程序的连接，将下列行添加至 /etc/services 文件或 TCPIP.ETC.SERVICES 文件：

```
wkernal 11000/tcp vwd 11001/tcpvlogger 11002/tcp
```

要设置 OS/390 代理程序和数据库之间的连接，将所有远程数据库添加至 OS/390 通信数据库 (CDB)。以下是一些样本 CDB 插入：

```
INSERT INTO SYSIBM.LOCATIONS
(LOCATION, LINKNAME, PORT)
VALUES
('NTDB', 'VWNT704', '60002');

INSERT INTO SYSIBM.IPNAMES
(LINKNAME, SECURITY_OUT, USERNAMES, IPADDR)
VALUES
('VWNT704', 'P', 'O', 'VWNT704.STL.IBM.COM');

INSERT INTO SYSIBM.USERNAMES
(TYPE, AUTHID, LINKNAME, NEWAUTHID, PASSWORD)
VALUES
('O', 'MVSUID', 'VWNT704', 'NTUID', 'NTPW');
```

有关设置连接以及更新通信数据库的详情，参见 *DB2 UDB for OS/390 Installation Guide* (GC26-9008-00) 中的『连接分布式数据库系统』一章。

绑定 CLI

因为 OS/390 代理程序使用 CLI 与 DB2 进行通信，所以必须将 CLI 计划绑定至代理程序将访问的所有远程数据库。以下是本地“DB2 OS/390 版”数据库的一些样本绑定软件包语句：

```
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLICS) ISO(CS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLINC) ISO(NC)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIRR) ISO(RR)
```

```
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIRS) ISO(RS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIUR) ISO(UR)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIMS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIC1)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIC2)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIF4)
```

以下是在 Windows NT 上运行的 DB2 数据库的一些样本绑定软件包语句:

```
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLICS) ISO(CS)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLINC) ISO(NC)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIRR) ISO(RR)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIRS) ISO(RS)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIUR) ISO(UR)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIC1)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIC2)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIQR)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIF4)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIV1)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIV2)
```

以下是用来将这些 CLI 软件包一起绑定到计划中的样本绑定语句:

```
BIND PLAN(DWC6CLI) PKLIST(*.DWC6CLI.* )
```

设置 ODBC 初始化文件

样本 ODBC 初始化文件 `inisamp` 包括在 `usr/lpp/DWC/` 目录中。可编辑此文件以与系统配合使用, 或创建您自己的文件。要确保文件正常工作, 验证它是否被正确配置为:

- `DSNAOINI` 环境变量必须指向初始化文件。
- 文件名应该使用命名约定 `dsnaoini.location_name`。
- 该文件必须包括 `CONNECTTYPE=2` 和 `MVSATTACHTYPE=CAF` 参数。

有关绑定 CLI 和 `DSNAOINI` 文件的详情, 参见 *DB2 UDB for OS/390 ODBC Guide and Reference* (SC26-9005)。

设置权限

OS/390 代理程序是一个守护进程。可使用常规 UNIX 安全性或 OS/390 UNIX 安全性运行该代理程序守护程序。

因为该代理程序需要守护程序权限, 所以必须对“PACF 程序控制”定义下列代理程序可执行文件:

- `libt1s4d.dll`
- `iwhcomnt.dll`
- `vwd`

要为 RACF 程序控制定义可执行程序, 切换至“数据仓库中心”可执行文件所在的目录, 并运行下列命令:

```
extattr +p libt1s4d.dll
extattr +p iwhcomnt.dll
extattr +p vwd
```

要使用带有 +p 参数的 extattr 命令，必须至少对 BPX.FILEATTR.PROGCTL FACILITY 类具有读访问权。以下示例显示了用来对用户标识 SMORG 授予此许可权的 RACF 命令：

```
RDEFINE FACILITY BPX.FILEATTR.PROGCTL UACC(NONE)
PERMIT BPX.FILEATTR.PROGCTL CLASS(FACILITY) ID(SMORG) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

有关权限的详情，参见 *OS/390 UNIX System Services Planning* (SC28-1890)。

启动代理程序守护程序

在配置完系统之后，启动代理程序守护程序：

1. 通过 OS/390 主机名和 USS 端口 Telnet 至 OS/390 上的“UNIX 系统服务”。
2. 启动代理程序守护程序：

- 要在前台启动守护程序，应在命令行上输入 vwd。
- 要在后台启动守护程序，输入：

```
vwd>/usr/lpp/DWC/logs/vwd.log 2>&1 &
```

要验证 OS/390 代理程序守护程序是否在运行，在 UNIX shell 命令行上输入以下命令：

```
ps -e | grep vwd
```

或者，在 OS/390 控制台上输入 D OMVS,a=all ，并搜索字符串 vwd

设置附加代理程序功能

“DB2 仓库管理器”软件包包括下列用户定义程序：

- vwprftp: 运行 FTP 命令文件。
- vwpmvs: 提交 JCL 作业流。
- vwprcpy: 使用 FTP 复制文件。
- XTClient: 客户机触发器程序。
- etidlmvs: 来自 ETI (Evolutionary Technologies International) 的实用程序；删除 MVS 上的文件。
- etircmvs: 来自 ETI 的实用程序；在 MVS 主机上运行 FTP。
- etiexmvs: 来自 ETI 的实用程序；在 MVS 上运行 JCL。

另外，可以在“数据仓库中心”中创建用户定义程序和存储过程。OS/390 代理程序支持运行于“UNIX 系统服务”之下的任何可执行程序。

为一个或多个步骤指定了用户定义程序。在运行用户定义程序时，执行了下列操作：

- 代理程序启动。
- 代理程序运行用户定义程序。
- 用户定义程序对代理程序返回返回码和反馈文件。
- 代理程序将结果返回给内核。

要在 OS/390 上运行 ETI 程序，必须先将“修订包 2”应用于“DB2 通用数据库”版本 7.1。

使用 VWP_LOG 环境变量来定义用户定义程序可以将输出写至的目录。

如果使用用户定义程序以利用 FTP 提交作业，则必须先创建 JCL 和想要提交的数据。JCL 中的作业名必须是 USERIDx，其中 x 是 1 字符的字母或数字（示例：MYUSERA）。包含在 JCL 中的 MSGCLASS 和 SYSOUT 文件的输出类必须指定 JES 挂起的输出类。

限制：提交的作业的最大 LRECL 为 254 个字符。JES 只扫描 JCL 的头 72 个字符。

更改“数据仓库中心”模板以便支持 FTP

“数据仓库中心”安装了 JCL 模板，以便使用 FTP 来传输文件。如果计划让 OS/390 代理程序使用 FTP 命令 GET 或 PUT 来将文件从 OS/390 主机传输至另一远程主机，则需要对 OS/390 系统更改 JCL 模板中的帐户信息：

1. 使用有权复制和更新 /usr/lpp/DWC 目录中的文件的标识来进行登录。
2. 查找 ftp.jcl，并使用新文件名 systemname.ftp.jcl 来复制该文件，其中 *systemname* 是 OS/390 系统的名称。
3. 对计划运行转换程序 vwpmvs 或 ETI extract 的每个 OS/390 系统创建此文件的副本。例如，如果想要在 STLMVS1 上运行这两个程序中的任何一个，创建称为 STLMVS1.ftp.jcl 的文件副本。
4. 使用文本编辑器来定制 JCL 以满足站点的需求。更改帐户信息以与 MVS 系统的标准帐户信息相匹配。不要更改包含在括号内的任何参数，例如 [USERID] 和 [FTPFILE]。（这些括号分别是十六进制字符 x'AD' 和 x'BD'。如果在 SPF 选项 0 中未将 TSO 终端类型设置为 3278A，则这些值可能会显示为特殊字符而不是括号。如果不更改 x'AD' 或 x'BD' 或这些字符之间的任何数据，就不会存在此问题）。
5. 将环境变量 VWS_TEMPLATES 更新为指向复制的模板文件的目录。

“数据仓库中心”包括此样本 JCL 模板：

```
//[USERID]A JOB , 'PUT/GET',  
// CLASS=A,  
// USER=&SYSUID,  
// NOTIFY=&SYSUID,  
// TIME=(,30),  
// MSGCLASS=H  
//STEP1 EXEC PGM=FTP,PARM='( EXIT'  
//INPUT DD DSN=[FTPFILE],DISP=SHR  
//OUTPUT DD SYSOUT=*  
//SYSPRINT DD SYSOUT=*
```

对表或文件的内容进行采样

通过使用 OS/390 代理程序，可对 DB2 表和平面文件（例如“UNIX 系统服务”文件和 OS/390 本机平面文件）的内容进行采样。还可使用 OS/390 代理程序通过 Classic Connect 对 IMS 或 VSAM 文件的内容进行采样。对于平面文件，代理程序会查看文件定义特性中的参数来确定文件格式。

使用触发器程序 (XTClient) 调度仓库步骤

使用触发器程序来从 OS/390 平台调度仓库步骤。您或 OS/390 作业调度程序可提交一个作业，它会在“数据仓库中心”中触发一个步骤。如果该步骤成功执行，则 JCL 中的触发步骤会返回返回码 0。

必须已经在“OS/390 UNIX 系统服务”上安装了“Java 开发工具箱” (JDK) 1.1.8 或更新版本，才能使用该触发器程序。要启动触发器，首先在运行仓库服务器的机器上启

动 XTServer。此过程在数据仓库中心管理指南的第 5 章的主题『从“数据仓库中心”外部启动步骤』中作了描述。启动 XTServer 之后，在 OS/390 上启动 XTClient。以下示例显示了启动触发器的样本 JCL。

```
//DBA1A JOB 1,'XTCLIENT',CLASS=A,MSGCLASS=H,
//          MSGLEVEL=(1,1),REGION=4M,NOTIFY=&SYSUID
//*****
//* submit iwhetrig
//*****
//BRADS EXEC PGM=BPXBATCH,
//          PARM=('sh cd /usr/lpp/DWC/; java XTClient 9.317.171.133 1100x
//          9 drummond pw bvmvs2nt 1 1 100')
//STDOUT DD PATH='/tmp/xtclient.stdout',
//          PATHOPTS=(OWRONLY,OCREAT),
//          PATHMODE=SIRWXU
//STDERR DD PATH='/tmp/xtclient.stderr',
//          PATHOPTS=(OWRONLY,OCREAT),
//          PATHMODE=SIRWXU
//
```

注：上述样本 JCL 代码显示了如何使参数换行。为此，输入参数直至第 71 列，在第 72 列中输入 'X'，并在下一行的第 16 列中继续输入参数。

参数的第一部分是一个切换至安装有 OS/390 代理程序的目录的语句 (cd /usr/lpp/DWC/;)。参数的第二部分将启动 XTClient，并传送下列 8 个参数：

- “数据仓库中心”服务器主机名或 IP 地址
- “数据仓库中心”服务器端口（通常为 11009）
- “数据仓库中心”用户标识
- “数据仓库中心”口令
- 要运行的步骤的名称
- “数据仓库中心”服务器命令，其中：
 - 1 = 填充该步骤
 - 2 = 将步骤提升至测试方式
 - 3 = 将步骤提升至生产方式
 - 4 = 将步骤降级至测试方式
 - 5 = 将步骤降级至开发方式
- 用来表示是否等待步骤完成的选项，其中 1=是，0=否
- 最大行数（使用 0 或空白来取装所有行）

变换器

介绍

以下 12 个变换器是 Java 存储过程，它们提供了一些基本的数据转换。为了运行这些变换器，必须首先在 DB2 子系统上设置 Java 存储过程。有关这些变换器的附加信息，可以在 *IBM DB2 通用数据库数据仓库中心管理指南* 版本 7 (SB84-0226-00) 中获得。

```
IWH.CLEAN
IWH.PERIODTABLE
IWH.KEYTABLE
IWH.CHISQUARE
IWH.CORRELATION
IWH.STATISTICS
IWH.INVERTDATA
```

IWH.PIVOTDATA
IWH.REGRESSION
IWH.ANOVA
IWH.SUBTOTAL
IWH.MOVINGAVERAGE

设置 Java 存储过程

这些指导是关于如何设置 Java 存储过程的完整指导的简要版本，可以在 *Application Programming Guide and Reference for Java™* (SC26-9018) 中找到这些指导。

1. 对 DB2 子系统应用 PTF UQ46170 和 UQ46114。
2. 在 OS/390 系统上安装 Visual Age for Java 2.0 或更高版本。
3. 在 DB2 上安装 JDBC，并在 DB2 子系统中绑定 JDBC 软件包。
4. 为 DB2 子系统设置 RRS 和 DB2 WLM 存储过程。
5. 为 DB2 设置 Java 存储过程。这包括为 Java 存储过程地址空间创建 Java WLM 启动过程。
6. 在 WLM 中，必须将 Java WLM 启动过程与 WLM 环境名相关联。使用标题为“创建应用程序环境”的 WLM 应用程序环境面板来将环境名与 JCL 过程相关联。
7. 为 CREATE 或 ALTER PROCEDURE 上的 WLM_ENVIRONMENT 选项指定 WLM 应用程序环境名，以便使存储过程或用户定义函数与应用程序环境相关联。
8. 确保 DB2 已启动任务的所有者对 Java WLM 启动过程中的库具有访问权。

设置仓库变换器的步骤

这些指导是完整指导的简要版本，可以在 *IBM DB2 通用数据库数据仓库中心管理指南* 版本 7 (SB84-0226-00) 中找到这些指导。

1. 对“DB2 通用数据库 NT 版的版本 7”应用“修订包 3”，或将“仓库控制数据库”更新为 TRANSREGISTERED = 1 和 TRANSFENCED=1。
要更新“仓库控制数据库”，在“DB2 通用数据库”命令行处理器上输入以下 SQL:

```
CONNECT TO your_vw_control_database
UPDATE IWH.INFORESOURCE SET TRANSREGISTERED = '1' WHERE SUBDBTYPE = 'DB2 MVS'
UPDATE IWH.INFORESOURCE SET TRANSFENCED = '1' WHERE SUBDBTYPE = 'DB2 MVS'
```
2. 对 DB2 定义变换器
 - 如果有“DB2 OS/390 版的版本 7”，则在 /usr/lpp/DWC/createXfSQLV7 中使用 SQL 语句。
 - 如果有“DB2 OS/390 版的版本 6”，则在 /usr/lpp/DWC/createXfSQL 中使用 SQL 语句。
 - 如果有“DB2 OS/390 版的版本 5”，则在 /usr/lpp/DWC/createXfSQL 中使用注释了的 SQL 语句。注释掉所有 CREATE PROCEDURE 语句。然后取消注释，并使用 INSERT INTO SYSIBM.SYSPROCEDURES 语句来为“DB2 OS/390 版的版本 5”定义变换器。

在设置 Java 存储过程时，使用 WLM 来将 Java WLM 启动过程与 WLM 环境名相关联。环境名是在 CREATE PROCEDURE 语句的 WLM_ENVIRONMENT 选项中指定的。DSNWLMJ 是以上描述的变换器定义中所包括的 WLM 环境名。可以添加 WLM 关联名 DSNWLMJ，或者将每个变换器定义的 WLM_ENVIRONMENT 选项更改为已经与启动过程相关联的名称。

3. 在 IWH710.SIWHLOAD 中设置从“UNIX 系统服务”到变换器装入模块的链接。
 - Telnet 至 OS/390 主机系统上的“UNIX 系统服务”。
 - 切换至安装有 OS/390 代理程序的目录。缺省安装目录为 /usr/lpp/DWC。
 - 如果您正在使用 DB2 V7，则跳至第 4 步。如果您正在使用 DB2 V5 或 V6，则编辑已安装的目录中的 trlinks 数据。通过在第 1 列中放置井号 (#) 来注释此行：


```
ln -e IWHXFV7 xf.j11;
```

通过除去第 1 列中的井号 (#) 来取消注释此行。

```
#ln -e IWHXF xf.j11;
```

保存所作的更改。
 - 输入 trlinks，并按 Enter 键。这将在目录中创建一个 xf.j11 链接，它将引导代理程序装入 IWHXF 或 IWHXFV7 模块。
4. APF 授权 IWH710.SIWHPDSE，然后将它添加到 DB2 Java 存储过程启动过程的 STEPLIB 并置中。
5. 将 xf.j11 链接所在的目录（缺省值为：/usr/lpp/DWC）添加到 WLM 环境数据集中的 CLASSPATH 和 LIBPATH 环境变量中。
 - 如果您不能确定 WLM 环境数据集的位置，则查看 DB2 Java 存储过程启动过程。WLM 环境数据集就是 JAVAENV DD 卡所指向的数据集。
6. 启动存储过程，然后创建并运行仓库步骤。

对 Java 存储过程的限制

只有“DB2 OS/390 版的版本 7”才支持存储过程的签名中的 Java 对象。因此，在“DB2 OS/390 版”的版本 5 或版本 6 中，变换器不支持参数中的空值。在这些版本中，如果您传送一个空参数，则它将相当于为零。版本 5 和版本 6 变换器将零参数视作空字符串。

仅在“DB2 OS/390 版的版本 7”中，DB2 才支持存储过程中的 COMMIT SQL 语句。INVERTDATA 存储过程删除存储过程中的表再重新创建该表；因此，它需要提交语句。因此，IWH.INVERTDATA 在“DB2 OS/390 版”的版本 5 或版本 6 中是不受支持的。

“DB2 OS/390 版”不支持 Java 用户定义函数，因此，IWH.FORMATDATE 在 390 平台上是不受支持的。

Java 存储过程的样本启动过程

（在“DB2 for OS/390 Application Programming Guide and Reference for Java”中作了描述）：

```
//DSNWLMJ PROC DB2SSN=DSN,NUMTCB=5,APPLENV=DSNWLMJ <-- WLM ENVIRONMENT value in CREATE PROC
//*****
//*      THIS PROC IS USED TO START THE WLM-ESTABLISHED SPAS      *
//*      ADDRESS SPACE FOR THE DSNWLMJ APPLICATION ENVIRONMENT    *
//*      V WLM,APPLENV=DSNWLMJ,RESUME                             *
//*****
//DSNWLMJ EXEC PGM=DSNX9WLM,TIME=1440,REGION=0M,
//          PARM='&DB2SSN, &NUMTCB, &APPLENV'
//STEPLIB DD DSN=DSN.TESTLIB,DISP=SHR
//          DD DSN=IWH710.SIWHPDSE,DISP=SHR <-- This has the transformers in it
//          DD DSN=DSN.HPJSP.PDSE.JDBC,DISP=SHR <-- HPJ DLLs from HPJ setup
```

```

//      DD DSN=SYS1.PP.PDSELINK,DISP=SHR          <-- HPJ runtime libraries
//      DD DSN=DSN710.SDSNEXIT,DISP=SHR
//      DD DSN=DSN710.SDSNLOAD,DISP=SHR
//      DD DSN=SYS1.SCEERUN,DISP=SHR
//      DD DSN=DSN.PDSE,DISP=SHR                <-- HPJ setup info
//JAVAENV DD DSN=DSN.WLMENVJ.JSPENV,DISP=SHR    <-- Environment variables, see below
//CEEDUMP DD SYSOUT=A
//DSSPRINT DD SYSOUT=A
//JSPDEBUG DD SYSOUT=A
//SYSABEND DD SYSOUT=A
//SYSPRINT DD SYSOUT=A

```

样本环境变量数据集

(在“DB2 for OS/390 Application Programming Guide and Reference for Java”中作了描述):

```

ENVAR("TZ=PST07",
"DB2SQLJPROPERTIES=/usr/lpp/db2/jdbc/db2710/classes/db2sqljdbc.properties",
"LIBPATH=/usr/lpp/DWC",
"VMSPPATH=/usr/lpp/DWC",
"CLASSPATH=/usr/lpp/db2/jdbc/db2710/classes:/usr/lpp/DWC:/usr/lpp/hpj/lib"),
MSGFILE(JSPDEBUG)

```

变换器的国家语言支持

OS/390 代理程序生成的大多数消息都被发送到 NT 平台进行解释，因此，在大多数情况下，消息语言取决于“UDB NT 版”是如何安装的。但变换器例外。OS/390 代理程序为变换器提供了下列消息文件:

文件名:	语言:
Xf.properties_Fi_FI	芬兰的芬兰语
Xf.properties_No_NO	挪威的挪威语
Xf.properties_Ru_RU	俄罗斯的俄语
Xf.properties_Zh_CN	中国(中华人民共和国)的汉语
Xf.properties_Zh_TW	台湾地区的汉语
Xf.properties_Da_DK	丹麦的丹麦语
Xf.properties_De_DE	德国的德语
Xf.properties_En_US	美国的英语
Xf.properties_Es_ES	西班牙的西班牙语
Xf.properties_Fr_FR	法国的法语
Xf.properties_It_IT	意大利的意大利语
Xf.properties_Ja_JP	日本的日语
Xf.properties_Ko_KR	韩国的韩语
Xf.properties_Pt_BR	巴西的葡萄牙语
Xf.properties_Sv_SE	瑞典的瑞典语

如果您的变换器消息应采用英语之外的语言，则选择以上其中一个文件，并将其内容复制到 Xf.properties 中。

访问 DB2 系列之外的数据库

为了访问非“DB2 通用数据库”系统，OS/390 代理程序使用了 DataJoiner。DataJoiner 允许代理程序对它使用普通的 DRDA 流，就好像它是 UDB 数据库一样。如果 ODBC 请求指向非 DB 系列数据库源，则 DataJoiner 会调用附加层代码来访问外部数据库。

DataJoiner 可以访问 Oracle、Sybase、Informix、Microsoft SQL Server、Teradata，以及其 ODBC 驱动程序在 Windows NT、AIX 或 Sun 公司的“Solaris 操作环境”上运行的任何其他数据库。OS/390 代理程序可将 DataJoiner 作为源访问，但不能作为目标访问。DataJoiner 不支持两阶段提交。尽管在版本 2.1 和 2.1.1 中 DataJoiner 支持 TCP/IP

作为应用程序请求器，但是它没有应用程序服务器。因为 OS/390 代理程序需要应用程序服务器才能使用 TCP/IP，所以必须转而使用 SNA 连接来从 OS/390 访问 DataJoiner。

在 OS/390 上访问 IMS 和 VSAM

Classic Connect 的购买和安装是独立于数据仓库代理程序进行的。OS/390 代理程序可以通过 Classic Connect ODBC 驱动程序来访问 IMS 和 VSAM。通过 Classic Connect，可以为 IMS 和 VSAM 数据集设置类似 DB2 的定义，并使用 ODBC 来访问它们。

OS/390 代理程序根据请求是针对 Classic Connect 还是 DB2 来装入正确的 ODBC 驱动程序。如果在访问 DB2 源，则代理程序会装入 DB2 ODBC 驱动程序。如果正在访问 VSAM 或 IMS 源，代理程序会装入 Classic Connect ODBC 驱动程序。然后，会处理代理程序的请求。

设置 Classic Connect ODBC 驱动程序和仓库访问

Classic Connect 的购买和安装是独立于 OS/390 代理程序进行的。Classic Connect 可查看作为一个或多个关系表的单个文件或文件的一部分。必须映射 IMS 和 VSAM 数据，以便使 Classic Connect 能够访问它。可以人工映射数据或者使用 Microsoft Windows Classic Connect 非关系数据映射器来映射数据。

1. 在 OS/390 上安装 Classic Connect 数据服务器。
2. 可选：在 NT 上安装“Classic Connect 数据映射器”产品。
3. 定义 Classic Connect 的逻辑表定义，以便 Classic Connect 可以根据关系来访问数据。可以使用数据映射器来为 IMS 和 VSAM 结构创建定义，或人工创建定义。
4. 在设置 Classic Connect 之后，就可以设置对仓库的访问。
 - a. 创建 Classic Connect .ini 文件。

样本 Classic Connect 应用程序配置文件 cxa.ini 在 /usr/lpp/DWC/ 目录中，而在此处重新生成了该文件：

```
* national language for messages
NL = US English
* resource master file
NL CAT = usr/lpp/DWC/v4r1m00/msg/engcat
FETCH BUFFER SIZE = 32000
DEFLOC = CXASAMP
USERID = uid
USERPASSWORD = pwd
DATASOURCE = DJX4DWC tcp/9.112.46.200/1035
MESSAGE POOL SIZE = 1000000
```

- b. 更新 .ini 文件中的 DATASOURCE 行。此行包含一个数据源名称和一个协议地址。该数据源名称必须对应于在“Classic Connect 数据服务器”上定义的“查询处理器”名称，该名称在数据服务器配置文件的 QUERY PROCESSOR SERVICE INFO ENTRY 中。协议地址可在 TCP/IP SERVICE INFO 项的同一文件中找到。定义仓库数据源时将使用此文件中的 USERID 和 USERPASSWORD。
- c. 将 CXA_CONFIG 环境变量导出至 Classic Connect 程序文件，该文件通常在 .ini 文件所在的目录中。
- d. 更新 LIBPATH 环境变量以包括指向 Classic Connect 程序文件的路径，这些文件通常在 .ini 文件所在的目录中。

- e. 可选：利用测试程序 `cxasamp` 来验证安装：从包含 `.ini` 文件的目录中输入 `cxasamp`。 `location/uid/pwd` 是在 `.ini` 文件中定义的数据源名称 / 用户标识 / 用户口令。
- f. 按照定义任何 DB2 数据源的方式来对仓库定义数据源。

不需要更新 `dsnaoini` 文件，因为“DB2 OS/390 版”没有驱动程序管理器。Classic Connect 的驱动程序管理器被构建为 OS/390 代理程序。

运行“DB2 OS/390 版”实用程序

必须对 OS/390 代理程序应用 APAR PQ44904，以便使用代理程序来运行实用程序。

DSNUTILS 是在 WLM 和 RRS 环境中运行的“DB2 OS/390 版”存储过程。可使用它来运行通过使用用户定义存储过程接口安装的任何 DB2 实用程序。DB2 OS/390 版 LOAD、REORG 和 RUNSTATS 实用程序具有特性表，用来更改实用程序的运行方式。要更改实用程序的特性，在“进程造型程序”窗口中用鼠标右键单击该实用程序，并单击特性。

“仓库管理器”还为 DSNUTILS 提供了一个接口，以便可在“仓库管理器”步骤中包含 DB2 实用程序。

要设置 DSNUTILS 存储过程：

1. 安装 DB2 时，运行 DSNTIJSJ 作业来设置和绑定 DSNUTILS 存储过程。确保 DSNUTILS 的定义具有 PARAMETER STYLE GENERAL。
2. 启用 WLM 管理的存储过程。
3. 设置 RRS 和 WLM 环境。
4. 运行由 DB2 提供的样本批处理 DSNUTILS 程序。（建议执行此步骤，但不是必需的。）
5. 将 DSNUTILS 计划与 DSNCLI 计划绑定在一起，以便 CLI 可以调用存储过程：
`BIND PLAN(DSNAOCLI) PKLIST(*.DSNAOCLI.*, *.DSNUTILS.*)`
6. 使用“仓库管理器”来设置步骤，并运行该步骤。填充类型应该为 APPEND。如果不是该类型的话，则“仓库管理器”将在运行实用程序之前删除表中的所有内容。

使用 LOAD 实用程序来复制“DB2 OS/390 版”表之间的数据

假设您想通过将表卸装到平面文件中来复制该表，则应将平面文件装入到另一个表中。为此，通常需要卸装数据、编辑卸装生成的装入控制语句，然后装入数据。通过使用仓库，可以指定您想重新装入到另一个表中，而不需要在步骤之间停止并人工编辑控制语句。操作方式是：使用 Reorg/Generic 接口来创建一个步骤，以使用 UNLOAD 实用程序或 REORG TABLESPACE 实用程序来卸装文件。这两个实用程序生成两个输出数据集，一个带有表数据，一个带有可作为 LOAD 的输入的实用程序控制语句。在实用程序生成的控制语句中，INTO TABLE 表名就是卸装的表的名称。以下是可以用于 Reorg Unload 步骤的 DSNUTILS 参数的示例：

表 33. Reorg Unload 步骤的特性

UTILITY_ID	REORGULX
RESTART	NO
UTSTMT	REORG TABLESPACE DBVW.USAINENT UNLOAD EXTERNAL

表 33. Reorg Unload 步骤的特性 (续)

UTILITY_NAME	REORG TABLESPACE
RECDSN	DBVW.DSNURELD.RECDSN
RECDEVT	SYSDA
RECSpace	50
PNCHDSN	DBVW.DSNURELD.PNCHDSN
PNCHDEVT	SYSDA
PNCHSPACE	3

使用 Reorg/Generic DSNUTILS 接口来创建装入步骤。通常，DSNUTILS 实用程序语句参数指定实用程序控制语句。仓库实用程序接口还允许实用程序语句字段中存在文件名。可以使用关键字 :FILE: 来指定包含有效控制语句的文件，还可以使用关键字 :TABLE: 来指定您想装入的表的名称。要使用 LOAD 实用程序来处理先前示例的输出，在 LOAD 特性中应用下列参数值：

注：在 UTSTMT 字段中，输入装入语句或者从带有 UNLOAD EXTERNAL 选项的 REORG 实用程序中输出的文件的名称。

表 34. LOAD 步骤特性

UTILITY_ID	LOADREORG
RESTART	NO
UTSTMT	:FILE:DBVW.DSNURELD.PNCHDSN:TABLE:[DBVW].INVENTORY
UTILITY_NAME	LOAD
RECDSN	DBVW.DSNURELD.RECDSN
RECDEVT	SYSDA

这将适用于在相同或不同 DB2 子系统上的“DB2 OS/390 版”源表和目标表。控制语句平面文件可以是 HFS 或本机 MVS 文件。

有关可供 OS/390 使用的 DSNUTILS 和 DB2 实用程序的更详细的信息，参见 DB2 for OS/390 Utility Guide and Reference。

复制

可使用 OS/390 代理程序来使 Data Propagator 复制应用步骤自动进行。复制需要源数据库、控制数据库和目标数据库。这些数据库可以是不同或相同的数据库。Capture 作业会读取 DB2 日志，以确定在源数据库中添加、更新或更改了哪些行。然后，该作业会将更改写入更改数据表。然后，会运行 Apply 作业以对目标数据库应用这些更改。

“DB2 仓库管理器”软件包可通过创建复制步骤来自动执行 Apply 作业。使用“仓库管理器”来定义要运行的 Apply 作业的类型以及何时运行该作业。您需要将 SASNLINK 库导出至 steplib 环境变量。

将复制支持添加至“数据仓库中心”模板

“数据仓库中心”包括用于复制支持的 JCL 模板。如果计划使用 OS/390 代理程序来运行 Apply 程序，则需要对 OS/390 系统更改此模板中的帐户和数据集信息。要更改模板：

1. 以有权复制和更新 /usr/lpp/DWC/ 目录中的文件的标识进行登录。
2. 查找 apply.jcl, 并以 systemname.apply.jcl 为文件名来复制此文件, 其中, systemname 是 MVS 系统的名称。例如, 在 STLMVS1 上, 创建称为 STLMVS1.apply.jcl 的文件的副本。
3. 使用文本编辑器来定制 JCL 以满足站点的需求。更改帐户信息以与标准帐户信息相匹配, 并更改 STEPLIB DD 和 MSGS DD 的数据集以供 MVS 系统使用。
4. 如果需要的话, 更改 EXEC 卡上的程序名。有关更改程序名的详细信息, 参见 *DB2 Replication Guide and Reference*。不要更改包含在括号内的任何参数, 例如 [USERID] 和 [APPLY_PARMS]。(这些括号分别是十六进制字符 x'AD' 和 x'BD'。如果在 SPF 选项 0 中未将 TSO 终端类型设置为 3278A, 则这些值可能会显示为特殊字符而不是括号。如果不更改 x'AD' 或 x'BD' 或这些字符之间的任何数据, 就不会存在此问题)。
5. 将环境变量 VWS_TEMPLATES 更新为指向复制的模板文件的目录。

以下示例显示了包括在“数据仓库中心”中的 JCL 模板:

应用 JCL 模板:

```
//[USERID]A JOB ,MSGCLASS=H,MSGLEVEL=(1,1),
// REGION=2M,TIME=1440,NOTIFY=&SYSUID
//* DON'T CHANGE THE FIRST LINE OF THIS TEMPLATE.
//* THE REMAINING JCL SHOULD BE MODIFIED FOR YOUR SITE.
//*****
//* RUN APPLY/MVS ON OS/390 DB2 6.1 *
//*****
//ASNARUN EXEC PGM=ASNAPV66,REGION=10M,
// [APPLY_PARMS]
//STEPLIB DD DISP=SHR,DSN=DPROPR.V6R1M0.SASNLINK
// DD DISP=SHR,DSN=DSN610.SDSNLOAD
//MSGS DD DSN=DPROPR.V2R1M0A.MSGS,DISP=SHR
//ASNASPL DD DSN=&&ASNASPL,DISP=(NEW,DELETE,DELETE),
// UNIT=SYSDA,SPACE=(CYL,(10,1)),
// DCB=(RECFM=VB,BLKSIZE=6404)
//SYSTEM DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//
```

代理程序日志记录

许多“DB2 仓库管理器”组件(如服务器、记录器、代理程序和某些“数据仓库中心”程序)会将日志写至日志记录目录, 该目录是在 VWS_LOGGING 环境变量中指定的。这些日志文件是纯文本。

可以从“数据仓库中心”来启动代理程序日志记录。在左窗格中, 用鼠标右键单击**仓库**, 并单击**特性**。在“跟踪级别”标记上, 将设置更改为您想要的跟踪级别。代理程序跟踪支持级别 0 至 4:

- 级别 1 - 入口 / 出口跟踪
- 级别 2 - 级别 1 加上调试跟踪
- 级别 3 - 级别 2 加上数据跟踪
- 级别 4 - 内部缓冲区跟踪

当跟踪被设置得比级别 1 要高时, 性能会有所降低。仅当用于调试目的时才打开跟踪。跟踪信息存储在文件 AGNTxxx.LOG 中。环境信息存储在文件 AGNTxxx.SET 中。

在 Windows NT 上进行客户机方高速缓存

如果用户尝试使用有效令牌通过共享驱动器访问驻留在安装有 DB2 Datalinks 的 Windows NT 服务器上的 READ PERM DB 文件，该文件就会如期打开。但是，在此之后，使用同一令牌的后续打开请求并不会真正到达服务器，但会从客户机上的高速缓存中接受服务。即使令牌到期，文件的内容对用户来说仍然是可见的，因为该项仍在高速缓存中。但是，如果文件驻留在 Windows NT 工作站上，则不会发生此问题。

解决方案就是在 Windows NT 服务器上注册表项 `\\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Lanmanserver\Parameters\EnableOpLocks` 设置为零。通过此注册表设置，每当通过共享驱动器从客户机工作站访问驻留在服务器上的文件时，请求总是会到达服务器，而不是从客户机高速缓存接受服务。因此，必须对所有请求重新确认令牌。

此解决方案的负面影响就是这会影响到通过共享驱动器对服务器上的所有文件进行访问时的总体性能。即使有了此设置，如果该文件是通过映射服务器本身的共享驱动器进行访问的，与通过另一客户机高速缓存进行访问相比，该请求仍然是从高速缓存接受服务的。因此，令牌到期没有影响。

注：在所有情况下，如果文件访问是本地访问而不通过共享驱动器进行，则令牌验证和后续令牌到期将会如期发生。

企业版 UNIX CD-ROM 上的试用产品

“DB2 通用数据库 (UDB) 企业版 (EE) CD-ROM UNIX 版”平台的版本 6 和版本 7 中包含“DB2 Connect 企业版 (CEE)”的 90 天试用版本。因为 DB2 Connect 功能被构建到 DB2 UDB EE 产品中，所以，在安装了 DB2 UDB EE 的系统上不需要安装 DB2 CEE 产品就可以使用 DB2 Connect 功能。如果您安装了 DB2 CEE 的 90 天试用版本，并决定升级至发放了许可证的版本，则您必须购买 DB2 CEE 产品并安装 DB2 CEE 许可证密钥。您不需要重新安装该产品。在 DB2 EE 或 DB2 CEE UNIX 版“快速入门”一书中提供了关于安装许可证密钥的指导。

如果随 EE 安装一起安装了试用 CEE 产品，且您不想永久安装 CEE，则可以按照这些指导来除去 CEE 90 天试用版本。如果您除去 Connect EE 的试用版本，则 DB2 EE 仍将具有 DB2 Connect 功能。

要除去 DB2 Connect 版本 7，从各相应平台卸载下列文件集：

- 在 AIX 上，卸载 db2_07_01.clic 文件集。
- 在 NUMA-Q 和 Solaris 操作环境中，卸载 db2clic71 软件包。
- 在 Linux 上，卸载 db2clic71-7.1.0-x RPM。
- 在 HP-UX 上，卸载 DB2V7CONN.clic 文件集。

要除去 DB2 Connect 版本 6，从各相应平台卸载下列文件集：

- 在 AIX 上，卸载 db2_06_01.clic 文件集。
- 在 NUMA-Q 和 Solaris 操作环境中，卸载 db2cplic61 软件包。
- 在 Linux 上，卸载 db2cplic61-6.1.0-x RPM。
- 在 HP-UX 上，卸载 DB2V6CONN.clic 文件集。

DB2 Connect 企业版 UNIX CD-ROM 上的试用产品

“DB2 Connect 企业版 (EE)” CD-ROM UNIX 版”平台的版本 6 和版本 7 中包含“DB2 通用数据库 (UDB) 企业版 (EE)”的 90 天试用版本。DB2 UDB EE 90 天试用版的目的是供评估之用，而不为 DB2 Connect 所必需。

如果您安装了 DB2 UDB EE 的 90 天试用版本，并决定升级至发放了许可证的版本，则您必须购买 DB2 UDB EE 产品，并安装 DB2 UDB EE 许可证密钥。您不需要重新安装该产品。在 DB2 EE 或 DB2 CEE UNIX 版“快速入门”一书中提供了关于安装许可证密钥的指导。如果随 Connect EE 安装一起安装了试用 UDB EE 产品，并且您不想永久安装 UDB EE，则可以按照这些指导来除去 EE 90 天试用版本。如果您除去了 DB2 UDB EE 的试用版本，这不会影响 DB2 Connect EE 的功能。

要除去 DB2 UDB EE 版本 7，从各相应平台卸载下列文件集：

- 在 AIX 上，卸载 db2_07_01.elic 文件集。
- 在 NUMA-Q 和 Solaris 操作环境中，卸载 db2elic71 软件包。
- 在 Linux 上，卸载 db2elic71-7.1.0-x RPM。
- 在 HP-UX 上，卸载 DB2V7ENTP.elic 文件集。

要除去 DB2 UDB EE 版本 6，从各相应平台卸载下列文件集：

- 在 AIX 上，卸载 db2_06_01.elic 文件集。
- 在 NUMA-Q 和 Solaris 操作环境中，卸载 db2elic61 软件包。
- 在 Linux 上，卸载 db2elic61-6.1.0-x RPM。
- 在 HP-UX 上，卸载 DB2V6ENTP.elic 文件集。

删除 Data Links Manager

现在可以删除指定数据库的 DB2 Data Links Manager 了。某些与 Data Links 相关的 SQL 请求的处理以及实用程序（如，备份 / 恢复）涉及到与为数据库配置的所有 DLM 的通信。以前，DB2 不能删除已配置的 DLM，尽管它可能尚未运作。这在 SQL 和实用程序处理中导致了附加的开销。一旦添加了 DLM，引擎在处理请求时就会与它通信，这可能导致了某些 SQL 请求（例如，删除表 / 表空间 / 数据库）的故障。

使用 SMIT 卸载 DLFM 组件时可能会除去附加文件集

在从在其上安装了 Data Links Manager 的 AIX 机器中卸载 DB2（版本 5、6 或 7）之前，应遵循下列步骤：

1. 作为 root 用户，使用以下命令来制作 /etc/vfs 的副本：

```
cp -p /etc/vfs /etc/vfs.bak
```
2. 卸载 DB2。
3. 作为 root 用户，将 /etc/vfs 替换为在第 1 步中制作的备份副本。

```
cp -p /etc/vfs.bak /etc/vfs
```

在 Windows 2000 上使用 CLP 时发生错误 SQL1035N

如果 DB2 被安装到只有某些用户（例如，管理员）才具有写访问权的目录中，则当常规用户试图使用 DB2 “命令行处理器”时可能会接收到错误 SQL1035N。

要解决此问题，应将 DB2 安装到一个所有用户都具有写访问权的目录中。

SQL 辅助程序的增强功能

“SQL 辅助程序”工具现在允许用户指定除了 “=” 之外的连接运算符来进行表连接。“连接类型”对话框已经被增强为包括连接运算符的下拉列表，可以通过单击“SQL 辅助程序”工具的“连接”页上的“连接类型”按钮来启动该对话框。

可用的运算符是 “=”、“<>”、“<”、“>”、“<=” 和 “>=”。“SQL 辅助程序”是一个帮助用户创建简单 SQL 语句的工具。可以从“命令中心”（“交互式”选项卡）、“控制中心”（“创建视图”和“创建触发器”对话框）、“存储过程构建器”（“插入 SQL 存储过程”向导）和“数据仓库中心”（“SQL 进程”步骤）中得到它。

在 Linux 上对 DB2 的 Gnome 和 KDE 桌面集成

DB2 现在包括一组实用程序，可以用来创建 DB2 桌面文件夹和图标，以便在 Gnome 和 KDE 桌面上启动最常用的 DB2 工具，从而进行受支持的基于 Intel 的 Linux 分发。缺省情况下，这些工具是由 DB2 版本 7.2 安装的，并且在安装之后可以用来为一个或多个选择的用户创建和除去桌面图标。

要为一个或多个用户添加一组桌面图标，使用以下命令：

```
db2icons <user1> [<user2> <user3>...]
```

注：注意，如果图标是在 Gnome 或 KDE 桌面环境正在运行时生成的，则用户可能需要强制人工刷新桌面才能查看新图标。

要为一个或多个用户除去一组桌面图标，使用以下命令：

```
db2rmicons <user1> [<user2> <user3>...]
```

注：您必须具有足够的权限才能为其他用户生成或除去图标。通常，如果您是普通用户，则只能使用 **db2icons** 和 **db2rmicons** 来为您自己创建或除去图标，仅当您是 root 用户或者是有权写入指定用户的主目录的另一用户，您才能为其他用户创建或除去图标。

在 Windows 2000 Terminal Server, Administration Mode 下运行 DB2

对于 DB2 UDB 版本 7.1 修订包 3 和更新版本，DB2 可以在 Windows 2000 Terminal Server, Administration Mode 下运行。在此之前，您不能在 Windows 2000 Terminal Server, Administration Mode 的“客户机”会话中运行 DB2。

关于备份和恢复命令的联机帮助

当您输入 db2 ? backup 时会出现不正确的信息。正确的输出是：

```
BACKUP DATABASE database-alias [USER username [USING password]]
[TABLESPACE (tblspace-name [ {,tblspace-name} ... ])] [ONLINE]
[INCREMENTAL [DELTA]] [USE TSM [OPEN num-sess SESSIONS]] |
TO dir/dev [ {,dir/dev} ... ] | LOAD lib-name [OPEN num-sess SESSIONS]]
[WITH num-buff BUFFERS] [BUFFER buffer-size] [PARALLELISM n]
[WITHOUT PROMPTING]
```

当您输入 db2 ? restore 时出现了不正确的信息。正确的输出是:

```
RESTORE DATABASE source-database-alias { restore-options | CONTINUE | ABORT }";
```

```
restore-options:";
[USER username [USING password]] [{TABLESPACE [ONLINE] |};
TABLESPACE (tblspace-name [ {,tblspace-name} ... ])] [ONLINE] |";
HISTORY FILE [ONLINE]}} [INCREMENTAL [ABORT]]";
[USE TSM [OPEN num-sess SESSIONS] |";
FROM dir/dev [ {,dir/dev} ... ] | LOAD shared-lib";
[OPEN num-sess SESSIONS]}} [TAKEN AT date-time] [TO target-directory]";
[INTO target-database-alias] [NEWLOGPATH directory]";
[WITH num-buff BUFFERS] [BUFFER buffer-size]";
[DLREPORT file-name] [REPLACE EXISTING] [REDIRECT] [PARALLELISM n]";
[WITHOUT ROLLING FORWARD] [WITHOUT DATALINK] [WITHOUT PROMPTING]";
```

“仓库管理器” 应为 “DB2 仓库管理器”

产品屏幕和产品文档中所有出现短语“仓库管理器”的地方都应为“DB2 仓库管理器”。

第9部分 附加信息

附加信息

DB2 通用数据库和 DB2 Connect 联机支持

要获取 DB2 信息（包括有关在此文档发布之后发现的问题的信息）的完整和最新的来源，使用“DB2 通用数据库和 DB2 Connect 联机支持”Web 站点，网址为 <http://www.ibm.com/software/data/db2/udb/winos2unix/support>。

DB2 Magazine

有关 DB2 系列产品的最新信息，可通过免费预订 DB2 Magazine 来获得。联机版本的杂志是在 <http://www.db2mag.com> 提供的；有关申请预订的指导在此站点上。

第10部分 附录

附录. 声明

IBM 可能未在所有国家中提供本文中讨论的产品、服务或功能部件。关于您所在区域目前可用的产品及服务的信息，请向当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并不说明或暗示只能使用 IBM 的产品、程序或服务。凡是同等功能的产品、程序或服务，只要不侵犯 IBM 的知识产权，都可以用来替代 IBM 产品、程序或服务。当然，评估和验证非 IBM 产品、程序或服务均由用户自行负责。

本文档的议题可能涉及 IBM 的某些专利或正在申请中的专利的应用。提供本文档并不表示允许您使用这些专利。您可以将许可证查询以书面形式寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

关于双字节 (DBCS) 许可证查询的信息，请与您所在国家的 IBM 知识产权部门联系，将查询以书面形式寄往：

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

以下段落不适用于英国与其他当地法律不允许这种供应方式的国家：国际商用机器公司“按原样”出版此书，不做任何明确或暗示的担保，包括但不限于有关非伪造、商业性或符合特殊目的的隐含保证。一些地区在某些事务中不允许否认拒绝明确或暗示的担保，因此本条款可能不适合您。

本信息中可能有技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些信息将包含在本书新的版本中。IBM 可以随时对本书中说明的产品和/或程序进行改进和/或改动，而不必通知您。

此信息中对非 IBM Web 站点的任何引用仅是为了方便起见，而不以任何方式为那些 Web 站点作保证。那些 Web 站点的资料并非此 IBM 产品资料的一部分，使用那些 Web 站点的风险由您自己承担。

对于您所提供的任何信息，IBM 有权利以任何她认为适当的方式使用或散发，而不必对您负任何责任。

为了以下目的：(1) 允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换
(2) 允许对已经交换的信息进行相互使用，而希望获取本程序有关信息的合法用户请与下列地址联系：

IBM Canada Limited
Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario

M3C 1H7
CANADA

只要遵守适当的条款和条件，包括某些情形下的一定数量的付款，都可获取这方面的信息。

这些信息中描述的特许程序及其所有可用的特许资料，按 IBM 客户协议、IBM 国际程序许可证协议或任何等价的协议中的条款，由 IBM 提供。

此处包含的所有性能数据都是在受控环境中确定的。因此，在其他操作环境中获得的结果可能与之相差很大。某些测量可能是在开发级的系统上进行的，不能保证这些测量方法在通用系统上同样可用。此外，某些测量方法可能是通过外推法归纳来估计的。实际结果可能会有所不同。此文档的用户应针对他们的特定环境验证数据是否适用。

涉及非 IBM 产品的信息可从这些产品的供应商、其发行公告或其他公众可用源得到。IBM 未测试这些产品，因此不能确认性能的精确度、兼容性或其他对非 IBM 产品的索赔要求等。有关非 IBM 产品功能方面的问题可向它们的供应商提出。

所有关于 IBM 未来方向或意向的声明都可能随时更改或撤销，而不作任何通知，并且仅代表发展目标。

此信息包含了用于日常商业处理的数据和报表的示例。为了尽可能完整地说明问题，这些示例中包含了个人、公司、品牌和产品的名称。所有这些名称都是虚构的，如与实际商业企业所使用的名称和地址相似，纯属巧合。

版权许可证:

本信息中可能包含用源语言编写的示例应用程序，它们说明了各种不同的操作平台上的编程技术。您可以为了开发、使用、市场营销或分布应用程序(这些应用程序遵守编写这些示例程序的操作平台的应用程序接口)的目的，以任何形式复制、修改和分布这些示例程序，不用向 IBM 付费。这些例子未经所有条件下的完整测试。因此，IBM 不能保证或暗示其可靠性、可用性 or 这些程序的功能。

这些样本程序或任何派生产品的每个副本或任何部分必须包含如下的版权公告:

© (您的公司名称) (年度)。此代码各部分派生自“IBM 公司样本程序”。© Copyright IBM Corp. _输入年份_。All rights reserved.

注册商标

以星号 (*) 标出的下列术语是 IBM 公司在美国和 / 或其他国家的商标。

ACF/VTAM	IBMIMSIMS/ESA
AISPO	LAN DistanceMVS
AIXAIX/6000	MVS/ESAMVS/XA
AIXwindows	Net.Data
AnyNet	OS/2
APPNAS/400	OS/390
BookManager	OS/400PowerPC
CICSC Set++	QBIC
C/370	QMF
DATABASE 2	RACFRISC System/6000
DataHub	RS/6000
DataJoinerDataPropagator	S/370
DataRefresher	SP
DB2	SQL/DS
DB2 Connect	SQL/400
DB2 Extender	System/370
DB2 OLAP Server	System/390
DB2 Universal Database	SystemView
Distributed Relational Database Architecture	VisualAge
DRDA	VM/ESA
eNetwork	VSE/ESA
Extended Services	VTAM
FFST	WebExplorer
First Failure Support Technology	WIN-OS/2

下列术语是其他公司的商标或注册商标:

Microsoft、Windows 和 Windows NT 是微软公司的商标或注册商标。

Java 或所有基于 Java 的商标和徽标以及 Solaris 是 Sun 公司在美国和 / 或其他国家的商标。

Tivoli 和 NetView 是 Tivoli 系统公司在美国和 / 或其他国家的商标。

UNIX 是经 X/Open 有限公司唯一许可的在美国和 / 或其他国家的注册商标。

以双星号 (**) 标出的其他公司、产品或服务名, 可能是其他公司的商标或服务标志。

索引

[A]

安装

- 当使用可更换驱动器时挂起 6
- 停止 DB2 进程 15
- Windows 上需要的服务帐户 437

安装先决条件

- OS/390 的 Linux 上的 UDB EE 和 CEE 16

安装、安装、迁移、可执行文件 195

按需归档日志 73

按需日志归档 73

[B]

备份

- 脱机
 - 从分割镜像象 71
- 增量 69
- 至命名管道 71

备份服务 API

- DB2 支持 438

标记语言文件、DWC、ICM、ERwin 文件、ER1 文件 196

标量函数

- DAYOFWEEK_ISO 321
- WEEK_ISO 321

标识列

- 改变 65

标准输入文件 80

表达式

- 串联运算符 415
- 序列 332
- NEXTVAL 63
- PREVVAL 63
- string 415

表结构 147

表空间

- 设置为 ONLINE 状态 66

并行恢复 71

补丁级别

- 在 Solaris 版本 2.6 上 4

[C]

操作数

- string 415

查询结果, 查看 180

串联

- 运算符 415

存储过程

- 在“数据仓库中心”中 438

错误消息

- 将节点添加到分区数据库时 4
- 拒绝访问文件 176
- 无效快捷方式 176
- 在迁移期间 6
- 找不到联编文件 432
- SQL10012N 179

[D]

代码页

- 映射至语言环境 56
- 在使用 CLI 的应用程序中 59
- Baltic Rim 语言 59

导入、标记语言文件、ICM、信息目录管理器 198

导入、正在导入、标记语言文件、

- DWC 197

动态复合语句 84, 371

多站点更新测试连接

- 实例版本不兼容性 176

[F]

返回身份列值

- IDENTITY_VAL_LOCAL 函数 339

范围大小

- “创建数据库”向导中的设置 183

访问多个索引 88

非 DB2 仓库源, AIX 214

非 DB2 仓库源, Solaris 操作环境 214

非 db2 源, Sybase 的样本条目 214

分布式数据库

- 设置以更新 53

分割镜像

- 作为备用数据库 68

分割镜像处理

- 联机 68

分区键

- 更新 53

分区数据库

- 添加节点时发生错误 4

复合语句

- 动态 84

复合语句(动态)

- 变量 371

复合 SQL, 使用 318

复制

- 对表和列名的限制 142
- 对于非 IBM 服务器 141
- 方案 141

复制(续)

- 计划 143
- 随处更新先决条件 143
- 问题确定 144
- DATALINK 142
- 复制的总结表 53, 87
- 复制分析器 144
- 复制守护程序 433

[G]

高可用性 68

隔离级别

- 语句级别 83
- 在 SELECT 语句中 363

公共仓库元模型

- XML 支持 230

管理器配置文件 90

归档和检索

- 用户出口程序 80

归档日志 120

国家语言支持 60

[H]

函数

标量

- DECRYPT 335
- ENCRYPT 336
- GETHINT 338
- IDENTITY_VAL_LOCAL 339
- MQPUBLISH 343
- MQREAD 344
- MQRECEIVE 345
- MQSEND 346
- MQSUBSCRIBE 348
- MQUNSUBSCRIBE 349
- REC2XML 351

表

- MQREADALL 357
- MQRECEIVEALL 358

过程 360

- 在 Unicode 数据库中 415

DECRYPT 67

ENCRYPT 67

GETHINT 67

红旗 Linux

- 简体中文语言环境 6

缓冲池

- AWE 89

恢复

- 并行 71

恢复 (续)
 与 DB2 Data Links Manager 交互作用
 77
 增量 69
 DATALINK 列的问题 434

[J]

加密
 数据 67
加密信息
 ENCRYPT 函数 336
 GETHINT 函数 338
简体中文
 Linux 上的语言环境设置 6
解密信息
 DECRYPT 函数 335
进程造型程序
 数据仓库中心 230
进程, 在安装之前停止 15
警报中心 61
静态摘要表, 在 JDBC/ODBC/CLI 应用程序中 315
镜像
 日志 74

[K]

开放式客户机, 非 DB2 仓库源和 214
可访问性 437
可访问性功能部件 3
客户机
 三层环境 317
 下载 15
克隆数据库
 创建 68
控制中心
 版本 6, 用来管理卫星环境 110
 对并行小应用程序的限制 179
 故障诊断 178
 双向支持
 在 Windows NT 上 175
 用来管理 VM 和 VSE 数据库 175
 与 Internet Explorer 有关的错误 6
 运行已翻译版本 58
 在 OS/390 平台上 176
 在 UNIX 平台上进行故障诊断 178
 “工具设置”笔记本 179
 OS/2 上的 infopops 显示问题 178
 OS/390 需要的修正 178
 Windows 95 上的脚本中心问题 180
快速导览 431

[L]

例程
 过程 360

联合系统
 别名 33
 封装器 33
 约束 33
联合系统, 限制 435
列
 字符串指定, 基本规则 413

[M]

名称和地址清理, Trillium 202
命名管道
 备份至 71
命名规则 55
命名约定
 字符限制 55
模板
 Commit.tag
 标记 234
 示例值 234
 ForeignKeyAdditional.tag
 标记 236
 示例值 237
 ForeignKey.tag
 标记 234
 示例值 235
 PrimaryKeyAdditional.tag
 标记 239
 示例值 240
 PrimaryKey.tag
 标记 237
 示例值 238
模糊游标 383
模式名
 字符限制 55
模式造型程序
 数据仓库中心 231
目录视图
 SEQUENCES 403

[N]

内存窗口
 在 HP-UX 11 上 432

[P]

排序堆大小 91
排序堆阈值 91
配置参数
 最大锁定数 91
 app_ctl_heap_sz 93
 maxappls 93
 MIN_DEC_DIV_3 92
 mon_heap_sz 93

配置参数 (续)
 softmax 93

[Q]

启动 DB2
 在 Windows 95、98 和 ME 上使用
 db2start 151
迁移
 错误在 6
 视图的问题 15
 运行 db2iupdt 来更新实例 16
 在 Windows 2000 上 15

[R]

认证
 通过使用 DB2DOMAINLIST 的域
 433
日语
 Linux 上的语言环境设置 6
日志
 按需归档 73
 镜像 74
容器
 添加至 SMS 表空间 65
软件需求, ERwin 标记语言文件、数据仓库中心、DWC、信息目录管理器、ICM 195

[S]

使用“运行时客户机”来绑定数据库实用程序 315
视图
 在迁移之后不可用 15
授权
 对序列的公共创建 386
 授予对序列的创建 386
数据仓库中心
 使用存储过程 438
 iwh2exp2 命令语法 438
 OLE DB 支持 212
数据加密 67
数据类型
 在 Unicode 数据库中提升 412
数据清理, 名称和地址清理 202
数据移动
 对 Unicode 客户机的限制 421
双重记录 74
双向语言支持
 利用 Windows NT 上的“控制中心”
 175
死锁 154
搜索联机信息
 在 Solaris 操作环境上 9

锁定等待 154
锁定列表
 最大锁定数配置参数 91
锁定, 有利的 317
索引
 访问多个 88
 较大键 84
索引键
 较大 84

[T]

特权
 USAGE 67

[W]

网关
 三层环境 317
为列指定字符串, 规则 413
卫星
 安装说明 102
 交互式安装 103
 响应文件安装 102
 DB2 个人版 101
 DB2 工作组版 101
卫星管理中心
 版本 6, 用来管理卫星环境 110
卫星环境
 安装说明 102
 使用版本 6 卫星管理中心管理 110
 卫星的交互式安装 103
 卫星的响应文件安装 102
 先决条件 101
 “DB2 个人版”卫星 101
 “DB2 工作组版”卫星 101
无效快捷方式错误 176

[X]

系统目录统计信息
 收集分发统计信息 85
下一个键锁定 154
响应文件安装
 特定于卫星的关键字 102
 DB2 控制服务器关键字 101
向导
 创建数据库 183
 MQSeries 辅助程序 183
 OLE DB 辅助程序 183
消息
 关于 Capture 和 Apply 148
消息传递, MQ Series 209, 210
卸载
 DB2 DFS 客户机使能器 433
信息目录管理器
 采用希伯来语 435

信息目录管理器 (续)
 实用程序 253
性能监控器 61
序列 64
 创建 63
 调用 332
 改变 65
 删除 66
 特权 67
 nextval-expression 332
 prevval-expression 332
序列调用 332
序列, 描述 305

[Y]

样本内容, 查看 180
已触发的 SQL 语句
 SET Variable 语句 390
应急恢复
 使用 db2inidb 工具 68
用户标识
 字符限制 55
游标
 并行 317
 键集驱动的
 更新 317
 可更新 317
 可滚动 317
 服务器端 317
 键集驱动的 317
 静态 317
 应用程序故障诊断 318
 OS/390 317
 类型 317
 灵敏度 317
 属性, 缺省值 317
 值并行性 317
语句级别隔离 83
语言环境
 国家语言版本 56
 在红旗 Linux 上的简体中文 6
 “管理服务器”与实例之间的兼容性
 59, 436
 Linux 上的日语和简体中文 6
语言环境设置
 在使用 CLI 的应用程序中 59
预取器
 在并行 DMS 容器创建和扩展中 61
远程客户机访问
 为 NT 配置通信服务器 434

[Z]

暂挂的 I/O 功能
 支持持续可用性 68

增量备份和恢复 69
增量提交与替换 213
指定
 字符串, 基本规则 413
注册表变量
 新的和更改过的 95
 以支持备份服务 API 438
 DB2DOMAINLIST 433
 DB2_AWE 90
转换规则
 用于字符串比较 415
 用于, 运算组合字符串 415
子元素统计信息 85
自动装入程序
 在创建子进程期间挂起 434
字符串
 指定, 覆盖 413
字符转换
 规则, 用于运算组合字符串 415
 规则, 在比较字符串时 415
组件跟踪数据文件名 213

[特别字符]

“创建数据库”向导
 设置范围大小 183
“拒绝访问文件”错误 176
“MQSeries 辅助程序”向导 183
“OLE DB 辅助程序”向导 183

A

Address Windowing Extensions (AWE) 89
Adobe Acrobat Reader
 利用非英语语言环境来访问 10
ADT 变换 315
AIX 4.3.3
 自动装入程序在创建子进程之后挂起
 434
ALTER SEQUENCE 语句
 详细描述 365
app_ctl_heap_sz 93
AS/400
 Capture 和 Apply 145

B

Baltic Rim 语言
 代码页支持 59

C

Capture 和 Apply
 从应用程序内启动 148
 错误消息 148
 在 AS/400 上 145

CLI
 编写 Unicode 应用程序 416
CLI 存储过程
 限制 318
 自动绑定 318
CLI 函数
 SQLBindFileToParam 319
 SQLNextResult 319
CLI 接口
 代码页设置 59
Commit.tag 模板
 标记 234
 示例值 234
CREATE PROCEDURE 语句
 动态复合语句 371
 FOR 语句 396
 RETURN 语句 399
 SIGNAL 语句 401
CREATE PROCEDURE statement
 SQL procedure statement 394
CREATE SEQUENCE 语句
 详细描述 377
CWM
 XML 支持 230

D

Data Links File System
 与“Norton 实用程序”不兼容 56
DataJoiner
 约束 435
DATALINK
 复制 142
 恢复时的问题 434
DAYOFWEEK_ISO 标量函数 321
DB2 服务器 VM 版和 VSE 版
 从“控制中心”管理数据库 175
DB2 个人版
 设置为卫星 101
 先决条件 101
DB2 控制服务器
 版本 6, 升级 104
 响应文件安装 101
 最小先决条件 101
DB2 企业版
 版本 6, 用作 DB2 控制服务器 104
 设置为卫星 101
 先决条件 101
DB2 Connect
 有关 Sun Cluster 2.2 的先决条件 54
DB2 Data Links Manager
 与恢复交互作用 77
DB2 Relational Connect
 在 AIX、Solaris 和 Linux 上安装 35
 在 Windows NT 上安装 34
db2ArchiveLog 276

DB2CODEPAGE 注册表变量
 在使用 CLI 的应用程序中 59
db2expln
 运行 99
db2inidb 工具 68
db2iupdt 命令, 在升级 DB2 之后运行 16
DB2MCS 实用程序, 机器重新引导以设置 PATH 53
db2start
 在 Windows 95、98 和 ME 上 151
db2_all 80
DB2_AWE 注册表变量 90
DB2_INDEX_2BYTEVARLEN 64
DECLARE CURSOR 383
DECRYPT 函数
 详细描述 335
 值和自变量 335
DFS 客户机使能器
 卸载 433
dlfm client_conf
 故障原因 433
dl_expint 设置建议 92
DMS 表空间
 并行创建和调整大小 66
DMS 容器
 以并行方式创建和扩展 61
DWC
 CWM XML 支持 230
 OLE DB 支持 212
dynexpln
 运行 99

E

enable_MQFunctions 169, 171
ENCRYPT 函数
 详细描述 336
 值和自变量 336
error messages
 executing triggers 383
ERwin、元数据、抽取、导入 194

F

FOR 语句 396
ForeignKeyAdditional.tag 模板
 标记 236
 示例值 237
ForeignKey.tag 模板
 标记 234
 示例值 235

G

GETHINT 函数
 详细描述 338

GETHINT 函数 (续)
 值和自变量 338
GRANT
 CREATE ON SEQUENCE 386
GRANT (序列特权) 语句
 详细描述 386
GUI 工具
 绑定错误 432

H

HP 和 Sun Solaris
 备份和恢复支持 74
HP-UX
 不支持 Query Enabler 437
HP-UX 11
 内存窗口 432

I

IDENTITY 列 64
IDENTITY_VAL_LOCAL 函数
 详细描述 339
 值和自变量 339
inoperative triggers
 detailed description 382
Internet Explorer
 与“控制中心”有关的错误 6
IPX/SPX 协议支持, 在 Windows 2000 上 15
isolation levels
 in DELETE statement 384
 in INSERT statement 387
 in SELECT INTO statement 388
 in UPDATE statement 392
iwh2exp2 命令
 新选项 438

J

Java 方法
 在 CREATE PROCEDURE 或
 CREATE FUNCTION 中 312
JDBC
 可滚动游标, 访问 317
JDBC/ODBC/CLI 静态摘要表, 限制 315
JDK 1.1
 安装路径 179

L

Linux
 日语和简体中文的语言环境设置 6
 为“控制中心”设置环境 16

M

maxapps 93
Microsoft SQL Server 数据源
 代码页 47
 环境变量 43
 链接至 DB2 44
 提高性能 44
 DSN 名 45
 MERANT 封装器 45
 MERANT 库 42
 ODBC 跟踪 47
 ODBC 驱动程序 42
mon_heap_sz 93
MQ 函数 169, 171
MQPUBLISH 函数
 详细描述 343
 值和自变量 343
MQREAD 函数
 详细描述 344
 值和自变量 344
MQREADALL 函数
 详细描述 357
 值和自变量 357
MQRECEIVE 函数
 详细描述 345
 值和自变量 345
MQRECEIVEALL 函数
 详细描述 358
 值和自变量 358
MQSEND 函数
 详细描述 346
 值和自变量 346
MQSUBSCRIBE 函数
 详细描述 348
 值和自变量 348
MQUNSUBSCRIBE 函数
 详细描述 349
 值和自变量 349

N

Netscape
 出错信息 10
 错误消息 181
 用来访问联机信息 9
NEXTVAL 63
nextval-expression 332
NLV 支持
 Unicode 409
Norton 实用程序
 与 Data Links File System 不兼容 56
NUMA-Q
 不支持 Query Enabler 或 Tracker 437

O

ODBC
 可滚动游标, 访问 317
 取装方向 317
ODBC Unicode 应用程序 420
OLE DB 支持 212
Oracle 数据源
 环境变量 35
 Linux 封装器 35
 Solaris 封装器 35
OS/2
 安装“控制中心” 176
 消息必需的 JDK 级别 295

P

PREVVAL 63
prevval-expression 332
PrimaryKeyAdditional.tag 模板
 标记 239
 示例值 240
PrimaryKey.tag 模板
 标记 237
 示例值 238

Q

Query Enabler
 在 HP-UX 和 NUMA-Q 上不受支持 437
Query Patroller
 在迁移之后失去用户特权 437
Query Patroller Tracker
 在 NUMA-Q 上不受支持 437

R

REC2XML 函数
 详细描述 351
 值和自变量 351
RETURN 语句 399

S

search discovery 432
SET CONSTRAINTS 语句
 被 SET INTEGRITY 替换 56
SET ENCRYPTION PASSWORD 67
SET ENCRYPTION PASSWORD 语句
 详细描述 389
SET INTEGRITY 语句
 替换 SET CONSTRAINTS 56
SET Variable 语句
 详细描述 390
SIGNAL 语句 401

SIGTTIN 80
SMS 表空间
 添加容器 65
SNA 服务器
 从主机或者 AS/400 应用程序访问
 UDB 436
SNA SPM
 在重新引导之后验证启动 436
softmax 93
Solaris
 缺乏对 UltraSparc 之前的版本的支持 4
 在版本 2.6 上必需的补丁级别 4
Solaris 操作环境
 搜索联机信息 9
Solaris 操作系统
 64 位
 配置 155
 约束 155
Solaris 上的高可用性 54
SQL 变量 371
SQL 过程
 变量 371
 动态复合语句 371
 DECLARE 语句 371
 FOR 语句 396
 RETURN 语句 399
 SIGNAL 语句 401
SQL 语句
 ALTER SEQUENCE 365
 CREATE SEQUENCE 377
 GRANT (序列特权) 386
 SET ENCRYPTION PASSWORD 389
 SET Variable 390
SQL, 复合 318
SQLBindFileToParam CLI 函数, 校正 319
SQLNextResult 函数 319
string
 表达式 415
 操作数 415
Sun Cluster 2.2
 DB2 Connect 先决条件 54
Sun Solaris 和 HP
 备份和恢复支持 74
SWITCH ONLINE 子句 66
Sybase
 与 Windows 上 ICM 的不兼容性 7
Sybase 数据源 37
 代码页 42
 环境变量 37
 链接至 DB2 38
 提高性能 39
Sybase Open Client 37

T

triggers

- error messages 383
- inoperative 382

U

Unicode

- 函数 415
- 数据库和应用程序 409
- 文档更新 411
- 在 ODBC 应用程序中 420

Unicode 客户机

- 数据移动限制 421

USAGE 特权 67

V

Veritas

- Solaris 上的高可用性 54

VI

- 设置以与 DB2 配合使用 81

W

web 浏览器

- 对于 Windows 2000 的建议 9

WEEK_ISO 标量函数 321

Windows

- Sybase 与 ICM 之间的不兼容性 7

Windows 2000

- 建议的 web 浏览器 9
- 迁移 15
- IPXX/SPX 协议支持 15

X

XML 支持

- 对于“公共仓库元模型”(CWM) 230



Printed in the United States of America