



系统管理指南第一卷

Sybase IQ 15.4

文档 ID: DC00955-01-1540-01

最后修订日期: 2011 年 11 月

版权所有 © 2011 Sybase, Inc. 保留所有权利。

除非新版本或技术声明中另有说明, 否则本出版物适用于 Sybase 软件及所有后续版本。本档中的信息如有更改, 恕不另行通知。本出版物中描述的软件按许可证协议提供, 其使用或复制必须符合协议条款。

要订购其它文档, 美国和加拿大的客户请拨打客户服务部门电话 (800) 685-8225 或发传真至 (617) 229-9845。

持有美国许可证协议的其它国家/地区的客户可通过上述传真号码与客户服务部门联系。所有其它国际客户请与 Sybase 子公司或当地分销商联系。仅在软件的定期发布日期提供升级内容。未经 Sybase, Inc. 的事先书面许可, 不得以任何形式、任何手段 (电子的、机械的、手工的、光学的或其它手段) 复制、传播或翻译本出版物的任何部分。

可在 <http://www.sybase.com/detail?id=1011207> 上的 Sybase 商标页中查看 Sybase 商标。Sybase 和列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

SAP 和此处提及的其它 SAP 产品与服务及其各自的徽标是 SAP AG 在德国和世界各地其它几个国家/地区的商标或注册商标。

Java 和基于 Java 的所有标记都是 Sun Microsystems, Inc. 在美国和其它国家/地区的商标或注册商标。

Unicode 和 Unicode 徽标是 Unicode, Inc. 的注册商标。

本书中提到的所有其它公司和产品名均可能是与之相关的相应公司的商标。

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568。

目录

Sybase IQ 系统管理概述	1
读者	1
Sybase IQ 设计和用途	1
Sybase 数据库体系结构	1
快速访问众多数据源	1
数据仓库应用和 Sybase IQ	1
系统管理工具	2
数据库服务器	3
Sybase IQ 中的数据存储	3
SQL Anywhere 和 Sybase IQ	3
安全概述	4
并发操作	5
Multiplex 功能	5
存储过程	5
系统表和视图	6
命令和函数	6
SQL 语句的类型	7
函数	7
消息日志内容	7
消息日志中的版本信息	8
消息日志文件管理	10
实用程序数据库	13
可伸缩性	14
内存使用	14
数据装载	14
处理线程	15
磁盘空间	15
中间版本控制	15
页大小	16
基于列的索引	16
查询优化程序	16

schema 设计	16
UNION ALL 视图	16
Sybase IQ 启动	17
启动数据库服务器的方法	17
服务器启动实用程序	18
从 Windows “开始” 菜单启动服务器	19
服务器作为 Windows 服务	19
自动服务器启动	19
命令行开关	19
显示命令行选项	20
配置文件	20
Sybase IQ 文档中的命令行示例格式	21
必需的命令行选项	21
缺省配置文件	21
演示数据库的配置文件	22
命名限制	22
服务器和数据库名称	22
用于性能的命令行选项	23
控制权限的命令行选项	27
最大目录页大小	28
客户端/服务器环境选项	29
强制恢复模式选项	30
从 Interactive SQL 启动服务器	30
共享内存冲突	31
服务器活动日志	32
为服务器日志文件命名	34
UNIX 日志文件	34
关闭数据库服务器	34
何时停止和重新启动服务器	34
停止数据库服务器的方法	35
谁可以停止服务器	37
关闭操作系统会话	37
启动和停止数据库的方法	38
数据库启动准则	38

停止数据库的方法	39
如何启动 iqdemo 数据库	39
Sybase Central 启动和停止方法	40
Sybase IQ 连接	41
连接状态	41
Sybase IQ 如何建立连接	41
连接学习路线图	42
连接参数的工作原理	43
连接字符串的格式	43
应用程序如何传递连接参数	44
ODBC 数据源中的连接参数	44
Sybase Central 或 Interactive SQL 连接	45
“连接”对话框	45
打开“连接”对话框 (Sybase Central)	46
打开“连接”对话框 (Interactive SQL)	46
用于连接的驱动程序	46
Sybase Central 中的连接快捷方式	47
服务器对象	47
创建连接配置文件	48
简单连接示例	48
从 Sybase Central 连接到演示数据库	49
从 Interactive SQL 连接到演示 数据库	49
数据库状态对本地连接的影响	50
连接到本地服务器上正在运行的数据库	50
连接到未运行的数据库	51
在 UNIX 上从 Interactive SQL 连接到 数据库	51
从 UNIX 系统连接	52
从 Windows 系统连接	52
连接到嵌入式数据库	53
使用数据源进行连接	54
缺省连接参数	55
从 Sybase IQ 实用程序连接	56
ODBC 数据源	57
数据源保存在何处	57

从 ODBC 管理器中创建数据源	58
从命令行创建 ODBC 数据源	59
测试 ODBC 数据源	60
在 ODBC 管理器中配置 ODBC 数据源	60
“ODBC” 选项卡	60
文件数据源	61
使用 ODBC 管理器创建文件数据源	62
文件数据源和文本编辑器	62
UNIX 中的 ODBC 数据源	62
iAnywhere Solutions Oracle 驱动程序	64
在 UNIX 中创建 Oracle DSN	65
在 Windows 中创建 Oracle DSN	65
使用 IQDSN 创建 Oracle DSN	65
使用 OLE DB 连接数据库	66
OLE DB 提供程序	66
从 ADO 连接	67
从其它数据库连接	68
避免在 UNIX 上发生端口号冲突	68
如何测试连接	69
集成登录	69
使用集成登录	70
安全性问题：不受限制的数据库访问	71
临时公共选项可以提高安全性	72
集成登录的网络方面	73
缺省集成登录用户	73
连接归集	73
临时连接	74
逻辑服务器配置	74
在 Simplex 中连接	74
Multiplex 中的连接	74
如何终止连接	75
连接记录	75
连接和通信参数	77
连接参数	77

AppInfo 连接参数 [Appinfo]	78
AutoPreCommit 连接参数 [AutoPreCommit]	80
AutoStart 连接参数 [Astart]	80
AutoStop 连接参数 [Astop]	81
CharSet 连接参数 [CS]	81
CompressionThreshold 连接参数 [COMPTH]	82
CommBufferSize 连接参数 [CBSize]	82
CommBufferSpace 连接参数 [CBSpace]	83
CommLinks 连接参数 [Links]	84
ConnectionName 连接参数 [CON]	85
ConnectionPool 连接参数 [CPOOL]	85
PrefetchBuffer 连接参数 [PBUF]	86
DatabaseFile 连接参数 [DBF]	86
DatabaseName 连接参数 [DBN]	87
DatabaseSwitches 连接参数 [DBS]	88
DataSourceName 连接参数 [DSN]	89
DBKEY 连接参数 [DBKEY]	90
DisableMultiRowFetch 连接参数 [DMRF]	90
EngineName 连接参数 [ENG]	90
EncryptedPassword 连接参数 [ENP]	91
Encryption 连接参数 [ENC]	92
Escape 连接参数 [ESCAPE]	93
FileDataSourceName 连接参数 [FileDSN]	94
Idle 连接参数 [IDLE]	94
Integrated 连接参数 [INT]	95
Language 连接参数 [LANG]	96
LazyClose 连接参数 [LCLOSE]	96
LivenessTimeout 连接参数 [LTO]	97
ConnectionPool 连接参数 [CPOOL]	98
Logfile 连接参数 [LOG]	98
NewPassword 连接参数 [NEWPWD]	99
Password 连接参数 [PWD]	99
PrefetchRows 连接参数 [PROWS]	99

RetryConnectionTimeout 连接参数	
[RetryConnTO]	100
ServerName 连接参数 [ENG]	100
StartLine 连接参数 [START]	101
Unconditional 连接参数 [UNC]	101
Userid 连接参数 [UID]	102
网络通信参数	102
Broadcast 通信参数 [BCAST]	103
BroadcastListener 通信参数 [BLISTENER]	104
Certificate 通信参数	105
Certificate_Password 通信参数	105
ClientPort 通信参数 [CPort]	106
DatabaseName 通信参数 [DBN]	107
DoBroadcast 通信参数 [DBROAD]	107
Host 通信参数 [IP]	108
LDAP 通信参数 [LDAP]	110
LocalOnly 通信参数 [LOCAL]	110
LogFile 通信参数 [LOG]	111
LogFormat 通信参数 [LF]	111
LogMaxSize 通信参数 [LSIZE]	112
LogOptions 通信参数 [LOPT]	113
MaxConnections 通信参数 [MAXCONN]	114
MaxRequestSize 通信参数 [MAXSIZE]	114
MyIP 通信参数 [ME]	114
PreFetchOnOpen 通信参数	115
ReceiveBufferSize 通信参数 [RCVBUFSZ]	116
SendBufferSize 通信参数 [SNDBUFSZ]	116
ServerPort 通信参数 [PORT]	117
Sessions 通信参数	118
TDS 通信参数	119
Timeout 通信参数 [TO]	119
VerifyServerName 通信参数 [Verify]	120
数据库对象管理	123
Sybase IQ 数据库设计	123

使用数据库对象的工具	123
Sybase Central	124
Interactive SQL	124
设置 Sybase IQ 数据库	125
调度数据定义任务准则	125
用于性能监控的 Dummy 表	126
数据定义的权限	126
设备选择	126
空间分配	126
范围分区	133
对分区执行 DDL 操作的限制	135
对分区执行 DML 操作	135
未分区的表的对象放置	135
分区表的对象放置	136
数据库定义	137
使用 SQL 创建数据库	137
在 Sybase Central 中设置数据库选项	145
在 Interactive SQL 中设置数据库选项	145
IQ 主存储区和 IQ 临时存储空间管理	146
显示 Sybase Central 中的系统对象	147
显示 Interactive SQL 中的系统对象	147
在 Sybase Central 中与数据库断开连接	147
在 Interactive SQL 中与数据库断开所有连接	147
在嵌入式 SQL 中与数据库断开连接	148
使其他用户与数据库断开连接	148
删除数据库	148
数据存储	149
dbfile 属性和操作	149
数据库空间属性和操作	149
只读和读写数据库空间和文件	150
数据库空间重命名准则	162
其它数据库空间	162
删除数据库空间的准则	163
数据库空间管理示例	164

表管理	170
创建表的准则	171
更改表的准则	176
删除表的准则	177
创建主键	177
创建外键	178
系统视图中的表信息	179
视图管理	179
创建视图	180
使用视图的准则	180
修改视图的准则	181
针对视图的权限	182
如何删除视图	182
系统视图中的视图信息	182
索引管理	183
索引简介	183
创建索引	183
系统视图中的索引信息	183
索引验证	184
如何重命名索引	185
如何删除索引	185
Sybase IQ 索引	187
索引概述	187
Sybase IQ 索引类型	187
相较于传统索引的优点	190
创建 Sybase IQ 索引的工具	190
Interactive SQL 索引创建	190
并行列索引创建	191
选择索引的标准	191
索引中的唯一值数目	192
查询类型	192
磁盘空间使用情况	194
索引中的数据类型	194
组合索引类型	196

Sybase IQ 索引类型	196
快速投影 (FP) 缺省索引类型	196
Low_Fast (LF) 索引类型	200
High_Group (HG) 索引类型	201
High_Non_Group (HNG) 索引类型	203
比较 (CMP) 索引类型	204
包含 (WD) 索引类型	205
日期 (DATE)、时间 (TIME) 和日期时间 (DTTM) 索引类型	206
TEXT 索引	210
优化即席连接的性能	210
选择索引	210
在插入数据后添加列索引	211
使用连接索引	212
连接索引提高查询性能	212
连接索引的装载注意事项	212
如何使用连接索引进行查询	212
连接索引中的关系	212
当连接成为即席连接时	213
连接层次概述	213
连接索引中的列	214
查询解析中的连接层次	214
多表连接和性能	216
创建连接索引的步骤	216
同步连接索引	217
定义表之间的连接关系	218
发出 CREATE JOIN INDEX 语句	221
在 Sybase Central 中创建连接索引	221
连接层次类型	221
对修改连接索引表的限制	224
在连接索引表中执行插入或删除操作	225
表版本控制可控制对连接索引的访问	225
连接索引的大小和优势	226
如何估计连接索引大小	226

如何确定连接索引优势	226
数据导入和导出	227
导入和导出概述	227
导入和导出方法选择	227
输入和输出数据格式	228
修改数据的权限	228
调度数据库更新	229
从数据库导出数据的方法	229
输出重定向	229
数据抽取工具	230
使用 LOAD TABLE 语句执行 bulk load 操作	238
指定输入数据格式的装载	240
从客户端直接装载数据	242
分区表装载注意事项	242
装载和插入消息	243
完整性约束违规消息	243
二进制装载格式	246
IQ 二进制装载格式和装载效率	246
操作系统本机数据类型	246
DATE	247
TIME	247
TIMESTAMP	247
NUMERIC 和 DECIMAL	247
空值装载	248
使用 INSERT 语句	250
逐行插入指定值	251
插入数据库中的选定行	252
从其它数据库插入	252
交互式数据导入	254
在采用不同端格式的系统之间移动数据	254
插入到连接索引的表中	256
插入到主键列和外键列	256
装载或提取大对象数据	257
在插入时转换数据	257

Sybase IQ 15 版之前的版本中的数据	258
装载转换选项	258
列宽问题	262
更快的日期和时间装载	263
ASCII 输入转换	264
DATE 选项	265
DATETIME 转换选项	267
空值数据转换	269
影响数据显示的其它因素	270
匹配 Adaptive Server Enterprise 数据类型	270
不支持的 Adaptive Server Enterprise 数据类型	271
Adaptive Server Enterprise 等效数据类型	272
数据导入时的转换错误	276
调优数据的 bulk load 操作	276
定义数据库期间的装载性能	277
装载时间环境调整	278
增量装载中的 IQ 主存储区空间使用情况	278
装载期间的线程使用情况	279
对表行的更改	279
删除数据的方法	280
管理用户 ID 和权限	281
数据库权限概述	281
继承权限	282
继承的权限	283
权限概述	283
所有权权限概述	286
数据库空间管理权限	286
表和视图权限概述	287
组权限概述	287
服务器命令行权限选项	287
登录管理	288
用户帐户和连接管理	289
在失败的登录尝试后阻止连接	290

创建 DBA 恢复帐户	290
使用 DBA 恢复帐户登录	290
锁定用户	291
解锁用户	291
实用程序数据库服务器安全性	291
启动实用程序数据库	292
定义实用程序数据库口令	292
执行文件管理语句的权限	293
管理单个用户 ID 和权限	294
使用 Interactive SQL 创建用户	294
在 Sybase Central 中创建用户	294
在 Sybase Control Center 中创建用户	295
更改口令	295
向用户授予 DBA 权限	296
向用户授予 OPERATOR 权限	296
向用户授予 PERMS ADMIN 权限	296
向用户授予 RESOURCE 权限	296
授予 SPACE ADMIN 权限	297
向用户授予 USER ADMIN 权限	297
针对表和视图的权限	297
授予“授予权限”的权限	299
在 Interactive SQL 中授予针对过程的权限	300
在 Interactive SQL 中撤消用户权限	301
组管理	302
在 Interactive SQL 中创建组	302
在 Sybase Central 中创建组	303
在 Sybase Control Center 中创建组	303
在 Interactive SQL 中添加组成员	303
在 Sybase Central 中添加组成员	304
在 Sybase Control Center 中添加组成员	304
组的权限	304
组拥有的表	305
没有口令的组	305
自动创建的组	306

对象名的所有者前缀	306
视图和过程提供了更高的安全性	308
视图提供了定制的安全性	308
过程提供了定制的安全性	310
如何确定用户权限	311
连接使用的资源	312
控制用户资源的数据库选项	312
针对数据库连接的限制	313
使用过程禁用连接	313
系统对象中的用户和权限	313
系统表中的用户 ID、组和权限信息	313
系统视图中的用户 ID、组和权限信息	314
传输层安全性	314
IPv6 支持	315
数据完整性	317
数据完整性概述	317
哪些情况会导致数据无效	317
有效数据的规则和检查	317
更改数据库内容的语句	318
数据完整性工具	319
用于实施完整性约束的 SQL 语句	319
列缺省值有助于数据完整性	320
支持的缺省值	320
缺省值限制	320
创建列缺省值	321
更改列缺省值	321
删除列缺省值	321
支持的列缺省值	321
Sybase Central 中的列缺省值	323
日期、时间和时间戳缺省值	323
USER 缺省值	324
IDENTITY 或 AUTOINCREMENT 缺省值	324
NEWID 缺省值	325
空值缺省值	325

字符串和数字缺省值	325
常量表达式缺省值	325
表约束和列约束	325
对列或表的 UNIQUE 约束	326
对列的 IQ UNIQUE 约束	326
列的 CHECK 条件	327
用户定义数据类型的 CHECK 条件	327
在 Sybase Central 中添加、更改和删除列约束	328
列的 CHECK 条件	328
表的 CHECK 条件	329
使用 Interactive SQL 删除表的 CHECK 条件	329
实体和参照完整性	329
如何声明实体完整性	329
如何强制实施实体完整性	330
如果客户端应用程序破坏了实体完整性	330
参照完整性	330
参照完整性丢失	333
并发操作	334
如何禁用参照完整性检查	336
系统表中的完整性规则	336
事务和版本控制	337
事务和版本控制概述	337
事务	337
并发概述	339
版本控制概述	340
版本控制可防止不一致	346
锁定的工作方式	347
DML 操作锁	347
DDL 操作锁	348
主键和锁定	350
管理锁的工具	350
显示活动锁	350
调查锁争用的工具	350

隔离级别	351
检查点、保存点和事务回退	352
Checkpoints	352
事务内的保存点	353
事务回退	354
系统恢复	355
事务信息如何帮助进行恢复	356
快照版本控制的性能影响	356
重叠版本和删除	357
事务中的游标	358
游标和版本控制	358
游标敏感性	358
游标滚动	359
持有游标	359
定位操作	359
如何控制游标的消息记录	360
国际语言和字符集	361
缺省归类	362
了解软件中的字符集	362
Windows 中的代码页	362
归类序列如何对字符排序	363
了解区域设置	363
了解归类	364
显示归类	364
ANSI 和 OEM 代码页	364
ANSI ISO_1 归类	364
ANSI 1252LATIN1 归类	365
ANSI ISO1LATIN1 归类	365
ANSI ISO9LATIN1 归类	365
多字节归类	365
了解字符集转换	366
数据库消息的字符转换	366
连接字符串和字符集	367
避免字符集转换	368

配置字符集环境	368
区域设置信息	369
设置区域设置	369
语言标签值列表	370
为 INSERT LOCATION 语句设置区域设置	371
使用指定归类创建数据库	372
在数据库服务器中禁用字符集转换	372
将数据库从一种归类更改为另一种归类	372
兼容性问题	372
性能问题	373
数据备份、恢复和存档	375
数据保护	375
如何备份数据库	375
数据存储类型	376
备份类型	376
选择存档设备	379
准备备份	380
运行备份	382
指定是否存在操作员	382
指定备份类型	383
指定虚拟备份	383
指定存档设备	384
其它备份选项	386
等待磁带设备	387
使用只读硬件备份和恢复	387
备份示例	387
备份时从错误中恢复	388
完成备份之后	389
使用非 Sybase 产品执行备份	389
虚拟备份	390
虚拟备份类型	390
使用 SAN 快照或影子硬件执行虚拟备份	392
系统级备份	392
关闭数据库	392

备份正确的文件	392
从系统级备份恢复	394
验证数据库	394
恢复数据库	395
进行恢复之前	395
RESTORE 语句	398
按正确顺序进行恢复	402
在恢复后重新连接	403
在恢复后重命名事务日志	404
在恢复后验证数据库	404
恢复需要独占写入访问权限	405
显示标头信息	405
从恢复过程中发生的错误中恢复	406
检验数据库备份	406
备份与符号链接（仅适用于 UNIX）	408
获取备份和恢复的相关信息	409
定位备份日志	409
备份日志的内容	409
维护备份日志	410
记录数据库空间名称	411
确定数据备份和恢复策略	412
调度例行备份	412
指定备份和恢复责任	413
提高备份和恢复的性能	413
使用只读硬件存档数据	414
使用只读硬件	415
系统恢复与数据库修复	419
恢复和修复概述	419
常规恢复	419
数据库检验	420
sp_iqcheckdb 存储过程	420
sp_iqcheckdb 输出	423
运行 sp_iqcheckdb 的资源问题	426
数据库修复	427

索引错误分析	427
索引错误修复	430
分配问题分析	431
使用 DBCC 修复分配问题	433
强制恢复模式	435
强制恢复之前	435
以强制恢复模式启动服务器	435
恢复泄漏的空间	436
恢复 Multiplex 数据库	438
DBCC 报告的问题	438
DBCC 无法修复的索引问题	439
删除不一致的索引、表或列	439
DBCC 错误消息	440
故障排除提示	443
在线支持来源	443
特定情况的解决方案	443
服务器恢复和数据库修复的决策流程	443
服务器操作问题	444
数据库连接问题	452
Interactive SQL (dbisql) 问题	454
资源问题	455
处理问题	463
性能问题	466
Sybase Central 问题	467
网络通信故障排除	467
确保正在使用兼容的协议	467
确保拥有最新的驱动程序	468
在重新启动之间关闭计算机	468
逐层诊断协议堆栈	468
测试 TCP/IP 协议堆栈	468
诊断布线问题	469
检查常见网络通信问题	470
诊断工具	470
恢复至新临时文件拓扑	470

sp_iqstatus 存储过程	471
解释通知消息	473
sp_iqcheckdb 存储过程	478
检查数据库和服务器启动选项值	479
查找当前执行的语句	479
记录服务器请求	479
用于收集诊断信息的连接	482
诊断通信问题	482
向技术支持报告问题	482
使用 getiqinfo 收集诊断信息	482
由 getiqinfo 收集的信息	484
使 .srvlog 和 .iqmsg 文件的连接信息相关联	485
另一个有用的信息源	486
清单：技术支持信息	486
索引	489

Sybase IQ 系统管理概述

了解 Sybase® IQ 设计和功能是进行系统管理的基础。

读者

Sybase IQ 系统管理员管理数据库设置、连接、装载、安全性、备份和故障排除。

有关管理 Sybase IQ 系统的主题的目标读者是数据库管理员以及需要了解数据库创建、装载和操作问题的其他人员。

Sybase IQ 设计和用途

Sybase IQ 是专为数据仓库应用而设计的高性能决策支持服务器。

这个跨平台产品可以在多个常见 Unix、Linux 和 Windows 平台上运行。

Sybase IQ 是 Adaptive Server® 系列的一部分，该系列包含面向企业事务环境和混合工作负载环境的 *Adaptive Server® Enterprise*，此外，它还包含作为 Adaptive Server 袖珍版的 *SQL Anywhere®*，该版本通常用于移动计算，偶尔也用于连接计算。

Sybase 数据库体系结构

Sybase 数据库体系结构通过优化了工作负载的数据存储为 Sybase IQ 和 SQL Anywhere 提供公用基础代码。

您可以使用 IQ 存储进行数据仓储。这些产品共享公用的命令语法和用户界面，从而简化了应用程序开发和用户访问过程。

快速访问众多数据源

Sybase IQ 可以集成来自多种源的数据，源数据不仅来自 IQ 数据库，还可来自 SQL Anywhere 数据库和 ASE 数据库，以及非 Sybase 数据库和平面文件。

您可以将此数据导入 IQ 数据库，以便利用 IQ 的快速访问功能。您也可以使用 Sybase IQ 的远程数据访问功能，直接查询其它数据库。

数据仓库应用和 Sybase IQ

*数据仓库*是一些数据集，其设计目的是供业务分析员分析信息。

通常，数据仓库不同于生产数据库，以避免干扰日常运作。数据仓库常常用作建立决策支持系统 (DSS) 的数据存储。*决策支持系统*是一个软件应用程序，其设计目的是供组织分析数据以支持业务决策。

所有 Sybase IQ 功能的设计目的都是为 DSS 应用程序提供便利。独有的索引系统加速了数据分析。即使结果包含数千或数百万行数据，查询优化也可以为您提供快速响应。适用多个查询用户的并发数据访问，以及不中断查询处理便可更新数据库的功能，可以为用户提供他们所期望的全天候访问。

Sybase IQ *Multiplex* 是一种可高度扩展的共享磁盘网格技术，该技术允许通过连接到共享数据源的独立数据处理节点来并发执行数据装载和查询操作。有关详细信息和语法，请参见《使用 Sybase IQ Multiplex》。

系统管理工具

用于管理 Sybase IQ 数据库的工具具有 Sybase Central、Sybase Control Center 和 Interactive SQL。

Sybase Central 是用于管理数据库对象的图形应用程序。可以使用 Sybase Central 执行常见的管理任务，例如，创建数据库、备份数据库、添加用户、添加表和索引、启用数据库的 Multiplex 功能以及监控数据库性能。Sybase Central 有基于 Java 的图形用户界面，可用于任何支持图形工具的操作系统。

Interactive SQL 是一个实用程序，您可以使用它来通过交互方式输入 SQL 语句并将它们发送到数据库或将它们作为脚本来执行。Interactive SQL 有两种形式：基于 Java 的 **dbisql** 和 Open Client 实用程序 **iqisql**。

《Sybase IQ 简介》介绍如何使用 Sybase Central 和 Interactive SQ 来执行简单的管理任务。如果您还不熟悉这些工具，则应该阅读《Sybase IQ 简介》以了解它们并使用其中提供的教程。

Sybase IQ 的 Sybase Control Center 是基于 Web 的工具，用于管理和监控 Sybase IQ 单节点和 Multiplex 服务器。该工具的两个主要功能是管理和监控。

Sybase Control Center 体系结构允许少量 Sybase Control Center 服务器使用 Sybase Control Center 代理来监控企业中的所有 Sybase IQ 服务器。

Sybase Control Center 代理随每个 Sybase IQ 服务器一起安装。

请参见《Sybase Control Center 3.2.0》> “Sybase IQ 的 Sybase Control Center”。

除了这些工具外，Sybase IQ 还提供了许多执行系统管理功能的存储过程。您也可以创建自己的过程和批处理。您可能希望利用 IQ 的事件处理功能来开发自己的系统管理工具。

请参见《系统管理指南第二卷》> “使用日程表和事件自动完成任务”。

选择归类之类的若干管理任务依赖于《实用程序指南》中描述的命令行实用程序。

另请参见

- 存储过程（第 5 页）

数据库服务器

数据库服务器是 Sybase IQ 系统的“大脑”。

通过数据库服务器访问数据，从不直接访问。数据库服务器接收用户对信息的请求，然后连接到指定的数据库并执行指令。

Sybase IQ 中的数据存储

表空间、数据库空间、dbfile 和存储是 Sybase IQ 存储数据的单位。

表空间是数据库中的存储单位，可作为总存储的逻辑子集进行管理。可以将各个对象和子对象分配到各个表空间。每个数据库均包括多个表空间。

在 Sybase IQ15.4 中，*dbspace* 是由一个或多个操作系统文件组成的表空间。

该术语的含义因所使用的产品版本而异。Sybase IQ 12.7 在 *dbspace* 和数据库文件之间实现了一对一映射。在 `DEFAULT_DISK_STRIPING` 选项为“ON”的情况下，Sybase IQ 会自动在所有可用 *dbspace* 中分发数据，以获得最佳性能并简化管理。用户无法指定应在哪个 *dbspace* 中创建表或索引。

dbfile 一词连同相应的逻辑文件名和物理文件路径指的是每个操作系统文件。

每个物理文件路径均必须唯一。dbfile 名可以与数据库空间名称相同。

存储是一个或多个 *dbspace*，用来存储永久或临时数据以用于特殊目的。Sybase IQ 有三种存储：

- *Catalog* 存储包含 SYSTEM 数据库空间，还可包含最多十二个附加 Catalog 数据库空间。
- *IQ 主存储区*包含 IQ_SYSTEM_MAIN 数据库空间和其它用户定义的数据库空间。
- *IQ 临时存储*包含 IQ_SYSTEM_TEMP 数据库空间。

另请参见

- 数据库空间类型（第 127 页）

SQL Anywhere 和 Sybase IQ

SQL Anywhere 是可与 IQ 共存或独立于 IQ 存在的关系数据库系统。

Catalog 存储和 SQL Anywhere 存储非常相似。除 IQ 表以外，Catalog 存储中还可以有 SQL Anywhere 样式的表，或者您也可以有单独的 SQL Anywhere 数据库。

SQL Anywhere 表的格式不同于 IQ 表。虽然在 SQL Anywhere 数据库中创建对象时所使用的命令可能与用于 IQ 存储的命令相同，但您可以在这些命令中指定的功能会有

所不同。请始终使用本书或《参考：语句和选项》中的命令语法来执行 IQ 存储中的操作。

本指南介绍如何管理 IQ 存储及其关联的 Catalog 存储。如果您有 SQL Anywhere 数据库，或 Catalog 存储中有 SQL Anywhere 样式的表，请参见 SQL Anywhere 文档以了解有关如何创建、维护和使用这些数据库和表的详细信息。

安全概述

Sybase IQ 通过您可以分配给用户的特权来提供安全性控制。

数据库管理员 (DBA) 负责维护数据库的安全。

用户类型

Sybase IQ 可以识别每个 Sybase IQ 数据库的三类用户：DBA、所有者和公共用户。

数据库管理员 (DBA) 拥有 对该数据库执行所有操作的完整权限。DBA 通常执行 大多数管理任务。

创建特定数据库对象的用户是该对象的 *所有者*，可以对该对象执行任何操作。

所有其他用户都被视为 *公共用户*。某个对象的所有者将被视为其他用户拥有的对象的公共用户。

任务权限

除了可以执行任何任务的 DBA 以外，用户必须获得权限才能执行特定任务。

例如，您需要适当的权限才能：

- 连接到数据库
- 创建数据库对象，例如数据库、表、索引 或外键
- 将对象放在特定的 dbspace 上
- 更改数据库对象的结构
- 插入、更新或删除数据
- 选择（查看）数据
- 执行过程

另请参见

- 管理用户 ID 和权限（第 281 页）
- 修改数据的权限（第 228 页）
- 事务和版本控制（第 337 页）

并发操作

Sybase IQ 允许多个用户同时查询某个数据库，与此同时，别的用户还可以插入或删除数据，或者备份该数据库。

更改数据库结构的操作，如创建、删除或改变表会暂时禁止其他用户使用这些表，但只从其它地方访问这些表的查询操作可以继续。

Sybase IQ 会通过维护多个版本的表数据，保持数据库在这些并发操作期间的一致性。

Multiplex 功能

Multiplex 是 Sybase IQ 中的一项强大功能，通过该功能，您可以使用群集系统中的多台计算机来装入和打开驻留在共享磁盘存储上的单个 IQ 数据库，从而对应用程序进行伸缩。

有关详细信息，请参见《使用 Sybase IQ Multiplex》。

另请参见

- CREATE DATABASE 语句缺省值（第 140 页）
- 使用 SQL 创建数据库（第 137 页）
- 相对路径名（第 141 页）
- 实用程序数据库服务器安全性（第 291 页）

存储过程

Sybase IQ *存储过程*为您提供有关数据库和用户的信息，并对数据库执行各种操作。

存储过程通常运行于执行该存储过程的数据库上。例如，如果您在 `iqdemo` 数据库中运行存储过程 `sp_addlogin`，它会将用户添加到 `iqdemo`。

注意： 示例中显示的语句一般使用 `iqdemo` 数据库（一个可以使用随 Sybase IQ 一起安装的脚本来创建的演示数据库）。

您也可以创建自己的存储过程。

Sybase IQ 存储过程

Sybase IQ 提供了许多用来查看和管理信息的存储过程。

例如，`sp_iqstatus` 显示常规状态信息，而 `sp_iqcheckdb` 检查当前数据库的有效性并且可以修复分配问题。

请参见《系统管理指南第二卷》>“系统过程”。

还有几个 **Multiplex** 存储过程也可使用。请参见《使用 Sybase IQ Multiplex》> “**Multiplex 引用**” > “**系统过程**”。

目录存储过程

除了大多数 **Adaptive Server Enterprise** 目录存储过程外，还有其它系统和目录存储过程。

有关完整列表，请参见《参考：构件块、表和过程》> “**系统过程**”。

Sybase IQ 不支持 **Adaptive Server Enterprise** 目录存储过程 **sp_column_privileges**、**sp_databases**、**sp_datatype_info** 和 **sp_server_info**。

另请参见

- 系统管理工具（第 2 页）

系统表和视图

Sybase IQ 系统表和视图是包含数据库服务器管理 Sybase IQ 系统所需的所有信息的对象。系统视图以更便于阅读的格式显示其相应的系统表中的信息。

系统表位于 **Catalog** 存储中，有时也称为 **catalog** 表。只能通过系统视图来访问系统表。**SYS** 用户 ID 拥有系统表。

系统表中包括以下信息：

- 数据库特性
- 表特性，包括表定义和有关各表大小和位置的信息
- 有关索引的信息
- 数据库和 **Interactive SQL** 选项的当前设置

有关系统表和视图及其内容的完整说明，请参见《参考：构件块、表和过程》。

命令和函数

所有 Sybase IQ 命令都是 SQL 语句。SQL 表示结构化查询语言，一种常用于数据库应用程序的语言。

Sybase IQ SQL 使用与 SQL Anywhere SQL 相同的语法；唯一的不同之处在于某些产品功能要么仅受 Sybase IQ 支持，要么仅受 SQL Anywhere 支持。Sybase IQ SQL 也提供与 Transact-SQL（**Adaptive Server Enterprise** 使用的 SQL 术语）的高度兼容性。

本节介绍您可以使用的语句和函数的类型。本书其它各章将向您介绍用于执行各种管理任务的语句。

有关支持的函数和语句的完整详细信息，请参见《参考：构件块、表和过程》以及《参考：语句和选项》。

SQL 语句的类型

SQL 语句包含三种类型：数据定义语言 (DDL)、数据操作语言 (DML) 和程序控制语句。

DDL (数据定义语言) 语句供您定义和修改数据库 schema 以及表和索引定义。DDL 语句示例包括 **CREATE TABLE**、**CREATE INDEX**、**ALTER TABLE** 和 **DROP**。

DML (数据操作语言) 语句供您查询数据以及将数据移入和移出数据库。DML 语句示例包括 **SELECT**、**SET** 和 **INSERT**。

程序控制语句控制程序执行流。它们不会直接在您的 Sybase IQ 表上操作。示例包括：**IF**、**CALL** 和 **ROLLBACK**。

函数

函数返回数据库中的信息。

在允许使用表达式的任何地方都允许使用函数。Sybase IQ 提供了实现以下操作的函数：

- 集合数据 (例如, **AVG**、**COUNT**、**MAX**、**MIN**、**SUM**、**STDDEV**、**VARIANCE**)
- 处理数值数据 (例如, **ABS**、**CEILING**、**SQRT**、**TRUNCATE**)
- 处理字符串数据 (例如, **LENGTH**、**SOUNDEX**、**UCASE**)
- 处理日期和时间数据 (例如, **TODAY**、**DATEDIFF**、**DATEPART**、**MINUTES**)
- 将检索的数据从一种格式转换为另一种格式 (**CAST**、**CONVERT**)
- 处理分析数据 (例如, **DENSE_RANK**、**NTILE**、**PERCENT_RANK**、**PERCENTILE_CONT**、**PERCENTILE_DISC**、**RANK**)

消息日志内容

每个数据库都有一个 IQ 消息日志文件，用于从服务器捕获有关操作的消息。

消息日志文件具有缺省名 `dbname.iqmsg`，并且是在启动新创建的数据库时在 Catalog 存储所在的目录中创建的。数据库创建者可以指定其它位置和/或其它文件名。

缺省情况下，Sybase IQ 在消息日志文件中记录以下类型的消息：

- 错误消息
- 状态消息
- 插入通知消息
- 查询计划

您可以像查看任何其它文本文件一样查看此文件。

在此文件开头以及启动数据库时，会看到类似如下的输出：

```
I. 09/06 17:20:57. 0000000000 OpenDatabase Completed
I. 09/06 17:20:57. 0000000000 IQ cmd line srv opts: -iqmc 64 -iqtc
```

```

64
I. 09/06 17:20:57. 0000000000 IQ full cmd line: -n system1_demo -c
48m -gd all -gm 10
-gp 4096 -x tcpip{port=9030} -iqmc 64 -iqtc 64 -gc 20 -gl all -ti
4400 -gn 25
-o /system1/users/user1/sybase/IQ-15_4/logfiles/system1.0001.srvlog
-hn 5
I. 09/06 17:20:57. 0000000000 DB: r/w, Main 63MB Buffs=510, Temp 63MB
Buffs=510,
Pgsz=131072/8192blksz/16bpc, BP=4
I. 09/06 17:20:57. 0000000000 DB: Frmt#: 23F/2T/1P (FF: 03/18/1999)
I. 09/06 17:20:57. 0000000000 DB: Versn: 15.4.0.6556/110905/P/
Mainline/Sun_x64/OS 5.10
/64bit/2011-09-05 23:25:52
I. 09/06 17:20:57. 0000000000 DB: Name: /system1/users/user1/
iqdemo.db
I. 09/06 17:20:57. 0000000000 DB: Txn ID Seq: 1
I. 09/06 17:20:57. 0000000000 DB: DBID Blk: 2610
I. 09/06 17:20:57. 0000000000 DB: IQ Server system1_demo, PID 11033,
LOGIN user1
I. 09/06 17:20:57. 0000000000 DB: Database encryption is OFF.
I. 09/06 17:20:57. 0000000000 Mem: 160mb/M163
Main Blks: U2721/21%, Buffers: U6/L1
Temporary Blks: U65/2%, Buffers: U4/L0
Main I: L24/P4 O: C2/D2/P0 D:0 C:100.0
Temporary I: L34/P0 O: C4/D4/P0 D:0 C: 0.0
I. 09/06 17:20:57. 0000000000 Collation ISO_BINENG, Case Ignore,
Blank Padding On, Comparisons are Conditioned
I. 09/06 17:20:57. 0000000000 RcvyCmpl
I. 09/06 17:20:57. 0000000000 Chk
I. 09/06 17:20:57. 0000000000 ChkDone [NumTxnCP: 0]
I. 09/06 17:20:57. 0000000000 PostChk
I. 09/06 17:20:57. 0000000000 CloseDatabase
I. 09/06 17:20:57. 0000000000 Warning: Using default value for -iqmc
or -iqtc is
insufficient for most applications
    
```

消息日志中的版本信息

版本信息显示在靠近消息日志文件开头的地方。

版本信息类似于以下信息：

```

Versn:
15.4.0.6556/100301/P/GA/Sun_Opteron/OS 5.10/64bit/2011-09-06
01:21:40
    
```

表 1. 版本字符串元素

版本字符串元素	示例	注释
主版本	15.4	前两段表示主版本
次要版本	15.4. <i>n</i>	第三段表示次要版本

版本字符串元素	示例	注释
内部版本号	6556	供内部使用的唯一内部版本号
内部版本日期	100301	供内部使用的日期标识符
内部版本类型	P	P = 生产版本
版本类型	GA	GA = 一般可用性; ESD = 工程软件分发 (维护)
硬件平台	Sun_Opteron	硬件提供商的标识符
操作系统版本	OS 5.10	OS 版本和位模式指构建软件所在的系统, 而不是指其当前运行所在的系统。
位模式	64 位	
构建日期和时间	2011-09-06 01:21:40	显示为 YYYY-MM-DD hh:mm:ss (ISO 日期时间格式) : 表示年份的 4 位数 表示月份的 2 位数 (0 - 12) 表示月份中日期的 2 位数 (01 - 31) 表示自午夜开始过去的完整小时数的 2 位数 (00 - 23) 表示自小时开始过去的完整分钟数的 2 位小数 (00 - 59) 表示自分钟开始过去的完整秒数的 2 位数 (00 - 59)

消息日志中的归类信息

当您启动数据库时, 消息日志会捕获归类信息。

消息日志中捕获的归类信息显示归类名称、区分大小写、空白填充状态以及字符转换表要求。Comparisons are conditioned 表示比较字符数据时无需使用转换表。

消息日志中的连接信息

在第一个事务开始后、下一个事务开始前, 消息日志会捕获连接信息。

每个数据库连接都会在消息日志中输出一行连接行, 其中显示连接句柄和数据库用户名。

例如:

```
I. 04/23 13:02:37. 0000000003 Connect: SA connHandle: 1 SA connID:
3 IQ connID: 0000000003 User: DBA
```

您可以使用此连接信息匹配带查询文本的 .iqmsg 文件中的查询计划。运行存储过程 **sp_iqcontext** 可以确定每个连接在给定时刻执行的语句。

在消息日志中插入通知

消息日志会捕获插入和装载操作中的通知消息。

可以在 **LOAD** 和 **INSERT** 语句中使用参数关闭通知消息。

另请参见

- 连接索引中的列 (第 214 页)
- 解释通知消息 (第 473 页)
- 装载和插入消息 (第 243 页)
- 消息日志回卷 (第 10 页)

消息日志文件管理

缺省情况下，消息日志文件会增加到无限大小，而且将一直存在，直到数据库被删除。要控制消息日志文件的大小，可以设置文件大小限制并启用消息日志回卷或日志存档。

在停止数据库之后，可以随时删除、重命名或复制消息文件。

消息日志管理由服务器属性 **IQMsgMaxSize** 和 **IQMsgNumFiles** 或者服务器启动开关 **-iqmsgsz** 和 **-iqmsgnum** 控制：

- **IQMsgMaxSize** 或 **-iqmsgsz** 可设置活动消息日志大小的上限（以兆字节 [MB] 为单位）。允许的值为 0 到 2047（包括 0 和 2047）之间的整数。缺省值为 0，表示对消息日志文件的大小无限制。
- **IQMsgNumFiles** 或 **-iqmsgnum** 可设置消息日志存档数。允许的值为 0 到 64（包括 0 和 2047）之间的整数。缺省值为 0，表示消息将在主消息日志文件中回卷，而且没有存档。

服务器属性值的优先级高于相应服务器启动开关的值。服务器启动时，系统会将 **-iqmsgsz** 和 **-iqmsgnum** 服务器开关的值写入服务器日志文件。

有关设置 **IQMsgMaxSize** 和 **IQMsgNumFiles** 服务器属性的信息，请参见《参考：构件块、表和过程》>“系统过程”>“sa_server_option 系统过程”。

有关设置 **-iqmsgsz** 和 **-iqmsgnum** 服务器启动开关的信息，请参见《实用程序指南》>“start_iq 数据库服务器启动实用程序”。

注意： 不建议使用 **IQMSG_LENGTH_MB** 数据库选项；请将它从现有的脚本和代码中删除。尝试使用 **IQMSG_LENGTH_MB** 将返回错误。

消息日志回卷

消息日志回卷在达到最大日志大小后，将用新消息覆盖文件开头位置的现有消息。

如果启用消息日志回卷，则一旦日志文件达到 **IQMsgMaxSize** 服务器属性或 **-iqmsgsz** 服务器启动开关中指定的最大大小，便会在该文件开头位置写入新消息。现有消息会逐行被覆盖。

可以通过以下方法来启用消息日志回卷：将 **IQMsgMaxSize** 或 **-iqmsgsz** 设置为大于零的值（缺省值为 0，表示对于消息日志文件的大小没有限制），将 **IQMsgNumFiles** 服务器属性或 **-iqmsgnum** 服务器启动开关设置为零（缺省值）。

启用回卷后，消息日志文件中的 `<next msg insertion place>` 标记会告诉您新消息将放在哪里。在文件的开头和末尾的其它标记会提醒您日志回卷已启用，并且文件中的最后一条消息可能不是最新的消息。

另请参见

- 连接索引中的列（第 214 页）
- 解释通知消息（第 473 页）
- 装载和插入消息（第 243 页）
- 消息日志中的版本信息（第 8 页）

消息日志存档

消息日志存档在维护写入消息日志文件的所有信息时会限制文件大小。

消息日志存档是一个文件，用来保存处于活动状态的 `.iqmsg` 消息日志文件的内容。

可以通过以下方法来启用消息日志存档：使用 **sa_server_option** 系统存储过程将 **IQMsgMaxSize** 服务器属性或 **-iqmsgsz** 服务器启动开关设置为大于零的值，将 **IQMsgNumFiles** 服务器属性或 **-iqmsgnum** 服务器启动开关设置为消息日志存档的数量。服务器属性值的优先级高于相应服务器开关的值。

消息日志存档名称

`.iqmsg` 消息日志存档的名称采用一种模式。

`.iqmsg` 消息日志存档名称的模式为 `logname.iqmsg.n`，其中 *n* 是大于零的整数，`logname` 是在 **CREATE DATABASE** 语句中指定的消息路径的名称，或者是带有 `.iqmsg` 后缀的数据库名称。存档是在消息日志文件所在的目录中创建的只读文件。

当消息日志文件 `logname.iqmsg` 已满，而且消息日志存档的数量小于在 **IQMsgNumFiles** 或 **-iqmsgnum** 中指定的数字时，服务器会重命名当前的消息日志以创建新存档。系统会创建一个名为 `logname.iqmsg` 的新消息日志。

例如，如果 **-iqmsgnum** 等于 5，则系统会按以下顺序创建消息日志存档：`logname.iqmsg.1`、`logname.iqmsg.2`、`logname.iqmsg.3`、`logname.iqmsg.4` 和 `logname.iqmsg.5`。

当消息日志文件 `logname.iqmsg` 已满，而且消息日志存档已经达到最大数量时，系统会删除最旧的存档（文件扩展名为 `.1`），然后再对当前的消息日志进行存档。

例如，如果 **-iqmsgnum** 为 5 且 `logname.iqmsg.5` 已存在，则：

- 系统将删除 `logname.iqmsg.1` 文件。

- logname.iqmsg.2 至 logname.iqmsg.5 的各个文件将分别重命名为 logname.iqmsg.1 至 logname.iqmsg.4。
- 处于活动状态的消息日志文件将重命名为 logname.iqmsg.5。
- 系统将创建一个新的消息日志文件 logname.iqmsg。

使用此方法时，如果 **IQMsgNumFiles**（或 **-iqmsgnum**）的值大于零，服务器将始终保留最新的消息日志。

消息日志管理错误

如果磁盘在消息记录期间变满，则消息记录会停止。系统会在服务器日志中写入错误 "Disk Full!!! Message logging stopped."。

如果在消息日志管理期间出现以下错误，则消息记录会停止：

- 文件重命名错误：服务器无法对当前处于活动状态的日志进行存档或者无法重命名现有的任何日志存档
- 文件创建错误：服务器无法创建新的消息日志
- 文件删除错误：服务器无法删除旧存档
- 磁盘 I/O 错误

在错误情况得以解决之后，服务器会自动重新开始消息记录。当消息记录停止和记录重新开始时，会向服务器日志中写入一则消息。

例如，如果重命名文件失败，则系统会在服务器日志中写入消息 "Renaming of <filename> failed. Message log could not be archived. Message logging stopped."。当记录重新开始时，系统会在服务器日志中写入消息 "Message logging resumed."。

如果由于文件不存在而导致服务器无法删除或重命名文件，则会忽略该错误，并且日志管理进程继续执行。

何时备份消息日志和存档

即使在恢复时不需要消息日志文件 dbname.iqmsg 和消息日志存档，Sybase 也建议您为这些文件创建备份。

如果在恢复期间出现问题，则 .iqmsg 文件会包含证明数据库在启动备份前已关闭的信息。这些消息日志文件可用于诊断要从其中恢复的数据库故障是由什么原因导致的。请务必在恢复前创建副本，以供以后分析使用。

如果启用了 IQ 消息日志回卷 (**IQMsgMaxSize** 服务器选项或 **-iqmsgsz** 服务器启动开关不等于零，**IQMsgNumFiles** 服务器选项或 **-iqmsgnum** 服务器启动开关为零)，则您应备份 .iqmsg 文件，以便在需要用消息进行诊断时可访问所有消息。

如果启用了消息日志存档 (**IQMsgMaxSize** 服务器选项或 **-iqmsgsz** 服务器启动开关不等于零，**IQMsgNumFiles** 或 **-iqmsgnum** 服务器启动开关不等于零)，则服务器会自动备份消息日志存档。最多可存档 128 GB 的消息日志，在大多数情况下该大小已足够。

注意：在服务器重新启动之前，需要备份消息日志存档。服务器重新启动之后，系统将忽略现有日志存档，并在 `dbname.iqmsg` 文件已满时创建新的存档。如需保留旧存档日志，请在重新启动服务器之前备份这些文件。

夏令时和消息日志

如果在遵循夏令时的国家/地区运行 Sybase IQ，请在更改回标准时间之后重新启动 Sybase IQ 服务器。

通过执行重新启动操作，可更正 Sybase IQ 消息日志和通过对服务器启动指定 `-zr` 开关所生成的 SQL 日志之间的时间差异。在更改为夏令时之后，还需要重新启动 Sybase IQ 服务器。

实用程序数据库

`utility` 数据库从不保存数据，但系统会自动创建它来执行特定的管理任务。

数据库服务器在它需要连接数据库，但不存在实际数据库或没有应该运行的数据库时使用实用程序数据库。Sybase IQ 安装会自动创建实用程序数据库。

请确保不要删除此数据库。您需要它来执行以下任何操作：

- 在不指定任何数据库的情况下，使用 **START ENGINE** 命令启动数据库服务器
- 在没有其它可以连接的数据库的情况下，创建或删除一个数据库
- 在您拥有的任何其它数据库不可用（例如，由于介质故障的原因而不可用）的情况下，启动数据库服务器或连接到数据库
- 恢复数据库

缺省情况下，实用程序数据库具有用户 ID `dba` 和口令 `sql`。您可以在安装过程中将这些值更改为其它值，也可以在以后通过编辑可执行文件目录下 `util_db.ini` 文件中的连接参数将这些值更改为其它值。

另请参见

- 进行恢复之前（第 395 页）
- 控制权限的命令行选项（第 27 页）
- 移动数据库文件（第 399 页）
- 实用程序数据库服务器安全性（第 291 页）
- 检验数据库备份（第 406 页）

可伸缩性

Sybase IQ 的专利设计允许数据库扩展以容纳数千吉字节数据。

其基于索引的结构使 IQ 存储的数据大小比原始输入数据小得多，并且访问速度远快于传统的关系数据库。这些功能使得 Sybase IQ 很适合存储和访问非常大的数据库 (VLDB)。

数据库管理员需要了解影响性能的选项和功能，并遵守文档所述的准则。虽然很多缺省设置自动提供最高效率，但您可能需要根据配置、装载要求和查询来试验某些选项设置，以最快地获得结果。为了获得任何 Sybase IQ 数据库的最高性能，有必要恰当地设置这些选项，而且随着数据库增长到数千吉字节规模，这一点变得尤为重要。

本节介绍帮助您管理非常大的数据库的 Sybase IQ 功能，并指导您获取更为详细的讨论和建议。

内存使用

恰当分配内存是所有 Sybase IQ 数据库性能的关键因素。

Sybase IQ 在装载和查询的缓冲区高速缓存中使用内存。它也使用一些内存来管理连接、事务、缓冲区以及数据库对象。

Sybase IQ 有两个缓冲区高速缓存，一个用于 main 存储，另一个用于临时存储。这些高速缓存的缺省大小对于生产数据仓库来说是不够的。您必须调整它们来体现您的数据库和表的大小、装载和查询的混合以及可以影响可用内存量的其它因素（例如，操作系统和其它应用程序）。

数据装载

随着数据库的增长，正确管理数据装载变得至关重要。

以下功能确保装载可以扩展以满足您的需求：

- 缓冲区管理器分区以避免锁争用。缺省情况下，基于 CPU 数量的缓冲区分区已启用，并且可以通过设置服务器或数据库选项进行调整。
- 允许足够内存进行装载，不必分配系统上可用内存以外的内存。
- 保留释放保存点、提交和检查点操作期间使用的数据结构的空间。
请参见《参考：语句和选项》>“数据库选项”>“常规数据库选项”>“MAIN_RESERVED_DBSPACE_MB 选项”。

另请参见

- 调查锁争用的工具（第 350 页）
- 调优数据的 bulk load 操作（第 276 页）

处理线程

Sybase IQ 使用操作系统线程来处理查询和装载。

控制线程使用的缺省选项设置一般足以提供良好的性能。但在有些情况下，您可能需要更改这些设置。请参见《性能和调优指南》>“管理系统资源”>“进程线程模型”以了解 Sybase IQ 如何使用线程。请参见《性能和调优指南》>“管理系统资源”>“用于调优资源使用的选项”以设置控制线程使用的服务器选项。

磁盘空间

了解管理 Sybase IQ 系统的磁盘 I/O 的信息。

在管理 Sybase IQ 系统的磁盘 I/O 中，最重要的因素是：

- 拥有足够的查询和装载磁盘空间
- 高效使用磁盘空间，以使最快的 I/O 可用于支持大功率、多 CPU 系统的处理速度

sp_iqstatus 存储过程指示 IQ 主存储区和临时存储中使用的空间所占的百分比。如果没有足够的临时或主数据库空间可用于缓冲区或数据库空间分配请求，则进行请求的语句将回退。您可以创建一个基于计时器的事件来监视空间使用情况，从而帮助避免在执行非特权操作期间因空间不足而发生意外回退。

磁盘条带化是获取最佳 I/O 性能的重要方式。磁盘条带化在多个磁盘驱动器上随机分配数据。您可以利用操作系统或磁盘管理软件和硬件中的磁盘条带化功能，以及 Sybase IQ 的内部条带化功能。缺省情况下，Sybase IQ 磁盘条带化功能已启用。

另请参见

- 数据存储 (第 149 页)
- 数据库空间管理示例 (第 164 页)
- DDL 操作锁 (第 348 页)
- 临时表的版本控制 (第 346 页)
- 在 Interactive SQL 中与数据库断开所有连接 (第 147 页)

中间版本控制

管理较大数据库中的装载和查询的关键方面在于 Sybase IQ 的事务级版本控制。

特别是，Sybase IQ 提供了将事务回退到中间保存点的功能，以便在长事务未能完成时，您可以无需重复整个装载。

另请参见

- 事务和版本控制 (第 337 页)
- 表版本控制可控制对连接索引的访问 (第 225 页)

页大小

当您创建 Sybase IQ 数据库时，选择正确的 IQ 页大小特别重要。

对于非常大的数据库，需要 128KB 或更大的 IQ 页大小。

另请参见

- IQ PAGE SIZE 参数准则（第 143 页）

基于列的索引

Sybase IQ 的基于列的索引结构优化了在您感兴趣的属性上执行选择或计算的能力。

为了获得最佳性能，您需要适用于您的数据和查询的一组正确索引。数据库的每列上应该都应有可以影响性能的索引。

查询优化程序

Sybase IQ 查询优化程序评估每条查询，从各种处理选项中进行选择，生成提供最佳性能的查询计划。

优化程序针对每个 Sybase IQ 版本进行调优，为大多数查询和数据库（包括最大的）选择最佳计划。

schema 设计

Sybase IQ 通常更适合于使用数据仓库设计中常见的非规范化 schema。

在传统关系数据库中，规范化通过消除冗余和改善一致性来改进事务处理。在数据仓库中，尤其是在非常大的数据仓库中，非规范化可改进对大量数据进行查询时的处理性能。

UNION ALL 视图

装载行数很多的表可能耗费很长时间。

UNION ALL 视图是解决此问题的一种方法。Sybase IQ 允许您通过将数据分成若干独立的基表来对表进行分区（例如，按日期）。稍后再通过 UNION ALL 视图重新将它们合成一个逻辑整体。

UNION ALL 视图易于管理。例如，如果数据按月分区，您可以通过删除表并更新相应的 UNION ALL 视图定义，删除整个月的数据。您可以有年份、季度等的许多视图定义，而不必添加额外的日期范围谓词。

有关建立 UNION ALL 视图的信息以及优化引用这些视图的查询的建议，请参见《性能和调优指南》>“管理系统资源”>“使用 UNION ALL 视图 以便更快装载”。

Sybase IQ 启动

Sybase IQ 根据平台和管理工具，提供多种方法来启动和停止服务器和数据库。

启动数据库服务器的方法

运行 Sybase IQ 的第一步是启动数据库服务器。

您可以采用以下所有方式启动服务器：

- 使用 Sybase 提供的服务器启动实用程序 **start_iq** 启动服务器。
- 在 Sybase Central 中运行启动数据库服务器向导。请参见 Sybase Central IQ 插件的联机帮助。要以交互方式启动和停止 Multiplex 服务器，请始终使用 Sybase Central。
- 使用 Sybase Control Center。请参见《Sybase Control Center 3.2.0》>“Sybase IQ 的 Sybase Control Center”。
- 从 Windows “开始” 菜单启动服务器。
- 使用 Sybase 提供的配置文件启动服务器和 **iqdemo** 数据库。
- 将服务器启动命令放在快捷方式或桌面图标中。

注意： 还可以将 Windows 系统配置为在系统启动时自动启动 IQ 服务器（作为 Windows 服务）。有关详细信息，请参见《安装和配置指南》。

- 将服务器 **startline** 包含在 ODBC 数据源中。
- 将服务器 **startline** 包含在实用程序命令中。
- 从 Interactive SQL 发出 SQL 命令来启动另一个服务器。

注意： 如果将要使用远程数据访问功能插入其它数据库中的数据或发出对其它数据库的查询，请参见《系统管理指南第二卷》>“访问远程数据”和《系统管理指南第二卷》>“用于远程数据访问的服务器类”。

另请参见

- 如何启动 **iqdemo** 数据库（第 39 页）
- 国际语言和字符集（第 361 页）
- 服务器名（第 23 页）
- 从命令行创建 ODBC 数据源（第 59 页）
- 从 Interactive SQL 启动服务器（第 30 页）
- ODBC 数据源（第 57 页）
- 启动和停止数据库的方法（第 38 页）
- ODBC 数据源中的连接参数（第 44 页）
- JDBC 驱动程序（第 47 页）

- 从 Interactive SQL 连接到演示 数据库（第 49 页）

服务器启动实用程序

除了介绍的几个例外，启动实用程序 **start_iq** 可在所有平台上运行，并确保正确设置必需的参数。

在包含多个命令行选项的示例中，为清楚起见，我们将这些选项显示在不同的行上，与它们写在配置文件中的方式一致。如果直接在命令行中输入这些选项，则必须将它们全部输入在一行中（即无需任何回车符）。

您可以从多个命令行开关中进行选择，以指定诸如启动数据库或停止服务器所需的权限以及要使用的网络协议之类的功能。命令行选项是一种调优 Sybase IQ 行为和性能的方法。

另请参见

- 命令行开关（第 19 页）

同时启动数据库和数据库服务器

为便于使用，可通过在启动服务器时指定数据库名称来同时启动数据库和服务器。

缺省情况下服务器根据数据库名称获取其名称，您也可以为服务器指定其它名称。

另请参见

- 服务器和数据库名称（第 22 页）

启动数据库服务器而不启动数据库

要启动服务器而不启动任何数据库，请在 **start_iq** 命令中省略数据库文件并指定一个服务器名。

如果省略数据库名称，则必须使用 **-n** 服务器开关显式命名服务器。可在以下情况下使用此方法：创建或恢复数据库；仅希望控制服务器的启动和停止，让客户端软件来控制数据库的使用。

使用启动实用程序启动服务器

start_iq 命令将指定服务器作为后台进程启动（如果指定了数据库，则启动该指定数据库），并设置所有必需的启动选项。

1. 转到一个可写目录。
2. 在系统提示符处，输入：

```
start_iq servername [ database ]
```

如果不指定数据库，则必须使用 **-n <server name>**，否则服务器将无法启动。在前一示例中，在缺省端口 2638 上启动服务器。

服务器日志

服务器启动后，该命令会向启动服务器的窗口或主控台发送一条消息，以指示服务器正在运行。

服务器还会显示有关服务器环境的其它信息，以及有关启动失败的“可能的问题”消息。

所有服务器消息都会写入到服务器日志中。缺省情况下，在 Windows 平台上安装时，系统将设置 %IQLLOGDIR15%，并且服务器日志位于 %IQLLOGDIR15%\servername.nnnn.srvlog 中，其中 nnnn 为服务器已启动的次数。还可以使用 -o 启动选项来命名服务器日志。

从 Windows “开始” 菜单启动服务器

启动数据库服务器的某些方法是特定于 Windows 系统的。还可以使用任何通用方法来启动数据库服务器。

在 Windows 中启动服务器的最简单方法是从“开始”菜单来启动。选择“程序” > “Sybase” > “Sybase IQ15.4”。

您可以从此处启动 Sybase IQ 演示数据库、Sybase Central 和 Interactive SQL。

还可以将自己的数据库放到“程序”组中。

服务器作为 Windows 服务

您可以将服务器作为 Windows 中的服务来运行。

这样，即使在您注销计算机的情况下，它也能继续运行。有关此功能及其它特定于 Windows 的功能的详细信息，请参见《安装和配置指南》。

自动服务器启动

使用 Sybase IQ Service Manager 可定义将启动 IQ 服务器的服务。

然后，可以将该服务配置为在主机启动时自动启动服务器。该服务可以启动非 Multiplex 服务器或 Multiplex 服务器。

命令行开关

命令行开关用于定义 Sybase IQ 环境。

有关命令行开关的完整列表和完整参考信息，请参见《实用程序指南》> “start_iq 数据库服务器启动实用程序”。

您可以通过命令行选项设置的某些值也可以通过 **SET OPTION** 命令进行更改。请参见《参考：语句和选项》>“数据库选项”以及《参考：语句和选项》>“SET OPTION 语句”。

另请参见

- 服务器启动实用程序（第 18 页）

显示命令行选项

命令行开关是区分大小写的。

要显示所有可用的命令行开关，请在操作系统提示符处输入以下命令：

```
start_iq -h
```

配置文件

命令行开关集合可以存储在配置文件中。

如果使用大量命令行开关，则可将它们存储在配置文件中，并在服务器命令行中调用该文件。在配置文件中指定开关如同在命令行中指定开关，不同的是您可以在多行输入开关。

配置文件列表

Sybase IQ 提供了下列配置文件：

表 2. 配置文件

文件名	位置	使用
default.cfg	\$IQDIR15/ scripts (UNIX)、 %IQDIR15%\ scripts (Windows)	通用配置文件。此文件用于 start_iq 和 Multiplex 启动的缺省选项。Sybase IQ 将 default.cfg 复制到每个新的数据库目录，并将其重命名为 params.cfg。对 default.cfg（在脚本目录中）的更改会由在该文件更改后创建的所有数据库继承。
params.cfg	数据库目录	用户创建的数据库的配置文件（如果该数据库通过 Sybase Central 创建）。对该文件的更改仅会影响使用此特定文件的服务器。
iqdemo.cfg	\$IQDIR15/da- ta/demo (UNIX)、 %IQDIR15%\da- ta\demo (Windows)	启动演示数据库，并将启动开关设置为建议的缺省值。

配置文件示例

您可以将这些文件用作模板来创建自己的文件。

例如，以下配置文件会在名为 `Elora` 的数据库服务器上启动数据库 `mydb.db`，并进行如下设置：高速缓存为 `32MB`，检查点时间间隔为 `20` 分钟，允许任何人启动或停止数据库以及装载数据，用户连接数限制为 `10`，目录页大小为 `4096` 字节，缺省客户端连接超时为 `72` 小时，使用 `TCP/IP` 作为网络协议并使用指定的端口号 `1870`：

```
-n Elora -c 32m -gc 20
-gd all -gl all -gm 10 -gp 4096 -ti 4400 -x tcpip(port=1870) path
\mydb.db
```

可以按如下方式使用这些命令行选项：

```
start_iq @mydb.cfg
```

注意： 当使用 `DBSTOP` 命令停止服务器时，您需要指定与启动服务器时所使用的参数相同的参数。使用配置文件启动服务器可确保在需要这些参数时能够找到它们。

Sybase IQ 文档中的命令行示例格式

为清楚起见，整个 Sybase IQ 文档中的示例都在不同的行上显示多个命令行开关，与它们写在配置文件中的方式一致。

如果直接在命令行中输入它们，则必须将它们全部输入在一行中。

必需的命令行选项

并非所有命令行开关都是可选的。

虽然《实用程序指南》>“运行数据库服务器”中介绍的大多数命令行开关都是可选的，但您必须指定 `-n` 开关才能有效运行 Sybase IQ。

注意： 在所有 32 位平台上，建议使用 `-c 32M`。在所有 64 位平台上，建议使用 `-c 48M`。

如果使用 `TCP/IP` 来连接服务器，则还应包含网络连接参数。如果启动服务器时未设置参数 `-x 'tcpip(port=nnnn)'`，则服务器会使用缺省的 `TCP/IP` 端口号 `2638`。如果指定的端口号已在使用，服务器将无法启动。

另请参见

- 无法连接到数据库（第 453 页）

缺省配置文件

缺省配置文件 (`default.cfg`) 包含所有必需的开关。

此文件由 Windows 服务和 Sybase Central 用来启动服务器，并且是 UNIX `start_iq` 命令使用的 `params.cfg` 文件的源文件。您可以通过在 `start_iq` 命令行中指定新的开关来覆盖配置文件中的开关，但 `-n servername` 开关除外。

演示数据库的配置文件

用于启动演示数据库的 `iqdemo.cfg` 文件可将启动开关设置为建议的缺省值。

当使用 Sybase IQ 提供的脚本创建演示数据库时，可以创建此文件。

命名限制

请勿将带连字符的名称或保留字用于数据库名称、用户标识符或服务器名，即使用引号引起来也不能使用。

例如，以下内容是不允许的：

```
grant
```

```
june-1999-prospects
```

```
“foreign”
```

有关保留字（关键字）的完整列表，请参见《参考：构件块、表和过程》>“SQL 语言元素”>“关键字”>“保留字”。

服务器和数据库名称

将 `-n` 命令行开关用作服务器开关（以命名服务器）。这样可防止无意间连接到错误的服务器。

服务器名和数据库名称是客户端应用程序连接数据库时可以使用的连接参数。在 Windows 中，服务器名显示在桌面图标上和服务器窗口的标题栏中。

注意：虽然您可以启动多个数据库，但 Sybase 强烈建议您在 IQ 服务器上仅运行一个数据库。如果必须运行多个数据库，请分别在不同的 IQ 数据库服务器和不同的端口上启动各数据库。

另请参见

- 同时启动数据库和数据库服务器（第 18 页）

缺省服务器名

如果未提供服务器名，则缺省服务器名为启动的第一个数据库的名称。

数据库名称

您可以通过在数据库文件后面提供一个 `-n` 开关来命名数据库。例如，以下命令行可启动数据库并对其进行命名：

```
start_iq -n MyServer mydb.db -n MyDB
```

通过对数据库进行命名，您就可以使用昵称替代可能难以记住的文件名。

服务器名

通过在第一个数据库文件之前提供一个 **-n** 开关，可以命名服务器。（其余的参数可从 `default.cfg` 文件进行添加。）例如，以下命令行可启动名为 `Cambridge_iqdemo` 的服务器以及该服务器上的 `iqdemo` 数据库：

```
start_iq -n Cambridge_iqdemo iqdemo.db
```

在局域网（域）内，每个服务器名都必须是唯一的。这样可防止无意间连接到错误的服务器。主机名合并端口号不能唯一标识服务器。在服务器名后面追加一个唯一标识符是一种有用的约定。这在多用户、联网环境中尤为重要，在这种环境中共享内存将用于本地数据库连接。此约定可确保所有用户都能够连接到正确的数据库，即使同名的其它数据库已在其它主机系统上启动也是如此。

要允许 Sybase IQ 查找服务器，而不管使用的是何种字符集，请在服务器名中仅包含七位 ASCII（较低页）字符。

通过指定服务器名，可在不装载任何数据库的情况下启动数据库服务器。下面的命令在不装载任何数据库的情况下启动名为 **Galt** 的服务器：

```
start_iq -n Galt -gm 10 -gp 4096
```

注意：虽然您可以通过依靠缺省服务器名启动服务器，但最好同时包含服务器名和数据库名称，并且使这两个名称不同。此方法有助于用户区分服务器和其上运行的数据库。您必须指定服务器名，才能在不启动特定数据库的情况下启动服务器。

另请参见

- 如何启动 `iqdemo` 数据库（第 39 页）
- 国际语言和字符集（第 361 页）
- 启动数据库服务器的方法（第 17 页）
- 从命令行创建 ODBC 数据源（第 59 页）
- 从 Interactive SQL 启动服务器（第 30 页）

区分大小写和命名约定

服务器名和数据库名称在 Windows 中是不区分大小写的，在 UNIX 中则区分大小写。

您应当为服务器和数据库以及所有其它数据库对象采用一套包含大小写规范的命名约定。强制命名约定可防止用户遇到问题。

用于性能的命令选项

有若干命令行选项可能会影响数据库服务器性能。

本节介绍的大多数选项都可控制用于 IQ 存储操作的资源，它们可能会对性能产生重要的影响。仅对可用于 Catalog 存储操作的资源产生影响的选项可能对总体性能产生轻微的影响。如果需要指定仅影响 Catalog 存储的选项，请参见《参考：语句和选项》>“数据库选项”。

在本指南中会给出性能调优方面的建议。

内存选项

Sybase IQ 将内存用于多种用途。

- 解析查询时从磁盘读取数据的缓冲区
- 从平面文件中装载时从磁盘读取数据的缓冲区
- 用于管理连接、事务、缓冲区和数据库对象的开销

IQ 缓冲区高速缓存大小控制

IQ 缓冲区高速缓存大小选项以及在服务器运行后可以设置的其它选项，决定了有多少内存可用于这些用途。

main 高速缓存的缺省 **IQ 缓冲区高速缓存大小**为 16MB，临时高速缓存的为 8MB，对于任何活动数据库的使用来说，这样的内存量都太少了。

您需要通过下面两种方式之一为 **IQ** 主存储区和临时存储设置缓冲区高速缓存大小：

- 要在服务器范围为当前服务器会话设置缓冲区高速缓存大小，请指定数据库启动实用程序 `start_iq` 的选项 **-iqmc** (**main** 高速缓存大小) 和 **-iqtc** (临时高速缓存大小)。推荐的方法。有关语法，请参见《实用程序指南》>“`start_iq` 数据库服务器启动实用程序”。
- 要为数据库设置高速缓存大小，请使用带有 **main_cache_memory_mb** 或 **temp_cache_memory_mb** 参数的 **sa_server_option** 存储过程。

不过，如果设置的 **IQ 缓冲区高速缓存大小**高于系统允许的大小，则 Sybase IQ 无法打开数据库。

通过服务器选项 (**-iqmc** 和 **-iqtc**)，您还可以使用系统所允许使用的最大内存量，唯一的限制就是计算机上的物理内存量。因此，在 64 位系统中，您应使用 **-iqmc** 和 **-iqtc**。**-iqmc** 和 **-iqtc** 选项不会覆盖 **sa_server_option** 过程的设置。

由 **-iqmc** 和 **-iqtc** 设置的高速缓存大小应用于所有已启动的数据库，直到服务器关闭为止。所以，举例来说，如果将 **-iqmc** 和 **-iqtc** 都设置为 500 (MB)，在服务器启动时启动一个数据库，然后在同一服务器上启动另一个数据库，则您至少需要有 2GB 可供两个 **main** 高速缓存和两个临时高速缓存使用。

缺省情况下，基于 CPU 数量的缓冲区分区已启用，并且可以通过设置 `start_iq` 数据库启动实用程序的 **-iqpartition** 选项进行调整：

- **-iqpartition** - 指定 **IQ** 主缓冲区高速缓存和 **IQ** 临时缓冲区高速缓存中分区的数量。该数字必须为 2 的次方。可用值包括：0 (缺省值)、1、2、4、8、16、32、64、128、256。缺省情况下，**IQ** 自动将分区数计算为 $number_of_cpus/8$ ，并舍入为最接近的 2 的乘方 (最大为 64 次方)。通过调整高速缓存分区数，也许可以提高性能。The **-iqpartition** 开关为 **IQ** 服务器设置此值，并覆盖 `Cache_Partitions` 数据库选项所设置的值。

有关语法，请参见《实用程序指南》>“`start_iq` 数据库服务器启动实用程序”以及《参考：语句和选项》>“数据库选项”>“按字母顺序排列的选项列表”>“`CACHE_PARTITIONS` 选项”。

并发用户开关

您的许可证可设置并发用户的绝对数目。

您还必须设置 **-gm** 开关。利用此必需的开关，可以限制特定服务器上的并发用户连接数目。

-gn 开关可设置多个用户同时运行时将用于 **Catalog** 存储和连接的执行线程数目。它适用于所有操作系统和服务器。

在 **Windows** 中，**start_iq** 可计算此参数的值并使用以下公式来设置它：

```
gn_value = gm_value + 5
```

指定最小值 25。

对于 **UNIX** 平台，请参见针对所用平台的《安装和配置指南》以了解详细信息。

有时您可能希望通过将用户连接数限制为小于许可证允许的连接数来调优特定操作的性能。或者，您可能希望使用 **-iqgovern** 开关来控制查询使用。

并发查询开关

通过 **-iqgovern** 开关，您可以指定特定服务器上的并发查询数量。

这与连接数不同，连接数是由许可证控制的。通过指定 **-iqgovern** 开关，您可以帮助 **IQ** 优化将缓冲区数据分页调度到磁盘上，避免过量使用内存。**-iqgovern** 的缺省值是 (2 x CPU 数) + 10。您可能需要试验确定理想值。对于使用大量活动连接的站点，请尝试将 **-iqgovern** 设置得略低些。

连系内存开关

-iqwmem 开关仅在某些 **UNIX** 平台上创建“连系”内存池。

警告！ 仅当有足够的内存，可分出一部分用于此目的时，才应使用此开关。否则，可能导致性能大幅下降。

处理线程数

使用 **-iqmt** 开关可设置 **Sybase IQ** 可以使用的处理线程数。

Sybase IQ 基于该进程所完成的处理类型、可用线程总数和各种选项设置，将不同数目的内核线程分配给每个用户连接。增加线程数可提高性能。

处理器数

如果是在多处理器计算机上运行，则可以通过 **-gt** 选项设置数据库服务器用于 **Catalog** 存储操作的处理器数。

缺省情况下，会使用所有可用处理器。

目录存储高速缓存大小

使用 **-c** 开关可为 **Catalog** 存储设置高速缓存中的内存量。

start iq 命令以及 `iqdemo.cfg` 和 `default.cfg` 配置文件可将 **-c** 参数设置为 48MB (在 64 位系统上) 和 32MB (在 32 位系统上)。Sybase 建议您使用上述方法之一。

如果启动服务器时未使用 **start iq**、`iqdemo.cfg` 或 `default.cfg`，则系统会根据物理内存量、操作系统和数据库文件大小来计算缺省的初始高速缓存大小。在可用的高速缓存耗尽时，数据库服务器会增加用于目录的高速缓存。

任何小于 10000 的高速缓存大小都会被认为是以 KB 为单位 (1K = 1024 字节)；任何不小于 10000 的高速缓存大小都会被认为以字节为单位。还可以将高速缓存大小指定为 *nK* 或 *nM*。

警告! 要控制目录存储高速缓存大小，必须在配置文件 (`.cfg`) 中或用于启动服务器的 UNIX 命令行上执行以下操作之一 (但不要同时执行这两项操作)：

- 设置 **-c** 参数
- 使用 **-cl** 和 **-ch** 参数为目录存储高速缓存大小设置具体上限和下限

请勿指定这些参数的其它组合。

注意: IQ 存储的高速缓存大小不依赖于 Catalog 高速缓存大小。

有关设置 Catalog 高速缓存大小的详细信息，请参见《实用程序指南》。

CPU 开关数

Sybase IQ 启动命令中的 **-iqnumbercpus** 开关可指定可用于 IQ 的 CPU 数。

它会出于资源规划目的覆盖实际的 CPU 数目。该参数的缺省值为 CPU 的总数，但可用值范围为 1 到 512。

注意: Sybase 建议仅在以下情况下使用 **-iqnumbercpus**：

- 在配有 Intel® CPU 且已启用超线程的计算机上，将 **-iqnumbercpus** 设置为可用的 CPU 数
- 在已使用操作系统实用程序将 Sybase IQ 限制为仅可使用其内 CPU 子集的计算机上

将 **-iqnumbercpus** 设置为大于可用 CPU 数的值可能会影响性能。

影响计时的选项

有若干命令行选项控制出现某些数据库服务器事件的时间。

检查点时间间隔

Sybase IQ 使用检查点生成参照点和它为恢复数据库而需要的其它信息。

使用 **-gc** 开关可设置数据库服务器在不执行检查点的情况下可运行的最大分钟数。

在数据库服务器运行多个数据库时，会使用最先启动的数据库指定的检查点时间，除非它被此开关覆盖。如果输入值 0，则使用缺省值 20 分钟。

恢复时间

利用 **-gr** 参数可设置数据库服务器从系统失败中恢复将花费的最大分钟数。

在数据库服务器运行多个数据库时，会使用最先启动的数据库指定的恢复时间，除非它被此开关覆盖。

网络性能选项

有若干选项可帮助调优网络性能。

它们包括 **-gb**（Windows 中的数据库进程优先级）和 **-p**（最大包大小）。

控制权限的命令行选项

某些命令行选项可控制执行某些全局操作所需的权限。

启动和停止数据库的开关

使用 **-gd** 选项，可限制只有满足以下条件的用户才能够在运行的服务器上启动或停止数据库：在他们已连接到的数据库中具有特定级别的权限。具体值有：

- **DBA** - 只有具备 DBA 权限的用户才能启动其它数据库。
- **ALL** (**start_iq** 和 **default.cfg** 中的缺省值) - 任何用户都可启动和停止数据库。此设置表示 DBA 不需要发出 **START DATABASE** 命令。（请注意，在用户启动特定数据库后，仍需要具备相应权限才能访问它。）
- **NONE** - 任何人都无法在运行的服务器上从 Interactive SQL 启动或停止数据库。

注意： 如果在启动服务器时未设置 **-gd ALL**，则只有 DBA 可以在该服务器上启动其它数据库。这意味着用户无法连接到在服务器启动时或之后 DBA 未启动的数据库。但它还允许非 DBA 停止数据库。因此，某些站点可能希望将此设置更改为生产数据库上的 DBA。

创建和删除数据库的开关

使用 **-gu** 选项，可限制只有满足以下条件的用户才能创建和删除数据库：在他们已连接到的数据库中具有特定级别的权限。

- **DBA** - 只有具备 DBA 权限的用户才能创建和删除数据库。
- **ALL** (缺省值) - 任何用户都可以创建和删除数据库。
- **NONE** - 任何用户都无法创建或删除数据库。
- **UTILITY_DB** - 只有那些能够连接到 **utility_db** 数据库的用户才能创建和删除数据库。

停止服务器开关

使用 **-gk** 选项，可限制能够使用 **DBSTOP** 实用程序或 **STOP ENGINE** 命令关闭服务器的用户：

- **DBA** (缺省值) - 只有具备 DBA 权限的用户才能停止服务器。

- **ALL** - 任何用户都可以停止服务器。
- **NONE** - 任何用户都无法使用 **DBSTOP** 实用程序或 **STOP ENGINE** 命令关闭服务器。

装载和卸载数据库的开关

使用 **-gl** 选项，可限制只有在数据库中具备特定级别权限的用户才能使用 **LOAD TABLE** 装载数据。

- **DBA** - 只有具备 **DBA** 权限的用户才能装载数据。
- **ALL** (**start iq** 和 **default.cfg** 的缺省值) - 任何用户都可以装载数据。
- **NONE** - 无法装载数据。

另请参见

- 进行恢复之前 (第 395 页)
- 移动数据库文件 (第 399 页)
- 实用程序数据库 (第 13 页)
- 实用程序数据库服务器安全性 (第 291 页)
- 检验数据库备份 (第 406 页)

最大目录页大小

数据库服务器高速缓存是按页 (内存中大小固定的区域) 来进行组织的。由于服务器将单个高速缓存用于 **Catalog** 存储 (直到它关闭)，因此所有目录页都必须具有相同的大小。

Catalog 文件也是按大小为 4096、8192、16384 或 32768 字节的页进行组织的。每个数据库页都必须与高速缓存页相匹配。

使用 **-gp** 选项可显式设置目录页大小。通过将 **-gp** 设置为最大大小 32768，可使 Sybase IQ 可以支持的每个表的列数达到最大。

在缺省情况下，服务器页大小与命令行上各个数据库的最大页大小相等。**-gp** 选项可覆盖此缺省值。在服务器启动之后，您就不能装载目录页大小比服务器页大的数据库了。除非指定 **-gp**，否则无法装载目录页大小比在命令行中启动的数据库大的数据库文件。

如果您使用较大的页大小，切记要增加您的高速缓存大小。具有相同大小的高速缓存将只能容纳一部分较大页面，从而限制了空间安排的灵活性。

注意: **-gp** 选项和此处所列的页大小仅适用于 **Catalog** 存储。您可以在 **CREATE DATABASE** 命令的 **IQ PAGE SIZE** 参数中设置 **IQ** 存储的页大小。

另请参见

- **IQ PAGE SIZE** 参数准则 (第 143 页)

客户端/服务器环境选项

有三个选项可以帮助您设置客户端/服务器环境。

- **-x** 指定通信协议选项。
- **-tl** 设置网络连接超时。
- **-ti** 设置客户端连接超时。

通信协议

客户端应用程序与数据库服务器之间的任何通信都需要使用通信协议。Sybase IQ 支持一组通信协议，这些协议用于在网络之间进行通信以及在同一台计算机上进行通信。

数据库服务器支持以下协议：

- *共享内存*用于同一台计算机通信，缺省情况下会被装载。
- *TCP/IP*在所有平台上都受到支持。
- *命名管道*仅在 Windows 2000/2003/XP 中受到支持。“命名管道”可用于以使用 ODBC 或嵌入式 SQL 的 Windows 客户端应用程序为源和目标的同一计算机通信，但是，一般情况下建议不要将它用于此目的。命名管道不用于网络通信。

服务器协议开关

在缺省情况下，数据库服务器启动所有可用的协议。通过使用 **-x** 命令行开关，可以限制可用于数据库服务器的协议。在客户端，可以使用 **CommLinks** 连接参数控制许多相同的选项。

下面的命令使用 **TCP/IP** 协议启动服务器：

```
start_iq -x "tcpip" -n myserver
```

此示例中未严格要求使用引号，但如果 **-x** 的任何参数中有空格，则需要使用引号。如果省略此开关并且使用的是 **TCP/IP**，或者不指定端口号，则使用缺省端口 2638。

您可以添加参数，以便针对每个协议调优服务器的行为。例如，下面的命令行指示服务器使用两块网卡，其中一块使用一个指定的端口号。此命令必须全部输入在一行中，尽管它在此处显示在多行中。

```
start_iq -x "tcpip(MyIP=192.75.209.12:2367,192.75.209.32)" path
\iqdemo.db
```

另请参见

- 网络通信参数（第 102 页）

限制不活动的连接的开关

Sybase IQ 使用两个开关 **-tl** 和 **-ti** 来确定它应当关闭用户连接的时间。

缺省网络超时开关

活动包通过客户端/服务器 **TCP/IP** 通信协议定期发送，目的在于确认连接的完好性。如果服务器运行了一个活动超时周期（缺省为 2 分钟）后未检测到活动包，通信将断

开。服务器将删除与该客户端关联的任何连接。没有警告。任何打开的事务范围内的所有活动都会回退。

服务器上的 `-ti` 开关可以为连接时未指定 `-ti` 开关的所有客户端设置活动超时（以秒为单位）。活动包会以 $(\text{活动超时})/4$ 的间隔发送。

您可能需要在服务器级别为此开关设置更高的值。许多用户（特别是那些使用过 Sybase IQ 早期版本的用户）都不希望在仅处于不活动状态 2 分钟后就断开连接。

请尝试将活动超时设置为 300，并将其与下一节介绍的 `-ti` 的推荐值一起使用。请按如下所示设置此开关：

```
-ti 300
```

如果此值效果不理想，请尝试 `-ti 1200`，它可将活动超时设置为 20 分钟。

注意： 对于在同一台计算机上运行客户端和服务器的用户来说，不存在活动超时。

缺省客户端超时开关

Sybase IQ 会断开在使用 `-ti` 开关指定的分钟数内未提交请求的客户端连接。通过断开不活动的连接，此选项可释放这些连接持有的任何锁。`start_iq` 缺省值为 4400（大约 72 小时），例如，这样您可以从周末开始时启动长时间运行，并确保任何中间结果都不会回退。

有关详细信息，请参见《实用程序指南》>“`start_iq` 数据库服务器启动实用程序”>“`start_iq` 服务器选项”。

强制恢复模式选项

如果在出现故障后需要重新启动服务器，您通常可以使用与往常相同的启动选项来重新启动。

偶尔，您可能需要提供启动选项以强制恢复或恢复泄漏的存储。有关服务器选项，请参见《实用程序指南》>“`start_iq` 数据库服务器实用程序”>“`start_iq` 服务器选项”。

从 Interactive SQL 启动服务器

如果已连接到某个正在运行的数据库服务器，则您可以从 Interactive SQL 启动一个新的服务器。

下面的 Interactive SQL 命令输入在一行中，该命令启动一个数据库服务器，将其命名为 `jill_newserv`，并指定网络连接、连接数和目录页大小：

```
START ENGINE AS jill_newserv STARTLINE 'start_iq -x tcpip(port=5678) -gm 10 -gp 4096'
```

使用 **START ENGINE** 命令可从 Interactive SQL 启动指定的服务器。

在大多数情况下，不建议使用此方法。如果使用该方法，请确保满足以下条件：在预期的系统上启动服务器；在命令中包含了相应的服务器参数；在要启动服务器的系统上适当地设置了环境变量。

另请参见

- 从命令行创建 ODBC 数据源 (第 59 页)
- 如何启动 iqdemo 数据库 (第 39 页)
- 国际语言和字符集 (第 361 页)
- 服务器名 (第 23 页)
- 启动数据库服务器的方法 (第 17 页)

共享内存冲突

在生产环境中，同一系统中一般不运行多个服务器，而且 Sybase 极力反对这样做。但在开发环境中，这种情况可以出现。

如果在同一 UNIX 计算机上运行多个服务器或客户端，并且启用了共享内存，则必须采取特定预防措施来阻止用户连接到错误的服务器。

在尝试启动服务器时，您可能会看到下面的消息：

```
DBSPAWN ERROR -96 -- database engine already running
```

此错误表示启动过程正在查找之前启动的服务器的共享内存段，并且无法创建共享内存段。当 Sybase IQ 或 SQL Anywhere 服务器正在运行时，可能会出现此错误。（如果之前启动的服务器的共享内存端口可见，则 Interactive SQL 还会连接到此服务器，即使您希望它连接到之后启动的服务器也是如此。）如果您在每个系统上只运行一个服务器（Sybase IQ 或 SQL Anywhere），则可以避免该错误。

避免共享内存冲突的方法

要避免在使用共享内存时发生冲突，请考虑执行以下一项或多项操作：

- 创建每台服务器专用的临时目录。通过在两个系统中显式设置 IQTMP15 环境变量，确保每个客户端都与其服务器使用相同的临时目录。有关设置环境变量的详细信息，请参见《参考：构件块、表和过程》>“文件位置和安装设置”>“环境变量”。
- 在 .odbc.ini 文件（在 UNIX 上）中为每个服务器创建一个数据源名称，并提供详细的连接信息。
- 使用指定显式参数而不是依赖于缺省值的连接字符串。
- 通过发出以下命令确认连接：

```
SELECT "database name is" = db_name(), "servername_is" =  
@@servername
```

如果您在每个系统上运行多个服务器，则 Sybase IQ 要求您：

- 确保每个服务器都有一个在启动时使用 **-n** 参数指定的唯一名称。
- 确保每个服务器都有一个使用 **-x** 参数指定的唯一端口号。

有关使用这些参数的示例，请参见《实用程序指南》>“start_iq 数据库服务器启动实用程序”。

另请参见

- UNIX 中的 ODBC 数据源（第 62 页）

服务器活动日志

使用适合于自己所用平台的命令，可以指示 Sybase IQ 在日志文件中捕获服务器活动。

服务器启动消息

当您启动 IQ 服务器时，服务器日志窗口中会显示一系列消息。您将看到的确切的消息集，取决于您的平台和许可选项。以下示例来自于 AIX 系统：

```
Starting server myserver_iqdemo on myserver at port 3658 (09/06
17:25:23)

Run Directory      : /myserver/users/sybase/iqdemo_154_sep05
Server Executable  : /myserver/users/sybase/154_sep05/IQ-15_4/
bin64/iqsrv15
Server Output Log  : /myserver/users/sybase/154_sep05/IQ-15_4/
logfiles/iqdemo_3658.0001.srvlog
Server Version     : 15.4.0.6556/Mainline
Open Client Version : N/A
User Parameters    : '@iqdemo.cfg' 'iqdemo.db'
Default Parameters : -ti 4400 -gn 25

I. 09/06 17:25:26. Sybase IQ
I. 09/06 17:25:26. Version 15.4
I. 09/06 17:25:26. (64bit mode)
I. 09/06 17:25:26. Copyright 1992-2011 by Sybase, Inc. All rights
reserved
I. 09/06 17:25:26. Copyright (c) 2001-2011, Sybase, Inc.
I. 09/06 17:25:26. Portions copyright (c) 1988-2011, iAnywhere
Solutions, Inc. All rights reserved.
I. 09/06 17:25:26. Use of this software is governed by the Sybase
License Agreement.
I. 09/06 17:25:26. Refer to http://www.sybase.com/softwarelicenses.
I. 09/06 17:25:26.
I. 09/06 17:25:26. Processors detected: 4 (containing 16 logical
processors)
I. 09/06 17:25:26. Maximum number of physical processors the server
will use: 4
I. 09/06 17:25:26. Running SunOS 5.10 Generic_144489-12 on X86_64
I. 09/06 17:25:26. Server built for X86_64 processor architecture
I. 09/06 17:25:26. 49152K of memory used for caching
I. 09/06 17:25:26. Minimum cache size: 49152K, maximum cache size:
262144K
I. 09/06 17:25:26. Using a maximum page size of 4096 bytes
I. 09/06 17:25:27. Starting database "iqdemo" (/myserver/users/
sybase/iqdemo_154_sep05/iqdemo.db) at Tue Sep 06 2011 17:25
```

```

I. 09/06 17:25:27. Transaction log: iqdemo.log
I. 09/06 17:25:27. Starting checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2011 17:25
I. 09/06 17:25:27. Finished checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2011 17:25
I. 09/06 17:25:28. Database "iqdemo" (iqdemo.db) started at Tue Sep
06 2011 17:25
I. 09/06 17:25:28. IQ Server iqdemo_3658.
I. 09/06 17:25:28. Database server started at Tue Sep 06 2011 17:25
I. 09/06 17:25:28. Trying to start SharedMemory link ...
I. 09/06 17:25:28. SharedMemory link started successfully
I. 09/06 17:25:28. Trying to start TCPIP link ...
I. 09/06 17:25:28. Starting on port 3658
I. 09/06 17:25:33. TCPIP link started successfully
I. 09/06 17:25:33. Now accepting requests
I. 09/06 17:25:45. Database server shutdown due to HUP signal
I. 09/06 17:25:45. TCPIP listener on IP address (::):3658 is exiting
I. 09/06 17:25:45. TCPIP listener on IP address 0.0.0.0:3658 is
exiting
I. 09/06 17:25:45. Starting checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2011 17:25
I. 09/06 17:25:45. Finished checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2011 17:25
I. 09/06 17:25:46. Database server stopped at Tue Sep 06 2011 17:25

```

start_iq 日志文件

当您使用 **start_iq** 实用程序启动服务器时，服务器活动将记录在由 `$IQLOGDIR15` 定义的目录下放置的 ASCII 文本文件中。此文件包含服务器的标准输出和服务器状态。

日志文件名的格式如下：

```
your_server_name.nnnn.srvlog
```

每次启动服务器时，此编号都会递增。例如，目录可能类似于：

```
demo.0001.srvlog demo.0002.srvlog testdemo.0001.srvlog
```

有关最近一次会话的信息，请选择所需服务器上编号最大的日志。发出 **tail -f** 命令可查看该日志的内容。例如：

```
% tail -f demo.0002.srvlog
```

如果未定义 `$IQLOGDIR15` 目录，则在 UNIX 上，该日志将写入到 `$IQDIR15/logfiles/` 目录中，而在 Windows 上，该日志将写入到安装 Sybase IQ 时定义的 `$IQLOGDIR15` 目录中。

当运行 **start_iq** 时，请指定 **-z** 选项以在日志文件中增加有关连接的其它信息。这将有助于新用户或其他用户排除连接故障。

在 UNIX 系统中，有两种方法可用来检查特定服务器是否正在运行：

- 登录到启动服务器的计算机，然后发出以下命令：

```
ps -eaf | grep iqsrv
```

此输出在不同的 UNIX 平台上略有不同。对于 IBM AIX，这些列为：

UID	PID	PPID	C	STIME	TTY	TIME	CMD
-----	-----	------	---	-------	-----	------	-----

例如:

```
jones 422034 1 0 17:47:36 - 0:04
/ibm64srv/users/sybase/iq152/IQ-15_4/bin64/
iqsrv15
@iqdemo.cfg iqdemo.db -ti 4400 -gn 25 -o
/ibm64srv/users/sybase/iq154/IQ-
15_4/logfiles/ibm64srv_iqdemo.0003.srvlog -hn 7
```

- 使用下一节介绍的 **stop_iq** 实用程序，该实用程序可以显示所有正在运行的 Sybase IQ 进程。

在 Windows 系统中，在系统任务栏中查找一个或多个 Sybase IQ 图标。将光标放在每个图标上，就可以看到服务器名。

为服务器日志文件命名

服务器日志名称缺省为 `server.nmnn.srvlog`。

在 **start_iq** 启动命令中使用 **-o** 开关可更改服务器日志文件名。

例如，要将输出保存到服务器启动目录下名为 `results` 的文件中，请按如下方式启动服务器：

```
start_iq -n imyserver -o results
```

也可以使用 **-o** 开关指定日志文件的完整路径。

UNIX 日志文件

在 UNIX 平台中，有一个额外的日志文件会捕获操作系统输出，包括 `stdout` 和 `stderr` 输出。

该文件名的格式如下：

```
your_server_name.####.stderr
```

对于意外异常，Sybase IQ 将写入堆栈跟踪文件。在 UNIX 系统中，包含堆栈跟踪信息的文件的名称格式如下：

```
stktrc-YYYYMMDD-HHNNSS_#.iq
```

关闭数据库服务器

系统管理员不仅需要知道如何停止 IQ 数据库服务器，而且需要知道何时要停止它、如何控制谁能够停止它以及如何在您关闭操作系统时停止服务器。

何时停止和重新启动服务器

在有限的几种情况下，可能需要停止和重新启动服务器。

这些情况包括：

- 安装新版本的 Sybase IQ
- 重置某些服务器命令行选项
- 使少量服务器范围的数据库选项生效。有关完整列表，请参见《参考：语句和选项》>“数据库选项”>“数据库选项简介”>“数据库选项的作用域和持续时间”。
- 在关闭操作系统会话之前

停止数据库服务器的方法

在 Sybase IQ 中停止数据库服务器的首选方法如下：

- 在 Sybase Central (UNIX 或 Windows) 中，右键单击服务器名并选择“停止服务器”。
- 要关闭 IQ Multiplex 中的服务器，请打开“Multiplex”文件夹，右键单击需要停止的服务器并选择“控制”>“停止”。
- 在 UNIX 中，请在操作系统命令行中使用 **stop_iq** 实用程序。有关详细信息，请参见《实用程序指南》>“stop_iq 数据库关闭实用程序”。
- 当您运行 **stop_iq** 时，它会显示以下消息：

```
"Please note that 'stop_iq' will shut down a server completely
without regard for users, connections, or load process status.
For more control, use the 'dbstop' utility, which has options that
control stopping servers based on active connections."
```

- 在 Windows 中，单击数据库服务器显示屏上的“关闭”，或右键单击系统任务栏中的 IQ 图标并选择“退出”。
- 在 Windows 中，如果该服务器作为服务运行，请在“控制面板”中打开“Service Manager”。选择相应服务并单击“停止”。

通常，在服务器仍与一个或多个客户端保持连接状态时，不应关闭服务器。如果尝试此操作，将会警告您所有未提交的事务都将丢失。请在断开或关闭所有客户端后重新尝试。

您还可以使用以下方法停止数据库服务器：

- 在操作系统命令行中，发出带有适当参数的 **DBSTOP** 命令。使用在启动服务器时使用的相同参数。如果没有正确的连接参数，**DBSTOP** 将不知道如何连接到服务器以指示它关闭。有关使用 **DBSTOP** 的详细信息，请参见《实用程序指南》>“dbstop 数据库关闭实用程序”。
- 在 Interactive SQL 窗口或命令文件中，发出 **STOP ENGINE** 命令以停止指定的数据库服务器。
- 在 UNIX 中，在启动数据库服务器的窗口中键入：

```
q
```

如果您将输入重定向到其它设备或使用 **start_iq** 启动服务器，则此命令将不起作用。它仅适用于 **iqsrv15**。

- 在 UNIX **cron** 或 **at** 作业中，使用带有适当 **-stop** 选项的 **stop_iq**。该实用程序会停止与启动 **cron** 或 **at** 作业的用户关联的一个或所有服务器，具体取决于指定的参

数。用户必须是启动服务器的同一用户。系统不会显示任何操作员提示，也不需要任何操作员操作。

要在此类作业中使用 **stop_iq**，请为该实用程序指定适当的 **-stop** 选项：

```
stop_iq -stop one
```

如果有且仅有一个正在运行的服务器是由启动 **cron** 或 **at** 作业的用户 ID 启动的，则设置 **-stop one** 将关闭一个服务器。

```
stop_iq -stop all
```

设置 **-stop all** 将关闭由启动 **cron** 或 **at** 作业的用户 ID 启动的所有服务器。

注意： 必须在 **cron** 语句中指定 **stop_iq** 可执行文件的完整路径名。

示例 - 使用 **stop_iq** 停止服务器

下面的示例在 UNIX 操作系统命令行中使用 **stop_iq** 实用程序来关闭 Sybase IQ 服务器并关闭到它的所有用户连接。

当您发出 **stop_iq** 命令时，Sybase IQ 会列出其他用户拥有的所有服务器，后跟您拥有的服务器。然后，它会询问您是否要停止您的服务器。例如：

```
% stop_iq

Checking system for IQ 15 Servers ...
The following 2 server(s) are owned by other users.
##      Owner      PID      Started      CPU_Time      Additional Information
--      -
      handari 19895   15:43:44   183:38
start_iq @iqdemo.cfg iqdemo.db -gn 105 -o /server1/users/surya/
IQ-15_4/logfiles/surya_ibm2.001.srvlog -hn 8      pamela      409802
18:05:02      0:05   SVR:ibml_iqdemo2 DB:iqdemo
PORT:2678/ibml/users/sybase/iq151/IQ-15_4/bin64/iqsrv15 @iqdemo.cfg
iqdemo.db -ti 4400 -gn 25 -o /ibml/users/sybase/iq151/IQ
15_4/logfiles/ibm64qa_iq
The following 1 server(s) are owned by 'kermit'
##      Owner      PID      Started      CPU_Time      Additional Information
--      -
1:      kermit  422034   15:11:37      0:07   SVR:myserver_iqdemo
DB:iqdemo PORT:2638 /myserver/users/sybase/iq151/IQ-15_4/bin64/
iqsrv15
@iqdemo.cfg iqdemo.db -ti 4400 -gn 25 -o /myserver/users/sybase/
iq151/IQ-
15_4/logfiles/myserver_iq
start_iq -c 32m -gd all -gm 10 -gn 25 -gp 4096 -ti 4400 -tl 300
@iqdemo.cfg

--

Please note that 'stop_iq' will shut down a server completely
without regard for users connections or load processes status.
For more control, use the 'dbstop' utility, which has options
that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>?
```

要关闭服务器，请键入 **Y** (是)。系统将显示类似于以下内容的消息：

```
Shutting down server (422034) ...
Checkpointing server (422034) ...
Server shutdown.
```

要让服务器继续运行，请键入 **N** (否)。您将返回到系统提示并且 IQ 不会关闭服务器。

如果任何正在运行的服务器都不是由您的用户 ID 启动的，则 Sybase IQ 会显示有关其他用户运行的服务器的信息，然后显示类似于以下内容的消息：

```
There are no servers owned by 'kermit'
```

示例 - 从 Interactive SQL 停止服务器

下面的示例将从 Interactive SQL 停止服务器：

```
STOP ENGINE Ottawa UNCONDITIONALLY
```

可选关键字 **UNCONDITIONALLY** 指定，即使存在到数据库服务器的连接，该服务器也将会停止。

注意： 如果您以 DBA 身份连接到运行在该服务器上的其中一个数据库（包括 utility_db 数据库），或者该服务器是使用 **-gk ALL** 选项启动的，则您可以从 Interactive SQL 停止服务器。

谁可以停止服务器

启动服务器时，可以使用 **-gk** 选项设置用户通过 **DBSTOP** 或 **STOP ENGINE** 停止服务器时所需的权限级别。

所需的缺省权限级别为 **DBA**，但是您还可以将该值设置为 **ALL** 或 **NONE**。如果将其设置为 **NONE**，即使是 **DBA** 也无法执行 **STOP ENGINE**。在生产环境中，Sybase 强烈建议只允许 **DBA** 停止数据库服务器。

通过在 **UNIX** 命令行中运行 **stop_iq**，或在 **Windows** 平台上运行 **Shutdown**，仍可在已启动服务器的计算机上停止服务器和数据库。

关闭操作系统会话

务必在关闭操作系统会话之前显式停止数据库服务器。

如果您关闭正在运行数据库服务器的操作系统会话，或者使用操作系统命令来停止数据库服务器（除上一节中介绍的 **UNIX** 命令以外），则服务器会关机，但并不是干净地关掉。下次装载数据库时，会自动进行恢复。

注意： 例如，在“Windows 任务管理器”的“进程”窗口中停止进程，就属于不干净地停止服务器的命令。

另请参见

- 数据保护（第 375 页）
- 运行 sp_iqcheckdb 的资源问题（第 426 页）

- 系统恢复 (第 355 页)
- 系统恢复与数据库修复 (第 419 页)
- 验证数据库 (第 394 页)

启动和停止数据库的方法

您可以在启动服务器时或在服务器运行后启动数据库。

Sybase 建议在每个服务器上仅运行一个数据库，特别是在生产环境中更应如此。

在正在运行的服务器上启动数据库的方法

有多种方法可用在正在运行的服务器上启动数据库。

- 要从 **Interactive SQL** 或 **嵌入式 SQL** 启动数据库，请使用 **START DATABASE** 语句。请参见《参考：语句和选项》>“SQL 语句”>“START DATABASE 语句”。
- 要从 **Interactive SQL** 或 **Sybase Central** 启动并连接到某个数据库，请使用指定该数据库文件的数据源。
- 要在从系统命令提示符启动 **Interactive SQL** 时启动并连接到数据库，请在连接参数中包含参数 **“DBF=db-file”**。
- 要从 **Sybase Central** 启动数据库，请参见《Sybase IQ 简介》。
- 要在连接到服务器的情况下启动嵌入式数据库，请使用 **DBF** 参数连接到数据库。此参数可为新连接指定数据库文件。相应数据库文件将被装载到当前服务器上。

另请参见

- ODBC 数据源 (第 57 页)
- 启动数据库服务器的方法 (第 17 页)
- ODBC 数据源中的连接参数 (第 44 页)
- JDBC 驱动程序 (第 47 页)
- 从 **Interactive SQL** 连接到演示 数据库 (第 49 页)

数据库启动准则

若干问题可影响数据库启动。

文件访问

为启动数据库，**IQ_SYSTEM_MAIN** 的所有文件、**IQ_SYSTEM_TEMP** 的所有文件以及 **Catalog** 文件 **SYSTEM** 必须可用。启动数据库时可以跳过无法完全打开的数据库空间。如果在启动服务器时无法打除 **IQ_SYSTEM_MAIN** 以外的 **IQ** 主存储区数据库空间的任何可写文件或除 **SYSTEM** 以外的任何 **Catalog** 数据库空间文件，**Sybase IQ** 则会记录一个错误，并将该数据库空间标记为动态脱机（在内存中标记为脱机，而不是在目录中标记为脱机）。如果无法打开 **IQ_SYSTEM_TEMP** 的所有文件，除非使用 **-iqnotemp** 启动参数，否则将无法启动数据库。

Sybase IQ 检查每个数据库空间文件标头中的 `commit_id` 与系统表 `ISYSDBFILE` 和 `ISYSIQDBSPACE` 中的值是否一致，并按照上述方法将任何不匹配的文件或数据库空间标记为脱机。

通过 **ALTER DBSPACE ONLINE** 语句，可以使启动时标记为脱机的数据库空间联机，并假定问题已得以更正并且可以打开该数据库空间。要更正路径问题，可以使用 **ALTER DBSPACE *dbspace name* ALTER FILE *logical filename* RENAME PATH *new pathname*** 更正数据库空间文件的路径。

无法使用驻留在脱机数据库空间中的表对象。对脱机数据库空间中的任何表对象执行除 **ALTER DBSPACE ONLINE** 以外的任何 DDL 或 DML 请求都会产生错误。请注意，数据库空间脱机后，缓冲区高速缓存中可能仍存在有数据页。如果表非常小，即使数据库空间已脱机，整个表也可能包含在缓冲区高速缓存的内存中，并且可以临时使用。

页大小限制

服务器使用大小固定的页在内存中保存数据库信息。一旦服务器已经启动，您就不能装载目录页大小或 IQ 页大小比服务器页大的数据库了。因此，应当始终使用 **-gp** 开关将目录页大小设置为其最大值，即 32768 字节。

权限限制

-gd 服务器命令行选项用于确定启动数据库所需的权限级别。缺省情况下，此选项将会设置为 **DBA**，以便只有具备数据库管理员特权的用户才能启动 IQ 数据库。您还可以将此选项设置为 **ALL** 或 **NONE**。**ALL** 表示所有用户都可以启动数据库。**NONE** 表示任何用户（包括 **DBA**）都无法启动数据库。

另请参见

- Catalog 存储（第 128 页）

停止数据库的方法

您可以用以下方法停止数据库：

- 与由连接字符串启动的数据库断开连接。当最后一个用户与数据库断开连接时，该数据库会自动停止，除非您将 **AUTOSTOP** 连接参数显式设置为 **NO**。
- 从 Interactive SQL 或嵌入式 SQL 中，使用 **STOP DATABASE** 语句。

请参见《参考：语句和选项》>“SQL 语句”>“STOP DATABASE 语句 [Interactive SQL]”。

如何启动 iqdemo 数据库

使用安装时提供的脚本，可以创建 `iqdemo` 数据库以及可轻松启动该数据库的配置文件。

此配置文件名为 `iqdemo.cfg`，它包含启动演示数据库所需的所有参数。请参见《快速入门》>“演示数据库”>“创建和使用 IQ 演示数据库”。

另请参见

- 国际语言和字符集 (第 361 页)
- 服务器名 (第 23 页)
- 启动数据库服务器的方法 (第 17 页)
- 从命令行创建 ODBC 数据源 (第 59 页)
- 从 Interactive SQL 启动服务器 (第 30 页)

Sybase Central 启动和停止方法

如果系统支持图形用户界面，则您可以使用 **Sybase Central** 执行许多管理任务。

请参见 «Sybase IQ 简介» > “运行和连接到服务器” 或使用联机帮助。

Sybase IQ 连接

Sybase IQ 在客户端/服务器环境中运行，在该环境中，多个用户可通过网络连接到数据库服务器。

您可以同时连接多个数据库服务器。您选择的连接选项必须考虑这些情况。

注意：您可以从 Windows 或 Linux 客户端上的 Sybase Central 或 Interactive SQL 连接到 UNIX 服务器上的 Sybase IQ。

客户端应用程序可以从 ODBC、OLE DB 和嵌入式 SQL 应用程序连接到数据库，也可以从 Sybase Central 和 Interactive SQL 连接到数据库。

有关从 Sybase Open Client™ 应用程序连接到数据库的详细信息，请参见《系统管理指南第二卷》>“作为数据服务器的 Sybase IQ”。

有关通过 JDBC（如果您使用的不是 Sybase Central 或 Interactive SQL）进行连接的详细信息，请参见“SQL Anywhere Server - 编程”。

任何使用数据库的客户端应用程序在执行任何工作前都必须建立与该数据库的连接。客户端应用程序的所有活动都必须借助该连接才可以进行。例如，您的用户 ID 确定在数据库上执行操作的权限，并且数据库服务器包含您的用户 ID 信息，因为它是建立连接的请求的一部分。

连接状态

某些客户端工具不会明确指明连接状态。命令失败是连接不存在的第一个提示。

可通过查询数据库名称来快速确认连接。

要显示当前数据库，请使用以下语法：

```
select db_name()
```

要指定其它数据库，请使用以下语法：

```
select db_name([ database_id ])
```

Sybase IQ 如何建立连接

为了建立连接，客户端应用程序调用某个支持接口中的函数。

Sybase IQ 支持以下接口：

- ODBC - 本章讨论 ODBC 连接。
- OLE DB - 本章讨论 OLE DB 连接。

Sybase IQ 连接

- 嵌入式 SQL - 本章讨论嵌入式 SQL 连接。
- - 本章不介绍 Open Client 连接。有关从 Open Client 应用程序连接到 IQ 的信息，请参见《系统管理指南第二卷》>“作为数据服务器的 Sybase IQ”。
- JDBC — 本章介绍 JDBC 连接。有关通过 JDBC 进行连接的详细信息，请参见《Sybase IQ 简介》>“管理数据库”。要创建 JDBC 数据源，请参见“SQL Anywhere Server - 编程”>“JDBC 支持”。

注意： 此参考指向 SQL Anywhere 文档。

注意： JDBC 提供执行 Java 对象和数据库操作的链接。

该接口使用客户端应用程序的调用中包含的连接信息，可能还会使用文件数据源中的磁盘所存储的连接信息，以找到并连接至运行请求的数据库的服务器。

另请参见

- 缺省连接参数（第 55 页）
- 从 Sybase IQ 实用程序连接（第 56 页）

连接学习路线图

连接主题面向各种 IQ 用户而设计。该路线图显示了哪些主题可解决特定的连接需求。

如果需要...	单击本页底部的相关主题链接……
从 Sybase Central 或 Interactive SQL 进行连接的概述 (包括涉及到的驱动程序的说明)	Sybase Central 或 Interactive SQL 连接
快速入门的一些示例	简单连接示例
概念概述	连接参数的工作原理
了解有哪些连接参数可供使用	连接和通信参数
创建数据源	ODBC 数据源
查看如何建立连接的深入说明	ODBC 数据源
添加用户并向其授予权限	数据库权限概述
诊断网络特定的连接问题	网络通信故障排除
了解影响连接的字符集问题	连接字符串和字符集

另请参见

- 连接和通信参数（第 77 页）
- 数据库权限概述（第 281 页）
- 连接参数的工作原理（第 43 页）
- ODBC 数据源（第 57 页）

- 简单连接示例（第 48 页）
- Sybase Central 或 Interactive SQL 连接（第 45 页）
- 网络通信故障排除（第 467 页）

连接参数的工作原理

当应用程序连接到数据库时，它使用一组连接参数来定义连接。

连接参数包括诸如服务器名、数据库名称和用户 ID 之类的信息。

关键字-值对（采用 *parameter=value* 形式）指定每个连接参数。例如，按如下方式用缺省口令指定口令连接参数：

```
Password=sql
```

连接参数组合到连接字符串中。在连接字符串中，按如下方式用分号分隔每个连接参数：

```
ServerName=host_iqdemo;DatabaseName=iqdemo
```

许多连接参数影响服务器的启动方式。建议使用以下连接参数，而不要在 StartLine (START) 连接参数中提供相应的服务器选项：

- EngineName (ENG)
- DatabaseFile (DBF)
- DatabaseName (DBN)

另请参见

- 连接学习路线图（第 42 页）
- CharSet 连接参数 [CS]（第 81 页）
- Language 连接参数 [LANG]（第 96 页）
- 设置区域设置（第 369 页）

连接字符串的格式

本文档中的示例用以下形式表示连接字符串：

```
parameter1=value1  
parameter2=value2  
...
```

这等同于以下连接字符串：

```
parameter1=value1;parameter2=value2
```

一个连接字符串必须在一行中输入，并用分号分隔参数设置。

应用程序如何传递连接参数

连接参数以*连接字符串*形式传递到接口库。

该字符串由一组用分号隔开的参数组成。

通常，由应用程序构建并传递到接口库的连接字符串并不直接对应于用户输入信息的方式。但是，用户可以填写对话框，或者应用程序可以从初始化文件中读取连接信息。

某些 Sybase IQ 实用程序以 **-c** 命令行选项的形式接受连接字符串，并且不加更改即将该连接字符串传递到接口库。例如，要停止服务器 `myserver` 上名为 `iqdemo` 的数据库，请输入：

```
dbstop -c "uid=DBA;pwd=sql;eng=myserver;dbn=iqdemo"
```

注意： Interactive SQL 在内部处理连接字符串，而不仅仅将连接参数传递到接口库。请不要使用 Interactive SQL 从命令提示符测试连接字符串。

ODBC 数据源中的连接参数

许多客户端应用程序（包括应用程序开发系统）使用 ODBC 接口访问 Sybase IQ。

当连接到数据库时，ODBC 应用程序通常使用 ODBC 数据源。ODBC 数据源是一组存储在注册表或文件中的连接参数。

对于 Sybase IQ，不仅 Windows 上的 ODBC 应用程序可以使用 ODBC 数据源，其它应用程序也可以使用此数据源：

- UNIX 以及 Windows 操作系统上的 Sybase IQ 客户端应用程序可以使用 ODBC 数据源。在 UNIX 上，数据源作为文件存储。
- 使用 OLE DB 或嵌入式 SQL 接口的 Sybase IQ 客户端应用程序以及 ODBC 应用程序可以使用 ODBC 数据源。
- Interactive SQL 可使用 ODBC 数据源。
- 使用 iAnywhere JDBC 驱动程序的 JDBC 连接可以使用 ODBC 数据源。

另请参见

- ODBC 数据源（第 57 页）
- 启动和停止数据库的方法（第 38 页）
- 启动数据库服务器的方法（第 17 页）
- JDBC 驱动程序（第 47 页）
- 从 Interactive SQL 连接到演示数据库（第 49 页）

Sybase Central 或 Interactive SQL 连接

必须连接到数据库，以便使用 Sybase Central 或 Interactive SQL 进行管理。

在“连接”对话框中，您告知 Sybase Central 或 Interactive SQL 要连接到哪个数据库、它位于何处以及希望如何连接它。

连接过程取决于您的具体情况。例如，如果您的计算机上已经运行了一个服务器，且该服务器只包含一个数据库，您在“连接”对话框中必须执行的全部操作就是提供用户 ID 和口令。那样，Sybase Central 或 Interactive SQL 就知道要立即连接到正在运行的服务器上的数据库。

如果上述正在运行的服务器装载了多个数据库，或者该服务器尚未运行，或者它运行于另一台计算机上，则需要您在“连接”对话框中提供更详细的信息，以便 Sybase Central 或 Interactive SQL 连接到相应数据库。

本节介绍了如何在 Sybase Central 和 Interactive SQL 中访问“连接”对话框。

注意：为避免发生混淆，请为 Interactive SQL 指定连接参数，而不是依赖于缺省值。您可以在命令行、配置文件或初始化文件（如 `.odbc.ini` 或 `odbc.ini`）中指定连接参数。

例如，如果服务器上启动了多个数据库，则应指定数据库名称。在带有子网的网络中，请指定带有协议选项（含主机编号）的 **CommLinks** 参数。

在 `.odbc.ini` 文件中，必须使用长格式形式的参数。例如，请使用 `DatabaseFile`，而不要使用 `DBF`。如果参数不完整或不正确，将显示如下错误：

```
Database name required to start engine
```

另请参见

- 连接学习路线图（第 42 页）
- 连接和通信参数（第 77 页）
- 简单连接示例（第 48 页）

“连接”对话框

使用“连接”对话框，您可以定义用来连接服务器或数据库的参数。Sybase Central 和 Interactive SQL 中使用的是同一个对话框。

“连接”对话框包含以下选项卡：

- “标识”选项卡用于向数据库标识您自己并指定数据源。
- “数据库”选项卡用于标识要连接到的服务器和/或数据库。

Sybase IQ 连接

- “网络”选项卡用于指定共享内存或 TCP/IP 连接协议，选择安全选项，指定加密参数。
- “高级选项”选项卡用于添加连接参数以及为连接指定驱动程序。

连接之后，主窗口的左窗格中会出现数据库名称，具体位置在运行数据库的服务器的名称下方。该连接的用户 ID 会出现在数据库名称之后的中括号内。

在 Interactive SQL 中连接后，连接信息（包括数据库名称、用户 ID 以及数据库服务器）将显示在“SQL 语句”窗格之上的标题栏中。

打开“连接”对话框 (Sybase Central)

Sybase Central 中的“连接”对话框允许您连接到数据库。

在 Sybase Central 中，选择“连接” > “与 Sybase IQ 15 连接”。

如果您安装了多个 Sybase Central 插件，请从列表中选择“Sybase IQ”。

也可以单击主工具栏上的“连接”按钮或按 F11，打开“连接”对话框。

注意：以后您可以使用*连接配置文件*更方便快速地建立到给定数据库的连接。

另请参见

- 连接到本地服务器上正在运行的数据库（第 50 页）
- 连接到未运行的数据库（第 51 页）
- 简单连接示例（第 48 页）

打开“连接”对话框 (Interactive SQL)

Interactive SQL 中的“连接”对话框允许您连接到数据库。

在 Interactive SQL 中，选择“文件” > “新建窗口”或“SQL” > “连接”。

或者，您可以按 F11 打开“连接”对话框。

在出现“连接”对话框之后，您必须指定在连接时所需的连接参数。例如，使用“数据库”选项卡上的“浏览”按钮将 `iqdemo.db` 指定为数据库文件，在“标识”选项卡上键入 User ID DBA 和 Password sql 并单击“确定”，从而连接到 Sybase IQ 演示数据库。

如果服务器是远程服务器，请务必选择“数据库”选项卡上的“搜索网络中的数据库服务器”。

注意：当连接到用户创建的数据库时，必须完成“数据库文件”和“数据库名称”字段。提供完整的路径名。

用于连接的驱动程序

使用数据库时，所有请求和命令均通过驱动程序到达数据库本身。

另请参见

- 使用 ODBC 管理器创建文件数据源 (第 62 页)
- 使用数据源进行连接 (第 54 页)

JDBC 驱动程序

Interactive SQL 和 Sybase Central 仅支持 Sybase IQ 附带的 iAnywhere JDBC 驱动程序。

有关 JDBC 驱动程序 (包括所需软件) 的详细信息, 请参见 “SQL Anywhere Server - 编程” > “JDBC 支持” > “选择 JDBC 驱动程序” 以及 “SQL Anywhere Server - 编程” > “JDBC 支持” > “使用 jConnect JDBC 驱动程序”。

注意: 此参考指向 SQL Anywhere 文档。

另请参见

- ODBC 数据源 (第 57 页)
- 启动和停止数据库的方法 (第 38 页)
- 启动数据库服务器的方法 (第 17 页)
- ODBC 数据源中的连接参数 (第 44 页)
- 从 Interactive SQL 连接到演示 数据库 (第 49 页)

Sybase Central 中的连接快捷方式

按照这些主题中介绍的过程, 您可以简化 Sybase Central 连接。

服务器对象

要从 Adaptive Server Enterprise 数据库向 Sybase IQ 数据库插入, 每个服务器在 interfaces 文件中都必须有一个条目, 该条目也称为 *服务器对象*。

Sybase IQ 提供的连接信息快捷方式可用于:

- 使用 IQISQL 连接
- 简化数据库启动
- 简化从 Sybase Central 连接

使用 IQDSEEDIT (目录服务编辑器) 可以在 interfaces 文件中创建条目。您必须是 Sybase 主目录 (%SYBASE%) 的所有者或者拥有写入权限才能运行 IQDSEEDIT。

在该文件中添加服务器之后, 当 Sybase Central 请求连接信息时, 将启用 “服务器名” 下拉框。当使用 Tab 键切换至下拉框时, 按空格键将列出使用 IQDSEEDIT 创建的所有条目。您可以从该列表中选择服务器, 也可以只是按要使用的服务器名的第一个字母。多次按相同字母可在以该字母开头的所有值中循环。

创建连接配置文件

连接配置文件使单个用户在启动系统时可以更方便地自动连接到数据库，也可以在不键入连接参数的情况下连接。

1. 从 Sybase Central 菜单中选择“连接” > “连接配置文件”或按 F9。
2. 在“连接配置文件”对话框中，单击“新建”，创建配置文件。
3. 单击“确定”。

由于连接配置文件包含用户名和口令，因此适合单独使用，而不适合在整个安装中使用。通过 Sybase Central，您可以从现有配置文件列表中进行选择，创建、编辑或删除配置文件，或者选择在启动 Sybase Central 时自动使用的配置文件。

如果希望 Sybase Central 在每次启动计算机时连接到此连接，请单击“连接配置文件”窗口上的“设置启动 (Alt-S)”。

如果选择在启动时不进行自动连接，现在，您只需选择“连接” > “连接配置文件”并单击“连接”即可从 Sybase Central 进行连接。

简单连接示例

尽管 Sybase IQ 的连接模型是可配置的而且可能变得非常复杂，但是在许多情况下，连接到数据库是很简单的。

本节介绍连接到 Sybase IQ 数据库的应用程序的一些简单示例。此外，本节还包含快速入门时所需的全部说明，例如，当在本地服务器上运行 `iqdemo` 演示数据库且未连接到网络时。但是，在大多数 IQ 环境中，完整的连接参数集对确保您能够正确连接和断开连接至关重要。

有关使用 Sybase Central 连接到数据库的步骤，请参见《Sybase IQ 简介》。

另请参见

- 连接学习路线图（第 42 页）
- 连接和通信参数（第 77 页）
- Sybase Central 或 Interactive SQL 连接（第 45 页）
- 连接到本地服务器上正在运行的数据库（第 50 页）
- 打开“连接”对话框 (Sybase Central)（第 46 页）
- 连接到未运行的数据库（第 51 页）
- 连接字符串和字符集（第 367 页）

从 Sybase Central 连接到演示数据库

本文档中的许多示例和练习都始于连接到演示数据库。

1. 根据需要为系统启动 Sybase Central。

在 *UNIX* 中，先执行 `IQ-15_4.csh` (或 `.sh`) 脚本，然后再调用 Sybase Central 或 IQ 代理等实用程序。

在 Multiplex 环境中，如果未启动 IQ 代理，请键入：

```
%S99SybaseIQAgent15
```

要启动 Sybase Central，请键入：

```
% scjview
```

注意： 如果已按照《安装和配置指南》中的说明设置环境变量，则可从任何目录发出 `scjview` 命令。

在 *Windows* 中，要启动 Sybase Central，请选择“程序” > “Sybase” > “Sybase IQ 15.4” > “Sybase Central”。

2. 选择“Sybase IQ”。

这将打开一个右侧含有多个选项卡的面板。

3. 在“实用程序”选项卡中，双击“打开 Interactive SQL”。
4. 在“标识”选项卡中，为“用户”和“口令”分别键入 `DBA` 和 `sql`。
5. 在“数据库”选项卡中，选择“查找”。
6. 从“查找服务器”屏幕中选择您的 `iqdemo` 服务器，然后单击“确定”。

按照相同方式不能连接到已在运行的任何数据库服务器。还可以指定非缺省字符集和语言。

另请参见

- 其它数据库空间 (第 162 页)
- 创建主键 (第 177 页)

从 Interactive SQL 连接到演示数据库

本文档中的许多示例和练习都始于从 Interactive SQL (也称为 `dbisql`) 连接到演示数据库。

1. 选择“程序” > “Sybase” > “Sybase IQ 15.4” > “Interactive SQL”。
2. 按前面过程中的第 4 至 6 步进行操作。

按照相同方式不能连接到已在运行的任何数据库服务器。还可以指定非缺省字符集和语言。

有关使用 **Interactive SQL** 的详细信息，请参见《实用程序指南》>“dbisql Interactive SQL 实用程序”。

另请参见

- ODBC 数据源 (第 57 页)
- 启动和停止数据库的方法 (第 38 页)
- 启动数据库服务器的方法 (第 17 页)
- ODBC 数据源中的连接参数 (第 44 页)
- JDBC 驱动程序 (第 47 页)
- 其它数据库空间 (第 162 页)
- 创建主键 (第 177 页)

数据库状态对本地连接的影响

最简单的连接情况是：要连接到的数据库驻留在自己的计算机上。

在这种情况下：

- 该数据库是否已经在服务器上运行？如果是，您可以在“连接”对话框中指定较少的参数。如果不是，则需要标识数据库文件，以便 Sybase Central 或 Interactive SQL 可以启动它。
- 您的计算机上是否有多个数据库正在运行？如果是，您需要标识 Sybase Central 或 Interactive SQL 要连接到的数据库。如果只有一个数据库正在运行，则 Sybase Central 或 Interactive SQL 假定您要连接到该数据库，您不需要在“连接”对话框中指定它。

连接到本地服务器上正在运行的数据库

如果数据库已在本地服务器上运行，您可以指定比平常较少的“连接”对话框参数。

1. 启动 Sybase Central 或 Interactive SQL 并打开“连接”对话框（如果该对话框没有自动显示）。
2. 在该对话框的“标识”选项卡上，输入用户 ID 和口令。
- 3.

执行下列操作之一：

- 如果该服务器只包含一个数据库，请单击“确定”连接到它。
- 如果该服务器包含多个数据库，请单击该对话框的“数据库”选项卡并指定一个数据库名称。这通常是不带路径或扩展名的数据库文件名。

注意：如果已在服务器上装载（启动）数据库，您只需提供成功连接的数据库名称。不必给出数据库文件。

另请参见

- 打开“连接”对话框 (Sybase Central) (第 46 页)

- 连接到未运行的数据库（第 51 页）
- 简单连接示例（第 48 页）

连接到未运行的数据库

如果数据库尚未运行，则必须指定要连接的数据库文件名和路径。

1. 启动 Sybase Central 或 Interactive SQL 并打开“连接”对话框（如果该对话框没有自动显示）。
2. 打开该对话框的“标识”选项卡，输入用户 ID 和口令。
3. 单击该对话框的“数据库”选项卡。
4. 在“数据库文件”字段中指定一个文件（包括完整路径、名称和扩展名）。您可以通过单击“浏览”来搜索文件。
5. 如果您希望以后进行连接的数据库名称不同于该文件名，请在“数据库名称”字段中输入一个名称（不包括路径或扩展名）。

另请参见

- 连接到本地服务器上正在运行的数据库（第 50 页）
- 打开“连接”对话框 (Sybase Central)（第 46 页）
- 简单连接示例（第 48 页）

在 UNIX 上从 Interactive SQL 连接到数据库

以下过程显示如何从 Interactive SQL 连接到正在运行的数据库。

1.

通过在系统命令提示符下键入以下命令来启动服务器和数据库：

```
start_iq dbname
```

2.

通过在系统命令提示符下键入以下命令来启动 Interactive SQL：

```
dbisql -c "uid=userID;pwd=password" -host hostname -port portnum -n servername
      dbname.db
```

-c 参数指定连接参数。

例如，要连接到远程主机 *fiona* 上的演示数据库，请输入：

```
dbisql -c "uid=DBA;pwd=sql" -host fiona -port 1870 -n fiona_iqdemo
$IQDIR15/demo/iqdemo.db
```

如果要连接到本地计算机上的数据库，则不必指定主机和端口。

从 UNIX 系统连接

下面的过程显示如何从 UNIX 系统上的命令行连接到正在运行的数据库。

1. 确保已按照《参考：构件块、表和过程》>“文件位置和安装设置”中的说明正确设置 PATH 和其它环境变量。

2.

为了确保正在运行的服务器上已装载演示数据库，请在 UNIX 提示符下输入：

```
ps -eaf | grep iqdemo
```

要启动演示数据库，请输入：

```
cd $IQDIR15/demo
```

```
start_iq @iqdemo.cfg iqdemo.db
```

3.

启动 Interactive SQL：

```
dbisql -c "uid=DBA;pwd=sql;eng=servername;links=tcPIP"
```

将 *servername* 替换为 **start_iq** 命令中提供的相同服务器名以启动该服务器。

注意： 如果您希望使用更早的实用程序 Interactive SQL Classic 而不是基于 Java 的版本，请输入 **dbisqlc**，而不是 **dbisql**。请注意，尽管 **dbisqlc** 受支持，但 **dbisqlc** 并未包含 **dbisql** 的所有功能。

-c 参数用于指定连接参数。您还可以按照本章后面部分的说明在数据源中指定这些参数。

注意： 仅当使用 TCP/IP 连接数据库时才需要 links=tcPIP (或 CommLinks=tcPIP) 参数。如果使用共享内存端口连接本地数据库，则可以省略 links 参数；然而，包含完整的网络参数始终是比较安全的，并且这在某些平台上是必需的。

要连接到远程主机上的数据库，必须添加主机名和端口号。例如：

```
dbisql -c "uid=DBA;pwd=sql;eng=SERV1_iqdemo;links=tcPIP(host=SERV2;port=1234)"
```

您可以根据个人喜好使用链接子句的以下替代形式，该形式具有相同结果：

```
"links=tcPIP(host=SERV2:1234)"
```

从 Windows 系统连接

下面的过程显示如何从 Windows 系统上的命令提示符连接到正在运行的数据库。

1. 选择“程序”>“Sybase”>“Sybase IQ 15.4”>“Interactive SQL”或在 Windows 命令提示符下输入：

```
dbisql
```

您可以在 `dbisql` 命令中包括 `-c` 参数，以便指定连接参数，如上面的连接到 UNIX 的过程所述。如果省略这些参数，则会显示“Interactive SQL 连接”对话框。

2. 在“连接”对话框中，输入您的用户名和口令。

例如，对于 `iqdemo` 数据库，请输入 `DBA` 和 `sql`，即创建 Sybase IQ 数据库时该数据库的缺省用户和口令组合。

3. 单击“数据库”选项卡，键入启动服务器时所使用的服务器名（例如，对于 `iqdemo` 数据库，服务器名为“`hostname_iqdemo`”）。该名称在局域网中必须唯一。

对于远程服务器，请在“网络”选项卡上以 `host name` 和 `port number` 的格式指定服务器。

缺省端口号是 2638，但是如果服务器启动时使用的是另一端口号，则会改为使用后者。可以通过运行“Sybase IQ 15.4” > “ODBC 管理器 (32 位)”或“Sybase IQ 15.4” > “ODBC 管理器 (64 位)”来查找端口号。在“用户 DSN”选项卡上选择“用户数据源”，然后单击“配置”。可以通过在命令提示符下键入 `dblocate` 来查找端口号。

此过程使您连接到此服务器上启动的第一个数据库。如果正在运行多个数据库，则需要单击“浏览”选择所需数据库。

4. 单击“确定”以连接到数据库。

如果弹出“连接”对话框或有关丢失信息的错误消息，则可能需要在“高级”选项卡中输入 `-host` 和 `-port` 或其它丢失信息。如果数据库位于远程服务器上，则输入 `-host` 和 `-port` 参数（分两行输入），如下所示：

```
-host fiona -port 1870
```

5. 连接到数据库后，系统将显示“Interactive SQL”窗口。“Interactive SQL”窗口在其标题栏上显示连接的数据库名称、用户 ID 和服务名。

如果使用 Interactive SQL Classic 进行连接，则“统计”窗口中将显示“已连接到数据库”字样，并显示一则消息，用于显示数据库所使用的归类。

连接到嵌入式数据库

*嵌入式数据库*旨在供单一应用程序使用，它运行于该应用程序所在的同一台计算机上，而且，应用程序用户基本上无法看到它。

如果应用程序使用嵌入式数据库，则该数据库在应用程序进行连接时通常并不运行。在这种情况下，您可以使用连接字符串来启动数据库，方法是在连接字符串的 `DatabaseFile (DBF)` 参数中指定数据库文件。

数据库文件参数

`DBF` 参数指定要使用哪个数据库文件。数据库文件会自动装载到缺省服务器上；或者，如果未运行任何服务器，则会启动一个服务器。

当不再有到数据库的连接时（通常是当启动该连接的应用程序断开连接时），数据库将被卸载。如果服务器是由该连接启动的，它在数据库卸载之后停止。

下列连接参数显示如何将演示数据库作为嵌入式数据库来装载：

```
dbf=path\iqdemo.db  
uid=DBA  
pwd=sql
```

其中 *path* 是 Sybase IQ 安装目录的名称。

Start 参数

下列连接参数显示如何自定义作为嵌入式数据库的演示数据库的启动。如果您希望使用命令行选项（如高速缓存大小），这会非常有用：

```
Start=start_iq -gd all  
-gl all -gm 10 -gn 25 -gp 4096 -c 32M  
-ti 4400 -tl 300  
dbf=path\iqdemo.db  
uid=DBA  
pwd=sql
```

示例：从 Interactive SQL 连接

在此示例中，演示数据库是 Interactive SQL 中的嵌入式数据库。

在 Windows 中从 Interactive SQL 连接到嵌入式数据库

要连接到嵌入式数据库，请执行以下操作：

1. 在没有运行数据库的情况下启动 Interactive SQL。您可以使用以下方式之一：
 - 从 Windows “程序” 菜单中，选择 “Sybase” > “Sybase IQ 15.4” > “Interactive SQL”，或者
 - 在系统命令提示符下键入 dbisql。

Interactive SQL 启动时，它不会与任何数据库相连。

2. 在命令窗口中键入 CONNECT，然后按 F9 以执行该命令。连接对话框即会出现。
3. 如果数据库使用 ODBC 数据源，请选择该数据源。
4. 输入 DBA 作为用户 ID，输入 sql 作为口令。然后单击“数据库”选项卡。在“数据库文件”字段中输入演示数据库的完整路径。例如，如果安装目录是 c:\sybase\IQ-15_4，请输入：

```
c:\sybase\IQ-15_4\iqdemo.db
```

5. 将所有其它字段留空，然后单击“确定”。Sybase IQ 随即启动并装载演示数据库，Interactive SQL 将连接到该数据库。

使用数据源进行连接

您可以在数据源中保存多组连接参数。使用 iAnywhere JDBC 驱动程序的 ODBC 和 JDBC 使用数据源，这与嵌入式 SQL 应用程序（如 Interactive SQL Classic）相同。

您可以从 ODBC 管理器创建数据源。

所有应用程序都可以从使用数据源中受益。

iqdemo 数据源保存一组连接参数，包括数据库文件和一个用于启动演示数据库的 Start 参数。此数据源中的服务器名为 “hostname_iqdemo”，其中 hostname 表示您的系统名。

另请参见

- 使用 ODBC 管理器创建文件数据源（第 62 页）
- 用于连接的驱动程序（第 46 页）

使用数据源从 Sybase Central 或 Interactive SQL 连接

使用 ODBC 数据源从 Sybase Central 或 Interactive SQL 连接。

1. 启动 Sybase Central 或 Interactive SQL 并打开“连接”对话框（如果该对话框没有自动显示）。
2. 在“标识”选项卡（在 Interactive SQL Classic 中为“登录”选项卡）上，输入用户 ID 和口令，例如 DBA 和 sql。

3.

在“标识”选项卡的下半部分，执行下列操作之一：

- 选中“ODBC 数据源名称”选项并指定一个数据源名称（等同于 DSN 连接参数，该参数引用注册表中的数据源）。要查看数据源列表，请单击“浏览”。
- 选中“ODBC 数据源文件”选项并指定一个数据源文件（等同于 FileDSN 连接参数，该参数引用保存在文件中的数据源）。您可以通过单击“浏览”来搜索文件。

Sybase IQ 演示数据源保存一组连接参数，包括数据库文件和 Start 参数。

注意： 也可以通过在启动 Interactive SQL 时包括 dsn 连接参数来指定数据源名称：

```
dbisql -c "dsn=Sybase IQ Demo"
```

缺省连接参数

您可以不指定各种连接参数，而是使用缺省行为建立连接。

注意： 在生产环境中，依赖缺省行为一定要格外谨慎。有些客户可能会在其计算机中安装其它 Sybase IQ 或 SQL Anywhere 应用程序，给他们分发应用程序时尤其要谨慎。

缺省数据库服务器

如果要连接到本地服务器上的数据库，并且该服务器上已启动多个数据库，则需要指定要连接的数据库，但是也可以保留该服务器的缺省值：

```
dbn=db_name  
uid=user_id  
pwd=password
```

注意： 如果正在运行多个本地服务器，切勿使用上述参数，否则可能会连接到错误服务器。

缺省数据库

如果正在运行多个服务器，则需要指定希望连接到哪个服务器。如果该服务器上只启动了一个数据库，则无需指定数据库名称。下面的连接字符串使用缺省数据库连接到指定的服务器：

```
eng=server_name  
uid=user_id  
pwd=password
```

不使用缺省值进行连接

下面的连接字符串使用指定数据库连接到指定服务器：

```
eng=server_name  
dbn=db_name  
uid=user_id  
pwd=password
```

另请参见

- Sybase IQ 如何建立连接（第 41 页）
- 从 Sybase IQ 实用程序连接（第 56 页）

从 Sybase IQ 实用程序连接

Sybase IQ 数据库实用程序使用嵌入式 SQL 与服务器通信（而不是直接与数据库文件通信）。

数据库实用程序如何获取连接参数值

许多管理实用程序都通过以下方法来获取连接参数值：

1. 使用在命令行上指定的值（如果有的话）。例如，以下命令使用用户 ID DBA 和口令 sql 在 demo 数据库上备份 Catalog 存储：

```
dbbackup -y -x -c  
'uid=DBA;pwd=sql;eng=iqdemo;dbn=iqdemo.db;links=tcPIP{host=localh  
ost:2638}' -d '/mydir'
```

2. 使用 SQLCONNECT 环境变量设置（如果缺少任何命令行值）。Sybase IQ 数据库实用程序不会自动设置此变量。此选项提供的口令安全性高于任何其它方法。有关 SQLCONNECT 环境变量的说明，请参见《参考：构件块、表和过程》>“文件位置和安装设置”>“环境变量”。
3. 提示您输入用户 ID 和口令以连接到缺省服务器上的缺省数据库（如果未在命令行或 SQLCONNECT 环境变量中设置参数）。

有关各数据库实用程序的命令行选项的说明，请参见《实用程序指南》。

另请参见

- Sybase IQ 如何建立连接 (第 41 页)
- 缺省连接参数 (第 55 页)

ODBC 数据源

您可以将一组 Sybase IQ 连接参数以数据源形式存储在系统注册表或文件中。需要数据源才能使用通过开放式数据库互连 (ODBC) 接口进行连接的应用程序。

ODBC 接口由 Microsoft Corporation 定义，它是 Windows 环境中用于连接客户端应用程序与数据库管理系统的标准接口。许多客户端应用程序 (包括应用程序开发系统) 都使用 ODBC 接口访问多种数据库系统。

在 Windows 中，使用 ODBC 管理器可以集中创建和管理 ODBC 数据源。

注意：当使用 32 位 ODBC 驱动程序时，请使用 32 位 ODBC 管理器管理数据源；当使用 64 位客户端时，请使用 64 位 ODBC 管理器。

以下过程使用 ODBC 管理器向现有 `odbc.ini` 添加新的数据源或根据需要创建新文件。

Sybase IQ 还包括一个用于创建数据源的名为 `iqdsn` 的跨平台命令行实用程序。

准备工作

本节介绍如何创建 ODBC 数据源。在创建数据源之前，您需要了解要在其中包括哪些连接参数。

您可以在 ODBC 管理器中使用“用户数据源”、“文件数据源”和“系统数据源”。

另请参见

- 启动和停止数据库的方法 (第 38 页)
- 启动数据库服务器的方法 (第 17 页)
- ODBC 数据源中的连接参数 (第 44 页)
- JDBC 驱动程序 (第 47 页)
- 从 Interactive SQL 连接到演示数据库 (第 49 页)
- 连接学习路线图 (第 42 页)

数据源保存在何处

在基于 Windows 或 UNIX/Linux 的系统上可以使用数据源帮助客户端简化连接。

当使用 ODBC 连接数据库时，您可以使用 ODBC 数据源。数据源包含一组连接参数。对于要连接到的每个数据库，都需要在客户端计算机上有一个 ODBC 数据源。

一旦有了数据源，您的连接字符串就可以简单地指明要使用的数据源：

- 数据源 - 使用 DSN 连接参数引用用户或系统数据源:

```
DSN=my data source
```

在 Windows 中, 用户和系统数据源存储在注册表和 `odbc.ini` 文件中。在 UNIX 平台中, 用户数据源位于 `.odbc.ini` 文件中。

- 文件数据源 - 使用 FileDSN 连接参数引用保存在文件中的数据源:

```
FileDSN=mysource.dsn
```

除 FileDSN 允许的加密口令以外, 您可以在 DSN 和 FileDSN 中放置相同的连接信息。

ODBC 数据源可用于帮助众多客户端连接包括下列各项在内的应用程序:

- Windows、Linux 和 Unix 上使用 ODBC 接口的应用程序
- 使用 iAnywhere JDBC 驱动程序的 Java 应用程序

从 ODBC 管理器中创建数据源

ODBC 管理器提供创建 ODBC 数据源的简便方式。

1. 启动 ODBC 管理器:

选择“设置” > “控制面板” > “管理工具” > “数据源 (ODBC)”。

或

Sybase IQ 15.4 > “ODBC 管理器 (32 位)”

此时将出现“ODBC 数据源管理程序”。

2. 单击“添加”。

“创建新数据源”向导随即出现。

3. 从驱动程序列表中选择“Sybase IQ”，然后单击“完成”。

4. 在“ODBC 配置”窗口中, 键入数据源名称。

5. 现在单击“登录”选项卡。键入数据库的“用户 ID”和“口令”。例如, 使用 DBA 和 sql。

6. 单击“Database”选项卡。如果数据源位于本地计算机上, 请键入服务器名、启动行和数据库文件 (包括路径)。

7. 如果数据源位于远程系统中, 请单击“网络”选项卡。单击相应协议的框, 指定框旁边的选项。例如, 要使用 TCP/IP 协议和端口 1870 连接到系统 PUSHKIN 上的服务器, 则应单击“TCP/IP”, 然后键入:

```
host=pushkin:1870
```

还可以使用主机网络地址。例如:

```
host=157.133.66.75:1870
```

8. 数据源定义完成后, 单击“确定”。

此时 ODBC 数据源管理程序将返回“用户 DSN”选项卡。

注意：指定网络连接时，需要针对每个数据库服务器使用不同的 *systemname:port#* 组合。端口号必须与启动服务器所使用的端口号相符。

从命令行创建 ODBC 数据源

您可以使用 `iqdsn` 命令行实用程序创建“用户数据源”和“系统数据源”。不能使用 `iqdsn` 创建“文件数据源”。还可以使用 ODBC 管理器创建“用户数据源”、“系统数据源”和“文件数据源”。

1. 打开命令提示符。
- 2.

输入 `iqdsn` 命令，指定希望使用的连接参数。例如，以下命令为 Sybase IQ 演示数据库创建数据源。该命令必须在一行中输入：

```
iqdsn -w "My DSN"  
"uid=DBA;pwd=sql;dbf=c:\Program Files\Sybase\IQ-15_4\demo  
\iqdemo.db"
```

`iqdsn` 输出包含下面一行：

```
User Data Source "My DSN" written to registry.
```

`iqdsn` 实用程序列出在 Windows 命令行上创建的 Sybase IQ 用户数据源。

有关 `iqdsn` 实用程序的详细信息，请参见《实用程序指南》>“`iqdsn` 数据库管理实用程序”。

要编辑数据源，请从 ODBC 管理器窗口中的列表中选择一个数据源，并单击“配置”。

要跨网络访问 Windows 以创建 ODBC 数据源，请参见《安装和配置指南》。

另请参见

- 从 Interactive SQL 启动服务器（第 30 页）
- 如何启动 `iqdemo` 数据库（第 39 页）
- 国际语言和字符集（第 361 页）
- 服务器名（第 23 页）
- 启动数据库服务器的方法（第 17 页）

测试 ODBC 数据源

可以使用 ODBC 管理器来测试 ODBC 数据源。

1. 启动数据库。例如，要启动 演示数据库，请选择“程序” > “Sybase” > “Sybase IQ15.4” > “启动 Sybase IQ 演示数据库”。
2. 在 ODBC 数据源管理程序中，从“用户数据源”列表中选择您的新数据源。
3. 单击“配置”。
4. 在“ODBC 配置”对话框中单击“测试连接”。

如果无法访问数据源，请检查是否已使用正确的文件名和路径名填充各选项卡。

在 ODBC 管理器中配置 ODBC 数据源

“ODBC 配置”对话框中的选项是按选项卡进行组织的。

选项卡上的选项与 Sybase IQ 连接参数相对应，“ODBC”选项卡除外。

另请参见

- 使用 ODBC 管理器创建文件数据源（第 62 页）

“ODBC”选项卡

使用“ODBC 配置”对话框中的“ODBC”选项卡指定数据源名称等属性。

区域	描述
“ODBC”选项卡 - 数据源名称	“数据源名称”用于标识 ODBC 数据源。对于该数据源，您可以使用任何说明性名称（允许使用空格），但是建议您将该名称设置得简短些，因为您需要在连接字符串中输入它。
“ODBC”选项卡 - 说明	您可以输入任意长度的数据源说明。
“ODBC”选项卡 - 隔离级别	Sybase IQ 数据源的隔离级别始终为有效级别 3。但是，缺省 Catalog 存储隔离级别为 0。通常应保留为空白。
“ODBC”选项卡 - Microsoft 应用程序 (SQL Statistics 中的键)	如果希望 SQL Statistics 返回外键，请选中此框。ODBC 规范指出 SQL Statistics 不应当返回主键和外键；但是，某些 Microsoft 应用程序（例如 Visual Basic 和 Access）假定主键和外键是由 SQL Statistics 返回的。
“ODBC”选项卡 - Delphi 应用程序	选中此框可提高 Borland Delphi 应用程序的性能。如果选中此选项，则向每一行都指派一个书签值，而不是另外指派两个（一个用于向前读取，另一个用于向后读取）。

区域	描述
“ODBC”选项卡 - 抑制读取警告信息	选中此框可以不显示读取过程中从数据库服务器返回的警告消息。
“ODBC”选项卡 - 防止“驱动程序不支持”错误	Sybase IQ 15 ODBC 驱动程序返回 <code>Driver not capable</code> 错误代码，因为它不支持限定符。某些 ODBC 应用程序不能正确处理该错误。选中此框可禁用此错误代码，以便运行此类应用程序。
“ODBC”选项卡 - 延迟自动提交直到语句关闭	强制 Sybase IQ 15 ODBC 驱动程序将提交操作延迟到语句关闭以后。
“ODBC”选项卡 - 描述游标行为	选择您希望在执行或恢复一个过程时重新描述游标的频率。
“ODBC”选项卡 - 转换器	如果您的数据库使用 OEM 代码页，请选择“MS 代码页转换器”。如果您的数据库使用 ANSI 代码页（缺省设置），请不要选中此选项。
“ODBC”选项卡 - 测试连接	测试提供的信息能否产生正确的连接。为了使测试有效，必须指定用户 ID 和口令。

另请参见

- DataSourceName 连接参数 [DSN]（第 89 页）
- FileDataSourceName 连接参数 [FileDSN]（第 94 页）

文件数据源

在 Windows 操作系统上，ODBC 数据源通常存储在系统注册表中。文件数据源是一种以文件形式存储的替代形式。Windows 和 UNIX 系统均支持文件数据源。

在 Windows 中，文件数据源的扩展名通常为 `.dsn`。它们由不同部分组成，每一部分都以用中括号括起来的名称开始。DSN 文件的布局与初始化文件非常相似。

要使用文件数据源进行连接，请使用 **FileDSN** 连接参数。不能在同一连接中同时使用 **DSN** 和 **FileDSN**。

可分发文件数据源

文件数据源的其中一个优点是，可将文件分发到用户，这样，就不必在每台计算机上重建连接信息。如果文件放在文件数据源的缺省位置，则它由 ODBC 自动获取。按照这种方式，可以简化对多个用户连接的管理工作。

注意：由于 DSN 存储在 Windows 注册表中，因此它们是公用信息。为此，不应在 DSN 中放置口令，除非对口令加密。如果要在数据源中存储口令，请使用“文件 DSN”。

嵌入式 SQL 应用程序也可以使用 ODBC 文件数据源。

使用 ODBC 管理器创建文件数据源

可以使用 ODBC 管理器工具来创建文件数据源。

1. 启动 ODBC 管理器，单击“文件 DSN”选项卡，然后单击“添加”。
2. 从驱动程序列表中选择“Sybase IQ 12”，然后单击“下一步”。
3. 按照说明创建数据源。

另请参见

- 使用数据源进行连接（第 54 页）
- 用于连接的驱动程序（第 46 页）
- 在 ODBC 管理器中配置 ODBC 数据源（第 60 页）

文件数据源和文本编辑器

文件数据源是一个文本文件，可使用任何文本编辑器进行编辑。使用文本编辑器的局限性之一是不能在文件中存储加密口令。

文件数据源示例

```
[Sample File Data Source]
ENG = iqdemo
DBA = DBA
PWD = sql
```

请参见《实用程序指南》>“iqdsn 数据库管理实用程序”。

UNIX 中的 ODBC 数据源

在 UNIX 之类的操作系统中，odbc.ini 文件包含一个数据源列表。

创建 .odbc.ini 文件时，您必须使用各个标识符的长格式形式，例如：

```
[My Data Source] EngineName=myserver CommLinks=tcPIP(port=1870)
Userid=DBA Password=sql
```

除以上示例 DSN 中的连接参数之外，还可以在 .odbc.ini 文件中包含其它参数。有关受支持的所有连接参数的信息，请参见“连接参数”。网络通信参数是作为 CommLinks (LINKS) 参数的一部分来添加的。有关网络参数的完整列表，请参见“网络通信参数”。

您可以在 UNIX 上使用 iqdsn 命令行实用程序来创建和管理 ODBC 数据源。请参见“从命令行创建 ODBC 数据源”。

ODBC 数据源文件位置

对 ODBC 函数的引用在运行时进行解析。

若要与 ODBC 数据源进行连接，`.odbc.ini` 文件的位置必须由以下变量之一引用。Sybase IQ 可按照以下顺序搜索由以下变量指定的目录：

1. `$ODBCINI` - 必须包含 `.odbc.ini` 文件确切的完整路径名。
2. `$HOME`
3. 当前目录
4. `$PATH`

Sybase IQ 客户端在搜索 `.odbc.ini` 时将忽略以下变量：

1. `$ODBC_HOME`
2. `$ODBC_INI`

使用文本编辑器编辑 `.odbc.ini`。

在 UNIX 之类的操作系统中，Sybase IQ 会安装 ODBC 驱动程序和驱动程序管理器。驱动程序文件名包含操作系统特定扩展名，例如，对于 Solaris 系统，特定扩展名为 `so`。如果您使用的是使用 `libodbc.so` (`libodbc.so.1`) 或 `libodbcinst.so` (`libodbcinst.so.1`) 的 ODBC 应用程序，则只需为这些应用程序创建指向 `$SYBASE/IQ-15_4/lib64/libdbodbc12.so.1` 的符号链接。如果要创建自定义 ODBC 应用程序，则可以直接链接到 `libdbodbc12.so`。

如果 ODBC 驱动程序管理器不存在，IQ ODBC 驱动程序（通过符号链接查找）则使用 `.odbc.ini` 来获取数据源信息。

ODBC 跟踪输出

要创建 ODBC 跟踪文件，请参见“SQL Anywhere Server - 编程” > “ODBC 支持” > “构建 ODBC 应用程序” > “在 UNIX 上使用 SQL Anywhere ODBC 驱动程序管理器”。

注意： 此参考指向 SQL Anywhere 文档。

使用 `libdbodbc12.so` 驱动程序并由该驱动程序选择多线程驱动程序或非线程驱动程序。跟踪功能位于开关 (`libdbodbc12.so`) 中，而不在各个驱动程序 (`libdbodbc12_n.so` 或 `libdodbc12_r.so`) 中。如果将驱动程序更改为指向 `_r` 版本，则会从调用序列删除开关，从而防止跟踪。

另请参见

- 避免共享内存冲突的方法（第 31 页）
- 连接参数（第 77 页）
- 检查常见网络通信问题（第 470 页）
- 网络通信参数（第 102 页）
- 从命令行创建 ODBC 数据源（第 59 页）

iAnywhere Solutions Oracle 驱动程序

iAnywhere Solutions 11 - Oracle ODBC 驱动程序是专门定制用于 Sybase IQ 或 iAnywhere 的。此驱动程序不能与第三方软件一起使用。

如果将 Oracle 和 OMNI 一起使用，则必须在此 Oracle 驱动程序所在的计算机上安装 Oracle 客户端。

使用 ODBC 管理器、.odbc.ini 文件（在 UNIX 中）或 iqdsn 实用程序来配置 Oracle 驱动程序。

表 3. iAnywhere Oracle 驱动程序配置选项

Windows ODBC 管理器	针对 iqdsn 命令或 .odbc.ini 文件进行的配置	描述
数据源名称	对于 iqdsn，请使用 -w 选项。	用于标识数据源的名称。
用户 ID	在 iqdsn 中，在连接字符串中设置 UserID 选项。	应用程序用来连接到 Oracle 数据库的缺省登录 ID。如果将此字段留空，就必须在连接时提供信息。
口令	在 iqdsn 中，在连接字符串中设置此选项。	应用程序用来连接到 Oracle 数据库的口令。如果将此字段留空，就必须在连接时提供信息。
SID	SID	在 Oracle 安装目录下的 network/admin/tnsnames.ora 中存储的 TNS 服务名。
启用 Microsoft 分布式事务	对于 iqdsn，在连接字符串中使用 enableMSDTC 选项。odbc.ini 不支持。	选择此选项可以在 Microsoft 分布式事务协调器中征用您的事务。选择此选项后，Oracle ODBC 驱动程序会需要 Oracle 二进制文件 oramts.dll（用于 Oracle 9i 客户端）或 oramts10.dll（用于 Oracle 10g 客户端）。
加密口令	对于 iqdsn，请使用 -pe 选项。odbc.ini 不支持。	选择此选项可以在数据源中存储加密格式的口令。
过程返回结果	在 iqdsn 中，在连接字符串中设置 ProcResults 选项。	如果存储过程可以返回结果，请选择此选项。缺省设置是过程不返回结果（不选择）。如果您的 download_cursor 或 download_delete_cursor 脚本是存储过程调用，则请将此选项设置为 yes。

Windows ODBC 管理器	针对 iqdsn 命令或 .odbc.ini 文件进行的配置	描述
数组大小	在 iqdsn 中，在连接字符串中设置 ArraySize 选项。	用来以语句为单位预取行的字节数组的大小（以字节为单位）。缺省值是 60000。增大该值可显著提高读取性能，但需要分配额外的内存。

在 UNIX 中创建 Oracle DSN

在 UNIX 中，您可以在 ODBC 系统信息文件（通常称为 .odbc.ini）中设置驱动程序。

1. 使用文本编辑器编辑该系统信息文件。此驱动程序所在的部分应显示如下（每个字段都输入相应的值）：

```
[sample_dsn_using_the_ias_odbc_driver_for_oracle]Driver=full-path/libdboraodbc10_r.soUserID=user-idPassword=passwordSID=TNS-service-nameProcResults=[yes|no]ArraySize=bytes
```

2. 单击“添加”。
3. 选择“iAnywhere Solutions 11 - Oracle”，然后单击“完成”。
4. 指定配置选项。
5. 单击“测试连接”，然后单击“确定”。

在 Windows 中创建 Oracle DSN

使用 ODBC 管理器在 Windows 中创建 Oracle 数据源名称。

1. 选择“开始” > “程序” > “Sybase” > “Sybase IQ 15.4” > “ODBC 管理器 (32 位)”。
2. 单击“添加”。
3. 选择“iAnywhere Solutions 12 - Oracle”，然后单击“完成”。
4. 指定配置选项。
5. 单击“测试连接”，然后单击“确定”。

使用 IQDSN 创建 Oracle DSN

在 UNIX 中，还可以使用 IQDSN 实用程序来创建 Oracle DSN。

要使用 iqdsn 实用程序创建 Oracle DSN，请使用：

```
iqdsn -w data-source-name -or -c configuration-options
```

例如：

```
iqdsn -w MyOracleDSN -or -pe -c
Userid=dba;Password=sql;SID=abcd;ArraySize=100000;ProcResults=y;enableMSDIC=n
```

例如：

```
iqdsn -w MyOracleDSN -or -pe -c  
Userid=dba;Password=sql;SID=abcd;ArraySize=100000;ProcResults=y;enableMSDIC=n
```

使用 OLE DB 连接数据库

OLE DB 使用组件对象模型 (COM) 使应用程序可以使用各种来源的数据。

关系数据库是一类可通过 OLE DB 访问的数据源。

本节介绍在下面的环境中，如何使用 OLE DB 连接到 Sybase IQ 数据库：

- Sybase PowerBuilder® 可以访问 OLE DB 数据源，您可以将 Sybase IQ 用作 PowerBuilder OLE DB 数据库配置文件。
- Microsoft ActiveX 数据对象 (ActiveX Data Objects, 简称 ADO) 为 OLE DB 数据源提供了编程接口。您可以通过 Microsoft Visual Basic 之类的编程工具访问 Sybase IQ。

OLE DB 需要 Windows 客户端。但是，使用 OLE DB 既可以访问 Windows 服务器又可以访问 UNIX 服务器。

本节介绍如何在 Sybase PowerBuilder 和 Microsoft ADO 环境 (如 Visual Basic) 中使用 OLE DB。本文档并未完整讲述如何使用 ADO 或 OLE DB 进行编程。开发主题的主要信息来源是开发工具的文档。

有关 OLE DB 的详细信息，请参见“SQL Anywhere Server - 编程” > “OLE DB 和 ADO 开发”。

注意： 此参考指向 SQL Anywhere 文档。

注意： Sybase IQ 支持的某些与 OLE DB 一起使用的功能与 SQL Anywhere 支持的不同：

- Sybase IQ 不支持 Windows CE。
 - Sybase IQ 不支持通过游标进行远程更新。
 - Sybase IQ 支持动态 (动态滚动)、静态 (不敏感) 和仅向前 (无滚动) 游标，但不支持键设置 (滚动) 游标。
 - 在 Sybase IQ 中，无论如何指定，隔离级别始终为 3。
-

OLE DB 提供程序

您希望访问的每类数据源都需要一个 *OLE DB 提供程序*。

每个提供程序都是一个动态链接库。您可以使用两个 OLE DB 提供程序来访问 Sybase IQ：

- Sybase SQL Anywhere OLE DB 提供程序。SQL Anywhere OLE DB 提供程序作为 OLE DB 数据源提供对 Sybase IQ 的访问，而无需使用 ODBC 组件。此提供程序的简称是 **SAOLEDB**。
SAOLEDB 提供程序在安装过程中自行注册。注册过程包括在注册表的 COM 部分创建注册表条目，以便 ADO 可以在 **SAOLEDB** 提供程序被调用时找到 DLL。如果您更改 DLL 的位置，则必须重新注册。
如果使用 SQL Anywhere OLE DB 提供程序，则在部署中不需要 ODBC。
有关 OLE DB 提供程序的详细信息，请参见“SQL Anywhere Server - 编程”。
- 针对 ODBC 的 Microsoft OLE DB 提供程序。Microsoft 提供了一个简称为 **MSDASQL** 的 OLE DB 提供程序。
MSDASQL 提供程序使 ODBC 数据源以 OLE DB 数据源的形式出现。它需要使用 SQL Anywhere ODBC 驱动程序。

从 ADO 连接

ADO 是一种面向对象的编程接口。在 ADO 中，**Connection** 对象代表与数据源建立的唯一会话。

您可以使用以下 **Connection** 对象的特性来启动连接：

- 用于保存提供程序名称的 **Provider** 属性。如果您不提供 **Provider** 名称，ADO 将使用 **MSDASQL** 提供程序。
- 用于保存连接字符串的 **ConnectionString** 属性。该属性保存 Sybase IQ 连接字符串，其使用方式与 ODBC 驱动程序相同。您可以提供 ODBC 数据源名称，也可以显式提供 **UserID**、**Password**、**DatabaseName** 和其它参数，正如在其它连接字符串中一样。
- **Open** 方法用于启动连接。

有关 ADO 的详细信息，包括与数据库连接的示例代码，请参见“SQL Anywhere Server - 编程” > “OLE DB 和 ADO 开发”。

注意： 此参考指向 SQL Anywhere 文档。

注意： 要运行“SQL Anywhere Server - 编程”中的示例，请替换数据源“Sybase IQ Demo”和示例目录 %ALLUSERSPROFILE%\All Users\SybaseIQ\samples\SQLAnywhere\VBSampler\vbsampler.sln。安装 Sybase IQ 会自动注册相应的驱动程序，但必须既注册 32 位驱动程序又注册 64 位驱动程序，才能在 64 位 Windows 计算机上基于 OLE DB 开发和部署应用程序。在运行 regsvr32 时指定 dboledb12.dll 和 dboledba12.dll 的完整路径。

另请参见

- 连接参数（第 77 页）

从其它数据库连接

您可以从 Adaptive Server Enterprise 以外部数据源形式访问 Sybase IQ 表中的数据。要利用此功能，请使用组件集成服务 (CIS) 接口，客户端可以通过该接口使用分布式异构源中的数据。

CIS 准备就绪后，请在 Adaptive Server Enterprise 中定义表示 Sybase IQ 表的“代理表”。然后，您可以从 Adaptive Server Enterprise 查询代理表。请参见《Adaptive Server Enterprise 组件集成服务 用户指南》。

CIS 和 Sybase IQ 提供了其它多种方式来连接到其它数据库和共享数据，以使用户应用程序可通过公用接口访问整个数据仓库。使用 CIS，您可以：

- 从 Sybase IQ 访问 Adaptive Server Enterprise 数据库中的数据。只有某些平台才支持此功能。请参见针对您所用平台的《安装和配置指南》。
- 访问其它数据库服务器上 Sybase IQ 和 SQL Anywhere 数据库中的数据。
- 访问其它外部数据源，包括其他供应商的关系数据库、Excel 电子表格数据和文本文件。
- 连接不同 Sybase IQ 数据库中的表。

请参见《系统管理指南第二卷》>“访问远程数据”。

另请参见

- 匹配 Adaptive Server Enterprise 数据类型（第 270 页）

避免在 UNIX 上发生端口号冲突

更新配置文件以避免端口号冲突。

1. 在 \$IQDIR15/scripts/default.cfg 中添加带有未使用的端口号的下面一行，例如：

```
-x tcpip{port=4444}
```

2. 在每个配置文件中查找端口号定义。例如，/usr/summers/mydemo/iqdemo.cfg 包含下面一行：

```
-x tcpip{port=2638}
```

3. 编辑该行，并将缺省端口号替换为新端口号，例如：

```
-x tcpip{port=4444}
```

4. 完成后，请保存每个文件。

如果 SQL Anywhere 与 Sybase IQ 位于相同子网中，服务器名必须唯一。

如何测试连接

提供 **dbping** 命令行实用程序是为了帮助排除连接故障。特别是，您可以使用它来测试具有特定名称的服务器在网络上是否可用。

dbping 实用程序将连接字符串视为命令行选项，但是在缺省情况下只使用那些查找服务器时需要的部分。它不会尝试启动服务器。

下面一行测试是否可通过 **TCP/IP** 连接使用名为 **Ciaran** 的服务器：

```
dbping -c "eng=Ciaran;CommLinks=tcpip"
```

下面的命令测试是否可在当前计算机上使用缺省服务器：

```
dbping
```

另请参见

- 安全性问题：不受限制的数据库访问（第 71 页）
- 集成登录（第 69 页）
- Logfile 连接参数 [LOG]（第 98 页）

集成登录

集成登录功能允许您针对数据库连接及操作系统和/或网络登录，维护单个用户 **ID** 和口令。

支持的操作系统

仅 **Windows** 服务器提供了集成登录功能。支持的 **Windows** 平台上的客户端可以使用集成登录连接到运行 **Windows** 的网络服务器。

集成登录的优点

集成登录是从一个或多个 **Windows** 用户配置文件映射到数据库中现有的用户。如果一名用户已经成功地确保了其用户配置文件的安全，并且已经登录到其计算机上，则这名用户不用另外提供用户 **ID** 或口令，即可连接到数据库。

要完成此操作，数据库必须能使用集成登录，且必须在用于登录到计算机和/或网络的用户配置文件和数据库用户之间建立一个映射。

使用集成登录对于用户更加方便，而且允许针对数据库和网络安全使用单个安全系统。它的优点包括：

- 在使用集成登录连接到数据库时，用户无需输入用户 **ID** 或口令。
- 如果您使用集成登录，用户鉴定是由操作系统而非数据库完成的：单个系统用于数据库安全以及计算机或网络安全。
- 多个用户配置文件可映射到单个数据库用户 **ID**。

- 用于登录到 Windows 计算机的名称和口令不必与数据库用户 ID 和口令一致。

警告! 集成登录提供了单个安全系统的方便性，但是它在安全性方面涉及到一些值得引起关注的内容，数据库管理员应当熟悉这些内容。

另请参见

- 如何测试连接（第 69 页）
- 安全性问题：不受限制的数据库访问（第 71 页）
- Logfile 连接参数 [LOG]（第 98 页）

使用集成登录

为了通过集成登录进行连接，必须执行几个步骤。

1. 通过将 LOGIN_MODE 数据库选项的值设置为 **Mixed** 或 **Integrated**（该选项不区分大小写）来代替缺省值 **Standard**，可以在数据库中启用集成登录功能。该步骤需要 DBA 权限。
2. 在用户配置文件和现有的数据库用户之间创建集成登录映射。使用 SQL 语句可以实现这一点。
3. 按照触发集成登录功能的方法从客户端应用程序建立连接。

以下各节介绍上述每个步骤。

启用集成登录功能

LOGIN_MODE 数据库选项确定是否启用集成登录功能。因为数据库选项只应用于它们所在的数据库，所以，不同的数据库即使是在同一服务器内装载和运行，也可以有不同的集成登录设置。

LOGIN_MODE 数据库选项接受下面的三个值之一（不区分大小写）。

- 这是缺省设置，不允许集成登录。如果尝试集成登录连接，则会发生错误。
- 此设置允许集成登录和标准登录。
- 此设置表示到数据库的所有登录都必须使用集成登录来完成。

警告! 如果将 LOGIN_MODE 数据库选项设置为 **Integrated**，则只有那些已被授予集成登录映射的用户才能进行连接。试图使用用户 ID 和口令进行连接时产生错误。只有具有 DBA 权限（完全的管理权限）的用户不受此限制。

在 Sybase Central 中创建集成登录

用户配置文件只能映射到现有的数据库用户 ID。当该数据库用户 ID 从数据库中删除时，基于该数据库用户 ID 的所有集成登录映射都将被自动删除。

1. 以具有 DBA 权限的用户身份连接到数据库。
2. 右键单击数据库的“登录映射”文件夹，选择“登录映射”。此时将显示“集成登录”向导。

3. 在该向导的第一页上，输入要为其创建集成登录的系统（计算机）用户的名称。您可以从列表中选择名称，也可以输入一个名称。

此外，选择该用户映射到的数据库用户 ID。该向导显示可用的数据库用户。您必须从中选择一个。不能添加新的数据库用户 ID。

4. 请按照向导中的其余说明进行操作。

在 SQL 中创建集成登录

用户配置文件只能映射到现有的数据库用户 ID。当该数据库用户 ID 从数据库中删除时，基于该数据库用户 ID 的所有集成登录映射都将被自动删除。

以下 SQL 语句允许 Windows 用户 `fran_whitney` 和 `matthew_cobb` 以用户 DBA 身份登录到数据库，而不必知道或者提供 DBA 用户 ID 或口令。

```
GRANT INTEGRATED LOGIN
TO fran_whitney, matthew_cobb
AS USER DBA
```

有关详细信息，请参见《参考：语句和选项》>“SQL 语句”>“GRANT 语句”。

撤消集成登录权限 (SQL)

您可以使用 `Interactive SQL` 删除集成登录映射。

1. 使用 DBA 权限连接到数据库。
2. 执行 `REVOKE INTEGRATED LOGIN FROM` 语句。

从客户端应用程序连接

客户端应用程序可以按照下列方法之一使用集成登录连接到数据库：

- 将连接参数列表中的 `INTEGRATED` 参数设置为 **yes**。
- 在连接字符串或连接对话框中既不指定用户 ID 也不指定口令。此方法仅适用于嵌入式 SQL 应用程序，包括 `Sybase IQ` 管理实用程序。

如果在连接字符串中指定 `INTEGRATED=yes`，则尝试进行集成登录。如果连接尝试失败，且 `LOGIN_MODE` 数据库选项设置为 `Mixed`，则服务器会尝试进行标准登录。

如果在不提供用户 ID 或口令的情况下试图连接到数据库，则试图进行集成登录。连接尝试的成败取决于当前的用户配置文件名是否与数据库中的集成登录映射相匹配。

安全性问题：不受限制的数据库访问

集成登录功能使用 Windows 的登录控制系统来工作，而不会使用 `Sybase IQ` 用于控制数据库访问的系统。

实质上，如果您可以登录到运行数据库的计算机，并且满足了本章中概述的其它条件，您就能通过数据库安全性机制。

如果用户以“`dsmith`”身份成功登录到 Windows 服务器，只要您有集成登录映射或者缺省的集成登录用户 ID，您就不必进一步验证标识即可连接到数据库。

在使用集成登录时，数据库管理员应当特别考虑 Windows 实施登录安全的方法，以禁止对数据库进行不必要的访问。

特别是，一定要注意，在缺省情况下，在安装 Windows Workstation 或 Server 时创建并启用了“Guest”用户配置文件。

警告！ 如果启用 Guest 用户配置文件，可允许对由该服务器运行的数据库进行不受限制的访问。

如果 Guest 用户配置文件处于启用状态且口令为空，则登录到该服务器的任何尝试都将成功。用户配置文件无需存在于服务器上，或者所提供的登录 ID 无需具有域登录权限。准确地说，任何用户都可以使用任何登录 ID 和任何口令登录到服务器：缺省情况下他们会登录到 Guest 用户配置文件。

这对于在启用集成登录功能的情况下连接到数据库有着重要的含义。

请考虑下面的情况：假定运行数据库的 Windows 服务器有一个用空口令启用的“Guest”用户配置文件。

- 在用户 dsmith 和数据库用户 DBA 之间存在集成登录映射。用户 dsmith 使用正确的登录 ID 和口令连接服务器时，她将以 DBA 身份（具有全部管理权限的用户）连接到数据库。
- 但是，对于尝试以“dsmith”身份连接到服务器的任何其他用户，因为 Windows 将该连接尝试缺省设置为“Guest”用户配置文件，所以无论他们提供什么口令，都将成功登录到服务器。在使用“dsmith”登录 ID 成功登录到服务器之后，未经授权的用户可以使用集成登录映射，以 DBA 身份成功连接到数据库。

注意：为了安全而禁用“Guest”用户配置文件。最安全的集成登录策略是在运行 Sybase IQ 数据库的所有 Windows 计算机上禁用“Guest”。这可以通过使用 Windows 用户管理器实用程序来完成。

另请参见

- 如何测试连接（第 69 页）
- 集成登录（第 69 页）
- Logfile 连接参数 [LOG]（第 98 页）

临时公共选项可以提高安全性

如果使用下面的 SQL 语句将给定数据库的 LOGIN_MODE 选项的值设置为 **Mixed** 或 **Integrated**，将对数据库永久启用集成登录。

```
SET OPTION Public.LOGIN_MODE = Mixed
```

如果关闭并重新启动该数据库，则该选项的值将保持不变，集成登录仍处于启用状态。

如果临时更改 LOGIN_MODE 选项，将仍允许用户通过集成登录进行访问。下面的语句将临时更改该选项的值：

```
SET TEMPORARY OPTION "Public".LOGIN_MODE = Mixed
```

如果永久选项值为 **Standard**，数据库将在关闭时恢复为该值。

由于启用集成登录意味着该数据库要依赖运行它的操作系统的安全，所以，可将设置临时公共选项视为数据库访问的附加安全措施。如果关闭该数据库并将其复制到另一台计算机（如用户的计算机），则对于数据库的访问将恢复到 **Sybase IQ** 安全模型，而非在其中复制数据库的计算机操作系统的安全模型。

请参见《参考：语句和选项》>“SQL 语句”>“SET OPTION 语句”。

集成登录的网络方面

如果该数据库位于网络服务器上，则对于要使用的集成登录必须满足下列两个条件之一：

- 用于集成登录连接尝试的用户配置文件必须同时存在于本地计算机和服务器上。而且，这两台计算机用户配置文件的名称和口令都必须相同。
例如，当用户 **jsmith** 尝试使用集成登录连接到在网络服务器上装载的数据库时，在本地计算机和运行该数据库的应用程序服务器上必须同时存在相同的用户配置文件名和口令。必须允许 **jsmith** 登录到本地计算机和运行网络服务器的服务器。
- 如果网络访问由 **Microsoft** 域来控制，试图进行集成登录的用户必须对域控制器服务器拥有域权限且必须登录到网络中。网络服务器上并不需要与本地计算机上的用户配置文件相匹配的用户配置文件。

缺省集成登录用户

您可以创建缺省的集成登录用户 **ID**，这样，即使对于当前正在使用的用户配置文件不存在集成登录映射，也允许通过集成登录成功进行连接。

例如，如果对于用户配置文件 **JSMITH** 不存在集成登录映射，则当 **JSMITH** 是正在使用的用户配置文件时，集成登录连接尝试通常会失败。

但是，如果您在数据库中创建一个名为 **Guest** 的用户 **ID**，则即使没有显式地为用户配置文件 **JSMITH** 建立集成登录映射，集成登录也将成功映射到 **Guest** 用户 **ID**。

缺省集成登录用户允许任何尝试集成登录的用户成功连接到包含名为 **Guest** 的用户 **ID** 的数据库。向新连接的用户授予的权限由向 **Guest** 用户 **ID** 授予的权限确定。

连接归集

连接归集可以提高建立到数据库服务器的多个短暂连接的应用程序的性能。

如果为某个应用程序启用连接归集，则在该应用程序断开连接时，系统将自动高速缓存连接，并且可能会在该应用程序重新连接时重新使用该连接。

请参见“SQL Anywhere Server - 数据库管理”>“启动并连接到数据库”>“SQL Anywhere 数据库连接”>“SQL Anywhere 连接归集”。

临时连接

临时连接执行运行备份或初始化数据库之类的操作。

有关临时连接的信息，请使用系统过程 `sa_conn_info` 或 `sa_conn_list`。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “SQL Anywhere 数据库连接” > “临时连接”。

注意： 此参考指向 SQL Anywhere 文档。

逻辑服务器配置

在 Multiplex 中，您只能使用逻辑服务器访问服务器。

由于 Multiplex 中可能出现故障或不合成员节点等原因，因此并非所有成员节点在任何时候对其逻辑服务器都可用。在任何给定时间，有效的逻辑服务器配置均表示逻辑服务器的当前动态构成，包括实际提供可用于逻辑服务器的所有成员节点。有效的逻辑服务器配置实质上是静态逻辑服务器配置和 Multiplex 动态状态的一个函数。

注意： 只有 Multiplex 配置才支持逻辑服务器。有关内置逻辑服务器和逻辑服务器策略的信息可保留在 Simplex 环境的目录中，但不会使用。

在 Simplex 中连接

在 Simplex 中，连接不受逻辑服务器分配的登录策略设置的影响。

连接没有逻辑服务器环境。登录策略选项“locked”的基础设置在服务器接受连接之前应用。

Multiplex 中的连接

在 Multiplex 中，登录策略控制对逻辑服务器的访问。登录策略的所有用户只能访问属于分配的逻辑服务器的有效成员的那些 Multiplex 服务器。

当您连接到 Multiplex 服务器时，系统将根据用户登录策略和当前节点的有效逻辑服务器分配来确定连接的逻辑服务器环境。

另请参见

- 连接的逻辑服务器环境（第 74 页）

连接的逻辑服务器环境

在建立用户连接时，用户的登录策略和当前的节点可确定连接的逻辑服务器环境。

- 如果针对用户登录策略的有效逻辑服务器分配是一个或多个逻辑服务器，则基于当前节点在某个指定逻辑服务器中的明确成员关系来确定连接的逻辑服务器环境。

注意： 给定登录策略允许访问的逻辑服务器没有重叠的成员关系。

- 如果当前的节点并非分配给用户登录策略的任何逻辑服务器的成员，则连接将失败。
- 另外，如果用户登录策略的有效逻辑服务器分配设置为 **NONE**，则连接也将失败。

如何终止连接

在某些情况下，必须终止连接。

在下列情况下，连接将终止：

- 在 **Interactive SQL** 或嵌入式 **SQL** 中，用户或应用程序针对当前连接、指定连接或该应用程序的所有连接发出显式 **DISCONNECT** 语句
- 在 **Interactive SQL** 中，用户选择“SQL” > “断开连接”
- 在 **Sybase Central** 中，用户选择“连接” > “断开连接”
- 关闭具有活动连接的应用程序，例如 **Interactive SQL** 或 **Sybase Central**

具有 **DBA** 权限的用户还可以删除特定连接：

- 在 **Interactive SQL** 或嵌入式 **SQL** 中，通过发出 **DROP CONNECTION** 语句
- 在 **Sybase Central** 中，通过访问数据库的“用户和组”文件夹

连接记录

缺省情况下，用户每次连接数据库或断开数据库连接时，`.iqmsg` 日志将记录此操作。

使用数据库选项 **LOG_CONNECT** 可以控制用户对连接和断开连接的记录。如果在用户连接时禁用连接记录，并在用户断开连接前启用连接记录，消息日志将显示用户断开连接，而不会显示连接。

有关详细信息，请参见《参考：语句和选项》>“数据库选项”>“按字母顺序排列的选项列表”>“**LOG_CONNECT** 选项”。

连接和通信参数

连接参数用于建立和描述从客户端应用程序到数据库的连接。

另请参见

- 连接学习路线图（第 42 页）
- 简单连接示例（第 48 页）
- Sybase Central 或 Interactive SQL 连接（第 45 页）
- 连接字符串和字符集（第 367 页）

连接参数

连接参数包括在连接字符串中。

可以在以下位置输入连接参数：

- 在应用程序的连接字符串中
- 在 ODBC 数据源中
- 在“Sybase IQ 连接”对话框中

在每个参数名称后的中括号中，显示参数的简写形式。在连接命令中您可以将该简写形式作为缩写。

在 Windows 操作系统中，“ODBC 配置”对话框和“Sybase IQ 连接”对话框共用相同的格式。有些参数与这些对话框中的复选框和字段相对应，而有些参数则可输入到“高级”选项卡上的文本框中。

用法说明

连接参数不区分大小写。

每个连接参数的用法均描述了使用该参数的环境。常见的用法条目包括：

- 嵌入式数据库 - 当 Sybase IQ 用作嵌入式数据库时，连接会启动服务器并装载数据库。当该应用程序从该数据库断开连接时，该数据库就会被卸载，而且服务器会停止运行。
- 网络服务器 - 当 Sybase IQ 用作网络服务器时，客户端应用程序必须查找已在网络上运行的服务器并连接到数据库。

您可以使用 **dbping** 实用程序来测试连接字符串。有关示例，请参见《实用程序指南》>“dbping 数据库管理实用程序”。

布尔（真或假）参数为真时其值为 YES、ON、1 或 TRUE，为假时其值为 NO、OFF、0 或 FALSE。

由接口库使用的连接参数可从下列位置（按优先级顺序）获取：

- 连接字符串
- **SQLCONNECT** 环境变量
- 数据源

服务器名必须由 **ASCII** 字符集中值为 1 到 127 之间的字符组成。对于其它参数没有这种限制。

下列规则控制参数的优先级：

- 连接字符串中的条目的读取顺序是从左到右。如果对同一个参数指定了多次，则使用字符串中的最后一个。
- 如果字符串包含 **DSN** 或 **FILEDSN** 条目，则从配置文件中读取配置文件，如果尚未设置该文件中的条目，则使用这些条目。例如，如果连接字符串包含一个数据源名称并显式设置数据源中包含的一些参数，则在发生冲突时使用显式参数。

另请参见

- 从 ADO 连接（第 67 页）

AppInfo 连接参数 [Appinfo]

帮助管理员从数据库服务器识别特定客户端连接的源。

用法

任何位置

缺省值

空字符串

描述

此连接参数会从嵌入式 **SQL**、**ODBC** 或 **OLE DB** 客户端以及 **UNIX** 上的 **DBISQLC** 发送到数据库服务器。它在 **Open Client** 或 **jConnect** 应用程序（例如 **Java** 版本的 **Interactive SQL [dbisql]** 或 **Sybase Central**）中不可用。

它由生成的字符串组成，用来保存有关客户端进程的信息，如客户端计算机的 **IP** 地址、客户端进程运行时所用的操作系统等。该字符串在数据库服务器中与连接相关联，您可以使用下面的语句检索它：

```
SELECT connection_property( 'AppInfo' )
```

客户端还可以指定自己的字符串，该字符串会附加到生成的字符串。**AppInfo** 属性字符串是一系列由分号分隔的 **key=value** 对。有效键如下所示：

- **API - DBLIB**、**ODBC**、**OLEDB** 或 **ADO.NET**（iAnywhere JDBC 驱动程序返回 **ODBC**）
- **APPINFO** - 如果在连接字符串中指定了 **AppInfo**，则输入该字符串

- EXE - 客户端可执行文件的名称 (仅限 Windows)
- HOST - 客户端计算机的主机名
- IP - 客户端计算机的 IP 地址
- OS - 操作系统名称和版本号 (例如, Sun Solaris 2.9)
- PID - 客户端的进程 ID
- THREAD - 客户端的线程 ID
- VERSION - 正在使用的连接协议的版本, 包括主要值、次要值和内部版本号 (例如 9.0.1.1549)
- TIMEZONEADJUSTMENT - 必须添加到协调通用时间 (UTC) 中以显示连接本地时间的分钟数

如果您在客户端连接参数中指定调试日志文件, 则 APPINFO 字符串会添加到该文件中。

示例

- 从 Interactive SQL 连接到演示数据库 (缺省情况下使用 iAnywhere JDBC 驱动程序) :

```
dbisql -c "uid=DBA;pwd=SQL;dbf=C:\Program Files\Sybase
\IQ-15_4\demo\iqdemo.db"
```

查看应用程序信息:

```
SELECT connection_property('AppInfo')
```

结果如下所示 (在单个字符串中) :

```
IP=10.187.99.116;
HOST=machine-name;
OSUSER=smith;
OS='Windows XP Build 2600 Service Pack 3';
EXE='C:\\WINDOWS\\Sybase\\Shared\\Sybase Central
6.1.0\\win32\\scjview.exe';
PID=0x12a4;
THREAD=0x58c;
VERSION=12.0.1.6554;
API=iAnywhereJDBC;
TIMEZONEADJUSTMENT=-240
```

- 从 Interactive SQL 连接到缺省数据库, 并将您自己的信息附加到 AppInfo 属性中:

```
dbisql -c "uid=DBA;pwd=SQL;app=ISQL connection"
```

查看应用程序信息:

```
SELECT connection_property('appinfo')
```

结果如下所示 (在单个字符串中) :

```
HOST=machine-name;
OS=Sun_Solaris 2.10;
PID=0x10e;
THREAD=0xe1;
VERSION=12.0.1.6554;
```

```
TIMEZONEADJUSTMENT=-300  
APPINFO=ISQL connection
```

AutoPreCommit 连接参数 [AutoPreCommit]

强制每个语句在执行前提交。

用法

ODBC

缺省值

NO

描述

缺省情况下，语句在执行后发出 COMMIT。如果 AutoPreCommit = 'yes'，则在每个 select 语句前发出 commit 语句，以使用户可以始终看到所有数据库对象的最新版本。

示例

您可以将 AutoPreCommit 选项设置为 YES (Y) 以启用在执行前提交，或设置为 NO (N) 以将其禁用。在 .odbc.ini 文件中设置此选项或在“连接”对话框的“高级”选项卡上设置此选项。

例如，以下示例会使每个语句在执行前提交：

```
[Sample DSN]  
DatabaseFile=c:\Program Files\Sybase\IQ-15_4\demo\iqdemo.db  
AutoPreCommit=Y  
UserID=DBA  
Password=SQL
```

AutoStart 连接参数 [Astart]

防止在找不到连接时启动数据库服务器。

用法

任何位置

缺省值

是

描述

缺省情况下，如果在尝试连接期间找不到服务器，并且指定了数据库文件，则启动同一台计算机上的数据库服务器。您可以通过将连接字符串中的 AutoStart 参数设置为 OFF 来禁用此行为。

示例

- 下面的数据源片段会防止在没有找到任何网络服务器的情况下启动数据库服务器：

```
[My Demo Database]
DatabaseFile=c:\sybase\IQ-15_4\demo\iqdemo.db
Autostart=No
UserID=DBA
ENG=network_server
```

AutoStop 连接参数 [Astop]

防止当数据库不再存在打开的连接时立即卸载数据库。

用法

嵌入式数据库

缺省值

是

描述

缺省情况下，从连接字符串启动的任何服务器都会在不再有到它的连接时停止。同样，从该连接字符串装载的任何数据库都将在不再有到它的连接时立即卸载。此行为等效于 **Autostop=Yes**。

如果提供 **Autostop=No**，则当从连接中启动的任何数据库不再存在任何连接时，都不会被卸载。因此，数据库服务器也将不会关闭。

只有在连接到当前未运行的数据库时，才使用 **AutoStop** 参数。如果已装载数据库，则忽略该参数。

示例

下面的 Windows 连接配置文件会防止在删除连接时卸载数据库：

```
[Sample Embedded Database]
DatabaseFile=c:\sybase\IQ-15_4\demo\iqdemo.db
Autostop=No
UserID=DBA
```

CharSet 连接参数 [CS]

指定要在该连接上使用的字符集。

用法

任何位置

缺省值

区域设置字符集。

描述

如果为 **CharSet** 提供一个值，则指定的字符集用于当前的连接。**CharSet=none** 可以禁用连接的字符集转换。

卸载数据时，可以使用 **CharSet** 连接参数指定字符集。

为避免有损字符集转换，Sybase 建议在使用 Unicode 客户端 API 时不要设置 **CharSet** 连接参数。Unicode 客户端 API 包括 ADO.NET、OLE DB 和 iAnywhere JDBC 驱动程序。当使用 **wide** (Unicode) 函数时，ODBC 同样也是 Unicode 客户端 API。

另请参见

- **Language** 连接参数 [**LANG**] (第 96 页)
- 连接参数的工作原理 (第 43 页)
- 区域设置信息 (第 369 页)
- 设置区域设置 (第 369 页)

CompressionThreshold 连接参数 [**COMP**TH]

控制供数据库服务器写入有关 **Web** 请求的信息的日志文件的最大大小。

CompressionThreshold 参数是一个 SQL Anywhere 协议选项。请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “连接参数” > “**CompressionThreshold** [**COMP**TH]”。

注意： 此参考指向 SQL Anywhere 文档。

CommBufferSize 连接参数 [**CBS**ize]

设置通信包的最大大小（以字节为单位）。使用 **k** 来指定千字节的单位。

用法

任何位置

值

整数 [**k**]

缺省值

如果未设置 **CommBufferSize** 值，则 **CommBufferSize** 由服务器上的设置（缺省值为 1460 字节）控制。

描述

CommBufferSize 参数指定通信包的大小，以字节为单位。**CommBufferSize** 的最小值为 500，最大值为 16000。

协议堆栈设置网络上包的最大大小。如果您设置的 **CommBufferSize** 值大于网络所允许的值，则网络软件会拆分最大缓冲区。您应当将缓冲区大小设置得比网络所允许的值小一些，这是因为网络软件在通过网络发送信息之前，可能会向每个缓冲区添加信息。缺省值 1460 允许以太网包在使用 **TCP/IP** 时被完全填充。

较大的包大小可以提高多行提取和较大行提取的性能，但同时会增加客户端和服务器的内存使用量。

如果未在客户端指定 **CommBufferSize**，则连接使用服务器的缓冲区大小。如果在客户端指定 **CommBufferSize**，则连接使用该 **CommBufferSize** 值。

使用 **-p** 数据库服务器选项设置 **CommBufferSize** 时，会让所有未指定自己的 **CommBufferSize** 的客户端都使用 **-p** 数据库服务器选项指定的大小。

示例

将缓冲区大小设置为 600 字节：

```
...  
CommBufferSize=600  
...
```

另外，您可以通过在连接窗口的“**Network**”选项卡的“**Buffer size**”文本框中输入参数的值来设置此参数。

CommBufferSpace 连接参数 [CBSpace]

指定启动时为网络缓冲区分配的空间量（以千字节为单位）。

用法

任何位置

值

整数

缺省值

10

描述

指定启动时为网络缓冲区分配的空间量（以千字节为单位）。

该值是适用于所有连接的全局设置。

示例

以下配置文件片段指示网络库在启动时为网络缓冲区分配 200 KB。

```
...  
CBSpace=200  
...
```

您可以通过在连接窗口的“**Network**”选项卡的“**Buffer space**”文本框中输入参数的值来设置此参数。

CommLinks 连接参数 [Links]

指定客户端网络通信链接。

用法
任何位置

值
String

缺省值
仅使用共享内存通信链接进行连接。

描述
如果指定 `CommLinks=ALL`，则客户端会使用所有可用通信协议来搜索服务器。由于指定 `CommLinks=ALL` 时可能会影响系统性能，因此，只有在不知道使用哪个协议时才使用此设置。

如果在 `CommLinks (LINKS)` 连接参数中指定一个或多个协议，则客户端将按指定的顺序使用指定的通信协议搜索网络数据库服务器。如果连接无法使用指定的协议进行连接，则会出现一个连接错误并且该连接尝试会中止，即使列表中仍有可尝试的协议也是如此。

如果未指定 `CommLinks (LINKS)` 连接参数，则客户端只搜索当前计算机上的服务器，并且只使用共享内存连接。这是缺省行为，该行为等效于 `CommLinks=ShMem`。共享内存协议用于在同一台计算机的同一个操作系统下运行的客户端和服务器之间的通信。

`CommLinks` 参数的可用值如下所示：

- **SharedMemory (ShMem)** - 启动同一台计算机通信的共享内存协议。这是缺省设置。如果协议显示在协议列表中，则不管协议显示的顺序如何，客户端首先尝试使用共享内存。
- **ALL** - 首先使用共享内存协议尝试连接，然后使用所有其余可用的通信协议。如果不确定使用哪个协议，则使用该设置。
- **TCPIP** - 启动 TCP/IP 通信链接。所有操作系统都支持 TCP/IP。

其中的每个值都可以提供其它网络通信参数。

由于以下原因，您可能希望使用特定协议，而不是 **ALL**：

- 如果不启动不必要的网络链接，则网络库的启动会稍微快一些。
- 与数据库的连接速度可能会快些。
- 如果您希望通过提供其它网络通信参数来调整特定协议的广播行为，则必须显式指定链接。

可以为每个链接提供其它网络通信参数，从而调整链接的广播行为。

CommLinks 参数与数据库服务器的 **-x** 命令行开关相对应。缺省情况下，网络服务器启动所有可用的协议，这等效于 **-x ALL**。请参见《实用程序指南》中的 **-x list start_iq** 实用程序启动选项。

示例

- 以下连接字符串片段只启动 TCP/IP 协议：

```
CommLinks=tcPIP
```

- 下面的连接字符串片段启动共享内存协议并通过共享内存搜索数据库服务器。如果搜索失败，它会启动 TCP/IP 端口，除在直接的 TCP/IP 网络上搜索服务器外还搜索主机 **kangaroo**。

```
CommLinks=tcPIP(HOST=kangaroo),shmem
```

另请参见

- 网络通信参数（第 102 页）

ConnectionName 连接参数 [CON]

对连接进行命名，便于在多连接应用程序中切换到该连接。

用法

不可用于 ODBC

缺省值

无连接名称

描述

一个可选参数，为您所建立的特定连接提供名称。除非您要建立多个连接并在它们之间切换，否则不要指定该参数。

连接名称与数据源名称不同。

示例

连接，将连接命名为 FirstCon：

```
CON=FirstCon
```

ConnectionPool 连接参数 [CPOOL]

控制客户端连接归集的行为。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “连接参数” > “ConnectionPool (CPOOL) 连接参数”。

注意： 此参考指向 SQL Anywhere 文档。

PrefetchBuffer 连接参数 [PBUF]

允许用户设置最大内存量。

为缓冲行设置最大的内存量（以字节为单位）。

用法
任何位置

值
Integer [k | m]

缺省值
512 (KB)

描述

PrefetchBuffer 连接参数控制在客户端上分配的用于存储预取行的内存。该值是以字节为单位的，但您可以使用 **k** 或 **m** 来指定千字节或兆字节的单位。此连接参数接受 64KB 到 8MB 之间的值。在某些情况下，增加客户端从数据库服务器中预取的行数可以提高查询性能。可以使用 **PrefetchRows** 和 **PrefetchBuffer** 连接参数增加预取的行数。

增加 **PrefetchBuffer (PBUF)** 连接参数会增加用于缓冲 GET DATA 请求的内存量。对于一些处理许多 GET DATA (SQLGetData) 请求的应用程序来说，这可能会改善性能。

示例

要确定 PrefetchBuffer 内存限制是否减少预取的行数，请使用下面的连接字符串片段：

```
...prefetchrows=100;logfile=c:\ client.txt
```

要将内存限制增加到 256KB，请使用：

```
...prefetchrows=100;prefetchbuffer=256
```

另请参见

- PrefetchRows 连接参数 [PROWS]（第 99 页）

DatabaseFile 连接参数 [DBF]

指定在启动数据库时要装载或连接的目标数据库文件。

要连接到已在运行的数据库，请使用 **DatabaseName(DSN)** 参数。

现在，在特殊情况下 Sybase IQ 需要 DBF 参数和数据库文件名才能连接。

用法
嵌入式数据库

*值**String**缺省值*

无缺省设置。

描述

DatabaseFile (DBF) 连接参数用于装载并连接到尚未在数据库服务器上运行的特定数据库文件。

- 如果使用与 **DatabaseFile** 参数相同的名称装载数据库，但不带 **.db** 扩展名，则会与该数据库建立连接。
- 如果文件名不包含扩展名，则会查找名称为 **.db** 的文件。
- 文件的路径相对于数据库服务器的工作目录。如果您从命令提示符启动服务器，则工作目录是您在输入命令时所在的目录。如果您从图标或快捷方式启动服务器，则它是该图标或快捷方式中所指定的工作目录。建议提供完整路径和文件名。
- 如果同时指定了数据库文件和数据库名称，则忽略该数据库文件，且不用于尝试连接到已在运行的数据库。

您也可以使用 **UNC** 文件名。

警告! 数据库文件必须与数据库服务器位于同一台计算机上。管理位于网络驱动器上的数据库文件可能导致文件损坏。

示例

要装载并连接到演示数据库（在 Windows 上，安装在目录 **C:\Program Files\Sybase\IQ-15_4\demo** 中），请使用以下 **DBF** 参数：

```
DBF=C:\Program Files\Sybase\IQ-15_4\demo\iqdemo.db
```

另请参见

- **DatabaseName** 通信参数 [DBN]（第 107 页）
- **DatabaseName** 连接参数 [DBN]（第 87 页）

DatabaseName 连接参数 [DBN]

指定需要与之建立连接的已装载的数据库。用于连接到已在运行的数据库。

要连接到尚未运行的数据库，请使用 **DatabaseFile (DBF)** 参数。

用法

运行网络服务器

*值**String*

缺省值

无缺省设置。

描述

任何时候在服务器上启动数据库时，都会为它指派一个数据库名称，其方法是管理员使用 `-n` 选项，或者是服务器使用去掉了扩展名和路径的基本文件名。

如果要连接到的数据库已在运行，则应该指定数据库名称而不是数据库文件。

只有运行的数据库的名称与 `DatabaseName (DBN)` 参数中指定的名称匹配时，连接才会发生。

注意： 如果同时指定了数据库名称和数据库文件，则忽略该数据库文件，且不用于尝试连接到已在运行的数据库。

示例

- 要启动名为 `cities.db` 的数据库文件并将数据库重命名为 `Kitchener`，可以使用以下命令：

```
start_iq cities.db -n Kitchener
```

- 假定已运行上述命令，则可以如下所示成功地连接到正在运行的名为 `Kitchener` 的数据库：

```
DBN=Kitchener
```

- 另外，也可以使用下面的方法成功地连接到正在运行的名为 `Kitchener` 的数据库：

```
DBN=Kitchener;DBF=cities.db
```

- 但是，指定下面的参数将无法连接到名为 `Kitchener` 的数据库：

```
DBF=cities.db
```

另请参见

- `DatabaseFile` 连接参数 [DBF]（第 86 页）
- `DatabaseName` 通信参数 [DBN]（第 107 页）

DatabaseSwitches 连接参数 [DBS]

在启动数据库时提供特定于数据库的开关。

用法

在未装载数据库时连接到服务器。

缺省值

没有开关

描述

只有在连接到当前未运行的数据库时，才应提供 **DatabaseSwitches**。当服务器启动 **DatabaseFile** 指定的数据库时，该服务器将所提供的 **DatabaseSwitches** 用作命令行选项，从而确定该数据库的启动选项。

使用此参数只能提供数据库开关。必须使用 **START** 连接参数提供服务器开关。

示例

以下 UNIX 命令（全部输入在一行上）将连接到缺省数据库服务器，装载数据库文件 /IQ-15_4/demo/iqdemo.db（**DBF** 参数），将该数据库文件命名为 my_db（**DBS** 参数）并连接到具有该名称的数据库（**DBN** 参数）。

```
dbcollat -c "uid=DBA;pwd=SQL;dbf=/IQ-15_4/demo/iqdemo.db;
dbn=my_db;dbcs=-n my_db" /tmp/temp.col
```

另请参见

- StartLine 连接参数 [START]（第 101 页）

DataSourceName 连接参数 [DSN]

通知 ODBC 驱动程序管理器或嵌入式 SQL 库在 .odbc.ini 文件（在 UNIX 上）或 odbc.ini 文件或注册表（在 Windows 上）中的什么位置进行查找，从而找到 ODBC 数据源信息。

用法

任何位置。

缺省值

没有缺省数据源名称。

描述

对于 ODBC 应用程序来说，通常的做法是只向 ODBC 发送数据源名称。ODBC 驱动程序管理器和 ODBC 驱动程序查找包含其余连接参数的数据源。在 Sybase IQ 中，嵌入式 SQL 应用程序还可以使用 ODBC 数据源存储连接参数。

示例

下面的参数使用数据源名称：

```
DSN=Dynamo Demo
```

另请参见

- FileDataSourceName 连接参数 [FileDSN]（第 94 页）
- “ODBC”选项卡（第 60 页）

DBKEY 连接参数 [DBKEY]

用连接请求启动加密数据库。

用法

在数据库启动时。在连接到正在运行的数据库时不使用。

缺省值

缺省情况下，不对数据库加密。

描述

当您用连接请求启动加密数据库时，必须指定该参数。

示例

下面的配置文件片段表示使用加密密钥 “is!seCret” 连接到已在运行的名为 `marvin.db` 的强加密数据库。

```
...
UID=dba;PWD=sql;DBF=marvin.db;DBKEY='is!seCret'
...
```

DisableMultiRowFetch 连接参数 [DMRF]

关闭通过网络进行的多重记录读取。

用法

任何位置

缺省值

否

缺省情况下，在数据库服务器获取简单读取请求时，应用程序将要求提供额外行。您可以通过将该参数设置为 ON 来禁用该行为。

将 **DisableMultiRowFetch** 参数设置为 ON 等效于将 PREFETCH 选项设置为 OFF。

示例

- 以下连接字符串片段禁止预读：

```
DMRF=Yes
```

EngineName 连接参数 [ENG]

指定要连接到的正在运行的数据库服务器。这是 **ServerName** 的同义词。

用法

网络服务器

*值**String**缺省值*

缺省本地数据库服务器。

描述

您需要提供 **EngineName** 连接到网络服务器。在“连接”对话框和“ODBC 管理器”中，它是“**Server Name**”字段。

服务器名是按照客户端计算机的字符集解释的。建议不要在服务器名中使用多字节字符。

名称必须是有效的标识符。长服务器名将被截断为不同的长度，具体长度取决于协议。

有关数据库服务器命名限制，请参见《实用程序指南》。

协议	截断长度
UNIX 共享内存	31 字节
非 UNIX 共享内存	40 字节
TCP/IP	40 字节

示例

连接到名为 **Guelph** 的服务器：

```
ENG=Guelph
```

另请参见

- **ServerName** 连接参数 [ENG] (第 100 页)

EncryptedPassword 连接参数 [ENP]

提供口令，以加密形式存储在数据源中。

用法

任何位置。(DSN 和 FILEDSN 连接参数不支持详细形式的关键字。)

*值**String**缺省值*

无

描述

数据源以文件形式存储在磁盘上，或者存储在注册表中。在磁盘上存储口令可能会引发安全问题。因此，当您将口令输入到数据源中时，它会以加密形式存储。

如果同时指定了 **Password** 和 **EncryptedPassword**，则 **Password** 优先级较高。

另请参见

- **Password** 连接参数 [PWD]（第 99 页）

Encryption 连接参数 [ENC]

对在客户端应用程序和服务器之间发送的包进行加密。

用法

仅适用于 **ECC_TLS** (Certicom)、**RSA_TLS**、TCP/IP。

对于 **NONE** 或 **SIMPLE**，为任何位置。

值

String

缺省值

NONE

如果未设置 **Encryption** 值，则加密由服务器上的设置（缺省为未加密）控制。

描述

如果考虑网络包的安全性，则可以使用此参数。加密对性能有很轻微的影响。

Encryption(ENC) 连接参数接受以下参数：

- 接受未加密的通信包。该值等效于先前版本的 **Sybase IQ** 中的 **NO**。
- 接受用所有平台上和 12.6 版之前的 **Sybase IQ** 上支持的简单加密进行加密的通信包。该值等效于先前版本的 **Sybase IQ** 中的 **YES**。
- （以前称为 **Certicom**）接受使用 **Certicom** 加密技术加密的通信包。要使用该加密类型，服务器和客户端必须都运行于 **Solaris**、**Linux** 或所有支持的 **Windows** 操作系统上，该连接必须通过 **TCP/IP** 端口。除 **Solaris** 和 **Linux** 外，**UNIX** 平台不识别客户端或服务器 **Certicom** 参数。要鉴定服务器，**Certicom** 软件使用以下参数来检验服务器的证书值是否与您提供的有关客户端的任何值匹配：
 - 指定客户端用于鉴定服务器的认证文件。
 - 为组织字段指定值。服务器的值和客户端的值必须匹配。
 - 为组织单位字段指定值。服务器的值和客户端的值必须匹配。
 - 指定证书的公用名。服务器的值和客户端的值必须匹配。
- 接受使用 **RSA** 加密技术加密的通信包。要使用该加密类型，服务器和客户端必须都运行于 **Solaris**、**Linux** 或所有支持的 **Windows** 操作系统上，该连接必须通过 **TCP/IP** 端口。除 **Solaris** 和 **Linux** 外，**UNIX** 平台不识别客户端或服务器 **RSA_TLS** 参

数。要鉴定服务器，**Certicom** 软件使用以下参数来检验服务器的证书值是否与您提供的有关客户端的任何值匹配：

- 指定客户端用于鉴定服务器的认证文件。
- 为组织字段指定值。服务器的值和客户端的值必须匹配。
- 为组织单位字段指定值。服务器的值和客户端的值必须匹配。
- 指定证书的公用名。服务器的值和客户端的值必须匹配。

警告！ 示例证书仅可用于测试目的。示例证书无法为所部署的环境提供安全性，因为该证书和其相应的口令已通过 **Sybase** 软件广泛分发。要保护您的系统，必须创建自己的证书。

可以使用 **connection_property** 系统函数来检索当前连接的加密设置。该函数会返回以下三个值之一：**none**、**simple** 或 **Certicom**，具体取决于使用的加密类型。

另请参见

《实用程序指南》 > “start_iq 数据库服务器启动实用程序” 中的 **-ecencryption-options** 开关

“SQL Anywhere Server - 数据库管理” > “安全性” > “传输层 安全性”

注意： 此参考指向 **SQL Anywhere** 文档。

示例

- 下面的连接字符串片段使用 **Certicom** 加密和示例受托证书，通过 **TCP/IP** 链接连接到 **myeng** 数据库服务器：

```
"ENG=myeng; LINKS=tcPIP; Encryption=ECC_TLS
(trusted_certificates=sample.crt)"
```

- 下面的连接字符串片段使用 **RSA** 加密和示例受托证书，通过 **TCP/IP** 链接连接到 **myeng** 数据库服务器：

```
"ENG=myeng; LINKS=tcPIP; Encryption=RSA_TLS
(trusted_certificates=sample.crt)"
```

Escape 连接参数 [ESCAPE]

允许您指定返回表或列的列表时，由 **ODBC** 驱动程序生成的 **SQL** 语句的 **LIKE** 子句中使用的转义字符。

描述

只有 **Sybase IQ** 支持 **Escape** 参数。

缺省情况下，**ODBC** 驱动程序使用否定符号 (~) 作为转义字符，但某些应用程序假定转义字符是反斜杠 (\)。例如，以十六进制代码和符号存储的字符的缺省转义字符是反斜杠 (\)，因此 \x0A 表示换行符。

示例

要将感叹号用作转义字符，请输入：

```
ESCAPE='!'
```

FileDataSourceName 连接参数 [FileDSN]

通知 Client Library 有一个 ODBC 文件数据源保存有关要连接到的数据库的信息。

用法
任何位置

值
String

缺省值
没有缺省名。

描述
文件数据源中保存的信息与注册表中存储的 ODBC 数据源所保存的信息相同。文件数据源可轻松分发到最终用户，这样，就不必在每台计算机上重建连接信息。

ODBC 和嵌入式 SQL 应用程序都可以使用文件数据源。

示例
下面是在文件数据源中保存的数据源描述：

```
[Sample File Data Source]
ENG=iqdemo
DBA=DBA
PWD=SQL
```

另请参见

- DataSourceName 连接参数 [DSN] (第 89 页)
- “ODBC” 选项卡 (第 60 页)

Idle 连接参数 [IDLE]

Idle 连接参数确定连接超时的时间。

功能
指定连接的空闲超时期限。

用法
除 Tabular Data Stream™ (TDS) 连接和共享内存连接以外的任何位置。共享内存连接和 TDS 连接（包括 jConnect）忽略 Sybase IQ Idle (IDLE) 连接参数。

值
Integer

缺省值
-ti 的值

描述

Idle (IDLE) 连接参数只适用于当前连接。您可以将同一台服务器上的多个连接设置为不同的超时值。

如果未设置连接空闲超时值，则空闲超时值由服务器上的设置（由 **start_iq** 设置时缺省为 4400 分钟）控制。如果超时值之间有冲突，则无论是否指定服务器超时值，连接超时值都会取代任何服务器超时值。

对于空闲超时和活动超时（分别是 **-ti** 和 **-tl**）可以选择包括相关服务器命令行参数。

另请参见

《实用程序指南》> “start_iq 数据库服务器启动实用程序” 中的 **-ti** 服务器命令行选项

示例

- 以下连接字符串片段将此连接的超时值设置为 10 分钟：

```
"ENG=myeng;LINKS=tcPIP;IDLE=10"
```

Integrated 连接参数 [INT]

使用集成登录功能。

用法
任何位置

值
YES, NO

缺省值
NO

另请参见

有关详细信息，请参见《参考：语句和选项》> “数据库选项”> “按字母顺序排列的选项列表”> “LOGIN_MODE 选项”。

描述

Integrated 参数具有以下设置：

- 尝试进行集成登录。如果连接尝试失败，且 LOGIN_MODE 选项设置为 Mixed，则尝试进行标准登录。
- 这是缺省设置。不尝试进行集成登录。

对于要使用集成登录的客户端应用程序，服务器必须在将 LOGIN_MODE 数据库选项设置为 Mixed 或 Integrated 的情况下运行。

示例

下面的数据源片段使用集成登录:

```
INT=yes
```

Language 连接参数 [LANG]

指定连接的语言。

用法

任何位置

值

代表语言的双字母组合。例如，设置 **LANG=DE** 可将缺省语言设置为德语。

缺省值

由 **IQLANG** 环境变量或安装程序指定的（按顺序）语言。

描述

该连接参数为连接设定语言。来自服务器的任何错误或警告消息都以指定的语言来传送（假定服务器支持该语言）。

如果未指定语言，则使用缺省语言。缺省语言是由 **IQLANG** 环境变量或安装程序指定的（按顺序）语言。

该连接参数仅影响连接。从 **SQL Anywhere** 的不同工具和实用程序返回的消息采用缺省语言显示，而从服务器返回的消息则采用该连接的语言显示。

另请参见

- **CharSet** 连接参数 [CS]（第 81 页）
- 连接参数的工作原理（第 43 页）
- 区域设置信息（第 369 页）
- 设置区域设置（第 369 页）

LazyClose 连接参数 [LCLOSE]

导致 **CLOSE cursor-name** 数据库请求排队，然后和下一个数据库请求一起发送到服务器。每次关闭游标时，就无需进行网络请求。

用法

任何位置

值

YES, NO

缺省值

NO

描述

启用该参数时，游标实际上未关闭，直到有下一个数据库请求为止。当 `CLOSE cursor-name` 数据库请求处于排队状态时，任何隔离级别为 1 的游标稳定性锁仍旧适用于该游标。

在下列情况下，启用此选项可提高性能：

- 网络的等待时间较长
- 应用程序发送过多的打开和关闭游标请求

注意，以下情况很少发生：如果在 `CLOSE cursor-name` 数据库请求之后取消下一个请求，则可能会使游标在客户端上看起来处于关闭状态，但实际上它在服务器端上并没有关闭。如果随后尝试打开其它同名的游标，则此操作会失败。如果应用程序经常取消请求，则建议您不要使用 `LazyClose`。

LivenessTimeout 连接参数 [LTO]

在连接发生变化时控制连接的终止。

用法

采用 `TCP/IP` 通信协议的网络服务器。

除非线程 `UNIX` 应用程序外的所有平台。

值

Integer (以秒为单位)

缺省值

120

如果未设置 `LivenessTimeout` 值，则活动超时由服务器上的设置（缺省值为 120 秒）控制。

描述

活动包通过客户端/服务器 `TCP/IP` 通信协议定期发送，目的在于确认连接的完好性。如果客户端的运行时间达到了活动超时期限且没有检测到活动请求或响应包，则通信将断开。

如果连接在 `LivenessTimeout` 值的三分之一到三分之二这么长的一段时间内未发送任何数据包，则将发送活动数据包。

通信断开时，客户端计算机会忘记服务器的地址。客户端计算机会查找地址，直到下次有从该计算机到服务器的连接，从而删除到该服务器的所有当前连接。

当与服务器的连接超过 200 个时，服务器会根据指定的 **LivenessTimeout** 值自动计算一个更大的 **LivenessTimeout** 值。这使服务器可以更高效地处理大量连接。

另外，您可以通过在“ODBC 配置”对话框中“网络”选项卡的“LivenessTimeout”文本框中输入该参数的值来设置该参数。

示例

以下示例将活动超时值设置为 60 秒：

```
LTO=60
```

ConnectionPool 连接参数 [CPOOL]

控制客户端连接归集的行为。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “连接参数” > “ConnectionPool (CPOOL) 连接参数”。

注意： 此参考指向 SQL Anywhere 文档。

Logfile 连接参数 [LOG]

将客户端错误消息和调试消息发送到文件中。

用法

任何位置

值

String

描述

如果要将客户端错误消息和调试消息保存到文件中，请使用 **LogFile (LOG)** 参数。

如果文件名中包括路径，则它相对于客户端应用程序的当前工作目录。

LogFile (LOG) 连接参数是针对特定连接而言，因此，您可以从单个应用程序为不同的连接设置不同的 **LogFile** 参数。

示例

下面的命令行片段指定此连接的客户端消息应发送到该客户端的当前工作目录中的 `error.log` 文件：

```
...  
LogFile=error.log  
...
```

另请参见

- 如何测试连接（第 69 页）
- 安全性问题：不受限制的数据库访问（第 71 页）

- 集成登录（第 69 页）

NewPassword 连接参数 [NEWPWD]

NewPassword 连接参数允许用户在无需 DBA 干预的情况下更改口令，即使口令已过期也是如此。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “连接参数” > “NewPassword (NEWPWD) 连接参数”。

注意： 此参考指向 SQL Anywhere 文档。

Password 连接参数 [PWD]

为连接提供口令。

用法
任何位置

缺省值
不提供口令

描述
数据库的每个用户都有口令。必须提供口令，用户才可以连接数据库。缺省情况下，口令与数据具有相同的区分大小写设置。缺省情况下，IQ 数据库区分大小写。

不会对口令参数进行加密。如果将口令存储在连接配置文件中，应使用 **EncryptedPassword** 参数。 Sybase Central 和 Sybase IQ ODBC 配置工具均使用加密参数。

如果同时指定了 **Password** 和 **EncryptedPassword**，则 **Password** 优先级较高。

示例
以下连接字符串片段提供用户 ID DBA 和口令 SQL。

```
uid=DBA;pwd=SQL
```

另外，您可以在连接窗口的“User ID”和“Password”文本框中设置这些参数。

另请参见

- EncryptedPassword 连接参数 [ENP]（第 91 页）

PrefetchRows 连接参数 [PROWS]

PrefetchRows 连接参数可设置要在查询数据库时预取的最大行数。

用法
任何位置

缺省值

10 (当使用 .NET 数据提供程序时为 200)

描述

如果增加客户端可从数据库服务器预取的行数，则可以提高只提取相对位置为 0 或 1 (通过单行提取或宽提取) 的游标的性能。宽提取包括嵌入式 SQL 阵列提取和 ODBC 块提取。

尤其是在下列情况下会得到改善：

- 应用程序较少使用绝对读取来读取许多行 (几百或更多)。
- 应用程序以高速率提取行，客户端和服务端位于同一台计算机上或通过快速网络连接。
- 客户端/服务器通信是通过速度较慢的网络 (如拨号链接或广域网) 进行的。

预取的行数同时受到 `PrefetchRows` 连接参数和 `PrefetchBuffer` 参数的限制，后者限制可用于存储预取的行的内存。

可以预取的最大行数为 1000。

示例

下面的连接字符串片段将预取的行数设置为 100：

```
...prefetchrows=100;...
```

另请参见

- `PrefetchBuffer` 连接参数 [PBUF] (第 86 页)

RetryConnectionTimeout 连接参数 [RetryConnTO]

指示 Client Library (dblib、ODBC、ADO 等) 在指定的时间段内只要未找到服务器就一直重试连接。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “连接参数” > “RetryConnectionTimeout (RetryConnTO) 连接参数”。

注意： 此参考指向 SQL Anywhere 文档。

ServerName 连接参数 [ENG]

`EngineName` 连接参数 [ENG] 的同义词

另请参见

- `EngineName` 连接参数 [ENG] (第 90 页)

StartLine 连接参数 [START]

启动从应用程序运行的数据库服务器。

用法

嵌入式数据库

缺省值

无 StartLine 参数

描述

只有当您连接到当前未在运行的数据库服务器时，才应提供 StartLine 参数。StartLine 参数是用于启动服务器的命令行。

示例

- 下面的数据源片段用 32MB 的高速缓存启动数据库服务器。

```
StartLine=dbeng6 -c 32M iqdemo.db
```

另请参见

- DatabaseSwitches 连接参数 [DBS]（第 88 页）

Unconditional 连接参数 [UNC]

使用 dbstop 停止服务器（即使是有到服务器的连接时）。

用法

任何位置

缺省值

否

另请参见

《实用程序指南》 > “stop_iq 数据库关闭实用程序”

描述

dbstop 命令行实用程序关闭数据库服务器。如果在连接字符串中指定 **Unconditional=Yes**，则即使存在活动连接，也会关闭服务器。如果 Unconditional 未设置为 **Yes**，则只有当没有活动连接时才关闭服务器。

示例

- 下面的命令行无条件关闭服务器：

```
dbstop -c "uid=DBA;pwd=SQL;eng=server-name;unc=yes"
```

Userid 连接参数 [UID]

指定登录到数据库时使用的用户 ID。

用法

任何位置。（DSN 和 FILEDSN 连接参数不支持详细形式的关键字。）

缺省值

无

描述

在连接到数据库时必须始终提供用户 ID。用户 ID 不区分大小写，并且不会受到 Case Respect 数据库属性设置的影响。

示例

以下连接字符串片段提供用户 ID DBA 和口令 SQL:

```
uid=DBA;pwd=SQL
```

网络通信参数

如果遇到客户端/服务器网络通信问题，您可以对客户端和服务器同时设置多个命令行参数。这些参数使您能够利用不同网络协议实现的特性。

在服务器或客户端命令行上提供网络通信参数，如下例所示：

```
start_iq -x tcpip(PARM1=value1;PARM2=value2;. . .),...
```

从客户端将通信参数作为 CommLinks 通信参数输入：

```
CommLinks=tcpip(PARM1=value1;PARM2=value2;. . .),...
```

如果参数中有空格，则必须用引号引起网络通信参数，系统命令解释器才能正确分析它：

```
start_iq -x "tcpip(PARM1=value 1;PARM2=value 2;. . .),..."  
start_iq -x "tcpip(PARM1=value1;PARM2=value2;. . .)"
```

如果提供了多个参数，则在 UNIX 下需要用引号，因为 UNIX 将分号解释为命令分隔符。

布尔参数用 YES、ON、TRUE 或 1 中的任何一个启用，用 NO、OFF、FALSE 和 0 中的任何一个禁用。参数不区分大小写。

所提供的示例应全部输入在一行上；您还可以在配置文件中包括它们并使用 @ 服务器或客户端命令行开关调用配置文件。

TCP/IP、HTTP 和 HTTPS 通信参数

目前，TCP/IP、HTTP 和 HTTPS 可使用下列参数。

TCP/IP	HTTP & HTTPS
Broadcast [BCAST]	Certificate
BroadcastListener [BLISTENER]	Certificate_Password
ClientPort [CPORT]	DatabaseName [DBN]
DLL	LocalOnly [LOCAL]
DoBroadcast [DOBROAD]	LogFile [LOG]
Host [IP]	LogMaxSize [LSize]
LocalOnly [LOCAL]	LogOptions [LOpt]
LDAP [LDAP]	LogFormat [LF]
MyIP [ME]	MaxConnections [MaxConn]
ReceiveBufferSize [RCVBUFSZ]	MaxRequestSize [MaxSize]
SendBufferSize [SNDBUFSZ]	MyIP [ME]
ServerPort [PORT]	ServerPort [PORT]
TDS	Timeout [TO]
Timeout [TO]	
VerifyServerName [VERIFY]	

另请参见

- 通信协议 (第 29 页)
- CommLinks 连接参数 [Links] (第 84 页)

Broadcast 通信参数 [BCAST]

指定广播地址。

用法

TCP/IP

值

String (采用 IP 地址的形式)

缺省值

广播到同一子网上的所有地址。

描述

缺省广播地址是使用本地 IP 地址和子网掩码创建的。子网掩码指示 IP 地址的哪个部分标识网络，以及哪个端口标识主机。

对于子网 10.24.98.x，掩码 255.255.255.0 缺省广播地址是 10.24.98.255。

在 Windows 平台上指定 IPv6 地址时，请使用接口标识符。UNIX 平台既支持接口标识符，也支持 IPv6 地址中的接口名称。接口标识符在 Linux (kernel 2.6.13 和更高版本) 上是必需的。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “通信协议” > “使用 TCP/IP 协议”。Sybase IQ 不支持 Mac OS X 平台。

注意： 此参考指向 SQL Anywhere 文档。

示例

要告诉客户端在使用 IPv6 时仅在编号为 2 的接口上进行广播，请使用：

```
LINKS=tcpip(BROADCAST=ff02::1%2)
```

另请参见

- LocalOnly 通信参数 [LOCAL] (第 110 页)

BroadcastListener 通信参数 [BLISTENER]

控制此端口的广播监听。

用法

TCP/IP, 服务器端

值

YES, NO

缺省值

YES

描述

NO 值关闭此端口的监听。使用 **-sb 0** 与在 TCP/IP 上指定 BroadcastListener=NO 等效。

示例

- 启动接受使用 BroadcastListener=NO 的 TCP/IP 连接的服务器：

```
start_iq -x tcpip(BroadcastListener=NO)
```

Certificate 通信参数

允许您指定加密证书的名称。该证书的口令必须使用 `Certificate_Password` 参数指定。

HTTP、HTTPS

值

String

缺省值

无缺省证书名。

示例

- 启动要求 Web 连接使用特定的加密证书的服务器。

```
start_iq -xs
http(Certificate=cert.file;Certificate_Password=secret) ...
```

另请参见

- ClientPort 通信参数 [CPort] (第 106 页)
- DoBroadcast 通信参数 [DBROAD] (第 107 页)
- Host 通信参数 [IP] (第 108 页)
- ServerPort 通信参数 [PORT] (第 117 页)

Certificate_Password 通信参数

允许您指定与通过 `Certificate` 参数指定的加密证书匹配的口令。

用法

HTTP、HTTPS

值

String

缺省值

无缺省证书口令。

示例

- 启动要求 Web 连接使用特定的加密证书的服务器。

```
start_iq -xs
http(Certificate=cert.file;Certificate_Password=secret) ...
```

ClientPort 通信参数 [CPort]

指定客户端应用程序通过 TCP/IP 进行通信所用的端口号。您可以指定一个端口号，也可以指定单个端口号和端口号范围的组合。

用法

TCP/IP。仅限客户端。

缺省值

为每个网络连接动态地分配端口。如果您没有防火墙限制，则建议您不要使用该参数。

描述

该选项是为通过防火墙的连接提供的，因为防火墙软件按照 TCP/UDP 端口进行过滤。除非您由于防火墙原因而需要此参数，否则不要使用它。

如果您希望使用给定的数据源或给定的连接字符串建立多个连接，最好指定一系列端口号或端口号的范围。如果您指定一个端口号，则应用程序一次只能保持一个连接。实际上，即使是在关闭一个连接之后，也会有很短的停顿周期，在此期间无法使用指定的端口建立任何新连接。当您指定一系列端口号和/或端口号的范围时，该应用程序会一直尝试端口号，直到它找到一个可成功绑定到的端口号。

示例

- 下面的字符串建立从使用端口 6000 的应用程序到使用端口 5000 的名为 my_server 的服务器的连接：

```
CommLinks=tcpip{ClientPort=6000;ServerPort=5000};  
ServerName=my_server
```

- 下面的字符串从可以使用端口 5050 到 5060 以及端口 5040 和 5070 的应用程序建立连接，以便与使用缺省服务器端口的名为 my_server 的服务器进行通信：

```
CommLinks=tcpip{ClientPort=5040,5050-5060,5070};ServerName=my_server
```

另请参见

- Certificate 通信参数 (第 105 页)
- DoBroadcast 通信参数 [DBROAD] (第 107 页)
- Host 通信参数 [IP] (第 108 页)
- ServerPort 通信参数 [PORT] (第 117 页)

DatabaseName 通信参数 [DBN]

指定处理 Web 请求时要使用的数据库的名称，或者使用 **REQUIRED** 或 **AUTO** 关键字指定是否需要在 URI 中使用数据库名称。

用法

HTTP、HTTPS

值

AUTO, **REQUIRED**, *database-name*

缺省值

AUTO

描述

如果此参数设置为 **REQUIRED**，则 URI 必须指定数据库名称。

如果此参数设置为 **AUTO**，则 URI 可以指定数据库名称，也可以不指定。如果 URI 中不包含数据库名称，将使用服务器上的缺省数据库来处理 Web 请求。既然在设置为 **AUTO** 时服务器必须推测 URI 中是否包含数据库名称，您在设计 Web 站点时就应该避免多义性。

如果此参数设置为数据库的名称，则使用该数据库处理所有 Web 请求。URI 中一定不能包含数据库名称。

示例

- 以下命令启动两个数据库，但是只允许通过 **HTTP** 访问其中的一个。

```
start_iq -xs http(dbn=web) iqdemo.db web.db
```

另请参见

- DatabaseFile 连接参数 [DBF]（第 86 页）
- DatabaseName 连接参数 [DBN]（第 87 页）

DoBroadcast 通信参数 [DBROAD]

执行广播以搜索服务器。

用法

TCP/IP（所有平台）

值

ALL、**NONE**、**DIRECT**（客户端）

YES、**NO**（服务器端）

缺省值

ALL

描述

客户端用法 - 在 `DoBroadcast=ALL` (以前是 `DoBroadcast=YES`) 时, 执行广播以搜索服务器。广播首先进入本地子网。如果指定了 `HOST=`, 还将广播包发送到每个主机。对于 `TCP`, 所有的广播包都是 `UDP` 包。

在 `DoBroadcast=DIRECT` (以前是 `DoBroadcast=NO`) 时, 不会通过对本地子网执行广播来搜索数据库服务器。广播包只发送到列在 `HOST (IP)` 通信参数中的主机。如果您指定 `DoBroadcast=DIRECT`, 则需要 `HOST (IP)` 通信参数。

指定 `DoBroadcast=NONE` 将导致不使用 `UDP` 广播。将直接与指定的 `HOST/PORT` 建立 `TCP/IP` 连接, 并会检验服务器名。使用 `TCP/IP` 时, 通过将 `VerifyServerName (VERIFY)` 通信参数设置为 `NO`, 您可以选择不检验服务器名。`HOST (IP)` 通信参数是必需的参数, `ServerPort (PORT)` 通信参数是可选的参数。

对于 `DIRECT` 和 `NONE`, 则必须用 `HOST` 选项指定服务器主机。

服务器用法 - 如果设置 `DoBroadcast=NO`, 则会禁止数据库服务器进行广播以查找同名的其它服务器。这在极少的情况下有用, 但是通常不建议这样做。

示例

- 以下命令将启动客户端, 但不会通过广播来搜索数据库服务器。而是只在名为 `silver` 的计算机上查找服务器。

```
dbisql -x tcpip(DOBROADCAST=DIRECT;HOST=silver) iqdemo
```

- 在 `UNIX` 上, 必须用引号将选项引起来:

```
dbisql -x "tcpip(DOBROADCAST=DIRECT;HOST=silver)" iqdemo
```

另请参见

- `Certificate` 通信参数 (第 105 页)
- `ClientPort` 通信参数 [`CPort`] (第 106 页)
- `Host` 通信参数 [`IP`] (第 108 页)
- `ServerPort` 通信参数 [`PORT`] (第 117 页)

Host 通信参数 [IP]

指定要由 `Client Library` 在直接网络外部搜索的其它计算机。

用法

`TCP/IP` (所有平台) 服务器端和客户端

其中

服务器端和客户端

描述

在服务器上，执行该搜索以避免使用重复名称启动服务器。

对于 TCP/IP，可以使用 *hostname* 或用句点分隔的 IP 地址。

如果使用 **-z** 开关，则服务器会在启动期间输出此寻址信息。此外，如果指定了 **LogFile** (**Debug** 设置为 **TRUE**)，则应用程序会将该信息写入其日志文件。

您可以使用以分号分隔的地址列表来搜索多台计算机。您还可以将端口号附加到 IP 地址，并使用冒号作为分隔符。或者，也可以显式地指定主机和服务器端口，如 *Host=nnn.nn.nnn.nnn;ServerPort=pppp*。

在使用 TCP/IP 时，**IP** 和 **HOST** 是同义词。

在 Windows 平台上指定 IPv6 地址时，请使用 接口标识符。UNIX 平台既支持在 IPv6 地址中使用接口标识符，也支持在 IPv6 地址中使用接口名称。接口标识符在 Linux (内核 2.6.13 和更高版本) 上是必需的。Sybase IQ 不支持 Mac OS X 平台。

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “通信协议” > “使用 TCP/IP 协议”。

注意： 此参考指向 SQL Anywhere 文档。

值

String

缺省值

无其它计算机。

示例

- 以下连接字符串片段指示客户端在计算机“kangaroo”和 197.75.209.222 (端口 2369) 上查找名为 **iqdemo** 的数据库服务器：

```
...ENG=iqdemo CommLinks=tcPIP(IP=kangaroo;IP=197.75.209.222:2369)
```

- 对于 UNIX，需要在 TCP/IP 选项两边加引号：

```
dbisql -x "tcPIP(HOST=kangaroo;HOST=197.75.209.222)" iqdemo
```

- 下面的连接字符串片段命令客户端在计算机 **my_server** 和 **kangaroo** 上查找数据库服务器。尝试与第一个响应的主机建立连接。

```
dbisql -c
"UID=DBA;PWD=sql;LINKS=tcPIP(HOST=my_server,kangaroo;PORT=2639)"
```

另请参见

- **Certificate** 通信参数 (第 105 页)
- **ClientPort** 通信参数 [**CPort**] (第 106 页)
- **DoBroadcast** 通信参数 [**DBROAD**] (第 107 页)
- **ServerPort** 通信参数 [**PORT**] (第 117 页)

LDAP 通信参数 [LDAP]

可以使通过 WAN 或穿过防火墙运行的客户端找到服务器，而无需指定 IP 地址。

用法

可以使通过 WAN 或穿过防火墙运行的客户端找到服务器，而无需指定 IP 地址。这还使定位实用程序 (dblocate) 能够查找这样的服务器。

TCP/IP (仅限服务器端)

值

YES、**NO** 或 *filename*

缺省值

ON

缺省 *filename* 为 *asaldap.ini*

描述

通过让数据库服务器自行注册到 LDAP 服务器使客户端 (以及定位实用程序 [dblocate]) 能够查询 LDAP 服务器。

指定 **LDAP=filename** 会启用 LDAP 支持并将指定文件用作配置文件。指定 **LDAP=YES** 会启用 LDAP 支持并将 *saldap.ini* 用作配置文件。

LDAP 仅与 TCP/IP 一起使用，而且仅在网络服务器上使用。

另请参见

“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “通信协议” > “使用 TCP/IP 协议” > “使用 LDAP 服务器进行连接”。

注意： 此参考指向 SQL Anywhere 文档。

LocalOnly 通信参数 [LOCAL]

允许客户端选择只连接到本地计算机上的服务器 (如果存在)。

用法

TCP/IP、HTTP、HTTPS

值

YES、**NO**

缺省值

NO

描述

如果在本地计算机上找不到具有匹配的服务器名的服务器，则服务器将不会自动启动。

只有当指定了 `DoBroadcast=ALL`（缺省值）时，`LocalOnly (LOCAL)` 通信参数才有用。

如果设置 `LocalOnly=YES`，则使用常规的广播机制，但是将忽略来自其它计算机上的服务器的广播响应。

可以对服务器使用 `LocalOnly (LOCAL)` 通信参数，以将连接限制到本地计算机。尝试从远程计算机连接将无法找到此服务器，并且定位 `[dblocate]` 实用程序将无法看到此服务器。在将 `LocalOnly (LOCAL)` 通信参数设置为 `YES` 的情况下运行服务器，可以让网络服务器作为个人服务器运行，而不受连接或 CPU 限制。

另请参见

- `Broadcast` 通信参数 `[BCAST]`（第 103 页）

LogFile 通信参数 [LOG]

指定数据库服务器将有关 Web 请求的信息写入到的文件的名称。

用法

HTTP、HTTPS

值

Filename

缺省值

无

另请参见

- `LogMaxSize` 通信参数 `[LSIZE]`（第 112 页）
- `LogOptions` 通信参数 `[LOPT]`（第 113 页）

LogFormat 通信参数 [LF]

控制写入日志文件的消息的格式及消息中显示的字段。

用法

HTTP、HTTPS

值

Format-string

缺省值

@T - @W - @I - @P - "@M @U @V" - @R - @L - @E

描述

如果消息字符串包含以下任何代码，则在写入每条消息时将使用当前值替换这些代码。

- @ 字符。
- 开始处理请求的日期和时间（由于错误而无法排队的请求除外）。
- 客户端连接的日期和时间。
- 与请求关联的数据库的名称。
- 错误消息的文本（如果发生错误）。
- 请求处理结束的日期和时间。
- 客户端的 IP 地址。
- 包括消息头和消息体的响应的长度（以字节为单位）。
- HTTP 请求方法。
- 与请求关联的监听器端口。
- 请求排队等候处理的日期和时间（由于错误而无法排队的请求除外）。
- HTTP 响应的状态码和说明。
- HTTP 状态码。
- 当前日志条目写入的日期和时间。
- 请求的 URI。
- 请求的 HTTP 版本。
- 用于处理请求的时间 (@F - @B)，或者在由于出错而没有处理请求时为 0.000。

LogMaxSize 通信参数 [LSIZE]

控制供数据库服务器写入有关 Web 请求的信息的日志文件的最大大小。

LogMaxSize 参数是一个 SQL Anywhere 协议选项。请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “网络协议选项” > “LogMaxSize (LSIZE) 协议选项”。

注意： 此参考指向 SQL Anywhere 文档。

另请参见

- LogFile 通信参数 [LOG]（第 111 页）
- LogOptions 通信参数 [LOPT]（第 113 页）

LogOptions 通信参数 [LOPT]

控制记录的消息类别。

用法

HTTP、HTTPS

值

NONE, OK, INFO, ERRORS, ALL, *status-codes*, REQHDRS, RESHDRS, HEADERS

缺省值

ALL

描述

可用的值包括用于选择特定类型消息和 HTTP 状态码的关键字。可以指定多个值，以逗号分隔。

下面的关键字控制记录的消息类别：

- 不记录任何内容。
- 记录成功完成的请求（20x HTTP 状态码）。
- 记录返回结束或未修改的状态码的请求（30x HTTP 状态码）。
- 记录所有错误（40x 和 50x HTTP 状态码）
- 记录所有请求。

还提供以下常用 HTTP 状态码。它们可用于记录返回特定状态码的请求：

- 确定
- 错误请求
- 未授权
- 禁止
- 未找到
- 请求超时
- 未实现
- 服务不可用

此外，下列关键字可用于获取有关记录的消息的详细信息：

- 记录请求时，同时将请求的消息头写入日志文件。
- 记录请求时，同时将响应的消息头写入日志文件。
- 记录请求时，同时将请求和响应的消息头写入日志文件（与 REQHDRS,RESHDRS 相同）。

另请参见

- LogFile 通信参数 [LOG]（第 111 页）

- LogMaxSize 通信参数 [LSIZE] (第 112 页)

MaxConnections 通信参数 [MAXCONN]

指定服务器所接受的同时发生的连接数。

用法

HTTP、HTTPS

值

Number

缺省值

许可的连接数

描述

值 0 指示无限制。

另请参见

- MaxRequestSize 通信参数 [MAXSIZE] (第 114 页)

MaxRequestSize 通信参数 [MAXSIZE]

指定数据库服务器可接受的最大请求的大小。

描述

MaxRequestSize 参数是一个 SQL Anywhere 协议选项。请参见“SQL Anywhere Server-数据库管理” > “启动并连接到数据库” > “网络协议选项” > “MaxRequestSize (MAXSIZE) 协议选项”。

注意： 此参考指向 SQL Anywhere 文档。

另请参见

- MaxConnections 通信参数 [MAXCONN] (第 114 页)

MyIP 通信参数 [ME]

控制客户端如何尝试确定寻址信息。

用法

TCP/IP、HTTP、HTTPS

值

String

描述

MyIP (ME) 参数是为具有多个网络适配器的计算机提供的。

每个适配器都有一个 IP 地址。缺省情况下，**Sybase IQ** 使用它找到的第一个网卡。如果您希望数据库服务器使用多个网卡，请在 **MyIP (ME)** 参数中指定每个网卡的地址。

如果将关键字 **NONE** 作为 IP 号提供，则不尝试确定寻址信息。在有些计算机上执行此操作成本会比较高，例如，有多块网卡的计算机或者装有远程访问 (**RAS**) 软件和一块网卡的计算机，关键字 **NONE** 主要用于运行在此类计算机操作系统上的客户端。它不适用于在服务器上使用。

在 **Windows** 平台上，对于具有多个 IP 地址的计算机，可多次使用该选项。

请用逗号分隔多个 IP 地址。您可以选择将端口号附加到 IP 地址，用冒号分隔。

在 **Windows** 平台上指定 IPv6 地址时，应使用接口标识符。**Unix** 平台既支持在 IPv6 地址中使用接口标识符，也支持在 IPv6 地址中使用接口名称。接口标识符在 **Linux** (内核 2.6.13 和更高版本) 上是必需的。**Sybase IQ** 不支持 **Mac OS X** 平台。

请参见“**SQL Anywhere Server - 数据库管理**” > “启动并连接到数据库” > “通信协议” > “使用 **TCP/IP** 协议”。

注意： 此参考指向 **SQL Anywhere** 文档。

示例

- 下面的 **Windows** 命令行 (全部输入在一行上) 指示服务器使用两块网卡，其中一块使用一个指定的端口号。

```
start_iq -x tcpip(MyIP=192.75.209.12:2367,192.75.209.32)
c:\sybase\IQ-15_4\demo\iqdemo.db
```

- 下面的命令行 (全部输入在一行上) 指示服务器使用 IPv6 网卡。

```
start_iq -x tcpip(MyIP=fe80::5445:5245:444f)
"c:\sybase\IQ-15_4\demo\iqdemo.db"
```

- 下面的连接字符串片段指示客户端不尝试确定寻址信息。

```
...CommLinks= tcpip(MyIP=NONE)...
```

PreFetchOnOpen 通信参数

将预取请求与游标打开请求一起发送，从而不必在每次打开游标时发送提取行的网络请求。

用法

ODBC

值

YES, NO

缺省值

NO

描述

列必须已被绑定，才能在打开时进行预取。使用 **PreFetchOnOpen** 时，如果重新绑定游标打开和首次提取之间的列，则将使性能降低。

对返回结果集的查询或存储过程调用 ODBC 的 **SQLExecute** 或 **SQLExecDirect** 会导致游标打开。

在下列情况下，启用此选项可提高系统性能：

- 网络的等待时间较长
- 应用程序发送过多的打开和关闭游标请求

ReceiveBufferSize 通信参数 [RCVBUFSZ]

为由 TCP/IP 协议堆栈使用的缓冲区设置大小。如果网络上的 LOB 性能非常重要，您可能希望增加该值。

用法

TCP/IP

值

Integer [**k** | **m** | **g**]

最大允许值

1048576 字节 (1MB)

缺省值

与计算机有关

描述

分别使用 **k**、**m** 或 **g** 将单位指定为千字节、兆字节或千兆字节。

SendBufferSize 通信参数 [SNDBUFSZ]

为由 TCP/IP 协议堆栈使用的缓冲区设置大小。

用法

TCP/IP

*值**Integer* [**k** | **m** | **g**]*最大允许值*

1048576 字节 (1MB)

缺省值

与计算机有关

描述

缺省值以字节为单位，但可使用 **k**、**m** 或 **g** 将单位分别指定为千字节、兆字节或千兆字节。如果网络上的 LOB 性能非常重要，您可能希望增加该值。

另请参见

请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “通信协议” > “使用 TCP/IP 协议”。

注意： 此参考指向 SQL Anywhere 文档。

向服务器提供 RSA 加密功能的 RSA 安全模块是随 Sybase IQ 免费提供的，但 Kerberos、ECC 和 FIPS 安全模块需要单独的许可证。有关许可证的详细信息，请参见《安装和配置指南》。

ServerPort 通信参数 [PORT]

对于数据库服务器，**ServerPort** 选项指定使用 TCP/IP 在其上进行通信的端口号。在数据源中，**ServerPort** 选项将数据库服务器正在监听 TCP/IP 通信的端口通知客户端。

用法

TCP/IP (所有平台)、HTTP、HTTPS

*值**Integer**缺省值*TCP/IP 的缺省值为 **2638**。HTTP 的缺省值为 **80**。HTTPS 的缺省值为 **443**。*描述*

Internet 编号授权委员会已分配用于 TCP/IP 通信的 IQ 数据库服务器端口号 **2638**。但是，并不禁止应用程序使用该保留端口，这可能会导致在数据库服务器和其它应用程序之间发生寻址冲突。

客户端向在 **ServerPort** 参数中指定的每个端口进行广播以查找该服务器。如果未指定 **ServerPort**，则使用缺省端口 **2638**。

数据库服务器使用在 **ServerPort** 中为 TCP/IP 连接指定的端口号。如果未指定 **ServerPort**，则使用缺省端口 **2638**。因此，应用程序可在不指定端口号的情况下连接到数据库服务器。**HP-UX** 操作系统是一种例外情况，在该操作系统上，如果在其它端口上启动服务器，则该服务器不会在端口 **2638** 上监听。

缺省情况下，数据库服务器会监听标准的 **HTTP** 和 **HTTPS** 端口（分别为 80 和 443）。

示例

1. 在 Windows 上，启动 IQ 网络服务器：

```
start_iq -x tcpip c:\sybase\IQ-15_4\demo\iqdemo.db
```

现在采用端口号 2638。

2. 尝试启动其它数据库服务器：

```
start_iq -x tcpip c:\sybase\IQ-15_4\demo\iqdemo2.db
```

缺省端口已分配，服务器在其它端口上启动。

3. 在端口 2639 上启动其它数据库服务器：

```
start_iq -x "tcpip(ServerPort=2639)" c:\sybase\IQ-15_4\demo\iqdemo3.db
```

只要 2639 不是保留端口且没有其它应用程序分配此端口，此操作就会成功。

4. 如果计算机上的另一个 **Web** 服务器已在使用端口 80，或者您没有权限在此类低端口号上启动服务器，请启动一台监听替代端口（例如 8080）的服务器：

```
start_iq -xs http{port=8080} -n server3 web.db
```

另请参见

- Certificate 通信参数（第 105 页）
- ClientPort 通信参数 [CPort]（第 106 页）
- DoBroadcast 通信参数 [DBROAD]（第 107 页）
- Host 通信参数 [IP]（第 108 页）

Sessions 通信参数

设置通过单个 LAN 适配器可以同时与服务器通信的客户端的最大数目。

用法

NetBIOS。仅限服务器端。

描述

缺省设置因操作系统而异。值为整数，且最大值为 254。

NetBIOS 网络软件对每台计算机允许的 *命令* 数目有限制。Sybase IQ 使用这些 NetBIOS 命令，且如果系统没有更多可用的命令，则会禁止进一步的连接，即使是小于 Sessions 参数的值也会如此。

缺省值

因操作系统而异。在 Windows 上，缺省值为 16。

示例

下面的语句使用名为 `iqdemo` 的数据库启动服务器，允许 200 个 NetBIOS 连接。

```
start_iq -x netbios(sessions=200) iqdemo.db
```

TDS 通信参数

控制服务器是否允许 TDS 连接。

用法

TCP/IP、NamedPipes。仅限服务器端。

值

YES, NO

缺省值

YES

描述

要禁止到数据库服务器的 TDS 连接，请将 TDS 设置为 NO。如果您希望确保针对服务器只建立加密连接，则这些端口选项是禁止建立 TDS 连接的唯一方法。

示例

- 以下命令使用 TCP/IP 协议启动数据库服务器，但是禁止从 Open Client 或 jConnect 应用程序建立连接。

```
start_iq -x tcpip(TDS=NO) ...
```

Timeout 通信参数 [TO]

指定在建立通信和断开连接时等待响应的的时间长度（以秒为单位）。

用法

TCP/IP（所有平台）、HTTP、HTTPS

值

Integer（以秒为单位）

最大允许值

3600 秒

缺省值

5

描述

如果您在建立 **TCP/IP** 通信时遇到问题，则可能希望尝试等待更长的时间。

在 **HTTP** 或 **HTTPS** 应用程序中，该参数指定接收请求时允许的最大空闲时间。当达到该限制时，会关闭连接并向客户端返回 **408 REQUEST TIMEOUT**。值 **0** 禁用空闲超时，但应慎用。无此限制时，恶意的客户端可能会消耗服务器资源并阻止其它客户端连接。

示例

下面的数据源片段只启动 **TCP/IP** 通信链接，超时期限为 20 秒。

```
...  
CommLinks=tcpip(TO=20)  
...
```

VerifyServerName 通信参数 [Verify]

VerifyServerName 通信参数指定连接到该主机时是否检验服务器名。

用法

TCP/IP (仅限客户端)

值

YES, NO

缺省值

YES

描述

通常，不应设置此选项。此选项仅在需要平衡这些服务器之间的查询负载时用于连接到 **Multiplex** 辅助服务器。

当使用 **DoBroadcast=NONE** 参数通过 **TCP** 连接时，客户端会建立 **TCP** 连接，然后检验找到的服务器的名称与它要查找的服务器的名称是否相同。如果指定 **VerifyServerName=NO**，则跳过对服务器名的检验。这会使 **IQ** 客户端能够在只知道一个 **IP** 地址/端口的情况下连接到 **IQ** 服务器。

服务器名仍必须在连接字符串中指定，但是它会被忽略。只有在指定了 **DoBroadcast=NONE** 的情况下才使用 **VerifyServerName (VERIFY)** 通信参数。

按照示例所示方法使用时，如果将此选项设置为 **NO**，则会允许您指定到特定 **IP** 地址和端口号的连接。该 **IP** 地址和端口号用于充当 **IQ** 客户端和 **IQ** 服务器之间的网关的负载均衡计算机。

示例

要使用此选项，在客户端计算机上的“ODBC 管理器”中创建一个新的 ODBC DSN，并如下所示指定参数：

- 在“数据库”选项卡上，指定一个将用于连接到所有辅助服务器的通用服务器名，例如 `qserv`。服务器名是必需的，但由于“网络”选项卡中的参数，会忽略服务器名。
- 在“网络”选项卡上，选中“TCP/IP”复选框，然后在文本框中键入：

```
host=ip_address:port#;DOBROADCAST=NONE;VERIFY=NO
```

例如：

```
host=123.456.77.888:2222;DOBROADCAST=NONE;VERIFY=NO
```

IQ 客户端连接到此 DSN 时，负载均衡器会根据计算机的工作负载调度到特定辅助服务器的连接。

数据库对象管理

Sybase IQ 允许您创建、更改和删除数据库以及数据库对象（例如表、视图、索引、过程和登录策略）。

注意：请记住，Sybase IQ 是由 Catalog 存储和 IQ 存储二者共同组成的。本章说明如何创建这两种存储以及 IQ 存储中的对象。在 Catalog 存储中创建的表具有 SQL Anywhere 表的特性。如果要在 Catalog 存储中创建表，则需要参考 SQL Anywhere 文档。

Sybase IQ 数据库设计

适当的数据库设计可增强数据可用性和数据检索速度。

在创建数据库之前，必须决定在哪里存储数据、数据库需要多少空间以及谁将能够定义或修改数据库对象。Sybase IQ 提供了一些工具和进程来帮助您作出决定。

Sybase PowerDesigner® 可以帮助您设计数据库，具体来说，可以帮助您生成概念、物理或面向对象的数据模型，然后从模型生成数据库。它还允许您实施反向工程，从现有数据库创建模型。

无论使用哪个设计工具，数据库管理员 (DBA) 通常都要设计数据库，并定义其内容。要创建有效的设计，DBA 需要与整个组织中的每个人合作，以了解将如何使用数据。DBA 还需要了解 IQ 数据库所基于的概念。

Sybase IQ 数据库是为了用作数据仓库而经过优化的关系数据库。作为关系数据库，它由一组用于组织数据的相关表组成；作为数据仓库，它以索引的方式提供对非常大的数据集的有效访问。

创建数据库时，需要指定这些表的结构、表所允许的数据类型、表间关系、用于存储表数据的索引以及用于控制谁能访问数据的视图。在创建 IQ 数据库之前，请确保了解在《Sybase IQ 简介》中介绍的关系数据库和数据仓库应用概念。

使用数据库对象的工具

Sybase IQ 包括三个使用数据库对象的实用程序：Sybase Central、Sybase Control Center 和 Interactive SQL。此外，PowerDesigner 可以用于设计和创建整个数据仓库。

Sybase Central

Sybase Central 是用于在窗口系统上使用数据库对象的主要工具。

可以使用 Sybase Central 管理服务器、数据库和数据库空间。它允许您创建、修改和删除所有种类的数据库对象，包括表、过程、视图、索引、用户和组。要在 Multiplex 服务器上使用 Sybase Central，请参见《使用 Sybase IQ Multiplex》。

本章主要介绍使用数据库对象的 SQL 语句。如果使用 Sybase Central，则系统会自动为您生成这些 SQL 语句。Sybase Central 联机帮助是有关 Sybase Central 的信息的主要来源。本章仅为可以使用 Sybase Central 执行的任务提供简短指示。

有关使用 Sybase Central 的简介，请参见《Sybase IQ 简介》>“管理数据库”。

Interactive SQL

Interactive SQL (**dbisql**) 是用于输入 SQL 语句的实用程序。

如果您使用 Interactive SQL 处理数据库 schema，而不是逐句执行 SQL 语句，请在 **dbisql** 命令文件中创建命令集。然后，可以在 **dbisql** 中执行此文件以生成数据库。

数据库对象的定义构成数据库 schema。您可以将 schema 视为一个空数据库。用于创建和修改 schema 的 SQL 语句称为 *数据定义语言* (DDL)。

注意：每次只能有一个用户可以对表执行 DDL 语句。IQ 将在对表执行 DDL 操作期间锁定该表。但是，用户可以同时对相同数据库中的其它对象执行 DDL。

如果使用除了 Interactive SQL 以外的其它工具，则以下主题提供的关于 SQL 语句的所有信息仍然适用。

Interactive SQL 命令文件

Interactive SQL 命令文件是在命令的末尾放置分号的文本文件，如下所示。

```
CREATE TABLE t1 ( .. );
CREATE TABLE t2 ( .. );
CREATE LF INDEX i2 ON t2 ( .. );
..
```

Interactive SQL 命令文件通常具有扩展名 **.sql**。要执行命令文件，请将文件的内容粘贴到 Interactive SQL 命令窗口中（如果文件小于 500 行），或者输入可以将文件读取到命令窗口中的命令。例如，**READ** 语句：

```
read makedb
```

读取 makedb.sql 文件中的 Interactive SQL 命令。

另请参见

- 不能写入已锁定的表（第 465 页）
- 锁定的工作方式（第 347 页）
- 管理对表的写入锁争用（第 465 页）

设置 Sybase IQ 数据库

创建 IQ 数据库是从安装开始到数据库对用户可用时结束的较大设置过程的一部分。

1. 安装和配置 Sybase IQ。

此步骤将安装客户端和服务环境以及 `iqdemo` 数据库。有关详细信息，请参见针对所用平台的《安装和配置指南》。

2. 创建 IQ 数据库。

此步骤将创建 IQ 存储和 Catalog 存储。可以使用 Sybase Central、SQL 语句 **CREATE DATABASE** 或 `iqinit` 实用程序。

3. 在 IQ 数据库中创建表。

使用 **CREATE TABLE** 语句或 Sybase Central 创建表向导。

4. 为表创建索引。

使用 **CREATE INDEX** 语句或 Sybase Central 创建索引向导。创建表时，还可以自动创建某些索引。

5. 将数据装载到表中。

使用 **LOAD TABLE** 语句从文件批量装载数据，或者使用 **INSERT** 语句从现有数据库提取数据行。

调度数据定义任务准则

只要数据库存在，且其他用户可以访问它，就应在需要执行其它数据定义操作（例如添加或修改表或索引）时遵循这些准则。

可以将数据定义操作调度到数据库使用率低时进行。当您创建或更改表时，尽管只是很短的时间，所有其他用户仍将被禁止对该表进行读取或写入。如果表是连接索引的一部分，则用户不能对连接索引中的任何表进行读取或写入，直到数据定义操作完成。有关数据定义期间的并发规则的详细信息，请参见“DDL 操作锁”。

另请参见

- DDL 操作锁（第 348 页）
- 锁定的工作方式（第 347 页）
- IQ PAGE SIZE 参数准则（第 143 页）
- 不同的值数目（第 277 页）
- 有效数据的规则和检查（第 317 页）

用于性能监控的 Dummy 表

要启动 IQ 缓冲区高速缓存监视器，必须指定一个永久表或临时表，最好是仅用于监控的虚表。

数据定义的权限

为了执行数据定义任务，必须有合适的权限。

- 使用 **DBA** 权限，可以执行所有数据定义任务。还可以向其他用户授予权限，以执行特定任务。这包括向其他用户授予 **DBA** 权限的能力。
- 要创建任何数据库对象，您需要有该类型对象的 **资源权限**。
- 创建对象时，您将成为该对象的所有者。对象的所有者会自动获得对该对象执行所有操作的权限，以及授予其他用户更新表中信息的权限。

具有 **DBA** 或 **PERMS ADMIN** 权限的用户以及对象所有者可以向单个用户和用户组授予权限。还可以使用 **-gu** 命令行选项设置创建或删除数据库所需的权限级别。

设备选择

可以将数据库和数据库对象存储在设备上。在所有平台上，这些设备可以是操作系统文件。它们还可以是磁盘的组成部分，称为原始分区。创建数据库时，**Sybase IQ** 自动确定它是原始分区还是磁盘文件。

在生产环境中，原始分区安装可以提供更好的处理性能和更好的恢复功能。另一方面，文件系统可以使您更容易管理设备，并且可能更适合用于开发环境。

注意： Catalog 存储和事务日志不能位于原始分区上。

另请参见

- 数据库文件放置 (第 139 页)

空间分配

所有 **Sybase IQ** 数据库都是预分配的，无论它们驻留在文件系统中，还是在原始分区中。

每个数据库均包括多个表空间。表空间是数据库中的存储单位，可作为总存储空间的逻辑子集进行管理。可以将各个对象和子对象分配到各个表空间。

数据库空间是由一个或多个操作系统文件组成的表空间。

存储是一个或多个数据库空间，用于针对某一特殊用途存储永久数据或临时数据。

Sybase IQ 有四个存储：

- **Catalog** 存储包含 **SYSTEM** 数据库空间，还包含用户定义的附加 **Catalog** 数据库空间。
- **IQ** 主存储区包含 **IQ_SYSTEM_MAIN** 数据库空间和其它用户数据库空间。

- IQ 临时存储包含 IQ_SYSTEM_TEMP 数据库空间。
- IQ 共享临时存储包含 IQ_SHARED_TEMP 数据库空间。

数据库空间类型

每个类型的数据库空间存储特定类型的 Sybase IQ 数据。

数据库空间类型	存储的数据	数据库空间包含的文件	数据库空间数目
SYSTEM 数据库空间	系统表、视图、存储过程、SQL Anywhere 表和函数定义	一个	一个或更多
其它 Catalog 数据库空间	SQL Anywhere 表	一个	一个或更多
IQ_SYSTEM_MAIN	IQ 数据库结构（包括每个已提交事务和每个活动检查点事务的 IQ 前滚/回退数据）、增量备份元数据以及数据库空间和标识元数据。可以在此数据库空间中存储 IQ 用户对象，但 Sybase 建议您将这些对象放置到其它 main 数据库空间。	一个或更多	一个
其它 main 数据库空间（也称为用户数据库空间）	表、索引、连接索引和表元数据等 IQ 对象。	一个或更多	一个或更多
IQ_SYSTEM_TEMP	1 到 n 个临时 dbfile 组成的集，用于为单独的数据库或 Multiplex 节点定义一个临时数据库空间	一个或更多	一个
IQ_SHARED_TEMP	1 到 n 个临时 dbfile 组成的集，用于定义一个所有 Multiplex 节点共享的临时数据库空间。	一个或多个（最初没有文件）	一个
IQ_SYSTEM_MSG	记录有关数据库活动的消息的外部文件	每个 Multiplex 节点各一个	一个

隐式或显式指定表或连接索引的数据库空间。对于基表和连接索引，**DEFAULT_DBSPACE** 选项的值可隐式确定数据库空间的位置，也可以使用 **CREATE TABLE IN dbspace_name clause** 或 **CREATE JOIN INDEX IN dbspace_name** 子句显式指定该位置。基表通常在 IQ 主存储区的数据库空间中创建，但是也可以在 Catalog 存储的数据库空间中创建没有 IQ 索引的基表。

对于全局临时表，请指定 **IN SYSTEM** 子句以显式创建 SA 全局临时表。缺省情况下，在 IQ_SYSTEM_TEMP 中创建 IQ 临时表。

另请参见

- Sybase IQ 中的数据存储 (第 3 页)

Catalog 存储

Catalog 存储包含 IQ 数据库的元数据。

元数据描述 IQ 表、列和索引的布局。Catalog 存储有时也简称为目录。

- SYSTEM 数据库空间 - 名为 SYSTEM 的 IQ Catalog 数据库空间包含 IQ 数据库的元数据，并且其存储格式与 SQL Anywhere 关系数据库系统中表的格式相同 SQL Anywhere 可与 IQ 共存或独立于 IQ 存在。除 IQ 表以外，Catalog 存储中还可以有 SQL Anywhere 样式的表，或者您也可以有单独的 SQL Anywhere 数据库。每个 Catalog 数据库空间正好包含一个文件。
- 其它 Catalog 数据库空间 - 可以在独立于 SYSTEM 数据库空间的数据库空间中创建 SQL Anywhere 表。

您可以为 Catalog 存储预分配空间。请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “使用数据库文件” > “使用附加 dbspace” > “为数据库文件预分配空间”。

注意： 此参考指向 SQL Anywhere 文档。

另请参见

- 数据库启动准则 (第 38 页)

IQ_SYSTEM_MAIN 数据库空间

IQ_SYSTEM_MAIN 是一个特殊数据库空间，它包含打开数据库所必需的结构：IQ 检查点日志、每个已提交事务和每个活动检查点事务的 IQ 前滚/回退数据、增量备份元数据以及数据库空间和标识元数据。

IQ_SYSTEM_MAIN dbspace 在创建数据库时 或将早期版本的 IQ 数据库升级到 Sybase IQ15.4 时创建。当打开数据库时，IQ_SYSTEM_MAIN 始终处于联机 状态。

另请参见

- 磁盘空间不足 (第 455 页)
- IQ 主存储区和 IQ 临时存储空间管理 (第 146 页)
- 定义数据库期间的装载性能 (第 277 页)
- 主 IQ 存储块消息 (第 474 页)
- 监控磁盘空间使用情况 (第 459 页)
- 处理问题 (第 463 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- Sybase IQ 停止处理或停止响应 (第 449 页)

其它用户主数据库空间

创建用户主数据库空间，这样，用户就不会在 IQ_SYSTEM_MAIN 中放置用户表或索引。

最佳实践是避免在 IQ_SYSTEM_MAIN 中放置用户表或索引。管理员可以允许在 IQ_SYSTEM_MAIN 中创建用户表，特别是当这些表比较小且包含非常重要内容时。但是，推荐方法是在创建数据库之后，管理员应立即创建另一个 main 数据库空间（用户 main 数据库空间），撤消 PUBLIC 在数据库空间 IQ_SYSTEM_MAIN 中的 CREATE 特权，向选定用户或 PUBLIC 授予对新的 main 数据库空间的 CREATE 特权，并将 PUBLIC.DEFAULT_DBSPACE 设置为新的用户数据库空间。

例如：

```
CREATE DBSPACE user_main USING FILE user_main 'user_main1' SIZE
10000; GRANT CREATE ON user_main TO PUBLIC; REVOKE CREATE ON
IQ_SYSTEM_MAIN FROM PUBLIC; SET OPTION PUBLIC.DEFAULT_DBSPACE =
'user_main';
```

IQ 临时数据库空间

IQ 临时存储的单一 dbspace IQ_SYSTEM_TEMP 在创建数据库 或将早期版本的 IQ 数据库升级到 Sybase IQ 15.4 时创建。

每个 IQ 数据库空间都可以包含任意数量的文件。唯一的限制是 IQ 文件的总数为 16383。

确保 IQ_SYSTEM_TEMP dbspace 至少与 start_iq 实用程序的 -iqtc 服务器选项中指定的值一样大。请参见《实用程序指南》。

IQ 共享临时数据库空间

您在创建数据库或者将 12.7 ESD#5 或更高版本的 IQ 数据库升级到 Sybase IQ15.4 时，系统将创建 IQ 共享临时存储 IQ_SHARED_TEMP 的单一 dbspace。此 dbspace 存储分布式查询处理的节点之间共享的临时结构。

最初，此数据库空间不包含任何文件。要添加文件，请使用 ALTER DBSPACE ADD FILE。将文件分配到此 dbspace 是可选的操作，只有在 Multiplex 服务器中进行分布式查询处理时才必须执行该操作。

IQ 消息文件数据库空间

IQ_SYSTEM_MSG 是一个系统数据库空间，指向数据库 IQ 消息日志文件的文件路径。

IQ_SYSTEM_MSG 不存储任何数据，因此不将其视为存储。它在每个 Multiplex 节点上各具有一个文件。缺省情况下，Simplex 服务器或 Multiplex 协调器上的消息文件的物理文件名为 <dbname>.iqmsg。Multiplex 中辅助节点上的 IQ 消息文件的物理文件名为 <servername>.iqmsg。

由于它并非 IQ 存储数据库空间，**ALTER** 命令（如 **READONLY** 和 **OFFLINE**）不适用于 **IQ_SYSTEM_MSG**。

数据库空间

创建数据库时，系统会自动创建每个存储的第一个数据库空间。可以根据需要创建其它数据库空间。

创建和装载表时，**Sybase IQ** 会将数据分布到该存储中具有可用空间的所有现有数据库空间中。创建数据库空间时，可以为它预留空间，以便让它增长。可以将数据库空间的大小调整到最大预留值。假如所有数据均已从数据库空间的截断部分移出，则还可以使数据库空间更小。可以根据需要从指定的数据库空间中单个数据库对象移出。

不要将所有磁盘空间都分配给 **IQ** 数据库。请保留百分之十作为预留空间。**Sybase IQ** 需要此空间来正常处理空间不足情况。

请在创建数据库时创建所有数据库空间，而不要在旧数据库空间已满时再逐渐添加它们。这可确保数据库空间将更均匀地被填充，从而改进磁盘 **I/O**。

创建单独的数据库以用于调试目的。因为开发工作会增加服务器发生故障的可能性，所以 **Sybase** 建议您避免在生产数据库上执行开发工作。

IQ 存储和临时存储的空间要求

数据的数量以及您创建的索引的数目和类型将确定 **IQ** 数据库需要多少空间。

如果将数据装载到或插入数据库时空间用完，**IQ** 服务器将回退整个事务或回退到某一保存点。

除了显式定义的任何临时表以外，**Sybase IQ** 会在装载和删除期间使用临时存储作为排序、散列和位图的临时结果空间。所发出的查询类型、并发使用的程度以及数据的大小都会决定临时存储需要多少空间。

main 存储和临时存储的大小调整准则

Sybase IQ 体系结构会影响数据存储的准则。

- **IQ_SYSTEM_MAIN** 数据库空间保存除 **IQ** 表元数据以外的所有数据库元数据。**IQ** 表元数据存储于表的数据空间 and 表版本 (TLV) 日志中。如果已关闭某一节点，**Multiplex** 必须存储多个版本，以便在该节点重新启动时同步这些版本。这些版本可能会占用大量空间。
- 现在，大约有 20% 的 **IQ_SYSTEM_MAIN** **dbspace** 用于预分配的空闲列表空间，而非用于用户数据。
- 由于此版本的 **Sybase IQ** 并行执行更多的操作，因此它占用比早期版本更多的临时空间。

影响 **IQ_SYSTEM_MAIN** 存储所需的空间的因素有三个：

- 版本控制 - 维护的版本容量各不相同。
- 数据和索引的性质。

- 数据的动态性质 - 随时装载更多数据的能力。

尽管 Sybase 可以提供一般准则，但是这些因素的组合使各数据库的要求变得独一无二。对于总大小小于 500 GB 的开发或报告服务器，10 至 20 GB 的 IQ_SYSTEM_MAIN 文件可能已完全足够。

表 4. 生产数据库中的 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP 的大小准则

任务	准则	注释
从 <code>iqunload -n</code> 输出或针对小型测试数据库装载空 schema	10 GB main 存储, 5 GB 临时存储	CREATE DATABASE 大小以 MB 为单位。服务器必须为 12.7 ESD #5 或更高版本, 才能使用 <code>iqunload -n</code> 。
创建新的生产数据库	<ul style="list-style-type: none"> • 如果您正在迁移数据库, 并对当前 IQ_SYSTEM_MAIN 使用原始设备, 请分配新的标准大小的未使用原始设备。 • IQ_SYSTEM_MAIN 的总大小应至少为数据库总大小的 1/100, 并且 main 存储至少为 100 GB, 保留空间至少为 100 GB。 • 如果对 IQ_SYSTEM_MAIN 使用原始磁盘, 请尽可能使用多个原始磁盘。多个原始磁盘使 Sybase IQ 能够跨多个设备对数据执行条带化操作, 从而提高性能。 • 对于生产环境中的 IQ dbspace, Sybase 建议您使用具有容错性并由高性能的冗余磁盘阵列 (例如, RAID 5) 实现的文件系统。 	如果指定原始设备, 则省略 <code>ms_size</code> 。 始终将 main 保留大小设置为 IQ_SYSTEM_MAIN 大小的 20%。要设置 main 保留大小, 请使用数据库选项 <code>MAIN_RESERVED_DBSPACE_MB</code> 。
为 Multiplex 创建 main 存储	将 Simplex 数据库推荐空间增大一倍, 或者使 main 数据库空间至少为 200 GB, 使保留数据库空间至少为 200 GB。	

示例 1

在 **CREATE DATABASE** 语法中, 缺省大小单位是 MB, 而不是 GB。

下面的语句可创建具有 100GB IQ_SYSTEM_MAIN 及 100GB 保留空间 (以供将来扩展使用) 的数据库:

```
CREATE DATABASE 'test.db' IQ PATH 'test.iq' IQ SIZE 100000 IQ RESERVE
100000 TEMPORARY PATH 'test.iqtmp' TEMPORARY SIZE 5000
```

示例 2

`MAIN_RESERVED_DBSPACE_MB` 用于控制 Sybase IQ 在 IQ 主存储区中为释放保存点、提交和检查点操作期间使用的某些小型但极为重要的数据结构所留出的空间量。

Sybase 建议您将 `MAIN_RESERVED_DBSPACE_MB` 选项值设置为 `IQ_SYSTEM_MAIN` 大小的 20%。例如，如果 `IQ_SYSTEM_MAIN` 为 100GB，请将该选项值设置为 20GB，如下所示：

```
SET OPTION PUBLIC.MAIN_RESERVED_DBSPACE_MB = 20000
```

示例 3

可以在数据库迁移命令中指定 `IQ_SYSTEM_MAIN` 大小。

-ms_size 参数要求值以 **MB** 为单位而不是以 **GB** 为单位。如果指定原始设备，则省略 **-ms_size**。对于原始设备，必须指定未使用的原始分区。有关迁移的详细信息，请参见《安装和配置指南》。

下面的语句可在 UNIX 上的原始设备上创建 `IQ_SYSTEM_MAIN`：

```
iqunload -au -ms_filename /dev/rdsk/clt0d1 -c
"UID=DBA;PWD=SQL;DBF=latest.db"
```

下面的语句可在 Windows 上的原始设备上创建 `IQ_SYSTEM_MAIN`：

```
iqunload -au -ms_filename \\.\.\PhysicalDrive1 -c
"UID=DBA;PWD=SQL;DBF=latest.db"
```

另请参见

- 磁盘空间不足（第 455 页）
- IQ 主存储区和 IQ 临时存储空间管理（第 146 页）
- `IQ_SYSTEM_MAIN` 数据库空间（第 128 页）
- 定义数据库期间的装载性能（第 277 页）
- 主 IQ 存储块消息（第 474 页）
- 监控磁盘空间使用情况（第 459 页）
- 处理问题（第 463 页）
- Sybase IQ 停止处理或停止响应（第 449 页）

估计所需的空间和数据库空间

要避免在数据库或特定数据库空间已满时出现困难，请在创建数据库和其中的对象之前估计数据库空间要求。

可以运行 Sybase IQ 存储过程来估计数据库将需要多少空间和多少数据库空间。有关每个过程的语法和使用注意事项，请参见《参考：构件块、表和过程》>“系统过程”。

1. 运行 `sp_iqestspace` 以基于基础数据库表中的行数来估计创建数据库所需的空间数量。对于您计划创建的每个表：`sp_iqestspace table_name, rows[, iqpagesize]`
每个表所需的空间数量将作为“RAW DATA index_size”返回。
2. 将所有表的“RAW DATA index_size”下的总计相加。

- 运行存储过程 `sp_iqestjoin` 以估计对要频繁连接的表创建连接索引所需的额外空间的数量。对每一对表运行一次该过程，如下所示：`sp_iqestjoin table1, table1rows, table2, table2rows [,relation] [,iqpagesize] ...`

`sp_iqestjoin` 将根据您的查询建议不同的索引大小。

每次运行 `sp_iqestjoin` 时，选择一个建议的索引大小。如果您知道将始终完全按一对一匹配来连接表，请使用“Min Case index_size”。如果预测偶尔进行一对多连接，请使用“Avg Case index_size”。如果预测使用许多一对多连接，请使用“Max Case index_size”。

- 计算为所有表对选择的 `index_sizes` 的总计。
- 将从步骤 4 得到的连接空间总计与从步骤 2 得到的表空间总计相加，执行单独的计算以获得最小和最大连接空间。
- 运行 `sp_iqestdbspaces` 以确定要从给定的空间创建多少数据库空间文件，以及它们的大小应当是多少。使用在步骤 5 中计算的总计索引大小作为 `minsize` 和 `maxsize` 参数：

```
sp_iqestdbspaces (dbsize [,iqpagesize]
                 [,minsize] [,maxsize] ...
```

所有这些计算都是估计。结果将根据您为数据库创建的列和索引不同而异。有关每个过程的语法和使用注意事项，请参见《参考：构件块、表和过程》>“系统过程”。

另请参见

- 使其他用户与数据库断开连接（第 148 页）

dbfile 和输出文件

`SYSDATABASE` 视图可显示数据库中的所有 `dbfile`，包括 Catalog 数据库空间文件、IQ 消息文件、IQ 主数据库空间和临时数据库空间中的 `dbfile`、事务日志文件以及 SA 临时文件。

`dbfile` 以外的文件不会出现在 `SYSDATABASE` 视图中。这些文件包括服务器启动时可能生成的文件，如主控台日志（由 `-o` 开关指定）和 SQL 日志（由 `-zo` 指定）。这些日志文件不会显示为数据库属性或服务器属性，并且可以通过存储过程（例如，`sa_db_properties()` 或系统函数 `db_property()`）进行检查。（有关语法，请参见第 628 页的“表 14-1”。）

范围分区

分区是一种将大对象划分为子对象的方案。

例如：

- 存储空间分区为多个表空间
- 表可以分区为多个表分区

表对象可以是表分区、列、索引、IQ 基表、连接索引、唯一约束、主键或外键。

表分区是一个行集合，该集合是某个用户创建的表的子集。不能将一个给定行置于两个不同的分区中。每个分区可置于自己的数据库空间中，并可单独进行管理。一个分区共享其父表的逻辑属性：

- 具有相同完整性约束和缺省值的列定义
- 相同的参照完整性约束
- 相同的唯一约束和主键约束
- 相同的检查约束

表创建者可选择是否对表进行分区、如何对表进行分区以及分区的数目。表创建者可定义分区键，这是一个确定表如何分区的表列。有关语法，请参见《参考：语句和选项》>“SQL 语句”>“CREATE TABLE 语句”以及《参考：语句和选项》>“SQL 语句”>“ALTER TABLE 语句”。

分区的基本管理概念是使子对象成为只读对象的能力。一旦子对象设置为只读并且经过验证和备份后，子对象只需要最少的维护。

数据库空间管理和表分区的优点

分区和数据库空间管理允许在比表级别或数据库级别更精细的粒度上执行管理操作（数据放置、dbcc、备份、恢复）。

在 Sybase IQ 中，dbspace 管理和范围表分区：

- 提供数据放置功能
- 通过将更少的关键数据重新部署到更便宜的存储来提供分层存储管理

dbspace 管理和范围表分区通过以下方式提高可维护性和可用性：

- 支持数据库空间只读 (RO) 与读写 (RW) 状态
- 支持数据库空间联机与脱机状态
- 通过允许备份或恢复一个或多个 RO 数据库空间和/或文件或者所有 RW 文件来减少备份/恢复时间
- 支持对数据库空间或表分区目标执行数据验证。（请参见《参考：构件块、表和过程》>“系统过程”>“sp_iqcheckdb 过程”。）
- 允许使用不可用和/或无法使用的标记为“脱机”的 dbspace（Catalog 存储和 Sybase IQ 系统 dbspace 除外）启动 Sybase IQ 服务器。

分区按照业务要求组织存储和数据，从而改善管理和运行时操作，特别是备份、恢复和数据库验证。

Sybase IQ 范围分区根据单个表列的值将表分为若干逻辑分区。只能对基表进行分区；不能对全局临时表或声明的本地临时表进行分区。表分区的所有行在物理上处于同一位置，用户必须命名每个分区。

Sybase IQ 支持范围分区的最多 1024 个分区。

最常见的范围分区形式是按日期对表进行分区；例如，June_2009、July_2009 等。范围表分区可以分配给数据库空间。

对分区执行 DDL 操作的限制

可以对表分区进行删除、重命名、分区、取消分区、合并、拆分和移动，但具有下列限制：

运算	限制
删除	不能删除分区键列或分区表的最后一个分区。
重命名	不适用
对未分区表进行分区	不适用
合并两个相邻分区	两个分区必须位于同一数据库空间中。不需要移动数据。
拆分区	拆分后，所有行必须隶属于两个分区之一。拆分区必须与原始分区位于同一数据库空间上，以免移动数据。
将分区移至新数据库空间	该分区的所有行均被移至新数据库空间的数据页中。在新数据库空间中需要 CREATE 权限。

分区表不能参与连接索引。

另请参见

- 分区表装载注意事项（第 242 页）

对分区执行 DML 操作

可以执行 DML 操作，其中包括装载、插入、删除、截断和截断范围分区的表分区。支持更新操作，但不支持更新分区键列。

Sybase IQ 对只读表或只读表分区上的 DML 操作返回错误。如果给定行与指定的分区范围不匹配，插入和装载语句或者通过可更新的游标插入将产生错误，并且操作将回退。

禁止对分区表装载使用 START ROW ID 选项。在对分区表进行装载时，使用 APPEND_LOAD 选项会将新行附加到相应分区的末尾。

另请参见

- 分区表装载注意事项（第 242 页）

未分区的表的对象放置

创建对象或移动对象时，可以为表对象（包括任何数据类型的列、索引、主键和外键、唯一约束、连接索引以及未分区的表）指定数据库空间。

您必须在该数据库空间中具有 CREATE 特权。例如：

```
CREATE TABLE tab1 (
    col1 INT IN dsp1,
    col2 VARCHAR(20),
    col3 CLOB IN dsp2,
    col4 DATE,
    col5
```

```
BIGINT, PRIMARY KEY (col5) IN dsp4) IN dsp3; CREATE DATE INDEX
col4_date ON tab1(col4) IN dsp5;
```

将得到如下数据分配:

dbspace 名称	数据
dsp1	col1 的 FP 索引
dsp2	col3 的 FP 索引
dsp3	col2、col4 和 col5 的 FP 索引
dsp4	主键 (col5 上的 HG)
dsp5	DATE 索引 col4_date

分区表的对象放置

对于分区表, 可以将每个表分区放置在单独的数据库空间中。

也可以将表分区的各个列放置在单独的数据库空间中。但一般来讲, 建议只针对 BLOB 或 CLOB 列或者 CHAR、VARCHAR 或 VARBINARY 类型大于 255 字节的列使用单独的数据库空间。例如:

```
CREATE TABLE tab2( col1 INT IQ UNIQUE(65500), col2
VARCHAR(20), col3 CLOB PARTITION (p1 IN dsp11, p2 IN
dsp12, p3 IN dsp13), col4 DATE, col5 BIGINT, col6
VARCHAR(500) PARTITION (p1 IN dsp21, p2 IN dsp22), PRIMARY
KEY (col5) IN dsp2) IN dsp1 PARTITION BY RANGE (col4) (p1
VALUES <= ('2006/03/31') IN dsp31, p2 VALUES <= ('2006/06/30') IN
dsp32, p3 VALUES <= ('2006/09/30') IN dsp33 );CREATE DATE
INDEX c4_date ON tab2(col4) IN dsp3;
```

将得到如下数据分配:

分区	数据库空间	数据
p1	dsp11	col3 的 FP 索引 (CLOB 数据)
	dsp21	col6 的 FP 索引 (VARCHAR(500) 数据)
	dsp31	col1、col2、col4 和 col5 的 FP 索引
p2	dsp12	col3 的 FP (CLOB 数据)
	dsp22	col6 的 FP (VARCHAR(500) 数据)
	dsp32	col1、col2、col4 和 col5 的 FP 索引
p3	dsp13	col3 的 FP 索引 (CLOB 数据)
	dsp33	col1、col2、col4、col5 和 col6 的 FP 索引 (varchar(500) 数据)
未分区	dsp1	col1 的查找存储和其它共享数据 (对于所有分区)

分区	数据库空间	数据
未分区	dsp2	col5 上的主键 HG (对于所有分区)
未分区	dsp3	DATE 索引 col4_date (对于所有分区)

数据库定义

要定义数据库，可以使用 SQL 语句，也可以使用数据库设计工具。

某些应用程序设计系统（例如 Sybase PowerDesigner®）包含创建数据库对象的工具。这些工具通常通过其 ODBC 接口构建提交到服务器的 SQL 语句。如果您在使用这些工具中的某一个工具，则不需要构建 SQL 语句就可以完成诸如创建表、指派权限等操作。

诸如 Sybase PowerDesigner 之类的数据库设计工具可以为开发设计合理的数据库提供彻底、可靠的方法。

使用 Sybase Central 创建数据库

要在 Sybase Central 中创建 Sybase IQ 数据库，请参见《Sybase Central 插件帮助》。

使用 Sybase Central Center 创建数据库

要在 Sybase Control Center 中创建 Sybase IQ 数据库，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

Multiplex 数据库创建

要创建 Multiplex 数据库，请参见《使用 Sybase IQ Multiplex》> “Multiplex 创建”> “将数据库转换为 Multiplex”。

使用 Sybase Central Center 设置数据库选项

有关使用 Sybase Control Center 设置数据库选项的说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

使用 SQL 创建数据库

如果通过 Interactive SQL 工具（如 dbisql）生成数据库，则可以直接使用 SQL 语句定义数据库对象。

即使您使用应用程序设计工具，也可能需要使用 SQL 语句向数据库添加那些设计工具所不支持的功能。

创建完数据库之后，您就可以连接到该数据库并在该数据库中建立所需的表和其它对象。

另请参见

- CREATE DATABASE 语句缺省值 (第 140 页)
- Multiplex 功能 (第 5 页)
- 相对路径名 (第 141 页)
- 实用程序数据库服务器安全性 (第 291 页)

创建数据库之前

在使用 SQL 语句创建数据库之前执行前提操作。

要使用 SQL 语句创建数据库，必须：

- 启动数据库服务器
- 启动 **dbisql**

要在 **dbisql** 中创建数据库，您需要连接到现有数据库，否则启动 *实用程序数据库*，这是一个没有数据库文件、没有数据的幻像数据库。

注意： 如果以 **-m** 服务器选项启动服务器，则不能创建数据库。

启动实用程序数据库

如果尚未建立数据库，则必须在创建新数据库之前启动实用程序数据库。

可以用以下任何方式启动实用程序数据库：

1. 通过在启动命令中仅指定 **-n enginename**，在没有数据库的情况下启动数据库服务器。
2. 从命令行启动 **dbisql**，在连接字符串中将数据库名称设置为 **utility_db**，如下所示：

```
dbisqlc -c "uid=dba;pwd=sql;eng=myserver;dbn=utility_db;..."
```

(不得将它指定为数据库文件，因为 **utility_db** 没有数据库文件。)

3. 在 Sybase Central 中，在创建数据库向导中选择“使用远程主机上的 IQ 代理启动实用程序服务器”。

另请参见

- 实用程序数据库服务器安全性 (第 291 页)

自动创建数据库空间

创建数据库时，数据库服务器将创建以下四个数据库空间：

dbspace 名称	用途	缺省操作系统文件名
IQ_SYSTEM_MAIN	main (永久) IQ 存储文件	<i>dbname.iq</i>
IQ_SYSTEM_MSG	消息日志文件	<i>dbname.msg</i>

dbspace 名称	用途	缺省操作系统文件名
IQ_SYSTEM_TEMP	临时 IQ 存储文件	<i>dbname.iqtmp</i>
IQ_SHARED_TEMP	临时 IQ 存储	最初，此数据库空间不包含任何文件。
SYSTEM	Catalog 存储文件	<i>dbname.db</i>

SYSTEM 数据库空间包含系统表，构建数据库时，这些表保存 schema 定义。它还保存单独的检查点日志、回退日志以及（可选）Catalog 存储的写入文件、事务日志和事务日志镜像。

注意：除了这些数据库文件以外，数据库服务器还使用临时文件以保存会话期间所需的信息。此临时文件不同于 IQ 临时存储，只要数据库服务器关闭，就不再需要该文件。该文件的名称由服务器生成，扩展名为 .tmp。其位置由 TEMP 环境变量确定，或者在 UNIX 上由协调器环境变量确定。

数据库文件放置

创建数据库之前，请考虑是否仍将需要移动数据库。数据库空间文件的位置还可能会影响性能。

可以安全移动 IQ 目录 (.db) 和事务日志 (.log) 文件。*永远不要尝试复制正在运行的数据库。*如果使用相对路径名创建数据库，则可以通过关闭服务器并使用操作系统复制文件命令来移动文件。如果使用绝对（完全限定）路径名创建数据库，则必须通过使用 **BACKUP** 命令建立完整备份，并使用带有 **RENAME** 选项的 **RESTORE** 命令恢复备份来移动文件。

数据库关闭时，可以将原始分区上的 IQ 数据库空间移动到其它分区上。新分区必须至少与当前数据库空间大小一样大。新分区还必须有相同路径，才能使数据库空间启动。

警告！为数据库空间（系统数据库空间、IQ 主数据库空间或 IQ 临时数据库空间）分配文件系统文件时，请不要将文件放在通过局域网共享的文件系统上。这样做可能会导致 I/O 性能较差以及其它问题，包括使局域网过载和数据库空间文件问题。在 UNIX 和 Linux 平台上，请避免由网络文件系统 (NFS) 装入的文件系统。在 Windows 上，请不要将数据库空间文件放置在由另一个节点拥有的网络驱动器上。这些文件放置建议也适用于日志文件。

为避免冲突，Sybase 建议由单个连接上的单个数据库管理员执行 dbspace 管理。

通过增加专门用于存储随机访问的文件（包括系统 dbfile、IQ 主 dbfile 和 IQ 临时 dbfile）的磁盘驱动器数，可以提高与这些文件相关的性能。通过将顺序访问的文件（包括事务日志和消息日志文件）放在专用磁盘驱动器上，可以提高与这些文件相关的性能。

关于降低文件放置对性能影响的建议包括：

- 将随机磁盘 I/O 和顺序磁盘 I/O 分开。

- 将数据库文件、临时数据库空间和事务日志文件放在与数据库服务器相同的物理计算机上。
- 将 Sybase IQ 数据库 I/O 与其它数据库（例如 Adaptive Server Enterprise）中的代理表的 I/O 隔离。
- 将事务日志和消息日志放在单独的磁盘上，与 IQ 存储、Catalog 存储和临时存储分离，并与任何代理数据库（例如 Adaptive Server Enterprise）分离。

如果 IQ 要求的大小和复杂性足以需要多个物理系统，请考虑使用 Sybase IQ Multiplex 功能。

原始设备权限

确保所有原始设备均具有读写权限，然后您才能创建数据库或添加数据库空间。请查看 `/dev/rawctl` 是否具有读取权限。Linux 中的原始设备名使用 `/dev/raw/rdevname`。例如，`/dev/raw/raw10`。

数据库文件兼容性

Sybase IQ 服务器不能管理用 Sybase IQ 12.6 之前的版本创建的数据库；同样，旧版本服务器不能管理新数据库。

另请参见

- 系统恢复与数据库修复（第 419 页）

CREATE DATABASE 语句缺省值

CREATE DATABASE 语句具有两个必需的参数和几个可选参数。

必须指定 Catalog 存储的文件名和 **IQ PATH**。所有其它参数均可选。

如果使用所有缺省值，则数据库具有以下特性：

- 区分大小写 (**CASE RESPECT**)。“ABC”与“abc”比较不相等。缺省登录是用户 ID DBA（大写）和口令 sql（小写）。缺省情况下，口令区分大小写。用户名始终不区分大小写。
- 4096 字节的目录页大小 (**PAGE SIZE 4096**)。
- 比较不等长的两个字符串时，IQ 将较短的字符串视为用空白填充到较长字符串的长度，以便“abc”与“abc”比较相等 (**BLANK PADDING ON**)。
- 与 Adaptive Server Enterprise 不兼容。
- IQ 页大小为 128KB (**IQ PAGE SIZE 131072**)。
- IQ 消息文件和 IQ 临时存储与 Catalog 存储位于同一目录中。
- 对于原始设备，**IQ SIZE** 和 **TEMPORARY SIZE** 是原始分区的最大大小。对于操作系统文件，请参见下面对此参数的讨论。
- IQ 临时存储大小是 IQ 大小的一半。
- jConnect JDBC 驱动程序已启用 (**JCONNECT ON**)。

- 已使用归类 `ISO_BINENG`。归类顺序与 `ASCII` 字符集中的字符顺序相同。在区分大小写的数据库中，所有大写字母都位于所有小写字母的前面（例如，“A”和“B”都位于“a”的前面）。
- `IQ RESERVE` 和 `TEMPORARY RESERVE` 为 0。

另请参见

- 使用 `SQL` 创建数据库（第 137 页）
- `Multiplex` 功能（第 5 页）
- 相对路径名（第 141 页）
- 实用程序数据库服务器安全性（第 291 页）

相对路径名

您可以使用数据库中的每个文件的相对路径名或完全限定的路径名来创建数据库。

`Sybase` 建议 您用相对路径名创建数据库。如果指定 绝对路径名，则只能通过备份和恢复数据库的方法将文件移动到不同 路径名。

如果数据库在 `UNIX` 上，则可以定义每个路径名的符号链接，如《参考：语句和选项》>“`SQL` 语句”>“`CREATE DATABASE` 语句”中所述。

如果省略目录路径，则 `Sybase IQ` 按如下方式定位文件：

- `Catalog` 存储创建在服务器工作目录的相对目录下。
- `IQ` 存储、临时存储和消息日志文件都创建在 `Catalog` 存储所在的目录下或其相对目录下。
- 事务日志创建在 `Catalog` 存储所在的目录下。（如果未指定任何文件名，则也会发生这种情况。）但是，应该将它放在相同物理计算机上与 `Catalog` 存储和 `IQ` 存储不同的物理设备上。

注意： 对于任何以相对路径名创建的数据库，必须从数据库所在的目录启动数据库服务器。通过使用配置文件启动服务器，可以确保从一致的位置启动服务器。

另请参见

- `CREATE DATABASE` 语句缺省值（第 140 页）
- 使用 `SQL` 创建数据库（第 137 页）
- `Multiplex` 功能（第 5 页）
- 实用程序数据库服务器安全性（第 291 页）

IQ PATH 参数准则

所需的 `IQ PATH` 参数告诉 `Sybase IQ` 您正在创建 `IQ` 数据库，而不是 `SQL Anywhere` 数据库。

需要在此参数中指定 `IQ` 存储的位置。

请仔细选择数据库的位置。尽管可以将 `IQ` 数据库或其任何文件移动到另一位置，但要执行此操作，必须关闭数据库，并且可能必须执行备份和恢复。

可以在不同驱动器上添加空间，但只能使用此额外空间存储新数据。不能轻松地将数据的特定索引、表或行从一个位置移动到另一个位置。

每个操作系统的原始设备名都有其自己的格式。有关初始化 Sun Solaris 上的原始设备的重要注意事项，请参见《参考：构件块、表和过程》>“物理限制”。

表 5. UNIX 上的原始设备名

UNIX 平台	示例
AIX	/dev/rraw121v
HP-UX	/dev/vg03/rrchee12g
Sun Solaris	/dev/rsd0c
Sun AMD	/dev/rdisk/c5t0d0s1

表 6. Windows 上的原始设备名

设备类型	所需名称格式	示例
分区	分配给该分区的字母	\\.\C: (在 Sybase Central 中), \\.\C: (在 SQL 中)
未分区	<i>PhysicalDriveN</i> , 其中, <i>N</i> 是从 0 开始并且根据需要增大的数字。通过运行管理工具中的磁盘管理器, 可以查找物理驱动器号。	\\.\ PhysicalDrive32 (在 Sybase Central 中), \\.\ PhysicalDrive32 (在 SQL 中)

在 Windows 系统中, 如果要指定包含反斜杠的设备名, 则必须双写反斜杠以避免系统将反斜杠/字母的组合误认为转义序列 (如制表符或换行符命令)。

在 Windows 上的 SQL 语句中命名原始设备时, 必须始终双写反斜杠。

示例 1

以下语句创建名为 `company.db` 的 IQ 数据库。此数据库由四个 Windows 文件组成:

- Catalog 存储位于启动服务器的目录下的 `company.db` 中 (在此示例中为 `c:\company`)
- IQ 存储位于 `c:\company\iqdata\company.iq` 中
- 临时存储位于 `c:\company\company.iqtmp` 中
- IQ 消息日志文件位于 `c:\company\company.iqmsg` 中

```
CREATE DATABASE 'company.db'
IQ SIZE 200
IQ PATH 'c:\\company\\iqdata\\company.iq'
```

示例 2

以下语句创建名为 `company.db` 的 IQ 数据库。此数据库由四个 UNIX 文件组成：

- Catalog 存储位于启动服务器的目录下的 `company.db` 中（在此示例中为 `/disk1/company`）
- IQ 存储位于 `/disk1/company/iqdata/company.iq` 中
- 临时存储位于 `/disk1/company/iqdata/company.iqtmp` 中
- IQ 消息日志文件位于 `/disk1/company/iqdata/company.iqmsg` 中

```
CREATE DATABASE 'company.db'
IQ SIZE 2000
IQ PATH '/disk1/company/iqdata/company.iq'
```

示例 3

以下 UNIX 示例创建名为 `company` 的 IQ 数据库，具有 IQ PATH 的原始分区。

```
CREATE DATABASE 'company' IQ PATH '/dev/rdisk/c0t0d0s0'
```

示例 4

以下 Windows 示例创建名为 `company` 的 IQ 数据库，具有 IQ PATH 的原始分区。

```
CREATE DATABASE 'company' IQ PATH '\\.\.\D:'
```

IQ PAGE SIZE 参数准则

IQ PAGE SIZE 参数确定内存和磁盘使用率。

用 **IQ PAGE SIZE** 选项设置 IQ 存储的页大小。**IQ PAGE SIZE** 必须为 2 的幂，从 65536 到 524288 字节。IQ 页大小对于 IQ 存储中的所有数据库空间都是相同的。

为了获得最佳性能，Sybase 建议使用以下最小 IQ 页大小：

- **64KB (IQ PAGE SIZE 65536)**，针对其最大表不超过 10 亿行的数据库，或总大小小于 8TB 的数据库。这是新数据库的绝对最小值。在 32 位平台上，IQ 页大小设置为 64KB 可实现最佳性能。
- **128KB (IQ PAGE SIZE 131072)**，针对 64 位平台上的数据库，其最大表包含多于 10 亿且少于 40 亿的行数，或者可能增长到总大小 8TB 或更大。128KB 为缺省 IQ 页大小。
- **256KB (IQ PAGE SIZE 262144)**，针对 64 位平台上的数据库，其最大表包含多于 40 亿的行数，或者可能增长到总大小 8TB 或更大。

多用户环境和具有内存约束的系统都受益于至少 64 KB 的 IQ 页大小，因为此大小会最大程度地减少分页。

Sybase IQ 以压缩形式在磁盘上存储数据。它将数据解压缩，然后将数据页移动到内存中进行处理。IQ 页大小确定磁盘压缩的数量和 IQ 存储的缺省 I/O 传输块大小。对于大多数应用程序，此缺省值是最佳的。有关这些设置和影响资源使用和性能的其他选项的信息，请参见《性能和调优指南》>“管理系统资源”。

宽数据的页大小

如果数据库包括非常宽的表，您可能发现为给定数目的行提供下一个更高的 IQ 页大小会产生更好的性能。例如，具有多列宽 CHAR 或 VARCHAR 数据的表（从 255 到 32,767 字节的列）可能需要大于常用值的 IQ 页大小。

因为 IQ 在列中存储数据，所以它没有真正的最大行长度。但是，实际限制是 IQ 页大小的一半，因为这是查询保证能够返回到客户端的最宽结果集。请选择至少是可能的最宽表宽度的两倍的值作为 IQ 页大小。

另请参见

- 页大小（第 16 页）
- 最大目录页大小（第 28 页）
- 创建表的准则（第 171 页）
- 锁定的工作方式（第 347 页）
- 不同的值数目（第 277 页）
- 有效数据的规则和检查（第 317 页）

数据库大小准则

创建数据库时，需要设置以 MB 为单位的初始 IQ 数据库文件（IQ_SYSTEM_MAIN 数据库空间）的大小和保留大小。

对于 main 存储，这些值是在 IQ SIZE 和 IQ RESERVE 参数中定义的；对于临时存储，这些值是在 TEMPORARY SIZE 和 TEMPORARY RESERVE 中定义的。

- 对于原始分区，不需要指定 IQ SIZE 或 TEMPORARY SIZE；Sybase IQ 会确定原始设备的大小，并自动设置 IQ SIZE 和 TEMPORARY SIZE。如果指定大小，则该大小不能大于实际原始分区大小。
- 对于操作系统文件，您可以依赖于缺省值；也可以基于数据的大小指定值，范围从所需最小值到最大值 4TB（以 1MB 为单位递增）。

IQ 存储和临时存储的缺省大小和最小大小

IQ RESERVE 和 TEMPORARY RESERVE 参数会保留一个块范围，以便随后可以调整数据库空间的大小。但是，如果使 IQ RESERVE 大于所需值，则可以使用额外磁盘空间。

表 7. IQ 存储和临时存储的缺省大小和最小大小

IQ 页大小	IQ 存储的缺省大小	临时存储的缺省大小	显式指定时 IQ 存储的最小大小	显式指定时临时存储的最小大小
65536	4096000	2048000	4MB	2MB
131072	8192000	4096000	8MB	4MB

<i>IQ 页大小</i>	<i>IQ 存储的缺省大小</i>	<i>临时存储的缺省大小</i>	<i>显式指定时 IQ 存储的最小大小</i>	<i>显式指定时临时存储的最小大小</i>
262144	16384000	8192000	16MB	8MB
524288	32768000	16384000	32MB	16MB

PAGE SIZE 参数准则

可以用 **CREATE DATABASEPAGE SIZE** 选项为 Catalog 存储选择一个页大小。此选项的缺省值和最小值为 4096 (4KB)。

示例

以下语句创建具有 4KB 目录 **PAGE SIZE** 的数据库，其中，IQ 存储位于 UNIX 原始分区上，并且有 128KB 的 **IQ PAGE SIZE**。缺省情况下，IQ 存储大小是原始分区的大小，而临时存储是该大小的一半。因为没有为临时存储指定路径，所以将在 Catalog 存储所在的目录中创建它。

```
CREATE DATABASE 'company' IQ PATH '/dev/rdsk/c2t6d0s3' PAGE SIZE 4096
IQ PAGE SIZE 131072
```

块大小准则

在几乎所有情况下，都应当依赖于缺省块大小，它基于 IQ 页大小。

在 Sybase Central 中设置数据库选项

数据库选项是更改数据库的行为或执行方式的可配置设置。

1. 打开服务器。
2. 右键单击数据库，选择“选项”。
3. 对值进行编辑。

注意： 使用“数据库选项”对话框，还可以设置特定用户和组的数据库选项。

设置数据库自身的选项时，实际上是为该数据库中的 **PUBLIC** 组设置选项，因为所有用户和组都从 **PUBLIC** 继承选项设置。

- 请参见《参考：语句和选项》>“数据库选项”。

另请参见

- 资源问题（第 455 页）

在 Interactive SQL 中设置数据库选项

数据库选项是更改数据库的行为或执行方式的可配置设置。

在 **SET OPTION** 语句中指定属性。

注意： 设置数据库自身的选项时，实际上是为该数据库中的 **PUBLIC** 组设置选项，因为所有用户和组都从 **PUBLIC** 继承选项设置。

IQ 主存储区和 IQ 临时存储空间管理

选项 `MAIN_RESERVED_DBSPACE_MB` 和 `TEMP_RESERVED_DBSPACE_MB` 为检查点、提交和发布保存点操作提供空间。

这些选项分别确定 `IQ_SYSTEM_MAIN` 或 `IQ_SYSTEM_TEMP` 中的最后一个读写 `dbfile` 中保留空间分配大小。

具有 `DBA` 权限的用户可以限制每个连接所使用的空间量。此外，当 Sybase IQ 的 `IQ` 主存储区或 `IQ` 临时存储的空间不足时，在添加新空间之前，服务器将不再挂起空间不足的事务。`IQ` 主存储区或 `IQ` 临时存储中空间不足的事务将失败并回退。

数据库选项 `MAX_TEMP_SPACE_PER_CONNECTION` 限制每个连接使用的 `IQ` 临时存储的空间量，并跟踪所有数据操作语言 (DML) 语句和查询的临时存储使用情况。`MAX_TEMP_SPACE_PER_CONNECTION` 监控和限制语句的实际运行时临时存储使用情况。如果连接超过 `MAX_TEMP_SPACE_PER_CONNECTION` 选项设置的限额，则会返回错误，当前语句将回退。

`QUERY_TEMP_SPACE_LIMIT` 数据库选项的缺省值为 0，这意味着不会限制查询的临时存储使用情况。要限制每个连接的临时存储使用情况，`DBA` 可以对所有 DML 语句（包括查询）设置 `MAX_TEMP_SPACE_PER_CONNECTION` 选项。

当从 15.0 版本之前的版本升级 Sybase IQ 数据库时，`MAX_TEMP_SPACE_PER_CONNECTION` 数据库选项设置为缺省值 0。您可以在升级前后使用 `sp_iqcheckoptions` 查找选项的缺省值和当前值，以帮助确定新选项设置是否适用于已升级的数据库。

请参见《参考：语句和选项》>“数据库选项”。

另请参见

- 磁盘空间不足（第 455 页）
- `IQ_SYSTEM_MAIN` 数据库空间（第 128 页）
- 定义数据库期间的装载性能（第 277 页）
- 主 `IQ` 存储块消息（第 474 页）
- 监控磁盘空间使用情况（第 459 页）
- 处理问题（第 463 页）
- `main` 存储和临时存储的大小调整准则（第 130 页）
- Sybase IQ 停止处理或停止响应（第 449 页）
- 资源问题（第 455 页）
- 查找当前执行的语句（第 479 页）
- 记录服务器请求（第 479 页）

显示 Sybase Central 中的系统对象

在数据库中，表、视图、存储过程或域是系统对象。

1. 打开服务器。
2. 右键单击连接的数据库，然后选择“配置所有者/容器过滤”。
3. 启用 **SYS** 和 **dbo**，然后单击“确定”。

系统表、系统视图、系统过程和系统域出现在其各自的文件夹中（例如，系统表出现在 Tables 文件夹中常规表的旁边）。系统视图由 **SYS** 用户 ID 拥有。

另请参见

- 优化存储和查询性能（第 174 页）

显示 Interactive SQL 中的系统对象

在 Interactive SQL 中，不能查询系统表，但可以浏览系统视图的内容。

大多数系统表都具有可查询的等同系统视图。

在数据库中，表、视图、存储过程或域是系统对象。系统表存储有关数据库自身的信息，而系统过程和域大多支持 Sybase Transact-SQL 兼容性。

1. 使用 Interactive SQL 连接到数据库。
2. 执行 **SELECT** 语句，指定要浏览的表的系统视图。

要浏览 ISYSTAB 系统表，请在“结果”窗格中显示 **SYS.SYSTAB** 视图的内容。

```
SELECT *  
FROM SYS.SYSTAB
```

在 Sybase Central 中与数据库断开连接

处理完数据库后，可以与它断开连接。

1. 打开所需的服务器。
2. 选择所需数据库。
3. 在工具栏上，单击“断开连接”按钮。

在 Interactive SQL 中与数据库断开所有连接

处理完数据库后，可以断开指定的连接或所有连接。
执行 **DISCONNECT** 语句。

以下语句显示如何通过 Interactive SQL 使用 **DISCONNECT** 断开所有连接：

```
DISCONNECT ALL
```

另请参见

- 数据存储 (第 149 页)
- 数据库空间管理示例 (第 164 页)
- 磁盘空间 (第 15 页)
- DDL 操作锁 (第 348 页)
- 临时表的版本控制 (第 346 页)

在嵌入式 SQL 中与数据库断开连接

处理完数据库后，可以断开指定的连接或所有连接。
执行 **EXEC SQL DISCONNECT** 语句。

以下语句显示如何在嵌入式 SQL 中使用 DISCONNECT:

```
EXEC SQL DISCONNECT :conn_name
```

另请参见

- 数据库空间管理权限 (第 286 页)

使其他用户与数据库断开连接

Sybase IQ 还可以让您使其他用户与给定数据库断开连接。

通过使用 **connection_property** 函数请求连接号，可以获得用户的 *connection-id*。下面的语句将返回当前连接的连接 ID:

```
SELECT connection_property( 'number' )
```

1. 使用 DBA 权限连接到现有数据库。
2. 使用 Interactive SQL 执行 **DROP CONNECTION** 语句。

以下语句将删除 ID 号为 4 的连接。

```
DROP CONNECTION 4
```

另请参见

- 估计所需的空间和数据库空间 (第 132 页)

删除数据库

如果删除数据库，将从磁盘中删除所有表和数据，包括记录数据库更改的事务日志。它还会删除与数据库关联的所有数据库空间。

要删除数据库，请使用以下 SQL 语句:

```
DROP DATABASE dbname
```

必须完全按创建数据库时指定的相同信息指定数据库名称及其路径名。

例如，在 Windows 系统上:

```
DROP DATABASE 'c:\sybase\data\mydb.db'
```

必须先停止数据库，然后才能删除它。如果使用了 **AUTOSTOP=no** 连接参数，您可能需要发出 **STOP DATABASE** 语句。

数据存储

DBA 可以确定哪些表、索引和连接索引驻留在给定数据库空间上，将这些对象重新部署到其它数据库空间，并在清空任意数据库空间中的数据后将其删除。

DBA 还可以定义在将磁盘条带化算法移动到下一带区之前每个数据库空间的写入次数。

另请参见

- 数据库空间管理示例（第 164 页）
- 磁盘空间（第 15 页）
- DDL 操作锁（第 348 页）
- 临时表的版本控制（第 346 页）
- 在 Interactive SQL 中与数据库断开所有连接（第 147 页）

dbfile 属性和操作

dbfile 具有读写或只读状态。

添加 dbfile 时，该文件为读写状态，其运行时读写状态取决于数据库空间和 dbfile 的读写状态。管理员可以更改 dbfile 的读写/只读状态，但不能更改 dbfile 的联机/脱机状态。

可对 dbfile 执行的操作包括添加、删除、重命名逻辑名和重命名文件路径。请参见《参考：语句和选项》>“SQL 语句”>“ALTER DBSPACE 语句”。

数据库空间属性和操作

数据库空间状态可以是联机、脱机或动态脱机。

动态脱机表示数据库空间在内存中标记为脱机（相对于在目录中标记为脱机）。如果数据库已启动，并且无法打开一个或多个 dbfile，仍会启动该数据库，但会将数据库空间标记为动态脱机。在修复问题之后，管理员可以使用 **ALTER DBSPACE ONLINE** 使数据库空间重新联机，但这只能更改数据库空间的内存中状态。

除联机、脱机或动态脱机状态以外，数据库空间还具有读写或只读状态。创建后，数据库空间即处于联机和读写状态。

数据库空间还有条带化属性。管理员可以指定是否启用条带化功能及带区大小。

可对数据库空间执行的操作包括添加、删除和重命名。请参见《参考：语句和选项》>“SQL 语句”>“CREATE DBSPACE 语句”、《参考：语句和选项》>“SQL 语句”>“DROP 语句”以及《参考：语句和选项》>“SQL 语句”>“ALTER DBSPACE 语句”。

有关 Multiplex 数据库空间，请参见《使用 Sybase IQ Multiplex》> “Multiplex 事务”。要在 Sybase Control Center 中更改数据库空间的状态，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

只读和读写数据库空间和文件

如果文件状态为只读或读写，但拥有数据库空间状态为只读，该文件则为只读。

将数据库空间更改为只读并不会将其关联文件的目录状态更改为只读，但会在操作系统级别上将关联文件更改为只读。换言之，文件的目录只读或读写状态会保持不变，但无法修改该文件中的数据。

对于只读数据库空间，管理员可以：

- 添加文件
- 重命名数据库空间中的 dbfile 的文件路径（要求主数据库空间脱机）
- 删除空文件
- 重命名数据库空间或数据库空间中的 dbfile

数据库空间和关联文件的状态

数据库空间及其关联文件可以具有独立的只读 (RO) 或读写 (RW) 状态，例如：

对象	状态	有效状态	表的数据库空间	表的状态
dbspace1	RW	RW	dbspace1	RW
- file1	RO	RO		
- file2	RW	RW		
dbspace2	RO	RO	dbspace2	RO
- file1	RO	RO		
- file2	RW	RO		
dbspace3	RW	RO	dbspace3	RO
- file1	RO	RO		
- file2	RO	RO		

将表或连接索引分配给只读数据库空间时，表或连接索引为只读。将表分区分配给只读数据库空间时，表分区为只读。不允许对只读表或只读表分区执行任何数据修改，例如，插入、删除、更新、装载、截断表和通过可更新游标执行的插入/删除/更新。不允许对只读表或只读表分区执行任何 DDL 操作，例如，ALTER TABLE 添加/删除列、创建/删除索引。

将修改过的页刷新到磁盘时，系统将检测到对只读 dbspace 的写入尝试。在执行 **INSERT...VALUES** 语句期间修改的页不会写入到数据库，直到执行下一个除

INSERT...VALUES 语句以外的命令为止。（**INSERT...VALUES** 是具有这种行为的唯一命令。） Sybase IQ 对只读表或只读表分区上的 DML 操作返回 错误。

如果任何连接表为只读，对连接索引执行的操作（包括创建、删除和同步连接索引）将失败。

允许的数据库空间转换

更改数据库空间配置的能力取决于更改操作的类型以及特定的数据库空间属性。更改操作由数据库空间的状态（联机或脱机）、数据库空间的读写状态以及数据库空间的类型控制。

表 8. 允许的数据库空间配置转换

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
联机数据库空间			
	ALTER DBSPACE OFFLINE	是（如果为 RO）	否
	ALTER DBSPACE ONLINE	否	是（对于 IQ_SYSTEM_MAIN），否（对于 IQ_SHARED_TEMP 和 IQ_SYSTEM_TEMP）
	ALTER DBSPACE READONLY	是（如果为 RW）	否
	ALTER DBSPACE READWRITE	是（如果为 RO）	否
	ALTER STRIPING 或 STRIPESIZEKB	是	是（在 Simplex 和 Multiplex 协调器上）
	RENAME DBSPACE	是	否
	ADD FILE	是	是
	DROP FILE	是（如果为空和 RO）	是（如果为空和 RO）。删除 IQ_SHARED_TEMP 中的文件要求必须在以单节点模式启动协调器后在其中执行此操作。Simplex 中也支持删除 IQ_SHARED_TEMP 中的文件。IQ_SHARED_TEMP 中第一个设置为 RW 的文件必须是要删除的最后一个文件。

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER FILE READONLY	是 (如果为 RW)	是 (如果为 RW, 且不是最后一个 RW dbfile)。不允许将 IQ_SHARED_TEMP 中的文件设置为只读。
	ALTER FILE READWRITE	是 (如果为 RO)	是 (如果为 RO)
	ALTER FILE SIZE	是 (如果为 RW 数据库空间和 dbfile)	是 (如果为 RW)
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是 (对于 IQ_SHARED_TEMP 和 IQ_SYSTEM_TEMP) ; 在数据库重新启动后生效。否 (对于 IQ_SYSTEM_MAIN) 。
脱机数据库空间			
	ALTER DBSPACE OFFLINE	否	不适用
	ALTER DBSPACE ONLINE	是	不适用
	ALTER DBSPACE READONLY	否	不适用
	ALTER DBSPACE READWRITE	否	不适用
	ALTER STRIPING 或 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	否	不适用
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE RO	是	不适用
	ALTER FILE RW	是	不适用

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	是	不适用
动态脱机数据库空间			
	ALTER DBSPACE OFFLINE	是 (如果为 RO)	不适用
	ALTER DBSPACE ONLINE	是	不适用 (对于 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP), 是 (对于 IQ_SHARED_TEMP)
	ALTER DBSPACE READONLY	是 (如果为 RW)	不适用
	ALTER DBSPACE READWRITE	否	不适用
	ALTER STRIPING 或 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	否	不适用
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE READONLY	否	不适用
	ALTER FILE READWRITE	否	不适用
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	否	不适用

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
只读数据库空间			
	ALTER DBSPACE OFFLINE	是 (如果联机)	不适用
	ALTER DBSPACE ONLINE	是 (如果脱机)	不适用
	ALTER DBSPACE READONLY	否	不适用
	ALTER DBSPACE READWRITE	是 (如果联机)	不适用
	ALTER STRIPING 和 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	是	不适用
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE READONLY	是 (如果为 RW)	不适用
	ALTER FILE READWRITE	是 (如果为 RO)	不适用
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	是 (如果脱机)	不适用
读写数据库空间			
	ALTER DBSPACE OFFLINE	否	否
	ALTER DBSPACE ONLINE	是 (如果动态脱机)	否 (对于 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP), 是 (如果 IQ_SHARED_TEMP 动态脱机)

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER DBSPACE READONLY	是	否
	ALTER DBSPACE READWRITE	否	否
	ALTER STRIPING 和 STRIPESIZEKB	是	是
	RENAME DBSPACE	是	否
	ADD FILE	是	是
	DROP FILE	是 (如果为空)	是 (如果为空和 RO)。删除 IQ_SHARED_TEMP 中的文件要求必须在以单节点模式启动协调器后在其中执行此操作。Simplex 中也支持删除 IQ_SHARED_TEMP 中的文件。IQ_SHARED_TEMP 中第一个设置为 RW 的文件必须是删除的最后一个文件。
	ALTER FILE READONLY	是 (如果为 RW)	是 (如果为 RW)。不允许将 IQ_SHARED_TEMP 中的文件设置为只读。
	ALTER FILE READWRITE	是 (如果为 RO)	是 (如果为 RO)
	ALTER FILE SIZE	是 (如果为 RW)	是 (如果为 RW)
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是, 在数据库重新启动后生效
只读文件			
	ALTER FILE READONLY	否	否
	ALTER FILE READWRITE	是	是
	ALTER FILE SIZE	否	否

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	是 (如果脱机)	是, 在数据库重新启动后生效
读写文件			
	ALTER FILE READONLY	是	是
	ALTER FILE READWRITE	否	否
	ALTER FILE SIZE	是 (如果数据库空间为 RW 且处于联机状态)	是
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是, 在数据库重新启动后生效
<p>注释</p> <ul style="list-style-type: none"> • 动态脱机表示数据库空间在内存中标记为脱机 (相对于在目录中标记为脱机)。 • 只读 (RO) IQ_SYSTEM_MAIN dbfile 可以为动态脱机。 • 对于 IQ_SYSTEM_MSG 而言, 允许的唯一修改是重命名路径, 这是通过使用 ALTER DBSPACE IQ_SYSTEM_MSG RENAME 'filepath' 命令来实现的。 			

表 9. 允许的 dbspace 配置转换

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
联机数据库空间			
	ALTER DBSPACE OFFLINE	是 (如果为 RO)	否

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER DBSPACE ONLINE	否	是 (对于 IQ_SYSTEM_MAIN) , 否 (对于 IQ_SHARED_TEMP 和 IQ_SYSTEM_TEMP)
	ALTER DBSPACE READONLY	是 (如果为 RW)	否
	ALTER DBSPACE READWRITE	是 (如果为 RO)	否
	ALTER STRIPING 或 STRIPESIZEKB	是	是 (在 Simplex 和 Multiplex 协调器上)
	RENAME DBSPACE	是	否
	ADD FILE	是	是
	DROP FILE	是 (如果为空和 RO)	是 (如果为空和 RO) 。删除 IQ_SHARED_TEMP 中的文件要求必须在以单节点模式启动协调器后在其中执行此操作。Simplex 中也支持删除 IQ_SHARED_TEMP 中的文件。IQ_SHARED_TEMP 中第一个设置为 RW 的文件必须是要删除的最后一个文件。
	ALTER FILE READONLY	是 (如果为 RW)	是 (如果为 RW, 且不是最后一个 RW dbfile) 。不允许将 IQ_SHARED_TEMP 中的文件设置为只读。
	ALTER FILE READWRITE	是 (如果为 RO)	是 (如果为 RO)
	ALTER FILE SIZE	是 (如果为 RW 数据库空间和 dbfile)	是 (如果为 RW)
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是 (对于 IQ_SHARED_TEMP 和 IQ_SYSTEM_TEMP) ; 在数据库重新启动后生效。否 (对于 IQ_SYSTEM_MAIN) 。

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
脱机数据库空间			
	ALTER DBSPACE OFFLINE	否	不适用
	ALTER DBSPACE ONLINE	是	不适用
	ALTER DBSPACE READONLY	否	不适用
	ALTER DBSPACE READWRITE	否	不适用
	ALTER STRIPING 或 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	否	不适用
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE RO	是	不适用
	ALTER FILE RW	是	不适用
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	是	不适用
动态脱机数据库空间			
	ALTER DBSPACE OFFLINE	是 (如果为 RO)	不适用
	ALTER DBSPACE ONLINE	是	不适用 (对于 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP), 是 (对于 IQ_SHARED_TEMP)

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER DBSPACE READONLY	是 (如果为 RW)	不适用
	ALTER DBSPACE READWRITE	否	不适用
	ALTER STRIPING 或 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	否	不适用
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE READONLY	否	不适用
	ALTER FILE READWRITE	否	不适用
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	否	不适用
只读数据库空间			
	ALTER DBSPACE OFFLINE	是 (如果联机)	不适用
	ALTER DBSPACE ONLINE	是 (如果脱机)	不适用
	ALTER DBSPACE READONLY	否	不适用
	ALTER DBSPACE READWRITE	是 (如果联机)	不适用
	ALTER STRIPING 和 STRIPESIZEKB	是	不适用
	RENAME DBSPACE	是	不适用
	ADD FILE	是	不适用

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	DROP FILE	是 (如果为空)	不适用
	ALTER FILE READONLY	是 (如果为 RW)	不适用
	ALTER FILE READWRITE	是 (如果为 RO)	不适用
	ALTER FILE SIZE	否	不适用
	ALTER FILE RENAME LOGICAL NAME	是	不适用
	ALTER FILE RENAME PATH	是 (如果脱机)	不适用
读写数据库空间			
	ALTER DBSPACE OFFLINE	否	否
	ALTER DBSPACE ONLINE	是 (如果动态脱机)	否 (对于 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP), 是 (如果 IQ_SHARED_TEMP 动态脱机)
	ALTER DBSPACE READONLY	是	否
	ALTER DBSPACE READWRITE	否	否
	ALTER STRIPING 和 STRIPESIZEKB	是	是
	RENAME DBSPACE	是	否
	ADD FILE	是	是
	DROP FILE	是 (如果为空)	是 (如果为空和 RO)。删除 IQ_SHARED_TEMP 中的文件要求必须在以单节点模式启动协调器后在其中执行此操作。Simplex 中也支持删除 IQ_SHARED_TEMP 中的文件。IQ_SHARED_TEMP 中第一个设置为 RW 的文件必须是删除的最后一个文件。

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
	ALTER FILE READONLY	是 (如果为 RW)	是 (如果为 RW)。不允许将 IQ_SHARED_TEMP 中的文件设置为只读。
	ALTER FILE READWRITE	是 (如果为 RO)	是 (如果为 RO)
	ALTER FILE SIZE	是 (如果为 RW)	是 (如果为 RW)
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是, 在数据库重新启动后生效
只读文件			
	ALTER FILE READONLY	否	否
	ALTER FILE READWRITE	是	是
	ALTER FILE SIZE	否	否
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	是 (如果脱机)	是, 在数据库重新启动后生效
读写文件			
	ALTER FILE READONLY	是	是
	ALTER FILE READWRITE	否	否
	ALTER FILE SIZE	是 (如果数据库空间为 RW 且处于联机状态)	是
	ALTER FILE RENAME LOGICAL NAME	是	是
	ALTER FILE RENAME PATH	否	是, 在数据库重新启动后生效

状态	更改类型	用户 Main 是否允许	IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP 是否允许
<p>注意</p> <ul style="list-style-type: none"> • 动态脱机表示 dbspace 在内存中标记为脱机（相对于在目录中标记为脱机）。 • 只读 (RO) IQ_SYSTEM_MAIN dbfile 可以为动态脱机。 • 对于 IQ_SYSTEM_MSG 而言，允许的唯一修改是重命名路径，这是通过使用 ALTER DBSPACE IQ_SYSTEM_MSG RENAME 'filepath' 命令来实现的。 			

数据库空间重命名准则

可以重命名数据库空间或 dbfile 名称，但不能重命名或删除 Catalog 数据库空间 SYSTEM、IQ 主数据库空间 IQ_SYSTEM_MAIN、IQ 临时数据库空间 IQ_SYSTEM_TEMP、共享临时数据库空间 IQ_SHARED_TEMP 和 IQ 消息数据库空间 IQ_SYSTEM_MSG。

可以重命名 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP 中的文件的逻辑名，并且可以更改 IQ_SYSTEM_MSG 文件的逻辑名，但不能更改 SYSTEM 中的文件的逻辑名。可以使用 ALTER DBSPACE RENAME TO 重命名数据库空间 IQ_SYSTEM_MAIN、IQ_SYSTEM_TEMP、IQ_SYSTEM_MSG 或 SYSTEM。

其它数据库空间

使用 **CREATE DBSPACE** 语句或 Sybase Central 创建数据库空间向导创建数据库空间。

新数据库空间可以位于与现有数据库空间相同或不同的磁盘驱动器上。您必须具有 DBA 或 SPACE ADMIN 权限。

有关原始设备上的数据库空间和操作系统文件的最大大小，请参见《参考：构件块、表和过程》>“物理限制”。在一些平台上，必须使大文件系统文件能够达到最大大小。

只能为 IQ 存储和 IQ 临时存储指定 **SIZE** 和 **RESERVE**，不能为 Catalog 存储指定它们。

Sybase 建议您在原始设备上创建 main 存储。

当您为新的数据库空间指定原始设备时，Sybase IQ 会自动确定其文件大小，并将整个设备分配为 IQ 存储。这可能会在文件设备上产生不可预料的结果。

如果您指明该设备不是原始设备，则可以指定文件大小。向导会检验指定的路径是否存在。如果该路径不存在，则 Sybase IQ 会返回错误。

数据库空间数如何影响资源使用和性能

每个数据库的最大 dbspace 数目是可以调整的操作系统限制；每个 IQ 数据库最多拥有 2,047 个 dbspace，再加上用于 Catalog 存储的最多 12 个 dbspace。增加 dbspace 的数目不会对内存使用或性能造成真正的影响；应避免出现接近最大值的情况。

注意：在 HP 和 AIX 平台上，将数据分配到更多数据库空间时，会提高重叠的 I/O 性能。

将数据存储于原始分区上时，每个驱动器都可以有一个数据库空间。有关数据库空间大小限制，请参见《参考：构件块、表和过程》>“物理限制”。

将数据存储于文件系统中时，可以利用存储系统中的条带化功能。如果在多用户系统上使用操作系统或硬件带区，则带区大小应该至少为 1 MB，或可能的最大大小。带区大小应该是 IQ 页大小的数倍。还可以将 IQ 配置为执行软件条带化。

在添加数据库空间之前，可能需要估计空间要求。为了达到最有效的资源使用，应使数据库空间小到足以适合放在备份介质上，并且大到足以填满磁盘。

示例

以下命令创建一个名为 `library` 的新数据库空间，指向 `IQ_SYSTEM_MAIN` 数据库空间所在的目录中的名为 `library.iq` 的操作系统文件：

```
CREATE DBSPACE library
USING FILE library
'library.iq' SIZE 100 MB IQ STORE
```

要在 Sybase Central 中创建数据库空间，请参见联机帮助。要在 Sybase Control Center 中创建数据库空间，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

在添加或删除数据库空间之后，发出 **CHECKPOINT**。系统恢复将从最近的检查点之后开始。

另请参见

- 从 Sybase Central 连接到演示数据库（第 49 页）
- 从 Interactive SQL 连接到演示数据库（第 49 页）
- 创建外键（第 178 页）

删除数据库空间的准则

发出 **DROP DBSPACE** 命令可以删除数据库文件。

要删除数据库空间：

- 您必须具有 DBA 或 SPACE ADMIN 权限。
- 它不能包含用户表或连接索引中的任何数据。Sybase IQ 不允许删除不为空的任何数据库空间。
- 它不能是必需的数据库空间：SYSTEM、IQ_SYSTEM_MAIN、IQ_SYSTEM_TEMP 或 IQ_SYSTEM_MSG。永远不能删除这些数据库空间，但是如果数据库空间不包含用户创建的对象，则可以删除 IQ 主存储区或 Catalog 存储中的其它数据库空间。

要清空数据库空间，必须：

- 重新部署或删除数据库空间上的所有对象。
- 仅提交或回退使用早期版本的表的事务。

由于 Sybase IQ 用数据填充数据库空间的方式，数据库空间不可能仅在显式重新部署表和连接索引之后成为空的，尤其是磁盘条带化正在使用的情况下。通常，无法通过截断数据库空间中的表来清空数据库空间，因为即使空表也会占用一些空间。要重新部署表，请使用 **ALTER TABLE MOVE**。

如果在其他用户正在从表中读取数据时重新部署该表，将应用表版本控制的常规规则，也就是，旧表版本将一直存在到读取者的事务完成为止。

通过运行存储过程 **sp_iqspaceinfo**、**sp_iqdbspaceinfo** 和 **sp_iqindexinfo**，DBA 可以确定表和索引位于哪个数据库空间中。这些过程显示每个表、连接索引和索引在每个数据库空间中所使用的块数。

要确定是否可以删除特定数据库空间，请运行 **sp_iqdbspace**。请查看“Block Types”列 (Blk Types)，该列显示每个数据库空间的内容。如果数据库空间只包含块类型“H”、“F”、“R”、“B”、“X”和“C”，则可以删除该数据库空间。

块类型“A”是来自活动表版本的数据。使用 **sp_iqdbspaceinfo** 可以确定要重新部署哪些表。

块类型“O”表示可能仍然正在使用中的旧版本。必须回退或提交活动连接，才能释放此空间。块类型“M”表示 Multiplex。

有关使用 Sybase Control Center 删除数据库空间的说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

数据库空间管理示例

本节演示数据库空间管理过程：从创建新数据库并将对象及数据添加到数据库，到重新部署对象和删除空的数据库空间。此示例包括相关系统存储过程的示例 SQL 代码和输出。

创建数据库对象

使用以下方式创建小型数据库 **dbspacedb**：

```
CREATE DATABASE 'D:\IQ\dbspacedb'      IQ_PATH 'D:\IQ\dbspacedb.iq'  
      IQ_SIZE 10      IQ_RESERVE 100      TEMPORARY_SIZE 10  
      TEMPORARY_RESERVE 10      JCONNECT OFF;
```

连接到 **dbspacedb** 数据库：

```
CONNECT DATABASE dbspacedb  
      user DBA identified by sql;
```

将两个数据库空间添加到 **dbspacedb** 数据库：

```
CREATE DBSPACE dbspacedb2 USING FILE dbspace2 'D:\IQ\dbspacedb.iq2'
SIZE 10 RESERVE 20MB;CREATE DBSPACE dbspacedb3 USING FILE dbspace3
'D:\IQ\dbspacedb.iq3' SIZE 10 RESERVE 40MB;
```

更改数据库空间的大小

上例中的 **main** 存储对于表来说太小，无法添加到下一个示例中。本节中的 **ALTER DBSPACE** 命令可更改数据库空间大小。

数据库 **dbspacedb** 有用于 **IQ** 主存储区的 100MB 的保留大小，这是使用 **CREATE DATABASE** 语句的 **IQ RESERVE** 参数设置的。此 **IQ** 主存储区 (**IQ_SYSTEM_MAIN** 数据库空间) 可以扩展 100MB。原始 **IQ_SYSTEM_MAIN** 在创建时的大小为 10 MB (**CREATE DATABASE** 的 **IQ SIZE** 参数)。以下包含 **ADD** 参数的 **ALTER DBSPACE** 命令将 **IQ_SYSTEM_MAIN** 数据库空间由 10MB 扩展到 20MB:

```
ALTER DBSPACE IQ_SYSTEM_MAIN ADD 10mb; sp_iqdbspace;
```

DBSpaceName	DBSpaceType	Writable	Online
IQ_SYSTEM_MAIN	MAIN	T	T
IQ_SYSTEM_TEMP	TEMPORARY	T	T
dbspacedb2	MAIN	T	T
dbspacedb3	MAIN	T	T

Usage	TotalSize	Reserve	NumFiles	NumRWFiles
25	20M	90M	1	1
7	10M	10M	1	1
1	10N	20M	1	1
1	10M	40M	1	1

Stripingon	StripeSize	BlkTypes	OK ToDrop
T	1K	1H,1248F,32D,177A,128M	N
T	1K	1H,64F,16A	N
T	1K	1H	Y
T	1K	1H	Y

如果创建的 **dbspacedb** 数据库不包含 **IQ RESERVE** 值，则无法扩展数据库空间。但是，可以使数据库空间变得更小，从数据库空间中扣除的大小将添加到保留大小中。

IQ_SYSTEM_MAIN 数据库空间的大小现在为 20MB。将数据库空间的大小调整为 15MB:

```
ALTER DBSPACE IQ_SYSTEM_MAIN SIZE 15mb; sp_iqdbspace;
```

DBSpaceName	DBSpaceType	Writable	Online
IQ_SYSTEM_MAIN	MAIN	T	T
IQ_SYSTEM_TEMP	TEMPORARY	T	T
dbspacedb2	MAIN	T	T
dbspacedb3	MAIN	T	T

Usage	TotalSize	Reserve	NumFiles	NumRWFiles
25	15M	95M	1	1
7	10M	10M	1	1
1	10N	20M	1	1
1	10M	40M	1	1

Stripingon	StripeSize	BlkTypes	OK ToDrop
T	1K	1H,1248F,32D,177A,128M	N
T	1K	1H,64F,16A	N
T	1K	1H	Y
T	1K	1H	Y

减小数据库空间大小也只能在截断的部分未被使用的情况下才能进行。使用 **sp_iqdbspaceinfo** 可以确定数据库空间上的对象正在使用哪些块。

添加数据库对象

在 dbspacedb 数据库中创建两个表，创建索引，并添加一些数据:

```
CREATE TABLE t1(c1 int); CREATE TABLE t2(c1 int); CREATE hg INDEX t1c1hg ON t1(c1); CREATE hng INDEX t2c1hng ON t2(c1); INSERT t1 VALUES(1); INSERT t2 VALUES(2); COMMIT;
```

显示有关数据库空间的信息

显示 dbspacedb 数据库中的所有数据库空间的相关信息。以下示例演示 iqdemo 数据库中的数据库空间。输出可分为两部分，以提高可读性:

```
sp_iqdbspace;
```

DBSpaceName	DBSpaceType	Writable	Online	Usage	TotalSize	Reserve
IQ_SYSTEM_MAIN	MAIN	T	T	25	50M	100M
IQ_SYSTEM_TEMP	TEMPORARY	T	T	7	10M	10M
dbspacedb2	MAIN	T	T	1	10N	20M
dbspacedb3	MAIN	T	T	1	10M	40M

NumFiles	NumRWFiles	Stripingon	StripeSize	BlkTypes	OK ToDrop
1	1	T	1K	1H,1248F,32D, 177A,128M	N
1	1	T	1K	1H,64F,16A	N
1	1	T	1K	1H	Y
1	1	T	1K	1H	Y

显示特定数据库空间的对象放置和空间使用情况的相关信息。

注意： 下面的两个示例演示 iqdemo 数据库中的对象，以便更好地描述输出。请注意，iqdemo 包括一个名为 iq_main 的用户数据库空间示例，您自己的数据库中可能不存在该示例。

以下输出可分为几部分，以提高可读性：

```
sp_iqdbspaceinfo;
```

dbspace_name	object_type	owner	object_name	object_id	id
iq_main	table	DBA	empl	3,813	743
iq_main	table	DBA	iq_dummy	3,801	742
iq_main	table	DBA	sale	3,822	744
iq_main	table	GROUPO	Contacts	3,662	734
iq_main	table	GROUPO	Customers	3,639	733
iq_main	table	GROUPO	Departments	3,756	740
iq_main	table	GROUPO	Employees	3,765	741

dbspace_name	object_type	owner	object_name	object_id	id
iq_main	table	GROUPO	FinancialCodes	3,736	738
iq_main	table	GROUPO	FinancialData	3,745	739
iq_main	table	GROUPO	Products	3,717	737
iq_main	table	GROUPO	SalesOrderItems	3,704	736
iq_main	table	GROUPO	SalesOrders	3,689	735

columns	indexes	metadata	primary_key
96K	0B	1.37M	0B
24K	0B	464K	0B
96K	0B	1.22M	0B
288K	0B	5.45M	24K
240K	48K	4.63M	24K
72K	0B	1.78M	24K
408K	0B	8.03M	24K
72K	0B	1.53M	24K
96K	0B	2.19M	24K
272K	192K	4.67M	24K
120K	0B	2.7M	24K
144K	0B	3.35M	24K

unique_constraint	foreign_key	dbspace_online
0B	0B	Y
0B	0B	Y
0B	0B	Y

unique_constraint	foreign_key	dbspace_online
0B	48K	Y
0B	0B	Y
0B	48K	Y
0B	48K	Y
0B	0B	Y
0B	48K	Y
0B	0B	Y
0B	104K	Y
0B	144K	Y

使用 **sp_iqindexinfo** 系统存储过程显示特定表或索引的对象放置和空间使用情况。以下信息源自 iqdemo 数据库。

```
sp_iqindexinfo 'table GROUPO.Customers';
```

Object	DBSpaceName	ObjSize	DBSpPct	MinBlk	MaxBlk
GROUPO.Customers	iq_main	200K	1	1,045,460	1,051,032
GROUPO.Customers. ASIQ_IDX_T733_C10_FP	iq_main	440K	1	1,046,689	1,047,147
GROUPO.Customers. ASIQ_IDX_T733_C1_FP	iq_main	440K	1	1,046,641	1,047,213
GROUPO.Customers. ASIQ_IDX_T733_C2_FP	iq_main	440K	1	1,046,961	1,047,203
GROUPO.Customers. ASIQ_IDX_T733_C3_FP	iq_main	440K	1	1,046,833	1,047,196
GROUPO.Customers. ASIQ_IDX_T733_C4_FP	iq_main	440K	1	1,046,737	1,047,189
GROUPO.Customers. ASIQ_IDX_T733_C5_FP	iq_main	440K	1	1,046,929	1,047,182

Object	DBSpaceName	ObjSize	DBSpPct	MinBlk	MaxBlk
GROUPO.Customers. ASIQ_IDX_T733_C6_FP	iq_main	440K	1	1,047,009	1,047,175
GROUPO.Customers. ASIQ_IDX_T733_C7_FP	iq_main	440K	1	1,046,945	1,047,168
GROUPO.Customers. ASIQ_IDX_T733_C8_FP	iq_main	440K	1	1,046,785	1,047,161
GROUPO.Customers. ASIQ_IDX_T733_C9_FP	iq_main	440K	1	1,046,881	1,047,154
GROUPO.Customers. ASIQ_IDX_T733_I11_HG	iq_main	152K	1	1,047,121	1,047,206
GROUPO.Customers. IX_customer_name	iq_main	304K	1	1,050,995	1,051,038

有关 **sp_iqdbspace**、**sp_iqdbspaceinfo** 和 **sp_iqindexinfo** 系统存储过程的完整语法，请参见《参考：构件块、表和过程》。

另请参见

- 数据存储 (第 149 页)
- 磁盘空间 (第 15 页)
- DDL 操作锁 (第 348 页)
- 临时表的版本控制 (第 346 页)
- 在 Interactive SQL 中与数据库断开所有连接 (第 147 页)

表管理

当创建数据库时，系统表是该数据库中的唯一表，用于保存数据库 schema。

您可能需要在数据库中创建、更改和删除表。可以使用 **dbisql** 执行文档中的示例，但 SQL 语句与所使用的管理工具无关。

您可以创建含有 **CREATE TABLE** 和 **ALTER TABLE** 语句的命令文件，这些语句用于在数据库中定义表并将其存储到源代码控制系统中。您可以根据需要使用这些命令文件重新创建数据库，还可以按标准方式创建表，并且您可以复制和修订这些表。

创建表的准则

使用 Sybase Central 或 Interactive SQL 创建表。

使用 Sybase Central 创建表

要使用 Sybase Central 创建表，请参见联机帮助。

使用 Interactive SQL 创建表

用于创建表的 SQL 语句为 **CREATE TABLE**。

本节说明如何使用 **CREATE TABLE** 语句。本节中的示例使用样本数据库。要尝试这些示例，请运行 **dbisql**，并使用用户 ID **DBA** 和口令 **sql** 连接到演示数据库。

除了 **Interactive SQL** 以外，您还可以使用其它工具创建表。此处介绍的 SQL 语句独立于您使用的工具。

示例

以下语句可创建一个新的永久 **IQ** 表来描述公司中雇员的资格。表中的列分别保存每种技能的标识号、名称和类型（例如 **technical** 或 **administrative**）。

```
CREATE TABLE skill (
skill_id INTEGER NOT NULL,
skill_name CHAR( 20 ) NOT NULL,
skill_type CHAR( 20 ) NOT NULL
)
```

在 **dbisql** 命令窗口中键入命令，并按执行键 (F9) 来执行上述命令。

- 每一列都具有数据类型。skill_id 为 integer（例如 101），skill_name 是宽度固定的 CHARACTER 字符串，最多包含 20 个字符，依此类推。
- 数据类型后的 NOT NULL 短语表示此示例中的所有列都必须包含值。
- 通常，您不会创建不含主键的表。

通过在创建表之前在内部执行 **COMMIT** 语句，Sybase IQ 使之前对数据库执行的所有更改都成为永久更改。还需要在创建表之后执行 **COMMIT**。

有关 **CREATE TABLE** 语句的完整说明，请参见《参考：语句和选项》。

警告! 更改或创建全局表或基表可能会对数据库的其他用户产生影响。对于较大的表，**ALTER** 或 **CREATE TABLE** 操作可能十分耗时。**CREATE TABLE** 处理将延迟其它 IQ 进程的执行，直至语句完成。尽管可以在其它连接处于活动状态时执行 **ALTER TABLE** 语句，但当任何其它连接使用要更改的表时，便无法执行这些语句。**ALTER TABLE** 处理在语句处理过程中会排除引用所提供表的其它请求。

指定数据类型

当创建表时，请指定每列保存的数据类型。

您还可以为数据库定义自定义数据类型。有关所支持数据类型的列表，请参见《参考：构件块、表和过程》。有关如何创建自定义数据类型的详细信息，请参见《参考：构件块、表和过程》中的 **CREATE DOMAIN** 语句。

另请参见

- 锁定的工作方式（第 347 页）
- IQ PAGE SIZE 参数准则（第 143 页）
- 不同的值数目（第 277 页）
- 有效数据的规则和检查（第 317 页）

表类型

Sybase IQ 可识别四种表类型。

- 基表
- 本地临时表
- 全局临时表
- 连接虚拟表

基表是永久表

在显式删除基表之前，这些表永久属于数据库的一部分，因此基表有时也称为主表、持久表或永久表。

在用户断开连接、服务器重新启动和恢复之后，这些表仍保留在数据库中。具有相应权限的所有用户均可以访问基表和基表中的数据。上一示例中显示的 **CREATE TABLE** 语句创建一个基表。

创建临时表

临时表有两种类型：全局和局部。

您可以使用 **CREATE TABLE** 的 **GLOBAL TEMPORARY** 选项 *创建* 全局临时表，也可以在 Sybase Central 中使用全局临时表创建向导进行创建。

当创建全局临时表时，在使用 **DROP TABLE** 语句显式删除之前，该表一直存在于数据库中。

数据库仅包含一个全局临时表定义，这与基表相同。但是，每个用户在全局临时表中均有一个单独的数据实例。这些行仅对插入它们的连接可见。当断开连接或已提交时，将删除这些行。给定连接继承用户首次引用全局临时表时所存在的表模式。在 Multiplex 服务器上创建全局临时表时还将在所有其它 Multiplex 服务器上创建此表。请参见《使用 Sybase IQ Multiplex》> “Multiplex 事务” > “DML 命令” > “表数据范围”。

要选择临时表，请使用如下语法：

```
SELECT * INTO #TableTemp FROM lineitem WHERE l_discount < 0.5
```

注意： Sybase 强烈建议在编写对 **SELECT INTO** 临时表执行的脚本时，在 **CAST** 表达式中对不是基列的选择列表项进行封装。这样可保证临时表的列数据类型是所需的数据类型。

使用 **DECLARE LOCAL TEMPORARY TABLE** 语句可以仅针对您的连接 *声明* 本地临时表。本地临时表在断开连接或提交前一直存在，或存在于声明此表的复合语句中。其他用户完全无法访问此表及其数据。

如果该连接中存在某一本地临时表，尝试创建同名的基表或全局临时表将失败，这是因为 *owner.table* 无法唯一地标识新表。

但是，可创建同名的本地临时表，作为现有基表或全局临时表。由于将首先解析本地临时表，因此对表名的引用将访问本地临时表。

例如，考虑以下序列：

```
CREATE TABLE t1 (c1 INT); INSERT t1 VALUES (9); DECLARE LOCAL
TEMPORARY TABLE t1 (c1 INT); INSERT t1 VALUES (8); SELECT * FROM t1;
```

返回的结果为 8。对 *t1* 的任何引用都将引用本地临时表 *t1*，直到连接已删除本地临时表为止。

删除和更改全局临时表

全局临时表与基表的删除方式相同，即使用 **DROP TABLE** 语句或 Sybase Central 删除。

当其它连接正在使用全局临时表时，您不能删除或更改此表。

另请参见

- 使用连接索引（第 212 页）

表位置

Sybase IQ 在当前数据库中创建表。

如果连接到 IQ 数据库，则按如下方式放置表：

表 10. 表位置

表类型	允许位置	缺省位置
永久表	IQ 存储、Catalog 存储	IQ 存储
全局临时表	IQ 临时存储、Catalog 存储	IQ 临时存储
局部临时表	IQ 临时存储或 Catalog 存储；仅对创建此表的用户可见	IQ 临时存储

连接虚拟表

连接虚拟表是类似于常规表的非规范化表；它具有名称、列、行和索引。

执行 **CREATE JOIN INDEX** 以后，Sybase IQ 将创建连接虚拟表，以用于内部处理目的；执行 **DROP JOIN INDEX** 以后，Sybase IQ 将删除连接虚拟表。不能创建、修改或删除

连接虚拟表，但是，当尝试使用或修改这些表时，将显示与其相关的错误消息。Sybase 建议您忽略所有连接虚拟表。

在 Multiplex 中运行的服务器不能创建或删除连接索引。有关详细信息，请参见《使用 Sybase IQ Multiplex》。

为 IQ 表自动创建索引

通过创建具有 **PRIMARY KEY** 或 **UNIQUE** 约束的表，您可以为特定列自动创建索引。

这些选项使 Sybase IQ 能够为强制执行唯一性的列创建 **HG** 索引。

如果使用 **ALTER TABLE** 命令向现有表添加 **UNIQUE** 列，或将现有列指定为 **UNIQUE**，则会自动创建 **HG** 索引。

另请参见

- 索引概述 (第 187 页)
- 选择索引的标准 (第 191 页)

优化存储和查询性能

当在 IQ 数据库中创建永久表时，Sybase IQ 会将此表自动存储到有利于投影查询类型的缺省索引中。

Sybase IQ 根据下列因素优化此结构，以便提高查询性能并降低存储要求：

- **IQ UNIQUE** 选项 (**CREATE TABLE** 或插件的“列属性”页)
- **MINIMIZE_STORAGE** 选项 (**SET OPTION** 或插件的“数据库选项”对话框)
- 列的数据类型及其宽度
- **IQ PAGE SIZE** 选项 (**CREATE DATABASE** 或插件的创建数据库向导)

另请参见

- 显示 Sybase Central 中的系统对象 (第 147 页)

IQ UNIQUE 的影响

有关 **CREATE TABLE** 的选项 **IQ UNIQUE** 的影响，请参见下表。

表 11. IQ UNIQUE 的影响

IQ UNIQUE 小于或等于 256	IQ UNIQUE 小于或等于 65536	IQ UNIQUE 小于或等于 16777216	未指定 IQ UNIQUE 或大于 16777216
为较少数量的唯一值优化存储	为中等数量的唯一值优化存储	为包含 3 字节的 FP 索引优化存储	为较多数量的唯一值优化存储

IQ UNIQUE 小于或等于 256	IQ UNIQUE 小于或等于 65536	IQ UNIQUE 小于或等于 16777216	未指定 IQ UNIQUE 或大于 16777216
查询性能更快, 所需的 main IQ 存储空间更小	查询性能更快, 所需的 main IQ 存储空间更小	查询性能更快, 所需的 main IQ 存储空间更小	查询可能更慢
需要少量额外高速缓存, 以便 IQ 临时存储进行装载	需要额外高速缓存, 以便 IQ 临时存储进行装载。高速缓存大小取决于唯一值数量和数据类型。	需要大量额外高速缓存, 以便 IQ 临时存储进行装载。高速缓存大小取决于唯一值数量和数据类型。	不需要额外高速缓存即可进行装载
如果具有大量 IQ UNIQUE <256 的列, 装载可能更慢	装载可能更慢	装载可能更慢	装载更快

MINIMIZE_STORAGE 选项的影响

如果 MINIMIZE_STORAGE 为 ON, 则它等效于为所有新列指定 **IQ UNIQUE 255**。

MINIMIZE_STORAGE 缺省为 OFF。有关详细信息, 请参见《参考: 语句和选项》> “数据库选项” > “常规数据库选项” > “MINIMIZE_STORAGE 选项”。

索引和 IQ UNIQUE

如果估计的 **IQ UNIQUE** 不正确, 不会对装载产生任何影响; 优化程序仅使用下一个较大的索引。

对于查询, 如果估计的 **IQ UNIQUE** 不正确, 并且具有 **HG**、**LF** 或已优化存储的缺省索引, 优化程序将忽略 **IQ UNIQUE** 值, 并使用该索引中的实际值数量。如果没有上述任何一种索引, 并且估计差距非常大 (例如, 如果您指定 **IQ UNIQUE 1000000**, 但实际唯一值数量为 1200 万), 则可能会对查询性能产生影响。

要为现有索引更改 **IQ UNIQUE** 值, 请运行 **sp_iqrebuildindex** 过程。有关详细信息, 请参见《参考: 构件块、表和过程》> “系统过程”。

UNIQUE 和 IQ UNIQUE 之间的区别

IQ UNIQUE (count) 提供给定列中可能包含的不同的值数量的近似值。每个不同的值可能会出现多次。例如, 在 **employee** 表中, 可以在 **state** 列中显示一组有限的不同的值, 但是其中的每个值都可以在许多行中出现。

相反, 当指定 **UNIQUE** 或 **PRIMARY KEY** 时, 每个值只能在该列中出现一次。例如, 在 **employee** 表中, **ss_number** 的每个值 (即雇员的社会保险号) 只能在该列中出现一次。此唯一性扩展至空值。因此, 指定为 **UNIQUE** 的列还必须具有 **NOT NULL** 约束。

更改表的准则

本节介绍如何使用 **ALTER TABLE** 语句更改表的结构。

示例 1

以下命令将一列添加到 **skill** 表中，以便用于输入技能的可选说明：

```
ALTER TABLE skill
ADD skill_description CHAR( 254 )
```

示例 2

以下语句将 **skill_type** 列的名称更改为 **classification**：

```
ALTER TABLE skill
RENAME skill_type TO classification
```

示例 3

以下语句删除 **classification** 列。

```
ALTER TABLE skill
DELETE classification
```

示例 4

以下语句更改整个表的名称：

```
ALTER TABLE skill
RENAME qualification
```

这些示例显示了如何更改数据库的结构。**ALTER TABLE** 语句可更改表的诸多特性，例如，可以添加或删除外键等。但是，不能使用 **MODIFY** 更改表或列约束，而必须 **DELETE** 旧约束，然后再 **ADD** 新约束。在所有这些情况下，只要进行了更改，则存储过程、视图和引用该列的任何其它项都将无法工作。

有关完整说明，请参见《参考：语句和选项》>“SQL 语句”>“ALTER TABLE 语句”。

在 Sybase Central 中更改表

要在 Sybase Central 中更改表定义，请参见 Sybase Central 联机帮助。

更改连接索引中的表

不能对参与连接索引的连接条件的基表列执行 **ADD**、**DROP** 或 **MODIFY**。要更改连接列，必须先删除该连接索引，更改表，然后再重新创建连接索引。有关连接索引的完整信息，请参见“使用连接索引”。

另请参见

- 表约束和列约束（第 325 页）

删除表的准则

以下 **DROP TABLE** 语句会删除 `skill` 表中的所有记录，然后从数据库中删除 `skill` 表的定义

```
DROP TABLE skill
```

与 **CREATE** 语句类似，**DROP** 语句在删除表前后会自动执行 **COMMIT**。这会使上次执行 **COMMIT** 或 **ROLLBACK** 后对数据库的所有更改成为永久更改。

DROP 语句还删除对表的所有索引，但参与连接索引的表中的所有列除外。

如果只希望删除数据行，而不是表自身，请使用 **TRUNCATE TABLE** 语句。如果您在其他用户读取时截断某一个表，则应用表版本控制的常规规则，也就是说，将在读取者的事务完成之前保留旧的表版本；有关详细信息，请参见第 10 章“事务和版本控制”。

执行 **DROP TABLE** 和 **TRUNCATE TABLE** 语句只需几秒钟时间。数据大小不会影响操作速度。

有关完整说明，请参见《参考：语句和选项》>“SQL 语句”>“DROP 语句”。

在 Sybase Central 中删除表

您可以在 Sybase Central 或 Interactive SQL 中执行 **DROP TABLE** 语句。

1. 连接到数据库。
2. 单击该数据库的“Tables”文件夹。
3. 右键单击要删除的表，然后从弹出菜单中选择“删除”。

创建主键

表中的每一行由它的主键唯一地标识。

使用 **CREATE TABLE** 和 **ALTER TABLE** 语句可以设置表的诸多属性，包括列约束和检查约束。

创建主键

以下语句创建与以前相同的 `skill` 表，唯一不同之处是它添加了一个主键：

```
CREATE TABLE skill (
    skill_id INTEGER NOT NULL,
    skill_name CHAR( 20 ) NOT NULL,
    skill_type CHAR( 20 ) NOT NULL,
    primary key( skill_id )
)
```

该表中每一行的主键值必须是唯一的，这就意味着您不能给多行指定同一个 `skill_id`。

不允许主键中的列包含 **NULL**。您必须在主键中的列上指定 **NOT NULL**。

另请参见

- 从 Sybase Central 连接到演示数据库 (第 49 页)
- 从 Interactive SQL 连接到演示数据库 (第 49 页)
- 创建外键 (第 178 页)

创建外键

每一外键关系将一系列中的候选键 (主键和唯一约束) 与另一个表中的某一列关联起来 (后者则成为外键)。

例如, 您可以创建名为 `emp_skill` 的表, 该表保存每个雇员的每项合格技能的技能等级说明, 如下所示:

```
CREATE TABLE emp_skill(  
emp_id INTEGER NOT NULL,  
skill_id INTEGER NOT NULL,  
"skill level" INTEGER NOT NULL,  
PRIMARY KEY( emp_id, skill_id ),  
FOREIGN KEY REFERENCES employee,  
FOREIGN KEY REFERENCES skill  
)
```

`emp_skill` 表定义具有由以下两列组成的一个主键: `emp_id` 列和 `skill_id` 列。一个雇员可能有多项技能, 因此会显示在多行中, 同时, 多个雇员也可能都具有某一给定技能, 因此 `skill_id` 也可能出现多次。

`emp_skill` 表还具有两个外键。外键条目指示 `emp_id` 列必须包含 `employee` 表中的有效雇员号 (在 `employee` 中用作主键), `skill_id` 必须包含 `skill` 表中的有效条目 (在 `skill` 表中用作主键)。

一个表只能定义一个主键, 但可以按需要具有多个外键。

不能在任何类型 (局部、全局或自动) 的临时表上创建外键约束。

有关有效字符串和标识符的详细信息, 请参见《参考: 构件块、表和过程》> “SQL 语言元素”。

有关使用主键和外键的详细信息, 请参见第 9 章 “确保数据完整性”。

另请参见

- 其它数据库空间 (第 162 页)
- 创建主键 (第 177 页)
- 实体和参照完整性 (第 329 页)
- 星形连接 (第 222 页)

系统视图中的表信息

有关数据库中表的信息保存在系统视图 `SYS.SYSTAB` 中。

可以使用 **Interactive SQL** 浏览此视图中的信息。在 `dbisql` 命令窗口中键入以下语句，即可查看到 `SYS.SYSTAB` 视图中的所有列：

```
SELECT *  
FROM SYS.SYSTAB
```

视图管理

视图是计算出的表。

您可以使用视图以一种可以控制的格式向数据库用户显示所要提供的信息。

视图和基表之间的相似之处

视图和数据库的永久表（永久表也称为基表）在多个方面具有相似之处：

- 您可以为视图分配访问权限，就像为基表分配访问权限一样。
- 您可以对视图执行 **SELECT** 查询。
- 您可以对某些视图执行 **INSERT** 和 **DELETE** 操作。
- 您可以基于某些视图创建另一些视图。

视图和永久表之间的差异

视图和永久表之间存在一些差异：

- 不能对视图创建索引。
- 只能对特定视图执行 **INSERT**、**DELETE** 和 **UPDATE** 操作。
- 您不能将完整性约束和键指派给视图。
- 视图引用基表中的信息，但不保留该信息的副本。您每次调用视图时都重新计算视图。

定制数据访问的好处

使用视图可以定制对数据库中数据的访问。定制访问可以有以下几个好处：

- 因为不允许访问不相关的信息。
- 因为可以以一种比基表中的数据更易于理解的格式向用户和应用程序开发人员提供数据。
- 因为可以在数据库中集中一般查询的定义。

另请参见

- 视图和过程提供了更高的安全性（第 308 页）

创建视图

SELECT 语句对一个或多个表进行操作并产生一个结果集，该结果集也是一个表。

与基表类似，**SELECT** 查询的结果集具有列和行。视图为特定的查询指定一个名称，并在数据库系统表中保存定义。

示例

假设您需要经常列出每个部门的雇员数。您可以使用以下语句得到该列表：

```
SELECT DepartmentID, COUNT(*)
FROM Employees
GROUP BY DepartmentID
```

可以创建包含此语句的结果的视图，如下所示：

```
CREATE VIEW DepartmentSize AS
SELECT DepartmentID, COUNT(*)
FROM Employees
GROUP BY DepartmentID
```

视图中的信息不单独存储于数据库中。每次引用视图时，将执行关联的 **SELECT** 语句，以检索合适的数据库。

另一方面，之所以应当这样，是因为它意味着如果某人修改 `Employees` 表，则 `DepartmentSize` 视图中的信息将自动更新。另一方面，当您每次使用视图时，复杂的 **SELECT** 语句可能增加 SQL 查找正确信息所需的时间量。

要在 Sybase Central 中创建视图，请参见联机帮助。

使用视图的准则

对于用于创建视图的 **SELECT** 语句以及能否在视图中插入数据、从视图中删除数据或更新视图，都存在特定的限制。

对于 **SELECT** 语句的限制

您不能在 **SELECT** 查询中使用 **ORDER BY** 子句。关系表的一个特性是行或列的顺序没有特殊意义，而使用 **ORDER BY** 子句会对视图的行强制排序。可以在视图定义中使用 **GROUP BY** 子句、子查询和连接。

只有顶层 **SELECT** 列表（非视图、派生表或子查询）中支持标量值子查询。有时，顶层 **SELECT** 的 **FROM** 子句中使用的视图或派生表非常简单，可以向上展平到顶层 **SELECT** 中。因此，前述规则实际上只对子查询、未展平的视图和未展平的派生表强制执行。例如：

```
SELECT * FROM test_view
Msg 21, Level 14, State 0:
SQL Anywhere Error -1005004: Subqueries are allowed only as arguments
of
comparisons, IN, and EXISTS,
-- (opt_Select.cxx 2101)
```

要规划一个视图，请调优 **SELECT** 查询本身，直到其以所需格式提供完全符合需要的结果。一旦有了正确的 **SELECT** 查询，就可以在查询的前面添加短语，以创建视图。例如：

```
CREATE VIEW viewname AS
```

插入和删除视图的准则

有些视图允许使用 **UPDATE**、**INSERT** 和 **DELETE** 语句，但有些视图却不允许使用这些语句，具体取决于其关联的 **SELECT** 语句。

对于包含以下内容的视图，您无法更新这些视图、在这些视图中插入数据或从这些视图中删除数据：

- 集合函数，例如 **COUNT(*)**
- **SELECT** 语句中的 **GROUP BY** 子句
- **UNION** 操作

在所有这些情况下，无法将 **UPDATE**、**INSERT** 或 **DELETE** 转换为对基础表的操作。

警告！ 请勿删除由拥有系统对象的 **dbo** 用户 ID 拥有的视图。删除此类视图或将它们转化为表可能导致意外的问题。

另请参见

- 视图提供了定制的安全性（第 308 页）

修改视图的准则

可以使用 **ALTER VIEW** 语句修改视图。

ALTER VIEW 语句将视图定义替换为新的定义；它不对现有的视图定义进行修改。

ALTER VIEW 语句会保留视图的权限。

示例

例如，若要在上述 **DepartmentSize** 视图中将列名替换为具有更多信息的名称，可以使用以下语句：

```
ALTER VIEW DepartmentSize
  (DepartmentID, NumEmployees)
AS
  SELECT DepartmentID, COUNT(*)
  FROM Employees
  GROUP BY DepartmentID
```

另请参见

- 对修改连接索引表的限制（第 224 页）

针对视图的权限

如果具备以下条件中的一个和多个，用户可以通过视图执行操作：

- **DBA** 已授予该用户对视图执行相应操作的权限。
- 该用户拥有对所有对象（例如基表、视图、过程）执行相应操作的权限。
- 一个非 **DBA** 用户给该用户授予了对视图进行相应操作的权限。该非 **DBA** 用户必须是视图的所有者，或者必须具有对视图带 **WITH GRANT OPTION** 的相应权限。视图的所有者是：
 - **DBA**，或
 - 非 **DBA**，而且是该视图引用的所有对象的所有者，或
 - 非 **DBA**，并且不是该视图引用的某些对象或所有对象的所有者，但该视图所有者针对非其拥有的对象具备带 **WITH GRANT OPTION** 的 **SELECT** 权限，以及操作非其拥有的对象的带 **WITH GRANT OPTION** 的任何其它所需权限。权限可能已授予 **PUBLIC**，而不是针对对象具备带 **WITH GRANT OPTION** 的权限的所有者。其中包括针对系统表的 **SELECT** 权限。

仅能针对整个视图授予 **UPDATE** 权限。与表不同的是，不能针对视图中的单个列授予 **UPDATE** 权限。

如何删除视图

若要用 **Interactive SQL** 从数据库中删除视图，请使用 **DROP** 语句。下面的语句可删除 **DepartmentSize** 视图：

```
DROP VIEW DepartmentSize
```

如何删除 Sybase Central 中的视图

要删除 **Sybase Central** 中的视图，请右键单击希望删除的视图，然后从弹出菜单中选择“删除”。

有关详细信息，请参见 **Sybase Central** 联机帮助。

系统视图中的视图信息

有关数据库中视图的信息保存在系统视图 **SYS.SYSVIEW** 中。

请参见《参考：构件块、表和过程》>“系统表和视图”。

可以使用 **Interactive SQL** 浏览此视图中的信息。在 **dbisql** 命令窗口中键入以下语句，即可查看到 **SYS.SYSVIEW** 视图中的所有列：

```
SELECT *  
FROM SYS.SYSVIEW
```

要提取包含特定视图定义的文本文件，请使用如下语句：

```
SELECT view_def FROM SYS.SYSVIEW  
WHERE view_object_id = 1583;
```

```
OUTPUT TO viewtext.sql
FORMAT ASCII
```

索引管理

与传统关系数据库中的搜索相比，Sybase IQ 索引大大改进了数据库搜索的性能。

在设计和创建数据库时，性能是一个需要重点考虑的因素。

但是，甚至在 Sybase IQ 中，为了取得最佳的性能并更好地利用内存、磁盘和 CPU 周期，为数据选择正确的索引也是很重要的。

索引简介

所有包含数据的 IQ 数据库列都需要索引。

在 IQ 存储中创建数据库时，系统将在每个表的每个列上自动创建缺省索引。还可以选择其它几种索引类型：

- 四种列索引类型可以优化针对索引列的特定类型查询。
- 连接索引可以优化与两个或更多表的列相关的查询。

几乎可以肯定的是，您也需要通过为数据库中的很多列选择一个或多个索引来补充缺省索引。您还需要在用户查询中为以相同的方式进行连接的任何表列定义连接索引。应当基于数据库的大小、可用的磁盘空间和用户提交的查询类型来选择索引。

索引是针对于单个特定表创建的，对于一组表，则将创建连接索引。不能创建视图索引。

创建索引

创建列索引和连接索引。

可以用三种方式创建列索引：

- 使用 **CREATE INDEX** 语句
- 使用 Sybase Central 中的索引创建向导
- 使用 **CREATE TABLE** 的 **UNIQUE** 或 **PRIMARY KEY** 列约束，这将自动创建唯一索引。

可以用两种方式创建连接索引：

- 使用 **CREATE JOIN INDEX** 语句
- 使用 Sybase Central 中的创建连接索引向导

系统视图中的索引信息

索引的信息包含于系统视图 **SYSINDEX**、**SYSIQINDEX**、**SYSIXCOL** 和 **SYSIQJOINIXTABLE**（对于连接索引）中。

有关这些视图的说明，请参见《参考：构件块、表和过程》>“系统表和视图”。

使用存储过程显示索引

还可以使用存储过程 `sp_iqindex` 显示索引的列表及其相关信息。

例如，要列出 `Departments` 表中的索引，请发出命令：

```
sp_iqindex 'Departments'
```

将显示以下信息。数据将以两部分显示，以便于阅读：

table_name	table_owner	column_name	index_type
Departments	GROUPO	DepartmentHeadID	FP
Departments	GROUPO	DepartmentHeadID	HG
Departments	GROUPO	DepartmentID	FP
Departments	GROUPO	DepartmentID	HG
Departments	GROUPO	DepartmentName	FP

如果在命令中省略表名，则 `sp_iqindex` 将显示数据库中所有表的此类信息。

index_name	unique_index	dbspace_id	remarks
ASIQ_IDX_T740_C3_FP	N	16,387	(NULL)
ASIQ_IDX_T740_C3_HG	N	16,387	(NULL)
ASIQ_IDX_T740_C1_FP	U	16,387	(NULL)
ASIQ_IDX_T740_I4_HG	Y	16,387	(NULL)
ASIQ_IDX_T740_C2_FP	N	16,387	(NULL)

索引验证

可以对 `Catalog` 存储中 `SQL Anywhere` 表的索引进行验证，以确保在索引中引用的每个行都在表中确实有效。

对于外键索引，验证检查还可以确保对应的行存在于主表中，并且它们的散列值匹配。此检查等同于由 `SQL Anywhere VALIDATE TABLE` 语句执行的有效性检查。

若要验证索引，请打开命令提示符，并运行 `dbvalid` 实用程序。

例如，下面的语句验证名为 `EmployeeIndex` 的索引。 `-i` 开关指定每个给定的对象名都是索引。（请在一行中键入该命令。）

```
dbvalid -c "uid=dba;pwd=sql;eng=myserver" -i EmployeeIndex
```

请参见《实用程序指南》>“`dbvalid` 数据库管理实用程序”。

如何重命名索引

可以对所有者类型为 **USER** 的基表或全局临时表中的索引进行重命名。

有关重命名索引和更改外键角色名称的详细信息，请参见《参考：语句和选项》> “SQL 语句” > “ALTER INDEX 语句”。注意，不能重命名为强制键约束而创建的索引。

如何删除索引

如果不再需要列索引或连接索引，则可以使用 **DROP** 语句从数据库中将其删除。

您还可以在 **Sybase Central** 中删除索引，方法是：单击表名，再右键单击以显示选项，然后单击“删除”选项。

另请参见

- 对修改连接索引表的限制（第 224 页）

Sybase IQ 索引

在为 Sybase IQ 数据创建索引之前，您必须了解 Sybase IQ 索引类型，如何创建索引，如何确定哪些索引类型最适合您对数据库中的数据的使用方式，以及与建立索引相关的性能和资源问题。

索引概述

索引可以提高数据检索性能。Sybase IQ 索引与传统索引不同，它们旨在加快数据仓库应用环境中的查询速度。

传统索引通常使用 B 树索引策略指向数据记录。仅当使用大量唯一数据值并将这些数据值筛选为诸如多列订单号或客户名的很小记录集合时，此策略才有用，此过程与事务处理系统相仿。

Sybase IQ 索引实际上表示和存储数据，以便使用此数据加速多种查询。此策略是为数据仓库环境设计的，在该环境中，查询通常检查往往含有相对较少的唯一值的大量记录，并且该环境通常需要集合结果。

Sybase IQ 索引类型

Sybase IQ 为所有列创建用于优化投影的 *缺省索引*。

根据数据库大小、可用磁盘空间和用户提交的查询类型，您几乎毫无疑问地需要为该缺省索引补充一个或多个 Sybase IQ 逐位索引类型。

在表中装载数据时，Sybase IQ 实际上按列而不是按行为表中每一行存储数据。相较于基于行的传统索引而言，列方向为 IQ 索引提供了若干重要优势。

从逻辑上讲，仍可以逐行访问数据，正如在较传统的基于行的 SQL 数据库一样。列存储根据您希望跟踪的属性为数据创建结构。在数据仓库环境中，您通常需要查看成千上万行数据的特定属性，而不是仅仅查看完整的单行数据，后者是事务处理的传统重点。列存储优化了对您关注的属性执行选择或计算的功能。

Sybase IQ 为各列创建的缺省列存储结构实际上是一个针对存储和投影数据进行优化的索引。定义的列索引作为各表的一部分创建。创建连接索引时请小心；它们会产生大量的装载、更新和删除成本。

除了列索引之外，Sybase IQ 还允许您定义连接索引。连接索引是为连接相关表而优化的。您可能需要为用户因解析查询而经常连接的任何表集合创建连接索引。涉及这些列的任何连接索引都以列索引为基础。

包含的唯一值数目少于 1600 万的列可以存储在已经过优化的缺省索引中，这可以极大地降低存储要求。此格式支持已经过 IQ 优化程序改进的性能，并支持集合函数 **SUM**、**SUM DISTINCT**、**MAX**、**MIN** 和 **COUNT DISTINCT**。它适用于下列各项：

- 指定了 **IQ UNIQUE()** 的任何列
- 当 **MINIMIZE_STORAGE** 数据库选项为 **ON** 时创建的所有列

但是，为了最大程度地提高查询性能，还应为大多数列选择一种或多种其它索引类型（如下表所示），以便以最佳方式显示列数据的基数和使用情况：

表 12. Sybase IQ 索引类型

索引类型	描述
比较或 CMP	存储具有相同数据类型、精度和小数位数的任何两个不同的列的二进制比较 (<、> 或 =)。
DATE	数据类型为 DATE 的列的索引，用于处理涉及日期数量的查询。
日期时间或 DTTM	数据类型为 DATETIME 或 TIMESTAMP 的列的索引，用于处理涉及日期时间数量的查询。
High_Group 或 HG	增强型 B 树索引，用于对高基数数据处理相等和 group by 运算（建议用于具有超过 1,000 个不同的值的列）。
High_Non_Group 或 HNG	非基于值的位图索引，适合大部分与范围或集合有关的高基数决策支持运算。
Low_Fast 或 LF	基于值的位图索引，用于处理对低基数数据的查询（建议用于最多具有 1,000 个不同的值但可最多支持 10,000 个不同的值的列）。
TIME	数据类型为 TIME 的列的索引，用于处理涉及时间数量的查询。
WD	用于通过将 CHAR 、 VARCHAR 或 LONG VARCHAR 列的内容视为分隔列表为关键字创建索引。
TEXT	用于对词语（单词）及其位置编制索引。允许搜索单个词语、词组、指定距离内和给定顺序的词语对，以及这些条件的组合。

选择列索引时应基于该列中的数据类型和列数据的预期操作。通常，您可以对任何列使用任何索引或索引组合。但是，也存在一些例外情况。

当使用 **DATE** 数据类型创建表时，将对 **DATE** 字段创建一个优化的双字节 **FP** 索引，该索引独立于数据库选项 **MINIMIZE_STORAGE** 中的设置。

如果要对 **DATE** 字段创建三字节 **FP** 或平面样式的 **FP** 索引，请在创建表时使用下列 **IQ UNIQUE** 值：

- 对于三字节 **FP - IQ UNIQUE** 应介于 65537 和 16777216 之间。
- 对于平面样式的 **FP - IQ UNIQUE** 应大于 16777216。

为了对具有非整数数值数据的列利用 **High_Non_Group** 索引类型，请使用 **NUMERIC** 或 **DECIMAL** 数据类型，这些数据类型支持多达 254 位的精度。

某些索引类型具有重复功能；创建不必要的索引会浪费磁盘空间。有关如何选择索引的详细信息，请阅读下面各节。

将一列指定为 **FOREIGN KEY**、**PRIMARY KEY** 或 **UNIQUE** 时，Sybase IQ 会自动为其创建一个 **High_Group** 索引。对于每一外键，Sybase IQ 创建非唯一的 **High_Group** 索引。

注意：您还可以对一组列显式创建 **High_Group** 索引。有关详细信息，请参见《参考：语句和选项》中的 **CREATE INDEX** 语句。

Sybase IQ 如何使用索引

您可能还希望对列定义其它索引以获得最佳性能。Sybase IQ 使用适用于当前查询或连接谓词的最快索引。如果为列创建的索引类型不正确，Sybase IQ 仍可以解析涉及该列的查询，但响应时间可能会慢于使用正确索引类型所需的时间。

如果对特定列定义了多个索引，Sybase IQ 将基于相同输入数据为该列生成所有索引。

优化程序的索引指导

如果对数据库设置了 **INDEX_ADVISOR** 选项，Sybase IQ 将在消息日志或查询计划中发出消息，并给出可能提高性能的其他索引的建议。这些消息侧重于下列几个方面：

- 本地谓词列
- 单列连接键列
- 相关子查询列
- 分组列

如果您决定遵循建议，请亲自创建索引。

添加和删除索引

如果您在稍后发现需要其它索引，您可以方便地添加索引；但是，在插入任何数据之前创建所有需要的索引，速度往往会更快一些。

您只能使用所有者类型 **USER** 在基表或全局临时表中重命名或更改索引。有关重命名索引和更改外键角色名称的详细信息，请参见《参考：语句和选项》中的 **ALTER INDEX** 语句。

您可以删除确定不需要使用的任何可选索引。有关删除索引的详细信息，请参见《参考：语句和选项》的 **DROP** 语句中的 **DROP INDEX** 命令。

注意：您可能需要删除外键约束，但应保留基础 **HG** 索引。非唯一的 **HG** 索引可提高查询性能，但生成此索引会增大开销。

注意，**ALTER TABLE DROP FOREIGN KEY CONSTRAINT** 不会删除自动创建的非唯一 **HG** 索引。不能在保留关联外键的情况下删除主键。要删除此类索引，请发出 **ALTER TABLE DROP FOREIGN KEY** 命令，然后显式删除该索引。

相较于传统索引的优点

Sybase IQ 索引提供以下传统索引技术无法提供的优点：

- 索引大小通常非常小。可以对整个数据库完全建立索引，并且即席查询可在存储原始数据所需的相同空间中使用这些索引。传统数据库通常需要三倍或更大的空间。
- 通过仅对相关列的索引进行有效组合和操作来解析查询。这样可避免十分耗时的表扫描。
- 最大程度地降低 I/O，并消除潜在瓶颈情况。
- 由于索引非常紧凑，因此可以在内存中为后续查询存储更多数据，从而加速迭代分析上的吞吐量。
- 调优依赖于数据，允许为任何多次的即席查询一次性优化数据。

创建 Sybase IQ 索引的工具

可以使用两种工具之一显式创建列索引。

- **Interactive SQL**
要创建 Sybase IQ 列索引，请使用《参考：语句和选项》> “SQL 语句” > “CREATE INDEX 语句” 中的 SQL 语法。
要创建 Sybase IQ 列索引，请使用《参考：语句和选项》> “SQL 语句” > “CREATE INDEX 语句” 中的 SQL 语法。
- **Sybase Central**
要使用 Sybase Central 创建列索引，请参见《Sybase IQ 简介》> “创建索引和装载数据” 中的“创建列索引”。

Interactive SQL 索引创建

如果未指定 *index-type*，Sybase IQ 将创建 **HG** 索引。

为此，若干前端工具将自动创建 **HG** 索引。

索引创建示例

以下两个示例说明了索引创建功能。

第一个示例对 SalesOrderItems 表的 ShipDate 列创建名为 ShipIx 的 High_Non_Group (**HNG**) 索引。

```
CREATE HNG INDEX ShipIx ON dbo.SalesOrderItems (ShipDate)
```

第二个示例对 SalesOrder 表的 Region 列创建名为 SalesOrderRegionIX 的 Low_Fast (**LF**) 索引。

```
CREATE LF INDEX SalesOrderRegionIx ON dbo.SalesOrder (Region)
```

另请参见

- 比较 (CMP) 索引类型 (第 204 页)

索引装载的状态消息

缺省情况下，在索引中插入和装载每 100,000 条记录后，您将收到一则进度消息。

要更改记录数目，请指定 **CREATE INDEX** 的 **NOTIFY** 选项或 **NOTIFY_MODULUS** 选项。要防止出现这些消息，请指定 **NOTIFY 0**。

如何执行 CREATE INDEX 语句组合

使用关键字 **BEGIN PARALLEL IQ** 和 **END PARALLEL IQ** 来分隔您希望作为一组同时执行的任意数目的 **CREATE INDEX** 语句。

仅当对 **IQ** 表（而不是临时表或 SQL Anywhere 表）创建索引时，才能使用这些关键字。如果其中一个 **CREATE INDEX** 语句失败，则回退所有语句。请参见《参考：语句和选项》。

并行列索引创建

在某些情况下，您可以同时创建多个列索引。

- 每个 **CREATE INDEX** 语句只能创建一个索引。
- 如果两个连接对同一个表发出 **CREATE INDEX** 语句，则运行第一个语句；另一个语句将收到错误，通知只允许一个写入程序。
- 如果两个连接对不同表发出 **CREATE INDEX** 语句，则并行执行两个语句。
- 如果两个连接对参与同一连接索引的不同表发出 **CREATE INDEX** 语句，则只运行其中一个 **CREATE INDEX**；另一个语句将收到错误，通知只允许一个写入程序。

选择索引的标准

为任何给定列定义的索引集合可以对查询处理的速度产生巨大影响。

主要有四种选择索引的标准：

- 唯一值数目
- 查询类型
- 磁盘空间使用情况
- 数据类型

请配合使用所有标准所对应的建议，而不要将其分隔开来。同时，请记住，所有列将按照有助于快速投影的方式自动存储。要决定其它索引，请仔细查看各列中的数据。尝试预计唯一值数目和值总数、由这些数据生成的用户所需查询结果，以及数据将用于即席连接还是连接索引。

有关索引类型的详细信息以及选择正确类型所使用的标准，请参见后面各节。

索引中的唯一值数目

Sybase IQ 索引根据其包含的唯一（不同的）值数目进行优化。

当不同的值的数目达到特定级别时，请按照下表中的建议来选择索引。

表 13. 考虑顺序

唯一值数目	建议的索引类型
低于 1,000	LF (HG [如果表的行数 <25,000])
1000 或更大	HG 和/或 HNG

当 `MINIMIZE_STORAGE` 选项为 `ON` 时创建的列，或为其指定 `IQ UNIQUE 65536` 或更小的列将自动放置到缺省索引格式中，该索引已经过优化，可以减少存储，并提高特定查询类型的性能。

下面列出了一些具有不同唯一值数目的列的示例：

- 用于保存婚姻状况的列仅有几个唯一值（单身、已婚或空值）
- 用于保存州名或省名的列所包含的唯一值数目小于 100
- 用于保存日期数据的列所包含的唯一值数目可能大于 100 但小于 65536
- 用于保存帐户或社会保险号的列可能包含成千上万个唯一值

查询类型

Sybase 建议对特定类型的查询采用特定的索引类型。

您应预先了解通常将如何查询列中的数据。例如：

- 该列是否包含在连接谓词中？
- 如果列具有大量唯一值，该列是否将在 `GROUP BY` 子句中使用，是否用作 `COUNT DISTINCT` 的参数以及/或者是否用于 `SELECT DISTINCT` 投影？
- 该列是否将经常与其它具有相同数据类型、精度和标度的列进行比较？

通常，列中的数据类型的列的使用方式给出了很好的指示。例如，日期列将可能在 `WHERE` 子句中用于范围搜索，包含价格或销售额的列可能会作为集合函数（`SUM`、`AVG` 等）的参数在投影中使用。

注意： Sybase IQ 仍可以解析索引类型不正确的列所涉及的查询，但不能以同等有效方式执行。

下表显示了基于查询的建议索引类型。此表首先列出对每种查询而言通常最快的索引，最慢的索引列在最后。选择索引类型时不应以这些建议作为唯一标准，还应考虑唯一值数目和磁盘空间。请参见本节中的其它表。

表 14. 查询类型/索引

查询用法类型	建议的索引类型
用于 SELECT 投影列表	缺省值
用于 SUM(A+B) 等计算表达式	缺省值
作为 AVG/SUM 参数	HNG 、 LF 、 HG 、缺省值
作为 MIN/MAX 参数	LF , HG , HNG
作为 COUNT 参数	缺省值
作为 COUNT DISTINCT , SELECT DISTINCT 或 GROUP BY 参数	LF 、 HG 、缺省值
用作分析函数参数	LF 、缺省值
如果字段不允许重复项	HG
在即席连接条件中使用列	缺省值、 HG 、 LF
在连接索引中使用列	HG , LF
用作 WHERE 子句的 LIKE 参数	缺省值
作为 IN 参数	HG , LF
用于等于或不等于 (= 、 !=)	HG 、 LF ；另外还有 CMP
用于 WHERE 子句中的范围谓词 (> 、 < 、 >= 、 <= 、 BETWEEN)	LF 、 HG 或 HNG ；另外还有 CMP 、 DATE 、 TIME 、 DTTM
用于 DATEPART 等于谓词、范围谓词和 IN 列表谓词	DATE 、 TIME 、 DTTM
用于 CONTAINS 谓词	WD 、 TEXT

虽然建议使用 **HNG**，但是，在某些情况下，**LF** 或 **HG** 相对较快，且常常可以用于替代 **HNG**。**HNG** 往往可以提供一致的性能，而带有范围的 **LF** 或 **HG** 的性能取决于选择的范围大小。

为了优化查询性能，在连接谓词、子查询谓词、**GROUP BY** 和 **DISTINCT** 子句中使用的列都应具有 **HG** 或 **LF** 索引，因为 **IQ** 不包含除优化程序要使用的索引之外的任何其它统计信息。对高基数列使用 **HG**，对低基数列使用 **LF**，但是，对于所含行数少于 100,000 的表，则应使用 **HG**。

上述估计通常有效；但是，其它因素可能具有更高的优先级：

- 对于范围谓词，更为重要的因素是唯一值数目。
- 对于集合函数 **COUNT**、**COUNT DISTINCT**、**SUM**、**MIN**、**MAX** 和 **AVG**，为了使用除缺省索引以外的任何索引，必须能够使用单个表或连接索引解析整个查询。
- **BIT** 数据只能在缺省索引中使用；大于 255 字节的 **VARBINARY** 数据只能在缺省值、**TEXT** 和 **CMP** 索引类型中使用；大于 255 字节的 **CHAR** 和 **VARCHAR** 数据只能在缺省值、**CMP**、**TEXT** 和 **WD** 索引类型中使用；**LONG VARCHAR** 数据只能在缺省

值、**TEXT** 和 **WD** 索引类型中使用；只有 **DATE** 数据可用于 **DATE** 索引类型；只有 **TIME** 数据可用于 **TIME** 索引类型；只有 **DATETIME** 和 **TIMESTAMP** 数据可用于 **DTTM** 索引类型。

磁盘空间使用情况

下表相较于源数据库或平面文件中的列数据量对每种索引使用的空间大小进行了估计。

表 15. 索引磁盘空间使用情况

索引类型	估计空间与原始数据	注释
缺省值	小于或等于	如果不同的值数目小于 255，此索引使用的空间会远远小于原始数据空间
High_Group	从小于到大 2 倍	随着不同的值数目的减少（即，每组条目数的增加），使用的空间将根据原始数据大小按比例减小
High_Non_Group	小于或等于	在大多数情况下小于原始数据
Low_Fast	从小于到大 2 倍	与 High_Group 相同
日期	小于或等于	大于 High_Non_Group
时间	小于或等于	大于 High_Non_Group
Datetime	小于或等于	大于 High_Non_Group

对于 **LF** 和 **HG** 索引，索引大小取决于唯一值数目。唯一值越多，索引所需的空间越大。

由于 **CMP** 索引始终是附加索引，因此不会节省磁盘空间。

另请参见

- 重叠版本和删除（第 357 页）

索引中的数据类型

在对列创建索引时，仅选择该列的数据类型支持的索引类型。

缺省索引允许任何数据类型。

表 16. 数据类型支持的索引

数据类型	支持的索引	不支持的索引
tinyint	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT

数据类型	支持的索引	不支持的索引
smallint	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
int	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
unsigned int	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
bigint	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
unsigned bigint	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
numeric、decimal	CMP、HG、HNG、LF	WD、DATE、TIME、DTTM、TEXT
double	LF (允许但不推荐使用 HG)	CMP、HNG、WD、DATE、TIME、DTTM、TEXT
float	LF (允许但不推荐使用 HG)	CMP、HNG、WD、DATE、TIME、DTTM、TEXT
real	LF (允许但不推荐使用 HG)	CMP、HNG、WD、DATE、TIME、DTTM、TEXT
bit	(仅限缺省索引)	CMP、HG、HNG、LF、WD、DATE、TIME、DTTM、TEXT
date	CMP、HG、HNG、LF、DATE	WD、TIME、DTTM、TEXT
time	CMP、HG、HNG、LF、TIME	WD、DATE、DTTM、TEXT
datetime、timestamp	CMP、HG、HNG、LF、DTTM	WD、DATE、TIME、TEXT
char <= 255 字节、字符	CMP、HG、HNG、LF、WD、TEXT	DATE、TIME、DTTM
char >255 字节	CMP、WD、TEXT	HG、HNG、LF、DATE、TIME、DTTM
varchar <= 255 字节	CMP、HG、HNG、LF、WD、TEXT	DATE、TIME、DTTM
varchar >255 字节	CMP、WD、TEXT	HG、HNG、LF、DATE、TIME、DTTM

数据类型	支持的索引	不支持的索引
long varchar	WD、TEXT	CMP、HG、HNG、LF、DATE、TIME、DTTM
二进制数据 <= 255 个字节	CMP、HG、LF、TEXT	HNG、WD、DATE、TIME、DTTM
二进制数据 > 255 个字节	CMP、TEXT	HG、HNG、LF、WD、DATE、TIME、DTTM
varbinary <= 255 字节	CMP、HG、LF、TEXT	HNG、WD、DATE、TIME、DTTM
varbinary > 255 字节	CMP、TEXT	HG、HNG、LF、WD、DATE、TIME、DTTM

组合索引类型

如果将在多种查询类型中使用某一列，可能需要使用多种列索引类型。

这些索引类型可以构成优良组合。

表 17. 索引组合

现有索引	要添加的索引					
	HG	HNG	LF	CMP ^a	WD	DATE、TIME 或 DTTM
HG	-	1	2	1	1	1
HNG	1	-	1	1	2	2
LF	2	1	-	1	2	1
1 = 合理组合 2 = 不建议使用的组合 a. CMP 索引适用于一对列。其中的每一列始终至少包含一个其它索引。						

Sybase IQ 索引类型

本节深入分析可能使用各种列索引类型的原因。

快速投影 (FP) 缺省索引类型

当在 Sybase IQ 数据库中创建永久表时，IQ 将所有列值存储在缺省索引中。此缺省索引称为 **FP** (快速投影) 索引，可优化投影并支持计算特定类型的搜索条件。

每一列都具有一个 **FP** 索引，每个 **FP** 都是一个包含 n 个固定长度条目的数组，其中 n 表示表中的行数。每个列值都按 RecordID 升序顺序进行有序存储。

利用少量非重复值或唯一值（例如，状态、日期或月份字段），可以创建一个优化的 **FP** 格式，该格式将减少所需的磁盘页数，并极大地降低列和 I/O 投影开销所需的存储。

这些优化的 **FP** 索引包含以下两部分：一个查找表，其中列中的每个不同的值仅显示一次，以及逻辑值数组，其中逻辑数组的每个元素都是一个指向单元值在查找表中的存储位置的键。

sp_iqindexmetadata 存储过程可以生成一个描述指定的索引或属于指定所有者或表的索引的报告。**sp_iqindexmetadata** 输出显示给定索引是单字节索引、双字节索引、3 字节索引还是平面样式的 **FP** 索引。有关详细信息，请参见《参考：构件块、表和过程》中的“**sp_iqindexmetadata** 过程”。

可以对所含不同的值超过 64K 的列应用标识化。

FP(1) 索引

当列值数少于 256 时，会创建单字节 **FP** 索引。

每一行的实际键值存储在查找表中，**FP** 条目（查找键）包含查找表中的索引。对于单字节 **FP** 索引，每个 **FP** 条目各占 1 字节。

如果将缺省 main 高速缓存设置为 32MB，将缺省 **FP_LOOKUP_SIZE_PPM** 设置为 2500，使用 **LOAD**、**INSERT** 或 **UPDATE** 向 **char(2)**、**binary(2)** 或小整数添加新的不同的值可将 **FP(1)** 转换为平面 **FP**。

FP(2) 索引

如果列值数超过 256 但小于等于 65536，则会生成查找表。每个 **FP** 条目（查找键）包含查找表中的索引，但各占 2 字节而不是 1 字节。

为避免将单字节条目转换为双字节条目所带来的开销，用户可以在创建表时将 **IQ UNIQUE** 指定为大于 256 且小于或等于 65536 的值。

FP(3) 索引

如果离散列值个数超过 65536，则可以创建 3 字节 **FP** 索引。

FP(3) 索引在结构上与 **FP(1)** 和 **FP(2)** 索引类似，但主要具有下列区别：

- **FP(3)** 查找表的最大大小为 16777216，而不是 65536。
- **FP(3)** 索引缓冲区存储包含的每个查找键均为 3 字节。

只要离散值个数不超过 16777216，3 字节索引则将值存储到某个列（列数据）中。仅当列数据大小大于 3 字节时，用户才会对列创建 3 字节索引。因此，您不能对数据类型为 **BIT**、**TINYINT**、**SMALLINT**、**CHAR(<=3)**、**VARCHAR(<=3)**、**BINARY(<=3)** 和 **VARBINARY(<=3)** 的列创建 **FP(3)** 索引。对于 **LONG VARCHAR** 和 **LONG BINARY** 数据类型，Sybase IQ 也不支持 **FP(3)** 索引。

要创建 **FP(3)** 索引，必须将 **MINIMIZE_STORAGE** 设置为 **ON**，或者必须使用介于 65537 和 16777216 之间（包括边界值）的 **IQ UNIQUE** 约束值创建列。只要唯一值计数超过 65536，也可以从 **FP(2)** 索引创建 **FP(3)**，但仅限以下 Sybase IQ 快速投影 (**FP**) 索引表所示的数据大小，且不包括数据类型 **BIT**、**TINYINT**、**SMALLINT**、

CHAR(<=3)、VARCHAR(<=3)、BINARY(<=3) 和 VARBINARY(<=3)。请参见《参考：语句和选项》中的 **MINIMIZE_STORAGE** 选项。

行为更改

FP 索引的行为与早于 Sybase IQ 15.0 的版本存在一些不同之处。

表 18. Sybase IQ 快速投影 (FP) 索引

离散值个数	列数据大小 = 1 字节	列数据大小 = 2 字节	列数据大小 = 3 字节	列数据大小 >3 字节
<257	FP(1)	FP(1)	FP(1)	FP(1)
257 - 65536	-	FP(2)	FP(2)	FP(2)
65537-16777216	-	-	平面	FP(3)
>16777216	-	-	-	平面

与 FP(1) 和 FP(2) 索引类似，数据类型宽度大于 255 字节或小于 4 字节的列不支持 FP(3) 索引。

仅当查找表使用的空间小于 FP_LOOKUP_SIZE 选项的当前值，且小于 FP_LOOKUP_SIZE_PPM 的当前设置所指定的相应 main 高速缓存部分时，才允许创建 FP(3) 索引、溢出或强制转换为 FP(3) 索引。

Sybase IQ 中使用的查找页的最大数目由 FP_LOOKUP_SIZE 选项和 FP_LOOKUP_SIZE_PPM 选项控制（取二者中的较小者）。请参见《参考：语句和选项》。

下表基于以下公式计算 FP(3) 索引中可支持的最大不同的值个数：

$$FP_LOOKUP_SIZE / (Column-Data\ size + Cardinality\ size)$$

基数大小是为存储查找存储中的所有单独数据而保留的空间。基数大小的值可以为 4 或 8 字节。在本示例中，其值为最大值 8 字节。

此表还包含值数目小于 16777216 的相应示例，在该表中，对于小于预期的 16777216 的唯一值计数，将滚动更新为平面 FP。

表 19. FP(3) 中的最大唯一值

FP_LOOKUP_SIZE (MB)	列数据类型宽度 (字节)					
	4	8	32	64	128	255
1 MB	87381	65536	26214	14563	7710	3986
4 MB	349525	262144	104857	58254	30840	15947
8 MB	699050	524288	209715	116508	61680	31895

16 MB	1398101	1048576	419430	233016	123361	63791
32 MB	2796202	2097152	838860	466033	246723	127583
64 MB	5592405	4194304	1677721	932067	493447	255166
128 MB	11184810	8388608	3355443	1864135	986895	510333
256 MB	16777216	16777216	6710886	3728270	1973790	1020667

注意：此表中所示的值是根据给定的 `FP_LOOKUP_SIZE` 选项值估算的列中唯一值个数；实际值可能有所不同。由于计数可能存储为 4 字节或 8 字节，因此可能存在这种变化。

此表的计算条件是 `FP_LOOKUP_SIZE` 的值小于或等于 `FP_LOOKUP_SIZE_PPM` 的值。

另请参见

- 对列或表的 UNIQUE 约束（第 326 页）
- 配置 FP(3) 索引（第 199 页）

配置 FP(3) 索引

当配置 3 字节索引时，您可能需要调整临时高速缓存大小。

可以使用服务器启动命令行参数 `-iqtc` 或 `sa_server_option` 系统过程 `temp_cache_memory_mb` 选项设置值，如下所示：

```
CALL sa_server_option('temp_cache_memory_mb', value)
```

枚举 FP 索引使用散列对象管理在列中表示的值。根据不同的值的数目和列宽，3 字节 FP 使用的散列对象的大小可能会很大。利用分配的足够大的临时高速缓存，通过允许将整个散列对象保留在高速缓存中，将 `HASH_PINNABLE_CACHE_PERCENT` 选项的值增大到缺省值 20% 以上可以提高性能。

FP(3) 高速缓存使用情况

为了充分利用 FP(3) 索引，请设置 `FP_LOOKUP_SIZE` 选项的值，使其大于缺省值 16MB。

FP(3) 装载

带有 3 字节索引的列需要额外的高速缓存来装载数据。在装载带有 3 字节索引的列之前，请将 `FP_LOOKUP_SIZE` 设置为相应的值。

如果已占用的缓冲区不足，Sybase IQ 将在 `.iqmsg` 文件中返回一个警告，该文件还包含有关可能的抖动的通知：

```
Warning: Hash Insert forced buffer unpinning detected for FP Index
Warning: Hash Insert thrashing detected for FP Index
```

另请参见

- FP(3) 索引 (第 197 页)
- 对列或表的 UNIQUE 约束 (第 326 页)

平面 FP 索引

如果不同的值数目超过 16777216, 则不会创建任何查找表。每个平面样式的 FP 索引都包含一个实际列单元值。

如果 MINIMIZE_STORAGE 为 ON, 则可以避免将查找 FP 条目转换为平面样式的开销。当特定字段的非重复行计数增至 16777216 以上时, FP(3) 索引则自动转换为平面样式的 FP 索引。要创建平面 FP, 请在创建表时将 IQ UNIQUE 值指定为大于 16777216 的值。

注意: 当使用 DATE 数据类型创建表时, 将对 DATE 字段创建一个双字节 FP 索引, 该索引独立于数据库选项 MINIMIZE_STORAGE 中的设置。

要对 DATE 字段创建三字节 FP 或平面样式的 FP 索引, 请在创建表时使用下列值:

- 对于 3 字节 FP 索引 - IQ UNIQUE 值介于 65537 和 16777216 之间。
 - 对于平面样式的 FP 索引 - IQ UNIQUE 值大于 16777216。
-

Low_Fast (LF) 索引类型

此索引适用于所含唯一值非常少 (低于 1,000) 的列, 这些唯一值可能包括: 性别、是/否、True/False、相关项编号、工资级别等。LF 是 Sybase IQ 中最快的索引。

当测试相等性时, 只需查找一次即可快速提供结果集。要测试不等性, 您可能需要多查找几次。使用此索引还可以快速计算 SUM、AVG 和 COUNT 等。

随着列中唯一值数量的增多, 性能将开始下降, 插入和某些查询的内存和磁盘要求则开始增加。但是, 当执行相等测试时, 此索引的速度仍然最快, 即使对于具有大量唯一值的列也是如此。

建议用法

LF 索引的建议用法如下所示:

请在下列情况下使用 LF 索引:

- 列的唯一值数目少于 1,000。
- 列的唯一值数目少于 1,000 并且该列用于连接谓词中。

切勿对唯一值数目大于等于 10,000 的列使用 LF 索引。如果表具有的行数少于 25,000 行, 请使用 HG 索引, 因为使用该索引执行相同操作需要更少的磁盘 I/O 操作。

Low_Fast 的优缺点

下表列出了 Low_Fast 索引的优缺点。

表 20. LF 优点/缺点

优点	缺点
此索引速度很快，对于单表 SUM 、 AVG 、 COUNT 、 COUNT DISTINCT 、 MIN 和 MAX 运算尤其如此。	最多只能用于 10,000 个唯一值。 如果列中的数据为 BIT ， VARBINARY > 255 字节、 CHAR > 255 字节或 VARCHAR > 255 字节，则不能使用此索引。

与其它索引比较

熟悉 LF 索引与其它索引的比较情况。

HNG/HG

考虑的主要因素是列中的唯一值数目。如果该数目较低，请使用 **LF**。

其它索引

High_Non_Group 索引类型也适用于 **Low_Fast** 列。

注意：如果唯一值数目较小（小于 1,000），则最好始终使用 **LF** 索引。如果列显示在 **WHERE** 子句中，应首先考虑此索引。仅当唯一值数目较大时，才应考虑其它索引（**HG** 和 **HNG**）。对于具有大量唯一值的范围查询，也请考虑使用 **HNG** 索引。

High_Group (HG) 索引类型

High_Group 索引通常用于带有整数数据类型的连接列。由于该索引可有效处理 **GROUP BY**，因此其使用频率高于 **High_Non_Group**。

建议用法

HG 索引和多列 **HG** 索引提供不同的功能，在选择要使用哪种索引时必须考虑其不同的功能。

请在下列情况下使用 **HG** 索引：

- 该列将用于连接谓词中
- 列的唯一值数目大于 1000

使用多列 **HG** 索引可以提高带有对多个列的引用的 **ORDER BY** 查询的性能。此更改对于用户来说是透明的，但改进了查询性能。

注意：外键列需要其自己的单独 **HG** 索引。但是，如果存在连接索引，同一列不能同时具有显式创建的 **HG** 索引和外键约束。

High_Group 的优缺点

High_Group 索引具有以下优缺点。

表 21. HG 优点/缺点

优点	缺点
<p>可快速处理具有 GROUP BY 的查询。</p> <p>此索引有助于连接索引处理。它是其中一个针对在连接关系中使用的列而推荐的索引。另一个索引是 LF。</p>	<p>同 HNG 索引相比，此索引需要额外的磁盘空间（可能需要三倍多的原始数据空间）。</p> <p>此索引类型填充数据和删除的时间最长。</p> <p>如果列中的数据为 BIT，VARBINARY > 255 字节、CHAR > 255 字节或 VARCHAR > 255 字节，则不能使用此索引。</p> <p>不建议将此索引用于 FLOAT、REAL 和 DOUBLE 数据。</p>

与其它索引比较

下面是与其它索引的比较。

LF

唯一值数目是决定因素。如果列的唯一值数量较多，请使用 **High_Group**。如果唯一值数量较少，请使用 **Low_Fast**。

HNG

决定因素是列是否是连接列以及/或者是否在该列上处理 **GROUP BY**。如果满足上述任一条件，请单独使用 **High_Group**，也可以与 **High_Non_Group** 配合使用。在其它情况下，请使用 **High_Non_Group** 以节省磁盘空间。

其它索引

在某些情况下，可以在查询中使用符合 **High_Group** 索引条件的列，以便通过不同索引类型提高速度。在这种情况下，请为该列创建其它索引。

自动创建 High_Group 索引

当发出 **CREATE INDEX** 语句时，Sybase IQ 将在缺省情况下创建 **High_Group** 索引，而不必指定索引类型。

Sybase IQ 自动为任何 **UNIQUE**、**FOREIGN KEY** 或 **PRIMARY KEY** 约束创建 **High_Group** 索引。对于单列外键，Sybase IQ 创建一列非唯一的 **High_Group** 索引。对于多列外键，将隐式创建非唯一的组合 **High_Group** 索引。非唯一的 **HG** 索引允许重复值，并且以可选方式允许 **NULL**。它为参照完整性提供构件块，并且可用于提高查询性能。

如果列定义允许使用空值并且不强制执行约束（主键或唯一），则 Sybase IQ 允许在用户创建的唯一多列 **HG** 索引的数据值中使用空值。有关详细信息，请参见《参考：语句和选项》>“SQL 语句”>“CREATE INDEX 语句”>“注释”中的“多列索引”。

由于非唯一的组合 **High_Group** 索引可对连接和结果大小的基数提供更准确的估计，因此使用多列连接或多列 **group by** 子句进行查询可提高性能。此外，还可以优化推入和子查询。

High_Non_Group (HNG) 索引类型

当需要执行范围搜索时，请添加 **HNG** 索引。

HNG 索引需要的磁盘空间比 **HG** 索引需要的磁盘空间大约少三倍。仅基于这一点，当不需要执行组操作时，请使用 **HNG** 索引，而不要使用 **HG** 索引。

相反，如果您确定 **HG** 索引可以更有效地处理要执行的查询，或者如果列属于连接的一部分和/或您希望强制执行唯一性，请使用 **HG** 索引。

注意： 对于连接四个或更多表的复杂即席查询，使用 **HNG** 索引代替 **HG** 索引可能会严重降低性能。如果查询性能对应用程序中的此类查询非常重要，请同时选择 **HG** 和 **HNG**。

建议用法

请注意 **HNG** 索引的建议用法。

请在下列情况下使用 **HNG** 索引：

- 唯一值数目很大（大于 1000）
- 无需对列执行 **GROUP BY**

High_Non_Group 的优缺点

有关使用 **High_Non_Group** 索引的优缺点，请参见下表。

表 22. **HNG** 优点/缺点

优点	缺点
由于采用压缩算法，因此可以降低磁盘要求，而不会以牺牲性能为代价。 如果列包含大量唯一值，此索引的速度最快，但下述情况例外。	不建议将此索引用于 GROUP BY 查询。 如果强制执行唯一性则不可建立索引。 如果列中的数据为 FLOAT 、 REAL 、 DOUBLE 、 BIT 、 BINARY 、 VARBINARY 、 CHAR > 255 字节或 VARCHAR > 255 字节，则不能使用此索引。

与其它索引的比较

HNG 索引与类似索引的比较。

- **HNG** 需要的磁盘空间比 **HG** 少，但无法有效地执行 **GROUP BY**。
- 在 **LF** 和 **HNG** 之间选择时，唯一值数量是决定因素。如果唯一值数量大于 1000，请使用 **HNG**。

其它索引

High_Group 索引同样适用于 **HNG** 列。

比较 (CMP) 索引类型

比较 (CMP) 索引是两列之间的关系的索引。您可以对具有相同数据类型、精度和标度的任何两个不同列创建比较索引。**CMP** 索引存储对该索引的两个列的二进制比较结果 (<、> 或 =)。

另请参见

- 索引创建示例 (第 190 页)

建议用法

CMP 索引类型的建议用法。

可以在空值、非空值或混合有空值和非空值的列上创建 **CMP** 索引。**CMP** 索引不能唯一。注意，数值数据类型和小数数据类型均视为相同类型。当精度和标度相同时，您可以对这些数据类型的列创建 **CMP** 索引。对于 CHAR、VARCHAR、BINARY 和 VARBINARY 列，精度表示具有相同的列宽。

例如，以下命令显示如何创建表以及在之后创建相应的比较索引：

```
CREATE TABLE f(c1 INT NOT NULL, c2 INT NULL, c3 CHAR(5), c4 CHAR(5))
CREATE CMP INDEX c1c2cmp ON f(c1, c2)
```

以下索引为非法索引，因为建立索引的列具有不同的数据类型、精度和标度：

```
CREATE CMP INDEX c1c3cmp ON f(c1, c3)
```

限制

以下限制适用于 **CMP**：

- 您可以删除 **CMP** 索引。
- **CMP** 索引不能唯一。
- **CMP** 索引不能在基础连接索引中复制。
- 如果尝试更改或删除在 **CMP** 索引中定义的列，则会引发异常。
- 用户不能对在 **CMP** 索引中定义的现有列执行 **ALTER TABLE MODIFY**。
- **CMP** 索引不支持 BIT、FLOAT、DOUBLE 和 REAL 数据类型。

包含 (WD) 索引类型

使用包含 (WD) 索引，可以存储包含 CHAR、VARCHAR 和 LONG VARCHAR 数据的列字符串中的单词。

注意：要创建 LONG VARCHAR 列，您必须获得专门许可才能使用非结构化数据分析功能。有关非结构化数据分析选项的详细信息，请参见《Sybase IQ 中的非结构化数据分析》。

建议用法

使用 **WD** 索引以最快速度访问含有关键字列表的列（例如，在书目记录或网页中）。

以下限制适用于 **WD**：

- 不能指定 UNIQUE 属性。
- **WD** 索引仅与 CONTAINS 或 LIKE 谓词一起使用。
- 列名必须标识基表中的一个 CHAR、VARCHAR 或 LONG VARCHAR 列。
- 允许的最小列宽为 3 字节，允许的最大列宽为 LOB 列的最大宽度。（最大长度为 4 GB 乘以数据库页大小。）
- 必须将分隔符列表用单引号引起来。Sybase Central 创建索引向导在提示输入分隔符时不会指明这一点，如果省略分隔符，则会返回错误。
- 如果省略 DELIMITED BY 子句或指定的 *separators-string* 值为空字符串（单引号），则 Sybase IQ 将使用缺省分隔符集。缺省分隔符集包括非 7 位 ASCII 字母数字字符的所有 7 位 ASCII 字符，但连字符和单引号除外，在缺省情况下，连字符和单引号属于字词的一部分。缺省分隔符集中有 64 个分隔符。
- 如果指定了多个 DELIMITED BY 和 LIMIT 子句，则不会返回错误，但仅使用每种类型的最后一个子句。例如，下面两个语句返回相同结果：

语句 1:

```
CREATE WD INDEX clwd on foo(c1) DELIMITED BY 'f' LIMIT 40 LIMIT 99
DELIMITED BY 'g' DELIMITED BY 'h';
```

语句 2:

```
CREATE WD INDEX clwd on foo(c1) DELIMITED BY 'h' LIMIT 99;
```

- 在创建 **WD** 索引后，到其列的任何插入都将使用分隔符进行分析，并且在创建索引后，最大单词大小将无法更改。

对于 CHAR 列，Sybase 建议您至少指定一个空格作为其中的一个分隔符或使用缺省分隔符集。Sybase IQ 会自动将 CHAR 列填充至最大列宽。如果列包含空白和字符数据，查询 **WD** 索引数据返回的结果可能会造成误解。例如，列 `company_name` 包含由分隔符分隔的两个单词，但第二个单词以空白填充：

```
'Concord' 'Farms'
```

假设用户输入以下查询：

```
SELECT COUNT(*)FROM Customers WHERE CompanyName contains ('Farms')
```

分析程序确定字符串包含：

'Farms'

而不是:

'Farms'

并返回 1。您可以通过使用 VARCHAR 列而不是 CHAR 列避免此问题。

- **sp_iqcheckdb** (DBCC 一致性检查程序) 的分配、检查、检验和修复模式支持 CHAR、VARCHAR 和 LONG VARCHAR 列的 **WD** 索引。

WD 的优缺点

有关使用 **WD** 索引的优缺点，请参见下表。

表 23. WD 优点/缺点

<i>优点</i>	<i>缺点</i>
大量装载时可极大地提高性能。 使用此索引，某些 LIKE 谓词的执行速度将变快。 与此索引一起使用的 CONTAINS 谓词的优先级高于 LIKE 谓词。 是对关键字或部分 URL 建立索引的最佳方式。	潜在磁盘空间要求可能会非常大。 如果强制执行唯一性则不可建立索引。 只有在列中的数据为 CHAR、VARCHAR 或 LONG VARCHAR 时，才能使用此索引。

日期 (DATE)、时间 (TIME) 和日期时间 (DTTM) 索引类型

用于处理涉及日期、时间或日期时间数量的索引类型有三种：

- **DATE** 索引用于对 DATE 数据类型的列处理涉及日期数量的特定查询。
- **TIME** 索引用于对 TIME 数据类型的列处理涉及时间数量的特定查询。
- **DTTM** 索引用于对 DATETIME 或 TIMESTAMP 数据类型的列处理涉及日期时间数量的特定查询。

DATE、TIME 和 DTTM 索引类型的建议用法

如果在含有日期和时间函数及运算的查询中使用 DATE、TIME、DATETIME 或 TIMESTAMP 列，请在下列情况下使用 **DATE**、**TIME** 或 **DTTM** 索引。

- 使用 DATEPART 相等谓词 (=、!=)、DATEPART 范围谓词 (>、<、>=、<=、!>、!
<、BETWEEN) 和 DATEPART IN 列表谓词的查询
- 使用范围谓词 (>、<、>=、<=、BETWEEN) 的查询

注意： 对于用于 DATE、TIME、DATETIME 或 TIMESTAMP 列的简单相等谓词 (无 DATEPART)，使用 **LF** 和 **HG** 索引可获得最佳性能。如果 **LF** 或 **HG** 索引不可用，则使用 **DATE**、**TIME** 或 **DTTM** 索引来获得结果。

对于相等谓词 (包括连接条件) 或 IN 谓词，如果在 GROUP BY 子句或 WHERE/HAVING 子句中使用 DATE、TIME、DATETIME 或 TIMESTAMP 列，则该列需要 **LF** 或 **HG** 索引，因为只有这些索引才能实现快速相等。

本节示例中所使用的表 `tab` 包含按如下方式定义的列：

```
CREATE TABLE tab (col1 DATE, col2 DATETIME, col3 TIME);
```

另请参见

- 其它索引（第 209 页）

使用 DATEPART 相等谓词、范围谓词和 IN 列表谓词的查询

对于使用相等谓词 (= 或 !=) 的查询，如果比较一侧为 DATEPART 表达式或某些其它日期和时间函数（例如，YEAR、QUARTER、DAY、MINUTE），比较的另一侧为常量表达式（包括常量值或宿主变量），则使用 **DATE**、**TIME** 或 **DTTM** 索引（如果索引可用）来获得结果集。

例如，在下列查询中使用 **DATE**、**TIME** 或 **DTTM** 索引：

```
SELECT * FROM tab WHERE DATEPART(YEAR, col1) = 2002; SELECT * FROM
tab WHERE DATEPART(HOUR, col2) = 20; SELECT * FROM tab WHERE MINUTE
(col3) != 30; SELECT * FROM tab WHERE DATEPART(MONTH, col2) = @tmon;
```

其中，@tmon 为 INTEGER 宿主变量。

用于与 **DATE**、**TIME** 和 **DTTM** 索引一起处理的相应 DATEPART 范围谓词和 IN 列表谓词条件包括：

- **COMPARISON** 条件 >、<、>=、<=、!>、!<

运算符一侧是日期/时间函数或 DATEPART 函数，其参数为表列或视图列。运算符另一侧是常量表达式，如整数或整数类型宿主变量。例如，

```
DATEPART(WEEK, col1) !<23
DATEPART(YEAR, col1) = 2001
HOUR(col3) >= 1
```
- **BETWEEN**……**AND** 条件

BETWEEN 左侧是日期/时间函数或 DATEPART 函数，其参数为表列或视图列。**AND** 两侧均为常量表达式，如整数或整数类型宿主变量。例如，

```
DATEPART(YEAR, col1) BETWEEN host-var1 AND host-var2
```
- **IN** 条件

IN 左侧是日期/时间函数或 DATEPART 函数，其参数为表列或视图列。**IN** 列表中的值为常量表达式。例如，

```
DATEPART(MONTH, col1) IN (1999, 2001, 2003)
```

注意： **DATE**、**TIME** 和 **DTTM** 索引不支持某些日期分量（Calyearofweek、Calweekofyear、Caldayofweek、Dayofyear、Millisecond）。例如，

```
SELECT * FROM tab WHERE DATEPART(MILLISECOND, col3) = 100; SELECT *
FROM tab WHERE DATEPART(DAYOFYEAR, col1) <= 89;
```

在这些情况下，查询优化程序选择其它索引来获取结果。

使用范围谓词的查询

检查查询在您为谓词列创建索引时如何使用范围谓词。

在使用范围谓词的下列情况下，选择 **DATE**、**TIME** 或 **DTTM** 索引处理查询：

- 比较条件：

```
SELECT * FROM tab WHERE col1 < '2002/10/09' ; SELECT * FROM tab
WHERE col2 >= '2002/01/01 09:12:04.006' ;
```

比较运算符一侧是列名，另一侧是常量表达式（常量值或宿主变量）。

- **Between** 条件：

```
SELECT * FROM tab WHERE col3 BETWEEN '09:12:04.006' AND
'20:12:04.006' ; SELECT * FROM tab WHERE col2 BETWEEN
tmp_datetime1 AND tmp_datetime2;
```

对于这些查询类型，**DATE**、**TIME** 或 **DTTM** 索引通常比 **HNG** 索引执行得更快。

在下面三种特定情况下，使用 **DATE** 或 **DTTM** 索引可以大幅提高性能：

- 谓词范围恰好是一年或多年（实际开始日期是某一年的开始，并且实际结束日期是某一年的结束）。例如，

```
SELECT * FROM tab WHERE col1 BETWEEN '1993-01-01' AND
'1996-12-31' ; SELECT * FROM tab WHERE col1 >= '1993-01-01' AND
col1 < '1997-01-01' ; SELECT * FROM tab WHERE col2 BETWEEN
'1993-01-01 00:00:00.000000' AND '1996-12-31 23:59:59.999999' ;
```

- 谓词范围恰好是一个月或同一年中的多个月（实际开始日期是某个月的开始，并且实际结束日期是某个月的结束）。例如，

```
SELECT * FROM tab WHERE col1 > '1993-01-31' AND col1 <=
'1993-06-31' ; SELECT * FROM tab WHERE col2 >= '1993-01-01
00:00:00.000000' AND col1 < '1993-06-01 00:00:00.000000' ;
```

- 谓词范围恰好是一天。例如，

```
SELECT * FROM tab WHERE col2 >= '1993-01-31 00:00:00.000000' AND
col2 <= '1993-01-31 23:59:59.999999' ;
```

注意：在上面三种情况下，必须注意年范围、月范围和恰好一天的概念。例如，识别为年范围的 **DTTM** 索引包含四种情况：

```
col2 > ' year1/12/31 23:59:59.999999' and col2 < ' year2/01/01
00:00:00.000000' col2 >= ' year1/01/01 00:00:00.000000' and col2
< ' year2/01/01 00:00:00.000000' col2 > ' year1/12/31
23:59:59.999999' and col2 <= ' year2/12/31 23:59:59.999999' col2
>= ' year1/01/01 00:00:00.000000' and col2 <= ' year2/12/31
23:59:59.999999'
```

以下示例中的范围不符合年范围：

```
col2 > ' year1/12/31 23:59:59.999999' and col2 <= ' year2/01/01
00:00:00.000000' col2 > ' year1/01/01 00:00:00.000000' and col2
< ' year2/01/01 00:00:00.000000'
```

第一个范围不相符，因为除年范围之外，它还包括值“year2/01/01 00:00:00:000000”。第二个范围缺少值“year1/01/01 00:00:00:000000”。

类似具体信息也适用于 **DTTM** 和 **DATE** 索引的月范围和恰好一天。

如果较小的日期范围（小于 60 个值）不适用于上面三种特定情况，则 **LF** 和 **HG** 索引比 **DATE** 索引执行得更快。

DATE/TIME/DTTM 的优缺点

有关使用 **DATE**、**TIME** 或 **DTTM** 索引的优缺点，请参见下表。

表 24. DATE/TIME/DTTM 优点/缺点

优点	缺点
同其它索引类型相比，对日期、时间或日期时间数量的查询可以更快进行解析。 可以创建和删除 DATE 、 TIME 或 DTTM 索引。	使用的磁盘空间大于 HNG 索引。 实现快速相等仍需要 LF 或 HG 索引。 只有在列中的数据为 DATE 、 TIME 、 DATETIME 或 TIMESTAMP 数据类型时，才能使用这些索引。

对 DATE/TIME/DTTM 索引的限制

以下限制当前适用于 **DATE**、**TIME** 和 **DTTM** 索引：

- 不能使用 **UNIQUE** 关键字。
- 只能对单个列创建。
- 不支持日期分量 **Calyearofweek**、**Calweekofyear**、**Caldayofweek**、**Dayofyear**、**Millisecond**。

与其它索引的比较

DATE、**TIME** 和 **DTTM** 索引的性能与 **HNG** 索引相当。

同 **HNG** 相比，在支持的情况下，**DATE**、**TIME** 和 **DTTM** 索引通常比 **HNG** 执行得更快（最快相当于两倍的速度）。在“建议用法”一节讨论的特殊情况下，**DATE**、**TIME** 和 **DTTM** 索引的性能甚至更佳。因此，不必在 **DATE**、**TIME**、**DATETIME** 或 **TIMESTAMP** 数据类型的列上与 **DATE**、**TIME** 或 **DTTM** 索引一起使用 **HNG** 索引。

其它索引

如果在 **WHERE** 子句、**ON** 条件或 **GROUP BY** 子句中引用 **DATE**、**TIME**、**DATETIME** 或 **TIMESTAMP** 数据类型的列，建议始终对这些列建立 **DATE**、**TIME** 或 **DTTM** 索引。

此外，**HG** 或 **LF** 索引也适用于 **DATE**、**TIME**、**DATETIME** 或 **TIMESTAMP** 列，特别是对这些列计算相等谓词时。如果常常在 **GROUP BY** 子句中使用这些列，并且不同的值数量少于 1000（即少于三年的时间），则还推荐使用 **LF** 索引。

另请参见

- **DATE**、**TIME** 和 **DTTM** 索引类型的建议用法（第 206 页）

TEXT 索引

要使用 **TEXT** 索引，您必须获得专门许可才能使用非结构化数据分析功能。

注意： 请参见《Sybase IQ 中的非结构化数据分析》。

与在列字符串中使用关键字的包含 (WD) 索引不同，**TEXT** 索引可存储带索引的列中各个词的位置信息。使用 **TEXT** 索引的查询会比必须扫描表中所有值的查询要快。

创建 TEXT 索引

使用 Sybase Central 创建 **TEXT** 索引。

请参见《Sybase IQ 中的非结构化数据分析》。

优化即席连接的性能

可以创建索引来优化即席连接的性能。

为了以最快速度处理即席连接，请对下列各项可能引用的所有列创建 **Low_Fast** 或 **High_Group** 索引：

- 即席连接查询的 **WHERE** 子句
- 集合函数之外的即席连接查询的 **HAVING** 子句条件

例如：

```
SELECT n_name, sum(l_extendedprice*(1-l_discount)) AS revenue
FROM customer, orders, lineitem, supplier, nation, region
WHERE c_custkey = o_custkey AND o_orderkey =
l_orderkey AND l_suppkey = s_suppkey AND
c_nationkey = s_nationkey AND s_nationkey =
n_nationkey AND n_regionkey = r_regionkey AND
r_name = 'ASIA' AND o_orderdate >= '1994-01-01'
AND o_orderdate < '1995-01-01' GROUP BY n_name HAVING n_name LIKE
"I%" AND SUM(l_extendedprice*(1-l_discount)) > 0.50 ORDER BY 2
DESC
```

在此查询中引用的所有列 (**l_extendedprice** 和 **l_discount** 除外) 都应具有 **LF** 或 **HG** 索引。

选择索引

特定条件 (例如唯一值的数目) 有助于为数据确定适当的索引类型。

标识条件	选择的索引
指明对所有列自动创建的索引。	缺省索引
指明对具有 UNIQUE 或 PRIMARY KEY 约束的列自动创建的索引。	已强制执行 UNIQUE 的 HG

标识条件	选择的索引
标识在连接谓词中使用的所有列，并根据唯一值数量选择索引类型。	HG 或 LF
标识包含少量唯一值且尚未使用多个索引的列。	LF
标识具有大量唯一值且包含在 SELECT DISTINCT 或 DISTINCT COUNT 的选择列表的 GROUP BY 子句中的列。	HG
标识可在即席连接查询的 WHERE 子句中使用且没有 HG 或 LF 索引的列。	HG 或 LF
标识具有大量唯一值且不会与 GROUP BY 、 SELECT DISTINCT 或 DISTINCT COUNT 一起使用的列。	HNG
标识可能需要经常比较的具有相同数据类型、精度和标度的列对。	CMP
标识包含关键字或 URL 列表的列。	WD
标识具有大量唯一值且不会与 GROUP BY 、 SELECT DISTINCT 或 DISTINCT COUNT 一起使用的 DATE 、 TIME 、 DATETIME 或 TIMESTAMP 列。	DATE 、 TIME 或 DTTM
查看所有其余列，并根据唯一值数量、查询类型和磁盘空间确定其它索引。此外，对于所有列，确保您选择的索引类型允许该列的数据类型。	

在插入数据后添加列索引

当创建其它列索引时，**CREATE INDEX** 命令将创建新索引，以作为各个表和包含该列的任何连接索引的一部分。此规则的唯一例外情况是 **CMP** 和多列 **HG** 索引。

如果各个表中的现有列索引已含有数据，**CREATE INDEX** 语句还会从现有索引向新索引插入数据。这样可确保各个表中多个列的所有列索引之间的数据完整性。当为以前装载的作为连接索引一部分的表添加索引时，也会自动插入和同步数据。

如果您在插入数据后发现某一列需要其它索引，此功能则非常有用。您可以添加索引，而不必重新开始。

注意： 插入现有索引中的数据可能会非常慢。在插入数据之前创建所有相应索引，然后使用 **LOAD TABLE** 或 **INSERT** 语句一次插入所有索引，此方法的速度始终相对较快。

另请参见

- 同步连接索引（第 217 页）

使用连接索引

如果您知道同一数据库中的某些特定表通常将以一致方式连接，您可能希望为这些表创建 *连接索引*。

当创建连接索引时，Sybase IQ 生成一个可将表的各个列关联起来的新的内部结构。它表示两个或多个表，包括内部行、左外行和右外行。

另请参见

- 表类型（第 172 页）
- 有效数据的规则和检查（第 317 页）

连接索引提高查询性能

同最初在查询时定义表连接（即席连接）相比，连接索引通常可提高查询性能。但是，在大多数情况下，您可以在不创建连接索引的情况下基于连接列获取最佳性能。

连接索引的装载注意事项

同其它 IQ 索引相比，装载连接索引需要更多空间和时间。要装载连接索引，必须先装载基础表，然后装载连接索引。

如何使用连接索引进行查询

创建连接索引后，其用途通过 **SELECT** 语句的标准确定。

如果存在的连接索引按 **WHERE** 子句中指定的关系连接 **FROM** 子句中的表，或者如果存在的连接索引基于自然连接或键连接的 ANSI 连接语法，则连接索引用于加快查询速度。在其它情况下，将在查询时执行各个表上的索引之间的即席连接。如果 **SELECT** 中的表子集存在连接索引，则 Sybase IQ 使用它来加快所生成的即席连接的速度。

连接索引中的关系

Sybase IQ 连接索引支持一对多连接关系。一对多关系的一个简单示例是销售代表和客户。销售代表可有多个客户，但客户只能有一个销售代表。

此关系可以有多个级别。但是，应始终指定两个表或表与较低级别的连接之间的连接关系。表示关系的“多”方的表称为 *顶级表*。

另请参见

- 连接层次概述（第 213 页）
- 查询解析中的连接层次（第 214 页）
- 创建连接索引的步骤（第 216 页）

当连接成为即席连接时

如果没有任何连接索引处理查询中涉及的所有引用表，则可以使用即席连接解析查询。

由于无法创建连接索引来表示多对多连接关系，因此只能针对此关系发出即席查询。即席查询非常灵活，但是在某些情况下，这种灵活性是以牺牲性能为代价的。如果为连接索引留有足够空间，并且您不需要多对多关系或多级星形连接索引，在性能至关重要的情况下创建连接索引可能会很有用。

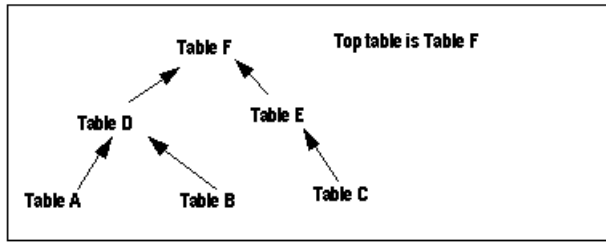
连接层次概述

Sybase IQ 支持的所有连接关系必须具有层次。可以把连接层次看作说明连接中的所有表如何连接的树。

Sybase IQ 连接层次在连接结束的树的顶部具有一个表。此表称为 *顶级表*，它不与任何其它表相连，但其它表可以与该表相连。顶级表始终表示一对多关系中的“多”方。

根据连接的复杂程度，沿着树往下至树的底部和连接开头的多个表可能呈一条直线，也可能会有很多侧枝。下图显示具有两个分支的连接层次。

图 1： 连接关系层次



在连接层次中：

- 表只能出现一次
- 表只能向外连接一次（具有一个离开箭头）
- 必须连接所有表

另请参见

- 连接索引中的关系（第 212 页）
- 查询解析中的连接层次（第 214 页）
- 创建连接索引的步骤（第 216 页）
- 表定义中的连接关系示例（第 219 页）
- 创建连接索引时指定连接类型（第 219 页）
- 创建连接索引时指定关系（第 220 页）

- 连接层次类型 (第 221 页)

连接索引中的列

假定使用名为 **ABCDE** 的连接索引连接表 **A** 至表 **E**。如果每个表具有两列数据，预计连接索引总共有十四列。

Sybase IQ 为每个连接表（顶级表除外）创建一个附加列，即 **ROWID** 列。这样，总共有十列（五个表中的每个表各具有两列）和四个 **ROWID** 列。

在列索引中插入时，可使用 **LOAD TABLE** 或 **INSERT** 语句的 **NOTIFY** 选项接收通知消息。这些消息标识连接索引中的每个列，包括 **ROWID** 列。

您可以使用 **NOTIFY_MODULUS** 选项设置这些消息的频率，并在 **CREATE DATABASE** 或 **LOAD TABLE** 命令中覆盖选项值。

连接索引列必须具有相同的数据类型、精度和标度。

另请参见

- 解释通知消息 (第 473 页)
- 装载和插入消息 (第 243 页)
- 消息日志回卷 (第 10 页)
- 消息日志中的版本信息 (第 8 页)

查询解析中的连接层次

Sybase IQ 可使用相同连接索引解析涉及在此连接索引中指定的完全连接关系的查询，或涉及此关系的任何连续子集的查询。您不必为子集关系创建单独的连接索引。

例如，假定连接索引 **ABCDEF** 连接连接关系层次图中说明的表。**Sybase IQ** 可使用连接索引 **ABCDEF** 解析涉及下列各项的任何查询：

- 整体关系
- 表 **A** 至表 **D**
- 表 **A** 至表 **D** 至表 **F**
- 表 **B** 至表 **D**
- 表 **B** 至表 **D** 至表 **F**
- 表 **D** 至表 **F**
- 表 **C** 至表 **E**
- 表 **E** 至表 **F**
- 表 **C** 至表 **E** 至表 **F**

但是，**Sybase IQ** 不能使用连接索引 **ABCDEF** 解析针对表 **E** 至表 **D** 等的查询。

另请参见

- 连接层次概述 (第 213 页)

- 连接索引中的关系（第 212 页）
- 创建连接索引的步骤（第 216 页）

一对多关系

在一对多连接关系中，一个表中的一行可能与另一个表中的一行或多行匹配，并且第一个表中只能有一行与第二个表中的相同行匹配。

为此，第一个表中的连接列中的值必须唯一。

其中一个表可能不与另一个表匹配。这就构成了一个外连接。Sybase IQ 完全支持外连接。

如果连接列由多个列组成，“一”方中各个值的组合必须唯一。例如，在 `iqdemo` 数据库中，`Customers` 表中的 `ID` 以及 `SalesOrders` 表中的 `CustomerID` 各含一个客户 ID。`Customers` 表中针对每个客户都包含一行内容，因此，在每行的 `ID` 列中都有一个唯一的值。`SalesOrders` 表为客户执行的每笔交易各包含一行。有可能每个客户对应有多笔交易，因此 `SalesOrders` 表中的 `CustomerID` 列具有多个含有相同值的行。

如果将 `Customers.ID` 连接到 `SalesOrders.CustomerID`，则连接关系为一对多。如以下示例所示，对于 `Customers` 中的每一行，`SalesOrders` 可能具有多个行与之匹配。（示例输出已限制为 15 行。）

```
SELECT SalesOrders.ID, SalesOrders.CustomerID,
Customers.GivenName
from SalesOrders, Customers
where SalesOrders.CustomerID = Customers.ID
```

ID	CustomerID	GivenName
2001	101	Michaels
2005	101	Michaels
2125	101	Michaels
2206	101	Michaels
2279	101	Michaels
2295	101	Michaels
2337	101	Michaels
2389	101	Michaels
2447	101	Michaels
2560	101	Michaels
2583	101	Michaels
2002	101	Beth
2142	101	Beth
2318	101	Beth
2338	101	Beth

警告! 如果一对多关系错误，则无法同步连接，直到您删除“一”表中的多余行为止。如果尝试同步，则会收到“重复行”错误，并且事务将回退。

当创建连接索引时，您可以使用 `ANSI FULL OUTER` 连接语法。Sybase IQ 将索引存储为完全外连接。稍后，当针对连接索引中的列发出查询时，您可以指定内连接、左

外连接和右外连接关系以及完全外连接。 Sybase IQ 仅使用给定查询所需的部分连接索引。

多表连接和性能

多表连接的规则包括：

- 表只能在一对多关系的“一”方出现一次。例如，您不能在以下情况下创建连接索引或连接查询：表 A 以一对多关系连接到表 B，同时表 A 又以一对多关系连接到表 C。您需要为上述每一关系单独创建连接索引。
- 表只能在关系层次中出现一次。因此，举例来说，您不能预定义这样的连接关系查询：表 A 连接到表 B，表 B 连接到表 C，并且表 C 又连接到表 A。只能使用预定义连接分别查询表 A 至表 B 和表 C 至表 A 关系。为此，请为上述每一关系单独创建连接索引。
- 表可以连接到其它表或连接定义。例如，您可以创建用于连接表 A 和表 B 的连接索引，或用于连接表 C 和表 A 与表 B 的连接的连接索引。
- 在层次中的顶级表与层次中的其余部分的一对多关系中，顶级表是此关系的“多”方。
- 最有用的连接索引通常为两个表连接。

在某些情况下，您可能希望为连接关系的子集创建单独的连接索引。如果连接索引子集中的顶级表的行数远远小于完全连接索引中的顶级表的行数，当查询仅使用子集中的表时，查询子集的速度可能会比查询同一完全连接索引的速度更快。

当然，此方法需要更大的磁盘空间来创建额外连接索引，还需要更多的索引创建时间（还包括增加的维护）。对于子集连接索引，额外连接索引重复完全连接索引中已有的信息子集。对于这种特定连接关系，您必须确定哪方面更重要，是查询速度还是应用程序的磁盘空间使用。此外，请牢记在 Sybase IQ 的当前版本中，连接索引提供的性能优势可能不同于早期版本，特别是当关系层次包含多个级别时。

创建连接索引的步骤

为了创建连接索引，必须执行下列所有步骤。

1. 使用 **CREATE TABLE** 命令或 Sybase Central 创建连接索引中所涉及的表。这些表必须是永久表；您不能使用临时表创建连接索引。
2. 确定将任一连接中涉及的基础表中的特定列对关联起来的 *连接条件*。
为数据库定义 **schema** 非常重要，可以用来阐明连接条件和有关数据结构的其它假定。**schema** 应表示外键-主键关系，并遵循 **schema** 设计的其它最佳实践。通过外键关联的列必须具有匹配的数据类型、精度和标度。
对于关系基于键连接的情况，您必须在步骤 1 中的 **CREATE TABLE** 命令或 **ALTER TABLE** 命令中将连接条件定义为参照完整性约束（主键和外键声明）。
3. 为连接中涉及的每个列创建主键。
4. 为要连接的表创建列索引。

当 Sybase IQ 在各个表之间创建连接索引时，将在连接索引中使用已针对单个表定义的 IQ 列索引类型和数据类型。基表上的多列索引在使用那些基表创建的连接索引中不会进行复制。

5. 使用 **LOAD TABLE** 命令将数据装载到表中。您也可以使用 **INSERT INTO** 命令将数据添加到现有表中。

注意： 必须以单表插入方式插入到连接索引的每个表中，而不是插入到连接索引自身。此方法符合索引数据的 ANSI 规则。

6. 通过发出 **CREATE JOIN INDEX** 命令，或者在 Sybase Central 中使用创建连接索引向导来创建连接索引。指定连接层次以作为此步骤的一部分。
7. 根据您执行上述步骤的顺序，您可能需要同步连接索引中的表，如下所述。如果数据存在于连接表中，则会自动进行同步。

在完成所有步骤之前，索引保持不可用状态。但是，您可以根据站点需求调整某些步骤的顺序：

- 通过在创建表时定义关系，您可以组合步骤 1 和步骤 2。
- 您可以在创建连接索引之前或之后装载数据。如果您在创建连接索引之后将数据装载到基础列索引中，则必须执行同步步骤。

另请参见

- 连接层次概述（第 213 页）
- 连接索引中的关系（第 212 页）
- 查询解析中的连接层次（第 214 页）

创建连接索引所需的特权

您必须是表所有者或 DBA 才能创建、更改或同步含有该表的连接索引。如果您不是 DBA，则需要是表所有者并具有 RESOURCE 权限才能创建连接索引。

有关完整语法，请参见《参考：语句和选项》中的 **CREATE TABLE** 语句、**ALTER TABLE** 语句、**LOAD TABLE** 语句、**INSERT** 语句和 **SYNCHRONIZE JOIN INDEX** 语句。下列各节介绍了有关创建连接索引的详细信息。

另请参见

- 数据导入和导出（第 227 页）

同步连接索引

必须先同步连接索引表中的数据，然后才能使用连接索引。同步可确保数据按连接的正确顺序装载。

创建连接索引时，将自动进行同步。在完成装载或插入数据的事务之前进行同步也可使表对所有读取者立即可用。但是，数据一旦装载，则必须显式同步连接索引，以下情况除外：当更改连接层次的顶级表时，系统将自动同步连接索引。

要显式同步，请发出以下命令：

```
SYNCHRONIZE JOIN INDEX [join-index-name [, join-index-name]
```

如果省略索引名，Sybase IQ 将同步所有连接索引。

另请参见

- 在插入数据后添加列索引（第 211 页）

同步的性能提示

同步可能需要很长时间。

要提高性能，请尝试下列建议：

- 在非高峰时间调度同步。
- 单独同步连接索引，而不要一次同步所有连接索引。
- 在执行整组插入和删除操作后同步。最好不要在每次插入或删除操作后同步，因为更新连接索引的时间在很大程度上取决于对表的更新顺序。同步多组更新使 Sybase IQ 能够选择对连接索引应用表更改的最佳顺序。

定义表之间的连接关系

当创建连接索引时，必须指定连接中每一关联对之间的关系。关联对始终为两个表，但是，您也可以通过将某个表与其它连接关系相关联来指定关系。

您可以根据关系一次或两次指定该关系：

- **键连接**将一个表的主键和另一个表的外键关联起来。对于键连接，当使用 **CREATE TABLE** 或 **ALTER TABLE** 命令创建或更改基础表时，必须指定 **PRIMARY KEY** 和 **FOREIGN KEY**。
- 对于所有连接，使用 **CREATE JOIN INDEX** 命令创建连接索引时应指定关系。连接按照表的列出顺序、指定列或连接类型：键连接、自然连接或 **ON** 子句连接。

连接关系规则包括：

- 连接关系中的每对表必须至少具有一个连接列。
- 连接列必须存在于两个表中。
- 一对表可以具有多个连接列，条件是只要这些表具有相同列数，并且在指定连接列时在每个表列表中的位置相同。两个表的列表顺序确定列的匹配方式。

使用外部引用

Sybase IQ 使用外键定义将在连接索引中使用的各个列之间的关系，并优化查询。

注意，对于某些特定类型的连接索引，依赖于外键的键连接是必需的。

Sybase IQ 不支持基于多列外键的键连接索引。

表定义中的连接关系示例

以下示例显示如何通过主键和外键指定连接关系。

在本示例中，一个客户可具有多个销售订单，因此，Customers 表的 ID 列（其主键）和 SalesOrders 表的 CustomerID 列之间存在一对多关系。因此，请将 SalesOrders 中的 CustomerID 指定为 **FOREIGN KEY**，它引用 Customers 表的 ID 列。

第一个示例创建 Customers 表，该表将列 ID 用作其主键。为了简化本示例，此处使用省略号 (...) 表示其它列。

```
CREATE TABLE GROUPO.Customers ( ID INTEGER NOT NULL, ... PRIMARY KEY (ID),)
```

然后，创建含有六列的 SalesOrders 表，并指定 CustomerID 列作为主键。您还必须向 Customers 表的 ID 列添加与 SalesOrders 表的 CustomerID 列关联的外键。

您可以在创建表时添加外键，也可以在稍后添加。本示例通过在 **CREATE TABLE** 语句中包含 **REFERENCES** 子句作为列约束来添加外键。

```
CREATE TABLE GROUPO.MySalesOrders (ID INTEGER NOT NULL, CustomerID INTEGER REFERENCES GROUPO.Customers(ID), OrderDate DATE NOT NULL, FinancialCodesID CHAR(2), Region CHAR(7), SalesRep INTEGER NOT NULL, PRIMARY KEY (ID),)
```

或者，您也可以在不使用 **REFERENCES** 子句的情况下创建表，并在稍后添加外键，这与以下 **ALTER TABLE** 语句的操作结果相同。您可以发出下列两个语句之一，但不能同时发出这两个语句：

```
ALTER TABLE GROUPO.MySalesOrders
ADD FOREIGN KEY ky_so_customer (CustomerID)
REFERENCES GROUPO.Customers (ID)
```

另请参见

- 连接层次概述（第 213 页）
- 创建连接索引时指定连接类型（第 219 页）
- 创建连接索引时指定关系（第 220 页）
- 连接层次类型（第 221 页）

创建连接索引时指定连接类型

连接类型始终为 **FULL OUTER**，关键字 **OUTER** 为可选关键字。

您还需要执行以下操作之一：

- 如果要连接两个表中具有相同名称的等效列，请将该连接指定为 **NATURAL JOIN**。
- 如果要基于键连接列，还必须将基础表中的关系指定为引用 **PRIMARY KEY** 的 **FOREIGN KEY**。

- 如果要连接两个表中的列的等效值（*等值连接*），请指定 **ON** 子句。

上述规则符合 ANSI 语法要求。

另请参见

- 表定义中的连接关系示例（第 219 页）
- 连接层次概述（第 213 页）
- 创建连接索引时指定关系（第 220 页）
- 连接层次类型（第 221 页）

创建连接索引时指定关系

对于非键连接，创建连接索引时表的指定顺序确定各表之间的连接关系层次。

CREATE JOIN INDEX 语句支持两种指定连接层次的方法：

- 从层次中最低一级的表开始列出每个表，并说明每对表之间的连接关系。列表中的最后一个表将为层次中的顶级表。例如，在连接关系层次图中，**F** 为顶级表，**E** 位于顶级表之下，**C** 位于层次底部。可以按如下方式指定上述三个表的连接层次：

```
C FULL OUTER JOIN E FULL OUTER JOIN F
```

- 使用括号控制连接关系的计算顺序。括号对计算顺序的控制与在数学中相似，也就是说，将首先计算最内部的表对。采用此方法时，请将顶级表放到最外面的一组括号中，并以此作为开始，然后是任何中间级，接着将最低的两个级别放置到最内部的括号中。使用此方法，您可以按如下方式指定上面相同的三个表：

```
(F FULL OUTER JOIN (C FULL OUTER JOIN E))
```

请注意，最低一级的表首先显示在最内部的括号中，这与第一种方法中的情况相同。

注意： 您可以使用此处介绍的方法连接上述三个表，但是，为了创建连接关系层次图中显示的完整层次，您将需要使用键连接。

当创建连接索引时，日志中的消息可标识连接中的顶级表。例如，

```
[20691]: Join Index 'join_on_tabletable' created from the following
join relations: [20694]:      Table Name      Relationship
[20697]:
-----
[20696]: 1. join_on_table_a joined to 'join_on_table_b' One >> Many
[20692]: The ultimate/top table is join_on_table_b [20697]:
-----
```

另请参见

- 表定义中的连接关系示例（第 219 页）
- 连接层次概述（第 213 页）
- 创建连接索引时指定连接类型（第 219 页）
- 连接层次类型（第 221 页）

发出 **CREATE JOIN INDEX** 语句

以下示例说明如何发出 **CREATE JOIN INDEX** 语句。

示例 1: 键连接

本示例为 **SalesOrders** 表和 **Customers** 表之间的键连接创建连接索引。这是基于外键 **ky_so_customer** 的键连接, 此外键将 **SalesOrders** 的 **CustomerID** 列与 **Customers** 表的主键 **ID** 相关联。您可以为索引指定任何名称。本示例将其命名为 **ky_so_customer_join**, 以便标识键连接所基于的外键。

```
CREATE JOIN INDEX ky_so_customer_join
FOR GROUPO.Customers FULL OUTER JOIN GROUPO.SalesOrders
```

示例 2: **ON** 子句连接

本示例使用 **ON** 子句为上面相同的两个表创建连接索引。无论是否存在外键, 您都可以使用此语法。

```
CREATE JOIN INDEX customer_sales_order_join
FOR GROUPO.Customers FULL OUTER JOIN GROUPO.SalesOrders
ON Customers.ID=SalesOrders.CustomerID
```

示例 3: 自然连接

要创建自然连接, 连接列必须具有相同的名称。如果要为前一示例中的表创建自然连接, 则不会获得预期结果。以下命令不会将 **Customers** 的 **ID** 列连接到 **SalesOrders** 的 **CustomerID** 列, 而会连接两个表中的不同 **ID** 列, 这是不允许的:

```
CREATE JOIN INDEX customers_sales_order_join FOR GROUPO.Customers
NATURAL FULL OUTER JOIN GROUPO.SalesOrders
```

SalesOrders 和 **SalesOrderItems** 的 **id** 列之间的自然连接更有意义。在本示例中, 同名列应包含匹配值。基于上述两个表之间的自然连接创建连接索引的命令如下:

```
CREATE JOIN INDEX sales_order_so_items_join FOR GROUPO.SalesOrders
NATURAL FULL OUTER JOIN GROUPO.SalesOrderItems
```

在 **Sybase Central** 中创建连接索引

按照以下步骤操作可以在 **Sybase Central** 中创建连接索引。

要在 **Sybase Central** 中创建连接索引, 请参见《**Sybase IQ 简介**》>“创建索引和装载数据”>“创建连接索引”。

连接层次类型

Sybase IQ 支持两种不同类型的连接层次。

支持的层次为:

- 线性连接
- 星形连接

可以为线性连接和星形连接创建即席连接。连接索引是为与线性连接一起使用而设计的。

另请参见

- 表定义中的连接关系示例 (第 219 页)
- 连接层次概述 (第 213 页)
- 创建连接索引时指定连接类型 (第 219 页)
- 创建连接索引时指定关系 (第 220 页)

线性连接

可将线性连接看作是没有分支的树。

层次中的每个表都与该表上面的表相关联，直到达到顶级表为止。表 A、D 和 F 构成一个线性连接层次。表 C、E 和 F 构成另一个线性连接层次。

在线性连接中，每对表都表示一对多关系，在每对表中，较低一级的表为“一”方，较高一级的表为“多”方。线性连接层次可依赖于下列任一基础连接条件：键连接、自然连接或 ON 子句连接。

星形连接

您可以将星形连接视为具有多个分支的结构，其中每个分支都与中间的某个表直接关联。

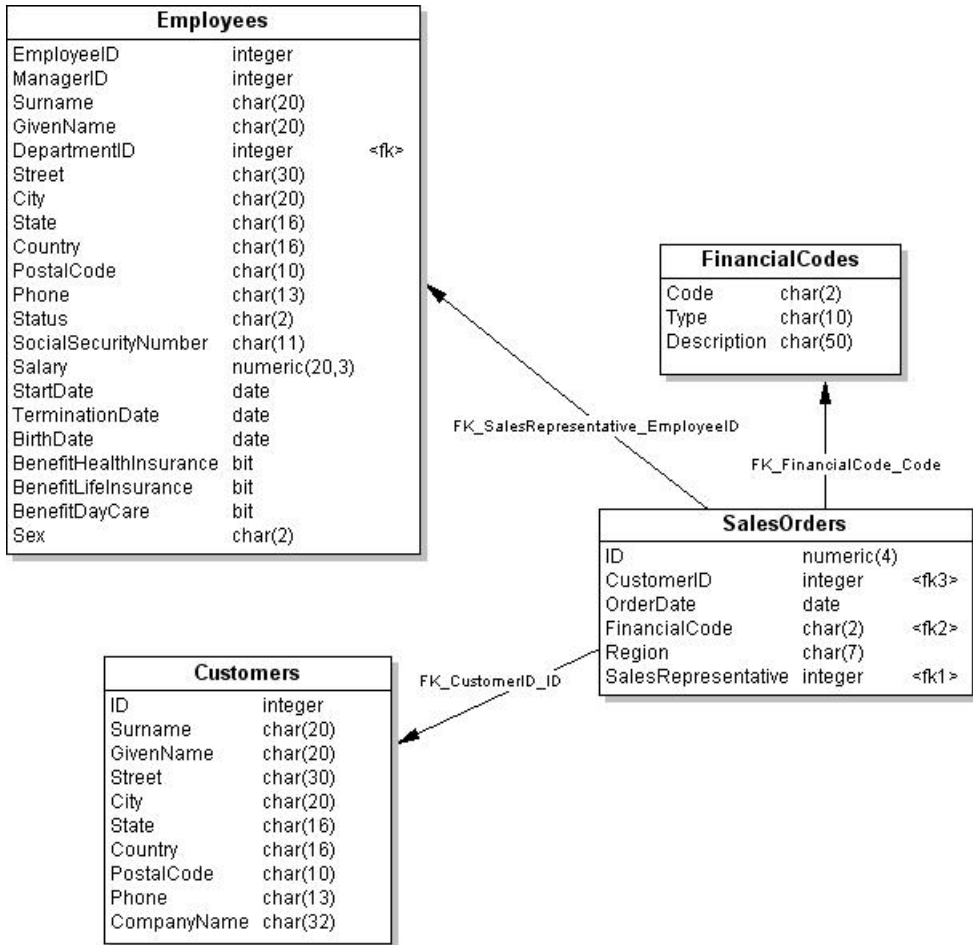
在连接关系层次图中，表 D、F 和 E 构成了一个非常简单的星形连接。更常见的是，表 F 位于多个表的中心，每个表都连接到表 F。

在星形连接中，一对多关系中的多个表与连接中心的一个表相关联。连接中心的一个表表示关系的“多”方，而该表周围的每个表表示关系的“一”方。“一”方的每个表都具有一组值，该组值具有其自己的唯一主键。关系“多”方中的表的外键将此表与关系“一”方中的表的主键相关联。

星形中心的“多”表有时称为*事实数据表*。与该表关联的“一”表称为*维度表*。

示例

在以下示例中，SalesOrders 表包含三个外键，每个外键都与另一个表的主键相关联。



可使用下列命令创建此表:

```
CREATE TABLE GROUPO.SalesOrders (
  ID NUMERIC (4) NOT NULL IQ
  UNIQUE (648),
  CustomerID INTEGER NOT NULL IQ UNIQUE (109),
  OrderDate date NOT NULL IQ UNIQUE (376),
  FinancialCode CHAR (2)
  NULL IQ UNIQUE (1),
  Region CHAR (7) NULL IQ UNIQUE (5),
  SalesRepresentative INTEGER NOT NULL IQ UNIQUE (11) );
COMMENT ON TABLE GROUPO.SalesOrders is 'sales orders that customers
have submitted to the sporting goods company';
ALTER TABLE GROUPO.SalesOrders ADD FOREIGN KEY FK_CustomerID_ID (CustomerID)
REFERENCES GROUPO.Customers (ID) ON DELETE RESTRICT ON UPDATE
RESTRICT;
ALTER TABLE GROUPO.SalesOrders ADD FOREIGN KEY
FK_FinancialCode_Code (FinancialCode) REFERENCES
GROUPO.FinancialCodes (Code) ON DELETE RESTRICT ON UPDATE RESTRICT;
ALTER TABLE GROUPO.SalesOrders ADD FOREIGN KEY
FK_SalesRepresentative_EmployeeID (SalesRepresentative) REFERENCES
GROUPO.Employees (EmployeeID) ON DELETE RESTRICT ON UPDATE RESTRICT;
```

如图所示，Salesorders 表位于星形连接的中心。该表中的每个外键列都可包含其引用的主键的多个实例。例如，如果输入：

```
SELECT SalesRepresentative FROM SalesOrders WHERE  
SalesRepresentative = 299
```

结果将显示 SalesRepresentative 列中带有 299 的 114 行。

不过，如果您输入：

```
SELECT EmployeeID FROM Employees WHERE EmployeeID = 299
```

结果则仅显示 EmployeeID 列中带有 299 的一行。

注意： 针对所有连接的查询优化主要依赖基础主键。它们不需要外键。然而，您可以从使用外键中受益。如果您对负载进行设置以检查主键-外键关系，Sybase IQ 将会强制使用外键。

Sybase IQ 不支持对任何给定连接使用多个连接键列的星形连接样式的连接索引。

对于真正意义上的星形连接（即，没有任何维度与其它维度共享连接键的星形连接），IQ 查询优化程序允许单个子句中最多有 24 个维度表。但是，由于处理查询所需的时间随维度数目以指数方式增加，因此，当接近此最大值时，性能将下降。

另请参见

- 实体和参照完整性（第 329 页）
- 创建外键（第 178 页）
- 创建外键（第 331 页）

对修改连接索引表的限制

一旦创建连接索引，您即受到对连接索引及其基础表和索引的更改类型的限制。

不能删除参与连接索引的任何表。类似地，也不能使用 **ALTER TABLE** 添加、删除或修改参与连接索引的列。在上述两种情况下，您都必须先删除连接索引。然后，才能删除参与连接索引的表或修改任何列。

您可以向参与连接索引的表添加列。但是，向这些列插入数据也存在一定限制，如下节所述。

您可以删除未涉及连接关系的列上的索引，并且您可以添加、删除或修改连接索引中的表的非连接列。但是，您不能删除连接列上的索引或该连接列自身。涉及预定义连接关系的列至少需要一个索引。很有必要在作为连接索引一部分的所有列上创建 **HG** 或 **LF** 索引。

Sybase IQ 在更改基表的同时会自动应用对连接索引的更改。对非连接列执行 **ALTER TABLE** 之后，您无需同步连接索引。

连接索引的其它 **ALTER TABLE** 限制包括下列各项：

- 您不能为使列满足或不满足 **NATURAL** 连接条件而对该列进行重命名。

- 不能添加将参与先前指定的 **NATURAL** 连接的列。
- 如果 **PRIMARY KEY/FOREIGN KEY** 关系与连接索引中正在使用的连接条件匹配，则不能删除该关系。
- 不能从参与连接条件的列中删除 **NOT NULL** 约束。
- 不能修改参与连接条件的列的数据类型。

另请参见

- 修改视图的准则（第 181 页）

在连接索引表中执行插入或删除操作

应始终在基础表而不是连接索引自身中插入、装载或删除。

当首次创建连接索引时，Sybase IQ 将自动同步连接表，而不管您先前是否已向该表装载数据。

如果在参与现有连接索引的表中插入或删除，必须显式同步连接索引，除非您更新的是连接层次中的顶级表。如果先插入行，随后又在执行同步前删除这些行，Sybase IQ 将优化同步以省略插入内容。

操作连接表中的数据所需的特权

创建连接索引时，您即已具有对该连接执行相应操作的所需特权。

但是，您必须先向组中的其他用户显式授予对基础“连接虚拟表”的权限，然后这些用户才能操作此连接中的表。必须授予对连接虚拟表的特权以及对参与连接的表的相应特权。

在授予对连接虚拟表的特权之前，您必须先确定其名称，该名称存储在系统表中。如果连接索引名为 `emp_location`，以下查询将返回连接虚拟表名：

```
select table_name from sys.systable      where table_id in (select
jvt_id from      sys.sysiqjoinindex    where
joinindex_name='emp_location')
```

使用此查询返回的连接虚拟表名 `jvt_name` 授予对该连接虚拟表的权限：

```
grant all on jvt_name to user_names
```

在授予对基础连接虚拟表的必要特权之后，组中的其他用户即可对该连接索引中的表执行相应操作，而不会收到权限错误。

表版本控制可控制对连接索引的访问

任何表在任意给定时间都只能供单个用户写入使用。

对于连接索引，这意味着，当某一用户正在更新连接索引中的任何表时，其他用户均不能更新该索引中的任何表。所有连接表将保持为不可用，直到提交第一个用户的事务并使用 **SYNCHRONIZE** 命令同步表为止。

当连接索引表正在使用时，其他用户将收到以下错误：

```
Cannot write to this table in current transaction. Another user has write mode access.
```

这些用户的当前事务不会写入到任何连接索引表；他们必须开始新的事务才能写入这些表。

另请参见

- 中间版本控制（第 15 页）
- 事务和版本控制（第 337 页）

连接索引的大小和优势

创建连接索引之前，请估计其大小和潜在优势。

如何估计连接索引大小

Sybase IQ 提供的存储过程 `sp_iqestjoin` 可帮助您估计连接索引的大小。

应对要连接的每对表运行此过程。每次运行该过程时，必须提供下列参数：

- 要连接的第一个表的名称
- 第一个表中的行数
- 要连接的第二个表的名称
- 第二个表中的行数
- 关系（缺省为一对多）
- IQ 页大小（缺省为 131072 字节或 128KB）

影响连接索引大小的因素有多种，特别是它包含的外连接数。为此，此过程为您提供了三种结果类型。如果您知道将始终完全按一对一匹配来连接表，请使用“`Min Case index_size`”。如果预测偶尔进行一对多连接，请使用“`Avg Case index_size`”。如果预测使用许多一对多连接，请使用“`Max Case index_size`”。

通过这些计算，您可以大致了解连接索引所需的磁盘空间大小。结果包含段大小（以字节为单位）和块数。此过程还可以告知创建连接索引将需要多少时间。

如果您希望知道现有连接索引的实际大小，请使用另一个存储过程 `sp_iqjoinindexsize`。

如何确定连接索引优势

当考虑创建连接索引时，请比较顶级表中的行数和关联表中的行数。

一般说来，如果顶级表中的行数与关联表中的行数的比率小于 10 比 1，对于大多数查询而言，连接索引的性能将提高。某些用户可能会发现，当比率高达 100 比 1 时，连接索引会很有用；而其他用户则可能发现，当比率低至 10 比 1 时，连接索引将有损于查询性能。如果考虑使用更高的比率，请进行测试并确保该比率有助于您的查询。

数据导入和导出

Sybase IQ 提供了几种将数据移入和移出数据库的方法。

您需要知道每种方法在何时使用以及如何解决从其它类型的数据库插入的数据的转换问题。

另请参见

- 创建连接索引所需的特权 (第 217 页)

导入和导出概述

使用 Sybase IQ，可以从平面文件导入数据或直接从数据库表导入数据。您也可以直接向数据库中输入指定的值。使用 **Interactive SQL** 实用程序和 IQ 数据抽取工具，可以将数据导出为其它格式。

Sybase IQ 表是逻辑表；它不包含数据。解析查询时所需的所有信息（包括数据）包含在 Sybase IQ 索引中。将数据插入到 IQ 表的列中时，不会实际将数据添加到表的列中，而是添加到列索引中。通过按表插入数据，可以创建索引。

导入和导出方法选择

Sybase IQ 提供了用于添加、更改或删除数据的方法选项。

- 要有效地从平面文件批量装载表，请使用 SQL 语句 **LOAD TABLE**。
- 要逐行地向表中插入指定的值，请将 SQL 语句 **INSERT** 与 **VALUES** 选项配合使用。
- 要从表插入所选择的行，请将 SQL 语句 **INSERT** 与 **SELECT** 语句子句配合使用。
- 要从驻留在其它数据库中的表中插入行，请将 SQL 语句 **INSERT** 与 **LOCATION** 语句子句配合使用。
- 要删除表中的特定行，请使用 **DELETE** 语句。**TRUNCATE** 语句将表初始化为 0 行。
- 要更改表中的现有行，还可以使用 **UPDATE** 语句。

IQ 数据抽取工具将数据以二进制或 ASCII 格式导出，这种格式的数据可以装载到另一数据库中。在进行大容量数据移动，或者需要可用于装载的输出文件时，可使用此功能。

利用 **Interactive SQL**，可以将数据以多种格式导出到另一数据库中，也可以生成文本文件作为输出。有关格式列表及如何选择格式，请参见下一节。您还可以重定向任何命令的输出。

注意： Sybase IQ 通过 **LOAD TABLE FORMAT BCP** 选项支持 BCP。Sybase IQ 还支持使用 **LOAD TABLE USING CLIENT FILE** 选项批量装载远程数据。请参见《参考：语句和选项》。

可以对 SQL Anywhere 表执行 BCP，然后将内容传输至 Sybase IQ；但是，行的传输却是一次一行地进行的。Sybase IQ 不支持 Open Client Bulk-Library (bklb) 例程，因此 BCP(它使用 Open Client Bulk-Library) 在批量装载模式下不起作用。(Bulk-Library/C 提供一些例程，能让 Client-Library 和 Server-Library 应用程序使用 Adaptive Server Enterprise 的批量复制接口来实现高速数据传输。) Sybase IQ 和 Adaptive Server Enterprise 中的 BCP 格式都支持在作为日期的一部分存在的单个数位之前放置前导空白。

另请参见

- 提取选项 (第 230 页)

输入和输出数据格式

LOAD TABLE 语句可从文件中逐行导入数据。ASCII 和二进制输入文件均受支持，固定长度字段或可变长度字段均以分隔符结尾。

INSERT 语句可将数据从指定的值集或直接从表移动到 Sybase IQ 表中。

有关 Interactive SQL 支持的输出文件格式，请参见《实用程序指南》>“dbisql”>“用于 Interactive SQL 的 SQL 语句”>“OUTPUT 语句 [Interactive SQL]”。

IQ 数据抽取工具将数据以二进制或 ASCII 格式导出。

另请参见

- 提取选项 (第 230 页)

指定 Interactive SQL 的输出格式

可以指定 Interactive SQL 的缺省输出格式。

1. 从 Interactive SQL 菜单栏中，选择“命令”>“选项”。
2. 选择输出格式。
3. 要将该格式设为缺省输出格式，请单击“永久”。

修改数据的权限

只有对要修改的数据库表具有适当的权限时，您才可以执行数据修改语句。

数据库管理员和数据库对象的所有者使用 GRANT 和 REVOKE 语句来决定哪些用户有权使用哪些数据修改功能。

要插入数据，您需要对该表或视图具有 INSERT 权限。要删除数据，您需要对该表或视图具有 DELETE 权限。要更新数据，您需要 UPDATE 权限。DBA 可以在任意表中插入或删除数据。表的所有者对表具有 INSERT、DELETE 和 UPDATE 权限。

可以将权限授予单个用户、组或公共组，也可以撤消其权限。

另请参见

- 管理用户 ID 和权限 (第 281 页)
- 安全概述 (第 4 页)
- 事务和版本控制 (第 337 页)

调度数据库更新

当一个用户向某数据库表中插入数据时，多个用户可以对该表进行查询。多个用户可同时更新数据库，但前提是他们在不同的表中插入或删除数据。

如果您允许在更新期间并发使用数据库，则性能会降低，并且磁盘使用率会增加。

另请参见

- 调优数据的 bulk load 操作 (第 276 页)

从数据库导出数据的方法

要从此版本的 Sybase IQ 中的数据库导出 IQ 数据，Sybase 建议您使用本指南中记录的方法。

您也可以使用由您或第三方编写的前端工具导出数据，该工具可根据需要有效查询 IQ 数据库并设置数据格式。

要从 Catalog 存储中导出（除系统表以外的）表，请使用本指南中的方法。

输出重定向

输出重定向可用于导出查询结果。

您可以通过在任意命令中的任意位置放置 **>#** 重定向符号来将命令的输出重定向到某文件或设备。重定向符号后面必须跟有文件名。（在命令文件中，文件名后跟用作语句终结符的分号。）文件会相对于 Interactive SQL 的启动目录而放置。

在以下示例中，输出重定向到文件 `empfile`：

```
SELECT *  
FROM Employees  
># empfile
```

请勿用引号将文件名引起来。

输出重定向对于 **SELECT** 语句最为有用。

在重定向符号中使用两个 **>** 字符而不是一个（例如，**>>#**）可将输出附加到指定文件中，而不是重新放置文件的内容。如果 **SELECT** 语句的输出开始于指定文件的开头并且输出格式支持标题，则输出中包括标题。

重定向输出和消息

>& 重定向符号可重定向命令的所有输出，包括显示的错误消息和统计信息。例如：

```
SELECT *  
FROM Employees  
>& empfile
```

请勿用引号将文件名引起来。

本示例将 **SELECT** 语句输出到文件 `empfile`，后跟 **SELECT** 语句的输出以及与此命令相关的一些统计信息。

`>&` 重定向方法用于获取在 **READ** 命令执行期间发生的情况的日志。每个命令的统计信息和错误都会随命令一起写入重定向的输出文件中。

空值输出

尽管提取数据的最常见原因是为了在其它软件产品中使用，但这些产品有时候在处理空值时可能会出现问題。

使用 **dbisql** 选项 **NULLS**，可以选择如何输出空值。或者，您可以使用 **IFNULL** 函数在存在空值时输出特定值。

有关设置 **dbisql** 选项的信息，请参见《参考：语句和选项》中的第 2 章“数据库选项”。

数据抽取工具

数据抽取工具是一组数据库选项，可大幅提高对大型结果集执行查询的性能。

与其它数据库选项一样，可将数据提取选项设置为临时的或永久的。通常情况下，会将这些选项设置为临时的。提取选项是为连接而设置的。

使用提取选项有两项优势：

- 支持二进制格式，这样，便可以将输出数据装载到同一个或不同的 **IQ** 数据库中。
- 输出内容较多的 **SELECT** 语句的运行速度对于 **ASCII** 输出最多可加快 4 倍，而对于二进制输出，最多可加快 9 倍。

提取选项

使用提取选项，您可以从标准界面中重定向 **SELECT** 语句的输出，以直接转至一个或多个磁盘文件或命名管道。

选项名称	允许值	缺省值
Temp_Extract_Append	ON 或 OFF	OFF
Temp_Extract_Binary	ON 或 OFF	OFF
Temp_Extract_Column_Delimiter	string	'\'
Temp_Extract_Directory	string	"
Temp_Extract_Name1	string	"

选项名称	允许值	缺省值
Temp_Extract_Name2	string	"
Temp_Extract_Name3	string	"
Temp_Extract_Name4	string	"
Temp_Extract_Name5	string	"
Temp_Extract_Name6	string	"
Temp_Extract_Name7	string	"
Temp_Extract_Name8	string	"
Temp_Extract_Null_As_Empty	ON 或 OFF	OFF
Temp_Extract_Null_As_Zero	ON 或 OFF	OFF
Temp_Extract_Quote	string	"
Temp_Extract_Quotes	ON 或 OFF	OFF
Temp_Extract_Quotes_All	ON 或 OFF	OFF
Temp_Extract_Row_Delimiter	string	"
Temp_Extract_Size1	因平台而异*	0
Temp_Extract_Size2	因平台而异*	0
Temp_Extract_Size3	因平台而异*	0
Temp_Extract_Size4	因平台而异*	0
Temp_Extract_Size5	因平台而异*	0
Temp_Extract_Size6	因平台而异*	0
Temp_Extract_Size7	因平台而异*	0
Temp_Extract_Size8	因平台而异*	0
Temp_Extract_Swap	ON 或 OFF	OFF

*Temp_Extract_Size*n* 选项的缺省值因平台而异：

- AIX 和 HP-UX： 0 - 64GB
- Sun Solaris： 0 - 512GB
- Windows： 0 - 128GB
- Linux： 0 - 512GB

如果大型文件系统（如 JFS2）所支持的文件大小大于缺省值，请将 TEMP_EXTRACT_SIZE_n 设置为该文件系统允许的值。例如，要支持 1 TB 设置选项：

```
SET OPTION TEMP_EXTRACT_SIZE1 = 1073741824 KB
```

注意： 对于所有接受整数值的数据库选项，Sybase IQ 会截去 *option-value* 设置的小数部分，只保留整数值。例如，值 3.8 将被截断为 3。

在这些选项中，最重要的选项就是 TEMP_EXTRACT_NAME1。如果 TEMP_EXTRACT_NAME1 设置为其缺省设置（空字符串），则将禁用提取并且不重定向输出。要启用提取，请将 Temp_Extract_Name1 设置为可能的路径名。提取功能即开始将相应数据提取至具有该名称的文件中。一定要选择未在使用的文件的路径名。如果该文件尚不存在，则数据抽取工具会创建该文件。

包含指定文件的目录或文件夹以及该指定文件都必须拥有针对用于启动 IQ 的用户名（例如，*sybase*）而设置的写入权限。在附加模式中，数据抽取工具向文件末尾添加提取的行并且不会覆盖文件中的已有数据。如果该文件不存在，则数据抽取工具会创建该文件。

警告！ 如果选择一个现有文件的路径名，并将 TEMP_EXTRACT_APPEND 选项设置为 OFF（缺省值），则该文件的内容将被覆盖。例如，如果该文件是每周报告，则您可能希望覆盖该文件的内容，但如果该文件是某个数据库文件，则您可能不希望覆盖。

除了 TEMP_EXTRACT_NAME1 之外，还可以使用 TEMP_EXTRACT_NAME2 到 TEMP_EXTRACT_NAME8 选项指定多个输出文件的名称。这些选项必须按顺序使用。例如，除非事先已设置 TEMP_EXTRACT_NAME1 和 TEMP_EXTRACT_NAME2 选项，否则 TEMP_EXTRACT_NAME3 将无效。

选项 TEMP_EXTRACT_SIZE1 至 TEMP_EXTRACT_SIZE8 用于指定相应的输出文件的最大大小。TEMP_EXTRACT_SIZE1 指定 TEMP_EXTRACT_NAME1 所指定的输出文件的最大大小，TEMP_EXTRACT_SIZE2 指定 TEMP_EXTRACT_NAME2 所指定的输出文件的最大大小，依此类推。

请注意，数据提取大小选项的缺省值为 0。IQ 可将此缺省值转换为以下各值：

设备类型	Size
磁盘文件	AIX 和 HP-UX: 0 - 64GB Sun Solaris 和 Linux: 0 - 512GB Windows: 0 - 128GB
磁带*	524288 KB (0.5 GB)
其它	无限制

*目前不支持磁带设备。

TEMP_EXTRACT_APPEND 与 TEMP_EXTRACT_SIZE_n 选项不兼容。如果您尝试限制提取追加输出文件的大小，Sybase IQ 将报告错误。

如果要提取到单个磁盘文件或单个命名管道，请将选项 TEMP_EXTRACT_NAME₂ 到 TEMP_EXTRACT_NAME₈ 以及 TEMP_EXTRACT_SIZE₁ 到 TEMP_EXTRACT_SIZE₈ 保留为缺省值。

注意： 如果 **SELECT** 不返回行并且没有要重定向的输出，则会创建长度为零的空文件。如果已指定多个提取文件，但却没有足够的数据来填充所有这些文件，则这些文件仍会被创建。

控制访问

TEMP_EXTRACT_DIRECTORY 选项控制是否允许用户使用数据抽取工具。它还控制临时提取文件的放置目录，并覆盖在 TEMP_EXTRACT_NAME_n 选项中指定的目录路径。

如果针对某一用户将 TEMP_EXTRACT_DIRECTORY 选项设置为字符串 FORBIDDEN (不区分大小写)，则禁止该用户执行数据提取。该用户尝试使用数据抽取工具将产生错误：“您无权执行提取”。

如果针对 PUBLIC 组的 TEMP_EXTRACT_DIRECTORY 设置为 FORBIDDEN，则任何人均不能运行数据提取。

如果将 TEMP_EXTRACT_DIRECTORY 设置为有效目录路径，则将临时提取文件放置到该目录中，并覆盖在 TEMP_EXTRACT_NAME_n 选项中指定的路径。

如果将 TEMP_EXTRACT_DIRECTORY 设置为无效目录路径，则会发生错误：“文件不存在: <无效路径>”

如果 TEMP_EXTRACT_DIRECTORY 为空，则根据在 TEMP_EXTRACT_NAME_n 中的相应指定，将临时提取文件放置到目录中。如果 TEMP_EXTRACT_NAME_n 中未指定任何路径，缺省情况下，提取文件将放置到服务器启动目录中。

TEMP_EXTRACT_DIRECTORY 选项可限制在用户对其具有写入访问权限的目录中创建大型数据提取文件的操作，从而增强了安全性并有助于控制磁盘管理。要求具有 DBA 权限才能设置此选项。

提取类型

数据提取有三种类型：

- 二进制
- 二进制/交换
- ASCII

二进制提取生成的文件可通过 **LOAD TABLE** 语句以总体“二进制”格式以及按列以“含空字节的二进制”格式装载。

二进制/交换提取与二进制提取相同，只不过前者是使用相反的端模式在另一计算机上装载。

ASCII 提取可生成文本文件。

Temp_Extract_Binary 和 Temp_Extract_Swap 这两个选项可确定执行三种提取类型中的哪一种：

类型	Temp_Extract_Binary	Temp_Extract_Swap
binary	ON	OFF
二进制/交换	ON	ON
ASCII	OFF	OFF

缺省提取类型为 ASCII。

请注意，如果使用数据抽取工具卸载数据时将 TEMP_EXTRACT_BINARY 选项设置为 ON，则在装载二进制数据时，必须针对每个列使用 LOAD TABLE 语句的 BINARY WITH NULL BYTE 参数。

列和行分隔符

就 ASCII 提取而言，缺省情况下使用逗号分隔列值，并以换行符作为行结尾（在 UNIX 平台中）或以回车/换行符对作为行结尾（在 Windows 平台中）。字符串不带引号。如果这些缺省值不适用，请使用下列选项更改分隔符：

- Temp_Extract_Column_Delimiter
- Temp_Extract_Row_Delimiter
- Temp_Extract_Quote
- Temp_Extract_Quotes
- Temp_Extract_Quotes_All

分隔符必须占 1 至 4（最大值）字节。如果您使用的是多字节归类顺序，则分隔符必须在当前使用的归类顺序中是有效的。请务必选择未在任何数据输出字符串本身中出现的分隔符。

请注意，Temp_Extract_Row_Delimiter 选项的缺省值为空字符串。IQ 将此选项的空字符串缺省值转换为换行符（在 UNIX 平台中）或回车符/换行符对（在 Windows 平台中）。

选项 Temp_Extract_Column_Delimiter 可控制各列之间的分隔符。对于 ASCII 提取，如果将此选项设置为空字符串，则将用不带列分隔符的固定宽度 ASCII 写入提取的数据。数值数据类型和二进制数据类型在具有 *n* 个空白的字段上右对齐，其中 *n* 为该类型的任何值所需的最大字节数。字符数据类型在具有 *n* 个空白的字段上左对齐。

注意：固定宽度 ASCII 提取的最小列宽是四字节，允许使用字符串“NULL”表示空值。例如，如果所提取的列为 CHAR(2) 并且 Temp_Extract_Column_Delimiter 设置为空字符串，则提取的数据后面会有两个空格。

在 ASCII 提取期间，下列选项可控制引号的使用：

选项	ASCII 提取操作
Temp_Extract_Quotes	将字符串字段用引号引起来
Temp_Extract_Quotes_All	将所有字段均用引号引起来
Temp_Extract_Quote	指定要用作引号的字符串

在 `Temp_Extract_Quote` 选项中指定的引号字符串与分隔符具有相同的限制。此选项的缺省值为空字符串，IQ 可将其转换为单引号。

空值的表示形式

`Temp_Extract_Null_As_Zero` 和 `Temp_Extract_Null_As_Empty` 选项控制要用于 ASCII 提取的空值的表示形式。当 `Temp_Extract_Null_As_Zero` 选项设置为 ON 时，空值的表示形式如下：

- '0'，表示算术类型
- " (空字符串)，表示 CHAR 和 VARCHAR 字符类型
- " (空字符串)，表示日期
- " (空字符串)，表示时间
- " (空字符串)，表示时间戳

当 `Temp_Extract_Null_As_Empty` 选项设置为 ON 时，对于所有数据类型，空值的表示形式均为 " (空字符串)。

请注意，提取输出文件中不包含上述引号。当 `Temp_Extract_Null_As_Zero` 和 `Temp_Extract_Null_As_Empty` 选项设置为 OFF 时，在所有情况下均使用字符串“NULL”表示空值。OFF 为缺省值。

如果 `Temp_Extract_Null_As_Zero` 设置为 ON，则 ASCII 提取向文件的 CHAR 或 VARCHAR 列所写入的字符数等于列中的字符数，即使该数字小于四也是如此。在早期版本中，Sybase IQ 始终返回至少四个字符来容纳单词 NULL。

消息记录

当 `Query_Plan` 选项设置为 ON 时，所提取列的列表（带时间戳）会显示在 IQ 消息日志中。

另请参见

- 导入和导出方法选择（第 227 页）
- 输入和输出数据格式（第 228 页）

启用数据提取选项

请谨慎使用数据提取选项。

警告！ 如果设置提取选项，则执行一条 **SELECT** 语句，然后在不更改提取文件名的情况下执行第二条 **SELECT** 语句，第二条 **SELECT** 语句的输出会覆盖第一条 **SELECT**

语句的输出。除非 `Temp_Extract_Append` 选项设置为 `ON`，否则每当执行 `SELECT` 语句时，无论是一秒钟后还是一周后，提取都会重新开始。

此外，还要注意为连接设置的提取选项。如果您设置了提取选项而另一用户也使用同一连接来连接到数据库，则该用户也可使用该提取功能。您的提取输出可能会被另一使用同一连接的用户覆盖。

同样，如果另一用户使用同一用户 `ID` 登录，则该用户运行的查询的输出会定向到提取文件，直到该选项禁用。如果您正在使用提取，请确保使用唯一用户 `ID` 运行您的请求。

-
1. 将需要保留的任何旧输出提取保存在单独的位置。
 2. 删除以前使用的任何提取文件。
 3. 设置所需的提取选项，确保将 `Temp_Extract_Name1` 设置为要接收输出的文件路径。
 4. 发出 `SELECT` 语句以提取所需的数据。
 5. 完成提取后，将 `Temp_Extract_Name1` 重置为空字符串，或者如果是临时设置的，则断开连接。

数据提取选项示例

这些示例描述了各种数据提取情况。

示例：提取到单个磁盘文件

下面的语句提取到单个磁盘文件 `daily_report.txt`：

```
SET TEMPORARY OPTION Temp_Extract_Name1 = 'daily_report.txt';  
SET TEMPORARY OPTION Temp_Extract_Name2 = '';  
SELECT ....;  
SET TEMPORARY OPTION Temp_Extract_Name1 = '';
```

请注意，在执行 `SELECT` 语句前，`Temp_Extract_Name2` 设置为空字符串以将输出限制到单个文件。

另请注意，在 `SELECT` 语句后，`Temp_Extract_Name1` 设置为空字符串以禁用提取。如果未禁用提取，则所执行的下一条 `SELECT` 语句会覆盖 `daily_report.txt` 文件。

示例：在附加模式下提取

在此示例中，已创建磁盘输出文件 `hourly_report.txt`，该文件为用户 `sybase` 设置了写入权限。下面的语句提取到 `hourly_report.txt`，并将每条 `SELECT` 语句的输出附加到文件的末尾：

```
SET TEMPORARY OPTION Temp_Extract_Append = ON; SET TEMPORARY OPTION  
Temp_Extract_Name1 = 'hourly_report.txt'; SET TEMPORARY OPTION  
Temp_Extract_Name2 = ''; SELECT ....; SELECT ....; SELECT ....; SET  
TEMPORARY OPTION Temp_Extract_Name1 = '';
```

这三条 **SELECT** 语句的所有输出均会写入文件 `hourly_report.txt`。在最后一条 **SELECT** 语句后，`Temp_Extract_Name1` 设置为空字符串以禁用提取。如果未禁用提取，则所执行的下一条 **SELECT** 语句的输出也会添加到文件 `hourly_report.txt` 的末尾。

示例：提取到多个磁盘文件

下面的语句提取到磁盘文件 `file1.out`、`file2.out` 和 `file3.out`。

首先设置文件名选项：

```
SET TEMPORARY OPTION Temp_Extract_Name1 = 'file1.out'; SET TEMPORARY
OPTION Temp_Extract_Name2 = 'file2.out'; SET TEMPORARY OPTION
Temp_Extract_Name3 = 'file3.out'; SET TEMPORARY OPTION
Temp_Extract_Name4 = '';
```

现在，通过设置相应的提取大小选项来限制文件的大小（例如，将每个文件的大小限制为 1MB）：

```
SET TEMPORARY OPTION Temp_Extract_Size1 = '1024'; SET TEMPORARY
OPTION Temp_Extract_Size2 = '1024'; SET TEMPORARY OPTION
Temp_Extract_Size3 = '1024';
```

大小选项以 KB（1024 字节）为单位。

使用这些设置时，提取输出首先写入 `file1.out`。如果要写入 `file1.out` 的下一行将会导致文件大小超过 1MB，则输出会重定向到 `file2.out`。如果 `file2.out` 已满（再将一行写入 `file2.out` 将会导致文件大小超过 1MB），则输出会重定向到 `file3.out`。如果在 IQ 提取所有行之前，`file3.out` 的大小超过 1MB，则系统会报告错误。

提取限制

这些约束和限制影响数据提取工具。

- 只能对 IQ 存储中存储的数据执行提取操作。
- 对系统表或跨数据库连接无法进行提取。
- 对于使用用户定义的函数或除 `suser_id()` 和 `suser_name()` 以外的系统函数的查询，不能使用提取功能。
- 二进制 **LOAD TABLE** 始终会剪裁 `VARCHAR` 数据中的空白。如果您有包含尾随空白的 `VARCHAR` 数据，则通过二进制装载来插入时，系统不会保留这些尾随空白。
- 在提取期间，尾随零会填补到 `VARBINARY` 数据中。例如，在提取期间，将用零填补一个声明为 `varbinary(6)` 且包含数据 `0x1234` 的字段，提取后，会将该字段作为 `0x123400` 进行装载。
- 在未来版本中，二进制格式将会改变。
- 如果您需要精确地重新生成浮点数据，请使用二进制选项。
- 此时不支持磁带设备。
- 如果运行带有 `-q`（安静模式）选项的 `dbisql`（Interactive SQL），并且数据提取命令位于命令文件中，则必须先设置 `dbisql` “显示多个结果集”选项并使该选项永久有效。如果未设置此选项，则系统不会创建输出文件。

要设置“显示多个结果集”选项，请单击 **dbisql** 窗口中的“工具” > “选项”，然后选中“显示多个结果集”框并单击“设为永久”。

另请注意，如果设置了 `Temp_Extract_Name1`，则不能执行下列操作：

- 针对连接中顶层表的 **LOAD**、**DELETE**、**INSERT** 或 **INSERT...LOCATION** 操作
- **SYNCHRONIZE JOIN INDEX**（显式发出或作为 **CREATE JOIN INDEX** 的一部分执行）
- **INSERT...SELECT**

事件不支持执行将返回结果集的语句。服务器日志会返回类似下面的错误：

```
Handler for event 'test_ev' caused SQLSTATE '09W03' Result set not permitted in 'test_ev'
```

为了通过事件执行查询，请创建一个调用存储过程的事件，然后将存储过程结果插入到临时表中。如果使用提取，则临时表始终为空并且需要的开销很少。

例如：

```
CREATE PROCEDURE procl() BEGIN      SET TEMPORARY OPTION
temp_extract_name1 = 'testproc.out';      SELECT * FROM iq_table;
END; CREATE EVENT "test_ev" ENABLE HANDLER BEGIN      SELECT * INTO
#tmp FROM procl(); END; TRIGGER EVENT test_ev;
```

使用 **LOAD TABLE** 语句执行 **bulk load** 操作

LOAD TABLE 语句有效地将数据从文本或二进制文件导入到现有的数据库表中。

该语句将数据装载到自动创建或由用户定义的列索引中。

执行 **LOAD TABLE** 语句所需的权限是在服务器命令行上使用 **-gl** 选项设置的。Sybase 建议使用 **-gl all** 设置，利用该设置，任何用户都可以装载或卸载表。这是 **start iq** 设置的缺省设置。如果设置了 **-gl all**，则您必须是表的所有者、拥有 **DBA** 权限或拥有 **ALTER** 权限，才能使用 **LOAD TABLE** 语句。您还需要对表具有 **INSERT** 权限。

使用命令文件装载数据

为了装载大量数据，大多数用户会创建命令文件。

*事务处理和 **LOAD TABLE***

当您对 **IQ** 表发出 **LOAD TABLE** 语句时，保存点会在数据装载前自动出现。

如果装载成功完成，则 Sybase **IQ** 会释放该保存点。如果装载失败，则事务会回退到该保存点。使用此方法，可以灵活地提交事务。例如，如果您发出两条 **LOAD TABLE** 命令，则可以确保两条命令均提交或均不提交。

当您对 **Catalog** 存储表发出 **LOAD TABLE** 时，没有自动保存点。如果装载成功，则它会自动提交。如果装载失败，则它会回退。您无法回退成功装载的 **Catalog** 存储表。

从平面文件装载：UNIX 示例

下面的语句将数据从文件 `dept.txt` 装载到 `department` 表的所有列中。

此示例假设不需要显式数据转换，并且输入列的宽度与 `Departments` 表中列的宽度匹配。平面文件 `dept.txt` 必须存在于指定位置中。

```
LOAD TABLE Departments
( DepartmentID, DepartmentName, DepartmentHeadID )
FROM '/d1/MILL1/dept.txt'
```

装载的文件规范要求

在 **FROM** 子句中，使用 *filename-string* 指定文件，并用逗号分隔多个字符串。

一次读取一个文件，并按 **FROM** 子句中指定的从左向右的顺序处理这些文件。任何 **SKIP** 或 **LIMIT** 值都只在装载开始时应用，而不是针对每个文件应用。

如果装载无法完成（例如，由于内存不足），则整个装载事务将会回退。

filename-string 以字符串的形式传递到服务器。因此，该字符串遵循的格式要求与其它 SQL 字符串相同。需要特别指出的是：

- 如果反斜杠 (\) 位于字符 `n`、`x` 或 `\` 前面，则会将它视为转义字符。因此，为了表示 Windows 系统中的目录路径，必须用两个反斜杠表示反斜杠字符（如果下一个字符是上面列出的任意一个）。（使用两个反斜杠总是很安全。）

若要将从文件 `c:\newinput.dat` 装载到 `employee` 表中，请输入以下命令：

```
LOAD TABLE employees FROM 'c:\\newinput.dat' ...
```

- 对于服务器端装载 (**LOAD TABLE ... USING FILE**)，路径名相对于数据库服务器，而不是客户端应用程序。如果在另一台计算机上的数据库服务器上运行此语句，则目录名是指服务器计算机上的目录，而不是指客户端计算机上的目录。要装载的输入文件必须位于该服务器计算机上。
- 对于客户端数据装载 (**LOAD TABLE ... USING CLIENT FILE**)，路径名必须相对于客户端应用程序。目录名是指客户端计算机上的目录。

指定命名管道的装载

从 Windows 上的命名管道（或 FIFO）装载时，向管道写入数据的程序必须以特殊方式关闭该管道。

管道写入程序必须先调用 `FlushFileBuffers()`，然后调用

`DisconnectNamedPipe()`。（否则，Sybase IQ 会从 `hos_io::Read()` 报告异常。）这会发出 `PIPE_NOT_CONNECTED` 错误，以通知 Sybase IQ 管道已正常关闭，而不是在无法控制的情况下断开连接。有关这些调用的详细信息，请参见 Microsoft 文档。

另请参见

- 直接从 Adaptive Server Enterprise 插入数据（第 252 页）
- 装载时间环境调整（第 278 页）
- 调查锁争用的工具（第 350 页）

指定输入数据格式的装载

可以指定大量装载选项，这些选项告诉 Sybase IQ 如何解释和处理输入文件，以及发生错误时如何处理。

您可以按任意顺序指定装载选项。有关所有选项的详细信息，请参见《参考：语句和选项》中的 **LOAD TABLE** 语句。

示例：显示双引号的装载
 请看定义如下的表：

```
CREATE TABLE t1 (c1 INT, c2 VARCHAR(20), c3 VARCHAR(20))
```

该表是使用以下输入数据定义的：

```
1, apple , fruit1      , 2, "banana" , "fruit2", 3, " pear ", " fruit3
",
```

通过运行下面的查询，可显示装载这些数据的结果：

```
SELECT c1, c2, c3, LENGTH(c2), LENGTH(c3) FROM t1
```

利用在执行 **LOAD TABLE** 命令期间所使用的 **QUOTES** 和 **STRIP** 选项的值，下表显示上述查询的结果，并将每个结果用 “<” 和 “>” 括起来：

LOAD TABLE 选项		SELECT c1, c2, c3, LENGTH(c2), LENGTH(c3) FROM t1 的结果				
QUOTES	STRIP	c1	c2	c3	length(c2)	length(c3)
ON	RTRIM	<1>	<apple>	<fruit1>	<5>	<6>
		<2>	<banana>	<fruit2>	<6>	<6>
		<3>	< pear >	< fruit3 >	<6>	<8>
ON	OFF	<1>	<apple>	<fruit1>	<6>	<7>
		<2>	<banana>	<fruit2>	<6>	<6>
		<3>	< pear >	< fruit3 >	<6>	<8>
OFF	RTRIM	<1>	<apple>	<fruit1>	<6>	<7>
		<2>	< “banana” >	< “fruit2” >	<9>	<9>
		<3>	< “ pear ” >	< “ fruit3 ” >	<9>	<11>
OFF	OFF	<1>	<apple>	<fruit1>	<7>	<8>
		<2>	< “banana” >	< “fruit2” >	<10>	<9>
		<3>	< “ pear ” >	< “ fruit3 ” >	<9>	<11>

结果注释:

- 使用 **QUOTES ON** 和 **STRIP RTRIM**，系统会将不带引号的字段 c2 行 1 的前导空格和尾随空格剪裁掉。
- 使用 **QUOTES ON** 和 **STRIP OFF**，系统仅会将不带引号的字段 c2 行 1 的前导空格剪裁掉。
- 使用 **QUOTES OFF** 和 **STRIP RTRIM**，系统仅会将不带引号的字段 c2 行 1 的尾随空格剪裁掉。
- 使用 **QUOTES OFF** 和 **STRIP OFF**，系统不会将不带引号的字段 c2 行 1 的前导空格和尾随空格剪裁掉。
- 使用 **QUOTES ON** 和 **STRIP RTRIM**，系统不会将带引号字段 c2 和 c3 行 3 的引号中的前导空格和尾随空格剪裁掉。

示例: 跳过指定字段的装载

以下是一个 Windows 示例:

```
LOAD TABLE nn (l_orderkey, l_quantity ASCII(PREFIX 2),
FILLER(2), FROM 'C:\\iq\\archive\\mill.txt' BYTE ORDER LOW
```

示例: 限制插入行数的装载

在下面的 Windows 示例中，不会跳过任何行，并且最多插入 1,000,000 行。

```
LOAD TABLE lineitem (l_shipmode ASCII(15), l_quantity
ASCII(8), FILLER(30)) FROM 'C:\\iq\\archive\\mill.txt' PREVIEW ON
LIMIT 1000000
```

示例: 包含制表符和换行符的装载

以下 Windows 示例将 l_orderkey 列的列分隔符设置为制表符，并将行分隔符设置为换行符 (\x0a) 后跟回车 (\x0d):

```
LOAD TABLE mm (l_orderkey '\x09', l_quantity ASCII(4),
FILLER(6), l_shipdate DATE('YYYY/MM/DD')) FROM 'C:\\iq\\archive\\
mill.txt' ROW DELIMITED BY '\x0a\x0d'
```

示例: 跳过行的装载

在此 UNIX 示例中，Sybase IQ 从输入文件读取 9,000 行，跳过前 5,000 行，装载接下来的 4,000 行。如果输入文件中只有 8,000 行，则只装载 3,000 行。

```
LOAD TABLE lineitem( l_shipmode ASCII(15), l_quantity
ASCII(8), FILLER(30)) FROM '/dl/MILL1/tt.t' LIMIT 4000 SKIP 5000
PREVIEW ON
```

LOAD TABLE 添加行

LOAD TABLE 语句将文件的内容附加到表中现有的行；它不替换表中现有的行，但如果指定 **START ROW ID** 装载选项则例外。

要清空现有表，请使用 **TRUNCATE TABLE** 语句删除所有行。

另请参见

- 装载转换选项 (第 258 页)

从客户端直接装载数据

Sybase IQ 支持通过 **LOAD TABLE USING CLIENT FILE** 语句对远程数据执行 bulk load 操作。**LOAD TABLE USING FILE** 在本地服务器上装载数据，并替换不建议使用的实用程序 **iq_bcp**。

服务器和客户端必须是 Sybase IQ 15.0 版或更高版本。组合使用 15.0 或更高版本的服务器和 12.7 客户端将返回“文件 I/O 错误”。

有关语法，请参见《参考：语句和选项》的 **LOAD TABLE** 语句中的 **USING** 子句。

分区表装载注意事项

针对装载分区表，有以下几个特殊注意事项：

- 当修改分区表时，如果分区列是命令的列列表中的第一个列，则可以获得最佳性能。

在 **INSERT...LOCATION** 语句的 **SELECT** 语句子句的大对象 (LOB) 列之前列出分区列，并装载主文件中的数据。如有可能，应使用预装载进程重新排列主文件中的数据。

- 不允许对分区表使用 **LOAD TABLE** 和 **INSERT** 语句的 **START ROW ID** 子句。系统将报告以下错误，并对装载操作执行回退：

```
"Option START ROW ID not allowed on a partitioned table."
```

```
(SQLCODE -1009416L, SQLSTATE QCB14, Sybase error code 21054)
```

- APPEND_LOAD** 数据库选项对分区表和未分区表具有不同的行为。每个分区都指派有行 ID 范围。

对于分区表，当 **APPEND_LOAD** 为 **ON** 时，新行将附加到相应分区的末尾。如果 **APPEND_LOAD** 为 **OFF**，则装载将重新使用相应分区已删除的行中的第一个可用行 ID 和空格。

对于未分区的表，当 **APPEND_LOAD** 为 **ON** 时，新行将添加到位于表行末尾的最大行 ID 后面。当 **APPEND_LOAD** 为 **OFF** 时，装载操作将重新使用已删除行的 ID。还可以使用 **LOAD** 或 **INSERT START ROW ID** 子句指定插入的开始行。

- 尝试更新分区列结果的内容将导致以下错误：

```
"Updating partition key column on a partitioned table is not allowed."
```

```
(SQLCODE -1009417L, SQLSTATE QCB15, Sybase error code 21055)
```

表分区是单独许可的 Sybase IQ VLDB 管理选项的一部分。

另请参见

- 对分区执行 DDL 操作的限制（第 135 页）

装载和插入消息

可以使用数据库选项和服务器启动开关来控制插入和装载消息。

消息可能在插入和装载操作期间显示。`NOTIFY_MODULUS` 数据库选项调整装载期间通知消息的缺省频率，或忽略这些消息。`LOAD` 命令中的 `NOTIFY` 选项会覆盖 `NOTIFY_MODULUS` 设置。请参见《参考：语句和选项》。

`IQMsgMaxSize` 服务器属性和 `-iqmsgsz` 服务器启动开关控制消息日志回卷和消息日志文件的大小。

另请参见

- 连接索引中的列（第 214 页）
- 解释通知消息（第 473 页）
- 消息日志回卷（第 10 页）
- 消息日志中的版本信息（第 8 页）

完整性约束违规消息

使用 `LOAD TABLE` 可以控制违反完整性约束时的装载行为并且有选择地记录有关违规的信息。

将 `MESSAGE LOG ... ROW LOG` 选项与 `ONLY LOG` 子句一起使用，您可以指示装载将有关特定类型的完整性约束违规的信息按违规记录在消息日志文件中，并按行记录在行日志文件中。如果未指定 `ONLY LOG` 子句，则只会将指示开始和完成装载的时间戳记录在这些文件中。

请注意，完整性约束违规的消息日志文件和行文件与 `IQ` 消息日志文件 (`.iqmsg`) 是不同的。

您可以指定是否忽略在装载期间发生的 `UNIQUE`、`NULL`、`DATA VALUE` 和/或 `FOREIGN KEY` 约束违规以及在启动回退前忽略的最大违规数量。您还可以指示装载将有关特定类型的完整性约束违规的信息按违规记录在消息日志中，并按行记录在行日志中。

MESSAGE LOG 内容和格式

`MESSAGE LOG` 文件包含所记录的每个完整性约束违规的行和列信息。

对于给定的负载，记录三种类型的消息：带时间戳的标头、行信息和带时间戳的结尾。每次装载出现一次标头。如果成功执行语句，则结尾出现一次。对于所记录的每个完整性约束违规，将会显示一次行信息。

标头消息的格式如下：

```
<datetime load started> Load Table <table-name>: Integrity
Constraint Violations
```

例如：

```
2009-05-24 23:04:31 Load Table Customers: Integrity Constraint  
Violations
```

行信息消息包括三个部分:

- 如果未发生完整性约束违规, 则为表中应已装载行的行号。
- 检测到的完整性约束违规的类型。
- 由 `schema` 指定的列编号。

例如,

```
1267 DATA VALUE 4 3216 UNIQUE 1 3216 NULL 3 3216 NULL 6 9677 NULL 1
```

结尾消息的格式如下:

```
<datetime load completed> Load Table <table-name> Completed
```

例如:

```
2009-05-24 23:05:43 LOAD TABLE Customers: Completed
```

注意: `MESSAGE LOG` 文件中的行数 (已报告错误) 可能超出 `IGNORE CONSTRAINT` 选项限制, 因为装载是由并行运行的多个线程执行的。多个线程可以报告约束违规次数已超出指定限定值。

ROW LOG 内容和格式

ROW LOG 文件包含在其中记录了所发生的完整性约束违规的每一行的行 ID 和数据值。

无论对于给定行发生多少次完整性约束违规, 都会只为该行显示一次行数据。对于给定的负载, 记录三种类型的消息: 带时间戳的标头、行数据和带时间戳的结尾。每次装载出现一次标头。如果成功执行语句, 则结尾出现一次。

标头消息的格式如下:

```
<datetime load started> Load Table <table-name>: Integrity  
Constraint Violations <formatting information>
```

其中 `<formatting information>` 是在设置行数据格式时使用的日期、时间和日期时间格式。例如:

```
2009-05-24 23:04:31 Load Table Customers: Integrity Constraint  
Violations Date Format: yyyy/mm/dd Time Format: hh:mm:ss Datetime  
format: yyyy/mm/dd hh:mm:ss
```

行数据消息包括两个部分:

- 如果未发生完整性约束违规, 则为表中应已装载行的行号。
- 行中的数据值, 由逗号或用户指定的 `LOG DELIMITED BY` 分隔符分隔。

例如,

```
3216 #Jones John#NULL#NULL#S#1945/01/12#NULL#
```

行数据消息中数据值的格式按下列规则确定:

- 当数据类型是 `VARBINARY` 或 `BINARY` 时，数据由 ASCII 十六进制字符表示。
- `DATE` 值以 `DATE_FORMAT` 数据库选项指定的格式表示。缺省格式为 `YYYY-MM-DD`。
- `DATETIME` 和 `TIMESTAMP` 值以 `TIMESTAMP_FORMAT` 数据库选项指定的格式表示。缺省为 `YYYY-MM-DD HH:NN:SS.SSS`。
- `TIME` 值以 `TIME_FORMAT` 数据库选项指定的格式表示。缺省为 `HH:NN:SS.SSS`。
- 空值由令牌 `NULL` 表示。

注意： 补白字段不会显示在行数据消息中。

结尾消息的格式如下：

```
<datetime load completed> Load Table <table-name>: Completed
```

例如：

```
2009-05-24 23:05:43 Load Table Customers: Completed
```

注意： `MESSAGE LOG` 文件中的不同错误数可能与 `ROW LOG` 文件中的行数不完全匹配。行数存在差异是由于多个线程执行的并行装载处理造成的。多个线程可以报告约束违规次数已超出指定限定值。

MESSAGE LOG 和 ROW LOG 示例

此示例说明了 `MESSAGE LOG` 和 `ROW LOG` 文件的内容和格式。

下面的 `CREATE TABLE` 语句创建使用 `LOAD TABLE` 语句装载的表：

```
CREATE TABLE Customers(name VARCHAR(80) NOT NULL, age TINYINT NULL,
sex CHAR(1) NOT NULL, marital_status CHAR(1) NULL, birthdate DATE NOT
NULL, credit_card VARCHAR(20)NOT NULL)
```

下面的 `LOAD TABLE` 语句将数据装载到 `Customers` 表中：

```
LOAD TABLE Customers ... IGNORE CONSTRAINT UNIQUE 200 MESSAGE LOG
'msg.log' ROW LOG 'row.log' ONLY LOG UNIQUE, NULL, DATA VALUE LOG
DELIMITED BY '#'
```

下面的原始数据是使用上面的 `LOAD TABLE` 语句从磁盘文件装载的：

```
Jones John, 19, M, S, 06/19/83, CC Cleven Bill, 56, M, OSIDJFJ,
02/23/43, CC Jones John, 339, M, NULL, 01/12/45, NULL NULL, 55, F, M,
10/02/37, ST
```

在 `LOAD TABLE` 完成后，`MESSAGE LOG` 文件 `msg.log` 包含以下信息：

```
2009-05-24 23:04:31 LOAD TABLE Customers: Integrity Constraint
Violations 1267 DATA VALUE 4 3216 UNIQUE 1 3216 NULL 6 9677 NULL 1
2009-05-24 23:05:43 LOAD TABLE Customers Completed
```

在 `LOAD TABLE` 完成后，`ROW LOG` 文件 `row.log` 包含以下信息：

```
2009-05-24 23:04:31 LOAD TABLE Customers Integrity Constraint
Violations Date Format: yyyy/mm/dd Time Format: hh:mm:ss Datetime
format: yyyy/mm/dd hh:mm:ss 1137 #Jones John#19#M#S#1983/06/19#CC#
1267 #Cleven Bill#56#M#OSIDJFJ#1943/02/23#CC# 3216 #Jones
```

```
John#NULL#NULL#S#1945/01/12#NULL# 9677 #NULL#55#F#M#1937/10/02#ST#  
2009-05-24 23:05:43 LOAD TABLE Customers Completed
```

二进制装载格式

要加快将数据装载到 Sybase IQ 中的速度，请创建 Sybase IQ 二进制格式的数据文件，然后使用 **LOAD TABLE** 的 **FORMAT BINARY** 和 **BINARY** 列规范子句将此数据装载到 Sybase IQ 中。

有关使用 **LOAD TABLE** 的说明，请参见《参考：语句和选项》。

创建具有这些二进制格式的数据文件可装载到具有相应数据类型的列中。大多数情况下，Sybase IQ 使用特定于平台的二进制格式。以下数据类型是使用特定于 Sybase IQ 的二进制格式的例外：

- DATE
- TIME
- DATETIME
- NUMERIC

IQ 二进制装载格式和装载效率

Sybase IQ 二进制装载格式是固定宽度格式。

一般来说，固定宽度装载比可变宽度装载完成得快。当装载逻辑知道列和行的宽度时，可以更高效地处理数据。使用分隔符分隔宽度不同的列和行将强制装载花费时间扫描输入数据来查找它们。

IQ 二进制装载格式是固定宽度装载。装载可以从表定义中的信息确定每列的宽度和每行的长度。

注意： 二进制装载格式是与端模式相关的。这是因为二进制装载格式使用本机二进制数据类型表示数据。

操作系统本机数据类型

以下数据类型的数据以本机操作系统二进制格式存储，能够以该格式直接写入数据文件。Sybase IQ 将各自的字节数直接读入相关数据类型而不进行转换。

- BIT (1 字节)
- TINYINT (1 字节)
- SMALLINT (2 字节)
- INT/UNSIGNED INT (4 字节)
- BIGINT/UNSIGNED BIGINT (8 字节)
- FLOAT (4 字节)
- DOUBLE (8 字节)

- CHAR/VARCHAR (字符数据)
- BINARY/VARBINARY (二进制数据)

缺省情况下, VARCHAR 和 VARBINARY 列读入 **LOAD TABLE column-spec** 指定的字节数。

DATE

DATE 列数据以 4 字节 (32 位无符号整数) 的形式存储在 Sybase IQ, 表示自 0000-01-01 以来的天数。

要将给定年份、月份和日期的日历日期转换为 Sybase IQ 二进制格式, 请使用:

```
year = current_year - 1;
days_in_year_0000 = 366;
binaryDateValue = (year * 365)
+ (year / 4)
- (year / 100)
+ (year / 400)
+ days_in_year_0000
+ day_of_current_year
- 1;
```

对于上述公式中的 *day_of_current_year* 值, 请参见以下示例: 2 月 12 日为第 43 天。

TIME

TIME 数据存储为 64 位无符号数量, 它表示以微秒 (换言之, 1.0e-6 秒) 为单位的数字。

对于给定小时、分钟、秒和微秒 (*usec*) 计算微秒量:

```
binaryTimeValue = (hour * 3600 + minute * 60 + second + microsecond )
* 1000000
```

TIMESTAMP

TIMESTAMP 数据存储为 64 位无符号整数, 表示以微秒为单位的数量。可以按如下方式对给定的年份、月份、日期、小时、分钟、秒钟和微秒计算二进制 **TIMESTAMP** 值:

计算如上所示日期的 *binaryDate Value*。

计算如上所示时间的 *binaryTime Value*。

```
binaryDateTimeValue = binaryDateValue * 86400000000 +
binaryTimeValue
```

NUMERIC 和 DECIMAL

NUMERIC 和 DECIMAL 数据类型的格式随精度的变化而变化。

必须用零填充值的右侧, 以达到值的全部位数。还必须用零完全填充值的左侧, 但填充随二进制编程自动进行。填充值后, 将删除小数点。例如, 对于值 12.34 将具有如下情况:

- NUMERIC(4,2): 1234
- NUMERIC(6,4): 123400
- NUMERIC(8,4): 00123400
- NUMERIC(12,6): 000012340000
- NUMERIC(16,8): 0000001234000000

填充值并删除小数点之后，将应用以下规则：

- 如果精度 ≤ 4 ，则二进制格式与 2 字节整数数量的本机操作系统二进制格式相同。
- 如果精度在 5 和 9 之间，则二进制格式与 4 字节整数数量的本机操作系统二进制格式相同。
- 如果精度在 10 和 18 之间，则二进制格式与 8 字节整数数量的本机操作系统二进制格式相同。
- 如果精度 ≥ 19 ，则有一个使用以下 C++ struct 定义的特殊格式：

```
struct { unsigned char sign; // sign 1 for +, 0 for - unsigned char
ndig; // # digits unsigned char exp; // exponent unsigned char
erracc; // should be 0 unsigned short digits[80]; };
```

指数是 excess-80 格式，除非值为零。“零”值表示为：

```
sign = 1 ndig = 0 erracc = 0 exp = 0
```

最大指数值是 159。支持的最大位数是 288。“digits[0]”包含重要性最低的数字。以打包表示形式存储的数字每“无符号短型值”（2 字节）数量有两个数字。对于给定的“数字”：

- lower order digit = digit[i] & 0x00FF
- high order digit = digit[i] & 0xFF00

例如，考虑以 NUMERIC(20) 表示的值 100。此值的二进制布局为：

```
0x0101 0x5000 0x0064 0x0000 0x0000 ..... Sign = 0x01 Number
digits = 0x01 Exponent = 0x50 Erracc = 0x00 Digits = 0x0064
```

再例如，考虑值 32769：

```
0x0102 0x5000 0x0ad1 0x0003 0x0000 0x0000 .... Sign = 0x01
Number digits = 0x02 Exponent = 0x50 Erracc = 0x00 Digits = 0x0ad1
0x0003
```

如果将数字转换成基数 10，则有：

```
0x0ad1 = 2769 0x0003 = 3
```

空值装载

插入空值最方便的方法是在输入文件中使用空值字节并在 **LOAD TABLE** 语句的列规范中指定 **WITH NULL BYTE**。

这通过使用“x00”或“x01”终止输入文件中的每个数据字段来完成。使用“x01”终止输入文件中的数据字段指示装载将 NULL 插入相应列。例如：

```
create table d1 ( c1 date ); load table d1 ( c1 binary with null
byte ) from 'filename' quotes off escapes off format binary;
```

如果装载输入文件的内容是 000b32cb00000b32cc00，则两行将装载到表中。第一行将是 May 7, 2009，第二行将是 May 8, 2009。NULL 字节添加到输入文件中每个二进制日期之后。如果要将空值装载到第一行中，请在输入文件中将空值字节的值更改为 “x01”。

```
000b32cb01000b32cc00
```

再例如，若要将值 32769 装载到 NUMERIC(20) 列中，输入文件应包含：

```
0x0102 0x5000 0x0ad1 0x0003 0x0000 0x00
```

这包括空值字节。

若要将 23456789012345678.12 装载到以 NUMERIC(19,2) 定义的列中，装载输入文件应包含：

```
0x0106 0x4f00 0x04b0 0x162e 0x04d2 0x1ed2 0x0d80 0x0002 0x0000 0x00
```

数字后跟空值字节 (0x00)。

在表示该数值数量的结构的数字数组中，有七个无符号短型值（编号 0 - 6）。

“digits[0]” 包含重要性最低的数字。

```
digits[0] = 0x04b0 (decimal 120) digits[1] = 0x162e (decimal 5678)
digits[2] = 0x04d2 (decimal 1234) digits[3] = 0x1ed2 (decimal 7890)
digits[4] = 0x0d80 (decimal 3456) digits[5] = 0x0002 (decimal 2)
digits[6] = 0x0000
```

装载到表列中时，列规范的 NULL 部分指示如何将某些输入值视为 NULL 值。这些字符可能包括 **BLANKS**、**ZEROS** 或您定义的任何其它文字列表。当指定空值或从源文件读取空值时，目标列必须能够包含 NULL。

ZEROS 的解释如下所示：

- 如果输入数据均为二进制零（而不是字符零），则将列设置为空值。
- 如果输入数据为字符零，则：
 - NULL(ZEROS) 永远不会导致列变为空值。
 - NULL('0') 导致列变为空值。例如：

装载：

```
CREATE TABLE t1 ( c1 INT, c2 INT );
```

查看输入数据文件，它使用大端字节排序：

```
od -x data.inp 3030 3030 0000 04d2
```

执行：

```
LOAD TABLE t1 ( c1 ASCII(4) NULL( '0000' ), c2 BINARY ) FROM
'data.inp' FORMAT BINARY QUOTES OFF ESCAPES OFF;
```

结果：

```
SELECT * FROM t1; c1 c2 NULL 1234
```

- 如果输入数据为二进制零（清除所有位），则：
 - NULL(ZEROS) 导致列变为空值。

- `NULL('0')` 永远不会导致列变为空值，例如：

装载：

```
CREATE TABLE t1 ( c1 INT, C2 INT );
```

查看输入数据文件，它使用大端字节排序：

```
od -x data.inp 0000 0000 0000 04d2
```

执行：

```
LOAD TABLE t1 ( c1 ASCII(4) NULL( zeros ), c2 BINARY ) FROM  
'data.inp' FORMAT BINARY QUOTES OFF ESCAPES OFF;
```

结果：

```
SELECT * FROM T1; c1 c2 NULL 1234
```

再例如，如果 **LOAD TABLE** 语句包含 `col1 date('yyymmdd') null(zeros)` 以及要装载的数据为 `000000`，您会收到一个错误，指示 `000000` 无法转换为 `DATE(4)`。若要使 **LOAD TABLE** 在数据为 `000000` 时在 `col1` 中插入空值，请将 `NULL` 子句编写为 `null('000000')`，或将数据修改为等同二进制零并使用 `NULL(ZEROS)`。

在二进制装载过程中装载空值的另一种方法是，如果目标列接受空值，则不为 **LOAD TABLE** 语句中的列提供数据。例如：

```
CREATE TABLE t1 ( c1 INT, c2 INT ); LOAD TABLE T1 ( c2 BINARY ) FROM  
'data.inp' FORMAT BINARY QUOTES OFF ESCAPES OFF; SELECT  
* FROM T1; c1 c2 NULL 1234 NULL 1234
```

查看输入数据文件，它使用大端字节排序：

```
od -x data.inp 0000 04d2 0000 04d2
```

使用 INSERT 语句

使用 **INSERT** 语句，可以直接插入数据，而无需先将数据置于平面文件中。

使用此命令，可以：

- 逐行插入一组指定的值
- 直接从数据库表插入

有关命令的这两种形式的详细信息，请参见下面各节。

另请参见

- 交互式数据导入（第 254 页）

逐行插入指定值

要逐行向表中插入指定值，请使用 **INSERT** 的语法 1。

Sybase IQ 将您指定的第一个值插入您指定的第一列中，将您指定的第二个值插入第二列中，依此类推。如果省略列名的列表，则按照创建列的顺序（该顺序与 **SELECT *** 的检索顺序相同）将值插入到表列中。只要有空间可用，**Sybase IQ** 便会将行插入到表中。

值可以是空值、任何正数或负数，或文字。

- 将 **CHAR**、**VARCHAR**、**DATE**、**TIME** 和 **TIMESTAMP** 或 **DATETIME** 列的值用单引号或双引号引起来。要表示含引号的值，请将一组不同的引号用作外部引号，如 “Smith' s”。
- 对于 **DATE**、**TIME** 和 **TIMESTAMP** 或 **DATETIME** 列，必须使用特定的格式。

注意： **TIMESTAMP** 和 **DATETIME** 数据类型是相同的。

允许空值

在只为行中的某些列指定值时，如果未指定值的列允许空值，则会为该列插入空值。

如果指定空值，则目标列必须允许空值，否则将拒绝 **INSERT**，并会在消息日志中生成一条错误消息。缺省情况下，**Sybase IQ** 列允许空值，但可以通过在 **CREATE TABLE** 语句的列定义中指定 **NOT NULL** 或通过其它方式（例如，使用主键）来改变这种情况。

示例

下面的示例在 **lineitem** 表中将 1995-06-09 添加到 **l_shipdate** 列，并将 123 添加到 **l_orderkey** 列。

```
INSERT INTO lineitem      (l_shipdate, l_orderkey)
VALUES('1995-06-09', 123)
```

如果要插入的数据行数较多，则使用以下方法更加有效：直接从数据库插入所选定行（如下一节所述），或者使用 **LOAD TABLE** 语句从平面文件装载数据，而不是逐行插入值。考虑配合使用 **SELECT** 语句和一些联合，而不是为一些行插入值，因为这样做仅需要到服务器的单个行程。

另请参见

- 在插入时转换数据（第 257 页）

插入数据库中的选定行

可以根据常规 **SELECT** 语句的结果插入任意数量的数据行。

要将当前数据库的其它表中的数据或定义为 **Specialty Data Store** 的数据库中的数据插入到 **Sybase IQ** 中，请使用 **INSERT** 语句的语法 2。

为了获得最高效率，请通过一个 **INSERT** 语句插入尽可能多的行。要在首次插入后插入另外几组行，请使用其它 **INSERT** 语句。

像其它 **SQL** 数据库一样，**Sybase IQ** 通过将目标列表和选择列表中指定的列的顺序相匹配来插入数据；即，来自选择列表第一列的数据将插入到第一个目标列中，依此类推。对于 **INSERT SELECT** 和 **INSERT VALUES**，如果省略目标列名，则 **Sybase IQ** 会按列的创建顺序向列中插入数据。

您要向其中插入数据的表必须存在于您当前连接到的数据库中。**Sybase IQ** 将数据插入到目标列的所有索引中。

选择列表的表中的列和表中的列必须具有相同或兼容的数据类型。换言之，选择项的值必须是或者必须能够转换为表中列的数据类型。

使用此形式的 **INSERT** 语句，可以指定任意一个 **insert-load-options**。

示例

此示例说明了从一个表 **partsupp** 向同一数据库中的另一个表 **lineitem** 插入数据。

来自源列 **l_quantity** 的数据将插入到目标列 **ps_availqty** 中。

```
INSERT INTO partsupp(ps_availqty) SELECT l_quantity FROM lineitem
```

从其它数据库插入

您可以从任何可访问数据库中的表插入数据。

- 您当前连接到的数据库的 **IQ** 存储或 **Catalog** 存储中的表。
- **Adaptive Server Enterprise** 数据库中的表。
- 当前数据库中的代理表，它对应于远程服务器上数据库中的表。

直接从 **Adaptive Server Enterprise** 插入数据

要从 **Adaptive Server Enterprise** 或 **SQL Server** 数据库插入数据，请使用 **INSERT** 语句的 **LOCATION** 语法。

要直接从 **Adaptive Server** 使用插入数据，必须满足下列所有条件：

- **Sybase** 连接库必须安装在您的系统上，并且您的平台上的装载库路径环境变量必须指向它们。

- 您所连接到的 Adaptive Server Enterprise 服务器必须存在于本地计算机上的 `interfaces` 文件中。
 - 您必须对源 ASE 或 Sybase IQ 数据库具有读取权限，并对目标 Sybase IQ 数据库具有 `INSERT` 权限。
1. 使用相同的用户 ID 和口令连接到 Adaptive Server Enterprise 和 Sybase IQ 数据库。
 2. 对 Sybase IQ 数据库，使用下面的语法发出语句：

```
INSERT INTO iq_table LOCATION 'ase_servername.ase_dbname'
{ SELECT col1, col2, col3,... FROM owner.ase_table }
```

3. 发出 `COMMIT` 以提交插入。

当 Sybase IQ 连接到远程服务器时，`INSERT...LOCATION` 还可以使用当前连接的用户 ID 的远程登录名，但前提是已使用 `CREATE EXTERNLOGIN` 创建远程登录名并且已使用 `CREATE SERVER` 语句定义了远程服务器。

注意： 您还可以使用此方法在两个 Sybase IQ 数据库之间移动选定列。

装载 ASE 文本和图像

Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 `TEXT`，但是，可从长度大于 255 字节的 `IQ CHAR` 或 `VARCHAR` 列、`LONG VARCHAR` 列和数据类型为 `TEXT` 的 ASE 数据库列执行 `INSERT...LOCATION`（语法 3）。如果 Sybase IQ 支持内部转换，可将 ASE `TEXT` 和 `IMAGE` 列插入到其它 Sybase IQ 数据类型的列中。另请注意，`INSERT...LOCATION` 不支持在 `SELECT` 语句中使用变量。缺省情况下，如果远程数据列包含 2GB 以上的数据，Sybase IQ 则会以无提示方式将列值截断为 2GB。

用户必须获得专门许可才能使用非结构化数据分析功能。有关详细信息，请参见《Sybase IQ 中的非结构化数据分析》。

您可以替代分隔 `SELECT` 语句的单引号的大括号 `{}`。（请注意，在 ODBC 标准中，大括号表示转义序列的开头和结尾，在 ODBC 环境中可能会导致产生错误。）

示例

以下命令从服务器 `detroit` 上的 Sybase IQ 数据库 `iq11db.dba` 中将 `lineitem` 表的 `l_shipdate` 和 `l_orderkey` 列中的数据插入到当前数据库的 `lineitem` 表的相应列中。

```
INSERT INTO lineitem (l_shipdate, l_orderkey) LOCATION
'detroit.iq11db' { SELECT l_shipdate, l_orderkey FROM
lineitem }
```

- 目标列和源列的名称可能不同。
- 指定列的顺序非常重要，因为来自所指定的第一个源列的数据将插入到所指定的第一个目标列中，依此类推。
- 您可以在 `INSERT` 命令中使用 `SELECT` 语句的谓词，以便仅插入表的某些行中的数据。

示例

此示例与上面的示例插入相同的列，但只为 `l_orderkey` 值为 1 的行插入。此外，在此示例中，将 TDS 包大小指定为 512 字节。

```
INSERT INTO lineitem (l_shipdate, l_orderkey) LOCATION
'detroit.iqdb' PACKETSIZE 512 { SELECT l_shipdate,
l_orderkey FROM lineitem WHERE l_orderkey = 1 }
```

另请参见

- 使用 `LOAD TABLE` 语句执行 bulk load 操作（第 238 页）

交互式数据导入

如果您要插入少量数据，可能会喜欢使用 `INSERT` 语句通过 Interactive SQL 以交互方式进行输入数据。

例如，您可以使用以下命令一次一行地插入所列出的值：

```
INSERT INTO T1
VALUES ( ... )
```

注意：不要使用 Interactive SQL “数据”菜单中的“导入”选项。Sybase IQ 数据库不支持使用此选项。

另请参见

- 使用 `INSERT` 语句（第 250 页）

在采用不同端格式的系统之间移动数据

您可以将大端格式的数据库中的数据移至小端格式的数据库中。

前提条件

注意：开始之前，确保您具有用于捕获数据库 `schema` 和表模式的进程。

以下示例装载名为 `lineitem` 的表，并在 UFS（文件系统）上标识一个名为 `lineitem_binary.inp` 的提取文件。

查看操作系统文档，了解系统的最大文件大小。例如，Sun Solaris x64 上的提取文件的最大大小为 512 GB。

过程

此过程移动表定义，但不包含对数据库对象的迁移，例如，必须重新创建的存储过程或事件。

例如，在 Sun64 SPARC 系统上建立的 Sybase IQ 数据库以大端（最重要字节在前）格式存储二进制数据。因为 Sun Solaris x64 是小端系统，所以在 Sun64 SPARC 上构建的 Sybase IQ 数据库无法通过 **ALTER DATABASE UPGRADE** 升级以在 Sun Solaris x64 系统上运行。

要对采用不同端模式结构的跨硬件平台移动各数据库的数据，必须执行下列操作：

- 从源平台复制数据库 schema（例如，表、索引等）。
- 在目标平台上创建新的数据库。
- 从源数据库执行二进制数据转储。
- 将数据装载到新的目标数据库。

1. 激活提取实用程序：

```
SET TEMPORARY OPTION Temp_Extract_Name1 =
'lineitem_binary.inp'
```

```
SET TEMPORARY OPTION Temp_Extract_Name2 = ''
```

2. 对 lineitem 表设置二进制提取：

```
SET TEMPORARY OPTION Temp_Extract_Binary = 'on'
```

```
SET TEMPORARY OPTION Temp_Extract_Swap = 'off'
```

3. 将输出放置到文件 lineitem_binary.inp 中：

```
SELECT * FROM lineitem
```

4. 关闭提取实用程序：

```
SET TEMPORARY OPTION Temp_Extract_Name1 = ''
```

5. 在目标系统上创建数据库副本。

6. 假定表 lineitem 的定义方式如下，并按如下方式装载 lineitem 表：

```
LOAD TABLE lineitem ( l_orderkey      BINARY WITH NULL BYTE,
l_partkey      BINARY WITH NULL BYTE, l_suppkey      BINARY
WITH NULL BYTE, l_linenumber      BINARY WITH NULL BYTE,
l_quantity      BINARY WITH NULL BYTE, l_extendedprice BINARY
WITH NULL BYTE, l_discount      BINARY WITH NULL BYTE,
l_tax          BINARY WITH NULL BYTE, l_returnflag      BINARY WITH
NULL BYTE, l_linestatus      BINARY WITH NULL BYTE,
l_shipdate      BINARY WITH NULL BYTE, l_commitdate      BINARY
WITH NULL BYTE, l_receiptdate      BINARY WITH NULL BYTE,
l_shipinstruct  BINARY WITH NULL BYTE, l_shipmode      BINARY
WITH NULL BYTE, l_comment      BINARY WITH NULL BYTE ) FROM 'C:
\\mydata\\lineitem_binary.inp' FORMAT BINARY STRIP OFF QUOTES OFF
ESCAPES OFF PREVIEW ON BYTE ORDER HIGH; COMMIT
```

请特别注意下列两个子句：

- 当装载二进制文件时，需要 **BINARY WITH NULL BYTE**。
- **BYTE ORDER HIGH** 从生成数据的系统指定字节顺序。本示例中的源数据库为大端平台；因此，此数据需要 **HIGH** 字节顺序。（小端数据库需要 **LOW** 字节顺序。）

当装载 **Multiplex** 数据库时，请在所有文件名中使用绝对（完全限定）路径。请不要使用相对路径名。

插入到连接索引的表中

同其它任何索引一样，您可以将数据装载或插入到基于连接索引的表中。只有一个差别：必须先同步连接索引中的数据，然后才能使用连接索引来解析查询。

注意： 您不能更新参与任何连接索引中的基表。只能执行插入、装载或删除操作。

首次创建连接索引时，**Sybase IQ** 会自动为您同步该连接索引。无论是在装载前还是装载后创建连接索引都无关紧要。该顺序也不会影响装载或同步的性能。

但是，一旦创建了连接索引，如果将数据插入或装载到连接层次中除顶层表以外的任何基础表中，就必须显式同步它。为此，请使用 **SYNCHRONIZE JOIN INDEX** 语句。请参见《参考：语句和选项》。

一旦有用户更新了连接索引中的表，其它用户将无法更新基于该连接索引的任何表，直到该连接索引已同步为止。

从不同的连接更新可能会导致发生错误

当有多名用户针对参与同一连接索引的不同表插入或删除数据时，除非在第二个用户的事务开始前同步提交，否则第二个用户的更新将会失败。如果存在下列任意一种情况，就会出现这种失败：

- 第二个用户的事务在第一个用户的事务提交前开始。
- 第二个用户尝试在第一个用户的事务已提交但连接索引尚未同步前进行更新。

发生此问题的原因是，**Sybase IQ** 在它的任何基础表更新时都会创建新版的连接索引。该新版本对于已经开始的其它事务是不可见的。如果一个用户进行所有更改，则不会发生此问题，因为较新的表版本对进行原始更改的用户而言是可见的。

例如，假设表 **A**、**B** 和 **C** 是同一连接索引的所有成员。用户 2 开始一个事务，并向未参与连接的另一表中进行写入。现在，用户 1 向表 **B** 中进行插入。此操作会创建新版本的表 **B** 以及新版本的连接索引。然后，用户 2 尝试向表 **C** 中进行写入。即使没有其它用户更改过表 **C**，在同步连接索引前，表 **C** 也无法更新（因为表 **C** 是连接索引的成员）。

插入到主键列和外键列

将数据装载或插入到主键列和外键列中就像装载或插入到任何其它列中一样。

当插入到主键中时，**Sybase IQ** 会检查每个值是否唯一。如果不唯一，则会出错。

装载或提取大对象数据

用户必须获得专门许可才能使用非结构化数据分析功能。

要装载或提取大对象数据，请参见《Sybase IQ 中的非结构化数据分析》。

在插入时转换数据

您输入到 Sybase IQ 数据库中的数据将有可能来自各种来源。

并非所有的数据都与 Sybase IQ 数据类型完全匹配。其中的某些数据将需要进行转换。数据可通过以下两种方式来转换：显式和隐式。例如，要将 INT 数据插入 CHAR 列，您必须显式转换该数据。

隐式转换可能会在以下情况下发生：

- 插入从同一数据库的另一列中选择的数据时
- 插入从另一个数据库中选择的数据时
- 从平面文件中装载数据时

当需要进行显式转换时，您指定转换的方式取决于您是从平面文件中装载还是插入所进行：

- 在 **LOAD TABLE** 语句中，可以通过在 *column-spec* 中指定格式来显式转换数据。
- 在 **INSERT** 语句中，可以在 **SELECT** 语句或 **VALUES** 列表中使用数据转换函数 **CAST**、**CONVERT** 和 **DATEPART** 来显式转换数据。

尽管大多数 Sybase IQ 数据类型都与同名的 SQL Anywhere 和 Adaptive Server Enterprise 数据类型完全兼容，但是它们之间还是有些差异。

为了在不同的版本之间实现兼容性，已经将几个数据类型定义为其它数据类型的同义词：

- **DECIMAL** 是 **NUMERIC** 的同义词。
- **INTEGER** 是 **INT** 的同义词。
- **DATETIME** 是 **TIMESTAMP** 的同义词。
- **FLOAT** (*precision*) 是 **REAL** 或 **DOUBLE** 的同义词，具体取决于 *precision* 的值。对于 Adaptive Server Enterprise，*precision* 小于或等于 15 时使用 **REAL**，而 *precision* 大于 15 时使用 **DOUBLE**。对于 Sybase IQ 和 SQL Anywhere，分界值与平台有关，但所有平台上的分界值均大于 22。
- **MONEY** 是与 Adaptive Server Enterprise 兼容的 **NUMERIC(19,4)** 的同义词，允许使用空值。
- **SMALLMONEY** 是与 Adaptive Server Enterprise 兼容的 **NUMERIC(10,4)** 的同义词，允许使用空值。

您可以将同义词与其标准数据类型交换使用。当存在同义词时，数据在内部以标准数据类型存储。在错误消息中，将显示标准名称来代替同义词。

注意： 缺省情况下，Sybase IQ 假设输入数据为二进制（数值型数据）并尝试按照这种方式插入数据。但是，这会假设输入列的字节长度必须与目标列的字节长度相匹配。如果不匹配的话，插入操作将失败或者导致意外的结果。例如，如果您尝试在 SMALLINT 目标列中插入包含整数数据且长度为 4 字节的输入列，Sybase IQ 将只装载该输入列的前 2 字节。

另请参见

- 逐行插入指定值（第 251 页）
- IQ 中的显式数据转换（第 259 页）
- 匹配 Adaptive Server Enterprise 数据类型（第 270 页）

Sybase IQ 15 版之前的版本中的数据

如果要将数据从旧版本移到 Sybase IQ 15.x 版，必须在插入或装载之前转换某些数据类型。

请参见《安装和配置指南》中的“从早期版本迁移数据”。

装载转换选项

下表按字母顺序列出了 **LOAD TABLE** 语句的转换选项，并提供了每个选项所执行操作的简要说明。

表 25. 用来从平面文件装载的转换选项

选项	Sybase IQ 数据类型	操作
ASCII	TINYINT, SMALL-INT, INT (or INTEGER), UNSIGNED INT, BIGINT, UNSIGNED BIGINT, NUMERIC (or DECIMAL), REAL, DOUBLE, BIT, DATE, TIME, TIMESTAMP (or DATETIME)	缺省情况下，Sybase IQ 假设输入数据为二进制且具有适于该数据类型的宽度。使用 ASCII 将允许您通知 Sybase IQ 数据采用字符格式并允许您指定数据的宽度。此选项允许针对 REAL 数据采用 E 表示法，但这可能会影响性能。
ASCII	CHAR, VARCHAR	缺省情况下，Sybase IQ 假设源列和目标列的宽度相同，这可能会导致它无法正确地读取输入文件。此选项允许您为输入列指定不同于目标列的宽度。
DATE	DATE	将所输入的固定格式的 ASCII 日期转换为二进制。

选项	Sybase IQ 数据类型	操作
DATETIME	TIMESTAMP (或 DATETIME) 或 TIME	将所输入的固定格式的 ASCII 时间或日期/时间转换为二进制。输入规范基于 12 小时制或 24 小时制。
TIME	TIME	将所输入的固定格式的 ASCII 时间转换为二进制。
NULL	all	用于指定在插入时要将哪些输入数据值转换为空值。

注意： 在从平面文件装载时，如果系统让您在二进制数据和字符数据之间进行选择，请选择使用前者。使用二进制输入可以消除转换开销，从而提高性能。

另请参见

- 指定输入数据格式的装载 (第 240 页)

IQ 中的显式数据转换

在使用 **INSERT** 语句从数据库 (而非平面文件) 直接插入数据时，不能使用装载转换选项。

如果数据要求进行显式转换，您必须在用来指定要插入数据的 **SELECT** 语句或 **VALUES** 列表中使用转换函数 **CAST** 或 **CONVERT**。如果数据已隐式转换，Sybase IQ 会自动处理转换。

当需要对 **SELECT** 语句中的数据类型进行匹配时，需要进行隐式或显式转换，但并未进行转换。在通过执行 **INSERT SELECT** 将一种数据类型转换为另一种数据类型时，以及比较或计算不同数据类型的值时，会发生这种情况。

下表显示：

- Sybase IQ 隐式 (I) 执行的转换
- 您必须显式 (E) 执行的转换
- 不受支持 (U) 的转换

这些转换适用于 Sybase IQ 数据库中的数据、来自 SQL Anywhere 数据库中的数据或者以 Speciality Data Store 形式连接的任何其它数据库中的数据。

第一个表列出了当 **SELECT** 语句中没有 **WHERE** 子句时，或者当 **WHERE** 子句基于比较运算 (=、> 或 <) 时，隐式 (I)、显式 (E) 和不受支持 (U) 的转换。

表 26. 比较运算的 IQ 转换

	更改为：																
从：	ti	si	in	ui	bi	ub	nu	rl	dl	bt	dt	tm	ts	ch	vc	bn	vb
tinyint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
smallint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I

	更改为:																
从:	ti	si	in	ui	bi	ub	nu	rl	dl	bt	dt	tm	ts	ch	vc	bn	vb
int	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
unsigned int	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
bigint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
unsigned bigint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
numeric	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	U	U
real	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	U	U
double	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	U	U
bit	I	I	I	I	I	I	I	I	I	I	U	U	U	I	I	I	I
date	E	E	E	E	E	E	E	E	E	U	I	U	I	E	E	U	U
time	E	E	E	E	E	E	E	E	E	U	U	I	E	E	E	U	U
time-stamp	E	E	E	E	E	E	E	E	E	U	E	I	I	E	E	U	U
char	E	E	E	E	E	E	E	E	E	I	E	E	E	I	I	I	I
varchar	E	E	E	E	E	E	E	E	E	I	E	E	E	I	I	I	I
binary	I	I	I	I	I	I	U	U	U	U	U	U	U	I	I	I	I
varbinary	I	I	I	I	I	I	U	U	U	U	U	U	U	I	I	I	I

下面的列表包含这些表中所用代码的说明：

代码	数据类型	代码	数据类型	代码	数据类型
ti	tinyint	nu	numeric	ts	timestamp
si	smallint	rl	real	ch	char
in	int	dl	double	vc	varchar
ui	unsigned int	bt	bit	bn	binary
bi	bigint	dt	date	vb	varbinary
ub	unsigned bigint	tm	time		

第二个表中列出了当 **SELECT** 语句中的 **WHERE** 子句基于算术运算 (+、- 等) 时，隐式 (I)、显式 (E) 和不受支持 (U) 的转换。

表 27. 算术运算的 IQ 转换

	更改为:																
从:	ti	si	in	ui	bi	ub	nu	rl	dl	bt	dt	tm	ts	ch	vc	bn	vb
tinyint	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
smallint	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
int	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
unsigned int	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
bigint	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
unsigned bigint	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	I	I
numeric	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	U	U
real	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	U	U
double	I	I	I	I	I	I	I	I	I	I	U	U	U	E	E	U	U
bit	I	I	I	I	I	I	I	I	I	I	U	U	U	I	I	I	I
date	U	U	U	U	U	U	U	U	U	U	U	I	U	U	U	U	U
time	U	U	U	U	U	U	U	U	U	U	I	U	U	U	U	U	U
timestamp	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
char	E	E	E	E	E	E	E	E	E	I	U	U	U	I	I	I	I
varchar	E	E	E	E	E	E	E	E	E	I	U	U	U	I	I	I	I
binary	I	I	I	I	I	I	U	U	U	U	U	U	U	I	I	I	I
varbinary	I	I	I	I	I	I	U	U	U	U	U	U	U	I	I	I	I

注意: 在算术运算中, bit 数据会隐式转换为 tinyint。

第三个表列出了 INSERT 和 UPDATE 语句的隐式 (I)、显式 (E) 和不受支持 (U) 的转换。

表 28. INSERT 和 UPDATE 的 IQ 转换

	更改为:																
从:	ti	si	in	ui	bi	ub	nu	rl	dl	bt	dt	tm	ts	ch	vc	bn	vb
tinyint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
smallint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
int	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I

	更改为:																
从:	ti	si	in	ui	bi	ub	nu	rl	dl	bt	dt	tm	ts	ch	vc	bn	vb
unsigned int	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
bigint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
unsigned bigint	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	I	I
numeric	I	I	I	I	I	I	I	I	I	E	E	E	E	E	E	U	U
real	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	U	U
double	I	I	I	I	I	I	I	I	I	I	E	E	E	E	E	U	U
bit	I	I	I	I	I	I	I	I	I	I	U	U	U	I	I	I	I
date	E	E	E	E	E	E	E	E	E	E	I	U	I	E	E	U	U
time	E	E	E	E	E	E	E	E	E	E	U	I	E	E	E	U	U
time-stamp	E	E	E	E	E	E	E	E	E	E	E	I	I	E	E	U	U
char	I	I	I	I	I	I	I	I	I	I	E	E	E	I	I	I	I
varchar	I	I	I	I	I	I	I	I	I	I	E	E	E	I	I	I	I
binary	I	I	I	I	I	I	U	U	U	I	U	U	U	I	I	I	I
varbinary	I	I	I	I	I	I	U	U	U	I	U	U	U	I	I	I	I

另请参见

- 在插入时转换数据 (第 257 页)
- 匹配 Adaptive Server Enterprise 数据类型 (第 270 页)
- 支持的列缺省值 (第 321 页)

列宽问题

Sybase IQ 假定输入数据的宽度与目标列的宽度相同，并以相应方式读取输入文件。

如果二者的宽度不相同，Sybase IQ 可能会为该列读取过少或过多的输入文件字节数。结果，该列的读取可能不正确，并且对输入文件中后续各列的读取也将不正确，因为它们不会从输入文件中的正确位置开始。

例如，如果 input_column1 的宽度为 15 字节，destination_column1 的宽度为 10 字节，并且未指定 ASCII 转换选项，则 Sybase IQ 假定输入列宽度仅为 10 字节。这对于 destination_column1 并无大碍，因为在任何情况下，输入数据都将被截断为 10 字节。但是，这还表示 Sybase IQ 假定输入文件中的下一列将从第 11 字节开始（该字节仍位于第一列中），而不是从第 16 字节开始（该字节是下一列的正确起始位置）。

相反，如果 input_column1 的宽度为 10 字节，destination_column1 的宽度为 15 字节，并且未指定 ASCII 转换选项，则 Sybase IQ 假定输入列宽度为 15 字节。这表示

Sybase IQ 将读取所有 `input_column1` 和输入文件下一列中的 5 字节，并将此值插入到 `destination_column1`。这样，该值即被插入到 `destination_column1` 中，并且所有后续列均不正确。

要防止出现此类问题，请使用 **ASCII** 转换选项。Sybase IQ 通过此选项提供了几种指定固定宽度或可变宽度的输入列的方法。输入数据可以包含具有特定大小（字节）的固定宽度的输入列、带有列分隔符的可变宽度的输入列，以及根据二进制前缀字节定义的可变宽度的输入列。

更快的日期和时间装载

Sybase IQ 具有在装载时对日期、时间和日期时间数据进行 `ascii` 到二进制类型转换的内置性能优化功能。如果要装载的原始数据与其中一种格式完全相符，采用相应格式可以显著减少装载时间。

识别的格式包括：

- “YYYY-MM-DD”
- “YYYY/MM/DD”
- “YYYY.MM.DD”
- “YYYYMMDD”
- “MM-DD-YYYY”
- “MM/DD/YYYY”
- “DD-MM-YYYY”
- “DD/MM/YYYY”
- “DD.MM.YYYY”
- “HH:NN:SS”
- “HHNNSS”
- “HH:NN:SS.S”
- “HH:NN:SS.SS”
- “HH:NN:SS.SSS”
- “HH:NN:SS.SSSS”
- “HH:NN:SS.SSSSS”
- “HH:NN:SS.SSSSSS”
- “YYYY-MM-DD HH:NN:SS”
- “YYYYMMDD HHNNSS”
- “YYYY-MM-DD HH:NN:SS.S”
- “YYYY-MM-DD HH:NN:SS.SS”
- “YYYY-MM-DD HH:NN:SS.SSS”
- “YYYY-MM-DD HH:NN:SS.SSSS”
- “YYYY-MM-DD HH:NN:SS.SSSSS”
- “YYYY-MM-DD HH:NN:SS.SSSSSS”

如果装载的表具有一个或多个日期、时间或日期时间列，并且输入格式为上述格式之一，当在装载语句中显式指定相应格式时，可显著提高装载运行速度。否则，装载运行速度会非常慢。

假定表含有日期列，并按如下方式进行创建：

```
CREATE TABLE table1(c1 DATE);
```

要装载表，请使用如下语句：

```
LOAD TABLE table1 (c1 ASCII(10)) FROM ...
```

如果原始数据格式已经过优化（例如 YYYY-MM-DD），装载速度将快得多。

以下各节更详细地说明了转换选项。

ASCII 输入转换

将 ASCII 输入数据转换为二进制。

使用 ASCII 转换选项可以：

- 将 ASCII 输入数据转换为二进制类型，并指定输入列的宽度，以便正确读取该列中的数据，或者
- 当输入列的宽度不同于目标列的宽度时，将 ASCII 数据插入到 ASCII 数据类型列。您可以使用该选项指定应针对每列读取的输入数据量。

您可以将此选项用于任何 Sybase IQ 数据类型，带有 1、2 或 4 个前缀字节以及带有列分隔符的数据类型。

截断 VARCHAR 和 CHAR 列的数据

如果输入列宽度大于目标列宽度，Sybase IQ 将在插入时截断数据。

如果输入数据宽度小于目标列宽度，对于 CHAR 或 VARCHAR 数据类型，Sybase IQ 将在插入时在表中用空白填充数据。

向 VARCHAR 列插入可变宽度将不会剪裁尾随空白，而向 VARCHAR 列插入固定宽度则会剪裁尾随空白。例如，假定要插入到 varcolumn 表的 varcolumn 列。以下命令将构成一个固定宽度的插入，由于您显式指定应包含两个空白（此处通过 __ 表示），因此不会剪裁此插入中的值：

```
INSERT INTO varcolumn VALUES ('box__')
```

如果您改用分隔输入从平面文件插入相同值，此插入的宽度则是可变的，因此将剪裁尾随空白。

下表说明 ASCII 转换选项如何处理 Sybase IQ 数据类型。本示例从 ASCII 平面文件 shipinfo.t 向 Sybase IQ 表 lineitem 插入数据，并概述了输入数据和表的内容和格式。

表 29. 输入文件转换示例

文件 <i>shipinfo.t</i>			表 <i>lineitem</i>		
列	格式	宽度	列	数据类型	宽度
<i>l_shipmode</i>	CHAR	15	<i>l_shipmode</i>	VARCHAR	30
<i>l_quantity</i>	ASCII	8	<i>l_quantity</i>	INT	4

对于 *l_shipmode* 列，您需要将 ASCII 数据插入到 ASCII 列（数据类型为 VARCHAR）。请注意，两个列的宽度不同。为了使此列及后续 *l_quantity* 列上的插入准确无误，请指定 *l_shipmode* 列的宽度，以便在正确位置读取正确的输入数据量。

对于 *l_quantity* 列，您需要将 ASCII 数据插入到二进制列（INT 数据类型）。为了使此列上的插入准确无误，必须将输入数据转换为二进制类型，并指明输入列的宽度。

下面的 UNIX 示例显示了此命令。

```
LOAD TABLE lineitem(      l_shipmode ASCII(15),      l_quantity
ASCII(8), FILLER(1)) FROM '/dl/MILL1/shipinfo.t' PREVIEW ON
```

替换空值或空白字符

Sybase IQ 支持零长度 VARCHAR 数据。如果 VARCHAR 单元的长度为零并且该单元不为空值，则将获得一个零长度单元。

对于所有其它数据类型来说，如果单元长度为零，则 Sybase IQ 将插入空值。

这种零长度字符数据的处理方式属于 ANSI 行为。如果需要非 ANSI 行为，请参见《参考：语句和选项》中的 **NON_ANSI_NULL_VARCHAR** 选项。

DATE 选项

使用 **DATE** 转换选项可以将用固定格式存储的 ASCII 数据插入到 DATE 列。

此选项将 ASCII 数据输入转换为二进制类型并指定输入数据的格式。（DATE 格式在内部用于解释输入；它不会影响数据的存储或输出格式。）有关详细信息，请参见 **ASCII 转换格式**。

示例

在以下 Windows 示例中，*l_shipdate* 列的数据从指定格式转换为二进制类型。单字节 **FILLER** 将跳过输入文件中的回车。

```
LOAD TABLE lineitem(      l_orderkey NULL(ZEROS) ASCII(4),
l_partkey ASCII(3),      l_shipdate DATE('MM/DD/YY'),      l_suppley
ASCII(5), FILLER(1)) FROM 'C:\\MILL1\\shipinfo.t' PREVIEW ON
```

DATE 格式

指定输入数据格式，使用 y 或 Y 表示年份，m 或 M 表示月份，d 或 D 表示日，j 或 J 表示儒略日。

格式字符串的长度即为输入列的宽度。

表 30. 设置日期格式

选项	含义
yyyy 或 YYYY yy 或 YY	表示年份数字。缺省值为 1900。
mm 或 MM	表示月份数字。请始终对月份数字使用前导零（如有必要），例如，“05”表示五月。如果从 DATE 值中省略月份，则将日期视为儒略日。如果仅输入月份，如“03”，则 Sybase IQ 应用缺省年份和日期，并将其转换为“1900-03-01”。
dd 或 DD jjj 或 JJJ	表示日期数字。缺省日期为 01。请始终对日期数字使用前导零（如有必要），例如，“01”表示第一天。J 或 j 表示一年中的儒略日（1 至 366）。

输入时，忽略格式代码的大小写。

输出时，格式代码的大小写具有以下影响：

- 混合大小写（如“Dd”）表示不会补零。
- 相同大小写（如“DD”或“dd”）表示将补零。

例如，时间 17:23:03.774 使用缺省时间格式，而 17:23:3.774 使用“HH:NN:Ss.SSS”。

下表中的示例用于显示日期输入数据的格式以及如何使用 DATE 转换选项指定格式。此表后给出了指定日期的一般规则。

表 31. DATE 格式选项示例

输入数据	格式规范
12/31/09	DATE ('MM/DD/YY')
12-31-09	DATE ('MM-DD-YY')
20091231	DATE ('YYYYMMDD')
12/09	DATE ('MM/YY')
2009/123	DATE ('YYYY/JJJ')

- DATE 规范必须位于括号中，并用单引号或双引号引起来。
- Sybase IQ 仅存储年份、月份和日的数字；它不会存储可能在输入数据中显示的任何其它字符。但是，如果输入数据包含其它字符（如斜杠 (/)、短划线 (-) 或空白）

来分隔月份、日和年份，**DATE** 格式必须显示这些字符的显示位置，以便忽略它们。

- 使用除 Y、M、J 或 D 以外的任何其它字符表示希望 Sybase IQ 跳过的分隔符。您甚至可以使用空白。
- 如果 **DATE** 格式仅包含年份和该年中的日期数字，则 Sybase IQ 将此日期视为儒略日。例如，2009-33 表示 2009 年中的第 33 天，也即 2009 年 2 月 2 日。
- 如果仅使用两位数指定年份，例如“5/27/32”，Sybase IQ 则将此数字转换为 19yy 或 20yy，具体取决于年份和 **NEAREST_CENTURY** 选项的设置。

NEAREST_CENTURY 设置	指定年份	假定年份
缺省值 (50)	00-49	2000-2049
	50-99	1950-1999
0	any	二十世纪初
100	any	二十一世纪初

另请参见

- 指定 DATETIME 转换的格式（第 268 页）

DATETIME 转换选项

使用 **DATETIME** 转换选项可以将用固定格式存储的 ASCII 数据插入到 **TIME**、**TIMESTAMP** 或 **DATETIME** 列。

此选项将 ASCII 数据输入转换为二进制类型并指定输入数据的格式。（**DATETIME** 格式在内部用于解释输入；它不会影响数据的存储或输出格式。）有关详细信息，请参见 **ASCII 转换格式**。

注意： 为了与早期版本兼容，您可以指定包含 **DATETIME** 数据的列。但是，此类数据在内部存储为等效格式 **TIMESTAMP**。

语法如下：

```
DATETIME ('input-datetime-format')
```

在以下 UNIX 示例中，采用斜杠作为输入数据日期部分中的分隔符，并采用冒号作为时间部分中的分隔符：

```
LOAD TABLE lineitem(      l_quantity ASCII(4),      l_shipdate
DATETIME('MM/DD/YY hh:mm:ss'), FILLER(1)) FROM '/dl/MILL1/tt.t'
PREVIEW ON
```

在以下 UNIX 示例中，**FILLER(1)** 子句防止 Sybase IQ 在 **DATETIME** 列之后的下一列 (**VWAP**) 中插入空值：

```
LOAD TABLE snapquote_stats_base SYMBOL '\x09', snaptime
DATETIME('MM/DD/YY hh:mm:ss'), FILLER(1)) VWAP '\x09', RS_DAY
'\x09', FROM '/dl/MILL1/tt.t' PREVIEW ON
```

在以下 UNIX 示例中，目标列包含 TIME 数据，但输入数据为 DATETIME。您将使用 **DATETIME** 转换选项，并使用 **FILLER** 跳过日期部分。

```
LOAD TABLE Customers(      open_time DATETIME('hh:mmaa'),
close_time DATETIME('hh:mmaa'), FILLER(9)) FROM '/d1/MILL1/tt.t'
PREVIEW ON
```

指定 DATETIME 转换的格式

指定 DATETIME 输入的格式。

使用以下表示法指定格式：

- Y 或 y 表示年份
- M 或 m 表示月份
- D 或 d 表示日
- H 或 h 表示小时
- N 或 n 表示分钟（当使用冒号作为分隔符时，也接受 mm）
- S 或 s 表示秒和秒的小数部分

格式字符串的长度即为输入列的宽度。下表说明时间格式设置选项。

表 32. 设置时间格式

选项	含义
hh HH	表示小时。小时基于 24 小时制。请始终对小时使用前导零（如有必要），例如，“01”表示凌晨 1 点。“00”也是有效值，表示中午 12 点。
nn	表示分钟。请始终对分钟使用前导零（如有必要），例如，“08”表示 8 分钟。
ss[.ssssss]	表示秒和秒的小数部分。
aa	表示 a.m. 或 p.m. 标记。
pp	仅当需要时表示 p.m. 标记。（这与 Sybase IQ 12.0 之前的版本不兼容；以前，pp 与 aa 同义。）
hh	Sybase IQ 假设分钟和秒钟均为零。例如，如果您输入的 DATETIME 值为“03”，则 Sybase IQ 将该值转换为“03:00:00.0000”。
hh:nn 或 hh:mm	Sybase IQ 假设秒钟为零。例如，如果您输入的时间值为“03:25”，则 Sybase IQ 将该值转换为“03:25:00.0000”。

下表中的示例用于显示时间输入数据的格式以及如何指定 **DATETIME** 选项的格式。此表后给出了指定时间的一般规则。

表 33. DATETIME 格式选项

输入数据	格式规范
12/31/00 14:01:50	DATETIME ('MM/DD/YY hh:nn:ss')
123100140150	DATETIME ('MMDDYYhhnnss')
14:01:50 12-31-00	DATETIME ('hh:mm:ss MM-DD-YY')
12/31/00 14:01:12.456	DATETIME ('MM/DD/YY hh:nn:sssssss')
12/31/00 14:01:12.3456	DATETIME ('MM/DD/YY hh:mm:sssssss')
12/31/00 02:01:50AM	DATETIME ('MM/DD/YY hh:mm:ssaa')
12/31/00 02:01:50pm	DATETIME ('MM/DD/YY hh:mm:sspp')

- 时间部分的规范字母必须包括在括号中并用单引号或双引号引起来。
- 输入数据可针对秒包含多至九个数位（包括浮动小数点），以便允许小数秒数。在输入和查询时，小数点将浮动，因此，最多可以指定六位小数位数。但是，**Sybase IQ** 始终仅存储六位小数位数和两位整秒位数 (ss.ssssss)。不允许使用更多的小数位数。
- 分隔符用于各时间元素之间。您可以使用任何字符作为分隔符，包括空白。本示例使用 “:”（冒号）。
- **Sybase IQ** 仅存储小时、分钟和秒的数字；它不会存储可能在输入数据中显示的任何其它字符。但是，如果数据包含其它字符（如冒号 [:] 或空白）来分隔小时、分钟和秒，格式规范的时间部分必须显示这些字符的显示位置，以使 **Sybase IQ** 跳过这些字符。
- 要指明特定值是 a.m. 还是 p.m.，输入数据必须在一致位置包含大写或小写 “a” 或 “p”。要指明 **Sybase IQ** 应在哪个位置查找 a.m. 或 p.m. 标记，请在格式规范中的相应位置放置仅含小写字母的 “aa” 或 “pp”。“aa” 指定将始终表示 a.m./p.m.，而 “pp” 指定仅在需要时表示 pm。
- 格式规范必须包含与输入中的每个字符相匹配的字符；不能用格式规范中的 “m” 匹配输入中的 “m”，因为 “m” 已用于表示分钟。
- 在时间部分中，如果未指定小时、分钟或秒，则 **Sybase IQ** 假定每一项分别为 0。

另请参见

- DATE 格式（第 266 页）

空值数据转换

当插入到 **Sybase IQ** 列索引时，使用 **NULL** 转换选项可以将输入数据中的特定值转换为空值。

此选项可以用于任何列，但该列必须允许空值。您可以对任何 **Sybase IQ** 数据类型指定此转换选项。

语法如下：

```
NULL ({BLANKS | ZEROS | literal' ['literal']...})
```

其中:

- **BLANKS** 表示将空白转换为空值。
- **ZEROS** 表示将二进制零转换为空值。
- **literal** 表示将所有指定文字转换为空值。指定文字必须与输入文件中的值完全匹配, 包括前导空白和/或尾随空白, 以使 Sybase IQ 能够将该值识别为匹配项。最多可以列出 20 个实际值。

您可能需要对同一列使用其它转换选项。例如, 要将 ASCII 数据插入到以二进制格式存储的 INT 列中, 并在插入时将输入数据中的空白转换为空值, 请使用 **ASCII** 转换选项将输入转换为二进制类型, 并使用 **NULL** 转换选项将空白转换为空值。

下面是一个 Windows 示例:

```
LOAD TABLE lineitem( l_orderkey NULL(ZEROS) ASCII(4),  
l_partkey ASCII(3), l_shipdate date('MM/DD/YY'), l_suppkey  
ascii(5), FILLER(1)) FROM 'C:\\MILL1\\tt.t' PREVIEW ON
```

影响数据显示的其它因素

只要 Sybase IQ 要求在查询或插入时将一种数据类型显式或隐式转换为另一种数据类型, 则始终会截断结果。

下述内容对此类情况进行了说明:

- 当从较高标度显式转换为较低标度时, Sybase IQ 截断结果中的值。例如, 如果在查询时对存储为 4 位标度的列值执行 **CAST**, 将其转换为 2 位标度, 则诸如 2.4561 之类的值将变为 2.45。
- 如果 Sybase IQ 在插入时从较高标度隐式转换为较低标度, 则会在将数据插入到表之前截断值。例如, 如果从一个数据类型为 **NUMERIC(7,3)** 的表插入到另一个数据类型为 **DECIMAL(12,2)** 的表中, 诸如 2.456 之类的值将变为 2.45。
- 在使用《参考: 构件块、表和过程》>“SQL 数据类型”>“数值数据类型”中定义的规则确定标度后, 如果算术运算导致标度高于预定标度, 则 Sybase IQ 会截断结果, 使其与预定标度相符。

如果结果要求舍入而不是截断值, 则应在命令中使用 **ROUND** 函数。但是, 对于插入, **ROUND** 函数只能作为其查询表达式的一部分。

数值数据的最大精度为 126。

匹配 Adaptive Server Enterprise 数据类型

下表显示哪些 Sybase IQ 数据类型与 Adaptive Server Enterprise 数据类型兼容。

以下是一些一般的规则:

- Sybase IQ 字符串类型接受任何 Adaptive Server Enterprise 字符串类型。
- Sybase IQ 精确数值类型接受任何 Adaptive Server Enterprise 数值类型。但是，如果 Sybase IQ 数据类型含有的数据量小于 Adaptive Server Enterprise 类型，则值转换为空值（例如，当从基础数据库向表中插入数据时）。
- Sybase IQ 日期/时间类型接受任何 Adaptive Server Enterprise 日期/时间类型。

另请参见

- 在插入时转换数据（第 257 页）
- IQ 中的显式数据转换（第 259 页）

不支持的 Adaptive Server Enterprise 数据类型

并非所有的 Adaptive Server Enterprise 数据类型都受支持。

此版本的 Sybase IQ 不支持下列 Adaptive Server Enterprise 数据类型：

- nchar、nvarchar
- unichar、univarchar、unitext
- text
- image
- unsigned smallint
- native Java data types
- XML data type

请注意下列各项：

- Sybase IQ 通过二进制大对象 (BLOB) 和字符大对象 (CLOB) 数据类型支持 Adaptive Server Enterprise 文本类型和图像类型。有关详细信息，请参见《Sybase IQ 中的非结构化数据分析》。
- 在 Sybase Central 中，您可以仅在 SYSTEM 数据库空间 (Catalog 存储) 中创建具有特定数据类型的列。在创建表向导中，选择“在系统存储中创建表”。
- Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 DATE、TEXT、UNSIGNED SMALLINT、NCHAR、NVARCHAR、UNICHAR、UNIVARCHAR 或 UNITEXT，但可以使用 INSERT 语句的 LOCATION 语法从数据类型为 DATE、TEXT、UNSIGNED SMALLINT、NCHAR、NVARCHAR、UNICHAR、UNIVARCHAR 或 UNITEXT 的 Adaptive Server Enterprise 数据库列插入数据。

Adaptive Server Enterprise 等效数据类型

下表显示 Adaptive Server Enterprise 精确数值类型以及 Sybase IQ 等效项。

表 34. 整数数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
int	INT、BIGINT、UNSIGNED INT、UNSIGNED BIGINT 或 NUMERIC	Sybase IQ 不允许带有标度的整数，如 INT(7,3)。INT(<i>precision, scale</i>) 格式的数据转换为 NUMERIC(<i>precision, scale</i>)。这与 Sybase IQ 12.0 版之前的版本以及 Adaptive Server Enterprise 不相同，在这两种软件中，int 数据类型可以是 -2,147,483,648 和 2,147,483,647 之间的值，且包括这两个数。 要处理更大的整数值，可以使用 BIGINT、无符号整数 (UNSIGNED INT) 或 UNSIGNED BIGINT 数据类型。使用 UNSIGNED INT，最后一位将用作值的一部分。没有正负指示；所有数字都假定为正数，因此值最大为 4,294,967,295。
numeric	具有相应精度的 DECIMAL 或 NUMERIC	如果定义的 Sybase IQ 数据类型的精度太小，无法存储 Adaptive Server Enterprise 值，则该值转换为空值。
decimal	具有相应精度的 DECIMAL 或 NUMERIC	请参见上述内容。
smallint	SMALLINT 或 NUMERIC	Sybase IQ SMALLINT 不允许精度和标度。Adaptive Server Enterprise smallint(<i>precision, scale</i>) 转换为 NUMERIC(<i>precision, scale</i>) 请参见上述 INT。
tinyint	TINYINT	Sybase IQ TINYINT 列不允许精度和标度。Adaptive Server Enterprise tinyint(<i>precision, scale</i>) 转换为 NUMERIC(<i>precision, scale</i>)。请参见上述 INT。
bit	BIT	
unsigned smallint	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 unsigned smallint，但可以使用 INSERT...LOCATION 从数据类型为 unsigned smallint 的 Adaptive Server Enterprise 数据库列插入数据。

下表显示 Adaptive Server Enterprise 近似数据类型以及 Sybase IQ 等效项。

表 35. 近似数值数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
float (precision)	FLOAT (precision)	IQ 对 FLOAT 支持更大精度 HNG 索引不允许使用 FLOAT、REAL 或 DOUBLE 数据。
double precision	DOUBLE	
real	REAL	

下表显示 Adaptive Server Enterprise 字符数据类型及其 Sybase IQ 等效项。

表 36. 字符数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
char	CHAR	除了 Sybase IQ 可以处理空值以外，Sybase IQ 和 Adaptive Server Enterprise 字符 (char 或 CHAR) 数据类型相同。如果要使 Sybase IQ CHAR 列与 Adaptive Server Enterprise char 列完全匹配，请将 Sybase IQ 列指定为非空值。Sybase IQ 缺省值允许使用空值。允许空值的 Adaptive Server Enterprise char 列内部转换为 varchar。
varchar	VARCHAR	请参见上述 char 注释。
nchar	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 nchar，但可以使用 INSERT...LOCATION 从数据类型为 nchar 的 ASE 数据库列插入数据。
nvarchar	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 nvarchar，但可以使用 INSERT...LOCATION 从数据类型为 nvarchar 的 Adaptive Server Enterprise 数据库列插入数据。
text	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 text，但可以使用 INSERT...LOCATION 从数据类型为 text 的 Adaptive Server Enterprise 数据库列插入数据。
unicar	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 unicar，但可以使用 INSERT...LOCATION 从数据类型为 unicar 的 Adaptive Server Enterprise 数据库列插入数据。

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
univarchar	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 univarchar, 但可以使用 INSERT...LOCATION 从数据类型为 univarchar 的 Adaptive Server Enterprise 数据库列插入数据。
unitext	不支持	Sybase IQ 不支持 Adaptive Server Enterprise 数据类型 unitext, 但可以使用 INSERT...LOCATION 从数据类型为 unitext 的 Adaptive Server Enterprise 数据库列插入数据。

下表显示 Adaptive Server Enterprise money 数据类型以及 Sybase IQ 等效项。

表 37. 货币数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
money	NUMERIC(19,4)	money 数据会隐式转换为 NUMERIC(19,4)。
smallmoney	NUMERIC(10,4)	

下表显示 Adaptive Server Enterprise DATE/TIME 数据类型以及 Sybase IQ 等效项。

表 38. DATE/TIME 数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
datetime	TIMESTAMP、DATE 或 TIME	<p>Adaptive Server Enterprise datetime 列维护日期值的日期和时间，使虚拟日期 0/0/0000 的基准日期前后的日期数字保持为 4 字节，使日期时间保持为 8 字节，并精确到百万分之一秒以内。Sybase IQ TIMESTAMP（或 DATETIME）列使日期值的日期和时间保持为两个 4 字节的整数：自 1/1/0 以来的日期数字为 4 字节，日期时间也为 4 字节（基于 24 小时制），并精确到百万分之一秒以内。Sybase IQ 自动处理转换。</p> <p>Sybase IQ 还有一种单独的 DATE 数据类型，即单个 4 字节整数。如果要从 SQL Server 或 Adaptive Server Enterprise datetime 列仅提取日期，可以使用 Sybase IQ DATE 数据类型执行此操作。为此，请定义一个与 Adaptive Server Enterprise datetime 列同名的 Sybase IQ DATE 列。Sybase IQ 自动选取 datetime 值的相应部分。</p>
smalldatetime	TIMESTAMP、DATETIME、DATE 或 TIME	<p>根据 Sybase IQ 中的 TIMESTAMP（或 DATETIME）数据类型定义 Adaptive Server Enterprise smalldatetime 列。Sybase IQ 以适当方式处理转换。与常规 datetime 相同，如果要从 Adaptive Server Enterprise smalldatetime 列中仅提取日期，请使用 Sybase IQ DATE 数据类型执行此操作。</p>
date	date	<p>可以使用 INSERT...LOCATION 从数据类型为 date 的 Adaptive Server Enterprise 数据库列插入数据。</p>
time	time	<p>该 Sybase IQ 数据类型是每日时刻，其中包含小时、分钟、秒钟和秒的小数部分。小数部分存储为 6 位小数。time 值需要 8 字节的存储空间。</p> <p>Adaptive Server Enterprise 数据类型 time 介于 00:00:00:000 和 23:59:59:999 之间。您可以使用军用时间或者使用 12AM 代表中午并使用 12PM 代表午夜。time 值必须包含一个冒号，或者 AM 或 PM 标记。AM 或 PM 大小写不限。time 值需要 4 字节的存储空间。</p> <p>可以使用 INSERT...LOCATION 从数据类型为 time 的 Adaptive Server Enterprise 数据库列插入数据。</p>

下表显示 Adaptive Server Enterprise binary 数据类型以及 Sybase IQ 等效项。

表 39. 二进制数据类型

Adaptive Server Enterprise 数据类型	Sybase IQ 数据类型	注释
binary	BINARY	Sybase IQ 在所有 BINARY 列中填充尾随零。创建的 BINARY 列的长度应始终包含偶数个字符。 HNG 索引不允许使用 BINARY 数据。
varbinary	VARBINARY	Sybase IQ 不会填充或截断 VARBINARY 列中的尾随零。创建的 VARBINARY 列的长度应始终包含偶数个字符。 HNG 索引不允许使用 VARBINARY 数据。

由于不支持下列 Adaptive Server Enterprise 数据类型，必须省略具有这些数据类型的列：

- nchar、nvarchar
- univar、univarchar
- unsigned smallint
- native Java data types

这同样适用于任何自定义 Adaptive Server Enterprise 数据类型。

数据导入时的转换错误

当从外部来源装载数据时，数据中可能有错误。

例如，可能有某些日期不是有效的日期，某些数字不是有效的数字。

CONVERSION_ERROR 数据库选项允许您忽略这些转换错误，该选项会将它们转换为空值。

有关设置 **dbisql** 数据库选项的信息，请参见《参考：语句和选项》中的 **SET OPTION** 语句。

调优数据的 bulk load 操作

将大量数据装载到数据库可能会需要很长时间，并且会占用大量磁盘空间。您可以采用某些手段来节省时间。

另请参见

- 数据装载 (第 14 页)
- 调查锁争用的工具 (第 350 页)
- 调度数据库更新 (第 229 页)

定义数据库期间的装载性能

数据库、表和索引的定义方式可以对装载性能产生极大影响。

另请参见

- 磁盘空间不足 (第 455 页)
- IQ 主存储区和 IQ 临时存储空间管理 (第 146 页)
- IQ_SYSTEM_MAIN 数据库空间 (第 128 页)
- 主 IQ 存储块消息 (第 474 页)
- 监控磁盘空间使用情况 (第 459 页)
- 处理问题 (第 463 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- Sybase IQ 停止处理或停止响应 (第 449 页)

不同的值数目

根据 `MINIMIZE_STORAGE` 数据库选项设置以及在创建数据库和表时指定的参数，Sybase IQ 可优化对大小各异的不同的值集合的数据装载。

影响装载优化的参数包括：

- **UNIQUE** 和 **IQ UNIQUE** 选项以及列的数据类型和宽度，所有这些参数都在 **CREATE TABLE** 或 **ALTER TABLE** 命令中指定。
- **IQ PAGE SIZE**，该参数在 **CREATE DATABASE** 命令中指定。

另请参见

- 创建表的准则 (第 171 页)
- 锁定的工作方式 (第 347 页)
- **IQ PAGE SIZE** 参数准则 (第 143 页)
- 有效数据的规则和检查 (第 317 页)

索引创建

为了充分利用该系统资源，请在装载数据前创建所有所需索引。虽然您始终可以在稍后添加新索引，但是一次性装载所有索引的速度要快得多。

内存调整

在某些平台上，您可以设置命令行选项，以便调整可用内存量。

增加内存可以提高装载性能。

装载时间环境调整

您可以在装载数据时调整若干因素，以提高装载性能。

- 当有权访问 ASCII 或二进制格式的原始数据时，请使用 **LOAD TABLE** 命令，特别是超过一百行的所有装载。**LOAD TABLE** 命令是最快的插入方法。
- 在从平面文件装载时，如果系统让您在二进制数据和字符数据之间进行选择，请选择使用前者。这可以避免转换开销并减少 I/O，从而提高性能。
- 对 **LOAD TABLE** 命令选项进行相应设置。

如果要记录忽略的完整性约束违规，Sybase 强烈建议将 **LOAD TABLE IGNORE CONSTRAINT** 选项限制设置为非零值。记录过多的违规数会影响装载性能。

- 将数据文件与数据库文件放在不同的物理磁盘驱动器上，以避免装载过程中过多的磁头移动。
- 增加数据库高速缓存的大小。为装载提供充足内存是提高性能的关键因素。使用数据库启动实用程序 **start_iq** 命令行选项 **iqmc** 和 **iqtc** 可以增加高速缓存大小；请参见《实用程序指南》>“**start_iq** 数据库服务器启动实用程序”。为使这些选项生效，必须重新启动服务器。
- 为数据库或服务器调整缓冲区分区的程度，以避免出现锁争用。缺省情况下，将启用基于 CPU 数目进行缓冲区分区，并且可通过设置 **-iqpartition** 服务器命令行选项或 **Cache_Partitions** 数据库选项进行调整。
- 确保一次只有一个用户更新数据库。虽然多个用户可同时向不同表插入数据，但是并发更新可降低性能。
- 在使用率较低时安排主要更新。虽然多个用户可以在更新表时进行查询，但是查询用户需要 CPU 周期、磁盘空间和内存。您需要使用这些资源，以使插入更快。
- 如果要使用 **INSERT** 语句，请在服务器所在的相同计算机上运行 **Interactive SQL** 或客户端应用程序（如有可能）。通过网络装载数据将增加额外的通信开销。这可能意味着需要在下班时间装载新数据。

如果要使用 **INSERT...LOCATION** 通过网络从远程 Adaptive Server Enterprise 数据库装载大量文本或批量数据，请使用 **LOCATION** 子句的 **PACKETSIZE** 参数，以便增加 TDS 包大小。此更改会极大地提高装载性能。

另请参见

- 使用 **LOAD TABLE** 语句执行 bulk load 操作（第 238 页）
- 调查锁争用的工具（第 350 页）

增量装载中的 IQ 主存储区空间使用情况

增量装载可以修改要装载的表中的大量页面。

因此，将在 **main** 数据库空间中临时控制这些页面的版本，直到已提交事务且检查点可以释放旧版本为止。如果在从同一表中删除后执行增量装载，此版本控制功能可能非常流行。其原因在于，Sybase IQ 在缺省情况下重用删除记录中的行 ID。

为了减少带有版本的页面的空间使用量，请将 **APPEND_LOAD** 选项设置为 ON，这样，IQ 可在表末追加新数据。缺省情况下，**APPEND_LOAD** 为 OFF。

APPEND_LOAD 选项适用于 **LOAD**、**INSERT...SELECT** 和 **INSERT...VALUES** 语句。

装载期间的线程使用情况

如有可能，Sybase IQ 将使用多线程以提高装载性能。

只要具有充足的可用资源（内存和/或线程），固定宽度的装载和全部宽度且用行分隔的装载（其大小和限制为 0）将以完全多线程方式运行。如果资源不足，装载将在单个线程中运行，并在 .iqmsg 文件中显示以下消息：

```
The insert to the table will be single threaded.
```

如果具有充足资源，没有行分隔数据的可变宽度装载在最佳情况下将仅以部分多线程方式运行。对于部分多线程装载，.iqmsg 文件中将显示以下消息：

```
Portions of the insert/load will be multithreaded.
```

对表行的更改

要更新一行或多行，请使用 **UPDATE** 语句。新数据可以是指定的常量或表达式，或者是从其它表中拖出的数据。

像在所有数据修改语句中一样，您一次只能更改一个表或视图中的数据。

如果 **UPDATE** 语句违反了完整性约束，则不会进行更新并将显示一条错误消息。例如，如果正被添加的一个值是错误的数据类型，或者如果它违反为所涉及的某个列或数据类型定义的约束，则将不会进行更新。

此语法的简化版本如下：

```
UPDATE table-name SET column_name = expression WHERE search-condition
```

示例

如果 Newton Ent. 公司（在演示数据库的 Customers 表中）被 Einstein, Inc. 接管，则您可以使用如下语句更新该公司的名称：

```
UPDATE Customers
SET company_name = 'Einstein, Inc.'
WHERE company_name = 'Newton Ent.'
```

您可以在 **WHERE** 子句中使用任何条件。如果不确定如何输入公司名称，则您可以使用如下所示的语句尝试更新名为 Newton 的任何公司：

```
UPDATE Customers
SET company_name = 'Einstein, Inc.'
WHERE company_name LIKE 'Newton%'
```

搜索条件不需要引用被更新的列。Newton Entertainments 的公司 ID 是 109。由于 ID 值是表的主键，所以您可以确定使用以下语句更新正确的行：

```
UPDATE Customers
SET company_name = 'Einstein, Inc.'
WHERE id = 109
```

SET 子句

SET 子句指定要更新的列和它们的新值。**WHERE** 子句确定要更新的行。如果不具有 **WHERE** 子句，则将用 **SET** 子句中提供的值更新所有行中指定的列。

您可以在 **SET** 子句中提供数据类型正确的任何表达式。

WHERE 子句

WHERE 子句指定要更新的行。例如，以下语句用 **Extra Large Tee Shirt** 替换 **One Size Fits All Tee Shirt**。

```
UPDATE Products
SET size = 'Extra Large'
WHERE name = 'Tee Shirt'
      AND size = 'One Size Fits All'
```

FROM 子句

您可以使用 **FROM** 子句从一个或多个表中取出数据并将它们放入正在更新的表中。您还可以通过 **FROM** 子句针对其它表使用选择条件，以便控制更新的行。

删除数据的方法

使用 **DELETE**、**DROP TABLE** 和 **TRUNCATE TABLE** 语句可以删除数据。

要删除数据库中的数据，您可以执行下列任一操作：

- 使用 **DELETE** 语句从表中删除满足指定条件的所有行。
- 使用 **DROP TABLE** 语句删除整个表，包括所有数据行。
- 使用 **TRUNCATE TABLE** 语句删除表中的所有行，但不删除表定义。

用于删除的空间

当使用 **DELETE** 或 **TRUNCATE TABLE** 语句时，鉴于 **Sybase IQ** 对数据页版本的存储方式，您可能需要向数据库添加空间。

当使用 **DROP TABLE** 时，由于不需要额外的版本页，因此，您无需添加空间。

管理用户 ID 和权限

必须为数据库的每个用户都分配一个唯一的用户 ID：即连接到数据库时要键入的名称。本章介绍如何管理用户 ID。

另请参见

- 修改数据的权限（第 228 页）
- 安全概述（第 4 页）
- 事务和版本控制（第 337 页）

数据库权限概述

通过适当管理用户 ID 和权限，您可以有效地开展工作，同时保持数据库内相应信息的安全性和隐私。

使用 SQL 语句可以将用户 ID 指派给数据库的新用户，授予和撤销数据库用户的权限，以及显示用户的当前权限。您还可以使用 Sybase Control Center 来分配用户 ID，并授予和撤销权限；有关说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

权限授予创建、修改、查询、使用或删除数据库对象（如表、视图、用户等）的能力。授权授予在数据库级别执行任务（如备份数据库）的能力。

数据库权限是按用户 ID 指派的。在本章中，*用户*一词是用户 ID 的同义词。但请记住，权限是针对每个用户 ID 授予和撤销的。

设置单个用户 ID

即使不存在有关多用户数据库的安全问题，但为每位用户设置单个用户 ID 仍好处多多。如果建立具有适当权限的组，单个用户 ID 的管理开销会很低。用户组将在本章的后面部分加以讨论。

使用单个用户 ID 的原因有：

- 使用单个用户 ID，网络服务器屏幕和 Sybase Central 中的连接列表都会更为有效，因为您可以分辨哪些连接对应哪些用户。
- 备份日志会标识创建备份的用户 ID。
- 消息日志会显示每个数据库连接的用户 ID。

所有权限都是可继承的（从用户所属的组），而授权只有部分是可继承的。

除了拥有完全管理特权的 DBA 以外，每个权限都有权执行某些类型的任务。

另请参见

- 连接学习路线图 (第 42 页)

继承权限

某些权限是通过组成员资格继承的。

表 40. 权限

权限	通过组成员资格继承
BACKUP	否
DBA	否
MULTIPLY ADMIN	是
OPERATOR	是
PERMS ADMIN	是
PROFILE	是
READCLIENTFILE	是
READFILE	是
REMOTE DBA	否
RESOURCE	否
SPACE ADMIN	是
USER ADMIN	是
VALIDATE	否
WRITECLIENTFILE	是

有关 MULTIPLY ADMIN 权限，请参见《使用 Sybase IQ Multiplex》。

某些权限是由 SQL Anywhere 提供的。请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “管理用户 ID 和权限” > “权限概述”。

注意： 此参考指向 SQL Anywhere 文档。

如果针对在 Sybase IQ 15.2 之前的版本中创建的数据库授予或撤消权限 MULTIPLY ADMIN、OPERATOR、PERMS ADMIN、SPACE ADMIN、USER ADMIN，会返回错误代码 -1347：

系统不支持在当前的此数据库上使用权限“%1”。

另请参见

- 权限概述 (第 283 页)

继承的权限

下表列出了您可以分配给用户的权限，以及它们是否是通过组成员资格继承的。

权限	通过组成员资格继承
ALL	是
ALTER	是
CONNECT	否
DELETE	是
INSERT	是
INTEGRATED LOGIN	否
KERBEROS LOGIN	否
REFERENCES	是
SELECT	是
UPDATE	是

权限概述

以下几节提供了有关权限的详细信息。

另请参见

- 继承权限 (第 282 页)
- 如何确定用户权限 (第 311 页)

DBA 权限概述

创建数据库时，会创建一个用户 ID。

缺省情况下，DBA 的口令最初设置为“sql”。要覆盖缺省用户名或口令，请使用带 **DBA USER** 或 **DBA PASSWORD** 子句的 **CREATE DATABASE** 语句。DBA 用户 ID 已被自动授予数据库内的 DBA 权限。此级别的权限可让 DBA 用户 ID 在数据库中执行任何活动：创建表、更改表结构、创建新的用户 ID、撤消用户的权限，等等。

注意： 为了确保数据库安全性和责任，请避免使用“dba”之类的通用名作为第一个用户 ID，而应改用真实用户的登录名和强口令。

具有 DBA 权限的用户

虽然 DBA 权限可以被授予或转让给其它用户 ID，但 Sybase IQ 文档还是假定 DBA 用户 ID 是数据库管理员，而 DBA 这一缩写既表示 DBA 用户 ID，也表示任何拥有 DBA 权限的用户 ID。

为了防止有效 DBA 用户丢失口令，请创建一个或多个额外的 DBA 帐户（使用随机生成的用户名和口令）并锁定这些凭据。如果有效的 DBA 口令丢失，请使用额外的凭据之一登录到该 DBA 帐户，然后重置原始帐户口令。

警告！ 请勿从 Multiplex 数据库中删除 DBA 用户，否则会导致数据库无效。

示例

以下示例演示如何赋予非 DBA 用户能力，使其可以执行需要 DBA 特权才能执行的命令。它将创建一个策略，可让非 DBA 用户 (user1) 执行备份。

```
CREATE PROCEDURE "DBA".do_backup()
BEGIN
  BACKUP DATABASE
  CRC ON
  ATTENDED OFF
  BLOCK FACTOR 4
  FULL
  TO 'fileA' SIZE 2000
  TO 'fileB' SIZE 2000
  TO 'fileC' SIZE 2000;
END;
GRANT EXECUTE ON "DBA".do_backup TO user1;
```

添加新用户

DBA 可以为数据库添加新用户。此外，系统还会向新用户授予在数据库中执行任务的权限。有些用户只需要使用 SQL 查询来查看数据库信息，有些用户可能需要向数据库添加信息，而有些用户则可能需要修改数据库结构。虽然 DBA 的职责可以移交给其它用户 ID，但由于 DBA 具有 DBA 权限，因此要负责数据库的全面管理。

DBA 拥有创建数据库对象和为其它用户 ID 指派这些对象的所有权的权限。

区分大小写的数据库中的 DBA 用户 ID

用户 ID 和口令实际上是数据库对象。有关用户 ID 和口令是否区分大小写的详细信息，请参见《参考：构件块、表和过程》。

OPERATOR 权限概述

OPERATOR 权限有权对数据库执行检查点操作、删除连接（包括具有 DBA 权限的用户连接）、备份数据库以及监控系统。

OPERATOR 权限只能由具有 DBA 或 PERMS ADMIN 权限的用户授予其他用户。不具有 DBA 权限的用户必须具有 OPERATOR 权限，才能执行 **BACKUP**、**CHECKPOINT** 或 **DROP CONNECTION** 命令。

PERMS ADMIN 权限概述

PERMS ADMIN 权限是用以管理数据权限、组和口令的权限。

需要有 PERMS ADMIN 或 DBA 权限才能：

- 针对任何适用的用户对象（如表、视图或列）授予或撤消 **SELECT**、**INSERT**、**UPDATE**、**DELETE**、**ALTER** 或 **REFERENCES** 权限。
- 针对任何适用的用户对象（如过程或函数）授予或撤消 **EXECUTE** 权限。
- 创建和删除组，以及授予或撤消组成员资格。
- 授予或撤消所有现有的权限和新的权限（**DBA** 和 **REMOTE DBA** 除外）。更改任何帐户的口令。

只有具有 **DBA** 或 **PERMS ADMIN** 权限的用户才可以向其他用户授予 **PERMS ADMIN** 权限。

RESOURCE 权限概述

RESOURCE 权限是创建数据库对象（例如表、视图和存储过程）的权限。只有具有 **DBA** 或 **PERMS ADMIN** 权限的用户才可以向其他用户授予 **RESOURCE** 权限。

SPACE ADMIN 权限概述

SPACE ADMIN 权限是用以管理数据库空间的权限。

需要有 **SPACE ADMIN** 权限才能：

- 对数据库空间进行创建、更改、删除或注释。
- 授予或撤消对数据库空间的 **CREATE** 权限。
- 将数据和元数据移至数据库空间。
- 执行只读选择性恢复操作。

注意： **SPACE ADMIN** 权限可以创建数据库空间；但要在数据库空间中创建对象，用户必须被授予对该数据库空间的 **CREATE** 权限。既具有 **SPACE ADMIN** 权限又具有 **RESOURCE** 权限的用户仍须被授予对该数据库空间的 **CREATE** 特权，才能在该数据库空间中创建对象。如果没有 **CREATE** 特权，则尝试创建对象会返回 **Permission Denied** 错误。要添加 **CREATE** 特权，请使用 **GRANT CREATE ON**。

只有具有 **DBA** 或 **PERMS ADMIN** 权限的用户才可以向其他用户授予 **SPACE ADMIN** 权限。

另请参见

- 进行恢复之前（第 395 页）

USER ADMIN 权限概述

USER ADMIN 权限是用以管理用户、外部登录名和登录策略的权限。

需要有 **USER ADMIN** 权限才能：

- 创建和删除数据库用户，以及分配初始口令。
- 创建、更改或删除登录策略，以及向用户分配登录策略。
- 定义用户鉴定机制，如 **Kerberos** 或集成登录。
- 创建或删除外部登录。

- 在用户下次登录时强制更改口令。
- 重置用户登录策略。

只有具有 DBA 或 PERMS ADMIN 权限的用户才可以向其他用户授予 USER ADMIN 权限。

所有权权限概述

数据库对象的所有权包含它对该对象执行操作的权限。数据库对象的创建者会成为该对象的所有者。

所有权权限分配给用户的方式与本章中其它权限的分配方式不同。

所有者

在数据库中创建新对象的用户是该对象的*所有者*，并自动拥有对该对象执行任何操作的权限。例如，表的所有者可以修改表的结构，或者，可以向其他数据库用户授予更新表中信息的权限。

注意： 如果表的所有者是 DBA 或服务器在命令行上或配置文件中以 **-gl all** 开关开头，则只有该所有者才能装载该表。所有权和资源权限不够，不能使用 **LOAD TABLE**。为了使用 **LOAD TABLE** 语句，您还需要对表的 **INSERT** 权限。

例如，DBA 拥有修改数据库中任何组件的权限，因此可以删除其他用户所创建的表。DBA 拥有各个数据库对象所有者拥有的有关该对象的全部权限。

DBA 也能为其他用户创建数据库对象，在这种情况下，对象的所有者不是执行 CREATE 语句的用户 ID。尽管存在上述可能性，但本章提及数据库对象的所有者和创建者时，二者可彼此互换。

另请参见

- 没有口令的组（第 305 页）

数据库空间管理权限

要在数据库空间中创建表对象或连接索引，或者要将表对象或连接索引移至新数据库空间，则需要具有数据库空间的 **CREATE** 权限。

可以授予/撤消用户或组的数据库空间的 **CREATE** 权限。组的所有成员都将从组继承 **CREATE** 权限。缺省情况下，会将 **IQ_SYSTEM_MAIN**、**IQ_SYSTEM_TEMP** 和 **SYSTEM** 的 **CREATE** 权限授予 **PUBLIC**。对于其它 IQ 主数据库空间，系统管理员必须先向组/用户显式授予对数据库空间的 **CREATE** 权限，然后这些用户/组才能创建对象或将对象移动到该数据库空间中。例如，如果要将表置于新的 IQ 主数据库空间中，则用户必须具有对该数据库空间的 **CREATE** 权限。用户还必须具有 **RESOURCE** 权限才能创建对象。

另请参见

- 在嵌入式 SQL 中与数据库断开连接（第 148 页）

表和视图权限概述

您可以授予用户 ID 几种有关表和视图的独特权限：

权限	描述
ALTER	更改表结构的权限
DELETE	从表或视图中删除行的权限
INSERT	在表或视图中插入行的权限
REFERENCES	对表创建索引以及创建引用表的外键的权限
SELECT	在表或视图中查看信息的权限
UPDATE	更新表或视图中行的权限。只能针对表中的一组列授予此权限
ALL	所有上述权限

在 Multiplex 中，只有写入服务器可以修改针对写入服务器拥有的表的表权限。

组权限概述

为数据库的各个用户分别设置权限可能非常耗时而且容易出错。对于多数数据库，以组为单位（而不是针对单个用户 ID）对权限进行管理会更加有效。

给组指派权限的方法与给单个用户指派权限的方法完全相同。然后，您可以给数据库的各个新用户指派相应组的成员资格，这些用户就会通过其组成员资格获得一组权限。

示例

例如，您可以在公司数据库中为不同的部门（sales、marketing 等）分别创建不同的组并为这些组分配权限。每一名销售人员都会成为 sales 组的成员，并自动获得访问数据库相应区域的权限。

任何用户 ID 都可以是多个组的成员，而且它们继承各个组的所有权限。

服务器命令行权限选项

数据库服务器启动命令 `start iq` 拥有一些选项，可以设置某些数据库和服务器函数的权限级别。

表 41. 影响权限的启动选项

选项	描述	允许值	缺省值
<code>-gd level</code>	设置启动数据库所需的权限	DBA、ALL、NONE	DBA
<code>-gk level</code>	设置停止服务器所需的权限	DBA、ALL、NONE	DBA

选项	描述	允许值	缺省值
-gl level	设置装载数据所需的权限	DBA、ALL、NONE	ALL 适用于使用 start iq 启动的服务器；DBA 适用于其它服务器
-gu level	设置执行实用程序命令所需的权限，例如， CREATE DATABASE 和 DROP DATABASE	DBA、ALL、NONE、UTILITY_DB	ALL

有关这些选项、权限级别值和缺省值的详细信息，请参见《系统管理指南第一卷》>“控制权限的 Sybase IQ 启动命令行选项”以及《实用程序指南》>“start iq 数据库服务器启动实用程序”。

登录管理

Sybase IQ 定义了 在数据库对象中建立用户数据库连接时需要遵守的规则，这些规则称为 **登录策略**。

登录策略是数据库中的一个命名对象，其中包含一组选项。每个登录策略与称为登录策略选项的一组选项相关联。

您必须具有 **DBA** 特权或 **USER ADMIN** 权限，才能创建新登录策略或为用户分配现有登录策略。有关用于管理策略的 **SQL** 命令语法，请参见《参考：语句和选项》中的 **ALTER LOGIN POLICY** 语句、**ALTER USER** 语句、**CREATE LOGIN POLICY** 语句和 **DROP LOGIN POLICY** 语句。

每个新数据库创建时都具有称为“**root 策略**”的缺省登录策略。您可以修改 **root** 登录策略的选项值，但不能删除策略。如果在创建用户帐户时没有指定登录策略，则该用户会成为 **root** 登录策略的一部分。创建登录策略时未显式设置的任何选项都将从 **root** 策略继承其值。

请参见“SQL Anywhere Server - 数据库管理”>“配置数据库”>“管理用户 ID 和权限”。

注意： 此参考指向 **SQL Anywhere** 文档。

您可以在任意 **Multiplex** 服务器上执行登录管理命令，这些命令将自动传播到 **Multiplex** 中的所有服务器。为实现最佳性能，**Sybase** 建议您在协调器上执行这些命令或任何 **DDL**。请参见《使用 Sybase IQ Multiplex》。

有关使用 **Sybase Control Center** 管理登录策略的说明，请参见 **SCC** 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 **Sybase IQ** 的 **Sybase Control Center** 联机帮助。

从 12.7 迁移数据库会删除 现有的登录管理设置。要在迁移后重新创建登录管理设置，请使用下一节中的 **SQL** 语法。

用户帐户和连接管理

Sybase IQ 登录管理功能帮助您管理用户以及与数据库的连接。

具有 DBA 或 USER ADMIN 权限的用户可以通过以下方式来添加或删除用户和控制连接：

- 限制单个用户的活动登录的数目。
为此，请将用户分配到一个登录策略，在其中指定 **max_connections** 登录策略选项。
- 锁定用户。
为此，请将用户分配到一个登录策略，并将 **locked** 选项设置为 ON。
- 设置用户口令有效期。
为此，请为登录策略指定 **max_failed_login_attempts** 和 **max_days_since_login**，并将用户分配到此策略。还可以使用 SQL 语句 **CREATE/ALTER USER** 中的 **FORCE PASSWORD CHANGE** 子句显式使用用户口令过期。

有关用于管理策略的 SQL 命令语法，请参见《参考：语句和选项》中的 **CREATE LOGIN POLICY**、**ALTER LOGIN POLICY** 和 **DROP LOGIN POLICY**。

每个新数据库均包含一个名为 **root** 的登录策略。您可以修改 **root** 登录策略的选项值，但不能删除该策略。如果在创建用户帐户时没有指定其登录策略，则该用户会成为 **root** 登录策略的一部分。

登录管理的存储过程

下表列出了执行各个 Sybase IQ 登录管理函数所需调用的过程。

表 42. 登录管理的存储过程

存储过程	用途	需要的权限
sa_get_user_status	检索所有现有用户的当前状态	DBA 或 USER ADMIN
sp_expireallpasswords	导致所有用户口令立即过期	DBA 或 USER ADMIN
sp_iqaddlogin	添加用户，定义其口令，指定登录策略和下次登录时口令到期	DBA
sp_iqcopyloginpolicy	通过复制现有登录策略创建新的登录策略	DBA
sp_iqdroplogin	删除指定的用户	DBA
sp_iqmodify	为给定用户分配登录策略	DBA

存储过程	用途	需要的权限
sp_iqmodifyadmin	将命名登录策略的某个选项设置为特定值	DBA
sp_iqpassword	更改用户口令	DBA（用以更改其他用户的口令）。所有用户都可以运行 sp_iqpassword 来更改他们自己的口令。

另请参见

- 管理单个用户 ID 和权限（第 294 页）
- 针对数据库连接的限制（第 313 页）

在失败的登录尝试后阻止连接

以下示例显示您如何能在五次失败的登录尝试之后阻止用户连接。

1. 创建登录策略 **lp**，并将其登录策略选项 **max_failed_login_attempts** 设置为值 5。

```
CREATE LOGIN POLICY lp max_failed_login_attempts=5;
```

2. 创建属于登录策略 **lp** 的用户 **John**。

```
CREATE USER john IDENTIFIED BY j345 LOGIN POLICY lp;
```

由于 **John** 属于 **lp** 登录策略，其中 **max_failed_login_attempts=5**，因此只要此用户超过 **max_failed_login_attempts** 的限制 5，就无法登录到 IQ 服务器。

创建 DBA 恢复帐户

Sybase 建议您为生产系统创建一个 DBA 恢复帐户。DBA 恢复帐户是备份帐户，以防您丢失原始 DBA 帐户口令。

1. 使用随机生成的用户名和口令创建一个或多个额外的 DBA 帐户。
2. 将凭据锁藏在安全的位置。

另请参见

- 使用 DBA 恢复帐户登录（第 290 页）

使用 DBA 恢复帐户登录

如果原始 DBA 帐户口令丢失，请使用 DBA 恢复帐户登录，然后重置原始 DBA 帐户口令。

1. 从安全位置检索 DBA 恢复帐户的用户名和口令。
2. 使用恢复帐户登录。

3. 重置原始 DBA 帐户口令。

另请参见

- 创建 DBA 恢复帐户（第 290 页）

锁定用户

您可以专门为锁定用户创建一个登录策略来强制锁定用户。

1. 创建具有登录策略选项 **locked=ON** 的登录策略，如下所示：

```
CREATE LOGIN POLICY locked_users locked=ON
```

2. 将您要锁定的用户分配到 **locked_users** 策略，例如：

```
ALTER USER john LOGIN POLICY locked_users
```

解锁用户

解锁用户的方法有若干种。

1. 假定用户被锁定是因为他或她属于 **locked=ON** 的登录策略。要解锁这样的用户，请将他或她分配到选项 **locked=OFF** 的登录策略。
2. 如果用户超过了 **max_failed_login_attempts** 或 **max_days_since_login**，还可以锁定他/她。要解锁这样的用户，请使用类似下面的语句：

```
ALTER USER john RESET LOGIN POLICY
```

上面的语句在语义上等同于 `ALTER USER userid LOGIN POLICY current-policy-for-user`。

实用程序数据库服务器安全性

Sybase IQ 包括一个幻像数据库（称为 *实用程序数据库*），它没有物理表示。

此数据库没有数据库文件，此数据库也不含任何数据。实用程序数据库可以在任何 Sybase IQ 服务器上运行。在 Sybase Central 中，实用程序数据库的服务器称为实例服务器。

实用程序数据库允许小范围的专用函数。提供该数据库的目的在于让您无需先连接到物理数据库，即可执行数据库文件操作语句，例如 **CREATE DATABASE** 和 **DROP DATABASE**。

您也可以从实用程序数据库检索数据库和连接属性。这些属性在连接到实用程序数据库时应用于您所创建的数据库。

您的配置任务之一是设置实用程序数据库及其服务器的安全性。实用程序数据库服务器安全性有两个方面：

- 谁能够连接到实用程序数据库？
- 谁能够执行文件管理语句？

另请参见

- 进行恢复之前 (第 395 页)
- 控制权限的命令行选项 (第 27 页)
- 移动数据库文件 (第 399 页)
- 实用程序数据库 (第 13 页)
- 检验数据库备份 (第 406 页)
- **CREATE DATABASE** 语句缺省值 (第 140 页)
- 使用 **SQL** 创建数据库 (第 137 页)
- **Multiplex** 功能 (第 5 页)
- 相对路径名 (第 141 页)

启动实用程序数据库

在启动实用程序数据库时，无法指定数据库文件，因为没有数据库文件与该数据库相关联。

连接时，指定 `utility_db` 为数据库名称。

例如：

```
dbisqlc -c  
"uid=dba;pwd=sql;eng=myserver;dbn=utility_db"
```

注意： 当您连接到实用程序数据库创建拥有 Windows 原始分区的 IQ 数据库时，请注意 IQ PATH 中存在的语法差异。例如，要在设备 I: 上为实用程序数据库指定 Windows 原始分区，则可以使用规范 “\\I:”，而在其它 IQ 数据库上，则必须使用双倍的斜杠字符，因此，同一设备应指定为 “\\\\I:”。反斜杠字符在 IQ 数据库中被视为转义字符，但在实用程序数据库中被视为常规字符。

定义实用程序数据库口令

为了使用实用程序数据库，您必须指定用户 ID DBA。

1. 使用文本编辑器打开文件 `util_db.ini`，它存储在服务器的可执行文件目录中。

由于该目录位于服务器上，因此您可以控制对该文件的访问，从而控制谁能够访问该口令。

2. 编辑以下行，用所需口令替换单词 “password”：

```
[UTILITY_DB]  
PWD=password
```

`utility_db` 安全级别的使用依赖托管数据库服务器的计算机的物理安全，这是由于 `util_db.ini` 文件可通过使用文本编辑器方便地读取。

执行文件管理语句的权限

为了提供更高的数据库安全性，有一个独立的安全级别控制数据库的创建和删除操作。**-gu** 数据库服务器命令行选项控制谁可以执行文件管理语句。

对于使用文件管理语句有四个权限级别。这些级别是：**all**、**none**、**DBA** 和 **utility_db**。**utility_db** 级别只允许能够连接到实用程序数据库的用户使用文件管理语句。

表 43. 文件管理的权限

-gu 开关值	作用	适用于
all	任何用户都能够执行文件管理语句	包括实用程序数据库在内的任何数据库
无	任何用户都不能执行文件管理语句	包括实用程序数据库在内的任何数据库
DBA	只有具备 DBA 权限的用户才能够执行文件管理语句	包括实用程序数据库在内的任何数据库
utility_db	只有可以连接到实用程序数据库的用户才能执行文件管理语句	仅限实用程序数据库

示例

在 Sun、HP、Linux 和 Windows 平台上，要仅允许知道实用程序数据库口令的用户连接到实用程序数据库以及创建或删除数据库，请使用以下命令从命令行启动服务器：

```
start_iq -n testsrv -gu utility_db
```

在 AIX 上，要仅允许知道实用程序数据库口令的用户连接到实用程序数据库以及创建或删除数据库，请使用以下命令从命令行启动服务器：

```
start_iq -n testsrv -gu utility_db -iqmt 256
```

假定实用程序数据库口令已在安装过程中设置为 **IQ&Mine49**，则下面的命令会将 **Interactive SQL** 实用程序作为客户端应用程序启动、连接到名为 **testsrv** 的服务器、装载实用程序数据库并连接到该用户：

```
dbisql -c
"uid=DBA;pwd=IQ&Mine49;dbn=utility_db;eng=testsrv"
```

成功执行此语句将连接到实用程序数据库。您现在便能创建和删除数据库了。

注意：数据库名称、用户 ID 和口令均区分大小写。确保您在 **dbisql** 命令和 **util_db.ini** 文件中指定了相同的大小写。

管理单个用户 ID 和权限

在管理组之前，应了解用户权限。您可以使用 **Interactive SQL** 和 **Sybase Central** 向用户授予权限以及创建新用户。

对于多数数据库，应使用组来执行批量权限管理，而不是一次一个地给单个用户指派权限。但是，由于组只是附带有特殊属性的用户 ID，因此在继续阅读有关管理组的论述之前，您应该先阅读并理解本节的内容。

使用 IQ 存储过程管理用户

您也可以使用 **IQ** 系统过程创建新用户。您必须使用那些过程来添加用户以及修改他们的口令及其登录功能，以便管理具有 **Sybase IQ** 登录管理功能的那些用户。

要添加和修改具有 **Sybase IQ** 登录管理功能的用户，请使用针对用户帐户和连接管理的系统过程。

要授予用户对于数据库对象的权限，您仍须使用本节余下部分介绍的命令和过程。

使用 ASE 存储过程管理用户

本章介绍如何使用 **Interactive SQL** 和 **Sybase Central** 管理用户和组。您可以使用与 **Adaptive Server Enterprise** 兼容的存储过程执行很多相同的任务。如果您之前曾使用过 **Adaptive Server Enterprise** 或 12.0 版前的 **Sybase IQ**，您可能倾向于使用这些存储过程。请参见《参考：构件块、表和过程》。ASE 存储过程不允许您使用 **Sybase IQ** 登录管理功能限制连接。

另请参见

- 用户帐户和连接管理（第 289 页）
- 组管理（第 302 页）

使用 **Interactive SQL** 创建用户

DBA 或具有 **USER ADMIN** 权限的用户可以使用 **CREATE USER** 语句创建新用户。

1. 从 **Interactive SQL** 以具有 **DBA** 或 **USER ADMIN** 权限的用户身份连接到数据库。
2. 发出 **CREATE USER** 语句。

下面的示例将口令为 *welcome* 的用户 ID *M_Haneef* 添加到数据库中。

```
CREATE USER M_Haneef IDENTIFIED BY welcome
```

在 **Sybase Central** 中创建用户

请按照下列步骤创建用户。

1. 连接到数据库。

2. 右键单击“用户和组”文件夹。
3. 选择“新建” > “用户”。创建用户向导将引导您完成整个过程。

有关详细信息，请参见 IQ 插件的 Sybase Central 联机帮助。

在 Sybase Control Center 中创建用户

有关在 Sybase Control Center 中创建 Sybase IQ 用户的说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

SCC 提供了用于管理 Sybase IQ 和 SCC 本身的用户和组的工具。请务必查找 Sybase IQ 用户和组的说明。

更改口令

如果您具有 DBA 或 PERMS ADMIN 权限，可以更改任何现有用户的口令。使用以下命令更改口令：

```
ALTER USER userid IDENTIFIED BY password
```

如果您在本来要添加新用户时无意中输入了现有用户的用户 ID，则实际上是在更改现有用户的口令。您不会收到警告，因为这被视为正常操作。此行为与 Sybase IQ 12 版以前的版本有所不同。

为避免出现这种情况，请使用 `sp_iqaddlogin` 系统过程或 `CREATE USER` 添加用户。还可以使用 `sp_addlogin` 和 `sp_adduser` 系统过程。这些过程会在您尝试添加现有用户 ID 时提供错误消息，就像 Adaptive Server Enterprise 和 12 版前的 Sybase IQ 一样。

口令规则

您可以设置口令规则并检验分配的新口令是否符合规则。例如，您可能要求口令必须包含一个数字或不能是用户 ID。有关详细信息，请参见《参考：语句和选项》中的 `VERIFY_PASSWORD_FUNCTION` 选项。

要设置最小口令长度，请参见《参考：语句和选项》中的 `MIN_PASSWORD_LENGTH` 选项。

更改 DBA 口令

所有数据库的 DBA 用户 ID 的缺省口令都是 `sql`。您应更改此口令以防止对数据库的非授权访问。

前提条件

要更改 DBA 口令，您必须具有 DBA 或 PERMS ADMIN 权限。

如果您使用的是 `dbisql`，最好将您的权限授予放入命令文件作为参考，以便在需要重新创建权限时修改并重新运行它。

过程

用户 ID DBA 标识具有完全管理权限和资源创建权限的用户。下面的命令将用户 ID 为 DBA 的口令更改为 new_password:

```
ALTER USER DBA IDENTIFIED BY new_password
```

向用户授予 DBA 权限

DBA 权限可以授予任何权限。只有 DBA 可以向数据库用户授予 DBA 权限。

DBA 权限非常强大，赋予了对数据库执行任何操作以及访问数据库中所有信息的能力。一般来说，将 DBA 权限授予多个人是不明智的，应该授予极少数人。

1. 以具有 DBA 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句:

```
GRANT DBA TO userid
```

您应该给具有 DBA 权限的用户两个用户 ID: 一个有 DBA 权限，一个没有。这样，这些用户就可以只在需要时才以 DBA 身份进行连接。

向用户授予 OPERATOR 权限

大多数权限都是以同样的方式授予的。DBA 权限可以授予任何权限。PERMS ADMIN 权限可以授予除 DBA 和 REMOTE DBA 外的任何权限。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句:

```
GRANT OPERATOR TO userid
```

向用户授予 PERMS ADMIN 权限

PERMS ADMIN 权限可以授予除 DBA 和 REMOTE DBA 外的任何权限。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句:

```
GRANT PERMS ADMIN TO userid
```

向用户授予 RESOURCE 权限

具有 RESOURCE 权限的用户可以创建新的数据库对象，例如表、视图、索引或过程。

大多数权限都是以同样的方式授予的。DBA 权限可以授予任何权限。PERMS ADMIN 权限可以授予除 DBA 和 REMOTE DBA 外的任何权限。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句：

```
GRANT RESOURCE TO userid
```

授予 SPACE ADMIN 权限

大多数权限都是以同样的方式授予的。DBA 权限可以授予任何权限。PERMS ADMIN 权限可以授予除 DBA 和 REMOTE DBA 外的任何权限。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句：

```
GRANT SPACE ADMIN TO userid
```

向用户授予 USER ADMIN 权限

使用 USER ADMIN 权限，您可以管理外部登录名、登录策略和其他用户。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 键入并执行下面的 SQL 语句：

```
GRANT USER ADMIN TO userid
```

针对表和视图的权限

您可以对个别表和视图指派一组权限。用户可被授予这些权限组合来定义他们对表或视图的访问权限。

权限组合

- ALTER（更改表结构的权限）和 REFERENCES（创建索引和创建外键的权限）权限可授予修改数据库 schema 的权限，因此不会分配给大多数用户。这些权限不适用于视图。
- DELETE、INSERT 和 UPDATE 权限可授予修改表或视图中的数据的数据的权限。其中 UPDATE 权限可以被限制到表或视图中的一组列。
- SELECT 权限可授予在表或视图中查看数据的权限，但不授予更改数据的权限。
- ALL 权限可授予所有上述权限。

在 Interactive SQL 中授予针对表和视图的 Delete 权限

所有表和视图权限的授予方式都非常相似。

使用 GRANT 语句授予权限。例如，向 M_Haneef 授予从名为 sample_table 的表中删除行的权限：

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份或 sample_table 的所有者身份连接到数据库。
- 2.

键入并执行下面的 SQL 语句：

```
GRANT DELETE ON sample_table TO M_Haneef
```

另请参见

- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）
- 在 Sybase Central 中授予针对表的用户权限（第 299 页）
- 组的权限（第 304 页）

在 Interactive SQL 中授予针对表和视图的 Update 权限

表和视图权限只应用于表或视图中的所有数据（UPDATE 权限除外，它可能已被限制）。

通过创建对表执行操作的过程，然后授予用户执行该过程的权限，可以完成对用户权限的精细调优。

向 M_Haneef 授予仅在名为 sample_table 的表中更新 column_1 和 column_2 列的权限：

以具有 DBA 或 PERMS ADMIN 权限的用户身份或 sample_table 的所有者身份连接到数据库。

键入并执行下面的 SQL 语句：

```
GRANT UPDATE (column_1, column_2)  
ON sample_table  
TO M_Haneef
```

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）
- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Sybase Central 中授予针对表的用户权限（第 299 页）
- 组的权限（第 304 页）

在 Sybase Central 中授予针对表的用户权限

以下是一种在 Sybase Central 中授予对某个表的用户权限的方式：

1. 连接到数据库。
2. 双击该数据库的“表”文件夹，在左面板中显示表。
3. 右键单击一个表，然后从弹出菜单中选择“属性”。
4. 在“属性”对话框的“权限”选项卡上配置表的权限：
 - 单击“授权”，选择要为其授予全部权限的用户或组。
 - 单击该用户或组旁边的字段以设置特定权限。权限由选中标记表明，而权限选项由带有两个“+”号的选中标记表明。
 - 选择用户并单击“引用”、“选择”或“更新”旁边的按钮，以设置单个列的权限类型。
 - 在列表中选择用户或组，并单击“撤消”以撤消所有权限。

您还可以从“用户和组”属性表分配权限。要同时给多个用户和多个组指派权限，请使用表的属性表。要同时给多个表分配权限，请使用用户的属性表。

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）
- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）
- 组的权限（第 304 页）

授予“授予权限”的权限

要分配表权限和视图权限，请使用 **GRANT** 语句子句 **WITH GRANT OPTION**。

该选项授予将权限传递给其他用户的权力。

例如，向 **M_Haneef** 授予从名为 **sample_table** 的表中删除行的权限，以及将此权限传递给其他用户的权限：

1. 以具有 **DBA** 或 **PERMS ADMIN** 权限的用户身份或 **sample_table** 的所有者身份连接到数据库。
2. 键入并执行下面的 SQL 语句：

```
GRANT DELETE ON sample_table
TO M_Haneef
WITH GRANT OPTION
```

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）

- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）
- 在 Sybase Central 中授予针对表的用户权限（第 299 页）
- 组的权限（第 304 页）

在 Interactive SQL 中授予针对过程的权限

仅有一种权限可以授予一个过程，即执行（或 CALL）该过程的 EXECUTE 权限。

执行存储过程的权限可由 DBA 或 PERMS ADMIN 授予，也可由过程的所有者（创建过程的用户 ID）授予。

使用 GRANT EXECUTE 语句授予针对过程的权限。例如，向 M_Haneef 授予执行名为 my_procedure 的过程的权限：

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份或 my_procedure 过程的所有者身份连接到数据库。
2. 执行下面的 SQL 语句：

```
GRANT EXECUTE
ON my_procedure
TO M_Haneef
```

过程的执行权限

使用过程所有者的权限执行过程。更新表中信息的任何过程都仅在过程所有者拥有表的 UPDATE 权限时才能成功执行。

只要过程所有者确实拥有相应权限，当拥有执行该过程的权限的用户调用该过程时，无论该用户是否有对基础表的权限，该过程均将成功执行。您可以使用过程来允许用户对表执行明确定义的活动，而不必拥有对表的任何常规权限。

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）
- 在 Sybase Central 中授予针对表的用户权限（第 299 页）
- 组的权限（第 304 页）
- 过程提供了定制的安全性（第 310 页）

在 Sybase Central 中授予针对过程的权限

以下是一种在 Sybase Central 中授予针对某个表的用户权限的方式。

1. 连接到数据库。

2. 单击“用户和组”文件夹并定位要授予其权限的用户。
3. 右键单击用户，然后从弹出菜单中选择“复制”。
4. 在“存储过程”文件夹中定位要允许用户执行的过程。
5. 单击该过程，然后从主菜单中选择“编辑” > “粘贴”授予权限。

有关详细信息，请参见 Sybase Central 联机帮助。

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）
- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）
- 在 Sybase Central 中授予针对表的用户权限（第 299 页）
- 组的权限（第 304 页）
- 过程提供了定制的安全性（第 310 页）

在 Interactive SQL 中撤消用户权限

任何用户的权限都是已授予的和已撤消的权限的组合。通过撤消和授予权限，您可以管理用户对于数据库的权限的模式。

REVOKE 语句与 GRANT 语句完全相反。

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份或该过程的所有者身份连接。
2. 执行 REVOKE 语句。

例如，禁止 M_Haneef 执行 my_procedure:

```
REVOKE EXECUTE ON my_procedure FROM M_Haneef
```

禁止 M_Haneef 删除 sample_table 中的行:

```
REVOKE DELETE ON sample_table FROM M_Haneef
```

警告! 撤消特权或删除用户之前，必须要清楚以下限制:

- 在发出 **REVOKE CONNECT** 或 **sp_dropuser** 之前，必须删除该用户拥有的任何对象，例如表。如果尝试在用户拥有任何数据库对象的同时，撤消该用户的连接权限或使用存储过程 **sp_dropuser**，您将收到错误消息。
- 像 **sp_dropuser** 这样的过程提供与 Adaptive Server Enterprise 存储过程的最小兼容性。如果习惯于 Adaptive Server Enterprise（或 Sybase IQ 11.x）存储过程，请在 **dbisql** 中使用当前的 Sybase IQ 过程之前，比较两者的文本。要进行比较，请使用命令

```
sp_helptext 'owner.procedure_name'
```

对于所有由 Sybase 提供的系统存储过程来说，所有者都是 **dbo**。要查看由不同用户拥有的同名存储过程的文本，您必须指定该用户，例如：

```
sp_helptext 'myname.myprocedure'
```

组管理

了解如何管理单个用户的权限后（如上节所述），使用组就变得非常简单了。

DBA、RESOURCE 和 GROUP 权限

当授予或撤消组对表、视图和过程的权限时，组的所有成员都将继承这些更改。

DBA、RESOURCE 和 GROUP 权限不会被继承：这些权限必须要逐个地授予需要它们的每个用户 ID。

组只是一个拥有特殊权限的用户 ID。授予组的权限和撤消组的权限的方式与任何其他用户完全相同。

一个组也可以是另一个组的成员。您可以构建若干个组的层次结构，每个组都继承其父组的权限。

一个用户 ID 也可被授予多个组的成员资格，因此，用户与组的关系是多对多。

另请参见

- 管理单个用户 ID 和权限（第 294 页）

在 Interactive SQL 中创建组

由用户 ID 标识的组就像一个用户一样，但此用户被授予了拥有 *成员* 的权限。

1. 以具有 DBA 权限的用户身份或者具有 USER ADMIN 和 PERMS ADMIN 权限的用户身份连接到数据库。USER ADMIN 对 CREATE USER 来说是必需的，PERMS ADMIN 对 GRANT GROUP 来说是必需的。

2.

使用以下 SQL 语句像创建任何其它用户 ID 一样创建组的用户 ID：

```
CREATE USER personnel IDENTIFIED BY group_password
```

3.

使用以下 SQL 语句，授予 personnel 用户 ID 拥有成员的权限：

```
GRANT GROUP TO personnel
```

授予该用户 ID 能够拥有成员的 GROUP 权限不会被组的成员继承。如果不这样的话，每个用户 ID 都会由于特殊的 PUBLIC 组中的成员资格，而自动成为组。

前一示例使用用户 ID 和口令创建组。您还可以创建不带口令的组，以阻止任何人使用组用户 ID 登录。

在 Sybase Central 中创建组

可以使用 Sybase Central 创建组。

1. 连接到数据库。
2. 右键单击“用户和组”文件夹。
3. 选择“新建” > “组”。一个向导即会引导您完成整个过程。

有关详细信息，请参见 Sybase Central 联机帮助。

在 Sybase Control Center 中创建组

有关在 Sybase Control Center 中创建 Sybase IQ 组的说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

SCC 提供了用于管理 Sybase IQ 和 SCC 本身的用户和组的工具。请务必查找 Sybase IQ 用户和组的说明。

在 Interactive SQL 中添加组成员

GRANT 语句使用户成为组的成员。

前提条件

组的成员资格可以由 DBA 或 PERMS ADMIN 权限授予。

过程

1. 以具有 DBA 或 PERMS ADMIN 权限的用户身份连接到数据库。
2. 发出相应的 GRANT 语句。例如，向用户 M_Haneef 授予 personnel 组的成员资格：

```
GRANT MEMBERSHIP
IN GROUP personnel
TO M_Haneef
```

当用户获得某个组的成员资格时，他们将继承与该组关联的表、视图和过程的所有权限。如果不希望某个特定用户访问特定的表、视图或过程，则不要使用该用户成为对该对象拥有权限的组的成员。

不能针对组中的特定用户撤销权限。

在 Sybase Central 中添加组成员

当用户获得某个组的成员资格时，他们将继承与该组关联的表、视图和过程的所有权限。

如果不希望某个特定用户访问特定的表、视图或过程，则不要使该用户成为对该对象拥有权限的组的成员。

1. 连接到数据库。
2. 双击该数据库的“用户和组”文件夹，打开该文件夹。组显示在左面板中，用户和组显示在右面板中。
3. 在右面板中，选择您要添加到一个组的用户，然后将他们拖到该组上。

有关详细信息，请参见 Sybase Central 联机帮助。

在 Sybase Control Center 中添加组成员

有关在 Sybase Control Center 中向 Sybase IQ 组添加成员的说明，请参见 SCC 中或 <http://sybooks.sybase.com/nav/summary.do?prod=10680> 上的 Sybase IQ 的 Sybase Control Center 联机帮助。

SCC 提供了用于管理 Sybase IQ 和 SCC 本身的用户和组的工具。请务必查找 Sybase IQ 用户和组的说明。

组的权限

权限可以以和任何其它用户 ID 完全一样的方式授予给组。

针对表、视图和过程的权限是由组的成员（包括其它组及其成员）继承的。数据库管理员需要注意：组权限是比较复杂的。

DBA、RESOURCE 和 GROUP 权限不能被组的成员继承。即使 `personnel` 用户 ID 被授予 RESOURCE 权限，`personnel` 的成员也没有 RESOURCE 权限。

数据库对象的所有权与单个用户 ID 相关联，不由组成员继承。如果用户 ID `personnel` 创建了一个表，则 `personnel` 用户 ID 就是该表的所有者，它有权对该表进行任何更改，而且，有权将有关该表的权限授予其他用户。作为 `personnel` 的成员的其它用户 ID 不是此表的所有者，也就没有这些权限。不过，如果 SELECT 权限由 DBA 或 `personnel` 用户 ID 自己显式授予了 `personnel` 用户 ID，则所有组成员都拥有对该表的 `select` 访问权限。换句话说，只有授予的权限会被继承。

另请参见

- 在 Interactive SQL 中授予针对表和视图的 Delete 权限（第 297 页）
- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）
- 授予“授予权限”的权限（第 299 页）
- 在 Interactive SQL 中授予针对表和视图的 Update 权限（第 298 页）

- 在 Sybase Central 中授予针对表的权限 (第 299 页)

组拥有的表

组可以用于查找数据库中的表和过程。

例如, 查询

```
SELECT * FROM SYSGROUPS
```

总会查找表 SYSGROUPS, 因为所有用户都属于 PUBLIC 组, 而 PUBLIC 又属于拥有 SYSGROUPS 表的 SYS 组。(SYSGROUPS 表包含了一个 *group_name*、*member_name* 对的列表, 这些对表示数据库中的组成员资格。)

如果表 Employees 属于 personnel 用户 ID 所有, 并且 M_Haneef 是 Personnel 组的一位成员, 则 M_Haneef 可以在 SQL 语句中将 Employees 表简称为 Employees。非 Personnel 组成员的用户需要使用限定名 Personnel.Employees。

拥有表的组

Sybase 建议您创建一个其唯一目的就是拥有表的组。

不为该组授予任何权限, 但让所有用户都成为该组的成员。这使得所有人在访问表时都无需使用限定名。然后, 您可以创建权限组, 并在确有必要时给用户授予这些权限组中的成员资格。

另请参见

- 对象名的所有者前缀 (第 306 页)

没有口令的组

以组的用户 ID 连接的用户会拥有某些权限。另外, 此类用户还拥有以该组用户 ID 的名义创建的数据库中任意表的所有权权限。

可以建立一个数据库, 以便所有处理组及其数据库对象的工作仅由 DBA 或具有 PERMS ADMIN 权限的指定用户完成, 而不允许其它用户 ID 更改组成员资格。

这可以通过创建组时, 禁止以该组的用户 ID 进行连接来实现。操作方法是: 键入 GRANT CONNECT 语句, 不带口令。例如:

```
GRANT CONNECT
TO personnel
```

创建用户 ID personnel。此用户 ID 可被授予组权限, 而其它用户 ID 可被授予该组的成员资格, 继承已授予 personnel 的任何权限, 但无人可以使用 personnel 用户 ID 连接到数据库, 因为它没有有效口令。

虽然没有用户可以使用用户 ID personnel 来连接数据库, 但是此用户 ID 仍然可以是数据库对象的所有者。CREATE TABLE 语句、CREATE PROCEDURE 语句和 CREATE VIEW 语句都允许将该对象的所有者指定为执行该语句的用户之外的用户。这项所有权的指派操作只能由 DBA 执行。

另请参见

- 所有权权限概述 (第 286 页)

自动创建的组

在创建数据库时，会自动创建两个组。即 **SYS** 和 **PUBLIC**。这两个组都没有口令，因此不可以 **SYS** 或 **PUBLIC** 组身份连接到数据库。这两个组在数据库中起着重要作用。

SYS 组

SYS 组是数据库的系统表和视图的所有者，它们包含了有关数据库结构的完整说明 (包括所有数据库对象和所有用户 ID)。

有关系统表和视图的说明以及访问这些表的说明，请参见《参考：构件块、表和过程》。

PUBLIC 组

在创建数据库时，会自动创建 **PUBLIC** 组，它拥有对数据库的 **CONNECT** 权限以及对系统表的 **SELECT** 权限。

PUBLIC 组是 **SYS** 组的成员，并拥有对某些系统表和视图的读取访问权限，因此数据库的任何用户都可以了解有关数据库 **schema** 的信息。如果您希望限制此访问权限，可将 **PUBLIC** 在 **SYS** 组中的成员资格撤消。

任何新用户 ID 都会自动成为 **PUBLIC** 组的成员，并继承由 **DBA** 专门授予该组的任何权限。如果需要，您还可以撤消用户在 **PUBLIC** 中的成员资格。

对象名的所有者前缀

每个数据库对象的名称都是一个标识符。在某些情况下，您需要使用所有者前缀来标识表、视图和过程。

《参考：构件块、表和过程》>“SQL 语言元素”>“标识符”中介绍了有效标识符的规则。

在本指南中的所有查询和示例 SQL 语句中，提及演示数据库中的数据库对象时，通常采用的是它们的简单名称。例如：

```
SELECT *  
FROM Employees
```

表、过程和视图都有所有者。演示数据库中表的所有者为用户 ID **DBA**。在某些情况下，必须使用所有者用户 ID 作为对象名的前缀，如下面语句中所示。

```
SELECT *  
FROM "DBA".Employees
```

Employees 表引用被称为限定的。(这种情况下，所有者名称被括在双引号内，因为 **DBA** 是一个 SQL 关键字。) 在其它情况下，给出对象名就足够了。

当引用数据库对象时，需要使用前缀，但以下情况除外：

- 您是数据库对象的所有者。
- 您是拥有数据库对象的组 ID 的成员。

示例

假设以下示例是一个公司的数据库。所有表均由用户 ID `company` 创建。此用户 ID 由数据库管理员使用，因此被给予 DBA 权限。

```
GRANT CONNECT TO company
IDENTIFIED BY secret;
GRANT DBA TO company;
```

数据库中的表由 `company` 用户 ID 创建。

```
CONNECT USER company IDENTIFIED BY secret;
CREATE TABLE company.Customers ( ... );
CREATE TABLE company.Products ( ... );
CREATE TABLE company.Orders ( ... );
CREATE TABLE company.Invoices ( ... );
CREATE TABLE company.Employees ( ... );
CREATE TABLE company.Salaries ( ... );
```

并非公司的每个人都有权访问所有信息。假设销售部门中的两个用户 ID `Joe` 和 `Sally` 应该有权访问 `Customers`、`Products` 和 `Orders` 表。要实现这一点，可创建 `Sales` 组。

```
GRANT CONNECT TO Sally IDENTIFIED BY xxxxxx;
GRANT CONNECT TO Joe IDENTIFIED BY xxxxxx;
GRANT CONNECT TO Sales IDENTIFIED BY xxxxxx;
GRANT GROUP TO Sales;
GRANT ALL ON Customers TO Sales;
GRANT ALL ON Orders TO Sales;
GRANT SELECT ON Products TO Sales;
GRANT MEMBERSHIP IN GROUP Sales TO Sally;
GRANT MEMBERSHIP IN GROUP Sales TO Joe;
```

现在，`Joe` 和 `Sally` 有权使用这些表，但他们仍须限定他们的表引用，因为表的所有者是 `company`，而 `Sally` 和 `Joe` 不是 `company` 组的成员：

```
SELECT *
FROM company.customers
```

要改变这种情况，请让 `Sales` 组成为 `company` 组的成员。

```
GRANT GROUP TO company;
GRANT MEMBERSHIP IN GROUP company TO Sales;
```

现在，`Sales` 组的成员 `Joe` 和 `Sally` 间接地成为了 `company` 组的成员，可以不用限定符而引用他们的表。现在可以使用以下命令：

```
SELECT *
FROM Customers
```

注释

由于 Joe 和 Sally 是 company 组的成员，因此他们没有任何额外的权限。company 组没有被明确授予任何表权限。（company 用户 ID 拥有查看诸如 Salaries 之类的表的隐式权限，因为该 ID 创建了这些表并具有 DBA 权限。）因此，Joe 和 Sally 在执行以下命令之一时仍会遇到错误：

```
SELECT *  
FROM Salaries;  
SELECT *  
FROM company.Salaries
```

在这两种情况下，Joe 和 Sally 都没有查看 Salaries 表的权限。

另请参见

- 组拥有的表（第 305 页）

视图和过程提供了更高的安全性

您可以使用视图和存储过程来对权限进行量身定制，以适应您组织的需要。

对于安全性要求很高的数据库，直接对表定义权限会受到限制。给用户授予的对表的任何权限都会应用于整个表。在许多情况下，需要更精确地授予用户的权限，而不仅仅是逐个表地授权。例如：

- 不应将员工表中所存储的个人信息或敏感信息的访问权授予那些需要访问该表其它部分的用户。
- 您可能想要授予销售代表更新包含其销售电话记录说明的表的权限，但此权限要仅限于其自己的电话。

本节描述了使用视图和过程对权限进行管理的一些方法。

另请参见

- 视图管理（第 179 页）

视图提供了定制的安全性

在仅适合授予用户拥有对部分表的访问权限时，视图对于安全非常有用。

可以用行或列来定义此部分。例如，您可能需要禁止一组用户查看 Employees 表的 Salary 列，或者可能希望限制用户仅查看他们创建的表的行。

示例 1

销售经理需要访问数据库中有关该部门员工的信息。但是，该经理没有理由访问有关其它部门员工的信息。

此示例描述了如何为销售经理创建用户 ID，如何创建可提供其所需信息的视图，以及如何给此销售经理用户 ID 授予适当的权限。

1. 使用具有 DBA 权限的用户 ID，通过 GRANT 语句创建新用户 ID。输入：

```
CONNECT "DBA"
IDENTIFIED by sql;
GRANT CONNECT
TO SalesManager
IDENTIFIED BY sales
```

您必须将 DBA 用引号引起来，因为它是一个 SQL 关键字，就像 **SELECT** 和 **FROM** 一样。

2. 定义一个只查看销售部门员工的视图：

```
CREATE VIEW emp_sales AS SELECT EmployeeID, GivenName, Surname
FROM "DBA".Employees WHERE DepartmentID = 200
```

将该表标识为 “DBA”.Employees，并将该表的所有者显式标识出来，以便 SalesManager 用户 ID 能够使用该视图。否则，当 SalesManager 使用该视图时，**SELECT** 语句会引用该用户 ID 不能识别的表。

3. 授予 SalesManager 查看该视图的权限：

```
GRANT SELECT ON emp_sales TO SalesManager
```

使用和授予对表的权限相同的命令可以授予对视图的权限。

示例 2

下一个示例创建一个允许 Sales Manager 查看销售订单摘要的视图。定义此视图需要有来自多个表中的信息：

1. 创建视图。

```
CREATE VIEW order_summary AS
SELECT OrderDate, Region, SalesRepresentative
FROM "GROUPO".SalesOrders
KEY JOIN "GROUPO".Customers
```

2. 授予 Sales Manager 查看该视图的权限。

```
GRANT SELECT
ON order_summary
TO SalesManager
```

3. 要检查该过程是否正确执行，请连接到 SalesManager 用户 ID，然后查看您创建的视图：

```
CONNECT SalesManager IDENTIFIED BY sales ;
SELECT * FROM "GROUPO".emp_sales ;
SELECT * FROM "GROUPO".order_summary ;
```

没有授予 Sales Manager 查看基础表的权限。以下命令将产生权限错误。

```
SELECT * FROM "DBA".Employees ;
SELECT * FROM "DBA".SalesOrders ;
```

前面的示例说明了如何使用视图来定制 **SELECT** 权限。可以使用相同的方法授予对视图的 **INSERT**、**DELETE** 和 **UPDATE** 权限。

有关允许对视图进行数据修改的信息，请参见第 209 页的“使用视图的准则”。

另请参见

- 使用视图的准则（第 180 页）

过程提供了定制的安全性

视图可以数据为基础限制访问权限，而过程则限制用户可以执行的操作。

用户可以对过程有 EXECUTE 权限，而对过程所针对的表没有任何权限。

缺省情况下，过程是使用过程所有者的权限执行的。对于更新表的过程，如果过程所有者具有对表的 UPDATE 权限，该用户就可以执行该过程。过程的所有者可以通过指定 SQL SECURITY INVOKER（调用 CREATE/ALTER PROCEDURE）语句，将过程限制为使用执行该过程的用户的权限来执行。

另请参见

- 在 Interactive SQL 中授予针对过程的权限（第 300 页）
- 在 Sybase Central 中授予针对过程的权限（第 300 页）

设置基于任务的安全性限制

为了获得严格的安全性，您可以禁止对基础表的所有访问，并授予用户或用户组执行某些存储过程的权限。此方法严格定义可以对数据库中的数据进行修改的方式。

允许具有特定权限的用户使用 IQ 系统过程管理特定任务：

1. 为每个所需权限创建一个组。
2. 将权限授予指定的组。
3. 向组授予对 IQ 过程的 EXECUTE 权限以便执行权限任务。
4. 创建将被授予权限的新用户后，向此用户授予为该权限创建的组的成员资格。

向用户授予运行相关存储过程的权限

因为大多数权限都是通过组成员资格继承，所以用户可以从该组继承该权限以及对 IQ 过程的执行权限。

前提条件

下列步骤需要 DBA 或 PERMS ADMIN 以及 USER ADMIN 权限。**CREATE USER USERADMIN_GRP** 语句需要 USER ADMIN 权限。

向用户 user1 授予 USER ADMIN 权限以及用于执行与用户管理相关的过程的权限：

过程

1. 创建组 USER ADMIN_GRP

```
CREATE USER USERADMIN_GRP
GRANT GROUP TO USERADMIN_GRP
```

- 向 USERADMIN_GRP 授予 USER ADMIN 权限。

```
GRANT USER ADMIN TO USERADMIN_GRP
```

- 向 USERADMIN_GRP 授予对用于用户管理的 Sybase IQ 存储过程的 EXECUTE 权限。

```
GRANT EXECUTE on sp_iqaddlogin
to USERADMIN_GRP
GRANT EXECUTE on sp_iqcopyloginpolicy
to USERADMIN_GRP
GRANT EXECUTE on sp_iqdroplogin
to USERADMIN_GRP
GRANT EXECUTE on sp_iqmodifyadmin
to USERADMIN_GRP
GRANT EXECUTE on sp_iqmodifylogin
to USERADMIN_GRP
```

- 向 user1 授予 USERADMIN_GRP 组成员资格。用户 user1 通过 USERADMIN_GRP 组的成员资格继承 USER ADMIN 权限以及执行指定 IQ 过程的能力。

```
GRANT MEMBERSHIP IN GROUP USERADMIN_GRP TO user1
```

用于组访问的相关存储过程

您可以创建授予各种相关存储过程权限的组，例如：

组名	授予权限	授予对存储过程的 EXECUTE 权限
OPERATOR_GRP	OPERATOR	sp_iqbackupdetails、 sp_iqbackupsummary、 sp_iqconnection 和 sp_iqsysmon
SPACEADMIN_GRP	SPACE ADMIN	sp_iqdbspace、 sp_iqdbspaceinfo、 sp_iqdbspa- ceobjectinfo、 sp_iqemptyfile、 sp_iqestdbspaces、 sp_iqfile、 sp_iqobjectinfo 和 sp_iqspaceused

有关 Multiplex 示例，请参见《使用 Sybase IQ Multiplex》。

如何确定用户权限

组确实使单个用户的权限变得更加复杂。

假设用户 M_Haneef 已被单独授予对特定表的 SELECT 和 UPDATE 权限，但该用户还是两个组的成员，其中一个组根本没有访问表的权限，另一个组仅有 SELECT 权限。那么，此用户的实际权限是什么呢？

下面给出 Sybase IQ 如何确定用户 ID 是否有权限执行特定操作：

1. 如果用户 ID 拥有 DBA 权限，该用户 ID 可以执行数据库中的任何操作。如果用户具有权限，该用户就具有与该权限关联的权限。
2. 否则，权限取决于指派给单个用户的权限。如果该用户 ID 已被授予执行某操作的权限，则允许执行此操作。
3. 如果没有为该用户进行过单独设置，则权限取决于该用户所属的各个组的权限。如果这些组中的任何一组都有执行某操作的权限，则该用户 ID 由于具有该组中的成员资格而拥有相应的权限，因而允许执行此项操作。
如果不希望某个特定用户访问特定的表、视图或过程，则不要使该用户成为对该对象拥有权限的组的成员。

此方法最大程度地减少了与权限设置顺序相关的问题。

另请参见

- 权限概述（第 283 页）

连接使用的资源

通过构建一个用户和组的集合，您就可以管理对数据库的权限。数据库安全与管理的另一方面是限制单个用户可以使用的资源。

例如，您可能希望阻止单个连接占用过多的可用内存或 CPU 资源，以免该连接导致数据库其他用户的速度变慢。

控制用户资源的数据库选项

Sybase IQ 提供一组 DBA 可用于控制资源的数据库选项。这些选项被称为 *资源调控器*。

如何设置选项

使用 **SET OPTION** 语句可以设置数据库选项。有关语法，请参见《参考：语句和选项》。

可以管理的资源

以下数据库选项可用于管理资源。

- 定义缓冲区中的游标行数。
- 限制包含笛卡尔连接的查询结果的行数。
- 设置用于 IQ 操作的连接的可用处理线程数。
- 设置 IQ 临时存储的高速缓存大小。（建议使用服务器选项 **-iqtc** 设置临时高速缓存大小。）
- 限制可用于任何一个查询的临时数据库空间量。
- 告诉查询优化程序拒绝可能消耗过多资源的查询。如果优化程序估计查询结果集将超出此选项的值，优化程序将拒绝该查询并返回错误消息。

以下数据库选项会影响引擎，但对 Sybase IQ 的影响有限：

- 针对每个连接设置分配给 Java 应用程序的那部分内存的最大值（以字节为单位）。
- 限制一个连接的游标数量。
- 为一个连接限制预准备语句的数量。

数据库选项设置不能通过组结构继承。

针对数据库连接的限制

您可以利用 Sybase IQ 登录管理来控制用户可以使用的资源，这可以限制所有用户或单个用户到某个数据库的连接的数目，设置口令有效期，以及锁定单个用户，使其不能连接到数据库。

另请参见

- 用户帐户和连接管理（第 289 页）

使用过程禁用连接

要禁用连接，请使用存储过程 `sp_iqmodifylogin`。

有关示例，请参见《参考：构件块、表和过程》中的 `sp_iqmodifylogin` 过程。

系统对象中的用户和权限

数据库的当前用户及其权限的相关信息存储在数据库系统表和系统视图中。

大多数系统表由特殊的用户 ID `sys` 拥有。不可能连接到 `sys` 用户 ID。

DBA 对于所有的系统表都具有 `SELECT` 访问权限，就像对于数据库中的任何其它表一样。其他用户对某些表的访问会受到限制。例如，只有 DBA 有权访问 `SYS.SYSUSERPERM` 表，该表中包含所有有关数据库用户的权限的信息以及每个用户 ID 的口令。而 `SYS.SYSUSERPERMS` 是一个视图，其中包含 `SYS.SYSUSERPERM` 中除口令外的所有信息，而且缺省情况下所有用户都具有对该视图的 `SELECT` 访问权限。在新数据库中为 `SYS`、`PUBLIC` 和 `DBA` 设置的权限和组成员资格都完全可以修改。

系统表中的用户 ID、组和权限信息

下表对系统表进行了汇总，这些系统表中包含着有关用户 ID、组和权限的信息。

所有表和视图都属于用户 ID `sys` 所有，因此它们的限定名为 `SYS.SYSUSERPERM` 等。

适当对下面的表进行 `SELECT` 查询会生成存储在数据库中的所有用户 ID 和权限信息。

表	缺省值	目录
SYSUSERPERM	仅限 DBA	各个用户 ID 的数据库级别的权限和口令
SYSGROUP	PUBLIC	用于各组的各个成员的一行
SYSTABLEPERM	PUBLIC	由 GRANT 命令授予的对表的所有权限
SYSOLPERM	PUBLIC	具有 GRANT 命令所授予的 UPDATE 权限的所有列
SYSROCPERM	PUBLIC	每行都保存着一个被授予了使用一个过程的权限的用户

系统视图中的用户 ID、组和权限信息

下表汇总了包含有关用户 ID、组和权限信息的系统视图。

视图	缺省值	目录
SYSUSERAUTH	仅限 DBA	SYSUSERPERM 中除用户号以外的所有信息
SYSUSERPERMS	PUBLIC	SYSUSERPERM 中除口令以外的所有信息
SYSUSERLIST	PUBLIC	SYSUSERAUTH 中除口令以外的所有信息
SYSGROUPS	PUBLIC	SYSGROUP 中的信息，采用了更加易读的格式
SYSTABAUTH	PUBLIC	SYSTABLEPERM 中的信息，采用了更加易读的格式
SYSOLAUTH	PUBLIC	SYSOLPERM 中的信息，采用了更加易读的格式
SYSROCAUTH	PUBLIC	SYSROCPERM 中的信息，采用了更加易读的格式

除了这些以外，还有一些表和视图，它们包含了有关数据库中各个对象的信息。

传输层安全性

使用传输层安全性 (TLS) 可以保护客户端和 IQ 服务器之间或 IQ 客户端和数据库服务器之间通信的安全。

Sybase IQ 数据库的数据库文件加密与 SQL Anywhere 数据库的数据库文件加密类似，如“SQL Anywhere Server - 数据库管理” > “安全性” > “保护数据的安全” > “加密和解密数据库”中所述。

注意： 此参考指向 SQL Anywhere 文档。

Sybase IQ 还可以对列进行加密，如《Sybase IQ 中的高级安全性》中所述。

单独许可的 Sybase IQ “高级安全性”选项包括对 FIPS 加密、Kerberos 鉴定和列加密的支持，《Sybase IQ 中的高级安全性》对此进行了介绍。

IPv6 支持

Sybase IQ 支持 Internet 协议版本 6 (IPv6)，它包含通过 Internet 路由包的寻址信息和控制信息。

IPv6 支持 2^{128} 个唯一 IP 地址，与其之前版本 IPv4 支持的地址数相比有了大幅增加。对于任何可在客户端或服务器上指定 IP 地址的环境，Sybase IQ 均支持 IPv4 和 IPv6 地址。

ODBC 类支持将 IPv6 地址用于远程数据访问。JDBC 类不支持将 IPv6 地址用于远程数据访问。

有关 IPv6 支持的详细信息，请参见“SQL Anywhere Server - 数据库管理” > “启动并连接到数据库” > “通信协议” > “使用 TCP/IP 协议” > “SQL Anywhere 中的 IPv6 支持”。

注意： 此参考指向 SQL Anywhere 文档。

数据完整性

表约束和列约束以及相应的数据类型选择可以确保数据库中的数据有效、可靠。

数据完整性概述

数据完整性表示数据有效（即准确无误），并且数据库的关系结构是完整的。数据库的关系结构通过参照完整性约束（即，使数据在各个表之间保持一致的业务规则）进行描述。

Sybase IQ 支持存储过程和 JDBC，这样，您可以对数据进入数据库的方式进行具体的控制。《系统管理指南第二卷》对过程进行了论述。

有关 JDBC 的信息，请参见“SQL Anywhere Server - 编程” > “JDBC 支持”。

注意： 此参考指向 SQL Anywhere 文档。

哪些情况会导致数据无效

如果不进行适当的检查，下列几种情况可能会导致数据库中的数据无效。使用本章中介绍的功能可防止出现这些情况。

- 操作员在订单表中为客户表中不存在的 **customer_id** 输入订单。
- 操作员在需要数值数据的位置输入文本。
- 操作员输入的数值数据对该列来说太宽。
- 创建了一个 **dept_id** 为 200 的新部门，需要将该部门添加到公司数据库的 **department** 表中 - 但有两名人员将此信息输入到表中。

有效数据的规则和检查

为了便于确保数据库中的数据有效，需要创建检查语句以定义有效数据和无效数据，并设计数据必须遵守的规则。

这些规则通常称为业务规则。检查语句和规则统称为约束。用来维护给定列的数据完整性的规则为列约束。用来维护给定表的一个或多个列的完整性的规则为表约束。可同时表中的单个列应用表约束和列约束。表约束还可以为表中的某组列设置规则。

约束应为内置约束

构建数据库本身的约束比构建客户端应用程序的约束或将约束形成文字说明提供给数据库用户更为可靠。构建数据库的约束是数据库自身定义的一部分，并且可在所有应用程序中一致地强制实施这些约束。

只需在数据库中设置一次约束，对以后与数据库进行的所有交互都将实施该约束，而不管来自任何源。相反，构建客户端应用程序的约束在每次软件更改时都会受到影响，而且可能需要在多个应用程序，或一个客户端应用程序的多个位置中实施。

由于 IQ 数据通常仅由几个用户输入，并且往往从其它数据库直接装载，因此，同 OLTP 数据库相比，IQ 数据库可能更不容易受到导致无效数据的错误类型的影响，具体取决于您使用的提取、转换和装载进程。

应声明适用的任何约束，而不管 Sybase IQ 是否会强制实施这些约束。通过声明约束，您可以确保您了解您的数据需求，并确保设计的数据库符合组织的业务规则。

约束有助于 IQ 优化

Sybase IQ 根据指定的约束执行多种优化。此优化不依赖于约束的强制实施。为使查询和装载操作获取最佳性能，请将所有约束放置到数据库中。

下面列出了一些依赖于构建到数据库中的约束和其它功能的优化类型：

- 连接不同列中的数据的连接索引可优化查询。在大多数情况下，连接索引的连接关系依赖于为连接的表指定的外键约束。
- **FOREIGN KEY**、**PRIMARY KEY** 和 **UNIQUE** 列约束以及 **IQ UNIQUE** 参数可提高装载和查询的性能。

约束检查装载

Sybase IQ 在装载操作期间检查是否遵循特定约束：

- Sybase IQ 确保要装载的数据具有相应的数据类型和长度。
- 如果您具有依赖于外键-主键关系的连接索引，当同步此连接索引时，Sybase IQ 将检查基础表中的数据是否保留有连接列之间的一对多关系。

另请参见

- 创建表的准则（第 171 页）
- 锁定的工作方式（第 347 页）
- IQ PAGE SIZE 参数准则（第 143 页）
- 不同的值数目（第 277 页）
- 使用连接索引（第 212 页）

更改数据库内容的语句

客户端应用程序通过提交 SQL 语句来更改数据库表中的信息。

实际上，只有几个 SQL 语句会修改数据库中的信息：

- 要删除表的现有行，请使用 **DELETE** 语句。
- 要在表中插入新行，请使用 **INSERT** 或 **LOAD TABLE** 语句。
- 要更改单元中的值，请使用 **UPDATE** 语句。

数据完整性工具

为了帮助保持数据完整性，您可以使用数据约束和指定数据库参照结构的约束。

约束

可以对各列或各表中的数据使用几种类型的约束。例如：

- 非空值约束可以防止列包含空值条目。 Sybase IQ 强制实施此约束。
- **CHECK** 条件可指派给列，以便指定该列中的每一行都应满足特定条件。例如，您可以指定工资列中的条目应介于某一指定范围内。
- 可以针对不同列中的相对值设置 **CHECK** 条件，例如，指定图书馆数据库中的 `date_returned` 条目晚于 `date_borrowed` 条目。

列约束可继承自用户定义数据类型。

实体和参照完整性

关系数据库表中的信息通过各个表之间的关系联系起来。这些关系由构建到数据库设计中的候选键和外键定义。

foreign key 由一列或列组合构成。每个外键将一个表 (**foreign** 表) 中的信息与另一个表 (**referenced** 或 **primary**) 中的信息关联起来。外表中的特定列或列组合指定为主表的外键。

具有唯一约束的主键或列（或列集）称为 **candidate key**。引用列或列集必须为候选键，它被称作 **referenced key**。

以下限制可影响候选键：

- 如果存在连接索引，外键不能为候选键。
- 不能对同时作为外键的候选键指定外键约束。

以下完整性规则定义数据库的结构：

- 跟踪主键。确保给定表中的每一行都由一个非 **NULL** 主键唯一地标识。
- 跟踪定义表间关系的外键。所有外键值应与相应主键中的值匹配，或包含空值（如果定义为允许空值）。

另请参见

- 列缺省值有助于数据完整性（第 320 页）
- 表约束和列约束（第 325 页）

用于实施完整性约束的 SQL 语句

以下 SQL 语句用于实施完整性约束：

- 使用此语句可在创建数据库时实施完整性约束。
- 使用此语句可在现有数据库中添加完整性约束或删除约束。

列缺省值有助于数据完整性

每次有人在数据库表中输入一行新数据时，列缺省值都会自动为特定列或一组列分配一个指定的值。

分配的缺省值不需要客户端应用程序执行任何操作，但是，如果客户端应用程序确实为列指定了值，新值将替换列的缺省值。

使用列缺省值可以自动在列中快速填入信息，例如插入某一行的日期或时间或第一个修改表中某行的人员的用户 ID。使用列缺省值有助于保持数据完整性，但不能强制实施数据完整性。客户端应用程序总是可以替换缺省值。

另请参见

- 数据完整性工具（第 319 页）
- 表约束和列约束（第 325 页）

支持的缺省值

Sybase IQ 支持下列列缺省值：

- **CREATE TABLE** 语句或 **ALTER TABLE** 语句中指定的字符串
- **CREATE TABLE** 语句或 **ALTER TABLE** 语句中指定的数字
- 自动递增的数字：每增加一行，新行中该列的值将在原有最大值的基础上加一
- 由 **NEWID** 函数生成的 **UUID**（通用唯一标识符）值
- 当前日期、时间或时间戳
- 当前数据库的名称
- 当前数据库用户的用户 ID 和上次修改行的用户的名称
- **SQL Remote** 应用程序的数据库的发布者用户 ID
- 空值
- 不引用任何数据库对象的常量表达式
- 使用 **CREATE DOMAIN** 语句在用户定义域（数据类型）中指定的所支持的缺省值

缺省值限制

不支持特定的列缺省值。

Sybase IQ 不支持使用下列值作为列缺省值：

- 使用特殊值 **UTC TIMESTAMP**、**CURRENT UTC TIMESTAMP** 和 **GLOBAL AUTOINCREMENT** 的值
- 与该列的数据类型不兼容的缺省值
- 违反表或列的检查约束的缺省值
- 引用数据库对象的常量表达式

Sybase IQ 忽略 **DEFAULT_TIMESTAMP_INCREMENT** 数据库选项的设置。

创建列缺省值

可以使用 **CREATE TABLE** 语句在创建表时创建列缺省值，或使用 **ALTER TABLE** 语句在创建了表之后添加列缺省值。

还可以使用 **CREATE DOMAIN** 语句在创建用户定义域（数据类型）时指定缺省值。

存储过程 **sp_iqcolumn** 返回有关所有表中的所有列的信息。由 **sp_iqcolumn** 的结果集返回的一个列称为“缺省列”，其中显示该列的特定缺省值。

1. 创建名为 `tab1` 的表，并为 **CHARACTER** 列 `c1` 指定缺省特殊值 **LAST USER**：

```
CREATE TABLE tab1(c1 CHAR(20) DEFAULT LAST USER)
```

2. 为 `sales_order` 表中名为 `id` 的现有列添加一个条件，以使该列的值自动递增（除非客户端应用程序指定一个值）：

```
ALTER TABLE sales_order MODIFY id DEFAULT AUTOINCREMENT
```

3. 定义名为 `dom1` 的域，该域的数据类型为 **INTEGER**，缺省值为 45：

```
CREATE DOMAIN dom1 INTEGER DEFAULT 45
```

更改列缺省值

要更改列缺省值，请使用您用来创建缺省值的 **ALTER TABLE** 语句的相同形式。

以下语句将名为 `order_date` 的列的缺省值从其当前设置更改为 **CURRENT DATE**：

```
ALTER TABLE sales_order
MODIFY order_date DEFAULT CURRENT DATE
```

删除列缺省值

要删除列缺省值，请将其修改为空值。

以下语句将删除 `order_date` 列中的缺省值：

```
ALTER TABLE sales_order
MODIFY order_date DEFAULT NULL
```

支持的列缺省值

您可以在 Sybase IQ 中装载和插入列缺省值。

使用以下语句：

- **INSERT...VALUES**
- **INSERT...SELECT**
- **INSERT...LOCATION**
- **LOAD TABLE**
- **UPDATE**
- **SELECT...FROM...FOR UPDATE**

Sybase IQ 在处理列缺省值的定义和插入时应符合下列要求：

- Sybase IQ 允许您指定 Sybase IQ 无法计算的缺省值。如果表具有不支持的缺省值，当对此表执行 **INSERT**、**LOAD** 或 **ALTER ADD** 操作时，将报告错误。
- 当服务器尝试插入的缺省值与该列的数据类型不兼容时，Sybase IQ 将产生错误或警告。例如，如果对整数列定义缺省表达式“N/A”，未指定列值的任何插入或装载将产生错误或警告，具体取决于 **CONVERSION_ERROR** 数据库选项的设置。
- 如果缺省值对 **CHARACTER** 类型列来说太长，Sybase IQ 要么截断该字符串，要么产生异常，具体取决于 **STRING_RTRUNCATION** 数据库选项的设置。
- 如果 **VARCHAR** 或 **LONG VARCHAR** 列的缺省值为零长度字符串，Sybase IQ 要么插入空值，要么插入零长度字符串，具体取决于 **NON_ANSI_NULL_VARCHAR** 数据库选项的设置。
- 如果 **VARCHAR**、**CHAR** 或 **LONG VARCHAR** 列的缺省值的字符串包含部分多字节字符，则 Sybase IQ 可能会在插入值之前剪裁这部分多字节字符，具体取决于 **TRIM_PARTIAL_MBC** 数据库选项的设置。
- 如果列的缺省值违反表或列的检查约束，当服务器每次尝试插入该缺省值时，Sybase IQ 将产生错误。
- 在执行 **LOAD TABLE** 操作期间，因插入缺省值而导致的所有约束冲突将按照任何用户指定的 **IGNORE CONSTRAINT** 和 **MESSAGE LOG/ROW LOG** 选项进行应用。
- Sybase IQ 不支持 **UTC TIMESTAMP** 和 **CURRENT UTC TIMESTMAP** 的列缺省值。每次尝试插入或更新这些类型的列的缺省值时将报告错误。
- 在基表上定义的列缺省值不会传播到这些表参与的连接中。
- 参与连接索引的表上不允许列缺省值，如果尝试对此类表定义缺省值，Sybase IQ 将产生错误。此规则类似于对 **AUTOINCREMENT** 缺省值的支持。
- 如果在表中添加定义有缺省值的列，将使用此缺省值填充新列中的所有行。
- 更改表中现有列的缺省值不会更改表中的任何现有值。
- 要使用 **LOAD TABLE** 语句 **DEFAULT** 选项中指定的缺省值，必须将 **LOAD TABLE DEFAULTS** 选项设置为 **ON**。如果 **DEFAULTS** 选项为 **OFF**，则不会使用指定的装载缺省值，而会在列中插入空值。
- **LOAD TABLE DEFAULT** 规范包含至少一个需要从 **LOAD TABLE** 命令中指定的文件装载的列。
- **LOAD TABLE DEFAULT** *default-value* 必须属于数据库的字符集，且必须符合列支持的缺省值和缺省值限制。**LOAD TABLE DEFAULT** 选项不支持 **AUTOINCREMENT**、**IDENTITY** 或 **GLOBAL AUTOINCREMENT** 作为缺省值。
- 对于在 **LOAD TABLE DEFAULT** 子句中指定的装载缺省值，不支持对缺省值进行加密。

有关定义和插入列缺省值的详细信息，请参见本节稍后部分中特定缺省值类型的单独一节。有关缺省列值表达式中可使用的特殊值的详细信息，请参见《参考：构件块、表和过程》。

另请参见

- IQ 中的显式数据转换（第 259 页）

Sybase Central 中的列缺省值

在 Sybase Central 中，可以使用列属性表的“值”选项卡来添加、更改和删除列缺省值。

注意：当创建新列时，某些属性处于隐藏状态，您可以选择“数据类型”或“值”并单击省略号来显示这些属性。

日期、时间和时间戳缺省值

对于数据类型为 DATE、TIME 或 TIMESTAMP 的列，您可以使用 **CURRENT DATE**、**CURRENT TIME**、**TIMESTAMP** 或 **CURRENT TIMESTAMP** 特殊值作为缺省值。您选择的缺省值必须与该列的数据类型兼容。

CURRENT DATE 缺省值示例

使用 **CURRENT DATE** 作为缺省值可以：

- 在联系人数据库中记录通话日期
- 在销售条目数据库中记录订单日期
- 在图书馆数据库中记录借阅人的借书日期

CURRENT TIMESTAMP 缺省值

CURRENT TIMESTAMP 与 **CURRENT DATE** 缺省值相似，但精度更高。例如，联系管理应用程序的用户在一天中可能要与同一个客户联系多次：使用 **CURRENT TIMESTAMP** 缺省值可以区分这些联系。

由于 **CURRENT TIMESTAMP** 记录的日期和时间可以精确到百万分之一秒，因此，当事件的顺序在数据库中非常重要时，使用 **CURRENT TIMESTAMP** 很有用。

TIMESTAMP 缺省值

当使用 **DEFAULT TIMESTAMP** 声明列时，系统将为插入和装载操作提供缺省值。只要更新此行，便会用当前日期和时间更新该值。

在执行 **INSERT** 和 **LOAD** 操作时，**DEFAULT TIMESTAMP** 与 **CURRENT TIMESTAMP** 的效果相同。在执行 **UPDATE** 操作时，如果未显式修改缺省值为 **TIMESTAMP** 的列，则该列的值将更改为当前日期和时间。

Sybase IQ 不支持 **UTC TIMESTAMP** 或 **CURRENT UTC TIMESTAMP** 的缺省值，也不支持数据库选项 **DEFAULT_TIMESTAMP_INCREMENT**。每次尝试插入或者更新 **UTC TIMESTAMP** 或 **CURRENT UTC TIMESTAMP** 类型的列的缺省值时，Sybase IQ 都会产生错误。

USER 缺省值

将 **DEFAULT USER** 分配给列可作为一种简单又可靠的方法来标识在数据库中输入信息的人。例如，销售人员在管理佣金时可能需要该信息。

为表的主键设置用户 **ID** 缺省值对于偶尔连接的用户非常有用，并且有助于避免在更新信息时发生冲突。这些用户可以将与他们的工作相关的表复制到便携式计算机上，在不连接到多用户数据库的情况下进行更改，等到他们回来后，再将事务日志应用到服务器上。

USER 缺省值

特殊值 **USER** 和 **CURRENT USER** 返回的字符串包含当前连接的用户 **ID**，这些特殊值在字符数据类型的列中可用作缺省值。在执行 **UPDATE** 操作时，系统不会更改缺省值为 **USER** 或 **CURRENT USER** 的列。

LAST USER 缺省值

特殊值 **LAST USER** 返回上次修改行的用户的名称，此特殊值在字符数据类型的列中可用作缺省值。在执行 **INSERT** 和 **LOAD** 操作时，此常量与 **CURRENT USER** 的效果相同。在执行 **UPDATE** 操作时，如果未显式修改缺省值为 **LAST USER** 的列，则该列将更改为当前用户的名称。

与 **DEFAULT TIMESTAMP** 组合时，**LAST USER** 缺省值可用于记录（在单独的列中）用户和上次更改行的日期和时间。

IDENTITY 或 AUTOINCREMENT 缺省值

IDENTITY/AUTOINCREMENT 缺省值适用于数值值本身并没有含义的数值数据字段。

每新增一行，该列中新行的值将在原有最大值的基础上加一。可以使用 **IDENTITY/AUTOINCREMENT** 列记录订购单编号、标识客户服务呼叫或其它需要标识号的条目。

自动增量列通常是主键列，或限制为只能包含唯一值的列（请参见《参考：语句和选项》中的 **CREATE TABLE** 语句）。例如，自动增量缺省值在该列是索引的第一列时会起很大作用，因为服务器使用索引或键定义来查找最大值。

有时，您可以使用 **@@identity** 全局变量来检索插入到自动增量列中的最新值。

Sybase IQ 不支持特殊值 **GLOBAL AUTOINCREMENT**。

自动增量和负数

IDENTITY/AUTOINCREMENT 设计用于正整数。

创建表时，**IDENTITY/AUTOINCREMENT** 的初始值设置为 0。

自动增量和 IDENTITY 列

具有 **AUTOINCREMENT** 缺省值的列在 Transact-SQL 应用程序中称为 **IDENTITY** 列。Sybase IQ 支持两个关键字，以实现兼容。

NEWID 缺省值

UUID（通用唯一标识符）又称为 **GUID**（全局唯一标识符），可用于唯一标识表中的行。

这样生成值会导致在一台计算机上生成的值与在另一台计算机上生成的该值不匹配。因此，**UUID** 可以作为键用在复制和同步环境中。

请参见《参考：构件块、表和过程》中的 **NEWID** 函数。

空值缺省值

对于允许使用空值的列，指定 **NULL** 缺省值与不指定任何缺省值完全相同。

如果插入行的客户端未显式分配值，该行会自动获得一个空值。

如果某些列的信息是可选的或者并不总有信息，则可以使用 **NULL** 缺省值。

有关空值的详细信息，请参见《参考：构件块、表和过程》>“SQL 语言元素”>“空值”。

字符串和数字缺省值

只要某一列的数据类型为字符串或数字，就可以指定一个特定字符串或数字作为该列的缺省值。

必须确保指定的缺省值可以转换为该列的数据类型。

缺省字符串和数字适用于具有典型条目的给定列。例如，如果公司有两个办事处，总部位于 **city_1**，而较小的办事处位于 **city_2**，则可能需要将地点列的缺省条目设置为 **city_1**，以便更容易地输入数据。

常量表达式缺省值

可以使用不引用任何数据库对象的常量表达式作为缺省值。

GETDATE 和 **DATEADD** 等函数可用于常量表达式缺省值。如果缺省常量表达式不是函数或简单值，必须用括号将该表达式括起来。

例如，使用常量表达式可以使列缺省值包含自今天起第十五天的日期等这样的条目：

```
... DEFAULT ( DATEADD( DAY, 15, GETDATE() ) )
```

表约束和列约束

约束可帮助确保表中输入的数据准确无误，并向 **Sybase IQ** 提供信息以提高性能。

CREATE TABLE 语句和 **ALTER TABLE** 语句可以指定许多不同的表属性。除基本表结构（列的数量、名称和数据类型，表的名称和位置）外，您还可以指定用于控制数据完整性的其它功能。

警告! 更改或创建表可能会对数据库的其他用户有负面影响。对于较大的表，**ALTER TABLE** 或 **CREATE TABLE** 操作可能十分耗时。**CREATE TABLE** 处理将延迟其它 IQ 进程的执行，直至语句完成。尽管可以在其它连接处于活动状态时执行 **ALTER TABLE** 语句，但当任何其它连接正在使用要更改的表时，便无法执行这些语句。在 **ALTER TABLE** 期间，在处理该语句的过程中，不允许请求引用正在更改的表。

另请参见

- 列缺省值有助于数据完整性 (第 320 页)
- 数据完整性工具 (第 319 页)

对列或表的 UNIQUE 约束

UNIQUE 约束指定一列或多列唯一标识表中的各行。如果应用 **UNIQUE** 约束，Sybase IQ 将强制实施此条件。

UNIQUE 实质上与 **PRIMARY KEY** 约束相同，不同之处在于，您可以在表中指定多个 **UNIQUE** 约束。使用 **UNIQUE** 和 **PRIMARY KEY**，列不能包含任何 **NULL** 值。

示例 1

下面的示例在 `employee` 表中添加 `ss_number` 列，并确保该列中的每个值在整个表中唯一。

```
ALTER TABLE employee ADD ss_number char(11) UNIQUE
```

示例 2

在本示例中，需要添加三列才能生成一个唯一条目。

```
ALTER TABLE product ADD UNIQUE (name, size, color)
```

另请参见

- FP(3) 索引 (第 197 页)
- 配置 FP(3) 索引 (第 199 页)

对列的 IQ UNIQUE 约束

IQ UNIQUE 约束指定列中不同的值的估计数目。您可以对表中的任何列应用 **IQ UNIQUE** 约束。此约束有助于优化装载索引。

在 Sybase Central IQ 插件中，您可以在列属性页上添加 **IQ UNIQUE** 约束。有关详细信息，请参见联机帮助。

例如，在 `employee` 表的 `state` 列中，应指定 **IQ UNIQUE(50)** 以表明仅存在 50 个可能值（假定仅限美国）。每个可能值可以出现多次。

如果 `MINIMIZE_STORAGE` 数据库选项为 **ON**，则它等效于对所有新列指定 **IQ UNIQUE(255)**。从 12.6 版开始，缺省情况下，此选项为 **OFF**。

列的 CHECK 条件

使用 CHECK 条件可以指定一列中的值必须满足某个明确的条件。

您可以对单个列中的值应用 CHECK 条件，以便指定这些值应遵循的规则。可能必须满足这些规则才能确保数据的合理性，或者可能需要使用更严格的规则来反映组织的政策和程序。

如果某列的有效值限制在一定范围内，则可以单独对该列使用 CHECK 条件。下面是一些示例：

示例 1

可以指定输入的值应与一组有限数量的值中的一个相匹配。例如，要指定 city 列只包含几个允许城市（例如，组织设有办事处的那些城市）中的一个，可以使用类似如下的约束：

```
ALTER TABLE office MODIFY city CHECK ( city IN ( 'city_1', 'city_2', 'city_3' ) )
```

缺省情况下，比较字符串时不区分大小写，除非创建数据库时使用 **CASE RESPECT** 选项显式指定需要区分大小写。请参见《参考：语句和选项》的 **CREATE DATABASE** 语句中的 CASE 子句。

示例 2

可以指定日期或数字在特定的范围内。例如，您可能需要确保雇员表中的 **start_date** 列必须在组织成立日期和当前日期之间，如下所示：

```
ALTER TABLE employee MODIFY start_date CHECK ( start_date BETWEEN '1983/06/27' AND CURRENT DATE )
```

有多种日期格式可以使用：不管当前选项如何设置，数据库都可以识别本例中使用的 YYYY/MM/DD 格式。

用户定义数据类型的 CHECK 条件

您可以向用户定义数据类型附加 CHECK 条件。使用这些数据类型定义的列将继承 CHECK 条件。为列显式指定的 CHECK 条件将替换用户定义数据类型中的 CHECK 条件。

当对用户定义数据类型定义 CHECK 条件时，任何前缀为 @ 符号的变量都会在计算 CHECK 条件时替换为列名。例如，以下用户定义数据类型仅接受正整数：

```
CREATE DATATYPE posint INT  
CHECK ( @col > 0 )
```

可以使用任何前缀为 @ 的变量名代替 @col。任何使用 **posint** 数据类型定义的列都只接受正整数，除非该列显式指定了不同 CHECK 条件。

带有 DELETE CHECK 子句的 ALTER TABLE 语句将从表定义中删除所有 CHECK 条件，包括从用户定义数据类型继承的那些 CHECK 条件。

有关用户定义数据类型的信息，请参见《参考：构件块、表和过程》。

在 Sybase Central 中添加、更改和删除列约束

在 Sybase Central 中，添加、更改和删除列约束的所有操作都是在列属性表的“约束”选项卡中执行的。

1. 连接到数据库。
2. 单击该数据库的“表”文件夹，然后单击包含要更改的列的表。
3. 选择一列，然后单击“文件” > “属性”。

有关详细信息，请参见 Sybase Central 联机帮助。

列的 CHECK 条件

CHECK 条件可以作为约束应用于表，而不是单个列。

此 CHECK 条件通常指定要输入或修改的行中的两个值之间具有正确的关系。列 CHECK 条件单独保存在系统表中，可以单独替换或删除它们。这种情况比较灵活，因此建议您尽可能对单个列使用 CHECK 条件。

在 Interactive SQL 中添加 CHECK 条件

您可以向表或某个列添加新的 CHECK 条件。

例如，在图书馆数据库中，特定条目的 `date_returned` 列必须晚于 `date_borrowed` 条目，或与其相同：

```
ALTER TABLE loan
ADD CHECK(date_returned >= date_borrowed)
```

在 Interactive SQL 中删除 CHECK 条件

您可以从单个列中删除 CHECK 条件。

要删除列的 CHECK 条件，请将其设置为空值。

以下语句将删除 `customer` 表中 `phone` 列的 CHECK 条件：

```
ALTER TABLE customer MODIFY phone
CHECK NULL
```

在 Interactive SQL 中替换 CHECK 条件

您可以从单个列中删除 CHECK 条件。

您可以用与添加 CHECK 条件相同的方法来替换列的 CHECK 条件。

以下语句在 `office` 表的 `city` 列中添加或替换 CHECK 条件：

```
ALTER TABLE office
MODIFY city
CHECK ( city IN ( 'city_1', 'city_2', 'city_3' ) )
```

表的 CHECK 条件

同在列中定义的 CHECK 条件相比，有两种方法可以修改在表中定义的 CHECK 条件。

- 使用带有 ADD 表约束子句的 **ALTER TABLE** 添加新的 CHECK 条件。
- 使用 **ALTER TABLE DELETE CHECK** 删除所有现有 CHECK 条件，包括列 CHECK 条件，然后再添加新的 CHECK 条件。

从表中删除列并不会删除表约束中保存的与该列相关联的 CHECK 条件。如果未删除约束，则查询表中的数据任何尝试都将产生错误消息 `column not found`。

使用 Interactive SQL 删除表的 CHECK 条件

使用带有 **DELETE CHECK** 子句的 **ALTER TABLE** 语句可以删除表中的所有 CHECK 条件，包括该表所有列中的 CHECK 条件以及继承自用户定义数据类型的 CHECK 条件。

例如：

```
ALTER TABLE table_name  
DELETE CHECK
```

实体和参照完整性

数据库服务器通过数据库的关系结构识别数据库中的信息。Sybase IQ 还确保依赖于各个表之间的主键-外键关系的任何连接索引中的所有行都能够正确保持这些关系。

另请参见

- 星形连接 (第 222 页)
- 创建外键 (第 178 页)
- 创建外键 (第 331 页)

如何声明实体完整性

为每个表指定了主键后，客户端应用程序开发人员或数据库管理员不需要采取任何其它措施即可保持实体完整性。

表的所有者在创建表时为表定义主键。如果以后要修改表的结构，也可以使用 **ALTER TABLE** 语句子句 **DELETE PRIMARY KEY** 或 **ADD PRIMARY KEY** 重新定义主键。请参见《参考：语句和选项》。

使用某些应用程序开发系统和数据库设计工具可以创建或更改数据库表。如果使用这种系统，可能不必显式输入 **CREATE TABLE** 或 **ALTER TABLE** 命令：该应用程序自动使用您提供的信息生成语句本身。请参见《参考：语句和选项》。

如何强制实施实体完整性

插入或更新表中的行后，数据库服务器确保该表的主键仍然有效：该表中的每一行都由主键唯一地标识。

示例 1

演示数据库中的 **Employees** 表使用雇员 ID 作为主键。在该表中添加了一个新雇员后，IQ 将检查新雇员 ID 值是否唯一，以及是否非 NULL。有关演示数据库中表的列表，请参见《Sybase IQ 简介》>“关于 Sybase IQ”>“演示数据库”>“表名和所有者”。

示例 2

演示数据库中的 **SalesOrderItems** 表使用两个列来定义主键。

此表保存有关订购项的信息。一列包含指定订单的 **id**，但每个订单可能涉及几个项目，因此此列本身不能作为主键。另一 **line_id** 列标识与项目对应的行。将 **id** 和 **line_id** 这两个列结合起来可以唯一地指定项目，并形成主键。这称为 *多列主键*。

如果客户端应用程序破坏了实体完整性

实体完整性要求主键的每个值或唯一约束在该表中都是唯一的，并且不能有空值。

如果客户端应用程序尝试插入或更新一个主键值，但提供的值不唯一，则会破坏实体完整性。

如果实体完整性被破坏，将导致不会将新信息添加到数据库中，而会向客户端应用程序发送一条错误消息。

应用程序编程人员应确定如何将该信息传达给用户，以便于用户采取适当的措施。此种情况下，适当的措施通常很简单，只需为主键提供一个唯一的值。

Sybase IQ 为外键或候选键上的每个 **UPDATE**、候选键上的每个 **DELETE** 和外键上的每个 **LOAD/INSERT** 检查参照完整性。当发生参照完整性冲突时，将立即拒绝并回退 **UPDATE** 或 **DELETE** 请求。还将拒绝或回退违反参照完整性的 **LOAD/INSERT** 请求。Sybase IQ 也会选择拒绝违反用户指定的数据完整性的行。

参照完整性

要使外键关系有效，外键中的条目必须与参照表中某一行的主键值相对应。

有时，可能会引用其它某个唯一的列组合，而不引用主键。具有唯一约束的主键或列（或列集）称为 **candidate key**。引用列或列集必须为候选键，它被称作 **referenced key**。

创建外键

使用 **CREATE TABLE** 语句或 **ALTER TABLE** 语句创建外键，其创建方法与主键的创建方法相同。

注意：不能对本地临时表创建外键约束。全局临时表必须用 **ON COMMIT PRESERVE ROWS** 创建。

另请参见

- 实体和参照完整性（第 329 页）
- 星形连接（第 222 页）
- 创建外键（第 178 页）

外键示例

演示数据库包含一个雇员表和一个部门表。雇员表的主键是雇员 ID，部门表的主键是部门 ID。

例如，采用以下 schema：

```
DEPT table { DeptNo int primary key DeptName varchar(20), Mgr int,
foreign key MGR_EMPNO (Mgr) references EMPLOYEE(EmpNo) on update
restrict } EMPLOYEE table { EmpNo int primary key, DeptNo int
references DEPT(DeptNo) on delete restrict, LastName varchar(20),
FirstName varchar(20), Salary int }
```

在雇员表中，部门 ID 是部门表的外键；雇员表中的每个部门 ID 严格对应于部门表中的部门 ID。

外键关系是一个多对一关系。雇员表中的几个条目都具有相同的部门 ID 条目，但部门 ID 是部门表的主键，因此在部门表中是唯一的。如果外键引用部门表中包含重复条目的列，则无法获知部门表中的哪一行是适当的引用。这是一个强制外键。

参照完整性违规

Sybase IQ 通过 **RESTRICT** 操作（ANSI 缺省设置）在语句级支持参照完整性。

这表示，Sybase IQ 拒绝主键或具有唯一约束的列上的更新和删除请求，该唯一约束删除对应外键所依赖的任何值。（必须注意请求删除和更新的顺序。）Sybase IQ 将发出错误消息，并回退违反参照完整性的装载操作，但允许您指定要忽略的特定行。

另请参见

- 如何禁用参照完整性检查（第 336 页）

使用现有未受强制的外键强制实施参照完整性

您可以使用未受强制的外键强制实施参照完整性。

1. 标识候选键-外键关系。

在外键示例的 schema 中，有两种此类关系：

- 外键 (EMPLOYEE.DeptNo) 与候选键 (DEPT.DeptNo) 的关系
 - 外键 (DEPT.Mgr) 与候选键 (EMPLOYEE.EMPNO) 的关系
2. 通过 **ALTER TABLE** 语句添加主键或对候选键添加唯一约束 (如果不存在)。(在上面的示例中, 已存在主键。) 所有候选键值必须唯一, 并且非 **NULL**。
 3. 通过 **ALTER TABLE** 语句删除未受强制的外键约束 (如果有)。

例如:

```
ALTER TABLE DEPT DROP FOREIGN KEY MGR_EMPNO; ALTER TABLE EMPLOYEE  
DROP FOREIGN KEY DEPT;
```

在上面的 schema 中, 我们需要删除引用 DEPT(DeptNo) 的未受强制的外键约束 MGR_EMPNO 和 EMPLOYEE(DeptNo)。如果 EMPLOYEE(DeptNo) 与 DEPT(DeptNo) 的关系没有用户指定角色名称, 缺省角色名称则与主表相同, 也即 DEPT。

4. 添加外键约束。例如:

```
ALTER TABLE DEPT ADD FOREIGN KEY MGR_EMPNO(Mgr) REFERENCES  
EMPLOYEE(EmpNo);  
ALTER TABLE EMPLOYEE ADD FOREIGN KEY EMP_DEPT(DeptNo) REFERENCES  
DEPT(DeptNo);
```

在新表中强制实施参照完整性

创建新表时, 请按如下方式强制实施参照完整性:

强制实施参照完整性:

- 创建主表, 例如:

```
CREATE TABLE DEPT(DeptNo int primary key, DeptName varchar(20),  
Mgr int );
```

- 创建外表。例如, 在此语句中, 指定外键的缺省角色名称为 DEPT:

```
CREATE TABLE EMPLOYEE(EmpNo int primary key, DeptNo int references  
DEPT(DeptNo) on delete restrict, LastName varchar(20), FirstName  
varchar(20), Salary int);
```

- 添加外键约束。例如:

```
ALTER TABLE DEPT ADD FOREIGN KEY MGR_EMPNO(Mgr) REFERENCES  
EMPLOYEE(EmpNo);
```

示例 1

下面提供了创建外表的另一种方法。在此语句中, 同一外键的用户指定角色名称为 EMP_DEPT:

```
CREATE TABLE EMPLOYEE(EmpNo int primary key, DeptNo int, LastName  
varchar(20), FirstName varchar(20), Salary int, FOREIGN KEY  
EMP_DEPT(DeptNo) REFERENCES DEPT(DeptNo));
```

删除外键约束。

- 如果未指派有角色名称, 如在第一个 **CREATE TABLE** 示例中, 则指定外键的缺省角色名称为 DEPT:

```
ALTER TABLE EMPLOYEE DROP FOREIGN KEY DEPT;
```

如果有多个外键且角色名称未知，则可以使用 **sp_iqconstraint** 过程显示角色名称。请参见《参考：构件块、表和过程》。

- 在第二个 **CREATE TABLE** 示例中，分配的角色名称为 **EMP_DEPT**，因此必须在删除键时指定此角色名称，如下所示：

```
ALTER TABLE EMPLOYEE DROP FOREIGN KEY EMP_DEPT;
```

示例 2

上述语句无法删除隐式创建的 **EMPLOYEE(DeptNo)** 的非唯一 **HG** 索引。要删除该索引，请使用 **sp_iqindex** 找到 **HG** 索引名，并使用 **DROP INDEX** 语句删除它，如下所示：

```
sp_iqindex( 'EMPLOYEE' );
```

```
EMPLOYEE DBA DeptNO FP ASIQ_IDX_T27_C2_FP N EMPLOYEE DBA DeptNO HG
ASIQ_IDX_T27_C2_HG N EMPLOYEE DBA EmpNO FP ASIQ_IDX_T27_C1_FP N
EMPLOYEE DBA EmpNO HG ASIQ_IDX_T27_I11_HG N EMPLOYEE DBA FirstName FP
ASIQ_IDX_T27_C4_FP N EMPLOYEE DBA LastName FP ASIQ_IDX_T27_C3_FP N
EMPLOYEE DBA Salary FP ASIQ_IDX_T27_C5_FP N
```

```
DROP INDEX ASIQ_IDX_T27_C2_HG
```

示例 3

要删除表，必须删除关联的所有外键约束。请按以下顺序删除外键约束和表：

```
ALTER TABLE DROP FOREIGN KEY MGR_EMPNO; DROP TABLE EMPLOYEE; DROP
TABLE DEPT;
```

删除同一表的另一种方法是：按任意顺序使用下列两个 **ALTER TABLE** 语句，然后按任意顺序使用 **DROP TABLE** 语句：

```
ALTER TABLE DEPT DROP FOREIGN KEY MGR_EMPNO; ALTER TABLE EMPLOYEE
DROP FOREIGN KEY EMP_DEPT;
```

示例 4

假设该数据库中还包含一个办事处表，列出了办事处的地点。雇员表中可能有一个办事处表的外键，显示雇员工作的办事处的位置。该数据库的设计人员可以允许在聘用雇员时不指派办事处地点。在本示例中，外键应允许空值，因为办事处地点未知或该雇员不在办事处之外工作。>

参照完整性丢失

Sybase IQ 对参照完整性丢失提供保护。

如果有人执行了以下操作，数据库可能会丢失参照完整性：

- 更新或删除具有匹配外键值的主键值。引用该主键的所有外键都将违反参照完整性。
- 在外表中添加一个新行，并为外键输入一个没有相应候选键值的值。数据库将违反参照完整性。

Sybase IQ 对这两类完整性丢失都提供保护。

当更新或删除引用的候选键时，Sybase IQ 不允许 **UPDATE** 或 **DELETE**。

并发操作

Sybase IQ 的参照完整性功能限制在对外表执行装载或插入时对主表进行并发更新或删除。

表 44. 返回了 IQ 错误的并发操作

第一个请求	重叠事务请求
某个事务请求任何外表的 LOAD/INSERT/UPDATE/ ALTER TABLE ADD 外键/ALTER TABLE DROP 外键	对其关联的带有可删除行的主表执行 DELETE 。
	对其关联主表执行 UPDATE 。
	对其关联主表执行 TRUNCATE 。

Sybase IQ 也会在下面这种情况下产生错误：某个事务请求 **ALTER TABLE ADD 外键**或 **DROP 外键**，但其它事务正在使用外表和/或主表的旧版本。

对于已强制实施或未受强制的外键和主键，Sybase IQ 允许：

- 对一个或多个外表以及共享主表同时执行 **LOAD/INSERT**。
- 对外表执行 **LOAD/INSERT**，并同时对外另外一个或多个外表执行 **DELETE/UPDATE/ TRUNCATE TABLE**。
- 对 2 个或多个外表同时执行 **DELETE/UPDATE/TRUNCATE TABLE**，即使这些外表共享同一个主表。
- 对外表执行 **DELETE/TRUNCATE TABLE**，并同时共享主表执行 **DELETE/UPDATE/ TRUNCATE TABLE**。
- 如果没有任何事务正在使用外表/主表的旧版本，则对外键执行 **ALTER TABLE ADD** 或 **DROP**，并且这些未使用的旧版本将作为 **ADD/DROP 外键**操作的一部分删除。

对外表和主表执行并发操作

Sybase IQ 的表级别版本控制允许在对外表执行 **LOAD/INSERT/UPDATE** 操作的同时对主表并发执行 **LOAD/INSERT** 操作，并确保一致的参照完整性检查。

当请求 **DELETE** 或 **UPDATE** 的事务启动时，Sybase IQ 还检验删除的旧值是否不存在于外表中。这样，可以在对外表执行 **DELETE** 的同时对主表并发执行 **DELETE/ UPDATE**，并提供一致的参照完整性检查。

为了理解对外表和主表的并发操作，假定两个外表 (ftab1 与 ftab2) 和一个主表 (ptab) 之间存在两个外键约束。假定外键 ftab1 (fk1, fk2) 引用候选键 ptab(pk1, pk2)。外键 ftab2 (fk1, fk2) 引用同一候选键。候选键 ptab(pk1, pk2) 既可以是主键，也可以是唯一约束。

此表显示外表和主表上应允许哪些操作以及哪些操作将返回错误。该表仅适用于强制实施的外键和候选键。

表 45. 对外表和主表执行并发 DML

	LOAD 或 INSERT ftab1	DE- LETE/ TRUN- CATE TABLE ftab1	UPDATE ftab1 (fk1,fk2)	填充新索引 (非 FK ftab1 (fk1,fk2))	ADD FK ftab1 (fk1 fk2)	DROP FK ftab1 (fk2, fk2)
LOAD ftab2	允许	允许	允许	允许	允许	允许
LOAD ptab	允许	允许	允许	允许	允许	允许
INSERT ftab2	允许	允许	允许	允许	允许	允许
INSERT ptab	允许	允许	允许	允许	允许	允许
DELETE ftab2 TRUNCATE TA- BLE ftab2	允许	允许	允许	允许	允许	允许
DELETE ptab TRUNCATE TA- BLE ptab	错误	允许	错误	允许	错误	错误
UPDATE ftab2(fk1,fk2)	允许	允许	允许	允许	允许	允许
UPDATE ptab (pk1,pk2)	错误	允许	错误	允许	错误	错误
填充新索引	允许	允许	允许	允许	允许	允许
QUERY (正在使 用的带有或不带有 (fk1,fk2) 的旧版本 ftab1/ptab)	允许	允许	允许	允许	错误	错误
未使用旧版本 ftab2	不适用	不适用	不适用	不适用	允许 (删 除所有未 使用的旧 版本 ftab1)	允许 (删 除所有未 使用的旧 版本 ftab1)

如果某一事务装载外键列，而另一事务更新关联的候选键列，则会发生并发冲突。如果某一事务装载外键列，而另一事务更新其中一个关联候选表上的非关联候选键列，则不会发生并发冲突。

注意： 为了提高性能，查询 **union all** 视图将打开用作连接键的列或分组依据列所引用的表。在提交事务并释放表上的读取锁之前，您无法更改或删除其外键用作连接条件或分组列的表。但是，可以在运行查询时装载、插入、删除和更新这些表。

如何禁用参照完整性检查

您可以根据需要使用 Sybase IQ 选项 `DISABLE_RI_CHECK` 绕过参照完整性检查。

由于绕过参照完整性检查会使采用此功能的目的落空，Sybase 建议您慎重使用此选项。

另请参见

- 参照完整性违规（第 331 页）

系统表中的完整性规则

有关数据库中的完整性检查和规则的所有信息都存储在以下系统表和视图中：

系统表	描述
<code>SYS.SYSTABLE</code>	<code>CHECK</code> 约束保存在 <code>SYS.SYSTABLE</code> 的 <code>view_def</code> 列中。对于视图， <code>view_def</code> 包含创建视图的 CREATE VIEW 命令。您可以通过查看 <code>table_type</code> 列（是 <code>BASE</code> 或 <code>VIEW</code> ）来检查一个特定表是基表还是视图。
<code>SYS.SYSFOREIGN-KEYS</code>	此视图以更便于阅读的格式显示 <code>SYS.SYSFOREIGNKEY</code> 和 <code>SYS.SYSFKCOL</code> 两个表中的外键信息。
<code>SYS.SYSCOLUMNS</code>	此视图以更便于阅读的格式显示 <code>SYS.SYSCOLUMN</code> 表中的信息，其中包括列的缺省设置和主键信息。

有关每个系统表的内容说明，请参见《参考：构件块、表和过程》。可以使用 Sybase Central 或 `dbisql` 浏览这些表和视图。

事务和版本控制

Sybase IQ 的事务处理方法（称为“快照版本控制”）对数据管理的性能和其它方面具有影响。

另请参见

- 管理用户 ID 和权限（第 281 页）
- 修改数据的权限（第 228 页）
- 安全概述（第 4 页）
- 中间版本控制（第 15 页）
- 表版本控制可控制对连接索引的访问（第 225 页）

事务和版本控制概述

事务处理可确保将逻辑上相关的命令作为一个单元执行。

Sybase IQ 使用事务处理来允许多名用户在数据库更新时从数据库读取数据。事务是保持数据精确性以及当系统出现故障时执行数据恢复的基础。

事务处理的一个至关重要的方面是其所具有的使用户免受其他用户的事务影响的能力。Sybase IQ 的事务处理方法（称为*快照版本控制*）支持 ISO 识别的最高隔离级别。

事务

事务是指 SQL 语句组。

每个事务都执行一项将数据库从一种一致状态更改为另一种一致状态的任务。这些单元在保护数据库免受介质和系统故障影响方面以及保持数据一致性方面起着重要作用。

逻辑工作单元

事务是一个逻辑工作单元。

每个事务都是一系列在逻辑上相关的命令，这些命令可完成一项任务，并将数据库从一种一致状态转换为另一种一致状态。

事务是原子性的。换言之，Sybase IQ 将事务中的所有语句作为一个单元执行。在每个事务结束时，可提交更改以使其成为永久性更改。如果由于某种原因而无法正确处理事务中的任何命令，则会撤消或回退部分或全部中间更改。用户应用程序控制提交或回退更改的条件。

事务可将每名用户的工作分为一些小块。完成每个块都会标记一个点，信息在这个点处是自我一致的。事务处理是确保数据库包含正确信息的基础。

注意： Sybase IQ 处理事务的方式与 SQL Anywhere 在没有 IQ 的情况下运行时处理事务的方式极为不同。本章将介绍 Sybase IQ 如何处理事务。如果您使用的只是 Anywhere 数据库，请参见“SQL Anywhere Server - SQL 用法” > “使用事务和隔离级别” > “锁定的工作方式”以了解有关事务和锁定的信息。

注意： 此参考指向 SQL Anywhere 文档。

命令和事务

Sybase IQ 允许将命令分组为事务。

在大多数情况下，IQ 事务根据所发出的命令以及所设置的选项自动开始和结束。您也可以发出显式命令来开始或结束事务。

启动事务的事件

某些事件可以自动启动事务。

事务自动以下列事件之一开始：

- 在与数据库建立连接后执行的第一个语句。
- 在上一个事务结束后执行的第一个语句。

Sybase IQ 也支持 Transact-SQL 命令（例如 **BEGIN TRANSACTION**），以便和 Adaptive Server Enterprise 兼容。通过 IQ，可以使用 **BEGIN TRANSACTION** 命令显式启动事务。

结束事务的事件

某些事件可完成事务。

以下事件之一将完成事务：

- 执行 **COMMIT** 语句导致对数据库进行的更改成为永久性更改。
- 执行 **ROLLBACK** 语句撤消由事务执行的所有更改。
- 与数据库断开连接导致隐式回退（缺省设置）或提交。
- 执行具有自动提交副作用的语句。

数据库定义命令（例如 **ALTER**、**CREATE** 和 **DROP**）均具有自动提交的副作用。您也可以使用两个 **dbisql** 选项来促使提交自动进行。

数据高速缓存和页面更新

从高速缓存读取数据以及向高速缓存写入数据将减少 Sybase IQ 必须访问磁盘的次數。

执行写入操作时，Sybase IQ 不会立即将数据写至磁盘中。相反，它将数据写至数据高速缓存中，高速缓存是指 Sybase IQ 在数据库中的页处于使用状态时将其存储到的内存区域。它是 IQ 获得高性能的基本要素。

最后，IQ 必须将脏页（即已经更新的页）写至磁盘中。每次提交事务时，Sybase IQ 都会将脏页写至磁盘中。此方法对于 IQ 用户很有好处，因为它意味着，IQ 不必在事

务日志中记录数据插入操作。因为不记录 IQ 通常会执行的非常大的插入操作，所以用户大大节省了磁盘和性能成本。

事务子划分

您可以使用保存点标识事务中的重要状态，并有选择地返回到相应状态，或致使执行其它操作。

另请参见

- 事务内的保存点（第 353 页）

事务活动快照

sp_iqtransaction 存储过程显示事务活动的快照，如所创建的和正在使用的主空间和临时空间、打开的游标，以及保存点。

它为 IQ 事务管理器中的每个事务控制块返回一行。

并发概述

数据库服务器中的特殊机制允许 Sybase IQ 事务在不互相干扰的情况下并发执行。

Sybase IQ 并发的运行方式

当执行包含在一个事务中的 SQL 语句时，数据库服务器可以执行其它事务中的部分或全部语句。

Sybase IQ 的并发方法是为数据仓库而设计的。通常，在数据仓库环境中，许多用户需要从数据库进行读取，但只有 DBA 需要对数据库进行更新。但是，经常需要能够在其他用户继续请求和接收查询结果时执行这些更新。

Sybase IQ 允许多名用户同时与一个数据库建立连接。它还可以并发处理来自所连接的多个用户或应用程序的事务。

Sybase IQ 可确保在一个事务中执行所有数据库操作，并且这些操作不会相互干扰。它通过设置表级别的访问限制以及使用一种名为“快照版本控制”的技术来实现这一点。在给定表中，IQ 允许并发处理多个读取事务，但只允许处理一个写入事务。此方法可确保数据库的内部一致性。

另请参见

- 版本控制概述（第 340 页）

备份的并发执行

备份可与读取和写入操作并发执行。

备份是一个 DML 操作。备份操作可在备份命令开始执行起执行（检查点）。但是，恢复操作要向数据库中写入数据，因此需要独占访问权。

并发执行的优点

Sybase IQ 的并发方法为查询用户提供直接访问信息的权限，并使您可以确保他们所收到的信息是安全和准确的。

数据仓库是由许多人共享的常见信息存储库。这些人可能需要经常访问这些信息。为了避免妨碍他们的工作，数据库服务器必须能够同时处理许多事务。

而且，许多站点还要求经常更新数据库。在高度可用的站点中，DBA 无法将插入和删除操作推迟到可以执行独占访问时。同样，能够在不中断其他用户的活动的情况下定期备份数据库是非常重要的。

版本控制概述

Sybase IQ 使用快照版本控制来允许事务并发运行。

您可以将快照版本控制想像为您对使用照相机拍摄的快照进行版本控制。当您为对象或场景拍摄快照时，您会获得它在给定时刻的显示图像。同样，当 IQ 为数据库中的对象拍摄快照时，它会保留该对象在给定时刻的图像。

但是，与照相机不同的是，IQ 不需要在图像每次发生变化时为整个对象创建副本。相反，它仅复制发生变化的图像部分 - 数据库页。对于未发生变化的数据库页，将在数据库中的所有活动版本间共享。

当在连接、提交或回退后执行第一个命令时，IQ 会拍摄快照。用户可以通过执行显式 **BEGIN TRANSACTION** 命令来强制提前拍摄快照。通过事务，读取对象的用户可以查看未发生变化的图像或快照版本。

另请参见

- Sybase IQ 并发的运行方式（第 339 页）

表级版本控制

使用表级版本控制，Sybase IQ 可以在执行了写入操作且查询结果所集中的级别控制对数据的访问。

在 Sybase IQ 中的用户可见级别，版本控制的单元是表。

表级版本控制之所以对 Sybase IQ 有意义，原因如下：

- IQ 数据结构在表级别聚合列的数据。
- 大多数 IQ 插入和删除操作都会在表范围内写入数据。

但是，在内部，将在页级别对数据进行版本控制。此方法可以帮助节约系统资源。

给定 IQ 表可包含数以百万计的数据页。更新该表时，可能只会向这些页的一小部分中写入数据。需要大量磁盘空间来为完整表的每个版本维护完整的副本。Sybase IQ 允许表版本共享未更新的页来节省磁盘空间。

Sybase IQ 的表级版本控制针对其 Multiplex 数据库进行了扩展。当创建新版本表的事务在写入服务器上提交时，所有辅助服务器将立即可以使用描述新版本的控制信息。

在辅助服务器上开始的新事务自动查看表的这些新版本，就像写入服务器上的新事务执行的操作一样。

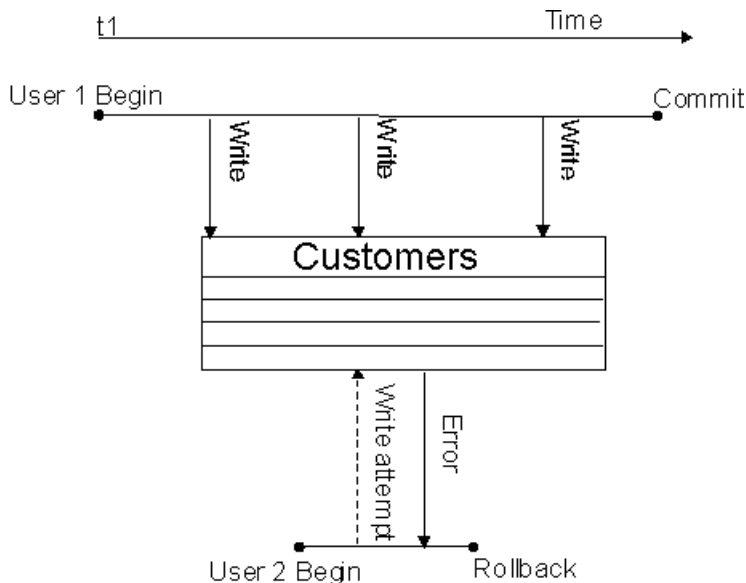
表级别允许一名写入者和多名读取者

在给定表中，IQ 只允许一名用户具有写入访问权限（以执行插入和删除操作），但允许多名读取者并发送出查询。

假设此图中所示的情形。首先，用户 1 开始事务，并开始将数据插入到 `customer` 表中。只要用户 1 的事务保持打开状态，任何其他用户都无法向 `customer` 表中写入数据。在用户 1 的事务提交之前，任何尝试向 `customer` 表中写入数据的事务都会收到错误。

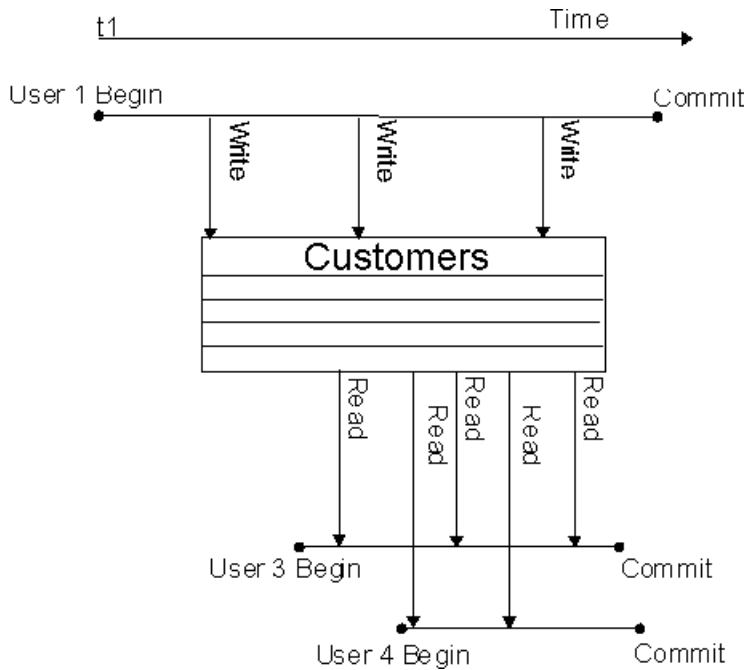
在图中，用户 2 由于尝试在用户 1 的事务提交前进行写入而收到错误。用户 2 的应用程序决定是回退事务，还是尝试向其它表中写入数据。但是，用户 2 无法在同一事务中再次向 `customer` 表中写入数据。

图 2：一次仅允许一名写入者



同时，其他用户在任何时间都可以从 `Customers` 表读取数据。这样，在数据库管理员插入和删除表数据时，查询可以继续。在下图中，用户 3 和用户 4 能够在用户 1 的写入事务保持打开状态时查询 `Customers` 表。

图 3： 一名写入者， 多名读取者



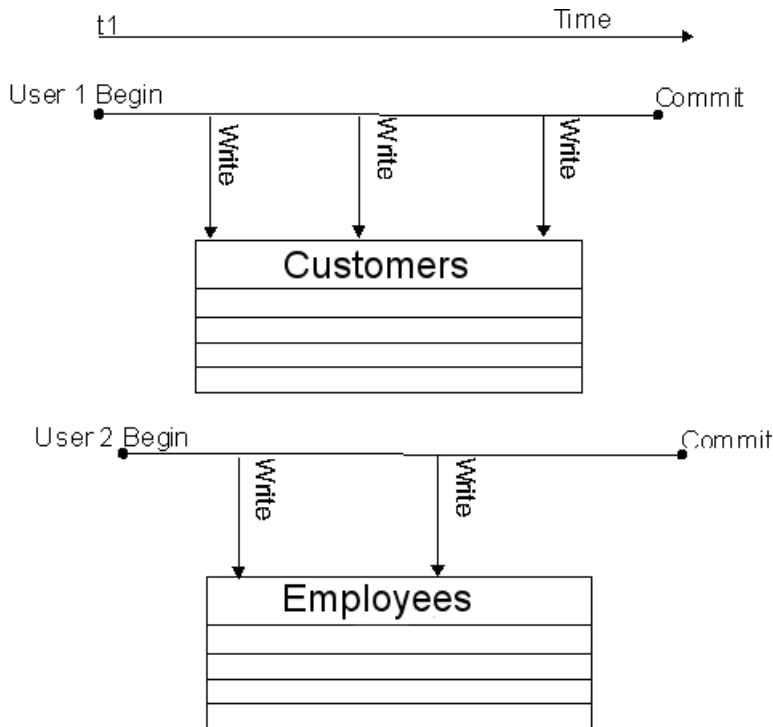
在数据库允许多名写入者和读取者

在 IQ 数据库中，多名只读用户和读写用户可以并发操作，但前提为该多名写入者是将数据插入到不同的表中（或从不同的表中删除数据）。

例如，当用户 1 的事务在 Customers 表中执行插入和删除操作时，用户 2 可以开始一个将数据装载到 Employees 表中的事务，如图中所示。同时，其他用户可以执行向这两个表或数据库中的任何其它表发出查询的事务。

一般而言，只读用户连接到任何辅助服务器，而读写用户连接到写入服务器。读写用户也可以连接到查询服务器，但只能修改全局表/临时表和 SQL Anywhere 基表中的本地数据。

图 4：对不同表的并发插入



对单个表执行的数据定义操作将禁止所有其他读取者和写入者对该表执行操作。

另请参见

- DDL 操作锁（第 348 页）

事务使用提交的数据

每个事务都使用自事务开始以来数据库的最新提交版本。

事务提交前，它将一直使用该版本。

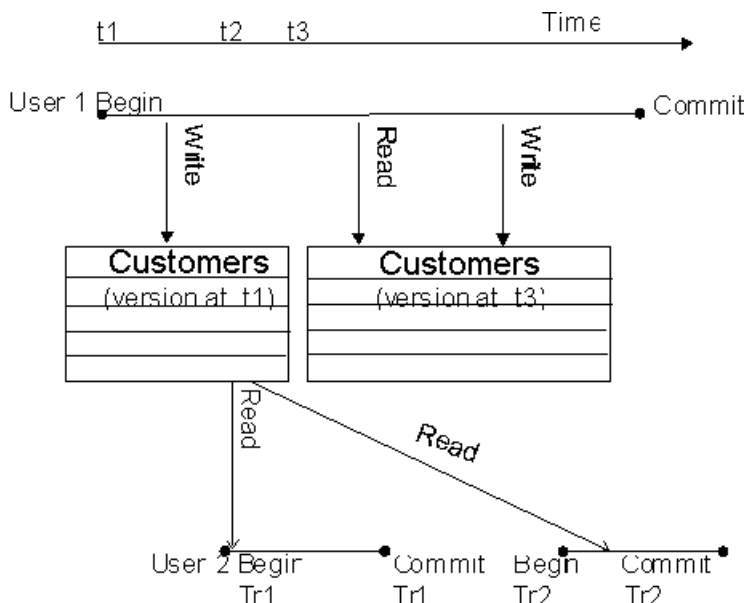
当写入事务提交时的提交数据结果。

事务的开始时间称为其“开始时间戳”。开始时间戳可以是事务首次读取前的任意时间。事务进行的任何插入和删除操作都会反映在快照中。因此，对于执行事务的用户，只要事务向表中写入数据，然后再次读取该数据，快照中的图像就会改变。对于所有其他用户，在事务提交前，图像将保持不变。

换言之，每个事务都以处于可靠状态的数据快照开始。您在发出查询时看到的数据快照不会改变，即使其他用户更新您所读取的表也是如此。例如，在下图中，当用户 1 的写入事务开始时，它使用最新提交的 `customer` 表版本。用户 2 的事务在用户 1 开始写入后但在用户 1 提交前开始。因此，用户 2 的第一个事务 (Tr1) 看不到用户 1

的任何更新。用户 2 的第二个事务在用户 1 提交后开始，因此它会看到用户 1 的所有更改。

图 5：事务使用提交的数据



写入者看到的数据只会根据他/她所做的更改而更改；在写入者的事务提交前，任何其它事务都无法改变写入者看到的内容。例如，在该图中，用户 1 插入一些数据，接着执行查询，然后删除一些数据。这些查询结果反映用户 1 刚刚进行的插入操作。

在用户 1 的事务开始后但在该事务提交前开始的其它事务会看到，自用户 1 的事务开始以来的数据版本。它们看不到最新的更改，原因是：这些更改尚未提交。只要用户 1 的事务提交，新事务就会看到用户 1 的更改。

另请参见

- 读取事务的提交操作时间影响版本（第 344 页）

读取事务的提交操作时间影响版本

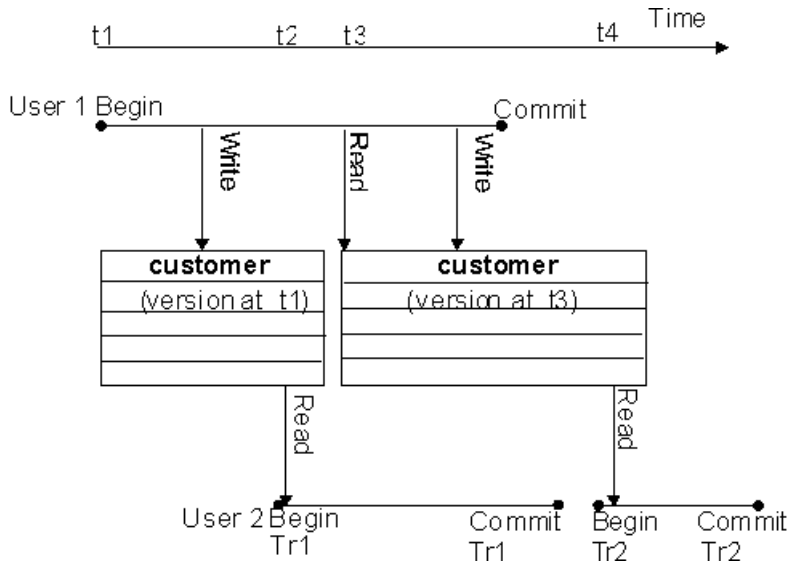
虽然读取事务无法影响现有写入事务看到的内容，但提交读取事务确实会影响其它事务。

- 如果用户的读取事务在并发写入事务执行之前提交，并且该用户开始一个新的读取事务，则版本保持不变。
- 如果读取事务在并发写入操作执行之后提交，则任何新事务（无论是只读还是读写）都会使用新版本。

事务使用提交的数据图是第一种情况的示例。用户 2 的两个事务使用的版本与用户 1 的事务开始于的版本相同，因为这是数据的最新提交版本。

下图显示了在第二种情况下发生的情形。此时，用户 2 的第一个读取事务 (Tr1) 在用户 1 的写入事务后提交。当用户 2 的第二个事务 (Tr2) 开始时，它使用反映用户 1 的已提交数据的新版本。

图 6：读取事务提交的影响



另请参见

- 事务使用提交的数据（第 343 页）

持有游标跨越事务

“事务始终使用最新提交版本”这一规则的唯一例外是在使用持有游标的事务中。

因为持有游标可以跨越事务，所以对它们的处理方式有所不同。

另请参见

- 事务中的游标（第 358 页）

Sybase IQ 如何跟踪版本

Sybase IQ 为元数据中存在的且寿命长于一个命令的持续时间的每个数据库对象分配一个版本标识符。

IQ 使用这些版本标识符确保对任何数据库对象的写入都始终基于该对象的最新版本。它将数据库对象的每个活动版本都保留在磁盘上。

当活动事务不再需要某个较旧的版本时，Sybase IQ 会将该版本从高速缓存中删除。除非使用某个版本的事务执行下列操作之一，否则将一直需要该版本：

- 提交
- 回退
- 发出用于释放该版本的 **RELEASE SAVEPOINT** 命令

此外，对于非 **Multiplex** 数据库，**Sybase IQ** 会确定何时其它事务一开始可以使用某个表的特定版本并释放空间，就等待最旧的活动事务提交或回退。如果您执行大量小型插入、删除和更新操作，则您最有可能从此功能受益。

另请参见

- 事务内的保存点（第 353 页）

临时表的版本控制

在数据库中创建的临时表称为 *全局临时表*。

所有具有适当权限的用户都可以访问全局临时表。但是，每名用户都具有自己的表实例。只有一名用户可以查看一组给定行。缺省情况下，全局临时表的行将在 **COMMIT** 时删除。您可以通过在创建临时表时指定 **ON COMMIT PRESERVE ROWS** 来覆盖此缺省值。

*本地临时表*是在数据库中声明而不是创建的。只有一名用户可以查看本地临时表中的任何行。当该用户断开连接时，将会删除该表。当您声明本地临时表时，**Sybase IQ** 会发出一个保存点而不是自动提交事务，就像它对任何其它类型的表的数据定义操作一样。在创建索引前，一定要提交本地临时表中的数据。如果您尝试使用未提交的数据创建索引，则可能会收到以下错误消息：“必须提交本地临时表 <表名> 才能够创建索引。”

为了进行版本控制，**Sybase IQ** 不区分基表（主数据库表）和全局临时表。因为只有一名用户可以访问任何临时表中的数据，所以永远不会有多个针对临时表的写入事务处于打开状态。

另请参见

- 数据存储（第 149 页）
- 数据库空间管理示例（第 164 页）
- 磁盘空间（第 15 页）
- DDL 操作锁（第 348 页）
- 在 **Interactive SQL** 中与数据库断开所有连接（第 147 页）

版本控制可防止不一致

如果没有版本控制，并发读取和写入操作可能会导致数据库中出现不一致的情况。

Sybase IQ 提供的表级版本控制通过 *序列化* 事务以及将表设为版本级别来防止不一致的情况。

Sybase IQ 允许多名写入者按顺序修改一个表（即，一名写入者接一名写入者地进行修改，永远不能同时有多名写入者进行修改），同时，多名读取者继续对表的原始副本进行读取。使用此方法，IQ 可完全确保防止不一致情况的发生。

虽然任何事务处理系统都设计为确保数据库保持一致，但 Sybase IQ 方法意味着用户不必担心能否将查询和更新置于适当的事务中。IQ 自动开始和结束事务，并可确保读取和写入操作不会彼此干扰。

锁定的工作方式

所有 Sybase IQ 锁都会根据用户请求的操作类型自动出现。

您不需要显式请求锁。可以访问表的事务被称为持有锁。

当在 Sybase IQ 中锁定某个表时，任何其它事务都无法对该表进行写入访问，但任何事务都可以对该表进行读取访问。数据定义操作是此通用读取访问的一个例外；有关详细信息，请参见下面的讨论。任何其它写入事务在尝试访问具有写入锁的表时都会收到错误。

锁可以通过禁止其它事务并发访问来保持数据库中信息的可靠性。数据库服务器将保留事务获取的所有锁，直到该事务由于提交或回退而完成。

可以使用 **LOCK TABLE** 语句保留新事务中一组表的 **WRITE** 锁。**LOCK TABLE** 将提交当前事务，并允许事务在锁可用之前进行排队。有关语法，请参见《参考：语句和选项》>“SQL 语句”>“LOCK TABLE 语句”。

另请参见

- 不能写入已锁定的表（第 465 页）
- Interactive SQL（第 124 页）
- 管理对表的写入锁争用（第 465 页）
- 创建表的准则（第 171 页）
- IQ PAGE SIZE 参数准则（第 143 页）
- 不同的值数目（第 277 页）
- 有效数据的规则和检查（第 317 页）

DML 操作锁

数据操作语言 (DML) 操作包括插入、删除、更新和查询。对于所有这种操作，Sybase IQ 允许一名写入者和多名读取者处理任意给定表。

此规则具有以下含义：

- 读取事务不会阻塞写入事务。
- 写入事务不会阻塞读取事务。

- 一名更新用户和多名只读用户可以并发访问某个表。
- 一次只有一名用户可以更新给定表中的数据。

要在写入模式下打开某个表的第一个事务会获取对该表的访问权。尝试在写入模式下打开该表的第二个事务会收到错误。在当前事务中，任何其它向该表中写入数据的尝试都会失败。事务可以继续，但只能执行读取操作或对其它表执行写入操作。

Sybase IQ 支持 **SHARE**、**WRITE** 和 **EXCLUSIVE** 锁排队，允许您在指定时间内锁定表。一次可以对多个表执行 **WRITE** 锁定。

为避免因后续 **DML** 语句导致的将来版本错误，请对计划修改的一个表或一组表保持 **WRITE** 锁定。请参见《参考：语句和选项》中的 **LOCK TABLE** 语句。

如果发生死锁，被阻塞的最后一个 **LOCK TABLE** 语句常常会回退，系统会将有关所发生死锁的形式的错误返回给该事务。

在某些情况下，必须发出 **COMMIT** 或 **ROLLBACK** 语句。如果 **SYNCHRONIZE JOIN INDEX** 因“表 *x* 没有数据可供连接其它表”而失败，参与连接索引和连接虚拟表的所有数据库表将保持锁定在 **WRITE** 模式下，直到显式断开连接或者发出 **COMMIT** 或 **ROLLBACK** 语句。当 **DML** 语句因完整性约束等原因而失败时，还需要显式执行 **COMMIT** 或 **ROLLBACK** 才能释放锁。

如果 **DML** 语句因参照表上的参照完整性违规或其它表上的锁不可用而失败，Sybase IQ 将返回“SQL Anywhere 错误 -210”。

DDL 操作锁

数据定义语言 (DDL) 操作包括 **CREATE**、**DROP** 和 **ALTER**。

对给定表或索引执行的 **DDL** 操作将会禁止所有其他读取者和写入者处理正在更改的任何表。此方法对于确保查询结果的准确性非常重要。例如，它可确保当您从数据库的某个列中选择数据时，该表列不会从数据库中消失。

CREATE、**DROP** 和 **ALTER** 命令具有下列特殊属性：

- 在任何其它事务正在使用这些命令要修改的表或索引时，这些命令无法开始。例如，如果用户针对某个表发出 **SELECT**，则该表将会锁定，无法进行修改，直到该用户注销、针对另一个表发出 **SELECT** 或发出 **ROLLBACK**。
- 这些命令在完成时自动执行 **COMMIT**。
- 尝试使用正在修改的表的现有事务会收到错误。换言之，如果您要访问某个表，而 **DDL** 命令正在更改该表，则您的命令将会失败。
- 在任何给定时间，命令 **CREATE DBSPACE**、**DROP DBSPACE** 和 **CHECKPOINT** 中只有其中之一可以执行数据库。

另请参见

- 数据存储 (第 149 页)
- 数据库空间管理示例 (第 164 页)
- 磁盘空间 (第 15 页)

- 临时表的版本控制（第 346 页）
- 在 **Interactive SQL** 中与数据库断开所有连接（第 147 页）
- 在数据库允许多名写入者和读取者（第 342 页）

DDL 锁定错误

如果在另一 **DDL** 命令正在执行过程中发出 **DDL** 命令，可能出现错误。

如果同时尝试执行多个 **DDL** 命令，则用户可能会收到以下错误消息：

```
Cannot perform DDL command now on table <tablename> as a DDL command is already in progress on that table.
```

如果正在执行 **CREATE DBSPACE** 或 **DROP DBSPACE** 命令，并且用户显式发出 **CHECKPOINT** 命令，则检查点会失败，并会出现以下消息：

```
Run time SQL Error
```

如果正在执行 **CHECKPOINT** 命令，发出 **CREATE DBSPACE** 或 **DROP DBSPACE** 命令的用户会收到以下消息：

```
Cannot perform requested command as there is a CHECKPOINT command in progress.
```

在删除期间发出 **CREATE DBSPACE** 的用户会收到以下消息：

```
Cannot perform requested command as there is a DROP DBSPACE command in progress.
```

在创建期间发出 **DROP DBSPACE** 的用户会收到以下消息：

```
Cannot perform requested command as there is a CREATE DBSPACE command in progress.
```

当一个事务针对给定表或索引发出 **DDL** 命令时，在 **DDL** 事务提交前开始的以及尝试访问该表的任何其它事务都会收到错误。

当出现此错误时，在当前事务中对该表进行的任何其它读取或写入尝试都会失败。

如果某个事务修改属于连接索引的表的定义，则它会锁定每个含该索引中连接的任意列的表。不管原始写入事务中的特定列是否连接，都会出现此结果。

对于索引创建命令，这些规则有所例外。**CREATE INDEX** 和 **CREATE JOIN INDEX** 可在受索引创建影响的表上与 **SELECT** 并发执行。在创建索引的事务提交之前，Sybase IQ 将禁止使用新索引或连接索引。

虽然命令 **GRANT**、**REVOKE** 和 **SET OPTION** 也被视为 **DDL** 操作，但它们不会导致并发冲突，因此不受限制。**GRANT** 和 **REVOKE** 始终会导致自动提交；除非将 **SET OPTION** 指定为 **TEMPORARY**，否则它会导致自动提交。当前连接到数据库的任何用户都不能使用 **GRANT** 和 **REVOKE**。**SET OPTION** 会影响所有发送到数据库服务器的后续 **SQL** 语句，但在重新启动数据库服务器后才会生效的某些特定选项例外。有关设置选项的详细信息，请参见《参考：语句和选项》。

主键和锁定

因为只有一名用户可以更新表，所以生成主键不会导致并发冲突。

管理锁的工具

虽然锁定和解锁自动进行，Sybase IQ 还可通过许多存储过程、IQ 监视器以及数据库和服务器选项帮助您管理锁。

显示活动锁

如果另一个事务对某个表进行了锁定，则尝试向该表写入数据将失败。

要标识锁定表的用户：

1. 运行 **sp_iqtransaction** 过程。
2. 在 **sp_iqtransaction** 的输出中查找事务标识符。
3. 在同一个输出行中查找用户的名称。

示例

例如，当另一事务持有锁时，系统将显示以下消息：

```
Cannot open the requested object for write in the current transaction (TxnID1). Another user has write access in transaction TxnID2.
```

在 **sp_iqtransaction** 的输出中查找 TxnID2，并在同一个输出行中查找用户的名称。

sp_iqlocks 中锁的相关信息

sp_iqlocks 过程显示当前在数据库中持有的锁的相关信息。

对于当前数据库的 Catalog 存储和 IQ 存储中的每个锁，**sp_iqlocks** 将说明：

- 持有锁的连接和用户 ID
- 控制锁的表。
- 锁的类型以及用于标识锁的名称

sp_iqtransaction 过程可提供有关事务的更为详细的信息。请参见《参考：构件块、表和过程》

调查锁争用的工具

有些装载或查询性能问题可能源于锁争用。

要弄清楚锁争用是否会影响系统的性能，请使用 IQ 或操作系统提供的功能：

- 使用 **-contention** 选项运行 IQ 监视器。
- 在 UNIX 平台上，运行 **sar** 或 **vmstat** 实用程序。

- 在 Windows 平台上，在“任务管理器”中检查 CPU 使用率。

如果您的内核系统时间大于 10%，则可能发生了锁争用问题。

Sybase IQ 通过将 IQ main 高速缓存和临时高速缓存分区来限制锁争用。分区的缺省级别基于 IQ 服务器上的 CPU 数量，在大多数情况下应该是足够用的。如果您怀疑存在锁争用的情况，则通过设置 `-iqpartition` 服务器启动选项或 `Cache_Partitions` 数据库设置选项直接控制分区级别可能会很有用。

注意： 如果内核系统时间高于正常值也表示您的内核调整不当。如果情况是这样，可能需要调整内核参数；更改 IQ 设置不会解决内核调整不当的问题。

另请参见

- 数据装载（第 14 页）
- 调优数据的 bulk load 操作（第 276 页）
- 使用 LOAD TABLE 语句执行 bulk load 操作（第 238 页）
- 装载时间环境调整（第 278 页）

隔离级别

事务处理的一个重要方面是数据库服务器隔离操作的能力。ANSI 标准定义了四个隔离级别。级别越高，为事务提供的与其它事务的隔离程度越高，因此更能确保数据库的内部一致性。

隔离级别控制一个事务中的操作和数据对其它并发事务中的操作的可见程度。IQ 快照版本控制支持最高的隔离级别。在此级别，可将所有调度序列化。

快照版本控制通过遵循下列规则在并发事务之间保持这种较高的隔离级别：

- 事务管理在每个事务开始时维护已提交数据的快照。
- 只要维护事务使用的快照版本，事务就始终可以读取数据。
- 事务的写入操作会反映在它看到的快照中。
- 一旦某个事务开始，它就无法看到其它事务进行的更新了。

Sybase IQ 为防止多种类型的不一致而提供的隔离级别。下面列出了最常遇到的不一致现象：

- **脏读** - 事务 A 修改了某个对象，但未提交或回退更改。事务 B 读取所修改的对象。然后，事务 A 通过执行 **COMMIT** 来进一步更改该对象。在这种情况下，事务 B 看到的对象处于从未被提交的状态。
- **不可重复的读取** - 事务 A 读取某个对象。然后，事务 B 修改或删除该对象并执行 **COMMIT**。如果事务 A 再次尝试读取同一对象，则该对象将被更改或删除。
- **幻像数据元素** - 事务 A 读取一组满足某种条件的数据。然后，事务 B 先后执行 **INSERT** 和 **COMMIT**。新提交的数据现在已满足以前不满足的条件。事务 A 之后重新执行最初的读取操作，并获得一组不同的数据。

- **更新丢失** – 在使用游标的应用程序，事务 A 写入一组数据的更改。然后，事务 B 保存基于以前的数据的更新。事务 A 的更改将完全丢失。

Sybase IQ 通过以下方法来防止所有这些不一致现象：确保在任何给定时间都只有一名用户可以修改表，在更改完成前使更改保持对其他用户不可见，以及在任何时间都为正在使用的数据对象维护带时间戳的快照。

虽然通过 IQ 可以使用 **SET OPTION ISOLATION_LEVEL** 将隔离级别设置为 0、1、2 或 3（相当于 ANSI 级别 1、2、3 或 4），但并不需要做。所有用户都在隔离级别 4 执行，即使您设置其它级别也是如此。设置更低的隔离级别不会使性能有所提高。

有关防止并发事务访问或修改表的详细信息，请参见《参考：语句和选项》中的 **LOCK TABLE** 语句。

检查点、保存点和事务回退

除允许并发以外，事务处理在数据恢复中也扮演着重要角色。数据库恢复始终会恢复每个已提交的事务。在数据库出现故障时尚未提交的事务不会恢复。

Sybase IQ 依赖于三个与事务相关的命令，这些命令可帮助您在系统或介质出现故障时恢复一组稳定数据。这些命令可设置检查点，设置并释放保存点，并回退事务。

Checkpoints

当 Sybase IQ 向磁盘写入它在内部跟踪的某些特定信息时，检查点会标记事务中的重要点。如果需要恢复数据库，IQ 会使用此信息。

Sybase IQ 使用检查点的方式与 OLTP 数据库（如 SQL Anywhere）不同。OLTP 数据库往往拥有较短的事务，这些事务只会影响少量行。对这些事务而言，将完整的页面写至磁盘会产生非常高的成本。相反，OLTP 数据库通常会在检查点处向磁盘中写入数据，并且只写入已更改的数据行。

Sybase IQ 是 OLAP 数据库。一个 OLAP 事务可以更改数以千计甚至数以百万计的数据行。因此，Sybase IQ 不会一直等到检查点出现才执行物理写入。在每个事务提交后，它都会将已更新的数据页写至磁盘中。对于 OLAP 数据库，将完整的数据页写至磁盘中比在任意检查点处写入少量数据更加有效。

恢复中的检查点辅助工具

为了从系统或介质故障中恢复，Sybase IQ 必须能够将数据库恢复到内部一致点。

IQ 使用检查点生成恢复数据库所需的参照点和其它信息。IQ 在每个检查点处写至磁盘中的信息对于恢复进程而言都是必需的。

检查点何时出现

大多数 Sybase IQ 检查点都会自动出现。您也可以设置显式检查点，但并不需要做。

在下列时间将会出现检查点：

- 当事务发出 **CHECKPOINT** 命令。
- 当超过 **CHECKPOINT_TIME** 时。
- 在备份进程开始和结束时。
- 当数据库服务器关闭时。

CHECKPOINT_TIME 是两个检查点之间间隔的最长时间。它缺省设置为 60 分钟。要调整检查点时间间隔，请使用 **SET OPTION** 语句。请参见《参考：语句和选项》。调整检查点时间或发出显式检查点可能是不必要的。与在 **OLTP** 数据库产品中相比，在 **Sybase IQ** 中，控制检查点并不那么重要，因为 **IQ** 会在每个事务提交后写入实际的数据页。

另请参见

- 事务信息如何帮助进行恢复（第 356 页）

事务内的保存点

Sybase IQ 支持事务中的保存点。

SAVEPOINT 语句定义事务执行过程中的中间点。因为一个 **IQ** 事务可以写入数以百万计的数据行，所以可能需要限制所提交（进而写至磁盘中）的数据量，使其小于整个事务的大小。通过设置保存点，可以细分事务。

您可以使用 **ROLLBACK TO SAVEPOINT** 语句撤消保存点后的所有更改。

另请参见

- 事务子划分（第 339 页）
- **Sybase IQ** 如何跟踪版本（第 345 页）

释放保存点

执行了 **RELEASE SAVEPOINT** 语句或事务结束后，则不能再使用该保存点。

释放保存点会释放已经使用的版本页，一直到相应保存点。请记住，对数据的版本控制是在内部的页面级别进行的。**Sybase IQ** 仅维护已更新页的单独副本，对于其余页，将与早期版本共享。通过释放保存点，可以释放与它们相关联的页面，从而更好地利用磁盘空间。

释放保存点 *n* 会释放该保存点后的所有资源，使您能够回退到任何中间保存点。

任何锁都不会由 **RELEASE SAVEPOINT** 命令释放。

回退到保存点

您可以通过发出 **ROLLBACK TO SAVEPOINT** 语句撤消保存点后的所有更改。

此命令会回退到您指定的保存点，或回退到最新的 **SAVEPOINT**（如果您没有指定命名的保存点）。回退到保存点 *n* 会撤消在大于或等于 *n* 的所有保存点后执行的所有操作。

通常情况下，仅在事务结束时释放锁。但是，在某些特定情况下，**ROLLBACK TO SAVEPOINT** 确实会释放锁，如以下情况所示：

假设您的事务中有一系列保存点，并且您执行的是写入操作。然后，您将事务回退到以前的保存点。回退会撤消该保存点后的所有操作，包括写入操作以及该操作在要回退到的保存点后获取的所有锁。

Sybase IQ 支持对可更新的游标执行保存点操作。

另请参见

- 事务中的游标（第 358 页）
- 回退的影响（第 355 页）

自动保存点和用户定义的保存点

IQ 在每个 DML 命令之前和之后设置隐式保存点。

当命令完成时，与这些保存点相关联的数据页版本将会释放。如果要在单个 DML 命令结束后继续保留数据页版本，则需要设置自己的命名保存点。

命名和嵌套的保存点

命名的嵌套保存点在事务内可提供多个活动保存点。

SAVEPOINT 和 **RELEASE SAVEPOINT** 之间的更改可以通过回退到前一个保存点或回退事务本身来取消。在提交事务之前，事务内的更改还没有永久地应用到数据库中。事务结束后，所有保存点都将释放。

保存点会导致 Sybase IQ 更新它所维护的有关可用磁盘空间位置的信息。此信息在事务回退期间使用。

使用保存点的过程中没有额外开销，但未释放的保存点可能会占用额外的磁盘空间（因为要将较旧的中间版本保持活动状态）。

事务回退

回退事务时，将撤消该事务中的所有操作。

回退数据库意味着将数据库恢复到以前的状态。

什么导致回退

回退可能会由于显式用户请求发生，也可能会自动发生。

使用 **ROLLBACK** 语句可撤消自上次执行 **COMMIT** 或 **ROLLBACK** 以来对数据库所做的所有更改。

使用 **ROLLBACK TO SAVEPOINT** 语句可撤消在您命名的 **SAVEPOINT**（或最后一个 **SAVEPOINT**）后对数据库所做的所有更改。

如果用户在执行事务，然后在未提交的情况下注销或断开连接，则 Sybase IQ 会自动回退数据库。回退的目标是最新的提交或回退。

回退的影响

回退可使 **main** 存储和临时存储恢复到它们以前的状态。

回退还会释放锁：

- 事务回退将释放事务持有的所有锁。
- 回退到保存点将释放在该保存点后获取的所有锁。

回退打开的游标将删除所有游标信息，并关闭持有游标和非持有游标：

- 事务回退将关闭所有游标。不管游标是在所回退的事务中打开的还是以前的事务中打开的，都同样如此。
- 回退到保存点将关闭在该保存点后打开的所有游标。

另请参见

- 事务中的游标（第 358 页）
- 回退到保存点（第 353 页）
- 游标和版本控制（第 358 页）

系统恢复

如果系统出现故障或者断电，或者如果在数据库服务器停止后将其重新启动，则 Sybase IQ 会自动尝试执行恢复。

在 Sybase IQ 数据库恢复期间，会回退任何未提交的事务，并且用于旧版本的任何磁盘空间都会返回到可用空间池中。此时，数据库只包含每个永久表的最新提交版本。

在从系统故障恢复期间，Sybase IQ 会重新打开出现故障时处于活动状态的所有连接。如果 **-gm** 参数（用于设置用户连接数目）在出现故障时处于有效状态，则需要至少使用发生故障时实际在使用的数量的连接，才能重新启动 IQ 服务器。临时表内容不可恢复。

如果发生故障，请尝试重新启动数据库服务器和数据库。您需要使用服务器日志和 IQ 消息日志中的信息进行恢复。

Sybase 建议您在系统出现故障后，允许用户连接前运行存储过程 **sp_iqcheckdb**。此过程会检查数据库中的每个锁，并生成可用于检查数据库的一致性和完整性的统计信息。

另请参见

- 数据保护（第 375 页）
- 关闭操作系统会话（第 37 页）
- 运行 **sp_iqcheckdb** 的资源问题（第 426 页）
- 系统恢复与数据库修复（第 419 页）
- 验证数据库（第 394 页）

事务信息如何帮助进行恢复

Sybase IQ 的恢复机制是为数据仓库而设计的。在这种环境中，执行的事务通常很少，但是每个事务都相当耗时。

为了最好地适应此模型，Sybase IQ 通过在实际数据库页的副本上进行更新来执行数据库更新，然后在写入事务提交时将数据写至磁盘中。它还记录以下信息：

- 每个事务的已更改数据的位置和数量。它将此信息存储在 *事务日志* 中。
- 磁盘上的所有版本页和可用空间的位置。当不再需要这些版本时，它使用此信息释放空间。当写入事务提交或回退时，在写入事务持续期间创建的所有版本都会过时。各个版本可在保存点处释放。
- 有关在事务执行期间出现的检查点的其它信息。

需要恢复数据库时，Sybase IQ 不会重复所有已发生的冗长事务，而是会利用事务日志中的信息和检查点信息进行快速恢复。它使用有关版本和可用空间的信息回退事务，并释放已过时的版本占用的磁盘空间。

事务日志需要使用的空间很少：对于每个已提交的事务，大约仅需要 128 字节。检查点和磁盘空间可用性信息的空间要求也非常小。但事务日志的大小却持续增长。如果系统中有大量会更改数据的事务，则在一段时间后，日志可能增长到非常大，从而需要定期截断日志。

检查点信息将在其下一个检查点处被删除。当释放或回退特定保存点时，将会删除与该保存点相关的信息。

另请参见

- 检查点何时出现（第 352 页）
- 运行备份（第 382 页）

快照版本控制的性能影响

快照版本控制对性能的影响很小。与由于能够在其他用户读取数据库中数据的同时更新数据库而获的灵活性相比，负面影响微乎其微。但是，您应该注意一些资源问题：

- 如果多名用户同时使用同一数据库页的不同版本，则缓冲区使用量可能会略有增加。
- 版本管理需要一些开销，但是对性能的影响很小。另请参见有关磁盘空间的项。
- 线程控件（用于确定用户获取多少个处理资源）和清理程序控件（可使用少量线程将磁盘中的脏数据页清理出去）对性能的影响很小。
- 磁盘空间有时会成为问题。存储重叠版本可能会使用大量磁盘空间，具体取决于同时使用的版本的数量和大小。元数据和数据库页版本将保留下来，直到在 **RELEASE SAVEPOINT** 处或在最后一个可以看到给定版本的事务提交或回退时将它们删除为止。然后，将会回收空间。

由于锁定而导致的延迟非常短暂。各个提交、回退和检查点只能在非常短的时间内阻塞其它读取或写入事务。

请记住，所有这些性能和磁盘使用因子对系统的影响程度都取决于您对 IQ 的并发读取和写入功能的利用程度。具体的磁盘空间要求可能会有很大不同，具体取决于写入事务在提交前占用的时间，在写入事务执行期间发生的读取事务数，这些事务影响的行数，以及您是否允许在中间保存点处释放数据页。

重叠版本和删除

为了删除数据，可能需要通过向 IQ 存储中添加数据库空间来实际增加磁盘空间。

执行删除所需的空间量取决于数据页上数据的分布，而不是取决于所删除的行的尺寸或数量。IQ 需要将每个含任意待删除数据的页面版本保留下来，从删除开始执行起一直保留到事务提交。如果所删除的行恰好分布在许多数据页中，则您需要使用 IQ 存储中的空间来保留所有这些额外的数据页。

例如，假设您需要从每页有 100 行的数据库中删除十行。如果这十行中的每一行都位于不同的数据页上，则 IQ 存储需要拥有用于存储十个版本页（每个页都足能容纳 100 行）的空间。虽然这种分布不太可能存在，但却不能排除其可能性。

删除数据所需的空间因索引类型而异。该空间与您要删除的索引的大小成比例，在最坏的情况下，等于该大小。

如果在删除数据时空间用尽，则 Sybase IQ 会停止删除，并在通知日志中显示下面的消息：

```
Out of disk space
```

添加空间后，删除操作继续进行。当删除事务提交时，该空间可用于执行其它删除或插入操作。如果通常不需要数据库中有大量空间，则可以删除数据库空间以重新获取额外磁盘空间来用于其它用途。在插入可能需要使用新数据库空间的任何数据之前，请务必执行此操作。

在删除期间用尽空间不会影响其他查询用户。

如果空间用尽，但没有足够的磁盘空间来添加其它数据库空间，您必须关闭数据库引擎，然后重新启动，以便数据库可以回退。然后，可以在更小的单独事务中删除行。

注意： DROP TABLE 和 DROP DATABASE 会删除表或数据库以及其中的所有数据，而不会创建任何版本页，因此，使用这些命令时不需要添加空间。

另请参见

- 磁盘空间使用情况（第 194 页）

事务中的游标

使用游标，可以将 **SELECT** 的结果以称为“游标”的数据类型的形式返回。

游标类似于表，但是拥有将一行标识为当前行的附加属性。可以使用各种命令在游标行间导航。例如，**FETCH** 命令从游标中检索一行，并将它标识为当前行。您可以通过重复调用此命令来逐步处理游标中的所有行。

当您编译过程时，或者当您编写使用嵌入式 **SQL** 访问数据库的应用程序时，游标最有用。许多前端查询工具也会使用游标。以交互方式使用 **dbisql** 时，游标不可用。

Sybase IQ 游标是可更新的，从而可以在处理游标时修改数据库中的基础数据。

游标中的行（类似表中的行）不具有与其关联的顺序。**FETCH** 命令逐步处理各个行，但是顺序可能看起来像随机的，甚至可能是不一致的。因此，您需要通过将 **ORDER BY** 短语附加到 **SELECT** 语句中来强制执行某种顺序。

sp_iqcursorinfo 存储过程可显示有关在服务器上当前打开的游标的信息。请参见《参考：构件块、表和过程》。

另请参见

- 持有游标跨越事务（第 345 页）
- 回退的影响（第 355 页）
- 回退到保存点（第 353 页）

游标和版本控制

使用游标时，**Sybase IQ** 需要能够在事务中管理多个版本。

例如，假设您在 *x* 时打开一个名为 `cust_cursor` 的游标，而该游标使用 `customer` 表。然后，您在 *y* 时更新该表。**Sybase IQ** 需要从 *x* 时到使用 `cust_cursor` 完成时保留 `customer` 表的版本。

Sybase IQ 提供的游标支持针对的是游标在 **DSS** 应用程序中的可能使用情况。以下各节讨论了具体的游标特征及其对事务处理的影响。

另请参见

- 回退的影响（第 355 页）

游标敏感性

如果游标的成员（它返回的数据行）从它打开到它关闭这段时间内会发生变化，则认为相应游标是敏感的。非敏感性游标在处于打开状态时，其成员固定不变。

敏感性游标的结果集的成员和值是否更改是不确定的。对值敏感的游标对于它的成员不敏感，而对于结果集的顺序和值敏感。**Sybase IQ** 支持敏感的可更新游标。

游标滚动

Sybase IQ 游标可以是滚动的，也可以是非滚动的。

非滚动游标只允许使用 **FETCH NEXT** 和 **FETCH RELATIVE 0** 这两种命令形式来查找和检索数据。非滚动游标不需要跟踪已提取了哪些行。声明为 **DYNAMIC SCROLL** 的游标与声明为 **SCROLL** 的游标相同。

如果您检索的行数非常多（数百万），则将选项 **FORCE_NO_SCROLL_CURSORS** 设置为 **ON** 可以按照临时存储要求进行保存。但如果前端应用程序经常使用向后滚动游标操作，将此选项设置为 **OFF** 会提高查询响应速度。

如果前端应用程序很少执行向后滚动，请将 **FORCE_NO_SCROLL_CURSORS = 'ON'** 作为永久 **PUBLIC** 选项。它将使用更少内存，并提高查询性能。

持有游标

如果事务以 **COMMIT** 结束，则在打开游标时指定 **HOLD** 选项会使游标在事务结束后依然保持打开状态。

在打开游标的 **ROLLBACK** 后，持有游标不会保持打开状态。

在 Sybase IQ 中，持有游标在提交前一直是可更新的。提交后，会在内部将持有游标标记为 **READ ONLY**，后续定位更新会产生错误。

虽然 **HOLD** 选项不常用于 **DSS** 环境中的较长事务，但在一些情况下，它也十分有用。例如，许多现有应用程序期望使用持有游标，而有些 **ODBC** 驱动程序则缺省使用持有游标。

Sybase IQ 提供持有游标所需的版本管理。

持有游标确实会影响性能。持有游标使用的所有资源（包括内存、磁盘空间和进程线程）会一直保留到游标关闭。

定位操作

在定位操作中，游标的当前位置决定读取或写入操作的开始位置。

Sybase IQ 支持定位提取，定位提取在非常长的查询事务中很有帮助。Sybase IQ 还支持定位更新和删除操作，这些操作适用于较短的插入和删除。通常，对 IQ 数据库的更新可能会涉及大量数据；而重新定位是这种写入操作的一个非常次要的部分。

在游标上，定位更新和删除被视为操作（因此，是作为事务的一部分）而不是单独的语句处理。在游标打开后发生的任何故障都会导致已通过此打开的游标执行的所有操作回退。

如何控制游标的消息记录

缺省情况下，不会在 IQ 消息文件中记录游标操作。

如果需要跟踪游标操作以确定问题的原因，请启用 `LOG_CURSOR_OPERATIONS` 选项以在每次打开或关闭游标时生成消息。通过可更新游标进行的数据更改也会记录在 IQ 消息文件中。

国际语言和字符集

配置 Sybase IQ 安装 来处理国际语言问题。

创建数据库时，会指定数据库使用的归类序列或*归类*。归类是 *字符集*和数据库中字符的 *排序顺序*的组合。

数据库归类用于排序和比较数据库中的所有字符数据类型，包括对象名，如表名和列名。

Sybase IQ 利用 SQL Anywhere 归类算法 的空间效率和速度。

- 数据库选项 `SORT_COLLATION` 允许在 **ORDER BY** 表达式中隐式使用 **SORTKEY** 函数。如果将此选项的值设置为有效的归类名称或归类 ID，则 **ORDER BY** 子句中的任何字符串表达式作为 **SORTKEY** 函数已调用处理。
- **SORTKEY** 函数使用用于 Unicode 的国际化组件 (ICU) 库，而不是 Sybase Unicode Infrastructure Library (Unilib[®])。使用 15.0 之前版本的 Sybase IQ 创建的排序键值不包含使用 15.0 版本和更高版本创建的相同值。应在数据库中重新生成使用 15.0 之前版本的 Sybase IQ 生成的任何排序键值。
- **CREATE DATABASE** 参数 **COLLATION** 支持为数据库指定归类。
数据库归类必须与操作系统使用的归类匹配。在 Sybase Central 创建数据库向导中，默认字符集是 `ISO_BINENG`；如果需要，请对其进行更改，以使其与操作系统使用的字符集匹配。
- **CP874toUTF8** 实用程序将数据由 CP874 字符集转换为 UTF8 归类，后者是 Sybase IQ 针对泰语提供支持的归类。**CP874toUTF8** 实用程序调用 ICU 库来执行数据转换。还可以装载 CP874 字符集中的数据，而不必使用该实用程序将数据转换为 UTF8。请参见《实用程序指南》>“CP874toUTF8 数据库管理实用程序”。

Sybase IQ 不再支持自定义归类。如果要重建具有自定义归类的数据库，请以单步方式重建 以便保留该自定义归类。如果卸载数据库，然后将 `schema` 和数据 装载到所创建的数据库中，则必须使用 提供的归类之一。

有关在使用多个字符集的环境中工作或 使用非英语语言时可能遇到的问题的简介，请参见“SQL Anywhere Server - 数据库管理”>“配置数据库”>“国际语言和字符集”>“SQL Anywhere 的本地化版本”>“何谓 ICU、何时需要它？”。

注意： 此参考指向 SQL Anywhere 文档。

有关 Sybase IQ 15.0 中 对数据库归类的更改以及不建议使用的归类列表的详细信息，请参见《Sybase IQ 15.0 中的新增功能》。使用 `iqunload` 实用程序可以从通过不建议使用的 归类创建的现有 12.7 数据库迁移到 当前版本的 Sybase IQ。有关 `iqunload` 的详细信息，请参见《安装和配置指南》。

另请参见

- 如何启动 `iqdemo` 数据库（第 39 页）

- 服务器名 (第 23 页)
- 启动数据库服务器的方法 (第 17 页)
- 从命令行创建 ODBC 数据源 (第 59 页)
- 从 Interactive SQL 启动服务器 (第 30 页)

缺省归类

如果在创建 Sybase IQ 数据库时采用缺省操作，数据库将具有 ISO_BINENG 归类。

ISO_BINENG 归类可为 IQ 数据库提供最佳性能，但不一定会提供最自然的排序顺序。

这一点与 SQL Anywhere 不同，后者根据在其中创建数据库的操作系统所用的字符集推断新数据库的缺省归类。

如果无法以这种缺省方式设置系统，则需要确定在数据库中使用哪个归类，以及是否使用字符集转换以确保在数据库系统的组件之间转换数据的一致性。

了解软件中的字符集

每个软件都会使用一个字符集，字符集是一组符号，包括字母、数字、空格和其它符号。

有关与国际语言和字符集 (包括单字节和多字节字符集) 有关的软件问题的一般信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “了解字符集”。

注意： 此参考指向 SQL Anywhere 文档。

Windows 中的代码页

很多语言只有极少的字符，用单字节字符集表示足矣。在此类字符集中，每个字符都以单个字节来表示：两位十六进制数。

使用单个字节最多可以表示 256 个字符。任何单字节字符集都无法囊括国际上使用的所有字符，其中包括带重音的字符。这一问题已通过开发一组代码页获得解决，每个代码页说明适合一种或多种国家/地区语言的一个字符集。例如，代码页 869 包含希腊语字符集，而代码页 850 包含一个国际字符集，该字符集适用于表示多种语言的众多字符集。

有关 Windows 中的 ANSI 和 OEM 代码页的信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “了解字符集” > “Windows 中的 ANSI 和 OEM 代码页”。

有关支持的代码页的列表，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “字符集和归类参考信息” > “建议的字符集和归类”。

注意： 这些链接可让您转到 SQL Anywhere 文档。

归类序列如何对字符排序

数据库归类序列包括字母顺序，并将其扩展到包含字符集中的所有字符，其中包括数字和空格字符。

每个排序位置可以与多个字符建立关联。例如，如果您希望将带重音符号的字符视为与不带重音符号的字符相同，这一点将十分有用。

数据库会将具有相同排序位置的两个字符视为在所有方面都相同。因此，如果归类将字符 *a* 和 *e* 指派到相同的排序位置，则具有以下搜索条件的查询：

```
WHERE col1 = 'want'
```

会使 **col1** 中包含条目 **went** 的行满足条件。

在每个排序位置，可以指明字符的大小写形式。对于区分大小写的数据库（Sybase IQ 数据库的缺省设置），不会将小写字符和大写字符视为等同。对于不区分大小写的数据库，则会将字符的小写和大写形式视为等同。

注意： 选择德语系统缺省归类的所有代码都应选择 1252LATIN1，而不是 1252DEU。1252DEU 区分带变音符号和不带变音符号的字符，而 1252LATIN1 则不区分。1252LATIN1 将 Muller 与 Müller 视为等同，但 1252DEU 不将其视为等同。由于 1252DEU 将带有变音符号的字符视为不同字符，因此其字母排序为：ob、öa。

多字节字符集的首字节归类排序

在多字节字符集中，只能为首字节指定字符的排序顺序。

首字节相同的字符会按后面字节的十六进制值排序。

了解区域设置

数据库服务器和 Client Library 都使用区域设置定义识别其语言和字符集环境。

请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “了解区域设置”。

注意： 此参考指向 SQL Anywhere 文档。

了解归类

归类描述了如何对来自特定字符集或编码的字符进行排序和比较。

有关提供的归类的信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “了解归类”。

有关在特定情况下使用哪些归类的建议，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “字符集和归类参考信息”。

有关可以与 SORTKEY 和 COMPARE 函数一起使用的兼容性归类，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “了解归类” > “支持的归类和替代归类”。

有关建议与 Windows 和 Unix 平台一起使用的字符集、代码页、编码以及归类的列表，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “字符集和归类参考信息” > “建议的字符集和归类”。

注意： 这些链接可让您转到 SQL Anywhere 文档。

显示归类

获得当前使用的归类。

在命令提示符下键入以下命令：

```
SELECT * FROM SYS.SYSCOLLATION
```

ANSI 和 OEM 代码页

Sybase IQ 归类是以代码页为基准的，这些代码页分别指定为 ANSI 代码页或 OEM 代码页。

在大多数情况下，推荐使用 ANSI 代码页。如果使用 OEM，需要在用户的客户端计算机上选择一种符合 OEM 代码页的代码页。

在任何情况下，都不应使用单独的转换驱动程序。转换驱动程序会干扰服务器的字符集转换。使用单独的转换驱动程序很可能导致数据损坏。

对于 Interactive SQL 和 Sybase Central，字符集转换由 iAnywhere JDBC 驱动程序处理。

ANSI ISO_1 归类

ISO_1 是为实现与 Adaptive Server Enterprise 缺省 ISO_1 归类的兼容而提供的。

与 Adaptive Server Enterprise 的差异在于：

- ß、小写拉丁字母 s (\xDF) 在 Sybase IQ 和 SQL Anywhere 中排在小写 s 之后，但在 Adaptive Server Enterprise 中排在 ss 之后。
- 在 Sybase IQ 和 SQL Anywhere 中，Æ 和 æ (\xC6 和 \xE6) 对应的连字分别排在 A 和 a 之后 (A、a、æ、Æ)。在 Adaptive Server Enterprise 中，排序顺序为 A、a、Æ、æ。

ANSI 1252LATIN1 归类

此归类包括欧元货币符号和若干其它字符（带有 caron 的 Z 和带有 caron 的 z）。

对于单字节 Windows 操作系统，这是大多数情况下建议使用的归类。对于使用英语或西欧语言的 Windows 用户，建议使用此归类。

ANSI ISO1LATIN1 归类

此归类与 ISO_1 相同，但仅限于范围 A0-BF 内的值的排序。

为与 Adaptive Server Enterprise 兼容，ISO_1 归类在 0xA0-0xBF 范围内没有对应字符。但是，作为其基准的 ISO 拉丁语 1 字符集在这些位置上却有对应的字符。ISO1LATIN1 归类反映了这些位置的对应字符。

如果您不关心与 Adaptive Server Enterprise 的兼容性，则一般建议使用 ISO1LATIN1，而不是 ISO_1。

对于使用英语或西欧语言的 UNIX 用户，建议使用 ISO1LATIN1，前提是愿意牺牲缺省归类 ISO_BINENG 的某些最佳性能。

ANSI ISO9LATIN1 归类

此归类与 ISO1LATIN1 相同，但是它包括欧元货币符号和 1252 LATIN1 归类中包括的其它新字符。

如果您的计算机使用 ISO 拉丁语 9 字符集，而且您愿意牺牲 ISO_BINENG 的某些最佳性能，则请使用此归类。

多字节归类

Sybase IQ 提供的归类采用若干多字节字符集。

Sybase IQ 支持可变宽度字符集。在这些字符集中，有些字符由一个字节表示，有些字符由多个字节（最多四个字节）表示。任何字符中首字节的值都指明用于该字符的字节数，还指明该字符是空格字符、数字字符还是字母字符。

对于 UTF8 归类，UTF-8 字符以 1 到 4 字节表示。对于其它多字节归类，则使用一个或两个字节。对于提供的所有多字节归类，由两个或多个字节组成的字符被视为“字母字符”，这样在将其用于标识符时无需加双引号。

Sybase IQ 不支持 16 位或 32 位字符集，如 UTF-16 或 UTF-32。

所有 Client Library（嵌入式 SQL 除外）都支持 Unicode，并使用 UTF-16 编码。系统会在客户端与服务器之间进行转换。

日语语言支持

对于日语 Windows 应用程序，Sybase 建议使用归类 932JPN。

归类 932JPN 支持装载 32 位多字节字符，这种字符是无法装载到 SJIS 或 SJIS2 中的。SJIS 和 SJIS2 是较早的归类。SJIS 可用作替代归类。SJIS2 则不再受支持。对于 Unix 应用程序，应使用 EUC_JAPAN。

泰语语言支持

Sybase IQ 提供 **CP874toUTF8** 实用程序，以将 CP874 格式的数据文件转换为 UTF8，后者是 Sybase IQ 针对泰语提供支持的归类。

有关语法，请参见《实用程序指南》。还可以装载 CP874 字符集中的数据，而不必使用该实用程序将数据转换为 UTF8。

SORTKEY() 函数按 **thaidict**（泰语字典，格式为 UTF8 的泰语字符集）排序顺序返回值。有关语法，请参见《参考：构件块、表和过程》。以下语句会产生相同的结果：

```
SELECT c1, SORTKEY(c1) from T1 where rid=3
SELECT c1, SORTKEY(c1, 'thaidict') from T1 where rid=3)
SELECT
'\340\270\201\340\271\207',SORTKEY( '\340\279\201\340\271\207' )
from T1 where rid=3
```

了解字符集转换

Sybase IQ 可以在包含相同字符的字符集之间执行字符集转换，但这些字符在字符集或代码页中应处于不同的位置。

字符集之间必须存在某种程度的兼容性，才能进行字符集转换。例如，EUC-JIS 与 cp932 字符集之间可以进行字符集转换，但 EUC-JIS 与 cp1252 之间则不能进行转换。

以下主题说明 Sybase IQ 如何执行字符集转换。此信息是为高级用户（如在多字符集环境中配置应用程序或数据库的用户）提供的。

数据库消息的字符转换

来自数据库软件的错误和其它消息包含在语言资源库中。此库的本地化版本随 Sybase IQ 的本地化版本提供。

客户端应用程序用户可看到来自数据库的消息和来自数据库的数据。某些数据库消息（来自语言库的字符串）可能包含由数据库中的字符填充的占位符。例如，如果使用不存在的列执行查询，返回的错误消息为：

```
找不到列 column-name
```

其中 *column-name* 用数据库中的字符填充。

为了以一致的方式为客户端应用程序展示这几种信息（即使数据库使用的是语言库中的不同字符集），数据库服务器会自动转换消息的字符，使其与数据库归类中使用的字符集匹配。

系统总是将消息转换为数据库归类字符集，无论是否启用字符集转换都是如此。

如果为数据库服务器启用了字符集转换（缺省设置），且客户端字符集不同于数据库归类中使用的字符集，则会执行进一步的字符集转换。

针对数据库消息使用字符转换

在读取数据库消息之前执行检查。

1. 请确保用于数据库的归类与计算机中使用的字符集兼容，并且与 **Sybase IQ** 语言资源库中使用的字符集兼容。对于 **Sybase IQ** 的不同本地化版本，语言资源库也有所不同。
2. 检查所需的字符在每个字符集中是否存在。

连接字符串和字符集

连接字符串是字符集转换的一种特殊情况。

连接字符串由 **Client Library** 进行分析，以定位或启动数据库服务器。此分析是在服务器的字符集或语言未知的情况下进行的。

接口库采用如下方法分析连接字符串：

- 将连接字符串分解为 *keyword = value* 的组成部分。只要不在 **CommLinks** 参数两边使用大括号 {}，此操作就可以脱离字符集独立完成。请使用建议的括号 ()。在某些多字节字符集中，大括号是有效的后随字节（除首字节之外的字节）。
- 定位服务器。服务器名是按照客户端计算机的字符集解释的。对于 **Windows** 操作系统，使用的是 **ANSI** 字符集。除非会导致客户端计算机与服务器计算机之间产生字符集转换问题，否则可以使用扩展字符。
为获得不同计算机之间最大程度的兼容性，应使用这样的服务器名：由字母 **ASCII** 字符 1 到 127（或 33 到 126）和下划线组成，且不含标点字符。服务器名的长度截断为 40 个字符。
- **DatabaseName (DBN)** 或 **DatabaseFile (DBF)** 参数在数据库服务器字符集中进行解释。
- 定位数据库后，根据数据库的字符集解释剩余的连接参数。

另请参见

- 连接和通信参数（第 77 页）
- 简单连接示例（第 48 页）

避免字符集转换

字符集转换会有相关的性能成本。

如果可以设置不需要字符集转换的环境，则不必付出此成本，而且设置的维护也比较简单。

如果使用单字节字符集，而且只关注七位 ASCII 字符（值 1 到 127），则无需进行字符集转换。即使数据库中的代码页与客户端操作系统中的代码页不同，它们在此字符范围内也是兼容的。许多英语语言系统都满足这些要求。在此版本中，缺省情况下已启用字符集转换。

如果需要扩展字符，也许可以采取另外一些措施：

- 如果客户端计算机操作系统中的代码页与数据库中使用的代码页匹配，则无需对数据库中的数据进行字符集转换。
- 如果能够使用针对您的语言生成的 Sybase IQ 版本，并且使用操作系统中的代码页，则无需对数据库消息进行字符集转换。Sybase IQ 消息字符串中使用的字符集如下：

语言	字符集
英语	cp1252
法语	cp1252
德语	cp1252
日语	cp932 (Shift-JIS)

另外请记住，在缺省情况下会进行客户端/服务器字符集转换。如果在连接字符串中指定 **CharSet=none**，则会禁用字符集转换。

配置字符集环境

设置计算环境以便正确处理字符集问题。

1. 确定您的环境中每个计算平台的缺省区域设置。缺省区域设置是每台计算机的字符集和语言。在 Windows 操作系统中，字符集是 ANSI 代码页。
2. 确定区域设置是否适用于您的环境。
3. 如果缺省设置不合适，请确定与您的数据匹配的字符集、语言和数据库归类，以避免字符集转换。
4. 将环境中每台计算机上的区域设置设置为这些值。
5. 使用缺省归类创建数据库。如果缺省归类不能满足您的需要，则可以使用指定归类创建数据库。
6. 在为数据库选择归类时，请考虑以下因素：

- a) 所选择的归类要使用适合于数据库中数据的字符集和排序顺序。经常会有多个备选归类符合要求，其中有些是 OEM 归类，有些是 ANSI 归类。
- b) 选择一个无需进行字符集转换的归类。在使用字符集转换时，会产生性能成本，并在系统配置方面增加额外的复杂性。通过使用数据库中与客户端计算机操作系统正在使用的字符集匹配的归类序列，可以避免字符集转换。如果客户端计算机使用的是 Windows 操作系统，则应选择 ANSI 字符集。对于 15.3 版或更高版本的 Sybase IQ 数据库服务器，缺省情况下已启用字符集转换。在命令行中使用 **CharSet=none**，可以关闭字符集转换。

有关字符集的详细信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “字符集和归类参考信息” > “建议的字符集和归类”。

注意： 此参考指向 SQL Anywhere 文档。

区域设置信息

使用系统函数来确定区域设置信息。

要查看如何使用系统函数返回有关客户端连接、数据库和数据库服务器的区域设置信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “有关国际语言和字符集的任务” > “确定区域设置信息”。

注意： 此参考指向 SQL Anywhere 文档。

另请参见

- CharSet 连接参数 [CS] (第 81 页)
- Language 连接参数 [LANG] (第 96 页)
- 设置区域设置 (第 369 页)

设置区域设置

您可以使用操作系统中的缺省区域设置，也可以显式设置区域设置以供计算机上的 Sybase IQ 组件使用。

1. 显示 IQ 引擎字符集：

```
select * from sa_eng_properties() where propname like '%CharSet%'
```

2. 设置 IQLANG 和/或 IQCHARSET 环境变量：

```
IQCHARSET=charset;IQLANG=language_code
```

其中 *charset* 是有效的字符集标签，*language_code* 是来自语言标签值列表的语言代码。

Sybase IQ 会在执行 **INSERT...LOCATION** 语句时 装载本地化信息。

另请参见

- 连接参数的工作原理 (第 43 页)
- 区域设置信息 (第 369 页)
- CharSet 连接参数 [CS] (第 81 页)
- Language 连接参数 [LANG] (第 96 页)

语言标签值列表

有效的语言标签值以及等效的 ISO 639 语言代码。设置 IQ_LANG 环境变量中的双字母 ISO_639 语言代码。

语言	ISO_639 语言代码	语言标签	替代标签
阿拉伯语	AR	arabic	不适用
捷克语	CS	czech	不适用
丹麦语	DA	danish	不适用
荷兰语	NL	dutch	不适用
英语	EN	us_english	english
芬兰语	FI	finnish	不适用
法语	FR	french	不适用
德语	DE	german	不适用
希腊语	EL	greek	不适用
希伯来语	HE	hebrew	不适用
匈牙利语	HU	hungarian	不适用
意大利语	IT	italian	不适用
日语	JA	japanese	不适用
朝鲜语	KO	korean	不适用
立陶宛语	LT	lithuanian	不适用
挪威语	NO	norwegian	norweg
波兰语	PL	polish	不适用
葡萄牙语	PT	portuguese	portugue
俄语	RU	俄语	不适用
简体中文	ZH	chinese	simpchin

语言	ISO_639 语言代码	语言标签	替代标签
西班牙语	ES	spanish	不适用
瑞典语	SV	swedish	不适用
泰语	TH	thai	不适用
繁体中文	TW	tchinese	tradchin
土耳其语	TR	turkish	不适用
乌克兰语	UK	ukrainian	不适用

为 INSERT……LOCATION 语句设置区域设置

如果数据库针对您的平台使用非缺省区域设置，则必须在本地客户端设置一个环境变量，使 Sybase IQ 能够装载有关语言、归类序列、字符集和日期/时间格式的正确信息。

在确定区域设置名称时，Sybase IQ 首先检查 LC_ALL 环境变量的值。如果未设置 LC_ALL，Sybase IQ 将使用 LANG 环境变量的值。如果这两个变量均未设置，Sybase IQ 将使用区域设置文件中的“default”条目。

1. 在文本编辑器中打开 \$SYBASE/locales/locales.dat 文件。

例如：

```
locale = default, us_english, roman8 locale = C, us_english,
roman8 locale = american, us_english, roman8 locale =
english.iso88591, us_english, iso_1
```

2. 将 LC_ALL 或 LANG 环境变量设置为正确的值。如果在步骤 1 中的平台上，数据库的归类为 iso_1 而且您使用的是英语，则需要将环境变量 LC_ALL 或 LANG 的值设置为“american.iso88591”。否则，Sybase IQ 将使用区域设置名称“缺省”，其归类为“roman8”。

例如，在 sh 或 ksh shell 中：

```
LC_ALL= american.iso88591;export LC_ALL
```

在 csh 或 tsch shell 中：

```
setenv LC_ALL american.iso88591
```

使用指定归类创建数据库

IQ 数据库的缺省归类始终为 ISO_BINENG。您可以在创建每个数据库时指定不同的归类。

有关使用指定归类创建数据库的信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “有关国际语言和字符集的任务” > “使用指定归类创建数据库”。

注意： 此参考指向 SQL Anywhere 文档。

在数据库服务器中禁用字符集转换

您可以在数据库服务器命令行中显式启用和禁用字符集转换。

如果客户端与服务器的区域设置不同，则会进行字符集转换。在 Sybase IQ 中，缺省情况下已启用字符集转换。

在连接字符串中使用 CharSet=none 参数连接。例如：

```
CharSet=none
```

将数据库从一种归类更改为另一种归类

将数据库从一种归类更改为另一种归类需要重建数据库。归类在创建数据库时选择，并且无法进行更改。

有关通过重建数据库更改归类的信息，请参见“SQL Anywhere Server - 数据库管理” > “配置数据库” > “国际语言和字符集” > “有关国际语言和字符集的任务” > “将数据库从一种归类更改为另一种归类”。

注意： 此参考指向 SQL Anywhere 文档。

兼容性问题

在 12.0 版之前，Sybase IQ 始终采用 ASCII 排序顺序，其中大写字符排在小写字符之前。

在 12.4.2 和更高版本中，IQ 数据库缺省采用 12 版 Sybase IQ 以前的方式进行数据排序。该缺省方式应用以下 **CREATE DATABASE** 选项：

```
CREATE DATABASE dbname COLLATION 'ISO_BINENG' BLANK PADDING ON CASE RESPECT
```

使用这些选项，大写字符在归类序列中排在所有小写字符之前。例如，使用这些选项，“XYZ”排在“abc”之前，这与 Sybase IQ 的早期版本相同。

性能问题

与非二进制字符集和归类序列相比，采用二进制字符集和归类序列的字符数据性能更好。

为充分发挥系统性能，应使用以下缺省选项设置创建数据库：

```
CREATE DATABASE dbname COLLATION 'ISO_BINENG' CASE RESPECT
```

这些选项将产生二进制字符集和归类序列。对于这两个选项，所有其它设置都将构成非二进制字符集和归类序列。

这些设置的缺点是大写字符总是排在小写字符之前。例如，BANANA 排在 apple 之前。如果您希望采用更自然的排序顺序，但仍需要创建区分大小写的数据库，而且愿意牺牲一定程度的性能，则可以使用归类 ISO_1 代替缺省归类 ISO_BINENG。

注意： 有关口令区分大小写的详细信息，请参见《参考：构件块、表和过程》>“与其它 Sybase 数据库的兼容性”>“数据定义语言”>“区分大小写”。

数据备份、恢复和存档

保护数据、作业并执行定期备份。您还可以使用只读硬件存档不可修改的数据以便于访问。

数据保护

Sybase IQ 提供的完整功能集可防止您的计算机出现两类故障，并避免数据库出现不一致情况。

- 如果在计算机或操作系统关闭时有部分完成的事务，则会出现 *系统故障*。这可能是由于计算机的关闭或重新启动不当、另一个应用程序导致操作系统崩溃或者电源故障而造成的。
- 如果数据库文件、文件系统或数据库文件存储设备不可用，则会出现 *介质故障*。

出现系统故障后，Sybase IQ 通常可以自动恢复，您无需恢复您的数据库。

出现介质故障之后，或者如果数据库中的数据因任何原因而不一致，则必须恢复数据库。要在上述所有情况下保护您的数据，请对数据库进行定期备份。特别是，应在每次完成向数据库插入大量新数据之后备份您的数据库。

出现故障后，恢复机制将事务作为工作基本单元进行正确处理：回退所有不完整的事务，保留所有已提交的事务。这样，即使出现故障，数据库中的数据也仍会保持一致状态。

另请参见

- 关闭操作系统会话（第 37 页）
- 运行 `sp_iqcheckdb` 的资源问题（第 426 页）
- 系统恢复（第 355 页）
- 系统恢复与数据库修复（第 419 页）
- 验证数据库（第 394 页）

如何备份数据库

使用 **BACKUP** 命令可以备份 IQ 数据库。

备份包括 Sybase IQ 数据（IQ 存储）和基础 SQL Anywhere 数据库（Catalog 存储）。

备份与数据库中的读写操作并发运行。比较来说，在恢复期间，该数据库不允许任何其它操作。

必须连接到数据库才能备份该数据库。**BACKUP** 命令无法指定其它数据库。

有关备份 Multiplex 数据库的信息，请参见《使用 Sybase IQ Multiplex》> “Multiplex 环境中的备份和恢复操作”。

数据存储类型

Sybase IQ 数据存储由一个或多个文件组成。

它们可包含用户数据和用于启动、恢复、备份和事务管理的内部数据库结构。通常，IQ 数据库包含下列存储：

- *db-name.db* 是包含描述数据库和您添加的任何标准 SQL Anywhere 数据库对象的系统表和存储过程的 Catalog 数据库空间。它称为 Catalog 存储，具有 dbspace-name SYSTEM。可以在 Catalog 存储中创建其它数据库空间。
- *db-name.iq* 是主要数据的数据库空间，包含 IQ 表数据和索引。它称为 IQ 存储，具有 dbspace-name IQ_SYSTEM_MAIN。dbfile 名称与 dbspace-name IQ_SYSTEM_MAIN 相符。可以在 IQ 存储中创建多个数据库空间，每个数据库空间可以保存多个 dbfile，其中包括 IQ_SYSTEM_MAIN。
- *db-name.iqtmp* 是包含某些查询生成的临时表的初始临时数据库空间。它称为 IQ 临时存储，具有 dbspace-name IQ_SYSTEM_TEMP。可以向 IQ 临时存储添加 dbfile。

上述任何存储和日志文件都可能出现故障。

备份类型

备份 Sybase IQ 数据有四种方式：

- 数据库备份
- 操作系统级备份
- 虚拟备份
- 存档备份（适用于日志文件）

数据库备份类型

Sybase IQ 提供四种数据库备份类型：

- *完全备份* 创建数据库的完整副本。
- *虚拟备份* 复制 IQ 存储中除表数据以外的所有数据库。
- *增量备份* 复制自上次任意备份类型以来的所有事务。
- *基于全备份的增量备份* 复制自上次完全备份以来的所有事务。

上述所有备份类型都对 Catalog 存储进行完全备份。在大多数情况下，Catalog 存储远小于 IQ 存储。但是，如果 Catalog 存储大于（或略小于）IQ 存储，IQ 的增量备份则大于所需大小或预期大小。

使用 **BACKUP** 语句可支持增量虚拟备份。

不会对临时存储数据进行备份。但是，将备份重新创建临时存储结构所需的元数据和任何其它信息。

备份 IQ 存储和 Catalog 存储

请在备份前阅读剩下的备份主题以获取完整详细信息。

此过程概述了备份步骤。

1. 使用拥有 DBA 特权的帐户连接到服务器。对于 **Multiplex** 数据库，必须连接到协调器。
2. 运行 **BACKUP** 命令。有关完整语法，请参见《参考：语句和选项》。

此步骤将备份下列文件：

- Catalog 存储（**SYSTEM** 数据库空间文件），通常命名为 *dbname.db*
- IQ 存储的所有数据库空间文件

3. 为每一个服务器创建 `params.cfg` 文件的副本。**BACKUP** 不会备份该文件。
- 4.

保存以下文件的长度：

- 协调器上的所有数据库空间文件
- IQ 临时存储

备份中的数据

BACKUP 仅备份已提交的数据。

借助提交和自动检查点开始备份。备份程序在此时确定要备份哪些数据。该程序备份截至此检查点时间为止的数据库当前快照版本。在此检查点操作发生时尚未提交的所有数据不会纳入备份范围。

第二个自动检查点在备份结束时发生。备份期间提交的所有数据将包含在任何后续备份中。

Sybase IQ 仅备份在备份时实际正在使用的可恢复数据库块，而不会备份空闲块。

Sybase IQ 备份数据库文件以及与连接的 IQ 数据库相关的目录信息。此备份不会备份事务日志文件。它不会使用事务日志恢复数据库。

如果由于某一原因而导致事务中的所有命令都无法正常处理，或者数据库缺少某些文件，备份将失败。

备份、恢复和恢复中的事务日志

从系统故障恢复时，Sybase IQ 将使用事务日志文件。

它不会使用事务日志为 Sybase IQ 数据库恢复 IQ 数据库、已提交的 IQ 事务或 Catalog 存储。对于完全恢复，一定不能存在事务日志。必须先删除此文件，然后才能启动完全恢复。

注意：SQL Anywhere 数据库以不同方式使用事务日志和其它日志。如果要恢复 SQL Anywhere 数据库，您需要其事务日志文件，**BACKUP** 将为您保留此文件。请参见“SQL Anywhere Server - 数据库管理” > “维护数据库” > “备份和数据恢复” > “备

份类型”。此外，如果 **Catalog** 存储中含有数据（系统表除外），则仅当在故障前已将该数据的事务写入到磁盘时，系统才会恢复这些事务。

注意： 此参考指向 **SQL Anywhere** 文档。

对事务日志执行实时备份

可以使用带有 **-l** 选项的 **dbbackup** 实用程序对事务日志执行实时备份。

1. 设置一个在联机计算机失败时可运行数据库的辅助计算机。（在该辅助计算机上安装和配置 **Sybase IQ**。）
2. 定期执行到辅助计算机的完全备份。
- 3.

运行将事务日志备份到辅助计算机的实时备份。

```
dbbackup -l path\filename.log -c "connection_string"
```

通常，您应当从辅助计算机运行 **dbbackup** 实用程序。如果主计算机变得不可用，则可以使用辅助计算机重新启动数据库。数据库文件和事务日志保存重新启动所需的信息。

分配备份数据

BACKUP 始终在第一台存档设备上对 **Catalog** 存储进行完全备份，然后在指定的所有设备中并行备份 **IQ** 存储中的数据。

块在存档介质中的分配并不均匀。根据各线程的处理速度，某一设备上的块可能会多于其它设备上的块。

注意： 由于文件集必须按照备份顺序进行恢复，因此备份数据的分配非常重要。

另请参见

- 获取备份和恢复的相关信息（第 409 页）
- 按正确顺序进行恢复（第 402 页）

确保数据库一致

虽然 **BACKUP** 将在备份数据库之前检查所有必要文件是否存在，但是它不会检查内部一致性。

要进行更彻底的检查，您可以在备份前运行存储过程 **sp_iqcheckdb**。

另请参见

- 在恢复后验证数据库（第 404 页）
- 验证数据库（第 394 页）

选择存档设备

您可以将任何 IQ 数据库备份到磁带或磁盘中，包括 WORM 设备。

Sybase IQ 支持以接近于设备的速度使用多个磁带驱动器备份和恢复，或者备份和恢复到多个磁盘（如果正在使用磁盘条带化）。在 **BACKUP** 命令的 *archive_device* 参数中指定备份设备名。

另请参见

- 使用只读硬件存档数据（第 414 页）

磁盘备份要求

磁盘备份必须在文件系统中进行；不支持将原始磁盘作为备份介质。独立设备冗余阵列 (RAID) 设备上的所有磁盘都视为单个设备。

磁带备份要求

如果定期备份大型数据库，则应使用 DLT 驱动器（如果您的平台支持这些驱动器）。在任何情况下，Sybase 都建议您使用多个磁带驱动器。

Sybase IQ **BACKUP** 可支持下列磁带驱动器：

- UNIX 系统上的数字线性磁带 (DLT)
- 4 mm 数字数据存储 (DDS)
- 8 mm

Sybase IQ 还允许带有多个磁带的栈式驱动器。

Sybase IQ **BACKUP** 不支持自动唱片点唱机或机器人装载程序。如果您需要使用它们，请使用第三方介质管理器。

Sybase IQ **BACKUP** 在 UNIX 系统上不支持固定长度的磁带设备，例如四分之一英寸盒式磁带 (QIC) 驱动器。

特定于平台的备份要求

请注意 AIX 和 IBM Linux 的备份要求。

请注意下列特定于平台的备份要求：

- AIX 系统上的磁带设备可配置为固定长度块模式或可变长度块模式。有关如何显示和更改块模式的信息，请参见《安装和配置指南》。Sybase IQ **BACKUP** 不支持固定长度块模式。
- 在 IBM Linux on POWER 上，要将 IQ 数据库备份到 SCSI 磁带，必须设置设备的块大小，使其接受可变长度的数据传输。执行任何 IQ 备份之前，请设置 SCSI 磁带设备的缺省块大小。以超级用户身份登录，然后运行 Linux shell 命令 **mt**，如下所示：

```
mt -f /dev/st0 defblksize 0
```

备份设备数目限制

当使用 Sybase IQ 的 **BACKUP** 命令时，用户可以通过指定多个 **TO** 子句在多个设备上并行执行操作。备份语句的设备数目上限为 36。如果超过此上限，该语句将产生错误。

该限制影响所有版本的 Sybase 风险分析平台 (Risk Analytics Platform) 和 Sybase RAP - Trading Edition™。

将来备份的更正操作

用户必须在 **BACKUP** 命令中使用 36 个或更少的 **TO** 子句来创建将来备份，该限制将在将来版本的 Sybase IQ 中强制实施。

保持较小的备份命令数目。设备数目不应太极端，这会导致更多设备争用 I/O 和硬件。作为实际准则，请在每台计算机上的每个核心中大约使用 1 台设备，以使 CPU 使用饱和。在较快系统上的每个核心中最多使用 2 台设备。

准备备份

为了运行 **BACKUP**，您必须符合下列章节中的要求。

获取 DBA 特权

您需要在数据库上拥有 DBA 特权才能运行 **BACKUP** 或 **RESTORE**。

您必须以 DBA 用户身份登录，或由 DBA 授予 DBA 权限。

回绕磁带

Sybase IQ 在使用磁带之前不会回绕磁带。

必须确保在将用于备份和恢复操作的磁带放入磁带设备之前这些磁带位于相应的起点位置。

如果使用回绕设备，磁带将在备份后进行回绕。如果您的磁带设备将自动回绕磁带，请注意不要覆盖磁带上的任何信息。

保留旧磁盘备份

BACKUP 覆盖具有相同名称的现有磁盘文件。

如果需要保留备份，请在新建备份时对存档设备使用不同文件名或路径名，或者在启动备份前将旧备份移至其它位置。

运行 BACKUP 的两种方式

可以用两种方法运行 **BACKUP**。

- **有人照管** - 在有人照管模式下，**BACKUP** 假定存在操作员，并在必要时提示您装载存档介质。如果要使用此方法，您必须从命令行以交互方式运行 **BACKUP**。

- **无人照管** – 在无人照管模式下，**BACKUP** 假定没有操作员，因此不会发出提示。您必须对所需空间进行相应估计，并由此设置您的设备。任何错误均被视为致命错误。

在某些情况下，您可以使用第三方软件创建备份。这些产品对无人照管的备份尤其有用。

注意： 您可以从批处理脚本或过程以及 **Interactive SQL** 运行 **BACKUP**。还可以使用事件处理程序进行自动备份。请参见《系统管理指南第二卷》。

另请参见

- 无人照管的备份（第 382 页）
- 估计介质容量（第 381 页）

估计介质容量

备份之前，确保存档介质具有足够的空间。

估计磁盘或磁带上的可用空间时，请牢记下列规则：

- 您需要足够的空间才能完全备份 **Catalog** 存储和对 **IQ** 存储进行完全备份或增量备份。如果 **Catalog** 存储包含 **SQL Anywhere** 数据和 **Sybase IQ** 系统表，则还需要相应空间来备份此数据。
- 您不必包括事务日志的空间，因为不会备份此日志。
- 对于磁带备份，指定的第一个磁带集必须能够容纳 **Catalog** 存储的完全备份，包括 **Catalog** 存储中的任何非 **IQ** 数据。（磁带集由给定存档设备上生成的一个或多个备份磁带组成。）
- 对于包含多个磁带驱动器的栈式设备，给定设备的所有磁带的大小必须相同。

Sybase 建议您始终为每次备份使用新磁带。

在启动到磁盘的备份之前，**Sybase IQ** 首先测试是否具有足够的磁盘文件空间来进行备份。对于到磁盘的操作员值守备份，当空间不足时，**BACKUP** 将提示您从磁盘中移动某些文件，然后再写入任何数据。在提供更多磁盘空间之前，不会启动备份。

类似地，如果在有人照管的磁盘备份期间空间不足，**BACKUP** 将关闭所有打开的备份文件，并一直等待，直到检测到您已清除某些空间为止。然后，该命令将使用新的备份文件重新启动。如果您愿意，您还可以停止备份。

缺省情况下，在继续备份之前，您必须至少提供 **8 KB** 可用磁盘空间。

无人照管的备份不能提示您提供更多空间。除非提供足够的空间，否则无人照管的备份将失败。对于无人照管的备份，**BACKUP** 以不同方式处理大小估计。

对于到磁带的操作员值守备份，**BACKUP** 只是简单地开始备份。如果空间不足，您必须装载其它磁带。

运行备份

使用 **BACKUP** 语句运行备份。

另请参见

- 事务信息如何帮助进行恢复（第 356 页）

并发和备份

可以将备份与大多数其它数据库操作并发运行。

以下情况除外：

- 在备份 **Catalog** 存储时，不能更改元数据。
- 备份期间不能运行发出检查点或 **DBCC** 的命令。

但是，请注意，不会备份启动备份时尚未提交的事务。如果在备份期间系统或介质出现故障，则不能恢复未提交的事务。

备份一旦启动，则不能执行 **CHECKPOINT** 命令。

指定是否存在操作员

ATTENDED ON 或 **OFF** 控制在需要新磁带或磁盘文件时是否应介入人工干预。

缺省值为 **ON**。

对于备份到磁盘的无人照管的备份，**BACKUP** 不会提示您添加更多磁盘空间。如果空间不足，将出现错误并暂停 **BACKUP**。

对于备份到磁带的无人照管的备份，**BACKUP** 不会提示装载新磁带。**SIZE** 和 **STACKER** 选项确定空间不足时会发生什么情况。

无人照管的备份

使用 **ATTENDED OFF** 选项，可以指定备份过程中没有任何操作员参与。

Sybase IQ 支持以下两种无人照管的备份功能：

- 在备份过程中，操作员无需对提示作出响应。
- 存档设备可以是栈式驱动器，这种驱动器会自动将一组磁带装载到同一驱动器中。对于有人照管的备份和无人照管的备份，均可使用栈式驱动器。

无人照管的备份会尝试检测备份失败的所有可能原因（磁带介质故障除外），并在尝试进行备份之前报告所有潜在错误，如磁盘或磁带的可用空间以及一致的大小和块因子。

对于以磁盘为目标的无人照管的备份，**Sybase IQ** 首先测试可用磁盘空间是否足以进行备份，但不会预先配备份文件以保留空间。如果另一用户写入该磁盘并导致备份空间不足，则备份将在磁盘空间用尽时失败。

对于以磁带为目标的备份，必须估计每个磁带将包含的数据量，并在 **BACKUP** 命令的 **TOarchive_device** 参数中指定该千字节数。备份程序会检查内部存储的信息，以查看

备份数据库所需的空间大小。如果程序确定磁带中的空间足够，备份将继续进行。但是，如果过高估计了磁带中的可用空间大小，备份将在空间用尽时失败。

对于无人照管的备份，如果省略 **SIZE** 参数，必须在一个磁带中容纳整个备份。

如果使用的是第三方备份产品，供应商信息字符串需要指明备份所需的所有信息，如设备规范、文件大小和栈式驱动器。有关详细信息，请参见供应商的文档。

注意： Sybase IQ 不允许进行无人照管的恢复。

指定备份类型

FULL | INCREMENTAL | INCREMENTAL SINCE FULL 指定备份类型。

请选择下列各项之一：

- **FULL** 会对 Catalog 存储和 IQ 存储进行完全备份。**FULL** 是缺省操作。对于虚拟备份，可以使用 **BACKUP** 语句的 **VIRTUAL DECOUPLED | VIRTUAL ENCAPSUATED** 选项。
- **INCREMENTAL** 对 Catalog 存储进行完全备份，然后备份自上次任意类型的 IQ 备份以来对 IQ 存储的所有更改。
- **INCREMENTAL SINCE FULL** 对 Catalog 存储进行完全备份，然后备份自上次完全 IQ 备份以来对 IQ 存储的所有更改。

使用 **BACKUP** 语句的 **VIRTUAL DECOUPLED** 和 **VIRTUAL ENCAPSULATED** 选项可支持 **INCREMENTAL** 和 **INCREMENTAL SINCE FULL** 虚拟备份。

使用 **READWRITE FILES ONLY** 关键字可以将数据库中的读写文件集限制为完全备份、基于全备份的增量备份或增量备份。备份的读写数据库空间或文件必须隶属于 IQ 主存储区。当备份命令检查目录中的读写状态时，系统将选择备份文件。

IQ 备份可以备份一组只读数据库空间和/或只读文件。只读数据库空间或文件必须隶属于 IQ 主存储区。备份文件由用户选择。

另请参见

- 调度例行备份（第 412 页）

指定虚拟备份

VIRTUAL DECOUPLED | VIRTUAL ENCAPSUATED 'shell-command' 选项指定虚拟备份类型。

VIRTUAL ENCAPSULATED 参数的 *shell-command* 变量允许 shell 命令执行系统级备份，以作为备份操作的一部分。

另请参见

- 虚拟备份（第 390 页）

指定存档设备

TO *archive_device* 子句指示备份的目标磁盘文件或系统磁带驱动器并控制存档设备的数量。

备份到磁盘的备份文件名

BACKUP 始终通过向您指定的 *archive_device* 名称附加一个后缀来为磁盘备份文件分配文件名。

该后缀包含 “.”，后跟一个随各新文件递增 1 的数字。例如，如果您指定 */iqback/mondayinc* 作为 *archive_device*，则备份文件为 */iqback/mondayinc.1*、*/iqback/mondayinc.2*，依此类推。此约定使您能够根据需要的大小存储备份，并允许您控制文件大小；有关详细信息，请参见 **SIZE** 选项。您的文件系统必须支持长文件名才能支持此约定。

必须确保为 *archive_device* 指定的目录名存在。**BACKUP** 不会创建缺少的目录。如果尝试在不存在的目录中启动备份，备份将失败。

应避免使用相对路径名指定磁盘文件位置。**BACKUP** 将此路径名解释为服务器启动的相关位置，执行备份时，您可能无法明确识别。此外，如果该路径指向的其它目录中存在数据，您可能没有足够的空间来进行备份。

定位磁带设备的位置

BACKUP 不会为您定位磁带的位置。您必须在启动备份前定位磁带的相应位置，并确保在使用回绕磁带设备时不会覆盖任何备份。为此，Sybase 建议您使用非回绕磁带设备。有关相应的命名约定，请参见适用于您的平台的操作系统文档。

在 UNIX 上指定磁带设备

下面给出了几个在 UNIX 平台上指定非回绕磁带设备的示例：

- 在 Sun Solaris 平台上，请在设备名后插入字母 **n** 以表示“无回绕”，例如，“*/dev/rmt/0n*”。
- 在 IBM AIX 平台上，使用后跟有数字的小数点，该数字指定具有回绕设置的相应压缩，例如，“*/dev/rmt0.1*”。
- 在 HP-UX 平台上，使用“**0m**”指定缺省磁带机制，并使用“**n**”表示“无回绕”，例如，“*/dev/rmt/0mn*”。

警告! 如果磁带设备名拼写错误，并且写入一个不是系统上的有效磁带设备的名称，**BACKUP** 将假定该名称为磁盘文件。

在 Windows 上指定磁带设备

Windows 系统不会指定回绕设备或没有任何回绕设备，并且仅支持对磁带设备执行固定长度的 I/O 操作。Sybase IQ 需要可变长度设备。要在 Windows 系统上容纳固定长度的磁带 I/O，它需要执行其它处理。

Windows 支持磁带分区，但 Sybase IQ 不使用磁带分区，因此，请不要为 Sybase IQ 备份或恢复使用其它应用程序设置磁带格式。在 Windows 中，第一个磁带设备为 “\\.\tape0”，第二个磁带设备为 “\\.\tape1”，依此类推。

警告! 对于备份（以及对于大多数其它情况），当反斜杠位于 n、x 或其它反斜杠之前时，Sybase IQ 会将字符串中的前导反斜杠视为转义字符。因此，当指定备份磁带设备时，必须根据 Windows 命名约定将每个反斜杠变为双反斜杠。例如，将您要备份到的第一个 Windows 磁带设备表示为 “\\\\.\tape0”，第二个表示为 “\\\\.\tape1”，依此类推。如果您忽略额外的反斜杠，或错误拼写了磁带设备的名称，并在您的系统中写入了一个无效磁带设备的名称，则 Sybase IQ 会将此名称解释为磁盘文件名。

指定磁带备份大小

TO 子句的 **SIZE** 选项用于标识该带区中备份数据的最大大小（以 KB 为单位）。

如果使用 Sybase 提供的备份（相对于第三方备份产品），则应在不能可靠地检测磁带结束标记的平台上为无人照管的磁带备份指定 **SIZE**。请注意，**SIZE** 的值基于每个输出设备而不同。在相应设备上使用的任何卷均不能小于此值。虽然 IQ 不要求您为有人照管磁带备份指定 **SIZE**，但最好提供准确的大小估计。

备份期间，如果任何磁带空间不足，并且您未指定 **SIZE**，则会收到错误。如果任何磁带的空间在用尽时并未达到指定大小，则不会立即收到错误；而会发生下列情况：

- 对于指定了 **SIZE** 和 **STACKER** 的无人照管的备份，备份将尝试打开下一磁带。
- 对于指定了 **SIZE** 但未指定 **STACKER** 的有人照管的备份，备份将要求您放入新磁带。
- 对于指定了 **SIZE** 和 **STACKER** 的无人照管的备份，备份将尝试打开下一磁带。如果没有任何卷可用，或者如果未指定 **STACKER**，则会收到错误。

任何附加磁带都不包含恢复所需的标头信息，因此，恢复时应按顺序谨慎装载磁带，否则，数据库可能会不一致。

在 Windows 中，磁带设备的 **SIZE** 选项还具有特殊要求：

- **SIZE** 的值必须是 64 的倍数。其它值则向下舍入为 64 的倍数。
- 如果未显式指定 **SIZE**，该值将自动设置为 1.5GB。

指定磁盘备份大小

TO 子句的 **SIZE** 选项用于标识该带区中备份数据的最大大小（以 KB 为单位）。请注意，**SIZE** 的值基于每个输出设备而不同。

如果使用 Sybase 提供的备份，无论是有人照管还是无人照管，当您命名为 *archive_device* 的任何磁盘文件大于缺省值 2GB (UNIX) 或 1.5GB (Windows) 时，请指定 **SIZE**。

备份期间，如果写入到给定 *archive_device* 的信息量达到 **SIZE**，备份将关闭当前文件，创建其它具有相同名称的文件，并在此文件名后附加下一升序数字。

例如，如果指定一个 *archive_device*（名为 *janfull* 的磁盘文件），并指定 **SIZE** 为 200000（表示文件最大大小为 200MB），但您的备份需要 2GB，则 **BACKUP** 将创建

十个 200MB 的文件：janfull.1、janfull.2……janfull.10。必须先确保您的磁盘能够容纳如此多的数据，然后再执行备份。

指定栈式设备

TO 子句的 **STACKER** 选项表示您要备份到自动装载的具有多个磁带的栈式设备，并指定该设备中的磁带数。如果 **ATTENDED** 为 **ON**，并且指定了 **STACKER**，**BACKUP** 将无限地等待装载下一磁带。给定栈式设备中的所有磁带的大小都必须相同。

为第三方备份指定设备

注意： 如果要使用第三方备份产品，请不要指定 **SIZE** 或 **STACKER**，因为 *vendor_specific_information* 字符串中含有大小信息。

另请参见

- 使用非 Sybase 产品执行备份（第 389 页）
- 记录数据库空间名称（第 411 页）
- 按正确顺序进行恢复（第 402 页）
- **RESTORE** 语句（第 398 页）
- 检验数据库备份（第 406 页）

其它备份选项

您可能希望设置多个其它 **BACKUP** 命令选项以自定义备份。

指定块因子

BLOCK FACTOR 指定一次写入到存档设备的 IQ 块数。

该值必须大于 0，否则 **BACKUP** 将返回错误消息。**BLOCK FACTOR** 在 UNIX 平台上缺省为 25。在 Windows 中，缺省 **BLOCK FACTOR** 基于数据库的块大小。例如，如果块大小为 512 字节，则 **BLOCK FACTOR** 为 120 个块。如果块大小为 32 KB，则 **BLOCK FACTOR** 为 1 个块。

此参数还控制备份期间用于缓冲区的内存量以及对备份性能的直接影响。块因子的影响是磁盘子系统速度、磁带速度和处理器速度共同作用的结果。某些系统采用较小块因子可获取更高备份性能，而其它系统采用较大块因子，可获得更好备份性能。有关平台的最佳 I/O 大小和块因子的信息，请参见平台操作系统文档。

指定错误检查

CRC ON 或 **OFF** 激活或停用针对每个块的 32 位循环冗余检查。

(**BACKUP** 还使用在硬件中提供的任何错误检测功能。) 通过 **CRC ON**，将在后续 **RESTORE** 操作期间检验对备份计算的校验和。缺省值为 **CRC ON**。

添加注释

WITH COMMENT 指定最大长度为 32KB 的字符串，以作为备份存档的标头信息的一部分。

如果忽略此选项，**BACKUP** 将输入空值。通过执行 **RESTORE DATABASE FROM CATALOG ONLY** 或显示 Sybase IQ 提供的备份日志 `backup.syb`，您可以查看注释字符串。

要备份仅用于 SQL Anywhere 的数据库，请参见“SQL Anywhere Server - SQL 参考” > “SQL 语句” > “BACKUP 语句”以了解其它 **BACKUP** 选项。

注意： 此参考指向 SQL Anywhere 文档。

等待磁带设备

在执行备份和恢复操作期间，如果 Sybase IQ 无法打开存档设备（例如，当需要装载的介质时），服务器将等待十秒钟，然后重试。

服务器将无限期地继续这些尝试，直至操作成功或通过 **Ctrl+C** 终止操作。会向 `.stderr` 文件中写入一条消息。没有任何主控台通知服务器无法打开存档设备。

使用只读硬件备份和恢复

Sybase IQ 对备份和恢复操作都使用只读硬件。

有以下一些规则：

- Sybase IQ 阻止在恢复期间写入到只读设备，因为该设备可能会在硬件级冻结为只读模式。
- 虚拟备份不会备份或恢复只读数据库空间的标头块或只读数据库空间的任何其它块。由于保证只读数据库空间不会发生更改，因此，在只读数据库空间发生介质故障后，虚拟备份和恢复只需要恢复只读数据库空间。
- 非虚拟完全备份将备份所有数据库空间，而不管模式如何。
- 非虚拟增量备份将不会备份下列只读数据库空间：
 - 执行增量备份所依赖的上一备份时的只读数据库空间，以及
 - 自此尚未更改的数据库空间。

上述数据库空间的内容将完全包含在上一先决备份中。将备份自先决备份以来已更改的只读数据库空间。

备份示例

示例主题说明了备份选项。

示例 1 - 完全备份

本示例在有人照管的情况下将数据库 `iquser` 完全备份到 UNIX 上的两个磁带设备。在运行此备份之前，必须将磁带定位到将写入备份文件的开始位置，然后连接到 `iquser`。接着发出以下命令：

```
BACKUP DATABASE
TO '/dev/rmt/0n'
TO '/dev/rmt/1n'
WITH COMMENT 'Jan 18 full backup of iquser'
```

Catalog 存储首先备份到 `/dev/rmt/0n`。然后，IQ 存储备份到两个磁带。

示例 2 - 增量备份

要对同一数据库执行增量备份，这一次请仅使用一个磁带设备，并发出如下命令：

```
BACKUP DATABASE
INCREMENTAL
TO '/dev/rmt/0n' SIZE 150
WITH COMMENT 'Jan 30 incremental backup of iquser'
```

其它示例

本章稍后部分提供了如何从这两个备份恢复此数据库的示例。

有关指定只读文件和数据库空间的备份的示例，请参见《参考：语句和选项》中的 **BACKUP** 语句。

备份时从错误中恢复

备份失败的可能原因有两种：空间不足或硬件故障。

第三方软件的问题也可能导致故障。

检查备份空间

BACKUP 使用 **STACKER** 和 **SIZE** 参数确定是否有足够的空间进行备份。

- 对于磁盘备份，如果该命令确定您提供的空间不足，则会在实际写入任何数据之前使备份失败。
- 如果该命令确定具有足够的空间来启动备份，但随后又在结束前用尽所有空间（例如，当估计不正确或其它应用程序中的用户在备份执行期间占用大量磁盘空间时），有人照管的备份将提示您装载新磁带或释放磁盘空间。无人照管的备份将在空间不足时失败。
- 如果未指定 **STACKER** 和 **SIZE**，备份将继续进行，直到完成或磁带/磁盘已满为止。如果空间不足，有人照管的备份将提示您装载新磁带或释放磁盘空间；无人照管的备份将失败。

恢复尝试

如果备份失败，备份程序将尝试进行恢复。

恢复进程包括：

- 如果备份在备份开始或结束时的检查点失败，则不会执行常规检查点恢复。
- 如果备份在两个检查点之间失败，则会回退备份。
- 如果系统在初始和最终检查点之间的任意时间失败，并且必须恢复数据库，则您必须使用较早的备份磁带集或磁盘文件进行恢复。
- 如果系统在 **FULL** 备份后的最终检查点期间失败，您可以从刚创建的备份磁带或文件恢复。

完成备份之后

如果您需要移动数据库或数据库中的一个数据库空间，您需要了解备份时该数据库中的每个数据库空间的名称。

Sybase IQ 包含一种机制，可使用 **RESTORE SQL** 语句的 **VERIFY** 子句来检验现有的 Sybase IQ 12.6 版或更高版本数据库备份。

使用非 Sybase 产品执行备份

Sybase IQ 支持使用多个第三方产品进行备份和恢复。使用的软件包必须符合 Adaptive Server Enterprise 备份接口。请查看您的产品文档，确保它支持 Sybase 数据库。

要执行此类备份或恢复，请发出 **BACKUP** 或 **RESTORE** 语句，如同您使用 Sybase IQ 执行操作一样，但需要注意以下例外情况：

- 对于每个 *archive_device*，请不要指定实际设备名，而应指定采用以下格式的字符串：

```
dll_name::vendor_specific_information
```

- 不要指定 **STACKER** 或 **SIZE** 参数。

dll_name 对应于在运行时装载的动态链接库。其长度可以为 1 至 30 字节，并且只能包含字母数字和下划线字符。*dll_name* 对于每个 *archive_device* 必须相同。

vendor_specific_information 的内容因产品而异，并且可能因各个 *archive_device* 而有所不同。总字符串（包括 *dll_name::* 和供应商信息）可以长达 255 字节。

备份程序将供应商信息自动传递到第三方案序。当请求第三方备份时，备份程序将此信息放置到备份标头文件中，并将标头文件写入到为每个指定的 *archive_device* 实际创建的第一个磁带或磁盘文件中。

注意： 只有某些第三方产品已使用此语法获得 Sybase IQ 认证。有关其它用法说明或限制，请参见《发行公告》。在使用任何第三方产品按上述方式备份 IQ 数据库之前，请确保该产品已经过认证。有关《技术文档》中的 Sybase IQ 产品，请参见 Sybase 认证报告。

另请参见

- 记录数据库空间名称（第 411 页）
- 按正确顺序进行恢复（第 402 页）
- 指定存档设备（第 384 页）
- **RESTORE** 语句（第 398 页）
- 检验数据库备份（第 406 页）

虚拟备份

虚拟备份有时也称为空值备份，它备份除 IQ 存储表数据以外的所有 IQ 数据库。

必须对相应 IQ 存储创建单独的操作系统级副本。要从虚拟备份恢复，必须首先恢复 IQ 存储的相应系统级副本，然后继续执行虚拟备份的 IQ 完全恢复。

虚拟备份可备份：

- 所有 IQ 目录数据
- 所有 IQ 元数据
- IQ 存储中非特定于单个表的所有元数据。（包括空闲列表、备份和检查点信息。）

除上述各项之外，虚拟备份不会备份表中的数据或元数据。

要进行虚拟备份，当执行完全 IQ 备份时，请在 **BACKUP** 命令中指定 **VIRTUAL DECOUPLED** 或 **VIRTUAL ENCAPSULATED** 参数。**VIRTUAL** 参数阻止 IQ 将 IQ 存储中的表数据和元数据复制到备份文件。

另请参见

- 指定虚拟备份（第 383 页）

虚拟备份类型

虚拟备份有以下两种类型。

- **封装式虚拟备份** - 恢复系统级备份后接着恢复 IQ 虚拟备份可生成经过全部恢复的数据库。
- **分离式虚拟备份** - 恢复系统级备份，接着恢复 IQ 虚拟备份，然后执行“基于全备份的增量备份”恢复，此过程可生成经过全部恢复的数据库。

执行封装式虚拟备份

为使表数据的系统级备份与虚拟备份一致，而不需要执行其它步骤，系统级备份必须在执行备份命令期间由备份事务执行。 T

参数 **VIRTUAL ENCAPSULATED 'shell-command'** 允许将任意 **shell** 命令作为备份操作的一部分执行，以确保上述语义。如果 **shell** 命令返回非零状态，备份操作将返回错误。用户必须确保 **shell** 命令正确执行系统级备份。

使用类似如下 SQL 语句：

```
BACKUP DATABASE FULL VIRTUAL ENCAPSULATED 'dd if=iqdemo.iq  
of=iqdemo.iq.copy' TO 'iqdemo.full'
```

从封装式虚拟备份恢复

请按照以下步骤从封装式虚拟备份进行恢复。

1. 恢复 IQ 存储的系统级副本。
2. 从备份文件执行完全 IQ 恢复。
3. 启动 IQ 数据库。

执行分离式虚拟备份

如果在备份事务之外执行系统级备份，IQ 存储备份将与 IQ 备份文件不一致。

但是，非虚拟 IQ 增量备份和虚拟完全备份将表示一致数据库。这是因为 IQ 增量备份将复制在虚拟完全备份期间或之后更改的所有 IQ 存储数据和元数据。请注意，即使是作为备份命令一部分的自动提交和检查点也会修改 IQ 存储，这将使独立的系统级备份不一致。尝试在不应用增量恢复的情况下使用数据库将导致不可预测的结果。

1.

使用类似如下 SQL 语句执行完全 IQ 备份：

```
BACKUP DATABASE FULL VIRTUAL DECOUPLED TO 'iqdemo.full'
```

2.

使用 shell 命令执行 IQ 存储的系统级备份：

```
dd if=iqdemo.iq of=iqdemo.iq.copy
```

3.

执行非虚拟增量 IQ 备份：

```
BACKUP DATABASE INCREMENTAL SINCE FULL TO 'iqdemo.isf'
```

从分离式虚拟备份恢复

请按照以下步骤从分离式虚拟备份进行恢复。

1.

恢复 IQ 存储的系统级副本，例如：

```
dd if=iqdemo.copy of=iqdemo.iq
```

2.

从 IQ 完全备份文件恢复。

```
RESTORE DATABASE iqdemo.db FROM 'iqdemo.full'
```

3.

从 IQ 增量备份文件恢复。

```
RESTORE DATABASE iqdemo.db FROM 'iqdemo.isf'
```

4. 启动 IQ 数据库。

使用 SAN 快照或影子硬件执行虚拟备份

存储区域网络 (SAN) 快照或影子硬件通过允许对影子副本而不是主数据库执行系统级备份，从而在备份过程方面提供了更大的灵活性。

可分隔影子，以替代作为虚拟备份一部分的系统级副本。然后，可根据 IQ 存储的影子副本执行系统级备份。这样，完全备份可快速完成。

系统级备份

BACKUP 命令是用于备份 IQ 数据的最可靠方法。如果您小心按照正确的过程执行操作，则可以对 IQ 数据库使用系统级备份。

使用系统级备份来备份 IQ 数据库时必须遵循这些过程。如果您在未采取上述安全措施的情况下尝试从系统级备份恢复 IQ 数据库文件，则可能因执行系统级备份时数据库中的活动或缺少的文件而导致数据丢失或不一致。

关闭数据库

执行系统级备份期间，必须关闭 IQ 数据库。有关在 Multiplex 系统上执行系统级备份的详细信息，请参见《使用 Sybase IQ Multiplex》。

启动系统级备份前，必须关闭 IQ 数据库。此外，还必须确保在系统级备份完成之前没有任何人启动 IQ 数据库。

确保数据库处于关闭状态

当数据库顺利关闭时，.db 文件的文件保护为只读，当数据库正在使用时，文件保护则设置为读/写。如果要写入脚本以执行备份，最好让脚本检查文件的访问模式，确保数据库已关闭。

为了确保数据库保持关闭，脚本可在该脚本的首尾处检查 .iqmsg 文件的大小，确保它未发生更改。如果数据库在脚本运行时已启动，则 .iqmsg 文件会变大。

另请参见

- 备份正确的文件（第 392 页）
- 从系统级备份恢复（第 394 页）

备份正确的文件

备份必需文件和可选文件。

所需文件

必须备份下列文件：

- 所有 SYSTEM 数据库空间文件，通常名为 dbname.db

注意： Catalog 存储中可能存在其它数据库空间，SYSDBSPPACES 中列出了这些数据库空间。

- 系统恢复所需的事务日志文件，通常名为 `dbname.log`
- `IQ_SYSTEM_MAIN` 数据库空间文件，通常名为 `dbname.iq`
- `IQ` 主存储区中已添加的任何其它数据库空间的文件

保存以下文件的长度：

- `IQ_SYSTEM_TEMP` 数据库文件，通常名为 `dbname.iqtmp`
- `IQ_SYSTEM_TEMP` 中已添加的其它文件

无需备份临时数据库空间。只要 `IQ` 知道数据库启动时的正确文件长度，`IQ` 即可重建任何临时数据库空间。因此，您只需记录用于保存临时数据库空间的文件或原始设备的大小。

可选文件

Sybase 建议您备份 ASCII 消息文件（例如 `dbname.iqmsg`）以及 `$IQDIR15/logfiles/*.srvlog` 和 `$IQDIR15/logfiles/*.stderr` 文件，即使无需恢复这些文件也是如此。如果在恢复期间出现问题，则 `.iqmsg` 文件会包含证明数据库在启动备份前已关闭的信息。

这些文件对诊断要从其中恢复的数据库故障的原因非常有用。请务必在恢复前创建副本，以供以后分析使用。

如果启用 `IQ` 消息日志回卷，您可能需要备份 `.iqmsg` 文件，以便在需要消息来进行诊断时可访问所有这些信息。

如果启用了消息日志存档（`IQMsgMaxSize` 服务器选项或 `-iqmsgsz` 服务器启动开关不等于零，并且 `IQMsgNumFiles` 服务器选项或 `-iqmsgnum` 服务器启动开关不等于零），则服务器会自动备份消息日志存档。最多可存档 128 GB 的消息日志，在大多数情况下该大小已足够。

注意： 在服务器重新启动之前，需要备份消息日志存档。服务器重新启动之后，系统将忽略现有日志存档，并在 `dbname.iqmsg` 文件已满时创建新的存档。如需保留旧存档日志，请在重新启动服务器之前备份这些文件。

更新备份列表

为系统备份规范添加已添加到数据库的任何数据库空间非常关键，不管这些数据库空间是位于 `SYSTEM`、`IQ_SYSTEM_MAIN` 还是 `IQ_SYSTEM_TEMP` 中。如果要在将来几个月内或在组织中周转后添加数据库空间，您可能会遗漏此步骤。

要确保备份您所需的所有文件，请使用系统级备份脚本。在该脚本中，启动备份之前，会将从 `SYSDBSPPACES`（用于系统数据库空间）和 `SYSDBSPPACES`（用于 `IQ` 数据库空间）中的所选内容与系统备份规范中已知的数据库空间列表相比较。

原始设备和符号链接

如果数据库文件位于原始设备中，请确保系统备份将备份原始设备内容，而不仅仅备份 `/dev/*` 中的设备名。

如果根据建议将符号链接用于原始设备名，一定要确保系统备份实用程序遵循符号链接并备份该设备。

另请参见

- 从系统级备份恢复（第 394 页）
- 关闭数据库（第 392 页）

从系统级备份恢复

如果必须从系统级备份恢复，则必须确保数据库服务器已关闭，正如在备份期间一样。

当恢复 **Multiplex** 数据库时，必须关闭所有辅助服务器和写入服务器。

确保所有文件存在

恢复之前，请查看备份目录，确保 **IQ** 所需的所有文件都存在。文件列表取决于您的应用程序。

对于临时数据库空间文件，请确保存在的文件或原始设备具有正确的文件名（或符号链接）和长度。在数据库重新启动之前，临时数据库空间文件的内容与此无关。

检查所有权和权限

确保所有权和权限级别不会在系统级恢复期间发生更改。

另请参见

- 备份正确的文件（第 392 页）
- 关闭数据库（第 392 页）

验证数据库

仅当数据库在内部一致时，备份数据库才有用。

备份始终确保数据库在继续操作之前处于可用状态。但是，最好在执行备份之前验证数据库，以便确保恢复的数据库处于稳定状态。恢复程序不会检查被恢复数据中的一致性，因为数据库甚至可能不存在。

要验证数据库，请发出以下命令：

```
sp_iqcheckdb 'check database'
```

sp_iqcheckdb 存储过程与服务器启动开关一起用作 **IQ** 数据库一致性检查程序 (DBCC) 的接口。

DBCC 具有不同的检验模式，用于执行按一定程度递增的一致性检查。有三种检查数据库一致性的模式，以及一种重置分配映射的模式。如果在 **sp_iqcheckdb** 命令字符串中指定“数据库”作为目标，每种模式都会检查所有数据库对象。也可以在该命令字符串中指定单个表和索引。如果指定个别表名，还将检查这些表中的所有索引。

数据库选项 **DBCC_LOG_PROGRESS** 指示 **sp_iqcheckdb** 将进度消息发送到 IQ 消息文件。通过这些消息，您可以在 **sp_iqcheckdb** 过程执行时紧跟其进度。

您应在备份之前或之后以及每次怀疑数据库存在问题时运行 **sp_iqcheckdb**。

验证 Multiplex 数据库

应仅在 IQ Multiplex 的写入服务器上运行 **sp_iqcheckdb**。

如果在 Multiplex 辅助服务器上运行 **sp_iqcheckdb**，则会返回错误。

sp_iqcheckdb 的并发问题

在对整个数据库运行 **sp_iqcheckdb** 时，**sp_iqcheckdb** 会读取正在使用的每个数据库页。

此过程占用数据库服务器的大部分时间，以便尽可能提高 I/O 效率。系统中所有其它并发活动的运行速度将低于正常速度。DBCC 的 CPU 利用率可通过指定 **sp_iqcheckdb** 参数 **resources resource-percent** 加以限制，该参数控制着与 CPU 数目有关的线程数。

如果运行 **sp_iqcheckdb** 时有其他用户处于活动状态，您看到的结果将仅反映事务获得的结果。

另请参见

- 数据保护（第 375 页）
- 关闭操作系统会话（第 37 页）
- 运行 **sp_iqcheckdb** 的资源问题（第 426 页）
- 系统恢复（第 355 页）
- 系统恢复与数据库修复（第 419 页）
- 确保数据库一致（第 378 页）
- 在恢复后验证数据库（第 404 页）

恢复数据库

创建数据库并完成完全备份后，可以在必要时恢复数据库。Sybase IQ 会将数据库恢复到截至备份开始时的自动 **CHECKPOINT** 的状态。

进行恢复之前

必须满足条件才能对数据库进行恢复。

- 您必须拥有 DBA 特权。

- 要从存档备份恢复只读文件或数据库空间，当发出 **RESTORE** 语句时，数据库可以正在运行，管理员可以连接到数据库。如果只读文件路径名与数据库系统表信息匹配，则无需与备份中的名称相匹配。

数据库必须未处于运行状态才能恢复 **READWRITE FILES ONLY** 或所有文件备份的 **FULL**、**INCREMENTAL SINCE FULL** 或 **INCREMENTAL** 恢复。

要恢复只读文件的备份，数据库可以正在运行，也可以未在运行。当恢复只读数据库空间中的特定文件或读写数据库空间 (**IQ_SYSTEM_MAIN** 除外) 中的只读文件时，数据库空间必须处于脱机状态。此限制并不适用于 **IQ_SYSTEM_MAIN**。选择性恢复可用于恢复只读数据库空间，条件是该数据库空间仍处于相同的只读状态。

- 要恢复数据库中的所有文件或从仅限读写文件的备份中恢复，必须连接到 **utility_db** 数据库。有关 **utility_db** 以及如何设置使用该数据库的特权的信息，请参见所用平台的《安装和配置指南》。
- 当恢复数据库中的所有文件或从仅限读写文件的备份中恢复时，没有任何用户可以连接到要恢复的数据库。如果指定的数据库有任何只读或读/写用户处于活动状态，**RESTORE** 将退出并返回错误。

Sybase 建议您使用两个启动开关对连接进行限制：

- 使用 **-gd DBA** 以便只有具备 **DBA** 权限的用户才能在正在运行的服务器上启动和停止数据库。（请注意，客户端必须已连接到服务器以便启动或停止数据库，因此此开关不阻止连接。）
- 使用 **-gm 1** 可以按照超出限值的方式启用一个连接和一个 **DBA** 连接，以使 **DBA** 可在紧急情况下采用其中一个连接并中断另一个连接。

限制连接的替代方法是：在启动执行恢复所用的连接后指定

```
sa_server_option('disable_connections', 'ON')
```

并且，在恢复后对同一连接指定

```
sa_server_option('disable_connections', 'OFF')
```

其缺点在于，此方法排除了从其它 **DBA** 连接执行的紧急访问。

- 必须将数据库恢复至适当的服务器，而且该服务器必须具有您需要的存档设备。如果使用 Sybase 提供的恢复功能，则需要与创建备份时相同数量的存档设备（即磁盘文件或磁带驱动器）。
- 若要进行完全恢复，存储文件（缺省为 **.iq** 文件）、**Catalog** 存储（缺省为 **.db** 文件）和事务日志（缺省为 **.log** 文件）不得存在于要恢复到的目标位置中。如果存在上述任一文件，则必须在执行完全恢复之前将其删除或移至其它目录。当完全恢复开始时，它会销毁所有旧的数据库文件，然后重新创建这些文件。必须手动删除存储、**Catalog** 存储和事务日志文件的要求会保护您以免意外地执行完全恢复。
- 对于所有增量恢复，**Catalog** 存储 (**.db**) 必须存在。如果它存在但所在位置与恢复目标位置不同，请移动数据库文件。如果它不存在，则只能进行完全恢复。（如果在任何增量恢复之前进行完全恢复，正确的文件将准备就绪。）
- 对于所有增量恢复，自上次恢复以来数据库必须未发生更改。

恢复需要对数据库和服务器进行独占访问。为使 DBA 更好地控制无意中打开数据库的行为，应在设置 **-gd DBA** 选项的情况下启动数据库服务器，但不启动要恢复的数据库。**RESTORE** 会自动以不允许任何其他用户连接到同一数据库的方式启动该数据库。

必须恢复整个备份或整组备份，包括完全备份和所有后续增量备份。仅对只读数据库空间和文件支持从备份存档恢复单独的文件。但是，您可以使用 **RESTORE** 命令的 **RENAME** 子句将数据库文件移至新的位置。

在恢复数据库之前，可以使用 **RESTORE SQL** 语句的 **VERIFY** 子句来检验数据库备份（现有的 Sybase IQ 12.6 版或更高版本数据库备份）。

另请参见

- 控制权限的命令行选项（第 27 页）
- 移动数据库文件（第 399 页）
- 实用程序数据库（第 13 页）
- 实用程序数据库服务器安全性（第 291 页）
- 检验数据库备份（第 406 页）
- **SPACE ADMIN** 权限概述（第 285 页）

恢复适应数据库空间更改

在一组增量恢复过程中，**RESTORE** 会根据需要创建和删除数据库空间，以匹配恢复所含的操作周期内进行的创建和删除。

例如，假定对数据库进行一次完全备份，然后为该数据库添加一个数据库空间，在添加数据库空间之后再进行一次增量备份。当您从这些备份进行恢复时，**RESTORE** 会在增量恢复开始时为该新数据库空间创建一个文件。类似地，如果删除某个数据库空间，则会在恢复过程中将其删除，但不会删除实际文件。

请注意，在恢复过程中，**SYSTEM** 数据库空间的 **SYSFILE** 系统表中的 **file_name** 列不会更新。对于 **SYSTEM** 数据库空间，创建数据库时，**file_name** 列始终显示其名称。**SYSTEM** 数据库空间的文件名是数据库文件的名称。

恢复磁盘备份文件

如果先备份至磁盘，然后将这些文件移至磁带，则将其移回磁盘文件中时必须使用与创建备份时相同的名称。**Sybase IQ** 无法恢复直接从磁带移至磁带的磁盘文件。

如果使用 **Sybase** 提供的备份和恢复功能进行恢复，则必须为恢复指定与创建备份所用的数量相同的存档设备（磁盘文件）。

恢复磁带备份文件

如果从磁带进行恢复，则必须将磁带定位于 **IQ** 数据的起始处。**RESTORE** 不会为您重新定位磁带。

如果使用 **Sybase** 提供的备份和恢复功能进行恢复，则必须为恢复指定与创建所恢复备份所用的数量相同的磁带驱动器。

为增量恢复指定文件

对于增量恢复，所恢复文件的数量和大小必须与其替换的文件相同，在 IQ 和 Catalog 存储中都是如此。

在恢复之间保持数据库不变

如果您正在进行一组增量恢复，但有用户在您完成整组恢复之前更改了数据库，恢复程序将不允许恢复剩余增量。

例如，如果您有一组恢复，其中包括一个完全恢复和两个增量恢复，某个用户的写入事务在完全恢复之后并且在您发出第二或第三个 **RESTORE** 命令之前提交，您将无法继续进行增量恢复。您必须重新恢复完全备份并应用增量恢复。

如果数据库自上次恢复后已发生更改，则当您尝试进行增量恢复时，将发生以下错误：

```
Database has changed since the last restore
```

注意： 如果数据库自上次恢复后已发生更改，Sybase IQ 将不允许进行增量恢复。但它并不阻止用户进行更改，而是由 DBA 或系统操作员确保在所有恢复完成之前数据库不发生任何更改。

从兼容备份进行恢复

RESTORE 允许恢复 Sybase IQ 15.0 和更高版本的数据库文件。由于数据库格式的变化，您不能从在 12.x 版本上创建的备份中进行恢复。

要将数据从 Sybase IQ 12.x 数据库移至 Sybase IQ15.4，请执行以下操作：

1. 使用 12.6 版《安装和配置指南》中所述的迁移过程升级到 12.6 ESD #11 或更高版本。
2. 按照当前版《安装和配置指南》中所述的迁移过程进行操作。

RESTORE 不允许将 Sybase IQ 备份恢复为 SQL Anywhere 数据库。

RESTORE 语句

要恢复数据库，请使用 **RESTORE** 语句。

您必须以 DBA 的身份连接到 `utility_db` 数据库才能发出此语句。

您必须指定 `db_file` 和至少一个 `archive_device`。

对于 `db_file`，需要为数据库（创建时缺省带有后缀 `.db`）指定 Catalog 存储文件的位置。可以指定完整路径名，也可以指定相对于数据库创建目录的路径名。如果指定一个新路径名，则会将 Catalog 存储以及相对于 Catalog 存储创建的所有文件移至该位置，但包含在 **RENAME** 子句中的所有文件除外。

正如备份那样，每个 `archive_device` 指定 API（第三方），对于 Sybase API，还指定作为恢复来源的物理磁带设备或磁盘文件名。对于第三方 API，`archive_device` 字符

串的内容取决于您的供应商。存档设备不能是原始磁盘设备。如果使用 Sybase API 从磁盘文件进行恢复，则必须提供与创建该备份时指定的数量相同的存档设备。

警告！ 如果拼错了磁带设备名，提供的名称不是系统中的有效磁带设备，**RESTORE** 会假定该名称是磁盘文件并尝试从中进行读取。

注意： 如果是在 Windows 中从磁带设备进行恢复，请注意在指定用于恢复的磁带设备时不需要像备份操作那样重复使用反斜杠。

示例 1 - 恢复至同一位置

此 Windows 示例将一个数据库恢复至 `iquser.db`。该数据库将从两个磁盘文件进行恢复。所有数据库文件均恢复至其初始位置。

```
RESTORE DATABASE 'iquser.db' FROM 'c:\\iq\\backup1' FROM 'c:\\iq\\
\\backup2'
```

另请参见

- 使用非 Sybase 产品执行备份（第 389 页）
- 记录数据库空间名称（第 411 页）
- 按正确顺序进行恢复（第 402 页）
- 指定存档设备（第 384 页）
- 检验数据库备份（第 406 页）

移动数据库文件

将数据库文件移到新位置。

如果需要将数据库文件移至新的位置（例如，如果某一磁盘驱动器发生故障），可采用以下方法之一：

- 要移动包含 Catalog 存储的数据库文件（缺省情况下为 `.db` 文件），只需将新名称指定为 `db_file` 即可。
- 若要移动或重命名事务日志文件，请使用事务日志实用程序 `dblog`。有关语法和详细信息，请参见《实用程序指南》>“`dblog` 数据库管理实用程序”。
- 要移动任何其它数据库文件，可使用 **RENAME** 选项。

恢复至原始设备

如果要恢复至原始设备，应确保该设备的大小足以保存所恢复的数据库空间。如果原始设备大小不足以恢复数据库空间，**IQ RESTORE** 会检查原始设备的大小并返回错误。

操作系统在原始设备中占用一小部分空间，**IQ** 数据库空间占用其余空间。在恢复数据库空间时，原始分区必须同时包含 **IQ** 数据库空间和为操作系统保留的空间。

要将 **IQ** 主数据库空间或临时数据库空间恢复至原始分区，请按下方系统表查找每个 **IQ** 数据库空间所需的原始设备大小：

```
SELECT segment_type, file_name, block_count, data_offset,
block_size, (block_count * block_size) + data_offset AS raw_size
```

```
FROM SYS.SYSIQFILE, SYS.SYSIQINFO where segment_type != 'Msg' ORDER BY 1,2
```

`segment_type` 和 `file_name` 用于提供信息。类型为“Main”或“Temp”的段可以存储在原始分区中，但消息文件（类型为“Msg”）则不能。其中 `file_name` 是数据库空间的名称。

`block_count` 列为整数，表示 IQ 使用的块数。

`data_offset` 列为整数，表示为操作系统保留的字节数。

`block_size` 列为整数，表示每个 IQ 块的字节数。

`raw_size` 列为整数，表示恢复此数据库空间所需的最小原始设备大小（以字节为单位）。Sybase 建议作为恢复目标的原始设备至少要比原本的原始设备大 10 MB。

示例 2 - 移动 Catalog 存储

本示例恢复的数据库与示例 1 相同。

但在示例 2 中，移动的是 Catalog 存储文件以及相对于该文件创建的所有数据库文件。为此，要将原始文件名替换为文件的新位置 `c:\newdir`，方法如下：

```
RESTORE DATABASE 'c:\newdir\iqnew.db' FROM 'c:\iq\backup1' FROM 'c:\iq\backup2'
```

Sybase IQ 对 Catalog 存储以外的数据库文件的移动方法如下：

- 如果指定 **RENAME** 子句，则会将文件移至该位置。
- 如果不指定 **RENAME** 子句，而且文件是使用相对路径名创建的，则会以相对于数据库文件新位置的方式恢复该文件。换言之，最初相对于 **SYSTEM** 数据库空间（包含 Catalog 存储文件）创建的文件将相对于 Catalog 存储文件进行恢复。最初相对于 Catalog 存储创建的文件将相对于 Catalog 存储进行恢复。
- 如果不指定 **RENAME** 子句，而且文件是使用绝对路径名创建的，则会将文件恢复至其初始位置。

换言之，如果希望移动整个数据库，则应在 **RENAME** 子句中指定数据库的每个 IQ 数据库空间（必需数据库空间、临时数据库空间和用户定义数据库空间）的新位置。**SYSTEM** 数据库空间是唯一不应包含于 **RENAME** 子句中的数据库空间。

如果只希望移动部分文件，并且可以覆盖原始文件，则只需对实际希望移动的文件进行重命名。

将每个 `dbfile` 名称指定为其在 `SYSIQDBFILE` 表中显示的值。将 `new_dbpace_path` 指定为该数据库空间的新原始分区，或者新的完整路径名或相对路径名。

不能使用 **RENAME** 选项指定部分恢复。

RENAME 子句中的相对路径名的工作方式与创建数据库或数据库空间时相同：主 IQ 存储数据库空间、临时存储数据库空间和消息日志将恢复至 `db_file` (Catalog 存储) 位置的相对位置；用户创建的 IQ 存储数据库空间将恢复至包含 Catalog 存储的目录的相对目录。

如果在同时恢复完全备份和增量备份的过程中对文件进行重命名，应确保在整组恢复中使用一致的数据空间名称和路径。这是确保文件重命名正确的最安全方法。

如果在完全备份与增量备份之间添加了一个数据库空间，而您正在对数据库文件进行重命名，则对于增量恢复，需要比完全恢复多使用一个 **RENAME** 子句。同样，如果在几次备份之间删除了某个数据库空间，则对于从删除该数据库空间后发生的任何备份进行的恢复，需要少使用一个 **RENAME** 子句。

示例 3 - 移动用户数据库空间

本示例显示如何恢复本章前面提供的示例中的完全备份和增量备份。在本例中，介质故障导致 **UNIX** 原始分区无法使用。必须将该原始分区中的用户定义 **dbfile IQ_USER** 移至新的原始分区 **/dev/rdisk/clt5d2s1**。此操作不影响其它数据库文件。

首先，连接到 **utility_db** 数据库。随后从两个磁带设备恢复完全备份。在本例中，这是进行备份所用的那两个磁带设备，但也可以使用不同的设备，前提是使用相同数量的存档设备、相同的介质类型（磁带或磁盘），并按正确顺序使用相同的磁带组。

第一个 **RESTORE** 命令为：

```
RESTORE DATABASE 'iquser' FROM '/dev/rmt/0n' FROM '/dev/rmt/1n'
RENAME IQ_SYSTEM_MAIN TO '/dev/rdisk/c2t0d1s1' RENAME IQ_SYSTEM_TEMP
TO '/dev/rdisk/c2t1d1s1' RENAME IQ_SYSTEM_MSG TO 'iquser.iqmsg'
RENAME IQ_USER TO '/dev/rdisk/clt5d2s1'
```

第二个 **RESTORE** 命令（用于恢复增量备份）为：

```
RESTORE DATABASE 'iquser' FROM '/dev/rmt/0n' RENAME IQ_SYSTEM_MAIN
TO '/dev/rdisk/c2t0d1s1' RENAME IQ_SYSTEM_TEMP TO '/dev/rdisk/
c2t1d1s1' RENAME IQ_SYSTEM_MSG TO 'iquser.iqmsg' RENAME IQ_USER TO '/
dev/rdisk/clt5d2s1'
```

注意：发出这些命令时也可以仅使用最后一个 **RENAME** 子句，因为只有一个数据库空间要恢复至新的位置。列出所有文件或原始分区（如本例所示）可确保您准确了解每个文件或分区的恢复位置。

另请参见

- 进行恢复之前（第 395 页）
- 控制权限的命令行选项（第 27 页）
- 实用程序数据库（第 13 页）
- 实用程序数据库服务器安全性（第 291 页）
- 检验数据库备份（第 406 页）

使用 **CATALOG ONLY** 选项显示标头信息

CATALOG ONLY 选项可显示数据库的标头信息，它将此信息置于 **.backup.syb** 文件中。

此选项不恢复任何数据，无论是从 **Catalog** 存储还是 **IQ** 存储。

指定 **CATALOG ONLY** 时必须包括 **FROM archive_device** 子句，但应省略 **RENAME** 子句。

另请参见

- 备份日志的内容（第 409 页）
- 显示标头信息（第 405 页）
- 获取备份和恢复的相关信息（第 409 页）

调整数据源和配置文件

在移动数据库时，您可能需要修改数据源、配置文件和集成登录，以反映数据库的新位置。

按正确顺序进行恢复

在从完全备份进行恢复时，会将进行备份时使用的每个块写入磁盘。在从增量备份进行恢复时，则只将上次备份（或上次完全备份）与此次备份之间发生更改的块写入磁盘。

必须按正确顺序恢复完全备份和增量备份，对所恢复的每个备份使用单独的 **RESTORE** 命令。**RESTORE** 可确保备份按顺序进行恢复，如果确定顺序不正确，则会给出以下错误：

```
SQL Code: -1012009
```

```
SQL State: QUA09
```

```
This restore cannot immediately follow the previous restore.
```

要确定正确顺序，需要备份日志中存储的有关备份文件的信息。

按如下方法恢复备份：

- 如果数据库不一致，或要将任何文件移至新位置，则必须恢复 **FULL** 备份。
- 如果最近一次备份是 **FULL** 备份，或需要将数据库恢复至进行任何现有增量备份之前的状态，则只需恢复完全备份。
- 如果在数据库发生故障之前进行了 **INCREMENTAL_SINCE_FULL** 备份，则应首先从上一个 **FULL** 备份进行恢复，然后恢复 **INCREMENTAL_SINCE_FULL** 备份。
- 如果您没有进行 **INCREMENTAL_SINCE_FULL** 备份，但自最后一个 **FULL** 备份后执行了一次或多次 **INCREMENTAL** 备份，则请按备份进行的顺序首先恢复 **FULL** 备份，然后恢复 **INCREMENTAL** 备份。

还可以使用建议的存储过程 **sp_iqrestoreaction** 给出获取稳定数据库集所需的恢复操作序列的建议。始终确认按照上述规则建议的步骤。此外，该存储过程不包括移动任何数据库文件。

在指定备份中，恢复磁带的顺序也很重要。尤其是，需要跟踪每个备份磁带组（即，使用指定存档设备在指定备份中生成的磁带组）中的磁带顺序：

- 必须首先恢复包含 **Catalog** 存储备份的磁带组，且该磁带组必须位于第一个存档设备中。
- 在每个组中，必须按磁带的创建顺序对其进行恢复。

- 不能交错恢复磁带组；必须先恢复一个组，才能恢复另一个组。
- 在第一组之后，各组的恢复顺序无关紧要，只要每组内部顺序正确即可。

应使用与生成备份所用数量相同的驱动器进行恢复，以免意外地交错恢复其它组中的磁带。

示例

假定您在所恢复的完全备份中使用三个存档设备，因此生成了三个磁带组 A、B 和 C。

每个组的内容以及恢复顺序如下：

- **组 A** - 磁带 A₁、A₂ 和 A₃。磁带 A₁ 和 A₂ 包含 Catalog 存储。此组必须首先进行恢复，且必须位于第一个设备中。
- **组 B** - 磁带 B₁ 和 B₂。这些磁带必须作为一组进行恢复，且必须在组 A 之后，但可以在组 C 之前或之后。它们可以位于第二或第三个设备中。
- **组 C** - 磁带 C₁、C₂ 和 C₃。这些磁带必须作为一组进行恢复，且必须在组 A 之后，但可以在组 B 之前或之后。它们可以位于第二或第三个设备中。

恢复程序将检查每组中的磁带在同一设备中的顺序是否正确。如果不正确，则会给出错误，在您提供正确的磁带之前恢复不会继续进行。除包含 Catalog 存储的组外，在指定设备中放置哪个组无关紧要。

注意： 您必须确保首先恢复 Catalog 存储磁带组。恢复程序不会对此进行检查。

尽管这些规则同样适用于磁盘文件，但将多个文件备份至某个指定磁盘设备的可能性不大。

另请参见

- 分配备份数据（第 378 页）
- 获取备份和恢复的相关信息（第 409 页）
- 使用非 Sybase 产品执行备份（第 389 页）
- 记录数据库空间名称（第 411 页）
- 指定存档设备（第 384 页）
- RESTORE 语句（第 398 页）
- 检验数据库备份（第 406 页）

在恢复后重新连接

在某些情况下，Sybase IQ 需要使用 DBF 参数和数据库文件名才能连接到数据库。

如果您使用 Interactive SQL，并且在连接到 utility_db 的情况下从备份恢复了该数据库，则会出现上述情况。

例如，按如下方法纳入 DBF 参数：

```
CONNECT USING 'uid=DBA;pwd=sql;dbf=node1/users/fiona/mydb.db;
links=tcpi{host=serv1;port=1234};eng=serv1_iqdemo'
```

在 Sybase IQ 12.6 ESD5 之前，可使用以下语法连接到恢复的数据库：

```
CONNECT DATABASE mydb USER DBA IDENTIFIED BY SQL
```

而现在，以上命令将返回 `specified database not found` 错误。

避免该错误的另一种方法是在连接到 `utility_db` 的情况下输入 **START DATABASE** 命令，例如：

```
START DATABASE mydb
```

在通过 **Interactive SQL** 进行连接时应使用此方法。

另请参见

- 数据库连接问题（第 452 页）

在恢复后重命名事务日志

如果对数据库中所有其它文件进行重命名或移动，也应当对日志文件进行同样的操作。

要移动或重命名日志文件，可使用事务日志实用程序 (**dblog**)。您应在以下情况下运行此实用程序：

- 对新的数据库名称使用 **RESTORE** 之后
- 对 **RENAME** 选项使用 **RESTORE** 之后

注意：更改事务日志文件名时，数据库服务器不得在该数据库上运行。

另外，在给定某些限制的情况下，还可以使用 **dblog** 重命名事务日志，即使尚未恢复数据库也可以。使用 **dblog** 命令行实用程序，可以通过系统命令行访问事务日志实用程序。请参见《实用程序指南》>“**dblog** 数据库管理实用程序”。

在恢复后验证数据库

为确保磁带已按正确顺序恢复，应在完成数据库的恢复后运行存储过程 **sp_iqcheckdb**。

如果恢复的是一组增量备份，则在恢复每个备份后运行 **sp_iqcheckdb** 最为安全。但为了节省时间，可以选择仅在恢复最后一个增量备份后运行 **sp_iqcheckdb**。

另请参见

- 确保数据库一致（第 378 页）
- 验证数据库（第 394 页）

恢复需要独占写入访问权限

RESTORE 开始后，便不允许其他用户访问指定的数据库。

如果先从完全备份进行恢复，然后从一个或多个增量备份进行恢复，则应确保在几次恢复之间没有任何用户修改数据库。进行修改是允许的，但您将无法再执行任何增量恢复。必须重新开始整个恢复。

如果系统在恢复过程中崩溃，这一限制将扩展到可能需要的所有增量恢复。如果需要从恢复过程中发生的系统或介质故障中恢复，则必须执行下列操作之一：

- 继续执行完全恢复和增量恢复操作的原始序列，或
- 执行完全恢复，然后执行完全恢复数据库所需的所有增量恢复。

缺省数据库服务器启动设置 **-gd DBA** 使 **DBA** 特权成为启动数据库的必要条件。当 **DBA** 运行 **RESTORE** 时，该命令会自动启动数据库，获取恢复所需的信息，然后停止数据库。恢复结束时，该命令将启动数据库，发出检查点，然后再次停止数据库。此过程可确保 **DBA** 在整个恢复过程中具有独占写入访问权限。

所有增量恢复完成后，**DBA** 将再次发出 **START DATABASE** 命令，以允许其他用户访问数据库。

要恢复 **Multiplex** 数据库，请参见《使用 **Sybase IQ Multiplex**》。

显示标头信息

可以使用带有 **CATALOG ONLY** 选项且不含 **FILE** 子句的 **RESTORE** 语句显示标头文件的内容。

带有 **CATALOG ONLY** 的 **RESTORE** 生成的信息格式与实际 **RESTORE** 的备份日志条目相同。

使用命令行实用程序 **db_backupheader**（接受与第一个备份存档对应的文件路径）可以获得有关备份存档的详细信息。该实用程序读取备份存档文件。它不会连接到数据库。

备份存档信息包括：

- 备份信息
- 备份时的数据库信息
- 数据库中各数据库空间的数据库空间信息
- 数据库空间中各 **dbfile** 的 **dbfile** 信息

另请参见

- 备份日志的内容（第 409 页）
- 使用 **CATALOG ONLY** 选项显示标头信息（第 401 页）
- 获取备份和恢复的相关信息（第 409 页）

从恢复过程中发生的错误中恢复

如果操作前段的某次增量恢复失败，数据库仍然可以使用（假定它在恢复之前存在且一致）。

如果完全恢复失败，数据库将无法使用。

如果故障发生在操作中某一点之后，恢复程序会将数据库标记为不一致。在这种情况下，只能通过 **FULL RESTORE** 进行恢复。如果故障发生时正在执行 **FULL RESTORE**，则可能需要返回上一个 **FULL BACKUP**。

检验数据库备份

Sybase IQ 包含一种机制，可使用 **RESTORE SQL** 语句的 **VERIFY** 子句来检验现有的 Sybase IQ 12.6 版或更高版本数据库备份。

检验过程会指令服务器验证指定的 Sybase IQ 数据库备份存档以进行完整、增量、基于完整的增量或虚拟备份，并且检查指定的存档有无恢复过程检查的错误，但不执行写入操作。所有状态和错误消息被写入服务器日志文件中。

备份检验过程可以在数据库所在主机之外的主机上运行。您必须具有 **DBA**、**BACKUP** 或 **OPERATOR** 权限才可运行 **RESTORE VERIFY**。

有关 **RESTORE VERIFY** 子句的语法和用法，请参见《参考：语句和选项》中的 **RESTORE** 语句。

注意： 备份存档的检验不同于数据库一致性检查程序 (**DBCC**) 检验模式 (**sp_iqcheckdb 'verify...'**)。 **RESTORE VERIFY** 验证备份存档的一致性，以确保它可以恢复，而 **DBCC** 验证数据库数据的一致性。

在开始备份之前先运行 **sp_iqcheckdb 'verify...'**。如果备份了不一致的数据库，然后从同一备份存档进行恢复，则数据会继续处于不一致状态，即使 **RESTORE VERIFY** 报告检验成功也是如此。

增量备份的检验

使用 **RESTORE... VERIFY COMPATIBLE** 子句可以对现有的数据库文件进行增量存档的兼容性检查。如果数据库文件不存在于调用 **RESTORE... VERIFY COMPATIBLE** 的系统上，就会返回错误。如果在检验完整备份时指定了 **COMPATIBLE**，此关键字就会被忽略；在恢复完整备份时不需要进行兼容性检查。

如果在不使用 **COMPATIBLE** 的情况下指定了 **RESTORE VERIFY** 进行增量恢复，Sybase IQ 就不查找任何数据库空间，不执行任何兼容性检查。不报告任何警告，即使文件不存在也是如此。只有在包含 **COMPATIBLE** 子句时，才会执行兼容性检查。

如果为增量恢复指定了 **RESTORE VERIFY COMPATIBLE**，而 IQ Catalog 存储或任何 Sybase IQ 数据库空间不存在，就无法进行兼容性检查；会报告错误，而且操作会失败。

在验证增量备份期间，**RESTORE VERIFY COMPATIBLE** 进程会以只读模式打开 Sybase IQ 数据库空间以执行一致性检查。不会修改任何数据库空间。

在增量恢复中，如果数据库已发生修改或者特定增量存档不是数据库的正确存档，则 **RESTORE VERIFY COMPATIBLE** 会报告错误 Database has changed since last restore. (SQLCODE -1012008、SQLSTATE QUA08) 或 This restore cannot immediately follow the previous restore. (SQLCODE -1012009、SQLSTATE QUA09)。

检验进度报告

RESTORE VERIFY 进程检验命令中指定的每个带区。在检验过程检查带区及其相应的文件时，会按照已检验的 IQ 块数报告进度。每检验 5000 个 IQ 块后，都会向服务器日志文件中写入一条消息：

```
5000/100000 (5%) Blocks verified
```

当 IQ 块 100% 检验完后，会向服务器日志中写入最终消息。

在检验操作开始和完成时，消息 **RESTORE VERIFY Started**、检验的 IQ 块数以及 **RESTORE VERIFY successfully completed** 也会分别写入服务器日志中。例如：

```
I. 11/17 06:45:24. VERIFY RESTORE Started I. 11/17 06:45:24. Total
number of IQ blocks to be verified: 764 I. 11/17 06:45:24. Total
number of IQ blocks verified: 764/764 ( 100 % ) I. 11/17 06:45:24.
VERIFY RESTORE Successfully Complete
```

检验错误报告

如果检验过程找到不影响继续进行的错误，该过程就会继续检查存档并记录有关检测到的错误的信息。

不影响检验过程继续进行的错误有：

- Header of block to be restored appears to be corrupted. (SQLCODE -10120111, SQLSTATE QUA11)
- Media data appears corrupted (bad checksum). (SQLCODE -1012012, SQLSTATE QUA12)
- Media meta data appears corrupted (boundary record). (SQLCODE -1012013, SQLSTATE QUA13)
- Media meta data appear corrupted (multiple begin boundary records). (SQLCODE -1012014, SQLSTATE QUA14)

如果发现任何上述错误，并且检验过程可以继续直到存档的结尾，Sybase IQ 就会报告以下错误：

```
The verification of the provided archive has failed. Please
check the server log for details of the errors thrown during
verify.
```

如果发现了与 **RESTORE** 有关的错误，而不是上述错误，则会报告发生的错误，检验过程也会停止。

另请参见

- 进行恢复之前 (第 395 页)
- 控制权限的命令行选项 (第 27 页)
- 移动数据库文件 (第 399 页)
- 实用程序数据库 (第 13 页)
- 实用程序数据库服务器安全性 (第 291 页)
- 使用非 Sybase 产品执行备份 (第 389 页)
- 记录数据库空间名称 (第 411 页)
- 按正确顺序进行恢复 (第 402 页)
- 指定存档设备 (第 384 页)
- RESTORE 语句 (第 398 页)

备份与符号链接 (仅适用于 UNIX)

在涉及符号链接的备份中, Sybase IQ 可能会在所需目录以外的目录中创建数据库空间。

例如, 假定在以下文件中创建数据库空间:

```
-rw-r--r-- 1 fiona sybase 122880000 Feb 26 18:27 iqdemo.db -rw-  
r--r-- 1 fiona sybase 122880000 Feb 26 18:27 iqdemo.iq1 -rw-r--  
r-- 1 fiona sybase 122880000 Feb 26 18:27 iqdemo.iq2 -rw-r--r--  
1 fiona sybase 122880000 Feb 26 18:27 iqdemo.iq3 -rw-r--r-- 1  
fiona sybase 122880000 Feb 26 18:27 iqdemo.iqtmp -rw-r--r-- 1  
fiona sybase 122880000 Feb 26 18:27 iqdemo.iqmsg
```

如果先创建以下链接, 则会在这些链接所指的目录 (或原始分区) 中创建数据库空间:

```
lrwxrwxrwx 1 fiona sybase 14 Feb 26 17:48 iqdemo.iq1 ->  
LINKS/iqdemo.iq1  
  
lrwxrwxrwx 1 fiona sybase 14 Feb 26 17:48 iqdemo.iq2 ->  
LINKS/iqdemo.iq2  
  
lrwxrwxrwx 1 fiona sybase 18 Feb 26 17:48 iqdemo.iq3 -> /  
dev/rdisk/c2t6d0s0
```

当您备份文件并使用 **CATALOG ONLY** 选项对其进行恢复时, 不会显示任何提示您这些文件是链接的信息; 事实上, 程序不保存此信息。

Sybase IQ 保存这些文件的方式就如同它们实际存在于符号链接所在的目录中。当您进行恢复时, 将在以数据库名称命名的目录或原始分区中重新创建这些文件。无论恢复时链接是否存在, 都不会再使用它们。数据库将恢复至其初始位置。

获取备份和恢复的相关信息

Sybase IQ 提供一个备份日志 `.backup.syb`，用于帮助您管理备份介质。

该日志并非用于创建备份或恢复数据库；但在备份和恢复过程中，程序会将备份或恢复的说明信息记录在该文件中。

注意： 要仅显示有关某个特定备份的信息，可以运行带有 **CATALOG ONLY** 选项的 **RESTORE**。此选项显示来自介质（而不是文件）的备份的标头文件，使 DBA 可以确定磁带或文件中的内容。

另请参见

- 分配备份数据（第 378 页）
- 按正确顺序进行恢复（第 402 页）
- 备份日志的内容（第 409 页）
- 使用 CATALOG ONLY 选项显示标头信息（第 401 页）
- 显示标头信息（第 405 页）

定位备份日志

`.backup.syb` 文件采用 ASCII 文本格式。

该文件的位置取决于启动服务器时环境变量的设置：

- 在 UNIX 中，服务器会尝试按以下顺序将其置于下列位置：
 - 由 `IQLOGDIR15` 环境变量指定的目录。
 - 由 `HOME` 环境变量指定的目录。
 - 从帐户信息中获取的主目录。
 - 当前目录（启动服务器的位置）。

如果文件位于主目录中，它将带有“.”前缀以成为隐藏文件。如果文件位于当前目录中，则没有前缀。

- 在 Windows 中，服务器会尝试按以下顺序将其置于下列位置：
 - 由 `IQLOGDIR15` 环境变量指定的目录。
 - 包含服务器可执行文件的目录。

备份日志的内容

对于您执行的每个备份或恢复，备份日志都包含一个逗号分隔的字段列表。

备份日志将提供：

- 操作（备份或恢复）
- 版本

- 数据库名称
- 数据库类型 (Sybase IQ 或 SQL Anywhere)
- 备份或恢复的日期和时间
- 创建者的用户 ID
- 备份/恢复的类型: 完全备份、增量备份或基于完全备份的增量备份, 或 Database File Only (仅用于 SQL Anywhere 数据库)
- 方法: Archive (用于 IQ 或 Anywhere 数据库) 或 Image (仅用于 Anywhere 数据库)
- 位置
- 用单引号引起来的注释 (前提是已在 **BACKUP** 命令中输入)。如果注释包括引号, 它们将显示为两个连续的单引号。

以下是样本备份日志。

```
BACKUP, 2.0, all_types.db, ASIQ, '2009-01-31 16:25:00.000', DBA, Full, Arch, TED_FULL00, ''
```

```
BACKUP, 2.0, all_types.db, ASIQ, '2009-01-31 16:53:00.000', DBA, Incr, Arch, TED_X_bkup_inc, ''
```

```
RESTORE, 2.0, all_types.db, ASIQ, '2009-01-31 16:25:00.000', DBA, Full, Arch, TED_FULL00, ''
```

```
RESTORE, 2.0, all_types.db, ASIQ, '2009-01-31 16:53:00.000', DBA, Incr, Arch, TED_X_bkup_inc, ''
```

```
BACKUP, 2.0, all_types.db, ASIQ, '2009-01-31 20:07:00.000', DBA, InSF, Arch, A_partial2_yes_sf, ''
```

```
BACKUP, 2.0, all_types.db, ASIQ, '2009-01-31 20:07:00.000', DBA, InSF, Arch, A_partial2_yes_sf, ''
```

另请参见

- 使用 **CATALOG ONLY** 选项显示标头信息 (第 401 页)
- 显示标头信息 (第 405 页)
- 获取备份和恢复的相关信息 (第 409 页)

维护备份日志

在清空备份介质后清理备份日志是很好的做法。可以使用文本编辑器进行此操作。

编辑时应小心谨慎: **BACKUP** 或 **RESTORE** 在此文件中记录信息后, 便不会检查其精确性。

每个服务器上只有一个备份日志。服务器必须能够对该文件进行读取和写入。系统管理员可能需要限制其他用户对该文件的访问权限。如果在同一系统中运行多个数据库服务器, 则应针对每个服务器设置不同的 **IQLOGDIR15** 环境变量, 以生成单独的备份日志。

警告! 不要在备份或恢复进行过程中编辑备份日志。如果在 **BACKUP** 或 **RESTORE** 写入文件时对文件进行修改, 则会使该文件中的信息失效。

记录数据库空间名称

为应对可能需要使用 **RESTORE** 的 **RENAME** 选项移动数据库或它的某个数据库空间的情况，您需要了解数据库中每个数据库空间的名称。

数据库空间名位于每个数据库的 **SYSFILE** 表中，但您在恢复时无法使用此表。对第一个备份存档文件路径运行 **db_backupheader** 可以查看此信息。或者，也可以在备份数据库时执行 **sp_iqdbspace** 和 **sp_iqfile** 存储过程或发出以下语句：

```
SELECT dbf.dbfile_name, f.* FROM SYSFILE f, SYSDBFILE dbf WHERE
f.file_id=dbf.dbfile_id
```

请将此查询的结果保存在数据库所在磁盘以外的位置，这样当您需要数据库空间名时可获得其完整列表。

此外，也可以在 **Interactive SQL** 中运行以下脚本。如果不实际更改任何文件的位置，此脚本将生成一个输出文件，其中包含您所用的重命名子句的集合。您可以替换任何新的文件位置，并使用 **RESTORE** 语句中产生的文件。

注意： 由于数据库在您需要恢复时可能不存在，因此您可能需要在备份数据库之后运行此脚本。

```
-- Get dbspace and IQ file names and add
-- rename syntax including quotation marks

select 'rename' as 'restore ... rename' ,
dbf.dbfile_name as 'IQ file' , 'to' as 'to' ,
'' + f.file_name + '' as 'file_path'
from SYSFILE f, SYSDBFILE dbf
where f.store_type=2 and f.file_id=dbf.dbfile_id

-- Send output to a file in proper format
-- without delimiters or extra quotation marks

output to restore.tst delimited by ' ' quote '';

-- This produces a restore.tst file like the following:
-- rename IQ_SYSTEM_MAIN to '/dev/rdisk/c2t0d1s7'
-- rename IQ_SYSTEM_TEMP to '/dev/rdisk/c2t1d1s7'
-- rename IQ_SYSTEM_MSG to 'all_types.iqmsg'
```

另请参见

- 使用非 Sybase 产品执行备份（第 389 页）
- 按正确顺序进行恢复（第 402 页）
- 指定存档设备（第 384 页）
- **RESTORE** 语句（第 398 页）
- 检验数据库备份（第 406 页）

确定数据备份和恢复策略

为开发用于备份系统的有效策略，需要确定适用于您站点的完全备份、增量备份与基于全备份的增量备份的最佳组合，然后设置执行备份的日程表。

应考虑各种备份选项对性能的影响，以及对您在数据库发生故障时进行快速恢复的能力的影响。

调度例行备份

创建每个数据库后应立即对其进行完全备份以提供基准点，之后应定期执行完全备份和增量备份。尤其重要的是在每次大量更改后备份数据库。

备份计划取决于以下因素：

- 系统的负载
- 数据库的大小
- 对数据的更改次数
- 较快备份和较快恢复的相对重要性

另请参见

- 指定备份类型（第 383 页）

确定备份的类型

在决定进行完全备份、增量备份还是基于全备份的增量备份时，需要在创建备份所用的时间与恢复所用时间之间取得平衡。

此外，还应考虑介质要求。相对而言，指定的增量备份速度较快，在磁带或磁盘中占用的空间也比较小。完全备份相对较慢，且需要大量空间。

基于全备份的增量备份介于两者之间。它在开始时与增量备份相同，但当数据库发生更改、备份数量自上次完全备份后有所增加时，基于全备份的增量备份会变得与完全备份一样耗费时间和介质，甚至比完全备份消耗更大。

一般来讲，对于恢复操作，情况正相反。例如，如果您需要从某个非常旧的完全备份和十几个增量备份进行恢复，则与进行新的完全备份相比，此恢复可能花费更长时间，而且备份占用的空间可能更多。

增量备份的明显优势在于，与备份整个数据库相比，其速度快得多，而且只备份自上次备份甚至上次完全备份后已更改的数据，因此占用的空间较少。过于依赖增量备份的缺点是每次最终恢复将花费更长时间。

例如，您对数据库进行了一次完全备份，理论上在这之后仅执行增量备份即可。但您不会希望如此操作，因为将来每次恢复将慢得难以接受，而且与定期进行完全备份相比，所需的磁带或磁盘空间更多。请记住，当您进行备份时，其他用户可以进行读写访问，但在您进行恢复时，任何其他用户都无法使用该数据库。您可能会发现自己需要恢复数十个增量备份，在此过程中用户将无法使用系统。

与此相比，混合增量备份与完全备份的方法要好得多。

数据库的更改量越大，进行备份的重要性就越高，增量备份的优势也越小。例如，如果您每晚更新数据库，且所做更改影响 10% 或更多数据，则可能需要每晚进行一次基于全备份的增量备份，每周进行一次完全备份。另一方面，如果更改趋于减少，则只需每月进行一次完全备份并在其间进行若干增量备份即可。

指定备份和恢复责任

许多组织设有负责执行所有备份和恢复操作的操作员。任何负责备份或恢复 Sybase IQ 数据库的人员必须拥有该数据库的 DBA 特权。

提高备份和恢复的性能

完成备份或恢复数据库所用的总时间在很大程度上取决于您针对混合完全恢复与增量恢复选择的策略。

还有另外几个因素会影响备份和恢复操作的速度：存档设备的数量、数据检验、可用于备份的内存以及 IQ 存储和 Catalog 存储的大小。

增加存档设备的数量

BACKUP 语句中的 **TO** 子句控制存档设备的数量。

消除数据检验

通过设置 **BACKUP** 命令中的 **CRC OFF**，也可以提高备份和恢复操作的速度。

此设置将失活循环冗余检查。使用 **CRC ON** 时，会在所有后续恢复操作过程中检验备份时计算的数字，因而影响这两个命令的性能。缺省值为 **CRC ON**。如果关闭此检查，请记住您是以数据精确度的更好保证为代价换取更高的性能。

假脱机备份数据

您可能会发现，在磁盘中创建备份然后将其假脱机到磁带中以进行存档存储的方法更加快速有效。如果选择这种方法，则需要在恢复数据之前对其进行取消假脱机。

增加备份过程中占用的内存

备份过程中用于缓冲区的内存量直接影响备份速度，尤其是磁带备份的速度。**BACKUP** 命令的 **BLOCK FACTOR** 参数可控制所用内存量。如果备份较慢，则可能需要增大 **BLOCK FACTOR** 的值以提高备份速度。

BLOCK FACTOR 的作用取决于操作系统，以及创建数据库时指定的块大小。对于新创建的数据库，缺省的 IQ 页大小为 128 KB，因此缺省块大小为 8192 字节。

在 UNIX 中，**BLOCK FACTOR** 的缺省值为 25。Sybase 建议至少将 **BLOCK FACTOR** 设置为 25。采用这种组合，在理想情况下 **BACKUP** 能够为大多数 UNIX 磁带驱动器缓冲数据，内存中的数据足以使驱动器在整个备份过程中经常保持繁忙状态。

在 Windows 中，**BLOCK FACTOR** 的缺省值基于数据库块大小进行计算。此值通常会实现 Windows 的最大吞吐量。由于 Windows 处理磁带设备的方式，您可能无法通过增大 **BLOCK FACTOR** 提高备份的速度。

平衡系统负载

Sybase IQ 允许备份与所有其它读/写操作并发执行，但对数据库结构产生影响的操作除外。

不过，为尽可能实现系统资源（磁盘、内存和 CPU 周期）的最佳利用，将备份安排在系统使用率较低的时间仍然是很好的做法。

控制 Catalog 存储的大小

IQ 数据库由一个 IQ 存储与一个基础 Catalog 存储组成。

BACKUP 在每次备份（完全备份和增量备份）开始时对 Catalog 存储进行完全备份。通常 Catalog 存储是很小的，仅包含系统表、元数据和 Sybase IQ 管理数据库所需的其它信息。但是，也可以在 Catalog 存储中创建非 IQ 表。通过始终将非 IQ 数据放在单独的仅限 SQL Anywhere 的数据库中（而不放在 Catalog 存储中），可以提高 IQ 备份性能。

备份仅复制数据库的最新提交版本。打开事务所用的其它版本页不进行备份。

使用只读硬件存档数据

最近的法规（如《美国健康保险转移和责任法案 (HIPAA)》和《萨班斯-奥克斯利法案》）针对数据保留和合规性指定了严格的规则，要求数据以易于访问的不可变形式进行存档。

数据量可能扩展到数千吉字节的范围，而数据保留周期则从几年到几十年不等。

为满足这些要求，**WORM** 磁盘存储解决方案经历了相应的发展。**WORM**（单写多读）存储开始时作为光盘技术，只允许每个存储位置进行一次永久写入。**WORM** 磁盘阵列在 Sybase IQ 中称为 *只读硬件*。只读硬件功能是通过添加了 **WORM** 保护层的低成本磁盘阵列硬件提供的。保护层在数据“冻结”之前允许对磁盘进行正常的读写使用。

数据冻结时，由用户指定不定或固定的保留周期。可以在卷级或文件级冻结磁盘。数据一旦冻结，便不能对其进行修改，保留周期可以延长，但决不能缩短。

只读硬件功能不仅限于 **WORM** 磁盘阵列硬件；您也可以在数据库空间更改为只读模式后删除原始设备或文件系统文件的写入特权。

另请参见

- 选择存档设备（第 379 页）

使用只读硬件

本节介绍典型情况下的只读硬件操作。

注意：只读硬件功能不需要使用 **WORM** 磁盘阵列硬件。在数据库空间更改为只读模式后，可以删除原始设备或文件系统文件的写入特权。

创建存档

请按照下列步骤创建存档。

假定某个 IQ 数据库包含一个名为 `db.db` 的 Catalog 存储数据库空间，它带有三个主 IQ 数据库空间：A、B 和 C。

1. 在时间 t_0 ，将三个主数据库空间全部更改为只读模式。
2. 将 `db.db` 复制到 `db.db0`，可通过关闭数据库并复制 `db.db` 实现，也可在数据库仍然运行时使用 **dbbackup** 生成副本。
3. 在硬件级冻结数据库空间 A、B 和 C。以不可变形式存储 `db.db0`，可采用将其存储在 **WORM** 设备上的文件系统文件中并对其进行冻结的方法。

此时，数据库已采用不可变形式完成截至时间 t_0 的存档。

另请参见

- 创建更多存档（第 417 页）

创建新数据库空间

请按照下列步骤创建新数据库空间。

1. 创建两个新的主数据库空间 D 和 E。
2. 继续使用数据库 `db.db` 作为生产数据库。

截至时间 t_0 存在的数据库对象（表、索引等）可能已更改，使 `db.db` 不同于 `db.db0`。只要在时间 t_0 存在的表仍然存在，并且其中包含截至 t_0 存在的数据的某些未修改行，`db.db` 就会继续从数据库空间 A、B 和 C 中读取数据。即使上述条件不再成立，`db.db` 仍会继续打开 A、B 和 C，除非将它们从 `db.db` 中删除，只有从 `db.db` 的角度来讲这些数据库空间为空时，才允许进行此删除。

检查存档数据

请按照下列步骤检查存档数据。

前提条件

假定您需要检查截至时间 t_0 的存档数据库。

过程

1. 将存档的只读 `db.db0` 复制到读写文件 `db.db0.working`。
2. 启动 `db.db0.working`。请注意，只要服务器名 `db.db0.working` 与生产系统 `db.db` 不存在冲突，就不需要停止生产系统。`db.db0.working` 将以只读模式打开 A、B、C 和 D。在 UNIX 中，此操作不会干扰 `db.db` 对这些文件的使用，但 Windows 会返回共享冲突。

请注意，Catalog 文件 `db.db0.working` 是以读写模式打开的。

3. 创建用户 `inv`，代表希望检查存档数据库的调查人员。
4. 为其赋予 `inv RESOURCE` 权限以创建视图、存储过程、全局或本地临时表，或调查所需的任何其它结构。

`db.db0` 以及 A、B 和 C 将保持不变。

更新可用存档

请按照下列步骤更新可用存档。

前提条件

如果自时间 `t0` 后经过了数年，则只要 **ALTER DATABASE UPGRADE** 未修改 IQ 主存储区中的任何对象，便可以对 `db.db0.working` 进行升级。

过程

1. 启动 `db.db0.working` 时无需使用截至时间 `t0` 存在的临时数据库空间。使用服务器启动开关 **-iqnotemp** 启动 `db.db0.working`。
2. 删除并创建新的临时数据库空间，或使用通过 **-iqnotemp** 参数创建的临时空间。

创建更多存档

请按照下列步骤创建更多存档。

前提条件

按如下方法在时间 `t1` 创建新存档。

过程

1. 将数据库空间 D 和 E 更改为只读模式。
2. 将 `db.db` 复制到 `db.db1`。
3. 冻结 D 和 E。
4. 以不可变形式保存 `db.db1`。
5. 创建新的主数据库空间，如 F 和 G。

6. 继续使用生产系统 db.db。

只需将 db.db0 和/或 db.db1 复制到可用文件并启动服务器，即可随时使用存档数据库 db.db0 或 db.db1，甚至可以同时使用这两者。

创建存档，然后按以下步骤创建任意数量的 db.db 的存档版本。

创建更多存档

请按照下列步骤创建更多存档。

另请参见

- 创建存档（第 415 页）

系统恢复与数据库修复

了解 Sybase IQ 服务器的常规恢复、特殊的恢复模式、如何检验数据库一致性以及如何修复数据库的不一致问题。

重新启动数据库服务器时，Sybase IQ 会自动尝试恢复。如果服务器无法恢复和重新启动，尤其是在系统出现故障或电源断电之后，数据库可能会出现不一致的情况。

另请参见

- 数据保护 (第 375 页)
- 关闭操作系统会话 (第 37 页)
- 运行 `sp_iqcheckdb` 的资源问题 (第 426 页)
- 系统恢复 (第 355 页)
- 验证数据库 (第 394 页)

恢复和修复概述

如果 Sybase IQ 服务器或数据库出现重新启动的情况，请使用以下信息来诊断数据库启动问题、检验数据库的一致性以及修复数据库。

如果能够在服务器出现故障后重新启动服务器，Sybase 建议您首先检验数据库，然后才能允许用户与数据库连接。可使用 `sp_iqcheckdb` 存储过程来检验数据库。

如果在启动服务器或数据库时遇到问题，或者数据库可启动但用户不能与其连接，或者在检验数据库期间出现问题，则可能需要执行强制恢复、恢复数据库、恢复泄漏的空间或修复索引。

检查服务器日志和 IQ 消息日志

要确定需执行哪种类型的恢复或修复，需要参见服务器日志 (`servername.nnnn.srvlog`) 和 IQ 消息日志 (`dbname.iqmsg`) 中的信息。一定要保留这些信息，以便在需要时可以将它们提供给 Sybase 技术支持部门。

例如，如果检测到数据不一致问题，则 `dbname.iqmsg` 文件中可能会包含详细的诊断信息。

常规恢复

在系统恢复期间，将回退所有未提交的事务，用于旧版本（未提交的事务所使用的数据库页的快照）的所有磁盘空间将返回至可用空间缓冲池。

在常规恢复后，数据库则仅包含各个永久表的最近提交的版本，除非它是 Multiplex 数据库。Multiplex 数据库包含辅助服务器可以访问的所有版本。

从系统故障或常规的系统关闭进行恢复期间，Sybase IQ 可重新打开之前处于活动状态的所有连接。如果用于设置用户连接数的 **-gm** 选项在发生故障时仍有效，则至少需要以与服务器停止时实际使用的连接数相等的连接数来重新启动 IQ 服务器。

数据库检验

使用 **sp_iqcheckdb** 进行数据库检验。

在服务器发生异常终止（如电源故障）又重新启动后，并在执行数据库备份之前，应尽可能快地对数据库进行一致性检查。

您可以使用 **sp_iqcheckdb** 存储过程来检测和修复数据库一致性方面的问题。

另请参见

- 分配问题分析（第 431 页）
- 数据库连接问题（第 452 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- 服务器操作问题（第 444 页）
- **sp_iqcheckdb** 存储过程（第 478 页）
- 以强制恢复模式启动服务器（第 435 页）
- 恢复泄漏的空间（第 436 页）
- 服务器恢复和数据库修复的决策流程（第 443 页）

sp_iqcheckdb 存储过程

IQ 数据库一致性检查程序 (DBCC) 执行数据库检验。与服务器启动选项结合使用的 **sp_iqcheckdb** 存储过程是 DBCC 的接口。

可通过指定 **sp_iqcheckdb** 命令字符串选择不同的检查和修复模式。除非在命令字符串中另有指定，否则 **sp_iqcheckdb** 将读取每个数据库页面并检查数据库的一致性。

注意： 在辅助服务器上，**sp_iqcheckdb** 不会检查空闲列表。但它会执行所有其它检查。

DBCC 有三种按一定的增量来执行一致性检查的模式，其中一种模式用于重置分配映射。除个别数据库空间、表、分区、索引或 **sp_iqcheckdb** 命令字符串中指定的索引类型之外，每种模式都检查所有数据库对象。如果指定个别表名，还将检查这些表中的所有索引。

注意： **sp_iqcheckdb** 存储过程不会检查参照完整性，也不会修复参照完整性违规。

有关 **sp_iqcheckdb** 的完整语法，请参见《参考：构件块、表和过程》的“系统过程”中的“**sp_iqcheckdb** 过程”。

DBCC 性能

DBCC 的执行时间根据进行全面数据库检查的数据库的大小、指定的表或索引的数目以及计算机性能的高低而有所不同。如果仅检查数据库的一个子集，例如仅检查指定的表、索引或索引类型，则所需时间比检查整个数据库要少。有关 **sp_iqcheckdb** 模式的处理时间，请参见《参考：构件块、表和过程》>“系统过程”中的“表 7-12”。

要想获得最佳的 DBCC 性能，请在 **sp_iqcheckdb** 命令字符串中尽可能具体地说明目的。如有可能，请使用“分配”或“检查”检验模式，如果您确切地知道哪个数据库对象需要检查，则可以指定表或索引的名称。

sp_iqcheckdb 检查模式

在检查模式中，**sp_iqcheckdb** 将对所有 IQ 索引执行内部一致性检查，并检查每个数据库块是否得到正确分配。所有可用的数据库统计信息均将报告出来。该模式可读取所有数据页，并能检测所有类型的分配问题和大部分类型的索引不一致问题。对大部分数据库来说，检查模式的运行速度应比检验模式快得多。

何时以检查模式运行：

- 如果在运行查询时返回元数据错误、计数为空错误或离散值个数错误。

检查模式示例：

表 46. sp_iqcheckdb 检查模式示例

命令	描述
<code>sp_iqcheckdb 'check database'</code>	对数据库中的所有表和索引进行内部检查
<code>sp_iqcheckdb 'check table t1'</code>	对表 t1 中的所有索引进行缺省检查
<code>sp_iqcheckdb 'check index t1c1hg'</code>	对索引 t1c1hg 进行内部检查
<code>sp_iqcheckdb 'check indextype FP database'</code>	对数据库中类型为 FP 的所有索引进行检查

sp_iqcheckdb 检验模式

在检验模式中，除了内部索引一致性检查和分配检查之外，**sp_iqcheckdb** 还会执行索引内的一致性检查。所有可用的数据库统计信息均将报告出来。每个非 FP 索引的内容根据其相应的 FP 索引进行检验。检验模式可读取所有数据页，并能检测所有类型的分配问题及所有类型的索引不一致问题。

何时以检验模式运行：

- 如果在运行查询时返回元数据错误、计数为空错误或离散值个数错误

检验模式示例：

表 47. sp_iqcheckdb 检验模式示例

命令	描述
sp_iqcheckdb 'verify database'	检验数据库中所有索引的内容
sp_iqcheckdb 'verify table t1'	检验表 t1 中所有索引的内容
sp_iqcheckdb 'verify index t1c1hg'	检验索引 t1c1hg 的内容
sp_iqcheckdb 'verify indextype HG table t1'	检验表 t1 中所有 HG 索引的内容

注意： 如果在检查模式下检查单个非 FP 索引，则相应的 FP 索引会用内部一致性检查自动进行检查并显示在 DBCC 结果中。

sp_iqcheckdb 分配模式

在分配模式中，**sp_iqcheckdb** 将根据内部物理页映射结构（块映射）来检查每个数据库块是否得到正确分配。与分配有关的数据库统计信息也会报告。该模式执行速度非常快。但是，分配模式不会检查索引一致性，并且也不能检测任一类型的分配问题。

何时以分配模式运行：

- 检查泄漏的块或由多个对象共有的块所造成的不一致的索引
- 强制运行恢复过程之后，以 **dropleaks** 模式运行 **sp_iqcheckdb** 以便重置分配映射（必须使用数据库作为目标）
- 检查重复块或无主块（使用数据库或特定表或索引作为目标）
- 遇到页头错误

分配模式示例：

表 48. sp_iqcheckdb 分配模式示例

命令	描述
sp_iqcheckdb 'allocation database'	对整个数据库进行分配检查
sp_iqcheckdb 'allocation database dumpleaks'	对整个数据库进行分配检查，并将泄漏块的块数显示到 IQ 消息文件
sp_iqcheckdb 'allocation table t1'	对表 t1 进行分配检查
sp_iqcheckdb 'allocation index t1c1hg'	对索引 t1c1hg 进行分配检查
sp_iqcheckdb 'allocation indextype LF table t2'	对表 t2 中的所有 LF 索引进行分配检查

如果表的某些分区处于脱机状态，您可以指定一个分区目标，以只检查表的一部分。可以组合所有模式，并可以在单个会话中对数据库运行多项检查。在下面的示例中，**sp_iqcheckdb** 将对表 **t2** 中的分区 **p1** 执行快速检查，对索引 **i1** 执行详细检查，并使用一半 CPU 对整个数据库执行分配检查：

```
sp_iqcheckdb 'check table t2 partition p1 verify index i1 allocation
database resources 50'
```

分配模式选项只能与 DBCC 命令 “allocation database” 一起使用。

以下分配模式选项可将受影响的数据库块的块数显示到 IQ 消息文件：

- **dumpleaks** - 泄漏块
- **dumpdups** - 重复块
- **dumpunallocs** - 未分配的块

一旦数据库内部版本控制时钟落后，**resetclocks** 将更正这些时钟的值。除非曾咨询过 Sybase IQ 技术支持，否则请勿将 **resetclocks** 选项用于任何其它目的。

resetclocks 选项必须在单用户模式下运行，并且只能与 DBCC 命令 “allocation database” 一起使用。**resetclocks** 命令的语法如下：

```
sp_iqcheckdb 'allocation database resetclocks'
```

sp_iqcheckdb 沙漏模式

当 Sybase IQ 服务器以单节点模式运行时，您可以将沙漏模式与数据库或数据库空间目标结合使用以重置整个数据库或指定数据库空间目标的分配映射。如果目标是 **dbspace**，则沙漏操作还必须阻止对命名 **dbspace** 的读写操作。数据库或 **dbspace** 列表中的所有 **dbspace** 都必须处于联机状态。

在下面的示例中，第一个语句重置整个数据库的分配映射，第二个语句重置数据库空间 **dbsp1** 的分配映射。

```
sp_iqcheckdb 'dropleaks database' sp_iqcheckdb 'dropleaks dbspace
dbsp1'
```

注意： 使用 **sp_iqrebuildindex** 可修复索引错误。目前尚不支持修复连接索引。

sp_iqcheckdb 输出

sp_iqcheckdb 的输出内容包括一大列统计信息及 DBCC 报告的所有错误。

仅显示非零值。含有错误的行都带有星号标记(*****)。请注意，如果遇到错误，DBCC 所报告的部分统计信息可能不准确。

sp_iqcheckdb 的输出始终会复制到 IQ 消息文件 (**.iqmsg**)。要将 **sp_iqcheckdb** 输出重定向到另一个文件，请输入以下命令：

```
sp_iqcheckdb ># file_name
```

其中 **file_name** 是接收输出的文件的名称。

当 **DBCC_LOG_PROGRESS** 选项为 ON 时，**sp_iqcheckdb** 会将进度消息发送至 IQ 消息文件。用户可通过这些消息跟踪 **sp_iqcheckdb** 过程的执行进度。

以下是该命令进度日志输出的示例 **sp_iqcheckdb 'check database'**

```
IQ Utility Check Database Start CHECK STATISTICS table: tloansf Start
CHECK STATISTICS for field: aqsn_dt Start CHECK STATISTICS processing
index: ASIQ_IDX_T444_C1_FP Start CHECK STATISTICS processing index:
tloansf_aqsn_dt_HNG Done CHECK STATISTICS field: aqsn_dt
```

后续版本错误

如果看到“DBCC 后续版本错误”消息，则说明在 DBCC 事务开始后执行了 DDL 操作。DBCC 会继续处理剩下的表，但不会检查泄漏的块，并且统计信息中并不包括被跳过的表。

要避免 DBCC 后续版本错误，请在运行 **sp_iqcheckdb** 之前执行 **COMMIT** 命令。

以下 DBCC 输出指示了后续版本错误：

```
=====|=====|=====
DBCC Verify Mode Report|          |
=====|=====|=====
** DBCC Future Version Errors| 1      |*****
```

有效数据库的输出样本

以下是在检验模式下运行 **sp_iqcheckdb** 的示例。如果未检测到错误，则没有泄漏的空间，数据库分配是一致的，所有索引也是一致的。

此示例的命令是 **sp_iqcheckdb 'verify database'**。请注意，DBCC 检验的是所有索引，但此处显示的索引检验输出是经过删节的。

DBCC 确定一致的每个索引都会在结果集中标记为已检验。

Stat	Value	Flags
===== ===== =====		
DBCC Verify Mode Report		
===== ===== =====		
DBCC Status	No Errors Detected	
DBCC Work units Dispatched	75	
DBCC Work units Completed	75	
===== ===== =====		
Index Summary		
===== ===== =====		
Verified Index Count	86	
===== ===== =====		
Allocation Summary		
===== ===== =====		
Blocks Total	8192	
Blocks in Current Version	4855	
Blocks in All		

Versions	4855	
Blocks in Use	4855	
% Blocks in Use	59	
=====	=====	=====
Allocation Statistics		
=====	=====	=====
DB Extent Count	1	
Blocks Created in Current TXN	211	
Blocks To Drop in Current TXN	212	
Marked Logical Blocks	8240	
Marked Physical Blocks	4855	
Marked Pages	515	
Blocks in Freelist	126422	
Imaginary Blocks	121567	
Highest PBN in Use	5473	
Total Free Blocks	3337	
Usable Free Blocks	3223	
% Total Space Fragmented	1	
% Free Space Fragmented	3	
Max Blocks Per Page	16	
1 Block Page Count	104	
3 Block Page Count	153	
...		
16 Block Hole Count	199	
=====	=====	=====
Index Statistics		
=====	=====	=====
...		
Verified Index	fin_data.DBA.ASIQ_IDX_T209_C3_HG	
Verified Index	fin_data.DBA.ASIQ_IDX_T209_C4_FP	
Verified Index	product.DBA.ASIQ_IDX_T210_C1_FP	
...		
Verified Index	employee.DBA.ASIQ_IDX_T212_C20_FP	
Verified Index	iq_dummy.DBA.ASIQ_IDX_T213_C1_FP	
FP Indexes Checked	68	
HNG Indexes Checked	1	
HG Indexes Checked	17	
=====	=====	=====
...		

DBCC 输出中还包含按标题（例如 Container Statistics、Buffer Manager Statistics、catalog Statistics、Connection Statistics 以及 Compression Statistics 等）进行分组的广泛的统计信息。在连接至 Sybase IQ 演示数据库 iqdemo 后执行命令 `sp_iqcheckdb 'verify database'`，可以查看可用统计信息的示例。

另请参见

- 索引错误分析（第 427 页）
- DBCC 错误消息（第 440 页）

- 索引错误修复 (第 430 页)
- 删除不一致的索引、表或列 (第 439 页)
- 恢复泄漏的空间 (第 436 页)

运行 **sp_iqcheckdb** 的资源问题

sp_iqcheckdb 报告执行时遇到的资源问题。

系统会在 **sp_iqcheckdb** 输出或 `.iqmsg` 文件中报告描述资源问题的消息。

- **Out of memory** 和 **DBCC Out of Memory Errors** 您没有足够的内存来执行此操作。可能需要阻止其它 IQ 操作或其它应用程序与 **sp_iqcheckdb** 存储过程同时运行。
- **No buffers available** 和 **DBCC Out of Buffers Errors** DBA 可能需要增加缓冲区高速缓存大小。

可使用数据库选项 **TEMP_CACHE_MEMORY_MB** 对缓冲区高速缓存大小进行永久性设置。使用服务器启动开关 **-iqmc** 和 **-iqtc** 可以覆盖使用数据库选项设置的缓冲区高速缓存大小的值。

不要同时运行多个数据库一致性检查，因为 **DBCC** 被优化为一次运行一个实例。

DBCC 的 CPU 利用率可通过指定 **sp_iqcheckdb** 参数 **resources resource-percent** 加以限制，该参数控制着与 CPU 数目有关的线程数。*resource-percent* 的缺省值为 100，它为每个 CPU 创建一个线程，应与大多数计算机的负载容量相匹配。如果将 **DBCC** 作为后台进程运行，请将 *resource-percent* 设置为小于 100 的值，以减少线程数。线程数的最小值为 1。

如果 *resource-percent* > 100，则有比 CPU 数更多的线程，对于某些计算机配置，这可能会提高性能。

数据库选项 **DBCC_PINNABLE_CACHE_PERCENT** 可用于调优 **DBCC** 缓冲区使用率。**DBCC_PINNABLE_CACHE_PERCENT** 的缺省设置为使用 50% 的高速缓存。请参见《参考：语句和选项》。

另请参见

- 数据保护 (第 375 页)
- 关闭操作系统会话 (第 37 页)
- 系统恢复 (第 355 页)
- 系统恢复与数据库修复 (第 419 页)
- 验证数据库 (第 394 页)

数据库修复

分配问题可通过以 **dropleaks** 模式运行 **sp_iqcheckdb** 来进行修复。

如果 **DBCC** 在试图修复分配问题时检测到索引不一致问题，则会生成一个错误，并无法修复分配问题。

另请参见

- 分配问题分析 (第 431 页)
- 数据库连接问题 (第 452 页)
- 数据库检验 (第 420 页)
- 索引错误修复 (第 430 页)
- 在执行检查点操作期间用完空间 (第 457 页)
- 服务器操作问题 (第 444 页)
- **sp_iqcheckdb** 存储过程 (第 478 页)
- 以强制恢复模式启动服务器 (第 435 页)
- 恢复泄漏的空间 (第 436 页)
- 服务器恢复和数据库修复的决策流程 (第 443 页)

索引错误分析

使用 **sp_iqcheckdb** 分析索引一致性。

含有不一致的索引的输出示例

以下是运行 **sp_iqcheckdb** 并存在索引不一致问题时您所看到的输出类型的示例。**DBCC** 将显示有关所检查的索引的摘要信息和详细信息。报告顶部的 **Index Summary** 部分指出是否发现不一致的索引。不一致索引的名称和问题类型可在索引统计信息部分找到。带有星号(*****)的行中包含有关不一致索引的信息。

在报告的错误中，**extra**、**missing** 或 **duplicate RID** 错误是最常见的错误类型。这些错误表明索引错误地表示了数据，并可能会给出不正确的结果或导致其它故障。通常，这些错误会伴随有其它指出不一致的具体信息的错误。

在该示例中，**DBCC** 报告了不一致的 **HNG** 索引。因为相应的 **FP** 索引检查结果为良好，所以该 **FP** 索引可与 **sp_iqrebuildindex** 一起用于修复损坏的 **HNG** 索引。

该示例执行的命令是 **sp_iqcheckdb 'verify database'**。请注意，**DBCC** 生成的是一个详细的报告，但在该示例中某些行的输出被删除了。

Stat	Value	Flags
DBCC Verify Mode Report		
** DBCC Status	Errors Detected	*****
DBCC Work units		

Dispatched	75	
DBCC Work units Completed	75	
=====	=====	=====
Index Summary		
=====	=====	=====
** Inconsistent Index		
Count	1	*****
Verified Index		
Count	85	
=====	=====	=====
Index Statistics		
=====	=====	=====
** Inconsistent Index	contact.DBA.idx01_HNG	*****
...		
Verified Index	fin_data.DBA.ASIQ_IDX_T209_C3_HG	
Verified Index	fin_data.DBA.ASIQ_IDX_T209_C4_FP	
...		
Verified Index	employee.DBA.ASIQ_IDX_T212_C19_FP	
Verified Index	employee.DBA.ASIQ_IDX_T212_C20_FP	
Verified Index	iq_dummy.DBA.ASIQ_IDX_T213_C1_FP	
** Extra Index RIDs	5	*****
FP Indexes Checked	68	
HNG Indexes Checked	1	
HG Indexes Checked	17	

sp_iqcheckdb 检测到的不一致的索引是 `contact.DBA.idx01_HNG`。

当再次运行 **sp_iqcheckdb** 仅检查不一致的索引时，会生成以下 DBCC 输出。该示例执行的命令行是 `sp_iqcheckdb 'verify index DBA.contact.idx01_HNG'`。

Stat	Value	Flags
=====	=====	=====
DBCC Verify Mode Report		
=====	=====	=====
** DBCC Status	Errors Detected	*****
DBCC Work units Dispatched	1	
DBCC Work units Completed	1	
=====	=====	=====
Index Summary		
=====	=====	=====
** Inconsistent Index		
Count	1	*****
Verified Index		
Count	1	
=====	=====	=====
Index Statistics		
=====	=====	=====
** Inconsistent Index	contact.DBA.idx01_HNG	*****

Verified Index	contact.DBA.ASIQ_IDX_T206_C1_FP	
** Extra Index RIDs	5	*****
FP Indexes Checked	1	
HNG Indexes Checked	1	
=====	=====	=====

DBCC 索引错误

DBCC 输出包含与索引问题有关的消息。

表 49. DBCC 索引错误

DBCC 消息	说明/操作
Inconsistent Index Count	DBCC 找到的不一致索引数。
Inconsistent Index	DBCC 找到的不一致索引名。
Extra Index RIDs Missing Index RIDs Duplicate Index RIDs	对于所有不一致的索引出现不一致的总行数。
Bitmap Verify Errors	所有数据库对象中不一致的位图的总数。
FP Lookup Table Inconsistencies	无法修复的错误，其中单字节或双字节 FP 内部不一致。
Non-Completed Index Count	因为检查时发生异常导致无法检验的索引的数目。
Non-Completed Index	因为检查时发生异常导致未检验的索引的名称。如果异常为版本过高、内存不足或缓冲区不足错误，请提交 DBCC 连接并重新运行 DBCC。
VDO Incorrect First Available Fields VDO Incorrect Next Available Fields VDO Incorrect Used Count Fields VDO Incorrect In-use Bitvec VDO Incorrect In-use Bitmap VDO Incorrect Partial Bitmap VDO Incorrect Deleted Bitmaps	无法修复的错误，可能导致整个表无法访问。您必须强制删除不一致的表以解决这些错误。

DBCC 消息	说明/操作
HG Missing Groups HG Extra Groups HG Extra Keys HG Missing Keys B-Tree Invalid Item Count B-Tree Invalid Item Count G-Array Empty Page Errors G-Array Bad Group Type Errors G-Array Out of Order Group Errors	HG 索引特定的错误。

另请参见

- DBCC 错误消息 (第 440 页)
- 索引错误修复 (第 430 页)
- sp_iqcheckdb 输出 (第 423 页)
- 删除不一致的索引、表或列 (第 439 页)
- 恢复泄漏的空间 (第 436 页)
- DBCC 无法修复的索引问题 (第 439 页)

索引错误修复

使用 **sp_iqrebuildindex** 过程修复索引，然后以检验模式运行 **sp_iqcheckdb** 以检查不一致。

如果某个索引仍不一致，请删除该索引并重新创建该索引，然后再重建该索引。

注意： **sp_iqrebuildindex** 过程不能修复 FP 索引。Sybase IQ 没有修复 FP 索引的功能。

另请参见

- 索引错误分析 (第 427 页)
- DBCC 错误消息 (第 440 页)
- sp_iqcheckdb 输出 (第 423 页)
- 删除不一致的索引、表或列 (第 439 页)
- 恢复泄漏的空间 (第 436 页)
- 分配问题分析 (第 431 页)
- 数据库连接问题 (第 452 页)
- 数据库检验 (第 420 页)

- 数据库修复 (第 427 页)
- 在执行检查点操作期间用完空间 (第 457 页)
- 服务器操作问题 (第 444 页)
- `sp_iqcheckdb` 存储过程 (第 478 页)
- 以强制恢复模式启动服务器 (第 435 页)
- 服务器恢复和数据库修复的决策流程 (第 443 页)

分配问题分析

使用 `sp_iqcheckdb` 分析分配问题。

数据库维护了一个分配映射 (也称作空闲列表), 用于跟踪数据库对象正在使用的块。

DBCC 共检测三种类型的分配问题:

- 泄漏的块 - 泄漏的块是指如下块: 已根据数据库分配映射进行分配, 但却未成为任何数据库对象的一部分。DBCC 可以恢复泄漏的块。
- 未分配的块 - 未分配的块是指如下块: 未根据数据库分配映射进行分配, 但正由某个数据库对象使用的块。DBCC 可以恢复未分配的块。
- 由多个对象共有的块 - 由多个对象共有的块是指正由多个数据库对象使用的块。所涉及的结构当中, 至少有一个含有不一致的数据。DBCC 无法修复这种类型的分配问题。如果遇到这种类型的错误, 请再次运行 DBCC, 指定一个索引列表, 直到标识了共享该块的索引为止。然后, 必须将这些索引全部删除, 以消除由多个对象共有的块。

泄漏的空间输出的示例

下面是当您运行 `sp_iqcheckdb` 并且有泄漏空间时所看到的输出示例。带有星号 (*****) 的行中包含有关分配问题的信息。在此示例中, DBCC 报告了 16 处泄漏块。

该示例执行的命令行是 `sp_iqcheckdb 'allocation database'`。

Stat	Value	Flags
=====		
DBCC Allocation Mode Report		
=====		
** DBCC Status	Errors Detected	*****
DBCC Work units Dispatched	164	
DBCC Work units Completed	164	
=====		
Allocation Summary		
=====		
Blocks Total	8192	
Blocks in Current Version	4785	
Blocks in All Versions	4785	
Blocks in Use	4801	
% Blocks in Use	58	
** Blocks Leaked	16	*****

```

=====|=====|=====
Allocation Statistics|          |=====
=====|=====|=====
...
** 1st Unowned PBN   | 1994   | *****
...
=====|=====|=====
    
```

如果一个或多个数据库空间处于脱机状态，请使用以下语法显示特定数据库空间的分配问题：

```
sp_iqcheckdb 'allocation dbspace dbspace-name'
```

DBCC 分配错误

分配问题在 DBCC 以分配模式或检验模式运行 **sp_iqcheckdb** 时所生成的输出中报告。如果 Allocation Summary 部分有带星号标记的值（例如 “** Blocks Leaked” 或 “** Blocks with Multiple Owners”），则说明存在分配问题。

DBCC 输出中与分配问题有关的消息都在下表中列出。

表 50. DBCC 分配错误

DBCC 消息	说明/操作
Block Count Mismatch	该计数总是伴随着其它分配错误。
Blocks Leaked 1st Unowned PBN	发现未由任何数据库对象正在使用的块。使用 sp_iqcheckdb dropleaks 模式进行修复。
Blocks with Multiple Owners 1st Multiple Owner PBN	正由多个数据库对象使用的块。删除报告为不一致的对象。
Unallocated Blocks in Use 1st Unallocated PBN	正由数据库对象使用但未标记为“正在使用”的块。使用 sp_iqcheckdb dropleaks 模式进行修复。

如果 Allocation Summary 行未指出任何问题，但 Index Summary 部分为 “Inconsistent Index Count” 报告了一个值，则说明存在一个或多个不一致的索引。

另请参见

- 数据库连接问题（第 452 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- 服务器操作问题（第 444 页）
- sp_iqcheckdb 存储过程（第 478 页）
- 以强制恢复模式启动服务器（第 435 页）

- 恢复泄漏的空间（第 436 页）
- 服务器恢复和数据库修复的决策流程（第 443 页）

使用 DBCC 修复分配问题

使用 `sp_iqcheckdb dropleaks` 修复数据库分配问题。

注意：此过程使用 `-gd` 和 `-gm` 开关限制数据库的访问。有关限制性更强的方法，请参见“系统恢复和数据库修复” > “数据库修复” > “强制恢复模式” > “以强制恢复模式启动服务器”。

1. 启动服务器。

例如：

```
start_iq -n my_db_server -x 'tcpip{port=7934}' -gd dba -gm 1 /
work/database/my_db.db
```

注意：必须以“.db”扩展名（而不是“.DB”）启动数据库。

Sybase 建议使用两个服务器启动开关限制访问：

- 使用 `-gd DBA` 可以使只有拥有 DBA 权限的用户才能启动和停止数据库。（请注意，客户端必须已连接到服务器以便启动或停止数据库，因此此开关不阻止连接。）
- 使用 `-gm 1` 可以按照超出限值的方式启用一个连接和一个 DBA 连接，以使 DBA 可在紧急情况下采用其中一个连接并中断另一个连接。

有关限制连接的详细信息，请参见“以强制恢复模式启动服务器”。

2. 以沙漏模式运行存储过程 `sp_iqcheckdb`：

```
sp_iqcheckdb 'dropleaks database'
```

如果一个或多个数据库空间处于脱机状态，则只需运行下面的命令来修复数据库空间的分配问题：

```
sp_iqcheckdb 'dropleaks dbspace dbspace-name'
```

如果分配修复成功，`sp_iqcheckdb` 会显示消息“空闲列表已更新。”如果检测到错误，`sp_iqcheckdb` 会返回消息“空闲列表未更新”或“检测到错误”。

3. `sp_iqcheckdb` 完成后，请停止服务器。要停止服务器，请在 UNIX 中使用 `stop_iq`，或在 Windows 中使用主控台窗口中的关闭按钮。

分配问题得到修复后，分配统计信息将出现在 DBCC 输出中，其中将不再有错误信息。

DBCC 在报告的顶部显示了 Allocation Summary 部分，其中列出了有关分配使用情况的信息。Allocation Statistics 部分提供了有关块的详细信息。DBCC 输出中不包含已恢复的泄漏的块的修复信息。

例如：

```
sp_iqcheckdb 'dropleaks dbspace mydbspace'; checkpoint;
```

sp_iqcheckdb 输出指示没有任何错误，因此执行了 **checkpoint**。

DBCC 会报告未在该简短输出中显示的统计信息。

Stat	Value	Flags
=====		
DBCC Allocation Mode Report		
=====		
DBCC Status	Freelist Updated	
DBCC Status	No Errors Detected	
DBCC Work units Dispatched	75	
DBCC Work units Completed	75	
=====		
Allocation Summary		
=====		
Blocks Total	8192	
Blocks in Current Version	4594	
Blocks in All Versions	4594	
Blocks in Use	4610	
% Blocks in Use	56	
=====		
Allocation Statistics		
=====		
DB Extent Count	1	
Marked Logical Blocks	8176	
Marked Physical Blocks	4594	
Marked Pages	511	
Blocks in Freelist	126177	
Imaginary Blocks	121567	
Highest PBN in Use	5425	
Total Free Blocks	3582	
Usable Free Blocks	3507	
% Free Space Fragmented	2	
Max Blocks Per Page	16	
1 Block Page Count	103	
3 Block Page Count	153	
...		
16 Block Hole Count	213	
=====		
=====		

注意： 执行强制恢复或恢复泄漏的块时，必须以 “.db” 扩展名（而不是 “.DB”）启动数据库。例如：

```
start_iq -n my_db_server -x 'tcpip{port=7934}'
-gd dba my_db /work/database/my_db.db
```


强制恢复模式

强制数据库恢复与常规数据库恢复不同。

- **强制恢复将数据库内的所有存储区标记为“正在使用”** - 为了恢复可能不一致的分配映射，数据库内的所有存储区都标记为“正在使用”。以 **dropleaks** 模式使用 **sp_iqcheckdb** 可将分配映射重置到正确的状态。
- **已禁用增量备份** - 数据库在强制恢复模式下打开后，增量备份即被禁用。下一次备份必须采用完全备份。进行完全备份可重新启用增量备份。
- **如果启动了服务器，则强制恢复参数适用于对数据库执行的所有打开操作** - 因此，在数据库打开之后，DBA 需要关闭服务器，然后再不用强制恢复标志重新启动服务器，以确保后续的打开以常规模式运行。在强制恢复处于启用的状态下重复打开数据库不会损害数据库，但可能会使 DBA 感到困惑。每次以强制恢复模式打开数据库时，该数据库内的所有存储区均标记为“正在使用”。

强制恢复之前

限制数据库访问可以对强制恢复期间无意中打开数据库这一行为进行更好地控制。

Sybase 建议使用两个服务器启动开关限制访问：

- 使用 **-gd DBA** 以便只有具备 DBA 权限的用户才能在正在运行的服务器上启动和停止数据库。（请注意，客户端必须已连接到服务器以便启动或停止数据库，因此此开关不阻止连接。）
- 使用 **-gm 1** 可以按照超出限值的方式启用一个连接和一个 DBA 连接，以使 DBA 可在紧急情况下采用其中一个连接并中断另一个连接。

限制连接的替代方法是：在启动执行强制恢复所用的连接后指定

```
sa_server_option('disable_connections', 'ON')
```

，并且在恢复后对同一连接指定

```
sa_server_option('disable_connections', 'OFF')
```

。其缺点在于，此方法排除了从其它 DBA 连接执行的紧急访问。

以强制恢复模式启动服务器

强制恢复允许服务器在分配映射存在不一致的情况下启动。

如果您无法在 Multiplex 中启动服务器或数据库，则可能需要强制恢复。只有当正常数据库恢复无法将数据库恢复到运行状态并且您在恢复过程中看到 **s_buf** 或空闲列表错误时，才应使用强制恢复。切勿使用强制恢复来响应 SQL Anywhere 错误，例如 SA 事务日志重放故障。

如果您遵循了文档所述的恢复过程，并且 Sybase 技术支持部门建议进行强制恢复，请按照以下步骤操作：

1. 使用 **stop_iq** 关闭所有辅助节点。
2. 使用 **-iqfrec** 和 **-iqmpx_sn 1** 标志启动服务器：

```
start_iq -n my_server -x 'tcpip(port=7934}'  
-gd dba -gm 1 -iqmpx_sn 1 -iqfrec  
my_db /database/my_db.db
```

3. 连接到服务器并运行：

```
sp_iqcheckdb 'dropleaks database'  
checkpoint
```

4. 更正错误并重新运行 **sp_iqcheckdb**。重复以上步骤，直到不产生错误为止。
5. 正常关闭服务器并重新启动（不使用步骤 2 中的标志）。

如果无法在强制恢复模式下启动服务器，请与 Sybase 技术支持部门联系。

使用强制恢复但不后接 **sp_iqcheckdb**

运行强制恢复可在有效但已完全分配的模式下启动数据库。换句话说，您应该可以进行一切操作，但一点永久性 **main** 数据库空间都没剩下。在进行其它任何操作之前，必须要么通过以 **dropleaks** 模式运行 **sp_iqcheckdb** 来恢复丢失的数据库空间，要么添加新的数据库空间。请注意，查询应该也会运行成功，因为它们不需要额外的永久性数据库空间；但不能装载、插入或删除数据。

警告！ 在不检验数据库的情况下运行查询不会使数据出现任何不一致的情况。但是，如果数据中存在导致服务器出现故障的问题，则服务器可能会再次出现故障或生成不正确的结果。

另请参见

- 恢复泄漏的空间（第 436 页）
- 分配问题分析（第 431 页）
- 数据库连接问题（第 452 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- 服务器操作问题（第 444 页）
- **sp_iqcheckdb** 存储过程（第 478 页）

恢复泄漏的空间

以 **dropleaks** 模式使用 **sp_iqcheckdb** 存储过程来恢复指定数据库内泄漏的存储空间。

服务器使用分配映射来确定某个页面是否正在 **IQ** 内使用。无论是通过系统故障还是由于使用强制恢复打开数据库所致，数据库的分配映射可能不会如实反映其使用情况的分配状况。发生这种情况时，我们假设该数据库有“泄漏的”存储区或“泄漏的块”。通常，当出现少量泄漏的块时，您无需太在意。如果泄漏的块达到了数兆字节，则可能需要恢复该空间。

在恢复泄漏的存储区期间，会更改分配映射的其它事务都被排除在外。此类操作包括修改数据库的检查点和命令。

恢复泄漏的存储区与强制恢复二者既可以同时进行，也可以分别进行。要在不进行强制恢复的情况下恢复数据库内泄漏的空间，请使用 **DBCC 修复分配问题**。要在执行强制恢复后恢复数据库内泄漏的空间，请使用以下过程恢复泄漏的空间。

如果使用 **DBCC 修复分配问题** 无法恢复泄漏的存储区，请使用以下过程。

注意： 此过程使用 **-gd** 和 **-gm** 开关限制数据库的访问。要使用限制性更强的方法，请以强制恢复模式启动服务器。

1. 结合 **-iqfrec** 选项使用 **start iq** 命令启动服务器。例如：

```
start iq -n my_db_server -x 'tcpip{port=7934}'
-gd dba -gm 1
-iqfrec my_db /work/database/my_db.db
```

需在同一行中两次指定数据库名称，一次是将其指定为要启动的数据库，一次是指定为要进行强制恢复的数据库。**-iqfrec** 选项要求提供数据库名称。

2. 连接到正在恢复的数据库。
- 3.

以沙漏模式运行存储过程 **sp_iqcheckdb**：

```
sp_iqcheckdb 'dropleaks database'
```

如果没有报错并且 **sp_iqcheckdb** 显示消息“空闲列表已更新”，则说明泄漏的空间已恢复并且是强制执行的恢复。继续下一步操作。

如果发现不一致性，请删除不一致的索引、表或列。然后，再次运行 **sp_iqcheckdb** 恢复泄漏的空间。

4. 发出检查点。
5. 使用常规方法停止服务器。
6. 使用常规方法重新启动服务器，然后继续正常的处理。

另请参见

- 索引错误分析（第 427 页）
- **DBCC 错误消息**（第 440 页）
- 索引错误修复（第 430 页）
- **sp_iqcheckdb** 输出（第 423 页）
- 以强制恢复模式启动服务器（第 435 页）
- 分配问题分析（第 431 页）
- 数据库连接问题（第 452 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 在执行检查点操作期间用完空间（第 457 页）

- 服务器操作问题 (第 444 页)
- sp_iqcheckdb 存储过程 (第 478 页)

恢复 Multiplex 数据库

在解决 Multiplex 数据库相关恢复问题之前，请参见《使用 Sybase IQ Multiplex》。

DBCC 报告的问题

报告 DBCC 无法修复的问题的消息。

表 51. 有关 DBCC 无法修复的问题的消息

DBCC 消息	说明/操作
FP Lookup Table Inconsistencies	无法修复的错误，其中单字节或双字节 FP 内部不一致。
VDO Incorrect First Available Fields VDO Incorrect Next Available Fields VDO Incorrect Used Count Fields VDO Incorrect In-use Bitvec VDO Incorrect In-use Bitmap VDO Incorrect Partial Bitmap VDO Incorrect Deleted Bitmaps	无法修复的错误，可能导致整个表无法访问。您必须强制删除不一致的表以解决这些错误。
Blocks with Multiple Owners 1st Multiple Owner PBN	正由多个数据库对象使用的块。删除报告为不一致的对象。
DBCC Meta-data Errors Blockmap Invalid Chunksize Error Count Blockmap Compression Bit Error Count Blockmap Invalid Block Number Error Count	内部页映射结构不一致且需要删除对象。
DBCC Inconsistent Disk Block Headers DBCC Decompress Errors	对象存储不一致且需要删除对象。

另请参见

- DBCC 错误消息 (第 440 页)

DBCC 无法修复的索引问题

请使用以下建议修复不一致的索引。

如果 DBCC 检测到某个索引有问题，会将该索引名连同问题类型报告出来。使用 **sp_iqrebuildindex** 可修复非 FP 索引。无法修复 FP 索引。当以缺省模式或检查模式运行 **sp_iqcheckdb** 时，分析报告为“不一致索引”的索引的索引错误。

请根据问题类型的不同，使用 **DROP INDEX**、**ALTER TABLE DROP COLUMN**、**DROP TABLE** 或 **FORCE_DROP** 选项来解决问题。

Sybase 建议您致电 Sybase 技术支持部门，请他们帮助确定修复不一致的索引或表的最佳操作过程。

另请参见

- 索引错误分析（第 427 页）

删除不一致的索引、表或列

请使用以下建议来解决无法修复的索引、列或表的相关问题。

如果 **sp_iqcheckdb** 报告无法修复的索引、列或表，则必须分别使用 **DROP INDEX**、**ALTER TABLE DROP COLUMN** 或 **DROP TABLE** 语句来删除这些对象。

注意： 不应试图强制删除对象，除非 Sybase 技术支持部门指示您这样做。

如果无法删除某个不一致的对象，请设置临时的 **FORCE_DROP** 选项。**FORCE_DROP** 会导致 IQ 服务器不发出提示即释放已删除对象的磁盘存储空间，而不是尝试回收它们。您稍后可以使用 **DBCC** 恢复泄漏的空间。这是对不一致的对象的最好处理方法，因为关于对象的存储区的唯一信息存在于该对象本身，并且该信息可能就是不一致的对象。

不允许对辅助节点使用 **FORCE_DROP** 数据库选项。如果试图在辅助节点上强制执行删除操作，将返回错误。**FORCE_DROP** 是一个临时选项，以便选项的值在同步时不会传播到辅助节点。

注意： 在强制删除对象时，必须确保只有 **DBA** 连接到数据库。强制删除之后应立即重新启动服务器。

此过程使用 **-gd** 和 **-gm** 开关限制数据库的访问。**-gd** 开关仅限制能够在运行的服务器上启动或停止数据库的用户。要使用限制性更强的方法，请以强制恢复模式启动服务器。

1. 重新启动服务器。

```
start_iq -n bad_db_server -x 'tcpip{port=7934}'
-gm 1 -gd dba bad_db.db
```

强制删除对象时，不能允许其他用户进行连接。

Sybase 建议使用两个服务器启动开关限制访问:

- 使用 **-gd DBA** 可以使只有拥有 **DBA** 权限的用户才能启动和停止数据库。(请注意, 客户端必须已连接到服务器以便启动或停止数据库, 因此此开关不阻止连接。)
- 使用 **-gm 1** 可以按照超出限值的方式启用一个连接和一个 **DBA** 连接, 以使 **DBA** 可在紧急情况下采用其中一个连接并中断另一个连接。

有关限制连接的详细信息, 请参见《安装和配置指南》。

2. 将 **FORCE_DROP** 临时选项设置为 **ON**。

```
set temporary option FORCE_DROP = 'ON'
```

3. 删除所有不一致的对象。

根据需要使用命令 **DROP INDEX**、**ALTER TABLE DROP COLUMN** 或 **DROP TABLE**。在重新启动服务器之前, 请勿输入任何其它 **DDL** 或 **DML** 命令。

4. 重新启动服务器。

要恢复泄漏的空间并将分配映射更新为正确的状态, 请启动服务器。

```
start_iq -n bad_db_server -x 'tcpip{port=7934}'  
-gm 1 -gd dba bad_db.db
```

5. 运行 **sp_iqcheckdb**。

```
sp_iqcheckdb 'dropleaks database';
```

此步骤将数据库分配映射重新设置为适当的分配映射。

另请参见

- 索引错误分析 (第 427 页)
- **DBCC** 错误消息 (第 440 页)
- 索引错误修复 (第 430 页)
- **sp_iqcheckdb** 输出 (第 423 页)

DBCC 错误消息

以下是 **DBCC** 输出中最重要的消息。

表 52. **DBCC** 错误消息

DBCC 消息	说明/操作
Inconsistent Index Count	DBCC 找到的不一致索引数。
Inconsistent Index	DBCC 找到的不一致索引名。
Extra Index RIDs Missing Index RIDs Duplicate Index RIDs	对于所有不一致的索引出现不一致的总行数。

DBCC 消息	说明/操作
Bitmap Verify Errors	所有数据库对象中不一致的位图的总数。
FP Lookup Table Inconsistencies	无法修复的错误，其中单字节或双字节 FP 内部不一致。
Non-Completed Index Count	因为检查时发生异常导致无法检验的索引的数目。
Non-Completed Index	因为检查时发生异常导致未检验的索引的名称。如果异常为版本过高、内存不足或缓冲区不足错误，请提交 DBCC 连接并重新运行 DBCC。
HG Missing Groups HG Extra Groups HG Extra Keys HG Missing Keys B- Tree Invalid Item Count B-Tree Invalid Item Count G-Array Empty Page Errors G-Array Bad Group Type Errors G-Ar- ray Out of Order Group Errors	HG 索引特定的错误。
VDO Incorrect First Available Fields VDO Incorrect Next Available Fields VDO Incorrect Used Count Fields VDO Incorrect In-use Bitvec VDO Incorrect In-use Bitmap VDO Incorrect Partial Bitmap VDO Incorrect Deleted Bitmaps	无法修复的错误，可能导致整个表无法访问。您必须强制删除不一致的表以解决这些错误。
Block Count Mismatch	该计数伴随其它分配错误。
Blocks Leaked 1st Unowned PBN	发现未由任何数据库对象正在使用的块。使用沙漏模式进行修复。
Blocks with Multiple Owners 1st Mul- tiple Owner PBN	正由多个数据库对象使用的块。删除报告为不一致的对象。
Unallocated Blocks in Use 1st Unallo- cated PBN	正由数据库对象使用但未标记为“正在使用”的块。使用沙漏模式进行修复。
Freelist Updated	表示成功修复分配。
Freelist Not Updated	表示分配修复期间检测到错误，并且分配修复未成功。
Invalid Blockmap Unique ID Generator Blockmap Unique ID Generator Upda- ted Invalid Transaction ID Counter Transaction ID Generator Updated	特定于 DBCC resetclocks 选项的错误和修复消息。
DBCC Future Version Errors	DBCC 无法打开该表，因为在其上执行过 DDL。请提交 DBCC，连接然后重新运行 DBCC。
DBCC Locked Table Access Conflict	DBCC 尝试打开其它连接已锁定的表。为了确保完成 DBCC 处理，请确保其他用户没有锁定数据库中的表。
DBCC Out of Buffers Errors	IQ main 高速缓存过小。请增大 main 高速缓存大小，或者对单个对象运行 DBCC。

DBCC 消息	说明/操作
DBCC Out of Memory Errors	系统内容不足，无法完成 DBCC 操作。
DBCC Meta-data Errors Blockmap Invalid Chunksize Error Count Blockmap Compression Bit Error Count Blockmap Invalid Block Number Error Count	内部页映射结构不一致且需要删除对象。
DBCC Page Read Errors	尝试读取对象时发生 I/O 错误。请执行硬件诊断程序。
DBCC Inconsistent Disk Block Headers DBCC Decompress Errors	对象存储不一致且需要删除对象。
DBCC Unknown Exceptions	发生 DBCC 未知的类型的异常。请检查 IQ 消息文件了解详细信息。
Unowned LVC cells Duplicate LVC cell rows Unallocated LVC cell rows	<p>这些消息表明与 VARCHAR 或 CLOB 列不一致。无主 LVC 单元表示少量的不可用磁盘空间，可安全地忽略。Duplicate 和 Unallocated LVC cells 为严重的错误，只能通过删除已损坏的列解决。</p> <p>要删除已损坏的列，请使用旧列的副本创建一个新列，然后删除原始列并将新列重命名为旧列。</p> <p>LVC 是一个宽度大于 255 个字符的 VARCHAR 列。CLOB 也使用 LVC。</p>

另请参见

- 索引错误分析 (第 427 页)
- 索引错误修复 (第 430 页)
- sp_iqcheckdb 输出 (第 423 页)
- 删除不一致的索引、表或列 (第 439 页)
- 恢复泄漏的空间 (第 436 页)
- DBCC 报告的问题 (第 438 页)

故障排除提示

Sybase IQ 提供用以解决遇到的任何问题的许多资源。

有关解决专门针对 Sybase IQ Multiplex 服务器相关的问题的信息，请参见《使用 Sybase IQ Multiplex》。

在线支持来源

如果无法使用文档解决问题，请参见 Sybase 在线支持站点 MySybase。

使用 MySybase 可以通过针对需求自定义的视图，搜索已完成的支持案例、软件公告和已解决的问题和已知问题。您甚至可以在线打开技术支持个案。

可以在大多数 Internet 浏览器中使用 MySybase。有关如何注册和使用此免费服务的信息，请将 Web 浏览器指向 MySybase。有关其它有用的 Sybase Web 站点，请参见《发行公告》。

特定情况的解决方案

您可能需要更多信息来诊断和解决特定的问题。

您可以使用诊断工具来诊断各种情况，包括在以下几节中描述的情况。

服务器恢复和数据库修复的决策流程

如果数据库可启动但用户不能与其连接，或者在检验数据库期间出现问题，则在启动服务器或数据库时可能会遇到问题。

1. 服务器是否已启动？

如果服务器已启动，则转到步骤 2。

如果服务器未启动，请参见“服务器操作问题”一节。如果在按照本节中的建议操作之后仍无法启动服务器，请参见“以强制恢复模式启动服务器”一节，并以强制恢复模式启动服务器。

如果无法以强制恢复模式启动服务器，请致电技术支持部门。这时，可能需要从备份恢复数据库。

2. 是否可以连接到数据库？

如果不能连接到数据库，请参见“数据库连接问题”一节，以获得故障排除建议。

如果可以连接到数据库，并且以前曾经通过强制恢复启动服务器，请参见“分配问题分析”一节，以了解检验数据库分配和恢复泄漏块的相关信息。

如果可以连接到数据库，但怀疑数据库可能不一致，请参见“数据库检验”一节，以了解有关检查数据库一致性的信息。

3. 服务器正在运行，并且可以连接，但您希望检验数据库的一致性。

如果以前曾经通过强制恢复启动服务器，或者怀疑数据库不一致，则应当运行 DBCC 检查，以验证数据库。有关同时检查索引一致性和数据库分配的信息，请参见“数据库检验”一节。

4. 服务器正在运行并且可以连接，已经运行 DBCC 检查，并且需要修复 DBCC 所检测到的索引不一致或分配问题。

如果 `sp_iqcheckdb` 在结果的“Index Summary”和“Index Statistics”部分中报告错误，请参见“索引错误修复”一节，以了解使用 DBCC 修复索引问题的过程。

如果 `sp_iqcheckdb` 在结果的“Allocation Summary”和“Allocation Statistics”部分中报告错误，请参见“使用 DBCC 修复分配问题”一节，以了解使用 DBCC 修复分配问题的过程。

另请参见

- 分配问题分析（第 431 页）
- 数据库连接问题（第 452 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- 服务器操作问题（第 444 页）
- `sp_iqcheckdb` 存储过程（第 478 页）

服务器操作问题

问题可能会影响服务器的操作，包括启动、关闭、不响应和异常终止故障。

另请参见

- 分配问题分析（第 431 页）
- 数据库连接问题（第 452 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- `sp_iqcheckdb` 存储过程（第 478 页）
- 以强制恢复模式启动服务器（第 435 页）
- 恢复泄漏的空间（第 436 页）
- 服务器恢复和数据库修复的决策流程（第 443 页）

Sybase IQ 无法启动

如果启动服务器时遇到问题，则 `start_iq` 返回一个非零值。

如果在启动时没有于 `-o` 开关后指定日志文件，则 Sybase IQ 会将错误写入以下最先定义的日志文件中：

- `$IQDIR15/logfiles/<servername>.nnnn.stderr`
- `$IQDIR15/logfiles/<servername>.nnnn.srvlog`
- 系统应用程序日志文件

可能的原因包括：

事务日志文件与数据库不匹配

以下消息出现在服务器日志文件 (`.srvlog`) 以及正在启动服务器的窗口中：

```
Starting database "dbname" (/dbdir/dbname.db)
at Fri Apr 27 2009 10:53 Transaction log: dbname.log
Error: Cannot open transaction log file
-- Can't use log file "dbname.log" since the database
file has been used more recently
Cannot open transaction log file
-- Can't use log file "dbname.log" since the database
file has been used more recently
Database server stopped at Fri Apr 27 2009 10:53
```

如果这些错误是您正在启动服务器时报告的，请检查以确保服务器正在使用正确的事务日志文件。如果找不到正确的事务日志文件，则从此情况恢复到正常的最安全方式是从最后一个有效备份恢复。

如果找不到正确的事务日志，并且无法选择从备份恢复，则执行不带事务日志的紧急恢复。

服务器找不到事务日志

如果服务器由于找不到事务日志而未能启动，则服务器日志文件中将出现相应消息。

将显示如下消息：

```
Transaction log: /dbdir/dbname.log... Error: Cannot open transaction
log file -- No such file or directory Cannot open transaction log
file -- No such file or directory
```

如果此错误是在您尝试启动服务器时报告的，请查找事务日志文件，并将该文件复制到与数据库 `.db` 文件相同的目录中。如果找不到正确的事务日志文件，请从最后一个有效备份恢复。

如果没有其它启动服务器选项可用，则可能能够使用紧急恢复 `-f` 选项来启动服务器。如有必要，请与 Sybase 技术支持部门联系以获得帮助。

警告！ 该过程具有非常高的危险性，因此建议您除非万不得已，否则尽量不要使用该过程。

服务器名在网络上不唯一

如果系统中有多个同名服务器，以下消息在您使用 **start iq** 尝试启动服务器时将出现在服务器日志文件 (*.srvlog 或 **-o** 启动选项中指定的名称) 中：

```
DBSPAWN ERROR: -85  
Communication error
```

如果在服务器日志文件中看到这些错误，并且服务器不启动，请尝试使用 **iqsrv15** 命令启动服务器。**iqsrv15** 命令返回更具体的错误消息：

```
A database server with that name has already started
```

一旦已经检验问题是由网络中重复的服务器名导致的，请用与已在运行的服务器的名称不同的其它名称启动服务器。

另请参见

- 无法运行 **start iq** (第 447 页)

日志文件具有非法名称

如果指定了单独的请求级别的记录文件，但文件名为非法标识符，则服务器启动时将出现错误。

```
Naming conflict: "iqdemo" --  
aborting
```

```
Database naming conflict --  
aborting startup
```

这些错误可能指示 **-zo** 选项指定的文件路径中有空格。

再次指定 **-zo** 选项并用引号将包含空格的任何文件名引起来。

服务器端口号在计算机上不唯一

如果 Sybase IQ 服务器正在运行，并且您试图使用相同端口号在相同计算机上启动另一个 Sybase IQ 服务器，则服务器日志文件 (*.srvlog) 中将出现消息。

将出现以下消息：

```
Trying to start TCPIP link ... TCPIP communication link not started  
Unable to initialize requested communication links ... DBSPAWN  
ERROR: -85 Communication error Server failed to start
```

如果在服务器日志文件中看见这些消息，并且服务器不启动，请运行 **stop iq** 命令 (UNIX) 以显示已在计算机上运行的 Sybase IQ 服务器的名称和端口号。然后尝试启动服务器，并指定未使用的端口号，或不指定端口号。如果启动服务器并且不提供端口号 (并且缺省端口号已在使用中)，则 Sybase IQ 将生成可用端口号。

下面是在启动服务器并且不指定端口号时在服务器日志文件中出现的消息：

```
Trying to start TCPIP link ... Unable to start on default port;  
starting on port 49152 instead TCPIP link started successfully Now  
accepting requests ... Server started successfully
```

使用不正确路径启动服务器

当启动新的 **Multiplex** 服务器时，数据库文件路径必须与创建该服务器时指定的数据库文件路径匹配。

如果使用错误的路径，则服务器启动将失败，并且以下消息将显示在服务器日志文件 (*.srvlog) 中：

```
E. 08/18 07:22:19. MPX: server myserver
has been started with an incorrect catalog path
(expected path: /work/IQ-15_4/demo/mympx/iqdemo.db).
-- (st_database.cxx 7883)
I. 08/18 07:22:19. Database server shutdown due
to startup error
DBSPAWN ERROR: -82
Unable to start specified database: autostarting
database failed
```

如果看到以下消息，请使用期望的路径重新启动服务器。如果您计划使用 **UNIX** 软（符号）链接作为服务器路径，则必须在运行 **CREATE MULTIPLEX SERVER** 之前创建软链接。

未正确设置环境变量

如果数据库配置文件参数不同于 **start_iq** 使用的参数，请确保使用正确的参数启动服务器。

请参见《参考：构件块、表和过程》>“文件位置和安装设置”>“运行 **UNIX** 环境源文件”。

无法运行 start_iq

如果无法运行 **start_iq** 命令，并且通常使用配置文件或其它命令行开关，请尝试仅使用 **start_iq** 和服务器名及数据库名称来启动服务器。

如果用此简单命令可以启动服务器，则问题可能是由在命令行或配置文件中输入的一个或多个开关或参数所导致的。尝试查明哪个参数或开关正在防止服务器启动。

如果无法用最基本的 **start_iq** 命令启动服务器，则尝试使用配置文件和命令行开关启动 **iqdemo** 演示数据库。如果服务器可以从 **iqdemo** 数据库启动，则可能是数据库有问题。

如果仍然无法运行 **start_iq** 命令，请使用 Sybase Central 中的“工具”>“**Sybase IQ 15**”>“启动服务器”或 **iqsrv15** 命令。

注意： 仅将 **iqsrv15** 用于排除服务器启动错误。始终使用 **start_iq** 启动 Sybase IQ 服务器。

在运行 **iqsrv15** 之前，必须执行以下任务（通常 **start_iq** 会自动执行它们）：

- 删除所有限制，然后为堆栈大小和描述符设置限制。为此，请转到 C Shell，并发出以下命令：

```
% unlimited % limit stacksize 8192 % limit descriptors 4096
```

注意： `unlimit` 仅影响软限制。必须通过设置内核参数才能更改任何硬限制。

- 设置所有适用于所用平台的服务器选项。有关合适的选项和如何在配置文件中设置它们的详细信息，请参见《安装和配置指南》。
- 向环境中添加路径 `$IQDIR/lib64/ocs`，以便在调用 `iqsrv15` 之前装载引擎和必需的库。仅在测试期间将此路径放入环境中，如下所示：

在 AIX 中：

```
% setenv LIBPATH "${LIBPATH}:%{IQDIR15}/lib64/ocs"
```

在其它 UNIX/LINUX 平台中：

```
% setenv LD_LIBRARY_PATH "${LD_LIBRARY_PATH}:%{IQDIR15}/lib64/ocs"
```

对于任何以相对路径名创建的数据库，必须从数据库所在的目录启动数据库服务器。

请注意启动服务器时所在的目录。服务器启动目录确定了您以相对路径名创建的任何新数据库文件的位置。如果在不同目录中启动服务器，则 Sybase IQ 找不到这些数据库文件。

任何服务器启动脚本都应当在发出服务器启动命令之前将目录更改为已知位置。

`iqsrv15` 的语法如下：

```
iqsrv15
-n
server-name
-gm
number [ other-server-switches ] [ database-
file [ database-switches ] ]
```

注意： 在 `iqsrv15` 命令行处，最后一个指定的选项优先，因此，如果要覆盖配置文件，请在配置文件名之后列出要更改的任何选项。例如：

```
iqsrv15 @iqdemo.cfg -x 'tcpip{port=1870}' iqdemo
```

此处的 `-x` 参数将覆盖 `iqdemo.cfg` 文件中的连接信息。

如果在运行 `iqsrv15` 命令时服务器未能启动，请尝试使用 `iqsrv15` 实用程序以最小开关和参数再次启动。例如：

```
iqsrv15 -n <servername> <dbname>.db -c 32m -gd all -gl all
```

如果以最小参数和开关启动了服务器，则通常用于启动服务器的某个参数或开关可能导致问题。尝试查明哪个参数或开关正在防止服务器启动。

用 `iqsrv15` 命令启动服务器时，它不在后台运行，并且消息不会自动记录到服务器日志中。但是，如果包括 `-o` 文件名服务器开关，则除了服务器窗口以外，消息还将发送到该指定文件。

另请参见

- 服务器名在网络上不唯一（第 446 页）

Sybase IQ 停止处理或停止响应

您可以通过查看 Sybase IQ 消息文件检测服务器不响应的原因。

可能原因

下面是服务器不响应的两个最常见原因：

- 磁盘空间不足。
- 主缓冲区高速缓存或临时缓冲区高速缓存中的空间不足。

操作

如果服务器似乎倾向于不响应，请在处理或关机时，使用 **start iq** 命令行选项 **-z** 和 Sybase IQ 数据库选项 **QUERY_PLAN = 'ON'**，以便在 Sybase IQ 消息 (.iqmsg) 和服务日志 (.srvlog) 文件中记录有用信息。

除了记录此信息以外，还可以执行其它步骤，以确定问题的原因：

- 检查 Sybase IQ 消息文件和 You have run out of space... 消息的服务器日志文件。如果您已用完 IQ 主存储区或 IQ 临时存储，请用 **CREATE DBSPACE** 命令添加适当的数据库空间。
另外一个要点是，请将数据库选项 **MAIN_RESERVED_DBSPACE_MB** 和 **TEMP_RESERVED_DB_SPACE_MB** 设置为足够大的值，以便能够在 **DDL COMMIT** 或 **CHECKPOINT** 期间处理空间不足。几百 MB 应当足够，但对于大型数据库，可以将这些选项设置得更高。
- 通过在操作系统级别对 CPU 使用率进行几分钟监控，确定 Sybase IQ 服务器进程 (iqsrv15) 是否正在使用 CPU 周期。记录此信息。如果 CPU 使用率更改，则通常表明正在处理 Sybase IQ 服务器进程。
如果 Sybase IQ 服务器 CPU 使用率正常，则可以检查服务器正在做什么，即，服务器当前正在执行什么语句。
- 如果没有指示空间不足，请对新连接或现有连接使用 Interactive SQL，以收集以下信息（依指定的顺序）。

表 53. 要收集的服务器不响应信息

命令	信息目的
SELECT db_name()	数据库名称
CHECKPOINT	检查点可以成功
sa_conn_properties ># sa_conn_properties.out	连接信息
sa_conn_info ># sa_conn_info.out	连接信息
sa_db_properties ># sa_db_properties.out	数据库属性信息
sa_eng_properties ># sa_eng_properties.out	服务器属性信息

命令	信息目的
<code>sp_iqstatus ># sp_iqstatus.out</code>	数据库状态信息
<code>sp_iqconnection ># sp_iqconnection.out</code>	连接信息
<code>sp_iqtransaction ># sp_iqtransaction.out</code>	事务信息

如果不能解决此问题，请与 Sybase 技术支持部门联系以获得帮助。他们可以使用刚才收集的这些信息来帮助诊断问题。

- 当服务器不响应时，可以通过在 `$IQDIR15/logfiles` 目录（Windows 64 平台上的 `%ALLUSERSPROFILE%\%SybaseIQ\logfiles` 文件夹、Vista 64 的 `C:\ProgramData\SybaseIQ\logfiles`）中创建一个名为 `DumpAllThreads` 或 `dumpallthreads` 的文件，为每个 Sybase IQ 线程生成一个堆栈跟踪。按照建议使用程序管理器或 `start_iq` 命令启动 Sybase IQ 时，系统将自动设置 `IQDIR15` 变量。如果未设置 `IQDIR15` 变量，将在启动 `iqsrv15` 的目录中创建 `DumpAllThreads` 文件。

Sybase IQ 服务器将检测 `DumpAllThreads` 文件是否存在，并在堆栈跟踪文件 `stktcr-YYYYMMDD-HHNNSS_#.iq` 中写入每个 IQ 线程的堆栈跟踪信息。在将堆栈跟踪信息写入堆栈跟踪文件之后，将删除 `DumpAllThreads` 文件。

此堆栈跟踪信息可供 Sybase 技术支持部门用于帮助诊断问题。

- 如果可以连接到数据库，请以 10 秒为间隔在持续 10 分钟内对主缓冲区高速缓存和临时（专用）缓冲区高速缓存运行 **IQ UTILITIES** 缓冲区高速缓存监控器：
 1. 连接到数据库或使用现有连接。
 2. `CREATE TABLE #dummy_monitor(c1 INT);`
 3. `IQ UTILITIES MAIN INTO #dummy_monitor START MONITOR '-append -debug -interval 10 -file_suffix iqdbgmon';`
 4. `IQ UTILITIES PRIVATE INTO #dummy_monitor START MONITOR '-append -debug -interval 10 -file_suffix iqdbgmon';` 让该进程运行 10 分钟，然后停止缓冲区高速缓存监控器：
 5. `IQ UTILITIES MAIN INTO #dummy_monitor STOP MONITOR;`
 6. `IQ UTILITIES PRIVATE INTO #dummy_monitor STOP MONITOR;`
- 检查靠近 Sybase IQ 消息文件的末尾处有无消息 `Resource count 0`，后面跟有 `Open Cursor` 消息。这些消息表示资源耗尽，这可能导致死锁。直接解决方案是使用 `CTRL-C` 或 **DROP CONNECTION** 命令以减少活动连接数。避免由于资源耗尽而导致死锁的长期解决方案是以下方案之一或它们的组合：
 - 通过减少 `-gm` 服务器启动选项的值，限制服务器的用户数
 - 将另一台辅助服务器添加到 **Multiplex**
 - 通过添加 CPU，增加硬件的处理容量

另请参见

- 磁盘空间不足（第 455 页）

- IQ 主存储区和 IQ 临时存储空间管理 (第 146 页)
- IQ_SYSTEM_MAIN 数据库空间 (第 128 页)
- 定义数据库期间的装载性能 (第 277 页)
- 主 IQ 存储块消息 (第 474 页)
- 监控磁盘空间使用情况 (第 459 页)
- 处理问题 (第 463 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- 查找当前执行的语句 (第 479 页)
- 记录服务器请求 (第 479 页)

系统故障/Sybase IQ 故障

您可以通过查看 Sybase IQ 消息文件检查系统/Sybase IQ 故障的原因。

可能原因

各种原因。

操作

- 在尝试重新启动数据库之前，复制或重命名消息日志文件 (dbname.iqmsg)。这将确保文件中的任何有用信息不会丢失。
- 在 UNIX 上，将堆栈跟踪的副本发送至 Sybase 技术支持部门。堆栈跟踪信息应当在启动数据库服务器的目录中名为 stkrtrc-YYYYMMDD-HHNNSS_#.iq 的文件中。如果故障发生时数据库是打开的，则堆栈跟踪信息还应当在 Sybase IQ 消息日志 (缺省名称是 dbname.iqmsg) 中。此信息将帮助 Sybase 技术支持部门确定失败发生的原因。
- 使用 **start iq** 命令重新启动服务器。当数据库重新启动时，将自动恢复。
- 尝试在不启动数据库的情况下启动服务器。如果能够启动服务器但不能启动数据库，请检查是否在启动行上和/或在连接配置文件中正确指定了数据库参数。
- 如果广泛查询 Catalog 存储表，请重新启动服务器并确保 **TEMP_SPACE_LIMIT_CHECK** 选项为 ON。使用此选项设置时，如果连接超过其 Catalog 存储临时文件空间的限额，它将收到非致命错误。

服务器无法关闭

要关闭服务器，请运行 **dbstop** 实用程序或 **stop iq**，在 UNIX 上的服务器窗口中键入 **q**，或在 Windows 上单击服务器窗口中的“关闭”。

可能原因

各种原因。

操作

如果服务器无法关闭，请执行以下操作。

在 UNIX 系统中：

故障排除提示

1. 捕获 **ps** 操作系统实用程序输出，以便可以将此输出提交给 Sybase 技术支持部门。在 Sun Solaris 上，有两个不同 **ps** 选项可用。二者都要使用。

```
ps -aAdeflclj|egrep "PPID|iqrsv15"
```

```
/usr/ucb/ps -awwwlx|egrep "PPID|iqrsv15"
```

2. 尝试在操作系统级别终止进程以生成核心转储。

```
kill -6 pid
```

小型核心文件是在运行 **start_iq** 的目录中创建的。如果能够以此方式终止服务器进程，请跳到步骤 5。

3. 如果服务器进程仍然不退出，请按步骤 1 捕获 **ps** 输出。保留两次运行 **ps**（在尝试终止进程之前和之后）的输出。然后用更强的信号终止进程：

```
kill -9 pid
```

4. 如果此方法不能使进程退出，请捕获另一组 **ps** 输出，然后重新启动系统。
5. 将所有 **ps** 输出、核心文件（如果在步骤 2 中生成）和 **stktrc-YYYYMMDD-HHNNSS_#.iq** 中的堆栈跟踪信息提交给 Sybase 技术支持部门。

在 Windows 系统上：

1. 通过右键单击任务栏并单击“**任务管理器**”，启动“任务管理器”。
2. 在“进程”选项卡中，选择 **iqrsv15.exe**，然后单击“**结束进程**”按钮以停止数据库服务器。
3. 如有必要，请重新启动 Windows。

数据库连接问题

尝试连接到数据库时可能遇到问题。

另请参见

- 分配问题分析（第 431 页）
- 数据库检验（第 420 页）
- 数据库修复（第 427 页）
- 索引错误修复（第 430 页）
- 在执行检查点操作期间用完空间（第 457 页）
- 服务器操作问题（第 444 页）
- **sp_iqcheckdb** 存储过程（第 478 页）
- 以强制恢复模式启动服务器（第 435 页）
- 恢复泄漏的空间（第 436 页）
- 服务器恢复和数据库修复的决策流程（第 443 页）
- 在恢复后重新连接（第 403 页）

无法连接到数据库

连接到数据库时可能会遇到问题。

可能原因

- 未定义数据源，或者已错误地输入或定义它。
数据源是一组连接参数，存储于注册表（在 Windows 上）或文件（Windows 和 UNIX）中。

- 指定了不正确的用户名或口令。返回的错误消息是：

```
Unable to connect
```

或

```
Could not connect to the database.
```

后跟如下消息：

```
Invalid user ID or password.
```

尝试以正确的用户 ID 和口令再次连接。

- 用户可能无权使用该数据库。
- 您正通过 TDS 进行连接（例如，使用 jConnect），并且用户 ID 或口令长于 30 字节。您会看到：

```
Invalid user ID or password CT-LIBRARY error: ct_connect():  
protocol specific layer: external error: The attempt to connect  
to the server failed.
```

- 提供的数据库文件名不正确。返回的错误消息是：

```
Unable to connect
```

或

```
Could not connect to the database.
```

后跟如下消息：

```
Specified database not found.
```

尝试以正确的数据库文件名再次连接。

如果您使用 **Interactive SQL** 且在连接到 `utility_db` 时已经通过备份恢复了数据库，则必须提供 **DBF** 参数和数据库文件名才能连接该数据库。

- 数据库文件可能已丢失。文件 `dbname.db`、`dbname.iq` 和 `dbname.iqmsg`（其中，`dbname` 是数据库的名称）必须均已存在。
- 可能超过了连接数限制或 DBA 定义的其他登录限制。返回的错误消息是：

```
Unable to connect Database server connection limit exceeded.
```

- 磁盘空间不足。检查 **Sybase IQ** 消息文件中是否有与磁盘空间相关的消息。
- 指定的服务器名不正确。返回的错误消息是：

```
Connection failed. Database server not running.
```

检查服务器的名称，并尝试以正确的服务器名再次连接。

- 服务器计算机名称或地址已经更改。
- 第一次从客户端连接并且未指定服务器名时，如果提供错误的端口号，可能会导致连接到数据库的操作失败。返回的错误消息是：

```
Could not connect to the database. Database server not found.
```

从 **Interactive SQL** 连接时，确保“服务器名”字段中的名称拼写正确、网络选项卡上的网络选项正确且数据库服务器已启动。请在连接时提供服务器名，或使用正确的端口号。若要确定服务器名和该服务器正在侦听的端口号，请运行命令 **stop_iq (UNIX)**，它会显示此信息。

- 端口号可能不在正确范围内，或正被另一个进程使用。
- 尝试启动客户端时，如果收到消息：

```
Unable to start — server not found
```

或

```
Database server not running.
```

则客户端无法在网络中找到数据库服务器。连接字符串可能不正确，或者服务器名高速缓存可能包含不正确或旧的连接信息。例如，如果从不同端口号启动服务器，则即使客户端应用程序在连接时指定了新端口号，系统仍然会从服务器名高速缓存中取得连接信息。

- 在 **CharSet** 连接参数中指定了字符集并尝试连接到不支持该字符集的服务器。如果服务器不支持所请求的字符集，则连接失败。

尝试在不指定 **CharSet** 的情况下重新连接。如果服务器不支持客户端的本地字符集，则连接会成功，但会显示不支持此字符集的公告。

注意： 不要将无法连接到数据库与 **Sybase IQ** 正在尝试打开数据库时出现的 **Sybase IQ** 服务器级别的错误混淆在一起。

操作

如果怀疑无法连接的原因是数据库有问题，则可以查看 `dbname.iqmsg` 文件，以确定哪里发生了问题。

如果出现消息“已完成打开数据库”，则数据库会打开而不显示错误，并且问题与客户端连接有关。如果不出现消息，则数据库可能已在打开或恢复时失败。

另请参见

- 必需的命令行选项（第 21 页）

Interactive SQL (dbisql) 问题

本节包含对 **dbisql** 的操作相关问题进行故障排除的信息。

在退出 dbisql 之后，目录保留下来

此问题只影响 NFS 文件系统的用户。

可能原因

- IQTMP15 环境变量未设置为指向本地目录。
每个客户端连接都会在临时目录中创建若干个目录和文件。当连接结束时，Sybase IQ 将删除这些文件。如果 IQTMP15 未指向本地目录，则无法找到 NFS 创建的 .nfs* 文件。

操作

- 将 IQTMP15 设置为本地目录，并重新启动服务器。

资源问题

资源问题可能包括磁盘空间不足、线程数不足、线程堆栈溢出和系统资源未使用。

另请参见

- 在 Sybase Central 中设置数据库选项（第 145 页）

磁盘空间不足

Sybase IQ 服务器在数据库空间不足的情况下不等待额外的空间，而是回退整个事务或回退到某个保存点。

如果没有足够的临时或主数据库空间可用于缓冲区或数据库空间分配请求，则进行请求的语句将回退。

此时，DBA 可使用 **ALTER DBSPACE** 或 **ALTER FILE** 命令向数据库空间添加更多空间。（您可以选择添加文件而不添加数据库空间。一个数据库空间可以有多个 dbfile。）

警告！ 磁盘空间用完时，如果 Sybase IQ 持有某些系统锁或正在执行检查点，您可能无法添加磁盘空间。因此，识别磁盘空间何时不足，并在空间不足之前添加新的数据库空间，这非常重要。

操作

- 在 Sybase IQ 消息日志 (dbname.iqmsg) 中检查最近的消息。如果看到“空间不足”消息，则必须添加另一个数据库空间。Sybase IQ 消息文件中的消息指示哪个数据库空间的空间已用完。如果在插入数据时出现该问题，则可能需要更多的 IQ 主存储区。如果在执行大型排序合并连接操作的查询过程中出现该问题，则可能需要更多的 IQ 临时存储空间。

检查 Sybase IQ 消息日志有无以下消息：

- 如果缓冲区或数据库空间分配请求因数据库空间中无空间而失败，则将在 dbname.iqmsg 消息文件中记录以下错误消息：

```
You have run out of space in %2 DBSpace. %1
```

```
[EMSG_OUT_OF_DBSPACE: SQL Code -1009170L, SQL State QSB66, Sybase Error Code 20223]
```

其中，%2 是数据库空间的名称。

此错误消息替换了错误消息 `You have run out of { IQ STORE | IQ TEMPORARY STORE } dbspace in database <dbname>. In another session, please issue a CREATE DBSPACE ... { IQ STORE | IQ TEMPORARY STORE } command and add a dbspace of at least nn MB.`

- 如果在数据库空间不足的情况下回退整个事务，则会报告以下错误消息：

```
%1 -- Transaction rolled back"
```

```
[IQ_TRANSACTION_ROLLBACK: SQL Code -1285L, SQL State 40W09, Sybase Error Code 2973]
```

其中，%1 在关键操作过程中服务器遇到的导致事务回退的错误。

- 如果缓冲区分配请求发现一个脏缓冲区，但缓冲区管理器由于空间不足情况而无法刷新缓冲区，则会返回以下错误消息且当前语句将回退：

```
%2: All buffer cache pages are in use, ask your DBA to increase the size of the buffer cache. %1
```

```
[EMSG_BUFMAN_ALLSLOTSLOCKED: SQL Code -1009031L, SQL State QSA31, Sybase Error Code 20052]
```

其中 %2 是抛出异常的特定缓冲区高速缓存。

- 尝试通过新连接连接到数据库。如果成功，即使查询正在等待，也可以知道数据库服务器正在运行。运行 `sp_iqstatus` 以获取详细信息。
- 如果无法连接到数据库，请通过监控该处理器的 CPU 使用率来检查 Sybase IQ 是否处于不可用状态。如果 CPU 使用率在一小段时间间隔后未发生变化，则 Sybase IQ 可能无法运行。如果 CPU 使用率发生变化，则 Sybase IQ 正在运行。
- 检查 `sp_iqstatus` 输出中是否有以下两行：

```
Main IQ Blocks Used: ,10188 of 12288, 82%, Max Block#: 134840
```

```
Temporary IQ Blocks Used: ,163 of 6144, 2%, Max Block#: 97
```

如果已使用块的百分比在九十以内，则需要用 `CREATE DBSPACE` 命令添加更多磁盘空间。在此示例中，主 IQ 块使用了 82%，临时 IQ 块使用了 2%，因此不久将需要在 IQ 主存储区中增加更多空间。

- 如果出现空间不足状况，或者 `sp_iqstatus` 显示 Multiplex 服务器上较高百分比的主块在使用时，请运行 `sp_iqversionuse`，以查明正在使用何种版本及通过释放该版本可以恢复的空间量。

另请参见

- IQ 主存储区和 IQ 临时存储空间管理（第 146 页）
- IQ_SYSTEM_MAIN 数据库空间（第 128 页）
- 定义数据库期间的装载性能（第 277 页）
- 主 IQ 存储块消息（第 474 页）

- 监控磁盘空间使用情况 (第 459 页)
- 处理问题 (第 463 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- Sybase IQ 停止处理或停止响应 (第 449 页)

在执行检查点操作期间用完空间

以强制恢复模式启动并尽快添加空间。

只有在添加了数据库空间的情况下，任何新的检查点才能成功。有关 Multiplex 服务器，请参见《使用 Sybase IQ Multiplex》。

另请参见

- 分配问题分析 (第 431 页)
- 数据库连接问题 (第 452 页)
- 数据库检验 (第 420 页)
- 数据库修复 (第 427 页)
- 索引错误修复 (第 430 页)
- 服务器操作问题 (第 444 页)
- sp_iqcheckdb 存储过程 (第 478 页)
- 以强制恢复模式启动服务器 (第 435 页)
- 恢复泄漏的空间 (第 436 页)
- 服务器恢复和数据库修复的决策流程 (第 443 页)

磁盘空间不足对检查点的影响

如果在请求检查点时 Sybase IQ 已经用完空间，则 **checkpoint** 命令将失败，并返回错误。

```
You have run out of space during the CHECKPOINT operation.
```

```
[EMSG_IQSTORE_OUTOFSPACE_CHECKPOINT:'QSB33', 1009133].
```

只有在添加了数据库空间的情况下，任何新的检查点才能成功。

在无法连接到服务器的情况下添加空间

如果在操作期间用完空间，并且由于无法连接到服务器而无法添加空间，请使用 **CREATE DBSPACE** 命令来添加空间。

1. 使用以下任意一种方法关闭服务器：

- 在任何平台上，运行 **dbstop**。
- 在 Windows 上，在 Windows 任务栏上单击正确的服务器图标以显示“Sybase IQ”窗口，然后单击“关闭”按钮。
- 在 UNIX 上，运行 **stop_iq**，或在启动服务器的窗口中键入 **q**。

2. 使用 **start_iq** 命令重新启动引擎。
3. 连接到数据库。
4. 使用 **CREATE DBSPACE** 命令 添加空间。
5. 重新运行最初由于空间不足而失败的操作。

管理数据库空间大小

Catalog 文件的生长正常并根据应用程序和目录内容而变化。

.db 文件的大小不影响性能，并且 .db 文件中的可用页将根据需要重用。最大限度地抑制 Catalog 文件的增大：

- 避免对 **CREATE TABLE** 语句使用 **IN SYSTEM**。
- 运行系统存储过程之后发出 **COMMIT** 语句。
- 在运行时间长的事务后发出 **COMMIT** 语句

如果 Catalog 存储不能扩展其某个文件 (.tmp、.db, 或 .iqmsg) , Sybase IQ 将返回错误 A dbspace has reached its maximum file size。防止出现此问题：

- 定期监控空间使用情况。
- 检验 .tmp、.db 或 .iqmsg 文件所在的操作系统没有操作系统文件大小限制（例如 Sun Solaris **ulimit**）。通常 .db 和 .tmp 文件位于主 Sybase IQ 数据库目录中。.tmp 文件位于 \$IQTMP15/<servername>/tmp 下，或者如果未设置 \$IQTMP15，则位于 /tmp/.SQLAnywhere/<servername>/tmp 下。

添加错误类型的空间

如果临时数据库空间用完了空间，并且您在 **CREATE DBSPACE** 命令中意外省略了 **TEMPORARY** 关键字，则无法创建临时数据库空间。

而是在作为 **IQ_SYSTEM_TEMP** 的现有临时数据库空间中添加文件。

碎片

Sybase IQ 通过利用甚至最小的未用空间来对碎片进行控制。

但是，仍然会出现碎片。如果数据库用完空间，则即使 **sp_iqstatus** 列出的内存使用情况或者 .iqmsg 文件显示“已使用的主 IQ 块”小于 100%，它通常也表示数据库有碎片。

释放空间

请注意，当连接的空间不足时，不可能通过删除另一个连接的表或索引来释放空间，因为空间不足的事务将在其快照版本中看见这些对象。

为未来保留空间

Sybase IQ 最少自动保留 200MB 和最后一个数据库空间大小的 50%。

若要确保未来如果用完空间后能有足够空间添加新的数据库空间，请设置数据库选项 **MAIN_RESERVED_DBSpace_MB** 和 **TEMP_RESERVED_DBSpace_MB**。请将这些选项

设置得足够大，以便能够处理在 COMMIT 或 CHECKPOINT 期间发生的空间用完情况。请参见《参考：语句和选项》。

监控磁盘空间使用情况

可以使用事件处理程序来监视磁盘空间使用情况，并在可用空间较少时向您发出通知。

本节中的第一个示例对于在装载期间监控空间尤其有用。可以在启动装载之前启用事件处理程序，并在装载完成之后禁用事件处理程序。

下面是示例事件处理程序代码。可以修改此代码以执行其它类型的监控。

```
-- This event handler sends email to the database -- administrator
whenever the IQ main DBSpace is more than -- 95 percent full. -- This
event handler runs every minute. The event handler uses --
sp_iqspaceused to sample the space usage. If the space is -- more
than 95 percent full, a file that contains the date and -- time is
created in the directory where iqsrv15 is -- running. The file
contents are then mailed to the database -- administrator and the
file is removed. -- This event can be enabled before a load and be
used -- to monitor disk space usage during loading. The event can --
then be disabled after the load. create event out_of_space schedule
start time '1:00AM' every 1 minutes handler begin declare mt
unsigned bigint; declare mu unsigned bigint; declare tt unsigned
bigint; declare tu unsigned bigint; call sp_iqspaceused(mt, mu, tt,
tu); if mu*100/mt > 95 then call xp_cmdshell('date > ./
temp_m_file'); call xp_cmdshell('mailx -s add_main_dbSPACE
iqdba@iqdemo.com < ./temp_m_file'); call xp_cmdshell('/bin/rm
-rf ./temp_m_file'); end if; if tu*100/tt > 95 then call
xp_cmdshell('date > ./temp_file'); call xp_cmdshell('mailx -s
add_temp_dbSPACE iqdba@iqdemo.com < ./temp_file'); call
xp_cmdshell('/bin/rm -rf ./temp_file'); end if; end
```

以下代码可创建一个基于计时器的事件，用于监控空间使用情况，以帮助避免执行非特权操作时空间不足状况可能导致的意外回退。会在示例 iqdemo 数据库中创建 DBSpaceLogger 事件。

```
CREATE EVENT DBSpaceLogger
SCHEDULE START TIME '00:00:01' EVERY 300 SECONDS
HANDLER
BEGIN
DECLARE DBSpaceName VARCHAR(128);
DECLARE Usage SMALLINT;
DECLARE cursor_1 CURSOR FOR
SELECT DBSpaceName, Usage
FROM sp_iqdbSPACE()
WHERE Usage > 0
ORDER BY Usage
FOR READ ONLY;

OPEN cursor_1;
idx1: LOOP
FETCH cursor_1 INTO DBSpaceName, Usage;
IF SQLCODE <> 0 THEN LEAVE idx1 END IF;
```

```
IF Usage >= 70 AND Usage < 80 THEN
call dbo.sp_iqlogtoiqmsg('Information: DBSpace' +
DBSpaceName + '''s usage is more than 70%');
ELSEIF Usage >= 80 AND Usage < 90 THEN
call dbo.sp_iqlogtoiqmsg('Warning: DBSpace ' +
DBSpaceName + '''s usage is more than 80%');
ELSEIF Usage >= 90 AND Usage < 100 THEN
call dbo.sp_iqlogtoiqmsg('Critical Warning: DBSpace
' + DBSpaceName + '''s usage is more than 90%');
END IF;
END LOOP;
CLOSE cursor_1;
END;
```

另请参见

- 磁盘空间不足 (第 455 页)
- IQ 主存储区和 IQ 临时存储空间管理 (第 146 页)
- IQ_SYSTEM_MAIN 数据库空间 (第 128 页)
- 定义数据库期间的装载性能 (第 277 页)
- 主 IQ 存储块消息 (第 474 页)
- 处理问题 (第 463 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- Sybase IQ 停止处理或停止响应 (第 449 页)

线程数不足

系统可能没有所需的服务器线程数来供查询使用。

可能原因

如果客户端收到消息 (如 “此查询的可用服务器线程不足 [-1010011] ['QXA11']”) , 则查询需要对 IQ 存储使用更多内核线程。

操作

- 等待另一个查询完成并释放它正在使用的线程。然后重新提交查询。
- 运行系统存储过程 **sp_iqconnection**。IQThreads 列中包含当前分配给连接的 IQ 线程的数目。此列可以帮助您确定哪些连接正在使用大多数资源。记住, 某些线程可能已分配, 但处于空闲状态。
- 如果此情况仍然存在, 则可能需要重新启动服务器并指定更多 IQ 线程。使用 **-iqmt** 服务器启动开关可以增加 Sybase IQ 可使用的处理线程数。

缺省值为: 60 个线程乘 CPU 数 (对于前 4 个 CPU) 加 50 个线程乘 CPU 数 (对于其余 CPU) , 加用于系统的 3 个线程, 再加数据库连接和后台任务所需的线程数。例如, 在具有 12 个 CPU 和 10 个连接的系统上: $60 * 4 + 50 * (\text{numCPUs} - 4) + \text{numConnections} + 6 = 656$ 。最小值为 $\text{numConnections} + 3$ 。在 64 位平台上, 服务器线程总数不能超过 4096; 在 32 位平台上, 服务器线程总数不能超过 2048。

- 如果服务器线程已用尽或恢复过程中没有充足的线程用于连接，则 Sybase IQ 可能会返回错误 Ran out of threads. Start up server with more threads. (SQLCODE -1012024)。RESTORE 命令会尝试分配一“组”线程用于恢复操作。Sybase IQ 会尝试为每个备份设备分配至少一个线程，为每个 CPU 分配两个线程，以及为该“组”分配一个线程以用于恢复操作。确保已经为每个连接、每组以及该服务器分配了充足的线程。使用 MAX_IQ_THREADS_PER_CONNECTION 和 MAX_IQ_THREADS_PER_TEAM 数据库选项。

堆栈溢出

如果线程堆栈溢出，则可能会遇到问题。

如果在堆栈跟踪文件 (stktrc-YYYYMMDD-HHNNSS_#.iq) 中看见错误 AbortIfEndofStack，则线程堆栈已经溢出。

可能原因

- 若要避免此问题，请在 32 位操作系统上将服务器参数 **-iqtss** 设置为 300，或者在 64 位操作系统上将该参数设置为 500，然后重新启动 Sybase IQ。服务器启动开关 **-iqtss** 可以指定以 KB 为单位的线程堆栈大小。如果此值不够，请将 **-iqtss** 的值提高 72，直到解决问题。
- 如果可能，请标识出导致错误的命令，并将其转发给 Sybase 技术支持部门。

在异常退出之后留下未用的信号和共享内存

异常退出之后可能留下未用的信号和共享内存。

AIX、HP-UX 和 Linux 平台使用信号在同一计算机中的客户端和服务端之间进行通信。每台客户端分配一个信号，每台服务器也分配一个信号。当客户端放置了包以供服务器读取时，客户端将发送服务器的信号；反之，当服务器放置了包以供客户端读取时，服务器将发送客户端的信号。给定系统所需的信号数取决于通过共享内存连接到本地服务器的本地客户端应用程序的数目。如果客户端需要为一台或多台服务器的 Multiplex 连接分配多个信号，则它会尝试分配信号组中的所有信号。有关 Linux 信号内核参数的设置建议，请参见《适用于 Linux 的安装和配置指南》。

可能原因

在 UNIX 系统上终止进程可能导致留下信号或共享内存而不是将其自动清理掉。若要消除不需要的信号，应当定期运行 UNIX **ipcs** 命令以检查信号和共享内存的状态。

ipcs -a 命令列出信号和共享内存段的 ID 号、所有者和创建时间。如果所有 Sybase IQ 实例均由相同用户启动 (Sybase 建议)，则可以在 OWNER 列中搜索该用户名。请识别未被使用的共享内存段和信号。

操作

在向这些共享内存段和信号的所有者确认它们未被使用之后，请运行 UNIX **ipcrm** 命令删除它们。使用 **-m** 参数可以指定内存段 ID，使用 **-s** 命令可以指定信号 ID 号，格式为：

```
ipcrm -m mid1 -m mid2 ... -s sid1 -s sid2 ...
```

例如:

```
% ipcrm -m 40965 -s 5130 -s36682
```

过程标识符不足

Sybase IQ 按顺序为过程分配内部目录 `proc_id`，并且不重用未使用的 `proc_id`。

当删除和创建过程时，在 Sybase IQ 12.6 之前创建的数据库可能最终达到 `proc_id` 的最大限制 32767，并导致 **CREATE PROCEDURE** 在 Sybase IQ 12.6 中返回“项已存在”错误。

对于用 Sybase IQ 12.6 GA 之前的版本创建的数据库，即使数据库已升级到 Sybase IQ 12.6 或更高版本，过程的最大 `proc_id` 仍然为 32767。此限制不适用于用 Sybase IQ 12.6 及更高版本创建的数据库。

如果 `proc_id` 列的数据类型为 `SMALLINT`，则应用最大 `proc_id` 32767。若要确定数据库的当前最大 `proc_id` 值，请运行以下查询：

```
SELECT MAX (proc_id) FROM sys.sysprocedure
```

Sybase IQ 12.6 ESD7 及更高版本确保对于在 Sybase IQ 12.6 之前创建的数据库，最大 `proc_id` 处于允许 **ALTER DATABASE UPGRADE** 完成的级别。如果最大 `proc_id` 更高，则 **ALTER DATABASE UPGRADE** 将失败，并返回消息“数据库不可升级”。

对于在 Sybase IQ 12.6 之前创建的数据库，为了解决此问题，在 12.6 ESD7 及更高版本中，**ALTER DATABASE UPGRADE** 支持 `PROCEDURE ON` 子句，以便通过重新创建所有存储过程来精简 `proc_id`。语法为 **ALTER DATABASE UPGRADE PROCEDURE ON**。对于在 12.6 和更高版本中创建的数据库，将忽略 `PROCEDURE ON` 子句。

ALTER DATABASE UPGRADE PROCEDURE ON 将重新创建所有过程，但没有注释。如果要在运行命令之后恢复过程中的注释，请对源代码运行 **ALTER PROCEDURE <procedure_name>**，以获得包含注释的过程。`sp_helptext <owner>.<procname>` 命令可以用于在运行 **ALTER DATABASE UPGRADE PROCEDURE ON** 之前保存包含注释的过程文本。

作为备份，请在运行 **ALTER DATABASE UPGRADE PROCEDURE ON** 之前，立即复制数据库的 `.db` 和 `.log` 文件。由于在 **ALTER DATABASE UPGRADE** 命令期间只修改目录，所以不需要完整备份。

缓冲区不足

如果资源管理器确定没有足够的高速缓存来完成操作，则系统将不会启动操作并返回缓冲区不足错误。

使用以下建议之一来解决问题：

- 增加缓冲区高速缓存大小并再次运行操作。
- 将操作重新安排到服务器不忙并且可能会有更多缓冲区高速缓存可用的时间。
- 在 **Multiplex** 环境中，将工作负载移到其它节点。

处理问题

处理问题可能与装载、查询、索引和表访问相关。

另请参见

- 磁盘空间不足 (第 455 页)
- IQ 主存储区和 IQ 临时存储空间管理 (第 146 页)
- IQ_SYSTEM_MAIN 数据库空间 (第 128 页)
- 定义数据库期间的装载性能 (第 277 页)
- 主 IQ 存储块消息 (第 474 页)
- 监控磁盘空间使用情况 (第 459 页)
- main 存储和临时存储的大小调整准则 (第 130 页)
- Sybase IQ 停止处理或停止响应 (第 449 页)

表中有太多索引

表中有太多索引时，可能会出现这个问题。

可能原因

Microsoft Access 用户正在尝试链接到有超过 32 个索引的表。

操作

创建一个选择表中所有列的视图，并链接到该视图，而不是基表。

意外的长期装载或查询

长期装载或查询可能导致出现问题。

可能原因

- IQ 缓冲区高速缓存太大，因此操作系统正在抖动。
- IQ 缓冲区高速缓存太小，因此 Sybase IQ 正在抖动，因为它无法将足够的查询数据放到高速缓存中。
- 尝试设置 IQ 缓冲区高速缓存大小，以使系统上的总计内存要求超过总计系统内存。因此缓冲区高速缓存会自动减少到其缺省大小。
- 需要 CIS 干预的用户定义的函数或交叉数据库连接。
- WHERE 子句和 GROUP BY 子句使用的列上缺少 HG 或 LF 索引。

操作

监控分页以确定抖动是否是问题。

- 要监控 IQ 分页，请运行 IQ 缓冲区高速缓存监控器。
- 若要监控操作系统分页，请使用 UNIX **vmstat** 实用程序或其它平台特定工具，或使用 Windows 性能监视器。

根据需要重置缓冲区大小。

如果您监控分页并确定存在抖动问题，您还可以限制在执行包含涉及散列算法的查询的语句过程中的抖动量。通过调整 `HASH_THRASHING_PERCENT` 数据库选项，可以控制在语句回退且返回错误之前允许的硬盘 I/O 百分比。

`HASH_THRASHING_PERCENT` 的缺省值为 10%。增大 `HASH_THRASHING_PERCENT` 将允许在回退之前对磁盘进行更多分页，而减小 `HASH_THRASHING_PERCENT` 将减少在回退之前允许的分页。

如果查询涉及在 Sybase IQ 的早期版本中执行的散列算法，则这些查询现在可能在达到缺省的 `HASH_THRASHING_PERCENT` 限制时发生回退。将报告错误 `Hash insert thrashing detected.` 或 `Hash find thrashing detected.` (`SQLState QFA43, SQLCode -1001047`)。执行以下一项或多项操作可为查询提供执行所需的资源：

- 通过增大 `HASH_THRASHING_PERCENT` 的值来放宽分页限制。
- 增加临时高速缓存（仅限 DBA）的大小。请切记，增加临时高速缓存的大小会减少 `main` 高速缓存的大小。
- 对于此语句，尝试确定 Sybase IQ 错误估计一个或多个散列大小的原因并缓解该错误估计所产生的后果。
- 减少数据库选项 `HASH_PINNABLE_CACHE_PERCENT` 的值。

若要标识查询可能出现的问题，请通过以临时数据库选项 `QUERY_PLAN = 'ON'` 和 `QUERY_DETAIL = 'ON'` 运行查询以生成查询计划，然后检查查询计划中的估计值。选项 `QUERY_PLAN_AFTER_RUN = 'ON'` 将提供其它信息，因为查询运行完之后将打印该查询计划。生成的查询计划存在于消息日志文件中。

由于唯一值数的原因，装载失败

查询中的唯一值数可能导致出现问题。

可能原因

日志文件中的以下消息指示在具有 **LF** 索引的列中有超过 10000 个唯一值：

```
1009103: Number of unique values exceeded for index. index_name_LF
10000
```

`Low_Fast` 索引将对 1000 个唯一值进行优化，但上限为 10000 个。

操作

将 **LF** 索引替换为 **HG** 索引。

为此，请发出 **DROP INDEX** 语句，以删除错误消息中标识出的 **LF** 索引。例如：

```
DROP INDEX DBA.employee.emp_lname_LF
```

然后，发出 **CREATE INDEX** 语句以创建新的 **HG** 索引。例如：

```
CREATE HG INDEX ON DBA.employee (emp_lname)
```

不能写入已锁定的表

已锁定的表可能导致出现问题。

可能原因

在对另一个用户已对其具有写入访问权限的对象进行写入操作时，将报告以下错误消息。“无法在当前事务 (TxnID1) 中打开请求写入的对象。另一用户在事务 TxnID2 中具有写入访问权限”。

操作

使用 **sp_iqlocks** 存储过程可以标识正在阻止其他用户写入表的用户。此过程将显示与数据库当前存在的锁相关的信息，包括持有锁的连接和用户 ID、被锁的表、锁的类型以及用于标识锁的名称。

错误消息还包括正在尝试写入的用户的事务 ID (TxnID1) 和当前正在写入的用户的事务 ID (TxnID2)。如果需要与已经锁定表的事务相关的更详细信息，请运行 **sp_iqtransaction** 存储过程。

另请参见

- 锁定的工作方式 (第 347 页)
- Interactive SQL (第 124 页)
- 管理对表的写入锁争用 (第 465 页)

管理对表的写入锁争用

如果大多数事务都能够获得锁，则对多个用户所使用的表进行的写入锁的高度争用可能会影响到处理。这一节中的示例存储过程是对表的写入锁的争用进行管理的方法示例。此过程不会消除对表的写入锁争用，而是对争用进行管理，以便事务能够获取写入锁。

以下存储过程代码将对用于记录事件的名为 `dbo.event` 的表的锁争用进行管理。该过程向调用者返回 `event_id`。此表处于高度的写入锁争用状态。存储过程 `dbo.log_event` 将在表 `dbo.event` 中记录信息。如果发生访问错误，将捕获错误，有希望的写入者将以五秒的间隔休眠，然后尝试再次写入表。五秒的重试间隔通常足以让争用得到解决，从而使 `dbo.event` 表的写入锁可用。

可以修改此代码以执行其它相似任务。

```

if exists (select 1
           from sys.sysprocedure a
                join sys.sysuserperm b on a.creator = b.user_id
           where a.proc_name = 'log_event' and b.user_name = 'dbo')
then drop procedure dbo.log_event; end if; create procedure
dbo.log_event(in @event varchar(255)) on exception resume begin
  declare @event_id bigint; declare @res char(5); set
@event_id=0; loop1: loop commit work; select
max(event_id)+1 into @event_id from
dbo.event; insert dbo.event values
(@event_id,@event,current timestamp,null,null); set
@res=sqlstate; if @res = ' ' or(@res <> 'QDA29' and @res <>

```

```
'QDA11') then          leave loop1          end if;          call  
dbo.sleep(5);          end loop loop1;          commit work;          return  
@event_id end
```

若要防止关键更新操作失败，可以提前保留所有所需表的 **WRITE** 锁。例如，下面的示例保留表 `SalesOrders`、`Customers` 和 `SalesOrderItems` 的 **WRITE** 锁，这些锁是假设性更新所必需的：

```
BEGIN WHILE TRUE LOOP          LOCK TABLE SalesOrders, SalesOrderItems,  
Customers IN WRITE MODE WAIT '30:00:00';          If SQLCODE indicates  
that lock could not be acquired          then          SET status_msg = 'lock  
for required tables          not yet acquired - retrying';  
          Message to client status_msg;          ELSE          BREAK;          ENDIF;  
END LOOP; // Locks on SalesOrders, SalesOrderItems, Customers are  
acquired Update table SalesOrders ...; INSERT INTO SalesOrderItems ...;  
LOAD INTO Customers ...; COMMIT; END;
```

另请参见

- 不能写入已锁定的表（第 465 页）
- 锁定的工作方式（第 347 页）
- Interactive SQL（第 124 页）

检查点提示

检查点时间和恢复时间的缺省值已经足够，不需要更改。

检查点之间的时间缺省为 60 分钟。

启动服务器时通过更改 **start_iq** 命令或 `dbname.cfg` 配置文件中的 **-gc** 和 **-gr** 选项，可调整检查点之间的时间。**-gc** 开关指定了检查点超时期限的分钟数。**-gr** 开关指定了最长恢复时间的分钟数。数据库引擎同时使用这两个开关来计算检查点时间。

有关 **start_iq** 数据库选项的详细信息，请参见《实用程序指南》。

性能问题

可能影响性能的设置。

有关诊断和解决性能问题的完整信息，请参见《性能和调优指南》。

多 CPU 或超线程计算机的性能下降

多 CPU 或超线程计算机可能遇到问题。

可能原因

如果 Sybase IQ 知道它可以使用多少个物理 CPU，它就能最有效地运行。在打开超线程功能的计算机上，或者在 Sybase IQ 无法访问所有可用 CPU 的情况下，Sybase IQ 将创建过多线程，并以低于其能力的低效率运行。

操作

在启动服务器时，将 **-iqnumbercpus** 设置为 Sybase IQ 可用的 CPU 数目，从而覆盖实际的 CPU 数目。

有关 `start_iq` 数据库选项的详细信息，请参见《实用程序指南》。

Sybase Central 问题

某些问题可能与 Sybase Central 的操作相关。

某些 Sybase Central 字段不显示

某些 Sybase Central 字段不显示。

可能原因

系统正在使用深色背景显示白色文本。

操作

使用窗口化系统提供的功能更改 Sybase Central 显示设置，以便在白色或淡色背景上使用深色文本。

网络通信故障排除

Windows 和基于 Windows 的客户端上可能出现通信问题。

网络软件包括几个不同的组件，从而增大了出现问题的可能性。虽然我们在本节中提供了一些有关网络故障排除的技巧，但是网络故障排除方面帮助的主要来源应为您的网络通信软件供应商提供的网络通信软件文档和技术支持。

使用诊断工具获得可以在诊断各种情况（包括后面几节描述的情况）时使用的信息。

另请参见

- 连接学习路线图（第 42 页）
- 诊断工具（第 470 页）

确保正在使用兼容的协议

如果在客户端或服务器计算机上安装了多个协议堆栈，则应当确保客户端和数据库服务器正在使用相同协议。

服务器的 `-x` 命令行开关将选择一个协议列表供服务器使用，并且 CommLinks 连接参数将对客户端应用程序执行相同操作。

您可以使用这些选项来确保每个应用程序正在使用相同的协议。

缺省情况下，数据库服务器和 Client Library 都会使用所有可用的协议堆栈。服务器支持针对任何活动协议的客户端请求，并且客户端会在所有活动协议上搜索服务器。

有关 `start_iq` 数据库启动实用程序 `-x` 开关的详细信息，请参见《实用程序指南》。

确保拥有最新的驱动程序

旧的网络适配器驱动程序是通信问题的常见原因。

应当确保网络适配器拥有合适的最新版本的 **NDIS** 或 **ODI** 驱动程序。应当能够从适配器卡的制造商或供应提供商那里获得最新网络适配器驱动程序。

网络适配器制造商和提供商会提供其适配器卡的最新版本的驱动程序。大多数适配器卡制造商都有网站，您可以从那里下载 **NDIS** 和 **ODI** 驱动程序的最新版本。

还可能从网络软件的提供商那里获得最新网络适配器驱动程序。

当下载 **Novell** 客户端软件时，除了用于所有网络适配器的 **Novell** 软件以外，还包括某些网络适配器的 **ODI** 驱动程序。

在重新启动之间关闭计算机

重新启动计算机时，某些网络适配器板不能完全重置。

如果正在进行故障排除，请关闭计算机，等待几秒钟，然后在重新启动之间将它打开。

逐层诊断协议堆栈

如果客户端应用程序在与数据库服务器通信时遇到问题，则需要确保客户端和数据库服务器正在使用兼容的协议堆栈。

隔离网络通信问题的有用方法是让协议堆栈逐级工作，以测试每个级别的通信是否工作正常。

如果可以用任何方式连接到服务器计算机，则不管连接是否使用将用于 **Sybase IQ** 的相同更高层协议建立的，都说明数据链接层正在工作。

例如，可能需要尝试从运行客户端应用程序的计算机连接到运行数据库服务器的计算机上的磁盘驱动器。

在检验数据链接层正在工作后，下一步是检验正在使用与 **Sybase IQ** 相同的网络和传输层的其它应用程序是否正常工作。

测试 TCP/IP 协议堆栈

如果正在 **TCP/IP** 下运行，则有几个应用程序可以用于测试客户端计算机和服务器计算机 **TCP/IP** 协议堆栈的兼容性。

很多 **TCP/IP** 包附带的 **ping** 实用程序对于测试 **IP** 网络层很有用。

使用 Ping 测试 IP 层

每个 **IP** 层都有关联的地址 - 以句点分隔的四个整数数字（例如 191.72.109.12）。**Ping** 以 **IP** 地址作为参数并尝试将单个包发送到指定的 **IP** 协议堆栈。

首先，请通过“Ping”自己的计算机，确定自己的协议堆栈是否工作正常。例如，如果您的 IP 地址是 191.72.109.12，请在命令行提示符下输入：

```
ping 191.72.109.12
```

并等待以查看包是否被完全路由。如果已经路由，则输出将与如下所示类似：

```
c:> ping 191.72.109.12
Pinging 191.72.109.12 with 32 bytes of data:
Reply from 191.72.109.12: bytes=32 time<.10ms TTL=32
Reply from 191.72.109.12: bytes=32 time<.10ms TTL=32
Reply from 191.72.109.12: bytes=32 time<.10ms TTL=32
...
```

如果 Ping 工作，则说明计算机能够将包路由到它自己。这样就有理由确信 IP 层的设置正确。请其他人运行其 IP 地址的 TCP/IP，并尝试对其计算机进行 Ping 操作。

在继续之前，确保可以从客户端计算机对运行数据库服务器的计算机进行 Ping 操作。

使用 Telnet 测试 TCP/IP 堆栈

若要进一步测试 TCP/IP 堆栈，请在一台计算机上启动服务器应用程序，并在另一台计算机上启动客户端程序，然后测试它们是否可以正常通信。

通常 TCP/IP 实现附带的几个应用程序可以用于此用途。以下过程显示如何使用 telnet 命令测试 TCP/IP 堆栈。

1. 在一台计算机上启动 Telnet 服务器进程（或守护程序）。查看 TCP/IP 软件文档以了解如何执行此操作。若要查看典型命令行 Telnet 程序，请在命令提示符下键入以下指令：

```
telnetd
```

2. 在其它计算机上启动 Telnet 客户端进程，并查看是否可以连接。再次查看 TCP/IP 软件文档以了解如何执行此操作。对于命令行程序，通常键入以下指令：

```
telnet server_name
```

其中，*server_name* 是运行 Telnet 服务器进程的计算机的名称或 IP 地址。

如果在这两台计算机之间建立了 Telnet 连接，则说明协议堆栈是稳定的，并且客户端和服务器应当能够使用 TCP/IP 链接在两台计算机之间进行通信。如果无法建立 Telnet 连接，则说明有问题。在继续之前，应当确保 TCP/IP 协议堆栈工作正常。

诊断布线问题

网络布线或接头有故障可能会导致难以发现的问题。

尝试在具有相同配置的类型计算机上再现问题。如果问题仅发生在一台计算机上，则问题可能是布线问题或硬件问题。

有关在 NetWare 下检测布线问题的信息，请参见 Novell NetWare 手册。Novell LANalyzer 程序对于诊断以太网或令牌环网络的布线问题很有用。NetWare 授权的零售商还可以提供能够帮助您诊断和解决布线问题的 NetWare 认证工程师的姓名。

检查常见网络通信问题

熟悉常见的网络通信问题及其解决方案。

“无法启动 - 未找到服务器”消息

如果在尝试启动客户端时接收到“无法启动 - 未找到服务器”消息，则表明客户端在网络中找不到数据库服务器。

检查是否存在以下问题：

- 客户端计算机上的网络驱动程序的网络配置参数与服务器计算机上的相应参数不同。例如，两个以太网适配器卡应当正在使用通用的帧类型。对于 Novell NetWare，应在 `net.cfg` 文件中设置帧类型。在 Windows 98、Windows NT 和 Windows 2000 下，可以通过控制面板中的网络设置访问这些设置。
- 在 TCP/IP 协议下，客户端通过广播请求来搜索数据库服务器。这样的广播通常不经过网关，所以找不到位于另一个网络（子网）内的计算机上的任何数据库服务器。如果是这种情况，则必须使用 `-x` 服务器启动命令行选项以提供正在运行服务器的计算机的主机名。这是通过 TCP 连接到 NetWare 服务器所必需的。
- 没有正确地安装网络驱动程序，或者网络布线安装不正确。
- 网络驱动程序的网络配置参数与 Sybase IQ 多用户支持不兼容。

“无法初始化任何通信链接”消息

如果收到以下消息：“无法初始化任何通信链接”，则表示无法建立链接。

导致此问题的原因可能是尚未安装网络驱动程序。除非已经另行指定使用 `-x` 服务器启动选项，否则服务器和客户端将尝试使用所有可用协议来启动通信链接。请参见网络文档，了解如何安装需要使用的驱动程序。

诊断工具

若干 Sybase IQ 工具可帮助您诊断各种情况。

另请参见

- 网络通信故障排除（第 467 页）

恢复至新临时文件拓扑

如果临时 `dbfile` 无法打开或者已损坏，可以将数据库恢复至其它临时文件拓扑。

1. 在恢复期间，通过让实用程序服务器忽略备份数据库中所有临时 IQ 文件定义的方式启动实用程序服务器：

```
start_iq -n utility_startup_svr -c 32MB  
-x tcpip(port=1234) -iqnotemp
```

2. 恢复数据库:

```
RESTORE DATABASE 'iqdemo'
FROM '/system1/IQ15/IQ-15_4/demo/backup/iqmain'
```

3. 使用 **-iqnotemp** 标志重新启动已恢复的数据库:

4. 删除 IQ_SYSTEM_TEMP 中的所有文件:

```
ALTER DBSPACE IQ_SYSTEM_TEMP DROP FILE ALL
```

5. 在不使用 **-iqnotemp** 标志的情况下重新启动服务器:

6. 向 IQ_SYSTEM_TEMP 添加新临时 dbfile:

sp_iqstatus 存储过程

sp_iqstatus 存储过程提供多种 IQ 状态信息。

注意: 下面的示例显示 iqdemo 样本数据库的输出。示例用户数据库空间 iq_main 可能不出现在您自己的用户创建数据库中。

以下输出来自 **sp_iqstatus** 存储过程:

Sybase IQ (TM)	版权所有 (c) 1992-2011 Sybase, Inc. 保留所有权利。
Version:	15.4.0.6552/110812/P/GA/Sun_Sparc/OS 5.10/64bit/2011-08-12 03:08:39
现在时间:	2011-09-12 10:33:19.979
内部版本时间:	2011-08-12 03:08:39
文件格式:	1999 年 3 月 18 日 23:00
服务器模式:	IQ Multiplex 协调服务器
目录格式:	2
存储过程修订:	1
页大小:	131072/8192blks/16bpp
主 DB 文件数:	2
main 存储空间不足:	N
共享临时 DB 文件数:	0
共享临时存储空间不足:	N
本地临时 DB 文件数:	1
本地临时存储空间不足:	N
DB 块: 1-12800	IQ_SYSTEM_MAIN
DB 块: 1045440-1058239	iq_main

故障排除提示

本地临时块: 1-3200	IQ_SYSTEM_TEMP
创建时间:	2011-08-17 11:31:03.313
更新时间:	2011-09-12 10:32:00.077
主 IQ 缓冲区:	510, 64Mb
临时 IQ 缓冲区:	510, 64Mb
已使用的主 IQ 块:	8076/19200, 42%=63Mb, 最大块数: 1051107
已使用的共享临时 IQ 块:	0/0, 0%=0Mb, 最大块数: 0
已使用的本地临时 IQ 块:	113/1600, 7%=0Mb, 最大块数: 834
可用的主保留块:	6400/6400, 100%=50Mb
可用的共享临时保留块:	0/0, 0%=0Mb
可用的本地临时保留块:	1600/1600, 100%=12Mb
IQ 动态内存:	当前: 150mb, 最大: 150mb
主 IQ 缓冲区: '	已使用: 509, 已锁定: 0
临时 IQ 缓冲区: '	已使用: 8, 已锁定: 0
主 IQ I/O: '	I: L184357/P71 O: C18370/D25255/P20297 D:5613 C:51.8
临时 IQ I/O: '	I: L248471/P0 O: C22502/D25269/P4896 D:22494 C:59.3
其它版本: '	2 = 0Mb
活动 Txn 版本: '	0 = C:0Mb/D:0Mb
上次完全备份 ID: '	0
上次完全备份时间: '	
上次备份 ID:	0
上次备份类型:	无
上次备份时间:	
DB 已更新:	1
下次 ISF 备份中的块:	0 块: =0Mb
下次 ISI 备份中的块:	0 块: =0Mb
主 Tlvlog 大小:	页: 2, 记录: 413, 重放: 0/0
DB 文件加密状态:	OFF

下面是了解 Main IQ I/O 和 Temporary IQ I/O 输出代码的关键:

- I: 输入
- L: 逻辑页读取 (“查找”)
- P: 物理页读取
- O: 输出
- C 创建的页
- D 脏页
- P: 物理写入
- D: 损坏的页
- C: 压缩比

检查以下信息:

- **Main IQ Blocks Used** 和 **Temporary IQ Blocks used** 行指示正在使用哪部分数据空间。如果正在使用的块的百分比 (这些行上的中间统计信息) 达到九十以上, 则需要添加数据库空间。
- **Main IQ Blocks Used** 和 **Temporary IQ Blocks Used** 基于自用户操作不能使用保留块以来 **DB Blocks** 行 (主 IQ 块总数) 减去 **Main Reserved Blocks Available** 和 **Temp Blocks** 行 (临时 IQ 块总数) 减去 **Temporary Reserved Blocks Available** 计算。
- **Main IQ Buffers** 和 **Temporary IQ Buffers** 行说明主和临时缓冲区高速缓存的当前大小。
- **Other Versions** 显示其它数据库版本和所消耗的总空间。当这些版本不再被活动事务引用或不可引用时, 最后将删除它们。
- **Active Txn Versions** 显示活动写入事务数以及它们已经创建和损坏的数据量。如果这些事务提交, 则“毁坏的”数据将成为旧版本, 并最后被删除。如果它们回退, 则将释放“创建的”数据。
- **Main Reserved Blocks Available** 和 **Temporary Reserved Blocks Available** 显示可用保留空间的数量。
- **Main IQ I/O** 和 **Temporary IQ I/O** 行以与 IQ 消息日志相同的格式显示 I/O 状态。

另请参见

- 主缓冲区高速缓存活动消息 (第 476 页)

解释通知消息

缺省情况下, Sybase IQ 会在执行插入和装载操作期间将有关数据库的信息显示在 IQ 消息日志 (.iqmsg 文件) 中。

这些消息中的统计信息指示何时需要执行维护和优化任务, 如添加更多数据库空间。这些消息还会报告有关装载的进度。本节解释了每条通知消息。

插入的开头是操作说明, 例如:

```
In table 'tab2', the full width insert
of 2 columns will begin at record 1.
```

```
I. 02/11 13:28:14. 0000000002 Insert Started:
I. 02/11 13:28:14. 0000000002 tab2
I. 02/11 13:28:14. 0000000227 [20895]: Insert Pass 1
completed in 0 seconds.
I. 02/11 13:28:14. 0000000227 [20895]: Insert Pass 2
completed in 0 seconds.
I. 02/11 13:28:14. 0000000227 [20834]:
    1 records were inserted into 'tab2'.
```

每当 Sybase IQ 插入 **NOTIFY** 装载选项中指定数量的记录时，服务器都会发送类似下面的消息：

```
2010-05-27 13:03:49 0000000002
[20897]: 100000 Records, 2 Seconds
```

第一行显示自上次发送通知消息读取这些其它行以来 Sybase IQ 到目前为止读取了多少行，以及读取它们所花费的秒数。即使 Sybase IQ 每次读取的行数相同，时间长度也会因所读取的数据而异（例如，需要多少次数据转换）。所报告的时间间隔小于 1 秒通常会被报告为“0 秒”。

另请参见

- 连接索引中的列（第 214 页）
- 装载和插入消息（第 243 页）
- 消息日志回卷（第 10 页）
- 消息日志中的版本信息（第 8 页）

内存消息

内存消息显示 Sybase IQ 服务器内存使用情况的相关信息。

IQ 消息日志（.iqmsg 文件）中的下面一行显示内存使用情况信息：

```
Mem: 469mb/M470
```

表 54. 内存使用消息

项目	描述
Mem: # mb	此 Sybase IQ 服务器当前所使用的内存，以 MB 为单位。
M# mb	此 IQ 服务器自启动以来使用的最大 MB 数。

主 IQ 存储块消息

主 IQ 存储块消息显示主 IQ 存储中块和缓冲区使用情况的相关信息。

IQ 消息日志（.iqmsg 文件）中的下面一行描述了永久主 IQ 存储：

```
Main      Blks: U63137/6%, Buffers: U12578/L7
```


表 55. 主 IQ 存储块消息

项目	描述
U#	正在使用的块数。
#%	已填充的数据库百分比。
Buffers: U#	<p>正在使用的缓冲区数。通常情况下，此值将是 100%，原因是：缓冲区管理器将缓冲区保留在内存中，直到需要使用缓冲区存储一些其它数据。一般而言，使用的缓冲区数和锁定的缓冲区数是没有意义的，因为 IQ 会尽量积极有效地使用缓冲区。</p> <p>注意： 此值将会增大为主缓冲区高速缓存所能包含的最大缓冲区数。只要分配缓冲区，此数字就会递增，但是仅当缓冲区损坏时，此数字才会递减，当缓冲区解锁或刷新时，此数字不会递减。临时高速缓存中的对象在完成后会释放它们的缓冲区，但在 main 高速缓存中，IQ 可能会也可能不会损坏缓冲区，原因是只要缓冲区解锁，就可以重新使用它，无论它是为空、包含数据还是包含损坏的数据都是如此。</p>
L#	<p>锁定的缓冲区数。锁定的缓冲区是指正在使用且无法从高速缓存中删除的缓冲区。IQ 锁定一些对象（如散列对象）的缓冲区以将它们保留在内存中。它是否锁定其它对象（如排序）的缓冲区要取决于工作负载以及它认为该对象拥有多少份额是公平的。</p> <p>只要您请求缓冲区，此数字就会递增。如果在运行脚本时超过最大值，则超过最大值的命令将会失败并且后续命令可能无法正确完成。</p> <p>注意： 缓冲区锁定不会占用任何内存。锁定的缓冲区在内存结构中设有一个标志，无论是否锁定该缓冲区，该标志都会存在。</p>

识别服务器的磁盘空间何时不足，并在服务器出现空间不足之前添加新的数据库空间，这非常重要。有关使用事件处理程序监控磁盘空间使用情况以及在装载期间可用空间不足时通知您的示例，请参见“监控磁盘空间使用情况”一节。

另请参见

- 磁盘空间不足（第 455 页）
- IQ 主存储区和 IQ 临时存储空间管理（第 146 页）
- IQ_SYSTEM_MAIN 数据库空间（第 128 页）
- 定义数据库期间的装载性能（第 277 页）
- 监控磁盘空间使用情况（第 459 页）
- 处理问题（第 463 页）
- main 存储和临时存储的大小调整准则（第 130 页）
- Sybase IQ 停止处理或停止响应（第 449 页）

IQ 临时存储块消息

IQ 临时存储块消息显示临时 IQ 存储中块和缓冲区使用情况的相关信息。

IQ 消息日志（.iqmsg 文件）中的下面一行描述了临时 IQ 存储：

Temporary Blks: U273/0%, Buffers: U1987/L1960

表 56. 临时 IQ 存储块消息

项目	描述
U#	正在使用的块数。
#%	已填充的数据库百分比。
Buffers: U#	正在使用的缓冲区数。通常情况下，此值将是 100%，原因是：缓冲区管理器将缓冲区保留在内存中，直到需要使用缓冲区存储一些其它数据。一般而言，使用的缓冲区数和锁定的缓冲区数是没有意义的，因为 IQ 会尽量积极有效地使用缓冲区。 注意： 临时高速缓存中的对象在完成后会释放它们的缓冲区。
L#	锁定的缓冲区数。锁定的缓冲区是指正在使用且无法从高速缓存中删除的缓冲区。IQ 锁定一些对象（如散列对象）的缓冲区以将它们保留在内存中。它是否锁定其它对象（如排序）的缓冲区要取决于工作负载以及它认为该对象拥有多少份额是公平的。 只要您请求缓冲区，此数字就会递增。如果在运行脚本时超过最大值，则超过最大值的命令将会失败并且后续命令可能无法正确完成。 注意： 缓冲区锁定不会占用任何内存。锁定的缓冲区在内存中结构中设有一个标志，无论是否锁定该缓冲区，该标志都会存在。

识别服务器的磁盘空间何时不足，并在服务器出现空间不足之前添加新的数据库空间，这非常重要。有关使用事件处理程序监控磁盘空间使用情况以及在装载期间可用空间不足时通知您的示例，请参见“监控磁盘空间使用情况”一节。

主缓冲区高速缓存活动消息

主缓冲区高速缓存活动消息显示主 IQ 存储缓冲区高速缓存的相关信息。

IQ 消息日志（.iqmsg 文件）中的下面一行显示主 IQ 存储缓冲区高速缓存的相关信息：

Main I: L331224/P22 O: D25967/P7805 C:D0

表 57. 主 IQ 存储缓冲区高速缓存消息

项目	描述
Main: I: L#	逻辑文件读取次数。
P#	物理文件读取次数。
O: D#	损坏缓冲区的次数。
P#	物理写入次数。

项目	描述
C: D#	缓冲区管理器数据压缩比率。该值 = (可压缩的总字节数 - 压缩后使用的字节数) / 可压缩的总字节数 * 100。换言之，它指出压缩了多少数据（即压缩的数据占未压缩时大小的百分比）。该数字越大越好。仅有某些数据块是可以压缩的。合格的块包括索引（数据库的 90-95%）和排序集。这仅仅反映了缓冲区管理器使用的数据压缩技术。其它数据压缩可能会在数据到达缓冲区管理器之前发生，因此，总数据压缩率可能比较高。

一般而言，假设缓冲区高速缓存已满，每次物理读取应该有 10 到 1000 次逻辑读取。如果值较小，则表示缓冲区管理器中的抖动过多。如果多于 1000 次，可能表示您为缓冲区高速缓存过度分配了内存。

另请参见

- sp_iqstatus 存储过程（第 471 页）

临时缓冲区高速缓存消息

临时缓冲区高速缓存活动消息显示临时 IQ 存储缓冲区高速缓存的相关信息。

IQ 消息日志（.iqmsg 文件）中的下面一行显示临时 IQ 存储缓冲区高速缓存的相关信息：

```
Temporary I: L25240/P8 O: D4749/P0 C:D0
```

表 58. 临时 IQ 存储缓冲区高速缓存消息

项目	描述
临时: I: L#	逻辑文件读取次数。
P#	物理文件读取次数。
O: D#	损坏缓冲区的次数。
P#	物理写入次数。
C: D#	缓冲区管理器数据压缩比率。该值 = (可压缩的总字节数 - 压缩后使用的字节数) / 可压缩的总字节数 * 100。换言之，它指出压缩了多少数据（即压缩的数据占未压缩时大小的百分比）。该数字越大越好。仅有某些数据块是可以压缩的。合格的块包括索引（数据库的 90-95%）和排序集。这仅仅反映了缓冲区管理器使用的数据压缩技术。其它数据压缩可能会在数据到达缓冲区管理器之前发生，因此，总数据压缩率可能比较高。

一般而言，假设缓冲区高速缓存已满，每次物理读取应该有 10 到 1000 次逻辑读取。如果值较小，则表示缓冲区管理器中的抖动过多。如果多于 1000 次，可能表示您为缓冲区高速缓存过度分配了内存。

用户名、连接句柄和连接 ID

在临时缓冲区高速缓存消息后，将会在 `.iqmsg` 文件中针对每个数据库连接记录一次连接句柄、连接 ID (SA connID) 和用户名。

IQ 消息日志 (`.iqmsg` 文件) 中的以下行显示连接信息：

```
2010-05-12 09:34:42 0000000002 Txn 173
2010-05-12 09:34:42 0000000002 Connect: 1550990889. SA connID: 1.
User: DBA.
```

连接句柄是 `sa_conn_info` 存储过程显示的值。

注意： 要将 `-zr` 日志文件中的连接信息与 `.iqmsg` 文件中的连接信息关联，请参见“使 `.srvlog` 和 `.iqmsg` 文件的连接信息相关联”。

另请参见

- 使 `.srvlog` 和 `.iqmsg` 文件的连接信息相关联 (第 485 页)

sp_iqcheckdb 存储过程

如果怀疑数据库中存在问題，请尝试运行存储过程 `sp_iqcheckdb`。

此过程可以从磁盘中将每个数据库页读取到内存中，并进行各种一致性检查。但是，根据数据库的大小，执行检查可能会用很长时间。

`sp_iqdbstatistics` 存储过程显示通过最近执行 `sp_iqcheckdb` 过程所收集的数据库统计信息。

另请参见

- 分配问题分析 (第 431 页)
- 数据库连接问题 (第 452 页)
- 数据库检验 (第 420 页)
- 数据库修复 (第 427 页)
- 索引错误修复 (第 430 页)
- 在执行检查点操作期间用完空间 (第 457 页)
- 服务器操作问题 (第 444 页)
- 以强制恢复模式启动服务器 (第 435 页)
- 恢复泄漏的空间 (第 436 页)
- 服务器恢复和数据库修复的决策流程 (第 443 页)

检查数据库和服务器启动选项值

诊断服务器启动、资源或处理问题时，可能需要检查数据库选项和服务器启动选项的当前值。

对于已连接的用户，**sp_iqcheckoptions** 存储过程将显示已更改为非缺省值的数据库选项的当前值和缺省值的列表。**sp_iqcheckoptions** 还会列出已更改为非缺省值的服务器启动选项。

当 **sp_iqcheckoptions** 运行时，DBA 会查看为所有组和用户设置的所有永久性选项，并查看为 DBA 设置的临时选项。非 DBA 用户可以查看自己的临时选项。所有用户都能查看非缺省的服务器启动选项。

sp_iqcheckoptions 存储过程不需要使用参数。在 Interactive SQL 中运行以下命令：

```
sp_iqcheckoptions
```

系统表 DBO.SYSOPTIONDEFAULTS 包含 Sybase IQ 和 ASA 选项的所有名称和缺省值。如果需要查看所有选项的缺省值，则可以查询此表。

查找当前执行的语句

诊断问题时，可能需要知道当问题发生时正在执行什么语句。

sp_iqcontext 存储过程可以显示当运行过程时系统正在运行什么语句，并标识发出该语句的用户和连接。可以将此实用程序与 **sp_iqconnection**、**.iqmsg** 日志和 **-zr** 服务器请求日志 (**.srvlog**) 提供的信息以及堆栈跟踪信息一起使用，从而确定发生问题时正在执行什么操作。

有关 **sp_iqcontext** 存储过程的详细信息和输出样本，请参见《参考：构件块、表和过程》。若要使用连接信息使 **.iqmsg** 日志与 **-zr** 服务器请求日志条目相匹配，请将 **.srvlog** 和 **.iqmsg** 文件之间的连接信息相关联。

另请参见

- IQ 主存储区和 IQ 临时存储空间管理（第 146 页）
- 记录服务器请求（第 479 页）
- Sybase IQ 停止处理或停止响应（第 449 页）
- 使 **.srvlog** 和 **.iqmsg** 文件的连接信息相关联（第 485 页）

记录服务器请求

为了找出某些类型的问题，尤其是查询的问题，记录服务器请求将对此有帮助。

可以按两种方式启用请求级别的记录：

- 通过在启动服务器时设置 **-zr** 命令行选项。
- 通过调用 **sa_server_option** 存储过程，此过程将覆盖 **-zr** 命令行选项的当前设置。

服务器请求将记录在服务器日志文件 ***.srvlog** 中。**-zr** 服务器启动选项将启用对操作进行请求级别的记录，并设置要记录的请求类型 (**ALL**、**NONE** 或 **SQL**)。**-zo** 选

项会将请求级别的记录信息重定向到与常规日志文件分离的文件中，而 **-zs** 会限制此文件的大小。请参见《实用程序指南》> “**start_iq** 数据库服务器启动实用程序”。

注意： 如果将写入日志的查询文本的大小超过了指定的限制，则查询文本不会被截断而是完整地进行记录。

可以使用 **sa_server_option** 存储过程启用和禁用请求级别的记录，而不用重新启动 Sybase IQ。以下命令将启用对一组有限的请求进行请求级别的记录，并将输出重定向到文件 `sqllog.txt` 中：

```
call sa_server_option('request_level_logging','SQL'); call
sa_server_option('request_level_log_file',
                 'sqllog.txt');
```

以下命令将禁用请求级别的记录：

```
call sa_server_option('request_level_log_file','');
```

若要查看 **SQL** 日志文件和记录级别的当前设置，请执行以下语句：

```
select property('RequestLogFile'),      property('RequestLogging');
```

若要使用连接信息使 `.iqmsg` 日志与 **-zr** 服务器请求日志 (`.srvlog`) 条目相匹配，请将 `.srvlog` 和 `.iqmsg` 文件之间的连接信息相关联。

在 Sybase IQ 15.1 版中，对请求日志进行了修改。常见信息记录为逗号分隔的文本，而不是固定格式的行前缀。如果可能，将时间记录为“=”（表示与上一行相同）或 `+nnn`（表示上一行后的 `nnn` 毫秒）。现在的请求日志比 Sybase IQ 15.1 之前的版本小很多。

此外，在请求日志中会记录更多信息。对于查询，记录的信息是隔离级别、读取的行数和游标类型。对于 **INSERT**、**UPDATE** 和 **DELETE** 语句，记录的信息是受影响的行数和已触发的触发器的数目。

还可以选择记录在过程和触发器中执行的语句。

可以选择在请求日志中记录简写形式的查询计划。如果启用了过程记录，则还将记录过程语句的计划。

下面的输出显示以 **-zr all** 选项启动服务器时来自请求日志的摘录。在本例中，用户将连接到 `iqdemo` 数据库，并执行命令 `sp_iqstatus`。

每行中有多个逗号分隔的字段且第一个字段指示时间。定期来看，完整的时间戳是以下形式的输出：

```
MMdd hhmmss.sss 0523 095954.807,[,1000000001,sp_iq_mpx_init,16,iq
utilities status 1
```

对于此行之后的行，例如“`+13,C,1,UID=DBA`”，偏移量来自上一行。在这种情况下，“`+13`”表示自上一行以来已过去约 13 毫秒。在某些情况下，“`=`”表示自上一行以来已过去约 0 毫秒。

下面是来自请求日志的摘要：

```

0523 095954.807,[,1000000001,sp_iq_mpx_init,16,iq utilities status 1
+2,],1000000001,sp_iq_mpx_init,16 +1,[,1000000001,sp_iq_mpx_init,
62,message STRING('IQ Server ',@@servername,') to console +2,],
1000000001,sp_iq_mpx_init,62 taj% pg iqdemo.sqllog 0523 095954.807,
[,1000000001,sp_iq_mpx_init,16,iq utilities status 1 +2,],
1000000001,sp_iq_mpx_init,16 +1,[,1000000001,sp_iq_mpx_init,
62,message STRING('IQ Server ',@@servername,') to console +2,],
1000000001,sp_iq_mpx_init,62 0523 100510.344,<,1,CONNECT +13,C,
1,UID=DBA +83,>,1,CONNECT,1 +1,<,1,PREPARE,SELECT @@version, if
'A'<>'a' then 1 else 0 endif, isnull(property('IsIQ'),'NO'),
isnull(connection_property('odbc_distinguish_char_and_
varchar'),'Off'),
isnull(connection_property('odbc_describe_binary_as_va
rbinary'),'Off'), connection_property('charset'),
db_property('charset') +1,>,1,PREPARE,65536 =,<,1,EXEC,65536 +79,P,
1,[S]DUMMY<seq> =,>,1,EXEC +1,<,1,DROP_STMT,65536 =,>,1,DROP_STMT
=,<,1,PREPARE,SET TEMPORARY OPTION time_format = 'hh:nn:ss';SET
TEMPORARY OPTION timestamp_format = 'yyyy-mm-dd hh:nn:ss.ssssss';SET
TEMPORARY OPTION date_format = 'yyyy-mm-dd';SET TEMPORARY OPTION
date_order = 'ymd';SET TEMPORARY OPTION isolation_level = 0; +1,>,
1,PREPARE,65537 +1,<,1,EXEC,65537 =,[,1,*batch*,1,set temporary
option time_format = 'hh:nn:ss' +11,],1,*batch*,1 =,[,1,*batch*,
1,set temporary option timestamp_format = 'yyyy-mm-dd
hh:nn:ss.ssssss' +11,],1,*batch*,1 +1,[,1,*batch*,1,set temporary
option date_format = 'yyyy-mm-dd' +11,],1,*batch*,1 =,[,1,*batch*,
1,set temporary option date_order = 'ymd'+11,],1,*batch*,1 =,[,
1,*batch*,1,set temporary option isolation_level = 0 +11,],
1,*batch*,1 =,>,1,EXEC

```

另请参见

- 查找当前执行的语句（第 479 页）
- IQ 主存储区和 IQ 临时存储空间管理（第 146 页）
- Sybase IQ 停止处理或停止响应（第 449 页）
- 使.srvlog 和.iqmsg 文件的连接信息相关联（第 485 页）

请求日志文件分析

在 Sybase IQ 15.1 中，请求日志文件（通过设置 **-zr** 服务器启动开关生成）中的输出格式已更改。使用存储过程 **sa_get_request_profile** 和 **sa_get_request_times** 可以读取 **-zr** 日志文件并汇总结果。

sa_get_request_profile 用于分析请求日志，以确定类似语句的执行时间并将结果汇总在全局临时表 **satmp_request_profile** 中。例如：

```

call sa_get_request_profile('/sys1/users/jones/iqreqs1_zr.log');
select * from satmp_request_profile;

```

sa_get_request_times 还用于分析请求日志，以确定语句执行时间并将结果汇总在全局临时表 **satmp_request_time** 中。例如：

```

call sa_get_request_times('/sys1/users/jones/iqreqs1_zr.log');
select * from satmp_request_time;

```

有关请求级别记录的详细信息，请参见《实用程序指南》>“start_iq 数据库服务器启动实用程序”中的 **start_iq -zo** 开关、《参考：构件块、表和过程》中的

sa_server_option 系统过程，以及“SQL Anywhere Server - SQL 用法” > “监控和提高数据库性能” > “诊断性能问题” > “使用其它诊断工具和技术” > “请求记录”。

注意： 此参考指向 SQL Anywhere 文档。

用于收集诊断信息的连接

数据库选项 **DEDICATED_TASK** 允许 DBA 将请求处理任务专用于处理来自单个连接的请求。

在数据库服务器意外地不响应时，可以使用此预先建立的连接收集有关该数据库服务器状态的信息。请参见《参考：语句和选项》中的 **DEDICATED_TASK** 选项。

诊断通信问题

如果在启动时服务器返回通信错误，则可能需要在启动服务器时设置 **-z** 命令行选项。

此开关将在服务器启动时提供有关通信链接的诊断信息。信息会记录到服务器从中启动的标准输出和 **srvlog** 文件中。

向技术支持报告问题

如果使用手册或联机帮助不能解决问题，则指定的人员应当与 Sybase 技术支持部门联系或与所在区域的 Sybase 子公司联系。

对于购买了支持合同的客户安装的每一个 Sybase 产品，都会有一位或多位指定人员获得与 Sybase 技术支持部门联系的授权。

技术支持人员需要您的 Sybase IQ 环境相关信息以解决您的问题。这一节描述此信息，使您了解如何使用自动 **getiqinfo** 工具收集它，并说明如何使各种 Sybase IQ 实用程序和日志文件的信息相关。

使用 **getiqinfo** 收集诊断信息

Sybase IQ 包括一个收集 Sybase 技术支持部门诊断问题所需信息的脚本。

getiqinfo 脚本将收集有关操作系统环境、Sybase IQ 环境和日志文件的信息。

在将问题报告到 Sybase 技术支持部门之前，请运行此脚本。通过执行此操作，可以帮助 Sybase 人员更快解决问题，并减轻您的负担。

getiqinfo 脚本将自动收集解决问题可能需要的信息。

getiqinfo 脚本不是为排除 Sybase IQ 安装故障而设计的，它不提供现场故障排除功能。仅当正确设置 Sybase IQ 环境并且服务器正在运行时，才能成功执行此脚本。

在运行 getiqinfo 之前

在运行 **getiqinfo** 脚本之前收集信息。

在运行脚本之前，请将以下信息准备好：

- 数据库文件的位置
- 用于启动服务器的配置文件（如果使用）的完整路径
- `.iqmsg` 文件的完整路径（如果 Sybase IQ 消息文件已重命名）

如果可能，请让 Sybase IQ 服务器保持运行，或在运行 **getiqinfo** 之前启动服务器。这将允许脚本收集仅当 Sybase IQ 正在运行时可用的内部数据库数据。脚本不会自动启动服务器。

脚本以用于启动 Sybase IQ 服务器的相同环境设置运行。**getiqinfo** 使用某些特定于 IQ 的环境变量搜索文件。

脚本将收集的数据放在当前目录（启动程序的目录）中。请确保在该目录下有足够空间。脚本不会提示您输入替代目录，但您可以通过重置变量 `DEST_DIR` 来修改脚本，以更改输出位置。

运行 getiqinfo 脚本

在 UNIX 平台上，**getiqinfo** 是 Shell 脚本。在 Windows 平台上，`getiqinfo.bat` 是 IQ-15_4\bin64 目录中的批处理脚本。

对于 UNIX 和 Windows 平台，步骤有所不同。

1. 请按照您的平台启动脚本：

- 在 UNIX 命令提示符处的 IQ-15_4/bin32 目录（在 32 位平台上）或 IQ-15_4/bin64 目录（在 64 位平台上）下，键入：
`getiqinfo.sh`
- 在 Windows 中，选择“开始”>“运行”，然后键入 `<install_path>\IQ-15_4\bin64\getiqinfo.bat`（在 64 位平台上）。

2. 当程序出现提示时，输入：

- 数据库文件的目录。这也是 `.iqmsg` 文件的缺省位置，在 UNIX 上是 `stktrc*.iq` 文件的缺省位置。
- 数据库文件的基名（没有 `.db` 后缀的文件名）。这也是 `.iqmsg` 文件的缺省基名。
- 要在其中搜索这些文件的其它目录
- 此数据库服务器的 Sybase IQ 引擎名称（服务器名）和端口号
- 对此数据库拥有 DBA 权限的用户 ID 和口令
- 用于启动 Sybase IQ 服务器的配置文件（如果已使用它）的完整路径
- `-zo` 服务器选项中的输出文件（如果已指定一个）的完整路径

程序还会指导您将列出的文件发送给 Sybase 技术支持 部门。

由 **getiqinfo** 收集的信息

getiqinfo 脚本将收集以下所有信息：

- 硬件的类型、内存量、CPU 类型、速度、CPU 数目
- 操作系统（例如，Sun Solaris 2.10）
- 交换空间大小
- Sybase IQ 版本和 EBF 级别以及 Anywhere 版本
- 发生此问题时的日期和时间的堆栈跟踪文件，名为 `stktrc-YYYYMMDD-HHMMSS_#.iq`，在启动数据库服务器的目录中。（仅适用于 UNIX 和 Linux 平台）
- 产生错误的命令或查询
- 消息日志文件，名为 `dbname.iqmsg`，缺省情况下位于启动数据库服务器的目录中
- 查询计划（记录于 `.iqmsg` 文件中；请参见下面的注释）
- 服务器日志
 - 对于 UNIX，为 `IQ-15_4/logfiles/<servername>.000n.stderr` 和 `IQ-15_4/logfiles/<servername>.000n.srvlog`
 - 在 Windows 平台上，如果需要，必须重新启动服务器，并手动收集主控台窗口的副本。
- 配置文件（缺省情况下为 `dbname.cfg`）中的启动和连接选项设置
- 数据库选项设置和 **sa_conn_properties** 的输出（如果服务器仍在运行）

getiqinfo 不收集以下信息，但技术支持也可能请求您提供它们：

- 使用的连接协议（例如 ODBC、JDBC、TDS）
- Open Client 版本
- 配置类型（单用户或多用户）
- 使用的前端工具（例如，Brio Query）
- 数据库的 schema 和索引
- **sp_iqcheckdb** 过程的输出

本章末尾提供了记录技术支持部门可能需要的信息的清单，以防万一需要手动收集这些信息。

注意： 如果设置了下面的选项，则 **getiqinfo** 将自动收集查询计划的详细信息。通过设置选项然后重新运行产生错误的命令，也可以手动收集此信息。

```
SET TEMPORARY OPTION QUERY_PLAN = 'ON'
```

```
SET TEMPORARY OPTION QUERY_DETAIL = 'ON'
```

查询计划位于消息日志文件中。这些选项的缺省值为 `QUERY_PLAN = ON` 和 `QUERY_DETAIL = OFF`。

如果您发现性能问题，请设置以下选项：

```
SET TEMPORARY OPTION QUERY_PLAN_AFTER_RUN = 'ON'
```

设置此选项使技术支持能够查看 查询处理中的哪些步骤占用了时间。

使.srvlog 和 .iqmsg 文件的连接信息相关联

技术支持可能要求您在配置文件中设置 `start_iq` 命令的 `-zr` 选项。

此服务器启动选项将设置请求记录级别，以跟踪发送到服务器的语句。参数是 `ALL`、`NONE` 或 `SQL`。此选项将产生以服务器命名并带有后缀 `.srvlog` 的日志文件。

在 Sybase IQ 消息文件 `.iqmsg` 中，每个与服务器的连接均以连接句柄进行标识。`.iqmsg` 消息文件将记录每个连接的错误、警告和跟踪信息。以下过程介绍如何将 `.srvlog` 和 `.iqmsg` 文件中的连接标识符关联起来以查找相关信息。

1.

在 `.iqmsg` 文件中，找到感兴趣的连接。例如：

```
Connect: SA connHandle: 1000000061
```

以下行显示该连接的 `.iqmsg` 日志文件内容：

```
16:14:59. 0000000062 Connect: SA connHandle: 1000000061
SA connID: 31 IQ connID: 0000000062 User: DBA 03/17
16:15:00. 0000000062 Cmt 12064 03/17 16:15:00. 0000000062 PostCmt
0 03/17 16:15:00. 0000000000 Disconnect: SA connHandle:
1000000061 SA connID: 31 IQ connID: 0000000062
User: DBA
```

2. 通过在 `.srvlog` 文件中搜索跟在 `.iqmsg` 文件中的“SA connHandle”后面的数字，来隔离所有用于连接的行。

例如，在 `.srvlog` 文件中搜索“1000000061”：

```
16:14:59. [,1000000061,sp_iqdbspace,48,select
str_replace(dbSPACEName,'',null) into dbSPACEName_literal 03/17
16:14:59. P,1000000061,[S][0]DUMMY<seq> 03/17 16:14:59. ],
1000000061,sp_iqdbspace,48 03/17 16:14:59. P,1000000061,
[1]ISYSIQDBFILE<seq> JNL dbf<ISYSDBFILE> JNL
ISYSDBSPACE<ISYSDBSPACE> 03/17 16:14:59. [,
1000000061,sp_iqdbspace,58,execute immediate with quotes on 'iq
utilities main into iq_dbSPACE_temp dbSPACE info ' || dbSPACEName
03/17 16:14:59. P,1000000061,[S]INSERT ROWS 03/17 16:14:59. P,
1000000061,[S]INSERT ROWS 03/17 16:14:59. P,1000000061,[S]INSERT
ROWS 03/17 16:14:59. P,1000000061,[S]INSERT ROWS03/17 16:14:59. P,
1000000061,[S]INSERT ROWS 03/17 16:14:59. ],
1000000061,sp_iqdbspace,58 03/17 16:14:59. [,
1000000061,sp_iqdbspace,60,select d.dbSPACE_name as
DBSPACEName, min(SegType) as DBSPACEType,... 03/17 16:15:00. ],
1000000061,sp_iqdbspace,60 03/17 16:15:00. P,
1000000061,Work[ Sort[ GrByH[ dbf<seq> JNL
ISYSIQDBSPACE<ISYSIQDBSPACE> JNL ISYSDBSPACE<ISYSDBSPACE> JH*
iq_dbSPACE_temp<seq> ] ] ] : ISYSIQPARTITIONCOLUMN<seq> :
idx<seq> : tab<seq> 03/17 16:15:00. [,1000000061,sp_iqdbspace,
105,drop table dbo.iq_dbSPACE_temp 03/17 16:15:00. ],
```

故障排除提示

```
1000000061,sp_iqdbspace,105 03/17 16:15:00. P,1000000061,  
[1]Work[ Sort[ sp_iqdbspace<call> ] ]
```

此示例中的连接句柄是 1000000061。

另请参见

- 用户名、连接句柄和连接 ID (第 478 页)
- 查找当前执行的语句 (第 479 页)
- 记录服务器请求 (第 479 页)

另一个有用的信息源

如果无法解决问题，则可以在 Sybase 在线支持站点 MySybase 上查找其它帮助信息。

使用 MySybase 可以通过针对需求自定义的视图，搜索已完成的支持案例、最新软件公告和已解决的问题和已知问题。您甚至可以在线打开技术支持个案。

可以在大多数 Internet 浏览器中使用 MySybase。有关如何注册和使用此免费服务的信息，请将 Web 浏览器指向 Technical Documents at <http://www.sybase.com/support/techdocs/>，并单击“MySybase”。

可以在大多数 Internet 浏览器中使用 MySybase。有关如何注册和使用此免费服务的信息，请将 Web 浏览器指向《技术文档》，并单击“MySybase”。

清单：技术支持信息

您可以运行 `getiqinfo` 脚本来收集以下多数信息。

请求的信息	值
Sybase IQ 版本 (例如, 15.4 GA 或 ESD 编号)	
<code>sp_iqlmconfig</code> 输出	
硬件类型	
内存量	
CPU 数	
操作系统名称和版本 (例如, Microsoft Windows 2008 Service Pack 1)	
操作系统修补程序级别	
所使用的前端工具 (例如, Business Objects Crystal Reports)	
所使用的连接协议 (例如, ODBC、JDBC、TDS)	
Open Client 版本	
配置类型 (单节点或 Multiplex)	

请求的信息	值
消息日志文件 (dbname.iqmsg)	
服务器日志文件 (server.nnnn.srvlog 和 server.nnnn.stderr)	
堆栈跟踪文件 (stktrc-YYYYMMDD-HHNNSS_#.iq)	
产生错误的命令或查询	
启动选项设置	
连接选项设置	
数据库选项设置	
数据库的 schema 和索引	
sp_iqstatus 输出	
查询计划: 设置选项 (Query_Plan、Query_Detail、Query_Plan_After_Run、Query_Plan_As_Html、Query_Plan_As_Html_Directory、Query_Timing), 重新运行命令或查询	
问题的屏幕快照 (如有可能)	

索引

符号

- c 开关 25
- iqnotemp 470

數字

- 2000 年
 - 转换选项 267

A

- Access
 - ODBC 配置 60, 61
- Adaptive Server Enterprise
 - unicar 数据类型 273
 - unitext 数据类型 273
 - univarchar 数据类型 273
 - 插入文本和图像 252
 - 从其中插入数据 47, 252
- ALL 权限 297
- ALTER DBSPACE 语句
 - ADD 参数 165
 - SIZE 参数 166
 - 示例 165
- ALTER TABLE 语句
 - 外键 331
- ALTER 权限 297
- ALTER 语句
 - 自动提交 338
- ANSI 代码页
 - 选择 364
- APPEND_LOAD 选项
 - 分区表 242
- AppInfo 连接参数 78
- ASCII
 - 插入时转换 264
 - 数据格式 228
 - 数据提取 233
 - 转换性能 263
 - 转换选项 258
 - 字符集 362
- AStart 连接参数 80
- AStop 连接参数 81
- AUTOINCREMENT
 - UltraLite 应用程序 324

- 带符号的数据类型 324
- 负数 324
- 缺省值 324
- AutoPreCommit 连接参数 80
- AutoStart 连接参数 80
- AutoStop 连接参数 81
- AVG 函数 200
- 安全
 - 登录失败 290
 - 关于 281
 - 过程 300, 308
 - 集成登录 71, 72
 - 视图 308
- 安全性
 - Encryption [ENC] 连接参数 92
 - FIPS 支持 314
 - IPv6 支持 315
 - Kerberos 鉴定 314
 - RSA 支持 314
 - 高级安全性选项 314
 - 列加密 314
 - 数据库加密 314

B

- BACKUP 语句 382
- BCAST 通信参数
 - 说明 103
- bcp 支持 228
- BIT 数据
 - 允许的索引 194
 - 转换 261
- BLISTENER 通信参数
 - 说明 104
- BLOB 数据 252
- BLOCK FACTOR
 - BACKUP 语句选项 386
- Broadcast 通信参数
 - 说明 103
- BroadcastListener 通信参数
 - 说明 104
- 版本号 8
- 版本控制
 - 隔离级别 351
 - 关于 337, 340

索引

- 临时表 346
- 性能影响 356
- 游标 358
- 在表级 340
- 在恢复中 419
- 在系统恢复中 356
- 版本字符串 8
- 包含索引
 - 请参见 WD 索引
- 保存点
 - 在事务中 353
- 备份
 - .iqmsg 文件 393
 - iqmsg 文件 12
 - Linux 379
 - Multiplex 376
 - 包含的数据 377
 - 并发 339
 - 并发问题 382
 - 从错误中恢复 388
 - 等待时间 387
 - 第三方 389
 - 调度 412
 - 更快 392
 - 关于 375
 - 检验 389, 397, 406
 - 检验增量 406
 - 空值 390
 - 权限 284
 - 设备 379, 384
 - 使用 SAN 进行虚拟 392
 - 所需特权 380
 - 完全 412
 - 无人照管 380, 382
 - 系统级 392
 - 显示标头文件 405
 - 消息日志 12, 393
 - 消息日志存档 12, 393
 - 性能问题 413
 - 虚拟 383, 390
 - 有人照管 380
 - 在 NT 上指定磁带设备 384
 - 责任 413
 - 增加内存 413
 - 增量 412
 - 只读硬件 387
- 备份日志
 - 关于 409

- 位置 409
- 本地临时表
 - 关于 172
- 比较索引
 - 请参见 CMP 索引
- 标量值子查询 180
- 表
 - 创建 170, 171
 - 访问权限已禁止 465
 - 更改 176
 - 管理已禁止的访问权限 465
 - 将键添加到 177, 178
 - 连接多个 216
 - 连接关系 218
 - 另请参见系统表中的信息 6
 - 权限 284, 285
 - 删除 177
 - 损坏 439
 - 所有者 286
 - 锁定 347
 - 无法修复的错误 438
 - 限定名 305, 306
 - 移至新数据库空间 286
 - 只读 150
 - 装载 238
 - 组所有者 305
- 表分区 134
- 表级版本控制
 - 关于 340
 - 另请参见版本控制 340
- 表空间
 - 分配 3
- 并发
 - 备份 339, 382
 - 插入、删除和查询 347
 - 读取和写入 341
 - 数据定义 348
 - 在 Sybase IQ 中 339
- 并行 CREATE INDEX 191
- 不一致的索引
 - 修复 430
- 不一致的状态 435
- 布线
 - 故障排除 469

C

- CATALOG ONLY
- RESTORE 选项 405

- Catalog 存储
 - 关于 128
 - 设置高速缓存大小 25
 - 预分配空间 128
- Catalog 文件
 - 增长 458
- CBSize 连接参数 82
- CBSpace 连接参数 83
- Certificate 通信参数
 - 说明 105
- Certificate_Password 通信参数
 - 说明 105
- CHAR 数据
 - 零长度单元 265
- CharSet 连接参数 81
- CHECK 条件
 - 表 328
 - 列 327
 - 删除 329
 - 修改 329
 - 用户定义数据类型 327
- ClientPort 通信参数 106
- CLOB 数据 252
- CMP 索引 204
 - 建议用法 204
 - 限制 204
- CommBufferSize 连接参数 82
- CommBufferSpace 连接参数 83
- CommLinks 连接参数 84
- ConnectFailed 事件处理程序 290
- ConnectionString 连接参数 85
- ConnectionPool 连接参数 85, 98
- CONVERSION_ERROR 数据库选项 276
- COUNT DISTINCT
 - 对索引选择的影响 192
- COUNT 函数 200
- CREATE DATABASE 语句 140
 - IQ RESERVE 参数 164
 - IQ SIZE 参数 164
 - TEMPORARY RESERVE 参数 164
 - TEMPORARY SIZE 参数 164
 - 归类 361
 - 原始设备 140
- CREATE DBSPACE 语句 162
 - RESERVE 参数 164
 - SIZE 参数 164
- CREATE INDEX 语句 190
- CREATE JOIN INDEX 语句 174, 221
- CREATE TABLE 语句
 - 和命令文件 124
 - 示例 171
- CREATE 权限 286
- CREATE 语句
 - 并发规则 348
 - 自动提交 338
- CS 连接参数 81
- CS_TEXT_TYPE 252
- 参照完整性
 - 列缺省值 320
 - 强制 329
 - 权限 228
 - 声明 330
 - 使用现有未受强制的外键强制实施 332
- 操作员
 - 任务 413
- 插入
 - 从 Adaptive Server Enterprise 数据库 252
 - 从旧版本 258
 - 从其它数据库 252
 - 概述 227
 - 连接索引表 256
 - 列宽问题 262
 - 列缺省值 321
 - 另请参见装载数据 258
 - 性能 276
 - 选定行 252
 - 以交互方式 254
 - 主键列和外键列 256
- 插入转换选项 258
- 查询
 - 并发规则 347
 - 抖动 463
 - 范围谓词 208
 - 使用 DATEPART 207
 - 限制并发 25
 - 性能 203
 - 性能问题 463
- 查询服务器
 - 替换写入服务器 423
- 查询类型
 - 索引类型 192
- 查找索引 197
- 产品支持 482
- 常量表达式缺省值 325
- 超线程
 - 服务器开关 26
- 持有游标 345, 359

索引

创建

- 列缺省值 321

词索引 205

磁带设备

- 用于备份 384

磁盘

- 监控空间使用情况 459

- 空间不足 449, 455

磁盘空间

- 保存 359

- 分配 162

- 索引 194

磁盘空间不足

- 监控空间使用情况 459

- 建议的操作 449, 455

磁盘阵列

- WORM 414

存储过程

- Sybase IQ 5

- 关于 5

- 授予执行权限 310

错误

- 报告 482

- 插入和删除 347

- 事务处理 347

- 数据库空间不足状况 455

- 数据转换 276

- 无法修复 438

错误消息

- PIPE_NOT_CONNECTED 239

- 重定向到文件 229

- 字符集转换 366

D

DatabaseFile 连接参数 86

DatabaseName 连接参数 87

DatabaseName 通信参数

- 说明 107

DatabaseSwitches 连接参数 88

DataSourceName 连接参数 89

DATE 格式

- 转换两位数的日期 267

DATE 数据类型

- 优化装载 263

- 指定要转换的格式 266

DATE 索引 206

- 建议用法 206

- 其它索引 209

- 缺点 209

- 优点 209

- 与其它索引的比较 209

DATE 选项 258, 265

DATEPART

- 查询 207

DATETIME

- 转换选项 258

- 装载转换选项 267

DATETIME 数据类型 267

- 优化装载 263

- 转换格式 268

DBA 权限

- 不可继承 302

- 关于 283

- 授予 284, 296

DBA (数据库管理员)

- 定义的 283

- 角色 4

DBCC

- sp_iqcheckdb 接口 420

- 分配检验和修复 431

- 分析分配问题 431

- 分析索引问题 427

- 检测分配错误 432

- 检测索引问题 439

- 检查分配 422

- 检查索引和分配 421

- 内部索引检查 421

- 输出 423

- 输出消息 440

- 输出样本 423

- 数据库检验 420

- 索引检验和修复 427

- 性能 421

- 修复分配 431, 433

- 修复索引 427

- 运行时间 421

DBCC_LOG_PROGRESS 选项 395, 423

DBF 连接参数 86

- 嵌入式数据库 53

dbisql

- 登录窗口 53

- 故障排除 454

- 另请参见 Interactive SQL 124

- 命令行参数 52

- 以交互方式插入数据 254

- 指定输出格式 228

- DBISQL
 - 简介 124
- DBKEY 连接参数 90
- dblog 实用程序 404
- DBN 连接参数 87
- DBN 通信参数
 - 说明 107
- dbo 用户 ID
 - 拥有的视图 181
- DBS 连接参数 88
- DBSpaceLogger 事件 459
- DDL
 - 关于 123
- DDL 锁定 348
- DDL (数据定义语言) 7
- DEFAULT 登录策略 288, 289
- DEFAULT_DISK_STRIPING 选项 3
- DELETE 权限 297
- DELIMITED BY 选项 240
- Delphi
 - ODBC 配置 60
- DisableMultiRowFetch 连接参数 90
- DML 135
- DML (数据操作语言) 7
- DMRF 连接参数 90
- DOBROADCAST 通信参数 107
- DROP JOIN INDEX 语句 174
- DROP TABLE 语句
 - 示例 177
- DROP VIEW 语句
 - 示例 182
 - 限制 181
- DROP 语句
 - 并发规则 348
 - 自动提交 338
- DSN 连接参数 89
 - 关于 57
- DSS (决策支持系统) 1
- DTM 索引 206
 - 建议用法 206
 - 其它索引 209
 - 缺点 209
 - 优点 209
 - 与其它索引的比较 209
- DumpAllThreads 文件 450
- dumpdups
 - sp_iqcheckdb 选项 423
- dumpleaks
 - sp_iqcheckdb 选项 423
- dumpunallocs
 - sp_iqcheckdb 选项 423
- 大对象数据 252
- 大括号 252
- 代码页
 - 关于 362
 - 支持 362
- 单节点模式 174
- 单字节字符集
 - 关于 362
- 当前日期和时间缺省值 323
- 导出数据
 - 概述 227
 - 关于 229
- 导入数据
 - LOAD TABLE 语句 238
 - 从 12 版以前的 IQ 数据库 252
 - 从 Adaptive Server Enterprise 252
 - 转换错误 276
- 登录
 - 集成 69, 70
 - 限制 289
- 登录策略 288
 - 权限 285
 - 锁定选项 291
 - 重置 291
- 登录尝试
 - 超过限制 291
- 登录管理
 - 过程列表 289
- 登录失败 290
- 顶级表 212
 - 大小和性能 216
- 抖动
 - HASH_THRASHING_PERCENT 选项 464
- 读取操作
 - 不显示警告 61
- 端口号
 - 缺省 52
 - 数据库服务器 117
 - 在 UNIX 上指定 52
- 断开连接 75
- 堆栈跟踪
 - 为线程生成 450
 - 位置 451
- 对数据执行 bulk load 操作
 - 客户端数据 242

索引

对象

- 限定名 306
- 多处理器计算机
 - 开关 23
- 多线程
 - 装载期间 279
- 多字节字符集
 - 使用 365

E

- ENC 连接参数
 - 说明 92
- EncryptedPassword 连接参数 91
- Encryption 连接参数
 - 说明 92
- ENG 连接参数 90
- EngineName 连接参数 90
- ENP 连接参数 91
- Escape 连接参数 93
- 二进制
 - 数据提取 233
- 二进制装载格式
 - LOAD TABLE 246
 - 数据文件 246

F

- FileDataSourceName 连接参数 94
- FileDSN
 - 创建 61
 - 分发 61
 - 连接参数 57
- FIPS 支持 314
- FORCE_DROP 选项 439
- FP(1) 索引 196
- FP(2) 索引 196
- FP(3) 索引 197
- FROM 子句
 - UPDATE 语句 280
 - 连接索引 212
- 反斜杠
 - Windows 原始设备 142
- 范围分区 134
 - 定义 133
- 防火墙
 - BroadcastListener [BLISTENER] 通信参数 104
 - LDAP 通信参数 110

- 连接 106
- 非结构化数据分析选项 252
- 非空值约束 319
- 分隔符
 - SELECT 语句 252
- 分配
 - DBCC 修复输出 433
 - 检验和修复 431
- 分配映射
 - 不一致 435
 - 检查分配 422
 - 修复错误 432
- 分区 227
 - DDL 操作 135
 - 定义 133
 - 只读 150
 - 最大数目 134
- 分区表
 - APPEND_LOAD 选项 242
 - INSERT 242
 - LOAD TABLE 242
- 分区键 134
- 分析
 - 非结构化数据分析选项 252
- 分析输出 481
- 服务器
 - CPU 使用率 449
 - 不响应 449, 450
 - 关闭并重新启动 34
 - 关闭问题 451
 - 恢复 419, 443
 - 空间不足 449
 - 连接 46
 - 命名 22
 - 命名冲突 446
 - 启动失败 435
 - 事务日志 445
 - 死锁 450
 - 停止处理 449
 - 唯一端口号 446, 447
 - 唯一名称 446
 - 在系统启动时自动启动 17
 - 另请参见 数据库服务器
- 服务器对象
 - 添加 47
- 服务器启动开关
 - iqmsgnum 10
 - iqmsgsz 10
- 服务器日志文件 33

服务器属性

- IQMsgMaxSize 10
- IQMsgNumFiles 10

G

getiqinfo 脚本 482

gm 开关 25
对恢复的影响 355, 420

GRANT 语句

WITH GRANT OPTION 299

创建组 302

过程 300

口令 295

没有口令 305

权限 297

新用户 294

组成员资格 303

GROUP BY 子句

对索引选择的影响 192

GROUP 权限

不可继承 302

高级安全性选项 314

高速缓存

写入 338

高速缓存大小

Catalog 存储的设置 25

隔离级别 351

ODBC 配置 60

跟踪

为线程生成 450

更改

归类 372

共享内存

信号 461

故障

介质 375

系统 375

故障排除 443

dbisql 454

Sybase Central 467

布线问题 469

常见问题 470

处理问题 463

服务器操作 444

数据库连接 452

协议 467

资源问题 455

关闭

故障排除 451

数据库 39

关闭并重新启动服务器 34

归类

ANSI 364

CP874toUTF8 实用程序 361

CREATE DATABASE 语句 361

ISO_1 364

SORT_COLLATION 选项 361

SORTKEY 函数 361

不再支持 361

创建数据库 372, 373

此版本的中的更改 361

多字节 365

更改 372

关于 361, 363, 364

兼容性问题 372

建议 364

建议用于日语 366

排序 363

迁移 361

迁移不建议使用的 361

缺省 362, 372, 373

替代 364

显示 364

性能问题 373

选择 368

支持 364

自定义 361

滚动

游标 359

国家/地区语言支持

多字节字符集 365

概述 361, 362

关于 361

归类 368

过程

安全 308

创建权限 284, 285

另请参见存储过程 5

权限 300

所有者 286

系统 5

H

HASH_THRASHING_PERCENT 选项 464

HG 索引

查询性能 203

建议用法 201

索引

- 具有空值的多列 202
- 空值 202
- 其它索引 202
- 缺点 202
- 外键约束 201
- 优点 202
- 与其它索引的比较 202
- 自动创建 202
- High_Group 索引
 - 请参见 HG 索引
- High_Non_Group 索引
 - 请参见 HNG 索引
- HIPAA 414
- HNG 索引 203
 - 建议用法 203
 - 其它索引 204
 - 缺点 203
 - 优点 203
 - 与其它索引的比较 203
- HOST 通信参数 108
- 函数
 - BFILE 252
 - 类型 7
 - 权限 284
 - 针对 BLOB 252
 - 针对 CLOB 252
- 候选键
 - 限制 319
- 环境变量
 - LANG 371
 - LC_ALL 371
 - SQLCONNECT 56
- 缓冲区不足
 - 缓冲区高速缓存 462
- 缓冲区高速缓存
 - IQ UTILITIES 命令 450
 - 监控器 450
 - 空间不足 449, 462
- 缓冲区空间
 - 连接参数 83
- 恢复
 - 版本控制 419
 - 常规 419
 - 服务器 419, 443
 - 强制 435
 - 事务 355, 419
 - 事务日志 356
 - 数据库检验 420
 - 数据库修复 420
 - 特殊模式 435
 - 替换写入服务器 423
 - 系统 355, 419
 - 系统故障 451
 - 泄漏的空间 436
- 恢复操作
 - 从错误中恢复 406
 - 关于 395
 - 恢复后的 SYSFILE 397
 - 检验备份 389, 397, 406
 - 排除其他用户 405
 - 确保顺序正确 402
 - 显示标头文件 405
 - 性能问题 413
 - 只读选择性 285
 - 至原始设备 399
- 恢复数据库
 - 检验备份 389, 397, 406
 - 重命名文件 399
- 恢复泄漏的空间 436
- 恢复帐户 290
- 回退
 - 数据库空间不足状况 146, 455
- 货币数据类型 274, 275

I

- IANA
 - 端口号 117
- iAnywhere JDBC 驱动程序
 - 连接到 ASA 数据库 46
- ICU 库 361
- IDLE 连接参数 94
- INSERT LOCATION 语句 252
- INSERT 权限 297
- INSERT 语句 211
 - VALUES 选项 251
 - 分区表 242
 - 关于 250
 - 和完整性 318
 - 性能 252
 - 增量 252
- INT 连接参数 95
- Interactive SQL
 - 另请参见 dbisql 124
 - 输出格式 228
- interfaces 文件
 - 添加条目 47

- IP 地址 78
 - Ping 468
- IP 通信参数 108
- IPv6 地址 115
- IPv6 支持 315
- IPX
 - 服务器配置 102
- IQ PAGE SIZE 143
- IQ UNIQUE
 - 更改值 175
 - 性能影响 174
- IQ UNIQUE 表选项 174
- IQ UTILITIES
 - 缓冲区高速缓存监控器 450
- IQ 唯一约束 326
- iq_bcp
 - LOAD TABLE USING FILE 替换 242
 - 不建议使用 242
- IQ_SYSTEM_MAIN
 - CREATE 权限 286
 - 大小准则 131
- IQ_SYSTEM_MSG 数据库空间 129
- IQ_SYSTEM_TEMP 470
 - CREATE 权限 286
- IQCHARSET 环境变量 369
- iqdemo 数据库 5
- IQDSEDIT 47
- iqdsn 命令 59
- iqgovern 开关 25
- IQLANG 370
- IQLANG 环境变量 369
- iqmc 开关 24
- iqmsg 日志 10
 - 存档 11
 - 回卷 10
 - 设置最大大小 10
- iqmsg 消息日志 7, 10
 - 存档 11
 - 回卷 10
 - 设置最大大小 10
- IQMsgMaxSize 服务器属性 10
- iqmsgnum 服务器启动开关 10
- IQMsgNumFiles 服务器属性 10
- iqmsgsz 服务器启动开关 10
- iqnumbercpus
 - 服务器开关 26
- iqpartition 开关 24
- iqtc 开关 24
- iqwmem 开关 25
- ISO_1 归类
 - 关于 364
- ISO_BINENG
 - 缺省归类 362, 372, 373
- ISO_BINENG 归类 362
- ISQL
 - 连接 47
- J**
- Java
 - 内存要求 25
 - 在 Sybase IQ 中使用 42
- jConnect 驱动程序
 - 连接到 IQ 数据库 46
- JDBC
 - 配置连接 47
- JDBC Sybase Open Client
 - 连接 42
- JDBC 连接
 - 关于 46
- 基表 172
 - 在连接索引中 256
- 即席连接
 - 性能 210
- 集成登录
 - 操作系统 69
 - 权限 285
 - 缺省用户 73
 - 使用 71
 - 网络方面 73
- 集合 200
- 记录连接 75
- 技术支持
 - MySybase 486
 - 将问题报告到 482
 - 联机帮助 486
 - 清单 486
- 技术支持的清单 486
- 加密
 - Encryption [ENC] 连接参数 92
 - FIPS 314
 - RSA 314
 - 列 314
 - 强 92
 - 数据库 314
 - 通信 119
- 剪裁尾随空白 237

索引

检查点

- 调整间隔 466
- 关于 352
- 权限 284
- 在恢复中 352
- 在系统恢复中 356
- 自动和显式 352

检验备份 389, 397, 406

- 错误报告 407
- 进度报告 407
- 增量 406

检验口令 295

键连接 218, 221

将 iqmsg 消息日志存档 11

接口库

- 连接 41

K

Kerberos

- 权限 285

Kerberos 鉴定 314

开放式数据库互连

- 请参见 ODBC

可变长度的数据传输 379

客户端

- Encryption [ENC] 连接参数 92
- 直接数据装载 242

客户端进程信息 78

空白

- 剪裁尾随空白 237
- 转换为空值 269

空间不足 457

- 缓冲区高速缓存 462

空间管理

- IQ 临时存储 146
- IQ 主存储区 146
- 等待空间状况 146, 455
- 数据库空间不足状况 146, 455

空值 267

- 备份 390
- 插入 251
- 缺省值 325
- 输出 230
- 在多列 HG 索引中 202
- 转换为 269
- 转换选项 258, 269

口令

- 丢失 290

分配 285

更改 295

规则 295

检验 295

区分大小写 373

缺省值 283

设置有效期 289

实用程序数据库 292

有效期 288

最小长度 295

块大小 145

块模式 379

块映射 422

快速投影索引 196

快照版本控制

- 另请参见版本控制 337

L

LANalyzer 469

LANG 环境变量 371

LANG 连接参数

- 说明 96

language_code 370

LazyClose 连接参数

- 说明 96

LC_ALL 环境变量 371

LCLOSE 连接参数

- 说明 96

LDAP 服务器

- LDAP 通信参数 110

LDAP 通信参数

- 说明 110

LF 索引 200

- 超过最大唯一值数 464

- 建议用法 200

- 其它索引 201

- 缺点 201

- 优点 201

- 与其它索引的比较 201

LF 通信参数

- 说明 111

Links 连接参数 84

LivenessTimeout 连接参数 97

LOAD TABLE

- BINARY 246

LOAD TABLE 语句 211

- BINARY FORMAT 246

- QUOTES 关键字 240

- QUOTES 选项示例 240
- STRIP 关键字 240
- USING CLIENT FILE 242
- USING FILE 子句 242
- 二进制格式数据文件 246
- 分区表 242
- 关于 238
- 完整性约束 243
- 性能 263
- LOB 252
- LOCAL 通信参数
 - 说明 110
- LocalOnly 通信参数
 - 说明 110
- LOG 通信参数
 - 说明 111
- LogFile
 - 连接参数 98
 - 通信参数 111
- LogFormat 通信参数
 - 说明 111
- LOGIN_MODE 数据库选项
 - 集成登录 70
- LogMaxSize 通信参数
 - 说明 82, 112
- LogOptions 通信参数
 - 说明 113
- LOPT 通信参数
 - 说明 113
- Low_Fast 索引
 - 请参见 LF 索引
- LSIZE 通信参数
 - 说明 82, 112
- LTO 连接参数 97
- 连接
 - BroadcastListener [BLISTENER] 通信参数 104
 - iAnywhere JDBC 驱动程序 46
 - Interactive SQL 49
 - jConnect 46
 - JDBC 42
 - LDAP 通信参数 110
 - 本地数据库 49
 - 到外部主机上的数据库 52
 - 多表 216
 - 防火墙 106
 - 概述 41
 - 管理 289
 - 记录 75
 - 建立 43
 - 临时 74
 - 嵌入式数据库 53
 - 权限 294
 - 删除权限 284
 - 使用连接进行更新 280
 - 使用数据源 54
 - 示例 48
 - 线性 222
 - 限制 396, 435
 - 限制并发 25
 - 星形 222
 - 性能影响 210
 - 学习路线图 42
 - 远程 46
 - 字符集 367
 - 最大数目 288
- 连接参数
 - CharSet 81
 - ConnectionPool 85, 98
 - Encryption [ENC] 92
 - Escape 93
 - IDLE 94
 - LANG 96
 - LazyClose [LCLOSE] 96
 - 不区分大小写 77
 - 关于 77
 - 缺省 55
 - 数据源 57
 - 优先级 78
 - 在连接字符串中 43
- 连接层次 213
- 连接关系
 - 定义 218
 - 指定 220
- 连接归集 73
- 连接句柄 9, 478
- 连接类型 221
- 连接列 218
- 连接配置文件 48
- 连接索引
 - 表中的列 214
 - 插入到 256
 - 创建 170, 216
 - 顶级表行数与关联表行数的比率 226
 - 更改列 176
 - 估计大小 226
 - 估计优势 226
 - 关于 212

- 候选键 319
- 基表 256
- 连接层次 213
- 连接关系 218
- 同步 217
- 修改基础表 225
- 在 Multiplex 中 174
- 在 Sybase Central 中创建 221
- 连接信息
 - IQ 消息文件 485
 - srvlog 文件 485
 - 请求日志 485
- 连接虚拟表 173
- 连接字符串
 - 表示 43
 - 字符集 367
- 连系内存
 - 设置 iqwmem 开关 25
- 列
 - 更改 176
 - 权限 284
 - 缺省值 320
 - 删除 176
 - 属性 323
 - 添加 176
 - 无法修复的错误 438
- 列分隔符
 - 装载格式选项 240
- 列宽
 - 插入问题 262
- 临时 dbfile 470
- 临时表
 - 版本控制 346
 - 关于 172
 - 装载 172
- 临时存储
 - 空间管理 146
 - 用于节省空间的选项 359
- 零
 - 转换为空值 269
- 路径名
 - 数据库 141
- 路由器
 - 广播 107
- 逻辑服务器 74
 - 在 Simplex 中 74

M

- main 存储
 - 空间管理 146
- max_days_since_login
 - 超过 291
- max_failed_login_attempts
 - 超过 291
- MAX_TEMP_SPACE_PER_CONNECTION 选项
 - 146
- MAXCONN 通信参数
 - 说明 114
- MaxConnections 通信参数
 - 说明 114
- MaxRequestSize 通信参数
 - 说明 114
- MAXSIZE 通信参数
 - 说明 114
- MESSAGE LOG
 - 内容 243
 - 示例 245
- Microsoft Access 463
 - ODBC 配置 60, 61
- Microsoft Visual Basic
 - ODBC 配置 60, 61
- Multiplex
 - 创建连接索引 174
 - 一致性检查 420
- Multiplex 服务器
 - 平衡负载 120
 - 停止 35
- Multiplex 数据库
 - 备份 376
 - 恢复 394
 - 验证 395
- MYIP 通信参数 114
- MySybase
 - 访问 486
 - 在线支持 486
- 敏感性
 - 游标行为 358
- 命令文件
 - dbisql 124
 - 创建数据库对象 124
- 命令行开关 19
 - 必需 21
 - 显示 20
- 命名冲突 446
- 命名管道 239

目录服务编辑器
请参见 IQDSEEDIT

N

NDIS
驱动程序 468
NEAREST_CENTURY 选项 267
net.cfg 文件 470
NetBIOS
服务器配置 102
NetWare
网络适配器设置 470
NEWID
缺省值 325
NEWPWD 连接参数 99
NOTIFY 选项
INSERT 语句 214
Novell 客户端软件 468
NULLS 选项
dbisql 230
内部版本号 8
内存
创建连系内存池 25
连接限制 312
用于目录存储高速缓存大小 25
内存消息
装载通知消息 474

O

ODBC
UNIX 的初始化文件 62
UNIX 支持 62
连接参数 77
配置数据源 60
数据源 57
ODBC 连接
关于 46
ODBC 数据源
与 jConnect 一起使用 46
ODBC 转换驱动程序
ODBC 配置 60
ODI 驱动程序 468
ON 子句连接 221
OPERATOR 权限
关于 284
授予 296
欧元符号
1252LATIN1 归类 365

P

params.cfg 文件 20
Password 连接参数 99
PERMS ADMIN 权限
关于 284
授予 296
PFP(1) 索引 196
PFP(2) 索引 196
Ping
TCP/IP 468
PIPE_NOT_CONNECTED 错误 239
PORT 通信参数 117
PreFetchOnOpen 通信参数
说明 115
PWD 连接参数 99
排序
归类 363
排序键
迁移 361
排序顺序
归类 361
配置
ODBC 数据源 60
配置参数
覆盖 448
配置文件
连接 48
使用 20
批量复制 228
平面 FP 索引 200
平面文件
装载 238
装载转换选项 258

Q

QUERY_TEMP_SPACE_LIMIT 选项 146
QUOTES
LOAD TABLE 关键字 240
启动 40
分配错误 435
故障排除提示 445
检查点错误 435
排除故障 435
启动参数 19
启动脚本 18
启动实用程序 18

索引

迁移

不再支持的归类 361

归类 361

排序键 361

嵌入式数据库 77, 81

Java 54

连接 53

启动 53

强加密

Encryption [ENC] 连接参数 92

强制恢复 435

服务器启动失败 435

过程 437

检测未分配的块 423

检测由多个对象共有的块 423

检测重复块 423

替换写入服务器 423

请求级别的记录 446

请求记录级别 485

请求日志文件 481

连接信息 485

使用 sa_get_request_profile 481

使用 sa_get_request_times 481

区分大小写

口令 373

连接参数 77

命令行 20

数据库名称和服务器名 23

区域设置

INSERT……LOCATION 371

关于 361, 363

确定 369

设置 369, 370

字符集 366

驱动程序

iAnywhere JDBC 驱动程序 46

jConnect JDBC 驱动程序 46

全局临时表

关于 172

权限

“授予权限”的权限 299

DBA 权限 283

INSERT 和 DELETE, 针对视图 310

OPERATOR 权限 284

PERMS ADMIN 权限 284, 285

RESOURCE 权限 285

USER ADMIN 权限 285

WITH GRANT OPTION 299

表 287, 297

冲突 311

单个 294

概述 281

管理 281

过程 300

继承 299, 302

口令 295

类型 4

连接 294

列表 313

命令行开关 27

视图 182, 297

授予口令 294

数据库空间管理 286

原始设备 140

在 Sybase Central 中 299

组 287, 302, 304

组成员资格 303

缺省

连接参数 55

缺省配置文件 20

缺省索引 196

缺省值

AUTOINCREMENT 324

NEWID 325

USER 特殊值 324

插入 321

常量表达式 325

创建 321

当前日期和时间 323

空值 325

列 320

在 Sybase Central 中创建 323

装载 321

字符串和数字 325

R

RCVBUFSZ 通信参数

说明 116

REAL 数据类型

匹配 Adaptive Server Enterprise 和 Sybase IQ
数据 272

ReceiveBufferSize 通信参数

说明 116

REFERENCES 权限 297

RELEASE SAVEPOINT 语句 353

resetclocks

sp_iqcheckdb 选项 423

- RESOURCE 权限
 - 不可继承 302
 - 关于 285
 - 授予 296
- RESTORE 语句
 - COMPATIBLE 子句 406
 - VERIFY 子句 389, 397, 406
 - 关于 398
 - 检验备份 389, 397, 406
- RetryConnectionTimeout 连接参数 100
- REVOKE 语句
 - 关于 301
- ROLLBACK TO SAVEPOINT 语句 353
- ROLLBACK 语句 353
- ROW LOG
 - 内容 244
 - 示例 245
- RSA 支持 314
- 日期时间索引
 - 请参见 DTTM 索引
- 日期数据类型
 - 匹配 Adaptive Server Enterprise 和 Sybase IQ 数据 274
- 日志文件
 - 服务器 33
 - 将连接信息相关联 485
- 软件版本号 8

- S**
- sa_get_request_profile
 - 分析请求日志文件 481
- sa_get_request_times
 - 分析请求日志文件 481
- SAVEPOINT 语句
 - 和事务 353
- schema
 - 创建 170
 - 更改 170
- SCSI 磁带备份 379
- SELECT DISTINCT 投影 192
- SELECT 权限 297
- SELECT 语句
 - 创建视图的限制 180
 - 分隔符 252
 - 连接索引 212
- SendBufferSize 通信参数
 - 说明 116
- ServerName 连接参数 90
- ServerPort 通信参数 117
- Service Manager
 - 启动服务器 19
- SESSIONS 通信参数 118
- SET 子句
 - UPDATE 语句 280
- SNDBUFSZ 通信参数
 - 说明 116
- SORT_COLLATION 选项 361
- SORTKEY 函数 361
- sp_iqcheckdb
 - DBCC 函数 420
 - DBCC_LOG_PROGRESS 395
 - DBCC_LOG_PROGRESS 选项 423
 - dumpdups 选项 423
 - dumpleaks 选项 423
 - dumpunallocs 选项 423
 - resetclocks 选项 423
 - 分配检验和修复 431
 - 分配模式 422
 - 分析分配问题 431
 - 分析索引问题 427
 - 检查分配 422
 - 检查模式 421
 - 检查数据库一致性 394
 - 检查索引和分配 421
 - 检验模式 421
 - 解释输出 432
 - 内部索引检查 421
 - 沙漏模式 423
 - 输出 423
 - 输出消息 440
 - 输出样本 423
 - 数据库检验 420
 - 索引检验和修复 427
 - 性能 421
 - 修复分配 431, 433
 - 修复索引 427
 - 语法 420
 - 运行时间 421
 - 重置分配映射 423
 - 资源问题 426
- sp_iqcheckoptions 存储过程 146
- sp_iqdbspace
 - 数据库空间使用信息 166
- sp_iqdbspaceinfo
 - 数据库空间使用信息 167

索引

- sp_iquestdbspaces
 - 估计数据库空间要求 132
- sp_iquestjoin 226
 - 估计连接索引空间要求 132
- sp_iquestspace
 - 估计数据库空间要求 132
- sp_iqindexinfo
 - 显示索引信息 169
- sp_iqrebuildindex 430
- sp_iqstatus
 - 输出样本 471
 - 在故障排除中使用 471
- sp_iqtransaction
 - 确定禁止写入者 465
- SPACE ADMIN 权限
 - 关于 285
 - 授予 297
- SQL 日志
 - 夏令时 13
- SQLCONNECT 环境变量
 - 连接 56
- srvlog
 - 将连接信息相关联 485
- srvlog 文件
 - 连接信息 485
- START ROW ID 选项
 - INSERT 语句 252
- Start 参数
 - 嵌入式数据库 54
- start_iq
 - 参数 447
 - 故障排除 447
 - 命令无法运行 447
- start_iq 实用程序 18
- StartLine 连接参数 101
- STRIP
 - LOAD TABLE 关键字 240
- SUM 函数 200
- Sybase Central 40
 - 创建表 171
 - 创建数据库空间 163
 - 创建用户 294
 - 创建组 303
 - 更改表 176
 - 和权限 299
 - 简介 124
 - 列缺省值 323
 - 列约束 328
 - 启动 49
 - 删除视图 182
 - 显示问题 461, 467
 - 在组中添加用户 304
- Sybase Control Center
 - 创建数据库空间 163
 - 创建用户 295
 - 创建组 303
 - 将成员添加到组中 304
- Sybase IQ
 - 与 Adaptive Server Enterprise 匹配数据类型 270
- SYS 组 306
- SYSCOLAUTH 视图
 - 权限 314
- SYSCOLUMN 表
 - 完整性 336
- SYSDUMMY 表
 - 权限 313
- SYSDFILE 表
 - 恢复后的 file_name 397
- SYSFOREIGNKEY 表
 - 完整性 336
- SYSGROUP 表
 - 权限 313
- SYSGROUPS 视图
 - 权限 314
- SYSPROCAUTH 视图
 - 权限 314
- SYSROCPERM 表
 - 权限 313
- SYSTABAUTH 视图
 - 权限 314
- SYSTABLE 表
 - 完整性 336
- SYSTABLEPERM 表
 - 权限 313
- SYSTEM 数据库空间 127
- SYSUSERAUTH 视图
 - 权限 314
- SYSUSERLIST 视图
 - 权限 314
- SYSUSERPERM 表
 - 权限 313
- SYSUSERPERMS 视图
 - 权限 314
- SYSVIEW 视图
 - 视图信息 182

- 删除
 - 列缺省值 321
 - 视图 181
- 设备类型
 - 数据库 126
- 时间数据类型
 - 匹配 Adaptive Server Enterprise 和 Sybase IQ 数据 274
- 实体完整性
 - 关于 319
 - 强制 329
- 实用程序
 - CP874toUTF8 361
 - 不建议使用的 iq_bcp 242
 - 事务日志 404
- 实用程序数据库
 - 安全性 291
 - 关于 13
 - 连接 293
 - 启动 292
 - 设置口令 292
 - 用于创建数据库的口令 293
- 使用列缺省值 320
- 事件
 - DBSpaceLogger 459
 - 监控磁盘空间使用情况 459
 - 监控空间使用情况 459
- 事件处理程序
 - ConnectFailed 290
- 事务
 - 保存点 353
 - 定义 337
 - 关于 337
 - 回退 354
 - 结束 338
 - 启动 338
 - 游标 358
 - 在恢复中 355, 419
 - 子事务和保存点 353
- 事务处理
 - 关于 337
- 事务日志
 - 在系统恢复中 356
 - 重命名 404
- 视图
 - SELECT 语句限制 180
 - 安全 308
 - 插入和删除 181
 - 创建 180
 - 权限 182, 284, 285, 287
 - 删除 181, 182
 - 使用 179, 180
 - 所有者 286
 - 修改 181
 - 与永久表的差异 179
- 输出格式
 - dbisql 228
- 输出重定向 229, 230
- 数据
 - 导出 227, 229
 - 导入 227
 - 客户端 242
 - 删除 280
 - 输入和输出格式 228
 - 提取 230
 - 无效 317
 - 在事务中 343
 - 重复 317
 - 装载 227
- 数据操纵语言 135
- 数据定义
 - 创建 170
- 数据定义语言
 - 并发规则 348
 - 关于 123
- 数据库
 - DBCC 一致性检查程序 394
 - schema 创建 170
 - SQL Anywhere 128
 - 安全概述 4
 - 初始大小 131
 - 初始化 137
 - 创建 137
 - 创建对象 170
 - 创建和删除权限 293
 - 创建数据库所需的特权 126
 - 大小 143
 - 非常大 14
 - 检查一致性 394
 - 块大小 145
 - 连接到 41, 46
 - 命名 22
 - 命名冲突 446
 - 启动 38
 - 启动权限 39
 - 启动时命名 22
 - 嵌入式 77, 81

- 权限 4, 27, 281
- 缺省特性 140
- 删除 148
- 设计 123
- 设置概述 125
- 实用程序 13
- 使用 **Interactive SQL** 管理 124
- 使用对象 123
- 使用实用程序数据库创建 292
- 所有者角色 4
- 停止 39
- 文件位置 139
- 无意中打开 435
- 相对路径名 141
- 卸载 38
- 修复 419, 443
- 选择位置 141
- 选择性恢复 285
- 验证 394
- 页大小 143
- 移动 139
- 移动文件 399
- 预分配空间 126
- 只读恢复操作 285
- 字符集 366
- 数据库访问
 - 限制 435
- 数据库服务器
 - 从“开始”菜单启动 19
 - 关于 3
 - 禁止启动 80
 - 连接 46
 - 名称开关 22
 - 命令行开关 19
 - 启动 17
 - 启动时命名 23
 - 停止 34, 37, 101
 - 远程 46
 - 在 Windows 上启动 19
 - 在命令提示符下启动 18
 - 作为 Windows 服务 19
- 数据库管理员
 - 定义的 283
 - 另请参见 DBA 283
- 数据库空间
 - IQ_SHARED_TEMP 127, 129
 - IQ_SYSTEM_MSG 129
 - IQ_SYSTEM_TEMP 127
 - SYSTEM 127
 - 创建 162
 - 创建时的文件位置 139
 - 创建示例 164
 - 定义 126
 - 更改大小 165
 - 管理示例 164
 - 恢复至原始设备 399
 - 监控空间使用情况 459
 - 空间不足错误消息 455
 - 命名 162
 - 权限 285
 - 删除 163
 - 数据库空间不足状况 146, 455
 - 脱机 432
 - 文件位置 139
 - 显示使用信息 166
 - 显示索引信息 166
 - 在原始设备上添加 140
 - 只读 150
- 数据库实用程序 56
- 数据类型
 - Adaptive Server Enterprise unichar 273
 - Adaptive Server Enterprise unitext 273
 - Adaptive Server Enterprise univarchar 273
 - FLOAT 272
 - REAL 272
 - 货币 274, 275
 - 匹配 Sybase IQ 和 Adaptive Server Enterprise 270
 - 在 Adaptive Server Enterprise 和 Sybase IQ 之间转换 271
 - 在创建表时指定 171
 - 整数 272
 - 转换 257
 - 装载过程中的转换 258
 - 字符 273
- 数据链接层
 - 故障排除 468
- 数据提取
 - ASCII 233
 - 二进制 233
 - 二进制/交换 233
 - 关于 230
 - 控制访问 233
 - 选项 230
 - 选项列表 230
- 数据完整性
 - 概述 317
 - 列缺省值 320

- 系统表中的规则 336
- 约束 319
- 数据修改
 - 权限 228
- 数据源
 - ODBC 57
 - UNIX 62
 - 创建命令行 59
 - 关于 57
 - 连接 54
 - 配置 60
 - 与 jConnect 一起使用 46
- 数据源名称
 - ODBC 60
- 死锁
 - 检测 450
 - 解决 450
- 所有者
 - 关于 286
 - 角色 4
- 索引
 - FP(1) 197
 - FP(1) 和 FP(2) 196, 197
 - FP(2) 197
 - FP(3) 197
 - sp_iqcheckdb 错误 432, 439
 - TEXT 索引 210
 - 表中的数目太多 463
 - 并行创建 191
 - 不一致 430
 - 查找 197
 - 创建 170, 190
 - 创建 TEXT 索引 210
 - 磁盘空间使用情况 194
 - 多列 HG 和空值 202
 - 关于 187
 - 检测逻辑问题 439
 - 检验和修复 427
 - 简介 183
 - 快速投影 196
 - 平面 FP 200
 - 缺省 196, 197
 - 删除 185
 - 删除损坏的内容 439
 - 添加和删除 189
 - 无法修复的错误 438
 - 系统表中的 183
 - 修复 430
 - 选择索引类型 210

- 验证 184
- 在装载表后添加 211
- 重建 175
- 重命名 185
- 自动创建 174
- 最大唯一值数 464
- 索引类型
 - LF 200
 - 关于 187
 - 建议 192
 - 选择 210
 - 选择标准 191
- 锁
 - 管理争用 465
- 锁定
 - DDL 操作 348
 - 表 347
 - 自动 290
- 锁争用
 - 管理 350

T

- TCP/IP
 - BroadcastListener [BLISTENER] 通信参数 104
 - LDAP 通信参数 110
 - 测试 469
 - 服务器端口号 117
 - 服务器配置 102
 - 故障排除 468
 - 通过防火墙连接 106
- TDS 通信参数 119
- Telnet
 - TCP/IP 测试 469
- TEMP_EXTRACT_NULL_AS_EMPTY 选项 235
- TEMP_EXTRACT_NULL_AS_ZERO 选项 235
- TEXT 索引 210
 - 创建 210
- TIME 数据类型
 - 优化装载 263
- TIME 索引 206
 - 建议用法 206
 - 其它索引 209
 - 缺点 209
 - 优点 209
 - 与其它索引的比较 209
- TIMEOUT 通信参数 119

索引

TO 通信参数 119

特权

定义数据库对象 126

用于插入和删除 228

提交事务

读取事务时间的影响 344

提取数据

关于 230

选项 230

选项列表 230

通信

Encryption [ENC] 连接参数 92

参数 102

故障排除 467

通信参数

Broadcast [BCAST] 103

BroadcastListener 104

Certificate 105

Certificate_Password 105

ClientPort 106

DatabaseName [DBN] 107

DOBROADCAST 107

HOST 108

LDAP [LDAP] 110

LocalOnly [LOCAL] 110

LogMaxSize [LSIZE] 82, 112

LogOptions [LOPT] 113

MaxConnections 114

MYIP 114

PreFetchOnOpen 115

ReceiveBufferSize [RCVBUFSZ] 116

SendBufferSize [SNDBUFSZ] 116

ServerPort 117

SESSIONS 118

TDS 119

TIMEOUT 119, 120

关于 77

说明 114

通知计数 191

通知消息 473

同步

Multiplex 174

关于 217

投影

优化 196

U

UID 连接参数 102

UNC 连接参数 101

Unconditional 连接参数 101

unicar Adaptive Server Enterprise 数据类型 273

Unicode

字符集 361, 365

Unilib 库 361

UNIQUE 约束 326

unitext Adaptive Server Enterprise 数据类型 273

univarchar Adaptive Server Enterprise 数据类型

273

UNIX

ODBC 支持 62

UPDATE 语句

使用 279

使用连接操作 280

USER 权限

授予 297

USER 特殊值

缺省值 324

Userid 连接参数 102

util_db.ini 文件 292

V

VALUES 选项

INSERT 语句 251

VARCHAR 数据

零长度单元 265

VERIFY 通信参数 120

Visual Basic

ODBC 配置 60, 61

VLDB

管理 14

W

WarehouseArchitect

关于 123

WD 索引 205

建议用法 205

缺点 206

优点 206

WHERE 子句

UPDATE 语句 280

对索引选择的影响 192

连接索引 212

Windows 服务 17

WITH GRANT OPTION 子句 299

WORM 存储 414

- 外键
 - 插入数据 256
 - 可选 332
 - 现有未受强制的 332
 - 限制 319
- 完整性
 - 概述 317
 - 列缺省值 320
 - 约束 319
- 完整性约束违规
 - 日志示例 243
- 网络号
 - 在 IPX 地址中 107
- 网络适配器
 - 驱动程序 468
- 网络通信
 - 命令行开关 102
- 网络协议
 - 故障排除 467
- 尾随空白
 - 剪裁 237
- 未受强制的外键 332
- 文件
 - 将输出重定向到 229, 230
- 文件大小
 - 控制 458
- 问题
 - 报告 482
- 物理层
 - 故障排除 469
- X**
- 系统表 6
 - SYSDATE 397
 - 关于 179
 - 权限 313
 - 索引 183
 - 显示 147
 - 用户和组 313
- 系统不响应 449
- 系统故障
 - 从故障中恢复 451
- 系统级备份 392
- 系统视图
 - 权限 313
 - 完整性 336
- 夏令时
 - SQL 日志文件 13
- 消息日志文件 13
- 线程
 - 不足 460
 - 生成堆栈跟踪 450
 - 装载期间的使用情况 279
- 限定对象名 306
- 消息
 - 内存通知 474
 - 数据库空间不足状况 455
 - 在消息日志中记录 7
 - 重定向到文件 229
- 消息记录 7, 10
- 消息日志 7
 - IQ_SYSTEM_MSG 数据库空间 129
 - 备份 12, 393
 - 备份存档 12, 393
 - 夏令时 13
- 消息日志管理 7, 10
- 消息文件
 - 连接信息 485
- 协议
 - 故障排除 467
 - 开关 29
- 写入访问权限已禁止
 - 管理争用 465
 - 确定禁止写入者 465
- 写入服务器
 - 替换 423
- 信号 461
- 星形连接 222
- 性能
 - 版本控制的影响 356
 - 插入 252
 - 查询和装载 463
 - 从平面文件装载 259
 - 即席连接 210
 - 索引 183
 - 约束的影响 318
 - 装载数据 276
- 修复
 - 表 439
 - 分配 431
 - 数据库 419, 443
 - 索引 439
- 修改
 - 列缺省值 321
- 修改和删除列缺省值 321
- 虚拟备份
 - 分离式 391

索引

- 封装式 390
- 选项
 - DBCC_LOG_PROGRESS 395, 423
 - MAX_TEMP_SPACE_PER_CONNECTION 146
 - QUERY_TEMP_SPACE_LIMIT 146
 - 设置 312
- 选项值
 - 截断 231, 232
- 选择驱动程序
 - 使用 iAnywhere JDBC 驱动程序 46
 - 使用 jConnect JDBC 驱动程序 46
- 选择性恢复操作 285

Y

- 演示数据库 5
- 页大小 143
 - 开关 28
 - 目录 28
- 一致性检查
 - Multiplex 420
- 以太网 470
- 引号
 - 在 SQL 中 306
- 用户
 - 创建单个 294
 - 登录失败 290
 - 解锁 291
 - 权限 285
 - 锁定 289, 291
 - 添加到组 304
 - 在 Sybase Central 中创建 294
- 用户 ID
 - 创建 294
 - 管理 281
 - 列表 313
 - 缺省值 283
 - 删除 301
 - 在消息日志中 9
- 用户定义数据类型
 - CHECK 条件 327
- 用户鉴定
 - 权限 285
- 用户名
 - 请参见 用户 ID
- 游标
 - ODBC 配置 61
 - 持有 345, 359

- 定位操作 359
- 滚动 359
- 连接限制 312
- 敏感性 358
- 消息记录 360
- 在事务中 358
- 语言支持
 - 多字节字符集 365
 - 概述 361
 - 关于 361
 - 归类 368
- 预分配空间
 - Catalog 存储 128
- 元数据
 - 在 Catalog 存储中 128
- 原始设备 126
 - 创建数据库 140
 - 恢复至 399
 - 权限 140
 - 实用程序数据库 292
 - 添加数据库空间 140
 - 在 Windows 上命名 142
- 远程数据
 - bulk load 242
 - 装载 242
- 远程数据访问
 - 代理表 252
- 约束
 - 对性能的影响 318
- 约束违规
 - 日志示例 243

Z

- zr 日志文件 481
- 在 LOAD TABLE 中 267
- 在 Sybase Central 中使用列缺省值 323
- 诊断工具 470
 - sa_server_option 479
 - sp_iqcheckdb 478
 - sp_iqcheckoptions 479
 - sp_iqconnection 479
 - sp_iqcontext 479
 - sp_iqdbstatistics 478
 - sp_iqstatus 471
 - 记录服务器请求 479
 - 检查服务器启动选项 479
 - 检查数据库选项 479
 - 通信问题 482

- 整数数据类型
 - 匹配 Adaptive Server Enterprise 和 Sybase IQ 272
- 帧类型 470
- 只读数据库空间 150
- 只读硬件
 - 备份 387
 - 示例 415
- 指定
 - 驱动程序 46
- 重定向
 - 输出到文件 229, 230
- 重命名数据库文件 399
- 重置登录策略 291
- 主键
 - AUTOINCREMENT 324
 - 插入数据 256
 - 创建 177, 178
 - 多列 330
 - 实体完整性 329
- 转换
 - 插入时 257
 - 插入选项 258
 - 导入时的错误 276
 - 在 Adaptive Server Enterprise 和 Sybase IQ 之间 271
- 转换驱动程序
 - ODBC 配置 60
- 转换选项
 - DATE 265
 - DATE 格式规范 266
 - DATETIME 267, 268
 - 平面文件装载 258
 - 替换零长度单元 265
 - 性能 263
- 装载
 - 线程使用情况 279
- 装载 schema
 - 推荐的数据库大小 131
- 装载时设置列 267
- 装载数据
 - Adaptive Server Enterprise 数据 252
 - ASCII 转换选项 264
 - 并发规则 347
 - 错误 464
 - 大对象 252
 - 概述 227
 - 格式选项 240
 - 记录约束违规 243
 - 监控空间使用情况 459
 - 客户端数据 242
 - 列缺省值 321
 - 另请参见插入 258
 - 命名管道 239
 - 通知消息 473
 - 完整性约束违规 243
 - 文件规范 239
 - 性能 276, 463
 - 需要的特权 228
 - 转换错误 276
 - 转换选项 258
- 装载性能
 - 锁争用 350
- 装载选项 241
- 装载优化 263
- 装载转换
 - 请参见 转换选项
- 资源规划
 - iqnumbercpus 开关 26
- 子查询
 - 标量值 180
- 子事务
 - 和保存点 353
- 字符串和数字缺省值 325
- 字符集
 - ISO_BINENG 362
 - Unicode 361, 365
 - UTF8 362
 - 避免转换 368
 - 编码 361, 362
 - 单字节 362
 - 多字节 365
 - 关于 361
 - 连接参数 81
 - 显示 369
 - 选择 368
 - 转换 372
- 字符集转换
 - 错误消息 366
 - 打开和关闭 372
 - 禁用 372
 - 启动服务器 372
- 字符数据类型
 - Adaptive Server Enterprise unichar 273
 - Adaptive Server Enterprise unitext 273
 - Adaptive Server Enterprise univarchar 273
 - 匹配 Adaptive Server Enterprise 和 Sybase IQ 数据 273

索引

自定义归类 361

自然连接 221

组

Sybase Central 303

 成员资格 303

创建 302

管理 302

没有口令 305

权限 284, 287, 304