



管理：数据库

SAP Sybase IQ 16.0 SP03

文档 ID: DC02008-01-1603-01

最后修订日期: 2013 年 12 月

© 2013 SAP 股份公司或其关联公司版权所有, 保留所有权利。

未经 SAP 股份公司明确许可, 不得以任何形式或为任何目的复制或传播本文的任何内容。本文包含的信息如有更改, 恕不另行事先通知。

由 SAP 股份公司及其分销商营销的部分软件产品包含其它软件供应商的专有软件组件。各国的产品规格可能不同。

上述资料由 SAP 股份公司及其关联公司 (统称“SAP 集团”) 提供, 仅供参考, 不构成任何形式的陈述或保证, 其中如若存在任何错误或疏漏, SAP 集团概不负责。与 SAP 集团产品和服务相关的保证仅限于该等产品和服务随附的保证声明 (若有) 中明确提出之保证。本文中的任何信息均不构成额外保证。

SAP 和本文提及的其它 SAP 产品和服务及其各自标识均为 SAP 股份公司在德国和其它国家的商标或注册商标。

如欲了解更多商标信息和声明, 请访问: <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark>。

目录

数据库管理工作流	1
调度数据定义任务准则	1
数据定义的系统特权	1
设备选择	2
运行数据库服务器	3
启动服务器	3
服务器启动实用程序	4
服务器作为 Windows 服务	5
命令行开关	5
服务器活动日志	29
为服务器日志文件命名	31
UNIX 日志文件	31
关闭数据库服务器	32
何时停止和重新启动服务器	32
停止服务器	32
停止服务器所需的权限	35
关闭操作系统会话	35
启动和停止数据库	35
数据库启动准则	36
停止数据库	37
启动 iqdemo 数据库	37
连接到服务器和数据库	39
连接路线图	39
连接方式	40
从 Interactive SQL 连接到演示数据库	40
数据库状态对本地连接的影响	41
连接到本地服务器上正在运行的数据库	41
在 UNIX 上从 Interactive SQL 连接到数据库	41
从 UNIX 系统连接	42
从 Windows 系统连接	42
连接到嵌入式数据库	43

TCP/IP 协议	44
使用数据源进行连接	47
缺省连接参数	48
从 SAP Sybase IQ 实用程序连接	49
受支持的连接接口	49
连接状态	50
连接参数的工作原理	51
连接字符串的格式	51
应用程序如何传递连接参数	52
ODBC 数据源中的连接参数	52
Interactive SQL 连接	53
“连接”对话框	54
打开“连接”对话框 (Interactive SQL)	54
用于连接的驱动程序	55
文件数据源	55
使用 ODBC 管理器创建文件数据源	56
文件数据源和文本编辑器	56
从其它数据库连接	56
避免在 UNIX 上发生端口号冲突	57
如何测试连接	58
集成登录	58
使用集成登录	60
安全性问题：不受限制的数据库访问	61
临时公共选项可以提高安全性	62
集成登录的网络方面	62
缺省集成登录用户	63
连接归集	63
临时连接	65
逻辑服务器配置	66
在 Simplex 中连接	66
Multiplex 中的连接	66
如何终止连接	67
连接记录	68
创建并管理数据库	69

创建数据库	69
使用 SQL 创建数据库	69
IQ 主存储库和 IQ 临时存储空间管理	76
删除数据库	77
从数据库中断开其他用户的连接	77
在嵌入式 SQL 中与数据库断开连接	78
在 Interactive SQL 中与数据库断开所有连接	78
显示 Interactive SQL 中的系统对象	78
在 Interactive SQL 中设置数据库选项	78
管理数据存储	79
空间分配	79
dbspace 类型	79
目录存储库	81
IQ_SYSTEM_MAIN dbspace	81
其它用户主 dbspace	81
IQ 临时 dbspace	81
IQ 共享临时 dbspace	82
IQ 消息文件 dbspace	83
数据库空间	83
IQ 存储库和临时存储库的空间要求	84
主存储库和临时存储库的大小调整准则	84
将 Windows 访问设置为原始设备	86
查看原始设备的访问权限	87
设置对原始设备的权限	87
在 Windows 中设置原始设备的符号链接	88
估计所需的空间和 dbspace	89
dbspace 管理示例	89
数据存储	95
dbfile 属性和操作	95
dbspace 属性和操作	97
只读和读写 dbspace 和文件	100
直连存储的高速缓存 Dbspace	106
创建高速缓存 Dbspace	107
删除高速缓存 Dbspace	108

将存储添加到高速缓存 Dbspace	108
删除高速缓存 Dbspace 中的存储	109
禁用及重新启用高速缓存 Dbspace	109
更改高速缓存 Dbspace Dbfile 的只读/读写状态	110
监控高速缓存 Dbspace	110
管理数据库对象	115
SAP Sybase IQ 数据库设计	115
视图管理	115
创建视图	116
使用视图的准则	117
修改视图的准则	118
针对视图的权限	118
如何删除视图	118
系统视图中的视图信息	119
表管理	119
创建表的准则	119
更改表的准则	123
删除表的准则	124
创建主键	124
创建外键	125
系统视图中的表信息	125
表分区	125
限制	126
范围分区	127
散列分区	128
散列范围分区	129
索引数据	131
索引概述	131
索引类型比较	131
包含 (WD) 索引	132
比较 (CMP) 索引	134
日期 (DATE) 和日期时间 (DTTM) 索引	134
快速投影 (FP) 索引	138

High_Group (HG) 索引	139
High_Non_Group (HNG) 索引	141
Low_Fast (LF) 索引	142
TEXT 索引	143
时间 (TIME) 索引	143
选择索引的标准	144
索引中的唯一值数目	144
查询类型	145
按查询类型建议的索引	146
按列值建议的索引	147
磁盘空间使用情况	147
索引中的数据类型	148
索引类型组合	150
创建索引	150
并行列索引创建	151
索引装载的状态消息	151
在插入数据后添加列索引	151
执行 CREATE INDEX 语句组	151
运行索引顾问	152
重命名索引	152
查看索引	152
系统视图中的索引信息	153
删除索引	153
在删除外键约束时保留索引	153
优化连接性能	154
实施数据完整性	155
数据完整性概述	155
哪些情况会导致数据无效	155
有效数据的规则和检查	155
更改数据库内容的语句	156
数据完整性工具	156
实施完整性约束的语句	157
列缺省值有助于数据完整性	157
支持的缺省值	157

缺省值限制	157
创建列缺省值	158
更改列缺省值	158
删除列缺省值	158
支持的列缺省值	158
日期、时间和时间戳缺省值	160
USER 缺省值	160
IDENTITY 或 AUTOINCREMENT 缺省值	161
NEWID 缺省值	161
空值缺省值	161
字符串和数字缺省值	162
常量表达式缺省值	162
表约束和列约束	162
对列或表的 UNIQUE 约束	162
对列的 IQ UNIQUE 约束	163
列的 CHECK 条件	163
用户定义数据类型的 CHECK 条件	164
列的 CHECK 条件	164
表的 CHECK 条件	165
删除表中的 Check 条件	165
实体和参照完整性	166
如果客户端应用程序破坏了实体完整性	166
参照完整性	167
并发操作	170
禁用参照完整性检查	172
系统表中的完整性规则	172
校验目录存储库索引	173
管理事务和版本控制	175
事务	175
查看事务活动	176
隔离级别	179
事务阻塞	180
启用连接阻塞	181
禁用连接阻塞	181

BLOCKING 选项	181
设置阻塞超时阈值	182
BLOCKING_TIMEOUT 选项	182
事务阻塞死锁	183
版本	189
表级快照版本控制	189
行级快照版本控制	193
查看事务的版本控制类型	199
临时表版本控制	199
调查锁争用对性能的影响	200
(不建议使用) -contention	200
检查点、保存点和事务回退	201
Checkpoints	202
事务内的保存点	202
事务回退	203
系统恢复	204
事务信息如何帮助进行恢复	204
备份的并发执行	205
事务中的游标	205
游标和版本控制	205
游标敏感性	206
游标滚动	206
持有游标	206
定位操作	206
游标的消息记录	207
远程事务	207
远程事务限制	207
创建过程和批处理	209
过程	209
创建过程	209
更改过程	210
调用过程	210
删除过程	210
执行存储过程的特权	210

在参数中返回过程结果	210
在结果集中返回过程结果	211
显示过程信息	211
显示过程参数信息	211
过程中的游标	212
使用 IQ UTILITIES 创建存储过程	214
用户定义的函数	216
创建用户定义的函数	216
调用用户定义的函数	222
删除用户定义的函数 (SQL)	223
执行用户定义的函数的权限	223
授予执行用户定义函数的能力 (SQL)	224
批处理	224
控制语句	227
复合语句	228
复合语句中的声明	228
原子复合语句	229
过程的结构	229
过程中允许的 SQL 语句	229
过程的参数声明	230
将参数传递给过程的方法	231
如何将参数传递给函数	231
过程结果	232
使用 RETURN 语句返回值	232
将结果作为过程参数返回的方法	233
从过程中以结果集形式返回的信息	234
返回多个结果集	236
过程的可变结果集	237
错误和警告处理	237
缺省的错误处理	238
使用 ON EXCEPTION RESUME 处理错误	239
缺省的警告处理	240
异常处理程序	241
嵌套的复合语句和异常处理程序	243

过程中的事务和保存点	245
隐藏过程、函数、触发器、事件或视图的内容	246
过程、触发器、事件和批处理中允许使用的语句	247
批处理中使用的 SELECT 语句	247
过程、触发器、用户定义函数和批处理中使用的 EXECUTE IMMEDIATE	248
使用调度和事件自动完成任务	251
使用调度和事件自动完成任务	251
事件	251
调度	251
系统事件	252
事件处理程序	254
事件处理程序	257
检索有关事件或调度的信息	258
审计数据库事件	259
dbtran 数据库管理实用程序	259
AUDITING 选项 [database]	262
故障排除提示	263
在线支持来源	263
特定情况的解决方案	263
服务器恢复和数据库修复的决策流程	263
服务器操作问题	264
数据库连接问题	271
Interactive SQL (dbisql) 问题	272
资源问题	273
处理问题	279
性能问题	283
网络通信故障排除	284
使用兼容协议	284
使用最新驱动程序	284
在两次重新启动之间关闭计算机电源	284
逐层诊断协议堆栈	284
测试 TCP/IP 协议堆栈	285
诊断布线问题	286

检查常见网络通信问题	286
诊断工具	287
恢复至新临时文件拓扑	287
sp_iqstatus 存储过程	287
解释通知消息	290
sp_iqcheckdb 存储过程	294
检查数据库和服务器启动选项值	294
查找当前执行的语句	294
记录服务器请求	295
用于收集诊断信息的连接	299
诊断通信问题	299
向技术支持报告问题	299
使用 getiqinfo 收集诊断信息	299
由 getiqinfo 收集的信息	301
使 .srvlog 和 .iqmsg 文件的连接信息相关联	302
支持 Web 站点	303
清单：技术支持信息	303
附录：连接参数和通信参数参考	305
连接参数	305
AppInfo 连接参数 [Appinfo]	306
AutoPreCommit 连接参数 [AutoPreCommit]	307
AutoStart 连接参数 [Astart]	308
AutoStop 连接参数 [Astop]	308
CharSet 连接参数 [CS]	309
CompressionThreshold (COMPTH) 连接参数	309
CommBufferSize 连接参数 [CBSize]	310
CommBufferSpace 连接参数 [CBSpace]	311
CommLinks 连接参数 [Links]	312
ConnectionName 连接参数 [CON]	313
ConnectionPool 连接参数 [CPOOL]	314
DatabaseFile 连接参数 [DBF]	314
DatabaseName 连接参数 [DBN]	315
DatabaseSwitches 连接参数 [DBS]	316
DataSourceName 连接参数 [DSN]	317

DBKEY 连接参数 [DBKEY]	317
DisableMultiRowFetch 连接参数 [DMRF]	318
EngineName 连接参数 [ENG]	318
EncryptedPassword 连接参数 [ENP]	319
Encryption 连接参数 [ENC]	319
Escape 连接参数 [ESCAPE]	320
FileDataSourceName 连接参数 [FileDSN]	321
Idle 连接参数 [IDLE]	321
Integrated 连接参数 [INT]	322
Language 连接参数 [LANG]	323
LazyClose 连接参数 [LCLOSE]	323
LivenessTimeout 连接参数 [LTO]	324
Logfile 连接参数 [LOG]	324
LogicalServer 连接参数 [LS]	325
LoginRedirection 连接参数 [REDIRECT]	325
NewPassword (NEWPWD) 连接参数	326
NodeType 连接参数	327
Password 连接参数 [PWD]	328
PrefetchBuffer 连接参数 [PBUF]	328
PrefetchRows 连接参数 [PROWS]	329
RetryConnectionTimeout (RetryConnTO) 连接 参数	330
ServerName 连接参数 [ENG]	330
StartLine 连接参数 [START]	330
Unconditional 连接参数 [UNC]	331
Userid 连接参数 [UID]	331
网络通信参数	332
Broadcast 通信参数 [BCAST]	333
BroadcastListener 通信参数 [BLISTENER]	334
ClientPort 通信参数 [CPort]	334
DatabaseName 通信参数 [DBN]	335
DoBroadcast 通信参数 [DBROAD]	336
Host 通信参数 [IP]	337
Identity 通信参数	338

Identity_Password 通信参数	338
LDAP 通信参数 [LDAP]	339
LocalOnly 通信参数 [LOCAL]	339
LogFile 通信参数 [LOG]	340
LogFormat 通信参数 [LF]	340
LogMaxSize (LSIZE) 协议选项	341
LogOptions 通信参数 [LOPT]	341
MaxConnections 通信参数 [MAXCONN]	342
MaxRequestSize (MAXSIZE) 协议选项	343
MyIP 通信参数 [ME]	343
PreFetchOnOpen 通信参数	344
ReceiveBufferSize 通信参数 [RCVBUFSZ]	345
SendBufferSize 通信参数 [SNDBUFSZ]	345
ServerPort 通信参数 [PORT]	346
Sessions 通信参数	347
TDS 通信参数	347
Timeout 通信参数 [TO]	348
VerifyServerName 通信参数 [Verify]	349
附录: SQL 语句和选项参考	351
数据库选项	351
数据库选项的作用域和持续时间	351
DEDICATED_TASK 选项	352
LOG_CONNECT 选项	352
MIN_PASSWORD_LENGTH 选项	353
VERIFY_PASSWORD_FUNCTION 选项	353
SQL 语句	356
ALTER DBSPACE 语句	356
ALTER INDEX 语句	360
ALTER LOGICAL SERVER 语句	363
ALTER LS POLICY 语句	364
ALTER TABLE 语句	368
CHECKPOINT 语句	383
COMMIT 语句	384
CREATE DATABASE 语句	385

CREATE DOMAIN 语句	395
CREATE INDEX 语句	397
CREATE LOGICAL SERVER 语句	404
CREATE LS POLICY 语句	406
CREATE TABLE 语句	410
DROP 语句	425
DROP LOGICAL SERVER 语句	428
GRANT INTEGRATED LOGIN 语句	429
IQ UTILITIES 语句	430
LOCK TABLE 语句	434
ROLLBACK 语句	436
SAVEPOINT 语句	437
SET OPTION 语句	438
STOP DATABASE 语句 [Interactive SQL]	440
索引	443

管理工作流

创建 SAP® Sybase® IQ 数据库属于更大的安装过程的一部分，此过程从安装开始到数据库对用户可用时结束。

1. 安装并配置 SAP Sybase IQ。

安装客户端环境和服务器环境。请参见《安装和配置指南》。

2. 启动并连接到数据库服务器。

启用数据库控制并打开与数据库的通信通道。使用 **start_iq** 实用程序、SAP Control Center、Windows “开始” 菜单或配置文件。

3. 创建数据库。

创建 IQ 存储库和目录存储库。可以使用 SAP Control Center、SQL 语句 **CREATE DATABASE** 或 **iqinit** 实用程序。

4. 在数据库中创建表。

使用 **CREATE TABLE** 语句或 SAP Control Center 的创建表向导。

5. 为表创建索引。

使用 **CREATE INDEX** 语句或 SAP Control Center 的创建索引向导。也可以在创建表时自动创建某些索引。

6. 将数据装载到表中。

使用 **LOAD TABLE** 语句从文件中批量装载数据，或使用 **INSERT** 语句从现有数据库中抽取数据行。请参见《管理：装载管理》。

调度数据定义任务准则

一旦数据库存在并可供其他用户进行访问，就必须执行附加数据定义操作，如添加或修改表或索引。

将数据定义操作调度为在数据库使用率低时进行。当您创建或更改某个表时，会禁止所有其他用户对该表进行读取或写入，虽然只持续很短的时间。

有关数据定义期间的并发规则的详细信息，请参见 DDL 操作的表锁。

数据定义的系统特权

必须具有相应的系统特权才能执行数据定义任务。

- 具有 **SERVER OPERATOR** 系统特权可以执行所有的数据定义任务。还可以将此系统特权授予其他用户供其执行特定任务。这包括向其他用户授予 **SERVER OPERATOR** 系统特权的能力。

- 要创建任意数据库对象，需具有针对此类型对象的 **CREATE ANY OBJECT** 系统特权。
- 对象创建后，您将成为该对象的所有者。对象所有者将自动获得对该对象执行所有操作的系统特权，并能够授予其他用户更新表中信息的系统特权。

对所需系统特权具有管理权限的用户和对象所有者可将所需的系统特权授予各用户。还可以使用 **-gu** 命令行选项来设置创建或删除数据库时所需的权限级别。

设备选择

将数据库和数据库对象存储在设备上。在所有的平台上，这些设备可以是操作系统文件，也可以是磁盘分区（称为原始分区）。创建数据库时，**SAP Sybase IQ** 将自动确定设备是原始分区还是磁盘文件。

在生产环境中，原始分区安装可能会提高处理性能和恢复能力。但文件系统可提供更简便的设备管理，并且可能更适合用于开发环境。

注意： 目录存储库和事务日志不能位于原始分区上。

运行 SAP Sybase IQ 数据库服务器

每个 SAP Sybase IQ 数据库都在一个服务器上运行。根据平台和管理工具，SAP Sybase IQ 提供多种方法来启动和停止服务器。

启动服务器

启动服务器的不同方式。

适用于任何平台的服务器启动方法
有几个方法可以启动数据库服务器。

要从此处启动服务器...	应执行的操作...
服务器启动实用程序	<p>在命令行中输入 start_iq 命令。</p> <p>语法：</p> <pre>start_iq [options]</pre> <p>如果输入多个命令行选项，则必须将其输入到一行中（不含回车符）。</p> <p>为便于使用，在启动服务器时指定数据库名。这样将同时启动数据库和服务器。</p> <p>省略数据库名时，会启动服务器而不启动数据库。如果省略数据库名，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以及让客户端软件来控制数据库的使用。</p>
配置文件	使用 SAP Sybase 提供的配置文件启动服务器和 iqdemo 数据库。
服务器启动命令	将服务器启动命令放在快捷方式或桌面图标中。
ODBC 数据源	将服务器启动行包含在数据源中。
实用程序命令	将服务器启动行包含在实用程序命令中。
Interactive SQL	从 Interactive SQL 发出 SQL 命令以启动其它服务器。
SAP Control Center	要在 SAP Control Center 中启动，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680 。

Windows 服务器启动方法

有几个方法可以启动 Windows 服务器。

要从此处启动服务器...	应执行的操作...
命令行	运行 stop_iq 或 dbstop 命令。
桌面	右键单击系统任务栏中的“服务器图标”，然后选择 Start server name 。
Windows “开始”菜单	单击“开始” > “所有程序” > “Sybase” > “Sybase IQ 16.0” > “启动 Sybase IQ 演示数据库”。
服务管理器	如果您将服务器作为 Windows 服务来启动： 1. 在“控制面板”上，选择“管理工具” > “服务”。 2. 选择 SAP Sybase IQ 服务，然后单击 Start the service 。

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，为清晰起见，这些选项显示在不同的行上，与它们写入配置文件时的方式一致。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（即不含任何回车符）。

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整 SAP Sybase IQ 行为和性能的方法。

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，服务器从数据库名称获取其名称，您也可以为服务器指定其它名称。

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称。

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以及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，服务器在缺省端口 2638 上启动。

服务器日志

服务器启动后，其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 平台上安装时将设置环境变量 %IQLOGDIR16%，服务器日志位于路径 %IQLOGDIR16%\servername.nnnn.srvlog 中，其中 *nnnn* 是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。

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也能继续运行。

有关这一功能及其它 Windows 特定功能的详细信息，请参见安装和配置指南 指南。

自动服务器启动

使用 SAP Sybase IQ 服务管理器定义启动 SAP Sybase IQ 服务器的服务。

然后，可以配置该服务以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

命令行开关定义您的 SAP 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”。

您通过命令行选项设置的某些值也可以通过 **SET OPTION** 命令更改。

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

可将命令行开关组存储到配置文件中。

如果使用大量命令行开关，则可将它们存储在配置文件中，然后在服务器命令行中调用该文件。在配置文件中指定开关与在命令行中指定开关的方法相同，但在文件中您可以在多行输入开关。

配置文件列表

SAP Sybase IQ 提供可用作模板的配置文件。

表 1. 配置文件

文件名	位置	使用
default.cfg	\$IQDIR16/scripts (UNIX)、 %IQDIR16%\scripts (Windows)	通用配置文件。此文件用于 start_iq 和 Multiplex 启动的缺省选项。SAP 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 scripts 目录中) 做出的任何更改都会由在该文件更改后创建的所有数据库继承。
iqdemo.cfg	\$IQDIR16/data/demo (UNIX)、 %IQDIR16%\data\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

您可以使用预定义配置文件作为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**，且高速缓存为 32MB，检查点时间间隔为 20 分钟。这样可以允许任何人启动或停止数据库并装载数据，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
```

可使用以下语句执行所有这些命令行选项：

```
start_iq @mydb.cfg
```

注意： 当使用 **DBSTOP** 命令停止服务器时，必须指定与启动服务器时所使用的参数相同的参数；如果使用配置文件启动服务器，请确保您可以在需要这些参数来停止服务器时找到它们。

SAP Sybase IQ 文档中的命令行示例格式

为保证清晰，整个 SAP Sybase IQ 文档中的示例都会将多个命令行开关分别显示在不同的行中，与将其写入到配置文件中时相同。

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 > **start_iq** 数据库服务器启动实用程序中介绍的大多数命令行开关都可选，但您必须指定 **-n** 开关才能有效运行 SAP Sybase IQ。

注意： 在所有 64 位平台上，建议使用 **-c 48M**。

如果使用 TCP/IP 连接服务器，则还应包括网络连接参数。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。如果指定的端口号正在使用，则服务器将无法启动。

缺省配置文件

缺省配置文件 (default.cfg) 包含所有必需的启动开关。

default.cfg 文件：

- 提供 UNIX **start_iq** 命令所使用的 params.cfg 文件的源
- 启动 Windows 服务的服务器

要在配置文件中覆盖除 **-n servername** 外的任意开关，可在 **start_iq** 命令行中指定新开关。

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。

可在使用 SAP Sybase IQ 提供的脚本创建演示数据库时自动创建该文件。

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称，即使用引号引起来也不能使用。

例如，不允许以下名称：

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 SAP Sybase 强烈建议在每个 SAP Sybase 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。

缺省服务器名称

如果没有提供服务器名称，则第一个启动的数据库将提供缺省服务器名称。

区分大小写和命名约定

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

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。

数据库名称

可以通过在数据库文件后提供一个 **-n** 开关来对数据库命名。

例如，以下命令行可启动数据库并对其命名：

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

可使用昵称命名数据库。

服务器名

可通过在第一个数据库文件之前提供一个 **-n** 开关来命名服务器。其余的参数可从 default.cfg 文件添加。

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唯一。这样可防止无意间连接到错误的服务器。主机名与端口号的组合不能唯一标识服务器。给服务器名称附加唯一标识符是一种有效的约定。指定唯一的服务器名称在多用户的联网环境中尤为重要，在这种环境中共享内存会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 SAP Sybase IQ 定位服务器，而不管使用的是何种字符集，请在服务器名称中仅包含七位 ASCII（较低页）字符。

指定服务器名称使您可在不装载任何数据库的情况下启动数据库服务器。以下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：虽然您可以通过使用缺省服务器名称启动服务器，但最好同时包括服务器名称和数据库名称，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称，才能在不启动特定数据库的情况下启动服务器。

适用于性能的命令行选项

若干命令行选项会影响数据库服务器性能。

大多数的性能命令行选项都可控制 IQ 存储库上操作的资源，它们可能会对性能产生重要的影响。仅影响可用于目录存储库上操作的资源的选项可能对总体性能产生轻微的影响。

另请参见

- 数据库选项（第 351 页）

内存选项

SAP 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同其它可在服务器运行后进行设置的选项一同决定了有多少内存可供使用。

主缓冲区高速缓存的缺省 IQ 缓冲区高速缓存大小为 16MB，临时高速缓存大小为 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。

使用以下两种方式中的一种来设置 IQ 主存储库和临时存储库的缓冲区高速缓存大小：

- （建议）要在服务器范围内设置当前服务器会话的缓冲区高速缓存大小，请指定数据库启动实用程序 **start_iq** 选项 **-iqmc**（主缓冲区高速缓存大小）和 **-iqtc**（临时高速缓存大小）。
- 要设置数据库的高速缓存大小，请以 **main_cache_memory_mb** 或 **temp_cache_memory_mb** 参数使用 **sa_server_option** 存储过程。

但如果设置的 IQ 缓冲区高速缓存大小高于系统可以承受的大小，则 SAP Sybase IQ 将无法打开数据库。

借助服务器选项（**-iqmc** 和 **-iqtc**），您还可以使用系统所允许的最大内存量，唯一的限制就是计算机的物理内存量。因此，在 64 位系统中使用 **-iqmc** 和 **-iqtc**，这两个服务器选项不会覆盖通过 **sa_server_option** 进行的设置。

通过 **-iqmc** 和 **-iqtc** 设置的高速缓存大小将应用于所有已经启动的数据库，直至服务器关闭为止。所以，举例来说，如果将 **-iqmc** 和 **-iqtc** 都设置为 500 (MB) 并在服务器

启动时启动一个数据库，稍后在同一台服务器上启动另一个数据库，则至少需要有 2GB 可供两个主高速缓存和两个临时高速缓存使用。

-iqmc iqsrv16 服务器选项

指定主 IQ 存储缓冲区高速缓存大小（以 MB 为单位）。

语法

-iqmc size

缺省值

64MB

注释

此开关将覆盖缺省值 64MB，并应用至从服务器启动到关闭这一期间启动的所有数据库。换句话说，如果在服务器启动时启动一个数据库而稍后又启动另一个数据库，则需要有 2 * **-iqmc** 的容量才能供主缓冲区高速缓存使用。

大容量内存需求是指所有可用物理内存的三分之一。为确保主存储库具有足够内存，请将 **-iqmc** 启动参数设置为所有可用物理内存的三分之一。

始终指定 size 值，但无需包括测量单位；例如，指定 **-iqmc 32** 而非 **-iqmc 32MB**。如果指定测量单位，**start_iq** 将忽略此开关；与此不同的是，SQL Anywhere 需要测量单位。

注意： 请勿使用 SAP Sybase IQ 服务器运行多个数据库。

-iqtc iqsrv16 服务器选项

指定 IQ 临时存储高速缓存大小（以 MB 为单位）。

语法

-iqtc size

缺省值

64MB

Remarks

此开关将覆盖缺省值 64MB，并应用至从 SAP Sybase IQ 服务器启动到关闭这一期间启动的所有数据库。换句话说，如果在服务器启动时启动一个数据库而稍后又启动另一个数据库，则需要有 2 * **-iqtc** 的容量才能供临时高速缓存使用。通常情况下，请避免使用 SAP Sybase IQ 服务器运行多个数据库。

大容量内存需求是指所有可用物理内存的三分之一。为确保 IQ 临时存储高速缓存具有足够内存，请将 **-iqtc** 启动参数设置为所有可用物理内存的三分之一。

始终指定 **size** 值，但无需包括测量单位；例如，指定 **-iqtc 32** 而非 **-iqtc 32MB**。如果指定测量单位，**start_iq** 将忽略此开关；与此不同的是，SQL Anywhere 需要测量单位。

确保 **IQ_SYSTEM_TEMP** dbspace 等于或大于 **-iqtc**。

IQ 缓冲区分区控制

缺省情况下，将启用基于 CPU 数进行的缓冲区分区。调整分区以提高装载或查询性能。

-iqpartition iqsrv16 服务器选项

设置 IQ 主缓冲区高速缓存和 IQ 临时缓冲区高速缓存中分区数的数量。

语法

-iqpartition num

注释

指定 IQ 主缓冲区高速缓存和 IQ 临时缓冲区高速缓存中的分区数。该数值必须为 2 的幂。缺省情况下，可用值包括：0（缺省值）、1、2、4、8、16、32、64、128、256。缺省情况下，SAP Sybase IQ 自动按 $\text{number_of_cpus}/8$ 计算分区数，计算结果将舍入到最接近的 2 的幂，最大值为 64。通过调整高速缓存分区数，也许可以提高性能。**-iqpartition** 开关为 SAP Sybase IQ 服务器设置此值，并覆盖 **Cache_Partitions** 数据库选项所设置的值。

- 从数据库中排除 **jConnect™ for JDBC™** 系统对象。要使用 **jConnect JDBC** 驱动程序来访问系统目录信息，需要 **jConnect** 目录支持（缺省情况下会安装）。指定此选项后，只要不访问系统信息，就仍然可以使用 **JDBC**。可在以后使用 **ALTER DATABASE** 语句添加 **jConnect** 支持。

CACHE_PARTITIONS 选项

设置要用于主缓冲区高速缓存和临时缓冲区高速缓存的分区数。

允许值

0、1、2、4、8、16、32、64

- **0** –（缺省）SAP Sybase IQ 自动将分区数计算为 $\text{number_of_cpus}/8$ ，并舍入为最接近的 2 的乘方（最大为 64）。
- **1** – 只有一个分区；此值将禁用分区操作。
- **2 到 64** – 分区数；必须是 2 的乘方。

缺省值

0

范围

只能在数据库 (PUBLIC) 级别设置选项。

必须具有 **SET ANY SYSTEM OPTION** 系统特权才能设置此选项。下次启动数据库服务器时将对该当前数据库生效。

注释

通过减少锁争用，分区缓冲区高速缓存有时可提高具有多个 CPU 系统的性能。通常，应当信赖 SAP Sybase IQ 自动计算的值，该值以系统中的 CPU 数为依据。但是，如果您发现多 CPU 配置中的装载或查询性能比预期低，则可以通过为 **CACHE_PARTITIONS** 设置其它值来提高装载或查询性能。

CPU 数和平台都可能对理想的分区数产生影响。可尝试使用不同的值以确定适用于您的配置的最佳设置。

您为 **CACHE_PARTITIONS** 设置的值同时适用于主缓冲区高速缓存和临时缓冲区高速缓存。对于每个缓冲区高速缓存，绝对最大分区数都是 64。

-iqpartition start_iq 服务器选项在服务器级别设置分区限制。如果在服务器启动时指定 **-iqpartition**，则它将覆盖 **CACHE_PARTITIONS** 设置。

分区数不会影响其它缓冲区高速缓存设置。而且，也不会影响 IQ 监视器收集的统计信息；所有分区的统计信息均作为单个值进行累计和报告。

示例

在具有 100 个 CPU 的系统中，如果您未设置 **CACHE_PARTITIONS**，则 SAP Sybase IQ 会自动将分区数设置为 16：

100 个 CPU/8 = 12，舍入为 16。

在此设置中，共有 16 个分区用于主缓冲区高速缓存，16 个分区用于临时高速缓存。

在具有 100 个 CPU 的相同系统中，要将分区数显式设置为 8，请指定：

```
SET OPTION "PUBLIC".CACHE_PARTITIONS=8
```

并发用户开关

您的许可证可设置并发用户的绝对数量。所需的 **-gm** 开关可以限制特定服务器上的并发用户连接数。

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

在 Windows 中，**start_iq** 可以通过以下公式来计算此参数值并对其进行设置：

```
gn_value >= gm_value * 1.5
```

将 **-gn** 值设置为 **-gm** 值的至少 1.5 倍。指定最小值 25。线程总数不得超过特定于平台的最大值；有关详细信息，请参见 **-iqmt num**。

-gm iqsrv16 数据库服务器选项

限制数据库服务器的并发连接数。

语法

```
iqsrv16 -gm integer ...
```

缺省值

个人服务器的缺省值为 10。网络数据库服务器的缺省值约为 32766，但操作过程中服务器所使用的内部临时连接会减少此值。

适用于

所有操作系统和数据库服务器。

注释

定义服务器的连接限制。如果此数字大于授权和内存约束所允许的数量，则该值无效。受计算机资源所限，网络服务器的连接数通常低于缺省值。

数据库服务器允许在连接限制之上建立一个额外的 DBA 连接，以便允许具有 DROP CONNECTION 系统特权的用户连接到该数据库服务器并断开其它连接。

-gn iqsrv16 服务器选项

设置多个用户同时运行时将用于目录存储库和连接的执行线程数。

语法

```
-gn integer
```

注释

此参数适用于所有操作系统和服务器。每个连接对每个请求使用一个线程，当请求完成后，该线程返回到池中供其它连接使用。由于连接每次只能执行一个请求，因此没有连接能够一次使用多个线程。

Java 应用程序可使用多个线程，是此规则的一个例外。Java 应用程序 中的每个线程都是一个数据库服务器执行线程。

在 Windows 上，在 **start_iq** 中指定此参数。若要计算其值，请使用：

```
gn_value >= gm_value * 1.5
```

将 **-gn** 值设置为 **-gm** 值的至少 1.5 倍。指定最小值 25。线程总数不得超过特定于平台的最大值；有关详细信息，请参见 **-iqmt num**。

并发查询开关

要将用户连接数限制为低于许可证所允许的范围，使用 **-iqgovern** 开关来控制查询的使用。

-iqgovern 开关可以控制特定服务器上的并发查询数。

使用 **-iqgovern** 开关来协助 IQ 优化缓冲区数据分页输出至磁盘的操作并避免过量使用内存。**-iqgovern** 的缺省值是 $(2 \times \text{CPU 数}) + 10$ 。您可能需要试验确定理想值。对于具有大量活动连接的站点，请将 **-iqgovern** 设置得略低些。

-iqgovern iqsrv16 服务器选项

设置服务器允许的并发 查询数。

语法

-iqgovern num

缺省值

RECOVERY_TIME 数据库选项的设置，该值缺省为 2 分钟。

Remarks

并发查询数与连接数不同。单个连接可具有多个打开的游标。**-iqgovern** 可帮助 SAP Sybase IQ 优化缓冲区数据分页输出至磁盘的操作并避免过量使用内存。此开关的缺省值等于计算机 CPU 数乘 2 再加上 10。您可能会发现其它值（如 CPU 数乘 2 再加上 4）将会提供更好的吞吐量，尤其是在连接有大量用户时。

绑定内存开关

-iqwmem 开关仅在特定的 UNIX 平台上创建“绑定”内存池。

警告！ 仅在内存中有足够的内存可专用于此目的时使用此开关。否则，可能会导致性能大幅下降。

-iqwmem iqsrv16 服务器选项

在 HP 和 Sun UNIX 系统上创建一个“绑定”内存池。

语法

-iqwmem size

Remarks

此内存处于锁定状态，因此操作系统不能 对其进行分页。指定内存的大小（以 MB 为单位）。仅当有足够内存可专用于此用途时，才应使用此开关。否则，可能导致性能大幅下降。

处理线程数

SAP 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，将不同数量的内核线程分配给每个用户连接。增加线程数可能会提高性能。

使用 **-iqmt** 开关可以设置 SAP Sybase IQ 可使用的处理线程数。

-iqmt iqsrv16 服务器选项

指定要创建的 SAP Sybase IQ 线程数。

语法

-iqmt num

注释

缺省值为：对于前四个 CPU，每个 CPU 有 60 个线程，对于其余的 CPU，每个 CPU 有 50 个线程，供系统使用的至少有 3 个线程，再加上用于数据库连接和后台任务的线程。例如，在具有 12 个 CPU 和 10 个连接的系统上： $60 * 4 + 50 * (\text{numCPUs} - 4) + \text{numConnections} + 6 = 656$ 。

num 的最小值为 $\text{num_conn} + 3$ 。

服务器线程总数不得超过 4096。

处理器数

在多处理器计算机中，使用 **-gt** 选项来设置数据库服务器进行目录存储库操作时所用的处理器数。

缺省情况下，会使用所有可用的处理器。

-gt iqsrv16 数据库服务器选项

设置最多可使用的物理处理器数（不超过授权使用的最大值）。此选项仅在多处理器系统中有用。

语法

```
iqsrv16 -gt num-processors ...
```

允许值

- **num-processors** – 介于 1 与以下各项中的最小值之间的整数：

计算机上物理处理器的数量

授权服务器使用的最大 CPU 数（如果 CPU 授权有效）

如果指定的 **-gt** 值在此范围之外，则会强制使用下限或上限值。

注释

对于每存取设备授权，网络数据库服务器使用计算机上可用的所有 CPU。对于基于 CPU 的授权，网络数据库服务器仅使用您得到授权的处理器。网络数据库服务器可以使用的 CPU 数量还可能会受到 SAP Sybase IQ 版本的限制。

若指定 **-gt** 选项的值，则数据库服务器将调整其处理器掩码（如果在该硬件平台受支持），以将数据库服务器限制为仅使用设置数量的物理处理器运行。如果许可数据库

服务器使用 n 个处理器，则缺省情况下，服务器将使用 n 个物理处理器的所有逻辑处理器（超线程和内核）运行。可使用 `-gtc` 选项对此行为做出进一步限制。

`-gt` 选项不能与 `-gta` 选项一起使用。

目录存储库高速缓存大小

使用 `-c` 开关来设置目录存储库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（64 位系统）和 32MB（32 位系统）。

任何小于 10000 的高速缓存大小都会被认为以 KB 为单位（1K = 1024 字节）；任何等于或大于 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 。

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的大小来计算缺省的初始高速缓存大小。当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。

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

- 设置 `-c` 参数，或者，
- 使用 `-cl` 和 `-ch` 参数为目录存储库高速缓存大小设置特定的上限和下限

请勿为这些参数指定其它组合。

IQ 存储库的高速缓存大小不依赖于目录高速缓存大小。

`-c iqsrv16` 服务器选项

设置为高速缓存目录存储库页和其它服务器信息而保留的初始内存。

语法

-c cache-size [$k|m|g|p$]

注释

如果在高速缓存中留出了内存，数据库服务器会使用额外内存来缓存数据库页面。所有小于 10000 的高速缓存大小均假定为以 KB 为单位（1K = 1024 字节）。所有等于或大于 10000 的高速缓存大小均假定为以字节为单位。您还可以指定高速缓存大小 nK ， nM 或 nP （1M = 1024 KB），其中 P 是物理系统内存的百分比。

在 `default.cfg` 文件中，`-c` 和 **start_iq** 的缺省值对于 Windows 平台为 32MB（`-c 32M`），对于 UNIX 平台则为 48MB（`-c 48M`）。请使用此缺省值或将 `-c` 设为更大值。

您可以使用 `%` 来代替 P ，但由于大多数非 UNIX 操作系统使用 `%` 作为环境变量转义字符，因此必须转义 `%` 字符。例如，要使用 20% 的物理系统内存，请指定：

```
start_iq -c 20%% ...
```


请不要在 **-ch** 或 **-cl** 所在的配置文件或命令行中使用 **-c**。有关详细信息，请参见 **-ch** 高速缓存大小选项和 **-ca 0** 选项。

*未指定 **start_iq** 服务器选项 **-c** 时的初始目录存储高速缓存大小*

如果没有为 **start_iq** 服务器选项 **-c** 开关提供值（在命令行中或使用 **start_iq** 缺省值），数据库服务器将计算初始目录存储高速缓存分配

服务器按照以下方式计算高速缓存分配：

- 数据库服务器使用 **32MB** 作为缺省高速缓存大小的最小值。
- 数据库服务器计算运行时特定的最小缺省高速缓存大小，该数值为以下各项中的较小者：
 - 计算机物理内存的 **25%**，或者
 - 命令行中指定的主数据库文件大小的总和。计算时不包括主数据库文件以外的其它 **dbspace**。如果没有指定文件，则该值为零。
- 数据库服务器分配这两个计算值中的较大者。

CPU 开关数

SAP Sybase IQ 启动命令中的 **-iqnumbercpus** 开关指定可用于 IQ 的 CPU 数，从而覆盖用于资源计划的实际 CPU 数。

该参数的缺省值为 CPU 的总数，但可用值范围为 1 到 512。

注意： 仅在以下计算机上使用 **-iqnumbercpus**：

- 使用 Intel® CPU 并启用超线程。将 **-iqnumbercpus** 设置为可用 CPU 的数量。
- 已使用操作系统实用程序将 SAP Sybase IQ 限定为计算机内 CPU 的子集

将 **-iqnumbercpus** 值设置为大于可用的 CPU 数可能会影响性能。

影响计时的选项

有若干命令行选项控制出现特定数据库服务器事件的时间。

检查点时间间隔

SAP Sybase IQ 使用检查点生成恢复数据库所需的参照点和其它信息。

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

当数据库服务器运行多个数据库时，使用由第一个启动的数据库指定的检查点时间，除非被 **-gc** 开关替换。如果输入值 0，则使用缺省值 20 分钟。

恢复时间

使用 **-gr** 参数可设置数据库服务器从系统失败中恢复所花费的最多分钟数。

当数据库服务器运行多个数据库时，使用由第一个启动的数据库指定的恢复时间，除非该值被 **-gr** 替换。

-gc iqsrv16 数据库服务器选项
设置检查点之间的最大间隔。

语法

```
iqsrv16 -gc minutes ...
```

缺省值
60 分钟

允许值

- **minutes** - 缺省值为 `checkpoint_time` 数据库选项的设置，该值缺省为 60 分钟。如果输入值 0，则使用缺省值 60 分钟。

适用于

所有操作系统和数据库服务器。

注释

使用此选项设置数据库服务器在不对各个数据库执行检查点操作的情况下运行的最长时间（以分钟为单位）。

检查点通常比指定时间发生得更为频繁。

网络性能选项

使用 **-gb**（Windows 中的数据库进程优先级）和 **-p**（最大包大小）选项可帮助调整网络性能。

-gb iqsrv16 数据库服务器选项
设置服务器进程优先级。

Windows 语法

```
iqsrv16 -gb { idle | normal | high | maximum } ...
```

Unix 语法

```
iqsrv16 -gb level ...
```

允许值

- **Unix** - 在 Unix 上，*level* 是一个介于 -20 和 19 之间的整数。Unix 上的缺省值与父进程的精确值相同。较低的 *level* 值代表更加合适的调度优先级。有关设置精确值的所有限制均适用于 **-gb** 选项。例如，在大多数 Unix 平台上，只有 root 用户能够降低进程的优先级（例如，将其从 0 更改为 -1）。

- **Windows** – 在 Windows 上，**normal** 和 **high** 为常用设置。提供值 **idle** 是为了使值集完整。而值 **maximum** 可能会影响计算机的运行。

适用于

Windows、Unix。

-p iqsrv16 数据库服务器选项
设置通信包的最大大小。

语法

```
iqsrv16 -p integer ...
```

缺省值

7300 字节

适用于

所有操作系统和数据库服务器。

注释

最小值为 500 个字节，最大值为 65535 个字节。

您可以通过设置 **CommBufferSize (CBSIZE)** 连接参数来更改连接的通信缓冲区大小。

控制特权的命令行选项

数据库服务器启动命令 **start_iq** 包含一些选项，可用于设置某些数据库和服务器功能的特权级别。

启动和停止数据库的开关

使用 **-gd** 选项，将仅允许在其所连接的数据库中具有以下特定特权的用户在运行的服务器上启动或停止数据库：

- **DBA** – (缺省值) 只有具有 **SERVER OPERATOR** 系统特权的用户才可以启动额外的数据库。
- **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。用户仍必须被授予特权才能访问自己启动的特定数据库。
- **NONE** – 任何人都无法通过 **Interactive SQL** 在运行的服务器上启动或停止数据库。

注意： 如果启动服务器时未设置 **-gd ALL**，则只有具有 **SERVER OPERATOR** 系统特权的用户才可以启动服务器上的其它数据库。这意味着用户无法连接到在服务器启动时、或随后由具有 **SERVER OPERATOR** 系统特权的用户启动服务器时未启动的数据库。但是，不具有 **SERVER OPERATOR** 系统特权的用户也可以停止数据库。因此，建议您将生产数据库上的此设置更改为 **DBA**。

创建和删除数据库的开关

-gu 选项仅允许在其所连接的数据库中具有特定特权的用户创建和删除数据库。

- **DBA** – 只有具有 **SERVER OPERATOR** 系统特权的用户才可以创建和删除数据库。
- **ALL** (缺省值) – 任何用户都可以创建和删除数据库。
- **NONE** – 任何用户都不能创建和删除数据库。
- **UTILITY_DB** – 只有能够连接到 `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) – 只有具有 **SERVER OPERATOR** 系统特权的用户才能停止服务器。
- **ALL** – 任何用户都可以停止服务器。
- **NONE** – 任何用户都不能使用 **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

-gl 选项仅允许在数据库中具有特定特权的用户使用 **LOAD TABLE** 装载数据。

- **DBA** – 任何具有 **LOAD ANY TABLE**、**ALTER ANY TABLE** 或 **ALTER ANY OBJECT** 系统特权的用户都可以装载数据。
- **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。

-gd iqsrv16 数据库服务器选项

设置启动或停止正在运行的数据库服务器上的数据库所需的特权。

语法

```
iqsrv16 -gd { DBA | all | none } ...
```

允许值

- **DBA** – 只有具有 **SERVER OPERATOR** 系统特权的用户才能启动或停止数据库。
- **all** – 所有用户都可以启动或停止数据库。不建议用于远程客户端可以访问的网络服务器。
- **none** – 除数据库服务器自行启动和停止外，不允许启动和停止数据库。

缺省值

网络数据库服务器的缺省设置为 **DBA**。大小写语法均可接受。

适用于

所有操作系统和数据库服务器。

注释

此选项指定用户在通过数据库服务器装载新数据库文件时或在正在运行的数据库服务器上停止数据库时所需的特权级别。

此选项设置为 **DBA** 时，客户端应用程序必须使用同一服务器上运行的另一数据库的现有连接才能启动或停止数据库。无法使用 **DatabaseFile** 连接参数启动未在运行的数据库。

使用 **StartDBPermission** 服务器属性，可获得 **-gd** 选项的设置：

```
SELECT PROPERTY ( 'StartDBPermission' );
```

停止数据库服务器所需的特权由 **-gk** 选项指定。

示例

下面这组步骤说明了怎样将 **-gd** 选项用于网络数据库服务器。

1. 启动网络数据库服务器：

```
iqsrv16 -su mypwd -gd DBA -n my_server
```

2. 从 **Interactive SQL** 连接到实用程序数据库：

```
dbisql -c "UID=DBA;PWD=mypwd;DBN=utility_db"
```

3. 启动数据库：

```
START DATABASE 'demo.db';
```

4. 连接到数据库：

```
CONNECT USING 'DBN=demo;UID=DBA;PWD=sql';
```

-gk iqsrv16 数据库服务器选项

设置停止数据库服务器所需的特权。

语法

```
iqsrv16 -gk { DBA | all | none } ...
```

允许值

- **DBA** - 只有具有 **SERVER OPERATOR** 系统特权的用户才能停止数据库服务器。这是网络服务器的缺省设置。
- **all** - 关闭数据库服务器不需要任何特权。
- **none** - 数据库服务器无法停止。

适用于

所有操作系统和数据库服务器。

注释

-gd 数据库服务器选项应用于 **dbstop** 实用程序以及以下语句：

- ALTER DATABASE *dbname* FORCE START 语句。
- STOP DATABASE 语句

-gl iqsrv16 服务器选项

设置使用 **LOAD TABLE** 装载数据时所需的权限。

语法

-gl level

Remarks

LOAD TABLE 语句用于从数据库服务器计算机中读取文件。要控制使用这些语句对文件系统进行访问，可使用 **-gl** 命令行开关来控制这些语句所需的数据库权限级别。*level* 是：

- DBA - 只有具有 **LOAD ANY TABLE**、**ALTER ANY TABLE** 或 **ALTER ANY OBJECT** 系统特权的用户才能装载数据。
- ALL - 所有用户都可以装载数据。
- NONE - 无法装载数据。

可以对这些选项使用大写和小写语法。

对于使用 **start iq** 启动的服务器，缺省设置为 **all**；对于其它服务器，缺省设置为 **dba**。为了与早期版本保持一致，请在所有系统中使用 **all** 值。在 `iqdemo.cfg` 和 `default.cfg` 配置文件中使用 **all** 设置。

-gu iqsrv16 数据库服务器选项

设置执行数据库文件管理语句（例如用于创建或删除数据库）所需的特权。

语法

```
iqsrv16 -gu { all | none | DBA | utility_db } ...
```

允许值

-gu 选项	效果	适用于
all	不建议使用此选项。任何用户都能够执行文件管理语句。	任何数据库（包括实用程序数据库）
none	不允许执行文件管理语句。	任何数据库（包括实用程序数据库）
DBA	只有具有 SERVER OPERATOR 系统特权的用户才能执行文件管理语句	任何数据库（包括实用程序数据库）
utility_db	只有能够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

缺省值

DBA

适用于

所有操作系统和数据库服务器。

注释

限制可执行以下数据库文件管理语句的用户：

- ALTER DATABASE dbfile ALTER TRANSACTION LOG
- CREATE DATABASE 语句
- CREATE DECRYPTED DATABASE 语句
- CREATE DECRYPTED FILE 语句
- CREATE ENCRYPTED DATABASE 语句
- CREATE ENCRYPTED FILE 语句
- DROP DATABASE 语句
- RESTORE DATABASE 语句。

如果指定 `utility_db`，则只能从实用程序运行这些语句。如果指定 `DBA`，则只能由具有 `SERVER OPERATOR` 系统特权的用户运行这些语句。如果未指定上述两项，则所有用户均无法执行这些语句。

示例

要防止文件管理语句被使用，请使用 `-gu` 选项的 `none` 特权级别启动数据库服务器。以下命令启动一个数据库服务器并将其命名为 `TestSrv`。它装载 `mytestdb.db` 数据库，但禁止任何用户使用该服务器来创建或删除数据库或者执行任何其它文件管理语句，而不管用户是否具有资源创建权限或者能否装载实用程序数据库并连接到该数据库。

```
iqsrv16 -n TestSrv -gu none c:\mytestdb.db
```

要只允许知道实用程序数据库口令的用户执行文件管理语句，通过运行以下命令启动服务器。

```
iqsrv16 -n TestSrv -su secret -gu utility_db
```

以下命令将 `Interactive SQL` 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用户。

```
dbisql -c
"UID=DBA;PWD=secret;DBN=utility_db;Host=host1;Server=TestSrv"
```

在成功执行上述命令之后，用户连接到该实用程序数据库，并能执行文件管理语句。

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目录存储库（直到它关闭），因此所有目录页都必须具有相同的大小。

目录文件也是按页（大小分别为 4096、8192、16384 或 32768 字节）进行组织。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，可将 SAP Sybase IQ 能够支持的每个表的列数最大化。

在缺省情况下，服务器的页面大小与命令行上各个数据库的最大页面大小相等。**-gp** 选项会替代此缺省值。服务器启动之后，即无法装载目录页大小超过该服务器页大小的数据库。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页大小大的数据库文件。

如果使用较大的页大小，则应增加高速缓存大小。由于相同大小的高速缓存只能容纳少许较大页面，因而降低了空间安排的灵活性。

注意： **-gp** 选项和此处所列的页大小仅应用于目录存储。您可以在 **CREATE DATABASE** 命令的 **IQ PAGE SIZE** 参数中设置 IQ 存储库的页大小。

-gp iqsrv16 数据库服务器选项

设置允许的最大数据库页面大小。

语法

```
iqsrv16 -gp { 2048 | 4096 | 8192 | 16384 | 32768 } ...
```

缺省值

4096（如果在不装载数据库的情况下启动数据库服务器）

适用于

所有操作系统和数据库服务器。

注释

不能装载页面大小大于服务器页面大小的数据库文件。此选项显式设置服务器的页面大小（以字节为单位）。

在缺省情况下，服务器的页面大小与命令行上各个数据库的最大页面大小相等。

在所有平台上，如果不使用此选项，并且在不装载数据库的情况下启动服务器，则缺省值为 4096。

客户端/服务器环境选项

-x（通信协议）、**-ti**（网络连接超时）和 **-ti**（客户端连接超时）选项可帮助您设置客户端/服务器环境。

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。SAP Sybase IQ 支持将一组通信协议用于跨网络通信和同一计算机通信。

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，但通常不建议使用。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，可使用 **CommLinks** 连接参数控制许多相同的选项。

要使用 **TCP/IP** 协议启动服务器，可以使用：

```
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。如果省略此开关且使用 **TCP/IP**，或者不指定端口号，则会使用缺省端口 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
```

-x iqsrv16 服务器选项

指定服务器端网络通信协议。

语法

-x list

Remarks

list 是一个由逗号分隔的 **tcpip** 的列表或 **namedpipes** 设置。例如：

```
-x tcpip,ipx
```

仅允许 **TCP/IP** 和 **IPX** 通信。

缺省情况下，将尝试使用操作系统上数据库服务器支持的所有设置。

对于某些协议，可以提供采用以下格式的附加参数：

```
-x tcpip(PARM1=value1;PARM2=value2;...)
```

对于 **UNIX**，如果提供多个参数，需要使用引号：

```
-x "tcip (PARM1=value1;PARM2=value2;...)"
```

限制不活动的连接的开关

服务器选项 **-tl** 和 **-ti** 确定 SAP Sybase IQ 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，用以确认连接的完好性。如果服务器运行了一个活动超时周期（缺省为 2 分钟）但没有检测到活动包，则切断通信。服务器将断开与该客户端关联的所有连接。没有警告。任何打开事务范围内的所有活动都会回退。

服务器上的 **-tl** 开关可以为连接时未指定 **-tl** 开关的所有客户端设置活动超时（以秒为单位）。活动包会以（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 SAP Sybase IQ 早期版本的用户）不希望仅在处于非活动状态 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 **-ti** 的建议值一起使用。请按如下所示设置此开关：

```
-tl 300
```

如果此值效果不理想，请尝试 **-tl 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

SAP 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，此选项可解除这些连接具有的所有锁。**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。

-ti iqsrv16 数据库服务器选项

断开不活动的连接。

语法

```
iqsrv16 -ti minutes ...
```

缺省值

240（4 小时）

适用于

所有操作系统和数据库服务器。

注释

断开在指定的 *minutes* 时间内没有提交请求的连接。最大值为 32767。客户端计算机在数据库事务期间保持锁定，直至事务结束或连接断开。可提供 **-ti** 选项来断开不活动的连接，解除客户端锁定。

由于大多数连接都要通过网络链接 (TCP)，所以当与 **iqsrv16** 结合使用时，**-ti** 选项会非常有用。

将该值设置为零将禁止检查不活动的连接，因此不会断开任何连接。如果未使用 **Idle** 连接参数，那么 **TCP/IP** 连接的空闲超时值将由 **-ti** 数据库服务器选项进行控制。如果同时指定了 **-ti** 数据库服务器选项和 **Idle** 连接参数，那么空闲超时值将由连接参数进行控制。

-tl iqsrv16 数据库服务器选项

设置发送活动包的周期。

语法

```
iqsrv16 -tl seconds ...
```

适用于

所有使用 **TCP/IP** 的数据库服务器。

注释

活动包通过客户端/服务器 **TCP/IP** 通信协议定期发送，用以确认连接的完整性。如果服务器在运行了 **LivenessTimeout** 这样一段时间（缺省值为 2 分钟）后未检测到连接上存在活动包，则将断开通信，并且服务器会删除与该客户端相关的连接。**Unix** 多线程客户端和 **TDS** 连接不进行活动检查。

服务器上的 **-tl** 选项为没有指定活动周期的所有客户端设置 **LivenessTimeout** 值。

如果连接在 **LivenessTimeout** 值的三分之一到三分之二这么长的时间段内未发送任何数据包，则将发送活动数据包。

当连接超过 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，这样服务器可以更有效地处理大量连接。活动包在 **LivenessTimeout** 值的三分之一到三分之二这么长的一段时间内对每个空闲连接发送。不会同时发送大量活动包。如果发送活动包需要很长时间，则会在经过 **LivenessTimeout** 值的三分之二这么长的时间后将其发送。如果发送活动包所用的时间很长，数据库服务器消息日志中会出现一个警告。如果出现这样的警告，请考虑增大 **LivenessTimeout** 的值。

尽管通常不建议这样做，但您可以指定以下语句，以禁止发送活动包：

```
iqsrv16 -tl 0 -n my_server
```

请考虑按如下所示的方法将 **LivenessTimeout** 值增加到一小时，而不要禁用 **LivenessTimeout** 选项：

```
iqsrv16 -tl 3600 -n my_server
```

强制恢复模式选项

出现故障后，可使用常用启动选项重启服务器。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。

-iqfrec iqsrv16 服务器选项

在强制恢复模式下打开数据库。

语法

-iqfrec dbname

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。

以下 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。

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 SAP 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：

```
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 SAP Sybase IQ 或 SQL Anywhere 服务器运行时，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（SAP Sybase IQ 或 SQL Anywhere），则可以避免此错误。

避免共享内存冲突的方法

多种方法可避免在使用共享内存时发生冲突。

- 创建每个服务器专用的临时目录。通过在两个系统中显式设置 IQTMP16 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。
- 在 .odbc.ini 文件（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
```

如果在每个系统上运行多个服务器，请确保每个服务器有唯一的：

- 名称，启动时使用 **-n** 参数指定。
- 端口号，使用 **-x** 参数指定。

服务器活动日志

使用适合于自己所用平台的命令，可以指定 SAP 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 SAP Sybase IQ 服务器时，在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。该示例来自 AIX 系统：

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

Run Directory      : /myserver/users/sybase/iqdemo_160_sep05
Server Executable  : /myserver/users/sybase/160_sep05/IQ-16_0/
bin64/iqsrv16
Server Output Log  : /myserver/users/sybase/160_sep05/IQ-16_0/
logfiles/iqdemo_3658.0001.srvlog
Server Version     : 16.0.0.6556/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 16.0
I. 09/06 17:25:26. (64bit mode)
I. 09/06 17:25:26. Copyright 1992-2012 by Sybase, Inc. All rights
reserved
I. 09/06 17:25:26. Copyright (c) 2001-2012, 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_160_sep05/iqdemo.db) at Tue Sep 06 2012 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 2012 17:25
I. 09/06 17:25:27. Finished checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2012 17:25
I. 09/06 17:25:28. Database "iqdemo" (iqdemo.db) started at Tue Sep
06 2012 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 2012 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 2012 17:25
I. 09/06 17:25:45. Finished checkpoint of "iqdemo" (iqdemo.db) at Tue
Sep 06 2012 17:25
I. 09/06 17:25:46. Database server stopped at Tue Sep 06 2012 17:25

```

start_iq 日志文件

使用 **start_iq** 实用程序启动服务器时，服务器活动将记录在由 **\$IQLOGDIR16** 定义的目录下放置的 **ASCII** 文本文件中。该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** 命令以查看日志内容。例如：

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

如果未定义 \$IQLOGDIR16 目录，则在 UNIX 上该日志将写入到 \$IQDIR16/logfiles/ 目录中，而在 Windows 上则写入到安装 SAP Sybase IQ 时定义的 \$IQLOGDIR16 目录中。

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，这样可以帮助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服务器是否正在运行。

- 登录到服务器已启动的计算机，发出：

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

该输出与 UNIX 平台上的输出略有不同。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/iql60/IQ-16_0/bin64/
iqsrv16
@iqdemo.cfg iqdemo.db -ti 4400 -gn 25 -o
/ibm64srv/users/sybase/iql60/IQ-
16_0/logfiles/ibm64srv_iqdemo.0003.srvlog -hn 7
```

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

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

为服务器日志文件命名

服务器日志名称缺省为 server.nnnn.srvlog。

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

例如，要将输出保存为服务器启动目录下名为 results 的文件，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，SAP 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

系统管理员必须知道如何停止 SAP Sybase IQ 数据库服务器，而且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

某些情况下，可能需要停止并重新启动服务器。

例如：

- 安装 SAP Sybase IQ 的新版本
- 要重置某些服务器命令行选项
- 使少数几个服务器范围的数据库选项生效
- 在关闭操作系统会话之前

另请参见

- 数据库选项的作用域和持续时间（第 351 页）

停止服务器

停止服务器的不同方式。

SERVER OPERATOR 系统特权是停止服务器所需的缺省权限级别，但您可以使用 **-gk** 启动选项将缺省值更改为 **ALL** 或 **NONE**。将权限级别设置为 **ALL** 时，所有用户都可以停止服务器；而将权限级别设置为 **NONE** 时，则意味着任何人都不能停止服务器。在生产环境中，只允许服务器操作员停止数据库服务器。

请不要关闭仍然连接客户端的服务器，否则将丢失未提交的事务。请在关闭服务器之前断开连接或关闭所有客户端。

UNIX 中的服务器关闭命令

有几个命令可以关闭数据库服务器。

命令	描述
stop_iq	在不考虑用户、连接或装载进程状态的情况下完全关闭服务器。 语法： <code>stop_iq [options]</code>

命令	描述
dbstop	<p>停止服务器或数据库。可以借助其它选项来停止服务器，即使在存在活动连接的情况下也是如此。</p> <p>语法：</p> <pre>dbstop [options] server-name</pre> <p>要使用 dbstop 停止服务器，必须指定 server-name 以及启动服务器时指定的所有连接参数。</p>
iqsrv16	<p>如果没有将输入重定向到其它设备，在服务器启动窗口中进行输入时停止服务器。</p> <pre>q</pre>
STOP ENGINE	<p>STOP ENGINE 是用于停止数据库服务器的 SQL 语句。</p> <p>语法：</p> <pre>STOP ENGINE engine-name [UNCONDITIONALLY]</pre> <p>UNCONDITIONALLY 参数可以停止数据库服务器，即使该服务器中存在客户端连接时也是如此。</p>

Windows 中的服务器关闭方法

有以下几种方法可用来关闭或停止服务器。

要从此处停止服务器...	应执行的操作...
命令行	运行 stop_iq 或 dbstop 命令。
桌面	右键单击系统任务栏中的“服务器图标”，然后选择 Shutdown server name 。
服务管理器	<p>如果您将服务器作为 Windows 服务来启动：</p> <ol style="list-style-type: none"> 1. 在“控制面板”上，选择“管理工具” > “服务”。 2. 选择 SAP Sybase IQ 服务，然后单击 Stop the service。
Interactive SQL	运行 stop engine 命令来停止指定的数据库服务器。

在 UNIX 的 Cron 或 At 作业中停止服务器

要在 **cron** 或 **at** 作业中使用 **stop_iq**，请指定 **stop_iq** 可执行文件的完整路径名和相应的 **-stop** 选项。

如果运行中的服务器仅有一个由启动 **cron** 或 **at** 作业的用户 ID 启动，则设置 **-stop one** 将关闭一个服务器。这样可以避免在运行多个服务器的情况下意外关闭错误的服务器：

```
stop_iq -stop one
```

设置 **-stop all** 将关闭所有由启动 **cron** 或 **at** 作业的用户 ID 启动的服务器。

```
stop_iq -stop all
```

示例 - 使用 **stop_iq** 停止服务器

以下示例在 UNIX 操作系统命令行中使用 **stop_iq** 实用程序来关闭 SAP Sybase IQ 服务器并关闭它的所有用户连接。

当您发出 **stop_iq** 命令时，SAP Sybase IQ 会列出其他用户拥有的所有服务器，后跟您自己的服务器。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 16 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-16_0/logfiles/surya ibm2.001.srvlog -hn 8          pamela    409802
18:05:02      0:05  SVR:ibml_iqdemo2 DB:iqdemo
PORT:2678/ibml/users/sybase/iql60/IQ-16_0/bin64/iqsrv16 @iqdemo.cfg
iqdemo.db -ti 4400 -gn 25 -o /ibml/users/sybase/iql60/IQ
16_0/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/iql60/IQ-16_0/bin64/
iqsrv16
@iqdemo.cfg iqdemo.db -ti 4400 -gn 25 -o /myserver/users/sybase/
iql60/IQ-
16_0/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）。您将看到与以下相类似的消息：

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

要使服务器继续运行，请键入 **N**（否）。您将返回到系统提示符，并且服务器不会关闭。

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 SAP 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 身份连接到在服务器上运行的其中一个数据库（包括 `utility_db` 数据库），或者如果服务器是使用 `-gk ALL` 选项启动的，则可以从 *Interactive SQL* 停止该服务器。

停止服务器所需的权限

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**，它需要 **SERVER OPERATOR** 系统特权，但也可以将该值设置为 **ALL** 或 **NONE**。如果将其设置为 **NONE**，即使是具有 **SERVER OPERATOR** 系统特权的用户也无法执行 **STOP ENGINE**。在生产环境中，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。

如果在数据库服务器运行时关闭其所在的操作系统会话，或使用操作系统命令（而不是 **stop_iq**）来停止数据库服务器，则服务器会关闭，但并未完全关闭。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。

启动和停止数据库

您可以在启动服务器时或在服务器运行后启动数据库。

在每个服务器上仅运行一个数据库，特别是在生产环境中。

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。
- 要从 Interactive SQL 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，请包括连接参数 **“DBF=db-file”**。
- 要在已连接到服务器的情况下启动嵌入式数据库，请使用为新连接指定了数据库文件的 DBF 参数连接到数据库。相应数据库文件将被装载到当前服务器上。

数据库启动准则

关于数据库启动，应考虑若干问题。

文件访问

要启动数据库，IQ_SYSTEM_MAIN 的所有文件、IQ_SYSTEM_TEMP 的所有文件以及目录文件 SYSTEM 必须可用。启动数据库时可以跳过无法完全打开的 dbspace。如果在启动服务器时无法打开除 IQ_SYSTEM_MAIN 以外的 IQ 主存储库 dbspace 的任何可写入文件或除 SYSTEM 以外的任何目录 dbspace 文件，则 SAP Sybase IQ 会记录一个错误，并将该 dbspace 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 的任意文件，则除非使用 **-iqnotemp** 启动参数，否则数据库将不会启动。

SAP Sybase IQ 检查每个 dbspace 文件头中的 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 dbspace 标记为脱机。

通过 **ALTER DBSPACE ONLINE** 语句，可以使启动时标记为脱机的 dbspace 联机，并假定问题已得以更正并且可以打开该 dbspace。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 dbspace 文件的路径。

驻留在脱机 dbspace 中的表对象不可用。对脱机 dbspace 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。dbspace 脱机后，缓冲区高速缓存中可能仍存在数据页。如果表非常小，即使 dbspace 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

服务器使用固定大小的页将数据库信息保存在内存中。一旦服务器已经启动，则不能装载目录页大小或 IQ 页大小比服务器页大的数据库。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 SERVER OPERATOR 系统特权的用户才能启动 SAP Sybase 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（包括具备 SERVER OPERATOR 系统特权的用户）都无法启动数据库。

停止数据库

停止数据库的不同方式。

- 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，除非您将 **AUTOSTOP** 连接参数显式设置为 NO，否则该数据库会自动停止。
- 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》中的 STOP DATABASE 语句 [Interactive SQL]。

另请参见

- STOP DATABASE 语句 [Interactive SQL]（第 440 页）

启动 iqdemo 数据库

使用安装时提供的脚本可以创建 iqdemo 数据库以及可启动该数据库的配置文件。

此配置文件名为 iqdemo.cfg，包含启动演示数据库所需的所有参数。请参见《快速入门》

连接到 SAP Sybase IQ 服务器和数据库

SAP 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。

您可以连接到多个数据库服务器；指定明确标识服务器的连接选项。

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

客户端应用程序可以从 ODBC、OLE DB、嵌入式 SQL 应用程序和 Interactive SQL 连接到数据库。

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。

连接路线图

该路线图显示了哪些主题可解决特定的连接需求。

为了...	单击本页底部的相关主题链接……
使用客户端应用程序进行连接	受支持的连接接口
自定义连接字符串	连接和通信参数
创建数据源	ODBC 数据源
诊断网络连接问题	网络通信故障排除

另请参见

- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）

- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 页)
- 附录：连接参数和通信参数参考 (第 305 页)
- 网络通信故障排除 (第 284 页)

连接方式

应用程序以多种方式连接到 SAP Sybase IQ 数据库。

要在本地服务器上查询 iqdemo 数据库，只需要显示的连接参数。

多数 IQ 应用程序环境需要一组更复杂的连接参数。

另请参见

- 连接路线图 (第 39 页)
- 受支持的连接接口 (第 49 页)
- 连接状态 (第 50 页)
- 连接参数的工作原理 (第 51 页)
- ODBC 数据源中的连接参数 (第 52 页)
- Interactive SQL 连接 (第 53 页)
- 文件数据源 (第 55 页)
- 从其它数据库连接 (第 56 页)
- 如何测试连接 (第 58 页)
- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 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 16.0” > “Interactive SQL”。
2. 在 “标识” 选项卡中，为 “用户” 和 “口令” 分别键入 DBA 和 sql。

3. 在“数据库”选项卡中，选择“查找”。
4. 从“查找服务器”屏幕中选择您的 `iqdemo` 服务器，然后单击“确定”。

使用相同方法连接到任意已经运行的数据库服务器。还可以指定非缺省字符集和语言。

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。

在以下情况中：

- 该数据库是否已在服务器上运行？如果是，您可以在“连接”对话框中指定较少的参数。如果未运行，请指定数据库文件以便 **Interactive SQL** 启动数据库。
- 您的计算机上是否有多个数据库正在运行？如果是，请指定希望 **Interactive SQL** 连接到的数据库。如果只有一个数据库，则 **Interactive SQL** 建立连接，且不需要在“连接”对话框中进行指定。

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. 如果“连接”对话框没有自动显示，可以启动 **Interactive SQL** 并打开该对话框。
2. 在“标识”选项卡上，输入用户 ID 和口令。
3. 执行下列操作之一：
 - 如果服务器只包含一个数据库，则单击“确定”以连接。
 - 如果该服务器包含多个数据库，则单击“数据库”选项卡并指定一个数据库名称。该名称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。而不需要数据库文件。

在 UNIX 上从 Interactive SQL 连接到数据库

对于本地计算机上的数据库，则不必指定主机和端口。

1. 在系统命令提示符处启动服务器和数据库：

```
start_iq dbname
```

2. 启动 **Interactive SQL**：

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

连接到 SAP Sybase IQ 服务器和数据库

```
dbisql -c "uid=DBA;pwd=sql" -host fiona -port 1870 -n fiona_iqdemo  
$IQDIR16/demo/iqdemo.db
```

从 UNIX 系统连接

从 UNIX 系统上的命令行连接到正在运行的服务器。

前提条件

确保正确设置了 PATH 和其它环境变量。

过程

1. 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
```

```
cd $IQDIR16/demo
```

2. 要启动演示数据库，请输入：

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

3. 启动 Interactive SQL：

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

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**。但是，**dbisqlc** 不包含 **dbisql** 的所有功能。

-c 参数指定连接参数。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=tcipip** (或 **CommLinks=tcipip**) 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=SERVER1_iqdemo;  
links=tcipip(port=1234)"
```

从 Windows 系统连接

从“开始”菜单或命令提示符连接到正在运行的数据库。

1. 选择“程序” > “Sybase” > “Sybase IQ 16.0” > “Interactive SQL”，或在 Windows 命令提示符处输入：

```
dbisql
```

您可以在 `dbisql` 命令中包括 `-c` 参数以指定连接参数。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 SAP 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称（例如，对于 `iqdemo` 数据库，服务器名称为“`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。

缺省的端口号是 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。您可以通过运行“Sybase IQ 16.0” > “ODBC 管理器 (32 位)”或“Sybase IQ 16.0” > “ODBC 管理器 (64 位)”来查找端口号。选择“用户 DSN 上的用户数据源”选项卡，然后单击“配置”。在命令提示符处输入 `dblocate`。

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。
如果显示“连接”对话框或有关丢失信息的错误消息，请在“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，则分行输入 `-host` 和 `-port` 参数，例如：

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

5. 连接数据库后，将显示 Interactive SQL 窗口。连接的数据库名称、用户 ID 和服务名称显示在标题栏中。
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，并显示一则表明数据库所使用的归类的消息。

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。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。

使用下列连接参数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 SAP 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 SAP Sybase IQ。您可以使用下列方式之一：

- 从 Windows “程序” 菜单中，选择 “Sybase” > “Sybase IQ 16.0” > “Interactive SQL”，或
- 在系统命令提示符下键入 dbisql。

启动 Interactive SQL 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”标签页。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-16_0，则输入：

```
c:\sybase\IQ-16_0\iqdemo.db
```

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

TCP/IP 协议

SAP Sybase IQ 使用 TCP/IP 将客户端连接到运行在不同计算机上的数据库。

使用 TCP/IP 协议时，可以使用传送层安全和 RSA 加密技术来保护客户端/服务器通信的安全。

UDP

UDP 是一种 IP 之上的传送层协议。SAP Sybase IQ 可使用 UDP 进行初始服务器名称解析，之后再使用 TCP 进行连接和通信。

数据库服务器为响应客户端广播而发送的 UDP 数据包不包含敏感信息。这些数据包中所含的数据仅限于下列项：

- 数据库服务器名称
- 端口号
- 数据库服务器版本
- 运行在数据库服务器上的数据库的名称

可使用 `-dh` 数据库选项对 UDP 广播响应隐藏数据库名称。

可指定 `-sb` 服务器选项以禁用 UDP 监听器。

在 Windows 上使用 TCP/IP

所有 Windows 平台上数据库服务器的 TCP/IP 实现都使用 Winsock 2.2。Windows Mobile 上的客户端使用 Winsock 1.1 标准。

对通过 TCP/IP 的客户端/服务器通信进行加密

缺省情况下，通信包是不加密的，但这样可能会引起潜在的安全风险。您可以使用简单加密技术或传送层安全来保护客户端应用程序和数据库服务器之间在 TCP/IP 上的通信安全。传送层安全提供了服务器验证、采用 RSA 加密技术的高度加密以及其它用于保护数据完整性的功能。

SAP Sybase IQ 中的 IPv6 支持

在启用了 IPv6 的计算机上，缺省情况下，网络数据库服务器会监听所有的 IPv4 和 IPv6 地址。Windows CE 上不支持 IPv6。

通常情况下，不需要更改数据库服务器启动行便可使用 IPv6。在需要指定 IP 地址的情况下，数据库服务器和客户端库都接受 IPv4 和 IPv6 地址。例如，如果计算机上启用了多个网卡，则其可能会有两个 IPv4 地址和两个 IPv6 地址。对于包括端口号的 IPv6 地址，必须将地址括在方括号或圆括号中。如果想要让数据库服务器只监听其中一个 IPv6 地址，则可以指定以下格式的地址：

```
iqsrv16 -x tcpip(MyIP=fd77:55f:5a64:52a:202:5445:5245:444f) ...
```

类似地，如果客户端应用程序需要指定服务器的 IP 地址，则连接字符串或 ODBC 数据源可以包含该地址，格式如下：

```
...HOST=fd77:55f:5a64::444f;...
```

每个接口都指定有一个接口标识符，该标识符出现在 IPv6 地址的结尾。例如，如果 `ipconfig.exe` 列出地址 `[fd77:55f:5a64::444f]`，则接口标识符为 7。在 Windows 平台上指定 IPv6 地址时，应使用该接口标识符。在 Unix 上，可指定接口标识符或接口名称（接口名称是 `ifconfig` 所报告的接口的名称）。例如，在以下 IPv6 地址中，接口名称为 `eth1`：`[fd77:55f:5a64::444f;]`。在 Linux（内核 2.6.13 及更

高版本) 上指定 IPv6 地址时, 接口标识符是必需的。此要求会影响以下各项指定的值:

示例

假设 `ipconfig.exe` 列出两个接口, 一个标识符为 1, 另一个标识符为 2。如果要查找网络上 2 号接口使用的数据库服务器, 则可以指示客户端库仅在该接口上进行广播:

```
LINKS=tcpip(BROADCAST=ff02::1%2)
```

IPv6 链路本地多播地址为 `ff02::1`。

防火墙连接

当客户端应用程序在防火墙的一边, 而数据库服务器在另一边时, 连接会受到限制。防火墙软件可以根据网络端口过滤网络数据包。而且, 通常不允许 UDP 数据包穿过防火墙。

通常, 您可以使用 **Host** 连接参数并提供数据库服务器地址和端口来穿过已正确配置的防火墙进行连接。如果数据库服务器使用的是缺省端口 2638, 则不需要提供端口。

以下连接字符串片段使用端口 2020 连接到运行在地址为 `serverhost` 的计算机上的名为 `myserver` 的数据库服务器。未使用 UDP 数据包, 因为 **Host** 连接参数指定了 TCP/IP 地址和端口。

```
Server=myserver;Host=serverhost:2020
```

只允许特定客户端端口的防火墙

如果必须将防火墙配置为只允许特定客户端端口, 则必须使用 **CommLinks(LINKS)** 连接参数, 而不能使用 **Host** 连接参数。使用 **CommLinks** 连接参数时需要以下 TCP/IP 协议选项:

- **Host** - 将此协议选项设置为运行数据库服务器的主机名。可以使用其简写形式 **IP**。
- **ServerPort** - 如果数据库服务器未使用缺省端口 2638, 则必须指定它所使用的端口。可以使用其简写形式 **PORT**。
- **ClientPort** - 将此协议选项设置为供客户端应用程序使用的某一范围的允许值。您可以使用简写形式 **CPORT**。
- **DoBroadcast=NONE** - 设置此协议选项可防止在连接服务器时使用 UDP。可以使用其简写形式 **DOBROAD**。

必须对防火墙进行配置, 使其允许在 SAP Sybase IQ 数据库服务器地址与所有 SAP Sybase IQ 客户端地址之间进行 TCP/IP 通信。SAP Sybase IQ 数据库服务器的地址由运行 SAP Sybase IQ 服务器的计算机的 IP 地址 (**HOST** 协议选项) 和 SAP Sybase IQ 数据库服务器的 IP 端口号 (**ServerPort** 协议选项, 缺省值为 2638) 组成。

请使用一个客户端端口数超过每个客户端计算机最大并发连接数的端口范围, 因为几分钟超时过后才可以再次使用客户端端口。在 **ClientPort** 协议选项中指定的客户端范围必须与防火墙允许的范围一致。

示例

以下连接字符串片段将客户端应用程序限制为只能使用端口 5050 到 5060，并使用服务器端口 2020 连接到运行在地址为 myhost 的计算机上的名为 myeng 的数据库服务器。未执行 UDP 广播，因为 DoBroadcast 协议选项已设置为 NONE。

```
Server=myeng;LINKS=tcip(ClientPort=5050-5060;HOST=myhost;PORT=2020;DoBroadcast=NONE)
```

拨号网络连接 (CommLinks 连接参数)

要通过拨号网络连接进行连接，必须使用 CommLinks 连接参数。

在客户端，您应该指定以下协议选项：

- **Host** - 您应该使用 Host (IP) 协议选项指定数据库服务器的主机名或 IP 地址。
- **DoBroadcast** - 如果指定 Host (IP) 协议选项，则无需执行广播搜索来查找数据库服务器。由于此原因，请使用直接广播。
- **MyIP** - 应在客户端上设置 [MyIP=NONE]。
- **TIMEOUT** - 设置 Timeout (TO) 协议选项以增加客户端在搜索服务器时所等待的时间。

示例

以下是通过拨号网络连接进行连接所使用的典型 CommLinks (LINKS) 连接参数的示例：

```
LINKS=tcip(MyIP=NONE;DoBroadcast=DIRECT;HOST=server-ip);TIMEOUT=15)
```

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 会使用数据源，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。

使用 ODBC 数据源连接

使用 ODBC 数据源从 Interactive SQL 连接。

1. 启动 Interactive SQL 并打开“连接”对话框（如果没有自动显示该对话框）。
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：

- 选择“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”。
- 选择“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）。可以通过单击“浏览”来搜索文件。

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

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，而是使用缺省行为来建立连接。

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 SAP 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的服务器。

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
```


从 SAP Sybase IQ 实用程序连接

与服务器通信的 SAP Sybase IQ 数据库实用程序使用嵌入式 SQL（而不是直接在数据库文件中）进行此连接。

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：

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

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

1. 使用 SQLCONNECT 环境变量设置（如果缺少任何命令行值）。SAP 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”。
2. 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命令行选项的说明，请参见《实用程序指南》。

受支持的连接接口

为了建立连接，客户端应用程序会调用某个支持接口中的函数。

SAP Sybase IQ 支持以下接口：

- ODBC
- OLE DB
- 嵌入式 SQL
- SAP Sybase Open Client - 请参见《编程》中的 SAP Sybase IQ 作为客户端应用程序的数据服务器。
- JDBC - 请参见《编程》中的 JDBC CLI。Interactive SQL 支持 SAP Sybase IQ 中包含的 iAnywhere JDBC 驱动程序。

注意： 不建议使用 iAnywhere JDBC 驱动程序。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）

- ODBC 数据源中的连接参数 (第 52 页)
- Interactive SQL 连接 (第 53 页)
- 文件数据源 (第 55 页)
- 从其它数据库连接 (第 56 页)
- 如何测试连接 (第 58 页)
- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 页)

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 ])
```

另请参见

- 连接路线图 (第 39 页)
- 连接方式 (第 40 页)
- 受支持的连接接口 (第 49 页)
- 连接参数的工作原理 (第 51 页)
- ODBC 数据源中的连接参数 (第 52 页)
- Interactive SQL 连接 (第 53 页)
- 文件数据源 (第 55 页)
- 从其它数据库连接 (第 56 页)
- 如何测试连接 (第 58 页)
- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 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

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 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 SAP 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 SAP 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 SAP 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 SAP 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 SAP 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。

注意： 不建议使用 iAnywhere JDBC 驱动程序。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- **Interactive SQL 连接**（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 页）

Interactive SQL 连接

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

在“连接”对话框中，您告知 **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。这样，**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，则需要在“连接”对话框中提供更详细的信息，以便 **Interactive SQL** 连接到相应数据库。

本节介绍了如何在 **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

- 连接路线图 (第 39 页)
- 连接方式 (第 40 页)
- 受支持的连接接口 (第 49 页)
- 连接状态 (第 50 页)
- 连接参数的工作原理 (第 51 页)
- ODBC 数据源中的连接参数 (第 52 页)
- 文件数据源 (第 55 页)
- 从其它数据库连接 (第 56 页)
- 如何测试连接 (第 58 页)
- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 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。

“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。
- “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 (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 SAP 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。

《编程》介绍受支持的驱动程序和编程接口。

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，每一部分都以包含在方括号中的名称开始。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，因此它们是公共信息。因此不应在 DSN 中设置口令，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。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）

- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 临时连接 (第 65 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 页)

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 SAP Sybase IQ，然后单击“下一步”。
3. 按照说明创建数据源。

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”。

从其它数据库连接

您可以从 SAP Adaptive Server® Enterprise 以外部数据源形式访问 SAP 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。

使用 CIS，在 Adaptive Server 中定义表示 SAP Sybase IQ 表的“代理表”。然后可以从 Adaptive Server 查询代理表。请参见《Adaptive Server Enterprise 组件集成服务用户指南》。

CIS 和 SAP Sybase IQ 提供另外一些连接其它数据库和共享数据的方法，因此用户应用程序可通过公共接口访问整个数据仓库。您可以使用 CIS 执行以下操作：

- 从 SAP Sybase IQ 访问 Adaptive Server 数据库中的数据。只有部分平台支持此功能。请参见针对所用平台的安装和配置指南。
- 在其它数据库服务器上访问 SAP 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。
- 连接单独 SAP Sybase IQ 数据库中的表。

请参见《编程》>“访问远程数据”。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 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. 将以下行添加到 \$IQDIR16/scripts/default.cfg 中，该行的端口号为未使用的端口号，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 包含以下行：

```
-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 SAP Sybase IQ 在相同子网中，则服务器的名称必须唯一。

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。

以下行测试通过 TCP/IP 连接是否可使用名为 **Ciaran** 的服务器：

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

以下命令用于测试当前计算机上是否使用了缺省服务器：

```
dbping
```

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 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

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 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**（不建议使用）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 **SET ANY SECURITY OPTION** 系统特权。
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三个值之一（不区分大小写）：

- **Standard** – 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。
- **Mixed** - 允许集成登录和标准登录。
- **Integrated** - 在这种设置中，必须使用集成登录执行所有数据库登录操作。

警告！ 如果将 **LOGIN_MODE** 数据库选项设置为 **Integrated**，则只有被授予集成登录映射的用户才能进行连接。尝试使用用户 **ID** 和口令连接会生成错误。只有具备 **MANAGE ANY USER** 系统特权的用户不受此限制。

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**”。

另请参见

- **GRANT INTEGRATED LOGIN** 语句（第 429 页）

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. 以对 **MANAGE ANY USER** 系统特权拥有管理权限的用户身份连接数据库。
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 **SAP 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**（不建议使用），则服务器尝试进行标准登录。

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** 的登录控制系统代替 **SAP Sybase IQ** 用于控制数据库访问的系统来工作。

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。

- 用户 dsmith 和数据库用户 ID 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其他任何人，无论他们提供什么口令都将成功登录到服务器，因为 Windows 将该连接尝试缺省设置为 "Guest" 用户配置文件。在使用 "dsmith" 登录 ID 成功登录到服务器之后，未经授权的用户便可通过集成登录映射以 DBA 身份成功连接到数据库。

注意：为安全起见禁用 "Guest" 用户配置文件。最安全的集成登录策略是在承载 SAP Sybase IQ 数据库的所有 Windows 计算机上都禁用 "Guest"。这可以通过使用 Windows 用户管理器实用程序来实现。

临时公共选项可以提高安全性

如果使用以下 SQL 语句将给定数据库的 LOGIN_MODE 选项值设置为 **Mixed**（不建议使用）或 **Integrated**，将为该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依赖于其所在操作系统的安全性，所以，设置临时公共选项可视作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），则对数据库的访问将恢复为 SAP Sybase IQ 安全模型，而非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”。

另请参见

- SET OPTION 语句（第 438 页）

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有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名 **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哪些权限和特权由向 **Guest** 用户 **ID** 授予的特权决定。

连接归集

连接池可以提高与数据库服务器建立多个简短连接的应用程序的性能。如果对连接启用连接池，当断开连接时，连接会自动进行高速缓存且可在应用程序重新连接时重复使用。通过 **ConnectionPool (CPOOL)** 连接参数控制连接池。当应用程序通过同一个连接字符串进行了指定数量的连接后，连接会被加入池中。

应用程序必须通过同一个连接字符串进行五个连接，连接才能进行高速缓存。连接名可以每次不同，但所有其它连接参数都必须相同，才能重复使用被高速缓存的连接。

如果应用程序进程再次连接且存在可用于相同连接字符串的高速缓存连接，则会重复使用高速缓存的连接。连接会在 **ConnectionPool (CPOOL)** 连接参数所指定的时间（缺省情况下为 60 秒）内保持高速缓存状态。

如果高速缓存的连接会更改应用程序的行为，则不会被重复使用。例如，以下情况下，高速缓存的连接不会被重复使用：

- 对于无连接时自动停止的数据库。
- 禁用连接。
- 数据库服务器达到连接限制。
- 口令发生改变。
- **login_mode** 选项已设置。

为确保连接池对应用程序是透明的，如果高速缓存连接时发生错误则会断开连接。如果在尝试重复使用高速缓存的连接时发生错误，则数据库服务器会尝试正常连接。

如果连接断开且尚未达到 **CPOOL** 连接参数指定的最大连接数，此连接将被放入高速缓存中。连接被重新初始化，高速缓存中的连接仍保持与数据库服务器的连通状态，即使应用程序已将其断开。连接的清除和重新初始化包括以下活动：

- 回退所有未完成的事务。
- 删除临时表、临时函数和变量。
- 重置连接选项和连接计数器。
- 递减和递增数据库服务器连接计数。如果仅带有高速缓存的连接的数据数据库服务器关闭时存在活动的连接，您不会获得通知。
- 执行所有定义的断开连接和连接事件。
- 执行 `login_procedure` 数据库选项并验证登录策略。
- 重置连接 ID。

将连接池和其它连接池产品一同使用

如果您使用的产品或 API 支持连接池，则产品或 API 的连接池比 SAP Sybase IQ 连接池拥有更高的优先级。两种类型的连接池可同时处于活动状态。

您的产品或 API 中连接池的行为可能会与 SAP Sybase IQ 连接池大不相同。如果您的产品或 API 中连接池的行为不适用于应用程序，可使用 SAP Sybase IQ 连接池并可能因此提高某些应用程序的性能。

连接池和只读向外扩展

如果连接还指定了 `NodeType (NODE)` 连接参数，应用程序通常会连接到主服务器，而主服务器会确定哪个复制节点的负载最轻。接着连接会被重定向到该节点。如果应用程序在短时间内几次建立又丢弃了这种连接，那么连接将被放入连接池中，且它也不会询问主服务器应该使用的复制节点。此行为可以降低主服务器上的负载，但不会发生预期行为。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 页）

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。

使用系统过程 `sa_conn_info` 或 `sa_conn_list` 查询关于临时连接的信息。

`ParentConnection` 属性可返回生成临时连接的连接的连接 ID。

临时连接带有的连接 ID 大于 10 亿 (1000000000)，并且它们的名称描述了连接的函数。

以下示例使用 `sa_conn_info` 系统过程返回结果集，其中显示是哪个连接创建了临时连接。

```
SELECT Number, Name, ParentConnection FROM sa_conn_info();
```

连接 8 生成了执行 `CREATE DATABASE` 语句的临时连接。

Number	Name	ParentConnection
1000000048	INT: CreateDB	8
9	SQL_DBC_14675af8	(NULL)
8	SQL_DBA_152d5ac0	(NULL)

另请参见

- 连接路线图 (第 39 页)
- 连接方式 (第 40 页)
- 受支持的连接接口 (第 49 页)
- 连接状态 (第 50 页)
- 连接参数的工作原理 (第 51 页)
- ODBC 数据源中的连接参数 (第 52 页)
- Interactive SQL 连接 (第 53 页)
- 文件数据源 (第 55 页)
- 从其它数据库连接 (第 56 页)
- 如何测试连接 (第 58 页)
- 集成登录 (第 58 页)
- 连接归集 (第 63 页)
- 逻辑服务器配置 (第 66 页)
- 如何终止连接 (第 67 页)
- 连接记录 (第 68 页)

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含成员节点等原因，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。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 如何终止连接（第 67 页）
- 连接记录（第 68 页）

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。

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**”
- 关闭具有活动连接的应用程序，例如 **Interactive SQL**

对于拥有 **DROP CONNECTION** 系统特权的用户，也可以通过发出 **DROP CONNECTION** 语句在 **Interactive SQL** 或嵌入式 **SQL** 中删除特定连接。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- **Interactive SQL** 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 连接记录（第 68 页）

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，<dbname>.iqmsg 日志都将记录此操作。

可以使用数据库选项 **LOG_CONNECT** 来控制对用户连接和断开连接的记录。如果在用户连接时禁用连接记录，然后在用户断开连接前启用，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”。

另请参见

- 连接路线图（第 39 页）
- 连接方式（第 40 页）
- 受支持的连接接口（第 49 页）
- 连接状态（第 50 页）
- 连接参数的工作原理（第 51 页）
- ODBC 数据源中的连接参数（第 52 页）
- Interactive SQL 连接（第 53 页）
- 文件数据源（第 55 页）
- 从其它数据库连接（第 56 页）
- 如何测试连接（第 58 页）
- 集成登录（第 58 页）
- 连接归集（第 63 页）
- 临时连接（第 65 页）
- 逻辑服务器配置（第 66 页）
- 如何终止连接（第 67 页）
- LOG_CONNECT 选项（第 352 页）

创建并管理 SAP Sybase IQ 数据库

使用 Interactive SQL 或 SAP Control Center 创建和管理 SAP Sybase IQ 数据库。

创建 SAP Sybase IQ 数据库

要定义数据库，需要使用 SQL 语句或数据库设计工具。

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 SAP Control Center 创建数据库

要在 SAP Control Center 中创建 SAP Sybase IQ 数据库，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

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。

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。

为使用 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，或者启动 *实用程序数据库*（不具有任何数据库文件和数据的幻像数据库）。

注意： 如果以 **-m** 服务器选项启动服务器，则无法创建数据库。

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。

可使用以下任一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：

```
dbisql -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。)

自动创建 dbspace

创建数据库时，数据库服务器将自动创建特定 `dbspace`。

DbSPACE 名称	作用	缺省操作系统文件名
IQ_SYSTEM_MAIN	主（永久）IQ 存储库文件	<i>dbname.iq</i>
IQ_SYSTEM_MSG	消息日志文件	<i>dbname.iqmsg</i>
IQ_SYSTEM_TEMP	临时 IQ 存储库文件	<i>dbname.iqtmp</i>
IQ_SHARED_TEMP	临时 IQ 存储库	最初，此 <code>dbspace</code> 不包含任何文件。
SYSTEM	目录存储库文件	<i>dbname.db</i>

`SYSTEM` `dbspace` 中包含系统表，当您构建数据库时，这些表可以保存模式定义。它还保存单独的检查点日志、回退日志，并可以选择保存目录存储库的写入文件、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保存会话期间所需的信息。此临时文件不同于 `IQ` 临时存储库，并且数据库服务器关闭后就不再需要该文件。该文件的名称由服务器生成，扩展名为 `.tmp`。其位置由 `TEMP` 环境变量或 `UNIX` 中的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。`dbspace` 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 DATABASE** 命令建立完全备份，并使用带有 **RENAME** 选项的 **BACKUP DATABASE** 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 dbspace 移动到其它分区上。新分区必须至少与当前 dbspace 大小一样大。新分区还必须有相同路径，才能使 dbspace 启动。

可以使用字节复制实用程序（如 dd）仅将原始数据复制到文件系统设备或另一原始设备。切勿使用字节复制将 dbfile 从文件系统移动到原始设备。

警告！ 为 dbspace（系统、IQ 主或 IQ 临时）分配系统文件时，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，包括局域网重载和 dbspace 文件问题。在 UNIX 和 Linux 平台上，请避免使用网络文件系统 (NFS) 装入的文件系统。在 Windows 上，请不要将 dbspace 文件放置在归另一节点所有的网络驱动器上。这些关于文件放置的建议也适用于日志文件。

为避免冲突，所有 dbspace 应由单个连接上的同一位数据库管理员进行管理。

通过增加专门用于存储随机访问文件（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。通过将顺序访问文件（包括事务日志和消息日志文件）放置在专用磁盘驱动器上，可以提高与这些文件相关的性能。

有关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 dbspace 和事务日志文件放在数据库服务器所在的物理计算机上。
- 将 SAP Sybase IQ 数据库 I/O 与其它数据库（如 Adaptive Server）中代理表的 I/O 隔离。
- 将事务日志和消息日志放在独立于 IQ 存储库、目录存储库和临时存储库并且独立于所有代理数据库（如 Adaptive Server）的磁盘上。

如果您的 IQ 需求非常大并且复杂到需要多个物理系统，请考虑使用 SAP Sybase IQ Multiplex 功能。

原始设备权限

在创建数据库或添加 dbspace 之前，需确保所有的原始设备均具有读写权限。检查 /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

SAP Sybase IQ 服务器不能管理使用 SAP Sybase IQ 12.7 之前的版本创建的数据库；同样，旧服务器也不能管理新数据库。

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。

必须指定目录存储库的文件名和 **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 不兼容。
- IQ 页大小为 128KB (**IQ PAGE SIZE 131072**)。
- IQ 消息文件和 IQ 临时存储库与目录存储库位于同一目录中。
- 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。
- **SYSTEM PROCEDURE AS DEFINER** 为 OFF。

提高口令安全性

口令在任何数据库安全系统中都是一个重要部分。有几种方法可以提高口令安全性。

- **实现登录策略** – 控制口令更改的频率、指定锁定帐户之前所允许的登录尝试次数, 或者强制执行口令到期。请参见登录策略。
- **实现最小口令长度** – 缺省情况下, 对口令的长度没有任何限制。要实现更高的安全性, 可以对所有新口令规定最小长度要求, 不允许使用短 (因而容易被猜到的) 口令。建议的最短长度为 6。请参见 **MIN_PASSWORD_LENGTH**。
- **实现口令规则** – 实现高级口令规则, 其中包括要求在口令中使用某些类型的字符、不允许口令重用和为口令设置有效期。创建新用户 ID 或更改口令时, 会进行规则校验。请参见 **VERIFY_PASSWORD_FUNCTION**。

另请参见

- **MIN_PASSWORD_LENGTH** 选项 (第 353 页)
- **VERIFY_PASSWORD_FUNCTION** 选项 (第 353 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。

使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其它路径名。

如果数据库位于 UNIX 上, 则可以定义每个路径名的符号链接。

如果省略目录路径, 则 SAP Sybase IQ 将按如下方式定位文件:

- 目录存储库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目录存储库所在的目录下或其相对目录下。
- 事务日志创建在目录存储库所在的目录下。（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目录存储库和 IQ 存储库不同的物理设备上。

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。

IQ PATH 参数准则

所需的 IQ PATH 参数将告知 SAP Sybase IQ 您正在创建 SAP Sybase 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。

请谨慎选择数据库的位置。虽然可以将 SAP Sybase IQ 数据库或其中任意文件移动到其它位置，但要执行此操作，必须关闭该数据库，并可能必须执行备份和还原。

可以向其它驱动器添加空间，但此额外空间只能用于存储新数据。不能轻易将特定的索引、表或数据行从一个位置移动到另一个位置。

每个操作系统都具有其自己的原始设备名称格式。有关在 Sun Solaris 上初始化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 2. UNIX 上的原始设备名

	示例
AIX	/dev/rraw121v
HP-UX	/dev/vg03/rrchee12g
Sun Solaris	/dev/rsd0c
Sun AMD	/dev/rdisk/c5t0d0s1

表 3. Windows 上的原始设备名

设备类型	要求的名称格式	SQL 示例
已分区	分配给该分区的字母	\\\\.\\C:
未分区	<i>PhysicalDriveN</i> ，其中 <i>N</i> 是从 0 开始的数字，可根据需要增至任意大小。通过在管理工具中运行磁盘管理器，可以查找物理驱动器编号。	\\\\.\\ PhysicalDrive32

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 中以 SQL 语句命名原始设备时，必须始终双写反斜线。

示例 1

以下语句将创建名为 company.db 的数据库。此数据库由四个 Windows 文件组成：

- 目录存储库，位于 company.db 中，此文件位于服务器的启动目录中（本例中为 c:\company）
- IQ 存储库，位于 c:\company\iqdata\company.iq 中
- 临时存储库，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

以下语句将创建名为 company.db 的数据库。此数据库由四个 UNIX 文件组成：

- 目录存储库，位于 company.db 中，此文件位于服务器的启动目录中（本例中为 /disk1/company）
- IQ 存储库，位于 /disk1/company/iqdata/company.iq 中
- 临时存储库，位于 /disk1/company/company.iqtmp 中
- IQ 消息日志文件，位于 /disk1/company/company.iqmsg 中

```
CREATE DATABASE 'company.db'  
IQ SIZE 2000  
IQ PATH '/disk1/company/iqdata/company.iq'
```

示例 3

以下 UNIX 示例将创建名为 company 的具有 IQ PATH 原始分区的数据库。

```
CREATE DATABASE 'company'  
IQ PATH '/dev/rdisk/c0t0d0s0'
```

示例 4

以下 Windows 示例将创建名为 company 的具有 IQ PATH 原始分区的数据库。

```
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 存储库的页大小。**IQ PAGE SIZE** 必须为 2 的幂，从 65536 到 524288 字节。IQ 页大小对于 IQ 存储库中的所有 dbspace 都是相同的。

为获得最佳性能，请使用以下最小 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。

SAP Sybase IQ 使用压缩格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 页大小决定着磁盘压缩量和 IQ 存储库的 I/O 传输块的缺省大小。对于大多数应用程序，此缺省值是最佳的。有关这些设置以及影响资源使用和性能的有关选项的信息，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，您可能会发现为给定数量的行提供下一个更高的 IQ 页大小将带来更好的性能。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。

数据库大小准则

创建数据库时，需要以 MB 为单位为初始数据库文件 (IQ_SYSTEM_MAIN dbspace) 设置大小并保留大小。

对于主存储库，这些值在 **IQ SIZE** 和 **IQ RESERVE** 参数中进行定义；对于临时存储库，则在 **TEMPORARY SIZE** 和 **TEMPORARY RESERVE** 中进行定义。

- 对于原始分区，无需指定 **IQ SIZE** 或 **TEMPORARY SIZE**；SAP 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，则此大小不能大于实际原始分区的大小。
- 对于操作系统文件，您可以依赖于缺省值，也可根据数据大小来指定一个值，范围从所需的最小值一直到最大值 4TB（增量为 1MB）。

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 **dbspace** 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 4. IQ 存储库和临时存储库的缺省大小和最小大小

IQ 页大小	IQ 存储库的缺省大小	临时存储库的缺省大小	显式指定时 IQ 存储库的最小大小	显式指定时临时存储库的最小大小
65536	4096000	2048000	4MB	2MB
131072	8192000	4096000	8MB	4MB
262144	16384000	8192000	16MB	8MB
524288	32768000	16384000	32MB	16MB

PAGE SIZE 参数准则

使用 **CREATE DATABASE PAGE SIZE** 选项设置目录存储库页大小。此选项的缺省值和最小值为 4096 (4KB)。

示例

下面的语句将创建具有 4096 字节 (4KB) 目录 **PAGE SIZE** 的数据库，其中 IQ 存储库位于 UNIX 原始分区上，并具有 131072 字节 (128KB) 的 **IQ PAGE SIZE**。缺省情况下，IQ 存储库大小即为原始分区大小，临时存储库是该大小的一半。因为没有指定临时存储库的路径，所以将在目录存储库所在的目录中进行创建。

```
CREATE DATABASE 'company'  
IQ PATH '/dev/rdisk/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。

IQ 主存储库和 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 **SET ANY PUBLIC OPTION** 系统特权的用户可对每个连接所用的空间量进行限制。此外，当 SAP Sybase IQ 的 IQ 主存储库或 IQ 临时存储库中空间不足时，如果未添加新的空间，服务器不会再暂停导致空间不足的事务。导致 IQ 主存储库或 IQ 临时存储库中空间不足的事务将失败并回退。

数据库选项 `MAX_TEMP_SPACE_PER_CONNECTION` 限制了每个连接所用的 IQ 临时存储库空间量，并跟踪所有数据操作语言 (DML) 语句及查询的临时存储库使用情况。`MAX_TEMP_SPACE_PER_CONNECTION` 监控并限制语句的实际运行时临时存储库使用情况。如果连接超过 `MAX_TEMP_SPACE_PER_CONNECTION` 选项设置的限额，则会返回错误并回退当前语句。

`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库使用。要限制每个连接的临时存储库使用，DBA 可为所有 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 SAP 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升级后的数据库。

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 `dbspace`。

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。

从数据库中断开其他用户的连接

SAP Sybase IQ 允许您从给定数据库中断开其他用户的连接。

通过使用 `connection_property` 函数请求连接编号，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. 以对 `DROP CONNECTION` 系统特权具有管理权限的用户身份连接到现有数据库。
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
```

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
```

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
```

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 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** 角色设置选项，因为所有用户和角色都从 **PUBLIC** 继承选项设置。

管理数据存储

指定 SAP Sybase IQ 如何分配磁盘空间，并确保在磁盘中对数据进行组织以获得最佳性能。

空间分配

所有 SAP Sybase IQ 数据库都是预分配的，无论其驻留在文件系统还是原始分区中。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，可将其作为总存储的逻辑子集进行管理。可将各个对象和子对象分配到各个表空间。

dbspace 是由一个或多个操作系统文件组成的表空间。

存储库是一个或多个 dbspace，用于存储永久数据或临时数据以作特殊用途。SAP Sybase IQ 具有五种存储库：

- 目录存储库，包含 SYSTEM dbspace 和附加的用户定义的目录 dbspace。
- IQ 主存储库，包含 IQ_SYSTEM_MAIN dbspace 和其它用户 dbspace。
- IQ 临时存储库，包含 IQ_SYSTEM_TEMP dbspace。
- IQ 共享临时存储库，包含 IQ_SHARED_TEMP dbspace。
- 行级版本控制 (RLV) 存储库，是用于高性能行级更新的内存存储库。

dbspace 类型

每种类型的 dbspace 中都存储着特定类型的 SAP Sybase IQ 数据。

Dbspace 类型	存储的数据	Dbspace 中包含的文件	Dbspace 数
SYSTEM dbspace	系统表、视图、存储过程、SQL Anywhere 表和函数定义。	一个	一个或多个
其它目录 dbspace	SQL Anywhere 表。	一个	一个或多个
IQ_SYSTEM_MAIN	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前进和回退数据）、增量备份元数据、数据库空间和标识元数据。可以在此处存储用户对象，但最好将这些对象存储在其它主 dbspace 中。	一个或多个	一个

Dbospace 类型	存储的数据	Dbospace 中包含的文件	Dbospace 数
其它主 dbospace (也称用户 dbospace)	数据库对象, 如表、索引和表元数据。	一个或多个	一个或多个。您必须获得 IQ_VLDBMGMT 选项的使用许可才能创建多个用户 dbospace。
RLV_STORE	已启用行级版本控制 (RLV) 的表中的实时内存数据。	一个	一个
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 dbospace。有关详细信息, 请参见下表。	一个或多个	一个
IQ_SHARED_TEMP	1 到 n 个临时 dbfile 组成的集, 用于定义一个由所有 Multiplex 节点共享的临时 dbospace。有关详细信息, 请参见下表。	一个或多个 (最初没有文件)	一个
IQ_SYSTEM_MSG	用于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

表的 dbospace 可以隐式指定, 也可显式指定。对于基表, **DEFAULT_DBSPACE** 选项的值隐式决定了 dbospace 的位置, 也可以使用 **CREATE TABLE IN *dbospace_name*** 子句显式指定该位置。通常在 IQ 主存储库的 dbospace 中创建基表, 但也可以在目录存储库的 dbospace 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。

始终存储在 IQ_SYSTEM_TEMP 中	始终存储在 IQ_SHARED_TEMP 中	可能存储在 IQ_SHARED_TEMP 或 IQ_SYSTEM_TEMP 中
用于进行服务器故障恢复的临时空间	用于进行分布式查询处理的分布式临时对象	IQ 临时用户对象, 如 IQ 临时表和索引
用于进行共享存储空间管理的本地视图		针对本地非版本控制临时对象的临时空间
临时存储事务状态 (保存点前进和回退)		

目录存储库

目录存储库中包含 SAP Sybase IQ 数据库的元数据。

元数据描述表、列和索引的布局。目录存储库有时也简称为目录。

- **SYSTEM dbspace** - 目录 **dbspace SYSTEM** 中包含数据库元数据，其存储格式与 **SQL Anywhere** 关系数据库系统中表的格式相同。**SQL Anywhere** 可以带或不带 **SAP Sybase IQ** 存在。您的目录存储库中可能具有 **SQL Anywhere** 样式的表连同 **SAP Sybase IQ** 表，或者您可能具有单独的 **SQL Anywhere** 数据库。每个目录 **dbspace** 中正好包含一个文件。
- 其它目录 **dbspace** - 可在独立于 **SYSTEM dbspace** 的 **dbspace** 中创建 **SQL Anywhere** 表。

可通过执行 **ALTER DBSPACE** 语句将空间预分配到目录存储库中。

IQ_SYSTEM_MAIN dbspace

IQ_SYSTEM_MAIN 是一个特殊的 **dbspace**，它包含打开数据库时所必需的结构：检查点日志、每个已提交事务和每个活动检查点事务的前进/回退数据、增量备份元数据、数据库空间和标识元数据。

IQ_SYSTEM_MAIN dbspace 是在创建数据库时或将旧版数据库升级为 **SAP Sybase IQ 16.0** 时进行创建的。打开数据库后，**IQ_SYSTEM_MAIN** 始终处于联机状态。

其它用户主 dbspace

创建用户主 **dbspace**，这样，用户就不会在 **IQ_SYSTEM_MAIN** 中放置用户表或索引。

最好避免在 **IQ_SYSTEM_MAIN** 中放置用户表或索引。管理员可能允许在 **IQ_SYSTEM_MAIN** 中创建用户表，特别是当这些表比较小而又非常重要时。但建议管理员在创建数据库之后立即创建第二个主 **dbspace**（用户主 **dbspace**），撤销 **PUBLIC** 在 **dbspace IQ_SYSTEM_MAIN** 中的 **CREATE** 特权，向所选用户或 **PUBLIC** 授予对新的主 **dbspace** 的 **CREATE** 特权，并将 **PUBLIC.DEFAULT_DBSpace** 设置为新的用户主 **dbspace**。

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';
```

注意： 您必须获得 **IQ_VLDBMGMT** 选项的使用许可才能创建多个用户 **dbspace**。

IQ 临时 dbspace

在创建或升级数据库时，将创建一个 **IQ** 临时存储库 **dbspace**，即 **IQ_SYSTEM_TEMP**。

每个 `dbspace` 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` 至少与在 `start_iq` 实用程序的 `-iqtc` 服务器选项中所指定的值的大小相同。

IQ 共享临时 `dbspace`

在创建数据库或将数据库升级到 SAP Sybase IQ 16.0 时，将创建一个 `IQ` 共享临时存储 `dbspace`，即 `IQ_SHARED_TEMP`。该 `dbspace` 存储在分布式查询处理节点之间共享的临时结构。

最初，此 `dbspace` 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。

在将 `TEMP_DATA_IN_SHARED_TEMP` 逻辑服务器策略选项设置为 `ON` 时，所有临时表数据和符合条件的临时数据都将写入共享的临时存储库中，前提是共享的临时存储库不为空。必须在设置该选项或在将读写文件添加到共享的临时存储库之后重新启动辅助节点。如果共享的临时存储库中不具有任何读写文件，或者如果您不希望重新启动辅助节点，数据则会写入到 `IQ_SYSTEM_TEMP` 中。当设置为 `OFF` 时，所有临时表数据和临时数据都将写入到本地临时存储库中。

共享系统临时存储库

具有共享临时存储库的 `Multiplex` 配置可使用 `IQ_SHARED_TEMP` `dbspace` 作为共享的系统临时存储库，而不需要为每个辅助服务器进行单独的本地存储。共享的系统临时存储库可简化 `Multiplex` 配置、提高性能并支持分布式查询处理。

在 `Multiplex` 系统中：

- 在将逻辑服务器策略选项 `TEMP_DATA_IN_SHARED_TEMP` 设置为 `ON` 时，SAP Sybase IQ 将在 `IQ_SHARED_TEMP` `dbspace` 中创建所有的临时对象。必须在设置该选项或在将读写文件添加到共享的临时存储库之后重新启动辅助节点。（如果共享的临时存储库中不含任何读写文件，或者如果您不希望重新启动辅助节点，数据则会写入到 `IQ_SYSTEM_TEMP` 中。）
- 您使用 **`IN IQ_SYSTEM_TEMP`** 子句创建的临时用户对象（如表或表索引）将进入 `IQ_SYSTEM_TEMP` 或 `IQ_SHARED_TEMP`，具体取决于逻辑服务器选项 `TEMP_DATA_IN_SHARED_TEMP` 的值：
 - 如果 `TEMP_DATA_IN_SHARED_TEMP` 为 `'OFF'`，对象将进入 `IQ_SYSTEM_TEMP`。
 - 如果 `TEMP_DATA_IN_SHARED_TEMP` 设置为 `'ON'`，对象将进入 `IQ_SHARED_TEMP`。

SAP Sybase IQ 不支持使用 **`IN IQ_SHARED_TEMP`** 子句创建临时用户对象。

- **`WITH STOP SERVER`** 子句自动关闭逻辑服务器中的所有服务器。这些语句支持 **`WITH STOP SERVER`**：
 - **`ALTER LOGICAL SERVER`**
 - **`ALTER LS POLICY`**

- **CREATE LOGICAL SERVER**
- **DROP LOGICAL SERVER**
- 如果使用 **ALTER LS POLICY ... WITH STOP SERVER** 更改 **TEMP_DATA_IN_SHARED_TEMP** 选项 'ON|OFF'，则该逻辑服务器中的所有服务器都将自动关闭。必须重新启动服务器，以强制逻辑服务器将临时数据放置在由 **TEMP_DATA_IN_SHARED_TEMP** 选项指定的存储库中。
- 如果使用 **ALTER LS POLICY** 将 **TEMP_DATA_IN_SHARED_TEMP** 设置为 'OFF'，逻辑服务器会在服务器下一次正常启动之后开始将临时数据放置在 **SYSTEM** 临时区域中。
- 也可以使用 **CREATE LOGICAL SERVER**、**ALTER LOGICAL SERVER** 或 **DROP LOGICAL SERVER** 语句和 **WITH STOP SERVER** 子句间接更改 **TEMP_DATA_IN_SHARED_TEMP** 值。

另请参见

- **ALTER LS POLICY** 语句（第 364 页）

IQ 消息文件 dbspace

IQ_SYSTEM_MSG 是一个系统 **dbspace**，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库。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。

由于它并不是 IQ 存储库 **dbspace**，所以 **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 **dbspace**。可以根据需要创建其它 **dbspace**。

创建和装载表时，**SAP Sybase IQ** 会将数据分发到该存储库中具有可用空间的所有现有 **dbspace** 中。创建 **dbspace** 时可为其保留增长空间。可将 **dbspace** 的大小调整到最大保留值。还可以使 **dbspace** 更小，前提是所有数据均已从 **dbspace** 的截断部分中移出。可根据需要从指定的 **dbspace** 中将单个数据库对象移出。

不要将所有磁盘空间都分配给数据库。请保留百分之十作为保留空间。**SAP Sybase IQ** 需要此空间来妥善处理空间不足的情况。

请在创建数据库时创建所有的 **dbspace**，而不要在旧 **dbspace** 已满时再逐个添加它们。这样可以确保 **dbspace** 能够更均匀地被填充，从而改进磁盘 I/O。

创建单独的数据库以便于调试。避免在生产数据库上执行开发工作，因为这样会增加服务器发生故障的可能性。

IQ 存储库和临时存储库的空间要求

数据量以及您所创建的索引的数量和类型决定了数据库中需要多少空间。

如果在向数据库进行装载或插入时空间不足，则服务器将回退整个事务或者回退到某一保存点。

除所有显式定义的临时表之外，SAP Sybase IQ 还将在装载和删除期间使用临时存储库作为排序、散列和位图的临时结果空间。所发出查询的类型、并发的使用程度以及数据的大小都将决定临时存储库需要多少空间。

主存储库和临时存储库的大小调整准则

- SAP Sybase IQ 体系结构影响着数据存储准则。
- **IQ_SYSTEM_MAIN** dbspace 保存除 IQ 表元数据以外的所有数据库元数据。IQ 表元数据存储存在表的 dbspace 和表版本 (TLV) 日志中。如果某个节点已关闭，则 Multiplex 必须存储多个版本，以便在该节点恢复时同步这些版本。这些版本可能会占用大量空间。
 - 目前大约有 20% 的 **IQ_SYSTEM_MAIN** dbspace 用于预分配空闲列表空间，无法用于用户数据。
 - 由于该版本的 SAP Sybase IQ 并行执行的操作更多，所以它占用的临时空间比先前版本更多。

有三个因素影响 **IQ_SYSTEM_MAIN** 存储库所需要的空间：

- 版本控制 – 所维护的版本量各不相同。
- 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。

虽然文档可以提供一般准则，但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 500GB 的开发或报告服务器，10 至 20GB 的 **IQ_SYSTEM_MAIN** 文件便已足够。

表 5. 生产数据库中 **IQ_SYSTEM_MAIN** 和 **IQ_SYSTEM_TEMP** 的大小准则

任务	准则	注释
从 iqunload -n 输出或针对小型测试数据库装载空模式	10GB 主存储，5GB 临时存储	CREATE DATABASE 大小以 MB 为单位。

任务	准则	注释
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，请分配标准大小的未使用的新原始设备。 IQ_SYSTEM_MAIN 的总大小应至少为数据库总大小的 1/100，且主存储至少为 100GB，保留空间至少为 100GB。 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 SAP Sybase IQ 能够跨多个设备对数据执行分条操作，从而提高性能。 对于生产中的 IQ dbspace，使用通过高性能的冗余磁盘阵列（如 RAID 5）实现的具有容错能力的文件系统。对于单服务器系统，可以使用本地文件系统，但 Multiplex 系统需要原始设备（理想情况下，位于存储区域网络设备中）。 	<p>如果指定原始设备，则省略 ms_size。</p> <p>在 Windows 系统中，只有具有管理员权限的用户帐户才能访问原始设备。rawaccredit 实用程序为当前会话的设备设置权限。</p> <p>始终将主保留大小设置为 IQ_SYSTEM_MAIN 大小的 20%。要设置主保留，请使用数据库选项 MAIN_RESERVED_DBSPACE_MB。</p>
为 Multiplex 创建主存储库	将 Simplex 数据库的建议空间增大一倍，或者使主 dbspace 至少为 200GB，使保留 dbspace 至少为 200GB。	

示例 1

在 **CREATE DATABASE** 语法中，缺省大小单位是 **MB**，而不是 **GB**。

下面的语句将创建具有 100GB **IQ_SYSTEM_MAIN** 和 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 允许您控制 **SAP Sybase IQ** 在 **IQ** 主存储库中为在释放保存点、提交和检查点操作期间所用的某些小型但关键的数据结构留出的空间量。

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

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。

在 UNIX 的原始设备上创建 IQ_SYSTEM_MAIN:

```
iqunload -au -ms_filename /dev/rdisk/clt0d1 -c
"UID=DBA;PWD=SQL;DBF=latest.db"
```

在 Windows 的原始设备上创建 IQ_SYSTEM_MAIN:

```
iqunload -au -ms_filename \\.\PhysicalDrive1 -c
"UID=DBA;PWD=SQL;DBF=latest.db"
```

将 Windows 访问设置为原始设备

Windows 系统限制原始设备对具有管理员特权的用户帐户进行访问。

要使用缺少管理员特权的帐户运行 SAP Sybase IQ 服务器，必须在每次重新启动系统之后为该帐户启用新的设备访问权限。

rawaccedit 实用程序为当前会话的设备设置权限。

为写入服务器设置读写访问，并为查询服务器设置读取访问。

1. 在命令提示符处键入以下命令:

```
rawaccedit
```

2. 在 IQ 原始设备访问窗口中，键入要授予访问权限的用户和设备的名称。

可以使用 Alt+N 切换到“用户名”框，使用 Alt+D 切换到“原始设备名”框。

要指定...	键入...
取消分区的原始设备	键入物理驱动器编号。 将取消分区的驱动器命名为 \\.\PhysicalDriveN，其中 N 是从 0 开始的数字。要查找物理驱动器编号，请运行“附件” > “系统工具” > “系统信息”。
已分区的原始设备	键入为该分区分配的字母。

3. 单击“添加”。
4. 更正顶部面板中所显示的用户名和设备名中的错误，然后单击“更新 ACL 并退出”。

设备访问权限将保留到重新启动 Windows。

查看原始设备的访问权限

显示 Windows 原始设备的当前访问权限。

rawaccess 实用程序必须以管理员用户身份运行。这与具有管理特权不同。右键单击“命令提示符”，然后选择“Run As Administrator”可完成此操作。以管理员身份运行时，窗口标题栏中将显示 "Administrator:"。

permission_types 遵守 Windows 的 Microsoft 访问控制项 (ACE)。有关详细信息，请参见 [http://msdn.microsoft.com/en-us/library/windows/desktop/aa374899\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa374899(v=vs.85).aspx)

1. 以管理员身份运行，打开命令提示符窗口。
2. 键入 **rawaccess**，然后键入原始设备字母：

```
rawaccess filename
```

其中，

- **filename** - 指向原始设备的 Windows 通用命名空间文件名路径，采用以下两种格式之一：
 - `\\.\logical_drive_name`
 - `\\.\physical_device_id`
- **logical_drive_name** - 用于表示磁盘分区的分区字母（例如：\\.\D:）。
- **physical_device_id** - 由 Windows 设备管理器分配的名称。可在控制面板中使用磁盘管理器搜索这些名称（例如：\\.\PhysicalDisk1）。

```
1. Allow Exec to \Everyone (Well Known Group)
2. Allow All to NT AUTHORITY\SYSTEM (Well Known Group)
3. Allow All to BUILTIN\Administrators (Alias)
4. Allow Exec to NT AUTHORITY\RESTRICTED (Well Known Group)
```

设置对原始设备的权限

在使用 Windows 上的原始设备之前，必须设置对该设备的访问权限。

要将原始设备与 SAP Sybase IQ 一起使用，运行 SAP Sybase IQ 的帐户不能使用 UAC（用户帐户控制）。

rawaccess 实用程序必须以管理员用户身份运行。这与具有管理特权不同。右键单击“命令提示符”，然后选择“Run As Administrator”可完成此操作。以管理员身份运行时，窗口标题栏中将显示 "Administrator:"。

缺省情况下，Windows 不允许直接对原始设备进行写访问，/GRANT 选项是添加访问权限的最简单的方法，因为该选项与操作系统设置的其它缺省条目合并在一起。

permission_types 遵守 Windows 的 Microsoft 访问控制项 (ACE)。请参见 [http://msdn.microsoft.com/en-us/library/windows/desktop/aa374899\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa374899(v=vs.85).aspx)

rawaccess 对 Win32 设备命名空间使用标准 Microsoft 通用命名约定。请参见 [http://msdn.microsoft.com/en-ca/library/windows/desktop/aa365247\(v=vs.85\).aspx](http://msdn.microsoft.com/en-ca/library/windows/desktop/aa365247(v=vs.85).aspx)。

1. 以管理员身份运行，打开命令提示符窗口。
2. 键入以下内容，指定要授予权限的用户：

```
rawaccess filename [{/permission_type} trustee]
```

其中，

- **filename** - 指向原始设备的 Windows 通用命名空间文件名路径，采用以下两种格式之一：
 - `\\.\logical_drive_name`
 - `\\.\physical_device_id`
- **logical_drive_name** - 用于表示磁盘分区的分区字母（例如：\\.\D:）。
- **physical_device_id** - 由 Windows 设备管理器分配的名称。可在控制面板中使用磁盘管理器搜索这些名称（例如：\\.\PhysicalDisk1）。
- **permission_type** - 访问操作的可选类型。如果未指定，则显示当前访问列表
 - **DENY** - 将访问拒绝条目添加到指定可信方的指定设备。
 - **GRANT** - 将访问允许条目添加到指定可信方的指定设备。该访问允许条目将与继承的所有现有访问允许权限合并在一起。
 - **REVOKE** - 从指定可信方的指定设备删除一个访问允许条目。
 - **SET** - 为指定可信方的指定设备设置一个访问允许条目。
- **trustee** - 用户帐户的有效登录 ID。

3. 查看访问权限以验证权限是否设置正确。

示例：

将原始设备 E: 的访问权限授予 user1。

```
rawaccess \\.\E: /Grant user1@company.com
```

现在，原始设备的访问权限显示为：

```
1. Allow All to COMPANY\user1 (User)
2. Allow Exec to \Everyone (Well Known Group)
3. Allow All to NT AUTHORITY\SYSTEM (Well Known Group)
4. Allow All to BUILTIN\Administrators (Alias)
5. Allow Exec to NT AUTHORITY\RESTRICTED (Well Known Group)
```

在 Windows 中设置原始设备的符号链接

符号链接简化了 Windows 系统中的原始设备访问，并会在重新启动之后一直保持。

前提条件

- Windows Vista 或更高版本的 Windows 操作系统。

- 您必须具有 Windows Vista 系统中的管理员特权，或在本地安全策略管理控制台中更改行为。

过程

1. 创建符号链接安装点，作为由驱动器盘符指定的 NTFS 文件系统，例如 T。
2. 在 Windows 命令提示符处，使用 **mklink** 命令创建符号链接，例如：

```
mklink T:\IQ_SYSTEM_MAIN \\.\Volume{GUID}
```

用于创建数据库或添加空间的脚本可以引用 T:\IQ_SYSTEM_MAIN 而不是格式 \\.\PhysicalDeviceN。

如果存储库处于脱机状态，则当系统恢复在线状态时符号链接将一直保留。

估计所需的空间和 dbspace

要避免在数据库或特定 dbspace 已满时出现困难，请在创建数据库和其中的对象之前估计 dbspace 的需求。

可以运行 SAP Sybase IQ 存储过程来估计数据库将需要多少空间和多少 dbspace。

1. 运行 **sp_iquestspace** 以根据基础数据库表中的行数来估计创建数据库时所需的空间量。对于您计划创建的每个表：**sp_iquestspace table_name, rows[, iqpagesize]**。
每个表所需的空间量将以 "RAW DATA index_size" 的形式返回。
2. 运行 **sp_iquestdbspaces** 根据给定的空间来确定要创建多少 dbspace 文件，以及它们的大小应当是多少。使用步骤 5 中计算的总索引大小作为 *minsize* 和 *maxsize* 参数：

```
sp_iquestdbspaces (dbsize [,iqpagesize]
[,minsize] [,maxsize] ...
```

所有这些计算均为估算。结果随您为数据库创建的列和索引不同而变化。有关各个过程的语法和用法说明，请参见参考：构件块、表和过程 > “系统过程”。

dbspace 管理示例

本节演示 dbspace 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 dbspace。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

使用以下内容创建小型数据库 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 dbspace 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 dbspace 的大小

上例中的主存储库对于表来说太小，无法添加到下一个示例中。本节中的 **ALTER DBSPACE** 命令可更改 dbspace 大小。

数据库 dbspacedb 具有 100MB 的 IQ 主存储库保留大小，这是使用 **CREATE DATABASE** 语句的 **IQ RESERVE** 参数设置的。此 IQ 主存储库 (IQ_SYSTEM_MAIN dbspace) 可以扩展 100 MB。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** 命令可将 IQ_SYSTEM_MAIN dbspace 扩展 10MB 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 `dbspace`。但是，可以使 `dbspace` 变得更小，从 `dbspace` 中扣除的大小将添加到保留大小中。

`IQ_SYSTEM_MAIN` `dbspace` 的大小现在为 20MB。将 `dbspace` 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 `dbspace` 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 `dbspace` 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 t1clhg ON t1(c1);
CREATE hng INDEX t2clhng ON t2(c1);
INSERT t1 VALUES(1);
INSERT t2 VALUES(2);
COMMIT;
```

显示有关 **dbspace** 的信息

显示 dbspacedb 数据库中所有 **dbspace** 的相关信息。以下示例将显示 iqdemo 数据库中的 **dbspace**。输出将分为两部分，以提高可读性：

```
sp_iqdbspace;
```

DBSpaceName	DBSpaceType	Writable	Online	Usage	TotalSize	Reserve
IQ_SYSTEM_MAIN	MAIN	T	T	25	10M	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 **dbspace** 的对象放置和空间使用情况的相关信息。

注意： 下面的两个示例将显示 iqdemo 数据库中的对象，以便更好地阐释输出。请注意，iqdemo 包括一个名为 iq_main 的用户 **dbspace** 示例，您自己的数据库中可能不存在该示例。

以下输出将分为几部分，以提高可读性：

```
sp_iqdbspaceinfo;
```

dbspace_name	object_type	owner	object_name	object_id	id
iq_main	table	DBA	emp1	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
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
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。以下信息来源于 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_cus- tomer_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》。

数据存储

DBA 可以确定哪些表和索引驻留在给定 **dbspace** 上，将这些对象重新定位到其它 **dbspace**，并在清空任意 **dbspace** 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 **dbspace** 的写入次数。

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 **dbspace** 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。

另请参见

- **ALTER DBSPACE** 语句（第 356 页）

在 RLV Dbspace 中添加文件

您可能希望在 RLV dbspace 中添加一个文件以获取额外容量来存储 RLV 事务日志。

前提条件

- SAP Sybase IQ 服务器具有一个 Simplex 数据库。
- 数据库中存在一个 RLV dbspace，且其处于联机状态。
- 在 dbspace 采用读写模式时，不存在任何启用 RLV 的对象。

过程

使用语句 **ALTER DBSPACE *dbspace name* ADD FILE *filename*:**

```
ALTER DBSPACE d1 ADD FILE 'rlv2.iq'
```

由于内存 RLV 存储库的性质，不能将文件指定为 READONLY。

从 RLV Dbspace 中删除文件

您可以从 RLV dbspace 中删除文件，前提是该文件不是 RLV dbspace 中唯一一个文件而且该文件未被使用。

前提条件

- SAP Sybase IQ 服务器具有一个 Simplex 数据库。
- 数据库中存在一个 RLV dbspace。
- RLV dbspace 是只读的，或者在 dbspace 采用读写模式时文件未在使用中。

过程

使用语句 **ALTER DBSPACE <*dbspace name*> DROP FILE <*dbspace filename*>:**

```
ALTER DBSPACE d1 DROP FILE rlv2
```

从 IQ_SYSTEM_TEMP 中删除 Dbfile

发出 **DROP DBSPACE** 命令以删除 dbfile。

前提条件

要删除 dbfile:

- 需要 **MANAGE ANY DBSPACE** 系统特权。
- dbfile 不能包含用户表中的任何数据。SAP Sybase IQ 不允许删除任何不为空的 dbfile。
- dbspace 中至少必须存在一个可用的读写 dbfile。
- 必须提交或回退所有在 dbfile 上使用较旧版本的表的事务。

过程

1. 使用系统表搜索全局临时表模式。运行 **DROP TABLE** 命令以重新定位驻留在 dbfile 上的所有对象。

```
SELECT * FROM systable WHERE table_type = 'GBL TEMP'
```

2. 验证 dbfile 上的所有空间是否均可用。

```
sp_iqfile IQ_SYSTEM_TEMP
```

如果 OkToDrop 列包含 "Y", 则可以删除 dbfile。

3. 删除 dbfile。对于 Multiplex 数据库, 使用绝对路径。

```
ALTER DBSPACE IQ_SYSTEM_TEMP DROP FILE FileHist3
```

dbfile 和输出文件

SYSDBFILE 视图可显示数据库中的所有 dbfile, 包括目录 dbspace 文件、IQ 消息文件、IQ 主 dbspace 和临时 dbspace 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 SYSDBFILE 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”。)

dbspace 属性和操作

Dbspace 状态可以是联机、脱机或动态脱机。

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)。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 dbspace 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 dbspace 重新联机, 但这只能更改 dbspace 在内存中的状态。

注意: 如果任何索引、数据或分区存在于脱机 dbspace 中, 则表数据将无法访问。

除联机、脱机或动态脱机状态以外, dbspace 还具有读写或只读状态。创建后, dbspace 即处于联机和读写状态。

dbspace 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。

可以创建、修改或删除 dbspace。

有关 Multiplex dbspace, 请参见管理: Multiplex。要在 SAP Control Center 中更改 dbspace 的状态, 请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助, 或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

dbspace 重命名准则

可以重命名 dbspace 或 dbfile 名称, 但不能重命名或删除目录 dbspace SYSTEM、IQ 主 dbspace IQ_SYSTEM_MAIN、IQ 临时 dbspace IQ_SYSTEM_TEMP、共享临时 dbspace IQ_SHARED_TEMP 和 IQ 消息 dbspace IQ_SYSTEM_MSG。

可以重命名 `IQ_SYSTEM_MAIN` 和 `IQ_SYSTEM_TEMP` 中文件的逻辑名，也可以更改 `IQ_SYSTEM_MSG` 文件的逻辑名，但不能更改 `SYSTEM` 中文件的逻辑名。不能使用 `ALTER DBSPACE RENAME TO` 对 `dbspace IQ_SYSTEM_MAIN`、`IQ_SYSTEM_TEMP`、`IQ_SYSTEM_MSG` 或 `SYSTEM` 进行重命名。

其它 dbspace

使用 **CREATE DBSPACE** 语句创建 `dbspace`。

新 `dbspace` 可以位于与现有 `dbspace` 相同或不同的磁盘驱动器上。需要 **MANAGE ANY DBSPACE** 系统特权。

有关原始设备和操作系统文件中 `dbspace` 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**，不能为目录存储库指定它们。

在原始设备上创建主存储库。

当您为新的 `dbspace` 指定原始设备时，**SAP Sybase IQ** 会自动确定其文件大小，并分配整个设备以用作 `IQ` 存储库。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，**SAP Sybase IQ** 将返回错误。

dbspace 数如何影响资源使用 and 性能

每个数据库的最大 `dbspace` 数量是可以调整的操作系统限制；每个数据库最多拥有 2,047 个 `dbspace`，再加上最多 12 个用于目录存储库的 `dbspace`。增加 `dbspace` 数量不会对内存使用或性能造成实际影响；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 `dbspace` 时会提高重叠的 **I/O** 性能。

将数据存储在原分区上时，每个驱动器可具有一个 `dbspace`。有关 `dbspace` 的大小限制，请参见参考：构件块、表和过程 > “物理限制”。

将数据存储在文件系统中时，可以利用存储系统中的分条功能。如果在多用户系统上使用操作系统或硬件分条，则带区大小应该至少为 **1MB**，或可能的最大大小。带区大小应该是 `IQ` 页大小的数倍。还可以配置 **SAP Sybase IQ** 以执行软件分条。

在添加 `dbspace` 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 `dbspace` 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

下面的命令将创建一个名为 `library` 的新 `dbspace`，指向 `IQ_SYSTEM_MAIN` `dbspace` 所在目录中名为 `library.iq` 的操作系统文件：

```
CREATE DBSPACE library
USING FILE library
'library.iq' SIZE 100 MB IQ STORE
```

要在 SAP Control Center 中创建 dbspace，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

在添加或删除 dbspace 之后，发出 **CHECKPOINT**。系统恢复将从最近的检查点之后开始。

删除 dbspace 的准则

发出 **DROP DBSPACE** 命令可以删除数据库文件。

要删除 dbspace：

- 需要 **MANAGE ANY DBSPACE** 系统特权。
- 它不能包含用户表中的任何数据。SAP Sybase IQ 不允许删除任何不为空的 dbspace。
- 它不能是必需的 dbspace：SYSTEM、IQ_SYSTEM_MAIN、IQ_SYSTEM_TEMP 或 IQ_SYSTEM_MSG。永远不要删除这些 dbspace，但如果 dbspace 中不包含用户创建的对象，则可从 IQ 主存储库或目录存储库中删除其它 dbspace。

要清空 dbspace，必须：

- 重新部署或删除 dbspace 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 SAP Sybase IQ 在 dbspace 中填充数据的方式，dbspace 不太可能仅在显式重定位表之后就变为空的，尤其是在正在使用磁盘分条的情况下。通常，无法通过截断 dbspace 中的表来清空 dbspace，因为即使空表也会占用一些空间。要重新定位表，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 dbspace 中。这些过程显示了每个表和索引在每个 dbspace 中所使用的块数。

要确定是否可以删除特定 dbspace，请运行 **sp_iqdbspace**。请查看“Block Types”列 (Blk Types)，该列显示每个 dbspace 的内容。如果 dbspace 只包含块类型“H”、“F”、“R”、“B”、“X”和“C”，则可以删除该 dbspace。

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 SAP Control Center 删除 dbspace 的说明，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

只读和读写 **dbspace** 和文件

如果文件状态为只读或读写，但拥有 **dbspace** 状态为只读，该文件则为只读。

将 **dbspace** 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 **dbspace**，管理员可以：

- 添加文件
- 重命名 **dbspace** 中的 **dbfile** 的文件路径（要求主 **dbspace** 脱机）
- 删除空文件
- 重命名 **dbspace** 或 **dbspace** 中的 **dbfile**

dbspace 和关联文件的状态

dbspace 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 dbspace	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		

将表分配给只读 **dbspace** 时，该表为只读表。将表分区分配给只读 **dbspace** 时，该分区为只读表分区。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** 是唯一一个具有这种行为的命令。）SAP Sybase IQ 对只读表或只读表分区上的 DDL 操作返回错误。

允许的 dbspace 转换

更改 dbspace 配置的能力取决于更改操作的类型以及特定的 dbspace 属性。更改操作由 dbspace 的状态（联机或脱机）、dbspace 的读写状态以及 dbspace 的类型控制。

表 6. 允许的 Dbspace 配置转换

状态	更改类型	是否允许用户主 Dbspace	是否允许 IQ_SHARED_TEMP、IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
联机 dbspace			
	ALTER DBSPACE OFF-LINE	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 RE-ADONLY	是（如果为 RW）	否
	ALTER DBSPACE READ-WRITE	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 的文件必须是最后一个要删除的文件。
	ALTER FILE READONLY	是（如果为 RW）	是（如果为 RW 并且不是最后一个 RW dbfile）。不允许将 IQ_SHARED_TEMP 中的文件设置为只读。

状态	更改类型	是否允许用户主 Dbospace	是否允许 IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
	ALTER FILE READWRITE	是（如果为 RO）	是（如果为 RO）
	ALTER FILE SIZE	是（如果为 RW dbospace 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 dbospace			
	ALTER DBSPACE OFF- LINE	否	N/A
	ALTER DBSPACE ONLINE	是	N/A
	ALTER DBSPACE RE- ADONLY	否	N/A
	ALTER DBSPACE READ- WRITE	否	N/A
	ALTER STRIPING 或 STRIPESIZEKB	是	N/A
	RENAME DBSPACE	是	N/A
	ADD FILE	否	N/A
	DROP FILE	是（如果为空）	N/A
	ALTER FILE RO	是	N/A
	ALTER FILE RW	是	N/A
	ALTER FILE SIZE	否	N/A
	ALTER FILE RENAME LOGICAL NAME	是	N/A

状态	更改类型	是否允许用户主 DbSPACE	是否允许 IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
	ALTER FILE RENAME PATH	是	N/A
动态脱机 dbSPACE			
	ALTER DBSPACE OFF- LINE	是（如果为 RO）	N/A
	ALTER DBSPACE ONLINE	是	N/A（对于 IQ_SYSTEM_MAIN 和 IQ_SYSTEM_TEMP），是（对于 IQ_SHARED_TEMP）
	ALTER DBSPACE RE- ADONLY	是（如果为 RW）	N/A
	ALTER DBSPACE READ- WRITE	否	N/A
	ALTER STRIPING 或 STRIPESIZEKB	是	N/A
	RENAME DBSPACE	是	N/A
	ADD FILE	否	N/A
	DROP FILE	是（如果为空）	N/A
	ALTER FILE READONLY	否	N/A
	ALTER FILE READWRITE	否	N/A
	ALTER FILE SIZE	否	N/A
	ALTER FILE RENAME LOGICAL NAME	是	N/A
	ALTER FILE RENAME PATH	否	N/A
只读 dbSPACE			
	ALTER DBSPACE OFF- LINE	是（如果联机）	N/A

状态	更改类型	是否允许用户主 DbSPACE	是否允许 IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
	ALTER DBSPACE ONLINE	是（如果脱机）	N/A
	ALTER DBSPACE RE- ADONLY	否	N/A
	ALTER DBSPACE READ- WRITE	是（如果联机）	N/A
	ALTER STRIPING 和 STRIPESIZEKB	是	N/A
	RENAME DBSPACE	是	N/A
	ADD FILE	是	N/A
	DROP FILE	是（如果为空）	N/A
	ALTER FILE READONLY	是（如果为 RW）	N/A
	ALTER FILE READWRITE	是（如果为 RO）	N/A
	ALTER FILE SIZE	否	N/A
	ALTER FILE RENAME LOGICAL NAME	是	N/A
	ALTER FILE RENAME PATH	是（如果脱机）	N/A
读写 dbSPACE			
	ALTER DBSPACE OFF- LINE	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 RE- ADONLY	是	否
	ALTER DBSPACE READ- WRITE	否	否

状态	更改类型	是否允许用户主 Dbospace	是否允许 IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
	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 的文件必须是最后一个删除的文 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
	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			

状态	更改类型	是否允许用户主 Dbospace	是否允许 IQ_SHARED_TEMP、 IQ_SYSTEM_MAIN 或 IQ_SYSTEM_TEMP
	ALTER FILE READONLY	是	是
	ALTER FILE READWRITE	否	否
	ALTER FILE SIZE	是（如果 dbospace 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

注意：

- 动态脱机表示 dbospace 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，允许进行的唯一修改是重命名路径，通过 ALTER DBSPACE IQ_SYSTEM_MSG RENAME 'filepath'. 命令实现

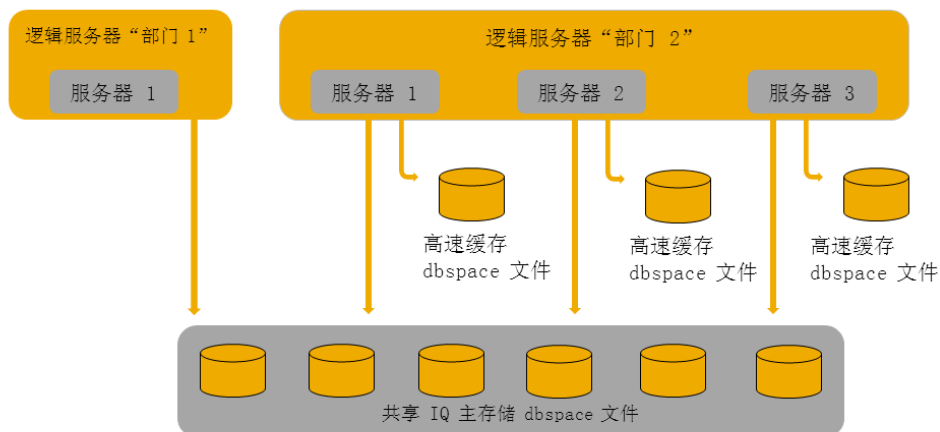
直连存储的高速缓存 Dbospace

通过为每个节点配置一个本地 *高速缓存 dbospace*，将直连存储设备（如固体状态硬盘 (SSD)）添加到 Multiplex 节点或 Simplex 服务器。本地直连存储为数据库提供了 SSD 缓冲池扩展，它能够总体提高 I/O 性能，尤其是提高共享 SAN 设备的性能。利用直连存储，Multiplex 配置得到的好处最大，Simplex 服务器次之。

高速缓存 dbospace 是独立的 dbospace，如 IQ 主存储库 (IQ_SYSTEM_MAIN)、IQ 临时存储库 (IQ_SYSTEM_TEMP) 以及 RLV (IQ RLV STORE) dbospace。与所有 dbospace 一样，高速缓存 dbospace 也可以包含一个或多个文件或设备。每个 Simplex 和 Multiplex 节点最多只能配置一个高速缓存 dbospace。

注意： 不要将 *高速缓存 dbospace* 与 IQ 主存储库的 *主缓冲区高速缓存* 相混淆。使用数据库启动实用程序 **start_iq** 选项 **-iqmc**（主缓冲区高速缓存大小）设置主缓冲区高速缓存大小。请参见实用程序指南。

创建并配置高速缓存 dbospace 之后，服务器将在该高速缓存 dbospace 中缓存本地使用只读页面（包含关联（首选）和非关联页面）的副本。当接下来从 IQ 主存储 dbospace 文件中读取内容时，服务器将首先检查该高速缓存 dbospace，查看是否存在相应的页面。如果存在，服务器将从高速 SSD 高速缓存 dbospace（而非共享 SAN IQ 主存储 dbospace）中读取页面，从而提高性能。



如果该高速缓存 dbspace 足够大，足以容纳节点的整个有效页集，将为您带来最大的性能优势。如果高速缓存 dbspace 比有效页集小，性能优势会大打折扣。

高速缓存 dbspace 与数据相似性功能一起使用可提高性能。数据相似性功能通过访问缓存数据来提升 Multiplex 服务器中的查询性能。系统将自动为 Multiplex 数据库启用相似性。相似性会尝试将分配给某个节点的分区页面存储在该节点的主缓冲区高速缓存内存中。但如果某个应用程序的有效页集超过了该节点的主缓冲区高速缓存内存大小，请添加直连存储作为辅助快速访问层。

创建高速缓存 Dbspace

创建一个高速缓存 dbspace，以将直连存储 (DAS) 设备添加到 Simplex 或 Multiplex 节点。该高速缓存 dbspace 通过减少从存储区域网络中读取内容来提高性能。每个 Multiplex 和 Simplex 节点只能有一个高速缓存 dbspace。

前提条件

需要 MANAGE ANY DBSPACE 系统特权。

过程

1. 连接到 Simplex 或 Multiplex 节点。

2. 创建高速缓存 dbspace 和 dbfile。在 CREATE DBSPACE 语句内指定 IQ CACHE STORE 关键字。

```
CREATE DBSPACE myDAS
USING FILE iqdas1
'/dev/rdisk/ssd104' IQ CACHE STORE
```

删除高速缓存 Dbspace

删除 Simplex 或 Multiplex 节点中的高速缓存 dbspace。与其它 dbspace 不同的是，无需在删除高速缓存 dbspace 之前先将其清空。

前提条件

- 一个高速缓存 dbspace 已在 Simplex 或 Multiplex 节点上。
- 需要 MANAGE ANY DBSPACE 系统特权。

过程

1. 连接到 Simplex 或 Multiplex 节点。
2. 删除高速缓存 dbspace:

```
DROP DBSPACE myDAS
```

将存储添加到高速缓存 Dbspace

通过指定文件或设备路径将更多直连存储添加到高速缓存 dbspace。

前提条件

- 一个高速缓存 dbspace 已在 Simplex 或 Multiplex 节点上。
- 需要 MANAGE ANY DBSPACE 系统特权。

过程

1. 连接到 Simplex 或 Multiplex 节点。
2. 指定一个文件或设备路径。

- 指定文件:

```
ALTER DBSPACE myDAS
ADD FILE iqdas2
'sampled.b.iqcache' SIZE 1024
```

- 指定设备路径:

```
ALTER DBSPACE myDAS
ADD FILE iqdas2
'/dev/rdisk/clt2d3s4'
```

删除高速缓存 DbSPACE 中的存储

通过删除文件或原始设备进而删除高速缓存 dbspace 中的直连存储。

前提条件

- 一个高速缓存 dbspace 已在 Simplex 或 Multiplex 节点上。
- 需要 **MANAGE ANY DBSPACE** 系统特权。

过程

即使高速缓存 dbspace 处于读写状态，也可以删除其中的 dbfile。

1. 连接到 Simplex 或 Multiplex 节点。
2. 使用 **ALTER DBSPACE** 语句删除单个文件或原始设备：

```
ALTER DBSPACE myDAS DROP FILE iqdas2
```

注意： 不允许使用多个 **DROP FILE** 子句。

禁用及重新启用高速缓存 DbSPACE

更改高速缓存 dbspace 的联机状态，以禁用高速缓存 dbspace 而不删除，或者更改高速缓存 dbspace 的脱机状态，以重新启用禁用的高速缓存 dbspace。

前提条件

- 一个高速缓存 dbspace 已在 Simplex 或 Multiplex 节点上。
- 需要 **MANAGE ANY DBSPACE** 系统特权。

过程

1. 连接到 Simplex 或 Multiplex 节点。
2. 使用 **ALTER DBSPACE** 语句禁用高速缓存 dbspace 或重新启用之前禁用的高速缓存 dbspace：

- 禁用高速缓存 dbspace：

```
ALTER DBSPACE myDAS OFFLINE
```

- 重新启用禁用的高速缓存 dbspace：

```
ALTER DBSPACE myDAS ONLINE
```

更改高速缓存 **Dbpace Dbfile** 的只读/读写状态

将高速缓存 **dbspace** 中的一个 **dbfile** 更改为只读或读写状态。

前提条件

- 一个高速缓存 **dbspace** 已在 **Simplex** 或 **Multiplex** 节点上。
- 需要 **MANAGE ANY DBSPACE** 系统特权。

过程

1. 连接到 **Simplex** 或 **Multiplex** 节点。
2. 使用 **ALTER DBSPACE** 语句将 **dbfile** 更改为只读或读写：

- 将高速缓存 **dpSPACE dbfile** 更改为只读：

```
ALTER DBSPACE myDAS ALTER FILE iqdas2 READONLY
```

- 将高速缓存 **dpSPACE dbfile** 更改为读写：

```
ALTER DBSPACE myDAS ALTER FILE iqdas2 READWRITE
```

监控高速缓存 **Dbpace**

使用 **sp_iqstatus**、**sp_iqdbspace**、**sp_iqfile** 和 **sp_iqcheckdb** 存储过程监控高速缓存 **dbspace** 及其 **dbfile** 的关键性能指示符。

前提条件

- **sp_iqstatus** 需要 **ALTER DATABASE**、**MANAGE ANY DBSPACE**、**MONITOR** 或 **SERVER OPERATOR** 系统特权。对于不具备上述系统特权之一的用户，必须为其授予 **EXECUTE** 特权。
- **sp_iqdbspace** 需要 **MANAGE ANY DBSPACE** 系统特权。对于不具备 **MANAGE ANY DBSPACE** 系统特权的用户，必须为其授予 **EXECUTE** 特权才能运行存储过程。
- **sp_iqfile** 需要 **MANAGE ANY DBSPACE** 系统特权。对于不具备 **MANAGE ANY DBSPACE** 系统特权的用户，必须为其授予 **EXECUTE** 特权。

过程

1. 运行 **sp_iqstatus** 以查看有关高速缓存 **dbspace** 及其 **dbfile** 的内存使用信息。

```
...
Main Store Out Of Space: N
Number of Cache Dbpace Files: 5
Number of Shared Temp DB Files: 0
Shared Temp Store Out Of Space: N
Number of Local Temp DB Files: 1
Local Temp Store Out Of Space: N
DB Blocks: 1-25600 IQ_SYSTEM_MAIN
```

```

DB Blocks: 522208-547807 MainUser
Cache Dbspace Blocks: 1-5120 ssd_dev_1
Cache Dbspace Blocks: 522208-527327 ssd_dev_2
Cache Dbspace Blocks: 1044416-1049535 ssd_dev_3
Cache Dbspace Blocks: 1566624-1571743 ssd_dev_4
Cache Dbspace Blocks: 2088832-2093951 ssd_dev_5
Local Temp Blocks: 1-25600 IQ_SYSTEM_TEMP
Create Time: 2013-08-21 06:27:05.444
Update Time: 2013-08-21 06:27:14.035
Main IQ Buffers: 1588, 100Mb
Temporary IQ Buffers: 1588, 100Mb
Main IQ Blocks Used: 5250 of 38400, 13%=20Mb, Max Block#: 5313
Cache Dbspace IQ Blocks Used: 197 of 25600, 0%=0Mb, Max Block#: 0
Shared Temporary IQ Blocks Used: 0 of 0, 0%=0Mb, Max Block#: 0
Local Temporary IQ Blocks Used: 65 of 12800, 0%=0Mb, Max Block#: 0
Main Reserved Blocks Available: 12800 of 12800, 100%=50Mb
Shared Temporary Reserved Blocks Available: 0 of 0, 0%=0Mb
Local Temporary Reserved Blocks Available: 12800 of 12800,
100%=50Mb
...

```

测量	描述
高速缓存 Dbspace 文件数	数据库中高速缓存 dbspace dbfile 的数量。
高速缓存 Dbspace 块	标识高速缓存 dbspace 块和相应的存储设备 dbfile 名称。
已用高速缓存 Dbspace IQ 块	已用 IQ 块数与 IQ 块总数进行比较。也可以按百分比显示使用情况。如果此百分比偏高，则考虑添加额外存储。

有关详细信息，请参见参考：构件块、表和过程中的 `sp_iqstatus` 过程。

- 运行 `sp_iqdbspace` 以查看有关高速缓存 dbspace 的详细信息，包括其读/写状态以及联机/脱机状态。查看与在 `CREATE DBSPACE` 语句中指定的高速缓存 dbspace 名称（本例中为 `myDAS`）相对应的行。高速缓存 dbspace 在 `DBSpaceType` 列中显示 `CACHE`：

```

IQ_SYSTEM_MAIN MAIN          T T 21 100M 0B 1 1 T 1K 1H,5088F,32D,128M
N
IQ_SYSTEM_TEMP TEMPORARY     T T 1   100M 0B 1 1 T 1K 1H,64F,16A N
myDAS CACHE                 T T 1   100M 0B 5 5 T 1K 5H,192F Y
MainUser MAIN                T T 1   100M 0B 1 1 T 1K 1H Y

```

列	描述
Writable	T（可写）或 F（不可写）。
Online	T（联机）或 F（脱机）。
Usage	所有文件当前使用的 dbspace 占整个 dbspace 的百分比。
TotalSize	dbspace 中所有文件的总大小，以 B（字节）、K（千字节）、M（兆字节）、G（千兆字节）、T（千吉字节）或 P（千万亿字节）为单位。

列	描述
Reserve	dbspace 中可以添加到所有文件的保留空间总大小。
NumFiles	dbspace 中的文件数。
NumRWFiles	dbspace 中的读/写文件数。
StripingOn	T (On) 或 F (Off)。
StripeSize	如果磁盘条带化已开启，则始终为 1。
BlkTypes	用户数据和内部系统结构占用的空间。
OkToDrop	"Y" 表示可删除 dbspace；否则为 "N"。

3. 运行 **sp_iqfile** 以显示有关高速缓存 dbspace 中每个 dbfile 的详细信息。查看与在 **CREATE DBSPACE** 语句中指定的高速缓存 dbspace 名称（本例中为 myDAS）相对应的行：

```
IQ_SYSTEM_MAIN IQ_SYSTEM_MAIN exampleDB.iq MAIN RW T 21 100M 0B 1K 1H,5088F,32D,128M 1 25600
MainUser MainUser exampleDB.iq2 MAIN RW T 1 100M 0B 1K 1H 522208 547807
myDAS ssd_dev_1 /dev/raw/ssd_dev_1 CACHE RW T 2 20M 0B 1K 1H,64F 1 5120
myDAS ssd_dev_2 /dev/raw/ssd_dev_2 CACHE RW T 1 20M 0B 1K 1H,32F 522208 527327
myDAS ssd_dev_3 /dev/raw/ssd_dev_3 CACHE RW T 1 20M 0B 1K 1H,32F 1044416 1049535
myDAS ssd_dev_4 /dev/raw/ssd_dev_4 CACHE RW T 1 20M 0B 1K 1H,32F 1566624 1571743
myDAS ssd_dev_5 /dev/raw/ssd_dev_5 CACHE RW T 1 20M 0B 1K 1H,32F 2088832 2093951
IQ_SYSTEM_TEMP IQ_SYSTEM_TEMP exampleDB.iqtmp TEMPORARY RW T 1 100M 0B 1K 1H,64F,16A 1 25600
```

列	描述
DBFileName	逻辑文件名。
Path	物理文件或原始分区的位置。
SegmentType	高速缓存。
RWMode	dbspace 的模式：读写 (RW) 或只读 (RO)。
Online	T (联机) 或 F (脱机)。
Usage	此文件当前使用的 dbspace 占整个 dbspace 的百分比。在 Multiplex 配置中针对辅助节点运行时，此列将显示 NA。

列	描述
DBFileSize	文件或原始分区的当前大小。对于原始分区来说，此大小值可以小于实际大小。
Reserve	dbspace 中可添加到此文件的保留空间。
StripeSize	如果磁盘条带化已开启，则始终为 1。
BlkTypes	用户数据和内部系统结构占用的空间。
FirstBlk	分配给文件的第一个 IQ 块号。
LastBlk	分配给文件的最后一个 IQ 块号。
OkToDrop	"Y" 表示可删除文件；否则为 "N"。

有关详细信息，请参见参考：构件块、表和过程中的 `sp_iqfile` 过程。

下一步

可使用 **sp_iqcheckdb** 存储过程对高速缓存 dbspace 执行数据库验证。请参见《参考：构件块、表和过程》中的“`sp_iqcheckdb` 过程”。

管理 SAP Sybase IQ 数据库对象

SAP Sybase IQ 用于创建、更改和删除数据库对象，如表和视图。

注意： SAP Sybase IQ 由目录存储库和 IQ 存储库组成。本节将介绍如何在 IQ 存储库中创建对象。在目录存储库中创建的表具有 SQL Anywhere 表的特性。要在目录存储库中创建表，请参见 SQL Anywhere 文档。

SAP 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，需要决定在哪里存储数据、数据库需要多少空间以及哪些人能够定义或修改数据库对象。SAP Sybase IQ 提供工具和进程来协助您做出决定。

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。

无论使用何种设计工具，通常由数据库管理员 (DBA) 来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各位成员合作，以了解数据是如何使用的。DBA 还需要了解 SAP Sybase IQ 数据库的基本概念。

SAP Sybase IQ 数据库是为了用作数据仓库而进行优化的关系数据库。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、表中所允许的数据类型、各表之间的关系、用于存储表数据的索引以及用于控制哪些人具有数据访问权限的视图。在创建数据库之前，请查看 SAP Sybase IQ 简介中的概念。

视图管理

视图是计算得出的表。

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。
- 无法向视图指派完整性约束和键。
- 视图引用基表中的信息，但不保留该信息的副本。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：

- 不允许访问不相关的信息。
- 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。
- 可以在数据库中集中一般查询的定义。

创建视图

在一个或多个表上操作 **SELECT** 语句并产生一个结果集，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，都将执行关联的 **SELECT** 语句来检索合适的数据库。

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。

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

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** 子句将在视图的行上强加顺序。可以在视图定义中使用 **GROUP BY** 子句、子查询和连接。

只在顶级 **SELECT** 列表中支持标量值子查询（视图、派生表或子查询均不支持）。有时顶级 **SELECT** 的 **FROM** 子句中使用的视图或派生表非常简单，可以“展平”到顶级 **SELECT**。因此，之前的规则实际仅针对子查询、未展平的视图以及未展平的派生表强制执行。例如：

```
CREATE VIEW test_view AS SELECT testkey, (SELECT COUNT(*) FROM
tagtests WHERE tagtests.testkey = testtrd.testkey ) FROM
testtrd
```

```
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，可以调整 **SELECT** 查询本身，直到其以所需格式提供完全符合需要的结果。调整好 **SELECT** 语句之后，即可在查询前面添加一个短语来创建视图。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，有些视图不允许，具体取决于其关联的 **SELECT** 语句。

不能在包含下面内容的视图中执行更新、插入或删除：

- 集合函数，如 **COUNT(*)**
- **SELECT** 语句中的 **GROUP BY** 子句
- **UNION** 操作

在所有这些情况中，无法将 **UPDATE**、**INSERT** 或 **DELETE** 转换成对基础表的操作。

警告！ 请不要删除 **dbo** 用户 ID 拥有的视图，该用户 ID 拥有系统对象。删除此类视图或将它们转化为表可能导致意外的问题。

修改视图的准则

可以使用 **ALTER VIEW** 语句修改视图。

ALTER VIEW 语句使用新的定义替换视图定义；它不对现有的视图定义进行修改。

ALTER VIEW 语句保留对视图的权限。

示例

例如，若要在上述 `DepartmentSize` 视图中将列名替换为更具信息性的名称，可以使用以下语句：

```
ALTER VIEW DepartmentSize
  (DepartmentID, NumEmployees)
AS
  SELECT DepartmentID, COUNT(*)
  FROM Employees
  GROUP BY DepartmentID
```

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，或者必须对视图具备带 **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

要用 **Interactive SQL** 从数据库中删除视图，应使用 **DROP** 语句。以下语句可删除 `DepartmentSize` 视图：

```
DROP VIEW DepartmentSize
```

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
```

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，并且可以复制和修订这些表。

创建表的准则

使用 **Interactive SQL** 创建表。

*使用 **Interactive SQL** 创建表*

用于创建表的 **SQL** 语句为 **CREATE TABLE**。

本节介绍如何使用 **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 **Interactive SQL** 以外，您还可以使用其它工具创建表。此处介绍的 **SQL** 语句与您使用的工具无关。

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，SAP 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**。

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，但是如有任何其它连接使用要更改的表，则将无法执行这些语句。**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：语句和选项中的 **CREATE DOMAIN** 语句。

表类型

SAP Sybase IQ 可识别四种表类型。

- 基表
- 局部临时表
- 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。

使用 **CREATE TABLE** 的 **GLOBAL TEMPORARY** 选项创建全局临时表。

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。当断开连接或提交后，将删除这些行。给定连接继承用户首次引用全局临时表时该表的模式。在某一个 **Multiplex** 服务器上创建的全局临时表也将所有其它 **Multiplex** 服务器上创建。

要选择添加到临时表，请使用如下语法：

```
SELECT * INTO #TableTemp FROM lineitem
WHERE l_discount < 0.5
```

注意： SAP Sybase 强烈建议，在编写 **SELECT INTO** 临时表的脚本时，在 **CAST** 表达式中包含非基列的选择列表项。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。

如果连接中存在某一局部临时表，则尝试创建同名的基表或全局临时表将失败，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，假定有以下顺序：

```
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局部临时表 t1，一直到局部临时表被连接删除。

删除和更改全局临时表

删除全局临时表的方式与删除基表的方式相同，即使用 **DROP TABLE** 语句删除。

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。

表位置

SAP Sybase IQ 在当前数据库中创建表。

如果连接到 SAP Sybase IQ 数据库，则表位置如下：

表 7. 表位置

表类型	允许位置	缺省位置
永久表	IQ 存储库、目录存储库	IQ 存储库

表类型	允许位置	缺省位置
全局临时表	IQ 临时存储库、目录存储库	IQ 临时存储
局部临时表	IQ 临时存储库或目录存储库；仅对创建此表的用户可见	IQ 临时存储

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建立索引。

这些选项使 SAP Sybase IQ 能够为实施唯一性的列创建 **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。

优化存储和查询性能

在 SAP Sybase IQ 数据库中创建永久表时，SAP Sybase IQ 自动将永久表列存储到缺省索引中，形成称为“投影”的查询类型。

SAP 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）
- 列的数据类型及其宽度
- **IQ PAGE SIZE** 选项（**CREATE DATABASE** 或 SAP Control Center 创建数据库向导）

IQ UNIQUE 的影响

IQ UNIQUE 是可选列约束，它显式定义列的基数并决定着该列装载为 Flat FP 还是 NBit。

IQ UNIQUE 定义列的预期基数并确定是将该列装载为 Flat FP 还是 NBit。IQ UNIQUE(*n*) 值显式设置为 0 会将该列装载为 Flat FP。不带 IQ UNIQUE 约束的列将隐式装载为 NBit，一直到 FP_NBIT_AUTOSIZE_LIMIT 和 FP_NBIT_LOOKUP_MB 选项定义的限制：

- FP_NBIT_AUTOSIZE_LIMIT 限制装载为 NBit 的不同值的数量
- FP_NBIT_LOOKUP_MB 为 NBit 字典总大小设置阈值
- FP_NBIT_ROLLOVER_MAX_MB 为从 NBit 至 Flat FP 的隐式 NBit 切换设置字典大小
- FP_NBIT_ENFORCE_LIMITS 强制执行 NBit 字典大小限制。此选项缺省设置为 OFF

不必使用 *n* 值小于 FP_NBIT_AUTOSIZE_LIMIT 的 IQ UNIQUE。自动大小调整功能会自动将所有中低基数列的大小调整为 NBit。如果想要将列装载为 Flat FP，或者想要在不同值的数量超出 FP_NBIT_AUTOSIZE_LIMIT 时将列装载为 NBit，请使用 IQ UNIQUE。

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 **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。如果没有上述任何索引，并且估计偏差非常大（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，需运行 **sp_iqrebuildindex** 过程。

UNIQUE 和 IQ UNIQUE 之间的区别

IQ UNIQUE (n) 接近给定列中非重复值的数量。**IQ UNIQUE (n)** 值显式设置为 "0" 会将该列装载为 **Flat FP**。没有 **IQ UNIQUE** 的列或 **IQ UNIQUE (n)** 值小于等于 **FP_NBIT_AUTOSIZE_LIMIT** 和 **FP_NBIT_LOOKUP_MB** 选项所定义的限制的列隐式装载为 **NBIT**。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，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（即雇员的社保号码）只能在该列中出现一次。此唯一性扩展至 **NULL** 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，用以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
```

这些示例说明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，一旦进行了更改，引用该列的存储过程、视图和任何其它项都可能无法工作。

删除表的准则

删除一个表会删除该表中的所有记录，然后删除表定义。

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 语句还会删除表上的所有索引。

如果只希望删除数据行，而不是表本身，请使用 **TRUNCATE TABLE** 语句。如果在其他用户从表中读取数据时截断表，将应用表版本控制的一般规则，即旧表版本将一直保留直到读取者完成事务为止。

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。

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。

创建主键

以下语句创建与之前相同的 `skill` 表，只是添加了一个主键：

```
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唯一，在本示例中，这就意味着您不能为多行指定同一个 `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**。

创建外键

每一外键关系将一列中的候选键（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，其中保存每个雇员的各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` 表定义具有由此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特定技能，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，但根据需要可以具有多个外键。

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”。

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 **SAP Control Center** 的执行 SQL 窗口或 **Interactive SQL** 浏览该视图中的信息。在命令窗口中键入以下语句，以查看到 `SYS.SYSTAB` 视图中的所有列：

```
SELECT *
FROM SYS.SYSTAB
```

表分区

表分区是将大表拆分为多个便于管理的小存储对象，这样可提高性能。各个分区共用父表的相同逻辑属性，但可放置在单独的 `dbspace` 中并分别进行管理。

注意： 如果分区数据存在于脱机 `dbspace` 中，则表数据将无法访问。确保所有 `dbspace` 均处于联机状态。

另请参见

- 管理数据存储 (第 79 页)

限制

表分区存在以下限制。

- 与散列范围分区相同，范围分区和复合分区模式要求有单独授权的 VLDB 管理组件。
- 只能对基表进行分区；不能对全局临时表或声明的局部临时表进行分区。一个表分区的所有行在物理上处于同一位置。
- 范围分区或子分区可驻留在不同的 `dbspace` 中，仅对于大于 255 个字节的 BLOB 或 CLOB 列或者 CHAR、VARCHAR 或 VARBINARY 列推荐使用单个 `dbspace`。

分区键

分区键不能包含：

- LOB(BLOB 或 CLOB) 列
- BINARY 或 VARBINARY 列
- 长度超过 255 个字节的 CHAR 或 VARCHAR 列
- BIT 列
- FLOAT/DOUBLE/REAL 列

分区键列可包含 NULL 和 DEFAULT 值。将所有的 NULL 值映射到同一个分区。

散列分区最多可含有 8 个分区键列，范围分区或子分区只能含有 1 个键列。

最多可以定义 1024 个范围分区或子分区。

DML 操作

您可以执行的 DML 操作包括 LOAD、INSERT、DELETE、TRUNCATE 和 TRUNCATE TABLE PARTITION。不支持对分区键列执行 UPDATE，这会生成一个错误。但可对分区表的所有其它列执行 UPDATE。

对 READ-ONLY 表或 READ-ONLY 表分区执行 DML 操作时，SAP Sybase IQ 会生成一个异常。如果给定行超出指定的分区范围，INSERT 和 LOAD 语句或通过可更新游标执行的 INSERT 将生成一条错误，同时操作回退。

DDL 操作

DROP、RENAME、PARTITION、UNPARTITION、MERGE、SPLIT 和 MOVE 分区存在以下限制：

表 8. 关于对分区执行 DDL 操作的限制

操作	限制
删除	不能从分区键或范围子分区键中删除列。不能删除分区表的最后一个分区。
对未分区表进行分区	不能对现有表进行散列分区。
合并两个相邻分区	两个分区必须位于同一个 dbspace 中。不需要移动数据。
拆分区	<p>将一个分区拆分为两个分区后，所有行必须属于第一个分区。拆分后的分区必须与原始分区位于同一个 dbspace 上，以免移动数据。</p> <p>仅当不必移动任何数据时才能拆分范围子分区。要拆分的子分区的所有现有行在拆分后必须保留在第一个子分区内。</p>
将分区移至新 dbspace	该分区的所有行均被移至新 dbspace 的数据页中。在新 dbspace 中需具备 CREATE 权限。

另请参见

- 范围分区（第 127 页）
- 散列分区（第 128 页）
- 散列范围分区（第 129 页）
- **ALTER TABLE** 语句（第 368 页）
- **CREATE TABLE** 语句（第 410 页）

范围分区

范围分区根据为每个分区建立的 *partition-key* 值范围拆分大表。

作为信息生命周期管理策略的一部分，范围分区能够缩短备份和存储时间；提供更精细粒度的数据验证；同时支持分层存储。

在 *range-partitioning-scheme* 中，*partition-key* 列的值确定该行所属的分区：

```
range-partitioning-scheme:
    PARTITION BY RANGE ( partition-key ) ( range-partition-decl [ ,
range-partition-decl ... ] )
```

range-partition-declaration 确定如何将指定的分区放在 **dbspace** 中：

```
range-partition-declaration:
    range-partition-name VALUES <= ( {constant | MAX } ) [ IN
dbspace-name ]
```

VALUES 子句标识每个分区的上限（按升序）。每个范围分区可置于自己的 **dbspace** 中，并可单独进行管理。在表分区集合中，分区名称必须唯一。

限制

范围分区被限定为有 1 个分区键列和最多 1024 个分区。

示例

最常见的范围分区形式是按日期对表进行分区。以下示例创建一个包含 6 列和 3 个分区的范围分区表 *bar*，用于基于日期向各个分区映射数据：

```
CREATE TABLE bar (
  c1 INT IQ UNIQUE(65500),
  c2 VARCHAR(20),
  c3 CLOB PARTITION (P1 IN Dsp11, P2 IN Dsp12,
    P3 IN Dsp13),
  c4 DATE,
  c5 BIGINT,
  c6 VARCHAR(500) PARTITION (P1 IN Dsp21,
    P2 IN Dsp22),
  PRIMARY KEY (c5) IN Dsp2) IN Dsp1
PARTITION BY RANGE (c4)
(P1 VALUES <= ('2006/03/31') IN Dsp31,
 P2 VALUES <= ('2006/06/30') IN Dsp32,
 P3 VALUES <= ('2006/09/30') IN Dsp33);
```

另请参见

- 限制 (第 126 页)
- 散列分区 (第 128 页)
- 散列范围分区 (第 129 页)
- ALTER TABLE 语句 (第 368 页)
- CREATE TABLE 语句 (第 410 页)

散列分区

散列分区基于内部散列函数处理的 *partition-key* 值向各个分区映射数据。

散列分区将数据分配到可并行执行的逻辑分区中，从而可增强大表和分布式查询 (DQP) 的连接性能。

在 *hash-partitioning-scheme* 声明中，*partition-key* 是一个或多个列，其合成值将确定存储有各行数据的分区：

```
hash-partitioning-scheme:
PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
```

散列分区键被限定为最多有 8 个列，其声明的列宽总和小于等于 5300 个字节。对于散列分区，表创建者仅确定分区键列；分区的数量和位置将在内部确定。

限制

- 不能添加、删除、合并或拆分散列分区。
- 不能为散列分区键添加列或从中删除列。

示例

在以下示例中，表 `tbl42` 包含一个 `PRIMARY KEY`（列 `c1`）和一个 `HASH PARTITION KEY`（列 `c4` 和 `c3`）。

```
CREATE TABLE tbl42 (
  c1 BIGINT NOT NULL,
  c2 CHAR(2) IQ UNIQUE(50),
  c3 DATE IQ UNIQUE(36524),
  c4 VARCHAR(200),
  PRIMARY KEY (c1)
  PARTITION BY HASH ( c4, c3 )
)
```

以下示例显示了一种常见情况，即连接键既为主键又为散列分区键。

```
CREATE TABLE bar (
  c1 BIGINT NOT NULL,
  c2 CHAR(2) IQ UNIQUE(50),
  c3 DATE IQ UNIQUE(36524),
  c4 VARCHAR(200),
  c5 CLOB,
  PRIMARY KEY ( c1 )
  PARTITION BY HASH ( c1 )
)
```

另请参见

- 限制（第 126 页）
- 范围分区（第 127 页）
- 散列范围分区（第 129 页）
- `ALTER TABLE` 语句（第 368 页）
- `CREATE TABLE` 语句（第 410 页）

散列范围分区

散列范围分区是一种复合分区模式，它按范围将散列分区表划分为多个子分区。

散列范围分区兼具散列分区和范围分区的优势。散列分区提供最佳分布式查询性能；范围子分区是信息生命周期管理策略的一部分，用于增强管理任务。

在 *hash-range-partitioning-scheme* 声明中，**`SUBPARTITION BY RANGE`** 子句将向现有散列范围分区表添加一个新的范围子分区：

```
hash-range-partitioning-scheme:
  PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
  [ SUBPARTITION BY RANGE
    ( range-partition-decl [ , range-partition-decl ... ] ) ]
```

散列分区指定如何在逻辑上分布和托管数据；范围子分区指定如何在物理上放置数据。新的范围子分区通过散列进行逻辑分区，其散列分区键与现有的散列范围分区表相同。范围子分区键被限定为仅一列。

示例

以下示例（表 tbl42）包含一个 PRIMARY KEY（列 c1）、一个散列分区键（列 c4 和 c2）以及一个范围子分区键（列 c3）：

```
CREATE TABLE tbl42 (
  c1 bigint not null,
  c2 char(2) iq unique(50),
  c3 date
  c4 varchar(200),
  PRIMARY KEY (c1)) IN Dsp1
  PARTITION BY HASH (c4, c2)
  SUBPARTITION BY RANGE (c3)
  (P1 VALUES <= ( '2011/03/31' ) IN Dsp31,
   P2 VALUES <= ( '2011/06/30' ) IN Dsp32,
   P3 VALUES <= ( '2011/09/30' ) IN Dsp33);
```

在以下示例中，表 tbl43 包含不同的 **dbspace**，用于存储列 c4 的 RANGE 子分区。该语法类似于 RANGE 分区的现有语法。请注意 SUBPARTITION 在列 c4 中的使用。

```
CREATE TABLE tbl43 (
  c1 bigint not null,
  c2 char(2) iq unique(50),
  c3 date
  c4 varchar(200)
  SUBPARTITION (P1 in Dsp331, P2 in Dsp332, P3 in Dsp333),
  PRIMARY KEY (c1)) IN Dsp1
  PARTITION BY HASH (c4, c2)
  SUBPARTITION BY RANGE (c3)
  (P1 VALUES <= ( '2011/03/31' ) IN Dsp31,
   P2 VALUES <= ( '2011/06/30' ) IN Dsp32,
   P3 VALUES <= ( '2011/09/30' ) IN Dsp33) ;
```

另请参见

- 限制（第 126 页）
- 范围分区（第 127 页）
- 散列分区（第 128 页）
- ALTER TABLE 语句（第 368 页）
- CREATE TABLE 语句（第 410 页）

索引 SAP Sybase IQ 列

SAP Sybase IQ 索引旨在对特定类型的查询充分利用内存、磁盘和 CPU 周期。

- 索引大小通常非常小。可以对整个数据库建立完整索引，并且即席查询可在存储原始数据所需的相同空间中使用该索引。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，允许为任意次数的即席查询一次性优化数据。

索引概述

索引可以提高数据检索性能。SAP 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。

在表中装载数据时，SAP Sybase IQ 实际上按列而不是按行为表中每一列存储数据。相较于基于行的传统索引而言，以列方向存储为 IQ 索引提供了许多重要优势。

SAP Sybase IQ 为各列创建的缺省列存储结构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。

从逻辑上讲，您仍可以像在基于行的传统 SQL 数据库中一样访问数据。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，通常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。列存储优化了对特定属性执行选择或计算的功能。

索引类型比较

选择适合于您任务的列索引类型。

表 9. 索引类型比较

索引类型	作用
比较 (CMP) 索引 (第 134 页)	具有相同数据类型、精度和小数位数的任意两个不同列之间的索引关系。
包含 (WD) 索引 (第 132 页)	存储 CHAR、VARCHAR 或 LONG VARCHAR 数据的列字符串中的单词。

索引类型	作用
日期 (DATE) 和日期时间 (DTTM) 索引 (第 134 页)	处理涉及日期或日期时间数量的查询。
快速投影 (FP) 索引缺省情况下，数据库引擎自动针对所有列创建快速投影 (FP) 索引。FP 索引优化投影并允许数据库引擎计算特定类型的搜索条件。	优化投影并支持计算特定类型的搜索条件。缺省分配给任意永久表。
High_Group (HG) 索引 (第 139 页)	用于含整数数据类型的连接列。有效处理 GROUP BY 和等式运算。建议用于含有超过 1000 个不同值的列。
High_Non_Group (HNG) 索引 (第 141 页)	用于涉及到范围或集合的高基数运算。
Low_Fast (LF) 索引 (第 142 页)	适用于所含唯一值少于 1000 的列（如“是/否”或“家属人数”），或含 SUM、AVG 和 COUNT 的列。SAP Sybase IQ 中最快速的索引类型。
TEXT 索引 (第 143 页)	用于非结构化数据分析功能，是 SAP Sybase IQ 授权选项。
时间 (TIME) 索引 (第 143 页)	处理涉及时间数量的查询。

包含 (WD) 索引

使用包含 (WD) 索引可以存储 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。请参见用户定义的函数。

WD 的建议用法

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 Web 页中）。

下面限制适用于 WD 索引：

- 不能指定 UNIQUE 属性。
- 它只能与 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。
- 省略 DELIMITED BY 子句或指定 *separators-string* 值为空（使用单引号），则 SAP 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。

- 如果指定多个 **DELIMITED BY** 和 **LIMIT** 子句，则不会返回任何错误，但只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，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。**SAP Sybase IQ** 自动将 **CHAR** 列填充至最大列宽。如果列包含空白和字符数据，**WD** 索引数据查询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** 列代替 **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

使用 **WD** 索引的优缺点。

表 10. WD 的优缺点

优点	缺点
大量装载时可极大地提高性能。	潜在磁盘空间要求可能会非常大。
使用此索引，某些 LIKE 谓词的执行速度将变快。	如果强制执行唯一性则不可建立索引。
与此索引一起使用的 CONTAINS 谓词的优先级高于 LIKE 谓词。	只有在列中的数据为 CHAR 、 VARCHAR 或 LONG VARCHAR 时，才能使用此索引。
是对关键字或部分 URL 建立索引的最佳方式。	

比较 (CMP) 索引

比较 (CMP) 索引表示两列之间的关系。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 =)。

CMP 的建议用法

可以在 NULL 值、非 NULL 值或混合有 NULL 值和非 NULL 值的列上创建 **CMP** 索引。**CMP** 索引不能唯一。

数值数据类型和小数数据类型视为相同类型。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，下面命令创建表，并随后创建相应的 **CMP** 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)
```

CMP 索引的限制

以下限制适用于 **CMP**。

- 您可以删除 **CMP** 索引。
- **CMP** 索引不能唯一。
- 如果尝试更改或删除在 **CMP** 索引中定义的列，则会引发异常。
- 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。

日期 (DATE) 和日期时间 (DTTM) 索引

DATE 和 **DTTM** 索引类型处理涉及日期或日期时间数量的查询。

- **DATE** 索引用于在 DATE 数据类型的列中处理涉及日期数量的特定查询。
- **DTTM** 索引用于在 DATETIME 或 TIMESTAMP 数据类型的列中处理涉及日期时间数量的特定查询。

DATE 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、DATETIME 或 TIMESTAMP 列，请使用 **DATE** 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、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** 或 **DTTM** 索引获取结果。

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、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。

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);
```

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 主机变量。

使用 **DATE**、**TIME** 和 **DTTM** 索引处理时相应的 DATEPART 范围谓词和 IN 列表谓词包括：

- **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，如整数或整数类型的主机变量。例如：
DATEPART(WEEK, col1) !<23
DATEPART(YEAR, col1) = 2001
HOUR(col3) >= 1
- **BETWEEN……AND** 条件
BETWEEN 的左侧是日期/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，如整数或整数类型的主机变量。例如：
DATEPART(YEAR, col1) BETWEEN host-var1 AND host-var2
- **IN** 条件
IN 的左侧是日期/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部分（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' ;
```

比较运算符的一侧是列名，另一侧是常量表达式（常量值或主机变量）。

- **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** 索引的执行速度更快。

在下面三种特定情况中，使用 **DATE** 或 **DTTM** 索引可以显著提高性能：

- 谓词范围恰好是一年或多年（实际开始日期是某一年年初，而实际结束日期是某一年年末）。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月初，而实际结束日期是某个月月末）。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** 索引的执行速度更快。

DATE/DTTM 的优缺点

使用 **DATE** 或 **DTTM** 索引的优缺点。

表 11. DATE/TIME/DTTM 的优缺点

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。	使用的磁盘空间大于 HNG 索引。
可以创建和删除 DATE 或 DTTM 索引。	实现快速相等仍需要 LF 或 HG 索引。
	只有列中的数据为 DATE 、 DATETIME 或 TIMESTAMP 数据类型时，才能使用这些索引。

DATE/DTTM 索引的限制

适用于 **DATE** 和 **DTTM** 索引的限制包括：

- 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** 索引通常可达到 **HNG** 执行速度的两倍。因此，不必在 **DATE**、**TIME**、**DATETIME** 或 **TIMESTAMP** 数据类型的列上使用除 **DATE**、**TIME** 或 **DTTM** 索引之外的 **HNG** 索引。

DATE/DTTM 列的其它索引

如果在 **WHERE** 子句、**ON** 条件句或 **GROUP BY** 子句中引用 **DATE**、**TIME**、**DATETIME** 或 **TIMESTAMP** 数据类型的列，建议始终对这些列使用 **DATE**、**TIME** 或 **DTTM** 索引。

此外，**HG** 或 **LF** 索引也可能适用于 **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，并且不同值的数量少于 1000（即少于三年的日期），则同样推荐使用 **LF** 索引。

快速投影 (FP) 索引

缺省情况下，数据库引擎自动针对所有列创建快速投影 (**FP**) 索引。**FP** 索引优化投影并允许数据库引擎计算特定类型的搜索条件。

FP 索引是一个包含 n 个固定长度条目的数组，其中 n 表示表中的行数。应用于列的 **IQ UNIQUE** 存储指令决定是将列装载为 **flat FP** 还是 **NBit**。**Flat FP** 索引包含实际列单元值。**NBit** 是压缩模式，使用 n 位为数据所存储的字典建立索引。除 **LOB** 外的所有数据类型（包括字符和二进制类型）以及 **BIT** 数据类型都可能是 **NBit** 列。

IQ UNIQUE

IQ UNIQUE 定义列的预期基数并确定是将该列装载为 **Flat FP** 还是 **NBit**。**IQ UNIQUE(n)** 值显式设置为 0 会将该列装载为 **Flat FP**。不带 **IQ UNIQUE** 约束的列将隐式装载为 **NBit**，一直到 **FP_NBIT_AUTOSIZE_LIMIT** 和 **FP_NBIT_LOOKUP_MB** 选项定义的限制：

- **FP_NBIT_AUTOSIZE_LIMIT** 限制装载为 **NBit** 的不同值的数量
- **FP_NBIT_LOOKUP_MB** 为 **NBit** 字典总大小设置阈值
- **FP_NBIT_ROLLOVER_MAX_MB** 为从 **NBit** 至 **Flat FP** 的隐式 **NBit** 切换设置字典大小
- **FP_NBIT_ENFORCE_LIMITS** 强制执行 **NBit** 字典大小限制。此选项缺省设置为 **OFF**

不必使用 n 值小于 **FP_NBIT_AUTOSIZE_LIMIT** 的 **IQ UNIQUE**。自动大小调整功能会自动将所有中低基数列的大小调整为 **NBit**。如果想要将列装载为 **Flat FP**，或者想要在不同值的数量超出 **FP_NBIT_AUTOSIZE_LIMIT** 时将列装载为 **NBit**，请使用 **IQ UNIQUE**。

重建 FP 列索引

sp_iqindexmetadata 显示有关列索引的详细信息，包括 FP 索引类型 (Flat FP 或 NBIT)、非重复计数、IQ UNIQUE *n* 值以及字典大小。

sp_iqrebuildindex 将 Flat FP 重建为 NBIT，或将 NBIT 重建为 Flat FP。
index_clause 可以将 IQ UNIQUE *n* 重置为显式值，范围从 '0' (将 NBIT 列重新转换为 Flat FP) 到 FP_NBIT_AUTOSIZE_LIMIT 和 FP_NBIT_LOOKUP_MB 中定义的界限。如果数值超过 *n* 值且 FP_NBIT_ENFORCE_LIMITS=ON，该操作将回退。如果 FP_NBIT_ENFORCE_LIMITS=OFF (缺省值)，NBIT 字典大小将继续增长。

其它信息

- 参考：语句和选项 > “数据库选项” > “按字母顺序排列的选项列表” > “FP_NBIT_AUTO_LIMIT”
- 参考：语句和选项 > “数据库选项” > “按字母顺序排列的选项列表” > “FP_NBIT_LOOKUP_MB”
- R 参考：语句和选项 > “数据库选项” > “按字母顺序排列的选项列表” > “FP_NBIT_ROLLOVER_MAX_MB”
- 参考：语句和选项 > “数据库选项” > “按字母顺序排列的选项列表” > “FP_NBIT_ENFORCE_LIMIT”
- 参考：构件块、表和过程 > “系统过程” > “按字母顺序排列的系统存储过程列表” > “sp_iqrebuildindex”
- 参考：构件块、表和过程 > “系统过程” > “按字母顺序排列的系统存储过程列表” > “sp_iqindexmetadata”

High_Group (HG) 索引

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。

HG 的建议用法

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。

在以下情况下使用 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 索引可以提高对多列执行 ORDER BY 查询的性能。这一更改对用户透明，但是可提高查询性能。

注意： 外键列需要自己单独的 HG 索引。

High_Group 的优缺点

High_Group 索引既有优点也有缺点。

表 12. HG 的优缺点

优点	缺点
可快速处理具有 GROUP BY 的查询。	该索引需要比 HNG 索引更多的磁盘空间（可能需要比原始数据多三倍的空间）。 此索引类型填充数据和删除需要的时间最长。 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。 不建议将此索引用于 FLOAT 、 REAL 和 DOUBLE 数据。

与其它索引比较

LF 和 HNG 索引的 HG 比较。

LF

决定因素是唯一值数目。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，则请使用 **Low_Fast**

HNG

决定因素为，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，则使用 **High_Group**（可以单独使用也可以与 **High_Non_Group** 配合使用）。在其它情况下，请使用 **High_Non_Group** 节省磁盘空间。

HG 列的其它索引

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，SAP Sybase IQ 将在缺省情况下创建 **High_Group** 索引，而不必指定索引类型。

SAP 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，SAP 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，并且可以选择允许空值。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 **NULL** 值并且不实施约束（主键或唯一），则 SAP Sybase IQ 允许在用户创建的唯一多列 **HG** 索引的数据值中使用 **NULL** 值。有关多列索引的详细信息，请参见参考：语句和选项中的 **CREATE INDEX** 语句。

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，还可以优化下推和子查询。

High_Non_Group (HNG) 索引

为范围搜索添加 **HNG** 索引。

HNG 索引需要的磁盘空间比 **HG** 索引大约少三倍。仅基于这一点，当不需要执行组操作时，请使用 **HNG** 索引代替 **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，或者您希望实施唯一性，则使用 **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** 索引可能会大幅度降低性能。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**。

HNG 的建议用法

不建议将 **HNG** 索引用于 **GROUP BY** 查询。

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

High_Non_Group 索引既有优点也有缺点。

表 13. HNG 优点/缺点

优点	缺点
<p>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。</p> <p>如果列包含大量唯一值，通常此索引的速度最快（也有一些例外情况）。</p>	<p>不建议将此索引类型用于 GROUP BY 查询。</p> <p>如果实施唯一性则不可用。</p> <p>如果列中的数据为 FLOAT、REAL、DOUBLE、BIT、BINARY、VARBINARY、CHAR > 255 字节 或 VARCHAR > 255 字节，则不能使用此索引类型。</p>

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**。

HNG 列的其它索引

High_Group 索引同样适用于 **HNG** 列。

Low_Fast (LF) 索引

此索引非常适用于含唯一值非常少（低于 1,000）的列，这些唯一值可能包括：性别、是/否、True/False、家属人数、工资级别等。**LF** 是 SAP 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

请在下列情况下使用 **LF** 索引：列的唯一值数目少于 1,000，或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

Low_Fast 索引既有优点也有缺点。

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。	此索引类型最多能用于 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，则不能使用此索引类型。

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 索引类型可能适用于 **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。仅在唯一值数目较大时考虑其它索引类型（**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。

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。

使用 Interactive SQL 创建 TEXT 索引

必须先在要搜索的列上创建 **TEXT** 索引，然后才能执行全文本搜索。

TEXT 索引存储带索引的列中各个术语的位置信息。

1. 以具有 **CREATE ANY INDEX** 或 **CREATE ANY OBJECT** 系统特权的用户身份或以基础表的所有者身份连接到数据库。
2. 执行 **CREATE TEXT INDEX** 语句。

以下示例在 `iqdemo` 数据库中 `Customers` 表的 `CompanyName` 列上创建一个 **TEXT** 索引，`myTxtIdx`。该示例使用 `default_char` 文本配置对象。

```
CREATE TEXT INDEX myTxtIdx ON Customers
( CompanyName ) CONFIGURATION default_char
```

时间 (TIME) 索引

使用 **TIME** 索引处理涉及到时间数量的查询。

TIME 索引用于数据类型为 **TIME** 的列中。

TIME 索引类型的推荐使用

在含有时间函数和运算的查询中使用 **TIME** 列时，将用到 **TIME**。

- 含 **DATEPART** 相等谓词 (`=`、`!=`)、**DATEPART** 范围谓词 (`>`、`<`、`>=`、`<=`、`!`、`>!`、`<!`、**BETWEEN**) 和 **DATEPART IN** 列表谓词的查询
- 含范围谓词 (`>`、`<`、`>=`、`<=`、**BETWEEN**) 的查询

注意： 对于用于 **TIME** 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 **TIME** 索引将获取结果。

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 **TIME** 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。

TIME 的优缺点

TIME 索引既有优点也有缺点。

优点	缺点
同其它索引类型相比，对时间数量的查询可以更快得到解决。 可以创建和删除 TIME 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。 只有列中数据为 TIME 数据类型时，才能使用该索引。

TIME 索引的限制

以下限制当前适用于 TIME 索引：

- 不能使用 UNIQUE 关键字。
- 只能在单个列中进行创建。
- 不支持在日期部分使用毫秒。

选择索引的标准

为任意给定列定义的一组索引都会对查询处理的速度产生巨大影响。

主要有四种用于选择索引的条件：

- 唯一值数目
- 查询的类型
- 磁盘空间使用情况
- 数据类型

请结合使用上述所有条件对应的建议，不要单独使用某些条件。同时，谨记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。尝试在预期查询结果中预计唯一值数目和值总数。

索引中的唯一值数目

索引根据其包含的唯一（不同）值数目进行优化。

IQ UNIQUE 定义列的预期基数并确定是将该列装载为 Flat FP 还是 NBit。IQ UNIQUE(*n*) 值显式设置为 0 会将该列装载为 Flat FP。不带 IQ UNIQUE 约束的列将隐式装载为 NBit，一直到 FP_NBIT_AUTOSIZE_LIMIT 和 FP_NBIT_LOOKUP_MB 选项定义的限制：

- FP_NBIT_AUTOSIZE_LIMIT 限制装载为 NBit 的不同值的数量
- FP_NBIT_LOOKUP_MB 为 NBit 字典总大小设置阈值

- **FP_NBIT_ROLLOVER_MAX_MB** 为从 NBit 至 Flat FP 的隐式 NBit 切换设置字典大小
- **FP_NBIT_ENFORCE_LIMITS** 强制执行 NBit 字典大小限制。此选项缺省设置为 OFF

不必使用 n 值小于 **FP_NBIT_AUTOSIZE_LIMIT** 的 **IQ UNIQUE**。自动大小调整功能会自动将所有中低基数列的大小调整为 NBit。如果想要将列装载为 Flat FP，或者想要在不同值的数量超出 **FP_NBIT_AUTOSIZE_LIMIT** 时将列装载为 NBit，请使用 **IQ UNIQUE**。

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 14. 考虑顺序

唯一值数目	建议的索引类型
低于 1,000	LF (如果表的行数 <100,000 则为 HG)
1000 或更大	HG 和/或 HNG

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

特定索引类型优化特定类型的查询。

预先确定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，该列是将用于 **GROUP BY** 子句、用作 **COUNT DISTINCT** 的参数，还是用于 **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： SAP Sybase IQ 仍可以解析索引类型不正确的列涉及的查询，但不能以同等有效方式执行。

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，最为重要的因素是唯一值数目。
- 使用集函数 **COUNT**、**COUNT DISTINCT**、**SUM**、**MIN**、**MAX** 和 **AVG** 时，为使用缺省索引以外的任何索引，必须使用单个表解析整个查询。
- **BIT** 数据只能用于缺省索引；大于 255 字节的 **VARBINARY** 数据只能用于缺省索引类型、**TEXT** 索引类型和 **CMP** 索引类型；大于 255 字节的 **CHAR** 和 **VARCHAR** 数据只能用于缺省索引类型、**CMP** 索引类型、**TEXT** 索引类型和 **WD** 索引类型；**LONG VARCHAR** 数据只能用于缺省索引类型、**TEXT** 索引类型和 **WD** 索引类型；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。

按查询类型建议的索引

使用为您的查询推荐的索引类型。

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。这些建议不应成为选择索引类型时所依据的唯一条件；还需要考虑唯一值的数量和磁盘空间。

查询使用类型	建议的索引类型
用于 SELECT 投影列表	缺省值
用于计算表达式，如 SUM(A+B)	缺省值
用作 AVG/SUM 参数	LF , HG 、缺省值
用作 MIN/MAX 参数	LF , HG
用作 COUNT 参数	缺省值
用作 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
用在字段不允许重复项的情况下	HG
即席连接条件中使用的列	缺省值、 HG , LF 、
用作 WHERE 子句中的 LIKE 参数	缺省值
用作 IN 参数	HG , LF
用于等式或不等式 (= 、 !=)	HG , LF ；还有 CMP
用于 WHERE 子句中的范围谓词 (> 、 < 、 >= 、 <= 、 BETWEEN)	LF 或 HG ；还有 CMP 、 DATE 、 TIME 、 DTTM
用于 DATEPART 相等谓词、范围谓词和 IN 列表谓词	DATE , TIME , DTTM
用于 CONTAINS 谓词	WD , TEXT

按列值建议的索引

诸如列中唯一值的数量等条件有助于确定相应的数据索引类型。

标识条件	要选择的索引
标识在所有列中自动创建的索引。	缺省索引
标识在所有含 UNIQUE 或 PRIMARY KEY 约束的列中自动创建的索引。	含强制 UNIQUE 的 HG
标识在连接谓词中使用的所有列，并根据唯一值的数量选择索引类型。	HG 或者 LF
标识唯一值数量较低且尚未使用多个索引的列。	LF
标识唯一值数量较高且在 SELECT DISTINCT 或 DISTINCT COUNT 的选择列表中属于 GROUP BY 子句一部分的列。	HG
标识可在即席连接查询的 WHERE 子句中使用且尚不具有 HG 或 LF 索引的列。	HG 或者 LF
标识唯一值数量较高且不会与 GROUP BY 、 SELECT DISTINCT 或 DISTINCT COUNT 一起使用的列。	HNG
标识可能需要经常比较且具有相同数据类型、精度和小数位数的列对。	CMP
标识包含关键字或 URL 列表的列。	WD
标识唯一值数量较高且不会与 GROUP BY 、 SELECT DISTINCT 或 DISTINCT COUNT 一起使用的 DATE 、 TIME 、 DATETIME 或 TIMESTAMP 列。	DATE 、 TIME 或 DTTM
查看所有剩余的列，并根据唯一值数量、查询类型和磁盘空间来确定其它索引。此外，对于所有列，确保您选择的索引类型适合该列的数据类型。	

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

索引类型	估计空间与原始数据	注释
High_Group	从小于到 2 倍大小	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 2 倍大小	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。

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
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

数据类型	支持的索引	不支持的索引
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
long varchar	WD、TEXT	CMP、HG、HNG、LF、DATE、TIME、DTTM
binary <= 255 字节	CMP、HG、LF、TEXT	HNG、WD、DATE、TIME、DTTM
binary > 255 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

索引类型组合

在多种查询类型中使用某一列时，可能需要分配多个列索引类型。

表 17. 索引组合

	要添加的索引					
现有索引	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 = 建议使用的组合。						
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。						

创建索引

当您在 IQ 存储库中创建数据库时，SAP Sybase IQ 会自动在每个表列上创建缺省索引。为获得最佳查询性能，需要创建附加索引。

前提条件

如果在分区表上创建索引，则所有 dbspace 都必须处于联机状态。

过程

1. 为数据选择正确的索引类型。确保所选的索引类型适合该列的数据类型。
如果没有指定索引类型，则 SAP Sybase IQ 会创建 **HG** 索引。为此，若干前端工具将自动创建 **HG** 索引。
2. 使用以下任何方法创建索引：
 - 输入 **CREATE INDEX** 语句。
例如，在 SalesOrderItems 表的 ShipDate 列上创建名为 ShipIx 的 High_Non_Group (**HNG**) 索引。

```
CREATE HNG INDEX ShipIx
ON dbo.SalesOrderItems (ShipDate)
```
 - 使用 SAP Control Center 创建索引向导。
 - 使用 **CREATE TABLE** 的 PRIMARY KEY 或 UNIQUE 列约束，其将自动创建 **HG** 索引。

另请参见

- CREATE INDEX 语句 (第 397 页)
- CREATE TABLE 语句 (第 410 页)

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；第二个语句将收到错误，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。

索引装载的状态消息

如果启用进度消息，则在插入每 100,000 条记录后会显示状态消息。

要启用消息，请将 **CREATE INDEX** 的 **NOTIFY** 选项设置为非零值。

要更改记录数目，请使用 **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**。

在插入数据后添加列索引

创建其它列索引时，**CREATE INDEX** 命令会创建新索引作为单个表的一部分。

如果单个表中的现有列索引已含有数据，则 **CREATE INDEX** 语句还会从现有索引向新索引插入数据。这样可确保单个表中多个列的所有列索引之间的数据完整性。

如果您在插入数据后发现某一列需要其它索引，则可使用此功能。这样，不必重新开始便可以添加索引。

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** 语句一次性插入到所有这些索引，这种方法的速度通常相对较快。

执行 CREATE INDEX 语句组

使用关键字 **BEGIN PARALLEL IQ** 和 **END PARALLEL IQ** 对希望作为一组语句同时执行的任意数量的 **CREATE INDEX** 语句进行分隔。

这些关键字仅用于在 IQ 主存储库的基表中创建索引，而不用于在临时表或目录存储表中创建索引。

如果其中任何一个 **CREATE INDEX** 语句失败，则所有语句都将回退。

```
BEGIN PARALLEL IQ
  CREATE HG INDEX c1_HG on table1 (col1);
  CREATE HNG INDEX c12_HNG on table1 (col12);
  CREATE LF INDEX c1_LF on table1 (col1);
  CREATE HNG INDEX c2_HNG on table1 (col2);
END PARALLEL IQ
```

运行索引顾问

如果在数据库中设置了 **INDEX_ADVISOR** 选项，SAP 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附加索引建议。

1. 在 Interactive SQL 中，设置 **INDEX_ADVISOR** 选项：

```
SET OPTION index_advisor = 'ON'
```

2. 检查消息日志文件。缺省情况下，会将消息日志命名为 `dbname.iqmsg`，并在目录存储库所在的目录中创建文件。

查找侧重于以下几个方面的消息：

- 本地谓词列
- 单列连接键列
- 子查询相关列
- 分组列

3. 如果您决定遵循索引顾问消息的建议，请创建以下建议索引：

```
CREATE HG INDEX id_hg
ON SalesOrderItems
( ID ) IN Dsp5
```

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。要重命名索引，使用 **ALTER INDEX** 语句。

注意： 不能重命名用于强制执行键约束的索引。

查看索引

使用存储过程 **sp_iqindex** 查看关于索引的信息。

要列出 `Departments` 表中的索引，请发出命令：

- `sp_iqindex 'Departments'`

为了便于阅读，此处输出将以两部分显示：

table_name	table_owner	column_name	index_type
Departments	GROUPO	DepartmentHeadID	FP
Departments	GROUPO	DepartmentHeadID	HG
Departments	GROUPO	DepartmentID	FP

table_name	table_owner	column_name	index_type
Departments	GROUPO	DepartmentID	HG
Departments	GROUPO	DepartmentName	FP

- 要显示所有表的信息，请在 **sp_iqindex** 中省略表名：

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)

系统视图中的索引信息

索引信息位于统一视图 SYSINDEXES 和系统视图 SYSIQIDX 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”。

删除索引

删除不再需要的列索引。

使用以下方法之一：

- 输入 **DROP** 语句。请参见《《参考：语句和选项》中的“DROP 语句”。
- 删除 SAP Sybase IQ 中的索引。

在删除外键约束时保留索引

您可能需要删除外键约束，但保留基础 **HG** 索引。非唯一的 **HG** 索引可提高查询性能，但构建此索引成本较高。

输入 **ALTER TABLE DROP FOREIGN KEY** 语句。

请参见参考：语句和选项中的 **ALTER INDEX** 语句。

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，请在发出 **ALTER TABLE DROP FOREIGN KEY** 之后将其显式删除。

优化连接性能

可以创建索引来优化连接性能。

为了以最快速度处理连接，请对下列位置可能引用的所有列创建 **Low_Fast (LF)** 或 **High_Group (HG)** 索引：

- 连接查询的 **WHERE** 子句
- 集合函数之外的连接查询的 **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。

实施数据完整性

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完整。

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。

SAP Sybase IQ 支持存储过程和 JDBC，这样可允许您对数据进入数据库的方式进行具体控制。有关 JDBC 的信息，请参见 SAP Sybase IQ 《编程》指南。

哪些情况会导致数据无效

如果不进行适当检查，可能会导致数据无效。

例如：

- 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。
- 多人在部门表中输入关于新部门（**dept_id** 为 200）的相同信息。

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列或多列的完整性的规则为表约束。可以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。

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定义的一部分，并且可在所有应用程序中一致实施这些约束。

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。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。

声明适用的任何约束，无论 SAP Sybase IQ 是否实施这些约束。通过声明约束，可以确保您了解自身的数据需求，并确保设计的数据库符合组织的业务规则。

SAP Sybase IQ 根据指定的约束执行多种优化。此优化不依赖于是否实施约束。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。**FOREIGN KEY**、**PRIMARY KEY**、**UNIQUE** 和 **IQ UNIQUE** 列约束可以提高装载和查询的性能。

SAP Sybase IQ 在装载操作期间检查是否遵循特定约束。例如，SAP Sybase IQ 确保要装载的数据具有相应的数据类型和长度。

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

使用数据约束和指定数据库参照结构的约束来保持数据的完整性。

约束

可以对各列或各表中的数据使用下面几种类型的约束：

- **NOT NULL** 约束禁止列包含 **NULL** 条目。SAP Sybase IQ 实施此约束。
- 可以向列指派 **CHECK** 条件，以指定该列中每一行都应满足的特定条件，例如，薪水列中的条目必须在特定范围内。
- 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相互联系。这些关系由构建到数据库设计中的候选键和外键定义。

外键 由一列或列组合构成。每个外键将一个表（外表）中的信息与另一个表（引用表或主表）中的信息相关联。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**候选键**。引用列或列集必须为候选键，它被称作**引用键**。不能对同时作为外键的候选键指定外键约束。

以下参照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。所有外键值必须与相应主键中的值匹配，或包含 **NULL** 值（如果定义为允许 **NULL** 值）。

实施完整性约束的语句

通过提交 SQL 语句实施完整性约束。

- **CREATE DATABASE** 语句可在创建数据库时实施完整性约束。
- **ALTER DATABASE** 语句可从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

只要在数据库表中输入一行新数据，列缺省值就会自动为特定列或一组列指派一个特定值。

列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。

支持的缺省值

SAP Sybase IQ 支持列的缺省值。

- **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

某些列缺省值不受支持。

SAP 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

SAP 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，或者使用 **ALTER TABLE** 语句在创建表后添加列缺省值。

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. 在 `sales_order` 表中为名为 `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，其数据类型为 `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，请使用创建缺省值时使用的 **ALTER TABLE** 语句格式。

例如，要将 **order_date** 列中的缺省值从当前设置更改为 **CURRENT DATE**，请使用：

```
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 **NULL**。

下面的语句将删除 **order_date** 列中的缺省值：

```
ALTER TABLE sales_order  
MODIFY order_date DEFAULT NULL
```

支持的列缺省值

可在 SAP 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**

在定义和插入列缺省值时：

- 可以指定无法由 SAP Sybase IQ 进行计算的缺省值。如果表中具有不受支持的缺省值，那么在对此表执行 **INSERT**、**LOAD** 或 **ALTER ADD** 操作时会报告错误。
- 如果服务器尝试插入的缺省值与该列的数据类型不兼容，则 SAP Sybase IQ 将生成错误或警告。例如，如果对整数列定义缺省表达式 "N/A"，则任何未指定列值的插入或装载都将生成错误或警告，具体取决于 `CONVERSION_ERROR` 数据库选项的设置。
- 如果缺省值对于 `CHARACTER` 类型列来说过长，则 SAP Sybase IQ 要么截断该字符串，要么生成异常，具体取决于 `STRING_RTRUNCATION` 数据库选项的设置。
- 如果 `VARCHAR` 或 `LONG VARCHAR` 列中的缺省值是长度为零的字符串，则 SAP Sybase IQ 将插入 `NULL` 或长度为零的字符串，具体取决于 `NON_ANSI_NULL_VARCHAR` 数据库选项的设置。
- 如果 `VARCHAR`、`CHAR` 或 `LONG VARCHAR` 列中的缺省值包含部分多字节字符，则 SAP Sybase IQ 可能会在插入值之前修整该字符，具体取决于 `TRIM_PARTIAL_MBC` 数据库选项的设置。
- 如果缺省值违反表或列的检查约束，则每次服务器尝试插入缺省值时，SAP Sybase IQ 都会生成错误消息。
- 在执行 **LOAD TABLE** 操作期间因插入缺省值而发生的约束违反永远不会被忽略，无论是否存在用户指定的 **IGNORE CONSTRAINT** 和 **MESSAGE LOG/ROW LOG** 选项。
- SAP Sybase IQ 不支持列缺省值 **UTC TIMESTAMP** 和 **CURRENT UTC TIMESTAMP**。
- 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。
- 如果将已定义缺省值的列添加到表中，则新列中的所有行都将使用此缺省值进行填充。
- 更改表中现有列的缺省值不会更改表中的任何现有值。
- **LOAD TABLE DEFAULTS** 选项必须为 **On** 才能使用 **LOAD TABLE** 语句 **DEFAULT** 选项中指定的缺省值。如果 **DEFAULTS** 选项为 **OFF**，则不使用指定的装载缺省值，而是在列中插入 `NULL` 值。
- **LOAD TABLE DEFAULT** 规范中必须至少包含一个要从 **LOAD TABLE** 命令所指定的文件中进行装载的列。
- **LOAD TABLE DEFAULT** *default-value* 必须与数据库具有相同的字符集，而且必须符合受支持的列缺省值和缺省值限制。**LOAD TABLE DEFAULT** 选项不支持 **AUTOINCREMENT**、**IDENTITY** 或 **GLOBAL AUTOINCREMENT** 作为装载缺省值。
- 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。

日期、时间和时间戳缺省值

对于数据类型为 **DATE**、**TIME** 或 **TIMESTAMP** 的列，可使用 **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

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。

SAP Sybase IQ 不支持缺省值 **UTC TIMESTAMP** 或 **CURRENT UTC TIMESTAMP**，SAP Sybase IQ 也不支持数据库选项 **DEFAULT_TIMESTAMP_INCREMENT**。每次尝试插入或更新类型为 **UTC TIMESTAMP** 或 **CURRENT UTC TIMESTAMP** 的列缺省值时，SAP Sybase IQ 都会生成错误。

USER 缺省值

将 **DEFAULT USER** 指派给列可以标识在数据库中输入信息的人员。例如，若销售人员从事按业绩提成的工作，可能需要此信息。

为表的主键生成用户 **ID** 缺省值对偶尔连接的用户来说非常有用，并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，在未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。

此功能会为每个新行指派一个值，该值比列中的当前最大值大一。可以使用 **IDENTITY/AUTOINCREMENT** 列记录采购订单编号、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有效，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。

SAP Sybase IQ 不支持特殊值 **GLOBAL AUTOINCREMENT**。

在创建表时，初始的 **IDENTITY/AUTOINCREMENT** 值为 0，并且仅适用于正整数。

具有 **AUTOINCREMENT** 缺省值的列在 Transact-SQL 应用程序中称为 **IDENTITY** 列。SAP Sybase IQ 支持这两个关键字。

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。

使用这种方式生成值会使在一台计算机上生成的值与在另一台计算机上生成的值不匹配。因此，可将 **UUID** 用作复制和同步环境中的键。

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 **NULL** 值的列，指定 **NULL** 缺省值与不指定任何缺省值是相同的。

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 **NULL** 值的详细信息，请参见参考：构件块、表和过程中的 **NULL** 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，则必须用括号将该表达式括起来。

例如，常量表达式允许列缺省值包含自今天起第十五天的日期等条目：

```
... DEFAULT ( DATEADD( DAY, 15, GETDATE() ) )
```

表约束和列约束

约束有助于确保表中所输入数据的正确性，并为 **SAP 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造成负面影响。对于较大的表，**ALTER TABLE** 或 **CREATE TABLE** 操作可能十分耗时。**CREATE TABLE** 处理将延迟其它进程的执行，直至语句完成。虽然可以在其它连接处于活动状态时执行 **ALTER TABLE** 语句，但如果有任何其它连接正在使用要更改的表，便无法执行这些语句。在 **ALTER TABLE** 期间，不允许在处理该语句的过程中进行任何其它引用所更改的表的请求。

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。

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** 时，列中不能包含任何 **NULL** 值。

示例 1

下面的示例将 `ss_number` 列添加到 `employee` 表中，并确保其中的每个值在整个表中都是唯一的：

```
ALTER TABLE employee
ADD ss_number char(11) UNIQUE
```

示例 2

在本示例中，三列组成一个唯一条目。

```
ALTER TABLE product
ADD UNIQUE (name, size, color)
```

对列的 IQ UNIQUE 约束

IQ UNIQUE 定义列的预期基数并确定是将该列装载为 Flat FP 还是 NBit。

IQ UNIQUE 定义列的预期基数并确定是将该列装载为 Flat FP 还是 NBit。IQ UNIQUE(*n*) 值显式设置为 0 会将该列装载为 Flat FP。不带 IQ UNIQUE 约束的列将隐式装载为 NBit，一直到 FP_NBIT_AUTOSIZE_LIMIT 和 FP_NBIT_LOOKUP_MB 选项定义的限制：

- FP_NBIT_AUTOSIZE_LIMIT 限制装载为 NBit 的不同值的数量
- FP_NBIT_LOOKUP_MB 为 NBit 字典总大小设置阈值
- FP_NBIT_ROLLOVER_MAX_MB 为从 NBit 至 Flat FP 的隐式 NBit 切换设置字典大小
- FP_NBIT_ENFORCE_LIMITS 强制执行 NBit 字典大小限制。此选项缺省设置为 OFF

不必使用 *n* 值小于 FP_NBIT_AUTOSIZE_LIMIT 的 IQ UNIQUE。自动大小调整功能会自动将所有中低基数列的大小调整为 NBit。如果想要将列装载为 Flat FP，或者想要在不同值的数量超出 FP_NBIT_AUTOSIZE_LIMIT 时将列装载为 NBit，请使用 IQ UNIQUE。

列的 CHECK 条件

使用 CHECK 条件为列中的值指定条件。

此条件可以指定为确保数据合理性而必须满足的规则，也可以指定用以反映组织政策和程序的规则。

如果某列的有效值限制在一定范围内，则可以单独在该列上使用 CHECK 条件。

示例 1

输入的值应与一组有限数量的值中的一个相匹配。要指定 `city` 列只包含几个允许城市（例如，组织在当地设有办事处的城市）中的一个，可以使用与以下约束相类似的约束：

```
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。

示例2

日期或数字在特定的范围内。您可能希望要求 **employee** 表中的 **start_date** 列必须处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：无论当前选项设置如何，始终识别 **YYY/MM/DD** 格式。

用户定义数据类型的 **CHECK** 条件

可将 **CHECK** 条件附加到用户定义的数据类型。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 **@** 而不是 **@col** 的变量名。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的 **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。

列的 **CHECK** 条件

可将 **CHECK** 条件作为约束应用到表中而不是单个列中。

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。建议尽可能对单个列使用 **CHECK** 条件。

添加 **Check** 条件

您可以向表或单个列中添加新的 **CHECK** 条件。

例如，在图书馆数据库中，特定条目的 **date_returned** 列必须等于或晚于 **date_borrowed** 条目：

```
ALTER TABLE loan
ADD CHECK (date_returned >= date_borrowed)
```

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 NULL。

以下语句将删除 **customer** 表中 **phone** 列的 CHECK 条件：

```
ALTER TABLE customer MODIFY phone
CHECK NULL
```

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。

以下语句将在 **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

有两种方法可用于修改表中定义的 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，则任何查询表中数据的尝试都将产生错误消息 `column not found`。

删除表中的 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

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。如果表的结构以后被修改，还可以使用 **ALTER TABLE** 语句的子句 **DELETE PRIMARY KEY** 或 **ADD PRIMARY KEY** 来重新定义主键。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和更改数据库表。如果使用此类系统，可能不必显式输入 **CREATE TABLE** 或 **ALTER TABLE** 命令。应用程序将通过您提供的信息来生成语句。

在插入或更新表行后，数据库服务器将确保表的主键仍有效：表中的每一行都由主键唯一标识。

示例 1

演示数据库中的 **Employees** 表使用员工 ID 作为主键。在表中添加新员工后，IQ 将检查新员工 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 SAP Sybase IQ 简介中的表名和所有者。

示例 2

演示数据库中的 **SalesOrderItems** 表使用两个列来定义主键。

该表保存有关可订购项目的信息。一列包含用于指定订单的 **id**，但每个订单可能涉及几个项目，因此该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。

SAP Sybase IQ 对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** 进行参照完整性检查。如果发生参照完整性违规，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。SAP Sybase IQ 还可以根据用户的指定选择性地拒绝违反数据完整性的行。

参照完整性

外键中的条目必须与所引用表中行的主键值相对应。

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**。

参照完整性丢失

SAP 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。所有引用该主键的外键均违反参照完整性。
- 在外表中添加一个新行，并为没有相应候选键值的外键输入一个值。数据库违反参照完整性。

SAP Sybase IQ 对这两类完整性丢失提供保护。

当更新或删除引用的候选键时，SAP Sybase IQ 不允许 **UPDATE** 或 **DELETE**。

外键

使用 **CREATE TABLE** 语句或 **ALTER TABLE** 语句创建外键，创建方法与主键相同。

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。

演示数据库中包含一个 **employee** 表和一个 **department** 表。**employee** 表的主键是员工 ID，**department** 表的主键是部门 ID。

例如，假设采用以下模式：

```
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 **employee** 表中，部门 ID 是 **department** 表的外键；**employee** 表中的每个部门 ID 严格对应于 **department** 表中的部门 ID。

外键关系是多对一关系，而且在本例中该关系是强制的。**employee** 表中的几个条目都具有相同的部门 ID 条目，但部门 ID 是 **department** 表的主键，因此在 **department** 表

中是唯一的。如果允许外键引用 **department** 表中包含重复条目的列，则无法确定 **department** 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

SAP Sybase IQ 通过 **RESTRICT** 操作（ANSI 缺省设置）在语句级支持参照完整性。

这意味着，SAP Sybase IQ 将拒绝主键或具有唯一约束的列上的更新和删除请求，该唯一约束可删除对应外键所依据的任何值。（必须慎重考虑请求删除和更新时所采用的顺序。）SAP Sybase IQ 将发出错误消息并回退违反参照完整性的装载操作，但允许您指定可对特定行进行忽略。

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。

假定下列模式：

```
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 }
```

1. 标识候选键-外键关系。

在该模式中，存在两种这样的关系：

- 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. 删除现有未实施的外键约束，例如：

```
ALTER TABLE DEPT DROP FOREIGN KEY MGR_EMPNO;
ALTER TABLE EMPLOYEE DROP FOREIGN KEY DEPT;
```

在该模式中，必须删除引用 **DEPT(DeptNo)** 的未实施外键约束 **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，则缺省角色名称与主表相同，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活动

在新表中强制实施参照完整性。

1. 创建主表，例如：

```
CREATE TABLE DEPT(DeptNo int primary key,
DeptName varchar(20),
Mgr int );
```

2. 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);
```

3. 指定外键约束。例如：

```
ALTER TABLE DEPT ADD FOREIGN KEY EMP_DEPT(DeptNo)
REFERENCES DEPT(DeptNo);
```

在此语句中，同一外键的用户指定角色名称为 EMP_DEPT。

删除外键

从表中删除外键。

1. 创建主表和外表。例如：

```
CREATE TABLE DEPT(DeptNo int primary key,
DeptName varchar(20),
Mgr int );
```

```
CREATE TABLE EMPLOYEE(EmpNo int primary key,
DeptNo int references DEPT(DeptNo)
on delete restrict,
LastName varchar(20),
FirstName varchar(20),
Salary int);
```

2. 如果未指定任何角色名称，则指定外键的缺省角色名称为 DEPT：

```
ALTER TABLE EMPLOYEE DROP FOREIGN KEY DEPT;
```

3. 如果有多个外键且角色名称未知，可使用 **sp_iqconstraint** 过程显示角色名称。请参见参考：构件块、表和过程。

4. 如果已分配角色名称，如 EMP_DEPT，则删除键时必须指定该角色名称：

```
ALTER TABLE EMPLOYEE DROP FOREIGN KEY EMP_DEPT;
```

上述语句不删除隐式创建的 EMPLOYEE(DeptNo) 的非唯一 HG 索引。要删除该索引，请使用 **sp_iqindex** 找到 HighGroup 索引名，并使用 **DROP INDEX** 语句，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
```

要删除表，必须删除所有关联的外键约束。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;
```

假设数据库中还包含一个列出了办事处地点的办事处表。employee 表中可能有办事处表的外键，显示雇员工作办事处的地点。该数据库的设计人员可以选择在聘用雇员时不指派办事处地点。此时，外键应允许 NULL 值，因为办事处地点未知或该雇员不在外面的办事处工作。

并发操作

SAP Sybase IQ 的参照完整性功能限制在对外表执行装载或插入时在主表上进行并发更新或删除。

表 18. 那些将返回错误的并发操作

第一个请求	重叠事务请求
一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 。

如果一个事务请求更改表、添加外键或删除外键，但其它事务正在使用外表和/或主表的旧版本，SAP Sybase IQ 也会生成错误。

对于已强制实施和未实施的外键与主键，SAP Sybase IQ 允许：

- 同时对一个或多个外表和共享主表进行装载或插入。
- 同时对外表进行装载或插入，并对其它一个或多个外表进行删除、更新或截断。
- 同时对两个或多个外表进行删除、更新或截断，即使在共享相同的主表时也是如此。
- 同时对外表进行删除、更新或截断，并对共享的主表进行删除、更新或截断。
- 如果没有任何事务在使用旧版本的外表/主表，则对外键执行 **ALTER TABLE ADD** 或 **DROP**，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

SAP Sybase IQ 的表级版本控制在允许对外表执行并发装载、插入或更新操作及对主表执行 **LOAD/INSERT** 操作的同时，可确保一致的参照完整性检查。

当用于请求 **DELETE** 或 **UPDATE** 的事务启动时，SAP Sybase IQ 还会验证外表中是否存在已删除的旧值。这样便可在对外表执行并发删除操作及对 **PRIMARY** 表执行删除或更新操作时提供一致的参照完整性检查。

假定两个外表 (ftab1 和 ftab2) 与一个主表 (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) 可以是主键，也可以是唯一约束。

下表显示在这两种表 (外表和主表) 中应允许哪些操作以及哪些操作将返回错误。表中的信息仅适用于强制实施的外键和候选键。

表 19. 对外表和主表执行并发 DML

	LOAD 或 INSERT ftab1	DELETE/ 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 (fk1、 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 TABLE ftab2	允许	允许	允许	允许	允许	允许
DELETE ptab TRUNCATE TABLE 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

	LOAD 或 INSERT ftab1	DELETE/ 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 (fk1、 fk2)
QUERY (正在 使用含或不 含 (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	N/A	N/A	N/A	N/A	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)

如果一个事务装载外键列，而另一事务更新关联的候选键列，则会发生并发冲突。如果一个事务装载外键列，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，无法对用作连接条件或分组列的外键所在的表进行更改或删除。但是，可以在运行查询时装载、插入、删除和更新这些表。

禁用参照完整性检查

可以使用 **SAP Sybase IQ** 选项 `DISABLE_RI_CHECK` 跳过参照完整性检查。

由于跳过参照完整性检查会使采用此功能的目的落空，因此请慎重使用此选项。

系统表中的完整性规则

有关数据库中完整性检查和规则的所有信息都存储在系统表和视图中。

系统表	描述
SYS.SYSTABLE	CHECK 约束保存在 SYS.SYSTABLE 的 <code>view_def</code> 列中。对于视图， <code>view_def</code> 保存创建视图的 CREATE VIEW 命令。可通过查看 <code>table_type</code> 列 (BASE 或 VIEW) 来检查某个特定表是基表还是视图。
SYS.SYSFOREIGN-KEYS	此视图以更便于阅读的格式显示 SYS.SYSFOREIGNKEY 和 SYS.SYSFKCOL 两个表中的外键信息。

系统表	描述
SYS.SYSCOLUMNS	此视图以更便于阅读的格式显示 SYS.SYSCOLUMN 表中的信息。其中包括列的缺省设置和主键信息。

可以使用 **Interactive SQL** 浏览这些表和视图。

校验目录存储库索引

可在目录存储库中对 **SQL Anywhere** 表的索引进行校验，以确保索引中引用的每一行在表中都实际有效。

要校验索引，请打开命令提示符，并运行 **dbvalid** 实用程序。

例如，下面的语句（应输入到一行中）将校验名为 **EmployeeIndex** 的索引。**-i** 开关指定每个给定的对象名是一个索引。

```
dbvalid -c "uid=dba;pwd=sql;eng=myserver"  
-i EmployeeIndex
```

下一步

请参见实用程序指南中的 **dbvalid** 数据库管理实用程序。

管理事务和版本控制

事务处理确保将逻辑上相关的命令组作为一个单元来执行。SAP Sybase IQ 使用事务处理来允许多个用户在数据库更新时从数据库中读取数据。事务是保持数据精确性以及当在系统出现故障时执行数据恢复的基础。

事务处理能够使用户不受其他用户事务的影响，这是其至关重要的一个方面。SAP 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。快照版本控制为每个数据库用户提供了一份数据库快照。在提交事务之前，其他数据库用户看不到您对数据库执行的任何更改。

另请参见

- 表级快照版本控制（第 189 页）
- 行级快照版本控制（第 193 页）

事务

每个事务都是一系列在逻辑上相关的命令，用于完成一项任务，并将数据库从一种一致状态转换为另一种一致状态。

事务是原子的；SAP 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，提交更改以使其成为永久性更改。如果由于某种原因使事务中的任何命令未得到正确处理，则可以撤消或回退部分或全部中间更改。用户应用程序控制着提交或回退更改所依据的条件。

事务由小块组成。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。

大多数情况下，SAP Sybase 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。您也可以通过发出显式命令来开始或结束事务。

事务自动通过以下事件之后的第一个语句启动：

- 与数据库连接
- 上一事务结束

SAP Sybase IQ 还支持 SAP Sybase IQ Transact-SQL 命令（如 **BEGIN TRANSACTION**），以便与 Adaptive Server 兼容。可以使用 **BEGIN TRANSACTION** 命令显式启动事务。

使用以下事件之一完成事务：

- 执行 **COMMIT** 语句使对数据库进行的更改成为永久更改。
- 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* 副作用的语句会对数据库进行更改。

数据库定义命令（如 **ALTER**、**CREATE** 和 **DROP**）都具有自动提交副作用。

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，或引发其它操作。

另请参见

- CHECKPOINT 语句（第 383 页）
- SAVEPOINT 语句（第 437 页）

查看事务活动

使用 **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。

运行 **sp_iqtransaction**。

该过程针对 IQ 事务管理器中的每个事务控制块返回一行信息。

sp_iqtransaction 过程

显示事务和版本的相关信息。

语法

sp_iqtransaction

适用于

Simplex 和 Multiplex。

特权

您必须具有系统过程的 EXECUTE 特权，以及 MONITOR 系统特权。

注释

sp_iqtransaction 将针对 SAP Sybase IQ 事务管理器中的每个事务控制块返回一行信息。列 Name、Userid 和 ConnHandle 分别是连接属性 **Name**、**Userid** 和 **Number**。行按 TxnID 排序。

sp_iqtransaction 输出只包含具有进行中事务的那些连接。要包括所有连接，请使用 **sp_iqconnection**。

注意： 尽管您可以使用 **sp_iqtransaction** 来标识阻止其他用户向表中写入内容的用户，但最好选择使用 **sp_iqlocks** 来实现此目的。

列名	描述
名称	应用程序的名称。
Userid	连接的用户 ID。

列名	描述
TxnID	此事务控制块的事务 ID。该事务 ID 在 begin transaction 时分配。它将出现在 .iqmsg 文件的 BeginTxn、CmtTxn 和 PostCmtTxn 消息中，并与打开数据库时所记录的 Txn ID Seq 相同。
CmtID	提交事务时事务管理器分配的 ID。对于活动事务，CmtID 为零。
VersionID	<p>对于 Simplex 和 Multiplex 节点，值 0 表示事务未实现版本化，且尚未分配 VersionID。</p> <p>对于 Multiplex 协调器，在事务建立表锁后分配 VersionID。Multiplex 辅助服务器自协调器接收 VersionID。VersionID 由 SAP Sybase IQ 内存中目录和 IQ 事务管理器在内部使用，以对 Multiplex 数据库中的所有节点唯一标识数据库版本。</p>
状态	<p>事务控制块的状态。此变量反映 SAP Sybase IQ 内部实现的详细信息，并可能在未来发生更改。当前，事务状态为 NONE、ACTIVE、ROLLING_BACK、ROLLED_BACK、COMMITTING、COMMITTED 和 APPLIED。</p> <p>NONE、ROLLING_BACK、ROLLED_BACK、COMMITTING 和 APPLIED 是瞬时状态，寿命很短。</p> <p>ACTIVE 指示事务处于活动状态。</p> <p>COMMITTED 指示事务已完成并正在等待 APPLIED，此时，将对对于任何事务都不可见的版本执行垃圾回收。</p> <p>一旦事务状态变为 ROLLED_BACK、COMMITTED 或 APPLIED，除了打开游标所持有的锁，该事务将不再拥有任何其它锁。</p>
ConnHandle	连接的 ID 号。
IQConnID	十位连接 ID，是 .iqmsg 文件中所有消息的一部分。它是一个单调递增整数，在整个服务器会话内唯一。
MainTableKBCr	此事务创建的 IQ 存储空间的大小（以 KB 为单位）。
MainTableKBDr	此事务删除的 IQ 存储空间的大小（以 KB 为单位），但由于该空间在其它数据库版本或此事务的其它保存点仍可见，因此该空间仍位于磁盘存储中。
TempTableKBCr	此事务创建的用于存储 IQ 临时表数据的 IQ 临时存储空间的大小（以 KB 为单位）。
TempTableKBDr	此事务删除的 IQ 临时表空间的大小（以 KB 为单位），但由于该空间对 IQ 游标仍可见，或者归此事务的其它保存点所有，因此该空间仍位于磁盘的 IQ 临时存储库中。

列名	描述
TempWorkspaceKB	<p>对于 ACTIVE 事务，该事务此时使用的工作空间的快照，如排序、散列和临时位图。该数值因您运行 sp_iqtransaction 的时间而异。例如，查询引擎可能会在临时高速缓存中创建 60MB，但很快释放其中的大部分空间，即使查询处理仍在继续也是如此。如果在查询完成后运行 sp_iqtransaction，该列显示要小得多的数值。当事务不再处于活动状态时，该列为零。</p> <p>对于 ACTIVE 事务，该列与 sp_iqconnection 的 TempWorkspaceKB 列相同。</p>
TxnCreateTime	事务的开始时间。一旦建立活动连接或者前一个事务执行提交或回退操作，所有 SAP Sybase IQ 事务将立即隐式开始。
CursorCount	引用该事务控制块的打开的 SAP Sybase IQ 游标数。如果事务处于 ACTIVE 状态，则它指示事务中创建的打开的游标数。如果事务处于 COMMITTED 状态，则它指示引用该事务控制块拥有的数据库版本的 HOLD 游标数。
SpCount	事务控制块中存在的保存点结构数。保存点可以隐式创建和释放。因此，此数值不指示事务中用户创建的保存点的数量。
SpNumber	事务的活动保存点数。这是实现详细信息，因此可能不能反映用户创建的保存点。
MPXServerName	指示活动事务是否来自节点间通信 (INC) 连接。如果来自 INC 连接，则值是发起事务的 Multiplex 服务器的名称。如果不是来自 INC 连接，则为 NULL 。如果事务不是活动的，则始终为 NULL 。
GlobalTxnID	与当前事务关联的全局事务 ID，如果没有则为 0（零）。
VersioningType	事务的快照版本控制类型；表级（缺省值）或行级。行级快照版本控制 (RLV) 仅适用于启用 RLV 的表。一旦启动某个事务，该值便不能进行更改。
Blocking	指示连接阻塞是已启用 (True) 还是已禁用 (False)。可使用 BLOCKING 数据库选项设置连接阻塞。如果为 true ，事务阻塞，即事务先等待解除冲突锁，然后尝试重试锁定请求。
BlockingTimeout	指示事务等待清除锁定冲突的时间（毫秒）。可使用 BLOCKING_TIMEOUT 数据库选项设置超时阈值。值 0（缺省值）表示事务无限期等待。

示例

示例 **sp_iqtransaction** 输出：

Name	Userid	TxnID	CmtID	VersionID	State	ConnHandle	IQConnID
=====	=====	=====	=====	=====	=====	=====	=====

red2	DBA	10058	10700	10058	Active	419740283	14
MainTableKBCr							
MainTableKBDr		TempTableKBCr		TempTableKBDr			
0		0		65824		0	
TempWorkSpaceKB		TxnCreateTime		CursorCount		SpCount	
SpNumber							
0		2013-03-26 13:17:27.612		1		3	
MPXServerName		GlobalTxnID		VersioningType		Blocking	
BlockingTimeout							
(NULL)		0		Row-level		True	
0							

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。

隔离级别控制着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。表级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。

表级快照版本控制通过遵循以下规则在并发事务之间维护这种较高的隔离级别：

- 事务管理在每个事务开始时维护所提交的数据的快照。
- 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。

表级快照版本控制所提供的隔离级别可防止出现多种类型的不一致，其中包括（最常遇到）：

- **脏读** - 事务 A 修改了一个对象，但未提交或回退所做更改。事务 B 读取所修改的对象。然后，事务 A 在执行 **COMMIT** 之前进一步更改该对象。在这种情况下，事务 B 看到的对象处于从未被提交的状态。
- **非可重复的读取** - 事务 A 读取一个对象。之后，事务 B 修改或删除了该对象，并执行了 **COMMIT**。如果事务 A 再次尝试读取同一对象，该对象将已被更改或删除。
- **幻像数据元素** - 事务 A 读取了一组满足某个条件的数据。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中，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。

表级快照版本控制可确保在任何给定时间都只有一名用户可以修改表，在更改完成前保持更改对其他用户不可见，并随时为使用中的数据对象维护带时间戳的快照。

虽然通过 **IQ** 可以将隔离级别设置为 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。

事务阻塞

如果将 **BLOCKING** 选项设置为 on，则如果事务尝试获取写入锁，而该写入锁与另一事务当前持有的写入锁冲突，则该事务需一直等待，直到冲突锁被释放或达到 **BLOCKING_TIMEOUT** 阈值。缺省情况下，**BLOCKING** 设置为 off。

连接阻塞将请求事务置于休眠状态（阻塞表级或行级锁），直到持有锁的连接释放相应锁（取消阻塞表级或行级锁）。通过设置 **BLOCKING_TIMEOUT** 值（以毫秒为单位）来控制阻塞时长。如果 **BLOCKING_TIMEOUT** 为 0（缺省值），连接中的所有阻塞事务需要无限期等待，直至该连接获得请求的锁。如果设置了 **BLOCKING_TIMEOUT**，但在指定时间内未释放锁，正在等待的事务会收到一条错误消息。

阻塞利用了延迟事务版本控制，事务管理器将在建立表级或行级锁后创建事务快照版本。

以下面的表级锁阻塞为例：

- 用户 **A** 连接。
- 用户 **A** 执行 `INSERT INTO iq_table1 VALUES(3, 300);`。
- 用户 **B** 连接。
- 用户 **B** 执行 `INSERT INTO iq_table1 VALUES(4, 400);`。

禁用连接阻塞时，用户 **B** 的事务将回退，他会收到一条错误，指出另一用户对 `iq_table1` 具有锁。

启用连接阻塞（并将 **BLOCKING_TIMEOUT** 设置为 0）时，用户 **B** 的 **INSERT** 语句置于休眠状态，直到用户 **A** 的 **INSERT** 语句提交，从而释放表写入锁。

启用连接阻塞且 **BLOCKING_TIMEOUT** 设置为 200 毫秒时，如果用户 **A** 的事务在 200 毫秒阻塞超时阈值内未提交，则用户 **B** 的事务回退，同时收到一条错误，指出另一用户对 `iq_table1` 具有锁。

有关行级锁的示例，请参见《管理：内存行级版本控制》>“管理 **RLV** 存储库中的阻塞”。

启用连接阻塞

如果某一事务尝试获取的锁与其它事务中的现有锁相冲突，则启用连接阻塞可以强制该事务进行等待：一直等到所有冲突锁均已被释放，或等到达到 **BLOCKING_TIMEOUT** 阈值。

前提条件

- SAP Sybase IQ 服务器具有一个 Simplex 数据库。

过程

将 **BLOCKING** 数据库选项设置为 ON。

```
set temporary option blocking = 'On';
```

注意： 可以在连接级别或 **PUBLIC** 级别上设置该阻塞选项。

禁用连接阻塞

如果某一事务尝试获取的锁与其它事务中的现有锁相冲突，则禁用连接阻塞可以强制该事务回退事务并显示错误：

前提条件

- SAP Sybase IQ 服务器具有一个 Simplex 数据库。

过程

将 **BLOCKING** 数据库选项设置为 OFF。

```
set temporary option blocking = 'Off';
```

注意： 可以在连接级别或 **PUBLIC** 级别上设置该阻塞选项。

BLOCKING 选项

控制响应锁定冲突的行为。**Multiplex** 辅助级节点不支持 **BLOCKING**。

允许值

ON、OFF

默认值

OFF

范围

可在数据库 (**PUBLIC**) 或用户级别设置选项。在数据库级别进行设置时，值将变为任何新用户的缺省值，但不会对现有用户产生任何影响。在用户级别进行设置时，仅替换该用户的 **PUBLIC** 值。为自身设置选项无需任何系统特权。在数据库或用户级别为任何其他用户设置选项都需要系统特权。

必须具有 **SET ANY PUBLIC OPTION** 系统特权才能设置此选项。可针对个别连接或 **PUBLIC** 角色进行临时设置。设置立即生效。

注释

当 **BLOCKING** 设置为 **OFF** 时，如果某个事务尝试执行写入操作但被另一事务的读取锁阻止，则会收到一条错误。

将 **BLOCKING** 选项设置为 **ON** 后，如果事务试图获取锁定，而该锁定与另一事务包含的现有锁定冲突，则该事务要等待所有冲突的锁定都解除或达到 **blocking_timeout**。如果该锁定在 **blocking_timeout** 毫秒的时间内没有被释放，则会向正在等待的事务返回错误。

设置阻塞超时阈值

可使用该阈值设置事务为获得锁而等待的时间长度（以毫秒为单位）。如果尝试获得锁的事务与其它事务中的现有锁相冲突，则该事务会一直等待，直至达到 **BLOCKING_TIMEOUT** 选项阈值。如果冲突仍然存在，则该事务会回退并显示错误。

前提条件

SAP Sybase IQ 服务器具有一个 Simplex 数据库。

过程

注意： 缺省值 0 表示受阻塞的事务必须进行无限期等待，直至所有冲突事务释放其中的锁。

将 **BLOCKING_TIMEOUT** 数据库选项值设置成您希望事务等待冲突事务释放锁的毫秒数。

```
set temporary option blocking_timeout = '400';
```

注意： 可以在连接级别或 **PUBLIC** 级别上设置该阻塞选项。

BLOCKING_TIMEOUT 选项

控制事务为获取锁而需要等待的时长。**Multiplex** 辅助节点不支持 **BLOCKING_TIMEOUT**。

允许值

整数（以毫秒为单位）。

缺省值

0

范围

可在数据库 (**PUBLIC**) 或用户级别设置选项。在数据库级别进行设置时，值将变为任何新用户的缺省值，但不会对现有用户产生任何影响。在用户级别进行设置时，仅替

换该用户的 **PUBLIC** 值。为自身设置选项无需任何系统特权。在数据库或用户级别为任何其他用户设置选项都需要系统特权。

必须具有 **SET ANY PUBLIC OPTION** 系统特权才能设置此选项。可针对个别连接或 **PUBLIC** 角色进行临时设置。设置立即生效。

注释

当 **blocking** 选项设置为 **on** 时，如果事务试图获取与现有锁冲突的锁，则该事务需等待指定的毫秒数以便释放相应的冲突锁。如果在 **blocking_timeout** 毫秒的时间内，未释放该锁，则会向正在等待的事务返回一条错误。

将此选项设置为 **0** 会强制所有试图获取锁的事务等待所有冲突的事务释放各自的锁。

事务阻塞死锁

事务阻塞可能会导致死锁。出现死锁情况时，一组事务将进入一种特殊状态，在该状态下这些事务都不能继续执行。

有两个原因会导致死锁：

- **循环阻塞冲突** – 事务 A 被事务 B 阻塞，而事务 B 又被事务 A 阻塞。等待并不能解决这个问题，必须取消其中一个事务，使另一个事务能够继续执行。如果多个（大于二）事务出现循环阻塞，也会导致这种情况。

为消除事务死锁，数据库服务器会从陷入死锁的连接中选择一个连接，回退对该连接上活动事务的更改并返回错误。数据库服务器使用内部启发式算法选择要回退的连接，该算法优先选择剩余阻塞等待时间最少的连接（由 **BLOCKING_TIMEOUT** 选项确定）。如果所有连接均设置为始终等待，那么促使服务器检测到死锁的连接会被选作牺牲品连接。

- **所有工作线程被阻塞** – 事务被阻塞时，不会释放其工作线程。例如，为数据库服务器配置三个工作线程。事务 A、B 和 C 被当前未执行请求的事务 D 阻塞。由于没有可用的工作线程，导致出现死锁情况。这种情况称为线程死锁。

假定数据库服务器有 n 个工作线程。当 $n-1$ 个工作线程被阻塞，并且最后一个工作线程也即将被阻塞时，就会发生线程死锁。数据库服务器的内核不会允许这最后一个工作线程被阻塞，因为这样会导致所有工作线程都被阻塞，数据库服务器将因此而停止响应。数据库服务器会转而结束即将阻塞最后一个工作线程的任务，回退对该连接上的活动事务的更改，并返回错误。

如果因数据库大小或阻塞问题而出现很多长时间运行的请求，则具有数十个或数百个连接的数据库服务器可能会出现线程死锁。此时，您可能需要增大 **start_iq** 应用程序的 **-gn** 服务器选项的值。

要查看 SAP Control Center 中的锁和死锁，请参见 SAP Control Center 联机帮助。

在 Interactive SQL 中创建死锁报告事件

创建一个表和一个系统事件，用于获取死锁相关信息。

前提条件

SAP Sybase IQ 服务器具有一个 Simplex 数据库。

过程

1. 创建一个表以存储从 **sa_report_deadlocks** 系统过程返回的数据。

```
CREATE TABLE DeadlockDetails (
    deadlockId INT PRIMARY KEY DEFAULT AUTOINCREMENT,
    snapshotId BIGINT,
    snapshotAt TIMESTAMP,
    waiter INTEGER,
    who VARCHAR(128),
    what LONG VARCHAR,
    object_id UNSIGNED BIGINT,
    record_id BIGINT,
    owner INTEGER,
    is_victim BIT,
    rollback_operation_count UNSIGNED INTEGER );
```

2. 创建一个事件，用于在发生死锁时发送电子邮件通知。

```
CREATE EVENT DeadlockNotification
TYPE Deadlock
HANDLER
BEGIN
    INSERT INTO DeadlockDetails WITH AUTO NAME
    SELECT snapshotId, snapshotAt, waiter, who, what, object_id,
    record_id,
        owner, is_victim, rollback_operation_count
    FROM sa_report_deadlocks ();
    COMMIT;
    CALL xp_startmail ( mail_user = 'John Smith',
                        mail_password = 'mypwd' );
    CALL xp_sendmail( recipient = 'DBAdmin',
                      subject = 'Deadlock details added to the
DeadlockDetails table.' );
    CALL xp_stopmail ( );
END;
```

此事件会将 **sa_report_deadlocks** 系统过程的结果复制到表中，并通知管理员有关死锁的情况。

3. 将 **log_deadlocks** 选项设置为 on。

```
SET OPTION PUBLIC.log_deadlocks = 'On';
```

4. 允许记录最近执行的语句。

```
CALL sa_server_option( 'RememberLastStatement', 'YES' );
```


sa_report_deadlocks 系统过程

从数据库服务器创建的内部缓冲区中检索有关死锁的信息。

语法

sa_report_deadlocks()

特权

您必须具有系统过程的 EXECUTE 特权，以及 MONITOR 系统特权。

注释

列名	数据类型	描述
snapshotId	BIGINT	死锁实例（与特定死锁相关的所有行都具有相同的 ID）。
snapshotAt	TIMESTAMP	发生死锁的时间。
waiter	INT	正等待的连接的连接句柄。
who	VARCHAR(128)	与正等待的连接关联的用户 ID。
what	LONG VARCHAR	正等待的连接所执行的命令。 仅当通过在数据库服务器命令行中指定 -zl 选项打开捕获最新准备的 SQL 语句的功能时，此信息才可用。
object_id	UNSIGNED BIGINT	包含该行的表的对象 ID。
record_id	BIGINT	系统表的行 ID。
owner	INT	拥有正等待的锁的连接的连接句柄。
is_victim	BIT	标识回退的事务。
rollback_operation_count	UNSIGNED INT	未提交操作的数量，如果事务回退，这些操作有可能丢失。
iq_rid	UNSIGNED BIGINT	启用 IQ RLV 的表的行 ID。
iq_txn_id	UNSIGNED BIGINT	关联行的事务 ID。

当 log_deadlocks 选项设置为 On 时，数据库服务器在内部缓冲区中记录有关死锁的信息。可以使用 sa_report_deadlocks 系统过程查看日志中的信息。

副作用

无。

LOG DEADLOCKS 选项

控制是打开还是关闭死锁报告功能。

允许值

On、Off

缺省值

Off

范围

只能在数据库 (PUBLIC) 级别设置选项。

必须具有 SET ANY SYSTEM OPTION 系统特权才能设置此选项。设置立即生效。

注释

当此选项设置为 On 时，数据库服务器在内部缓冲区中记录有关死锁的信息。缓冲区大小是固定的，即 10000 字节。您可以使用 sa_report_deadlocks 存储过程来查看死锁信息。当此选项设置为 Off 时，缓冲区的内容会被保留。

发生死锁后，仅报告死锁中涉及的那些连接的信息。报告连接的顺序取决于连接等待行的情况。对于线程死锁，将报告有关所有连接的信息。

死锁报告功能打开时，您还可以使用 Deadlock 系统事件在发生死锁时执行操作。

sa_conn_info 系统过程

报告连接属性信息。

语法

```
sa_conn_info( [ connidparm ] )
```

参数

- **connidparm** - 此可选 INTEGER 参数指定连接 ID 号。缺省值为 NULL。

结果集

列名	数据类型	说明
Number	INTEGER	返回当前连接的连接 ID（数字）。
Name	VARCHAR(255)	返回当前连接的连接 ID（数字）。 临时连接名在连接名前面附有 INT:。
Userid	VARCHAR(255)	返回连接的用户 ID。
DBNumber	INTEGER	返回数据库的 ID 编号。

列名	数据类型	说明
LastReqTime	VARCHAR(255)	返回指定连接的最后一个请求开始的时间。此属性可以为内部连接（如事件）返回空字符串。
ReqType	VARCHAR(255)	返回上次请求的类型。如果连接已由连接池进行了高速缓存，则其 ReqType 值为 CONNECT_POOL_CACHE。
CommLink	VARCHAR(255)	返回连接的通信链接。这是 SAP Sybase IQ 所支持的网络协议之一，或者 local，如果为相同计算机连接。
NodeAddr	VARCHAR(255)	返回客户端/服务器连接中客户端的地址。
ClientPort	INTEGER	返回客户端的 TCP/IP 端口号；或者如果连接不是 TCP/IP 连接，则返回 0。
ServerPort	INTEGER	返回数据库服务器的 TCP/IP 端口号或 0。
BlockedOn	INTEGER	如果当前连接未被阻塞，则返回零；如果当前连接被阻塞，则返回由于锁定冲突而阻塞连接的连接号。
LockRowID	UNSIGNED BIGINT	返回锁定行的标识符。 如果连接并未等待与行相关联的锁（即它未等待锁或正在等待不与行相关联的锁），则 LockRowID 为 NULL。
LockIndexID	INTEGER	返回锁定索引的标识符。 如果锁与 LockTable 中的表上的所有索引相关联，则 LockIndexID 为 -1。如果连接并未等待与索引相关联的锁（即它未等待锁或正在等待不与索引相关联的锁），则 LockRowID 为 NULL。
LockTables	VARCHAR(255)	如果连接正在等待锁，则返回与锁相关联的表名称。否则，LockTable 返回一个空字符串。
UncommitOps	INTEGER	返回未提交的操作数。
ParentConnection	INTEGER	返回已创建临时连接以执行数据库操作（如执行备份或创建数据库）的连接的连接 ID。对于其它类型的连接，此属性返回 NULL。

注释

如果 *connidparm* 小于零，则返回由当前连接的连接属性组成的结果集。如果 *connidparm* 未提供或为 NULL，则为数据库服务器上运行的所有数据库的所有连接返回连接属性。

当发生阻塞时，使用此过程所返回的 **BlockedOn** 值可以检查哪些用户被阻塞以及是谁阻塞的。**sa_locks** 系统过程可用于显示阻塞的连接所持有的锁。

有关基于任意这些属性的详细信息，可执行与以下内容：

```
SELECT *, DB_NAME( DBNumber ),
        CONNECTION_PROPERTY( 'LastStatement', Number )
FROM sa_conn_info( );
```

LockRowID 的值可用于在 **sa_locks** 过程的输出中查找锁。

LockIndexID 的值可用于在 **sa_locks** 过程的输出中查找锁。此外，**LockIndexID** 的值对应于 **ISYSIDX** 系统表的主键，可使用 **SYSIDX** 系统视图查看此系统表。

每个锁都具有关联的表，因此，**LockTable** 的值可用于明确地确定某连接是否正在等待锁。

特权

为当前连接 **ID** 执行此系统过程不需要任何特权。要为其它连接执行此系统过程，您必须具有 **SERVER OPERATOR**、**MONITOR** 或 **DROP CONNECTION** 系统特权。

副作用

无

示例

以下示例使用 **sa_conn_info** 系统过程为连接到服务器的所有连接返回一个汇总连接属性的结果集。

```
CALL sa_conn_info( );
```

Number	Name	Userid	DBNumber	...
79	SQL_DBC_10dcf810	DBA	0	...
46	setup	User1	0	...
...

以下示例使用 **sa_conn_info** 系统过程返回结果集，其中显示是哪个连接创建了临时连接。

```
SELECT Number, Name, ParentConnection FROM sa_conn_info();
```

Connection 8 创建了执行 **CREATE DATABASE** 语句的临时连接。

Number	Name	ParentConnection
1000000048	INT: CreateDB	8
9	SQL_DBC_14675af8	(NULL)
8	SQL_DBA_152d5ac0	(NULL)

版本

在某个事务取得表或表行的必要写入锁之后，数据库服务器将绑定创建该事务时的快照版本。表级版本控制启用单一写入程序访问和表级锁定。行级版本控制 (RLV) 启用对启用 RLV 的表的并发写入程序访问和行级锁定。

表级快照版本控制

缺省事务处理行为 **表级快照版本控制** 允许多个写入程序以串行方式修改一个表（即，写入程序相继进行修改，每次不超过一个），而多个读取程序继续使用该表的原始副本。

如果没有表级快照版本控制，并发读取和写入操作可能会导致 IQ 主存储库中出现不一致。**SAP Sybase IQ** 提供的表级版本控制可通过序列化事务以及将表设为版本级别的方式来防止出现不一致情况。

虽然任何事务处理系统都被设计为确保数据库保持一致，但 **SAP 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。**SAP Sybase 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。

在启用表级快照版本控制但将表配置为 **RLV** 存储时，仍不允许进行并发读取和写入操作。表级快照版本控制的设置会覆盖表的 **RLV** 设置。

启用表级快照版本控制

缺省情况下，表级快照版本控制为启用状态。如果更改了缺省的版本控制方法，请使用 **ALLOW_SNAPSHOT_VERSIONING** 数据库选项予以启用。

通过发出以下语句临时设置表级版本控制：

```
SET TEMPORARY OPTION Snapshot_Versioning = 'Table-level'
```

表级锁定

所有表级锁都会根据用户请求的操作类型自动产生。

无需显式请求锁；对表具有访问权限的事务持有锁。

当某个表在 **SAP Sybase IQ** 中遭到锁定时，除数据定义操作以外的其它事务都不具有对该表的写入权限。任何其它写入事务在尝试访问具有写锁定的表时都会收到错误。任何事务都可以获得读取权限。

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写锁定。**LOCK TABLE** 将提交当前事务，并允许事务在锁可用之前排入队列。

DML 操作的表锁

数据操作语言 (DML) 操作包括插入、删除、更新和查询。对于所有此类操作，表级快照版本控制允许在任意给定表上使用一个写入程序和多个读取程序。

在表级快照版本控制中：

- 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。

第一个在写入模式下打开表的事务会获取对该表的访问权限。第二个尝试在写入模式下打开该表的事务会收到错误。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。

SAP Sybase IQ 支持共享、写入和独占锁定排入队列，允许您将表锁定指定的一段时间。可以同时多个表执行写锁定。

为避免因后续 DML 语句导致的将来版本错误，请使用 **LOCK TABLE** 语句对计划修改的一个表或一组表保持写锁定。

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。例如，当 DML 语句因完整性约束原因而失败时，需要显式执行 **COMMIT** 或 **ROLLBACK** 才能释放锁。

如果 DML 语句因所引用的表中存在参照完整性违规或其它表中的锁不可用而失败，将会出现：“SQL Anywhere 错误 -210”。

DDL 操作的表锁

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。

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。

可能出现以下错误消息：

```
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，则检查点将失败，并显示以下消息：

```
Run time SQL Error
```

如果正在执行 **CHECKPOINT** 命令，则发出 **CREATE DBSPACE** 或 **DROP DBSPACE** 命令的用户会看到：

```
Cannot perform requested command as there is a
CHECKPOINT command in progress.
```

在删除过程中发出 **CREATE DBSPACE** 的用户会看到：

```
Cannot perform requested command as there is a
DROP DBSPACE command in progress.
```

在创建过程中发出 **DROP DBSPACE** 的用户会看到：

```
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。

这些索引创建命令规则存在一个例外。如果在受索引创建影响的表中使用含 **SELECT** 语句的 **CREATE INDEX** 命令，则 SAP Sybase IQ 将禁止使用新索引，直至创建索引的事务提交。

虽然 **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；**SET OPTION** 会导致自动提交，除非将其指定为 **TEMPORARY**。不允许当前连接到数据库的任何用户使用 **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，但某些在重新启动数据库服务器后才生效的特定选项除外。

主键和锁定

因为只有一名用户可以在启用表级快照版本控制时更新表，所以主键的生成不会导致并发冲突。

显示活动表锁

标识具有锁定表的用户。如果其它事务持有某个表的锁定，则向该表进行的写入尝试将失败。

1. 运行 **sp_iqtransaction**。
2. 在输出中查找事务标识符和用户名。
3. 运行 **sp_iqllocks** 过程。

对于当前数据库的目录存储库和 IQ 存储库中的每个锁，**sp_iqllocks** 将显示：

- 持有锁的连接和用户 ID
- 持有锁的表
- 锁类型，以及用于标识锁的名称

例如，当另一事务持有锁时，您会看到：

```
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，并在同一输出行中查找用户名。

表级快照版本控制的性能影响

通常来说，表级快照版本控制对性能的影响非常小。

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。
- 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、回退和检查点可能会阻塞其它读取或写入事务，但持续时间非常短。

所有这些性能和磁盘使用因素对系统的影响程度都取决于您对 SAP Sybase 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

要在使用表级版本控制时删除数据，可能需要通过向 IQ 主存储库中添加 **dbspace** 来实际增加磁盘空间。

执行删除时所需的空间量更多取决于数据页中的数据分布，而不是所删除的行的尺寸或数量。**SAP Sybase IQ** 必须将每个含任意待删除数据的页面版本保留下来，从删除开始执行时起一直保留到事务提交。如果所删除的行分布在许多数据页中，则需要使用 IQ 主存储库中的空间来保留所有这些额外的数据页。

例如，假设您必须从每页有 100 行的数据库中删除 10 行。如果这 10 行中的每一行都位于不同的数据页上，则 IQ 主存储库必须拥有用于存储 10 个版本页（每页都足以容纳 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素索引类型而异。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 **SAP Sybase IQ** 会停止删除，并在通知日志中写入以下消息：

```
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 **dbspace** 以重新获取额外磁盘空间来用于其它用途。在插入任何可能需要新 **dbspace** 的数据之前执行此操作。

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 **dbspace**，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。

行级快照版本控制

行级快照版本控制仅适用于为内存 **RLV** 存储库启用的表。行级快照版本控制允许多个写入程序对表进行并发 **DML** 更改，但永远也不要同时对相同的行进行并发 **DML** 更改。

行级快照版本控制使用行锁在行级锁定表。行锁将为某个表行提供写入锁，意味着事务被阻塞或失败，具体情况取决于 **BLOCKING** 和 **BLOCKING_TIMEOUT** 选项的设置。如果 **BLOCKING** 设置为 **ON**，则事务阻塞。如果 **BLOCKING** 设置为 **OFF**，事务立即失败，并抛出 **ALREADY LOCKED SQL** 异常。

事务阻塞允许行级快照版本控制同时写入到同一个表的不同行中。根据 **BLOCKING** 和 **BLOCKING_TIMEOUT** 选项设置，行锁争用将导致出错，或者，如果在指定超时期限内释放锁，则会尝试重新获取锁。当为表级版本控制配置的事务尝试向含有由行级版

本控制事务锁定的行的表中写入内容时，表级事务或者失败，显示一条错误，或者阻塞，并且如果锁在指定超时期限内被释放则尝试重新获取锁。

DDL 更改为表（**CREATE**、**DROP** 和 **ALTER**），但在表级锁定表。

指定快照版本控制

使用 **SNAPSHOT_VERSIONING** 选项将快照版本控制类型设置为 Row-level 或 Table-Level。您可以在数据库 (PUBLIC) 级别、连接级别 (TEMPORARY) 或者用户级别设置该选项。要使用内存 RLV 存储库，请为事务启用行级快照版本控制。要同时更新同一个表的不同行，每个事务或连接还要必须启用行级快照版本控制

前提条件

- 如果设置为行级，RLV 存储 dbspace 存在，且包含至少一个 dbfile。
- 如果设置为行级，表为启用 RLV 的表。
- 必须具备 SET ANY PUBLIC OPTION 系统特权才能为 PUBLIC 或者其他用户或角色设置此选项。

过程

为某个事务设置快照版本控制属性后，该属性将一直保持不变，直到该事务提交。

1. 确定 **SET OPTION** 命令的范围以将选项设置为数据库范围的选项、连接级别的选项或用户级别的选项：
 - **SET OPTION public.SNAPSHOT_VERSIONING...**
 - **SET TEMPORARY OPTION SNAPSHOT_VERSIONING...**
 - **SET OPTION *username*.SNAPSHOT_VERSIONING...**
2. 指定快照版本控制的类型。

级别	选项
Row-level	行级快照版本控制。内存 RLV 存储库需要此选项。行级快照版本控制允许多个写入程序对表进行并发 DML 更改，但永远也不要同时对相同的行进行并发 DML 更改。
Table-level	典型（向后兼容）SAP Sybase IQ 版本控制行为。在表级拍摄快照。多个写入程序不能对表进行并发 DML 更改。

```
SET TEMPORARY OPTION Snapshot_Versioning = 'Row-level';

CREATE TABLE rv_locks(c1 int, c2 int, c3 int);

INSERT INTO rv_locks VALUES (1,1,1);
INSERT INTO rv_locks VALUES (2,2,2);
INSERT INTO rv_locks VALUES (3,3,3);
INSERT INTO rv_locks VALUES (4,4,4);
COMMIT;
```

教程：监控意图写锁

在本教程中，您将创建启用 RLV 的表，执行事务，并使用 **sp_iqlocks** 存储过程报告数据库中的模式级锁和意图写锁。然后使用 **sp_iqconnection** 和 **sa_conn_info** 存储过程查看控制着意图写锁的内部连接。

前提条件

- SAP Sybase IQ 服务器具有一个 Simplex 数据库。
- 已配置 RLV 存储库。

过程

提示： 可以使用 SAP Control Center 对锁进行监控。有关详细信息，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

1. 创建启用 RLV 的表 **rv_locks** 和 **rv_locks2**，并配置表级快照版本控制。

```
SET TEMPORARY OPTION SNAPSHOT_VERSIONING = 'Table-level';

CREATE TABLE rv_locks(c1 INT, c2 INT, c3 INT);

INSERT INTO rv_locks VALUES (1,1,1);
INSERT INTO rv_locks VALUES (2,2,2);
INSERT INTO rv_locks VALUES (3,3,3);
INSERT INTO rv_locks VALUES (4,4,4);
COMMIT;

CREATE TABLE rv_locks2(c1 int, c2 int, c3 int);

INSERT INTO rv_locks2 VALUES (1,1,1);
INSERT INTO rv_locks2 VALUES (2,2,2);
INSERT INTO rv_locks2 VALUES (3,3,3);
INSERT INTO rv_locks2 VALUES (4,4,4);
COMMIT;

ALTER TABLE rv_locks ENABLE RLV STORE;
ALTER TABLE rv_locks2 ENABLE RLV STORE;
```

2. 启用连接阻塞并设置阻塞超时阈值：

```
SET TEMPORARY OPTION BLOCKING = 'ON';
SET TEMPORARY OPTION BLOCKING_TIMEOUT = '0';
```

3. 使用 **sp_iqlocks** 存储过程查看当前的数据库锁集。此时，未返回任何锁。

```
sp_iqlocks
```

启用 RLV 的表缺少意图写锁，表明该表的内存 RLV 部分仍有待创建。

4. 将事务的快照版本控制属性设置为行级。

```
SET TEMPORARY OPTION SNAPSHOT_VERSIONING = 'row-level';
```

5. 写入表中。

```
INSERT INTO rv_locks VALUES (5,5,5);
```

对启用 **RLV** 的表进行写入或查询将会在要求时在内存中创建该表的启用 **RLV** 的部分。

6. 重新执行 **sp_iqlocks**。

```
sp_iqlocks
```

这次，该过程返回意图写锁。

```
conn_name,conn_id,user_id,table_type,creator,table_name,index_id,
lock_class,lock_duration,lock_type,row_identifier,row_range
'SQL_DBC_13cd6038',
3,'DBA','BASE','DBA','rv_locks','','Schema','Transaction','Shared',
,
'RLV_CONN_T775',
1000000407','','BASE','DBA','rv_locks','','Table','Transaction','Int
ent',,
```

连接 ID 100000407 在 **rv_locks** 表中具有意图写锁。该锁类型设置为 Intent，表示意图写锁。

注意： 连接 ID 号 (100000407) 很大，因为它表示服务器本身的内部连接。此内部连接用于管理启用 **RLV** 的表中的锁。

ConnectionID 3 在该表中具有模式锁。该锁类型设置为 Shared，表示共享模式锁。

7. 返回到执行插入的未提交事务，并将其提交：

```
Commit
```

在提交过程中，数据库会释放该事务持有的锁。对于本教程，此操作仅释放共享模式锁。启用 **RLV** 的表现现在存在于内存中，其中含有已提交的数据。因此，此时仅存的锁是由该表启用 **RLV** 的部分所持有的意图写锁。

8. 重新执行 **sp_iqlocks**：

```
sp_iqlocks
```

模式锁已不存在，但意图写锁仍然存在：

```
conn_name,conn_id,user_id,table_type,creator,table_name,index_id,
lock_class,lock_duration,lock_type,row_identifier,row_range
'RLV_CONN_T775',
1000000407','','BASE','DBA','rv_locks2','','Table','Transaction','In
tent',,
```

注意： 自从上次执行 **sp_iqlocks** 后，conn_id 100000407 行一直没有发生更改。

9. 执行 **sp_iqconnection** 查看连接详细信息：

```
sp_iqconnection
```

您会看到：

```
ConnHandle,Name,UserId,LastReqTime,ReqType,IQCmdType,LastIQCmdTime,IQCursors,LowestIQCursorsState,IQthreads,TxnID,ConnCreateTime,TempTableSpaceKB,TempWorkSpaceKB,IQconnID,satoiq_count,iqtosa_count,CommLink,NodeAddr,LastIdle,MPXServerName,LSName,INCCConnName,INCCConnSuspended
1,'SQL_DBC_13de5fd8','DBA','2012-08-08
08:49:25.629','PREFETCH','NONE',2012-08-08 08:49:25.0,0,'NONE',
0,0,2012-08-08 08:49:24.0,0,0,70,40,2,'local','','0,,,','N'
3,'SQL_DBC_13cd6038','DBA','2012-08-08
09:25:32.920','OPEN','IQUTILITYOPENCURSOR',2012-08-08
09:25:32.0,0,'NONE',0,1008,2012-08-08
08:50:04.0,0,0,92,187,413,'local','','8789,,,','N'
1000000407,'INT: RLVLockConn','','','unknown (0)','NONE',
0001-01-01 00:00:00.0,0,'NONE',0,0,2012-08-08
09:00:40.0,0,0,410,2,0,'NA','NA',0,,,','N'
```

第三行 (ConnHandle 1000000407) 提供关于启用 RLV 的表在控制意图写锁时所用的内部连接 (RLVLockConn) 的信息。

注意： ConnHandle 1000000407 与 **sp_iqlocks** 输出中的 conn_id 1000000407 相对应。它也与 **sp_iqtransaction** 输出中的 ConnHandle 1000000407 相对应。

10. 执行 `sa_conn_info` 查看更多的连接详细信息。`sa_conn_info` 类似于 `sp_iqconnection`。

```
sa_conn_info
```

您会看到：

```
Number,Name,UserId,DBNumber,LastReqTime,ReqType,CommLink,NodeAddr
,ClientPort,ServerPort,BlockedOn,LockRowID,LockIndexID,LockTable,
UncommitOps,ParentConnection
1000000407,sa_ INT: RLVLockConn','',0','', 'unknown (0)', 'NA', 'NA',
0,0,0,0,,',0,
3,'SQL_DBC 13cd6038','DBA',0,'2012-08-08
09:30:43.799','FETCH','local','',0,0,0,0,,',0,
1,'SQL_DBC 13de5fd8','DBA',0,'2012-08-08
08:49:25.629','PREFETCH','local','',0,0,0,0,,',0,
```

注意：在第一行中，数字 1000000407 与 **sp_iqconnection** 输出中的 ConnHandle 1000000407 和 **sp_iqlocks** 输出中的 conn_id 1000000407 相对应。

Userid "INT:RLVLockConn" 表示内部连接。该连接由启用 RLV 的表用于控制意图写锁。

行鎖

行锁是表行写入锁，它允许持有该锁的事务向锁定行的任意列写入内容。一次只能存在此锁的一个持有者。意图写锁是前提条件，事务必须先持有意图写锁，这样锁管理器才能为其授予行锁。

表行写入锁允许持有该锁的事务向锁定行的任意列写入内容。请求事务必须先持有意图写锁，然后才能授予此锁。行写入锁是独占锁，任何时候，只有一个事务能够持有

某一行的写入锁。一个事务获取了写入锁后，其它事务要求锁定该行的请求都将被拒绝。

行锁只在行删除期间存在。RLV 存储库是仅附加存储库，意味着每个写入操作都会向存储库中附加一个新行。**INSERT** 语句向存储库中附加一个新行，就像执行 **UPDATE** 语句一样。RLV 存储库将 **UPDATE** 视为 **DELETE** 加上 **INSERT**。在删除某一行之前，数据库将在 **DELETE** 或 **UPDATE** 语句的上下文中删除行级锁。

监控 Interactive SQL 中的意图写锁

使用 **sa_locks** 系统过程查看有关意图写锁的详细信息。

前提条件

- 已配置 RLV 存储库。
- 已为 RLV 存储注册表。
- 将数据库选项 **SNAPSHOT_VERSIONING** 设置为 'Row-level'。

过程

1. 运行 **sa_locks** 系统过程。
2. 查看 **lock_class** 列，该列显示了表和行的意图锁。
3. 运行 **sp_iqlocks** 存储过程，以标识正在阻止其他用户对表进行写入操作的用户。此过程将显示关于数据库中当前持有的锁的信息，包括持有锁的连接和用户 ID、持有锁的表、锁类型以及用于标识锁的名称。

意图写锁

意图写锁是表写入锁，它授予事务在以后向表行中写入内容的权限。一个意图写锁可由多个请求连接持有。

如果内存中存在表的启用 RLV 部分，将始终存在意图写锁。使用 **sp_iqlocks** 存储过程可查看意图写锁的详细信息。

意图写锁与表写入锁和表独占锁相冲突。此冲突会阻止表级快照版本控制事务向表中写入内容或执行 DDL 操作，一直到锁管理器释放表的所有意图写锁。在表级快照版本控制事务和行级快照版本控制事务连接同时向表中写入内容的情况下，意图写锁将提供同步。考虑以下情形：

连接	操作
行级快照版本控制事务 A	<ul style="list-style-type: none">• 执行向 table_1 的多个行中写入内容的查询。• 锁管理器为 table_1 创建一个意图写锁。• 锁管理器为行级 DML 更新创建多个本地意图写锁。锁管理器创建行级锁。

连接	操作
表级快照版本控制事务 B	试图向 table_1 中写入内容。事务 B 被意图写锁阻塞。
行级快照版本控制事务 A	提交事务 A。将 RLV 存储库中的表更改合并到 IQ 主存储库中。意图写锁被释放。
表级快照版本控制事务 B	继续向 table_1 中写入内容。

行级 DDL 锁定的注意事项

数据定义语言 (DDL) 对启用 RLV 的表进行的更改 (如 **CREATE INDEX**、**DROP INDEX** 和 **ALTER TABLE ADD**、**ALTER** 或 **DROP**) 需要一个独占表级锁。对于 DDL 事件, 启用 RLV 的表具有与 IQ 主存储库相同的锁定行为: 写入连接在该表中有一个独占锁。当 **BLOCKING** 设置为 ON 时, 将阻塞针对该表的所有竞争 DML 和 DDL 事务, 直至提交 DDL 更改。当 **BLOCKING** 设置为 OFF 时, 竞争事务将立即停止锁定请求。

查看事务的版本控制类型

使用 **sp_iqtransaction** 存储过程可查看每个活动事务的版本控制类型 (表级或行级)。

1. 运行 **sp_iqtransaction**。
2. 检查 VersioningType 列。

VersioningType	描述
Row-level	事务使用行级版本控制。行级版本控制启用对启用 RLV 的表的并发写入程序访问和行级锁定。
Table-level	事务使用表级版本控制。表级版本控制启用单一写入程序访问和表级锁定。

另请参见

- **sp_iqtransaction** 过程 (第 176 页)

临时表版本控制

在数据库中创建的临时表称为**全局临时表**。声明 (而不是在数据库中创建) 的临时表称为**局部临时表**。

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。

只有一名用户可以查看局部临时表中的任何行。当该用户断开连接时, 将会删除该表。当您声明局部临时表时, SAP Sybase IQ 会发出一个保存点而不是自动提交事务,

就像它对其它任何类型表进行的数据定义操作一样。在创建索引前提交局部临时表中的数据。如果尝试使用未提交的数据创建索引，您可能会看到：“必须提交局部临时表<表名> 才能够创建索引。”

SAP Sybase IQ 不区分版本控制基表（IQ 主存储数据库表）和版本控制全局临时表。因为只有一名用户可以访问任何临时表中的数据，所以永远不会有一个以上针对临时表的写入事务处于打开状态。

您可以启用全局临时表或局部临时表进行 RLV 存储，即使临时表无法利用 RLV 的多用户并发写入功能也是如此。启用临时表进行 RLV 存储可以利用 RLV 存储的低延迟 DML。

调查锁争用对性能的影响

有些装载或查询性能问题可能源于锁争用。如果您的内核系统时间大于 10%，则可能发生了锁争用问题。

1. 要了解锁争用是否影响系统的性能，请执行以下操作之一：
 - 使用 **-contention** 选项运行 SAP Sybase IQ 监控器。
 - 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。
2. 如果您怀疑存在锁争用的情况，通过设置以下选项直接控制分区级别可能会很有用：
 - **-iqpartition** 服务器启动选项。
 - **cache_partitions** 数据库选项。

注意： 如果内核系统时间高于正常值也表示您的内核未合理调优。如果属于这种情况，可能需要调整内核参数；更改 SAP Sybase IQ 设置不会解决内核调整不当的问题。

另请参见

- **CACHE_PARTITIONS** 选项（第 11 页）

(不建议使用) -contention

显示很多关键的缓冲区高速缓存和内存管理器锁定。这些锁定和互斥计数器显示在缓冲区高速缓存和堆内存中的活动，以及解除这些锁定的速度。如果超时数超过 20%，则表示有问题。

用法

```
monitor_options -contention
```


输出

表 20. -contention 输出字段

输出字段	描述
<i>AU</i>	当前活动用户数
<i>LRULks</i>	LRU 的锁定次数（对临时高速缓存重复）
<i>woTO</i>	授予锁定而没有超时的次数（对临时高速缓存重复）
<i>Loops</i>	在授予锁之前 SAP Sybase IQ 重试的次数（对临时高速缓存重复）
<i>TOs</i>	SAP Sybase IQ 超时并且必须等待锁的次数（对临时高速缓存重复）
<i>BWaits</i>	高速缓存中的缓冲区忙等待数（对临时高速缓存重复）
<i>IOLock</i>	SAP Sybase IQ 锁定压缩 I/O 池的次数（对临时高速缓存重复）；可忽略
<i>IOWait</i>	SAP Sybase IQ 必须等待压缩 I/O 池锁的次数（对临时高速缓存重复）；可忽略
<i>HTLock</i>	SAP Sybase IQ 锁定块映射散列表的次数（对临时高速缓存重复）
<i>HTWait</i>	SAP Sybase IQ 必须等待块映射散列表的次数（对临时高速缓存重复）；HTLock 和 HTWait 指示您正在使用的块映射的数量
<i>FLLock</i>	SAP Sybase IQ 必须锁定空闲列表的次数（对临时高速缓存重复）
<i>FLWait</i>	SAP Sybase IQ 必须等待空闲列表锁的次数（对临时高速缓存重复）
<i>MemLks</i>	SAP Sybase IQ 取得内存管理器（堆）锁的次数
<i>MemWts</i>	SAP Sybase IQ 必须等待内存管理器锁的次数

注意： SAP Sybase IQ 不再使用旋转锁。因此，很少使用 woTO、Loops 和 TOs 统计信息。

检查点、保存点和事务回退

除允许并发以外，快照版本控制事务处理也在数据恢复中扮演着重要角色。

SAP Sybase IQ 依赖于与事务相关的命令，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。**CHECKPOINT** 命令设置检查点，**SAVEPOINT** 命令设置保存点，**RELEASE SAVEPOINT** 释放保存点，**ROLLBACK TRANSACTION** 回退事务。

Checkpoints

当 SAP Sybase IQ 向磁盘写入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。SAP Sybase IQ 在数据库恢复期间使用该信息。

SAP Sybase IQ 使用不同于 OLTP 数据库的检查点，如 SQL Anywhere。OLTP 数据库往往具有短事务，这些事务只会影响少数行。对这些事务而言，将整个页面写入到磁盘的成本非常昂贵。相反，OLTP 数据库通常会在检查点处向磁盘写入数据，并且只写入已更改的数据行。

SAP Sybase IQ 是 OLAP 数据库。一个 OLAP 事务可以更改上千行甚至几百万行数据。因此，数据库服务器不会一直等到出现检查点再执行物理写入。在每个事务提交后，它都会将已更新的数据页写入磁盘中。对于 OLAP 数据库，将整页数据写入磁盘比在任意检查点处写入少量数据更加有效。

大多数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。

在发生以下情况时会出现检查点：

- 当事务发出 **CHECKPOINT** 命令时
- 当超过 **CHECKPOINT_TIME** 时
- 当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。缺省情况下，此时间为 60 分钟。要调整检查点时间间隔，请使用 **SET OPTION** 语句。调整检查点时间或发出显式检查点可能不是必要操作。与在 OLTP 数据库产品中相比，在 SAP Sybase IQ 中，控制检查点并不那么重要，因为 SAP Sybase IQ 会在每个事务提交后写入实际数据页。

事务内的保存点

SAP 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（进而写入到 IQ 主存储库中）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。

释放保存点

释放保存点可以释放该保存点所用尽的版本页。

对数据的版本控制在页面级别内部进行。SAP Sybase IQ 仅维护已更新页的单独副本；剩余的页将与以前版本共享。

释放保存点可以充分利用磁盘空间。

以下 Interactive SQL 命令将释放保存点 *n*：

```
RELEASE SAVEPOINT n
```

保存点后的所有资源都将被释放，因此无法回退到任何中间保存点。

RELEASE SAVEPOINT 不会释放锁。

回退到保存点

ROLLBACK TO SAVEPOINT 命令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，或在未指定命名的保存点时回退到最近的保存点。回退到保存点 *n* 会撤消在所有大于或等于 *n* 的保存点执行的所有操作。

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，然后您执行了写入操作。将事务回退到以前的保存点。回退会撤消该保存点之后的所有操作，包括写入操作以及该事务在要回退到的保存点之后获得的所有锁。

SAP Sybase IQ 支持对可更新的游标执行保存点操作。

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。要在单个 DML 命令结束后保留数据页版本，请设置自己的命名保存点。

命名和嵌套的保存点

命名的嵌套保存点可在事务中提供多个活动保存点。

可以通过回退到前一个保存点或回退事务本身来取消 **SAVEPOINT** 和 **RELEASE SAVEPOINT** 之间的更改。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，所有保存点都将被释放。

保存点会导致 SAP 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，可自动或根据显式用户请求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。

回退原因

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** 语句撤消自上次执行 **COMMIT** 或 **ROLLBACK** 以来对数据库进行的所有更改。

使用 **ROLLBACK TO SAVEPOINT** 语句撤消在您命名的 **SAVEPOINT** 之后对数据库进行的所有更改，或者撤销最后一个 **SAVEPOINT** 之后的更改。

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 SAP Sybase IQ 会自动回退数据库。回退的目标是最近的提交或回退。

回退的影响

回退可使主存储库和临时存储库恢复到它们以前的状态，同时释放锁。

- 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：

- 无论游标是在所回退的事务中打开还是在以前的事务中打开，事务回退都会关闭所有游标。
- 回退到保存点将关闭在该保存点后打开的所有游标。

系统恢复

如果发生系统故障或电源中断，或在数据库服务器停止后进行重新启动时，SAP Sybase IQ 会自动尝试执行恢复。

在 SAP Sybase IQ 数据库恢复期间，会回退任何未提交的事务，而且任何用于旧版本的磁盘空间都会返回到可用空间池中。此时，数据库中只包含各个永久表最近提交的版本。

在系统故障恢复期间，SAP Sybase IQ 会重新打开故障发生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）在出现故障时处于有效状态，则会使用至少与故障发生时所使用的连接数量相同的连接重新启动 IQ 服务器。无法恢复临时表中的内容。

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。

在系统出现故障后运行存储过程 `sp_iqcheckdb`，最好在允许用户进行连接之前运行。此过程会检查数据库中的每个块，并生成可用于检查数据库的一致性和完整性的统计信息。

事务信息如何帮助进行恢复

SAP Sybase IQ 恢复机制专为数据仓库环境而设计，其中的事务通常数量少而长度长。

SAP Sybase IQ 在实际数据库页的副本上执行数据库更新，然后在提交写入事务时将数据写至磁盘中。SAP Sybase IQ 还会记录以下内容：

- 每个事务的所更改数据的位置和数量存储在 *事务日志* 中。
- 任意版本页和可用空间的位置存储在磁盘中。当不再需要这些版本时，SAP Sybase IQ 使用此信息释放空间。当提交或回退写入事务时，在写入事务持续期间创建的所有版本都会过时。可在保存点处释放各个版本。
- 有关在事务执行期间出现的检查点的其它信息。

当需要恢复数据库时，**SAP Sybase IQ** 会通过事务日志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，并释放由已过时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。但如果系统中有大量会更改数据的事务，则事务日志可能会变得相当大，需要定期截断。

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。

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。

事务中的游标

以名为“游标”的数据类型格式返回 **SELECT** 结果。

游标类似于表，只不过其中一行被标识为当前行。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。

您可能会发现在编写过程或编写使用嵌入式 **SQL** 访问数据库的应用程序时，游标非常有用。许多前端查询工具也会使用游标。使用 **Interactive SQL** 时，游标不可用。

SAP Sybase IQ 游标是可更新的，可用于在处理游标时修改数据库中的基础数据。

游标中的行与表中的行相似，不具有与其关联的顺序。**FETCH** 命令逐步处理各个行，但是顺序可能会随机出现而且会不一致。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打开的游标的信息。

游标和版本控制

使用游标时，**SAP Sybase IQ** 必须能够在一个事务中管理多个版本。

例如，假设您在 *x* 时打开一个使用 *customer* 表的名为 *cust_cursor* 的游标。您在 *y* 时更新该表。**SAP Sybase IQ** 会在从时间 *x* 至 *cust_cursor* 的使用完成这段时间内保留 *customer* 表的版本。

SAP Sybase IQ 的游标支持面向游标在 **DSS** 应用程序中的可能用途。

游标敏感性

如果游标的成员资格（其返回的数据行）从游标打开到关闭这段时间内可以发生变化，则认为相应游标是敏感的。不敏感游标处于打开状态时，其成员资格固定不变。

不敏感游标的结果集的成员资格和值是否更改是不确定的。对值敏感的游标对于它的成员资格不敏感，而对于结果集的顺序和值敏感。SAP Sybase IQ 支持可更新的不敏感游标。

游标滚动

SAP 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**，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的游标操作，则将此选项设置为 **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'** 设置为永久的 **PUBLIC** 选项。它将使用更少内存，并能提高查询性能。

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 SAP Sybase IQ 中，持有游标在提交前一直是可更新的。提交后，会将持有游标内部标记为只读，后续定位更新会产生错误。

虽然 **HOLD** 选项不常用于 DSS 环境中，但在一些情况下，该选项十分有用。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。

SAP 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。

SAP Sybase IQ 支持定位读取，该功能在较长的查询事务中很有帮助。SAP 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。大多数情况下，对 SAP Sybase IQ 数据库进行的更新将会涉及到大量数据；重新定位是此类写入操作的很小部分。

定位更新和删除将被视为操作在游标上进行处理，因此将作为其事务的一部分而不是单独的语句进行处理。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。

游标的消息记录

缺省情况下，不会在 SAP Sybase IQ 消息文件中记录游标操作。

要跟踪游标操作以确定问题的原因，请打开 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。

远程事务

涉及远程服务器的事务管理使用 *两阶段提交* 协议。

SAP Sybase IQ 执行的策略可以确保大多数情况的事务完整性。

远程事务限制

远程事务管理具有保存点和嵌套语句限制。

事务管理的限制包括：

- 不将保存点传播到远程服务器。
- 如果涉及远程服务器的事务中包含嵌套的 **BEGIN TRANSACTION** 和 **COMMIT TRANSACTION** 语句，则只处理最外层的一组语句。系统不会将包含 **BEGIN TRANSACTION** 和 **COMMIT TRANSACTION** 语句的不一致的一组语句传输到远程服务器。

创建过程和批处理

过程和批处理增强了 SAP Sybase IQ 数据库的安全性，提高了效率并促进了标准化。

过程在数据库中存储 SQL 语句，以供所有应用程序使用。它们增强了数据库的安全性，提高了效率并促进了标准化。用户定义的函数是将值返回到调用环境以供查询和其它 SQL 语句使用的过程。

出于多种目的，服务器端 JDBC 提供比 SQL 存储过程更灵活的方法来将逻辑内置于数据库中。

批处理是 SQL 语句的集合，它们作为一组提交到数据库服务器。过程中提供的许多功能（例如控制语句），批处理中同样会提供。

注意： 使用源代码控制软件可跟踪您部署到数据库的对于源代码的更改，以及对于从源（包括存储过程）创建的对象更改。

过程

创建过程

要创建 SAP Sybase IQ 过程，使用 SQL 语句 **CREATE PROCEDURE**。

前提条件

您必须具有 CREATE PROCEDURE 或 CREATE ANY PROCEDURE 系统特权。

过程

输入 **CREATE PROCEDURE** 语句。

该示例使用 SAP Sybase IQ 演示数据库 iqdemo.db。

```
CREATE PROCEDURE new_dept (IN id INT,
                           IN name CHAR(35),
                           IN head_id INT)
BEGIN
    INSERT
        INTO GROUP0.departments (DepartmentID,
                                DepartmentName,
                                DepartmentHeadID)
        values (id, name, head_id);
END
```

注意： 使用 **CREATE PROCEDURE** 的 AT *location-string* SQL 语法来创建代理存储过程。

更改过程

您可以使用 SAP Control Center 或 Interactive SQL 修改现有过程。您必须具有 **ALTER ANY PROCEDURE** 系统特权或成为此过程的所有者。

您还可以使用 **ALTER PROCEDURE** 语句来修改过程。

调用过程

CALL 语句可调用过程。过程可由应用程序调用，也可以由其它过程调用。

删除过程

过程一旦创建即会一直保留在数据库中，直到有人显式将其删除。只有过程的所有者或具有 **DROP ANY PROCEDURE** 系统特权的用户才可以将该过程从数据库中删除。

执行 **DROP PROCEDURE** 语句以从数据库中删除该过程。

执行存储过程的特权

过程由创建它的用户拥有，即使没有特权，该用户也可以执行它。

注意： 上述信息不适用于系统过程。有关授予特权以运行系统过程的详细信息，请参见参考：构件块、表和过程 > “系统过程”。

使用 **GRANT EXECUTE** 命令可为其他用户授予执行过程的权限。例如，**new_dept** 过程的所有者可以通过以下语句允许 **another_user** 执行此过程：

```
GRANT EXECUTE ON new_dept TO another_user
```

要撤消执行此过程的特权，请执行：

```
REVOKE EXECUTE ON new_dept FROM another_user
```

在参数中返回过程结果

过程会以不同方式将结果返回到调用环境中。

- 单独的值作为 **OUT** 或 **INOUT** 参数返回。
- 作为结果集。
- 可以使用 **RETURN** 语句返回单个结果。

注意： 该示例使用 SAP Sybase IQ 演示数据库 **iqdemo.db**。

```
CREATE PROCEDURE SalaryList (IN department_id INT)
RESULT ( "Employee ID" INT, "Salary" NUMERIC(20,3) )
BEGIN
    SELECT EmployeeID, Salary
    FROM Employees
    WHERE Employees.DepartmentID = department_id;
END
```

在结果集中返回过程结果

除了在单独参数中将结果返回到调用环境之外，过程可以在结果集内返回信息。结果集通常是查询的结果。

如果某个过程在存储过程中动态创建临时表并随后选择该表，您必须使用 **EXECUTE IMMEDIATE WITH RESULT SET ON** 语法来避免出现未找到列错误。

例如：

```
CREATE PROCEDURE p1 (IN @t varchar(30))
BEGIN
    EXECUTE IMMEDIATE
    'SELECT * INTO #resultSet FROM ' || @t;
    EXECUTE IMMEDIATE WITH RESULT SET ON
    'SELECT * FROM #resultSet';
END
```

另请参见

- 过程结果（第 232 页）

显示过程信息

使用 **sp_iqprocedure** 显示数据库中有系统过程和用户定义过程的信息。

在 Interactive SQL 中运行 **sp_iqprocedure**。

显示有关用户定义过程 **sp_test** 的信息：

```
sp_iqprocedure sp_test
```

proc_name	proc_owner	proc_defn	replicate	srvid	remarks
sp_test	DBA	create procedure DBA.sp_test(in n1 integer) begin message 'sp_test' end	N	(NULL)	(NULL)

显示过程参数信息

使用 **sp_iqprocparam** 显示有关存储过程参数的信息。

在 Interactive SQL 中运行 **sp_iqprocparam**。

显示有关系统过程 **sp_iqshowcompression** 的参数的信息：

```
sp_iqprocparam sp_iqshowcompression, dbo, system
```

proc_name	proc_owner	param_name	param_type	param_mode
sp_iqshowcompression	dbo	@owner_name	normal	in
char	128	0	(NULL)	
sp_iqshowcompression	dbo	@table_name	normal	in

char	128	0	(NULL)			
sp_iqshowcompression		dbo		@column_name	normal	in
char	128	0	(NULL)			
sp_iqshowcompression		dbo		Column	result	out
char	128	0	(NULL)			
sp_iqshowcompression		dbo		Compression	result	out
char	3	0	(NULL)			

过程中的游标

如果查询或存储过程的结果集中有多行，可使用游标从中检索行，一次一行。
游标是查询或过程的句柄或标识符，也是结果集内的当前位置的句柄或标识符。

游标管理

管理游标类似于通过编程语言管理文件。游标的管理步骤如下：

- 1. 使用 **DECLARE** 语句为特殊 **SELECT** 语句或过程声明游标。
- 2. 使用 **OPEN** 语句打开游标。
- 3. 使用 **FETCH** 语句从游标一次检索一行结果。
- 4. 警告 [未找到行] 说明已到达结果集的结尾。
- 5. 使用 **CLOSE** 语句关闭游标。

缺省情况下，游标会在事务结尾处（**COMMIT** 或 **ROLLBACK** 语句上）自动关闭。
而使用 **WITH HOLD** 子句所打开的游标对随后的事务也会保持打开状态，直至被显式关闭。

游标定位

游标定位非常灵活。当游标打开之后，它位于第一行之前。您可以将游标移到以查询结果的开头或结尾作为参照的一个绝对位置，或者移到以当前游标位置作为参照的一个相对位置。如何更改游标位置以及可以执行什么操作的具体情况受编程接口约束。

在游标中可读取的行位置编号受整数的大小制约。您最多可以读取到第 2147483646 行，这个数字比可以在整数中保存的值小 1。在使用负数（从末尾开始计算行数）时，您最多可以读取到的行数比整数中可保存的最大负值大 1。

您可以使用特殊定位的更新操作和删除操作来更新或删除位于游标当前位置的行。如果游标定位在第一行之前或最后一行之后，则会返回错误，指示没有相应的游标行。

注意：对敏感性游标进行的插入和某些更新会导致游标定位发生问题。除非 **SELECT** 语句中有 **ORDER BY** 子句，否则 **SAP Sybase IQ** 不会将插入的行放在游标内可预知的位置。有时，插入的行要等到关闭并再次打开游标后才会出现。对于 **SAP Sybase IQ**，如果必须创建工作表才能打开游标，则会出现这种情况。

UPDATE 语句可能导致行在游标中移动。如果游标有使用现有索引的 **ORDER BY** 子句（不必创建工作表），会发生这种情况。使用 **STATIC SCROLL** 游标会缓解这些问题，但需要更多的内存和处理。

注意： SAP Sybase IQ 将 FIRST、LAST 和 ABSOLUTE 选项视为从结果集的开头开始。它将 RELATIVE 视为从当前位置开始的值为负的行计数。

SELECT 语句上的游标

以下过程在 SELECT 语句上使用游标。该过程基于 ListCustomerValue 过程中使用的同一查询，说明了存储过程语言的几个特征。

```
CREATE PROCEDURE TopCustomerValue(
    OUT TopCompany CHAR(36),
    OUT TopValue INT )
BEGIN
    -- 1. Declare the "row not found" exception
    DECLARE err_notfound
        EXCEPTION FOR SQLSTATE '02000';
    -- 2. Declare variables to hold
    --     each company name and its value
    DECLARE ThisName CHAR(36);
    DECLARE ThisValue INT;
    -- 3. Declare the cursor ThisCompany
    --     for the query
    DECLARE ThisCompany CURSOR FOR
    SELECT CompanyName,
        CAST( sum( SalesOrderItems.Quantity *
            Products.UnitPrice ) AS INTEGER )
        AS value
    FROM Customers
        INNER JOIN SalesOrders
        INNER JOIN SalesOrderItems
        INNER JOIN Products
    GROUP BY CompanyName;
    -- 4. Initialize the values of TopValue
    SET TopValue = 0;
    -- 5. Open the cursor
    OPEN ThisCompany;
    -- 6. Loop over the rows of the query
    CompanyLoop:
    LOOP
        FETCH NEXT ThisCompany
            INTO ThisName, ThisValue;
        IF SQLSTATE = err_notfound THEN
            LEAVE CompanyLoop;
        END IF;
        IF ThisValue > TopValue THEN
            SET TopCompany = ThisName;
            SET TopValue = ThisValue;
        END IF;
    END LOOP CompanyLoop;
    -- 7. Close the cursor
    CLOSE ThisCompany;
END;
```

注意

TopCustomerValue 过程具有以下显著的功能：

- 声明异常。在过程的后半期，此异常会在针对查询结果的循环结束时通知用户。
- 声明两个局部变量 **ThisName** 和 **ThisValue**，以保存来自该查询的每一行的结果。
- 声明了游标 **ThisCompany**。**SELECT** 语句会生成公司名称及公司所下订单总值的列表。
- **TopValue** 的值被设置为初始值 0，以便以后在循环中使用。
- **ThisCompany** 游标打开。
- **LOOP** 语句对该查询的每一行执行循环，依次将每一公司名称及订单值放置在变量 **ThisName** 和 **ThisValue** 中。如果 **ThisValue** 大于当前的最高值，则将 **TopCompany** 和 **TopValue** 重置为 **ThisName** 和 **ThisValue**。
- 该游标在过程结束后关闭。
- 还可以通过在 **SELECT** 语句中添加 "**ORDER BY value DESC**" 子句来编写不含循环语句的此过程。那样就只需获取游标的第一行。

TopCompanyValue 过程中的 **LOOP** 构造是标准形式，在最后一行处理完毕后退出现。您可以使用 **FOR** 循环，以更紧凑的形式重新编写此过程。**FOR** 语句将上述过程的若干方面合并到单个语句中。

```
CREATE PROCEDURE TopCustomerValue2(
    OUT TopCompany CHAR(36),
    OUT TopValue INT )
BEGIN
    -- 1. Initialize the TopValue variable
    SET TopValue = 0;
    -- 2. Do the For Loop
    FOR CompanyFor AS ThisCompany
        CURSOR FOR
            SELECT CompanyName AS ThisName,
                CAST( sum( SalesOrderItems.Quantity *
                    Products.UnitPrice ) AS INTEGER )
                AS ThisValue
            FROM Customers
                INNER JOIN SalesOrders
                INNER JOIN SalesOrderItems
                INNER JOIN Products
            GROUP BY ThisName
    DO
        IF ThisValue > TopValue THEN
            SET TopCompany = ThisName;
            SET TopValue = ThisValue;
        END IF;
    END FOR;
END;
```

使用 **IQ UTILITIES** 创建存储过程

SAP Sybase IQ 中提供的系统存储过程在 SQL 中实现。

您必须按照与系统存储过程完全相同的方式使用本地临时表和 **IQ UTILITIES** 语句：

过程的所有 **SQL** 代码均会被加密并编译到共享库 **libiqscripts16_r.so** 文件（在 **UNIX** 上）和 **iqscripts16.dll** 文件（在 **Windows** 上）中。

警告！ 无法正确使用语句可导致 IQ 服务器或数据库出现严重问题。

要查看存储过程代码，请在 Interactive SQL 中输入 `sp_helptext 'owner.procname'`。

IQ UTILITIES 的语法为：

```
IQ UTILITIES MAIN INTO local-temp-table-name arguments
```

您可能想要创建您自己的过程变体。例如：

- 创建一个调用系统存储过程的过程。
- 创建一个与系统存储过程无关但执行类似功能的过程。
- 创建一个与系统存储过程使用相同结构但提供更多功能的过程。例如，在前端工具或浏览器中以图形（而不是文本）方式显示过程结果。
- 如果选择上两个选项之一，请确保您了解 **IQ UTILITIES** 语句以及使用该语句所需满足的严格要求。

IQ UTILITIES 命令

IQ UTILITIES 是基础语句，它会在您运行大多数 IQ 系统过程时执行。在大多数情况下，用户并不知道 **IQ UTILITIES** 正在执行。用户仅在运行 IQ 缓冲区高速缓存监视器时才会直接发出 **IQ UTILITIES**。

IQ UTILITIES 提供一种系统方法来收集和报告 IQ 系统表中所维护的信息。没有通用的用户界面，您只能按照与现有系统过程相同的方法来使用 **IQ UTILITIES**。

系统过程声明用来存储信息的局部临时表。执行系统过程时，将进而执行 **IQ UTILITIES** 以从系统表中获取信息并将其存储在局部临时表中。系统过程可以只是报告局部临时表中的信息，也可以执行其它处理。

在某些系统过程中，**IQ UTILITIES** 语句包括一个预定义的数字作为它的某个参数。这个数字执行特定的函数，例如，从系统表中的信息派生值。

选择要调用的过程

您可以安全地使用 **IQ UTILITIES** 来创建您自己的已归档系统过程，并用其报告数据库中的信息。

例如，`sp_iqspaceused` 显示有关 IQ 主存储库和 IQ 临时存储库中已用空间和可用空间的信息。验证您基于系统存储过程所创建的过程是否具有正确的所有者。

请勿基于系统过程创建用来控制 IQ 操作的过程。修改用来控制 IQ 操作的过程可能会导致严重问题。

IQ UTILITIES 使用的数字

在 **IQ UTILITIES** 命令中，将数字用作参数。每个数字都与一个系统过程一起使用。

表 21. 系统过程中使用的 IQ UTILITIES 值

数字	过程	注释
10000	<code>sp_iqtransaction</code>	
20000	<code>sp_iqconnection</code> 和 <code>sp_iqmpxcountdbremote</code>	
30000	<code>sp_iqspaceused</code>	
40000	<code>sp_iqspaceinfo</code>	
50000	<code>sp_iqlocks</code>	
60000	<code>sp_iqmpxversionfetch</code>	不使用
70000	<code>sp_iqmpxdumpltvlog</code>	
80000	<code>sp_iqcontext</code>	
100000	<code>sp_iqindexfragmentation</code>	
110000	<code>sp_iqrowdensity</code>	

过程测试

为了维护 **IQ** 服务器和数据库的稳定性，请先在开发环境中对过程进行测试，然后再在生产环境中运行这些过程。

用户定义的函数

用户定义的函数是一种将单个值返回给调用环境的过程。

创建用户定义的函数

使用 **CREATE FUNCTION** 语句来创建用户定义的函数。

前提条件

- 必须具有 **CREATE PROCEDURE** 系统特权才能创建归您所有的函数。
- 必须具有 **CREATE ANY PROCEDURE** 或 **CREATE ANY OBJECT** 系统特权才能创建归其他人所有的函数。
- 如果此过程包含外部引用，则除以上系统特权外，还要必须具有 **CREATE EXTERNAL REFERENCE** 系统特权。

要在 SAP Control Center 中创建用户定义的函数，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

CREATE FUNCTION 语句

在数据库中创建用户定义的函数。通过指定所有者名称，可以为其他用户创建函数。根据权限，可通过与其它非集合函数完全相同的方法使用用户定义的函数。

快速链接：

转至参数 (第 218 页)

转至示例 (第 221 页)

转至用法 (第 222 页)

转至标准 (第 222 页)

转至权限 (第 222 页)

语法

语法 1

```
CREATE [ OR REPLACE ] [ TEMPORARY ] FUNCTION [ owner.]function-name
( [ parameter, ... ] )
[ SQL SECURITY { INVOKER | DEFINER } ]
RETURNS data-type ON EXCEPTION RESUME
| [ NOT ] DETERMINISTIC
{ compound-statement | AS tsq-compound-statement
| EXTERNAL NAME library-call
| EXTERNAL NAME java-call LANGUAGE JAVA }
```

语法 2

```
CREATE FUNCTION [ owner.]function-name ( [ parameter, ... ] )
RETURNS data-type
URL url-string
[ HEADER header-string ]
[ SOAPHEADER soap-header-string ]
[ TYPE { 'HTTP[:{ GET | POST } ] ' | 'SOAP[:{ RPC | DOC } ] ' } ]
[ NAMESPACE namespace-string ]
[ CERTIFICATE certificate-string ]
[ CLIENTPORT clientport-string ]
[ PROXY proxy-string ]
```

parameter - (back to Syntax 1) or (back to Syntax 2)
IN parameter-name data-type [DEFAULT expression]

tsq-compound-statement - (back to Syntax 1)
sql-statement
sql-statement ...

library-call - (back to Syntax 1)
'[operating-system:]function-name@library; ...'

```

operating-system - (back to library-call)
    UNIX

java-call - (back to Syntax 1)
    '[ package-name.]class-name.method-name method-signature '

method-signature - (back to java-call)
    ( [ field-descriptor, ... ] ) return-descriptor

field-descriptor and return-descriptor - (back to method-signature)
    Z | B | S | I | J | F | D | C | V | [ descriptor | L class-name;

url-string - (back to Syntax 2)
    ' { HTTP | HTTPS | HTTPS_FIPS } ://[user:password@]hostname[:port] [/
    path] '

```

参数

(返回顶部) (第 217 页)

- **CREATE [OR REPLACE]** - 参数名必须符合数据库标识符规则。它们必须具有有效的 SQL 数据类型，而且必须以关键字 **IN** 作为前缀，以表明参数是为函数提供值的表达式。

CREATE 子句将创建一个新函数，而 **OR REPLACE** 子句将替换同名的现有函数。替换函数时会更改函数的定义，但保留现有权限。不能将 **OR REPLACE** 子句与临时函数一起使用。

- **TEMPORARY** - 该函数仅对创建它的连接可见，并在删除该连接时随之自动删除。也可以显式删除临时函数。无法对临时函数执行 **ALTER**、**GRANT** 或 **REVOKE** 操作，而且与其它函数不同，临时函数不会被记录在目录或事务日志中。

具有临时函数创建者（当前用户）权限才能执行临时函数，并且临时函数只能由其创建者所有。因此，创建临时函数时无需指定所有者。临时函数可在连接到只读数据库时加以创建和删除。

- **SQL SECURITY** - 定义是作为 **INVOKER**（调用函数的用户）还是作为 **DEFINER**（拥有函数的用户）执行函数。缺省值为 **DEFINER**。

指定 **INVOKER** 后，必须对每个调用该过程的用户加以标注，因此会使用更多内存。此外，还将以调用者身份执行名称解析。因此，需确保用适合的所有者限定所有对象名称（表、过程等）。

- **data-type** - 不允许将 **LONG BINARY** 和 **LONG VARCHAR** 作为返回值数据类型。
- **compound-statement** - 一组用 **BEGIN** 和 **END** 括起来的 SQL 语句，中间用分号分隔。请参见 **BEGIN ... END** 语句。
- **tsql-compound-statement** - 一批 Transact-SQL 语句。

- **external-name** – 包含对外部库函数调用的包装，在 **RETURNS** 子句之后不含任何其它子句。库名可包含文件扩展名，在 **Windows** 中通常为 **.dll**，在 **UNIX** 中通常为 **.so**。在没有扩展名的情况下，该软件附加平台特定的缺省库文件扩展名。
临时函数不支持 **external-name** 子句。
- **LANGUAGE JAVA** – 关于 Java 方法的包装。有关调用 Java 过程的信息，请参见 **CREATE PROCEDURE** 语句。
- **ON EXCEPTION RESUME** – 使用类似于 **Transact-SQL** 的错误处理。请参见 **CREATE PROCEDURE** 语句。
- **[NOT] DETERMINISTIC** – 每次在查询中调用函数时都将重新求值。不是以这种方式指定的函数结果可以存入高速缓存以便提高性能，并且每次在查询求值过程中使用相同参数调用函数时，都会重新使用缓存的结果。

对于具有副作用（如修改基础数据）的函数，应将其声明为 **NOT DETERMINISTIC**。例如，应将生成主键值且用于 **INSERT ... SELECT** 语句的函数声明为 **NOT DETERMINISTIC**：

```
CREATE FUNCTION keygen( increment INTEGER )
RETURNS INTEGER
NOT DETERMINISTIC
BEGIN
    DECLARE keyval INTEGER;
    UPDATE counter SET x = x + increment;
    SELECT counter.x INTO keyval FROM counter;
    RETURN keyval
END
INSERT INTO new_table
SELECT keygen(1), ...
FROM old_table
```

如果函数对给定输入参数总是返回相同的值，则该函数可以声明为 **DETERMINISTIC**。除非所有用户定义的函数都声明为 **NOT DETERMINISTIC**，否则它们将被视为确定型函数。确定型函数为相同的参数返回一致的结果，并且没有副作用。即，数据库服务器假定对具有相同参数的同一函数连续进行两次调用将返回相同的结果，并且不会对查询的语义产生任何不良的副作用。

- **URL** – 仅在定义 **HTTP** 或 **SOAP Web** 服务客户端函数时使用。指定 **Web** 服务的 **URL**。其中的用户名和口令参数是可选的，它们提供了一种用于提供 **HTTP** 基本验证所需的证书的方法。**HTTP** 基本验证对用户和口令信息进行基于 64 位的编码，并将其传递到 **HTTP** 请求的“验证”标头中。

对于 **Web** 服务客户端函数来说，**SOAP** 和 **HTTP** 函数的返回类型必须为字符数据类型之一，如 **VARCHAR**。返回值是 **HTTP** 响应的主体。其中不包括 **HTTP** 标头信息。如果需要详细信息（例如状态信息），请使用过程而非函数。

参数值将作为请求的一部分进行传递。使用的语法取决于请求的类型。对于 **HTTP:GET**，参数将作为 **URL** 的一部分进行传递；对于 **HTTP:POST** 请求，则 values 放在请求主体中。**SOAP** 请求的参数总是被绑定在请求主体中。

- **HEADER** – 创建 HTTP Web 服务客户端函数时，此子句用于添加或修改 HTTP 请求标头条目。仅可为 HTTP 标头指定可打印 ASCII 字符，且这些字符不区分大小写。有关如何使用此子句的详细信息，请参见 CREATE PROCEDURE 语句的 HEADER 子句。
- **SOAPHEADER** – 当将 SOAP Web 服务声明为函数时，此子句用于指定一个或多个 SOAP 请求标头条目。SOAP 标头可声明为静态常量，也可使用参数替代机制动态设置（为参数 hd1、hd2 等声明 IN、OUT 或 INOUT 参数）。Web 服务函数可定义一个或多个 IN 模式替代参数，但无法定义 INOUT 或 OUT 替代参数。
- **TYPE** – 指定创建 Web 服务请求时使用的格式。如果指定 SOAP 或未包括类型子句，则使用缺省类型 SOAP:RPC。HTTP 隐含 HTTP:POST。由于始终将 SOAP 请求作为 XML 文档发送，所以 HTTP:POST 始终用于发送 SOAP 请求。
- **NAMESPACE** – 仅适用于 SOAP 客户端函数，并且标识 SOAP:RPC 和 SOAP:DOC 请求通常都需要的方法命名空间。处理请求的 SOAP 服务器使用此命名空间来解释 SOAP 请求消息主体中的实体名称。可以通过 Web 服务服务器，从 SOAP 服务的 WSDL 描述中获取命名空间。缺省值是过程的 URL，但是不包括可选的路径组件。
- **CERTIFICATE** – 要创建安全 (HTTPS) 请求，客户端必须有权访问 HTTPS 服务器所用的证书。必要的信息在一个用分号分隔的键/值对字符串中指定。证书可置于文件中且文件的名称通过 file 键提供，或者整个证书可置于字符串中，但不能同时出现这两种情况。可使用以下键：

键	缩写	描述
file		证书的文件名
certificate	cert	证书
company	co	证书中指定的公司
unit		证书中指定的公司单位
name		证书中指定的公用名

只有定向到 HTTPS 服务器或可以从非安全服务器重定向至安全服务器的请求才需要证书。

- **CLIENTPORT** – 标识 HTTP 客户端过程使用 TCP/IP 通信时所在的端口号。该子句是为通过防火墙的连接提供的，并建议只用于此类连接，因为防火墙按照 TCP/UDP 端口进行过滤。您可以指定单个端口号、端口号范围或是两者的组合；例如 CLIENTPORT '85,90-97'。
- **PROXY** – 指定代理服务器的 URI。在客户端必须通过代理访问网络时使用。指示过程将要连接到代理服务器，并通过它将请求发送到 Web 服务。

示例

(返回顶部) (第 217 页)

- **示例 1** - 将 `firstname` 字符串与 `lastname` 字符串连接在一起:

```
CREATE FUNCTION fullname (
    firstname CHAR(30),
    lastname CHAR(30) )
RETURNS CHAR(61)
BEGIN
    DECLARE name CHAR(61);
    SET name = firstname || ' ' || lastname;
    RETURN (name);
END
```

此示例说明了 **fullname** 函数的用法。

- 根据提供的两个字符串返回全名:

```
SELECT fullname ('joe','smith')
```

fullname('joe', 'smith')
joe smith

- 列出所有雇员的姓名:

```
SELECT fullname (givenname, surname)
FROM Employees
```

fullname (givenname, surname)
Fran Whitney
Matthew Cobb
Philip Chin
Julie Jordan
Robert Breault
...

- **示例 2** - 使用 Transact-SQL 语法:

```
CREATE FUNCTION DoubleIt ( @Input INT )
RETURNS INT
AS
DECLARE @Result INT
SELECT @Result = @Input * 2
RETURN @Result
```

语句 `SELECT DoubleIt(5)` 返回值 10。

- **示例 3** - 创建一个用 Java 编写的外部函数:

```
CREATE FUNCTION dba.encrypt( IN name char(254) )  
RETURNS VARCHAR  
EXTERNAL NAME  
'Scramble.encrypt (Ljava/lang/String;)Ljava/lang/String;'  
LANGUAGE JAVA
```

用法

(返回顶部) (第 217 页)

要修改用户定义的函数或通过加扰函数定义来隐藏函数的内容，请使用 **ALTER FUNCTION** 语句。

执行函数时，不必指定所有参数。如果在 **CREATE FUNCTION** 语句中提供了缺省值，则系统会为缺少的参数指派缺省值。如果调用程序既未提供参数又未设置缺省值，则会给出错误。

副作用

- 自动提交

标准

(返回顶部) (第 217 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。

权限

(返回顶部) (第 217 页)

如果希望函数归自己所有 – 需要 **CREATE PROCEDURE** 系统特权。

如果希望函数可由任何用户拥有 – 则需要以下特权之一：

- **CREATE ANY PROCEDURE** 系统特权。
- **CREATE ANY OBJECT** 系统特权。

要创建包含外部引用的函数，则无论用户是否为函数的所有者，另外都需要 **CREATE EXTERNAL REFERENCE** 系统特权。

调用用户定义的函数

在您要使用内置的非集合函数的相同位置，可根据权限使用用户定义的函数。

以下 **Interactive SQL** 语句从包含名字和姓氏的两列返回全名：

```
SELECT fullname (GivenName, LastName)  
FROM Employees;
```

fullname (Employees.GivenName,Employees.LastName)
Fran Whitney
Matthew Cobb
Philip Chin
...

以下语句从提供的名字和姓氏返回全名：

```
SELECT fullname ('Jane', 'Smith');
```

fullname ('Jane', 'Smith')
Jane Smith

已被授予函数的执行权限的任何用户都可以使用 *fullname* 函数。

删除用户定义的函数 (SQL)

用户定义的函数保留在数据库中，直到它们被显式删除为止。

前提条件

您必须是用户定义函数的所有者，或者拥有以下其中一种系统特权：

```
DROP ANY PROCEDURE
DROP ANY OBJECT
```

过程

1. 连接到数据库。
2. 执行与以下语句类似的 **DROP FUNCTION** 语句：

```
DROP FUNCTION function-name;
```

用户定义的函数已删除。

执行用户定义的函数的权限

用户定义的函数由创建它的用户拥有，即使没有特权，该用户也可以执行它。

用户定义的函数的所有者可以使用 **GRANT EXECUTE** 命令向其他用户授予特权。

例如，*fullname* 函数的创建者可以通过以下语句允许 *another_user* 使用函数 *fullname*：

```
GRANT EXECUTE ON fullname TO another_user
```

以下语句将撤消使用该函数的特权：

```
REVOKE EXECUTE ON fullname FROM another_user
```

请参见《管理：用户管理和安全》>“安全管理”>“特权”>“对象级特权”>“授予和撤消对象级特权”>“授予对函数和过程的 EXECUTE 特权”。

授予执行用户定义函数的能力 (SQL)

通过授予 EXECUTE 对象级特权的方式来授予执行用户定义函数的能力。

前提条件

您必须是用户定义函数的所有者，或者拥有函数的 EXECUTE 特权及管理权限。

用户定义函数的所有权属于创建该函数的用户，因此该用户执行此函数时无需任何特权。

过程

您已创建一个函数，并且您希望其他用户能够使用该函数。

1. 连接到数据库。
2. 执行与以下语句类似的 GRANT EXECUTE 语句：

```
GRANT EXECUTE ON function-name TO user-id;
```

被授予者现在可以执行该过程。

下一步

无。

批处理

批处理是一起提交并作为一个组依次执行的 SQL 语句组。过程中使用的控制语句 (CASE、IF、LOOP 等) 也可以在批处理中使用。如果批处理由用 BEGIN/END 括起来的复合语句组成，则它还可以包含主机变量以及变量、游标、临时表或异常的本地声明。允许在批处理中引用主机变量，但具有以下限制：

- 批处理中只有一个语句可以引用主机变量
- 使用主机变量的语句之前不能出现返回结果集的语句

正在使用批处理时，建议使用 BEGIN/END 进行清晰地指示。

批处理内的语句可以使用分号分隔，这意味着该批处理符合 Watcom SQL 方言。不使用分号分隔语句的多语句批处理将符合 Transact-SQL 方言。批处理的方言决定在批处理中允许使用哪些语句，还决定如何处理批处理中错误。

在许多方面，批处理类似于存储过程；但是，存在一些差异：

- 批处理没有名称
- 批处理不接收参数
- 批处理不是永久存储在数据库中
- 批处理不能由不同的连接共享

简单的批处理由一组不带分隔符的 **SQL** 语句组成，后面跟有只含 **go** 一词的单独一行。以下示例会创建 **Eastern Sales** 部门并将所有销售代表从麻萨诸塞州转移到该部门。以下是一个 **Transact-SQL** 批处理示例。

```
INSERT
INTO Departments ( DepartmentID, DepartmentName )
VALUES ( 220, 'Eastern Sales' )

UPDATE Employees
SET DepartmentID = 220
WHERE DepartmentID = 200
AND State = 'MA'

COMMIT
go
```

go 一词由 **Interactive SQL** 识别，让其将前面的语句作为单个批处理发送到服务器。

以下示例尽管看起来与上面的很相似，但其实 **Interactive SQL** 会以截然不同的方式进行处理。此示例不使用 **Transact-SQL** 方言。每个语句以分号分隔。**Interactive SQL** 将每个以分号分隔的语句分别发送到服务器。它不会被视为批处理。

```
INSERT
INTO Departments ( DepartmentID, DepartmentName )
VALUES ( 220, 'Eastern Sales' );

UPDATE Employees
SET DepartmentID = 220
WHERE DepartmentID = 200
AND State = 'MA';

COMMIT;
```

要使 **Interactive SQL** 将其视为批处理，可将其改为使用 **BEGIN ... END** 的复合语句。以下是前一个示例的修订版。复合语句中的三个语句会作为批处理发送给服务器。

```
BEGIN
    INSERT
    INTO Departments ( DepartmentID, DepartmentName )
    VALUES ( 220, 'Eastern Sales' );

    UPDATE Employees
    SET DepartmentID = 220
    WHERE DepartmentID = 200
    AND State = 'MA';

    COMMIT;
END
```

在这个特殊的示例中，不论服务器执行批处理还是单个语句，最终结果都没有区别。但也存在结果不一样的情况。请看以下示例。

```
DECLARE @CurrentID INTEGER;
SET @CurrentID = 207;
SELECT Surname FROM Employees
WHERE EmployeeID=@CurrentID;
```

如果使用 **Interactive SQL** 执行此示例，则数据库服务器会返回一个错误，指示无法找到变量。发生这种情况是因为 **Interactive SQL** 将三个语句分别发送到服务器。它们不是作为批处理来执行。正如您已经看到的那样，这一问题的解决方法是使用复合语句来强制 **Interactive SQL** 将这些语句作为批处理发送给服务器。以下示例实现了这一目的。

```
BEGIN
  DECLARE @CurrentID INTEGER;
  SET @CurrentID = 207;
  SELECT Surname FROM Employees
  WHERE EmployeeID=@CurrentID;
END
```

用 **BEGIN** 和 **END** 将一组语句括起来可强制 **Interactive SQL** 将它们视作批处理。

IF 语句是复合语句的另一个例子。**Interactive SQL** 将以下语句作为单个批处理发送给服务器。

```
IF EXISTS ( SELECT *
            FROM SYSTAB
            WHERE table_name='Employees' )
THEN
  SELECT Surname AS LastName,
         GivenName AS FirstName
  FROM Employees;
  SELECT Surname, GivenName
  FROM Customers;
  SELECT Surname, GivenName
  FROM Contacts;
ELSE
  MESSAGE 'The Employees table does not exist'
  TO CLIENT;
END IF
```

使用其它技术准备和执行 SQL 语句时不会发生这种情况。例如，使用 **ODBC** 的应用程序可以将一系列用分号分隔的语句当作批处理来准备和执行。

对服务器混用 **Interactive SQL** 语句和 **SQL** 语句时必须小心。以下的示例说明混用 **Interactive SQL** 语句和 **SQL** 语句会引发问题。在此示例中，由于 **Interactive SQL** **OUTPUT** 语句嵌入到复合语句中，因此它与所有其它语句一起作为批处理发送给服务器，结果导致语法错误。

```
IF EXISTS ( SELECT *
            FROM SYSTAB
            WHERE table_name='Employees' )
THEN
  SELECT Surname AS LastName,
```

```

        GivenName AS FirstName
    FROM Employees;
    SELECT Surname, GivenName
    FROM Customers;
    SELECT Surname, GivenName
    FROM Contacts;
    OUTPUT TO 'c:\\temp\\query.txt';
ELSE
    MESSAGE 'The Employees table does not exist'
    TO CLIENT;
END IF

```

OUTPUT 语句的正确用法如下所示。

```

IF EXISTS (    SELECT *
              FROM SYSTAB
              WHERE table_name='Employees' )
THEN
    SELECT    Surname AS LastName,
              GivenName AS FirstName
    FROM Employees;
    SELECT Surname, GivenName
    FROM Customers;
    SELECT Surname, GivenName
    FROM Contacts;
ELSE
    MESSAGE 'The Employees table does not exist'
    TO CLIENT;
END IF;
OUTPUT TO 'c:\\temp\\query.txt';

```

控制语句

一些控制语句可用在过程、触发器或用户定义函数的主体或用在批处理中，以实现逻辑流和决策。可用的控制语句包括：

控制语句	语法
复合语句	<pre> BEGIN [ATOMIC] Statement-list END </pre>
条件执行: IF	<pre> IF condition THEN Statement-list ELSEIF condition THEN Statement-list ELSE Statement-list END IF </pre>

控制语句	语法
条件执行: CASE	<pre> CASE expression WHEN value THEN Statement-list WHEN value THEN Statement-list ELSE Statement-list END CASE </pre>
重复: WHILE、LOOP	<pre> WHILE condition LOOP Statement-list END LOOP </pre>
重复: FOR 游标循环	<pre> FOR loop-name AS cursor-name CURSOR FOR select-statement DO Statement-list END FOR </pre>
中断: LEAVE	<pre> LEAVE label </pre>
CALL	<pre> CALL procname(arg, ...) </pre>

复合语句

复合语句以关键字 **BEGIN** 起始，以关键字 **END** 结束。过程或触发器的主体是**复合语句**。复合语句还可以用于批处理。复合语句可以嵌套，可以与其它控制语句组合在一起，以定义过程和触发器中或批处理中的执行流。

复合语句允许将一组 **SQL** 语句组合在一起作为一个单元处理。复合语句内的 **SQL** 语句以分号分隔。

复合语句中的声明

复合语句中的局部声明紧随在 **BEGIN** 关键字之后。这些局部声明只存在于复合语句内。在复合语句内，您可以声明：

- 变量
- 游标
- 临时表
- 异常（错误标识符）

局部声明可由该复合语句中的任何语句引用，或者可由该复合语句内嵌套的任何复合语句中的任何语句引用。局部声明对于从复合语句调用的其它过程而言是不可见的。

原子复合语句

原子语句是或者完全执行，或者根本不执行的语句。例如，更新数千行的 **UPDATE** 语句在更新许多行后可能会遇到错误。如果该语句没有完成，则所有更改的行都恢复为其初始状态。**UPDATE** 语句即是原子语句。

所有非复合 **SQL** 语句都是原子语句。可以通过在 **BEGIN** 关键字后添加 **ATOMIC** 关键字来使复合语句成为原子语句。

```
BEGIN ATOMIC
    UPDATE Employees
    SET ManagerID = 501
    WHERE EmployeeID = 467;
    UPDATE Employees
    SET BirthDate = 'bad_data';
END
```

在此示例中，两个更新语句都属于原子复合语句。它们要么成功，要么失败，二者只能居其一。第一个更新语句将是成功的。第二个语句会引发数据转换错误，因为指派给 **BirthDate** 列的值无法转换为日期。

该原子复合语句会失败，因此两个 **UPDATE** 语句的结果都会被撤消。即使当前执行的事务最终被提交，原子复合语句中的这两个语句也都不会生效。

如果原子复合语句成功，则在复合语句中进行的更改直到当前执行的事务被提交的时候才会生效。如果原子复合语句成功而包含该语句的事务却回退，则原子复合语句也回退。在原子复合语句开始时建立一个保存点。语句中的任何错误都会导致回退到该保存点。

在自动提交（非链接）模式下执行原子复合语句时，直到完成语句执行后提交模式才变为到手动（链接）。在手动模式中，在原子复合语句中执行的 **DML** 语句不会引起快速 **COMMIT** 或 **ROLLBACK**。如果原子复合语句成功完成，将执行 **COMMIT** 语句；否则执行 **ROLLBACK** 语句。

在原子复合语句中不能使用 **COMMIT** 和 **ROLLBACK** 以及某些 **ROLLBACK TO SAVEPOINT** 语句。

过程的结构

过程的主体由复合语句组成。

复合语句由 **BEGIN** 和 **END** 组成，它们将一组 **SQL** 语句括起来，用分号分隔每一语句。

过程中允许的 **SQL** 语句

您可以在过程内使用几乎所有的 **SQL** 语句。

- **SELECT**、**UPDATE**、**DELETE**、**INSERT** 以及 **SET VARIABLE**

- 用来执行其它过程的 **CALL** 语句
- 控制语句
- 游标语句
- 异常处理语句
- **EXECUTE IMMEDIATE** 语句

不能在过程中使用的部分 SQL 语句：

- **CONNECT** 语句
- **DISCONNECT** 语句

可以在过程中使用 **COMMIT**、**ROLLBACK** 和 **SAVEPOINT** 语句，但有一些限制。

请参见参考：语句和选项 > “SQL 语句” 中每个语句的 “用法” 部分。

过程的参数声明

过程参数在 **CREATE PROCEDURE** 语句中以列表形式出现。参数名必须符合其它数据库标识符（如列名）的规则。它们必须具有有效数据类型，并且可以用关键字 **IN**、**OUT** 或 **INOUT** 作为前缀。缺省情况下，参数是 **INOUT** 参数。这些关键字具有以下含义：

- **IN** - 此参数是一个为过程提供值的表达式。
- **OUT** - 此参数是一个可由过程为其赋值的变量。
- **INOUT** - 此参数是一个为过程提供值的变量，并且可以由过程为其赋予新值。

可以在 **CREATE PROCEDURE** 语句中将缺省值指派给过程参数。缺省值必须是常量，可以是空值。例如，以下过程对一个 **IN** 参数使用 **NULL** 缺省值，来避免执行没有任何意义的查询：

```
CREATE PROCEDURE CustomerProducts (
    IN customer_ID
        INTEGER DEFAULT NULL )
RESULT ( product_ID INTEGER,
        quantity_ordered INTEGER )
BEGIN
    IF customer_ID IS NULL THEN
        RETURN;
    ELSE
        SELECT      Products.ID,
                    sum( SalesOrderItems.Quantity )
        FROM        Products,
                    SalesOrderItems,
                    SalesOrders
        WHERE SalesOrders.CustomerID = customer_ID
        AND SalesOrders.ID = SalesOrderItems.ID
        AND SalesOrderItems.ProductID = Products.ID
        GROUP BY Products.ID;
    END IF;
END;
```

以下语句指派 **DEFAULT NULL**，当遇到该值时，过程会返回而不执行查询。

```
CALL CustomerProducts();
```

将参数传递给过程的方法

您可以通过 **CALL** 语句的两种形式之一利用存储过程参数的缺省值的优越性。

位于 **CREATE PROCEDURE** 语句参数列表末尾的可选参数可以在 **CALL** 语句中省略。例如，假定有以下具有三个 **INOUT** 参数的过程：

```
CREATE PROCEDURE SampleProcedure(
    INOUT var1 INT DEFAULT 1,
    INOUT var2 int DEFAULT 2,
    INOUT var3 int DEFAULT 3 )
...
```

此示例假定调用环境已设置了三个变量来保存传递给过程的值：

```
CREATE VARIABLE V1 INT;
CREATE VARIABLE V2 INT;
CREATE VARIABLE V3 INT;
```

可以调用过程 **SampleProcedure**，仅提供第一个参数，如下所示。在这种情况下，参数 **var2** 和 **var3** 使用缺省值。

```
CALL SampleProcedure( V1 );
```

也可以通过使用第一个参数的 **DEFAULT** 值，并指定第二个参数的方法调用这个过程。如下所示：

```
CALL SampleProcedure( DEFAULT, V2 );
```

以可选参数调用过程的更灵活方法是按名称传递这些参数。可按如下所示调用 **SampleProcedure** 过程：

```
CALL SampleProcedure( var1 = V1, var3 = V3 );
```

或按如下所示调用该过程：

```
CALL SampleProcedure( var3 = V3, var1 = V1 );
```

如何将参数传递给函数

用户定义的函数不通过 **CALL** 语句进行调用，而是采用与内置函数同样的方式使用。例如，以下语句使用 **FullName** 函数来检索雇员的姓名：

注意

- 在调用函数中可以使用缺省参数。但不能按名称将参数传递给函数。
- 参数是按值而不是按引用传递的。即使函数更改了参数的值，这一更改也不会返回给调用环境。
- 不能在用户定义的函数中使用输出参数。
- 用户定义的函数不能返回结果集。

示例：列出所有雇员的姓名

在 **Interactive SQL** 中，执行以下查询：

```
SELECT FullName( GivenName, Surname ) AS Name
FROM Employees;
```

将显示以下结果：

Name
Fran Whitney
Matthew Cobb
Philip Chin
Julie Jordan
...

过程结果

过程能够返回单行或多行数据。

由单行数据组成的结果可作为参数传递回过程。由多行数据组成的结果作为结果集传递回过程。过程也可以返回在 **RETURN** 语句中给定的单个值。

使用 **RETURN** 语句返回值

RETURN 语句会将单个整数值返回给调用环境，并导致程序立即从过程退出。

前提条件

执行此任务没有前提条件。

过程

1. 执行以下语句：

```
RETURN expression
```

2. 所提供表达式的值被返回给调用环境。使用 **CALL** 语句的扩展将返回值保存在变量中：

```
CREATE VARIABLE returnval INTEGER;
returnval = CALL variable/procedure-name? myproc();
```

返回一个值并将其保存到变量。

将结果作为过程参数返回的方法

过程可以将结果作为作用于该过程的参数返回到调用环境中。在过程内，可以采用以下方式对参数和变量指派值：

- **SET 语句**

以下过程使用 **SET** 语句将值返回到指派的 **OUT** 参数中。您必须具有 **CREATE PROCEDURE** 系统特权才能执行以下语句：

```
CREATE PROCEDURE greater(
    IN a INT,
    IN b INT,
    OUT c INT )
BEGIN
    IF a > b THEN
        SET c = a;
    ELSE
        SET c = b;
    END IF ;
END;
```

- 具有 **INTO** 子句的 **SELECT** 语句

单行查询最多从数据库中检索一行。此类型的查询将 **SELECT** 语句与 **INTO** 子句一起使用。**INTO** 子句跟随在 **SELECT** 列表之后，位于 **FROM** 子句之前。它包含一组变量，用于为各个 **SELECT** 列表项检索值。变量的数目必须与 **SELECT** 列表项的数目相同。

在执行 **SELECT** 语句时，数据库服务器会检索 **SELECT** 语句的结果并将这些结果放置在变量中。如果查询结果包含多个行，则数据库服务器会返回一个错误。对于返回多行的查询，必须使用游标。

如果查询结果中没有选定的行，则不更新变量并将返回警告。

您必须具有对象的相应 **SELECT** 特权才能执行 **SELECT** 语句。

示例 1：创建一个过程并使用 **SELECT...INTO** 语句选择其结果

1. 启动 **Interactive SQL** 并连接到 **SAP Sybase IQ** 示例数据库。您必须拥有 **CREATE PROCEDURE** 系统特权，以及 **Employees** 表的 **SELECT** 特权或 **SELECT ANY TABLE** 系统特权这两种特权中的一种。
2. 在“**SQL 语句**”窗格中，执行下列语句来创建一个过程 (**AverageSalary**)，用于将雇员的平均薪水作为 **OUT** 参数返回：

```
CREATE PROCEDURE AverageSalary( OUT average_salary NUMERIC(20,3) )
BEGIN
    SELECT AVG( Salary )
    INTO average_salary
    FROM GROUPO.Employees;
END;
```

3. 创建用于保存过程输出值的变量。在此例中，输出变量是具有三位小数的数值。

```
CREATE VARIABLE Average NUMERIC(20,3);
```

4. 使用所创建的用来保存结果的变量调用过程：

```
CALL AverageSalary( Average );
```

5. 如果该过程已正确创建并正常运行，Interactive SQL “消息” 选项卡不会显示任何错误。

6. 要检查该变量的值，请执行以下语句：

```
SELECT Average;
```

7. 查看输出变量 **Average** 的值。“结果” 窗格的 “结果” 选项卡会显示此变量的值 49988.623，这是雇员的平均薪水。

示例 2：返回单行 **SELECT** 语句的结果

1. 启动 Interactive SQL 并连接到 SAP Sybase IQ 示例数据库。您必须拥有 **CREATE PROCEDURE** 系统特权，以及 **Customers** 表的 **SELECT** 特权或 **SELECT ANY TABLE** 系统特权这两种特权中的一种。
2. 执行以下语句以返回给定客户所下订单的数目：

```
CREATE PROCEDURE OrderCount (
    IN customer_ID INT,
    OUT Orders INT )
BEGIN
    SELECT COUNT(SalesOrders.ID)
        INTO Orders
    FROM GROUPO.Customers
        KEY LEFT OUTER JOIN SalesOrders
    WHERE Customers.ID = customer_ID;
END;
```

3. 使用以下语句测试此过程，将会显示 ID 为 102 的客户所下订单的数目：

```
CREATE VARIABLE orders INT;
CALL OrderCount ( 102, orders );
SELECT orders;
```

示例 2 的说明 -

- **customer_ID** 参数被声明为 **IN** 参数。此参数会保存传递给过程的客户 ID。
- **Orders** 参数被声明为 **OUT** 参数。用于保存返回给调用环境的订单数变量的值。
- **Orders** 变量不需要 **DECLARE** 语句，因为它是在过程参数列表中声明的。
- **SELECT** 语句返回单个行并将其放置在变量 **Orders** 中。

从过程中以结果集形式返回的信息

除了在单独的参数中将结果返回给调用环境外，过程还可以以结果集的形式返回信息。结果集通常是查询的结果。

RESULT 子句中变量的数目必须与 **SELECT** 列表项的数目相匹配。如果数据类型不匹配，会尽可能执行自动的数据类型转换。

RESULT 子句是 **CREATE PROCEDURE** 语句的一部分，没有语句分隔符。

SELECT 列表项的名称不必与 **RESULT** 子句中的那些名称相匹配。

要在视图上修改过程结果集，用户必须对基础表具有适当的特权。

如果存储过程或用户定义的函数返回一个结果，则不能同时支持输出参数或返回值。

缺省情况下，**Interactive SQL** 仅显示第一个结果集。要使过程在 **Interactive SQL** 中返回多行结果，可在“选项”窗口的“结果”选项卡上设置“显示多个结果集”选项。

示例 1

以下过程会返回已下订单的客户及其所下订单总值的列表。

在 **Interactive SQL** 中执行以下语句：

```
CREATE PROCEDURE ListCustomerValue()
RESULT ( "Company" CHAR(36), "Value" INT )
BEGIN
    SELECT CompanyName,
           CAST( SUM( SalesOrderItems.Quantity *
                     Products.UnitPrice )
                AS INTEGER ) AS value
    FROM Customers
       INNER JOIN SalesOrders
       INNER JOIN SalesOrderItems
       INNER JOIN Products
    GROUP BY CompanyName
    ORDER BY value DESC;
END;
```

执行 CALL ListCustomerValue (); 可返回以下结果集：

Company	Value
The Hat Company	5016
The Igloo	3564
The Ultimate	3348
North Land Trading	3144
Molly's	2808
...	...

示例 2

以下过程返回包含指定部门中每个雇员的薪水的结果集。在 **Interactive SQL** 中执行以下语句：

```
CREATE PROCEDURE SalaryList( IN department_id INT )
RESULT ( "Employee ID" INT, Salary NUMERIC(20,3) )
BEGIN
    SELECT EmployeeID, Salary
    FROM Employees
    WHERE Employees.DepartmentID = department_id;
END;
```

RESULT 子句中的姓名与查询的结果对应，用作所显示结果中的列标题。

要列出研发部门（部门 ID 为 100）中雇员的薪水，请执行下列语句：

```
CALL SalaryList( 100 );
```

以下结果集将出现在“结果”窗格中：

Employee ID	Salary
102	45700.000
105	62000.000
160	57490.000
243	72995.000
...	...

返回多个结果集

可使用 Interactive SQL 从一个过程返回多个结果集。

前提条件

执行此任务没有前提条件。

过程

缺省情况下，Interactive SQL 不显示多个结果集。

1. 在 Interactive SQL 中，连接到数据库。
2. 单击“工具”»“选项”。
3. 单击“SAP Sybase IQ”。
4. 在“结果”选项卡上，单击“显示所有结果集”。
5. 单击“确定”。

您启用此选项后，Interactive SQL 将显示多个结果集。该设置将立即生效，并会在将来的会话中仍然有效，直到其被禁用时为止。

下一步

如果在过程定义中包括 **RESULT** 子句，结果集必须是兼容的：项目数必须与 **SELECT** 列表中的项目数相同，数据类型必须全都可以自动转换为 **RESULT** 子句中所列的数据类型。

如果省略 **RESULT** 子句，则过程返回的各结果集之间所返回的列的数目和类型可以不同。

过程的可变结果集

RESULT 子句在过程中是可选的。通过省略 **RESULT** 子句，可以编写返回不同结果集的过程，结果集中可以有不同数目或类型的列，具体情况取决于过程的执行方式。

如果不使用可变结果集功能，则出于性能方面的原因，应使用 **RESULT** 子句。

例如，如果输入变量为 **Y**，以下过程返回两列，否则只返回一列。

```
CREATE PROCEDURE Names( IN formal char(1) )
BEGIN
    IF formal = 'y' THEN
        SELECT Surname, GivenName
        FROM Employees
    ELSE
        SELECT GivenName
        FROM Employees
    END IF
END;
```

过程中可变结果集的使用受到某些限制，具体限制取决于客户端应用程序所使用的接口。

- **嵌入式 SQL** - 要获取正确形式的结果集，必须在打开用于结果集的游标之后、但在返回任何行之前对过程调用执行 **DESCRIBE**。

创建不带 **RESULT** 子句的过程并且过程返回可变结果集时，引用该过程的 **SELECT** 语句的 **DESCRIBE** 可能失败。为防止 **DESCRIBE** 失败，建议在 **SELECT** 语句的 **FROM** 子句中包括 **WITH** 子句。或者，也可在 **DESCRIBE** 语句中使用 **WITH VARIABLE RESULT** 子句。您可使用 **WITH VARIABLE RESULT** 子句来确定每次执行 **OPEN** 语句后是否应描述过程调用。

- **ODBC** - ODBC 应用程序可以使用可变结果集过程。SAP Sybase IQ ODBC 驱动程序会对可变结果集进行适当的描述。
- **Open Client 应用程序** - Open Client 应用程序可以使用可变结果集过程。SAP Sybase IQ 会对可变结果集进行适当的描述。

错误和警告处理

应用程序在执行某个 **SQL** 语句后，可以检查**状态代码**。此状态码（即返回代码）指示执行的语句是成功还是失败，并给出失败的原因。可以使用相同的机制来指示对过程执行的 **CALL** 语句是成功还是失败。

错误报告使用 **SQLCODE** 或 **SQLSTATE** 状态说明。

只要执行 **SQL** 语句，就会在称作 **SQLSTATE** 和 **SQLCODE** 的特殊过程变量中出现值。特殊值指示在执行语句时是否遇到了任何意外情况。可以在执行 **SQL** 语句后通过 **IF** 语句来检查 **SQLSTATE** 或 **SQLCODE** 的值，并根据语句的成功与否采取措施。

例如，可使用 **SQLSTATE** 变量来指示是否成功地获取了某行。**TopCustomerValue** 过程使用 **SQLSTATE** 测试来检测是否已经处理了 **SELECT** 语句的所有行。

缺省的错误处理

本节介绍 **SAP Sybase IQ** 在您没有在过程中内置错误处理方式的情况下会如何处理过程执行期间所发生的错误。

要改变行为方式，可以使用异常处理程序。

警告与错误的处理方式稍有不同。

有两种无需使用显式错误处理即可处理错误的方法：

- **缺省的错误处理** – 过程或触发器失败，并且将错误代码返回到调用环境。
- **ON EXCEPTION RESUME** – 如果 **ON EXCEPTION RESUME** 子句出现在 **CREATE PROCEDURE** 语句中，则过程将在出现错误后继续执行，在导致错误的语句之后的下一语句恢复。

使用 **ON EXCEPTION RESUME** 的过程的精确行为由 **on_tsql_error** 选项设置来规定。

缺省的错误处理

通常，如果过程或触发器中的 **SQL** 语句失败，该过程或触发器会停止执行，并将控制权及对 **SQLSTATE** 值和 **SQLCODE** 值的适当设置返回给应用程序。即使错误发生在从第一个过程或触发器直接或间接调用的过程或触发器中，上述情况也同样适用。对于触发器，导致该触发器触发的操作也将撤消，错误被返回给应用程序。

以下演示过程说明了在发生下面情况时的结果：应用程序调用过程 **OuterProc**，**OuterProc** 又调用过程 **InnerProc**，然后遇到错误。

```
CREATE PROCEDURE OuterProc()
BEGIN
    MESSAGE 'Hello from OuterProc.' TO CLIENT;
    CALL InnerProc();
    MESSAGE 'SQLSTATE set to ',
        SQLSTATE, ' in OuterProc.' TO CLIENT
END;
CREATE PROCEDURE InnerProc()
BEGIN
    DECLARE column_not_found
        EXCEPTION FOR SQLSTATE '52003';
    MESSAGE 'Hello from InnerProc.' TO CLIENT;
    SIGNAL column_not_found;
    MESSAGE 'SQLSTATE set to ',
        SQLSTATE, ' in InnerProc.' TO CLIENT;
END;

CALL OuterProc();
```

Interactive SQL [消息] 选项卡会显示以下内容：

```
Hello from OuterProc.
Hello from InnerProc.
```

InnerProc 中的 DECLARE 语句为与服务器已知的错误条件相关联的某一预定义的 SQLSTATE 值声明符号名称。

MESSAGE 语句将消息发送到 Interactive SQL [消息] 选项卡。

SIGNAL 语句从 InnerProc 过程内生成错误条件。

InnerProc 中跟随在 SIGNAL 语句之后的任何语句都不会执行：InnerProc 会立即将控制权传递回调用环境，在本例中调用环境即为 OuterProc 过程。OuterProc 中没有任何跟随在 CALL 语句之后的语句执行。错误条件返回到调用环境中，并在那里进行处理。例如，Interactive SQL 通过显示描述该错误的消息窗口来处理该错误。

TRACEBACK 函数会提供在发生错误时正执行的语句的列表。可以输入以下语句以从 Interactive SQL 使用 TRACEBACK 函数：

```
SELECT TRACEBACK();
```

使用 ON EXCEPTION RESUME 处理错误

如果 ON EXCEPTION RESUME 子句出现在 CREATE PROCEDURE 语句中，则该过程会在发生错误时检查随后的语句。如果该语句处理错误，则过程继续执行，在导致错误的语句之后的下一语句恢复。当发生错误时，它不会将控制权返回给调用环境。

可以通过 on_tsq_error 选项设置修改使用 ON EXCEPTION RESUME 的过程的行为。

错误处理语句包括以下这些：

```
IF
SELECT @variable =
CASE
LOOP
LEAVE
CONTINUE
CALL
EXECUTE
SIGNAL
RESIGNAL
DECLARE
SET VARIABLE
```

以下演示过程说明了在发生下面情况时的结果：应用程序调用过程 OuterProc，OuterProc 又调用过程 InnerProc，然后遇到错误。这些演示过程基于在本节前面部分中所使用的过程：

```
DROP PROCEDURE OuterProc;
DROP PROCEDURE InnerProc;

CREATE PROCEDURE OuterProc()
ON EXCEPTION RESUME
BEGIN
```

```
DECLARE res CHAR(5);
MESSAGE 'Hello from OuterProc.' TO CLIENT;
CALL InnerProc();
SET res=SQLSTATE;
IF res='52003' THEN
    MESSAGE 'SQLSTATE set to ',
        res, ' in OuterProc.' TO CLIENT;
END IF
END;

CREATE PROCEDURE InnerProc()
ON EXCEPTION RESUME
BEGIN
    DECLARE column_not_found
    EXCEPTION FOR SQLSTATE '52003';
    MESSAGE 'Hello from InnerProc.' TO CLIENT;
    SIGNAL column_not_found;
    MESSAGE 'SQLSTATE set to ',
        SQLSTATE, ' in InnerProc.' TO CLIENT;
END;

CALL OuterProc();
```

Interactive SQL [消息] 选项卡随后会显示以下内容:

```
Hello from OuterProc.
Hello from InnerProc.
SQLSTATE set to 52003 in OuterProc.
```

使用的执行路径如下所示:

1. **OuterProc** 执行并调用 **InnerProc**。
2. 在 **InnerProc** 中, **SIGNAL** 语句指示一个错误。
3. **MESSAGE** 语句不是错误处理语句, 因此控制将被传递回 **OuterProc** 并且不显示该消息。
4. 在 **OuterProc** 中, 跟随在错误后的语句将 **SQLSTATE** 值指派给名为 **res** 的变量。这是错误处理语句, 因此过程的执行会继续并显示 **OuterProc** 消息。

缺省的警告处理

错误和警告的处理方式各不相同。对错误的缺省操作是为 **SQLSTATE** 和 **SQLCODE** 变量设置值, 并在遇到错误时将控制权返回给调用环境; 而对警告的缺省操作是设置 **SQLSTATE** 和 **SQLCODE** 值, 并继续执行过程。

以下演示过程说明缺省情况下警告的处理方式。

在本例中, **SIGNAL** 语句生成一个指明找不到行的条件。这是警告, 不是错误。

```
DROP PROCEDURE OuterProc;
DROP PROCEDURE InnerProc;

CREATE PROCEDURE OuterProc()
BEGIN
    MESSAGE 'Hello from OuterProc.' TO CLIENT;
    CALL InnerProc();
```



```

        MESSAGE 'SQLSTATE set to ',
            SQLSTATE, ' in OuterProc.' TO CLIENT;
END;
CREATE PROCEDURE InnerProc()
BEGIN
    DECLARE row not found
        EXCEPTION FOR SQLSTATE '02000';
    MESSAGE 'Hello from InnerProc.' TO CLIENT;
    SIGNAL row_not_found;
    MESSAGE 'SQLSTATE set to ',
        SQLSTATE, ' in InnerProc.' TO CLIENT;
END;

CALL OuterProc();

```

Interactive SQL [消息] 选项卡随后会显示以下内容:

```

Hello from OuterProc.
Hello from InnerProc.
SQLSTATE set to 02000 in InnerProc.
SQLSTATE set to 00000 in OuterProc.

```

通过由警告 (02000) 设置的 SQLSTATE, 这两个过程都在生成警告后继续执行。

执行 InnerProc 中的第二个 MESSAGE 语句时会重置该警告。任何 SQL 语句的成功执行都会将 SQLSTATE 重置为 00000, 将 SQLCODE 重置为 0。如果某一过程需要保存错误状态, 它必须在引发错误或警告的语句执行后立即指派状态值。

异常处理程序

对于某些类型的错误, 通常需要进行拦截并在过程或触发器内处理, 而不是将其传递回调用环境。这是通过使用**异常处理程序**完成的。

使用复合语句的 EXCEPTION 部分来定义异常处理程序。

只要在复合语句中发生错误, 异常处理程序就会执行。与错误不同, 警告不会导致执行异常处理代码。如果错误出现在嵌套的复合语句中, 或者出现在复合语句内任何地方所调用的过程或触发器中, 异常处理代码也会执行。

中断错误 SQL_INTERRUPT、SQLSTATE 57014 的异常处理程序应只包含不可中断的语句, 如 ROLLBACK 和 ROLLBACK TO SAVEPOINT 等。如果异常处理程序包含连接中断时调用的可中断语句, 数据库服务器将在第一个可中断语句处停止异常处理程序, 并返回中断错误。

异常处理程序可以使用 SQLSTATE 或 SQLCODE 特殊值以确定语句失败的原因。也可以使用 ERRORMSG 函数 (无需参数) 返回与 SQLSTATE 相关的错误状况。只有每个 WHEN 子句中的第一个语句可以指定此信息, 并且该语句不得为复合语句。

在本例中, 附加代码会在 InnerProc 过程中处理有关未找到的列的错误。

```

DROP PROCEDURE OuterProc;
DROP PROCEDURE InnerProc;

CREATE PROCEDURE OuterProc()

```

```
BEGIN
  MESSAGE 'Hello from OuterProc.' TO CLIENT;
  CALL InnerProc();
  MESSAGE 'SQLSTATE set to ',
    SQLSTATE, ' in OuterProc.' TO CLIENT
END;
CREATE PROCEDURE InnerProc()
BEGIN
  DECLARE column_not_found
  EXCEPTION FOR SQLSTATE '52003';
  MESSAGE 'Hello from InnerProc.' TO CLIENT;
  SIGNAL column_not_found;
  MESSAGE 'Line following SIGNAL.' TO CLIENT;
  EXCEPTION
    WHEN column_not_found THEN
      MESSAGE 'Column not found handling.' TO CLIENT;
    WHEN OTHERS THEN
      RESIGNAL ;
END;
CALL OuterProc();
```

Interactive SQL “消息” 选项卡随后会显示以下内容：

```
Hello from OuterProc.
Hello from InnerProc.
Column not found handling.
SQLSTATE set to 00000 in OuterProc.
```

EXCEPTION 子句声明异常处理程序。**EXCEPTION** 后的各行直到错误发生时才会执行。每一 **WHEN** 子句都指定一个异常名称（用 **DECLARE** 语句声明）以及在遇到该异常时要执行的语句。**WHEN OTHERS THEN** 子句指定当所发生的异常没有出现在之前的 **WHEN** 子句时要执行的语句。

在上述示例中，**RESIGNAL** 语句将该异常传递给更高级别的异常处理程序。如果在异常处理程序中不指定 **WHEN OTHERS THEN**，则 **RESIGNAL** 是缺省操作。

附加说明

- 在 **InnerProc** 中，执行 **EXCEPTION** 处理程序，而不是执行 **SIGNAL** 语句之后的那些行。
- 因为遇到的错误是有关未找到的列的错误，因此执行其中用来处理该错误的 **MESSAGE** 语句，**SQLSTATE** 重置为零（指示没有发生任何错误）。
- 在异常处理代码执行后，控制被传递回 **OuterProc**，就像没有遇到任何错误一样继续执行。
- 不应将 **ON EXCEPTION RESUME** 与显式异常处理一起使用。如果包括 **ON EXCEPTION RESUME**，将不会执行异常处理代码。
- 如果该错误的处理代码是一个 **RESIGNAL** 语句，控制权会返回 **OuterProc** 过程，**SQLSTATE** 仍然设置为值 52003。这就像是 **InnerProc** 中没有错误处理代码。由于在 **OuterProc** 中没有错误处理代码，因此该过程将失败。

异常处理和原子复合语句

如果错误出现在原子复合语句中，并且该语句具有可处理错误的异常处理程序，则在该复合语句完成时不会有活动的异常，并且在出现异常之前所做的更改也不会被反转。如果异常处理程序不处理错误或导致另一错误（包括通过 **RESIGNAL**），则在原子复合语句内所做的更改会被撤消。

嵌套的复合语句和异常处理程序

只有 **ON EXCEPTION RESUME** 子句出现在过程定义中，跟随在导致出现错误的语句之后的代码才会执行。

您可以使用嵌套的复合语句来增强相关控制，以控制哪些语句在出现错误之后仍执行，哪些语句不执行。

以下示例说明如何使用嵌套的复合语句来控制程序流。

```
DROP PROCEDURE OuterProc;
DROP PROCEDURE InnerProc;

CREATE PROCEDURE InnerProc()
BEGIN
    BEGIN
        DECLARE column_not_found
        EXCEPTION FOR SQLSTATE VALUE '52003';
        MESSAGE 'Hello from InnerProc' TO CLIENT;
        SIGNAL column_not_found;
        MESSAGE 'Line following SIGNAL' TO CLIENT
        EXCEPTION
            WHEN column_not_found THEN
                MESSAGE 'Column not found handling' TO
                CLIENT;
            WHEN OTHERS THEN
                RESIGNAL;
    END;
    MESSAGE 'Outer compound statement' TO CLIENT;
END;

CALL InnerProc();
```

Interactive SQL “消息” 选项卡随后会显示以下内容：

```
Hello from InnerProc
Column not found handling
Outer compound statement
```

在遇到导致错误的 **SIGNAL** 语句时，控制权会传递给复合语句的异常处理程序，并输出 [Column not found handling] 消息。控制权然后传递回外部复合语句，并输出 [Outer compound statement] 消息。

如果在内部复合语句中遇到了 [未找到列] (**SQLSTATE**) 以外的其它错误，则异常处理程序会执行 **RESIGNAL** 语句。**RESIGNAL** 语句将控制权直接传递回调用环境，不再执行其余的外部复合语句。

示例

以下示例显示的是使用 **RESIGNAL** 输出 **sa_error_stack_trace** 系统过程:

```
CREATE PROCEDURE DBA.error_reporting_procedure()
BEGIN
    SELECT *
    FROM sa_error_stack_trace();
END;

CREATE PROCEDURE DBA.proc1()
BEGIN TRY
    BEGIN TRY
        DECLARE v INTEGER = 0;
        SET v = 1 / v;
    END TRY
    BEGIN CATCH
        CALL DBA.proc2();
    END CATCH
END TRY
BEGIN CATCH
    CALL DBA.error_reporting_procedure();
END CATCH;

CREATE PROCEDURE DBA.proc2()
BEGIN
    CALL DBA.proc3();
END;

CREATE PROCEDURE DBA.proc3()
BEGIN
    RESIGNAL;
END;
```

在使用 **CALL proc1()** 调用上述过程时, 将产生以下结果集:

StackLevel	UserName	ProcName	LineNumber	IsResignal
1	DBA	proc1	8	0
2	DBA	proc2	3	0
3	DBA	proc3	3	1
4	DBA	proc1	5	0

以下示例显示的是使用 **RESIGNAL** 和 **BEGIN** 语句输出 **sa_error_stack_trace** 系统过程:

```
CREATE PROCEDURE DBA.error_reporting_procedure()
BEGIN
    SELECT *
    FROM sa_error_stack_trace();
END;

CREATE PROCEDURE DBA.proc1()
```

```
BEGIN
  BEGIN
    DECLARE v INTEGER = 0;
    SET v = 1 / v;
    EXCEPTION WHEN OTHERS THEN
      CALL DBA.proc2();
    END
  END
  EXCEPTION WHEN OTHERS THEN
    CALL DBA.error_reporting_procedure();
  END;

  CREATE PROCEDURE DBA.proc2()
  BEGIN
    CALL DBA.proc3();
  END;

  CREATE PROCEDURE DBA.proc3()
  BEGIN
    RESIGNAL;
  END;
```

在使用 CALL proc1() 调用上述过程时，将产生以下结果集：

StackLevel	UserName	ProcName	LineNumber	IsResignal
1	DBA	proc1	8	0
2	DBA	proc2	3	0
3	DBA	proc3	3	1
4	DBA	proc1	5	0

过程中的事务和保存点

过程中的 SQL 语句是当前事务的一部分。可以在一个事务内调用若干过程，或在一个过程中具有若干事务。

不允许将 COMMIT 和 ROLLBACK 用在任何原子语句中。

保存点可在过程内使用，但 ROLLBACK TO SAVEPOINT 语句永远不可以引用在原子操作开始前的保存点。此外，在完成原子操作后，释放该原子操作内的所有保存点。

隐藏过程、函数、触发器、事件或视图的内容

要分发应用程序和数据库而不透露过程、函数、触发器、事件和视图中包含的逻辑，可使用 ALTER PROCEDURE、ALTER FUNCTION、ALTER TRIGGER、ALTER EVENT 和 ALTER VIEW 语句的 SET HIDDEN 子句来隐藏这些对象的内容。

前提条件

您必须是对象的所有者，具有 ALTER ANY OBJECT 系统特权或者具有以下其中一种特权：

过程和函数 – ALTER ANY PROCEDURE 系统特权

视图 – ALTER ANY VIEW 系统特权

事件 – MANAGE ANY EVENT 系统特权

触发器 –

ALTER ANY TRIGGER 系统特权

基础表的 ALTER 特权和 CREATE ANY OBJECT 系统特权

对于视图上的触发器，您必须拥有 ALTER ANY TRIGGER 和 ALTER ANY VIEW 系统特权

过程

SET HIDDEN 子句会对关联对象的内容进行模糊处理，使之难以理解，但这些对象仍然可以使用。还可以卸载对象并将其重新装入其它数据库。

修改是无法撤消的，会删除对象的原始文本。因此需要在数据库外保留对象的原始数据源。

注意： 将 preserve_source_format 数据库选项设置为 On 时，数据库服务器会保存来自过程、视图、触发器和事件的 CREATE 和 ALTER 语句的格式化源代码，并将其置于相应系统视图的源列。在本例中，对象定义和源代码定义同时隐藏。

但是，将 preserve_source_format 数据库选项设置为 On 并不会阻止 SET HIDDEN 子句删除对象的原始源代码定义。

使用带有 SET HIDDEN 子句的相应 ALTER 语句。

选项	操作
隐藏单个对象	执行带有 SET HIDDEN 子句的相应 ALTER 语句，以隐藏单个过程、函数、触发器、事件或视图。

选项	操作
隐藏特定类型的所有对象	循环执行带有 SET HIDDEN 子句的相应 ALTER 语句，以隐藏所有过程、函数、触发器、事件或视图。

执行自动提交。对象定义不再可见。隐藏后的对象仍可直接引用，也仍可在查询处理中使用。

过程、触发器、事件和批处理中允许使用的语句

除了以下语句，所有 **SQL** 语句在批处理中都是可接受的：

ALTER DATABASE（语法 3 和 4）
CONNECT
CREATE DATABASE
CREATE DECRYPTED FILE
CREATE ENCRYPTED FILE
DISCONNECT
DROP CONNECTION
DROP DATABASE
FORWARD TO
Interactive **SQL** 语句，例如 **INPUT** 或 **OUTPUT**
PREPARE TO COMMIT
STOP SERVER

可以在过程、触发器、事件和批处理中使用 **COMMIT**、**ROLLBACK** 和 **SAVEPOINT** 语句，但有一些限制。

批处理中使用的 **SELECT** 语句

可以在批处理中包括一个或多个 **SELECT** 语句。例如：

```
IF EXISTS( SELECT *
            FROM SYSTAB
            WHERE table_name='Employees' )
THEN
    SELECT    Surname AS LastName,
             GivenName AS FirstName
    FROM Employees;
    SELECT Surname, GivenName
    FROM Customers;
    SELECT Surname, GivenName
    FROM Contacts;
END IF;
```

只需要在第一个 **SELECT** 语句中指定结果集的别名，因为服务器使用批处理中的第一个 **SELECT** 语句来描述结果集。

在每一查询后需要使用 **RESUME** 语句，以检索下一个结果集。

过程、触发器、用户定义函数和批处理中使用的 **EXECUTE IMMEDIATE**

EXECUTE IMMEDIATE 语句允许使用文字字符串（在引号中）和变量的组合来构建语句。例如，以下过程包含创建表的 **EXECUTE IMMEDIATE** 语句。

```
CREATE PROCEDURE CreateTableProcedure(  
    IN tablename CHAR(128) )  
BEGIN  
    EXECUTE IMMEDIATE 'CREATE TABLE '  
        || tablename  
        || '( column1 INT PRIMARY KEY )'  
END;
```

EXECUTE IMMEDIATE 语句可以与返回结果集的查询一起使用。您可将 **WITH RESULT SET ON** 子句与 **EXECUTE IMMEDIATE** 语句配合使用，以指示语句返回结果集—缺省行为是语句不返回结果集。指定 **WITH RESULT SET ON** 或 **WITH RESULT SET OFF** 同时影响创建过程时和执行过程时所发生的事件。

请考虑以下过程：

```
CREATE OR REPLACE PROCEDURE test_result_clause()  
BEGIN  
    EXECUTE IMMEDIATE WITH RESULT SET OFF 'SELECT 1';  
END;
```

当过程定义中不包括 **RESULT SET** 子句时，数据库服务器将试图确定过程是否生成结果集。在此处，**EXECUTE IMMEDIATE** 语句指定不生成结果集。因此，数据库服务器定义的过程没有结果集列，并且此过程的 **SYSPROCPARM** 系统视图中也没有任何行。对此过程的 **CALL** 执行 **DESCRIBE** 将不会返回任何结果列。如果嵌入式 **SQL** 应用程序使用该信息来决定是否要打开游标或执行语句，则其会执行语句，然后返回错误。

作为第二个示例，请考虑上述过程的修改版本：

```
CREATE OR REPLACE PROCEDURE test_result_clause()  
BEGIN  
    EXECUTE IMMEDIATE WITH RESULT SET ON 'SELECT 1';  
END;
```

在此处，**WITH RESULT SET ON** 子句将致使此过程的 **SYSPROCPARM** 系统视图中存在一个行。数据库服务器不知道结果集的外观—因为过程正在使用 **EXECUTE IMMEDIATE**—但其知道过程将返回结果集，所以数据库服务器在 **SYSPROCPARM** 中定义一个虚设的结果集列来表示过程的结果集（其名称为 "expression"，类型为 **SMALLINT**）。仅创建一个虚设的结果集列；使用 **EXECUTE IMMEDIATE** 语句时，服务器无法确定各结果集的列的数目和类型。因此，请考虑以下稍做修改的示例：

```
CREATE OR REPLACE PROCEDURE test_result_clause()  
BEGIN  
    EXECUTE IMMEDIATE WITH RESULT SET ON 'SELECT 1, 2, 3';  
END;
```


在此，虽然 **SELECT** 会返回含有三个列的结果集，但服务器仍然仅在 **SYSPROCPARM** 系统视图中放置一行。因此，以下查询

```
SELECT * FROM test_result_clause();
```

失败并生成 **SQLCODE -866**，因为运行时的结果集特性与 **SYSPROCPARM** 中的占位符结果不匹配。

要执行上述查询，您可以显式指定结果集列的名称和类型，如下所示：

```
SELECT * FROM test_result_clause() WITH (x INTEGER, y INTEGER, z
INTEGER);
```

在执行时，如果指定 **WITH RESULT SET ON**，数据库服务器处理返回结果集的 **EXECUTE IMMEDIATE** 语句。但是，如果指定 **WITH RESULT SET OFF** 或省略该子句，则数据库服务器仍会在已分析的字符串参数中查看第一个语句的类型。如果该语句是 **SELECT** 语句，则其会返回结果集。因此，在上面的第二个示例中：

```
CREATE OR REPLACE PROCEDURE test_result_clause()
BEGIN
    EXECUTE IMMEDIATE WITH RESULT SET OFF 'SELECT 1';
END;
```

从 **Interactive SQL** 中可以成功地调用此过程。但是，如果更改此过程以使其包含批处理，而不是仅包含一个 **SELECT** 语句：

```
CREATE OR REPLACE PROCEDURE test_result_clause()
BEGIN
    EXECUTE IMMEDIATE WITH RESULT SET OFF
    'begin declare v int; set v=1; select v; end';
END;
```

则对 **test_result_clause** 过程的 **CALL** 将会返回错误 (**SQLCODE -946**、**SQLSTATE 09W03**)。

最后一个示例说明您如何将 **SELECT** 语句构造为某个过程中的 **EXECUTE IMMEDIATE** 语句的参数，并使该过程返回结果集。

```
CREATE PROCEDURE DynamicResult(
    IN Columns LONG VARCHAR,
    IN TableName CHAR(128),
    IN Restriction LONG VARCHAR DEFAULT NULL )
BEGIN
    DECLARE Command LONG VARCHAR;
    SET Command = 'SELECT ' || Columns || ' FROM ' || TableName;
    IF ISNULL( Restriction, '' ) <> '' THEN
        SET Command = Command || ' WHERE ' || Restriction;
    END IF;
    EXECUTE IMMEDIATE WITH RESULT SET ON Command;
END;
```

如果按如下方式调用该过程：

```
CALL DynamicResult(
    'table_id,table_name',
    'SYSTAB',
    'table_id <= 10');
```

它将生成以下结果：

table_id	table_name
1	ISYSTAB
2	ISYSTABCOL
3	ISYSIDX
...	...

上述 **CALL** 将正确返回结果集，即使过程利用 **EXECUTE IMMEDIATE** 也是如此。一些服务器 API（例如 ODBC）将请求与 **PREPARE-DESCRIBE-EXECUTE-OR-OPEN** 结合使用，从而根据其是否返回结果集来决定是执行语句还是打开语句。如果打开语句，API 或应用程序可随即发出 **DESCRIBE CURSOR** 来确定实际结果集的外观，而不是依赖于创建过程时 **SYSPROCPARM** 系统视图的内容。**DBISQL** 和 **DBISQLC** 均使用此技术。在上述情况下，对以上过程的 **CALL** 将会正确执行。但是，依赖于语句的 **DESCRIBE** 结果的应用程序接口将无法处理任意语句。

在原子复合语句中，不能使用导致 **COMMIT**（提交）的 **EXECUTE IMMEDIATE** 语句，因为在该上下文中不允许 **COMMIT**。

使用调度和事件自动完成任务

使用调度和事件处理功能可自动完成数据库管理及其它任务。

使用调度和事件自动完成任务

很多数据库管理任务最好是有计划有步骤地完成。例如，定期备份过程是适当的数据库管理过程的重要组成部分。

在 SAP Sybase IQ 中，可以通过将某个**事件**添加到数据库中并提供该事件的调度，来自动完成例行的任务。每当调度中的某个时间经过时，数据库服务器都会运行被称为**事件处理程序**的一系列操作。

数据库管理还包括在出现某些情况时采取相应的操作。例如，当事务日志所在的磁盘快写满时，可能应该用电子邮件通知系统管理员，以便管理员进行处理。通过为一组**系统事件**中的某个事件定义事件处理程序，也可以自动完成这些任务。

事件

在 SAP Sybase IQ 中，可以通过将某个事件添加到数据库中并提供该事件的调度，来自动完成例行的任务。**SAP Sybase IQ** 支持以下类型的事件：

- **调度事件** - 具有相关联的调度并在指定时间执行。
- **系统事件** - 与由数据库服务器跟踪的特殊条件类型关联。
- **手动事件** - 通过使用 **TRIGGER EVENT** 语句显式地触发。
- **用户跟踪事件** - 用于将有关应用程序的信息记录到事件跟踪会话。这些事件对数据库的所有连接均可见。

每次执行事件处理程序后，如果没有出现错误便会发生 **COMMIT**。如果出现错误，则会发生 **ROLLBACK**。

调度

通过对活动进行调度，可以确保在一组预置的时间执行一组操作。调度信息和事件处理程序都存储在数据库本身中。

可以通过将多个调度与一个指定的事件相关联，来定义复杂的调度，但通常并不必要这样做。例如，一家零售店可能希望某事件在营业时间每小时发生一次，而营业时间根据一周中的不同日期而变化。通过定义各自拥有自己的调度的多个事件，以及调用一个公用存储过程，您可以达到同样的效果。

对事件进行调度时，可以使用完整的英文日期名称（**Monday**、**Tuesday** 等），也可以使用日期的简写形式（**Mon**、**Tue** 等）。如果要想以英文以外的其它语言运行的服务器能够识别日期名称，您必须使用完整的英文日期名称。

以下示例可以帮助您初步了解可能有用的调度操作。

示例

在每个工作日结束时汇总订单。此示例使用 `iqdemo.db`。

```
CREATE TABLE OrderSummary(c1 date, c2 int);
CREATE EVENT SummarizeSchedule
START TIME '6:00 pm' on ('Mon', 'Tue', 'Wed', 'Thu', 'Fri')
HANDLER
BEGIN
    INSERT INTO DBA.OrderSummary
        SELECT MAX(OrderDate),
        COUNT(*)
    FROM GROUP0.SalesOrders
    WHERE OrderDate = current date
END;
```

调度定义

为灵活起见，调度定义具有下列几个组成部分：

- **名称** – 每个调度定义都有一个名称。您可以将多个调度指派给某个特定事件，这在设计复杂调度时非常有用。
- **开始时间** – 您可以为事件定义开始时间，即开始执行事件的时间。
- **范围** – 如果不指定开始时间，可以指定事件处于活动状态的时间范围。事件在指定的开始和结束日期范围内发生。频率由指定的周期决定。
- **定期** – 每个调度都可以反复循环。事件是按一定频率触发的，可以按某些日期的点钟、分钟或秒指定频率，而这些日期可以指定为星期几或几号。反复出现的事件包含一个 `EVERY` 或 `ON` 子句。

系统事件

SAP Sybase IQ 跟踪着多个系统事件。每个系统事件都提供一个您可以在其上挂接一组操作的挂钩。数据库服务器为您跟踪事件，并在系统事件满足提供的**触发条件**时执行操作（操作在事件处理程序中定义）。

通过将事件处理程序定义为在发生选定系统事件类型且该事件类型符合定义的触发条件时执行，可以提高数据的安全性并有助于简化管理。如果在执行期间未检测到任何错误，则将提交事件处理程序的操作。如果检测到错误，将回退。

系统事件类型

- **BackupEnd** – 您可以使用 `BackupEnd` 事件类型在备份结束时执行操作。
- **连接事件** – 在建立连接时 (`Connect`) 或连接尝试失败时 (`ConnectFailed`)。为安全起见，您可能希望使用这些事件。作为连接事件处理程序的替代方法，可能要考虑使用登录过程。
- **DatabaseStart** – 您可以使用 `DatabaseStart` 事件类型在启动数据库时执行操作。
- **Deadlock** – 您可以使用 `Deadlock` 事件在发生死锁时执行操作。事件处理程序可以使用 `sa_report_deadlocks` 过程以获得有关导致死锁的条件的信息。使用 `Deadlock` 事件时，应配置数据库服务器以便通过将 `log_deadlocks` 选项设置为 `On` 来捕获死

锁信息，并且通过使用 `sa_server_option` 或 `-zl` 服务器选项来启用 `RememberLastStatement` 功能。

连接死锁和线程死锁触发 `Deadlock` 事件。`Deadlock` 事件仅能提供可通过 `sa_report_deadlocks` 系统过程获得的信息。但是，通过使用此事件可及时地对死锁发生作用。由于数据库服务器能够保留的与死锁相关的信息量是有限的，所以快速响应会非常重要。

- **Disconnect** – 您可以使用 `Disconnect` 事件在用户或应用程序断开连接时执行操作。
- **可用磁盘空间** – 跟踪数据库文件 (`DBDiskSpace`)、日志文件 (`LogDiskSpace`) 或临时文件 (`TempDiskSpace`) 所在设备的可用磁盘空间。此系统事件在 `Windows Mobile` 上不可用。

在磁盘空间不足时，您可能希望使用磁盘空间事件来提醒管理员。

您可以在启动数据库服务器时指定 `-fc` 选项，以在数据库服务器出现文件系统已满时执行回调函数。

- **文件大小** – 文件达到指定的大小。可以将此事件用于数据库文件 (`GrowDB`)、事务日志 (`GrowLog`) 或临时文件 (`GrowTemp`)。

您可能希望使用文件大小事件来跟踪对数据库的异常操作或监控批量操作。

- **GlobalAutoincrement** – 当用 `GLOBAL AUTOINCREMENT` 定义的列的剩余值数目不足其范围的百分之一时，就会触发 `GlobalAutoincrement` 事件。基于表和提供给此事件作为参数的剩余值数目，可使用此事件为 `global_database_id` 选项请求新值。要获得事件中表的其余值，请使用具有 `RemainingValues` 参数和 `TableName` 参数的 `EVENT_PARAMETER` 函数。`RemainingValues` 将会返回可为该列生成的剩余值的数量，而 `TableName` 则会返回含有 `GLOBAL AUTOINCREMENT` 列（此列接近其范围末尾）的表。
- **RAISERROR 错误** – 当执行一个 `RAISERROR` 语句时，您可以使用 `RAISERROR` 事件类型来执行操作。通过使用 `EVENT_CONDITION` 函数，可在事件处理程序中确定在 `RAISERROR` 语句中使用的错误号（例如，`EVENT_CONDITION('ErrorNumber')`）。
- **空闲时间** – 数据库服务器已经空闲了指定的时间 (`ServerIdle`)。您可能希望使用此事件类型在安静期间执行例行的维护操作。

事件的触发条件

每个事件定义都有一个与其关联的系统事件。它也有一个或多个触发条件。在满足系统事件的触发条件时，会触发事件处理程序。

触发条件包括在 `CREATE EVENT` 语句的 `WHERE` 子句中，可以使用 `AND` 关键字对触发条件进行组合。每个触发条件的形式如下：

```
event_condition( condition-name ) comparison-operator value
```

`condition-name` 参数是一组预置字符串中的一个字符串，这些字符串适用于不同的事件类型。例如，您可以使用 `DBSize`（以兆字节为单位的数据库文件大小）来构建适

合于 GrowDB 系统事件的触发条件。数据库服务器不检查条件名是否与事件类型匹配：您应负责确保该条件在事件类型上下文中有意义。

示例

- 通知管理员可能有人试图入侵数据库。此示例使用 `iqdemo.db`：

```
create event SecurityCheck
type ConnectFailed
handler
begin
  declare num_failures int;
  declare mins int;
  insert into FailedConnections( log_time )
  values ( current timestamp );
  select count( * ) into num_failures
  from FailedConnections
  where log_time >= dateadd(minute, -5, current timestamp );
  if( num_failures >= 3 ) then
    select datediff( minute, last_notification, current timestamp )
    into mins from Notification;
    if( mins > 30 ) then
      update Notification
      set last_notification = current timestamp;
    end if
  end if
end;
```

事件处理程序

事件处理程序与触发了该事件的操作在不同的连接上执行，因此不与客户端应用程序进行交互。它们是使用事件创建者的特权执行的。

不管是用于调度事件的处理，还是用于系统事件处理，事件处理程序都包含复合语句，而且在很多方面都与存储过程类似。您可以添加循环、条件执行等。

每次执行事件处理程序后，如果没有出现错误便会发生 **COMMIT**。如果出现错误，则会发生 **ROLLBACK**。

事件处理程序的上下文信息

与储存过程不同，事件处理程序不带任何参数。可使用 **EVENT_PARAMETER** 函数访问在其中触发了事件的上下文的相关信息。返回的信息包括导致事件触发的连接 ID 和用户 ID、事件名以及该事件的执行次数。

测试事件处理程序

在开发过程中，您希望在方便时触发事件处理程序。即使在没有出现触发条件或调度时间时，您也可以使用 **TRIGGER EVENT** 语句显式导致事件的执行。但是，**TRIGGER EVENT** 不会导致被禁用的事件处理程序执行。

尽管在生产数据库上开发事件处理程序不是很好的做法，但您可以显式使用 **ALTER EVENT** 语句来禁用事件处理程序。

代码共享

用同一组操作来处理多个事件是很有用的。例如，如果数据库文件或日志文件所在设备上磁盘空间有限，您可能需要执行通知操作。为此，创建一个存储过程并在每个事件处理程序的正文中调用它，从而将任何所需的上下文信息作为参数传递给该过程。

调试事件处理程序

调试事件处理程序与调试存储过程非常类似。事件处理程序出现在事件列表中。

隐藏事件处理程序

可以使用含有 **SET HIDDEN** 子句的 **ALTER EVENT** 语句来隐藏事件处理程序的定义。指定 **SET HIDDEN** 子句的结果是永久模糊处理存储在 **ISYSEVENT** 系统表的操作列表中的事件处理程序定义。

限制活动事件

通过使用含有 **NumActive** 上下文名称的 **EVENT_PARAMETER** 函数，您还可以确定某个特定事件处理程序有多少个实例当前处于活动状态。如果要限制事件处理程序以便在给定的时间只执行一个实例，则此函数是很有用的。

数据库服务器如何检查系统事件

系统事件是按照其**事件类型**进行分类的，事件类型在 **CREATE EVENT** 语句中指定。有两种事件类型：

- **活动的事件类型** - 有些事件类型是数据库服务器本身所执行的操作的结果。这些活动的事件类型包括增长的数据库文件或不同的数据库操作或 **RAISERROR** 的开始和结束（**BackupEnd** 等）。

当数据库服务器采取操作时，它会检查是否满足在 **WHERE** 子句中定义的触发条件，如果满足条件，则触发为该事件类型定义的任何事件。

- **轮询的事件类型** - 有些事件类型（例如，**DBDiskSpace** 等可用磁盘空间类型以及 **IdleTime** 类型）并不仅仅是由数据库操作触发的。

对于这些事件类型，数据库服务器每隔三十秒轮询一次，大约在启动数据库服务器三十秒后开始进行轮询。

对于 **IdleTime** 事件类型，数据库服务器会检查服务器是否已经空闲了整三十秒。如果没有启动请求且当前没有请求处于活动状态，则服务器会将空闲检查间隔时间（以秒为单位）与总空闲时间累加；否则，将总空闲时间重置为0。因此 **IdleTime** 的值始终是三十秒的倍数。当 **IdleTime** 大于在触发条件中指定的间隔时，会触发与 **IdleTime** 关联的事件处理程序。

数据库服务器如何检查调度事件

在数据库服务器启动时以及每当调度的事件处理程序完成时，都会进行调度的事件时间的计算。

根据在调度定义中指定的增量，将增量与上一开始时间累加来计算下一个调度时间。如果执行事件处理程序花费的时间比指定的增量大，导致下一时间早于当前时间，则数据库服务器会按指定的增量计算下一个调度时间，直到下一个调度时间是将来的时间。

例如，其执行需要六十五分钟且被请求在 9:00 和 5:00 之间每小时运行一次的事件处理程序，将每隔两小时运行一次，即在 9:00、11:00、1:00 等时间运行。

为运行一个在 9:00 和 5:00 之间工作并在下一次执行之前延迟一段时间的进程，您可以将处理程序定义为在其完成时间超过之前不断循环，并且每次迭代之间都有一个 **WAITFOR** 语句。

如果您正在间歇地运行数据库服务器，而且不是在调度时间运行，则在启动时事件处理程序不会运行。而是会在启动时计算下一个调度时间。例如，如果您将一个备份预定在每天凌晨一点发生，但在每个工作日结束时都会关闭数据库服务器，则该备份永远也不会发生。

如果一个事件的下一调度执行时间是在一小时以后，则数据库服务器将以小时为基础重新计算它的下一个调度时间。这允许事件在系统时钟调整（系统时钟更改为 **Daylight Saving Time** 或者由 **Daylight Saving Time** 更改为其它）时按预期触发。

事件处理程序是如何执行的

当触发事件处理程序时，会建立一个可在其上执行事件处理程序的临时内部连接。处理程序不是在导致处理程序触发的连接上执行的，因此诸如 **MESSAGE...TO CLIENT** 之类与客户端应用程序进行交互的语句在事件处理程序内是没有意义的。同样，也不允许会返回结果集的语句。

用于执行处理程序的临时连接不计入授权的连接限制，不会为 **login_procedure** 选项指定的过程执行事件连接。

事件处理程序以事件所有者特权在一个单独的连接上执行。也可以在事件处理程序中调用过程，此时，过程以过程所有者特权执行。事件处理程序的单独连接不计入个人数据库服务器只能有 10 个连接的限制中。

任何事件错误都会记录到数据库服务器消息日志中。

注意： 如果在执行期间未检测到任何错误，则将提交事件处理程序中的事务。如果检测到错误，将回退。

隐藏事件处理程序

为提高安全性，可以使用 `ALTER EVENT` 语句来隐藏某个事件处理程序的定义。

前提条件

您必须具有 `MANAGE ANY EVENT` 或 `ALTER ANY OBJECT` 系统特权。

过程

1. 连接到数据库。
2. 执行带有 `SET HIDDEN` 子句的 `ALTER EVENT` 语句：

```
ALTER EVENT event-name SET HIDDEN
```

事件处理程序在事件处理程序定义中进行了永久模糊处理，此定义存储在 `ISYSEVENT` 系统表的操作列中。

事件处理程序

事件处理程序与触发了该事件的操作在不同的连接上执行，因此不与客户端应用程序进行交互。它们是使用事件创建者的特权执行的。

不管是用于调度事件的处理，还是用于系统事件处理，事件处理程序都包含复合语句，而且在很多方面都与存储过程类似。您可以添加循环、条件执行等。

每次执行事件处理程序后，如果没有出现错误便会发生 `COMMIT`。如果出现错误，则会发生 `ROLLBACK`。

事件处理程序的上下文信息

与存储过程不同，事件处理程序不带任何参数。可使用 `EVENT_PARAMETER` 函数访问在其中触发了事件的上下文的相关信息。返回的信息包括导致事件触发的连接 ID 和用户 ID、事件名以及该事件的执行次数。

测试事件处理程序

在开发过程中，您希望在方便时触发事件处理程序。即使在没有出现触发条件或调度时间时，您也可以使用 `TRIGGER EVENT` 语句显式导致事件的执行。但是，`TRIGGER EVENT` 不会导致被禁用的事件处理程序执行。

尽管在生产数据库上开发事件处理程序不是很好的做法，但您可以显式使用 `ALTER EVENT` 语句来禁用事件处理程序。

代码共享

用同一组操作来处理多个事件是很有用的。例如，如果数据库文件或日志文件所在设备上磁盘空间有限，您可能需要执行通知操作。为此，创建一个存储过程并在每个事件处理程序的正文中调用它，从而将任何所需的上下文信息作为参数传递给该过程。

调试事件处理程序

调试事件处理程序与调试存储过程非常类似。事件处理程序出现在事件列表中。

隐藏事件处理程序

可以使用含有 **SET HIDDEN** 子句的 **ALTER EVENT** 语句来隐藏事件处理程序的定义。指定 **SET HIDDEN** 子句的结果是永久模糊处理存储在 **ISYSEVENT** 系统表的操作列中的事件处理程序定义。

限制活动事件

通过使用含有 **NumActive** 上下文名称的 **EVENT_PARAMETER** 函数，您还可以确定某个特定事件处理程序有多少个实例当前处于活动状态。如果要限制事件处理程序以便在给定的时间只执行一个实例，则此函数是很有用的。

检索有关事件或调度的信息

SAP Sybase IQ 在系统表 **SYSEVENT**、**SYSEVENTTYPE** 和 **SYSSCHEDULE** 中存储有关事件、系统事件和调度的信息。

使用 **ALTER EVENT** 语句更改事件时，请指定事件名称，也可以指定调度名称。使用 **TRIGGER EVENT** 语句触发事件时，请指定事件名称。

通过查询系统表 **SYSEVENT**，可以列出事件名称：

```
SELECT event_id, event_name FROM SYSEVENT
```

通过查询系统表 **SYSSCHEDULE**，可以列出调度名称：

```
SELECT event_id, sched_name FROM SYSSCHEDULE
```

每个事件都具有唯一的事件 **ID**。可使用 **SYSEVENT** 和 **SYSSCHEDULE** 的 **event_id** 列将事件与关联的调度匹配。

审计数据库事件

审计在事务日志中记录了数据库事件。

dbtran 数据库管理实用程序

在命令提示符处使用 **dbtran** 翻译日志文件实用程序将事务日志转换为 .sql 命令文件。

语法

针对数据库服务器运行：

```
dbtran [ options ] -c { connection-string } -n SQL-file
```

针对事务日志运行：

```
dbtran [ options ] [ transaction-log ] [ SQL-file ]
```

参数

选项	描述
@data	读取来自指定的环境变量或配置文件的选项。
-a	控制是否在事务日志中显示未提交的事务。事务日志仅包含任何事务在最近一次执行 COMMIT 之前做出的更改。如果未指定 -a ，则只有提交的事务显示在输出文件中。如果指定 -a ，将显示事务日志中所有未提交的事务。
-c "keyword=value; ..."	在针对数据库服务器运行该实用程序时，此选项用于指定连接字符串。
-d	指定按照从最早到最晚的次序写入事务。此功能用于审计数据库活动：不要对数据库应用 dbtran 输出。
-ek key	为高度加密的数据库指定加密密钥。如果您有一个高度加密数据库，则必须提供加密密钥才能使用数据库或事务日志。指定 -ek 或 -ep ，但不要同时指定两者。如果没有指定正确的加密密钥，则该命令将失败。如果要针对数据库服务器运行 dbtran （使用 -c 选项），请使用连接参数而非 -ek 选项来指定密钥。例如，下面的命令将从数据库服务器示例中获取有关数据库 enc.db 的事务日志信息，并将其输出保存到 log.sql 。 <div>dbtran -n log.sql -c "ENG=sample;DBF=enc.db;UID=DBA;PWD=sql;DBKEY=mykey"</div>

选项	描述
-ep	<p>提示输入加密密钥。使用此选项将显示一个窗口，可以在其中输入加密密钥。这样，加密密钥决不会以明文显示，从而提供了额外的安全保证。指定 -ek 或 -ep，但不要同时指定二者。如果没有指定正确的加密密钥，则该命令将失败。如果要针对数据库服务器运行 dbtran（使用 -c 选项），请使用连接参数而非 -ep 选项来指定密钥。例如，下面的命令将从数据库服务器示例中获取有关数据库 enc.db 的事务日志信息，并将其输出保存到 log.sql。</p> <pre>dbtran -n log.sql -c "ENG=sample;DBF=enc.db;UID=DBA;PWD=sql;DBKEY=mykey"</pre>
-f	只输出上一个检查点之后完成的事务。
-g	如果启用了审计数据库选项，则审计信息会添加到事务日志中。
-ir <i>offset1,offset2</i>	输出在两个指定偏移之间的事务日志部分。
-is <i>source,...</i>	<p>对于已经由以下一个或多个源（以逗号分隔列表形式指定）中的操作修改了的那些行，输出对这些行执行的操作：</p> <ul style="list-style-type: none"> • All - 所有行。这是缺省设置。 • SQLRemote - 只包括使用 SQL Remote 修改的那些行。您也可以使用简写形式 "SR"。 • RepServer - 只包括使用复制代理 (LTM) 和复制服务器修改的那些行。您也可以使用简写形式 "RS"。 • Local - 只包括没有复制的行。
-it <i>owner.table,...</i>	输出对以逗号分隔列表形式指定的表所执行的操作。以 owner.table 形式指定每个表。
-j <i>date/time</i>	只转换给定日期或时间之前的最后一次检查点操作以来的事务。用户提供的参数可以是日期、时间或日期和时间（用引号括起来）。如果您省略时间，则缺省时间为 00:00 。如果省略日期，则缺省日期为当天。可接受的日期及时间格式为： "YYYY/MMM/DD HH:NN" 。
-k	防止检测到错误时消除部分 .sql 文件。如果在 dbtran 运行期间检测到错误，通常会消除到该时刻为止所生成的 .sql 文件，以确保不会使用部分文件。如果试图从损坏的事务日志中保留事务，指定此选项将甚为有用。
-m	指定包含事务日志的目录。将此选项与 -n 选项一起使用。
-n <i>filename</i>	在针对数据库服务器运行 dbtran 时，使用此选项指定保存 SQL 语句的输出文件。
-o <i>filename</i>	将输出消息写入指定的文件中。

选项	描述
-r	删除所有未提交的事务。这是缺省行为。
-rsu <i>username,...</i>	指定以逗号分隔的列表形式的用户名，以覆盖缺省复制服务器用户名。缺省情况下，-is 选项假定缺省的复制服务器用户名为 dbmaint 和 sa 。
-s	控制如何生成 UPDATE 语句。如果不使用此选项，而且在表上没有主键或唯一索引，则在出现重复行的情况下， dbtran 将生成带有非标准 FIRST 关键字的 UPDATE 语句。如果使用此选项，则忽略 FIRST 关键字，以便与 SQL 标准兼容。
-sr	将生成的用以说明 SQL Remote 如何将操作分发到远程站点的注释包含到输出文件中。
-t	控制是否在命令文件中包含触发器。缺省情况下，触发器执行的操作不包括在命令文件中。如果数据库中存在匹配的触发器，则在针对该数据库运行命令文件时，相应触发器将自动执行操作。如果运行命令文件的数据库中不存在匹配触发器，则应该包括触发器操作。
-u <i>userid,...</i>	将事务日志的输出限制为只包括指定的用户。
-x <i>userid,...</i>	将事务日志的输出限制为不包括指定的用户。
-y	在不提示进行确认的情况下替换现有的命令文件。如果您指定了 -q，则还必须指定 -y，否则操作将失败。
<i>transaction-log</i>	指定要转换的日志文件。不能与 -c 或 -m 选项一起使用。
<i>SQL-file</i>	命名包含已翻译信息的输出文件。只与 <i>transaction-log</i> 一起使用。

用法

您可以运行 **dbtran**：

- 针对数据库服务器 - 它使用在 -c 选项之后指定的连接字符串连接到数据库服务器，并将输出保存在使用 -n 选项指定的文件中。要以这种方式运行，需要具有 **BACKUP DATABASE** 系统特权。例如，以下命令用于转换来自服务器 **iqdemo** 的日志信息，并将输出保存在名为 **iqdemo.sql** 的文件中：

```
dbtran -c "eng=iqdemo;dbn=iqdemo;dbf=iqdemo.db;uid=DBA;pwd=sql" -n iqdemo.sql
```
- 针对事务日志文件 - 直接作用于事务日志文件。若要阻止用户运行此语句，请禁止对事务日志文件的一般访问。

```
dbtran iqdemo.log iqdemo.sql
```

dbtran 显示事务日志中的最早日志偏移，可用于确定多个日志文件的生成顺序。

dbtran-c 尝试转换联机事务日志文件及其同一目录中的所有脱机事务日志文件。如果目录中包含不止一个数据库的事务日志文件，可能会显示错误消息。为避免出现这种情况，请确保每个目录仅包含一个数据库的事务日志文件。

一个事务可以跨多个事务日志。如果事务日志文件包含跨多个日志的事务，则只转换一个事务日志文件（例如，`dbtran demo.log`）可能会导致跨日志的事务丢失。为了使 **dbtran** 生成完整的事务，请将 `-c` 或 `-m` 选项与目录中的事务日志文件一起使用。

退出代码为 0（成功）或非零值（失败）。

此实用程序接受 `@filename` 参数。

AUDITING 选项 [database]

在数据库中启用和禁用审计。

允许值

ON、OFF

默认值

OFF

范围

只能在数据库 (PUBLIC) 级别设置选项。

必须具有 **SET ANY SECURITY OPTION** 系统特权才能设置此选项。设置立即生效。

注释

审计是事务日志中有关数据库中许多事件的详细信息的记录。审计可提供一些安全功能，但会降低一些性能。打开数据库的审计功能时，无法停止使用事务日志。必须关闭审计功能才能关闭事务日志。无法以只读模式启动打开审计功能的数据库。

为使 **AUDITING** 选项起作用，必须将审计选项设置为 **ON**，并使用

sa_enable_auditing_type 系统过程指定要审计的信息类型，包括权限检查、连接尝试、DDL 语句、公共选项和触发器的任意组合。满足以下条件时，不会进行审计：

- **AUDITING** 选项设置为 **OFF**，或
- 审计选项已禁用。

如果将 **AUDITING** 选项设置为 **ON** 但不指定审计选项，则会记录所有类型的审计信息。

故障排除提示

SAP Sybase IQ 提供许多资源以供解决问题。

在线支持来源

如果无法使用文档解决问题，请参见 SAP Sybase IQ 在线支持 Web 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及已知问题。您甚至可以在线打开技术支持案例。

可以在大多数 Internet 浏览器中使用 MySybase。有关如何注册并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 SAP Sybase Web 站点，请参见《发行公告》。

特定情况的解决方案

可能需要更多信息来诊断和解决特定的问题。可以使用诊断工具诊断各种情况

服务器恢复和数据库修复的决策流程

在启动服务器或数据库，或者连接或验证数据库时可能会出现问题。

1. 服务器是否已启动？

如果是，转到步骤 2。

如果否，请参见服务器操作问题。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”，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。您可能需要从备份中恢复数据库。

2. 是否可以连接到数据库？

如果无法连接到数据库，请参见数据库连接问题以获得故障排除建议。

如果可以连接到数据库并且之前通过强制恢复启动了服务器，请参见“分配问题分析”，以了解验证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”，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您可以进行连接，但您希望验证数据库的一致性。

如果以前通过强制恢复启动了服务器，或者怀疑数据库不一致，请运行 DBCC 检查以验证数据库。有关检查索引一致性和数据库分配的信息，请参见“数据库检验”。

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" 部分报告错误，请参见“索引错误修复”，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" 部分报告错误，请参见“使用 DBCC 修复分配问题”，以了解使用 DBCC 修复分配问题的过程。

服务器操作问题

可能影响服务器运行的问题包括启动、关闭、无响应和异常中止。

SAP Sybase IQ 无法启动

如果在启动服务器时遇到问题，则 **start_iq** 会返回非零的值。

如果没有在 **-o** 开关启动后指定日志文件，则 SAP Sybase IQ 会将错误写入以下日志文件中最先定义的那一个：

- \$IQDIR16/logfiles/<servername>.nnnn.stderr
- \$IQDIR16/logfiles/<servername>.nnnn.srvlog
- 系统应用程序日志文件

可能的原因有多种。

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，请验证服务器使用的事务日志文件是否正确。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，请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。

```
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，并将该文件复制到数据库 .db 文件所在的目录中。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，则可以使用紧急恢复 **-f** 选项来启动服务器。如有需要，请联系 SAP Sybase 技术支持部门获取帮助。

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，在您尝试使用 **start iq** 启动服务器时，消息会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没有启动，请尝试使用 **iqsrv16** 命令启动服务器。**iqsrv16** 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。

日志文件具有非法名称

如果指定了单独的请求级记录文件，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 SAP 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 SAP Sybase IQ 服务器，则消息会出现在服务器日志文件 (*.srvlog) 中。

```
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
```

如果服务器日志文件中出现这些消息，并且服务器没有启动，请运行 **stop_iq** 命令 (UNIX) 以显示已在此计算机上运行的 SAP Sybase IQ 服务器的名称和端口号。然后尝试启动服务器，并指定未被使用的端口号，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 SAP Sybase IQ 将生成可用端口号。

如果启动服务器时不指定端口号，则服务器日志文件中将显示以下消息：

```
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
```

使用错误路径启动服务器

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，服务器将无法启动，并且在服务器日志文件 (*.srvlog) 中写入以下消息：

```
E. 08/18 07:22:19. MPX: server myserver
has been started with an incorrect catalog path
(expected path: /work/IQ-16_0/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。

环境变量设置不正确

如果数据库配置文件参数与 **start_iq** 使用的参数不同，请确保使用正确的参数启动服务器。

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。

如果服务器不是通过最基本的 **start_iq** 命令启动的，则尝试使用配置文件和命令行开关启动 iqdemo 演示数据库。如果服务器从 iqdemo 数据库启动，则您的数据库可能出现問題。

如果仍然无法运行 **start_iq** 命令，请使用 **iqsrv16** 命令。

注意： **iqsrv16** 仅用于排除服务器启动错误。始终使用 **start_iq** 启动 SAP Sybase IQ 服务器。

在运行 **iqsrv16** 之前，必须执行以下任务（通常由 **start_iq** 执行）：

- 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。请参见安装和配置指南 指南。
- 向环境中添加路径 \$SYBASE/OCS-15_0/lib，以便在调用 **iqsrv16** 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 上：

```
% setenv LIBPATH "${LIBPATH}:{SYBASE}/OCS-15_0/lib"
```

在其它 UNIX/LINUX 平台上：

```
% setenv LD_LIBRARY_PATH "${LD_LIBRARY_PATH}:${SYBASE}/OCS-15_0/lib"
```

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。

记录您在启动服务器时所在的目录。服务器启动目录决定了您以相对路径名创建的任何新数据库文件的位置。如果在其它目录中启动服务器，则 SAP Sybase IQ 找不到这些数据库文件。

在发出服务器启动命令之前，任何服务器启动脚本都应当更改为已知位置。

iqsrv16 的语法为：

```
iqsrv16 -n server-name -gm number  
[ other-server-switches ] [ database-file [ database-switches ] ]
```

注意： 在 **iqsrv16** 命令行中，指定的最后一个选项优先，因此如果要覆盖配置文件，可在配置文件名之后列出要更改的所有选项。例如：

```
iqsrv16 @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 **iqsrv16** 命令时服务器无法启动，请尝试使用 **iqsrv16** 实用程序以最小开关和参数再次启动。例如：

```
iqsrv16 -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。

使用 **iqsrv16** 命令启动服务器时，服务器不会在后台运行，消息也不会自动记录到服务器日志中。但如果包括 **-o** 文件名服务器开关，则除服务器窗口外，消息还将发送到该文件。

SAP Sybase IQ 停止处理或停止响应

您可以通过查看 SAP Sybase IQ 消息文件来检测导致服务器无响应的原因。

可能原因

服务器无响应的最常见原因包括：

- 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

如果服务器在处理过程中或关闭过程中看上去倾向于无响应，使用 **start_iq** 命令行选项 **-z** 和 SAP Sybase IQ 数据库选项 **QUERY_PLAN = 'ON'** 在 SAP Sybase IQ 消息 (**.iqmsg**) 文件和服务器日志 (**.srvlog**) 文件中记录有用的信息。

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 SAP Sybase IQ 消息文件和服务器日志文件中是否存在 **You have run out of space...** 消息。如果 IQ 主存储库或 IQ 临时存储库已用尽，使用 **CREATE DBSPACE** 命令适当添加 **dbspace**。

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** 期间处理空间不足的情况；这一点也十分重要。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 使用情况进行几分钟监控，确定 SAP Sybase IQ 服务器进程 (**iqsrv16**) 是否正在消耗 CPU 周期。记录此信息。如果 CPU 使用率更改，则 SAP Sybase IQ 服务器进程应正在正常处理。

如果 SAP Sybase IQ 服务器 CPU 使用率正常，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。

- 如果没有提示空间不足，请对新连接或现有连接使用 **Interactive SQL**，以便按指定顺序收集以下信息。

表 22. 要收集的服务器无响应信息

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
sp_iqstatus ># sp_iqstatus.out	数据库状态信息
sp_iqconnection ># sp_iqconnection.out	连接信息
sp_iqtransaction ># sp_iqtransaction.out	事务信息

如果无法解决此问题，请联系 SAP Sybase 技术支持部门获取帮助。他们可以使用您刚才收集的这些信息来帮助诊断问题。

- 服务器无响应时，可以通过在 \$IQDIR16/logfiles 目录（Windows 64 平台上为 %ALLUSERSPROFILE%\SybaseIQ\logfiles 文件夹，Vista 64 平台上为 C:\ProgramData\SybaseIQ\logfiles）中创建名为 DumpAllThreads 或 dumpallthreads 的文件来为每个 SAP Sybase IQ 线程生成堆栈跟踪。

按照建议使用程序管理器或 **start_iq** 命令启动 SAP Sybase IQ 时，系统将自动设置 **IQDIR16** 变量。如果未设置 **IQDIR16** 变量，则在 iqsrv16 的启动目录中创建 DumpAllThreads 文件。

SAP Sybase IQ 服务器将检测 DumpAllThreads 文件是否存在，并在堆栈跟踪文件 stkttrc-YYYYMMDD-HHNNSS_#.iq 中写入每个 IQ 线程的堆栈跟踪信息。堆栈跟踪信息写入堆栈跟踪文件之后，将删除 DumpAllThreads 文件。

SAP Sybase 技术支持部门可使用此堆栈跟踪信息帮助您诊断问题。

- 如果可以连接到数据库，请以十秒为间隔在持续十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 SAP Sybase IQ 消息文件的末尾处是否存在消息 Resource count 0，其后面可能跟有 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一种方案或几种方案的组合：
 - 通过减少 **-gm** 服务器启动选项的值来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 来增加硬件的处理容量

系统故障/SAP Sybase IQ 故障

您可以通过查看 SAP Sybase IQ 消息文件来检测导致系统/SAP Sybase IQ 故障的原因。

可能原因
有多种。

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)。这样可以确保文件中的任何有用信息都不会丢失。
- 在 UNIX 中，将堆栈跟踪的副本发送至 SAP Sybase 技术支持部门。堆栈跟踪信息应位于数据库服务器的启动目录中，位于名为 stktcr-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位于 SAP Sybase IQ 消息日志（缺省名称为 dbname.iqmsg）中。此信息有助于 SAP Sybase 技术支持部门确定故障发生的原因。
- 使用 **start iq** 命令重新启动服务器。当数据库重新启动时，将自动进行恢复。
- 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，请检查启动行上和连接配置文件中指定的数据库参数是否正确。
- 如果广泛查询目录存储表，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，如果连接超过其目录存储库临时文件空间的限额，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 的服务器窗口中键入 **q**，或在 Windows 的服务器窗口中单击“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：

1. 捕获 **ps** 操作系统实用程序输出，这样您便可将此输出提交给技术支持部门。在 Sun Solaris 中，有两个不同的 **ps** 选项可用。二者都要使用。

```
ps -aAdeflclj|egrep "PPID|iqlsrv16"

/usr/ucb/ps -awwwlx|egrep "PPID|iqlsrv16"
```

2. 尝试在操作系统级别终止进程以生成核心转储。

```
kill -6 pid
```

小型核心文件将在 **start_iq** 的运行目录中创建。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（如果已在步骤 2 中生成）和 **stktrc-YYYYMMDD-HHNNSS_#.iq** 中的堆栈跟踪信息提交给技术支持部门。

在 Windows 系统中：

1. 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“**iqlsrv16.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

在尝试连接到数据库时可能会遇到问题。

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，或数据源定义不正确。使用正确的用户 ID 和口令再次尝试连接。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。
- 用户可能无权使用该数据库。
- 您正通过 TDS 进行连接（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。使用正确的数据库文件名再次尝试连接。
如果使用 **Interactive SQL** 并且在连接到 `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）都必须存在。
- 可能超过了连接数限制或 **DBA** 定义的其它登录限制。
- 磁盘空间不足。检查 **SAP Sybase IQ** 消息文件中是否存在与磁盘空间相关的消息。
- 指定的服务器名不正确。检查服务器名，并使用正确的服务器名再次尝试连接。
- 服务器计算机名称或地址已更改。
- 如果第一次从客户端进行连接并且未指定服务器名，则在提供错误的端口号时可能导致无法连接到数据库。返回的错误消息如下：

```
Could not connect to the database.  
Database server not found.
```

从 **Interactive SQL** 进行连接时，确保“服务器名”字段中的名称拼写正确、“网络”选项卡中的网络选项是正确的，并确保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。要确定服务器名和该服务器正在监听的端口号，请运行 **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。
- 如果在尝试启动客户端时接收到以下消息：

```
Unable to start — server not found
```

或：

```
Database server not running.
```

则表明客户端无法在网络上找到数据库服务器。连接字符串可能不正确，或者服务器名高速缓存中可能包含不正确的或旧的连接信息。例如，如果从其它端口号启动服务器，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获取连接信息。

- 您在 **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。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警告。

操作

如果怀疑无法连接的原因是数据库出现问题，请查看 `dbname.iqmsg` 文件以确定问题发生在哪里。

已完成打开数据库表示数据库已正确打开，且问题与客户端连接有关。如果未显示此消息，则数据库可能没有成功打开或恢复。

Interactive SQL (dbisql) 问题

在使用 **dbisql** 时可能会遇到问题。

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。

可能的原因

IQTMP16 环境变量没有指向本地目录。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。SAP Sybase IQ 在连接终止时删除这些文件。如果 IQTMP16 没有指向本地目录，则无法找到 NFS 创建的 .nfs* 文件。

操作

将 IQTMP16 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。

磁盘空间不足

SAP Sybase IQ 服务器在 dbspace 不足的情况下不会等待额外的空间，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 dbspace 或主 dbspace 可用于缓冲区或 dbspace 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 dbspace 添加更多空间。（您可以选择添加文件而不添加 dbspace。一个 dbspace 可以有多个 dbfile。）

警告！ 磁盘空间用完时，如果 SAP Sybase IQ 持有某些系统锁或正在执行检查点，那么您可能无法添加磁盘空间。识别磁盘空间何时不足，并在空间不足之前添加新的 dbspace 是非常重要的。

操作

- 在 SAP Sybase IQ 消息日志 (dbname.iqmsg) 中检查最近的消息。空间不足消息表示必须添加其它 dbspace。SAP Sybase IQ 消息文件中的消息指示哪个 dbspace 中的空间已用完。如果在插入数据时出现该问题，则 IQ 主存储库中可能需要更多空间。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 IQ 临时存储库中可能需要更多空间。

检查 SAP Sybase IQ 消息日志中是否存在以下消息：

- 如果缓冲区或 dbspace 分配请求由于 dbspace 空间不足而失败，则该错误消息会记录到 dbname.iqmsg 消息文件中：

```
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 dbspace 的名称。

- 如果在 dbspace 不足的情况下回退整个事务，您会看到：

```
%1 -- Transaction rolled back"
```

```
[IQ_TRANSACTION_ROLLBACK: SQL Code -1285L,  
SQL State 40W09, Sybase Error Code 2973]
```

其中 %1 是服务器在关键操作过程中所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遇到空间不足的情况而无法刷新缓冲区，则会显示以下消息并回退当前语句：

```
%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，数据库服务器也会运行。运行 **sp_iqstatus** 获取详细信息。
- 如果无法连接到数据库，通过监控该处理器的 CPU 使用率来检查 SAP Sybase IQ 是否处于不能使用的状态。如果 CPU 使用率在一小段时间间隔后未发生变化，那么 SAP Sybase IQ 可能并未运行。如果 CPU 使用率发生变化，则 SAP Sybase IQ 正在运行。
- 检查 **sp_iqstatus** 输出中是否存在：

```
Main IQ Blocks Used:,10188 of 12288,  
82%, Max Block#: 134840
```

```
Temporary IQ Blocks Used:,163 of 6144,  
2%, Max Block#: 97
```

如果已使用的块所占百分比在九十以内，使用 **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%，表示不久后 IQ 主存储库将需要更多的空间。

- 如果出现空间不足状况，或者 **sp_iqstatus** 显示 Multiplex 服务器中有较高百分比的主块正在使用时，请运行 **sp_iqversionuse** 查明正在使用哪些版本以及通过释放这些版本可以恢复多少空间量。

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 **dbspace** 的情况下，任意新的检查点才能成功。

检查点对磁盘空间不足情况的影响

如果在请求检查点时 SAP Sybase IQ 已经空间不足，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 **dbspace** 的情况下，任意新的检查点才能成功。

在无法连接到服务器时添加空间

如果在操作过程中空间不足，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 任务栏中单击正确的服务器图标以显示 SAP Sybase IQ 窗口，然后单击“关闭”按钮。
 - 在 UNIX 上，运行 **stop_iq** 或在服务器启动窗口中键入 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 dbspace 大小

目录文件增长正常，并随应用程序和目录内容而变化。.db 文件的大小不会不影响性能，并且可以根据需要重用 .db 文件中的空闲页。

为最大限度地抑制目录文件增大：

- 避免对 **CREATE TABLE** 语句使用 **IN SYSTEM**。
- 在运行系统存储过程之后发出 **COMMIT** 语句。
- 在长时间运行事务之后发出 **COMMIT** 语句。

如果目录存储库不能扩展其中的某个文件（.tmp、.db，或 .iqmsg），SAP Sybase IQ 将返回 A dbspace has reached its maximum file size。为避免出现此类问题，可以：

- 定期监控空间使用情况。
- 验证 .tmp、.db 或 .iqmsg 文件所在的位置是否不存在操作系统文件大小限制（如 Sun Solaris **ulimit**）。.db 和 .tmp 文件通常位于主 SAP Sybase IQ 数据库目录中。.tmp 文件位于 \$IQTMP16/<servername>/tmp 下；如果未设置 \$IQTMP16，则该文件位于 /tmp/.SQLAnywhere/<servername>/tmp 下。

添加错误类型的空间

如果临时 dbspace 空间不足并且您 **CREATE DBSPACE** 命令中省略了 **TEMPORARY** 关键字，则无法创建临时 dbspace。

而是在作为 IQ_SYSTEM_TEMP 的现有临时 dbspace 中添加文件。

碎片

SAP Sybase IQ 通过利用甚至最小的未使用空间来提供碎片控制。

但碎片仍然会出现。如果数据库空间不足，虽然 **sp_iqstatus** 或 **.iqmsg** 文件列出的内存使用情况显示“已使用的主 IQ 块”值小于 100%，通常也表示数据库中存在碎片。

释放空间

当连接空间不足时，无法通过删除其它连接中的表或索引来释放空间；空间不足的事务将在其快照版本中显示这些对象。

为未来保留空间

SAP Sybase IQ 自动保留最少 200MB 和最后一个 **dbspace** 大小的 50%。

要确保在未来空间不足时有足够的空间添加新的 **dbspace**，请将数据库选项 **MAIN_RESERVED_DBSPACE_MB** 和 **TEMP_RESERVED_DBSPACE_MB** 设置为足够大的值，以便在执行 **COMMIT** 或 **CHECKPOINT** 期间处理空间不足情况。

监控磁盘空间使用情况

可以使用事件处理程序来监控磁盘空间使用情况，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进行空间监控尤其有用。可以在启动装载之前启用事件处理程序，并在装载完成后将其禁用。

可以修改此示例事件的处理程序代码，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 iqsrv16 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

```

以下代码将创建一个基于计时器的事件，用于监控空间使用情况，以帮助避免在执行非特权操作时可能由空间不足状况引起的意外回退。DBSpaceLogger 事件创建于示例 iqdemo 数据库中。

```

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;

```

线程数不足

系统可能不具有所需的服务器线程数来供查询使用。

可能原因

类似没有足够的服务器线程可用于此查询 [-1010011] ['QXA11'] 的客户端消息表示查询需要对 IQ 存储库使用其它内核线程。

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 **sp_iqconnection**。IQThreads 列中包含当前分配给连接的 IQ 线程数。此列可以帮助您确定哪些连接使用了最多的资源。某些线程可能已分配，但仍处于空闲状态。
- 如果此情况仍然存在，则可能需要重新启动服务器并指定其它 IQ 线程。使用 **-iqmt** 服务器启动开关可以增加 SAP Sybase IQ 可用的处理线程数。
缺省值为：对于前四个 CPU，每个 CPU 有 60 个线程，对于其余的 CPU，每个 CPU 有 50 个线程，供系统使用的至少有 3 个线程，再加上用于数据库连接和后台任务的线程。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 $\text{numConnections} + 3$ 。服务器线程总数不得超过 10000。
- 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，您可能会看到错误 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。SAP Sybase IQ 会尝试为每个备份设备分配至少一个线程以及为每个 CPU 分配两个线程，并为组分配一个线程。确保已经为每个连接、每组以及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。

堆栈跟踪文件 (stkttrc-YYYYMMDD-HHNNSS_#.iq) 中的 AbortIfEndofStack 指示线程堆栈已溢出。

可能原因

- 为避免出现此问题，请重新启动 SAP Sybase IQ 并将服务器参数 **-iqtss** 设置为 300（在 32 位操作系统上），或设置为 500（在 64 位操作系统上）。服务器启动开关 **-iqtss** 指定线程堆栈大小（以 KB 为单位）。如果此值不够，请增大 **-iqtss** 值，每次增加 72，直到解决问题。
- 如果可能，请识别出导致出错的命令，并将其转发给技术支持部门。

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台计算机上的客户端和服务端之间通信。每个客户端分配一个信号，每个服务器也分配一个信号。当客户端发送一个数据包供服务器读取时，它会发出服务器信号，同样，当服务器发送一个数据包供客户端读取时，服务器会发出客户端信号。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数。如果某个客户端需要分配多个信号以便与一个或多个服务器建立 **Multiplex** 连接，该客户端会尝试分配信号组中的所有信号。

可能原因

在 UNIX 系统上终止进程可能会导致留下信号或共享内存，而不是将其自动清理掉。要消除不需要的信号，应定期运行 UNIX **ipcs** 命令以检查信号和共享内存的状态。

ipcs -a 命令用于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 SAP Sybase IQ 实例均由同一用户启动（建议做法），则可以在 **OWNER** 列中搜索该用户名。确定未使用的共享内存段和信号。

操作

在与所有者确认这些共享内存段和信号未使用之后，请运行 UNIX **ipcrm** 命令删除它们。使用 **-m** 参数指定内存段 ID，使用 **-s** 命令指定信号 ID 编号，格式为：

```
ipcrm -m mid1 -m mid2 ... -s sid1 -s sid2 ...
```

例如：

```
% ipcrm -m 40965 -s 5130 -s36682
```

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某项操作，则不启动该操作，并返回缓冲区不足。

使用以下建议之一可解决该问题：

- 增加缓冲区高速缓存大小并重新运行该操作。
- 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。

表中有太多索引

表中有太多索引时，可能会出现问题。

可能原因

一位 Microsoft Access 用户正尝试链接到包含 32 个以上索引的表。

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 缓冲区高速缓存过大，因此操作系统出现抖动。
- IQ 缓冲区高速缓存过小，由于高速缓存不能容纳足够的查询数据，因此 SAP Sybase IQ 出现抖动。
- 您尝试设置 IQ 缓冲区高速缓存大小，使得系统的总内存需求超过系统内存总大小。因此，缓冲区高速缓存自动减少到其缺省大小。
- 存在需要 CIS 干预的用户定义函数或跨数据库连接。
- **WHERE** 子句和 **GROUP BY** 子句中使用了列缺少的 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。
- 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。

您还可以限制涉及散列算法的查询执行期间发生的抖动量。通过调整 `HASH_THRASHING_PERCENT` 数据库选项，可以控制在超过多少硬盘 I/O 百分比之后回退语句及返回错误。

`HASH_THRASHING_PERCENT` 的缺省值为 10%。增大此值将允许在回退之前对磁盘进行更多分页，而减小此值将减少在回退之前允许的分页。

如果达到缺省的 `HASH_THRASHING_PERCENT` 限制，则在 SAP Sybase IQ 早期版本中执行的涉及散列算法的查询现在将回退，并可能会显示以下两条消息之一：

- Hash insert thrashing detected.
- Hash find thrashing detected. (SQLState QFA43, SQLCode -1001047)

要为查询提供执行所必需的资源，请执行以下一项或多项操作：

- 通过增大 `HASH_THRASHING_PERCENT` 值放宽分页限制。
- 增加临时高速缓存（仅限 DBA）的大小。增大临时高速缓存将减小主缓冲区高速缓存。
- 对于此语句，尝试确定 SAP Sybase IQ 错误估计一个或多个散列大小的原因并缓解其后果。

- 减小数据库选项 `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** 索引的列中您具有 10000 个以上的唯一值：

```
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。发出 **DROP INDEX** 语句，以删除错误消息中标识的 **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

在对某个对象进行写入操作，而另一个用户已对该对象具有写入访问权限时，将报告以下错误消息。在当前事务（TxnID1）中无法打开请求的对象进行写入。另一用户具有事务 TxnID2 的写访问权限。

操作

使用 **sp_iqlocks** 存储过程以标识正在阻止其他用户对表进行写入操作的用户。此过程将显示关于数据库中当前持有的锁的信息，包括持有锁的连接和用户 ID、持有锁的表、锁类型以及用于标识锁的名称。

错误消息还包括正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详细信息，请运行 **sp_iqtransaction** 存储过程。

管理对表的写入锁争用

如果大多数事务都能够获取锁，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处理。

以下示例存储过程介绍了一种管理表写入锁争用的方法。该过程并不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。

示例存储过程代码对记录事件的 `dbo.event` 表的锁争用进行管理。该过程将 `event_id` 返回给调用程序。此表存在写入锁的高度争用。存储过程 `dbo.log_event` 记录表 `dbo.event` 中的信息。如果发生访问错误，将捕获该错误，有希望获得写入锁的写入程序将进行五秒的休眠，然后尝试再次对表执行写入操作。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。

您可对这些代码进行修改，从而执行其它相似任务。

```
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
```

为防止关键的更新操作失败，可提前预留所有必需表的写入锁。例如，假设在更新时需要表 `SalesOrders`、`Customers` 和 `SalesOrderItems`，下面的示例将预留这些表的写入锁：

```

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;

```

检查点提示

检查点时间和恢复时间的缺省值已足够大，大多数情况下无需进行更改。

检查点之间的时间缺省为 60 分钟。

可在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** 选项来调整检查点之间的时间。**-gc** 开关指定检查点超时期限的分钟数。**-gr** 开关指定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

有些设置可能影响性能。

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 SAP Sybase IQ 已知可用的实际 CPU 数，则能够以最有效的方式运行。在开启超线程功能的计算机上，或者在 SAP Sybase IQ 无法访问所有可用 CPU 的计算机上，SAP Sybase IQ 将创建很多个线程，因而运行效率低于其应有水平。

操作

在启动服务器时，将 **-iqnumbercpus** 设置为 SAP Sybase IQ 可用的 CPU 数，从而覆盖实际 CPU 数。

有关 **start_iq** 数据库选项的详细信息，请参见实用程序指南。

网络通信故障排除

网络软件包括若干个不同的组件，从而可能会增多需要排除故障的问题。

帮助排除网络故障的主要来源是您的网络通信软件供应商所提供的网络通信软件文档和技术支持。但您可以采用最佳做法并利用诊断工具来获取各种情况的信息。

使用兼容协议

如果在客户端或服务器计算机上安装了多个协议堆栈，应确保客户端和数据库服务器使用同一个协议。

服务器的 `-x` 命令行开关用于选择一系列供服务器使用的协议，对于客户端应用程序，使用 **CommLinks** 连接参数可实现同一目的。

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。

使用最新驱动程序

根据需要，确保网络适配器有最新版本的 **NDIS** 或 **ODI** 驱动程序。

您应当能够从适配器卡的制造商或供应商那里获得最新的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。

下载 **Novell** 客户端软件时，除了用于所有网络适配器的 **Novell** 软件，还要包括某些网络适配器的 **ODI** 驱动程序。

在两次重新启动之间关闭计算机电源

某些网络适配器板在重新启动计算机时不能完全重置。如果要进行故障排除，请关闭计算机，等待几秒钟，然后再重新开启计算机。

逐层诊断协议堆栈

如果在客户端应用程序与数据库服务器通信期间遇到问题，请确保客户端与数据库服务器使用兼容的协议堆栈。

一种发现网络通信问题的方法是激发协议堆栈，测试是否每个级别的通信均能正常运行。

如果可以连接到服务器计算机，则数据链接层生效，而不管是否使用将要用于 **SAP Sybase IQ** 的相同更高层协议建立连接。

例如，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。

在验证出数据链接层生效后，接下来验证与 **SAP Sybase IQ** 使用相同网络和传输层的其它应用程序是否运行正常。

测试 TCP/IP 协议堆栈

如果使用 **TCP/IP** 运行，可利用多个应用程序来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。

使用 Ping 测试 IP 层

每个 **IP** 层都有一个关联的地址 - 以句点分隔的四个整数（如 191.72.109.12）。**ping** 将 **IP** 地址作为参数，并尝试将单个数据包发送到指定的 **IP** 协议堆栈。

首先，通过对您自己的计算机执行 "**Ping**" 操作来确定您自己的协议堆栈是否正常运行。例如，如果您的 **IP** 地址是 191.72.109.12，请在命令提示符处输入以下命令：

```
ping 191.72.109.12
```

等待一段时间，查看是否发送数据包。如果已发送数据包，您将看到以下类似输出内容：

```
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** 操作正常运行，将指出计算机能够将数据包发送到自身。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。

在进行其它诊断之前，确保可以从客户端计算机对运行数据库服务器的计算机执行 **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** 实现附带了几个能够用于实现此目的的应用程序。要使用 **telnet** 命令测试 **TCP/IP** 堆栈：

1. 在一台计算机上启动一个 **Telnet** 服务器进程（或守护程序）。在 **TCP/IP** 软件文档中查阅相关指令。对于典型的命令行 **Telnet** 程序，请在命令提示符下输入以下内容：

```
telnetd
```

2. 在其它计算机上启动该 **Telnet** 客户端进程，并查看是否可以连接。同样，请在 **TCP/IP** 软件文档中查阅相关指令。通常，输入以下类似指令：

```
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，并且客户端和服务端应当能够使用 **TCP/IP** 链接进行通信。如果无法建立 **Telnet** 连接，则说明存在问题。在继续执行其它诊断之前，确保 **TCP/IP** 协议堆栈正常运行。

诊断布线问题

网络布线或连接器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似计算机上再现问题。如果问题只出现在一台计算机上，可能是布线或硬件问题。

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** 授权的转售商还可以为您提供一位 **NetWare** 认证工程师的姓名，以帮助您诊断和解决布线问题。

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收到消息无法启动 - 未找到服务器，则表明客户端在网络中找不到数据库服务器。

- 客户端计算机上网络驱动程序的网络配置参数可能不同于服务器计算机上的相应参数。例如，两个以太网适配器卡应当使用一种通用帧类型。对于 **Novell NetWare**，在 `net.cfg` 文件中指定帧类型。在 **Windows** 中，在控制面板网络设置中查找帧类型设置。
- 使用 **TCP/IP** 协议时，客户端通过广播请求来搜索数据库服务器。此类广播通常不经过网关，因此，如果数据库服务器位于另一网络（子网）内的计算机上，将找不到这些数据库服务器。此时，必须使用 `-x` 服务器启动命令行选项来提供运行有服务器的计算机的主机名。要通过 **TCP** 连接到 **NetWare** 服务器，必须执行此操作。
- 网络驱动程序或布线安装不正确。
- 网络驱动程序的网络配置参数可能与 **SAP Sybase IQ** 的多用户支持不兼容。

“无法初始化任何通信链接”消息

消息无法初始化任何通信链接表示无法建立链接。

可能原因是尚未安装网络驱动程序。除非已经使用 `-x` 服务器启动选项另行指定协议，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，了解如何安装所需的驱动程序。

诊断工具

您可利用若干工具来诊断各种情况。

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. 启动实用程序服务器，以使其在恢复期间忽略已备份的数据库中的所有临时 IQ 文件定义：

```
start iq -n utility_startup_svr -c 32m
-x 'tcpip{port=1234}' -iqnotemp
```

- 2. 恢复数据库：

```
RESTORE DATABASE 'iqdemo'
FROM '/system1/IQ16/IQ-16_0/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` 添加新的临时 `dbfile`

sp_iqstatus 存储过程

`sp_iqstatus` 存储过程提供了各种 SAP Sybase IQ 状态信息。

注意： 以下示例显示 `iqdemo` 示例数据库的输出。示例用户 `dbspace iq_main` 可能不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)	Copyright (c) 1992-2013 by Sybase, Inc. All rights reserved.
Version:	16.0.0.6552/110812/P/GA/Sun_Sparc/OS 5.10/64bit/2012-08-12 03:08:39
现在时间:	2012-09-13 10:33:19.979
内部版本时间:	2012-08-13 03:08:39

故障排除提示

文件格式:	23 on 03/18/1999
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
主存储库空间不足:	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
本地临时块: 1-3200	IQ_SYSTEM_TEMP
创建时间:	2013-08-17 11:31:03.313
更新时间:	2013-09-12 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

Main IQ I/O 与 Temporary IQ I/O 输出代码的键:

- I: 输入
- L: 读取的逻辑页 (“查找”)
- P: 读取的物理页
- O: 输出
- C: 创建的页
- D: 脏页
- P: 物理写入
- D: 损坏的页
- C: 压缩率

检查以下信息:

- Main IQ Blocks Used 和 Temporary IQ Blocks Used 行指示正在使用 **dbspace** 的哪部分。如果已使用的块所占百分比 (这些行上的中间统计信息) 达到九十以上, 请添加 **dbspace**。
- 由于保留块不能用于用户操作, Main IQ Blocks Used 和 Temporary IQ Blocks Used 的计算公式分别为: 行 DB Blocks (主 IQ 块总数) 减去 Main Reserved Blocks Available 以及 Temp Blocks (临时 IQ 块总计) 减去 Temporary Reserved Blocks Available。
- Main IQ Buffers 行与 Temporary IQ Buffers 行告知主缓冲区高速缓存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及占用的总空间。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，最后将被删除。如果它们回退，则将释放“创建的”数据。
- Main Reserved Blocks Available 和 Temporary Reserved Blocks Available 显示可用的保留空间大小。
- Main IQ I/O 行和 Temporary IQ I/O 行按照 IQ 消息日志中的相同格式显示 I/O 状态。

解释通知消息

缺省情况下，SAP 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表示何时需要执行维护和优化任务，如添加更多 dbspace。这些消息还会报告装载进度。

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 SAP 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更多行以来 SAP 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 SAP Sybase IQ 每次读取的行数相同，时间长度也会因所读取的数据（例如，需要多少次数据转换）而异。所报告的时间间隔小于 1 秒通常会被报告为“0 秒”。

内存消息

内存消息显示 SAP 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 23. 内存使用消息

项目	描述
Mem: # mb	此 SAP Sybase IQ 服务器当前使用的内存（以 MB 为单位）。
M# mb	此 SAP Sybase IQ 服务器自启动以来使用的最大内存（以 MB 为单位）。

IQ 主存储块消息

IQ 主存储块消息显示 IQ 主存储库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 IQ 主存储库：

```
Main      Blks: U63137/6%, Buffers: U12578/L7
```

表 24. IQ 主存储块消息

项	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。此值通常为 100%，因为缓冲区管理器会将缓冲区保留在内存中，直到一些其它数据需要这些缓冲区。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 SAP Sybase IQ 会尽量积极有效地使用缓冲区。</p> <p>此值将会增大到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，但在主缓冲区高速缓存中，SAP Sybase IQ 可能会也可能不会损坏缓冲区，原因是只要缓冲解锁，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。</p>
L#	<p>锁定的缓冲区数。锁定的缓冲区正在使用中，无法将其从高速缓存中删除。SAP Sybase IQ 会锁定某些对象（如散列对象）的缓冲区，以便将其保存在内存中。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>缓冲区锁不占用任何内存。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>

当服务器磁盘空间较少时识别到这一点，并在服务器用光磁盘空间之前及时添加新的 dbspace，这一点非常重要。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向您发出通知的示例，请参见监控磁盘空间使用情况。

IQ 临时存储块消息

IQ 临时存储块消息显示 IQ 临时存储库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 存储库：

Temporary Blks: U273/0%, Buffers: U1987/L1960

表 25. 临时 IQ 存储块消息

项	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。此值通常为 100%，因为缓冲区管理器会将缓冲区保留在内存中，直到一些其它数据需要这些缓冲区。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。 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。锁定的缓冲区当前正在使用中，无法将其从高速缓存中删除。IQ 会锁定某些对象（如散列对象）的缓冲区，以便将其保存在内存中。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。 缓冲区锁不占用任何内存。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。

当服务器磁盘空间较少时识别到这一点，并在服务器用光磁盘空间之前及时添加新的 dbspace，这一点非常重要。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向您发出通知的示例，请参见监控磁盘空间使用情况。

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 IQ 主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 IQ 主存储缓冲区高速缓存的相关信息：

Main I: L331224/P22 O: D25967/P7805 C:D0

表 26. IQ 主存储缓冲区高速缓存消息

项	描述
Main:I:L#	逻辑文件读取次数。
P#	物理文件读取次数。

项	描述
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，它指出数据的压缩程度（即占未压缩时大小的百分比）。该数字越大越好。只有特定的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。排序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，总数据压缩率可能会更高。

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分配了过多内存。

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 IQ 临时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 IQ 临时存储缓冲区高速缓存的相关信息：

```
Temporary I: L25240/P8 O: D4749/P0 C:D0
```

表 27. 临时 IQ 存储缓冲区高速缓存消息

项	描述
Temporary: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，它指出数据的压缩程度（即占未压缩时大小的百分比）。该数字越大越好。只有特定的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。比率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，总数据压缩率可能会更高。

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

每建立一次数据库连接，都会在 .iqmsg 文件中临时缓冲区高速缓存消息的后面记录一次连接句柄、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 文件中的连接信息相关联，请参见“使 .srvlog 和 .iqmsg 文件的连接信息相关联”。

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**。

此过程用于将每个数据库页从磁盘读取到内存中，并执行各种一致性检查。但检查可能会运行很长一段时间，具体取决于数据库的大小。

sp_iqdbstatistics 显示在最近一次执行 **sp_iqcheckdb** 时收集的数据库统计信息。

检查数据库和服务端启动选项值

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端启动选项的当前值。

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 DBA 执行 **sp_iqcheckoptions** 时，他将看到为所有角色和用户设置的所有永久性选项，并看到为 DBA 设置的临时选项。非 DBA 用户只能看到自己的临时选项。所有用户都会看到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 中执行：

```
sp_iqcheckoptions
```

系统表 DBO.SYSOPTIONDEFAULTS 包含 SAP Sybase IQ 和 SQL Anywhere 选项的所有名称和缺省值。可以查询此表以查看所有选项缺省值。

查找当前执行的语句

诊断问题时，可能需要知道当问題发生时正在执行的语句。

sp_iqcontext 存储过程用于列出当前运行的语句，并识别发出相应语句的用户和连接。可将此实用程序与 **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。

要使用连接信息匹配 .iqmsg 日志与 **-zr** 服务器请求日志条目，请将 .srvlog 和 .iqmsg 文件之间的连接信息相关联。

记录服务器请求

要找出某些类型的问题，尤其是与查询相关的问题，请记录服务器请求。

可使用以下两种方式之一来启用请求级别的记录：

- 在启动服务器时设置 **-zr** 命令行选项，或者
- 调用 **sa_server_option** 存储过程，以覆盖 **-zr** 的当前设置。

服务器请求记录在 *.srvlog 中。**-zr** 服务器启动选项将启用操作的请求级别的记录，并设置要记录的请求类型 (SQL|HOSTVARS|PLAN|PROCEDURES|TRIGGERS|OTHER|BLOCKS|REPLACE|ALL|NONE)。**-zo** 选项将请求级别的记录信息重定向到一个非常规的日志文件中。**-zs** 用于限制此文件的大小。

注意： 如果要写入日志的查询文本的大小超过了指定的限制，查询文本不会被截断而是完整地进行记录。

一旦数据库服务器启动，您便可使用 **sa_server_option** 调整请求日志设置，以记录更多或更少的信息。以下命令将启用对有限的一组请求进行请求记录，并将输出重定向到文件 sqllog.txt 中：

```
call sa_server_option('RequestLogging','SQL');
      call sa_server_option('RequestLogFile',
                           'sqllog.txt')
```

要禁用请求级记录，请使用：

```
call sa_server_option('RequestLogging','NO');
```

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');
```

要使用连接信息匹配 .iqmsg 日志与 **-zr** 服务器请求日志 (.srvlog) 条目，请将 .srvlog 和 .iqmsg 文件之间的连接信息相关联。

注意： SAP Sybase IQ 版本 15.1 修改了请求日志。共同信息开始以逗号分隔的文本形式记录，而不是固定格式的行前缀。如果可能，将时间记录为 "="（表示与前一行相同）或 +nnn（表示前一行之后的 nnn 毫秒）。因此，请求日志比 SAP 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。

下面的输出摘录自以 **-zr all** 选项启动服务器时的请求日志。在本例中，用户将连接到 `iqdemo` 数据库，并执行命令 `sp_iqstatus`。

每行中有多个以逗号分隔的字段，第一个字段指示时间。完整的时间戳将定期按以下形式输出：

```
MMdd hhmmss.sss
0523 095954.807,[,1000000001,sp_iq_mpx_init,16,iq utilities status 1
```

对于此行之后的那些行（例如 `+13,C,1,UID=DBA`），显示了与前一行之间的偏移量。在此例中，`+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。

下面的内容摘录自请求日志：

```
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
```

请求日志文件分析

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 分析请求日志，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 也分析请求日志，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，以及《参考：构件块、表和过程》中的 **sa_server_option** 系统过程。

请求记录

请求记录会记录从应用程序接收的各个请求以及发送到应用程序的响应。请求记录在确定应用程序要求数据库服务器所执行的操作时用处最大。

在不清楚是数据库服务器还是客户端有故障时，要对特定应用程序进行性能分析，也最好先从请求记录开始。可以使用请求记录来确定问题的根源是否是向数据库服务器发出特定请求。

注意： 请求记录功能提供的所有功能和数据也可使用诊断跟踪来获得。诊断跟踪还可提供其它的功能和数据。

所记录的信息包括时间戳、连接 ID 和请求类型等。对于查询，记录的信息还包括隔离级别、读取的行数和游标类型。对于 **INSERT**、**UPDATE** 和 **DELETE** 语句，记录的信息还包括受到影响的行数以及触发的触发器数。

注意： 注释

请求日志包含的具有敏感信息的所有语句均已进行模糊处理。只有具有解析错误的语句才会包含未进行模糊处理的敏感信息。

当启动数据库服务器时，您可以使用 `-zr` 服务器选项来启动请求记录。使用 `-zo` 服务器选项，您可以将输出重定向到一个请求日志文件，以便进一步分析。`-zn` 和 `-zs` 选项允许您指定要保存的请求日志文件数以及请求日志文件的最大大小。

`sa_get_request_times` 系统过程读取请求日志，并用日志中的语句和它们的执行时间填充全局临时表 (`satmp_request_time`)。对于 `INSERT/UPDATE/DELETE` 语句，记录的时间是执行语句时的时间。对于查询，记录的时间是从 `PREPARE` 到 `DROP`（描述/打开/读取/关闭）所经过的总时间。这意味着您需要知道任何打开的游标。

分析 `satmp_request_time` 以找到可能需要改进的语句。开销低但经常执行的语句可能会出现性能问题。

您可以使用 `sa_get_request_profile` 调用 `sa_get_request_times`，并将 `satmp_request_time` 汇总到另一个名为 `satmp_request_profile` 的全局临时表中。此过程还可将语句组合在一起，并提供调用次数、执行时间，等等。

警告！ 小心

如果使用 `tracetime.pl` Perl 脚本对日志进行分析，应将 `max_client_statements_cached` 选项设置为 0，以便在捕获请求日志时禁用客户端语句缓存。

示例

使用 `sa_server_option` 系统过程，可以对输出到请求日志的内容进行过滤，以便只包括来自特定连接或特定数据库的请求。在监控具有多个活动连接或多个数据库的数据库服务器时，这有助于减小日志的大小。

- **根据连接过滤** – 使用以下语法：

```
CALL sa_server_option( 'RequestFilterConn' , connection-id );
```

可以通过执行 `CALL sa_conn_info()` 获得 *connection-id*。

- **根据数据库过滤** – 使用以下语法：

```
CALL sa_server_option( 'RequestFilterDB' , database-id );
```

当连接到数据库时，通过执行 `SELECT`

`CONNECTION_PROPERTY('DBNumber')` 可以获得 *database-id*。在显式重置过滤或关闭数据库服务器之前，过滤将一直有效。

- **重置过滤** – 使用下面两个语句中的任何一个，按连接或按数据库重置过滤：

```
CALL sa_server_option( 'RequestFilterConn' , -1 );
```

```
CALL sa_server_option( 'RequestFilterDB' , -1 );
```

- **将主机变量输出到请求日志** – 要在请求日志中包括主机变量值：

- 使用具有值 `hostvars` 的服务器选项 `-zr`

- 执行以下语句：

```
CALL sa_server_option( 'RequestLogging' , 'hostvars' );
```

请求日志分析过程 `sa_get_request_times` 可识别日志中的主机变量，并将它们添加到全局临时表 `satmp_request_hostvar` 中。

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

如果服务器在启动时返回通信错误，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 SAP 技术支持部门联系或与所在区域的 SAP 子公司联系。

对于购买了支持合同的客户安装的每一个 SAP 产品，都会有一位或多位指定人员获得与 SAP 技术支持部门联系的授权。

技术支持人员需要您的 SAP Sybase IQ 环境的相关信息，以便解决您的问题。

如果您存在 SAP Control Center 方面的问题，请参见 SCC 中针对 SAP Sybase IQ 的 SAP Control Center 联机帮助，或访问 <http://sybooks.sybase.com/sybooks/sybooks.xhtml?prodID=10680>。

使用 **getiqinfo** 收集诊断信息

SAP Sybase IQ 包括一个脚本，用于收集 SAP 技术支持部门在诊断问题时所需的信息。

getiqinfo 脚本将收集有关操作系统环境、SAP Sybase IQ 环境和日志文件的信息。

向 SAP 技术支持部门报告问题之前，请先运行此脚本。通过执行此操作，可以帮助 SAP 人员更快解决问题，并减轻您的负担。

getiqinfo 不是为排除 SAP 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 SAP 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。

在运行该脚本之前，需了解以下内容：

- 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（如果 SAP Sybase IQ 消息文件已经重命名）的完整路径

如果可能，请让 SAP 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 SAP Sybase IQ 在运行时才会提供的内部数据库数据。该脚本不会自动启动服务器。

该脚本以启动 SAP Sybase IQ 服务器时所使用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（程序的启动目录）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-16_0\bin64 目录中的批处理脚本。**getiqinfo** 脚本随 IQ 服务器一起安装，不能用于网络客户端。

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，在 IQ-16_0/bin64 目录中，键入：

```
getiqinfo.sh
```
- 在 Windows 中，选择“开始”>“运行”><install_path>
\\IQ-16_0\\bin64\\getiqinfo.bat。

2. 根据提示您的内容，请输入：

- 数据库文件的目录。这也是 .iqmsg 文件的缺省位置，在 UNIX 上是 stkttrc*.iq 文件的缺省位置。
- 数据库文件的基本名（不带 .db 后缀的文件名）。这也是 .iqmsg 文件的缺省基本名。
- 要在其中搜索这些文件的其它目录。
- SAP Sybase IQ 引擎名称（服务器名称）和该数据库服务器的端口号。
- 授予了以下特权之一的用户的用户 ID 和口令：
 - DROP CONNECTION 系统特权
 - MONITOR 系统特权
 - SERVER OPERATOR 系统特权

- 用于启动 SAP Sybase IQ 服务器的配置文件（如果已使用）的完整路径
- **-zo** 服务器选项中的输出文件（如果已指定一个输出文件）的完整路径。

程序还会指导您将列出的文件发送给 SAP Sybase 技术支持部门。

由 **getiqinfo** 收集的信息

getiqinfo 脚本用于收集信息。

- 硬件类型、内存大小、CPU 类型、速度、CPU 数目
- 操作系统（例如，Sun Solaris 2.10）
- 交换空间大小
- SAP Sybase IQ 版本和 EBF 级别以及 Anywhere 版本
- 与发生此问题时的日期和时间相对应的堆栈跟踪文件，名为 `stktrc-YYYYMMDD-HHMMSS_#.iq`，在数据库服务器的启动目录中。（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`，缺省情况下位于数据库服务器的启动目录中
- 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，`IQ-16_0/logfiles/<servername>.000n.stderr` 和 `IQ-16_0/logfiles/<servername>.000n.srvlog`
 - 在 Windows 平台上，如果需要，必须重新启动服务器，并手动收集控制台窗口的副本
- 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：

- 所使用的连接协议（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）
- 数据库的模式和索引
- **sp_iqcheckdb** 过程的输出

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`。

如果您存在性能问题，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 **SAP 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。

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_dbpace_temp dbspace info
' || dbpaceName
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.dbpace_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. ],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。

支持 Web 站点

如果无法解决某个问题，可能会在 SAP Sybase IQ 在线支持 Web 站点 MySybase 上找到更多帮助信息。

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/>，然后单击 "MySybase"。

清单：技术支持信息

可以运行 **getiqinfo** 脚本收集信息。

要求提供的信息	值
SAP Sybase IQ 版本（例如，16.0 GA 或 SP 编号）	
sp_iqlmconfig 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	
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）	
消息日志文件 (dbname.iqmsg)	
服务器日志文件 (server.nnnn.srvlog 和 server.nnnn.stderr)	

故障排除提示

要求提供的信息	值
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（如果可能）	

附录：连接参数和通信参数参考

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。

连接参数

连接参数包括在连接字符串中。

您可在以下位置输入连接参数：

- 应用程序的连接字符串
- ODBC 数据源
- “Sybase IQ 连接”对话框

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”对话框使用相同的格式。一些参数与这些对话框中的复选框和字段相对应，另外一些可在“高级”选项卡上的文本框中输入。

用法说明

连接参数不区分大小写。

每个连接参数的用法介绍了该参数的使用环境。常见的用法条目包括：

- **嵌入式数据库** - 当 SAP Sybase IQ 用作嵌入式数据库时，连接会启动服务器并装载该数据库。当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 SAP 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）参数为真时值为 YES、ON、1 或 TRUE；为假时值为 NO、OFF、0 或 FALSE。

您可以从下列位置获得接口库使用的连接参数（按优先级顺序）：

- 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按从左到右的顺序读取。如果指定同一个参数多次，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信息并使用该文件中的条目（如果它们尚未设置）。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。

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** 中不可用。

参数由存有客户端进程信息（如客户端计算机的 IP 地址、运行此客户端进程的操作系统等）的生成的字符串组成。该字符串在数据库服务器中与特定连接相关联，您可以使用以下语句对其进行检索：

```
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。有效的键值如下：

- **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 nogui -c "uid=DBA;pwd=sql" -host MachineName -port 1234
```

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=12.345.67.899;
HOST=MachineName;
OSUSER=sybase;
OS='SunOS 5.10 Generic_144489_04';
EXE='/Sybase/IQ16/shared/JRE-7_0_1/bin/amd64/java';
PID=0x52af;
THREAD=0x14;
VERSION=16.0.0.290;
API=iAnywhereJDBC;
TIMEZONEADJUSTMENT=-240
```

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

```
isql -U DBA -P sql -S MachineName:1234 --appname MyISQL
```

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=MachineName;
PID=21155;
EXE=MyISQL
```

AutoPreCommit 连接参数 [AutoPreCommit]

强制每个语句在执行前提交。

用法

ODBC

缺省值

否

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，以使用户始终可以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** 以启用在执行前提交，或设置为 **NO** 禁用。在执行前提交。在 `.odbc.ini` 文件中或在“连接”对话框的“高级”选项卡上设置此选项。

例如，以下示例将导致每个语句在执行前提交：

```
[Sample DSN]
DatabaseFile=c:\Program Files\Sybase\IQ-16_0\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-16_0\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**，则从连接中启动的数据库不再存在任何连接时并不会被卸载。因此，数据库服务器也不会关闭。

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-16_0\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，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。

CompressionThreshold (COMPTH) 连接参数

增大或减小压缩包的大小限制。

语法

```
{ CompressionThreshold | COMPTH }=size[ k ]
```

用法

除 **TDS** 以外的任何位置。只适用于压缩连接。

允许值

- **size** – 该整数指定压缩包的大小限制。该值以字节为单位，但可使 **k** 将单位指定为千字节。如果客户端和数据库服务器指定了不同的压缩阈值设置，则使用客户端设置。所支持的最小值为 1 个字节，所支持的最大值为 32767 字节。建议不要使用小于 80 字节的值。

缺省值

120

如果未设置 **CompressionThreshold** 值，则压缩阈值由服务器上的设置（缺省值为 120 字节）控制。

注释

通过更改压缩阈值，您可以只在压缩会提高数据包的传输速度时才对数据包进行压缩，从而提高压缩连接的性能。

如果启用了压缩功能，则单个数据包可能会压缩也可能不会压缩，具体情况视其大小而定。例如，即使启用了通信压缩，**SAP Sybase IQ** 也不压缩小于压缩阈值的数据包。此外，小数据包（小于 100 字节）通常不压缩。由于压缩数据包时需要 CPU 时间，因此尝试压缩小数据包实际上可能会降低性能。

在速度很慢的网络上降低压缩阈值可能会改善性能，而提高压缩阈值可能会通过减少 CPU 占用来改善性能。但是，由于降低压缩阈值在客户端和服务器上均会增加 CPU 使用率，因此应进行性能分析以确定更改压缩阈值是否有好处。

示例

连接，压缩阈值为 100 个字节。

```
CompressionThreshold=100
```

CommBufferSize 连接参数 [CBSize]

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。

语法

```
{ CommBufferSize | CBSize }=size[ k ]
```

用法

任何位置

允许值

- **size** – 指定通信包的最大大小。该缺省值以字节为单位，但可使用 **k** 将单位指定为千字节。**CommBufferSize** 的最小值是 500 字节，最大值是 65535 字节。

缺省值

如果未设置 **CommBufferSize** 值，则 **CommBufferSize** 由服务器上的设置（缺省值为 7300 字节）控制。

注释

协议栈决定了网络上包的最大大小。如果设置的 **CommBufferSize** 值大于网络所允许的值，则通信包由网络软件拆分。缺省值为标准以太网 TCP/IP 最大数据包大小（1460 字节）的倍数。

较大的数据包大小可以提高多行读取和较大行读取的性能，但同时会增加客户端和服务器的内存使用。

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，所有未指定自己的 **CommBufferSize** 值的客户端都将使用 **-p** 数据库服务器选项所指定的大小。

示例

将缓冲区的大小设置为 1460 字节：

```
...
CommBufferSize=1460
...
```

另外，您也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

CommBufferSpace 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

字符串

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** 可能会影响系统性能，因此，仅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，将会出现一个连接错误并中止连接尝试，即使列表中仍有其他协议也是如此。

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，该行为等同于 **CommLinks=ShMem**。共享内存协议用于在同一台计算机的相同操作系统下运行的客户端和服务端之间的通信。

CommLinks 参数的可用值有：

- **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** - 首先尝试使用共享内存协议进行连接，然后使用所有其余的可用通信协议进行连接。如果无法确定使用哪个（些）协议，则使用此设置。

- **TCP/IP** - 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来代替 **ALL**：

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

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 iqsrv16** 服务器选项相对应。缺省情况下，网络服务器启动所有可用的协议，这等同于 **-x ALL**。

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
```

ConnectionName 连接参数 [CON]

对连接进行命名，以便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。

语法

```
ConnectionPool = { NO | YES  
  [ ( [ Timeout=timeout-sec; ] [ MaxCached=max-cached-conn ] ) ] }
```

用法

除了非线程 Unix 客户端外的所有平台。

允许值

- **timeout-sec** - 连接池的空闲超时时间，以秒计算。缺省值为 60 秒。将断开高速缓存中在 timeout-sec 值指定的时间内未重新使用的连接，且不可重新使用。
- **max-cached-conn** - 各应用程序在高速缓存中的最大连接数。缺省值为五个连接。如果连接断开且没有达到 max-cached-conn 值指定的最大连接数，将此连接放入高速缓存中。连接被重新初始化，高速缓存中的连接仍保持与数据库服务器的连通状态，即使应用程序已将其断开。

缺省值

YES

注释

连接池可以提高与数据库服务器建立多个简短连接的应用程序的性能。连接断开后会自动被放入高速缓存中，以便应用程序再次连接时重新使用。对于将被放入池中的连接，连接名称可以不同，但其它所有连接参数都必须相同。

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。

在特殊情况下，SAP 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

描述

DBF 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（但不带 .db 扩展名）装载数据库，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。
- 文件的路径与数据库服务器的工作目录相关。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 上的安装目录为 C:\Program Files\Sybase\IQ-16_0\demo），请使用以下 DBF 参数：

```
DBF=C:\Program Files\Sybase\IQ-16_0\demo\iqdemo.db
```

DatabaseName 连接参数 [DBN]

指定需要与之建立连接的已装载数据库。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

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，才会进行连接。

注意： 如果同时指定了数据库名和数据库文件，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，则可以成功地连接到正在运行的名为 **Kitchener** 的数据库：

```
DBN=Kitchener
```

- 或者：

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

- 但是，这无法连接到名为 **Kitchener** 的数据库：

```
DBF=cities.db
```

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** 用作命令行选项来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

下面的 UNIX 命令（在一行中输入所有内容）将连接到缺省数据库服务器，装载数据库文件 `IQ-16_0/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-16_0/demo/iqdemo.db;  
dbn=my_db;dbf=-n my_db" /tmp/temp.col
```

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（UNIX 上）或 odbc.ini 文件或注册表（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
无

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 驱动程序将查找包含其余连接参数的数据源。在 SAP 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
```

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
以下配置文件片段使用加密密钥 "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]

关闭跨网络的多记录读取。

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]

指定所要连接的运行中的数据库服务器。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
要连接到网络服务器，需要提供 **EngineName**。在“连接”对话框和“ODBC 管理器”中，它是 **Server Name** 字段。

服务器名按照客户端计算机的字符集进行解释。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的信息，请参见实用程序指南。

协议	截断长度
UNIX 共享内存	31 个字节

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

示例

连接到名为 **Guelph** 的服务器：

```
ENG=Guelph
```

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**，**Password** 优先级较高。

Encryption 连接参数 [ENC]

使用传送层安全或简单加密方式加密在客户端应用程序和数据库服务器之间发送的数据包。

用法

对于 **TLS**，仅 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

如果您注重网络包的安全性，可以使用此参数。加密将轻微影响性能。**Encryption (ENC)** 连接参数接受以下参数：

- **None** 接受未加密的通信包。
- **Simple** 接受用所有平台上和 SAP Sybase IQ 12.6 之前版本上支持的简单加密进行加密的通信包。简单加密不提供服务器验证、RSA 加密或其它传送层安全性功能。
- **TLS** 接受使用 RSA 加密技术加密的通信包。对于 FIPS 认证的 RSA 加密，请指定 FIPS=Y。RSA FIPS 使用单独的认证库，但与指定 RSA 的 SAP Sybase IQ 服务器兼容。所有受支持的平台均支持客户端与服务器之间的 TLS 连接，但只有 LinuxAMD 和 Windows（32 位和 64 位）支持 FIPS=Y。要验证服务器，软件将使用下列参数验证服务器的证书值是否与您提供的任何客户端相关值相匹配：
 - **trusted_certificate** 指定客户端用于验证服务器身份的证书文件。
 - **certificate_company** 指定组织字段的值。服务器的值和客户端的值必须匹配。
 - **certificate_unit** 指定组织单位字段的值。服务器的值和客户端的值必须匹配。
 - **certificate_name** 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！ 示例证书仅用于测试目的。在部署环境下，它不具备任何安全性，因为它和对应的口令是通过 SAP Sybase IQ 软件进行广泛分发的。为保护您的系统，请创建您自己的证书。

可以使用 **CONNECTION_PROPERTY** 系统函数来检索当前连接的加密设置。

示例

- 此连接字符串片段使用 RSA 加密和示例受信任证书通过 TCP/IP 链接连接到名为 myeng 的数据库服务器：

```
"ENG=myeng; LINKS=tcip;  
Encryption=(FIPS=N;TRUSTED_CERTIFICATE=iq-16_0/samples/  
certificates/rsaroot.crt)"
```

Escape 连接参数 [ESCAPE]

允许指定返回表或列的列表时，ODBC 驱动程序生成的 SQL 语句的 LIKE 子句中所用的转义字符。

描述

只有 SAP Sybase IQ 支持 **Escape** 参数。

缺省情况下，ODBC 驱动程序使用代字号 (~) 作为转义字符，而某些应用程序则使用反斜线 (\) 作为转义字符。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线 (\)，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
无

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 应用程序均可使用文件数据源。

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
```

Idle 连接参数 [IDLE]

Idle 连接参数决定着连接超时的时间。

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 SAP Sybase IQ 的 **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 iqsrv16** 服务器选项。

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

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**（不建议使用），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**（不建议使用）或 **Integrated** 的情况下运行。

示例

以下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）。

如果未指定任何语言，则使用缺省语言。

该连接参数仅影响连接。从 **SQL Anywhere** 的工具和实用程序返回的消息会以缺省语言显示，而从服务器返回的消息则以连接的语言显示。

LazyClose 连接参数 [LCLOSE]

导致 **CLOSE *cursor-name*** 数据库请求排队，并和下一个数据库请求一起发送到服务器。每次关闭游标时，则无需进行网络请求。

用法
任何位置

值
YES, NO

缺省值
NO

描述
启用该参数时，直到下一个数据库请求之前游标实际上未关闭。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

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似处于关闭状态，但实际上它在服务器端并没有关闭。之

后尝试打开其它同名的游标将会失败。如果应用程序经常取消请求，则不要使用 `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。

除了非线程 **UNIX** 应用程序之外的所有平台。

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，用以确认连接的完好性。如果客户端的运行时间达到了活动超时时间且没有检测到活动请求或响应包，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。

通信断开后，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，从而删除与该服务器之间的所有当前连接。

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。这使服务器可以更有效地处理大量连接。

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
```

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*

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当前工作目录中的 `error.log` 文件：

```
...
LogFile=error.log
...
```

LogicalServer 连接参数 [LS]

指定连接的目标逻辑服务器。

用法

任何位置

*值**String**描述*

使用该参数显式指定目标逻辑服务器，从而使 SAP Sybase IQ 能够将连接从一个 **Multiplex** 节点重定向到另一个 **Multiplex** 节点。

如果没有在连接字符串中指定 **LogicalServer**，则用户登录策略中的缺省逻辑服务器设置将充当逻辑服务器上下文。

LoginRedirection 连接参数 [REDIRECT]

在连接级别控制登录重定向。

用法

必须在逻辑服务器级别启用登录重定向才能使该参数生效。

值

[**ON** | **OFF**]

缺省值

ON

描述

某些应用程序可能需要重定向到逻辑服务器中的特定节点。

当指定 **REDIRECT=OFF** 时，该连接不会发生登录重定向。如果初始连接节点无法满足目标逻辑服务器和所请求服务器角色的连接要求，则连接失败。

例如，如果逻辑服务器中包含 **SQL Anywhere** 表（该表不可在节点之间进行共享）数据，则可在连接级别上禁用向该逻辑服务器中的目标节点进行的登录重定向。

NewPassword (NEWPWD) 连接参数

让用户可以更改口令（即使口令已经到期），而无需 **DBA** 干预。

语法

```
{ NewPassword | NEWPWD }={ password-string | * }
```

用法

只有 **Microsoft Windows** 上支持提示输入新口令的客户端库。

允许值

- **password-string** - 如果用户提供新口令，数据库服务器就会验证用户 **ID** 和口令，并在调用 **login_procedure** 选项前尝试更改该口令。此过程允许用户更改到期口令，而不涉及 **DBA**。如果设置了 **verify_password_function** 选项，则会验证新口令。如果通过集成登录或 **Kerberos** 登录进行验证，则不校验原始口令，数据库服务器会忽略新口令值，口令也不会被更改。
- ***** - 在 **Microsoft Windows** 上，如果使用特殊值 *****，则只有在现有口令过期时客户端库才会在尝试连接期间提示输入新口令。用户必须提供现有口令和新口令，并确认新口令。用户填写完各字段并单击“**确定**”后，系统会验证旧口令，数据库服务器也会尝试更改该口令。如果设置了 **verify_password_function** 选项，则会验证新口令。验证用户口令是否过期的过程、系统提示输入口令、验证和更改口令都通过单一连接调用客户端库进行。

缺省值

不更改口令，客户端库不提示输入新口令。

注释

当您使用 **password_life_time** 或 **password_expiry_on_next_login** 选项实施登录策略时，此连接参数非常有用。也可以通过使 **login_procedure** 发出 [口令已到期] 出错信号来实施口令到期策略。

如果所处环境不支持口令提示，用户会收到一条 [口令已到期] 错误消息。在 **Microsoft Windows** 环境中，如果调用应用程序具有多个顶级窗口或者该应用程序的顶级窗口为

最小化，提示窗口可能不会成功避免与调用应用程序窗口（它可能不是对话框模式或者具有相应的父窗口）的相互作用。

在 Windows 环境下，如果使用 ODBC SQLDriverConnect 函数且 DriverCompletion 参数为 SQL_DRIVER_NOPROMPT 外的任意值，则当口令过期时，该连接就会提示输入新口令。如果 DBPROP_INIT_PROMPT 属性为 DBPROMPT_NOPROMPT 外的任意值，则在 OLE DB 中连接可能会提示输入新口令。以上两种情况的作用方式都类似于指定了 NewPassword=* 连接参数。

- 用户 ID 不能：
 - 以空格、单引号或双引号开头
 - 以空格结尾
 - 含有分号
- 口令区分大小写并且不能：
 - 以空格、单引号或双引号开头
 - 以空格结尾
 - 含有分号
 - 长度超过 255 个字节

示例

以下连接字符串用于更改用户 Test1 在连接时的口令：

```
"UID=Test1;PWD=welcome;NEWPWD=hello"
```

在 Windows 环境下，以下连接字符串用于在现有口令到期时提示用户 Test1 输入新口令：

```
"UID=Test1;PWD=welcome;NEWPWD=*"
```

NodeType 连接参数

为逻辑服务器成员节点指定服务器角色。

用法
任何位置

值
{ **READER** | **WRITER** | **ANY** }

缺省值
ANY

描述
逻辑服务器中可以包含读取程序节点和写入程序节点这二者作为成员。该参数允许应用程序使用特定角色连接到成员节点。可以将所需的角色指定为 **NODETYPE = {READER | WRITER | ANY}**

- **READER** – 执行用户定义函数的应用程序可能需要使用 **READER** 角色连接到成员节点。
- **WRITER** – 执行语句（如 **INSERT**、**UPDATE** 和 **DELETE**）的应用程序可能需要使用 **WRITER** 角色连接到成员节点。
- **ANY** – 大多数只读应用程序（不执行任何 UDF）可以连接到逻辑服务器中的任何成员节点。

当使用 **SERVER** 逻辑服务器上下文登录时，SAP Sybase IQ 会忽略连接参数 **NodeType**。

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
每个数据库用户拥有一个口令。必须提供口令，用户才可以连接数据库。缺省情况下，口令与数据都区分大小写，因而 SAP Sybase IQ 数据库也缺省区分大小写。

口令参数未加密。如果将口令存储在连接配置文件中，应使用 **EncryptedPassword** 参数。SAP Sybase IQ ODBC 配置工具使用加密参数。

如果同时指定了 **Password** 和 **EncryptedPassword**，**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。

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
```

PrefetchRows 连接参数 [PROWS]

PrefetchRows 连接参数可设置查询数据库时可预取的最大行数。

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;...
```

RetryConnectionTimeout (RetryConnTO) 连接参数

指示客户端库（DBLIB、ODBC、ADO 等）在找到服务器之前于指定的期间内不断进行连接尝试。

语法

```
{ RetryConnectionTimeout | RetryConnTO }=timeout-value
```

用法

任何位置

允许值

- **timeout-value** – 此连接指定的值是超时值（以秒为单位）。它不是重试连接尝试的计数器。缺省值零指出仅应尝试连接一次。

缺省值

0

注释

在迭代之间有半秒的延迟，并且仅在由于未找到数据库服务器而导致连接尝试失败的情况下才会重试。对于任何其它错误，将立即返回错误消息。如果未找到数据库服务器，尝试连接的时间至少不会小于由 **RetryConnectionTimeout** 连接参数指定的时间。

缺省 TCP 超时为 5 秒，因此，如果连接字符串中包含的 **RetryConnTO** 值少于 5 秒（例如 `Host=host-name;RetryConnTO=3`），则尝试连接仍将需要 5 秒。

示例

以下连接字符串片段通知客户端库继续重试连接尝试至少 5 秒:

```
...RetryConnTO=5;...
```

ServerName 连接参数 [ENG]

EngineName 连接参数 [ENG] 的同义词

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

只有连接到当前未在运行的数据库服务器时，才提供 **StartLine** 参数。**StartLine** 参数是用于启动服务器的命令行。

示例

以下数据源片段用 32MB 的高速缓存启动数据库服务器。

```
StartLine=dbeng6 -c 32M iqdemo.db
```

Unconditional 连接参数 [UNC]

使用 **dbstop** 停止服务器（即使该服务器存在连接）。

用法

任何位置

缺省值

否

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**，则仅在没有活动连接时才关闭服务器。

示例

以下命令行无条件关闭服务器：

```
dbstop -c "uid=DBA;pwd=SQL;eng=server-name;unc=yes"
```

另请参见

实用程序指南 > “stop_iq 数据库关闭实用程序”

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

连接数据库时必须始终提供用户 ID。用户 ID 不区分大小写，并且不受 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端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;. . .),...
```

从客户端输入通信参数，如下例所示：

```
CommLinks=tcpip(PARM1=value1;PARM2=value2;. . .),...
```

如果包含空格，那么必须用引号引起网络连接参数才能正确分析：

```
start_iq -x "tcpip(PARM1=value 1;PARM2=value 2;...),..."
start_iq -x "tcpip(PARM1=value1;PARM2=value2;...)"
```

在 UNIX 中，如果提供多个参数，也需要使用引号，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。

在同一行上输入命令以设置连接参数；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 和 HTTPS
Broadcast [BCAST]	标识
BroadcastListener [BLISTENER]	Identity_Password
ClientPort [CPORT]	DatabaseName [DBN]
DLL	LocalOnly [LOCAL]
DoBroadcast [DOBROAD]	LogFile [LOG]

TCP/IP	HTTP 和 HTTPS
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]	

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

缺省广播地址由本地 IP 地址和子网掩码生成。子网掩码指示哪部分 IP 地址识别网络，哪部分识别主机。

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。

示例

要通知客户端在使用 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)
```

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 关闭对此端口的监听。使用 **-sb 0** 等同于在 TCP/IP 上指定 BroadcastListener=NO。

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)
```

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

由网络实现为每个连接动态地指派。如果您没有防火墙限制，请不要使用该参数。

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。

指定端口号的列表或范围，使用给定的数据源或连接字符串建立多个连接。如果您指定一个端口号，则应用程序每次只能保持一个连接。实际上，即使是在关闭一个连接之后，也会有很短的停顿期间，在此期间无法使用指定的端口建立任何新连接。如果指定端口号的列表或范围，应用程序会一直尝试这些端口号，直到找到一个可成功捆绑到的端口号。

示例

- 以下字符串在使用端口 6000 的应用程序和使用端口 5000 的名为 `my_server` 的服务器之间建立连接：

```
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
```

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，但没有必要。如果 URI 中不包含数据库名，将使用服务器上的缺省数据库来处理 Web 请求。由于在设置为 **AUTO** 时数据库服务器必须确定 URI 中是否包含数据库名，因而在 Web 站点设计中应避免多义性。

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
```

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 - 如果 DoBroadcast=ALL (以前是 DoBroadcast=YES), 则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=, 还会将广播包发送到每个主机。对于 TCP, 所有的广播包都是 UDP 包。
- 如果 DoBroadcast=DIRECT (以前是 DoBroadcast=NO), 则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 HOST (IP) 通信参数中列出的主机。如果指定 DoBroadcast=DIRECT, 则需要 HOST (IP) 通信参数。
- 指定 DoBroadcast=NONE 将导致不使用 UDP 广播。TCP/IP 连接直接通过指定的 HOST/PORT 建立, 而且系统会验证服务器名。使用 TCP/IP 时, 通过将 VerifyServerName (VERIFY) 通信参数设置为 **NO**, 您可以选择不对服务器名进行验证。HOST (IP) 通信参数是必需参数, ServerPort (PORT) 通信参数是可选参数。
- 对于 DIRECT 和 NONE, 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。在极少数情况下此设置非常有用, 但通常不建议使用此设置。

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
```


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

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，应使用接口标识符。UNIX 平台既支持接口标识符，也支持 IPv6 地址中的接口名称。而在 Linux（内核 2.6.13 及更高版本）上，则需要使用接口标识符。

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=tcip(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 "tcip(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=tcip(HOST=my_server,kangaroo;PORT=2639)"
```

Identity 通信参数

指定标识文件的名称。

语法

Identity=*identity-file*

可用的协议

HTTPS

允许值

- **identity-file** - 此字符串用于指定标识文件的名称。

缺省值

无

注释

使用传送层安全时，标识文件包含公共证书及其专有密钥；对于不是自签名的证书，标识文件也包含所有签名证书，其中包括加密证书。该证书的口令必须通过 **Identity_Password** 参数指定。

示例

启动要求 **Web** 连接使用特定的加密证书的数据库服务器。

```
iqsrv16 -xs https (Identity=cert.file;Identity_Password=secret) ...
```

Identity_Password 通信参数

指定加密证书的口令。

语法

Identity_Password=*password*

可用的协议

HTTPS

允许值

- **password** - 此字符串指定加密证书的口令

缺省值

无

注释

使用传送层安全时，此选项用于指定与 **Identity** 通信参数指定的加密证书口令相匹配的口令。

示例

启动要求 Web 连接使用特定的加密证书的数据库服务器。

```
iqsrv16 -xs https(Identity=cert.file;Identity_Password=secret) ...
```

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

允许运行在 WAN 上或穿过防火墙的客户端查找服务器，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* 是 `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。

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服务器名的服务器，则不会自动启动任何服务器。

只有指定 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，以将连接限制为本地计算机。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。运行 LocalOnly (LOCAL) 通信参数被设置为 YES 的服务器时，可以将网络服务器作为个人服务器运行，而不受连接或 CPU 的限制。

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

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
- 与请求相关的数据库的名称
- 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
- 包括消息标头和消息主体在内的响应的长度（以字节为单位）
- HTTP 请求方式
- 与请求相关的监听器端口
- 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 响应的状态代码和说明
- HTTP 状态代码
- 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)，在由于出错而没有处理请求时为 0.000

LogMaxSize (LSIZE) 协议选项

控制数据库服务器在其中写入 Web 请求信息的日志文件的最大大小。

语法

```
{ LogMaxSize | LSIZE }=size[ k | m | g ]
```

可用的协议

HTTP、HTTPS

允许值

- **size** - 此整数指定写入 Web 请求信息的文件的最大大小。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 LogMaxSize 为 0，则日志文件大小没有限制。

缺省值

0

注释

当日志文件达到指定的大小时，将重命名该文件并创建另一个日志文件。

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）。

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。

MaxRequestSize (MAXSIZE) 协议选项

指定数据库服务器可接受的最大请求的大小。

语法

```
{ MaxRequestSize | MAXSIZE }=size[ k | m | g ]
```

可用的协议

HTTP、HTTPS

允许值

- **size** – 此整数指定数据库服务器可接受的最大请求的大小。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。值 **0** 禁用该限制，但应慎用。无此限制时，恶意的客户端可能造成数据库服务器过载，或者导致服务器内存不足。

缺省值

100k

注释

如果请求的大小超出此限制，将关闭连接，同时客户端会返回一个表示请求过大的响应。该值仅限制请求的大小，并不限制响应的大小。

示例

以下命令指示数据库服务器接受大小不超过 150000 字节的请求：

```
iqsrv16 -xs http(MaxRequestSize=150000)
```

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，**SAP 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，可在 **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），**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** 平台既支持接口标识符，也支持 **IPv6** 地址中的接口名称。而在 **Linux**（内核 2.6.13 及更高版本）上，则需要使用接口标识符。

示例

- 以下 **Windows** 命令行（在一行中输入所有内容）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-16_0\demo\iqdemo.db
```

- 以下命令行（在一行中输入所有内容）指示服务器使用 **IPv6** 网卡：

```
start_iq -x tcpip(MyIP=fe80::5445:5245:444f)
"c:\sybase\IQ-16_0\demo\iqdemo.db"
```

- 以下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，以便在打开时进行预取。使用 `PreFetchOnOpen` 时，在打开和首次读取之间重新绑定游标的列会使系统性能降低。

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**]

v

允许的最大值

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**]

允许的最大值

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** 分别指定千字节、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。

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 通信的端口通知给客户端。

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 编号授权委员会已为 SAP Sybase IQ 数据库服务器分配端口号 2638 用于进行 TCP/IP 通信的。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 **ServerPort** 中指定的端口号进行 TCP/IP 连接。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 中，启动 SAP Sybase IQ 网络服务器：

```
start_iq -x tcpip c:\sybase\IQ-16_0\demo\iqdemo.db
```

现在采用端口号 2638。

2. 尝试启动另一台数据库服务器：

```
start_iq -x tcpip c:\sybase\IQ-16_0\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 上启动另一台数据库服务器：

```
start_iq -x "tcpip(ServerPort=2639)" c:\sybase\IQ-16_0\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 服务器已经在使用端口 80，或者您没有权限在这样的低端口号上启动服务器，则可启动监听替代端口（如 8080）的服务器：

```
start_iq -xs http{port=8080} -n server3 web.db
```

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所允许的命令数量有限制。SAP Sybase IQ 使用这些 NetBIOS 命令，并在系统中没有更多可用命令的情况下禁止进一步的连接，即使在 NetBIOS 命令数小于 Sessions 参数值时也是如此。

缺省值

OS 特定。在 Windows 上，缺省值为 16。

示例

下面的语句将使用名为 iqdemo 的数据库启动服务器，并允许 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

要禁止与数据库服务器建立 **TDS** 连接，请将 **TDS** 设置为 **NO**。如果希望确保仅与服务
器建立加密连接，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）

允许的最大值

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，且应慎用。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

通常情况下，不应设置此选项。此选项仅用于连接到 **Multiplex** 辅助服务器以平衡这些服务器之间的查询负载。

使用 DoBroadcast=NONE 参数通过 **TCP** 建立连接时，客户端会建立 **TCP** 连接，然后验证找到的服务器的名称与要查找的服务器的名称是否相同。指定 VerifyServerName=NO 可跳过对服务器名的验证。这允许 **IQ** 客户端在只知道一个 **IP** 地址/端口的情况下连接到 **SAP Sybase IQ** 服务器。

服务器名仍必须在连接字符串中指定，但会被忽略。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，可在客户端计算机的“**ODBC 管理器**”中创建新的 **ODBC DSN**，并按如下所示指定参数：

- 在“数据库”选项卡上，指定一个用于连接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，但会因“网络”选项卡中的参数而被忽略。
- 在“网络”选项卡上，选中“**TCP/IP**”复选框，然后在文本框中输入以下内容：

```
host=hostname;port=port#;DOBROADCAST=NONE;VERIFY=NO
```

例如：

```
host=hostname;port=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。

附录：SQL 语句和选项参考

特定的 SQL 语句和数据库选项具有支持 SAP Sybase IQ 数据库管理的特殊的语法。

数据库选项

数据库选项修改数据库行为。

数据库选项的作用域和持续时间

可以将选项设置为三种级别的作用域：公共、用户和临时。

临时选项优先于用户设置和公共设置。用户级选项优先于公共设置。如果为当前用户设置用户级选项，也会设置对应的临时选项。

某些选项（如 COMMIT 行为）的作用域是数据库范围。设置这些选项要求具有 DBA 权限。其它选项（如 ISOLATION_LEVEL）也可以只应用于当前连接，并且不需要特殊权限。

根据选项的不同，选项设置更改的时间也不同。对全局选项（如 RECOVERY_TIME）的更改发生在下一次服务器启动时。一些选项在服务器重启后生效：

需要重启服务器的数据库选项
CACHE_PARTITIONS
CHECKPOINT_TIME
OS_FILE_CACHE_BUFFERING
OS_FILE_CACHE_BUFFERING_TEMPDB
PREFETCH_BUFFER_LIMIT
PREFETCH_BUFFER_PERCENT
RECOVERY_TIME
SWEEPER_THREADS_PERCENT
WASH_AREA_BUFFERS_PERCENT

通常，只有影响到当前连接的选项才会立即生效。例如，可以在一个事务中间更改选项设置。

警告！ 在游标打开时更改选项会导致不可靠的结果。例如，在游标打开时，更改 `DATE_FORMAT` 或许不会改变下一行的格式。根据游标检索的方式，或许要到几行之后更改才会对用户生效。

DEDICATED_TASK 选项

将请求处理任务专用于处理来自单个连接的请求。

允许值

ON、OFF

默认值

OFF

范围

只能针对单个连接、PUBLIC 角色或当前连接的持续期间将选项设置为临时选项。

必须具有 SET ANY SYSTEM OPTION 系统特权才能设置此选项。设置立即生效。

注释

DEDICATED_TASK 连接选项设置为 ON 时，一个请求处理任务将专门处理一个连接的请求。通过启用此选项预先建立连接，可以在数据库服务器意外不响应时，收集有关该服务器的状态信息。

LOG_CONNECT 选项

控制用户连接的记录。

允许值

ON、OFF

默认值

ON

范围

只能在数据库 (PUBLIC) 级别设置选项。

必须具有 SET ANY SECURITY OPTION 系统特权才能设置此选项。设置立即生效。

注释

此选项设置为 ON 时，每次用户连接到 SAP Sybase IQ 数据库或与其断开时，都将在 IQ 消息日志（.iqmsg 文件）中出现一条消息。

注意： 如果用户连接时将此选项设置为 OFF（禁用连接记录），然后在用户断开连接之前将其打开，则消息日志将显示该用户断开但未连接。

MIN_PASSWORD_LENGTH 选项

设置数据库中新口令的最小长度。

允许值

大于或等于零的整数

值以字节为单位。对于单字节字符集，它与字符数相等。

默认值

3 个字符

范围

只能在数据库 (PUBLIC) 级别设置选项。

必须具有 SET ANY SECURITY OPTION 系统特权才能设置此选项。设置立即生效。

注释

此选项对所有新口令强制使用一个最小长度，以实现更高的安全性。现有口令不受影响。

示例

将新口令的最小长度设置为 6 个字节：

```
SET OPTION PUBLIC.MIN_PASSWORD_LENGTH = 6
```

VERIFY_PASSWORD_FUNCTION 选项

指定可用于实现口令规则的用户提供的验证函数。

允许值

字符串

缺省值

" (空字符串) 。（设置口令时不调用函数。）

范围

可在数据库 (PUBLIC) 或用户级别设置选项。在数据库级别进行设置时，值将变为任何新用户的缺省值，但不会对现有用户产生任何影响。在用户级别进行设置时，仅替换该用户的 PUBLIC 值。为自身设置选项无需任何系统特权。在数据库或用户级别为任何其他用户设置选项都需要系统特权。

必须具有 SET ANY SECURITY OPTION 系统特权才能设置此选项。可针对个别连接或 PUBLIC 角色进行临时设置。设置立即生效。

注释

当 **VERIFY_PASSWORD_FUNCTION** 选项值设置为有效字符串时，语句 **GRANT CONNECT TO *userid* IDENTIFIED BY *password*** 将调用该选项值所指定的函数。

该选项值需要使用 *owner.function_name* 格式，以防止用户覆盖该函数。

该函数具有两个参数：

- *user_name* VARCHAR(128)
- *new_pwd* VARCHAR(255)

返回值类型为 VARCHAR(255)。

如果设置了 **VERIFY_PASSWORD_FUNCTION**，则无法使用 **GRANT CONNECT** 语句指定多个用户 ID 和口令。

示例

以下示例代码定义表和函数并设置一些登录策略选项。它们共同实现了高级口令规则，其中包括要求在口令中使用某些类型的字符、不允许口令重用和为口令设置有效期。当创建用户 ID 或更改口令时，数据库服务器会通过 *verify_password_function* 选项调用函数。应用程序可以调用 *post_login_procedure* 选项指定的过程，以报告口令应在到期前进行更改。

```
-- only DBA should have privileges on this table
CREATE TABLE DBA.t_pwd_history(
    pk          INT          DEFAULT AUTOINCREMENT PRIMARY KEY,
    user_name   CHAR(128),   -- the user whose password is set
    pwd_hash    CHAR(32) );  -- hash of password value to detect
                             -- duplicate passwords

-- called whenever a non-NULL password is set
-- to verify the password conforms to password rules
CREATE FUNCTION DBA.f_verify_pwd( uid      VARCHAR(128),
                                  new_pwd  VARCHAR(255) )
RETURNS VARCHAR(255)
BEGIN
    -- enforce password rules
    -- enforce minimum length (can also be done with
    -- min_password_length option)
    IF length( new_pwd ) < 6 THEN
        RETURN 'password must be at least 6 characters long';
    END IF;

    -- number of lowercase characters IN new_pwd
    SELECT count(*) INTO num_lower_chars
        FROM pwd_chars WHERE CAST( c AS BINARY ) BETWEEN 'a' AND 'z';

    -- enforce rules based on characters contained in new_pwd
    IF ( SELECT count(*) FROM pwd_chars WHERE c BETWEEN '0' AND '9' )
        < 1 THEN
        RETURN 'password must contain at least one numeric digit';
    ELSEIF length( pwd_alpha_only ) < 2 THEN
```

```

        RETURN 'password must contain at least two letters';
    ELSEIF num_lower_chars = 0
        OR length( pwd_alpha_only ) - num_lower_chars = 0 THEN
        RETURN 'password must contain both upper- and lowercase
characters';
    END IF;

    -- not the same as any user name
    -- (this could be modified to check against a disallowed words
table)
    IF EXISTS( SELECT * FROM SYS.SYSUSER
                WHERE lower( user_name ) IN
( lower( pwd_alpha_only ),
                                lower( new_pwd ) ) ) THEN
        RETURN 'password or only alphabetic characters in password '
||
        'must not match any user name';
    END IF;

    -- not the same as any previous password for this user
    IF EXISTS( SELECT * FROM t_pwd_history
                WHERE user_name = uid
                AND pwd_hash = hash( uid || new_pwd, 'md5' ) ) THEN
        RETURN 'previous passwords cannot be reused';
    END IF;

    -- save the new password
    INSERT INTO t_pwd_history( user_name, pwd_hash )
        VALUES( uid, hash( uid || new_pwd, 'md5' ) );

    RETURN( NULL );
END;

ALTER FUNCTION DBA.f_verify_pwd SET HIDDEN;
GRANT EXECUTE ON DBA.f_verify_pwd TO PUBLIC;
SET OPTION PUBLIC.verify_password_function = 'DBA.f_verify_pwd';

-- All passwords expire in 180 days. Expired passwords can be changed
-- by the user using the NewPassword connection parameter.
ALTER LOGIN POLICY DEFAULT password_life_time = 180;

-- If an application calls the procedure specified by the
-- post_login_procedure option, then the procedure can be used to
-- warn the user that their password is about to expire. In
particular,
-- Interactive SQL calls the post_login_procedure.
ALTER LOGIN POLICY DEFAULT password_grace_time = 30;

```

要关闭该选项，请将其设置为空字符串：

```
SET OPTION PUBLIC.VERIFY_PASSWORD_FUNCTION = ''
```

SQL 语句

SQL 语句执行数据定义和操作任务。

ALTER DBSPACE 语句

更改读/写模式、更改大小或扩展现有数据库空间。

快速链接:

[转至参数](#) (第 356 页)

[转至示例](#) (第 358 页)

[转至用法](#) (第 359 页)

[转至标准](#) (第 359 页)

[转至权限](#) (第 360 页)

语法

```
ALTER DBSPACE dbspace-name
{ ADD new-file-spec [, new-file-spec ... ]
| DROP FILE logical-file-name [, FILE logical-file-name ... ]
| RENAME TO newname | RENAME 'new-file-pathname'
| READONLY | READWRITE
| ONLINE | OFFLINE
| STRIPING{ ON | OFF }
| STRIPESIZEKB size-in-KB
ALTER FILE file-name
{ READONLY | [ FORCE ] READWRITE }
| SIZE file-size [ KB | MB | GB | TB ]
| ADD file-size [ KB | MB | GB | TB | PAGES ] }
RENAME PATH 'new-file-pathname'
RENAME TO newname

new-file-spec - (back to Syntax)
FILE logical-file-name 'file-path' iq-file-opts

iq-file-opts - (back to new-file-spec)
[ [ SIZE ] file-size ]
...[ KB | MB | GB | TB ] ]
[ RESERVE reserve-size [ KB | MB | GB | TB ] ]
```

参数

(返回顶部) (第 356 页)

- **ADD** - 向指定 *dbspace* 中添加一个或多个文件。必须为每个文件指定 *dbfile* 名称和物理文件路径，且该名称和路径必须唯一。您可以向 IQ 主 *dbspace*、IQ 共享临时 *dbspace*、IQ 临时 *dbspace* 或高速缓存 *dbspace* 添加文件。您可以向只读 *dbspace* 中

添加一个文件，但 `dbspace` 仍为只读。您只能以只读模式（`ADD FILE` 的缺省设置）向 `Multiplex` 共享临时 `dbspace` 添加文件。

一个目录 `dbspace` 只能包含一个文件，因此 `ADD FILE` 不可用于目录 `dbspace`。

对于 `RLV dbspace`，仅在 `Simplex` 服务器上使用 `ADD FILE`。不能将文件添加到 `Multiplex RLV dbspace`。

对于高速缓存 `dbspace`，在 `Multiplex` 或 `Simplex` 服务器上使用 `ADD FILE`。

在 `ALTER FILE` 子句中使用，以页、千字节 (KB)、兆字节 (MB)、千兆字节 (GB) 或千吉字节 (TB) 为单位扩展文件的大小。缺省值为 `MB`。仅当空闲列表（一种分配映射）有足够空间且 `dbspace` 有足够保留的空间时，您才能执行 `ADD` 子句。

- **DROP FILE** - 从 `IQ dbspace` 中删除指定文件。该文件必须为空。您不能删除指定 `dbspace` 中的最后一个文件。如果 `dbspace` 中仅包含一个文件，改为使用 `DROP DBSPACE`。
- **RENAME TO** - 与 `DROP FILE` 子句一起使用时，重命名包含单个文件的 `dbspace` 的路径名。从语义上看，它等同于 `RENAME PATH` 子句。如果 `dbspace` 包含多个文件，则系统会返回错误。不能重命名 `IQ_SYSTEM_MAIN`、`IQ_SYSTEM_MSG`、`IQ_SYSTEM_TEMP`、`IQ_SHARED_TEMP` 或 `SYSTEM`。

与 `ALTER FILE` 子句一起使用时，将指定文件的逻辑名重命名为新名称。新名称在数据库中必须唯一。

- **READONLY** - 与 `DROP` 子句一起使用时，将除 `IQ_SYSTEM_MAIN`、`IQ_SYSTEM_TEMP`、`IQ_SYSTEM_MSG`、`IQ_SHARED_TEMP` 和 `SYSTEM` 外的任何 `dbspace` 更改为只读。禁止对当前已分配至 `dbspace` 的任何对象执行 `DML` 修改。只能用于高速缓存 `dbspace` 和 `IQ` 主存储库中的 `dbspace`。

与 `ALTER FILE` 子句一起使用时，将指定文件更改为只读。文件必须与 `IQ` 主 `dbspace` 关联。不能将 `IQ_SHARED_TEMP` 中的文件更改为 `READONLY` 状态。

- **READWRITE** - 与 `DROP FILE` 子句一起使用时，将 `dbspace` 更改为读写。`dbspace` 必须联机。只能用于高速缓存 `dbspace` 和 `IQ` 主存储库中的 `dbspace`。

与 `ALTER FILE` 子句一起使用时，将指定的高速缓存 `dbspace`、`IQ` 主 `dbspace` 或临时存储 `dbfile` 更改为读写。文件必须与高速缓存 `dbspace`、`IQ` 主 `dbspace` 或临时 `dbspace` 关联。

- **ONLINE** - 将脱机 `dbspace` 及其全部关联文件设置为联机状态。只能用于高速缓存 `dbspace` 和 `IQ` 主存储库中的 `dbspace`。
- **OFFLINE** - 将联机只读 `dbspace` 及其全部关联文件设置为脱机状态。（如果 `dbspace` 为读写、已脱机或非高速缓存 `dbspace` 或 `IQ` 主存储库，则系统会返回错误。）只能用于高速缓存 `dbspace` 或 `IQ` 主存储库中的 `dbspace`。
- **STRIPING** - 按照指定条件更改 `dbspace` 的磁盘条带化。当磁盘条带化设置为 `ON` 时，会以轮转调度方式从 `dbspace` 内的每个文件分配数据。例如，所写的第一个

数据库页面将进入第一个文件，所写的第二个页面将进入给定 **dbspace** 内的下一个文件，以此类推。将跳过只读 **dbspace**。

- **STRIPESIZEKB** – 指定磁盘条带化算法移至指定 **dbspace** 的下一条带之前要写入每个文件的字节数 (KB)。
- **FORCE READWRITE** – 与 **ALTER FILE** 子句一起使用时，将指定的共享临时存储 **dbfile** 的状态更改为读写，尽管辅助节点可能会出现已知文件状态问题。该文件可能与 **IQ** 主 **dbspace**、共享临时 **dbspace** 或临时 **dbspace** 相关联，但因为 **IQ_SYSTEM_MAIN** 中的新 **dbfile** 和用户主 **dbspace** 创建为读写，所以此子句只会影响共享临时 **dbspace**。
- **SIZE** – 以千字节 (KB)、兆字节 (MB)、千兆字节 (GB) 或千吉字节 (TB) 为单位指定文件的新大小。缺省单位为兆字节。仅当空闲列表（一种分配映射）有足够空间且 **dbspace** 有足够保留的空间时，您才能增大 **dbspace**。同样，仅当要截断的部分未被使用时，您才能减小 **dbspace**。
- **RENAME PATH** – 与 **ALTER FILE** 子句一起使用时，重命名与指定文件关联的文件路径名。此子句只是将文件与新文件路径关联起来，而不再关联旧路径。子句并不实际更改操作系统文件名。必须通过操作系统才能更改文件名。**dbspace** 必须脱机才能重命名文件路径。当 **dbspace** 更改为联机状态或当数据库重新启动时，将使用新路径。

您不能重命名 **IQ_SYSTEM_MAIN** 中的文件路径，因为如果新路径不可访问，则数据库将无法启动。如果您需要重命名 **IQ_SYSTEM_MAIN** 中文件的路径，请将文件设置为只读，清空并删除该文件，然后以新文件路径名再次添加该文件。请将 **dbfile** 的物理文件路径用单引号引起来。

示例

(返回顶部) (第 356 页)

- **示例 1** – 将名为 **DspHist** 的 **dbspace** 的模式更改为 **READONLY**:

```
ALTER DBSPACE DspHist READONLY
```

- **示例 2** – 通过添加大小为 **500MB** 的文件 **FileHist3**，向 **dbspace DspHist** 添加 **500MB**:

```
ALTER DBSPACE DspHist
ALTER FILE FileHist3 ADD 500MB
```

- **示例 3** – 在 **UNIX** 系统上，向 **dbspace DspHist** 添加两个 **500MB** 文件:

```
ALTER DBSPACE DspHist ADD
FILE FileHist3 '/History1/data/file3' SIZE 500MB,
FILE FileHist4 '/History1/data/file4' SIZE 500
```

- **示例 4** – 将 **dbspace IQ_SYSTEM_TEMP** 的大小增大 **2GB**:

```
ALTER DBSPACE IQ_SYSTEM_TEMP ADD 2 GB
```

- **示例 5** - 从 dbspace DspHist 中删除两个文件。两个文件必须为空：

```
ALTER DBSPACE DspHist
DROP FILE FileHist2, FILE FileHist4
```

- **示例 6** - 将 dbspace IQ_SYSTEM_MAIN 的大小增大 1000 页。（ADD 子句的缺省单位为页）：

```
ALTER DBSPACE IQ_SYSTEM_MAIN ADD 1000
```

- **示例 7** - 将文件添加到高速缓存 dbspace myDAS：

```
ALTER DBSPACE myDAS ADD FILE iqdas2 'sampledb.iqcache' size 1024
```

- **示例 8** - 将 dbfile iqdas2 从高速缓存 dbspace myDAS 移除：

```
ALTER DBSPACE myDAS DROP FILE iqdas2
```

- **示例 9** - 禁用高速缓存 dbspace myDAS：

```
ALTER DBSPACE myDAS OFFLINE
```

- **示例 10** - 将 myDAS 高速缓存 dbspace dbfile iqdas2 设置为只读：

```
ALTER DBSPACE myDAS ALTER FILE iqdas2 READONLY
```

用法

(返回顶部) (第 356 页)

ALTER DBSPACE 更改读写模式，更改联机/脱机状态，更改文件大小，重命名 dbspace 名称、文件逻辑名或文件路径，或设置 dbspace 条带化参数。有关现有 dbspace 的详细信息，请运行 **sp_iqdbspace** 过程、**sp_iqdbspaceinfo** 过程、**sp_iqfile** 过程、**sp_iqdbspaceobjectinfo** 和 **sp_iqobjectinfo**。Dbspace 和 dbfile 名称始终不区分大小写。如果数据库为 CASE RESPECT，且操作系统支持区分大小写的文件，则物理文件路径区分大小写。否则，文件路径不区分大小写。

对于 dbspace 名称和 dbfile 名称，要么不用引号引起来，要么用双引号引起来。

在 Windows 中，如果您指定一个路径，则任何后跟 n 或 x 的反斜杠字符 (\) 都必须双写。这样可防止根据 SQL 中的字符串规则将这些反斜杠解释为换行符 (\n) 或十六进制数 (\x)。更安全的做法是始终使用两个反斜杠。

副作用：

- 自动提交
- 自动执行检查点操作
- 将模式更改为 READONLY 会立即将 dbspace 中的内部数据库结构重新分配到某个读/写 dbspace。

标准

(返回顶部) (第 356 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。

权限

(返回顶部) (第 356 页)

需要 **MANAGE ANY DBSPACE** 系统特权。

ALTER INDEX 语句

重命名基表或全局临时表中的索引、用户显式创建的索引和外键的外键角色名或在目录存储表上更改索引的聚簇性质。不能重命名用于强制执行键约束的索引。

快速链接:

[转至参数 \(第 360 页\)](#)

[转至示例 \(第 361 页\)](#)

[转至用法 \(第 361 页\)](#)

[转至标准 \(第 362 页\)](#)

[转至权限 \(第 362 页\)](#)

语法

```
ALTER { INDEX index-name
      | [ INDEX ] FOREIGN KEY role-name
      | [ INDEX ] PRIMARY KEY
      | ON [owner.]table-name { rename-clause | move-clause | cluster-clause }

rename-clause - (back to Syntax)
  RENAME TO | AS new-name

move-clause - (back to Syntax)
  MOVE TO dbspace-name

cluster-clause - (back to Syntax)
  CLUSTERED | NONCLUSTERED
```

参数

(返回顶部) (第 360 页)

- **ON** - 指定含有要重命名的索引或外键的表的名称。
- **RENAME** - 为索引或外键角色指定新名称。
- **MOVE** - 将指定的索引、唯一约束、外键或主键移动到指定的 **dbspace**。对于唯一约束或外键，您必须指定其唯一索引名称。

- **cluster-clause** - 指定索引应更改为 **CLUSTERED** 还是 **NONCLUSTERED**。仅适用于目录存储表并且一个表上只有一个索引可以是聚簇索引。

示例

(返回顶部) (第 360 页)

- **示例 1** - 将主键 (c5 的 HG) 从 dbspace Dsp4 移动到 Dsp8:

```
CREATE TABLE foo (
    c1 INT IN Dsp1,
    c2 VARCHAR(20),
    c3 CLOB IN Dsp2,
    c4 DATE,
    c5 BIGINT,
    PRIMARY KEY (c5) IN Dsp4) IN Dsp3);
CREATE DATE INDEX c4_date ON foo(c4) IN Dsp5;
ALTER INDEX PRIMARY KEY ON foo MOVE TO Dsp8;
```

- **示例 2** - 将 DATE 索引从 Dsp5 移动到 Dsp9:

```
ALTER INDEX c4_date ON foo MOVE TO Dsp9
```

- **示例 3** - 将表 jal.mytable 中的索引 COL1_HG_OLD 重命名为 COL1_HG_NEW:

```
ALTER INDEX COL1_HG_OLD ON jal.mytable
RENAME AS COL1_HG_NEW
```

- **示例 4** - 将表 dba.Employees 中的外键角色名称 ky_dept_id 重命名为 emp_dept_id:

```
ALTER INDEX FOREIGN KEY ky_dept_id
ON dba.Employees
RENAME TO emp_dept_id
```

用法

(返回顶部) (第 360 页)

您必须对新 dbspace 具有 **CREATE** 特权且成为表所有者，或者具有 **MANAGE ANY DBSPACE** 系统特权。

注意： 如果尝试更改局部临时表中的索引，则系统会返回错误未找到索引。尝试更改非用户创建的索引（如缺省索引 (FP)），将返回错误“无法更改索引。只能更改所有者类型为 **USER** 的基表或全局临时表中的索引。”

副作用：

- 自动提交。清除 **Interactive SQL** 的“结果”窗格中的“结果”选项卡。关闭用于当前连接的所有游标。

标准

(返回顶部) (第 360 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。

权限

(返回顶部) (第 360 页)

move-clause 针对实例化视图需要以下特权之一：

- **MANAGE ANY DBSPACE** 系统特权。
- **ALTER ANY INDEX** 系统特权。
- **ALTER ANY OBJECT** 系统特权。
- 您拥有实例化视图，外加以下特权之一：
 - **CREATE ANY OBJECT** 系统特权。
 - 对目标 dbspace 的 **CREATE** 特权。

move-clause 针对所有其它索引需要以下特权之一：

- **MANAGE ANY DBSPACE** 系统特权。
- **ALTER ANY INDEX** 系统特权。
- **ALTER ANY OBJECT** 系统特权。
- 您拥有基础表或对表具有 **REFERENCES** 特权，外加以下特权之一：
 - **CREATE ANY OBJECT** 系统特权。
 - 对目标 dbspace 的 **CREATE** 特权。

cluster-clause 针对实例化视图需要以下特权之一：

- **ALTER ANY INDEX** 系统特权。
- **ALTER ANY OBJECT** 系统特权。
- 您拥有实例化视图。

cluster-clause 针对所有其它索引需要以下特权之一：

- **ALTER ANY INDEX** 系统特权。
- **ALTER ANY OBJECT** 系统特权。
- 对表的 **REFERENCES** 特权。
- 您拥有该表。

所有其它子句需要以下特权之一：

- **ALTER ANY INDEX** 系统特权。
- **ALTER ANY OBJECT** 系统特权。
- 对表的 **REFERENCES** 特权。

- 您拥有基础表。

ALTER LOGICAL SERVER 语句

修改数据库中现有用户定义逻辑服务器的配置。此语句强制共享系统临时存储设置在逻辑服务器共享的物理节点中保持一致。

快速链接：

[转至参数](#)（第 363 页）

[转至示例](#)（第 363 页）

[转至用法](#)（第 364 页）

[转至权限](#)（第 364 页）

语法

```
ALTER LOGICAL SERVER logical-server-name
    { alter-ls-clause } [ WITH STOP SERVER ]

alter-ls-clause - (back to Syntax)
    { ADD MEMBERSHIP '(' { ls-member, ... } ')' |
      DROP MEMBERSHIP '(' { ls-member, ... } ')' |
      POLICY policy-name }

ls-member - (back to alter-ls-clause)
    FOR LOGICAL COORDINATOR | mpx-server-name
```

参数

[\(返回顶部\)](#)（第 363 页）

- **logical-server-name** - 是指现有的用户定义逻辑服务器名。
- **WITH STOP SERVER** - 在 **TEMP_DATA_IN_SHARED_TEMP** 数据库选项发生直接或间接更改时，会自动关闭逻辑服务器中的所有服务器。

应用于

仅限 Multiplex。

示例

[\(返回顶部\)](#)（第 363 页）

- **示例 1** - 通过将 Multiplex 节点 n1 和 n2 添加到逻辑服务器 ls1 来变更用户定义的逻辑服务器：

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (n1, n2)
```

- **示例 2** - 添加 COORDINATOR 逻辑成员资格，并从逻辑服务器 ls1 中删除当前协调器节点 n1 的指定成员资格：

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (FOR LOGICAL COORDINATOR)
ALTER LOGICAL SERVER ls1 DROP MEMBERSHIP (nl)
```

- **示例 3** – 将逻辑服务器 ls2 的逻辑服务器策略更改为策略 lsp1。

```
ALTER LOGICAL SERVER ls2 POLICY lsp1
```

用法

(返回顶部) (第 363 页)

SYS.ISSYSIQLSMEMBER 系统表存储逻辑服务器成员资格的定义。

当成员节点添加至某一逻辑服务器或从中删除后，该节点仅在运行与 **ALTER LOGICAL SERVER** 相对应的 TLV 日志之后，才会开始或停止接受逻辑服务器连接。从逻辑服务器删除某一节点后，该节点的现有逻辑服务器连接仍在运行，不过这些连接的分布式处理已停止。

出现以下情况时会返回错误：

- 任何通过 ADD MEMBERSHIP 子句指定的 ls-member 已经是逻辑服务器的成员。
- 任何通过 DROP MEMBERSHIP 子句指定的 ls-member 不是逻辑服务器的现有成员。
- 逻辑服务器成员资格的更改导致某个节点属于多个分配给一个登录策略的逻辑服务器。登录策略中的逻辑服务器成员资格不能重叠。

权限

(返回顶部) (第 363 页)

需要 MANAGE MULTIPLEX 系统特权。

ALTER LS POLICY 语句

修改根逻辑服务器策略或用户创建的逻辑服务器策略中的部分或全部选项值。此语句强制共享系统临时存储设置在逻辑服务器共享的物理节点中保持一致。

快速链接：

[转至参数](#) (第 365 页)

[转至示例](#) (第 365 页)

[转至用法](#) (第 365 页)

[转至权限](#) (第 367 页)

语法

```
ALTER LS POLICY ls-policy-name ls-option-value-list
[ WITH STOP SERVER ]
```

ls-option-value-list – (back to Syntax)

```
{ ls-option-name = ls-policy-option-value } ...
```

ls-option-name - (back to *ls-option-value-list*)

```
ALLOW_COORDINATOR_AS_MEMBER
| DQP_ENABLED
| ENABLE_AUTOMATIC_FAILOVER
| LOGIN_REDIRECTION
| REDIRECTION_WAITERS_THRESHOLD
| TEMP_DATA_IN_SHARED_TEMP
```

参数

(返回顶部) (第 364 页)

- **ls-policy-name** - 逻辑服务器策略的名称。指定用于修改根逻辑服务器策略的根。
- **ls-option-value-list** - 逻辑服务器策略选项的名称。请参见选项列表用法 (第 365 页)。
- **ls-policy-option-value** - 任何未指定的选项将从根逻辑服务器策略中继承值。请参见用法 (第 365 页)。
- **WITH STOP SERVER** - 当直接或间接更改 **TEMP_DATA_IN_SHARED_TEMP** 选项时，将自动关闭逻辑服务器中的所有服务器。

应用于

仅限 Multiplex。

示例

(返回顶部) (第 364 页)

- **示例 1** - 修改逻辑服务器策略：

```
ALTER LS POLICY root
ALLOW_COORDINATOR_AS_MEMBER=ON
```

- **示例 2** - 修改逻辑服务器策略，并使服务器在选项值发生更改时自动关闭：

```
ALTER LS POLICY root
TEMP_DATA_IN_SHARED_TEMP=ON WITH STOP SERVER
```

用法

(返回顶部) (第 364 页)

如果您需要一个较小的 **IQ_SYSTEM_TEMP** dbspace，请将 **TEMP_DATA_IN_SHARED_TEMP** 设置为 **ON**，这会将临时数据写入 **IQ_SHARED_TEMP** 而非 **IQ_SYSTEM_TEMP**。但在分布式查询处理环境中，将 **DQP_ENABLED** 和 **TEMP_DATA_IN_SHARED_TEMP** 同时设置为 **ON** 可能由于 **IQ_SHARED_TEMP** 中有更多数据而造成 **SAN** 饱和，而针对 **IQ_SHARED_TEMP** 执行的额外 I/O 操作可能会对 **DQP** 性能造成负面影响。

选项	描述
ALLOW_COORDINATOR_AS_MEMBER	<p>只能为 ROOT 逻辑服务器策略设置。设置为 ON (缺省值) 时，协调器可以是任何用户定义逻辑服务器的成员。OFF 表示阻止协调器用作任何用户定义逻辑服务器的成员。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - ON
DQP_ENABLED	<p>如果设置为 0，不分布查询处理。如果设置为 1 (缺省值)，只要可写共享临时文件存在，便分布查询处理。如果设置为 2，在整个网络上分布查询处理，不使用共享的临时存储库。</p> <ul style="list-style-type: none"> 值 - 0, 1, 2 缺省值 - 1
ENABLE_AUTOMATIC_FAILOVER	<p>只能为 ROOT 逻辑服务器策略设置。当设置为 ON 时，将启用由指定登录策略控制的逻辑服务器的自动故障转移。当设置为 OFF (缺省值) 时，禁用逻辑服务器级别的自动故障转移，允许手动故障转移。指定 DEFAULT 以设置回缺省值。</p> <ul style="list-style-type: none"> 值 - ON、OFF、DEFAULT 缺省值 - OFF
LOGIN_REDIRECTION	<p>当设置为 ON 时，将启用由指定登录策略控制的逻辑服务器的登录重定向。当设置为 OFF (缺省值) 时，禁用逻辑服务器级别的登录重定向，允许外部连接管理。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF
REDIRECTION_WAITERS_THRESHOLD	<p>指定在 SAP Sybase IQ 将与此逻辑服务器的连接重定向到其它服务器之前可排队的连接数。可以是任意整数；缺省值为 5。</p> <ul style="list-style-type: none"> 值 - 整数 缺省值 - 5
TEMP_DATA_IN_SHARED_TEMP	<p>设置为 ON 时，所有临时表数据和符合条件的临时数据都将写入共享临时存储库中，前提是共享临时存储库中至少添加了一个读写文件。必须在设置此选项之后或在向共享临时存储库添加读写文件之后重新启动所有 Multiplex 节点。（如果共享临时存储库不包含任何读写文件，或者如果您不重新启动节点，数据会改为写入 IQ_SYSTEM_TEMP 中。）当设置为 OFF (缺省值) 时，所有临时表数据和临时数据都会写入到局部临时存储库中。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF

权限

(返回顶部) (第 364 页)

需要 MANAGE MULTIPLEX 系统特权。

LS 策略选项

可用于根 LS 策略和用户定义 LS 策略的选项。

选项	描述
ALLOW_COORDINATOR_AS_MEMBER	<p>只能为 ROOT 逻辑服务器策略设置。设置为 ON (缺省值) 时，协调器可以是任何用户定义逻辑服务器的成员。OFF 表示阻止协调器用作任何用户定义逻辑服务器的成员。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - ON
DQP_ENABLED	<p>如果设置为 0，不分布查询处理。如果设置为 1 (缺省值)，只要可写共享临时文件存在，便分布查询处理。如果设置为 2，在整个网络上分布查询处理，不使用共享的临时存储库。</p> <ul style="list-style-type: none"> 值 - 0、1、2 缺省值 - 1
LOGIN_REDIRECTION	<p>当设置为 ON 时，将启用由指定登录策略控制的逻辑服务器的登录重定向。当设置为 OFF (缺省值) 时，禁用逻辑服务器级别的登录重定向，允许外部连接管理。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF
REDIRECTION_WAITERS_THRESHOLD	<p>指定在 SAP Sybase IQ 将与此逻辑服务器的连接重定向到其它服务器之前可排队的连接数。可以是任意整数；缺省值为 5。</p> <ul style="list-style-type: none"> 值 - 整数 缺省值 - 5
TEMP_DATA_IN_SHARED_TEMP	<p>设置为 ON 时，所有临时表数据和符合条件的临时数据都将写入共享临时存储库中，前提是共享临时存储库中至少添加了一个读写文件。必须在设置此选项之后或在向共享临时存储库添加读写文件之后重新启动所有 Multiplex 节点。（如果共享临时存储库不包含任何读写文件，或者如果您不重新启动节点，数据会改为写入 IQ_SYSTEM_TEMP 中。）当设置为 OFF (缺省值) 时，所有临时表数据和临时数据都会写入到局部临时存储库中。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF

ALTER TABLE 语句

修改表的定义。

快速链接：

[转至参数](#)（第 370 页）

[转至示例](#)（第 376 页）

[转至用法](#)（第 379 页）

[转至标准](#)（第 379 页）

[转至权限](#)（第 379 页）

语法

语法 1 - Alter Owner

```
ALTER TABLE table_name ALTER OWNER TO new_owner
[ { PRESERVE | DROP } PERMISSIONS ]
[ { PRESERVE | DROP } FOREIGN KEYS ]
```

语法 2

```
ALTER TABLE [ owner. ] table-name
| { ENABLE | DISABLE } RLV STORE
{ alter-clause, ... }
```

alter-clause - (back to Syntax 2)

ADD *create-clause*

```
| ALTER column-name column-alteration
| ALTER [ CONSTRAINT constraint-name ] CHECK ( condition )
| DROP drop-object
| RENAME rename-object
| move-clause
| SPLIT PARTITION range-partition-name
| INTO ( range-partition-decl, range-partition-decl )
| MERGE PARTITION partition-name-1 INTO partition-name-2
| UNPARTITION
| PARTITION BY
| range-partitioning-scheme
| hash-partitioning-scheme
| composite-partitioning-scheme composite-partitioning-scheme
```

create-clause - (back to *alter-clause*)

```
column-name column-definition [ column-constraint ]
| table-constraint
| [ PARTITION BY ] range-partitioning-scheme
```

column definition - (back to *create-clause*)

```
column-name data-type [ NOT NULL | NULL ]
[ IN dbspace-name ]
[ DEFAULT default-value | IDENTITY ]
```



```

column-constraint - (back to create-clause)
[ CONSTRAINT constraint-name ]
{ UNIQUE
  | PRIMARY KEY
  | REFERENCES table-name [ ( column-name ) ] [ actions ]
  | CHECK ( condition )
  | IQ UNIQUE ( integer )
}

table-constraint - (back to create-clause)
[ CONSTRAINT constraint-name ]
{ UNIQUE ( column-name [ , ... ] )
  | PRIMARY KEY ( column-name [ , ... ] )
  | foreign-key-constraint
  | CHECK ( condition )
}

foreign-key-constraint - (back to table-constraint)
FOREIGN KEY [ role-name ] [ ( column-name [ , ... ] ) ]
... REFERENCES table-name [ ( column-name [ , ... ] ) ]
... [ actions ]

actions - (back to foreign-key-constraint)
[ ON { UPDATE | DELETE } { RESTRICT } ]

column-alteration - (back to alter-clause)
{ column-data-type | alterable-column-attribute } [ alterable-column-attribute ... ]

  | ADD [ constraint-name ] CHECK ( condition )
  | DROP { DEFAULT | CHECK | CONSTRAINT constraint-name }

alterable-column-attribute - (back to column-alteration)
[ NOT ] NULL
| DEFAULT default-value
| [ CONSTRAINT constraint-name ] CHECK { NULL | ( condition )
}

default-value - (back to alterable-column-attribute)
CURRENT { DATABASE | DATE | REMOTE USER | TIME | TIMESTAMP | USER |
PUBLISHER )
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function ( constant-expression )
| AUTOINCREMENT
| NULL
| TIMESTAMP
| LAST USER
| USER

drop-object - (back to alter-clause)
{ column-name
| CHECK constraint-name
| CONSTRAINT

```

```

| UNIQUE ( index-columns-list )
| PRIMARY KEY
| FOREIGN KEY fkey-name
| [ PARTITION ] range-partition-name
}

rename-object - (back to alter-clause)
  new-table-name
  | column-name TO new-column-name
  | CONSTRAINT constraint-name TO new-constraint-name
  | [ PARTITION ] range-partition-name TO new-range-partition-name

move-clause - (back to alter-clause)
{ ALTER column-name
  MOVE
  { PARTITION ( range-partition-name TO new-dbspace-name)
    | TO new-dbspace-name }
  }
  | MOVE PARTITION range-partition-name TO new-dbspace-name
  | MOVE TO new-dbspace-name
  | MOVE TABLE METADATA TO new-dbspace-name
}

```

```

range-partitioning-scheme - (back to alter-clause)
  RANGE ( partition-key )
  ( range-partition-decl [, range-partition-decl ...] )

partition-key - (back to range-partitioning-scheme)
  column-name

range-partition-decl - (back to alter-clause) or (back to range-
partitioning-scheme)
  range-partition-name VALUES <= ( { constant | MAX } ) [ IN dbspace-
name ]

hash-partitioning-scheme - (back to alter-clause) or (back to composite-
partitioning-scheme)
  HASH ( partition-key, ... ] )

composite-partitioning-scheme - (back to alter-clause)
  hash-partitioning-scheme SUBPARTITION range-partitioning-scheme

```

参数

(返回顶部) (第 368 页)

- **{ ENABLE | DISABLE } RLV STORE** - 将该表注册到 RLV 存储库以进行实时内存更新。不支持 IQ 临时表，也不能在 Multiplex 环境中运行。此值替代数据库选项 **BASE_TABLES_IN_RLV** 的值。
- **ADD column-definition [column-constraint]** - 向表中添加新列。

表必须为空才可指定 NOT NULL。添加 IDENTITY 或 DEFAULT AUTOINCREMENT 列时，表可能包含数据。如果列有缺省 IDENTITY 值，新列的所有行将用连续值

来填充。也可添加 FOREIGN 约束作为单列键的列约束。IDENTITY/DEFAULT AUTOINCREMENT 列的值唯一标识表中的每一行。

IDENTITY/DEFAULT AUTOINCREMENT 列存储在插入和更新期间自动生成的顺序编号。DEFAULT AUTOINCREMENT 列也称为 IDENTITY 列。使用 IDENTITY/DEFAULT AUTOINCREMENT 时，列必须是小数位数为 0 的整数数据类型之一或精确的数值类型。有关列约束和 IDENTITY/DEFAULT AUTOINCREMENT 列的详细信息，请参见 CREATE TABLE 语句。

IQ UNIQUE 约束 - 定义列的预期基数并确定是将该列装载为 Flat FP 还是 NBit FP。IQ UNIQUE(*n*) 值显式设置为 0 会将该列装载为 Flat FP。不带 IQ UNIQUE 约束的列将隐式装载为 NBit，一直到 FP_NBIT_AUTOSIZE_LIMIT、FP_NBIT_LOOKUP_MB 和 FP_NBIT_ROLLOVER_MAX_MB 选项定义的限制。

不必使用 *n* 值小于 FP_NBIT_AUTOSIZE_LIMIT 的 IQ UNIQUE。自动大小调整功能会自动将所有中低基数列的大小调整为 NBit。如果想要将列装载为 Flat FP，或者想要在不同值的数量超出 FP_NBIT_AUTOSIZE_LIMIT 时将列装载为 NBit，请使用 IQ UNIQUE。

注意:

- 当指定高 IQ UNIQUE 值时，请考虑内存的使用情况。如果计算机资源受到限制，应避免带有 FP_NBIT_ENFORCE_LIMITS='OFF' (缺省设置) 的装载。
在 SAP Sybase IQ 16.0 之前，IQ UNIQUE *n* 值 > 16777216 时会切换到 Flat FP。在 16.0 中，支持对较大的 IQ UNIQUE 值进行标识化，但鉴于基数和列宽，该值可能需要非常多的内存资源。
 - BIT、BLOB 和 CLOB 数据类型不支持 NBit 字典压缩。如果 FP_NBIT_IQ15_COMPATIBILITY='OFF'，则在包含这些数据类型的 CREATE TABLE 或 ALTER TABLE 语句中指定的非零 IQ UNIQUE 列将返回一个错误。
-

• ALTER *column-name* column-alteration - 更改列定义:

- **SET DEFAULT** *default-value* - 更改表中现有列的缺省值。您也可以为此任务使用 MODIFY 子句，但 ALTER 与 ISO/ANSI SQL 兼容，而 MODIFY 与其不兼容。修改缺省值不会更改表中的任何现有值。
- **DROP DEFAULT** - 删除表中现有列的缺省值。您也可以为此任务使用 MODIFY 子句，但 ALTER 与 ISO/ANSI SQL 兼容，而 MODIFY 与其不兼容。删除缺省值不会更改表中的任何现有值。
- **ADD** - 在列中添加已命名约束或 CHECK 条件。新约束或条件仅在定义后应用于对表的操作。但不会对表中现有的值进行校验来确认其是否满足新约束或条件。
- **CONSTRAINT** *column-constraint-name* - 可选列约束名用于以后单独修改或删除约束，而不必修改整个列的约束。

- **[CONSTRAINT *constraint-name*] CHECK (*condition*)** – 此子句用于在列上添加 CHECK 约束。
- **SET COMPUTE (*expression*)** – 更改与计算列关联的表达式。当执行语句时，重新计算列中的值，如果新表达式无效，则语句执行失败。
- **DROP COMPUTE** – 将列从计算列更改为非计算列。此语句不更改表中的任何现有值。
- **ADD table-constraint** – 向表中添加约束。

也可单列键或多列键添加外键约束作为表约束。如果指定 **PRIMARY KEY**，表不得具有通过 **CREATE TABLE** 语句或其它 **ALTER TABLE** 语句创建的主键。有关表约束的完整说明，请参见 **CREATE TABLE** 语句。

注意： 不能通过 **MODIFY** 修改表约束或列约束。要更改约束，必须通过 **DELETE** 删除旧约束并通过 **ADD** 添加新约束。

- **DROP drop-object** – 删除表对象：
 - **DROP column-name** – 从表中删除列。如果列包含在任何多列索引、唯一性约束、外键或主键中，则必须删除索引、约束或键后才能删除列。这不会删除引用该列的 CHECK 约束。只有在 **IDENTITY_INSERT** 处于关闭状态并且表不是局部临时表的情况下，才能删除 **IDENTITY/DEFAULT AUTOINCREMENT** 列。
 - **DROP CHECK** – 删除表的所有检查约束。这包括表检查约束和列检查约束。
 - **DROP CONSTRAINT *constraint-name*** – 删除表或指定列的已命名约束。
 - **DROP UNIQUE (*column-name*, ...)** – 删除指定列上的唯一约束。同时也删除引用该唯一约束的任何外键（而不是主键）。如果有关联的外键约束，系统会报告错误。在删除主键约束之前，使用 **ALTER TABLE** 删除所有引用该主键的外键。
 - **DROP PRIMARY KEY** – 删除主键。同时也删除引用此表主键的所有外键。如果有关联的外键约束，系统会报告错误。如果主键是未实施的主键，并且存在关联的未实施的外键约束，则 **DELETE** 会返回错误。
 - **DROP FOREIGN KEY *role-name*** – 删除该表的具有给定角色名称的外键约束。系统将保留为该外键约束隐式创建的非唯一 **HG** 索引。用户可以使用 **DROP INDEX** 语句显式删除 **HG** 索引。
 - **DROP [PARTITION]** – 删除指定分区。分区 **p1** 中的行将被删除，分区定义也会被删除。不能删除最后一个分区，因为删除最后一个分区会将分区表转变为未分区的表。（要合并分区表，请改用 **UNPARTITION** 子句。）例如：

```
CREATE TABLE foo (c1 INT, c2 INT)
PARTITION BY RANGE (c1)
(P1 VALUES <= (100) IN dbbsp1,
 P2 VALUES <= (200) IN dbbsp2,
 P3 VALUES <= (MAX) IN dbbsp3
) IN dbbsp4);
LOAD TABLE ...
ALTER TABLE DROP PARTITION P1;
```

- **RENAME *rename-object*** – 重命名表中的对象：
 - **RENAME *new-table-name*** – 将表名更改为 *new-table-name*。必须修改任何使用旧表名的应用程序。此外，系统自动分配了旧表名的任何外键都不会更改名称。
 - **RENAME *column-name* TO *new-column-name*** – 将列名更改为 *new-column-name*。必须修改任何使用旧列名的应用程序。
 - **RENAME [PARTITION]** – 重命名现有分区。
 - **RENAME *constraint-name* TO *new-constraint-name*** – 将约束名更改为 *new-constraint-name*。必须修改任何使用旧约束名的应用程序。
- **MOVE clause** – 移动表对象。表对象只能驻留在一个 **dbspace** 内。任何类型的 ALTER MOVE 都会阻止在整个移动期间内对表执行任何修改。

注意： 无法将对象移动到高速缓存 **dbspace**。

- **MOVE TO** – 当表映射至新的 **dbspace** 时，移动同一 **dbspace** 内驻留的所有表对象，其中包括列、索引、唯一约束、主键、外键和元数据。不能在分区表上请求 ALTER Column MOVE TO 子句。
无法将 BIT 数据类型列显式置于 **dbspace** 中。BIT 数据类型不支持以下内容：

```
ALTER TABLE t2 alter c1_bit MOVE TO iq_main;
```
- **MOVE TABLE METADATA** – 将表的元数据移动到新的 **dbspace**。对于分区表，MOVE TABLE METADATA 还会移动各分区之间共享的元数据。
- **MOVE PARTITION** – 将指定分区移动到新的 **dbspace**。
- **PARTITION BY** – 将大表拆分为多个便于管理的小存储对象。各个分区共用父表的相同逻辑属性，但可放置在单独的 **dbspace** 中并分别进行管理。SAP Sybase IQ 支持多个表分区模式：
 - 散列分区
 - 范围分区
 - 复合分区

分区键是包含表分区键的一个或多个列。分区键可包含 NULL 和 DEFAULT 值，但不能包含：

- LOB (BLOB 或 CLOB) 列
- BINARY 或 VARBINARY 列
- 长度超过 255 个字节的 CHAR 或 VARCHAR 列
- BIT 列
- FLOAT/DOUBLE/REAL 列
- **PARTITION BY RANGE** – 根据分区列中的值范围对行进行分区。范围分区被限定为有 1 个分区键列和最多 1024 个分区。在范围分区模式中，分区键是包含表分区键的列：

```
range-partition-decl:
  partition-name VALUES <= ( {constant-expr | MAX } [ ,
```

```
{ constant-expr | MAX }]]... )
[ IN dbspace-name ]
```

分区名称是存储表行的新分区的名称。在表分区集合中，分区名称必须唯一。分区名称是必需项。

- **VALUE** – 按升序为每个分区指定上限（含上限）。用户必须为每个范围分区指定分区条件，确保将每一行只分配到一个分区。分区列允许 NULL 值，将 NULL 作为分区键值的行属于第一个表分区。但 NULL 不能是界限值。

第一个分区没有下限（MIN 值）。分区键第一列中单元格值为 NULL 的行属于第一个分区。对于最后一个分区，您可以指定一个上限（含上限）或 MAX。如果最后一个分区上限值不是 MAX，则加载或插入其分区键值大于最后一个分区上限值的任何行都将生成错误。

- **Max** – 表示无限大的上限，只能为最后一个分区指定。
- **IN** – 在 partition-decl 中指定其中应该包含分区行的 dbspace。

以下限制影响范围分区表的分区键和界限值：

- 只有所有现有行都属于第一分区，才能对未分区表进行范围分区。
- 分区界限必须是常量，而不是常量表达式。
- 分区界限必须按照分区的创建顺序以升序排列。也就是说，第二个分区上限必须大于第一个分区上限，依此类推。

此外，分区界限值必须与相应的分区键列数据类型兼容。例如，VARCHAR 与 CHAR 兼容。

- 如果某个界限值的数据类型不同于其对应分区键列的数据类型，SAP Sybase IQ 会将界限值的数据类型转换为分区键列的数据类型，但存在以下例外情况：
- 不允许进行显式转换。以下示例尝试将 INT 显式转换为 VARCHAR，并生成错误：

```
CREATE TABLE Employees(emp_name VARCHAR(20))
PARTITION BY RANGE(emp_name)
(p1 VALUES <=(CAST (1 AS VARCHAR(20))),
p2 VALUES <= (CAST (10 AS VARCHAR(20)))
```

- 不允许执行可导致数据丢失的隐式转换。在以下示例中，分区界限与分区键类型不兼容。舍入假设可导致数据丢失，并生成错误：

```
CREATE TABLE emp_id (id INT) PARTITION BY RANGE(id) (p1 VALUES
<= (10.5), p2 VALUES <= (100.5))
```

- 在以下示例中，分区界限与分区键数据类型兼容。界限值会直接转换为浮点值。无需进行舍入处理，且支持转换：

```
CREATE TABLE id_emp (id FLOAT)
PARTITION BY RANGE(id) (p1 VALUES <= (10),
p2 VALUES <= (100))
```

- 不允许将非二进制数据类型转换为二进制数据类型。例如，不允许进行以下转换，并会返回错误：

```
CREATE TABLE newemp (name BINARY)
PARTITION BY RANGE(name)
```

```
(p1 VALUES <= ("Maarten"),
p2 VALUES <= ("Zymmerman"))
```

- 在范围分区表中，不能将 **NULL** 用作界限。
- 如果分区键第一列中的单元格值计算结果为 **NULL**，则对应行将属于第一个分区。**SAP Sybase IQ** 仅支持一列分区键，因此分区键中的任何 **NULL** 都会将对应行分配到第一个分区。
- **PARTITION BY HASH** – 基于内部散列函数处理的分区-键值向各个分区映射数据。散列分区键被限定为最多有 8 个列，其声明的列宽总和小于等于 5300 个字节。对于散列分区，表创建者仅确定分区键列；分区的数量和位置将在内部确定。

在散列分区声明中，分区键是一个或多个列，其合成值将确定存储有各行数据的分区：

```
hash-partitioning-scheme:
HASH ( partition-key [ , partition-key, ... ] )
```

- **限制** –
 - 只能对基表进行散列分区。试图对全局临时表或局部临时表进行分区将会引发错误。
 - 只能对空的未分区表进行散列分区。
 - 不能添加、删除、合并或拆分散列分区。
 - 不能为散列分区键添加列或从中删除列。
- **PARTITION BY HASH RANGE** – 按范围对已进行散列分区的表划分子分区。在散列范围分区模式声明中，**SUBPARTITION BY RANGE** 子句将向现有散列范围分区表添加一个新的范围子分区：

```
hash-range-partitioning-scheme:
PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
  [ SUBPARTITION BY RANGE ( range-partition-decl [ , range-
partition-decl ... ] ) ]
```

散列分区指定如何在逻辑上分布和托管数据；范围子分区指定如何在物理上放置数据。新的范围子分区通过散列进行逻辑分区，其散列分区键与现有的散列范围分区表相同。范围子分区键被限定为仅一列。

- **限制** –
 - 只能对基表进行散列分区。试图对全局临时表或局部临时表进行分区将会引发错误。
 - 只有散列分区表为空表时才能按范围对其进行子分区。
 - 不能添加、删除、合并或拆分散列分区。
 - 不能为散列分区键添加列或从中删除列。

注意： 与散列范围分区相同，范围分区和复合分区模式要求有单独授权的 VLDB 管理组件。

- **MERGE PARTITION** – 将 *partition-name-1* 合并到 *partition-name-2*。如果两个分区相邻且数据驻留在同一个 **dbspace** 内，则可合并这两个分区。只能将分区值较低

的分区合并到分区值较高的相邻分区内。请注意，服务器不会检查分区所合并到的 dbspace 的 CREATE 特权。有关如何创建相邻分区的示例，请参见 CREATE TABLE 语句示例。

- **RENAME PARTITION** – 重命名现有 PARTITION。
- **UNPARTITION** – 从分区表中删除分区。每一列都置于单个 dbspace 内。请注意，服务器不会检查所有分区数据将移动到的目标 dbspace 的 CREATE 特权。ALTER TABLE UNPARTITION 会阻塞所有数据库活动。
- **ALTER OWNER** – 更改表的所有者。ALTER OWNER 子句不能与 ALTER TABLE 语句的任何其它 [alter-clause] 子句配合使用。
 - **[PRESERVE | DROP] PERMISSIONS** – 如果不希望新的所有者拥有与旧所有者相同的特权，则可使用 DROP 特权子句（缺省）来删除允许用户访问该表的所有显式授予的特权。隐式授予该表所有者的特权将被授予新的所有者，并从旧的所有者删除这些特权。
 - **[PRESERVE | DROP] FOREIGN KEYS** – 如果想防止新的所有者访问被引用表中的数据，可使用 DROP FOREIGN KEYS 子句（缺省）来删除该表中的所有外键以及引用该表的所有外键。除非所有引用表都属于新的所有者，否则无法将 PRESERVE FOREIGN KEYS 子句与 DROP PERMISSIONS 子句配合使用。

在以下情况下，ALTER TABLE ALTER OWNER 语句会失败：

- 存在另一个与原始表同名的表，并且该表属于新用户。
- 同时指定了 PRESERVE FOREIGN KEYS 和 PRESERVE PERMISSIONS 子句，并且有一个外键所属的用户不是引用该表的新表所有者，而且该外键依赖于隐式授予的特权（如授予表所有者的特权）。为了避免语句失败，可将 SELECT 特权显式授予引用表的原始所有者，或者删除外键。
- 指定了 PRESERVE FOREIGN KEYS 子句，但未指定 PRESERVE PERMISSIONS 子句，并且有一个外键所属的用户不是引用该表的新表所有者。为了避免语句失败，可删除外键。
- 指定了 PRESERVE FOREIGN KEYS 子句，并且表中包含一个依赖于隐式授予的特权（如授予表所有者的特权）的外键。为了避免语句失败，可将被引用表的 SELECT 特权显式授予新所有者，或者删除外键。
- 表包含一个具有缺省值且引用序列的列，而序列生成器的 USAGE 特权依赖于隐式授予的特权（如授予序列所有者的特权）。为了避免语句失败，可将序列生成器的 USAGE 特权显式授予表的新所有者。
- 存在依赖于原始表的已启用实例化视图。

示例

(返回顶部) (第 368 页)

- **示例 1** – 向 Employees 表添加一个新列，说明他们在哪个办公室工作：


```
ALTER TABLE Employees
ADD office CHAR(20)
```

- **示例 2** - 从 Employees 表中删除 office 列:

```
ALTER TABLE Employees
DROP office
```

- **示例 3** - 向 Customers 表添加一列, 用于为每个客户指派一个销售联系人:

```
ALTER TABLE Customers
ADD SalesContact INTEGER
REFERENCES Employees (EmployeeID)
```

- **示例 4** - 向 Customers 表添加一个新列 CustomerNum, 并指定缺省值 88:

```
ALTER TABLE Customers
ADD CustomerNum INTEGER DEFAULT 88
```

- **示例 5** - 将 c2、c4 和 c5 的 **FP** 索引从 dbspace Dsp3 移动到 Dsp6。c1 的 **FP** 索引保留在 Dsp1 中。c3 的 **FP** 索引保留在 Dsp2 中。c5 的主键保留在 Dsp4 中。**DATE** 索引 c4_date 保留在 Dsp5 中。

```
CREATE TABLE foo (
    c1 INT IN Dsp1,
    c2 VARCHAR(20),
    c3 CLOB IN Dsp2,
    c4 DATE,
    c5 BIGINT,
    PRIMARY KEY (c5) IN Dsp4) IN Dsp3);

CREATE DATE INDEX c4_date ON foo(c4) IN Dsp5;
ALTER TABLE foo
    MOVE TO Dsp6;
```

- **示例 6** - 只将 **FP** 索引 c1 从 dbspace Dsp1 移动到 Dsp7:

```
ALTER TABLE foo ALTER c1 MOVE TO Dsp7
```

- **示例 7** - 使用多个 **ALTER TABLE** 子句对分区进行移动、拆分、重命名和合并。

创建分区表:

```
CREATE TABLE bar (
    c1 INT,
    c2 DATE,
    c3 VARCHAR(10))
    PARTITION BY RANGE(c2)
    (p1 VALUES <= ('2005-12-31') IN dbbsp1,
    p2 VALUES <= ('2006-12-31') IN dbbsp2,
    p3 VALUES <= ('2007-12-31') IN dbbsp3,
    p4 VALUES <= ('2008-12-31') IN dbbsp4);
INSERT INTO bar VALUES(3, '2007-01-01', 'banana nut');
INSERT INTO bar VALUES(4, '2007-09-09', 'grape jam');
INSERT INTO bar VALUES(5, '2008-05-05', 'apple cake');
```

将分区 p2 移动到 dbbsp5:

```
ALTER TABLE bar MOVE PARTITION p2 TO DBSP5;
```

将分区 p4 拆分为 2 个分区：

```
ALTER TABLE bar SPLIT PARTITION p4 INTO
  (P41 VALUES <= ('2008-06-30') IN dbsp4,
   P42 VALUES <= ('2008-12-31') IN dbsp4);
```

下面的 **SPLIT PARTITION** 报告一个错误，因为它需要移动数据。拆分后，并非所有现有行都位于同一分区内。

```
ALTER TABLE bar SPLIT PARTITION p3 INTO
  (P31 VALUES <= ('2007-06-30') IN dbsp3,
   P32 VALUES <= ('2007-12-31') IN dbsp3);
```

会报告以下错误：

```
No data move is allowed, cannot split partition p3.
```

下面的 **SPLIT PARTITION** 报告一个错误，因为它更改了分区的边界值：

```
ALTER TABLE bar SPLIT PARTITION p2 INTO
  (p21 VALUES <= ('2006-06-30') IN dbsp2,
   P22 VALUES <= ('2006-12-01') IN dbsp2);
```

会报告以下错误：

```
Boundary value for the partition p2 cannot be changed.
```

将分区 p3 合并到 p2。由于不允许将边界值较高的分区合并到边界值较低的分区，因此，系统会报告错误。

```
ALTER TABLE bar MERGE PARTITION p3 into p2;
```

会报告以下错误：

```
Partition 'p2' is not adjacent to or before partition 'p3'.
```

将分区 p2 合并到 p3：

```
ALTER TABLE bar MERGE PARTITION p2 INTO P3;
```

将分区 p1 重命名为 p1_new：

```
ALTER TABLE bar RENAME PARTITION p1 TO p1_new;
```

对表 bar 取消分区：

```
ALTER TABLE bar UNPARTITION;
```

对表 bar 进行分区。此命令会报告错误，因为所有行必须位于第一个分区中。

```
ALTER TABLE bar PARTITION BY RANGE(c2)
  (p1 VALUES <= ('2005-12-31') IN dbsp1,
   P2 VALUES <= ('2006-12-31') IN DBSP2,
   P3 VALUES <= ('2007-12-31') IN dbsp3,
   P4 VALUES <= ('2008-12-31') IN dbsp4);
```

会报告以下错误：

```
All rows must be in the first partition.
```

对表 bar 进行分区：

```
ALTER TABLE bar PARTITION BY RANGE(c2)
  (p1 VALUES <= ('2008-12-31') IN dbbsp1,
   p2 VALUES <= ('2009-12-31') IN dbbsp2,
   p3 VALUES <= ('2010-12-31') IN dbbsp3,
   p4 VALUES <= ('2011-12-31') IN dbbsp4);
```

- **示例 8** - 更改表 tab1，使其不再注册到 RLV 存储库中以进行实时内存更新。

```
ALTER TABLE tab1 DISABLE RLV STORE
```

用法

(返回顶部) (第 368 页)

ALTER TABLE 语句可更改以前创建的表中的表属性（列定义和约束）。该语法允许使用一组变更子句；但在每个 **ALTER TABLE** 语句中，只能添加、修改或删除一个表约束或列约束。只要 **ALTER TABLE** 语句影响了当前正由其它连接使用的表，就会禁止该语句。**ALTER TABLE** 可能很耗时，并且在处理语句时，服务器不会处理引用相同表的请求。

注意： 您不能更改本地临时表，但可以在仅有一个连接使用全局临时表时，更改全局临时表。

SAP Sybase IQ 强制执行 **REFERENCES** 和 **CHECK** 约束。只有在满足以下条件时，才会评估添加到 **ALTER TABLE** 语句的表和/或列检查约束：变更表操作过程中在所添加的其中一个新列中定义这些约束。有关 **CHECK** 约束的详细信息，请参见“**CREATE TABLE** 语句”。

如果在视图定义中使用 **SELECT ***，并且变更 **SELECT *** 所引用的表，则您必须运行 **ALTER VIEW <viewname> RECOMPILE** 以确保视图定义正确，并防止在查询该视图时出现意外的结果。

副作用：

- 自动提交。**ALTER** 和 **DROP** 选项用于关闭当前连接的所有游标。Interactive SQL 数据窗口同时也会被清除。
- **ALTER TABLE** 操作开始时执行检查点。
- 一旦更改了列或表，引用更改列的任何存储过程、视图或其它项将不再有效。

标准

(返回顶部) (第 368 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 一些子句受 SAP Adaptive Server® Enterprise 的支持。

权限

(返回顶部) (第 368 页)

句法 1

需要以下特权之一：

- ALTER ANY TABLE 系统特权
- ALTER ANY OBJECT 系统特权
- 表的 ALTER 特权
- 您拥有该表

句法 2

语法 1 所需的系统特权因使用的子句而异。

子句	所需特权
Add	<p>需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY TABLE 系统特权• ALTER ANY OBJECT 系统特权• 基础表的 ALTER 特权• 您拥有基础表 <p>UNIQUE、PRIMARY KEY、FOREIGN KEY 或 IQ UNIQUE 列约束 – 需要以上特权，外加基础表的 REFERENCES 特权。</p> <p>FOREIGN KEY 表约束需要以上特权，外加以下特权之一：</p> <ul style="list-style-type: none">• CREATE ANY INDEX 系统特权• CREATE ANY OBJECT 系统特权• 基表的 REFERENCES 特权 <p>PARTITION BY RANGE 需要以上特权，外加下列特权之一：</p> <ul style="list-style-type: none">• CREATE ANY OBJECT 系统特权• 对要创建分区的 dbspace 的 CREATE 特权
Alter	<p>需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY TABLE 系统特权• ALTER ANY OBJECT 系统特权• 表的 ALTER 特权• 您拥有该表。 <p>变更主键或唯一约束还需要表的 REFERENCES 特权。</p>

子句	所需特权
Drop	<p>删除不含约束的列 - 需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY OBJECT 系统特权• ALTER ANY TABLE 系统特权• 基础表的 ALTER 特权• 您拥有基础表 <p>删除含有约束的列或表需要以上特权以及 REFERENCES 特权（如果使用 ALTER 特权）。</p> <p>删除自身拥有的表分区 - 无需任何权限。</p> <p>删除其他用户拥有的表分区 - 需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY TABLE 系统特权• ALTER ANY OBJECT 系统特权• 表的 ALTER 特权
RENAME	<p>需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY TABLE 系统特权• ALTER ANY OBJECT 系统特权• 表的 ALTER 特权• 您拥有该表
Move	<p>需要以下特权之一：</p> <ul style="list-style-type: none">• ALTER ANY TABLE 系统特权• ALTER ANY OBJECT 系统特权• 系统特权• 基础表的 ALTER 特权• 您拥有基础表 <p>另外，还需要以下特权之一：</p> <ul style="list-style-type: none">• CREATE ANY OBJECT 系统特权• 分区要移动到的目标 dbspace 的 CREATE 特权

子句	所需特权
Split Partition	<p>将自身拥有的表分区 - 无需任何权限。</p> <p>将其他用户拥有的表分区 - 需要以下特权之一：</p> <ul style="list-style-type: none"> • SELECT ANY TABLE 系统特权 • 表的 SELECT 特权 <p>另外，还需要以下特权之一：</p> <ul style="list-style-type: none"> • ALTER ANY TABLE 系统特权 • ALTER ANY OBJECT 系统特权 • 表的 ALTER 特权
Merge Partition、Unpartition	<p>自身拥有的表 - 无需任何权限。</p> <p>其他用户拥有的表 - 需要以下特权之一：</p> <ul style="list-style-type: none"> • ALTER ANY TABLE 系统特权 • ALTER ANY OBJECT 系统特权 • 表的 ALTER 特权
Partition By	<p>需要以下特权之一：</p> <ul style="list-style-type: none"> • CREATE ANY OBJECT 系统特权 • 对要创建分区的 dbspace 的 CREATE 特权 <p>另外，还需要以下特权之一：</p> <ul style="list-style-type: none"> • ALTER ANY TABLE 系统特权 • ALTER ANY OBJECT 系统特权 • 表的 ALTER 特权 • 您拥有该表
或 disable RLV store	<p>需要以下特权之一：</p> <ul style="list-style-type: none"> • ALTER ANY TABLE 系统特权 • ALTER ANY OBJECT 系统特权

另请参见

- 限制（第 126 页）
- 范围分区（第 127 页）
- 散列分区（第 128 页）
- 散列范围分区（第 129 页）

CHECKPOINT 语句

对数据库执行检查点操作。

快速链接：

转至用法（第 383 页）

转至标准（第 383 页）

转至权限（第 383 页）

语法

CHECKPOINT

用法

(返回顶部)（第 383 页）

CHECKPOINT 强制数据库服务器执行检查点操作。数据库服务器也会根据内部算法自动执行检查点操作。通常，应用程序无需发出 **CHECKPOINT**。

SAP Sybase IQ 使用不同于 OLTP 数据库的检查点，如 SQL Anywhere。OLTP 数据库往往具有短事务，这些事务只会影响少数行。对这些事务而言，将整个页面写入到磁盘的成本非常昂贵。相反，OLTP 数据库通常会在检查点处向磁盘写入数据，并且只写入已更改的数据行。SAP Sybase IQ 是 OLAP 数据库。一个 OLAP 事务可以更改上千行甚至几百万行数据。因此，数据库服务器不会一直等到出现检查点再执行物理写入。在每个事务提交后，它都会将已更新的数据页写入磁盘中。对于 OLAP 数据库，将整页数据写入磁盘比在任意检查点处写入少量数据更加有效。

调整检查点时间或发出显式检查点可能不是必要操作。与在 OLTP 数据库产品中相比，在 SAP Sybase IQ 中，控制检查点并不那么重要，因为 SAP Sybase IQ 会在每个事务提交后写入实际数据页。

标准

(返回顶部)（第 383 页）

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持。

权限

(返回顶部)（第 383 页）

需要 CHECKPOINT 系统特权。

COMMIT 语句

使数据库更改永久有效，或终止用户定义的事务。

快速链接：

[转至示例](#)（第 384 页）

[转至用法](#)（第 384 页）

[转至标准](#)（第 385 页）

[转至权限](#)（第 385 页）

语法

语法 1 – 结束事务并使所有更改永久生效

```
COMMIT [ WORK ]
```

语法 2 – 构造嵌套事务

```
COMMIT TRAN[SACTION ] [ transaction-name ]
```

示例

[\(返回顶部\)](#)（第 384 页）

- **示例 1** – 提交当前事务：

```
COMMIT
```

- **示例 2** – 此 Transact-SQL 批处理报告 @@trancount 的连续值为：0、1、2、1、0：

```
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
go
```

用法

[\(返回顶部\)](#)（第 384 页）

- **语法 1** – 数据定义语句自动执行提交。有关信息，请参见每个 SQL 语句的副作用列表。

如果数据库服务器检测到任何无效的外键，**COMMIT** 将失败。这产生的结果是：不可能通过任何无效外键结束事务。通常，在每个数据处理操作中会检查外键的

完整性。但是，如果数据库选项 `WAIT_FOR_COMMIT` 设置为 `ON`，或者使用 **CHECK ON COMMIT** 子句定义了具体的外键，则数据库服务器会将完整性检查推迟到执行 **COMMIT** 语句之后。

- **语法 2** - 嵌套事务类似于保存点。当在一组嵌套事务的最外层执行时，此语句使数据库更改永久生效。在事务内部执行时，**COMMIT TRANSACTION** 会将事务的嵌套级别减少一级。当事务被嵌套时，只有最外层的 **COMMIT** 可使数据库更改永久生效。

可选参数 *transaction-name* 是分配给该事务的名称。它必须是有效的标识符。仅在最外面的一对嵌套的 **BEGIN/COMMIT** 或 **BEGIN/ROLLBACK** 语句中使用事务名称。

可以使用一组选项来控制 **COMMIT** 语句的行为细节。请参见 `COOPERATIVE_COMMIT_TIMEOUT` 选项、`COOPERATIVE_COMMITS` 选项、`DELAYED_COMMITS` 选项以及 `DELAYED_COMMIT_TIMEOUT` 选项。可以使用 **Commit** 连接属性返回当前连接的提交数。

副作用：

- 除了使用 **WITH HOLD** 打开的游标之外，关闭所有游标。
- 删除此连接上声明的临时表的所有行，除非这些表是用 **ON COMMIT PRESERVE ROWS** 声明的。

标准

(返回顶部) (第 384 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持。语法 2 是 ISO/ANSI SQL 语法的 Transact-SQL 扩展。

权限

(返回顶部) (第 384 页)

必须连接到数据库。

CREATE DATABASE 语句

创建由多个操作系统文件组成的数据库。

快速链接：

转至参数 (第 386 页)

转至示例 (第 391 页)

转至用法 (第 391 页)

转至标准 (第 394 页)

转至权限 (第 395 页)

语法

```

CREATE DATABASE db-name
... [ [ TRANSACTION ] { LOG ON [ log-file-name ]
    [ MIRROR mirror-file-name ] } ]
... [ CASE { RESPECT | IGNORE } ]
... [ PAGE SIZE catalog-page-size ]
... [ COLLATION collation-label [ ( collation-tailoring-string ) ] ]
... [ ENCRYPTED { algorithm-key-spec | OFF } ]
... [ BLANK PADDING ON ]
... [ JCONNECT { ON | OFF } ]
... [ IQ PATH iq-file-name ]
... [ IQ SIZE iq-file-size ]
... [ IQ PAGE SIZE iq-page-size ]
... [ BLOCK SIZE block-size ]
... [ IQ RESERVE sizeMB ]
... [ TEMPORARY RESERVE sizeMB ]
... [ MESSAGE PATH message-file-name ]
... [ TEMPORARY PATH temp-file-name ]
... [ TEMPORARY SIZE temp-db-size ]
... [ DBA USER userid ]
... [ DBA PASSWORD password ]
... [ SYSTEM PROCEDURE AS DEFINER { ON | OFF } ]

```

catalog-page-size (bytes) - (back to Syntax)
 { 4096 | 8192 | 16384 | 32768 }

collation-label - (back to Syntax)
string

collation-tailoring-string - (back to Syntax)
keyword=value

algorithm-key-spec - (back to Syntax)
ON
 | [**ON**] **KEY** *key* [**ALGORITHM** *AES-algorithm*]
 | [**ON**] **ALGORITHM** *AES-algorithm* **KEY** *key*
 | [**ON**] **ALGORITHM** 'SIMPLE'

AES-algorithm - (back to algorithm-key-spec)
 'AES' | 'AES256' | 'AES_FIPS' | 'AES256_FIPS'

key : - (back to algorithm-key-spec)
quoted string

iq-page-size (bytes) - (back to Syntax)
 { 65536 | 131072 | 262144 | 524288 }

block-size (bytes) - (back to Syntax)
 { 4096 | 8192 | 16384 | 32768 }

参数

(返回顶部) (第 385 页)

- **TRANSACTION LOG** – 数据库服务器用来记录所有数据库更改的文件。事务日志在系统恢复中起着重要的作用。如果不指定任何 **TRANSACTION LOG** 子句，或者如果您省略文件名的路径，则事务日志将位于 **.db** 文件所在的同一目录中。但是，应该将它放在不同于 **.db** 和 **.iq** 的物理设备上。无法在原始分区上创建该文件。
- **MIRROR** – 是与事务日志完全相同的一个副本，通常在单独的的设备上进行维护，以便更好地保护您的数据。缺省情况下，**SAP Sybase IQ** 不使用镜像事务日志。如果您不想使用事务日志镜像，则必须提供文件名。如果使用相对路径，则相对于目录存储库 (**db-name.db**) 所在的目录创建事务日志镜像。提示:始终创建事务日志的镜像副本。
- **CASE** – 对使用 **CASE RESPECT** 创建的数据库而言，在比较和字符串操作中，所有受影响的值都区分大小写。数据库对象名称（如列、过程或用户 ID）则不受影响。无论是否指定 **CASE**，**Dbospace** 名称始终不区分大小写。缺省值 (**RESPECT**) 为所有比较均区分大小写。**CASE RESPECT** 提供了比 **CASE IGNORE** 更高的性能。
- **PAGE SIZE** – 数据库（包含目录表）的 **SQL Anywhere** 段的页大小可以为 4096、8192、16384 或 32768 字节。通常，使用缺省值 4096 (4KB)。大型数据库可能需要比缺省大小更大的页大小以便提高性能。较小的值可能会限制数据库能够支持的列数。如果指定一个小于 4096 的页大小，则 **SAP Sybase IQ** 将使用 4096 的页大小。
- **COLLATION** – 用于在数据库中对字符数据类型进行排序和比较的归类序列。归类为所使用的编码（字符集）提供字符比较和排序信息。如果不指定 **COLLATION** 子句，**SAP Sybase IQ** 会根据操作系统语言和编码选择一种归类。对大多数操作系统来说，缺省归类序列为 **ISO_BINENG**，该序列可提供最佳性能。在 **ISO_BINENG** 中，归类顺序与 **ASCII** 字符集中的字符顺序相同。所有大写字母均位于所有小写字母之前（例如，'A' 和 'B' 均位于 'a' 的前面）。

可以从支持归类的列表中选择归类。对于在 **SAP Sybase IQ** 服务器上创建的 **SQL Anywhere** 数据库，归类也可以是 **Unicode** 归类算法 (**UCA**)。如果指定了 **UCA**，则也会指定 **ENCODING** 子句。**SAP Sybase IQ** 不支持 **SAP Sybase IQ** 数据库的任何基于 **UCA** 的归类。如果在数据库的 **CREATE DATABASE** 语句中指定基于 **UCA** 的归类，则服务器将返回错误不支持 **UCA** 归类且数据库创建将会失败。创建数据库后无法更改归类序列。

或者，您还可以指定归类定制选项 (*collation-tailoring-string*)，以便能够对字符的排序和比较进行更多的控制。这些选项采用“关键字=值”对的形式，包括在圆括号内，位于归类名后。

SAP Sybase IQ 的归类定制选项包含 **SAP Sybase IQ** 数据库支持的关键字、允许的替代形式以及允许的归类定制选项 (*collation-tailoring-string*) 值。

表 28. SAP Sybase IQ 的归类定制选项

关键字	归类	替代形式	允许值
CaseSensitivity	支持的全部归类	CaseSensitive、Case	<ul style="list-style-type: none">• respect – 区分字母间的大小写差异。对于 UCA 归类，这等效于 UpperFirst。对于其它归类，respect 的值取决于归类自身。• ignore – 忽略字母间的大小写差异。• UpperFirst – 始终将大写排在前面 (Aa)。• LowerFirst – 始终将小写排在前面 (aA)。

注意： 如果针对在 SAP Sybase IQ 服务器上创建的 SQL Anywhere 数据库指定了 UCA 归类，则若干归类定制选项将受到支持。对于所有其它归类 (SAP Sybase IQ)，仅支持区分大小写定制。另外，使用归类定制选项创建的数据库无法使用 15.0 之前版本的数据库服务器启动。

- **ENCRYPTED** – 使存储在物理数据库文件中的数据变为不可读。使用 **CREATE DATABASE ENCRYPTED** 关键字（不包含 **TABLE** 关键字）可加密整个数据库。使用 **ENCRYPTED TABLE** 子句仅对 SQL Anywhere 表启用表加密。SAP Sybase IQ 表不支持表级别加密。启用表加密将意味着，会使用您在创建数据库时指定的设置，对随后使用 **ENCRYPTED** 子句创建或变更的表进行加密。

有两个级别的数据库加密：简单加密和高度加密。

- 简单加密等效于模糊处理。虽然这些数据无法阅读，但了解加密技术的人可能会破解数据。对于简单加密，请指定 **CREATE DATABASE** 子句 **ENCRYPTED ON ALGORITHM 'SIMPLE'**、**ENCRYPTED ALGORITHM 'SIMPLE'**，或者指定 **ENCRYPTED ON** 子句而不指定算法或密钥。
- 高度加密可通过使用 128 位算法和安全密钥实现。如果没有密钥，数据将不可读取，而且基本上无法被破解。对于高度加密，请指定使用 128 位或 256 位 AES 算法的 **CREATE DATABASE** 子句 **ENCRYPTED ON ALGORITHM**，并使用 **KEY** 子句来指定加密密钥。应选择满足以下条件的密钥值：长度至少为 16 个字符，混合使用大小写并包含数字、字母和特殊字符。

每次启动数据库时，都需要此加密密钥。

只能在数据库创建期间指定加密。要对某个现有数据库引入加密，需要完全卸载、重新创建数据库，然后重新装载所有数据。如果使用 **ENCRYPTED** 子句但未指定算法，则缺省算法为 AES。缺省情况下，加密处于关闭状态。

警告！ 请保护好您的加密密钥！将密钥副本存储在一个安全位置。丢失了密钥将导致数据库完全无法访问，而无法访问的数据库是无法进行恢复的。

- **BLANK PADDING** – 将忽略用于比较用途的尾随空白 (**BLANK PADDING ON**)，并且嵌入式 SQL 程序将填充读入字符数组中的字符串。提供此选项是为了符合 ISO/ANSI SQL 标准。**CREATE DATABASE** 不再支持 **BLANK PADDING OFF**。

- **JCONNECT** – 要使用 SAP Sybase jConnect for JDBC 驱动程序来访问系统目录信息，请安装 jConnect 支持。将 JCONNECT 设置为 OFF 以排除 jConnect 系统对象（缺省设置为 ON）。只要不访问系统信息，您就仍然可以使用 JDBC。
- **IQ PATH** – 包含 SAP Sybase IQ 数据的主要段文件的路径名。可以指定操作系统文件或 I/O 设备的原始分区。（针对您的平台的《安装和配置指南》介绍了指定原始分区时所使用的格式。）SAP Sybase IQ 将根据您指定的路径名自动检测相应的类型。如果使用相对路径，则将相对于目录存储库（.db 文件）所在的目录创建文件。

如果省略 IQ PATH 子句，则指定以下任一选项将生成错误：IQ SIZE、IQ PAGE SIZE、BLOCK SIZE、MESSAGE PATH、TEMPORARY PATH 和 TEMPORARY SIZE。

- **IQ SIZE** – 通过 IQ PATH 子句指定的原始分区或操作系统文件的大小（以 MB 为单位）。对于原始分区，应不指定 IQ SIZE 而始终采用缺省值，使用缺省值可允许 SAP Sybase IQ 使用整个原始分区；如果为 IQ SIZE 指定一个值，则该值必须匹配 I/O 设备的大小，否则 SAP Sybase IQ 将返回一个错误。对于操作系统文件，您可以指定一个值，范围从下表中的最小值一直到最大值 4TB。

操作系统文件的缺省大小取决于 IQ PAGE SIZE：

表 29. IQ 存储文件和临时存储文件的缺省大小和最小大小

IQ PAGE SIZE	IQ SIZE 缺省值	TEMPORARY SIZE 缺省值	显式 IQ SIZE 最小值	显式 TEMPORARY SIZE 最小值
65536	4096000	2048000	4MB	2MB
131072	8192000	4096000	8MB	4MB
262144	16384000	8192000	16MB	8MB
524288	32768000	16384000	32MB	16MB

- **IQ PAGE SIZE** – 数据库的 SAP Sybase IQ 段（包含 IQ 表和索引）的页大小（以字节为单位）。该值必须为 2 的幂，从 65536 到 524288 字节。缺省值为 131072 (128KB)。页大小的其它值都将更改为下一个更大的大小。IQ 页大小决定着缺省的 I/O 传输块大小和数据库的最大数据压缩。

为获得最佳性能，请使用以下最小页大小：

- 64KB (IQ PAGE SIZE 65536)，适用于最大表最多包含 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)，适用于 64 位平台上最大表所含行数超过 10 亿但少于 40 亿的数据库，或总大小可能增至 8TB 或更大的数据库。128KB 为缺省 IQ 页大小。

- 256KB (IQ PAGE SIZE 262144)，适用于 64 位平台上最大表所含行数超过 40 亿的数据库，或总大小可能增至 8TB 或更大的数据库。
- **BLOCK SIZE** – 数据库的 SAP Sybase IQ 段的 I/O 传输块大小（以字节为单位）。该值必须小于 IQ PAGE SIZE，并且必须为介于 4096 到 32768 之间的 2 的幂。该大小的其它值将更改为下一个更大的大小。缺省值取决于 IQ PAGE SIZE 子句的值。对于大多数应用程序，此缺省值是最佳的。
- **IQ RESERVE** – 指定要为主 IQ 存储库 (IQ_SYSTEM_MAIN dbspace) 保留的空间大小（以 MB 为单位），以便将来可以增加该 dbfile 的大小。sizeMB 参数可以为大于 0 的任意数字。创建 dbspace 后，保留大小将无法更改。指定 IQ RESERVE 后，数据库将更多空间用于内部（空闲列表）结构。如果保留大小过大，则内部结构所需的空間可能会大于指定大小，这会导致错误。
- **TEMPORARY RESERVE** – 指定要为临时 IQ 存储库 (IQ_SYSTEM_TEMP dbspace) 保留的空间大小（以 MB 为单位），以便将来可以增加该 dbfile 的大小。sizeMB 参数可以为大于 0 的任意数字。创建 dbspace 后，保留大小将无法更改。指定 TEMPORARY RESERVE 后，数据库将更多空间用于内部（空闲列表）结构。如果保留大小过大，则内部结构所需的空間可能会大于指定大小，这会导致错误。

注意： 如果从备份恢复数据库，则临时 dbspace 的保留大小和模式将丢失。

- **MESSAGE PATH** – 包含 SAP Sybase IQ 消息跟踪文件的段的路径名。您必须指定操作系统文件；消息文件不能位于原始分区上。如果使用相对路径或省略该路径，则将相对于 .db 文件所在目录创建消息文件。
- **TEMPORARY SIZE** – 通过 TEMPORARY PATH 子句指定的原始分区或操作系统文件的大小（以 MB 为单位）。对于原始分区，应不指定 TEMPORARY SIZE 而始终采用缺省值，使用缺省值可允许 SAP Sybase IQ 使用整个原始分区。操作系统文件的缺省值始终为 IQ SIZE 值的二分之一。如果 IQ 存储库位于原始分区上且临时存储为操作系统文件，则缺省的 TEMPORARY SIZE 为 IQ 存储原始分区大小的二分之一。
- **DBA USER** – 已授予 SYS_AUTH_DBA_ROLE 系统角色的缺省用户帐户的用户名。如果您不指定此子句，SAP Sybase IQ 会创建缺省的 DBA 用户 ID。
- **DBA PASSWORD** – 已授予 SYS_AUTH_DBA_ROLE 系统角色的缺省用户帐户的密码。
- **SYSTEM PROCEDURE AS DEFINER** – 定义是以调用者（执行过程的人员）特权还是定义者（过程的所有者）特权来运行特许系统过程。OFF（缺省值）或未指定，表示将通过调用者特权执行所有特许系统过程。使用 sp_proc_priv() 标识运行系统过程所需的系统特权。

ON 表示 16.0 之前版本的特许系统过程以定义者特权来执行。16.0 或更高版本的特许系统过程以调用者特权来执行。

示例

(返回顶部) (第 385 页)

- **示例 1** – 此 Windows 示例将在 C:\s1\data 目录中创建一个名为 mydb 的 SAP Sybase IQ 数据库及其相应的 mydb.db、mydb.iq、mydb.iqtmp 和 mydb.iqmsg 文件：

```
CREATE DATABASE 'C:\\s1\\data\\mydb'
BLANK PADDING ON
IQ PATH 'C:\\s1\\data'
IQ SIZE 2000
IQ PAGE SIZE 131072
```

- **示例 2** – 此 UNIX 命令将在 IQ PATH 和 TEMPORARY PATH 的原始设备中创建 SAP Sybase IQ 数据库。将应用 128KB 的缺省 IQ 页大小。

```
CREATE DATABASE '/s1/data/bigdb'
IQ PATH '/dev/md/rdisk/bigdb'
MESSAGE PATH '/s1/data/bigdb.iqmsg'
TEMPORARY PATH '/dev/md/rdisk/bigtmp'
```

- **示例 3** – 此 Windows 命令将在 IQ PATH 的原始设备中创建 SAP Sybase IQ 数据库。请注意原始设备名中使用的双反斜线（Windows 要求）：

```
CREATE DATABASE 'company'
IQ PATH '\\\\.\\E:'
JCONNECT OFF
IQ SIZE 40
```

- **示例 4** – 以下 UNIX 示例将使用 AES 加密算法和密钥 "is!seCret" 创建一个高度加密的 SAP Sybase IQ 数据库。

```
CREATE DATABASE 'marvin.db'
BLANK PADDING ON
CASE RESPECT
COLLATION 'ISO_BINENG'
IQ PATH '/filesystem/marvin.main1'
IQ SIZE 6400
IQ PAGE SIZE 262144
TEMPORARY PATH '/filesystem/marvin.temp1'
TEMPORARY SIZE 3200
ENCRYPTED ON KEY 'is!seCret' ALGORITHM 'AES'
```

用法

(返回顶部) (第 385 页)

创建具有所提供名称和属性的数据库。要创建 SAP Sybase IQ 数据库，必须使用 IQ PATH 子句；否则将创建一个标准的 SQL Anywhere 数据库。

当 SAP Sybase IQ 创建数据库时，将自动生成四个数据库文件以存储构成数据库的不同类型的数据。每个文件都对应一个 dbspace，即 SAP Sybase IQ 用来标识数据库文件的逻辑名：

- *db-name.db* 是保存目录 *dbspace* 的文件，内部逻辑名称为 **SYSTEM**。其中包含用于描述数据库及您添加的所有标准 SQL Anywhere 数据库对象的系统表和存储过程。如果您没有包含 *.db* 扩展名，SAP Sybase IQ 将会添加。此初始 *dbspace* 包含目录存储库，以后您可以添加 *dbspace* 来增加其大小。无法在原始分区上创建该文件。
- *db-name.iq* 是用来保存主要数据 *dbspace* 的缺省文件名称，内部名称为 **IQ_SYSTEM_MAIN**，其中包含 **IQ** 表和索引。您可以使用 **IQ PATH** 子句指定不同的文件名。此初始 *dbspace* 包含 **IQ** 存储库。

警告！ **IQ_SYSTEM_MAIN** 是一个特殊的 *dbspace*，它包含打开数据库时所必需的所有结构：**IQ db_identity** 块、**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** 前进/回退位图、增量备份位图和空闲列表根页。打开数据库后，**IQ_SYSTEM_MAIN** 始终处于联机状态。

管理员可能允许在 **IQ_SYSTEM_MAIN** 中创建用户表，特别是当这些表比较小而又非常重要时。但是，更常见的情况是在创建数据库后，管理员立即创建另一个主 *dbspace*，撤消所有用户在 *dbspace IQ_SYSTEM_MAIN* 中的 **create** 特权，向选定用户授予对新的主 *dbspace* 的 **create** 特权，并将 **PUBLIC.default_dbspace** 设置为新的主 *dbspace*。

- *db-name.iqtmp* 是用来保存初始临时 *dbspace* 的缺省文件名称，内部名称为 **IQ_SYSTEM_TEMP**。其中包含特定查询生成的临时表。此文件所需的大小可能会根据查询类型和数据量的不同而发生改变。您可以使用 **TEMPORARY PATH** 子句指定不同的名称。此初始 *dbspace* 包含临时存储库。
- *db-name.iqmsg* 是包含消息跟踪 *dbspace* 的缺省文件名称，内部名称为 **IQ_SYSTEM_MSG**。可使用 **MESSAGE PATH** 子句指定不同的文件名称。

除这些文件之外，数据库还拥有事务日志文件 (*db-name.log*)，并且还可能拥有一个事务日志镜像文件。

文件名 (*db-name*、*log-file-name*、*mirror-file-name*、*iq-file-name*、*message-file-name*、*temp-file-name*) 是包含操作系统文件名的字符串。作为字符串，它们必须用单引号括起来。

在 Windows 中，如果指定路径，则任何后面跟有 **n** 或 **x** 的反斜杠字符 (\) 都必须使用两个。这样可防止系统按照 SQL 中的字符串规则将这些反斜杠解释为换行符 (\n) 或十六进制数 (\x)。始终使用两个反斜杠是较为安全的做法。例如：

```
CREATE DATABASE 'c:\\sybase\\mydb.db'  
LOG ON 'e:\\logdrive\\mydb.log'  
JCONNECT OFF  
IQ PATH 'c:\\sybase\\mydb'  
IQ SIZE 40
```

如果未指定路径或指定了相对路径：

- 相对于服务器的工作目录创建目录存储文件 (*db-name.db*)。

- 在目录存储库的同一目录或相对目录下创建 **IQ** 存储库、临时存储库和消息日志文件。

建议使用相对路径名。

警告！ 数据库文件、临时 **dbspace** 和事务日志文件必须位于数据库服务器所在的同一物理计算机上。不要将数据库文件和事务日志文件放在网络驱动器上。但事务日志应与其镜像分别位于不同的设备上。

在类似于 **UNIX** 的操作系统中，您可以创建符号链接，此类链接为间接指针，其中包含所指文件的路径名。可将符号链接用作相对路径名。为数据库文件名创建符号链接具有以下几个优点：

- 指向原始设备的符号链接可以具有有意义的名称，而实际设备名语法可能模糊不清。
- 使用符号名称可能会消除在恢复自备份后已移至新目录的数据库文件时存在的问题。

要创建符号链接，请使用 **ln -s** 命令。例如：

```
ln -s /disk1/company/iqdata/company.iq company_iq_store
```

创建该链接后，您可以在诸如 **CREATE DATABASE** 或 **RESTORE DATABASE** 之类的命令中指定符号链接而不是全路径名称。

创建数据库或 **dbspace** 时，每个 **dbspace** 文件的路径必须唯一。如果您的 **CREATE DATABASE** 命令为这两种存储库指定了相同的路径名和文件名，将收到一条错误。

可使用以下任一方式创建唯一路径：

- 为每个文件指定一个不同的扩展名（如 **mydb.iq** 和 **mydb.iqtmp**）
- 指定一个不同的文件名（如 **mydb.iq** 和 **mytmp.iq**）
- 指定一个不同的路径名（如 **/iqfiles/main/iq** 和 **/iqfiles/temp/iq**）或不同的原始分区
- 在创建数据库时省略 **TEMPORARY PATH**。此时，系统将在目录存储库所在的同一路径下创建临时存储库，缺省名和扩展名为 **dbname.iqtmp**，其中 **dbname** 为数据库名称。

警告！ 在类似于 **UNIX** 的操作系统上，为保持数据库的一致性，必须指定链接到不同文件的文件名。**SAP Sybase IQ** 无法检测到链接文件指向的目标。即使命令中的文件名不同，也应确保它们不会指向同一个操作系统文件。

插入到表中的字符串始终以它们输入时的大小写形式存储，无论数据库是否区分大小写。如果将字符串 **Value** 插入到字符数据类型列中，该字符串将始终以 **V** 大写、其余字母小写的形式存储在数据库中。**SELECT** 语句以 **Value** 形式返回字符串。但如果数据库不区分大小写，所有比较都会将 **Value** 与 **value**、**VALUE** 等不同大小写形式的字符串视为相同。**SAP Sybase IQ** 服务器能够以大小写字母的任意组合返回结果，因此，在不区分大小写的数据库中，得不到区分大小写的结果 (**CASE IGNORE**)。

例如，假定有以下表和数据：

```
CREATE TABLE tb (id int NOT NULL,  
                  string VARCHAR(30) NOT NULL);  
INSERT INTO tb VALUES (1, 'ONE');  
SELECT * FROM tb WHERE string = 'oNe';
```

SELECT 的结果可以为 "oNe"（如在 **WHERE** 子句中所指定的那样），而不必一定为 "ONE"（如在数据库中所存储的那样）。

类似地，

```
SELECT * FROM tb WHERE string = 'One';
```

的结果可以为 "One",

```
SELECT * FROM tb WHERE string = 'ONE';
```

的结果可以为 "ONE"。

在创建所有数据库时至少使用一个用户 ID：

DBA

以及口令：

sql

在新数据库中，所有口令都区分大小写，而与数据库是否区分大小写无关。用户 ID 不受 **CASE RESPECT** 设置的影响。

启动数据库时，其页大小不能大于当前服务器的页大小。服务器页大小可从启动的第一组数据库中获取，或者在服务器命令行上使用 **-gp** 命令行选项来设置。

任何语句的命令行长度均限定为目录页大小。在大多数情况下，4KB 缺省值已足够用；但在某些情况下，为了容纳非常长的命令（如引用多个 **dbspace** 的 **RESTORE DATABASE** 命令），需要更大的 **PAGE SIZE** 值。此外，执行涉及大量表或视图的查询时，可能需要较大的页大小。

由于缺省目录页大小为 4KB，因此仅当连接指向 **utility_db** 等数据库（页大小为 1024）时才会出现问题。此限制可能会导致引用大量 **dbspace** 的 **RESTORE DATABASE** 命令失败。为避免此问题，请确保 **SQL** 命令行长度小于目录页大小。

或者，使用 **-gp 32768** 启动引擎以增大目录页大小。

副作用：

- 自动提交

标准

(返回顶部) (第 385 页)

- **SQL - ISO/ANSI SQL** 语法的服务商扩充。
- SAP Sybase 数据库产品 - Adaptive Server 提供了一个 **CREATE DATABASE** 语句，但具有不同选项。

权限

(返回顶部) (第 385 页)

执行此语句所需的权限是通过 **-gu** 服务器命令行选项设置的，如下所示：

- **NONE** – 任何用户都不能发出此语句。
- **DBA** – 需要 **SERVER OPERATOR** 系统特权。
- **UTILITY_DB** – 只有能够连接到 `utility_db` 数据库的那些用户才能发出此语句。

运行服务器时所使用的帐户必须对创建文件的目录具有写权限。

CREATE DOMAIN 语句

在数据库中创建用户定义数据类型。

快速链接：

转至参数 (第 396 页)

转至示例 (第 396 页)

转至用法 (第 396 页)

转至标准 (第 397 页)

转至权限 (第 397 页)

语法

```
CREATE { DOMAIN | DATATYPE } domain-name data-type
... [ NOT ] NULL ]
... [ DEFAULT default-value ]
```

default-value – (back to Syntax)

special-value

```
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function( constant-expression )
| AUTOINCREMENT
| CURRENT DATABASE
| CURRENT REMOTE USER
| NULL
| TIMESTAMP
| LAST USER
```

special-value (back to default-value)

CURRENT

```
{ DATE
| TIME
| TIMESTAMP
```

```
| USER  
| PUBLISHER }  
| USER
```

参数

(返回顶部) (第 395 页)

- **data-type** – 具有精度和小数位数的内置数据类型

示例

(返回顶部) (第 395 页)

- **示例 1** – 创建名为 `address` 的数据类型，它包含 35 个字符长的字符串并且可以为 NULL:

```
CREATE DOMAIN address CHAR( 35 ) NULL
```

用法

(返回顶部) (第 395 页)

用户定义数据类型是内置数据类型的别名，在适用的时候还会包括精度值和小数位数。它们提高了方便性并促进了数据库的一致性。

注意： 使用 **CREATE DOMAIN** 而不是 **CREATE DATATYPE**，因为 **CREATE DOMAIN** 是 ANSI/ISO SQL3 术语。

创建数据类型的用户会自动成为此数据类型的所有者。在 **CREATE DATATYPE** 语句中不能指定所有者。用户定义数据类型名称必须是唯一的，所有用户都可以访问这个数据类型且不必将所有者作为前缀。

用户定义数据类型是数据库内的对象。其名称必须遵守标识符规则。与内置数据类型名称一样，用户定义数据类型名称始终不区分大小写。

缺省情况下，用户定义的数据类型允许 NULL，除非 **allow_nulls_by_default** 数据库选项设置为 **OFF**。这种情况下，新的用户定义数据类型的缺省设置为不允许 NULL。基于用户定义数据类型创建的列可否为空值取决于该用户定义数据类型定义的设置，而不取决于引用该列时 **allow_nulls_by_default** 选项的设置。任何在列定义中显式设置的 NULL 或 NOT NULL 都将覆盖用户定义数据类型设置。

使用 **CREATE DOMAIN** 语句可以对用户定义数据类型指定 **DEFAULT** 值。指定的 **DEFAULT** 值将由基于数据类型定义的任何列继承。在列上显式指定的任何 **DEFAULT** 值都将覆盖为该数据类型指定的值。

使用 **CREATE DOMAIN** 语句可以将称为 **CHECK** 条件的规则添加至用户定义数据类型的定义中。

对于基表、全局临时表、局部临时表和用户定义的数据类型，SAP Sybase IQ 将强制执行 **CHECK** 约束。

要从数据库中删除数据类型，请使用 **DROP** 语句。您必须是数据类型的所有者或者具有 **CREATE DATATYPE** 或 **CREATE ANY OBJECT** 系统特权才能删除用户定义的数据类型。

副作用：

- 自动提交

标准

(返回顶部) (第 395 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。Transact-SQL 使用 **sp_addtype** 系统过程以及 **CREATE DEFAULT** 和 **CREATE RULE** 语句提供类似功能。

权限

(返回顶部) (第 395 页)

需要以下特权之一：

- **CREATE DATATYPE** 系统特权。
- **CREATE ANY OBJECT** 系统特权。

CREATE INDEX 语句

创建指定表或表的索引。索引一旦创建就再也不能在 SQL 语句中再次引用，除非使用 **DROP INDEX** 语句将它删除。

快速链接：

转至参数 (第 398 页)

转至示例 (第 400 页)

转至用法 (第 401 页)

转至标准 (第 403 页)

转至权限 (第 404 页)

语法

```
CREATE [ UNIQUE ] [ index-type ] INDEX [ IF NOT EXISTS ] index-name
  ...ON [ owner. ] table-name
  ... ( column-name [ , column-name ] ... )
  ...[ { IN | ON } dbspace-name ]
  ...[ NOTIFY integer ]
  ...[ DELIMITED BY 'separators-string' ]
  ...[ LIMIT maxwordsize-integer ]
```

index-type{ **CMP** | **HG** | **HNG** | **LF** | **WD** | **DATE** | **TIME** | **DTTM** }**参数**

(返回顶部) (第 397 页)

- **index-type** – 对于 SAP Sybase IQ 表中的列，可指定 **index-type** 为 **HG** (**High_Group**)、**HNG** (**High_Non_Group**)、**LF** (**Low_Fast**)、**WD** (**Word**)、**DATE**、**TIME** 或 **DTTM** (**Datetime**)。如果未指定 **index-type**，则缺省创建 **HG** 索引。

要为 **IQ** 主存储表中两列之间的关系创建索引，可指定 **index-type** 为 **CMP** (比较)。两个列必须具有相同的数据类型、精度和小数位数。对于 **CHAR**、**VARCHAR**、**BINARY** 或 **VARBINARY** 列，精度表示两列具有相同的宽度。

要想获得最快的查询速度，列索引的正确类型取决于：

- 列中的唯一值数
- 查询中的列的使用方式
- 可用的磁盘空间量

您可以为 **IQ** 主存储表中的某个列指定多个索引，但这些索引必须具有不同的索引类型。**CREATE INDEX** 不允许添加重复的索引类型。**SAP Sybase IQ** 会选择适用于当前查询或查询一部分的最快索引。但每增加一种索引类型都可能会显著增加对该表的空间要求。

- **column-name** – 指定要建立索引的列的名称。列名称是前面带有可选相关名的标识符。(相关名通常是表名。有关相关名的详细信息，请参见 **FROM** 子句。)如果列名含有字母、数字和下划线以外的其它字符，则用引号 (") 将它引起来。

如果省略 **UNIQUE** 子句，只能指定 **HG** 索引。外键要求非唯一索引，组合外键要求非唯一的组合 **HG** 索引。唯一和非唯一的 **HG** 索引的多列组合键的最大宽度均为 5300 字节。如果 **CHAR** 或 **VARCHAR** 数据是组合键或单列 **HG**、**LF**、**HNG**、**DATE**、**TIME** 或 **DTTM** 索引的一部分，则这两种数据都不能超过 255 字节。

- **UNIQUE** – 确保表中不会有两行的值在索引的所有列中相同。每个索引键都必须是唯一的，或者至少在一列中包含 **NULL**。可以对多个列创建唯一的 **HG** 索引，但不能使用其它索引类型创建多列索引。您不能对 **CMP**、**HNG**、**WD**、**DATE**、**TIME** 或 **DTTM** 索引类型指定 **UNIQUE**。

如果列定义允许使用 **NULL** 值并且不实施约束 (主键或唯一)，则 **SAP Sybase IQ** 允许在用户创建的唯一多列 **HG** 索引的数据值中使用 **NULL** 值。有关详细信息，请参见注释中的“多列索引”。

- **IF NOT EXISTS** – 如果指定的对象已存在，则不进行任何更改，也不会返回错误。
- **IN** – 指定索引位置。如果省略 **IN** 子句，将在创建表的 **dbspace** 中创建索引。索引始终与其表放在同一类型的 **dbspace** (**IQ** 存储库或临时存储库) 中。装载索引时，数据将跨所有具备可用空间的该类型数据库文件分布。**SAP Sybase IQ** 确保您

指定的任何 *dbspace-name* 都适用于该索引。如果尝试为临时表的索引指定 `IQ_SYSTEM_MAIN` 或其它主 *dbspace*，则会收到一条错误，反之亦然。无论是指定 **CREATE DATABASE...CASE IGNORE** 还是指定 **CASE RESPECT**，*Dbpace* 名称始终不区分大小写。

- **DELIMITED BY** - 指定用于将列字符串解析为要存储到相应列 **WD** 索引中的单词的分隔符。如果忽略该子句或指定该值为空字符串，则 **SAP Sybase IQ** 使用缺省分隔符集。缺省分隔符集针对缺省归类顺序 (**ISO-BINENG**) 而设计。它包括非 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。缺省情况下，连字符和单引号是单词的一部分。缺省分隔符集中有 64 个分隔符。例如，如果列值为以下字符串：

```
The cat is on the mat
```

并且使用了使用缺省分隔符的 **CASE IGNORE** 设置来创建数据库，则此字符串中的下列单词会存储到 **WD** 索引中：

```
cat is mat on the
```

如果指定多个 **DELIMITED BY** 和 **LIMIT** 子句，则不会返回任何错误，但只使用每种类型的最后一个子句。

- **separators-string** - 必须是数据库创建时使用的归类顺序中的 0 个或多个字符的序列。分隔符字符串中的每个字符都被当作一个分隔符。如果分隔符字符串中没有字符，则会使用缺省分隔符集。（每个分隔符必须是正在使用的归类序列中的单个字符。）分隔符字符串中的字符（分隔符）不能多于 256 个。

要将制表符指定为分隔符，可以在分隔符字符串内键入 `<TAB>` 字符，或者使用制表符的十六进制 **ASCII** 代码 `\x09`。`"\t"` 指定两个分隔符，`\` 和字母 `t`。要将换行符指定为分隔符，可以键入 `<RETURN>` 字符或十六进制 **ASCII** 代码 `\x0a`。

例如，子句 **DELIMITED BY** `' ; ; . \ / t '` 指定以下 7 个分隔符：

```
space : ; . \ / t
```

表 30. 制表符和换行符作为分隔符

分隔符	DELIMITED BY 子句的分隔符字符串
制表符	<code>' '</code> （键入 <code><TAB></code> ）或 <code>'\x09'</code>
换行符	<code>' '</code> （键入 <code><RETURN></code> ）或 <code>'\x0a'</code>

- **LIMIT** - 仅可用于创建 **WD** 索引。指定 **WD** 索引中允许的最大单词长度。分析过程中单词长度超过该值会导致错误。缺省值为 255 个字节。允许的最小值为 1，允许的最大值为 255。如果 **CREATE INDEX** 语句中指定的最大单词长度或缺省确定的最大单词长度超过列宽，则所用的最大单词长度会自动缩短至列宽。使用的最大允许单词长度越小，允许插入、删除和更新使用的空间会更小、时间会更短。空词（两个相邻的分隔符）将被自动忽略。创建 **WD** 索引后，使用创建时确定的

分隔符和最大单词大小来解析在相应列中插入的所有内容。创建索引之后，将无法更改这些分隔符和最大单词大小。

- **NOTIFY** – 在为索引成功添加 **n** 个记录后，给出通知消息。这些消息会发送到标准输出设备。消息包含与内存使用情况、数据库空间以及当前使用的缓冲区数相关的信息。缺省值为 100,000 条记录。要关闭 **NOTIFY**，请将其设置为 0。

示例

(返回顶部) (第 397 页)

- **示例 1** – 创建 `projected_earnings` 和 `current_earnings` 列的比较索引。这两列是十进制列，具有相同的精度和小数位。

```
CREATE CMP INDEX proj_curr_cmp
ON sales_data
( projected_earnings, current_earnings )
```

- **示例 2** – 为 `SalesOrderItems` 表的 `ID` 列创建一个 **High_Group** 索引。该索引的数据页是从 `dbspace Dsp5` 中分配的。

```
CREATE HG INDEX id_hg
ON SalesOrderItems
( ID ) IN Dsp5
```

- **示例 3** – 为 `SalesOrderItems` 表的 `ProductID` 列创建一个 **High_Group** 索引：

```
CREATE HG INDEX item_prod_hg
ON Sales_OrderItems
( ProductID)
```

- **示例 4** – 为 `SalesOrderItems` 表的同一 `ProductID` 列创建一个 **Low_Fast** 索引，且不会出现任何通知消息：

```
CREATE LF INDEX item_prod
ON SalesOrderItems
( ProductID)
NOTIFY 0
```

- **示例 5** – 为 `earnings_report` 表创建一个 **WD** 索引。将字符串的分隔符指定为空格、冒号、分号和句点。将字符串的长度限定为 25。

```
CREATE WD INDEX earnings_wd
ON earnings_report_table(varchar)
DELIMITED BY ' ; , . '
LIMIT 25
```

- **示例 6** – 为 `SalesOrders` 表的 `OrderDate` 列创建一个 **DTTM** 索引：

```
CREATE DTTM INDEX order_dttm
ON SalesOrders
( OrderDate )
```


用法

(返回顶部) (第 397 页)

- 无法在 **CREATE INDEX** 语句中指定索引所有者。索引将自动由定义了这些索引的表的所有者拥有。对于每个所有者，索引名称必须唯一。
- 无法为视图创建索引。对于给定表，每个索引的名称必须唯一。
- 只要 **CREATE INDEX** 影响了当前正由其它连接修改的表，就会禁止该语句。但允许对正在同时添加索引的表进行查询。
- 创建 **WD** 索引后，使用分隔符来解析在列中插入的所有内容，而且，在创建该索引后，将无法更改最大单词大小。对于 **CHAR** 列，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。**SAP Sybase IQ** 自动将 **CHAR** 列填充至最大列宽。如果列包含空白和字符数据，**WD** 索引数据查询返回的结果可能会造成误解。例如，列 **CompanyName** 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 0 而不是 1。您可以用 **VARCHAR** 列代替 **CHAR** 列来避免此问题。

- 数据类型：
 - 无法使用 **CREATE INDEX** 为包含 **BIT** 数据的列创建索引。
 - 对超过 255 字节的 **CHAR** 和 **VARCHAR** 数据，只能创建缺省索引、**CMP** 索引或 **WD** 索引。
 - 对 **LONG VARCHAR** 数据，只能创建缺省索引类型和 **WD** 索引类型。
 - 对超过 255 字节的 **BINARY** 和 **VARBINARY** 数据，只能创建缺省索引类型、**CMP** 索引类型或 **TEXT** 索引类型。
 - 无法对具有 **FLOAT**、**REAL** 或 **DOUBLE** 数据的列创建 **HNG** 索引或 **CMP** 索引。
 - 只能对具有 **TIME** 数据类型的列创建 **TIME** 索引。
 - 只能对具有 **DATE** 数据类型的列创建 **DATE** 索引。
 - 只能对具有 **DATETIME** 或 **TIMESTAMP** 数据类型的列创建 **DTTM** 索引。
- 您可对多个列创建一个唯一或非唯一 **HG** 索引。**SAP Sybase IQ** 可对组成外键的一组列隐式创建一个非唯一的 **HG** 索引。

仅 **HG** 和 **CMP** 是可以包含多列的索引类型。您不能对多个列创建唯一的 **HNG** 或 **LF** 索引，也不能对多个列创建 **DATE**、**TIME** 或 **DTTM** 索引。

多列连接键的最大宽度为 **5KB**（5300 字节）。允许的列数取决于多少个列的大小能凑成 **5KB**。超过 255 字节的 **CHAR** 或 **VARCHAR** 数据不允许作为单列 **HG**、**LF**、**HNG**、**DATE**、**TIME** 或 **DTTM** 索引中组合键的一部分。

多列索引上的 **INSERT** 必须包含索引的所有列。

使用多列 **HG** 索引可以提高在 **ORDER BY** 子句中对单列进行查询的运行速度。例如：

```
SELECT abs (x) from t1
ORDER BY x
```

在以上示例中，**HG** 索引按排序顺序垂直投影 **x**。

为增强查询性能，请在满足以下条件，在 **SELECT** 或 **ORDER BY** 子句中，使用多列 **HG** 索引对多个列（也可以包括 **ROWID**）执行 **ORDER BY** 操作：

- 索引中存在所有投影列以及所有排序列（**ROWID** 除外）
- 排序键按顺序与前导 **HG** 列匹配

如果有多个多列 **HG** 索引满足这些条件，则使用具有最低非重复计数的索引。

如果查询具有一个 **ORDER BY** 子句，且 **ORDER BY** 列列表为某个多列索引的前缀（其中，**SELECT** 列表中引用的所有列均存在于一个多列索引中），则该多列索引执行垂直投影；例如：

```
SELECT x,z,y FROM T
ORDER BY x,y
```

如果 **SELECT** 列表中的基列存在表达式，并且所有表达式中引用的所有列都存在于该多列索引中，则查询将使用多列索引；例如：

```
SELECT power(x,2), x+y, sin(z) FROM T
ORDER BY x,y
```

除以上两个示例之外，如果 **ROWID()** 函数位于 **SELECT** 列表表达式中，则将使用多列索引。例如：

```
SELECT rowid()+x, z FROM T
ORDER BY x,y,z
```

除以上三个示例之外，如果 **ROWID()** 存在于 **ORDER BY** 列表结尾处，并且该列表的列（不包括 **ROWID()**）以准确顺序使用多列索引，则多列索引将用于该查询。例如：

```
SELECT z,y FROM T
ORDER BY x,y,z,ROWID()
```

如果列定义允许使用 **NULL** 值并且不实施约束（主键或唯一），则 **SAP Sybase IQ** 允许在用户创建的唯一多列 **HG** 索引的数据值中使用 **NULL** 值。此功能的规则如下：

- 将 **NULL** 值视为未定义的值。
- 唯一索引列中允许多个值为 **NULL** 的行。
 1. 在单列索引中，一个索引列中允许多个值为 **NULL** 的行。
 2. 在多列索引中，只要其余列中的非 **NULL** 值可保证该索引中的唯一性，索引列允许多个值为 **NULL** 的行。

3. 在多列索引中，索引涉及的所有列中允许多个值为 **NULL** 的行。

以下示例阐释了这些规则。假定表 `table1`：

```
CREATE TABLE table1
(c1 INT NULL, c2 INT NULL, c3 INT NOT NULL);
```

对允许 **NULL** 值的列创建唯一的单列 **HG** 索引：

```
CREATE UNIQUE HG INDEX c1_hg1 ON table1 (c1);
```

根据上面的规则 1，可将 **NULL** 值插入到索引列的多个行中：

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,1,1);
INSERT INTO table1(c1,c2,c3) VALUES (NULL,2,2);
```

对允许 **NULL** 值的列创建唯一多列 **HG** 索引：

```
CREATE UNIQUE HG INDEX c1c2_hg2 ON table1(c1,c2);
```

根据上述规则 2，必须确保索引的唯一性。下面的 **INSERT** 不成功，因为行 1 和行 3 的多列索引 `c1c2_hg2` 具有相同的值：

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,1,3);
```

但根据规则 1 和 3，以下 **INSERT** 操作会成功：

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,NULL,3);
INSERT INTO table1(c1,c2,c3) VALUES (NULL,NULL,4);
```

唯一性会在多列索引中得到保留。

此 **UPDATE** 操作成功，因为规则 3 允许在多列索引的所有列中包含多个值为 **NULL** 的行：

```
UPDATE table1 SET c2=NULL WHERE c3=1
```

如果多列 **HG** 索引受唯一约束制约，则索引中涉及的任何列都不允许 **NULL** 值。

- 可以使用 **BEGIN PARALLEL IQ ... END PARALLEL IQ** 语句将要对多个 IQ 主存储表执行的 **CREATE INDEX** 语句分组在一起，这样便可像一条 DDL 语句那样来执行这些语句。有关详细信息，请参见 **BEGIN PARALLEL IQ ... END PARALLEL IQ** 语句。

警告！ 对包含未提交数据的局部临时表使用 **CREATE INDEX** 命令会失败并生成错误消息必须提交局部临时表 `<tablename>` 才能创建索引。请先提交局部临时表中的数据，然后再创建索引。

副作用

- 自动提交

标准

(返回顶部) (第 397 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 与 SAP Sybase IQ 相比，Adaptive Server 的 **CREATE INDEX** 语句更为复杂。虽然在 SAP Sybase IQ 中允许使用 Adaptive Server 语法，但一些

子句和关键字会被忽略。有关 Adaptive Server **CREATE INDEX** 语句的完整语法，请参见《Adaptive Server 参考手册》第二卷：命令。

Adaptive Server 索引可以是聚簇索引或非聚簇索引。聚簇索引检索数据几乎总是比非聚簇索引快。每个表只允许有一个聚簇索引。

SAP Sybase IQ 不支持聚簇索引。SQL Anywhere 允许使用 **CLUSTERED** 和 **NONCLUSTERED** 关键字，但 SAP Sybase IQ 将忽略它们。如果未指定 *index-type*，则 SAP Sybase IQ 会对指定的列创建 HG 索引。

SAP Sybase IQ 不允许使用 **DESC** 关键字。

对于 SAP Sybase IQ 和 Adaptive Server，给定表的索引名称必须唯一。

权限

(返回顶部) (第 397 页)

需要对要创建索引的 dbspace 具有 **CREATE** 特权。还需要具备以下特权之一：

- **CREATE ANY INDEX** 系统特权。
- **CREATE ANY OBJECT** 系统特权。
- 索引基础表的 **REFERENCES** 特权。
- 您拥有索引的基础表。

CREATE LOGICAL SERVER 语句

创建用户定义的逻辑服务器。此语句强制共享系统临时存储设置在逻辑服务器共享的物理节点中保持一致。

快速链接：

转至参数 (第 404 页)

转至示例 (第 405 页)

转至权限 (第 406 页)

语法

```
CREATE LOGICAL SERVER logical-server-name [  
    { ls-create-clause, ... } ] [ WITH STOP SERVER ]  
  
ls-create-clause - (back to Syntax)  
    { MEMBERSHIP ( { ls-member, ... } ) | POLICY ls-policy-name }  
  
ls-member - (back to ls-create-clause)  
    FOR LOGICAL COORDINATOR | mpx-server-name
```

参数

(返回顶部) (第 404 页)

- **logical-server-name** – 用户指定的任意标识符，但以下项目除外：
 - ALL
 - AUTO
 - COORDINATOR
 - DEFAULT
 - NONE
 - OPEN
 - SERVER
- **MEMBERSHIP** – 要定义协调器逻辑成员资格，在 **MEMBERSHIP** 子句中使用 **FOR LOGICAL COORDINATOR**。

如果在逻辑服务器创建过程中未指定成员，则创建的逻辑服务器为空。

注意： 系统根本不会存储隐式逻辑服务器成员资格定义（如用于 **OPEN** 和 **SERVER** 逻辑服务器的定义）。

SYS.ISSYSLOGICALMEMBER 系统表存储逻辑服务器成员资格的定义。

将根逻辑服务器策略的 **ALLOW_COORDINATOR_AS_MEMBER** 选项从 **ON** 改为 **OFF** 并不会影响目录中存储的成员资格信息，而只会影响逻辑服务器的有效配置。

可通过指定 **Multiplex** 服务器名或使用 **FOR LOGICAL COORDINATOR** 子句来定义当前协调器的逻辑服务器成员资格，即使在 **ALLOW_COORDINATOR_AS_MEMBER** 选项设置为 **OFF** 时也是如此。成员资格定义存储在目录中，但在该 **Multiplex** 服务器充当协调器时处于非活动状态。

目录中存储着逻辑服务器及其成员资格定义。

- **POLICY** – 将逻辑服务器与用户定义的逻辑服务器策略相关联。如果没有指定 **POLICY** 子句，则逻辑服务器将与根策略相关联。
- SYS.ISSYSIQLOGICALSERVER** 系统表存储相应逻辑服务器中逻辑服务器策略的相关信息。
- **ls-policy-name** – 用户指定的任意标识符，但 **ROOT** 除外。
 - **WITH STOP SERVER** – 在 **TEMP_DATA_IN_SHARED_TEMP** 选项发生直接或间接更改时，会自动关闭逻辑服务器中的所有服务器。

应用于
仅限 **Multiplex**。

示例

(返回顶部) (第 404 页)

- **示例 1** – 创建用户定义的逻辑服务器 *Is1*，其中包含三个充当其成员的 **Multiplex** 节点：

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( n1, n2, n3 )
```

- **示例 2** – 创建含三个成员节点的用户定义逻辑服务器 *ls1*，并定义逻辑服务器策略名 *lsp1*:

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( w1_svr, w2_svr, r2_svr )  
POLICY lsp1
```

- **示例 3** – 像示例 2 那样创建服务器，只不过 **WITH STOP SERVER** 会在 **TEMP_DATA_IN_SHARED_TEMP** 选项发生直接或间接更改时自动关闭逻辑服务器中的所有服务器:

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( w1_svr, w2_svr, r2_svr )  
POLICY lsp1 WITH STOP SERVER
```

- **示例 4** – 创建用户定义的逻辑服务器 *ls1*，其中含有逻辑服务器策略 *lspolicy1* 但不包含成员节点:

```
CREATE LOGICAL SERVER ls1 POLICY lspolicy1
```

- **示例 5** – 其中 *n1* 为当前协调器，创建逻辑服务器 *ls2*，该服务器含有 **Multiplex** 节点 *n1* 和 *n3* 的指定成员资格以及协调器的逻辑成员资格。还要将 *ls2* 的逻辑服务器策略设置为 *lspolicy2*。

```
CREATE LOGICAL SERVER ls2 POLICY  
MEMBERSHIP FOR LOGICAL COORDINATOR  
lspolicy1, n1, n2, n3 POLICY lspolicy2
```

权限

(返回顶部) (第 404 页)

需要 **MANAGE MULTIPLEX** 系统特权。

CREATE LS POLICY 语句

创建用户定义的逻辑服务器策略。此语句强制共享系统临时存储设置在逻辑服务器共享的物理节点中保持一致。

快速链接:

[转至参数](#) (第 407 页)

[转至示例](#) (第 407 页)

[转至用法](#) (第 407 页)

[转至标准](#) (第 409 页)

[转至权限](#) (第 409 页)

语法

```
CREATE LS POLICY ls-policy-name ls-option-value-list [ WITH STOP SERVER ]
```

ls-option-value-list – (back to Syntax)

```
{ ls-option-name = ls-policy-option-value } ...

ls-option-name - (back to ls-option-value-list)
  ALLOW_COORDINATOR_AS_MEMBER
  | DQP_ENABLED
  | ENABLE_AUTOMATIC_FAILOVER
  | LOGIN_REDIRECTION
  | REDIRECTION_WAITERS_THRESHOLD
  | TEMP_DATA_IN_SHARED_TEMP
```

参数

(返回顶部) (第 406 页)

- **ls-policy-name** - 逻辑服务器策略的名称。可以为策略名指定除 **root** 之外的任何标识符。
- **ls-option-value-list** - 逻辑服务器策略选项的名称。请参见用法 (第 407 页)。
- **ls-policy-option-value** - 任何未指定的选项将从根逻辑服务器策略中继承值。请参见用法 (第 407 页)。
- **WITH STOP SERVER** - 当直接或间接更改 **TEMP_DATA_IN_SHARED_TEMP** 选项时，将自动关闭逻辑服务器中的所有服务器。

应用于

仅限 Multiplex。

示例

(返回顶部) (第 406 页)

- **示例 1** - 创建名称为 *lspolicy1* 的用户定义逻辑服务器策略：

```
CREATE LS POLICY lspolicy1
ALLOW_COORDINATOR_AS_MEMBER=ON;
```

用法

(返回顶部) (第 406 页)

如果您需要一个较小的 **IQ_SYSTEM_TEMP** dbspace，请将 **TEMP_DATA_IN_SHARED_TEMP** 设置为 **ON**，这会将临时数据写入 **IQ_SHARED_TEMP** 而非 **IQ_SYSTEM_TEMP**。但在分布式查询处理环境中，将 **DQP_ENABLED** 和 **TEMP_DATA_IN_SHARED_TEMP** 同时设置为 **ON** 可能由于 **IQ_SHARED_TEMP** 中有更多数据而造成 **SAN** 饱和，而针对 **IQ_SHARED_TEMP** 执行的额外 I/O 操作可能会对 **DQP** 性能造成负面影响。

选项	描述
ALLOW_COORDINATOR_AS_MEMBER	<p>只能为 ROOT 逻辑服务器策略设置。设置为 ON (缺省值) 时, 协调器可以是任何用户定义逻辑服务器的成员。OFF 表示阻止协调器用作任何用户定义逻辑服务器的成员。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - ON
DQP_ENABLED	<p>如果设置为 0, 不分布查询处理。如果设置为 1 (缺省值), 只要可写共享临时文件存在, 便分布查询处理。如果设置为 2, 在整个网络上分布查询处理, 不使用共享的临时存储库。</p> <ul style="list-style-type: none"> 值 - 0, 1, 2 缺省值 - 1
ENABLE_AUTOMATIC_FAILOVER	<p>只能为 ROOT 逻辑服务器策略设置。当设置为 ON 时, 将启用由指定登录策略控制的逻辑服务器的自动故障转移。当设置为 OFF (缺省值) 时, 禁用逻辑服务器级别的自动故障转移, 允许手动故障转移。指定 DEFAULT 以设置回缺省值。</p> <ul style="list-style-type: none"> 值 - ON、OFF、DEFAULT 缺省值 - OFF
LOGIN_REDIRECTION	<p>当设置为 ON 时, 将启用由指定登录策略控制的逻辑服务器的登录重定向。当设置为 OFF (缺省值) 时, 禁用逻辑服务器级别的登录重定向, 允许外部连接管理。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF
REDIRECTION_WAITERS_THRESHOLD	<p>指定在 SAP Sybase IQ 将与此逻辑服务器的连接重定向到其它服务器之前可排队的连接数。可以是任意整数; 缺省值为 5。</p> <ul style="list-style-type: none"> 值 - 整数 缺省值 - 5
TEMP_DATA_IN_SHARED_TEMP	<p>设置为 ON 时, 所有临时表数据和符合条件的临时数据都将写入共享临时存储库中, 前提是共享临时存储库中至少添加了一个读写文件。必须在设置此选项之后或在向共享临时存储库添加读写文件之后重新启动所有 Multiplex 节点。(如果共享临时存储库不包含任何读写文件, 或者如果您不重新启动节点, 数据会改为写入 IQ_SYSTEM_TEMP 中。) 当设置为 OFF (缺省值) 时, 所有临时表数据和临时数据都会写入到局部临时存储库中。</p> <ul style="list-style-type: none"> 值 - ON、OFF 缺省值 - OFF

标准

(返回顶部) (第 406 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 的支持。

权限

(返回顶部) (第 406 页)

需要 MANAGE MULTIPLEX 系统特权。

LS 策略选项

可用于根 LS 策略和用户定义 LS 策略的选项。

选项	描述
ALLOW_COORDINATOR_AS_MEMBER	<p>只能为 ROOT 逻辑服务器策略设置。设置为 ON (缺省值) 时, 协调器可以是任何用户定义逻辑服务器的成员。OFF 表示阻止协调器用作任何用户定义逻辑服务器的成员。</p> <ul style="list-style-type: none"> • 值 - ON、OFF • 缺省值 - ON
DQP_ENABLED	<p>如果设置为 0, 不分布查询处理。如果设置为 1 (缺省值), 只要可写共享临时文件存在, 便分布查询处理。如果设置为 2, 在整个网络上分布查询处理, 不使用共享的临时存储库。</p> <ul style="list-style-type: none"> • 值 - 0、1、2 • 缺省值 - 1
LOGIN_REDIRECTION	<p>当设置为 ON 时, 将启用由指定登录策略控制的逻辑服务器的登录重定向。当设置为 OFF (缺省值) 时, 禁用逻辑服务器级别的登录重定向, 允许外部连接管理。</p> <ul style="list-style-type: none"> • 值 - ON、OFF • 缺省值 - OFF
REDIRECTION_WAITERS_THRESHOLD	<p>指定在 SAP Sybase IQ 将与此逻辑服务器的连接重定向到其它服务器之前可排队的连接数。可以是任意整数; 缺省值为 5。</p> <ul style="list-style-type: none"> • 值 - 整数 • 缺省值 - 5

选项	描述
TEMP_DATA_IN_SHARED_TEMP	<p>设置为 ON 时，所有临时表数据和符合条件的临时数据都将写入共享临时存储库中，前提是共享临时存储库中至少添加了一个读写文件。必须在设置此选项之后或在向共享临时存储库添加读写文件之后重新启动所有 Multiplex 节点。（如果共享临时存储库不包含任何读写文件，或者如果您不重新启动节点，数据会改为写入 IQ_SYSTEM_TEMP 中。）当设置为 OFF（缺省值）时，所有临时表数据和临时数据都会写入到局部临时存储库中。</p> <ul style="list-style-type: none">• 值 - ON、OFF• 缺省值 - OFF

CREATE TABLE 语句

在数据库或远程服务器上创建一个新表。

快速链接：

转至参数（第 412 页）

转至示例（第 419 页）

转至用法（第 422 页）

转至标准（第 423 页）

转至权限（第 424 页）

语法

```
CREATE [ { GLOBAL | LOCAL } TEMPORARY ] TABLE
[ IF NOT EXISTS ] [ owner. ] table-name
... ( column-definition [ column-constraint ] ...
[ , column-definition [ column-constraint ] ... ]
[ , table-constraint ] ... )
| { ENABLE | DISABLE } RLV STORE

...[ IN dbspace-name ]
...[ ON COMMIT { DELETE | PRESERVE } ROWS ]
[ AT location-string ]
[ PARTITION BY
  range-partitioning-scheme
  | hash-partitioning-scheme
  | composite-partitioning-scheme ]

column-definition - (back to Syntax)
  column-name data-type
  [ [ NOT ] NULL ]
  [ DEFAULT default-value | IDENTITY ]
  [ PARTITION | SUBPARTITION ( partition-name IN dbspace-name
  [ , ... ] ) ]

default-value - (back to column-definition)
```

```

special-value
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function( constant-expression )
| AUTOINCREMENT
| CURRENT DATABASE
| CURRENT REMOTE USER
| NULL
| TIMESTAMP
| LAST USER

special-value - (back to default value)
CURRENT
{ DATE
| TIME
| TIMESTAMP
| USER
| PUBLISHER }
| USER

column-constraint - (back to Syntax)
[ CONSTRAINT constraint-name ] {
  { UNIQUE
    | PRIMARY KEY
    | REFERENCES table-name [ ( column-name ) ] [ action ]
  }
  [ IN dbspace-name ]
  | CHECK ( condition )
  | IQ UNIQUE ( integer )
}

table-constraint - (back to Syntax)
[ CONSTRAINT constraint-name ]
{ { UNIQUE ( column-name [ , column-name ] ... )
  | PRIMARY KEY ( column-name [ , column-name ] ... )
  }
  [ IN dbspace-name ]
  | foreign-key-constraint
  | CHECK ( condition )
  | IQ UNIQUE ( integer )
}

foreign-key-constraint - (back to table-constraint)
FOREIGN KEY [ role-name ] [ ( column-name [ , column-name ] ... ) ]
...REFERENCES table-name [ ( column-name [ , column-name ] ... ) ]
...[ actions ] [ IN dbspace-name ]

actions - (back to foreign-key-constraint)
[ ON { UPDATE | DELETE } RESTRICT ]

location-string - (back to Syntax) or (back to composite-partitioning-
scheme)
{ remote-server-name. [ db-name ].[ owner ].object-name

```

```

| remote-server-name; [ db-name ]; [ owner ];object-name }

range-partitioning-scheme - (back to Syntax)
    RANGE ( partition-key ) ( range-partition-decl [,range-partition-decl ... ] )

partition-key - (back to range-partitioning-scheme) or (back to hash-
partitioning-scheme)
    column-name

range-partition-decl - (back to range-partitioning-scheme)
    VALUES <= ( {constant-expr
| MAX } [ , { constant-expr
| MAX } ]... )
    [ IN dbspace-name ]

hash-partitioning-scheme - (back to Syntax) or (back to composite-
partitioning-scheme)
    HASH ( partition-key [ , partition-key, ... ] )

composite-partitioning-scheme - (back to Syntax)
    hash-partitioning-scheme SUBPARTITION range-partitioning-scheme

```

参数

(返回顶部) (第 410 页)

- **IN** - 在 column-definition、column-constraint、table-constraint、foreign-key 以及 partition-decl 子句中用于指定将创建对象的 dbspace。如果省略 IN 子句，SAP Sybase IQ 将在已分配表的 dbspace 中创建对象。

指定 SYSTEM 和该子句，以将永久表或临时表放在目录存储库中。指定 IQ_SYSTEM_TEMP 以将临时用户对象（表、分区或表索引）存储在 IQ_SYSTEM_TEMP 中，或者在 TEMP_DATA_IN_SHARED_TEMP 选项设置为 'ON' 且 IQ_SHARED_TEMP dbspace 包含 RW 文件时，存储在 IQ_SHARED_TEMP 中。（无法同时指定 IN 子句和 IQ_SHARED_TEMP。）IN 子句的所有其它用法均被忽略。缺省情况下，所有永久表均放置在主 IQ 存储库中，所有临时表放置在临时 IQ 存储库中。全局临时表和局部临时表决不能出现在 IQ 存储库中。

不支持下列语法：

```
CREATE LOCAL TEMPORARY TABLE tab1(c1 int) IN IQ_SHARED_TEMP
```

无法将 BIT 数据类型列式置于 dbspace 中。BIT 数据类型不支持以下内容：

```
CREATE TABLE t1(c1_bit bit IN iq_main);
```

- **ON COMMIT** - 仅允许用于临时表。缺省情况下，临时表的行将在 COMMIT（提交）时被删除。
- **AT** - 创建映射到 location-string 子句指定的远程位置的代理表。代理表名不能超过 30 个字符。AT 子句支持分号 (;) 作为分隔符。如果分号出现在 location-string 子句中的任何位置，则分号将用作字段分隔符。如果没有分号，则使用句号作为字段分隔符。这样一来，便可在数据库和所有者字段中使用文件名和扩展名。

分号字段分隔符主要用于当前不支持的服务器类；但是，您也可以在句点用作字段分隔符的情况下使用它们。例如，此语句将表 proxy_a 映射到远程服务器 myasa 上的 SQL Anywhere 数据库 mydb：

```
CREATE TABLE proxy_al
AT 'myasa;mydb;;al'
```

忽略远程表的外键定义。引用远程表的局部表的外键定义也会被忽略。如果远程服务器支持主键，则主键定义会发送到此服务器。

在 Simplex 环境中，不能在同一节点上创建引用远程表的代理表。在 Multiplex 环境中，不能创建引用在 Multiplex 中定义的远程表的代理表。

- **IF NOT EXISTS** – 如果指定的对象已存在，则不进行任何更改，也不会返回错误。
- **{ ENABLE | DISABLE } RLV STORE** – 将该表注册到 RLV 存储库以进行实时内存更新。IQ 临时表对此不支持。此值替代数据库选项 **BASE_TABLES_IN_RLV** 的值。将此值设置为 ENABLE 需要 CREATE TABLE 系统特权以及 RLV 存储 dbspace 的 CREATE 权限。
- **column-definition** – 定义表列。所允许的数据类型在《参考：构件块、表和过程》> “SQL 数据类型” 中进行了介绍。同一表中的两列不能同名。最多可以创建 45,000 列；但如果一个表中的列多于 10,000，则可能会导致性能下降。
 - **[NOT] NULL** – 包括或排除 NULL 值。如果指定 NOT NULL，或者列具有 UNIQUE 或 PRIMARY KEY 约束，则该列不能含有任何 NULL 值。每个表中允许 NULL 的列数存在限制，最多为大约 $8 * (\text{database-page-size} - 30)$ 。
 - **DEFAULT default-value** – 通过 CREATE TABLE（和 ALTER TABLE）语句中的 DEFAULT 关键字指定缺省列值。DEFAULT 值用作未指定列值的任何 INSERT（或 LOAD）语句中列的值。
 - **DEFAULT AUTOINCREMENT** – DEFAULT AUTOINCREMENT 列的值唯一地标识表中的每一行。此类型的列也称为 IDENTITY 列，可与 Adaptive Server 兼容。IDENTITY/DEFAULT AUTOINCREMENT 列存储在插入和更新期间自动生成的顺序编号。使用 IDENTITY 或 DEFAULT AUTOINCREMENT 时，列必须是小数位数为 0 的整数数据类型之一或精确的数字类型。列值也可以是 NULL。必须使用所有者名称来限定指定的表名。

ON 可向表中插入数据。如果没有指定 IDENTITY/DEFAULT AUTOINCREMENT 列的值，则生成一个比列中的任何其它值都大的唯一值。如果 INSERT 指定了列值，则使用该值；如果指定的值不大于列的当前最大值，该值将用作后续插入的起点。

删除行不会递减 IDENTITY/AUTOINCREMENT 计数器的值。由于删除行而产生的间隙只能由使用插入时的显式赋值填充。数据库选项 IDENTITY_INSERT 必须设置为表名，才能向 IDENTITY/AUTOINCREMENT 列执行插入操作。

例如，以下示例创建一个带有 IDENTITY 列的表，并向其显式添加一些数据：

```
CREATE TABLE mytable(c1 INT IDENTITY);
SET TEMPORARY OPTION IDENTITY_INSERT = "DBA".mytable;
INSERT INTO mytable VALUES(5);
```

显式插入小于该列最大值的行号后，后面没有显式赋值的行仍自动递增为比上一个最大值大 1 的值。

通过检查 @@identity 全局变量，可以找到最近一次插入的列值。

- **IDENTITY** - 使用 AUTOINCREMENT 缺省值的替代方法，与 Transact-SQL® 兼容。在 SAP Sybase IQ 中，可使用 IDENTITY 或 DEFAULT AUTOINCREMENT 子句来创建标识列。
- **table-constraint** - 帮助确保数据库中数据的完整性。共有四种类型的完整性约束：
 - **UNIQUE** - 标识唯一识别表中各行的一列或多列。表中任何两行的值在所有指定的列中不能相同。表可以有多个唯一约束。
 - **PRIMARY KEY** - 与 UNIQUE 约束相同，但表只能有一个主键约束。不能为同一列同时指定 PRIMARY KEY 约束和 UNIQUE 约束。主键通常标识行的最佳标识符。例如，客户号可能是 customer 表的主键。
 - **FOREIGN KEY** - 将某组列的值局限于与另一个表的主键或唯一约束中的值匹配。例如，外键约束可用于确保 invoice 表中的客户号与 customer 表中的客户号相对应。

不能在局部临时表上创建外键约束。全局临时表必须用 ON COMMIT PRESERVE ROWS 创建。

- **CHECK** - 允许对任意条件进行校验。例如，检查约束可用于确保名为 Gender 的列只包含值 male 或 female。表中的任何行都不能违反约束。如果 INSERT 或 UPDATE 语句会导致行违反约束，则不允许执行相应操作并且撤消语句的作用。

列检查约束中以符号 '@' 开头的列标识符是实际列名的占位符。以下格式的语句：

```
CREATE TABLE t1(c1 INTEGER CHECK (@foo < 5))
```

与下面的语句完全相同：

```
CREATE TABLE t1(c1 INTEGER CHECK (c1 < 5))
```

表检查约束中以符号 '@' 开头的列标识符不是占位符。

如果一条语句会引起数据库的更改并且这些更改会使数据库违反完整性约束，则该语句不会得到有效执行，并且系统会报告一条错误。（有效是指系统检测到错误之前该语句所做的任何更改都被撤消。）

SAP Sybase IQ 通过为该列创建 HG 索引来强制执行单列 UNIQUE 约束。

注意： 不能将含有 BIT 数据类型的列定义为 UNIQUE 或 PRIMARY KEY 约束。另外，BIT 数据类型的列缺省设置为不允许 NULL 值；您可以通过显式定义该列允许 NULL 值来更改这种情况。

- **column-constraint** – 限制列可以容纳的值。列约束和表约束有助于确保数据库中数据的完整性。如果语句会导致违反约束，则该语句的执行不会完成。该语句在检测到错误前所做的任何更改都被撤消并报告错误。列约束是相应表约束的缩写。例如，下列语句是等效的：

```
CREATE TABLE Products (
    product_num integer UNIQUE
)
CREATE TABLE Products (
    product_num integer,
    UNIQUE ( product_num )
)
```

通常使用列约束，除非约束引用了表中的多个列。此类情况下，必须使用表约束。

- **IQ UNIQUE** – 定义列的预期基数并确定是作为 Flat FP 还是 NBit FP 来装载该列。IQ UNIQUE(*n*) 值显式设置为 0 会作为 Flat FP 装载该列。不带 IQ UNIQUE 约束的列将隐式装载为 NBit，一直到 FP_NBIT_AUTOSIZE_LIMIT、FP_NBIT_LOOKUP_MB 和 FP_NBIT_ROLLOVER_MAX_MB 选项定义的限制：
 - FP_NBIT_AUTOSIZE_LIMIT 限制装载为 NBit 的不同值的数量
 - FP_NBIT_LOOKUP_MB 为 NBit 字典总大小设置阈值
 - FP_NBIT_ROLLOVER_MAX_MB 为从 NBit 至 Flat FP 的隐式 NBit 切换设置字典大小
 - FP_NBIT_ENFORCE_LIMITS 强制执行 NBit 字典大小限制。此选项缺省设置为 OFF

不必使用 *n* 值小于 FP_NBIT_AUTOSIZE_LIMIT 的 IQ UNIQUE。自动大小调整功能会自动将所有中低基数列的大小调整为 NBit。如果想要将列装载为 Flat FP，或者想要在不同值的数量超出 FP_NBIT_AUTOSIZE_LIMIT 时将列装载为 NBit，请使用 IQ UNIQUE。

注意：

- 当指定高 IQ UNIQUE 值时，请考虑内存的使用情况。如果计算机资源受到限制，应避免带有 FP_NBIT_ENFORCE_LIMITS='OFF'（缺省设置）的装载。在 SAP Sybase IQ 16.0 之前，IQ UNIQUE *n* 值 > 16777216 时会切换到 Flat FP。在 16.0 中，支持对较大的 IQ UNIQUE 值进行标识化，但鉴于基数和列宽，该值可能需要非常多的内存资源。
 - BIT、BLOB 和 CLOB 数据类型不支持 NBit 字典压缩。如果 FP_NBIT_IQ15_COMPATIBILITY='OFF'，则在包含这些数据类型的 CREATE TABLE 或 ALTER TABLE 语句中指定的非零 IQ UNIQUE 列将返回一个错误。
-

- **column-constraint** 和 **table-constraint** 子句 – 列约束和表约束有助于确保数据库中数据的完整性。

- **PRIMARY KEY 或 PRIMARY KEY (column-name, ...)** – 表的主键由列出的列组成，指定的列都不可含有任何 NULL 值。SAP Sybase IQ 确保表中的每一行都有唯一的主键值。表只能有一个 PRIMARY KEY。

采用第二种格式 (PRIMARY KEY 后面跟有列的列表) 时，创建的主键所包括的列是按其定义的顺序排序的，而不是按列出的顺序排序。

将列指定为 PRIMARY KEY、FOREIGN KEY 或 UNIQUE 时，SAP Sybase IQ 会自动为其创建 High_Group 索引。对于多列主键，该索引位于主键上，而不是各个列上。为了获得最佳性能，应单独用 HG 或 LF 索引对各列创建索引。

- **REFERENCES primary-table-name [(primary-column-name)]** – 将列定义为主键的外键，或主表的唯一约束。通常，外键针对的是主键而不是唯一约束。如果指定主列名，则它必须与主表中的某一列相匹配，而该列受唯一约束或主键约束的制约，而且该约束必须仅包含这一列。否则，外键会引用第二个表的主键。主键与外键必须具有相同的数据类型、精度、小数位数和符号。仅会为单列外键创建非唯一单列 HG 索引。对于多列外键，SAP Sybase IQ 将创建非唯一复合 HG 索引。唯一或非唯一 HG 索引的多列组合键的最大宽度为 1KB。

临时表不能有引用基表的外键，而基表不能有引用临时表的外键。局部临时表不能有外键，也不能被外键引用。

- **FOREIGN KEY [role-name] [(...)] REFERENCES primary-table-name [(...)]** – 定义对另一个表中主键或唯一约束的外键引用。通常，外键针对的是主键而不是唯一约束。（此处所述的另一个表称为主表。）

如果未指定主表列名，主表列即是表的主键中的列。如果未指定外键列名，则外键列的列名与主表的列名相同。如果指定外键列名，则还必须指定主键列名，列名按照列表中的位置成对出现。

如果主表与外键表不一样，则表明唯一约束或主键约束已在被引用键上定义。被引用键和外键必须具有相同的列数，以及相同的数据类型、符号、精度和小数位数。

行的外键值必须作为主表中某一行的候选键值出现，除非外键的允许 null 值的列中有一列或多列包含 null 值。

所有非显式定义的外键列在创建时会被自动赋予与主表中相应列相同的数据类型。这些自动创建的列不能属于外表的主键。因此，必须显式创建同时在主键和外键中使用的列。

role-name 是外键的名称。*role-name* 的主要作用是区分同一表的两个外键。如果未指定 *role-name*，则按如下方式分配角色名：

1. 如果没有与表名同名的 *role-name* 外键，则将表名指派为 *role-name*。
2. 如果表名已被使用，则 *role-name* 为表名加上表的唯一 3 位零填充数字。

参照完整性操作可定义为维护数据库中的外键关系而采取的操作。更改主键值或从数据库表中删除主键值时，其它表中可能有一些相应的外键值需要以某种方式修改。可以指定 ON DELETE 子句，后跟 RESTRICT 子句。

- **RESTRICT** – 如果在数据库的其它地方存在相应的外键时您试图更新或删除主键值，则将生成错误。如果试图更新外键，以便通过候选键创建不匹配的新值，则将生成错误。这是缺省操作，除非您指定 **LOAD** 可以拒绝违反参照完整性的行。这样便会在语句级别强制实施参照完整性。

如果使用 **CHECK ON COMMIT** 时未指定任何操作，那么 **RESTRICT** 将被视为 **DELETE** 的一项操作。SAP Sybase IQ 不支持 **CHECK ON COMMIT**。

全局临时表不能有引用基表的外键，而基表不能有引用全局临时表的外键。局部临时表不能有外键，也不能被外键引用。

- **CHECK (条件)** – 不允许行违反条件。如果 **INSERT** 语句会导致行违反条件，则不允许进行该操作并且撤消语句的作用。

只有条件为 **FALSE** 时才会拒绝更改；特别是，当条件为 **UNKNOWN** 时，允许进行更改。SAP Sybase IQ 不会强制遵守 **CHECK** 条件。

注意： 如果可能，不要在 SAP Sybase IQ 中定义参照完整性外键-主键关系，除非您确定其中没有孤立外键。

- **远程表** – 忽略远程表的外键定义。引用远程表的局部表的外键定义也会被忽略。如果远程服务器支持主键，则主键定义会发送到此服务器。
- **PARTITION BY** – 将大表拆分为多个便于管理的小存储对象。各个分区共用父表的相同逻辑属性，但可放置在单独的 **dbspace** 中并分别进行管理。SAP Sybase IQ 支持多个表分区模式：
 - 散列分区
 - 范围分区
 - 复合分区

分区键是包含表分区键的一个或多个列。分区键可包含 **NULL** 和 **DEFAULT** 值，但不能包含：

- **LOB (BLOB 或 CLOB)** 列
- **BINARY 或 VARBINARY** 列
- 长度超过 255 个字节的 **CHAR** 或 **VARCHAR** 列
- **BIT** 列
- **FLOAT/DOUBLE/REAL** 列
- **PARTITION BY RANGE** – 根据分区列中的值范围对行进行分区。范围分区被限定为有 1 个分区键列和最多 1024 个分区。在范围分区模式中，分区键是包含表分区键的列：

```
range-partition-decl:
    partition-name VALUES <= ( {constant-expr | MAX } [ ,
{ constant-expr | MAX } ]... )
    [ IN dbspace-name ]
```

分区名称是存储表行的新分区的名称。在表分区集合中，分区名称必须唯一。分区名称是必需项。

- **VALUE** – 按升序为每个分区指定上限（含上限）。用户必须为每个范围分区指定分区条件，确保将每一行只分配到一个分区。分区列允许 **NULL** 值，将 **NULL** 作为分区键值的行属于第一个表分区。但 **NULL** 不能是界限值。

第一个分区没有下限（**MIN** 值）。分区键第一列中单元格值为 **NULL** 的行属于第一个分区。对于最后一个分区，您可以指定一个上限（含上限）或 **MAX**。如果最后一个分区的上限值不是 **MAX**，则加载或插入其分区键值大于最后一个分区上限值的任何行都将生成错误。

- **Max** – 表示无限大的上限，只能为最后一个分区指定。
- **IN** – 在 **partition-decl** 中指定其中应该包含分区行的 **dbspace**。

以下限制影响范围分区表的分区键和界限值：

- 分区界限必须是常量，而不是常量表达式。
- 分区界限必须按照分区的创建顺序以升序排列。也就是说，第二个分区的上限必须大于第一个分区的上限，依此类推。
此外，分区界限值必须与相应的分区键列数据类型兼容。例如，**VARCHAR** 与 **CHAR** 兼容。
- 如果某个界限值的数据类型不同于其对应分区键列的数据类型，**SAP Sybase IQ** 会将界限值的数据类型转换为分区键列的数据类型，但存在以下例外情况：
- 不允许进行显式转换。以下示例尝试将 **INT** 显式转换为 **VARCHAR**，并生成错误：

```
CREATE TABLE Employees(emp_name VARCHAR(20))
PARTITION BY RANGE(emp_name)
(p1 VALUES <=(CAST (1 AS VARCHAR(20))),
p2 VALUES <= (CAST (10 AS VARCHAR(20)))
```

- 不允许执行可导致数据丢失的隐式转换。在以下示例中，分区界限与分区键类型不兼容。舍入假设可导致数据丢失，并生成错误：

```
CREATE TABLE emp_id (id INT) PARTITION BY RANGE(id) (p1 VALUES
<= (10.5), p2 VALUES <= (100.5))
```

- 在以下示例中，分区界限与分区键数据类型兼容。界限值会直接转换为浮点值。无需进行舍入处理，且支持转换：

```
CREATE TABLE id_emp (id FLOAT)
PARTITION BY RANGE(id) (p1 VALUES <= (10),
p2 VALUES <= (100))
```

- 不允许将非二进制数据类型转换为二进制数据类型。例如，不允许进行以下转换，并会返回错误：

```
CREATE TABLE newemp (name BINARY)
PARTITION BY RANGE(name)
(p1 VALUES <= ("Maarten"),
p2 VALUES <= ("Zymlmerman"))
```

- 在范围分区表中，不能将 **NULL** 用作界限。
- 如果分区键第一列中的单元格值计算结果为 **NULL**，则对应行将属于第一个分区。**SAP Sybase IQ** 仅支持一列分区键，因此分区键中的任何 **NULL** 都会将对应行分配到第一个分区。

- **PARTITION BY HASH** – 基于内部散列函数处理的分区-键值向各个分区映射数据。散列分区键被限定为最多有 8 个列，其声明的列宽总和小于等于 5300 个字节。对于散列分区，表创建者仅确定分区键列；分区的数量和位置将在内部确定。

在散列分区声明中，分区键是一个或多个列，其合成值将确定存储有各行数据的分区：

```
hash-partitioning-scheme:
HASH ( partition-key [ , partition-key, ... ] )
```

- **限制 –**

- 只能对基表进行散列分区。试图对全局临时表或局部临时表进行分区将会引发错误。
- 不能添加、删除、合并或拆分散列分区。
- 不能为散列分区键添加列或从中删除列。

- **PARTITION BY HASH RANGE** – 按范围对已进行散列分区的表划分子分区。在散列范围分区模式声明中，**SUBPARTITION BY RANGE** 子句将向现有散列范围分区表添加一个新的范围子分区：

```
hash-range-partitioning-scheme:
PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
    [ SUBPARTITION BY RANGE ( range-partition-decl [ , range-
partition-decl ... ] ) ]
```

散列分区指定如何在逻辑上分布和托管数据；范围子分区指定如何在物理上放置数据。新的范围子分区通过散列进行逻辑分区，其散列分区键与现有的散列范围分区表相同。范围子分区键被限定为仅一列。

- **限制 –**

- 只能对基表进行散列分区。试图对全局临时表或局部临时表进行分区将会引发错误。
- 不能添加、删除、合并或拆分散列分区。
- 不能为散列分区键添加列或从中删除列。

注意：与散列范围分区相同，范围分区和复合分区模式要求有单独授权的 VLDB 管理组件。

示例

(返回顶部) (第 410 页)

- **示例 1** – 创建一个名为 SalesOrders2 的表，其中包含 5 列。FinancialCode、OrderDate 和 ID 列的数据页位于 dbspace Dsp3 中。整数列 CustomerID 的数据页位于 dbspace Dsp1 中。CLOB 列 History 的数据页位于 dbspace Dsp2 中。主键 (ID 的 HG) 的数据页位于 dbspace Dsp4 中：

```
CREATE TABLE SalesOrders2 (
    FinancialCode CHAR(2),
    CustomerID int IN Dsp1,
```

```
History CLOB IN Dsp2,  
OrderDate TIMESTAMP,  
ID BIGINT,  
PRIMARY KEY(ID) IN Dsp4  
) IN Dsp3
```

- **示例 2** – 创建一个名为 `fin_code2` 的表，其中包含 4 列。`code`、`type` 和 `id` 列的数据页位于数据库选项 `DEFAULT_DBSPACE` 的值所确定的缺省 `dbspace` 中。`CLOB` 列 `description` 的数据页位于 `dbspace Dsp2` 中。外键 `fk1` (`c1` 的 `HG`) 中的数据页位于 `dbspace Dsp4` 中：

```
CREATE TABLE fin_code2 (  
code INT,  
type CHAR(10),  
description CLOB IN Dsp2,  
id BIGINT,  
FOREIGN KEY fk1(id) REFERENCES SalesOrders(ID) IN Dsp4  
)
```

- **示例 3** – 创建表 `t1`，其中分区 `p1` 与 `p2` 相邻，分区 `p2` 与 `p3` 相邻：

```
CREATE TABLE t1 (c1 INT, c2 INT)  
PARTITION BY RANGE(c1)  
(p1 VALUES <= (0), p2 VALUES <= (10), p3 VALUES <= (100))
```

- **示例 4** – 创建包含 6 列和 3 个分区的 **RANGE** 分区表 `bar`，用于基于日期向各个分区映射数据：

```
CREATE TABLE bar (  
c1 INT IQ UNIQUE(65500),  
c2 VARCHAR(20),  
c3 CLOB PARTITION (P1 IN Dsp11, P2 IN Dsp12,  
P3 IN Dsp13),  
c4 DATE,  
c5 BIGINT,  
c6 VARCHAR(500) PARTITION (P1 IN Dsp21,  
P2 IN Dsp22),  
PRIMARY KEY (c5) IN Dsp2) IN Dsp1  
PARTITION BY RANGE (c4)  
(P1 VALUES <= ('2006/03/31') IN Dsp31,  
P2 VALUES <= ('2006/06/30') IN Dsp32,  
P3 VALUES <= ('2006/09/30') IN Dsp33  
) ;
```

每个分区的数据页分配：

分区	Dbospace	列
P1	Dsp31	c1、c2、c4、c5
P1	Dsp11	c3
P1	Dsp21	c6
P2	Dsp32	c1、c2、c4、c5

分区	Dbospace	列
P2	Dsp12	c3
P2	Dsp22	c6
P3	Dsp33	c1、c2、c4、c5、c6
P3	Dsp13	c3
P1、P2、P3	Dsp1	c1 的查找存储和其它共享数据
P1、P2、P3	Dsp2	主键 (c5 的 HG)

- **示例 5** – 创建包含 PRIMARY KEY (列 c1) 和 HASH PARTITION KEY (列 c4 和 c3) 的 HASH 分区表 (table tbl142)。

```
CREATE TABLE tbl142 (  
    c1 BIGINT NOT NULL,  
    c2 CHAR(2) IQ UNIQUE(50),  
    c3 DATE IQ UNIQUE(36524),  
    c4 VARCHAR(200),  
    PRIMARY KEY (c1)  
)  
PARTITION BY HASH ( c4, c3 )
```

- **示例 6** – 创建含有 PRIMARY KEY (列 c1)、散列分区键 (列 c4 和 c2) 和范围子分区键 (列 c3) 的散列范围分区表。

```
CREATE TABLE tbl142 (  
    c1 BIGINT NOT NULL,  
    c2 CHAR(2) IQ UNIQUE(50),  
    c3 DATE,  
    c4 VARCHAR(200),  
    PRIMARY KEY (c1)) IN Dsp1  
  
PARTITION BY HASH ( c4, c2 )  
SUBPARTITION BY RANGE ( c3 )  
( P1 VALUES <= (2011/03/31) IN Dsp31,  
  P2 VALUES <= (2011/06/30) IN Dsp32,  
  P3 VALUES <= (2011/09/30) IN Dsp33) ;
```

- **示例 7** – 为图书馆数据库创建用于保存借出图书信息的表：

```
CREATE TABLE borrowed_book (  
    date_borrowed DATE NOT NULL,  
    date_returned DATE,  
    book CHAR(20)  
    REFERENCES library_books (isbn),  
    CHECK( date_returned >= date_borrowed )  
)
```

- **示例 8** – 在远程服务器 SERVER_A 上创建表 t1，并创建映射到该远程表的名为 t1 的代理表：

```
CREATE TABLE t1
( a INT,
  b CHAR(10))
AT 'SERVER_A.db1.joe.t1'
```

- **示例 9** – 创建表 tab1，其中包含以特殊常量 LAST USER 为缺省值的列 c1:

```
CREATE TABLE tab1(c1 CHAR(20) DEFAULT LAST USER)
```

- **示例 10** – 创建包含列 c1 的局部临时表 tab1:

```
CREATE LOCAL TEMPORARY TABLE tab1(c1 int) IN IQ_SYSTEM_TEMP
```

在以下情况下，本例将在 IQ_SYSTEM_TEMP dbspace 中创建 tab1:

- DQP_ENABLED 逻辑服务器策略选项设置为 ON，但在 IQ_SHARED_TEMP 中没有读写文件
- DQP_ENABLED 选项为 OFF，TEMP_DATA_IN_SHARED_TEMP 逻辑服务器策略选项为 ON，但在 IQ_SHARED_TEMP 中没有读写文件
- DQP_ENABLED 选项和 TEMP_DATA_IN_SHARED_TEMP 选项均设置为 OFF

在以下情况下，本例将在 IQ_SHARED_TEMP dbspace 中创建相同的表 tab1:

- DQP_ENABLED 设置为 ON，且在 IQ_SHARED_TEMP 中具有读写文件
 - DQP_ENABLED 设置为 OFF，TEMP_DATA_IN_SHARED_TEMP 设置为 ON，且在 IQ_SHARED_TEMP 中具有读写文件
- **示例 11** – 创建表 tab1，允许在内存 RLV 存储库中使用行级版本控制和实时存储。

```
CREATE TABLE tab1 ( c1 INT, c2 CHAR(25) ) ENABLE RLV STORE
```

用法

(返回顶部) (第 410 页)

通过指定所有者名称，可为其他用户创建表。如果未指定 GLOBAL TEMPORARY 或 LOCAL TEMPORARY，该表被称为基表。否则，该表为临时表。

与基表一样，所创建的全局临时表也存在于数据库中，并一直保留在数据库中，直到通过 DROP TABLE 语句将其显式删除。临时表中的行仅对插入这些行的连接可见。来自相同或不同应用程序的多个连接可同时使用同一个临时表，但每个连接只能看到它自己的那些行。给定的连接会继承当该连接首次引用全局临时表时该表的模式。连接结束时删除临时表中的行。

创建局部临时表时，请不要指定所有者。如果在创建临时表时指定所有者（例如，CREATE TABLE dbo.#temp(coll int)），就会将其错误地创建成基表。

如果连接中存在某一局部临时表，则尝试创建同名的基表或全局临时表将失败，因为 owner.table 无法唯一地标识新表。

但是，当存在基表或全局临时表时，可创建同名的局部临时表。引用表名时将访问局部临时表，因为将首先解析局部临时表。

例如，请考虑以下序列：

```
CREATE TABLE t1 (c1 int);
INSERT t1 VALUES (9);

CREATE 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局部临时表 t1，一直到局部临时表被连接删除。

在过程中，如果要创建一个在过程完成后仍然保留的表，可使用 **CREATE LOCAL TEMPORARY TABLE** 语句，而不是 **DECLARE LOCAL TEMPORARY TABLE** 语句。使用 **CREATE LOCAL TEMPORARY TABLE** 语句创建的局部临时表会一直保留到它被显式删除或连接终止。

利用使用 **CREATE LOCAL TEMPORARY TABLE** 的 **IF** 语句创建的局部临时表，在 **IF** 语句完成后也会继续保留。

SAP Sybase IQ 不支持将 **CREATE TABLE ENCRYPTED** 子句用于对 SAP Sybase IQ 表进行表级别加密。但 SAP Sybase IQ 数据库中的 SQL Anywhere 表支持 **CREATE TABLE ENCRYPTED** 子句。

Side Effects

- 自动提交

标准

(返回顶部) (第 410 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
以下是服务商扩充：
 - { **IN** | **ON** } *dbspace-name* 子句
 - **ON COMMIT** 子句
 - 部分缺省值
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持，但存在一些差异。
 - **临时表** - 通过在 **CREATE TABLE** 语句中的表名前加上井号 (#)，可创建临时表。这些临时表是 SAP Sybase IQ 声明的临时表，只能在当前连接中可用。有关声明的临时表的信息，请参见 **DECLARE LOCAL TEMPORARY TABLE** 语句。
 - **实际放置** - SAP Sybase IQ 中表的实际放置方式不同于 Adaptive Server 中表的实际放置方式。Adaptive Server 支持的 **ON segment-name** 子句在 SAP Sybase IQ 中也受支持，但 *segment-name* 指的是 IQ dbspace。

- **约束** – SAP Sybase IQ 不支持已命名的约束或已命名的缺省值，但确实支持允许将约束和缺省值定义封装在数据类型定义中的用户定义数据类型。它还支持在 **CREATE TABLE** 语句中使用显式缺省值和 **CHECK** 条件。
- **NULL** – （缺省值）缺省情况下，Adaptive Server 中的列缺省设置为 NOT NULL，而在 SAP Sybase IQ 中，缺省设置为 NULL，从而允许 NULL 值。可使用 `ALLOW_NULLS_BY_DEFAULT` 选项控制此设置。请参见 `ALLOW_NULLS_BY_DEFAULT` 选项 [TSQL]。要使您的数据定义语句成为可移植的语句，应显式指定 NULL 或 NOT NULL。

权限

(返回顶部) (第 410 页)

表类型	所需特权
IQ 主存储库中的基表	<p>自有表 - 针对创建表的 <code>dbspace</code> 需要具备 CREATE 特权。还需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE TABLE 系统特权。• CREATE ANY OBJECT 系统特权。 <p>由任何用户拥有的表 - 需要对创建表的 <code>dbspace</code> 具有 CREATE 特权。还需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE ANY TABLE 系统特权。• CREATE ANY OBJECT 系统特权。
全局临时表	<p>由自身拥有的表 – 需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE TABLE 系统特权。• CREATE ANY OBJECT 系统特权。 <p>由任何用户拥有的表 – 需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE ANY TABLE 系统特权。• CREATE ANY OBJECT 系统特权。
代理表	<p>由自身拥有的表 – 需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE PROXY TABLE 系统特权。• CREATE ANY TABLE 系统特权。• CREATE ANY OBJECT 系统特权。 <p>由任何用户拥有的表 – 需要具备以下特权之一：</p> <ul style="list-style-type: none">• CREATE ANY TABLE 系统特权。• CREATE ANY OBJECT 系统特权。

另请参见

- 限制 (第 126 页)

- 范围分区 (第 127 页)
- 散列分区 (第 128 页)
- 散列范围分区 (第 129 页)

DROP 语句

从数据库中删除对象。

快速链接:

[转至参数 \(第 425 页\)](#)

[转至示例 \(第 426 页\)](#)

[转至用法 \(第 426 页\)](#)

[转至标准 \(第 427 页\)](#)

[转至权限 \(第 427 页\)](#)

语法

DROP

```
{ DBSPACE dbspace-name
| { DATATYPE [ IF EXISTS ]
| DOMAIN } datatype-name
| EVENT [ IF EXISTS ] event-name
| INDEX [ IF EXISTS ] [ [ owner ].table-name ].index-name
| MESSAGE message-number
| TABLE [ IF EXISTS ] [ owner ].table-name
| VIEW [ IF EXISTS ] [ owner ].view-name
| MATERIALIZED VIEW [ IF EXISTS ] [ owner ].view-name
| PROCEDURE [ IF EXISTS ] [ owner ].procedure-name
| FUNCTION [ IF EXISTS ] [ owner ].function-name }
```

参数

(返回顶部) (第 425 页)

- **IF EXISTS** – 如果您不希望在 **DROP** 语句试图删除不存在的数据库对象时返回错误，请使用此子句。
- **INDEX** – 删除任何显式创建的索引。仅当不存在唯一约束或外键约束或者不存在关联的主键时，此语句才删除隐式创建的索引。

DROP INDEX 如果存在关联的未实施外键，则删除非唯一 HG 索引失败。

警告! 请勿删除由 DBO 用户拥有的视图。删除此类视图或将它们转化为表可能导致出现问题。

只要 **DROP INDEX** 语句影响了当前正由其它连接使用的表，就会禁止该语句。

- **TABLE** – 如果主表具有与 **DROP TABLE** 相关联的外键约束（包括未实施的外键约束），则会禁止该语句

如果相应表包含 **IDENTITY** 列且 **IDENTITY_INSERT** 设置为该表，也会禁止 **DROP TABLE**。要删除该表，必须清除 **IDENTITY_INSERT**（即将 **IDENTITY_INSERT** 设置为空字符串 ''），或者将其设置为其它表名。

外键可以具有非唯一的单列或多列 **HG** 索引。主键可以具有唯一的单列或多列 **HG** 索引。无法删除为现有外键、主键和唯一约束隐式创建的 **HG** 索引。

四个初始 **dbspace** 为 **SYSTEM**、**IQ_SYSTEM_MAIN**、**IQ_SYSTEM_TEMP** 和 **IQ_SYSTEM_MSG**。您不能删除这些初始 **dbspace**，但您可以从可能包含多个 **dbspace** 的 **IQ** 主存储库或目录存储库中删除 **dbspace**，前提是至少一个 **dbspace** 仍处于读写模式。

必须先将 **dbspace** 中的表删除，然后才能删除 **dbspace**。如果该 **dbspace** 仍包含用户数据，则会返回错误；该 **dbspace** 删除后，将自动重新分配其它结构。只有使 **dbspace** 处于只读模式后，才能删除它。

注意： **dbspace** 被命令使用后，在任何时刻都可能包含数据，因而会禁止对它执行 **DROP DBSPACE**。

只要 **DROP TABLE** 语句影响了当前正由其它连接使用的表，就会禁止该语句。

- **PROCEDURE** – 其它连接正在使用相应过程时，禁止执行 **DROP PROCEDURE**。
- **DATATYPE** – 如果表中用到相应数据类型，则会禁止执行 **DROP DATATYPE**。要删除数据类型，必须更改以某种用户定义数据类型定义的所有列的数据类型。建议使用 **DROP DOMAIN** 而不是 **DROP DATATYPE**，因为 **DROP DOMAIN** 是 ANSI/ISO SQL3 草案中使用的语法。

示例

(返回顶部) (第 425 页)

- **示例 1** – 从数据库中删除 **Departments** 表：

```
DROP TABLE Departments
```

- **示例 2** – 从数据库中删除 **emp_dept** 视图：

```
DROP VIEW emp_dept
```

- **示例 3** – 从您连接的 **Simplex** 或 **Multiplex** 节点删除 **myDAS** 主高速缓存：

```
DROP DBSPACE myDAS
```

用法

(返回顶部) (第 425 页)

DROP 会移除指定数据库结构的定义。如果该结构是 **dbspace**，则必须先删除或重新分配该 **dbspace** 中包含任何数据的所有表，再删除该 **dbspace**；将自动重新分配其它结构。如果该结构是表，则在删除过程中将自动删除该表中的所有数据。另外，**DROP TABLE** 语句还会删除该表的所有索引和键。

只要 **DROP DBSPACE** 语句影响了当前正由其它连接使用的表，就会禁止该语句。其它连接正在使用相应过程时，禁止执行 **DROP PROCEDURE**。

副作用

- 自动提交。清除 **dbisql** 中的数据窗口。**DROP TABLE** 和 **DROP INDEX** 用于关闭当前连接的所有游标。
- 局部临时表是个特例，删除它时不执行提交。

标准

(返回顶部) (第 425 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持。

权限

(返回顶部) (第 425 页)

DBSPACE 子句 - 需要 **DROP ANY OBJECT** 系统特权并且用户必须是数据库的唯一连接。

DOMAIN 子句 - 需要以下特权之一：

- **DROP DATATYPE** 系统特权。
- **DROP ANY OBJECT** 系统特权。
- 您拥有该对象。

FUNCTION 子句 - 需要以下特权之一：

- **DROP ANY PROCEDURE** 系统特权。
- **DROP ANY OBJECT** 系统特权。
- 您拥有该函数。

INDEX 子句 - 需要以下特权之一：

- **DROP ANY INDEX** 系统特权。
- **DROP ANY OBJECT** 系统特权。
- 对要建立索引的基础表的 **REFERENCES** 特权。
- 您拥有要建立索引的基础表。

DBA 或具有适当特权的用户可以在不使用完全限定名的情况下删除其他用户拥有的表的索引。所有其他用户必须提供完全限定的索引名才能删除 DBA 拥有的基表的索引。

MATERIALIZED VIEW 子句 - 需要以下特权之一：

- **DROP ANY MATERIALIZED VIEW** 系统特权。

- DROP ANY OBJECT 系统特权。
- 您拥有实例化视图。

PROCEDURE 子句 - 需要以下特权之一：

- DROP ANY PROCEDURE 系统特权。
- DROP ANY OBJECT 系统特权。
- 您拥有该过程。

TABLES 子句 - 需要以下特权之一：

- DROP ANY TABLE 系统特权。
- DROP ANY OBJECT 系统特权。
- 您拥有该表。

除非引用全局临时表的所有用户都已经断开连接，否则不能删除相应的全局临时表。

VIEW 子句 - 需要以下特权之一：

- DROP ANY VIEW 系统特权。
- DROP ANY OBJECT 系统特权。
- 您拥有该视图。

所有其它子句 - 需要以下特权之一：

- DROP ANY OBJECT 系统特权。
- 您拥有该对象。

DROP LOGICAL SERVER 语句

删除用户定义的逻辑服务器。此语句强制共享系统临时存储设置在逻辑服务器共享的物理节点中保持一致。

快速链接：

[转至参数](#)（第 428 页）

[转至示例](#)（第 429 页）

[转至用法](#)（第 429 页）

[转至权限](#)（第 429 页）

语法

```
DROP LOGICAL SERVER logical-server-name  
[ WITH STOP SERVER ]
```

参数

[\(返回顶部\)](#)（第 428 页）

- **WITH STOP SERVER** – 在 `TEMP DATA IN SHARED_TEMP` 选项发生直接或间接更改时，会自动关闭逻辑服务器中的所有服务器。

应用于

仅限 **Multiplex**。

示例

(返回顶部) (第 428 页)

- **示例 1** – 删除用户定义的逻辑服务器 `ls1`:

```
DROP LOGICAL SERVER ls1
```

用法

(返回顶部) (第 428 页)

在删除逻辑服务器时，**SAP Sybase IQ** 将内部执行以下目录更改操作：

- 删除逻辑服务器的所有成员资格定义。
- 从显式分配给主逻辑服务器的各个登录策略中删除逻辑服务器分配。如果该服务器是分配给登录策略的唯一逻辑服务器，则 **SAP Sybase IQ** 会将该登录策略的逻辑服务器分配设置为 **NONE**。
- 从 **ISYSIQ.LOGICALSERVER** 中删除逻辑服务器条目。

权限

(返回顶部) (第 428 页)

需要 **MANAGE MULTIPLEX** 系统特权。

GRANT INTEGRATED LOGIN 语句

在一个或多个 **Windows** 用户配置文件与一个现有数据库用户 **ID** 之间创建显式集成登录映射，使成功登录到本地计算机的用户不必提供用户 **ID** 或口令就可以连接到数据库。

快速链接：

转至参数 (第 430 页)

转至标准 (第 430 页)

转至权限 (第 430 页)

语法

GRANT INTEGRATED LOGIN

```
TO user_profile_name [, ...]
AS USER userID [,...]
```

参数

(返回顶部) (第 429 页)

- **userID** – 必须是具有登录口令的现有用户或角色的名称。多个 userID 用逗号分隔。

标准

(返回顶部) (第 429 页)

- SQL - 其它语法是 ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - Adaptive Server 和 SAP Sybase IQ 中的安全模型不同，所以其它语法也不同。

权限

(返回顶部) (第 429 页)

需要 **MANAGE ANY USER** 系统特权。

IQ UTILITIES 语句

启动一个用于收集缓冲区高速缓存统计信息的高速缓存监控器。

快速链接：

[转至参数](#) (第 431 页)

[转至示例](#) (第 432 页)

[转至用法](#) (第 433 页)

[转至标准](#) (第 433 页)

[转至权限](#) (第 433 页)

语法

```
IQ UTILITIES { MAIN | PRIVATE }  
  [ INTO ] table-name  
  { START MONITOR [ 'monitor-options' ]  
  | STOP MONITOR }
```

```
monitor-options  
  {  
  | -summary  
  | { -append | -truncate } -bufalloc  
  | -cache  
  | -cache_by_type  
  | -contention  
  | -debug  
  | -file_suffix suffix  
  | -io
```

```
| -interval seconds  
| -threads }...
```

参数

(返回顶部) (第 430 页)

- **START MONITOR** - 启动 IQ 缓冲区高速缓存监控器。
- **MAIN** - 监控 IQ 存储库的主缓冲区高速缓存中的所有表。
- **PRIVATE** - 监控临时存储库的临时缓冲区高速缓存中的所有表。
- **dummy_table_name** - 可以是任意 SAP Sybase IQ 基表或临时表。为了与其它 **IQ UTILITIES** 命令语法兼容，必须提供表名。最好有一个仅用于监控的表。
- **monitor_options** - 控制缓冲区高速缓存监控器输出。可以指定多个，且必须用引号括起。

选项	描述
-summary	显示主缓冲区高速缓存和临时缓冲区高速缓存的汇总信息。如果不指定任何监控器选项，则会收到汇总报告。用法： monitor_options -summary
-cache	详细显示主缓冲区高速缓存或临时缓冲区高速缓存的活动。关键字段是 Finds、HR% 和 BWait。用法： monitor_options -cache
-cache_by_type	按照 IQ 页类型拆分 -cache 结果。（Bwaits 列除外，此列仅显示总计。）当需要向技术支持部门提供信息时，此格式最有用。用法： monitor_options -cache_by_type
-file_suffix	创建一个名为 <dbname>.<connid>-<main_or_temp>-<suffix> 的监控器输出文件。如果您没有指定可选文件扩展名，则缺省情况下文件扩展名为 .iqmon。用法： monitor_options -file_suffix {extension}
-io	显示在指定时间间隔内主缓冲区高速缓存或临时（专用）缓冲区高速缓存 I/O 速率和压缩率。这些计数器表示服务器的所有活动；这些信息不会被设备断开。用法： monitor_options -io
-bufalloc	显示有关主缓冲区分配器或临时缓冲区分配器的信息，该分配器可在缓冲区高速缓存中保留空间以供排序、散列和位图等对象使用。用法： monitor_options -bufalloc

选项	描述
-contention	显示多个关键缓冲区高速缓存和内存管理器锁。这些锁和互斥计数器显示在缓冲区高速缓存和堆内存中的活动，以及解除这些锁的速度。如果超时数超过 20%，则表示有问题。用法： <code>monitor_options -contention</code>
-threads	显示处理线程管理器计数。这些值适用于整个服务器（即，无论为主缓冲区高速缓存还是专用缓冲区高速缓存选中此选项均适用）。用法： <code>monitor_options -threads</code>
-interval	指定报告间隔（以秒为单位）。缺省值是每 60 秒一次。最小值是每 2 秒一次。 在运行有关性能问题的查询期间或在出现性能问题的那一天中的某个时间，以缺省间隔运行监控器通常可以获得有用的结果。较短的间隔可能不会给出有意义的结果。间隔应与作业时间成比例；一分钟通常就足够了。用法： <code>monitor_options -interval</code>
-append -truncate	将输出附加到现有输出文件或截断现有输出文件。截断是缺省设置。用法： <code>monitor_options -append -truncate</code>
-debug	显示性能监控器的所有可用信息，无论是否有包含相同信息的标准显示模式。 <code>-debug</code> 主要用于向技术支持部门提供信息。用法： <code>monitor_options -debug</code>

- **STOP MONITOR** – 与 **START MONITOR** 相似，但不需要指定任何选项：

注意：

- 若要简化监控器的使用，请创建存储过程以声明虚表，并指定其输出位置，然后启动监控器。
- 间隔（有两个例外）应用于每行输出，而不是每页。这两个例外是 `-cache_by_type` 和 `-debug` 子句，其中每次显示都会出现一个新页。

示例

(返回顶部) (第 430 页)

- **示例 1** – 启动缓冲区高速缓存监控器并记录 IQ 临时缓冲区高速缓存的活动：

```
IQ UTILITIES PRIVATE INTO monitor START MONITOR '-cache -interval 20'
```


用法

(返回顶部) (第 430 页)

发出单独命令来监控每个缓冲区高速缓存。在监控器收集结果时，使每个会话保持打开状态；关闭其连接时，监控器的运行会停止。一个连接可以运行最多两个监控器运行，一个用于主缓冲区高速缓存，另一个用于临时缓冲区高速缓存。

要控制监控器输出文件的存放目录，请设置 **MONITOR_OUTPUT_DIRECTORY** 选项。如果不设置此选项，则监控器会将输出发送到与数据库相同的目录。所有监控器输出文件均用于监控器运行期间。监控器运行停止后，这些文件会保留。

在创建任何 **Multiplex** 查询服务器之前，可以声明在监控中使用的临时表，或者在创建新数据库时创建永久性虚表。这些解决方案避免了 **DDL** 更改，以便在生产运行期间数据在查询服务器上保持活动状态。

在类似 **UNIX** 的操作系统上，可以在查询运行时监视监控器输出。例如：

使用此命令启动监控器：

```
iq utilities main into monitor_tab
start monitor "-cache -interval 2 -file_suffix iqmon"
```

将输出发送到名为 `dbname.conn#-[main|temp]-iqmon` 的 **ASCII** 文件。因此，对于 **iqdemo** 数据库，缓冲区监控器会将结果发送到 `iqdemo.2-main-iqmon`

缓冲区高速缓存监控器会将每次运行的结果写入到这些日志中：

- `dbname.connection#-main-iqmon` //for main buffer cache results
- `dbname.connection#-temp-iqmon` //for temp buffer cache results

前缀 `dbname.connection#` 表示数据库名称和连接编号。如果看到多个连接编号，且不能确定哪个是您的，则可以运行目录存储过程 **sa_conn_info**。此过程将为与数据库相连的每个活动连接显示连接编号、用户 **ID** 及其它信息。`-file_suffic` 子句将后缀 `iqmon` 更改为您所选择的后缀。使用文本编辑器显示或打印文件。请从相同数据库和连接编号再次运行监控器，覆盖之前的结果。要保存监控器的运行结果，可以将文件复制到其它位置或使用 `-append` 选项。

标准

(返回顶部) (第 430 页)

- **SQL - ISO/ANSI SQL** 语法的服务商扩充。
- **SAP Sybase 数据库产品** - 不受 **Adaptive Server** 支持。

权限

(返回顶部) (第 430 页)

无

LOCK TABLE 语句

禁止其它并发事务在指定时间内访问或修改表。

快速链接：

转至参数（第 434 页）

转至示例（第 435 页）

转至用法（第 435 页）

转至标准（第 436 页）

转至权限（第 436 页）

语法

```
LOCK TABLE table-list [ WITH HOLD ]  
      IN { SHARE | WRITE | EXCLUSIVE } MODE [ WAIT time ]  
  
table-list  
      [ owner. ] table-name [ , [ owner. ] table-name, ...]
```

参数

(返回顶部)（第 434 页）

- **table-name** – 必须是基表，而不是视图。**WRITE** 模式仅对 IQ 基表有效。**LOCK TABLE** 会锁定表列表中的所有表，或者不锁定任何表。必须为行级版本控制 (RLV) 启用该表。当为 SQL Anywhere 表获取锁，或者当获取 **SHARE** 或 **EXCLUSIVE** 锁时，只能指定一个表。标准 SAP Sybase IQ 对象限定规则用于分析 *table-name*。
- **WITH HOLD** – 该锁会一直保持到连接结束时为止。如果未指定此子句，则提交或回退当前事务时将释放锁定。不支持在同一语句中使用 **WITH HOLD** 和 **WRITE MODE** 子句，这将返回错误 SQLCODE=-131, ODBC 3 State="42000"。
- **SHARE** – 防止其它事务修改表，但允许它们读取访问。在此模式中，只要没有其它的事务锁定要修改的行，就可以间接地或显式地使用 **LOCK TABLE** 更改表中的数据。
- **WRITE** – 防止其它事务修改表的列表。无条件提交连接最外层的事务。事务的快照版本不是由 **LOCK TABLE IN WRITE MODE** 语句建立，而是通过执行由 SAP Sybase IQ 处理的下一命令建立。

当事务提交或回退或者断开连接时，会解除 **WRITE** 模式锁。

- **EXCLUSIVE** – 防止其它事务访问表。在此模式中，任何其它事务都不能对表执行任何种类的查询、更新或任何其它操作。
- **WAIT time** – 指定所有锁类型的最大阻塞时间。当锁定模式为 **WRITE** 时，此子句为必填。给定时间参数后，仅当服务器在指定时间内可用时，其才会锁定指定

表。可以使用 **hh:nn:ss:sss** 格式指定时间参数。如果指定了日期部分，服务器将忽略此部分，并将参数转换为时间戳。如果未给定时间参数，服务器将无限期等待，直到 **WRITE** 锁可用或发生中断为止。

示例

(返回顶部) (第 434 页)

- **示例 1** - 在 Customers 和 Employees 表中获取 **WRITE** 锁（如果在 5 分 3 秒内可用）：

```
LOCK TABLE Customers, Employees IN WRITE MODE WAIT
'00:05:03'
```

- **示例 2** - 无限期等待，直到在 Customers 和 Employees 表上的 **WRITE** 锁可用或发生中断为止：

```
LOCK TABLE Customers, Employees IN WRITE MODE WAIT
```

用法

(返回顶部) (第 434 页)

在协调器的 IQ 主存储库中的表上运行 **LOCK TABLE** 语句，不会影响从辅助服务器上的连接访问这些表。例如：

在协调器连接上，发出命令：

```
LOCK TABLE coord1 WITH HOLD IN EXCLUSIVE MODE
```

协调器上的 **sp_iqlocks** 会确认表 coord1 具有排它 (E) 锁。

在辅助服务器的连接上运行的 **sp_iqlocks** 的结果不会显示表 coord1 上的排它锁。此连接上的用户可查看协调器上表 coord1 的更新。

协调器上的其它连接可查看 coord1 上的排它锁，并尝试通过返回用户 DBA 具有在 coord1 中锁定的行的协调器上的其它连接从表 coord1 中选择。

不支持在视图上执行 **LOCK TABLE**。尝试锁定视图将获取共享模式锁，而不管命令中指定的模式如何。共享模式锁会阻止其它事务修改表模式。

Transact-SQL (T-SQL) 存储过程方言不支持 **LOCK TABLE**。例如，此语句返回 LOCK 附近的语法错误：

```
CREATE PROCEDURE tproc()
AS
BEGIN
COMMIT;
LOCK TABLE t1 IN SHARE MODE
INSERT INTO t1 VALUES(30)
END
```

Watcom-SQL 存储过程语言支持 **LOCK TABLE**。缺省命令分隔符为分号 (;)。例如：

```
CREATE PROCEDURE tproc()  
AS  
BEGIN  
COMMIT;  
LOCK TABLE t1 IN SHARE MODE  
INSERT INTO t1 VALUES(30)  
END
```

标准

(返回顶部) (第 434 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持。Adaptive Server 不支持 WITH HOLD 子句。Adaptive Server 提供不受 SQL Anywhere 支持的 WAIT 子句。

权限

(返回顶部) (第 434 页)

若要以 SHARE 模式锁定表，需要 SELECT 权限。

若要以 EXCLUSIVE 模式锁定表，则必须是表所有者或者具有以下系统特权之一：

- ALTER ANY OBJECT
- INSERT ANY TABLE
- UPDATE ANY TABLE
- DELETE ANY TABLE
- ALTER ANY TABLE
- LOAD ANY TABLE
- TRUNCATE ANY TABLE

ROLLBACK 语句

撤消自上次 **COMMIT** 或 **ROLLBACK** 以来所做的任何更改。

快速链接：

转至用法 (第 436 页)

转至标准 (第 437 页)

转至权限 (第 437 页)

语法

```
ROLLBACK [ WORK ]
```

用法

(返回顶部) (第 436 页)

ROLLBACK 结束一个逻辑工作单元（事务），并撤消该事务期间内对数据库进行的所有更改。事务是指在一次数据库连接中 **COMMIT** 或 **ROLLBACK** 语句所做的数据库工作。

副作用

- 关闭所有不以 **WITH HOLD** 方式打开的游标。
- 释放由发出 **ROLLBACK** 的事务所持有的锁。

标准

(返回顶部) (第 436 页)

- SQL - 符合 ISO/ANSI SQL 标准。
- SAP Sybase 数据库产品 - 受 Adaptive Server 支持。

权限

(返回顶部) (第 436 页)

无，但用户必须连接到数据库。

SAVEPOINT 语句

在当前事务中建立保存点。

快速链接：

转至参数 (第 437 页)

转至用法 (第 437 页)

转至标准 (第 438 页)

转至权限 (第 438 页)

语法

```
SAVEPOINT [ savepoint-name ]
```

参数

(返回顶部) (第 437 页)

- **savepoint-name** - 是可用于 **RELEASE SAVEPOINT** 或 **ROLLBACK TO SAVEPOINT** 语句的标识符。

用法

(返回顶部) (第 437 页)

事务结束后，所有保存点都自动释放。

执行触发器或原子复合语句时建立的保存点在原子操作结束时自动释放。

标准

(返回顶部) (第 437 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。若要以 Adaptive Server 兼容的方式实现类似的功能，可使用嵌套事务。

权限

(返回顶部) (第 437 页)

无

SET OPTION 语句

更改影响数据库行为及数据库与 Transact-SQL 兼容性的选项。设置选项的值可更改所有用户或某个用户的行为，作用域可以是临时的，也可以是永久的。

快速链接：

转至参数 (第 438 页)

转至示例 (第 439 页)

转至用法 (第 439 页)

转至标准 (第 440 页)

转至权限 (第 440 页)

语法

```
SET [ EXISTING ] [ TEMPORARY ] OPTION  
    ... [ userid. | PUBLIC.]option-name = [ option-value ]
```

参数

(返回顶部) (第 438 页)

- **option-value** - 主机变量（允许使用指示符）、字符串、标识符或数字。*option-value* 设置为字符串时，其最大长度为 127 个字节。

如果忽略 *option-value*，将从数据库中删除指定的选项设置。如果它是个人选项设置，则所用的值会恢复为 PUBLIC 设置。

注意： 对于所有接受整数值的数据库选项，SAP Sybase IQ 会将任何小数形式的 *option-value* 设置截断为整数值。例如，值 3.8 将被截断为 3。

- **EXISTING** – 如果某选项没有 PUBLIC 用户 ID 设置，则无法为单个用户 ID 设置此选项的值。
- **TEMPORARY** – 更改选项更改有效性的持续时间。如果没有 TEMPORARY 子句，则对选项的更改将是永久性更改：在使用 **SET OPTION** 进行显式更改之前，它不会发生变化。

如果使用单个用户 ID 应用 TEMPORARY 子句，则只要用户在数据库中处于登录状态，新选项值就会一直有效。

如果通过 PUBLIC 用户 ID 使用 TEMPORARY 子句，则更改在数据库运行时间内将一直有效。当数据库关闭时，PUBLIC 用户 ID 的 TEMPORARY 选项恢复为其永久值。

如果删除一个 TEMPORARY 选项，则选项设置会恢复为永久设置。

示例

(返回顶部) (第 438 页)

- **示例 1** – 设置 DATE_FORMAT 选项：

```
SET OPTION public.date_format = 'Mmm dd yyyy'
```

- **示例 2** – 将 WAIT_FOR_COMMIT 选项设置为 on：

```
SET OPTION wait_for_commit = 'on'
```

- **示例 3** – 嵌入式 SQL 示例：

```
EXEC SQL SET OPTION :user.:option_name = :value;
EXEC SQL SET TEMPORARY OPTION Date_format = 'mm/dd/yyyy';
```

用法

(返回顶部) (第 438 页)

选项的分类如下：

- 常规数据库选项
- Transact-SQL 兼容性数据库选项

指定用户 ID 或 PUBLIC 用户 ID 可确定该选项是为单个用户、为由 *userid* 表示的角色，还是为 PUBLIC 用户 ID（所有用户都属于该角色的成员）设置的。如果选项适用于角色 ID，则角色成员不会继承选项设置，即，仅将更改应用于角色 ID。如果未指定角色，则将所做选项更改应用于发出 **SET OPTION** 语句的当前登录用户 ID。例如，以下语句对 PUBLIC 用户 ID 应用选项更改：

```
SET OPTION Public.login_mode = standard
```

在嵌入式 SQL 中，只有数据库选项可以临时设置。

为 PUBLIC 用户 ID 更改选项的值，相当于为没有设置该值的任意用户设置此选项的值。如果某选项没有 PUBLIC 用户 ID 设置，则无法为单个用户 ID 设置此选项的值。

相对于永久性地设置选项的值，临时设置 **PUBLIC** 用户 ID 的选项更具安全优势。例如，在启用 **LOGIN_MODE** 选项时，数据库依赖于其所运行的系统的登录安全性。临时启用该选项意味着，对于依赖于 **Windows** 域的安全性的数据库，如果关闭该数据库并将它复制到本地计算机，它的安全不会受到威胁。在这种情况下，临时启用的 **LOGIN_MODE** 将恢复为它的永久值（可能是“标准”模式，这种模式不允许集成登录）。

警告！ 不支持从游标中读取行时更改选项设置，因为这会导致意外的行为。例如，在从游标中读取时更改 **DATE_FORMAT** 设置会在结果集的行中返回不同的日期格式。不要在读取行时更改选项设置。

标准

(返回顶部) (第 438 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 不受 Adaptive Server 支持。SAP Sybase IQ 支持一些使用 **SET** 语句的 Adaptive Server 选项。

权限

(返回顶部) (第 438 页)

设置自身的选项不需要特定系统特权。

必须具有 **SET ANY PUBLIC OPTION** 系统特权才能为其他用户设置数据库选项。

必须具有 **SET ANY SYSTEM OPTION** 系统特权才能为 **PUBLIC** 用户 ID 设置 **SYSTEM** 选项。

必须具有 **SET ANY SECURITY OPTION** 系统特权才能为 **PUBLIC** 用户 ID 设置 **SECURITY** 选项。

STOP DATABASE 语句 [Interactive SQL]

停止指定数据库服务器上的数据库。

快速链接：

转至参数 (第 441 页)

转至示例 (第 441 页)

转至标准 (第 441 页)

转至权限 (第 441 页)

语法

```
STOP DATABASE database-name
... [ ON engine-name ]
... [ UNCONDITIONALLY ]
```


参数

(返回顶部) (第 440 页)

- **database-name** - 启动数据库时在 **-n** 参数中指定的名称，或者在 **DBN (DatabaseName)** 连接参数中指定的名称。此名称通常是保存目录存储的数据库文件的文件名，不带 **.db** 扩展名，但也可以是用户定义的任何名称。
- **engine-name** - 如果未指定，则在所有运行的引擎中搜索具有指定名称的数据库。
- **UNCONDITIONALLY** - 如果指定，即使存在到数据库的连接也停止数据库。如果未指定，若存在到数据库的连接，则不停止该数据库。

示例

(返回顶部) (第 440 页)

- **示例 1** - 停止缺省服务器上名为 **sample** 的数据库：

```
STOP DATABASE sample
```

标准

(返回顶部) (第 440 页)

- **SQL - ISO/ANSI SQL** 语法的服务商扩充。
- **SAP Sybase 数据库产品** - 不适用。

权限

(返回顶部) (第 440 页)

需要 **SERVER OPERATOR** 系统特权。

STOP ENGINE 语句 [Interactive SQL]

停止数据库服务器。

快速链接：

转至参数 (第 442 页)

转至示例 (第 442 页)

转至标准 (第 442 页)

转至权限 (第 442 页)

语法

```
STOP ENGINE engine-name [ UNCONDITIONALLY ]
```

参数

(返回顶部) (第 441 页)

- **UNCONDITIONALLY** - 如果指定，即使存在到数据库服务器的连接也会停止该数据库服务器。如果未指定，如果存在到数据库服务器的连接，则不停止该数据库服务器。

示例

(返回顶部) (第 441 页)

- **示例 1** - 停止名为 sample 的数据库服务器：

```
STOP ENGINE sample
```

标准

(返回顶部) (第 441 页)

- SQL - ISO/ANSI SQL 语法的服务商扩充。
- SAP Sybase 数据库产品 - 不适用。

权限

(返回顶部) (第 441 页)

无

索引

符号

-c 开关 16
-contention 200
-iqnotemp 287

A

AES 加密算法
 CREATE DATABASE 语句 385
ALLOW_SNAPSHOT_VERSIONING 189
ALTER DBSPACE 语句
 ADD 参数 89
 SIZE 参数 91
 示例 89
 语法 356
ALTER INDEX 语句
 错误 360
ALTER LOGICAL SERVER 语句
 语法 363
ALTER LS POLICY 语句
 语法 364
ALTER TABLE 语句
 外键 167
 语法 368
ALTER VIEW 语句
 RECOMPILE 368
ALTER 语句
 自动提交 175
AppInfo 连接参数 306
AStart 连接参数 308
AStop 连接参数 308
AUDITING 选项 262
AUTOINCREMENT
 负数 161
 缺省值 161
AUTOINCREMENT 列缺省值 410
AutoPreCommit 连接参数 307
AutoStart 连接参数 308
AutoStop 连接参数 308
AVG 函数 142
安全
 集成登录 61, 62
 审计 262
 最小口令长度 353

安全性

 Encryption [ENC] 连接参数 319

B

BCAST 通信参数
 说明 333
BEGIN TRANSACTION 语句
 远程数据访问 207
BIT 数据
 允许的索引 148
BLISTENER 通信参数
 说明 334
BLOCKING 选项 181
BLOCKING_TIMEOUT 选项 182
Broadcast 通信参数
 说明 333
BroadcastListener 通信参数
 说明 334
版本控制 175, 189, 193
 隔离级别 179
 临时表 199
 性能影响 192
 游标 205
 在系统恢复中 204
绑定内存
 设置 iqwmem 开关 14
包含索引
 请参见 WD 索引
保存点
 在事务中 202
备份
 并发 205
本地临时表
 关于 120
本地意图写 198
比较索引
 请参见 CMP 索引
标记化 FP (NBit) 138
标量值子查询 117
表
 GLOBAL TEMPORARY 410
 创建 119, 410
 访问权限已禁止 281
 更改 123, 368

- 更改定义 368
- 管理已禁止的访问权限 282
- 将键添加到 124, 125
- 临时 410
- 删除 124, 425
- 锁定 189, 434
- 只读 100
- 表分区
 - DDL 操作 126
 - DML 操作 126
 - hash-partition-key 128
 - hash-partitioning-scheme 128
 - 定义 125
 - 范围分区 127
 - 分区键 126
 - 散列范围分区 129
 - 散列分区 128
- 表级快照版本控制 189
- 表锁
 - 查看 195
- 表锁争用
 - 管理 200
- 表约束 410
- 并发
 - 备份 205
 - 插入、删除和查询 190
 - 数据定义 190
 - 锁定表 434
- 并行 CREATE INDEX 151
- 布线
 - 故障排除 286

C

- CACHE_PARTITIONS 选项 11
- CALL 语句
 - 示例 210
- Catalog 文件
 - 增长 275
- CBSize 连接参数 310
- CBSpace 连接参数 311
- CharSet 连接参数 309
- CHECK ON COMMIT 子句
 - 参照完整性 410
- CHECK 条件
 - 表 164
 - 关于 410
 - 列 163
 - 删除 165

- 修改 165
- 用户定义数据类型 164
- CHECKPOINT 语句
 - 语法 383
- ClientPort 通信参数 334
- CMP 索引 134
 - 建议用法 134
 - 限制 134
- CommBufferSize 连接参数 310
- CommBufferSpace 连接参数 311
- COMMIT TRANSACTION 语句
 - Transact-SQL 384
- COMMIT 语句
 - 语法 384
 - 远程数据访问 207
- CommLinks 连接参数 312
- ConnectionName 连接参数 313
- ConnectionPool 连接参数 314
- COUNT DISTINCT
 - 对索引选择的影响 145
- COUNT 函数 142
- CREATE DATABASE 语句 71
 - IQ RESERVE 参数 89
 - IQ SIZE 参数 89
 - TEMPORARY RESERVE 参数 89
 - TEMPORARY SIZE 参数 89
 - 语法 385
 - 原始设备 70
- CREATE DBSPACE 语句 98
 - RESERVE 参数 89
 - SIZE 参数 89
- CREATE DOMAIN 语句
 - 语法 395
- CREATE FUNCTION 语句
 - 语法 217
- CREATE INDEX 语句
 - 表使用 397
 - 语法 397
- CREATE LOGICAL SERVER 语句 404
- CREATE LS POLICY 语句
 - 语法 406
- CREATE TABLE 语句
 - 示例 119
 - 语法 410
- CREATE TEXT INDEX 143
- CREATE 语句
 - 并发规则 190
 - 自动提交 175

CS 连接参数 309**参照完整性**

对现有未强制实施的外键强制实施 169

列缺省值 157

强制 166

声明 167

插入

列缺省值 158

查询

并发规则 190

抖动 280

范围谓词 136

使用 DATEPART 135

限制并发 13

性能 140

性能问题 280

查询计划

索引建议 152

查询类型

索引类型 145

产品支持 299**常量表达式缺省值 162****超线程**

服务器开关 17

持有游标 206**创建**

TEXT 索引 143

列缺省值 158

数据类型 395

词索引 132**磁盘**

监控空间使用情况 276

空间不足 268, 273

磁盘空间

保存 206

分配 98

索引 147

磁盘空间不足

监控空间使用情况 276

建议操作 268

建议的操作 273

存储过程

显示相关信息 211

错误

dbspace 不足情况 273

报告 299

错误 190

事务处理 190

D

DatabaseFile 连接参数 314

DatabaseName 连接参数 315

DatabaseName 通信参数

说明 335

DatabaseSwitches 连接参数 316

DataSourceName 连接参数 317

DATE 索引 134

建议用法 134

其它索引 138

缺点 137

限制 137

优点 137

与其它索引的比较 138

DATEPART

查询 135

DBF 连接参数 314

嵌入式数据库 43

dbisql

登录窗口 42

端口号 42

故障排除 272

连接 42

连接参数示例 40

命令行参数 42

DBKEY 连接参数 317

DBN 连接参数 315

DBN 通信参数

说明 335

dbo 用户 ID

dbo 用户 ID 拥有的视图 117, 425

DBS 连接参数 316

dbspace

dbspace 不足状况 76, 273

IQ_SHARED_TEMP 79, 82

IQ_SYSTEM_MSG 83

IQ_SYSTEM_TEMP 79

SYSTEM 79

创建 98

创建时的文件位置 70

创建示例 89

定义 79

更改大小 89

管理示例 89

监控空间使用情况 276

空间不足错误消息 273

命名 97

删除 99, 425

- 设置脱机 356
 - 脱机 79
 - 文件位置 70
 - 显示使用信息 89
 - 显示索引信息 89
 - 在原始设备上添加 70
 - 只读 100
 - DbSPACE
 - 自动创建 70
 - DBSpaceLogger 事件 276
 - dbtran 实用程序 259
 - 退出代码 259
 - 语法 259
 - DDL 199
 - 关于 115
 - DDL 锁定 190
 - DEDICATED_TASK 选项
 - 描述 352
 - DisableMultiRowFetch 连接参数 318
 - DMRF 连接参数 318
 - DOBROADCAST 通信参数 336
 - DQP
 - 性能 364, 406
 - DROP DATATYPE 语句
 - 语法 425
 - DROP DBSPACE 语句
 - 语法 425
 - DROP DOMAIN 语句
 - 语法 425
 - DROP EVENT
 - 语法 425
 - DROP FUNCTION 语句
 - 语法 425
 - DROP INDEX 语句
 - 语法 425
 - DROP LOGICAL SERVER 语句 428
 - DROP MESSAGE
 - 语法 425
 - DROP PROCEDURE 语句
 - 语法 425
 - DROP TABLE
 - IDENTITY_INSERT 选项 425
 - DROP TABLE 语句
 - 示例 124
 - 语法 425
 - DROP VIEW 语句
 - 示例 118
 - 限制 117, 425
 - 语法 425
 - DROP 语句
 - 并发规则 190
 - 语法 425
 - 自动提交 175
 - DSN 连接参数 317
 - DTTM 索引 134
 - 建议用法 134
 - 其它索引 138
 - 缺点 137
 - 优点 137
 - 与其它索引的比较 138
 - DumpAllThreads 文件 268
 - 当前日期和时间缺省值 160
 - 登录
 - 集成 58, 60
 - 另请参见 连接
 - 登录重定向 364
 - 调度 251
 - 抖动
 - HASH_THRASHING_PERCENT 选项 280
 - 端口号
 - 数据库服务器 346
 - 断开连接 67
 - 堆栈跟踪
 - 位置 270
 - 针对线程生成 268
 - 多处理器计算机
 - 开关 9
 - 多列索引 397
- ## E
- ENC 连接参数
 - 说明 319
 - EncryptedPassword 连接参数 319
 - Encryption 连接参数
 - 说明 319
 - ENG 连接参数 318
 - EngineName 连接参数 318
 - ENP 连接参数 319
 - Escape 连接参数 320
- ## F
- FileDataSourceName 连接参数 321
 - FileDSN
 - 创建 55
 - 分发 55

- Flat FP 138
 - 反斜线
 - Windows 原始设备 73
 - 范围分区
 - partition-key 127
 - range-partition-decl 127
 - range-partitioning-scheme 127
 - VALUES 子句 127
 - 范围分区 127
 - 示例 127
 - 限制 127
 - 防火墙
 - BroadcastListener [BLISTENER] 通信参数 334
 - LDAP 通信参数 339
 - 连接 334
 - 非空值约束 156
 - 分布式查询处理
 - 性能 364, 406
 - 分隔符
 - 示例 397
 - 在 WD 索引中 397
 - 分区
 - DDL 操作 126
 - DML 操作 126
 - hash-partition-key 128
 - hash-partitioning-scheme 128
 - hash-range-partitioning-scheme 129
 - partition-key 127
 - range-partitioning-scheme 127
 - SUBPARTITION BY RANGE 129
 - VALUES 子句 (范围分区) 127
 - 表 125
 - 范围分区 127
 - 分区键 126
 - 散列范围分区 129
 - 散列分区 128
 - 删除 368
 - 示例 127-129
 - 限制 127, 128
 - 只读 100
 - 分区限制 11
 - 分析输出 297
 - 服务器
 - CPU 使用 268
 - 不响应 268
 - 创建逻辑 404
 - 关闭并重新启动 32
 - 关闭问题 270
 - 恢复 263
 - 空间不足 268
 - 连接 55
 - 命名 8
 - 命名冲突 265
 - 删除逻辑 428
 - 事务日志 264
 - 死锁 268
 - 停止 32
 - 停止处理 268
 - 唯一端口号 266
 - 唯一名称 265
 - 无响应 268
 - 另请参见 数据库服务器
 - 服务器日志文件 29
 - 符号链接 385
- ## G
- getinfo 脚本 299
 - gm 开关 12
 - 对恢复的影响 204
 - GROUP BY 子句
 - 对索引选择的影响 145
 - gt 开关 15
 - 高度加密
 - CREATE DATABASE 语句 385
 - 高速缓存 dbspace 106
 - 创建 107
 - 监控 110
 - 将 dbfile 更改为读写 110
 - 将 dbfile 更改为只读 110
 - 禁用 109
 - 删除 108
 - 删除存储从 109
 - 添加更多存储到 108
 - 重新启用 109
 - 高速缓存大小
 - 目录存储库的设置 16
 - 高速缓存大小
 - 设置目录存储库 16
 - 隔离级别 179
 - 根逻辑服务器策略 364
 - 跟踪
 - 针对线程生成 268
 - 共享内存
 - 信号 279
 - 故障排除 263
 - dbisql 272

索引

- 布线问题 286
- 常见问题 286
- 处理问题 279
- 服务器操作 264
- 数据库连接 271
- 协议 284
- 资源问题 273
- 关闭
 - 故障排除 270
 - 数据库 37
- 关闭并重新启动服务器 32
- 管理
 - 事务 207
- 滚动
 - 游标 206
- 过程
 - 参数 211
 - 调用 210
 - 返回结果 210, 232
 - 关于 209
 - 结构 229
 - 结果集 211
 - 删除 210, 425
 - 所有者 211
 - 游标 212
 - 允许的 SQL 语句 229
 - 执行特权 210

H

HASH_THRASHING_PERCENT 选项 280

HG 索引

- NULL 值 140, 397
- 查询性能 140
- 建议用法 139
- 具有 NULL 的多列 140
- 具有 NULL 值的多列 397
- 其它索引 140
- 缺点 140
- 外键约束 139
- 优点 140
- 与其它索引的比较 140
- 在删除外键约束时保留 153
- 自动创建 140

High_Group 索引

请参见 HG 索引

High_Non_Group 索引

请参见 HNG 索引

HNG 索引 141

建议用法 141

其它索引 142

缺点 141

优点 141

与其它索引的比较 141

HOST 通信参数 337

函数

创建 217

删除 425

用户定义的 216

环境变量

SQLCONNECT 49

缓冲区不足

缓冲区高速缓存 279

缓冲区高速缓存

IQ UTILITIES 命令 268

分区 11

监控器 268

空间不足 268, 279

缓冲区空间

连接参数 311

换行

WD 索引分隔符 397

恢复

服务器 263

事务 204

事务日志 204

系统 204

系统故障 270

回退 175

dbspace 不足状况 76, 273

I

IANA

端口号 346

IDENTITY 列

以及 DROP TABLE 425

IDENTITY_INSERT 选项

删除表 425

IDLE 连接参数 321

INSERT 语句 151

和完整性 156

INT 连接参数 322

INTEGRATED LOGIN 语句

授予 429

IP 地址 306

Ping 285

IP 通信参数 337

IPv6 地址 344

- IPX
 - 服务器配置 332
- IQ PAGE SIZE 74
- IQ UNIQUE
 - 更改值 123
- IQ UNIQUE 表选项 122
- IQ UTILITIES
 - 缓冲区高速缓存监控器 268
- IQ UTILITIES 语句
 - 语法 430
- IQ 存储库
 - 原始设备访问 86
 - 原始设备权限 87
- IQ 唯一约束 163
- IQ_SYSTEM_MAIN
 - 大小准则 84
- IQ_SYSTEM_MSG dbspace 83
- IQ_SYSTEM_TEMP 287
- iqgovern 开关 13
- iqmc 开关 9
- iqmt 开关 14
- iqnumbercpus
 - 服务器开关 17
- iqpartition 开关 9
- iqpartition 启动开关 11
- iqwmem 开关 14

J

- Java
 - 内存要求 16
 - 在 SAP Sybase IQ 中使用 49
- jConnect 驱动程序
 - 连接到数据库 55
- JDBC 连接
 - 关于 55
- 基表 120
- 集成登录
 - 操作系统 58
 - 缺省用户 63
 - 使用 61
 - 网络方面 62
- 集合 142
- 记录连接 68
- 技术支持
 - MySybase 303
 - 将问题报告到 299
 - 联机帮助 303
 - 清单 303

- 技术支持的清单 303
- 加密
 - Encryption [ENC] 连接参数 319
 - 强 319
 - 通信 347
- 加密算法
 - CREATE DATABASE 语句 385
- 监控
 - 启动和停止 430
 - 在 IQ UTILITIES 语句中 430
- 监控器输出选项
 - contention 200
- 检查点
 - 调整间隔 283
 - 关于 202
 - 在系统恢复中 204
 - 自动和显式 202
- 将 dbspace 设置为联机 356
- 接口库
 - 连接 50
- 结果集
 - 过程 211
- 局部临时表 199

K

- 客户端
 - Encryption [ENC] 连接参数 319
- 客户端进程信息 306
- 空间不足 275
 - 缓冲区高速缓存 279
- 空间管理
 - dbspace 不足状况 76, 273
 - IQ 临时存储库 76
 - IQ 主存储库 76
 - 等待空间状况 76, 273
- 空值
 - 缺省值 161
- 口令
 - 最小长度 353
- 口令安全性 72
- 块大小 76
- 快速投影索引 138
- 快照版本控制 175
 - 表级 189
 - 行级 193, 194

L

- LANalyzer 286
- LANG 连接参数
 - 说明 323
- LazyClose 连接参数
 - 说明 323
- LCLOSE 连接参数
 - 说明 323
- LDAP 服务器
 - LDAP 通信参数 339
- LDAP 通信参数
 - 说明 339
- LF 索引 142
 - 超过最大唯一值数 281
 - 建议用法 142
 - 其它索引 142
 - 缺点 142
 - 优点 142
 - 与其它索引的比较 142
- LF 通信参数
 - 说明 340
- Links 连接参数 312
- LivenessTimeout 连接参数 324
- LOAD TABLE 语句 151
- LOCAL 通信参数
 - 说明 339
- LocalOnly 通信参数
 - 说明 339
- LOCK TABLE
 - 语法 434
- LOG 通信参数
 - 说明 340
- LOG_CONNECT 数据库选项 352
- LogFile
 - 连接参数 324
 - 通信参数 340
- LogFormat 通信参数
 - 说明 340
- LogicalServer
 - 连接参数 325
- LOGIN_MODE 数据库选项
 - 集成登录 60
- LogOptions 通信参数
 - 说明 341
- LOPT 通信参数
 - 说明 341
- Low_Fast 索引
 - 请参见 LF 索引

- LS 策略 367, 409
- LTO 连接参数 324
- 连接
 - BroadcastListener [BLISTENER] 通信参数 334
 - DEDICATED_TASK 选项 352
 - iAnywhere JDBC 驱动程序 55
 - Interactive SQL 40
 - jConnect 55
 - JDBC 49
 - JDBC SAP Sybase Open Client 49
 - LDAP 通信参数 339
 - 本地数据库 40
 - 防火墙 334
 - 概述 50
 - 记录 68, 352
 - 建立 51
 - 临时 65
 - 嵌入式数据库 43
 - 使用数据源 47
 - 示例 40
 - 限制并发 12
 - 性能 154
 - 性能影响 154
 - 学习路线图 39
 - 远程 55, 207
- 连接参数 325
 - CharSet 309
 - ConnectionPool 314
 - Encryption [ENC] 319
 - Escape 320
 - IDLE 321
 - LANG 323
 - LazyClose [LCLOSE] 323
 - 不区分大小写 305
 - 关于 305
 - 缺省 48
 - 优先级 305
 - 在连接字符串中 51
- 连接归集 63
- 连接句柄 294
- 连接信息
 - IQ 消息文件 302
 - srvlog 文件 302
 - 请求日志 302
- 连接字符串
 - 表示 51
- 联机
 - dbspace 356

链接

符号 385

列

更改 123, 368

缺省值 157

删除 123

添加 123

约束 410

临时 dbfile 287

临时表 410

版本控制 199

创建 410

关于 120

装载 120

临时存储

空间管理 76

用于节省空间的选项 206

路径

相对 385

路径名

数据库 72

路由器

广播 336

逻辑服务器 66, 67

更改 363

在 Simplex 中 66

逻辑服务器策略

创建 406

定义 404

更改 364

选项 367, 409

M

MAX_TEMP_SPACE_PER_CONNECTION 选项
76

MAXCONN 通信参数
说明 342

MaxConnections 通信参数
说明 342

MDSR 加密算法
CREATE DATABASE 语句 385

Microsoft Access 279

MIN_PASSWORD_LENGTH 选项 353

mklink 实用程序 88

Multiplex 服务器
平衡负载 349

Multiplex 数据库
创建 385

MYIP 通信参数 343

MySybase

访问 303

在线支持 303

敏感性

游标行为 206

命令行开关 5

必需 7

显示 5

命名冲突 265

目录存储库

关于 81

设置高速缓存 16

设置高速缓存大小 16

校验索引 173

预分配空间 81

N

NBit 138

NDIS

驱动程序 284

net.cfg 文件 286

NetBIOS

服务器配置 332

NetWare

网络适配器设置 286

NEWID

缺省值 161

NodeType

连接参数 327

Novell 客户端软件 284

NULL

在多列 HG 索引上 140, 397

NULL 值

在多列 HG 索引中 140, 397

内存

创建绑定内存池 14

用于目录存储库高速缓存 16

内存消息

装载通知消息 290

O

ODBC

连接参数 305

ODBC 连接

关于 55

索引

ODBC 数据源

与 jConnect 一起使用 55

ODI 驱动程序 284

P

Password 连接参数 328

Ping

TCP/IP 285

PORT 通信参数 346

PreFetchOnOpen 通信参数

说明 344

PREPARE 语句

远程数据访问 207

PWD 连接参数 328

配置参数

覆盖 267

配置文件

使用 6

批处理

关于 209

Q

QUERY_TEMP_SPACE_LIMIT 选项 76

启动

故障排除提示 264

启动参数 5

嵌入式数据库 43

启动脚本 4

启动实用程序 4

嵌入式数据库 305, 309

Java 43

连接 43

启动 43

强加密

Encryption [ENC] 连接参数 319

请求级别的记录 265

请求记录级别 302

请求日志文件 297

连接信息 302

使用 sa_get_request_profile 297

使用 sa_get_request_times 297

区分大小写

连接参数 305

命令行 5

驱动程序

jConnect JDBC 驱动程序 55

JDBC 49

SQL Anywhere JDBC 驱动程序 55

全局临时表 199

关于 120

权限

视图 118

原始设备 70

缺省

连接参数 48

缺省配置文件 6

缺省索引

Flat FP 138

NBit 138

标记化 FP (NBit) 138

缺省值

AUTOINCREMENT 161

NEWID 161

USER 特殊值 160

插入 158

常量表达式 162

创建 158

当前日期和时间 160

空值 161

列 157

装载 158

字符串和数字 162

R

rawaccredit 实用程序 86

rawaccess 实用程序 87

RCVBUFSZ 通信参数

说明 345

ReceiveBufferSize 通信参数

说明 345

REFERENCES 子句 368

RELEASE SAVEPOINT 语句 202

RESTRICT 操作 410

Rigndael 加密算法

CREATE DATABASE 语句 385

RLV dbspace

文件, 删除 96

文件, 添加 96

ROLLBACK TO SAVEPOINT 语句 202

ROLLBACK 语句 202

语法 436

日期时间索引

请参见 DTTM 索引

日志文件

服务器 29

将连接信息相关联 302

S

- sa_conn_info 195
- sa_get_request_profile
 - 分析请求日志文件 297
- sa_get_request_times
 - 分析请求日志文件 297
- sa_report_deadlocks 系统过程 184
- SAP Sybase IQ
 - 停止 32
- SAVEPOINT 语句
 - 和事务 202
 - 语法 437
- schema
 - 创建 119
 - 更改 119
- SELECT * 368
- SELECT DISTINCT 投影 145
- SELECT 语句
 - 视图创建的限制 117
- SELECT 语句限制 117
- SendBufferSize 通信参数
 - 说明 345
- ServerName 连接参数 318
- ServerPort 通信参数 346
- Service Manager
 - 启动服务器 5
- SESSIONS 通信参数 347
- SET OPTION 语句
 - 语法 438
- SET TEMPORARY OPTION 语句
 - 语法 438
- SNAPSHOT_VERSIONING 194
- SNDBUFSZ 通信参数
 - 说明 345
- sp_iqcheckoptions 存储过程 76
- sp_iqconnection 195
- sp_iqdbspace
 - dbspace 使用信息 89
- sp_iqdbspaceinfo
 - dbspace 使用信息 89
- sp_iquestdbspaces
 - 估计 dbspace 要求 89
- sp_iquestspace
 - 估计数据库空间要求 89
- sp_iqindex 存储过程 152
- sp_iqindexinfo
 - 显示索引信息 89
- sp_iqllocks 195
- sp_iqprocedure
 - 有关过程的信息 211
- sp_iqprocparm
 - 过程参数 211
- sp_iqstatus
 - 示例输出 287
 - 在故障排除中使用 287
- sp_iqtransaction 199
 - 确定禁止写入者 281
- sp_iqtransaction 系统过程 176
- SQL Anywhere JDBC 驱动程序
 - 连接到 SAP Sybase IQ 和 SQL Anywhere 数据库 55
- SQLCONNECT 环境变量
 - 连接 49
- srvlog
 - 将连接信息相关联 302
- srvlog 文件
 - 连接信息 302
- SSD 106
- start_iq
 - 参数 267
 - 故障排除 267
 - 命令无法运行 267
- start_iq 实用程序 4
- StartLine 连接参数 330
- STOP DATABASE 语句
 - 语法 440
- STOP ENGINE 语句
 - 语法 441
- SUM 函数 142
- Sybase Control Center
 - 创建 dbspace 98
- SYSCOLUMN 表
 - 完整性 172
- SYSPFOREIGNKEY 表
 - 完整性 172
- SYSTABLE 表
 - 完整性 172
- SYSTEM dbspace 79
- SYSVIEW 视图
 - 视图信息 119
- 散列范围分区
 - hash-range-partitioning-scheme 129
 - SUBPARTITION BY RANGE 129
 - 示例 129
- 散列分区
 - hash-partition-key 128

- hash-partitioning-scheme 128
- 示例 128
- 限制 128
- 删除
 - 列缺省值 158
 - 视图 117, 425
- 删除分区 368
- 设备类型
 - 数据库 2
- 实例化视图
 - 删除 425
- 实体完整性
 - 强制 166
- 实用程序语句 430
- 使用列缺省值 157
- 事件
 - DBSpaceLogger 276
 - 监控磁盘空间使用情况 276
 - 监控空间使用情况 276
 - 检索日程表名称 258
 - 检索事件名称 258
 - 删除 425
- 事务 175
 - ROLLBACK 语句 436
 - SAVEPOINT 语句 437
 - 保存点 202
 - 关于 175
 - 管理 207
 - 回退 203
 - 启动 175
 - 提交 384
 - 游标 205
 - 远程数据访问 207
 - 在恢复中 204
 - 子事务和保存点 202
 - 阻塞 183
- 事务处理
 - 关于 175
- 事务管理 207, 384
 - 在 Transact-SQL 中 384
- 事务日志
 - 在系统恢复中 204
- 视图 117
 - 插入和删除 117
 - 创建 116
 - 权限 118
 - 删除 117, 118, 425
 - 使用 115, 117
 - 视图中的变更表 368
- 索引 397
- 修改 118
- 与永久表的差异 115
- 输出选项, 监控器
 - contention 200
- 数据
 - 无效 155
 - 重复 155
- 数据定义
 - 创建 119
- 数据定义语言
 - 并发规则 190
 - 关于 115
- 数据库
 - schema 创建 119
 - SQL Anywhere 81
 - 初始大小 84
 - 初始化 69
 - 创建 69, 385
 - 创建对象 119
 - 创建数据库所需的特权 1
 - 大小 74
 - 块大小 76
 - 连接到 50, 55
 - 命名 8
 - 命名冲突 265
 - 启动 35
 - 启动权限 36
 - 启动时命名 8
 - 嵌入式 305, 309
 - 缺省特性 71
 - 删除 77
 - 设计 115
 - 设置概述 1
 - 使用对象 115
 - 特权 19
 - 停止 37, 440
 - 文件位置 70
 - 相对路径名 72
 - 卸载 35
 - 修复 263
 - 选择位置 73
 - 页大小 74
 - 移动 70
 - 预分配空间 79
- 数据库服务器
 - 禁止启动 308
 - 连接 55

- 名称开关 8
- 命令行开关 5
- 启动 3
- 启动时命名 8
- 停止 32, 35, 331, 441
- 远程 55
- 在命令提示符下启动 4
- 作为 Windows 服务 5
- 数据库空间
 - 更改 356
- 数据库实用程序 49
 - 日志转换 259
- 数据库文件
 - 更改 356
- 数据库选项
 - DEDICATED_TASK 352
 - 持续时间 351
 - 游标 351
 - 最大字符串长度 438
- 数据类型
 - 创建 395
 - 删除用户定义数据类型 425
 - 在创建表时指定 119
- 数据链接层
 - 故障排除 285
- 数据完整性
 - 概述 155
 - 列缺省值 157
 - 系统表中的规则 172
 - 约束 157
- 数据源
 - 连接 47
 - 与 jConnect 一起使用 55
- 死锁 183
 - 报告 184
 - 检测 268
 - 解决 268
- 索引
 - Flat FP 138
 - NBit 138
 - TEXT 索引 143
 - 标记化 FP (NBit) 138
 - 表使用 397
 - 表中的数目太多 279
 - 并行创建 151
 - 创建 119, 150, 209, 397
 - 磁盘空间使用情况 147
 - 多列 397
 - 多列 HG 和 NULL 140, 397

- 关于 131
- 建议组合 150
- 快速投影 138
- 列表 152
- 命名 397
- 缺省 138
- 删除 153, 425
- 所有者 397
- 唯一 397
- 系统视图中 153
- 显式有关信息 152
- 信息日志中的建议 152
- 选择索引类型 147
- 在装载表后添加 151
- 重建 123
- 重命名 152
- 自动创建 122
- 最大唯一值数 281
- 索引顾问 152
- 索引类型
 - LF 142
 - 建议 131, 146
 - 选择 147
 - 选择标准 144
- 锁
 - 管理争用 282
 - 使用 ROLLBACK 释放 436
- 锁定
 - DDL 操作 190
 - 表 189, 434
- 锁争用
 - iqpartition 服务器选项 11

T

- TCP/IP
 - BroadcastListener [BLISTENER] 通信参数 334
 - LDAP 通信参数 339
 - 测试 285
 - 服务器端口号 346
 - 服务器配置 332
 - 故障排除 285
 - 通过防火墙连接 334
- TDS 通信参数 347
- Telnet
 - TCP/IP 测试 285
- TEMP_DATA_IN_SHARED_TEMP
 - 逻辑服务器策略选项 364

索引

- TEXT 索引 143
 - 创建 143
- TIME 索引 143
 - 建议使用 143
 - 其它索引 138
 - 缺点 144
 - 限制 144
 - 优点 144
 - 与其它索引的比较 138
- TIMEOUT 通信参数 348
- TLV 189
- TO 通信参数 348
- Transact-SQL
 - COMMIT TRANSACTION 384
- 特权
 - 定义数据库对象 1
 - 过程 210
 - 命令行开关 19
 - 用户定义的函数 223
- 添加
 - TEXT 索引 143
- 停止
 - 服务器 32
- 停止数据库 440
- 通信
 - Encryption [ENC] 连接参数 319
 - 参数 332
 - 故障排除 284
- 通信参数
 - Broadcast [BCAST] 333
 - BroadcastListener 334
 - ClientPort 334
 - DatabaseName [DBN] 335
 - DOBROADCAST 336
 - HOST 337
 - LDAP [LDAP] 339
 - LocalOnly [LOCAL] 339
 - LogOptions [LOPT] 341
 - MaxConnections 342
 - MYIP 343
 - PreFetchOnOpen 344
 - ReceiveBufferSize [RCVBUFSZ] 345
 - SendBufferSize [SNDBUFSZ] 345
 - ServerPort 346
 - SESSIONS 347
 - TDS 347
 - TIMEOUT 348, 349
 - 关于 305
- 通知计数 151

- 通知消息 290
- 投影
 - 优化 138
- 脱机
 - dbspace 356

U

- UID 连接参数 331
- UNC 连接参数 331
- Unconditional 连接参数 331
- UNIQUE 约束 162
- USER 特殊值
 - 缺省值 160
- Userid 连接参数 331

V

- VERIFY 通信参数 349
- VERIFY_PASSWORD_FUNCTION 选项 353

W

- WarehouseArchitect
 - 关于 115
- WD 索引 132
 - CHAR 列 397
 - 分隔符 397
 - 建议用法 132
 - 缺点 133
 - 优点 133
- WHERE 子句
 - 对索引选择的影响 145
- 外键
 - 可选 169
 - 完整性约束 410
 - 未命名 410
 - 现有未强制实施 169
- 外键约束
 - 删除 153
- 完整性
 - 概述 155
 - 列缺省值 157
 - 约束 156, 157
- 网络号
 - 在 IPX 地址中 336
- 网络适配器
 - 驱动程序 284

网络通信
 命令行开关 332

网络协议
 故障排除 284

唯一
 约束 410

唯一索引 397

未强制实施的外键 169

文件
 设置联机 356
 设置脱机 356
 数据库空间 356

文件大小
 控制 275

问题
 报告 299

物理层
 故障排除 286

X

系统表
 关于 125

系统不响应 268

系统故障
 从故障中恢复 270

系统过程
 sp_iqtransaction 176

系统视图
 完整性 172

线程
 不足 278
 生成堆栈跟踪 268
 增加 14

相对路径 385

消息
 dbspace 不足情况 273
 内存通知 290
 删除 425

消息日志
 IQ_SYSTEM_MSG dbspace 83
 索引建议 152

消息文件
 连接信息 302

协议
 故障排除 284
 开关 25

写入访问权限已禁止
 管理争用 282

确定禁止写入者 281

信号 279

行级快照版本控制 193

行锁 197

性能
 表级版本控制的影响 192
 查询和装载 280
 连接 154
 约束的影响 156

修复
 数据库 263

修改
 列缺省值 158

修改和删除列缺省值 158

选项
 DEDICATED_TASK 352
 MAX_TEMP_SPACE_PER_CONNECTION
 76
 QUERY_TEMP_SPACE_LIMIT 76
 持续时间 351
 范围 351
 设置 438
 优先级 351
 游标 351

选项值
 截断 438

选择驱动程序
 使用 iAnywhere JDBC 驱动程序 55
 使用 jConnect JDBC 驱动程序 55

Y

页大小 74
 开关 24
 目录 24

以太网 286
 意图写锁 198
 查看 195, 198

用户定义的函数
 创建 216
 调用 222
 使用 216
 执行特权 223

用户定义数据类型
 CHECK 条件 164
 CREATE DOMAIN 语句 395
 删除 425

游标
 持有 206

索引

- 定位操作 206
- 滚动 206
- 过程 212
- 敏感性 206
- 数据库选项 351
- 消息记录 207
- 在事务中 205
- 预分配空间
 - 目录存储库 81
- 域 395
- 元数据
 - 在目录存储库中 81
- 原始设备 2
 - Windows 中的符号链接 88
 - 查看 Windows 上的权限 87
 - 创建数据库 70
 - 命名 385
 - 权限 70
 - 添加 dbspace 70
 - 在 Windows 上命名 73
 - 在 Windows 上设置权限 87
 - 在 Windows 中设置访问 86
- 远程服务器
 - 事务管理 207
- 远程数据访问 360
- 约束
 - 性能的影响 156

Z

- zr 日志文件 297
- 诊断工具 287
 - sa_server_option 295
 - sp_iqcheckdb 294
 - sp_iqcheckoptions 294
 - sp_iqconnection 294
 - sp_iqcontext 294
 - sp_iqdbstatistics 294
 - sp_iqstatus 287
 - 记录服务器请求 295
 - 检查服务器启动选项 294
 - 检查数据库选项 294
 - 通信问题 299
- 帧类型 286

- 直连存储 106
- 只读
 - 锁定表 434
- 只读 dbspace 100
- 指定
 - 驱动程序 55
- 制表符
 - WD 索引分隔符 397
- 重定向 325
- 主存储库
 - 空间管理 76
- 主高速缓存
 - 删除 425
- 主键
 - AUTOINCREMENT 161
 - 创建 124, 125
 - 多列 166
- 装载
 - 可伸缩性 11
- 装载模式
 - 建议数据库大小 84
- 装载数据
 - 并发规则 190
 - 错误 281
 - 监控空间使用情况 276
 - 列缺省值 158
 - 通知消息 290
 - 性能 280
- 装载性能
 - iqpartition 服务器选项 11
 - 表锁争用 200
- 资源规划
 - iqnumbercpus 开关 17
- 子查询
 - 标量值 117
- 子事务
 - 和保存点 202
- 字符串
 - 数据库选项的长度 438
- 字符串和数字缺省值 162
- 字符集
 - 连接参数 309
- 阻塞 183