SinoDB 管理员参考

星瑞格 SinoDB 产品系列

SinoDB

版本 12.10

目录

插图清单..... 17

表格清单..... 30

简介..... 38
关于本出版物..... 38
用户类型..... 38
软件依赖性..... 38
演示数据库..... 38
示例代码约定..... 38
语法图..... 39
符合行业标准..... 40

第I部分 配置与监视 SinoDB®..... 41
数据库配置参数..... 41
onconfig 文件..... 41
更改 onconfig 文件..... 42
显示 onconfig 文件中的设置..... 43
onconfig 门户: 按功能类别划分的配置参数..... 43
ADMIN_MODE_USERS 配置参数..... 63
ADMIN_USER_MODE_WITH_DBSA 配置参数..... 64
ALARMPROGRAM 配置参数..... 65
ALLOW_NEWLINE 配置参数..... 66
ALRM_ALL_EVENTS 配置参数..... 66
AUTO_AIOVPS 配置参数..... 67
AUTO_CKPTS 配置参数..... 67
AUTO_LLOG 配置参数..... 68
AUTO_TUNE_SERVER_SIZE 配置参数..... 69
AUTO_LRU_TUNING 配置参数..... 70
AUTO_READAHEAD 配置参数..... 71
AUTO_REPREPARE 配置参数..... 72
AUTO_STAT_MODE 配置参数..... 73
AUTO_TUNE 配置参数..... 73
AUTOLOCATE 配置参数..... 75
BATCHEDREAD_INDEX 配置参数..... 76
BATCHEDREAD_TABLE 配置参数..... 76
BLOCKTIMEOUT 配置参数..... 77
BTSCANNER 配置参数..... 77
BUFFERPOOL 配置参数..... 78
CHECKALLDOMAINSFORUSER 配置参数..... 83
CKPTINTVL 配置参数..... 83
CLEANERS 配置参数..... 84
CLUSTER_TXN_SCOPE 配置参数..... 84
CONSOLE 配置参数..... 86
CONVERSION_GUARD 配置参数..... 86
DATASKIP 配置参数..... 87
DBCREATE_PERMISSION 配置参数..... 87
DB_LIBRARY_PATH 配置参数..... 88
DBSERVERALIASES 配置参数..... 88
DBSERVERNAME 配置参数..... 90

DBSPACETEMP 配置参数..... 91
使用散列联接溢出 (Hash Join Overflow) 和 DBSPACETEMP..... 92
DD_HASHMAX 配置参数..... 92
DD_HASHSIZE 配置参数..... 93
DEADLOCK_TIMEOUT 配置参数..... 93
DEF_TABLE_LOCKMODE 配置参数..... 94
DEFAULTESCCHAR 配置参数..... 94
DELAY_APPLY 配置参数..... 95
DIRECT_IO 配置参数 (UNIX™)..... 96
DIRECTIVES 配置参数..... 97
DISABLE_B162428_XA_FIX 配置参数..... 97
DRDA_COMMBUFFSIZE 配置参数..... 98
DRAUTO 配置参数..... 98
DRIDXAUTO 配置参数..... 99
DRINTERVAL 配置参数..... 100
DRLOSTFOUND 配置参数..... 101
DRTIMEOUT 配置参数..... 101
DS_HASHSIZE 配置参数..... 102
DS_MAX_QUERIES 配置参数..... 102
DS_MAX_SCANS 配置参数..... 103
DS_NONPDQ_QUERY_MEM 配置参数..... 104
DS_POOLSIZE 配置参数..... 105
DS_TOTAL_MEMORY 配置参数..... 106
DS_TOTAL_MEMORY 的算法..... 106
DUMPCNT 配置参数 (UNIX™)..... 107
DUMPCORE 配置参数 (UNIX™)..... 107
DUMPDIR 配置参数..... 108
DUMPGCORE 配置参数 (UNIX™)..... 108
DUMPSHMEM 配置参数 (UNIX™)..... 109
DYNAMIC_LOGS 配置参数..... 109
EILSEQ_COMPAT_MODE 配置参数..... 110
ENABLE_SNAPSHOT_COPY 配置参数..... 111
EXPLAIN_STAT 配置参数..... 111
EXT_DIRECTIVES 配置参数..... 112
EXTSHMADD 配置参数..... 112
FAILOVER_CALLBACK 配置参数..... 113
FAILOVER_TX_TIMEOUT 配置参数..... 113
FASTPOLL 配置参数..... 114
FILLFACTOR 配置参数..... 114
FULL_DISK_INIT 配置参数..... 115
HA_ALIAS 配置参数..... 115
HA_FOC_ORDER 配置参数..... 116
HDR_TXN_SCOPE 配置参数..... 118
IFX_EXTEND_ROLE 配置参数..... 119
IFX_FOLDVIEW 配置参数..... 119
IFX_XA_UNIQUEXID_IN_DATABASE 配置参数..... 120
INFORMIXCONRETRY 配置参数..... 120
INFORMIXCONTIME 配置参数..... 121
LIMITNUMSESSIONS 配置参数..... 121
LISTEN_TIMEOUT 配置参数..... 122
LOCKS 配置参数..... 123
LOGBUFF 配置参数..... 123
LOGFILES 配置参数..... 124
LOG_INDEX_BUILDS 配置参数..... 125
LOG_STAGING_DIR 配置参数..... 125
LOGSIZE 配置参数..... 126

LOW_MEMORY_MGR 配置参数..... 127
LOW_MEMORY_RESERVE 配置参数..... 128
LTXEHWM 配置参数..... 128
LTXHWM 配置参数..... 129
MAX_FILL_DATA_PAGES 配置参数..... 130
MAX_INCOMPLETE_CONNECTIONS 配置参数..... 130
MAX_PDQPRIORITY 配置参数..... 131
MIRROR 配置参数..... 132
MIRROROFFSET 配置参数..... 132
MIRRORPATH 配置参数..... 132
MSG_DATE 配置参数..... 133
MSGPATH 配置参数..... 134
MULTIPROCESSOR 配置参数..... 134
NET_IO_TIMEOUT_ALARM 配置参数..... 134
NETTYPE 配置参数..... 135
NS_CACHE 配置参数..... 137
NUMFDSERVERS 配置参数..... 138
OFF_RECVRY_THREADS 配置参数..... 139
ON_RECVRY_THREADS 配置参数..... 139
ONDBSPACEDOWN 配置参数..... 140
当不应用 ONDBSPACEDOWN 时的数据库服务器的行为..... 140
ONLIDX_MAXMEM 配置参数..... 140
OPTCOMPIND 配置参数..... 141
OPT_GOAL 配置参数..... 142
PC_HASHSIZE 配置参数..... 142
PC_POOLSIZE 配置参数..... 143
PHYSBUFF 配置参数..... 143
PHYSFILE 配置参数..... 144
PLOG_OVERFLOW_PATH 配置参数..... 144
PLCY_HASHSIZE 配置参数..... 145
PLCY_POOLSIZE configuration parameter..... 145
PN_STAGEBLOB_THRESHOLD 配置参数..... 145
PRELOAD_DLL_FILE 配置参数..... 146
QSTATS 配置参数..... 147
REMOTE_SERVER_CFG 配置参数..... 147
REMOTE_USERS_CFG 配置参数..... 148
RESIDENT 配置参数..... 148
RESTARTABLE_RESTORE 配置参数..... 149
RESTORE_POINT_DIR 配置参数..... 150
ROOTNAME 配置参数..... 151
ROOTOFFSET 配置参数..... 151
ROOTPATH 配置参数..... 151
ROOTSIZE 配置参数..... 152
RSS_FLOW_CONTROL 配置参数..... 152
RTO_SERVER_RESTART 配置参数..... 153
S6_USE_REMOTE_SERVER_CFG 配置参数..... 154
SB_CHECK_FOR_TEMP 配置参数..... 154
SBSPACENAME 配置参数..... 155
SBSPACETEMP 配置参数..... 156
SDS_ALTERNATE 配置参数..... 156
SDS_ENABLE 配置参数..... 157
SDS_FLOW_CONTROL 配置参数..... 158
SDS_LOGCHECK 配置参数..... 159
SDS_PAGING 配置参数..... 159
SDS_TEMPDBS 配置参数..... 160
SDS_TIMEOUT 配置参数..... 161

SECURITY_LOCALCONNECTION 配置参数..... 162
SEQ_CACHE_SIZE 配置参数..... 162
SERVERNUM 配置参数..... 162
SESSION_LIMIT_LOCKS 配置参数..... 163
SESSION_LIMIT_LOGSPACE 配置参数..... 163
SESSION_LIMIT_MEMORY 配置参数..... 164
SESSION_LIMIT_TEMPSPACE 配置参数..... 165
SESSION_LIMIT_TXN_TIME 配置参数..... 165
SHMADD 配置参数..... 166
SHMBASE 配置参数..... 167
SHMNOACCESS 配置参数..... 167
SHMTOTAL 配置参数..... 168
SHMVIRT_ALLOCSEG 配置参数..... 169
SHMVIRTSIZE 配置参数..... 169
SINGLE_CPU_VP 配置参数..... 171
VPCLASS 值和 SINGLE_CPU_VP 配置参数..... 171
SMX_COMPRESS 配置参数..... 172
SMX_NUMPIPES 配置参数..... 172
SMX_PING_INTERVAL 配置参数..... 172
SMX_PING_RETRY 配置参数..... 173
SP_AUTOEXPAND 配置参数..... 174
SP_THRESHOLD 配置参数..... 174
SP_WAITTIME 配置参数..... 175
SQL_LOGICAL_CHAR 配置参数..... 176
SQLTRACE 配置参数..... 177
STACKSIZE 配置参数..... 178
STATCHANGE 配置参数..... 179
STMT_CACHE 配置参数..... 179
STMT_CACHE_HITS 配置参数..... 180
STMT_CACHE_NOLIMIT 配置参数..... 181
STMT_CACHE_NUMPOOL 配置参数..... 181
STMT_CACHE_SIZE 配置参数..... 182
STOP_APPLY 配置参数..... 182
STORAGE_FULL_ALARM 配置参数..... 183
SYSALARMPROGRAM 配置参数..... 184
SYSSBSPACENAME 配置参数..... 184
TBLSPACE_STATS 配置参数..... 186
TBLTBLFIRST 配置参数..... 186
TBLTBLNEXT 配置参数..... 187
TEMPTAB_NOLOG 配置参数..... 187
TENANT_LIMIT_CONNECTIONS 配置参数..... 188
TENANT_LIMIT_MEMORY 配置参数..... 188
TENANT_LIMIT_SPACE 配置参数..... 189
TXTIMEOUT 配置参数..... 189
UNSECURE_ONSTAT 配置参数..... 190
UPDATABLE_SECONDARY 配置参数..... 190
USELASTCOMMITTED 配置参数..... 191
USEOSTIME 配置参数..... 192
USERMAPPING 配置参数 (UNIX™, Linux™)..... 193
USRC_HASHSIZE 配置参数..... 193
USRC_POOLSIZE 配置参数..... 194
USTLOW_SAMPLE 配置参数..... 194
VP_MEMORY_CACHE_KB 配置参数..... 195
VPCLASS 配置参数..... 196
WSTATS 配置参数..... 198
sysmaster 数据库..... 199

sysmaster 数据库.....199
buildsmi 脚本.....199
bldutil.sh 脚本.....200
系统监视接口.....200
了解 SMI 表.....200
访问 SMI 表.....200
系统监视接口表.....201
sysutils 表.....203
sysadtinfo.....204
sysaudit.....204
syschkio.....205
syscheckpoint.....205
syschunks.....206
sysckptinfo.....207
syscluster.....208
syscmsm.....209
syscmsmsla.....209
syscmsmtab.....210
syscmsmunit.....210
syscompdicts_full.....210
sysconfig.....211
sysdatabases.....212
sysdbslocale.....212
sysdbspaces.....213
sysdri.....214
sysdual.....214
sysenv.....214
sysenvses.....214
sysextents.....215
sysextspaces.....215
sysfeatures.....215
sysha_lagtime 表.....216
sysha_type.....217
sysha_workload.....217
sysipl.....218
syslocks.....218
syslogs.....219
syslogfil 表.....220
sysmgminfo.....221
sysnetclienttype.....221
sysnetglobal.....222
sysnetworkio 表.....222
sysonlinelog.....223
sysprofile.....223
sysproxyagents.....225
sysproxydistributors.....225
sysproxysessions 表.....225
sysproxytxnops 表.....226
sysproxytxns 表.....226
sysptprof 表.....226
sysrepevtreg 表.....227
sysrepstats 表.....227
sysrsslog.....230
sysscblst.....231
syssesappinfo.....231
syssesprof.....231

syssessions..... 232
syssmx..... 233
syssmxses..... 234
syssqexplain 表..... 234
syssqltrace..... 235
syssqltrace_hvar..... 237
syssqltrace_info..... 237
syssqltrace_iter..... 237
syssrcrss..... 238
syssrcsds..... 238
systabnames..... 239
systhreads..... 239
systrgrss..... 240
systrgsds..... 240
sysvpprof..... 240
SMI 表映射..... 241
从 onstat 获得的 SMI 表中的信息..... 242
sysadmin 数据库..... 243
调度程序表..... 243
ph_task 表..... 244
ph_run 表..... 246
ph_group 表..... 247
ph_alert 表..... 247
ph_threshold 表..... 249
结果表..... 250
command_history 表..... 250
storagepool 表..... 251
tenant 表..... 252
磁盘结构和存储..... 252
数据库空间结构和存储..... 253
根数据库空间的结构..... 253
保留页..... 253
常规数据库空间的结构..... 254
块可用列表页的结构..... 255
表空间 Tblspace 的结构..... 255
数据库表空间的结构..... 257
扩展数据块的结构和分配..... 258
数据库空间页的结构和存储..... 261
分段表的结构..... 263
B 型树索引页的结构..... 264
R 型树索引页的结构..... 268
简单大对象的存储..... 268
BLOB 空间的结构..... 268
数据库空间 BLOB 页的结构..... 268
简单大对象存储和描述符..... 268
BLOB 空间页类型..... 269
BLOB 空间 BLOB 页的结构..... 270
智能大对象空间结构..... 270
元数据区域的结构..... 271
智能大对象页结构..... 271
时间戳记..... 271
数据库和表的创建：磁盘上发生什么..... 272
数据库的创建..... 272
表的创建..... 272
解释逻辑日志记录..... 273
关于逻辑日志记录..... 273

删除表或索引的事务..... 273
回滚的事务..... 274
带有活动事务的检查点..... 274
分布式事务..... 274
逻辑日志记录结构..... 274
逻辑日志记录头..... 274
逻辑日志记录类型和附加列..... 275
智能大对象的日志记录类型..... 286

第II部分 管理实用程序..... 289
实用程序概述..... 289
获取实用程序版本信息..... 289
设置实用程序的本地环境变量..... 290
finderr 实用程序..... 290
genoncfg 实用程序..... 291
oncheck 实用程序..... 294
oncheck 检查并修复..... 294
每个选项做些什么？..... 294
使用 -y 选项执行修复..... 295
修复智能大对象空间和外部空间中的索引..... 296
锁定与 oncheck..... 296
oncheck 实用程序语法..... 296
oncheck -cc 和 -pc: 检查系统目录表..... 300
oncheck -cd 和 oncheck -cD 命令: 检查页..... 300
oncheck -ce, -pe: 检查空闲块列表..... 301
oncheck -ci 和 -cI: 检查索引节点链接..... 302
oncheck -cr 和 -cR: 检查保留页..... 303
oncheck -cs, -cS, -ps, -pS: 检查并显示智能大对象空间..... 303
oncheck -pB: 显示 BLOB 空间统计信息..... 304
oncheck -pd 和 pD: 以十六进制格式显示行..... 304
oncheck -pk, -pK, -pl, -pL: 显示索引信息..... 305
oncheck -pp and -pP: 显示逻辑页的内容..... 306
oncheck -pr 和 pR: 显示保留页信息..... 308
oncheck -pt 和 -pT: 显示表或分段的表空间..... 308
使用 -x 打开锁定..... 311
使用 -u 将特殊参数发送给存取方法..... 311
退出时的返回码..... 311
onclean 实用程序..... 312
onshutdown 脚本..... 313
oncmsm 实用程序..... 314
onconfig_diff 实用程序..... 316
ondblog 实用程序..... 317
oninit 实用程序..... 319
-FILE 选项..... 322
oninit 实用程序的返回码..... 323
onlog 实用程序..... 327
onmode 实用程序..... 330
onmode 命令语法..... 330
onmode -a: 添加共享内存段..... 331
onmode -BC: 允许大块方式..... 331
onmode -c: 强制检查点..... 332
onmode -C: 控制 B 型树扫描程序..... 332
onmode -cache surrogates: 高速缓存 allowed.surrogates 文件..... 333
onmode -d: 设置数据复制类型..... 334
onmode -d: 设置高可用性服务器的特性..... 335

onmode -d 命令: 使用数据复制来复制索引..... 336
onmode -D, -M, -Q, -S: 更改决策支持参数..... 337
onmode -e: 更改 SQL 语句高速缓存的使用..... 338
onmode -F: 释放未使用的内存段..... 339
onmode -I: 控制诊断收集..... 339
onmode -k, -m, -s, -u, -j：更改数据库服务器模式..... 339
使用 -k 选项使数据库服务器处于脱机模式..... 340
使用 -m 选项使数据库服务器处于联机模式..... 340
使用 -s 选项以正常关闭数据库服务器..... 340
使用 -u 选项立即关闭数据库服务器..... 341
使用 -j 选项使数据库服务器更改为管理模式..... 341
onmode -l: 切换逻辑日志文件..... 341
onmode -n, -r: 更改共享内存驻留..... 342
onmode -O: 重设 ONDBSPACEDOWN WAIT 方式..... 342
onmode -p: 添加或删除虚拟处理器..... 343
添加和删除虚拟处理器的规则..... 344
使用 onstat 实用程序监视轮询线程..... 345
onmode -P: 动态地启动、停止或重启监听线程..... 345
onmode -R: 重新生成 .infos.*dbservername* 文件..... 346
onmode -W: 更改 SQL 语句高速缓存的设置..... 346
SQL 语句高速缓存示例..... 347
onmode -we: 导出包含当前配置参数的文件..... 347
onmode -wf，-wm: 动态更改某些配置参数..... 347
onmode -wm: 更改 LRU 调整状态..... 348
onmode -wi: 导入配置参数文件..... 348
onmode -Y: 动态更改 SET EXPLAIN..... 349
onmode -z: 杀死数据库服务器会话..... 350
onmode -Z: 杀死分布式事务..... 350
onparams 实用程序..... 351
onparams 语法..... 351
onparams -a -d *dbspace*: 添加逻辑日志文件..... 352
onparams -d -l *lognum*: 删除逻辑日志文件..... 352
onparams -p: 更改物理日志参数..... 353
更改物理日志大小或位置之后进行备份..... 354
更改物理日志的大小和使用非缺省页大小..... 354
onparams -b: 添加缓冲池..... 354
onparams 命令的示例..... 354
ifxclone 实用程序..... 355
onspaces 实用程序..... 362
onspaces 语法..... 362
onspaces -a: 向数据库空间或 BLOB 空间添加块..... 362
onspaces -a: 向智能大对象空间添加块..... 364
onspaces -c -b: 创建 BLOB 空间..... 365
onspaces -c -d: 创建数据库空间..... 367
onspaces -c -P: 创建物理日志空间..... 369
onspaces -c -S: 创建智能大对象空间..... 371
使用 -t 选项创建临时智能大对象空间..... 373
使用 -Df 选项创建智能大对象空间..... 373
更改 -Df 设置..... 375
使用 onspaces -g 选项..... 375
onspaces -c -x: 创建外部空间..... 375
onspaces -ch: 更改智能大对象空间缺省规范..... 376
onspaces -cl: 清除智能大对象空间中的游离智能大对象..... 377
onspaces -d: 删除数据库空间、BLOB 空间或智能大对象空间中的块..... 377
onspaces -d: 删除空间..... 379
onspaces -f: 指定 DATASKIP 参数..... 379

onspaces -m: 启动镜像..... 380
以 -f 选项使用文件来指定块位置信息..... 381
onspaces -r: 停止镜像..... 382
onspaces -ren: 重命名数据库空间、BLOB 空间、智能大对象空间或外部空间..... 382
在 Enterprise Replication 处于活动时重命名数据库空间、BLOB 空间、智能大对象空间或外部空间..... 383
重命名空间之后执行归档..... 383
onspaces -s: 更改镜像块的状态..... 383
避免覆盖块..... 384
onstat 实用程序..... 385
onstat Portal: 按功能类别排序的 onstat 实用程序命令..... 385
监视数据库服务器状态..... 400
onstat 命令语法..... 401
onstat 命令: 等同于 onstat -pu 命令..... 403
onstat - 命令: 打印输出头..... 403
onstat -- 命令: 打印 onstat 选项和功能..... 404
在共享内存转储文件上运行 onstat 命令..... 404
onstat -a 命令: 打印数据库服务器的整体状态..... 405
onstat -b 命令: 显示正在使用的缓冲区的信息..... 405
onstat -B 命令: 显示已用缓冲区的信息..... 406
onstat -c 命令: 显示 ONCONFIG 文件内容..... 408
onstat -C 命令: 打印 B 型树扫描程序信息..... 408
onstat -d 命令: 显示块信息..... 414
onstat -D 命令: 打印页读取和页写入的信息..... 419
onstat -f 命令: 打印受 dataskip 影响的数据库空间信息..... 419
onstat -F 命令: 显示计数..... 420
onstat -g 监视选项..... 421
onstat -g act 命令: 显示活动线程..... 421
onstat -g afr 命令: 打印已分配内存分段..... 421
onstat -g all 命令: 打印诊断信息..... 422
onstat -g aqt 命令: 打印数据市集和加速查询表的信息..... 422
onstat -g arc 命令: 打印归档状态..... 425
onstat -g ath 命令: 显示所有线程的信息..... 426
onstat -g bth 和 -g BTH: 显示被阻挡和等待中的线程..... 427
onstat -g buf 命令: 显示缓冲池的概要文件信息..... 429
onstat -g cac 命令: 显示有关高速缓存的信息..... 432
onstat -g ckp 命令: 打印检查点历史记录与配置建议..... 434
onstat -g cfg 命令: 显示配置参数的当前值..... 438
onstat -g cluster 命令: 打印高可用性集群信息..... 440
onstat -g cmsm command: 显示连接管理器信息..... 443
onstat -g con 命令: 打印条件和线程信息..... 446
onstat -g cpu: 显示运行时统计信息..... 447
onstat -g dbc 命令: 打印 dbScheduler 和 dbWorker 线程统计信息..... 448
onstat -g defragment 命令: 显示分区扩展数据块取消分段信息..... 450
onstat -g dic 命令: 打印表信息..... 451
onstat -g dis 命令: 打印数据库服务器信息..... 451
onstat -g dll 命令: 打印动态链接库文件列表..... 452
onstat -g dmp 命令: 打印原内存..... 453
onstat -g dri 命令: 显示高可用性数据复制信息..... 454
onstat -g dsc 命令: 显示分布式高速缓存信息..... 457
onstat -g dsk 命令: 显示当前正在运行的压缩操作的进度..... 458
onstat -g env 命令: 打印环境变量值..... 459
onstat -g ffr 命令: 打印可用分段..... 461
onstat -g glo 命令: 显示全局多线程信息..... 461
onstat -g his 命令: 打印 SQL 跟踪信息..... 463
onstat -g ioa 命令: 打印合并的 onstat -g 信息..... 467

onstat -g iob 命令: 打印大缓冲区的使用摘要..... 469
onstat -g iof 命令: 显示异步 I/O 统计信息..... 469
onstat -g iog 命令: 打印 AIO 全局信息..... 470
onstat -g ioq 命令: 显示 I/O 队列信息..... 471
onstat -g ipl 命令: 打印索引页日志记录状态信息..... 472
onstat -g iov 命令: 显示 AIO VP 统计信息..... 472
onstat -g lap 命令: 显示轻量追加状态信息..... 474
onstat -g laq 命令: 打印辅助服务器队列..... 474
onstat -g lmm 命令: 显示低内存管理信息..... 475
onstat -g lmx 命令: 打印所有锁定的互斥..... 477
onstat -g lsc 命令: 打印活动的轻扫描状态 (弃用)..... 478
onstat -g mem 命令: 显示池内存统计信息..... 479
onstat -g mgm 命令: 显示 MGM 资源信息..... 480
onstat -g nbm 命令: 打印块位图..... 482
onstat -g nsc 命令: 打印当前共享内存连接信息..... 483
onstat -g nsd 命令: 打印轮询线程的共享内存数据..... 486
onstat -g nss 命令: 打印共享内存网络连接的状态..... 486
onstat -g ntd 命令: 打印网络统计信息..... 487
onstat -g ntm 命令: 打印网络邮件统计信息..... 488
onstat -g ntt 命令: 打印网络用户时间..... 488
onstat -g ntu 命令: 打印网络用户统计信息..... 488
onstat -g opn 命令: 打印打开的分区..... 489
onstat -g osi: 打印操作系统信息..... 489
onstat -g pos 命令: 打印文件中的值..... 489
onstat -g ppd 命令: 显示分区压缩字典信息..... 490
onstat -g ppf 命令: 打印分区概要文件..... 491
onstat -g pqs 命令: 打印所有 SQL 查询的运算符..... 492
onstat -g prc 命令: 显示使用 UDR 或 SPL 例程的会话..... 493
onstat -g proxy 命令: 打印代理分发器信息..... 494
onstat -g qst 命令: 打印等待队列和条件队列的等待选项..... 498
onstat -g rah 命令: 显示预读请求统计信息..... 499
onstat -g rbm 命令: 打印共享内存的块映射..... 500
onstat -g rea 命令: 显示准备就绪的线程..... 501
onstat -g rss 命令: 打印 RS 辅助服务器信息..... 501
onstat -g rwm 命令: 打印读取和写入互斥..... 505
onstat -g sch 命令: 打印 VP 信息..... 506
onstat -g scn 命令: 显示扫描信息..... 506
onstat -g sds 命令: 打印 SD 辅助服务器信息..... 509
onstat -g seg 命令: 显示共享内存段的统计信息..... 512
onstat -g ses 命令: 显示会话相关信息..... 513
onstat -g shard 命令: 打印有关分片定义的信息..... 519
onstat -g sle 命令: 打印所有休眠的线程..... 522
onstat -g smb 命令: 显示智能大对象空间信息..... 523
onstat -g smx 命令：显示多路复用器组信息..... 524
onstat -g spi 命令: 显示使用长自旋的自旋锁..... 525
onstat -g sql 命令: 打印与 SQL 有关的会话信息..... 526
onstat -g spf: 打印已就绪语句的概要文件..... 528
onstat -g src 命令: 共享内存中的模式..... 529
onstat -g ssc 命令: 显示 SQL 语句出现次数..... 529
onstat -g stk 命令: 打印线程堆栈..... 531
onstat -g stm 命令: 打印 SQL 语句的内存使用..... 532
onstat -g stq 命令: 打印队列信息..... 532
onstat -g sts 命令: 打印每个线程的堆栈使用..... 533
onstat -g sym 命令: 打印 oninit 实用程序的符号表信息..... 533
onstat -g tpf 命令: 打印线程概要文件..... 534
onstat -g ufr 命令: 打印内存池分段..... 535

onstat -g vpcache 命令: 显示 CPU 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息..... 536
onstat -g wai 命令: 打印等待队列线程列表..... 538
onstat -g wmx 命令: 打印具有等待者的所有互斥..... 539
onstat -g wst 命令: 打印线程的等待统计信息..... 539
onstat -G 命令: 打印 TP/XA 事务信息..... 541
onstat -h 命令: 打印缓冲头哈希链信息..... 543
onstat -i 命令: 启动 交互模式..... 544
onstat -k 命令: 打印活动锁的信息..... 544
onstat -l 命令: 显示物理和逻辑日志信息..... 546
onstat -L 命令: 显示可用锁的数量..... 549
onstat -m 命令: 打印最近的系统消息日志信息..... 550
onstat -o 命令: 共享内存内容输出到文件..... 550
onstat -p 命令: 显示概要文件计数..... 550
onstat -P 命令: 打印分区信息..... 554
onstat -r 命令: 重复打印选择的统计信息..... 555
onstat -R 命令: 显示 LRU、FLRU 和 MLRU 队列信息..... 557
onstat -s 命令: 打印锁存器的信息..... 559
onstat -t and onstat -T 命令: 打印表空间信息..... 560
onstat -u 命令: 显示用户活动的概要文件..... 562
onstat -x 命令: 显示数据库服务器事务信息..... 564
确定逻辑日志记录的位置..... 566
确定全局事务的方式..... 567
onstat -X 命令: 显示线程信息..... 567
onstat -z 命令: 清除统计信息..... 569
退出 onstat 实用程序时的返回码..... 570

第III部分 SQL 管理 API..... 571
SQL 管理 API 函数..... 571
SQL 管理 API 概述..... 571
admin() 和 task() 函数语法行为..... 571
admin() 和 task() 参数大小规范..... 572
admin() 和 task() 函数返回码..... 572
SQL 管理 API 门户: 按权限组划分参数..... 573
add bufferpool 参数: 添加缓冲池 (SQL 管理 API)..... 583
add chunk 参数: 添加新块 (SQL 管理 API)..... 584
add log 参数: 添加新逻辑日志 (SQL 管理 API)..... 585
add memory 参数: 增加共享内存 (SQL 管理 API)..... 586
add mirror 参数: 添加镜像块 (SQL 管理 API)..... 586
alter chunk 参数: 更改块状态为联机或脱机 (SQL 管理 API)..... 587
alter logmode 参数: 更改数据库日志记录模式 (SQL 管理 API)..... 588
alter plog 参数: 更改物理日志 (SQL 管理 API)..... 589
archive fake 参数: 执行无记录的备份 (SQL 管理 API)..... 589
autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API).... 590
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API). 591
autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API)..... 591
autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API)..... 592
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API). 593
cdr 参数: 管理 Enterprise Replication (SQL 管理 API)..... 594
cdr add trustedhost 参数: 添加可信任主机 (SQL 管理 API)..... 595
cdr autoconfig serv 参数: 自动配置连接性和复制 (SQL 管理 API)..... 596
cdr list trustedhost 参数: 罗列可信任主机 (SQL 管理 API)..... 598
cdr remove trustedhost 参数: 移除可信任主机 (SQL 管理 API)..... 599
check data 参数: 检查数据一致性 (SQL 管理 API)..... 600
check extents 参数: 检查扩展数据块一致性 (SQL 管理 API)..... 601

check partition 参数: 检查分区一致性 (SQL 管理 API)..... 601
checkpoint 参数: 强制检查点 (SQL 管理 API)..... 602
clean sbspace 参数: 释放未引用的智能大对象 (SQL 管理 API)..... 603
create blobspace 参数: 创建 BLOB 空间 (SQL 管理 API)..... 603
create blobspace from storagepool 参数: 从存储池创建 BLOB 空间 (SQL 管理 API).. 604
create chunk 参数: 创建块 (SQL 管理 API)..... 605
create chunk from storagepool 参数: 从存储池创建块 (SQL 管理 API)..... 606
create database 参数: 创建数据库 (SQL 管理 API)..... 607
create dbaccessdemo 参数: 创建演示数据库 (SQL 管理 API)..... 608
create dbspace 参数: 创建数据库空间 (SQL 管理 API)..... 608
create dbspace from storagepool 参数: 从存储池创建数据库空间 (SQL 管理 API).... 610
create plogspace: 创建物理日志空间 (SQL 管理 API)..... 611
create sbspace 参数: 创建智能大对象空间 (SQL 管理 API)..... 612
create sbspace from storagepool 参数: 从存储池创建智能大对象空间 (SQL 管理 API). 613
create sbspace with accesstime 参数: 创建跟踪访问时间的智能大对象空间 (SQL 管理 API)..... 614
create sbspace with log 参数: 创建具有事务日志记录的智能大对象空间 (SQL 管理 API) 615
create tempdbspace 参数: 创建临时数据库空间 (SQL 管理 API)..... 616
create tempdbspace from storagepool 参数: 从存储池创建临时数据库空间 (SQL 管理 API)..... 617
create tempsbspace 参数: 创建临时智能大对象空间 (SQL 管理 API)..... 617
create tempsbspace from storagepool 参数: 从存储池创建临时智能大对象空间 (SQL 管理 API)..... 618
defragment 参数: 动态地对分区扩展块取消分段 (SQL 管理 API)..... 619
drop blobspace 参数: 删除 Blob 空间 (SQL 管理 API)..... 620
drop blobspace to storagepool 参数: 从空 BLOB 空间归还空间到存储池 (SQL 管理 API) 621
drop chunk 参数: 删除块 (SQL 管理 API)..... 621
drop chunk to storagepool 参数: 将空块的空间归还到存储池 (SQL 管理 API)..... 622
drop database 参数: 删除数据库 (SQL 管理 API)..... 623
drop dbspace 参数: 删除数据库空间 (SQL 管理 API)..... 623
drop dbspace to storagepool 参数: 从空数据库空间归还空间到存储池 (SQL 管理 API). 624
drop log 参数: 删除逻辑日志 (SQL 管理 API)..... 624
drop plogspace: 删除物理日志空间 (SQL 管理 API)..... 625
drop sbspace 参数: 删除智能大对象空间 (SQL 管理 API)..... 626
drop sbspace to storagepool 参数: 从空智能大对象空间归还空间到存储池 (SQL 管理 API)..... 626
drop tempdbspace 参数: 删除临时数据库空间 (SQL 管理 API)..... 627
drop tempdbspace to storagepool 参数: 从空临时数据库空间归还空间到存储池 (SQL 管理 API)..... 627
drop tempsbspace to storagepool 参数: 从空临时智能大对象空间归还空间到存储池 (SQL 管理 API)..... 628
export config 参数: 导出配置参数值 (SQL 管理 API)..... 628
file status 参数: 显示消息日志文件的状态 (SQL 管理 API)..... 629
grant admin 参数: 授予运行 SQL 管理 API 命令的权限..... 630
ha make primary 参数: 更改辅助服务器的模式 (SQL 管理 API)..... 631
ha rss 参数: 创建 RS 辅助服务器 (SQL 管理 API)..... 631
ha rss add 参数: 添加 RS 辅助服务器到主服务器 (SQL 管理 API)..... 632
ha rss change 参数: 更改 RS 辅助服务器的密码 (SQL 管理 API)..... 633
ha rss delete 参数: 删除 RS 辅助服务器 (SQL 管理 API)..... 633
ha sds clear 参数: 停止共享磁盘复制 (SQL 管理 API)..... 634
ha sds primary 参数: 将 SD 辅助服务器转换为主服务器 (SQL 管理 API)..... 635
ha sds set 参数: 创建共享磁盘主服务器 (SQL 管理 API)..... 635
ha set idxauto 参数: 复制索引到辅助服务器 (SQL 管理 API)..... 636
ha set ipl 参数: 在主服务器上构建日志索引 (SQL 管理 API)..... 637
ha set primary 参数: 定义 HDR 主服务器 (SQL 管理 API)..... 637
ha set secondary 参数: 定义 HDR 辅助服务器 (SQL 管理 API)..... 638

ha set standard 参数: 将 HDR 服务器转换成为标准服务器 (SQL 管理 API)..... 638
ha set timeout 参数: 更改 SD 辅助服务器超时 (SQL 管理 API)..... 639
import config 参数: 导入配置参数值 (SQL 管理 API)..... 640
index compress repack shrink 参数: 优化 B 型树索引的存储 (SQL 管理 API)..... 640
index estimate_compression 参数: 估计索引压缩 (SQL 管理 API)..... 642
message log delete 参数: 删除消息日志文件 (SQL 管理 API)..... 643
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API)..... 643
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API)..... 644
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API)..... 645
modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API)..... 646
modify chunk extendable off 参数: 标记块为不可扩展 (SQL 管理 API)..... 647
modify config 参数 : 更改配置参数 (SQL 管理 API)..... 647
modify space expand 参数: 扩大空间的大小 (SQL 管理 API)..... 648
modify space sp_sizes 参数: 更改可扩展的存储空间的大小 (SQL 管理 API)..... 649
onbar 参数: 备份存储空间 (SQL 管理 API)..... 650
onmode 和 a 参数: 添加共享内存段 (SQL 管理 API)..... 651
onmode 和 c 参数: 强制检查点 (SQL 管理 API)..... 652
onmode 和 C 参数: 控制 B 型树扫描程序 (SQL 管理 API)..... 653
onmode 和 d 参数: 设置数据复制类型 (SQL 管理 API)..... 654
onmode 和 D 参数: 设置 PDQ 优先级 (SQL 管理 API)..... 655
onmode 和 e 参数: 更改 SQL 语句高速缓存的使用 (SQL 管理 API)..... 655
onmode 和 F 参数: 释放未使用的内存段 (SQL 管理 API)..... 656
onmode 和 j 参数: 切换数据库服务器到管理模式 (SQL 管理 API)..... 657
onmode 和 l 参数: 切换到下一个逻辑日志 (SQL 管理 API)..... 657
onmode 和 m 参数: 切换到多用户模式 (SQL 管理 API)..... 658
onmode 和 M 参数: 临时更改决策支持内存 (SQL 管理 API)..... 658
onmode 和 n 参数: 解锁驻留内存 (SQL 管理 API)..... 659
onmode 和 O 参数: 标记禁用的数据库空间为 down (SQL 管理 API)..... 660
onmode 和 p 参数: 添加或移除虚拟处理器 (SQL 管理 API)..... 660
onmode 和 Q 参数: 设置决策支持查询的最大数目 (SQL 管理 API)..... 661
onmode 和 r 参数: 强制共享内存的驻留 (SQL 管理 API)..... 662
onmode 和 S 参数: 设置决策支持扫描的最大数目 (SQL 管理 API)..... 662
onmode 和 W 参数: 重设语句高速缓存属性 (SQL 管理 API)..... 663
onmode 和 wf 参数: 永久地更新配置参数 (SQL 管理 API)..... 664
onmode 和 wm 参数: 临时地更新配置参数 (SQL 管理 API)..... 664
onmode、wm 和 AUTO_LRU_TUNING 参数: 更改 LRU 调整状态 (SQL 管理 API)..... 665
onmode 和 Y 参数: 更改会话的查询计划度量 (SQL 管理 API)..... 666
onmode 和 z 参数: 终止用户会话 (SQL 管理 API)..... 667
onmode 和 Z 参数: 终止分布式事务 (SQL 管理 API)..... 667
onsmsync 参数: 与存储管理器目录同步 (SQL 管理 API)..... 668
onstat 参数: 监视数据库服务器 (SQL 管理 API)..... 669
ontape archive 参数: 备份数据库上的数据 (SQL 管理 API)..... 670
print error 参数: 打印错误消息 (SQL 管理 API)..... 671
print file info 参数: 显示目录或文件信息 (SQL 管理 API)..... 671
print partition 参数: 打印分区信息 (SQL 管理 API)..... 672
rename space 参数: 重命名存储空间 (SQL 管理 API)..... 673
reset config 参数: 恢复配置参数值 (SQL 管理 API)..... 674
reset config all 参数: 恢复所有动态地可更新的配置参数值 (SQL 管理 API)..... 674
reset sysadmin 参数: 移动 sysadmin 数据库 (SQL 管理 API)..... 675
restart listen 参数: 动态地停止并启动监听线程 (SQL 管理 API)..... 675
revoke admin 参数: 撤销运行 SQL 管理 API 命令的权限..... 676
scheduler 参数: 停止或启动调度程序 (SQL 管理 API)..... 677
scheduler lmm enable 参数: 指定自动低内存管理设置 (SQL 管理 API)..... 677
scheduler lmm disable 参数: 停止自动低内存管理 (SQL 管理 API)..... 679
set chunk 参数: 更改块的状态 (SQL 管理 API)..... 680
set dataskip 参数: 启动或停止跳过数据库空间 (SQL 管理 API)..... 680

set index compression 参数: 更改索引页压缩 (SQL 管理 API)..... 681
set onconfig memory 参数: 临时地更改配置参数 (SQL 管理 API)..... 682
set onconfig permanent 参数: 永久地更改配置参数 (SQL 管理 API).....683
set sbspace accesstime 参数: 控制访问时间跟踪 (SQL 管理 API)..... 683
set sbspace avg_lo_size 参数: 设置智能大对象的平均大小 (SQL 管理 API).....684
set sbspace logging 参数: 更改智能大对象空间的日志记录 (SQL 管理 API).....685
set sql tracing 参数: 设置全局 SQL 跟踪 (SQL 管理 API).....685
set sql tracing database 参数: 更改数据库跟踪 (SQL 管理 API)..... 686
set sql tracing session 参数: 控制对会话的跟踪 (SQL 管理 API)..... 687
set sql tracing user 参数: 控制对用户的跟踪 (SQL 管理 API)..... 688
set sql user tracing 参数: 设置对用户会话的全局 SQL 跟踪 (SQL 管理 API).....688
start json listener 参数: 启动 MongoDB API 有线监听器..... 689
start listen 参数: 动态地启动监听线程 (SQL 管理 API)..... 690
start mirroring 参数: 启动存储空间镜像 (SQL 管理 API)..... 690
stop json listener 参数: 停止有线监听器..... 691
stop listen 参数: 动态地停止监听线程 (SQL 管理 API)..... 692
stop mirroring 参数: 停止存储空间镜像 (SQL 管理 API)..... 692
storagepool add 参数: 添加存储池条目 (SQL 管理 API)..... 693
storagepool delete 参数: 删除存储池条目 (SQL 管理 API)..... 695
storagepool modify 参数: 更改存储池条目 (SQL 管理 API)..... 696
storagepool purge 参数: 删除存储池条目 (SQL 管理 API)..... 697
表和分段压缩和解压缩操作 (SQL 管理 API)..... 698
table 或 fragment 参数: 压缩数据和优化存储 (SQL 管理 API).....699
估计压缩操作的输出 (SQL 管理 API)..... 703
purge compression dictionary 参数: 移除压缩字典 (SQL 管理 API).....704
tenant create 参数: 创建租户数据库 (SQL 管理 API)..... 705
tenant drop 参数: 删除租户数据库 (SQL 管理 API)..... 710
tenant update 参数: 更改租户数据库属性 (SQL 管理 API)..... 710

附录 A 附录..... 715
数据库服务器文件..... 715
故障排除错误..... 719
使用 onmode -I 收集诊断信息..... 719
创建跟踪点..... 719
使用 ifxcollect 工具收集数据.....719
事件警报..... 721
使用 ALARMPROGRAM 捕获事件.....722
ph_alert 表中的事件..... 723
事件警报参数..... 724
事件警报 ID..... 725
连接管理器事件警报标识符..... 773
数据库服务器日志中的消息..... 776
本章中消息是如何排列的..... 776
消息: A-B..... 777
消息: C..... 778
消息: D-E-F..... 787
消息: G-H-I..... 791
消息: J-K-L-M..... 794
消息: N-O-P..... 798
消息: Q-R-S..... 804
消息: T-U-V..... 809
消息: W-X-Y-Z..... 813
消息: 符号..... 815
转换和复原错误消息..... 815
Conversion and Reversion Messages for Enterprise Replication.....818

动态日志消息..... 820
智能大对象空间元数据消息..... 821
截断表消息..... 822
SinoDB® 中的限制..... 822

插图清单

图 1: BTSCANNER 配置参数语法.....
图 2: memory 字段语法.....
图 3: buffers 字段语法.....
图 4: DATASKIP 配置参数语法.....
图 5: HA_FOC_ORDER 配置参数的语法.....
图 6: SQLTRACE 配置参数语法图.....
图 7: VPCLASS 配置参数语法图.....
图 8: SMI 表中的列..... 241
图 9: 调度程序表之间的关系..... 244
图 10: 常规数据库空间初始块..... 254
图 11: 附加数据库空间块..... 254
图 12: 镜像块结构..... 255
图 13: 可用列表页..... 255
图 14: 带有 partnum 值的 systables 查询的输出..... 256
图 15: 带有 partn 值的 sysfragments 表的输出..... 257
图 16: 数据库表空间在根数据库空间中的位置..... 257
图 17: 表的扩展数据块结构..... 259
图 18: 索引的扩展数据块结构..... 260
图 19: 下一扩展数据块分配策略..... 261
图 20: 剩余页..... 263
图 21: 分段表的磁盘结构..... 264
图 22: 完整 B 型树结构..... 265
图 23: 根节点..... 265
图 24: 在叶节点 3 填满后创建叶节点 4..... 266
图 25: 分支节点的典型内容..... 266

图 26: 叶节点 3 和 4.....267
图 27: 单个智能大对象空间块.....270
图 28: 单块智能大对象空间的元数据区域的结构.....271
图 29: 智能大对象页结构.....271
图 30: 新数据库.....272
图 31: onlog 输出中的智能大对象记录.....287
图 32: 语法：.....
图 33: 语法：.....
图 34: 语法：.....
图 35: 语法：.....
图 36: 语法：.....
图 37: 语法：.....
图 38: 语法：.....
图 39: 语法：.....
图 40: 语法：.....
图 41: 语法：.....
图 42: 语法：.....
图 43: 语法.....
图 44: UNIX 语法图：.....
图 45: Windows 语法图：.....
图 46: 语法.....
图 47: 语法：.....
图 48: 语法：.....
图 49: 语法：.....
图 50: 语法：.....
图 51: 语法：.....
图 52: 语法：.....
图 53: 语法：.....

图 54: 语法：.....
图 55: 语法：.....
图 56: 语法：.....
图 57: 语法：.....
图 58: 语法：.....
图 59: 语法：.....
图 60: 语法：.....
图 61: 语法：.....
图 62: 语法：.....
图 63: 语法：.....
图 64: 语句：.....
图 65: 语法：.....
图 66: 语法：.....
图 67: 语法：.....
图 68: 语法：.....
图 69: 语法：.....
图 70: 语法：.....
图 71: 语法：.....
图 72: 语法：.....
图 73: 语法：.....
图 74: 语法：.....
图 75: 语法.....
图 76: 语法：.....
图 77: 语法：.....
图 78: 语法：.....
图 79: 语法.....
图 80: 语法.....
图 81: 语法：.....

图 82: 语法：.....
图 83: 语法：.....
图 84: 语法：.....
图 85: 语法：.....
图 86: 语法：.....
图 87: 语法：.....
图 88: 语法：.....
图 89: 语法：.....
图 90: 语法：.....
图 91: 语法：.....
图 92: 语法：.....
图 93: 语法：.....
图 94: 语法：.....
图 95: 语法：.....
图 96: 语法：.....
图 97: 语法：.....
图 98: onstat -b 命令输出..... 406
图 99: 语法：.....
图 100: 语法：.....
图 101: 语法：.....
图 102: 使用 prof 选项的 onstat -C 命令输出..... 409
图 103: 使用 hot 选项的 onstat -C 命令输出..... 409
图 104: 使用 part 选项的 onstat -C 命令输出..... 410
图 105: 使用 clean 选项的 onstat -C 命令输出..... 411
图 106: onstat -C range..... 412
图 107: onstat -C map..... 412
图 108: onstat -C alice..... 413
图 109: 语法：.....

图 110: onstat -d 命令输出..... 414
图 111: 语法：.....
图 112: onstat -D 命令输出..... 419
图 113: 语法：.....
图 114: 语法：.....
图 115: onstat -F 命令输出..... 420
图 116: 语法：.....
图 117: onstat -g act 命令输出..... 421
图 118: 语法：.....
图 119: onstat -g afr 命令输出..... 422
图 120: 语法：.....
图 121: 语法：.....
图 122: onstat -g aqt 命令输出..... 422
图 123: onstat -g aqt aqt_name 命令输出..... 423
图 124: 语法：.....
图 125: onstat -g arc 命令输出..... 425
图 126: 语法：.....
图 127: onstat -g ath 命令输出..... 426
图 128: 语法：.....
图 129: onstat -g bth 命令输出..... 427
图 130: 语法：.....
图 131: onstat -g buf 对于 memory 设置的输出..... 429
图 132: 语法：.....
图 133: 语法：.....
图 134: onstat -g ckp 命令输出..... 434
图 135: 语法：.....
图 136: 语法：.....
图 137: onstat -g cluster 命令输出（在主服务器上运行）..... 440

图 138: onstat -g cluster verbose 命令输出（在主服务器上运行）..... 441
图 139: onstat -g cluster 命令输出（在辅助服务器上运行）..... 442
图 140: 语法：.....
图 141: 语法：.....
图 142: onstat -g con 命令输出..... 446
图 143: 语法：.....
图 144: onstat -g cpu 命令输出..... 447
图 145: 语法：.....
图 146: onstat -g dbc 命令输出..... 448
图 147: 语法：.....
图 148: onstat -g defragment 命令输出..... 450
图 149: 语法：.....
图 150: onstat -g dic 命令输出..... 451
图 151: 语法：.....
图 152: onstat -g dis 命令输出..... 452
图 153: 语法：.....
图 154: onstat -g dll 命令输出..... 453
图 155: 语法：.....
图 156: onstat -g dmp 命令输出..... 453
图 157: 语法：.....
图 158: onstat -g dri 命令输出..... 454
图 159: onstat -g dri ckpt 命令输出..... 455
图 160: onstat -g dri que 命令输出..... 456
图 161: 语法.....
图 162: onstat -g dsc 命令输出..... 457
图 163: 语法：.....
图 164: onstat -g dsk 命令的压缩操作输出..... 458
图 165: onstat -g dsk 命令的重打包操作输出..... 458

图 166: 语法：.....
图 167: onstat -g env 命令输出.....460
图 168: 语法：.....
图 169: onstat -g ffr aio 命令输出.....461
图 170: 语法：.....
图 171: onstat -g glo 命令输出.....461
图 172: 语法：.....
图 173: 语法.....
图 174: 语法：.....
图 175: onstat -g iob 命令输出.....469
图 176: 语法：.....
图 177: onstat -g iof 命令输出.....470
图 178: 语法：.....
图 179: onstat -g iog 命令输出.....470
图 180: 语法：.....
图 181: onstat -g ioq 命令输出.....471
图 182: 语法：.....
图 183: onstat -g ipl 命令输出.....472
图 184: 语法：.....
图 185: onstat -g iov 命令输出.....472
图 186: 语法：.....
图 187: onstat -g lap 命令输出.....474
图 188: 语法：.....
图 189: onstat -g laq 命令输出.....475
图 190: 语法：.....
图 191: onstat -g lmm 命令输出.....475
图 192: 语法：.....
图 193: onstat -g lmx 命令输出.....477

图 194: 语法：.....
图 195: onstat -g lsc 命令输出.....478
图 196: 语法：.....
图 197: onstat -g mem 命令输出.....479
图 198: 语法：.....
图 199: onstat -g mgm 命令输出.....480
图 200: 语法：.....
图 201: onstat -g nbm 命令输出.....483
图 202: 语法：.....
图 203: onstat -g nsc 命令输出.....483
图 204: 带有 client id 的 onstat -g nsc 命令输出.....484
图 205: 语法：.....
图 206: onstat -g nsd 命令输出.....486
图 207: 语法：.....
图 208: onstat -g nss 命令输出.....486
图 209: 语法：.....
图 210: onstat -g ntd 命令输出.....487
图 211: 语法：.....
图 212: onstat -g ntm 命令输出.....488
图 213: 语法：.....
图 214: onstat -g ntt 命令输出.....488
图 215: 语法：.....
图 216: onstat -g ntu 命令输出.....488
图 217: 语法：.....
图 218: onstat -g osi 命令输出.....489
图 219: 语法：.....
图 220: onstat -g pos 命令输出.....490
图 221: 语法：.....

图 222: onstat -g ppd 输出.....490
图 223: 语法：.....
图 224: onstat -g ppf 命令输出.....491
图 225: 语法：.....
图 226: onstat -g pqs 命令输出.....492
图 227: 语法：.....
图 228: onstat -g prc 命令输出.....493
图 229: 语法：.....
图 230: onstat -g proxy 命令输出（从主服务器上运行）.....494
图 231: onstat -g proxy 命令输出（从辅助服务器上运行）.....495
图 232: onstat -g proxy all 命令输出（从主服务器上运行）.....495
图 233: onstat -g proxy all 命令输出（从辅助服务器上运行）.....496
图 234: onstat -g proxy proxy_id 命令输出（从辅助服务器上运行）.....497
图 235: onstat -g proxy_id proxy_transaction_id 命令输出（从辅助服务器上运行）.....497
图 236: onstat -g proxy_id proxy_transaction_id sequence_number 命令输出（从辅助服务器上运行）.....498
图 237: 语法：.....
图 238: onstat -g qst 命令输出.....499
图 239: 语法：.....
图 240: onstat -g rah 命令输出.....499
图 241: 语法：.....
图 242: onstat -g rbm 命令输出.....501
图 243: 语法：.....
图 244: onstat -g rea 命令输出.....501
图 245: 语法：.....
图 246: onstat -g rss verbose 命令输出，在主服务器上运行。.....502
图 247: onstat -g rss log 命令输出，在主服务器上运行。.....503
图 248: onstat -g rss 命令输出，在 RS 辅助服务器上运行。.....504

图 249: onstat -g rss verbose 命令输出，在 RS 辅助服务器上运行。..... 504
图 250: 语法：.....
图 251: onstat -g rwm 命令输出..... 505
图 252: 语法：.....
图 253: onstat -g sch 命令输出..... 506
图 254: 语法：.....
图 255: onstat -g scn 输出显示表信息..... 507
图 256: onstat -g scn 输出显示索引扫描信息..... 507
图 257: 语法：.....
图 258: onstat -g sds 命令输出（在主服务器上运行）..... 509
图 259: onstat -g sds server_name 命令输出（在主服务器上运行）..... 509
图 260: onstat -g sds verbose 命令输出（在 SD 辅助服务器上运行）..... 511
图 261: 语法：.....
图 262: onstat -g seg 命令输出..... 512
图 263: 语法：.....
图 264: onstat -g ses 命令输出..... 513
图 265: onstat -g ses session_id 命令输出..... 514
图 266: 语法：.....
图 267: 对于使用一致性哈希算式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。..... 520
图 268: 对于使用哈希算式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。..... 521
图 269: 对于使用表达式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。..... 521
图 270: 对于使用 BSON 分片键和表达式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。..... 522
图 271: 语法：.....
图 272: onstat -g sle 命令输出..... 522
图 273: 语法：.....

图 274: 语法：.....
图 275: onstat -g smx 命令输出.....524
图 276: onstat -g smx ses 输出.....525
图 277: 语法：.....
图 278: onstat -g spi 命令输出.....526
图 279: 语法：.....
图 280: onstat -g sql 命令输出.....527
图 281: 语法：.....
图 282: onstat -g spf 命令输出.....528
图 283: 语法：.....
图 284: onstat -g src 命令输出.....529
图 285: 语法：.....
图 286: onstat -g ssc 命令输出.....530
图 287: 语法：.....
图 288: onstat -g stk tid 命令输出.....531
图 289: 语法：.....
图 290: onstat -g stm 命令输出.....532
图 291: 语法：.....
图 292: onstat -g stq 命令输出.....532
图 293: 语法：.....
图 294: onstat -g sts command output.....533
图 295: 语法：.....
图 296: onstat -g sym 命令输出.....533
图 297: 语法：.....
图 298: onstat -g tpf 命令输出.....534
图 299: 语法：.....
图 300: 指定池名的 onstat -g ufr global 命令输出.....535
图 301: 指定会话 ID 的 onstat -g ufr 命令输出.....535

图 302: 语法：.....
图 303: onstat -g vpcache 命令输出..... 536
图 304: 语法：.....
图 305: onstat -g wai 命令输出..... 538
图 306: 语法：.....
图 307: onstat -g wmx 命令输出..... 539
图 308: 语法：.....
图 309: 语法：.....
图 310: onstat -G 命令输出..... 541
图 311: 语法：.....
图 312: onstat -h 命令输出..... 543
图 313: 语法：.....
图 314: 语法：.....
图 315: onstat -k 命令输出..... 544
图 316: 语法：.....
图 317: onstat -l 命令输出..... 546
图 318: 语法：.....
图 319: onstat -L 输出..... 549
图 320: 语法：.....
图 321: onstat -m 命令输出..... 550
图 322: 语法：.....
图 323: 语法：.....
图 324: onstat -p 命令输出..... 551
图 325: 语法：.....
图 326: onstat -P 命令输出..... 554
图 327: 语法：.....
图 328: 命令输出.....555
图 329: 命令输出.....555

图 330: onstat -r 1 -h 命令输出.....555
图 331: onstat -rFh 命令输出.....556
图 332: 语法：.....
图 333: onstat -R 命令输出.....557
图 334: 语法：.....
图 335: onstat -s 命令输出.....559
图 336: 语法：.....
图 337: onstat -t 命令输出.....560
图 338: 语法：.....
图 339: onstat -u 命令输出.....562
图 340: 语法：.....
图 341: onstat -x 命令输出.....564
图 342: 语法：.....
图 343: onstat -X 命令输出.....567
图 344: 语法：.....
图 345: 语法.....
图 346: 语法.....
图 347: 语法.....
图 348: 语法：索引压缩命令参数.....
图 349: 语法：index estimate_compression 命令参数.....
图 350: 语法.....
图 351: 语法.....
图 352: 语法.....
图 353: 语法：表数据压缩命令参数.....
图 354: 语法：分段数据压缩命令参数.....

表格清单

表 1: 根数据库空间配置参数..... 45
表 2: 物理日志配置参数..... 45
表 3: 逻辑日志配置参数..... 45
表 4: 长事务配置参数..... 45
表 5: 服务器消息文件配置参数..... 46
表 6: 表空间配置参数..... 46
表 7: 临时数据库空间和智能大对象空间配置参数..... 46
表 8: 默认数据库空间和智能大对象空间配置参数..... 46
表 9: 系统配置参数..... 47
表 10: 网络配置参数..... 47
表 11: CPU 虚拟处理器配置参数..... 47
表 12: CPU 虚拟处理器配置参数..... 48
表 13: AIO 和缓冲区清除程序配置参数..... 48
表 14: 锁配置参数..... 48
表 15: 共享内存配置参数..... 49
表 16: 检查点、恢复时间目标和系统阻塞时间配置参数..... 49
表 17: 转换防护配置参数..... 49
表 18: 分布式事务配置参数..... 50
表 19: 磁带设备配置参数..... 50
表 20: 逻辑日志磁带设备配置参数..... 50
表 21: ON-Bar 配置参数..... 50
表 22: SinoDB 主要存储管理器 配置参数..... 51
表 23: 数据字典高速缓存配置参数..... 52
表 24: 数据分布配置参数..... 52
表 25: UDR 配置参数..... 52

表 26: SQL 语句高速缓存配置参数.....52
表 27: 操作系统和会话配置参数.....53
表 28: 索引配置参数.....53
表 29: PDQ 配置参数.....53
表 30: 优化器配置参数.....54
表 31: 扫描配置参数.....54
表 32: SQL 跟踪配置参数.....54
表 33: 安全配置参数.....55
表 34: LBAC 配置参数.....55
表 35: 内置字符数据类型配置参数.....55
表 36: 序列高速缓存数据类型配置参数.....56
表 37: Enterprise Replication 配置参数.....56
表 38: 并行分片查询配置参数.....57
表 39: 高可用性集群配置参数.....57
表 40: 逻辑恢复配置参数.....58
表 41: 诊断配置参数.....58
表 42: 报警程序配置参数.....59
表 43: 技术支持配置参数.....59
表 44: 字符处理配置参数.....59
表 45: 队列和等待统计配置参数.....59
表 46: 用户映射.....59
表 47: 存储供应配置参数.....60
表 48: 自动定位配置参数.....60
表 49: 默认转义字符配置参数.....60
表 50: MQ 配置参数.....60
表 51: 非 root 用户服务器安装.....61
表 52: 低内存配置参数.....61
表 53: 连接配置参数.....61

表 54: 会话限制配置参数.....62
表 55: 租户限制配置参数.....62
表 56: Java 配置参数.....62
表 57: 缓冲池和 LRU 调整配置参数.....63
表 58: 不在 onconfig.std 文件中的参数.....63
表 59: 样本脚本.....65
表 60: 对内存和存储空间分配的影响.....70
表 61: BTSCANNER 配置参数值选项.....78
表 62: BUFFERPOOL 配置参数值的选项.....80
表 63: DATASKIP 配置参数值的选项.....87
表 64: 排序溢出文件的位置.....92
表 65: DRINTERVAL、HDR_TXN_SCOPE 和日志记录设置的矩阵，及其生成的 HDR 复制模式.....100
表 66: DRINTERVAL、HDR_TXN_SCOPE 和日志记录设置，以及其结果的 HDR 复制模式的矩阵.....118
表 67: NETTYPE 配置参数值的选项.....136
表 68: 建议的 SHMADD 值.....166
表 69: SQLTRACE 配置参数值的选项.....177
表 70: 从 Windows 注册表来启用命令扩展.....184
表 71: 有效的辅助服务器 USELASTCOMMITTED 值.....192
表 72: VPCLASS 配置参数值的选项.....196
表 73: SMI 表.....201
表 74: The syschunks 表.....206
表 75: syscluster 表信息.....208
表 76: syscmsm 表信息.....209
表 77: syscmsmsla 表信息.....209
表 78: syscmsmtab 表信息.....210
表 79: syscmsmunit 表信息.....210
表 80: 压缩字典信息.....211

表 81: sysdatabases 视图信息..... 212
表 82: sysdbslocale 表信息..... 213
表 83: sysdbspaces 表信息..... 213
表 84: sysha_lagtime 表信息..... 216
表 85: sysha_type 表信息..... 217
表 86: sysha_type 表中值的描述.....217
表 87: sysha_workload 表信息..... 217
表 88: sysipl 表信息..... 218
表 89: syslocks 表信息..... 218
表 90: syslogs 表信息..... 219
表 91: syslogfil 表中的信息..... 220
表 92: sysmgminfo 表信息..... 221
表 93: sysrepevtreg 表信息..... 227
表 94: sysrepstats 表信息..... 228
表 95: 事件类.....228
表 96: REPEVT_CLUST_CHG 事件类的子事件.....228
表 97: REPEVT_CLUST_PERFSTAT 事件类的子事件.....229
表 98: REPEVT_CLUST_LATSTAT 事件类的子事件.....229
表 99: REPEVT_CM_ADM 事件类的子事件.....229
表 100: REPEVT_SRV_ADM 事件类的子事件.....230
表 101: REPEVT_ER_ADM 事件类的子事件.....230
表 102: sysrsslog 表信息..... 230
表 103: syssesappinfo 表列信息.....231
表 104: syssessions 表信息..... 232
表 105: syssmx 表列信息..... 233
表 106: syssmxses 表列信息..... 234
表 107: syssqexplain 伪表..... 234
表 108: syssrcrss 表列信息..... 238

表 109: ph_task 表..... 244
表 110: ph_run 表..... 246
表 111: ph_group 表..... 247
表 112: ph_alert 表..... 247
表 113: 警报类型和颜色..... 249
表 114: ph_threshold 表..... 249
表 115: 结果表..... 250
表 116: command_history 表..... 250
表 117: command_history 表中的示例信息..... 250
表 118: storagepool 表..... 251
表 119: tenant 表..... 252
表 120: 每个扩展数据块的默认大小和数据库空间类型..... 255
表 121: 表空间 tblspace 条目..... 256
表 122: onlog 的样本输出..... 274
表 123: onlog 头列的定义..... 275
表 124: 逻辑日志记录类型..... 275
表 125: 智能大对象的记录子类型..... 286
表 126: finderr 元素..... 290
表 127: genoncfg 实用程序的输入文件的参数..... 293
表 128: oncheck 选项和其功能..... 295
表 129: oncheck -pt 和 oncheck -pT 命令的选项..... 308
表 130: onclean 命令的语法元素..... 312
表 131: onshutdown 脚本的语法元素..... 313
表 132: ondblog 实用程序的返回码..... 318
表 133: oninit 命令元素..... 320
表 134: -FILE 选项..... 322
表 135: oninit 实用程序的返回码..... 323
表 136: size 参数值列表..... 357

表 137: 源服务器和目标服务器上允许的 MIRROR 配置参数的设置..... 359
表 138: -Df 缺省规范..... 373
表 139: onstat 实用程序归档信息选项.....386
表 140: onstat 实用程序高速缓存信息选项.....386
表 141: onstat 实用程序压缩选项.....387
表 142: onstat 实用程序调试选项.....388
表 143: onstat 实用程序 Enterprise Replication 选项.....388
表 144: onstat 实用程序高可用性复制选项.....389
表 145: onstat 实用程序 仓库加速器选项..... 390
表 146: onstat 实用程序 I/O 选项..... 390
表 147: onstat 实用程序锁和锁存器选项.....391
表 148: onstat 实用程序日志选项.....392
表 149: onstat 实用程序内存选项.....392
表 150: onstat 实用程序网络选项.....393
表 151: onstat 实用程序性能检查（第一层）..... 394
表 152: onstat 实用程序性能检查（第二层）..... 394
表 153: onstat 实用程序表选项.....395
表 154: onstat 实用程序线程选项.....396
表 155: onstat 实用程序用户/会话选项.....397
表 156: onstat 实用程序虚拟处理器选项.....398
表 157: onstat 实用程序等待选项.....398
表 158: 其他有用的 onstat 实用程序选项..... 398
表 159: 每个十六进制值的描述..... 415
表 160: 虚拟处理器摘要中列的描述..... 462
表 161: 虚拟处理器摘要中的列的描述..... 462
表 162: 各个虚拟处理器的列的描述..... 463
表 163: 每个 onstat -g pqs 命令调用的描述..... 492
表 164: 备份和复原的 admin() 和 task() 函数参数..... 574

表 165: 消息日志命令的 admin() 和 task() 函数参数..... 574
表 166: 授予和撤销权限的 admin() 和 task() 函数参数.....574
表 167: 高可用性集群命令的 admin() 和 task() 函数参数.....575
表 168: onstat 命令的 admin() 和 task() 函数参数..... 576
表 169: 配置参数命令的 admin() 和 task() 函数参数..... 576
表 170: 数据、分区和扩展数据块命令的 admin() 和 task() 函数参数..... 576
表 171: 监听线程命令的 admin() 和 task() 函数参数..... 577
表 172: 消息日志命令的 admin() 和 task() 函数参数..... 577
表 173: 内存命令的 admin() 和 task() 函数参数..... 577
表 174: PDQ 命令的 admin() 和 task() 函数参数..... 577
表 175: 服务器模式命令的 admin() 和 task() 函数参数.....578
表 176: SQL 语句高速缓存命令的 admin() 和 task() 函数参数.....578
表 177: 其他管理任务命令的 admin() 和 task() 函数参数.....578
表 178: 监视消息日志、Enterprise Replication 或压缩估算的 admin() 和 task() 函数参数..... 579
表 179: Enterprise Replication 命令的 admin() 和 task() 函数参数.....579
表 180: 数据库和错误消息的 admin() 和 task() 函数参数.....579
表 181: SQL 跟踪命令的 admin() 和 task() 函数参数..... 579
表 182: 表存储命令的 admin() 和 task() 函数参数..... 580
表 183: 压缩命令的 admin() 和 task() 函数参数..... 580
表 184: 日志命令的 admin() 和 task() 函数参数..... 580
表 185: 镜像命令的 admin() 和 task() 函数参数..... 581
表 186: 空间命令的 admin() 和 task() 函数参数..... 581
表 187: 存储供应空间命令的 admin() 和 task() 函数参数.....582
表 188: 租户数据库命令的 admin() 和 task() 函数参数.....583
表 189: export config 命令元素..... 628
表 190: import config command elements..... 640
表 191: 索引压缩操作的参数.....641

表 192: 索引压缩命令元素.....641
表 193: Index estimate_compression 命令元素.....642
表 194: modify config 命令元素.....647
表 195: reset config 命令元素.....674
表 196: 压缩和解压缩操作的参数.....699
表 197: 表压缩和存储优化命令元素.....700
表 198: 分段压缩和存储优化命令元素.....701
表 199: estimate_compression 命令显示的信息.....703
表 200: 您可使用的数据库服务器文件.....715
表 201: 仅供内部使用的数据库服务器文件.....718
表 202: 数据收集选项.....720
表 203: FTP 选项（如果还传输数据）.....720
表 204: 类和子类的组合.....720
表 205: 事件警报参数.....724
表 206: 事件严重性代码.....724
表 207: 事件警报.....726
表 208: Severity 5 event alarms.....770
表 209: 连接管理器事件警报.....773
表 210: 系统级参数限制.....823
表 211: 表级参数限制.....823
表 212: 系统缺省值.....824
表 213: 存取能力.....824

简介

本简介概述了本出版物中的信息，并描述了所使用的约定。

关于本出版物

本出版物提供有关 SinoDB® 的参考资料。它包含配置参数、sysmaster 数据库中的系统监视接口（SMI）表、数据库服务器实用程序（例如 onmode 和 onstat）的语法、逻辑日志记录、磁盘结构、事件警报和未编号的错误消息等的详细说明。本出版物有两个副卷，《*SinoDB*® 管理员指南》 和 《*SinoDB*® 性能指南》。

本节讨论本出版物的目标读者以及使用管理实用程序所必须具备的关联软件产品。

用户类型

本出版物是为以下用户编写的：

- 数据库管理员
- 系统管理员
- 性能工程师

本出版是在假设您已具备以下背景知识的基础上编写的：

- 具备计算机、操作系统以及操作系统所提供的实用程序的实践知识
- 具备使用关系数据库的一些经验，或了解数据库概念
- 具备数据库服务器管理、操作系统管理或网络管理的一些经验

软件依赖性

本出版物是在假设您使用 SinoDB® V12.10 作为数据库服务器的基础上编写的。

演示数据库

DB-Access实用程序随SinoDB®数据库服务器产品一起提供，它包括一个或多个以下演示数据库：

- stores_demo 数据库以一家虚构的体育用品批发商的有关信息举例说明了关系模式。星瑞格®出版物中的许多示例均基于 stores_demo 数据库。
- superstores_demo 数据库举例说明了对象关系模式。superstores_demo 数据库包含扩展数据类型、类型和表继承以及用户自定义例程的示例。

有关如何创建和填充演示数据库的信息，请参阅《*SinoDB® DB-Access* 用户指南》。有关数据库及其内容的描述，请参阅《*SinoDB® SQL* 指南*:* 参考》。

用于安装演示数据库的脚本位于 UNIX™ 平台上的 $INFORMIXDIR/bin 目录和 Windows™ 环境中的 %INFORMIXDIR%\bin 目录中。

示例代码约定

SQL 代码的示例出现在整个出版物中。除非另有说明，代码不特定于任何单个的 SinoDB® 应用程序开发工具。

如果示例中仅列出 SQL 语句，那么它们将不用分号定界。例如：您可能看到以下示例中的代码：

```
CONNECT TO stores_demo
...
```

```
DELETE FROM customer
   WHERE customer_num = 121
...

COMMIT WORK
DISCONNECT CURRENT
```

要将此 SQL 代码用于特定产品，必须应用该产品的语法规则。例如，如果使用的是 SQL API，那么必须在每条语句的开头使用 EXEC SQL ，并在每条语句的结尾使用分号（或其他合适的定界符）。 如果使用的是 DB-Access，那么必须用分号将多条语句隔开。

> 提示: 代码示例中的省略点表示在整个应用程序中可添加更多的代码，但对于正在讨论的概念是不需要显示的。

有关使用特定应用程序开发工具或 SQL API 的 SQL 语句的详细指导，请参阅您的产品文档。

语法图

语法图使用特殊组件描述 SQL 语句和命令的语法。

沿着线的路径，从左向右，从上往下阅读语法图。

右向双箭头和直线符号►►──表示语法图开始。

直线和右向单箭头符号──►表示语法换行继续。

右向箭头和直线符号►──表示语法接续前一行。

直线、右向和左向箭头符号──►◄表示语法图结束。

语法片段由管线符号|——开始，以线管符号——| 结束。

必选项在水平线（主路径）上。

required_item

可选项在主路径下方。

required_item
[*optional_item*]

如果有两个以上选项，则其会以堆栈形式呈现。

如果必须从选项中选一个选项，那么堆栈的其中一个选项会在主路径上。

required_item
{ *required_choice1required_choice2* }

如果选项为可选的，整个堆栈会在主路径下方。

required_item
[{ *optional_choice1optional_choice2* }]

如果其中一项是默认项，它会在主路径上方，剩余选项则在主路径下方。

required_item
[{ *default_choiceoptional_choiceoptional_choice* }]

在主路径的上方，有个返回到左边的箭头，表示该项可被重复。这种情况下，重复项必须用一个或多个空格间隔。

required_item
repeatable_item

如果重复箭头包含一个逗号，表示重复项目间必须使用逗号分隔。

required_item

, *repeatable_item*

堆栈上方重复箭头表示可以从在堆栈中选择多个选项，或者是重复某一项。

SQL 关键词是以大写来显示（例如，FROM），您必须完全按照显示来拼写。变量是以小写来显示（例如，column-name），它们表示语法中用户提供的字段名或变量值。

如出现标点符号、圆括号、算术运算符，或是其他类似符号，您必须将其视为语法的一部分输入。

有时候，单个变量表示一个语法段。例如，在下图中，变量parameter-block表示标记为 parameter-block的语法段。

required_item <parameter-block>
parameter-block
parameter-block
{ *parameter1* | *parameter2* { *parameter3* *parameter4* } }

符合行业标准

SinoDB® 产品符合各种标准。

基于 SinoDB® SQL 的产品完全符合 SQL-92 入门标准（发布为 ANSI X3.135-1992），即 ISO 9075:1992 标准。另外，SinoDB® 数据库服务器的许多功能都遵循 SQL-92 中级和最高级标准，以及 X/Open SQL 公共应用程序环境 (CAE) 标准。

第 I 部分

配置与监视 SinoDB®

数据库配置参数

SinoDB® 数据库服务器在初始话期间使用一个称为 onconfig 的配置文件。该文件包含默认配置参数值。您可更改这些参数值以提升实例或数据库的性能和其他特性。

ONCONFIG 环境变量标识您的 onconfig 文件。

onconfig 文件

添加或更改 onconfig 文件中的信息时，必须遵循该文件所使用的编辑约定。

参数说明与可能的值显示在 onconfig.std 文件中条目上方的注释中。

以下显示了参数行的语法：

```
PARAMETER_NAME parameter_value comments
```

onconfig 文件内容的规则如下：

- 一行一个参数。
- 以 # 符号开头的行为注释行。
- onconfig 文件的最大行限制为 512 字节。超出此限制的行将被截断，而且可能导致配置问题。
- 参数名、参数值和可选注释之间必须放置空格（制表符和/或空格）。参数值内不可使用制表符或空格。参数值和空白之后的任何字符都视为注释，不论这些字符的开头是否有 # 符号。
- 参数及其值是区分大小写的。参数名总是大写。如果值条目是以大写字母来描述的，那么您必须使用大写字母（例如，NETTYPE 参数的 CPU 值）。
- 大多数参数只能有一个有效条目。如果 onconfig 文件中这些参数有一个以上的条目，那么只有第一个条目是有效的。然而，有些参数可有多个条目，例如 DBSERVERALIASES 配置参数以逗号区隔条目。某些参数可出现多次，例如 VPCLASS 配置参数。
- 复制但忽略无法辨识的参数，且不会报错。

  提示: 如果在 onconfig.std 模板文件中运行一个像 grep 的实用程序，那么指定新行字符（^）以便只返回配置参数名称和值。若没有新行字符，那么参数说明也会被返回。

  例如，以下命令将返回配置参数说明和值：

  ```
  grep "MSGPATH" onconfig.std
  # MSGPATH      - The path of the Sinoregal DS message log file
  MSGPATH $INFORMIXDIR/tmp/online.log
  ```

然而，以下命令仅返回配置参数值：

```
grep "^MSGPATH" onconfig.std
MSGPATH $INFORMIXDIR/tmp/online.log
```

环境变量的约定

可以在适用环境变量的任何配置参数中使用环境变量作为配置参数值。例如，可使用以下环境变量来代替数据库服务器的名称作为 DBSERVERNAME 配置参数的值。

```
DBSERVERNAME      $MY_DBSERVERNAME
```

重要： 如果使用环境变量作为值，那么必须在读取 onconfig 文件的可执行程序或实用程序的环境中设置该环境变量。读取 onconfig 文件的实用程序包括 oninit、oncheck、onbar、ontape、onlog 和 archecker。

相关链接

设置实用程序的本地环境变量 在第290页

更改 onconfig 文件

您可以更改数据库服务器上的 onconfig 文件，以定制服务器功能或调整数据库行为。

默认情况下，onconfig 文件在 INFORMIXDIR/etc 目录中。ONCONFIG 环境变量指定 onconfig 文件的名称和位置。

onconfig.std 文件是一个模板配置文件，您可以从中复制配置参数设置。onconfig.std 文件是一个模板，不是一个功能配置。可以复制或重命名 onconfig.std 文件，但不可以更改或删除 onconfig.std文件。如果您在配置文件的副本中忽略一个参数值，那么数据库服务器将使用 onconfig.std 模板文件中的默认值或基于其他参数值计算出该值

可通过以下方式更改 onconfig 文件：

- 可使用 SinoDB® 开放管理工具 (OAT) 来监视和更新您的配置。OAT 提供配置参数值建议来优化您的数据库服务器配置。
- 可使用文本编辑器来更改配置参数值。在下一次数据库服务器关闭并重新启动之后，这些变更才会生效。
- 您可以通过运行 onmode -wf 永久更新配置参数，或通过运行 onmode -wm 命令更新内存中的配置参数来动态地更改配置参数值，而无需重新启动数据库服务器。
- 可通过运行 genoncfg 实用程序来生成一个 onconfig 文件，其中包含了针对您估计的连接、磁盘空间和 CPU 使用情况进行了优化的设置。
- 您可以导出、导入和更改组中的配置参数：
  - 使用 onmode -we 命令将当前配置的快照导出到一个文件。将该结果快照可归档、用作为配置文件或导入到另一个正在运行的实例。
  - 使用 onmode -wi 命令从先前导出的文件导入可调配置参数。该文件中不可动态地调整的配置参数会被忽略。
- 可使用 SQL 管理 API 命令来修改、重置、导出和导入配置文件：
  - 使用 admin() 或 task() 函数的 modify config 参数来更改配置参数值。
  - 使用 admin() 或 task() 函数的 export config 和 import config 参数来导出和导入包含多个可动态调整的配置参数的文件。
  - 使用 admin() 或 task() 函数的 reset config 和 reset config all 参数来将一个配置参数值或全部配置参数的值恢复为 onconfig 文件中的值。

通过运行 onconfig_diff 实用程序，可以比较两个 onconfig 文件。

相关链接

genoncfg 实用程序 在第291页
onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -we: 导出包含当前配置参数的文件 在第347页
onmode -wi: 导入配置参数文件 在第348页
modify config 参数 : 更改配置参数 *(SQL* 管理 *API)* 在第647页
reset config 参数: 恢复配置参数值 *(SQL* 管理 *API)* 在第674页
reset config all 参数: 恢复所有动态地可更新的配置参数值 *(SQL* 管理 *API)* 在第674页
import config 参数: 导入配置参数值 *(SQL* 管理 *API)* 在第640页
export config 参数: 导出配置参数值 *(SQL* 管理 *API)* 在第628页
onconfig_diff 实用程序 在第316页
《*SinoDB* 管理员指南》: 数据库服务器配置

显示 onconfig 文件中的设置

您可以使用若干工具来显示 onconfig 文件中的设置。

请使用以下工具之一来显示 onconfig 文件中的设置：

- 使用文本编辑器打开 onconfig 文件。
- 使用 onstat -c 命令或 SinoDB® 开放管理工具 (OAT) 来查看 onconfig 文件中的内容。
- 通过运行 onstat -g cfg 命令来查看配置参数列表及其当前值。如果动态地更新配置参数，那么当前值将与 onconfig 文件中的永久值不同。

  可以使用 onstat -g cfg 的其他选项来仅显示动态变更的配置参数，或显示有关所有配置参数的其他信息。

相关链接

onstat -c 命令: 显示 *ONCONFIG* 文件内容 在第408页
onstat -g cfg 命令: 显示配置参数的当前值 在第438页
《*SinoDB SQL* 指南: 参考》: *ONCONFIG* 环境变量

onconfig 门户: 按功能类别划分的配置参数

此节中的信息列出 UNIX™ onconfig.std 文件中的配置参数。

类别列表

要使用此节，您首先要从以下列表中确定适当的类别，然后跟著链接转到该类别的配置参数。这些类别罗列的顺序与 onconfig.std 文件中的顺序相同。不在 onconfig.std 文件中但您可添加到 onconfig 文件中的参数罗列在 *表 58:* 不在 *onconfig.std* 文件中的参数 在第63页。

- 根数据库空间配置参数 在第44页
- 物理日志配置参数 在第45页
- 逻辑日志配置参数 在第45页
- 长事务配置参数 在第45页
- 服务器消息文件配置参数 在第46页
- 表空间配置参数 在第46页
- 临时数据库空间和智能大对象空间配置参数 在第46页
- 数据库空间和智能大对象空间配置参数 在第46页
- 系统配置参数 在第47页
- 网络配置参数 在第47页
- *CPU* 相关的配置参数 在第47页
- 自动调整配置参数 在第48页
- *AIO* 和清除程序相关的配置参数 在第48页
- 锁相关的配置参数 在第48页
- 共享内存配置参数 在第49页

- 检查点和系统阻塞配置参数 在第49页
- 转换防护配置参数 在第49页
- 事务相关的配置参数 在第49页
- *ontape* 磁带设备配置参数 在第50页
- *ontape* 逻辑日志磁带设备配置参数 在第50页
- 备份和恢复配置参数 在第50页
- 主存储管理器配置参数 在第51页
- 数据字典高速缓存配置参数 在第51页
- 数据分布配置参数 在第52页
- 用户定义例程 *(UDR)* 配置参数 在第52页
- *SQL* 语句高速缓存配置参数 在第52页
- 操作系统会话相关的配置参数 在第53页
- 索引相关的配置参数 在第53页
- 并行数据库查询 *(PDQ)* 配置参数 在第53页
- 优化器配置参数 在第54页
- 扫描配置参数 在第54页
- *SQL* 跟踪配置参数 在第54页
- 安全配置参数 在第55页
- 基于标签的访问控制配置参数 在第55页
- 内置字符数据类型配置参数 在第55页
- 序列高速缓存配置参数 在第55页
- *Enterprise Replication* 配置参数 在第56页
- 并行分片查询配置参数 在第57页
- 高可用性集群配置参数 在第57页
- 逻辑恢复配置参数 在第58页
- 诊断转储配置参数 在第58页
- 报警程序配置参数 在第59页
- 技术支持配置参数 在第59页
- 字符处理配置参数 在第59页
- 统计配置参数 在第59页
- 用户映射配置参数 在第59页
- 存储供应配置参数 在第60页
- 数据库对象的自动定位 在第60页
- 默认转义字符配置参数 在第60页
- *WebSphere MQ* 服务器配置参数 在第60页
- 非 *root* 用户服务器安装配置参数 在第61页
- 低内存配置参数 在第61页
- 连接参数 在第61页
- 会话限制 在第62页
- 租户限制 在第62页
- *Java* 配置参数 在第62页
- 缓冲池和 *LRU* 调整配置参数 在第62页
- 附加参数 在第63页

根数据库空间配置参数

使用以下配置参数来配置根数据库空间。

表 1: 根数据库空间配置参数

配置参数	参考
ROOTNAME 配置参数 在第151页	根数据库空间名称。
ROOTPATH 配置参数 在第151页	根数据库空间的路径。
ROOTOFFSET 配置参数 在第151页	根数据库空间的偏移量。
ROOTSIZE 配置参数 在第152页	根数据库空间的大小。
MIRROR 配置参数 在第132页	启用或禁用镜像。
MIRRORPATH 配置参数 在第132页	镜像的根数据库空间的路径。
MIRROROFFSET 配置参数 在第132页	镜像的根数据库空间的偏移量。

物理日志配置参数

使用以下配置参数来配置物理日志。

表 2: 物理日志配置参数

配置参数	参考
PHYSFILE 配置参数 在第144页	物理日志的大小。
PLOG_OVERFLOW_PATH 配置参数 在第144页	物理日志文件的溢出目录。
PHYSBUFF 配置参数 在第143页	物理日志缓冲的大小。

逻辑日志配置参数

使用以下配置参数来配置逻辑日志。

表 3: 逻辑日志配置参数

配置参数	参考
LOGFILES 配置参数 在第124页	逻辑日志文件的数目。
LOGSIZE 配置参数 在第126页	每个逻辑日志文件的大小。
DYNAMIC_LOGS 配置参数 在第109页	动态日志分配的类型。
LOGBUFF 配置参数 在第123页	逻辑日志缓冲的大小。

长事务配置参数

使用以下配置参数来控制长事务的回滚的时间。

表 4: 长事务配置参数

配置参数	参考
LTXHWM 配置参数 在第129页	回滚一个长事务之前，逻辑日志文件可填充的百分比。
LTXEHWM 配置参数 在第128页	在服务器暂挂其他活动以便长事务可独占使用日志之前，逻辑日志文件可填充的百分比。

服务器消息文件配置参数

使用以下配置参数来配置服务器消息文件。

表 5: 服务器消息文件配置参数

配置参数	参考
MSGPATH 配置参数 在第134页	消息文件的路径。
CONSOLE 配置参数 在第86页	控制台消息文件的路径。

表空间配置参数

使用以下配置参数来配置根数据库空间中的 tblspace。

表 6: 表空间配置参数

配置参数	参考
TBLTBLFIRST 配置参数 在第186页	表空间 tblspace 的第一个扩展数据块大小。
TBLTBLNEXT 配置参数 在第187页	T表空间 tblspace 的下一个扩展数据块大小。
TBLSPACE_STATS 配置参数 在第186页	启用或禁用表空间统计信息。

临时数据库空间和智能大对象空间配置参数

使用以下配置参数来配置默认临时数据库空间和智能大对象空间。

表 7: 临时数据库空间和智能大对象空间配置参数

配置参数	参考
DBSPACETEMP 配置参数 在第91页	临时对象的数据库空间列表。
SBSPACETEMP 配置参数 在第156页	临时智能大对象的智能大对象空间列表。

数据库空间和智能大对象空间配置参数

使用以下配置参数来配置默认数据库空间和智能大对象空间。

表 8: 默认数据库空间和智能大对象空间配置参数

配置参数	参考
SBSPACENAME 配置参数 在第155页	存储智能大对象的默认智能大对象空间。
SYSSBSPACENAME 配置参数 在第184页	系统统计信息的默认智能大对象空间。
ONDBSPACEDOWN 配置参数 在第140页	指定当数据库空间不可用时服务器的行为。

系统配置参数

使用以下参数来设置服务器实例信息。

表 9: 系统配置参数

配置参数	参考
SERVERNUM 配置参数 在第162页	数据库服务器实例的唯一标识。
DBSERVERNAME 配置参数 在第90页	默认数据库服务器的名称。
DBSERVERALIASES 配置参数 在第88页	替代的数据库服务器名称的列表。
FULL_DISK_INIT 配置参数 在第115页	防止现存服务器实例的意外磁盘重新初始化。

网络配置参数

使用以下配置参数来配置网络。

表 10: 网络配置参数

配置参数	参考
NETTYPE 配置参数 在第135页	The configuration of poll threads for a specific protocol. 对特定协议的轮询线程的配置
LISTEN_TIMEOUT 配置参数 在第122页	数据库服务器等待连接的时间。
MAX_INCOMPLETE_CONNECTIONS 配置参数 在第130页	最大不完整连接数。
FASTPOLL 配置参数 在第114页	启用或禁用快速轮询。
NUMFDSERVERS 配置参数 在第138页	对于 UNIX™ 上的网络连接，使用 NUMFDSERVERS 配置参数来指定轮询线程数，这些轮询线程用于在处理虚拟处理器之间移动的网路连接。
NS_CACHE 配置参数 在第137页	定义位于主机名/IP 地址高速缓存、服务高速缓存、用户高速缓存和组高速缓存等中的单个条目的最大保留时间。

CPU 相关的配置参数

使用以下配置参数来配置 CPU 虚拟处理器。

表 11: CPU 虚拟处理器配置参数

配置参数	参考
MULTIPROCESSOR 配置参数 在第134页	设置为 1 支持多个 CPU VP。
VPCLASS 配置参数 在第196页	定义每个 CPU 虚拟处理器类的特性。
VP_MEMORY_CACHE_KB 配置参数 在第195页	CPU 虚拟处理器的专用内存块的容量。

配置参数	参考
SINGLE_CPU_VP 配置参数 在第171页	设置为 0 以启用用户定义 CPU VP，或设置为 1 启用单个 CPU VP。

自动调整配置参数

使用以下配置参数来自动调整数据库服务器的配置。

表 12: CPU 虚拟处理器配置参数

配置参数	参考
AUTO_TUNE 配置参数 在第73页	启用或禁用所有自动调整配置参数，其值不存在 onconfig 文件中。
AUTO_LRU_TUNING 配置参数 在第70页	启用或禁用 LRU 队列的自动调整。
AUTO_AIOVPS 配置参数 在第67页	启用或禁用 AIO 虚拟处理器的自动管理。
AUTO_CKPTS 配置参数 在第67页	启用或禁用自动检查点。
AUTO_REPREPARE 配置参数 在第72页	启用或禁用自动地重新优化存储过程和重新准备已准备就绪的语句。
AUTO_STAT_MODE 配置参数 在第73页	启用或禁用系统的选择性更新统计信息的模式。
AUTO_READAHEAD 配置参数 在第71页	更改自动预读模式，或者禁用或启用查询的自动预读。

AIO 和清除程序相关的配置参数

使用以下配置参数来配置 AIO 虚拟处理器和缓冲区清除程序。

表 13: AIO 和缓冲区清除程序配置参数

配置参数	参考
VPCLASS 配置参数 在第196页	配置 AIO 虚拟处理器。
CLEANERS 配置参数 在第84页	页清除程序线程的数目。
DIRECT_IO 配置参数 *(UNIX)* 在第96页	指定是否使用直接 I/O。

锁相关的配置参数

使用以下配置参数来设置锁行为。

表 14: 锁配置参数

配置参数	参考
LOCKS 配置参数 在第123页	启动时锁的初始数目。
DEF_TABLE_LOCKMODE 配置参数 在第94页	默认表锁模式。

共享内存配置参数

使用以下配置参数来配置共享内存。

表 15: 共享内存配置参数

配置参数	参考
RESIDENT 配置参数 在第148页	控制共享内存是否驻留。
SHMBASE 配置参数 在第167页	共享内存基地址。不可更改此值。
SHMVIRTSIZE 配置参数 在第169页	共享内存的虚拟段的初始大小，以 KB 为单位。
SHMADD 配置参数 在第166页	虚拟共享内存段的大小。
EXTSHMADD 配置参数 在第112页	在用户定义虚拟处理器中运行的用户定义例程和 DataBlade® 例程的每个虚拟扩展共享内存段的大小。
SHMTOTAL 配置参数 在第168页	数据库服务器的共享内存的最大容量。
SHMVIRT_ALLOCSEG 配置参数 在第169页	控制何时添加内存段。
SHMNOACCESS 配置参数 在第167页	罗列服务器不可访问的共享内存地址。

检查点和系统阻塞配置参数

使用以下配置参数来配置检查点、恢复时间目标和系统阻塞时间。

表 16: 检查点、恢复时间目标和系统阻塞时间配置参数

配置参数	参考
CKPTINTVL 配置参数 在第83页	检查是否需要检查点的频率。
RTO_SERVER_RESTART 配置参数 在第153页	故障后重启的恢复时间目标。
BLOCKTIMEOUT 配置参数 在第77页	系统阻塞的时间长度。

转换防护配置参数

在服务器升级到新版期间 ，使用以下配置参数来控制 SinoDB® 使用的信息。

表 17: 转换防护配置参数

配置参数	参考
CONVERSION_GUARD 配置参数 在第86页	指定在升级期间发生错误时是停止升级还是继续升级。
RESTORE_POINT_DIR 配置参数 在第150页	当启用 CONVERSION_GUARD 配置参数时，指定升级失败期间用于放置恢复点文件的空目录的路径名。

事务相关的配置参数

使用以下配置参数来控制分布式事务。

表 18: 分布式事务配置参数

配置参数	参考
TXTIMEOUT 配置参数 在第189页	分布式事务的超时时间长度。
DEADLOCK_TIMEOUT 配置参数 在第93页	分布式事务中等待锁的最长时间。

ontape 磁带设备配置参数

使用以下参数来配置用 ontape 实用程序备份的磁带设备。

表 19: 磁带设备配置参数

配置参数	参考
TAPEDEV configuration parameter	备份的磁带设备。
TAPEBLK configuration parameter	磁带块的大小。
TAPESIZE configuration parameter	一个备份磁带上可放置的最大数据量。

ontape 逻辑日志磁带设备配置参数

使用以下配置参数来配置用 ontape 实用程序备份逻辑日志的磁带设备。

表 20: 逻辑日志磁带设备配置参数

配置参数	参考
LTAPEDEV configuration parameter	逻辑日志备份的磁带设备。
LTAPEBLK configuration parameter	逻辑日志备份的磁带块大小。
LTAPESIZE configuration parameter	一个逻辑日志备份磁带上可放置的最大数据量。

备份和恢复配置参数

使用以下配置参数来控制 ON-Bar 实用程序的备份和恢复。除非另有说明，否则这些配置参数的说明在《*SinoDB®* 备份和还原指南》中。

表 21: ON-Bar 配置参数

配置参数	参考
BAR_ACT_LOG configuration parameter	ON-Bar 活动日志文件的位置。
BAR_DEBUG_LOG configuration parameter	ON-Bar 调试日志文件的位置。
BAR_DEBUG configuration parameter	ON-Bar 的调试级别。
BAR_MAX_BACKUP configuration parameter	备份中使用的备份线程数。
BAR_MAX_RESTORE configuration parameter	恢复中使用的恢复线程数。

配置参数	参考
BAR_RETRY configuration parameter	再次尝试备份或恢复的次数。
BAR_NB_XPORT_COUNT configuration parameter	每个备份进程使用的数据缓冲区数。
BAR_XFER_BUF_SIZE configuration parameter	每个数据缓冲区大小。
RESTARTABLE_RESTORE configuration parameter	失败后，启用 ON-Bar 来继续备份。
BAR_PROGRESS_FREQ configuration parameter	进展消息放置到活动日志中的频率。
BAR_BSALIB_PATH configuration parameter	ON-Bar 和存储管理器的共享库的路径。
BACKUP_FILTER configuration parameter	备份期间使用的过滤程式的路径。
RESTORE_FILTER configuration parameter	恢复期间使用的过滤程式的路径。
BAR_PERFORMANCE configuration parameter	报告的 ON-Bar 性能统计信息的类型。
BAR_CKPTSEC_TIMEOUT configuration parameter	辅助服务器中等待归档检查点完成的时间，以秒为单位。

主存储管理器配置参数

使用以下配置参数来配置 SinoDB® 主要存储管理器。

表 22: SinoDB® 主要存储管理器 配置参数

配置参数	参考
PSM_ACT_LOG configuration parameter	指定 SinoDB® 主要存储管理器活动日志的位置，否则其日志信息将包含在 ON-Bar 活动日志中。
PSM_DEBUG_LOG configuration parameter	指定 SinoDB® 主要存储管理器调试日志的位置，否则其日志信息将包含在 ON-Bar 调试日置中。
PSM_DEBUG configuration parameter	指定在 SinoDB® 主要存储管理器调试日志中打印的信息量，否则将使用与 ON-Bar 相同的调试级别。
PSM_CATALOG_PATH configuration parameter	指定包含 SinoDB® 主要存储管理器目录表的目录的完整路径。
PSM_DBS_POOL configuration parameter	指定 SinoDB® 主要存储管理器放置备份和恢复数据库空间数据的池的名称。
PSM_LOG_POOL configuration parameter	指定 SinoDB® 主要存储管理器放置备份和恢复日志数据的池的名称。

数据字典高速缓存配置参数

使用以下配置参数来配置数据字典高速缓存。

表 23: 数据字典高速缓存配置参数

配置参数	参考
DD_HASHSIZE 配置参数 在第93页	数据字典高速缓存中散列存储区的数目。
DD_HASHMAX 配置参数 在第92页	每个散列存储区中最大的表数目。

数据分布配置参数

使用以下配置参数来配置数据分布池。

表 24: 数据分布配置参数

配置参数	参考
DS_HASHSIZE 配置参数 在第102页	数据分布高速缓存和其他高速缓存中的散列存储区的数目。
DS_POOLSIZE 配置参数 在第105页	数据分布高速缓存和其他高速缓存中条目中最大的条目数。

用户定义例程 (UDR) 配置参数

使用以下配置参数来配置 UDR。

表 25: UDR 配置参数

配置参数	参考
PC_HASHSIZE 配置参数 在第142页	UDR 高速缓存中的散列存储区的数目。
PC_POOLSIZE 配置参数 在第143页	UDR 高速缓存中最大的条目数。
PRELOAD_DLL_FILE 配置参数 在第146页	服务器启动时加载的 C UDR 共享库路径名。

SQL 语句高速缓存配置参数

使用以下配置参数来配置 SQL 语句高速缓存。

表 26: SQL 语句高速缓存配置参数

配置参数	参考
STMT_CACHE 配置参数 在第179页	控制 SQL 语句高速缓存。
STMT_CACHE_HITS 配置参数 在第180页	SQL 语句在高速缓存之前运行的次数。
STMT_CACHE_SIZE 配置参数 在第182页	SQL 语句高速缓存大小。
STMT_CACHE_NOLIMIT 配置参数 在第181页	控制 SQL 语句高速缓存的额外内存消耗。

配置参数	参考
STMT_CACHE_NUMPOOL 配置参数 在第181页	SQL 语句高速缓存的池数目。

操作系统会话相关的配置参数

使用以下配置参数来配置操作系统和会话特性。

表 27: 操作系统和会话配置参数

配置参数	参考
USEOSTIME 配置参数 在第192页	SQL 语句计时的精度。
STACKSIZE 配置参数 在第178页	会话堆栈大小。
ALLOW_NEWLINE 配置参数 在第66页	SQL 语句中是否允许嵌入换行字符
USELASTCOMMITTED 配置参数 在第191页	控制 committed read 隔离级别。

索引相关的配置参数

使用以下配置参数来配置索引特性。

表 28: 索引配置参数

配置参数	参考
FILLFACTOR 配置参数 在第114页	索引页充满百分比。
MAX_FILL_DATA_PAGES 配置参数 在第130页	如果数据页有可变长度行，启用或禁用尽可能填满数据页。
BTSCANNER 配置参数 在第77页	配置 B 型数扫描程序线程。
ONLIDX_MAXMEM 配置参数 在第140页	pre-image 和 updator 日志池的内存量。

并行数据库查询 (PDQ) 配置参数

使用以下配置参数来配置 PDQ。

表 29: PDQ 配置参数

配置参数	参考
MAX_PDQPRIORITY 配置参数 在第131页	单个查询的最大资源百分比。
DS_MAX_QUERIES 配置参数 在第102页	最大并发决策支持查询数
DS_TOTAL_MEMORY 配置参数 在第106页	决策支持的最大内存量。
DS_MAX_SCANS 配置参数 在第103页	决策支持的最大扫描线程数。

配置参数	参考
DS_NONPDQ_QUERY_MEM 配置参数 在第104页	非 PDQ 查询的内存量
DATASKIP 配置参数 在第87页	处理查询时是否跳过数据库空间。

优化器配置参数

使用以下配置参数来影响查询执行优化器计划和指令。

表 30: 优化器配置参数

配置参数	参考
OPTCOMPIND 配置参数 在第141页	控制优化器如何决定最佳查询路径。
DIRECTIVES 配置参数 在第97页	启用或禁用 inline 优化器指令。
EXT_DIRECTIVES 配置参数 在第112页	启用或禁用外部指令。
OPT_GOAL 配置参数 在第142页	控制如何优化进行最快检索。
IFX_FOLDVIEW 配置参数 在第119页	启用或禁用折叠视图。
STATCHANGE 配置参数 在第179页	为更改阈的全局百分比指定一个正整数，以确定需要更新的数据分布统计信息。
USTLOW_SAMPLE 配置参数 在第194页	当以 LOW 模式运行 UPDATE STATISTICS 语句时，启用或者禁用基于抽样生成索引统计。

扫描配置参数

使用以下配置参数来设置预读行为。

表 31: 扫描配置参数

配置参数	参考
BATCHEDREAD_TABLE 配置参数 在第76页	在压缩表上、在行大于一页的表上，以及在具有 VARCHAR、LVARCHAR 和 NVARCHAR 数据的表上，启用或禁用轻扫描。
BATCHEDREAD_INDEX 配置参数 在第76页	使优化器对索引执行轻扫描。

SQL 跟踪配置参数

使用以下配置参数来设置 SQL 跟踪。

表 32: SQL 跟踪配置参数

配置参数	参考
EXPLAIN_STAT 配置参数 在第111页	启用或禁用在说明输出文件中包含查询统计。
SQLTRACE 配置参数 在第177页	配置 SQL 跟踪。

安全配置参数

使用以下配置参数来配置安全选项。

表 33: 安全配置参数

配置参数	参考
DBCREATE_PERMISSION 配置参数 在第87页	指定可以创建数据库的用户。
DB_LIBRARY_PATH 配置参数 在第88页	指定 UDR 或 UDT 共享库的位置。
IFX_EXTEND_ROLE 配置参数 在第119页	控制如何指定可注册外部例程的用户。
SECURITY_LOCALCONNECTION 配置参数 在第162页	数据库服务器是否检查本地连接的安全性。
UNSECURE_ONSTAT 配置参数 在第190页	是否非 DBSA 的用可以运行 onstat 命令。
ADMIN_USER_MODE_WITH_DBSA 配置参数 在第64页	控制可在管理模式下连接服务器的用户。
ADMIN_MODE_USERS 配置参数 在第63页	罗列可在管理模式下连接的用户

基于标签的访问控制配置参数

使用以下配置参数来配置基于标签的访问控制 （LBAC）高速缓存。这些配置参数的说明在《*SinoDB*® 安全指南》中。

表 34: LBAC 配置参数

配置参数	参考
PLCY_POOLSIZE configuration parameter 在第145页	LBAC 安全信息高速缓存中的散列存储区的数目。
PLCY_HASHSIZE 配置参数 在第145页	LBAC 安全信息高速缓存的每个散列存储区中的最大条目数。
USRC_POOLSIZE 配置参数 在第194页	LBAC 凭证内存高速缓存中的散列存储区的数目。
USRC_HASHSIZE 配置参数 在第193页	LBAC 凭证内存高速缓存的每个散列存储区中的最大条目数。

内置字符数据类型配置参数

使用以下配置参数来配置内置字符数据类型。

表 35: 内置字符数据类型配置参数

配置参数	参考
SQL_LOGICAL_CHAR 配置参数 在第176页	启用或禁用内置字符数据类型声明中大小规格的扩展。

序列高速缓存配置参数

使用以下配置参数来配置序列高速缓存。

表 36: 序列高速缓存数据类型配置参数

配置参数	参考
SEQ_CACHE_SIZE 配置参数 在第162页	指定内存中高速缓存的最大序列对象数。

Enterprise Replication 配置参数

使用以下配置参数来配置 Enterprise Replication（ER）。这些配置参数的说明在 《*SinoDB® Enterprise Replication* 指南》 中。

表 37: Enterprise Replication 配置参数

配置参数	参考
CDR_EVALTHREADS Configuration Parameter	求值程序线程数。
CDR_DSLOCKWAIT Configuration Parameter	数据同步线程等待数据库锁的时间量。
CDR_QUEUEMEM Configuration Parameter	发送和接收队列的最大内存量。
CDR_NIFCOMPRESS Configuration Parameter	网络接口压缩级别。
CDR_SERIAL Configuration Parameter	Serial 列的起始值和增量大小。
CDR_DBSPACE Configuration Parameter	syscdr 数据库的数据库空间的名称。
CDR_QDATA_SBSPACE Configuration Parameter	假脱机事务的智能大对象空间的名称。
CDR_SUPPRESS_ATSRISWARN Configuration Parameter	ATS 和 RIS 文件中不显示数据同步报警和错误。
CDR_DELAY_PURGE_DTC configuration parameter	保留删除表的时间量。
CDR_LOG_LAG_ACTION configuration parameter	当数据库服务器即将覆盖 Enterprise Replication 尚未处理的逻辑日志时采取的行动。
CDR_LOG_STAGING_MAXSIZE Configuration Parameter	Enterprise Replication 用于暂存日志文件的空间量。
CDR_MAX_DYNAMIC_LOGS Configuration Parameter	Enterprise Replication 在会话中可发出的最大动态逻辑请求数。
GRIDCOPY_DIR Configuration Parameter	ifx_grid_copy 程序使用的默认目录。.
CDR_TSINSTANCEID configuration parameter	复制的时间序列实例的唯一标识符。
CDR_MAX_FLUSH_SIZE configuration parameter	在日志清空到磁盘之前应用的最大事务数。
CDR_AUTO_DISCOVER configuration parameter	允许通过 cdr autoconfig serv 命令、安装向导，或 ifxclone 实用程序自动配置 Enterprise Replication。
CDR_MEM configuration parameter	指定 Enterprise Replication 的内存池分配的方式。

并行分片查询配置参数

使用以下配置参数来配置并行分片查询。

表 38: 并行分片查询配置参数

配置参数	参考
SHARD_MEM configuration parameter	指定分片服务器上分片查询的共享内存分配方式。
SHARD_ID configuration parameter	为分片集群中分片服务器设置唯一标识符。

高可用性集群配置参数

使用以下配置参数来配置高可用性集群。

表 39: 高可用性集群配置参数

配置参数	参考
DRAUTO 配置参数 在第98页	控制主服务器的自动故障转移。
DRINTERVAL 配置参数 在第100页	缓冲区清空之间的最大间隔。
HDR_TXN_SCOPE 配置参数 在第118页	在客户端应用程序、主服务器和 HDR 辅助服务器之间调整事务同步。
DRTIMEOUT 配置参数 在第101页	网络超时时间
DRLOSTFOUND 配置参数 在第101页	HDR 失而复得文件的路径
DRIDXAUTO 配置参数 在第99页	启用或禁用自动索引修理。
HA_ALIAS 配置参数 在第115页	高可用性集群的服务器别名。
HA_FOC_ORDER 配置参数 在第116页	定义连接管理器使用的单个故障转移规则。
LOG_INDEX_BUILDS 配置参数 在第125页	启用或禁用索引页日志记录。
SDS_ENABLE 配置参数 在第157页	启用或禁用 SD 辅助服务器。
SDS_TIMEOUT 配置参数 在第161页	T主服务器等待 SD 辅助服务器响应的时间。
SDS_TEMPDBS 配置参数 在第160页	SD 辅助服务器使用的临时数据库空间。
SDS_ALTERNATE 配置参数 在第156页	高可用性集群中主服务器和 SD 辅助服务器之间通信的替代方法。
SDS_PAGING 配置参数 在第159页	SD 辅助服务器调页文件的路径。
SDS_LOGCHECK 配置参数 在第159页	主服务器是否生成日志活动，以及是否允许或防止主服务器的故障转移。
UPDATABLE_SECONDARY 配置参数 在第190页	辅助服务器是否可接受来自客户端的更新、插入或删除操作。
FAILOVER_CALLBACK 配置参数 在第113页	辅助服务器切换为标准或主服务器时调用的程序。
TEMPTAB_NOLOG 配置参数 在第187页	临时表的默认日志记录模式。
DELAY_APPLY 配置参数 在第95页	辅助服务器上应用事务的延迟时间。
STOP_APPLY 配置参数 在第182页	在 RS 辅助服务器上停止应用事务

配置参数	参考
LOG_STAGING_DIR 配置参数 在第125页	暂存日志文件的目录。
RSS_FLOW_CONTROL 配置参数 在第152页	启用 RS 辅助服务器的流量控制。
FAILOVER_TX_TIMEOUT 配置参数 在第113页	启用或禁用故障转移期间的事务存活行为。
ENABLE_SNAPSHOT_COPY 配置参数 在第111页	能否通过 ifxclone 实用程序克隆服务器实例。
SMX_COMPRESS 配置参数 在第172页	T从源数据库服务器发送数据到目标数据库服务器时，数据库服务器使用的压缩级别。
SMX_PING_INTERVAL 配置参数 在第172页	超时间隔的秒数。
SMX_PING_RETRY 配置参数 在第173页	辅助服务器关闭对主服务器的 SMX 连接之前的超时间隔数
CLUSTER_TXN_SCOPE 配置参数 在第84页	控制事务提交返回客户端应用程序的时间。
SMX_NUMPIPES 配置参数 在第172页	设置 SMX 连接的管道数。

逻辑恢复配置参数

使用以下配置参数来设置逻辑恢复线程。

表 40：逻辑恢复配置参数

配置参数	参考
ON_RECVRY_THREADS 配置参数 在第139页	在热恢复期间并行运行的逻辑恢复线程数。
OFF_RECVRY_THREADS 配置参数 在第139页	在冷恢复和快速恢复中使用的逻辑恢复线程数。

诊断转储配置参数

使用以下配置参数来控制诊断转储信息。

表 41：诊断配置参数

配置参数	参考
DUMPDIR 配置参数 在第108页	断言失败诊断文件的位置。
DUMPSHMEM 配置参数 *(UNIX)* 在第109页	控制共享内存转储。
DUMPGCORE 配置参数 *(UNIX)* 在第108页	启用或禁用数据库服务器将核心转储到 gcore 文件。
DUMPCORE 配置参数 *(UNIX)* 在第107页	启用或禁用数据库服务器在断言失败后转储核心。
DUMPCNT 配置参数 *(UNIX)* 在第107页	会话共享内存转储的最大断言失败数。

报警程序配置参数

使用以下配置参数来配置报警程序。

表 42: 报警程序配置参数

配置参数	参考
ALARMPROGRAM 配置参数 在第65页	显示事件报警的报警程序。
ALRM_ALL_EVENTS 配置参数 在第66页	报警程序是否针对所有事件运行。
STORAGE_FULL_ALARM 配置参数 在第183页	当存储空间变满或分区用尽页或扩展数据块时，消息和事件发出的频率。
SYSALARMPROGRAM 配置参数 在第184页	断言失败后触发的系统报警程序。

技术支持配置参数

以下配置参数自动设置，并由技术支持使用。

表 43: 技术支持配置参数

配置参数	参考
RAS_PLOG_SPEED	保留供支持使用。
RAS_LLOG_SPEED	保留供支持使用。

字符处理配置参数

使用以下配置参数来控制是否 SinoDB® 检查字符是否对语言环境有效。

表 44: 字符处理配置参数

配置参数	参考
EILSEQ_COMPAT_MODE 配置参数 在第110页	启用或禁用字符有效性检查。

统计配置参数

使用以下配置参数来控制队列和等待统计的收集。

表 45: 队列和等待统计配置参数

配置参数	参考
QSTATS 配置参数 在第147页	启用或禁用收集队列统计。
WSTATS 配置参数 在第198页	启用或禁用收集等待统计。

用户映射配置参数

使用此配置参数来控制用户映射。

表 46: 用户映射

配置参数	描述
USERMAPPING 配置参数 (UNIX, Linux) 在第193页	映射用户是否可以连接到 SinoDB®，如果可以，映射用户是否具有管理权限。

存储供应配置参数

使用以下配置参数来控制在现有的存储空间（数据库空间、临时数据库空间、智能大对象空间、临时智能大对象空间或 BLOB 空间）需要更多空间时服务器可自动扩展或添加块的信息。

表 47: 存储供应配置参数

配置参数	参考
SP_AUTOEXPAND 配置参数 在第174页	启用或禁用在存储空间中自动创建或扩展块。
SP_THRESHOLD 配置参数 在第174页	定义存储空间中可存在的最小可用 KB 数。
SP_WAITTIME 配置参数 在第175页	指定线程在返回 "out of space" 错误之前等待存储池扩展的最大秒数。

数据库对象的自动定位

使用以下配置参数来启用自动定位和分段存储。

表 48: 自动定位配置参数

配置参数	参考
AUTOLOCATE 配置参数 在第75页	启用数据库和表的自动定位，以及表的自动分段存储。

默认转义字符配置参数

按需要使用以下配置参数。

表 49: 默认转义字符配置参数

配置参数	参考
DEFAULTESCCHAR 配置参数 在第94页	指定默认转义字符

WebSphere® MQ 服务器配置参数

Use the following configuration parameters to configure the database server for MQ messaging. These configuration parameters are documented in the 《*SinoDB*® 数据库扩展用户指南》.

使用以下配置参数来配置 MQ 消息的数据库服务器。这些配置参数的说明在《*SinoDB*® 数据库扩展用户指南》中。

表 50: MQ 配置参数

配置参数	参考
MQSERVER configuration parameter	定义通道、指定 WebSphere® MQ 服务器位置和使用的通信方式。

配置参数	参考
MQCHLLIB configuration parameter	指定包含了 WebSphere® MQ 客户端通道定义表的目录的路径。
MQCHLTAB configuration parameter	指定 WebSphere® 客户端通道定义表的名称

非 root 用户服务器安装配置参数

使用以下配置参数进行非 root 服务器安装。

表 51: 非 root 用户服务器安装

配置参数	参考
REMOTE_SERVER_CFG 配置参数 在第147页	指定罗列了数据库服务器计算机信任的远程主机的文件名。
REMOTE_USERS_CFG 配置参数 在第148页	指定罗列了远程主机上可信任用户名的文件名。
S6_USE_REMOTE_SERVER_CFG 配置参数 在第154页	指定在可信任网络环境中用于认证安全服务器连接的文件。

低内存配置参数

使用以下配置参数 来管理低内存。

表 52: 低内存配置参数

配置参数	参考
LOW_MEMORY_RESERVE 配置参数 在第128页	保留一个特定内存量，以便需要执行关键活动时且服务器可用内存有限时使用。
LOW_MEMORY_MGR 配置参数 在第127页	更改当服务器达到内存限制时的默认行为。

连接参数

使用以下配置参数来管理连接。

表 53: 连接配置参数

配置参数	描述
INFORMIXCONRETRY 配置参数 在第120页	指定在向数据库服务器的初始连接尝试失败后，可以进行的连接尝试次数。使用 INFORMIXCONTIME 配置参数指定 CONNECT 语句尝试连接数据库服务器的频率。
INFORMIXCONTIME 配置参数 在第121页	指定 CONNECT 语句尝试建立到数据库服务器的连接的秒数。使用 INFORMIXRETRY 配置参数指定 CONNECT 语句尝试连接数据库服务器的频率。

会话限制

使用以下配置参数来创建各个会话限制。

表 54: 会话限制配置参数

配置参数	参考
SESSION_LIMIT_LOCKS 配置参数 在第163页	限制锁的数目。
SESSION_LIMIT_MEMORY 配置参数 在第164页	限制可用内存。
SESSION_LIMIT_TEMPSPACE 配置参数 在第165页	限制临时表空间。
SESSION_LIMIT_LOGSPACE 配置参数 在第163页	限制个别事务可用日志空间。
SESSION_LIMIT_TXN_TIME 配置参数 在第165页	限制一个事务可运行的时间量

租户限制

使用以下配置参数来指定租户数据库上的限制。

表 55: 租户限制配置参数

配置参数	参考
TENANT_LIMIT_SPACE 配置参数 在第189页	限制租户数据库可用的存储空间量。
TENANT_LIMIT_MEMORY 配置参数 在第188页	限制连接到租户数据库的所有会话的共享内存量。
TENANT_LIMIT_CONNECTIONS 配置参数 在第188页	限制到租户数据库的连接数。

Java™ 配置参数

使用以下配置参数来配置 Java™ 虚拟处理器。这些配置参数的说明在《*SinoDB® J/Foundation* 开发者指南》中。

表 56: Java™ 配置参数

配置参数	参考
VPCLASS	配置 Java™ 虚拟处理器类。
JVPPROPFILE	Java™ VP 特性文件。
JVPLOGFILE	Java™ VP 日志文件。
JVPARGS	配置 Java™ VM。
JVPCLASSPATH	Java™ 类路径。

缓冲池和 LRU 调整配置参数

使用以下配置参数来配置缓冲池和调整 LRU 队列。

表 57: 缓冲池和 LRU 调整配置参数

配置参数	参考
BUFFERPOOL 配置参数 在第78页	配置缓冲池。

附加参数

某些配置参数不在 onconfig.std 文件中。如需要，您可添加这些参数到您的 onconfig 文件。

表 58: 不在 onconfig.std 文件中的参数

配置参数	参考
AUTO_TUNE_SERVER_SIZE 配置参数 在第69页	基于预期用户数来设置数据库服务器大小。 如果您在安装期间创建服务器，此参数会设置在 onconfig 文件中。
AUTO_LLOG 配置参数 在第68页	在指定的数据库空间中自动添加逻辑日志，以提升性能和限制逻辑日志文件的总计大小。 如果您在安装期间创建服务器，此参数会设置在 onconfig 文件中。
CDR_APPLY Configuration Parameter	指定最小和最大的数据同步线程数。
CDR_ENV Configuration Parameter	设置某些特定的 Enterprise Replication 环境变量。
CHECKALLDOMAINSFORUSER 配置参数 在第83页	指定在 Windows™ 网络环境中数据库服务器如何查找用户名。
DISABLE_B162428_XA_FIX 配置参数 在第97页	指定在回滚操作后是否释放全局事务。
DRDA_COMMBUFFSIZE 配置参数 在第98页	指定 DRDA® 通信缓冲区的大小。
IFX_XA_UNIQUEXID_IN_DATABASE 配置参数 在第120页	使事务管理器能够使用相同的 XID 来表示同一数据库服务器实例中不同数据库上的全局事务。
LIMITNUMSESSIONS 配置参数 在第121页	指定可连接到数据库服务器的最大会话数。
MSG_DATE 配置参数 在第133页	在打印到联机日志的消息的开头插入日期戳记。
NET_IO_TIMEOUT_ALARM 配置参数 在第134页	如果网络写操作受阻 30 分钟或更久，则发送通知。
PN_STAGEBLOB_THRESHOLD 配置参数 在第145页	为循环分段中的 BYTE 和 TEXT 数据保留空间。

相关链接

数据库服务器文件 在第715页

ADMIN_MODE_USERS 配置参数

ADMIN_MODE_USERS 配置参数指定可以在管理模式下访问数据库服务器的用户（除用户 informix 和 DBSA 组成员以外）。

onconfig.std 值

未设置。仅用户 informix 和 DBSA 组成员可以在管理模式下访问 SinoDB®。

分隔符

用逗号分隔用户名，例如：Karin,Sarah,Andrew，字符串最长为 127 字节。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

永久保存 ADMIN_MODE_USERS 配置参数中的用户列表。可使用 onmode -wm 或 onmode -wf 命令来移除用户。

在数据库运行时，使用 onmode -j -U 来允许一个或多个用户在管理模式下访问数据库服务器。

必须将 ADMIN_USER_MODE_WITH_DBSA 配置参数设为 1，才可使 ADMIN_MODE_USERS 配置参数中所列用户可在管理模式下连接到数据库服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

使用 *-j* 选项使数据库服务器更改为管理模式 在第341页

ADMIN_USER_MODE_WITH_DBSA 配置参数 在第64页

oninit 实用程序 在第319页

ADMIN_USER_MODE_WITH_DBSA 配置参数

ADMIN_USER_MODE_WITH_DBSA 配置参数指定 informix 以外的用户是否可在管理模式下连接到数据库服务器。

onconfig.std 值

未设置。仅用户 informix 可以在管理模式下连接到数据库服务器。

值

0 = 仅用户 informix 可以在管理模式下连接

1 = 如果未设置 ADMIN_USER_MODE 配置参数，则以下用户可以在管理模式下连接：

- 用户 informix
- DBSA 组成员

如果 ADMIN_USER_MODE 配置参数设置了一个或多个用户名，那么以下用户可以在管理模式下连接：

- 用户 informix
- 组列表中包含 informix 组的用户（仅 UNIX™）
- DBSA 组成员
- 罗列在 ADMIN_MODE_USERS 配置参数中的管理用户

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

ADMIN_MODE_USERS 配置参数 在第63页

ALARMPROGRAM 配置参数

使用 ALARMPROGRAM 配置参数来指定处理事件报警和控制逻辑日志备份的 alarmprogram 文件的完整路径名。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/etc/alarmprogram.sh

在 Windows™ 上：%INFORMIXDIR%\etc\alarmprogram.bat

如不存在

在 UNIX™ 上：$INFORMIXDIR/etc/no_log.sh

在 Windows™ 上：%INFORMIXDIR%\etc\no_log.bat

值

pathname = alarmprogram 文件的完整路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

可设置 ALRM_ALL_EVENTS 配置参数来指定 ALARMPROGRAM 配置参数是针对 MSGPATH 中记录的所有事件运行，还是仅对特定值得注意的事件（事件严重性大于 1）运行。

如果 ALARMPROGRAM 配置参数指定的脚本不存在，则由默认报警处理器 no_log.sh 或 no_log.bat 取代之。有正确脚本后，更新 ALARMPROGRAM 配置参数的值以指定该脚本。可在服务器联机时通过使用 onmode -wm 命令来进行此更新。

以下提供样本脚本。

表 59: 样本脚本

脚本名（UNIX™）	脚本名（Windows™）	描述
log_full.sh	log_full.bat	设置 ALARMPROGRAM 为 log_full.sh 或 log_full.bat，当数据库服务器发出日志已满事件报警时自动备份逻辑日志。 您可修改脚本，并将其设置到 onconfig 文件中 ALARMPROGRAM 的完整路径。
no_log.sh	no_log.bat	设置 ALARMPROGRAM 为 no_log.sh 或 no_log.bat 将禁用自动逻辑日志备份。
alarmprogram.sh	alarmprogram.bat	处理事件报警和控制逻辑日志备份。修改 alarmprogram.sh 或 alarmprogram.bat，并将 ALARMPROGRAM 设置为 alarmprogram.sh 或 alarmprogram.bat 的完整路径名。请参阅 定制 *ALARMPROGRAM* 脚本 在第722页。

不使用提供的脚本，您可以编写您自己的 shell 脚本、批处理文件或二进制程序来执行事件。设置 ALARMPROGRAM 为该文件的完整路径名。当值得注意的事件发生时，数据库服务器执行该脚本。这些事件包括数据库、表、索引或智能大对象故障；所有日志已满；内部子系统故障；初始化失败；和长事务。事件通知可以使用电子邮件或传呼邮件消息。

要生成事件报警，将 ALARMPROGRAM 设置为 $INFORMIXDIR/etc/alarmprogram.sh 或 %INFORMIXDIR%\etc\alarmprogram.bat，并修改相应文件。

重要： 当选择自动逻辑日志备份时，备份介质应随时可供备份进程使用。

如果您通过 ALARMPROGRAM 参数进行自动日志备份设置，请不要使用连续日志备份命令（onbar -b -l -C）。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
定制 *ALARMPROGRAM* 脚本 在第722页
编写自己的警报脚本 在第722页
ALRM_ALL_EVENTS 配置参数 在第66页
事件警报 在第721页
ALRM_ALL_EVENTS 配置参数 在第66页

ALLOW_NEWLINE 配置参数

使用 ALLOW_NEWLINE 配置参数来决定是否允许所有会话的带引号字符串中出现换行符。

在 onconfig 文件中指定 ALLOW_NEWLINE，则允许分布式查询中所有远程会话支持嵌入的换行符。

onconfig.std 值

ALLOW_NEWLINE 0

值

0 = 不允许所有会话的带引号字符串中出现换行符。

1 = 允许所有会话的带引号字符串出现换行符。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

可以指定数据库服务器允许所有会话或特定会话的带引号字符串中出现换行符（\n）。会话是客户端连接到数据库的持续时间。

在未设置 ALLOW_NEWLINE 时，要允许或不允许当前会话的带引号字符串中出现换行符，您可以执行内置 ifx_allow_newline() 例程，并使用 't' 或 'f' 为唯一参数。

- 't' 支持带引号字符串中的换行符。
- 'f' 相反效果。

ifx_allow_newline() 的调用仅影响调用此例程的用户会话。

相关链接

《*SinoDB SQL* 指南: 语法》: 引号字串符
《*SinoDB ESQL/C* 程序员指南》: 引号字串符中的换行符

ALRM_ALL_EVENTS 配置参数

使用 ALRM_ALL_EVENTS 配置参数来指定 ALARMPROGRAM 配置参数是针对 MSGPATH 配置参数中记录的所有事件运行，还是仅针对值得注意的事件运行。

onconfig.std 值

ALRM_ALL_EVENTS 0

值

0 = 仅对值得注意的事件。

1 = 此参数触发 ALARMPROGRAM 配置参数，且 ALRM_ALL_EVENTS 配置参数显示所有事件报警。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

ALARMPROGRAM 配置参数 在第65页

ALARMPROGRAM 配置参数 在第65页

事件警报 在第721页

AUTO_AIOVPS 配置参数

当 数据库服务器侦测到现有的 AIO VP 的性能跟不上 I/O 工作负载超过时，AUTO_AIOVPS 配置参数使数据库服务器能自动增加异步 I/O 虚拟处理器（AIO VP）和页清除程序线程的数目。

onconfig.std 值

未设置。如果 AUTO_TUNE 配置参数设置为 1，则 AIO VPs 和页清除程序线程会自动地增加。

值

0 = 关闭

1 = 开启

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

如果当前 onconfig 文件中未设置 AUTO_AIOVPS 值，编辑 AUTO_TUNE 配置参数并重启数据库服务器。

用法

VPCLASS aio 配置参数控制 AIO VP 的数目。如果在 onconfig 文件中未设置 VP aio 参数，则当启用 AUTO_AIOVPS 时，数据库服务器启动的 AIO VP 初始值等于 AIO 块的数目。未设置 aio 时，数据库服务器可以启动的最大 AIO VP 数为 128。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_TUNE 配置参数 在第73页

VPCLASS 配置参数 在第196页

《*SinoDB* 性能指南》: 自动检查点、*LRU* 调整以及 *AIO* 虚拟处理器调整

DIRECT_IO 配置参数 *(UNIX)* 在第96页

VPCLASS 配置参数 在第196页

AUTO_CKPTS 配置参数

AUTO_CKPTS 配置参数允许服务器更频繁地触发检查点来避免事务阻塞。

onconfig.std 值

未设置。如果 AUTO_TUNE 配置参数设置为 1，则启用自动检查点。

值

0 = 关闭

1 = 开启

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

如果当前 onconfig 文件中未设置 AUTO_CKPTS 值，编辑 AUTO_TUNE 配置参数并重启数据库服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_TUNE 配置参数 在第73页

《*SinoDB* 管理员指南》*:* 检查点

《*SinoDB* 性能指南》*:* 自动检查点、*LRU* 调整以及 *AIO* 虚拟处理器调整

AUTO_LLOG 配置参数

使用 AUTO_LLOG 配置参数来在指定的数据库空间内自动添加逻辑日志，以提升性能。

onconfig.std 值

未在 onconfig.std 文件中设置。

默认值（如果在安装期间创建服务器）

```
AUTO_LLOG 1,llog,max_size
```

max_size 值取决于 AUTO_TUNE_SERVER_SIZE 配置参数的值。

值

0 = 默认值。禁用。不会自动添加逻辑日志来提升性能。

1,*dbspace_name*,*max_size*

- 1 = 启用。当需要提升性能时，自动添加逻辑日志。
- *dbspace_name* = 要在其中添加逻辑日志文件的数据库空间的名称。该数据库空间必须有操作系统的默认页大小。
- *max_size* = 可选。默认为 2048000 KB（2 GB）。所有逻辑日志文件的最大 KB 数，包含不存储在 *dbspace_name* 数据库空间的任何逻辑日志文件。当大小到达最大时，数据库服务器将不再添加逻辑日志文件来提高性能。如果未指定 *max_size*，则 AUTO_TUNE_SERVER_SIZE 配置参数的设置将影响大小的最大值。请参阅下面的用法部份。

分隔符

使用逗号分隔字段。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果在安装期间创建了服务器，则自动地启用 AUTO_LLOG 配置参数。为逻辑日志创建名为 llog 的数据库空间。安装程序根据 AUTO_TUNE_SERVER_SIZE 配置参数的值来设置初始大小和数据库空间的 *max_size* 选项的值。可通过重置 AUTO_LLOG 配置参数的值来更改 *max_size* 选项。

如果在安装期间未创建服务器，则您可启用 AUTO_LLOG 配置参数，以便在数据库服务器侦测到添加逻辑日志文件可提升性能时自动添加逻辑日志文件。为了获得理想性能，在与根数据库空间和物理日志不同的单独磁盘上选择数据库空间。

当启用了 AUTO_LLOG 配置参数时，一旦逻辑日志不足导致过高检查点百分比、阻塞检查点或长检查点时，数据库服务器会添加逻辑日志。

当逻辑日志文件大小达到最大时，不再添加逻辑日志文件来提升性能。然而，如果启用了 DYNAMIC_LOGS 配置参数，则会添加逻辑日志以防事务阻塞。DYNAMIC_LOGS 配置参数和 AUTO_LLOG 配置参数的设置不会相互作用。同样地，您可以继续手动地添加逻辑日志文件。

如果 *max_size* 字段的值大于指定数据库的大小，请确定存储池有可用的空间。

> 示例
>
> 以下设置使得能自动添加逻辑日志文件直到所有逻辑日志文件的大小达到 204800 KB，并将逻辑日志文件的数据库空间设置为 llog：
>
> ```
> AUTO_LLOG 1,llog,204800
> ```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_TUNE_SERVER_SIZE 配置参数 在第69页

《SinoDB 性能指南》: AUTO_LLOG 及其对日志记录的影响

AUTO_TUNE_SERVER_SIZE 配置参数

使用 AUTO_TUNE_SERVER_SIZE 配置参数根据预期的并发用户数来设置要分配的内存和存储空间的大小。

onconfig.std 值

未在 onconfig.std 文件中设置。

默认值

未设置。

值（如果安装期间创建服务器）

取决于在安装程序中指定的用户数。

值

SMALL = 1 - 100 用户

MEDIUM = 101 - 500 用户

LARGE = 501 - 1000 用户

XLARGE = 多于1000 用户

生效

如果在安装期间创建服务器。

在编辑 onconfig 文件并第一次重启数据库服务器之后。

用法

如果在安装期间创建服务器，请指定数据库服务器的预期用户数。AUTO_TUNE_SERVER_SIZE 配置参数设置成相应的大小，其会影响以下特性的大小：

- 缓冲池大小。
- 在服务器停止为了提升性能而自动地添加逻辑日志之前的逻辑日志文件大小的最大值。
- 在安装期间以下自动地创建的存储空间的初始大小：
  - 物理日志的可扩展物理日志空间
  - 逻辑日志的数据库空间
  - 数据库和表的数据库空间
  - 临时数据库空间
  - 智能大对象空间
  - 临时智能大对象空间

下表显示 AUTO_TUNE_SERVER_SIZE 配置参数值对大小的影响。

表 60: 对内存和存储空间分配的影响

值	缓冲池（BUFFERPOOL）大小的最大值	自动创建的存储空间的初始大小	逻辑日志文件（AUTO_LLOG）大小的最大值
SMALL	可用共享内存的 10%	50 MB	200 MB
MEDIUM	20%	100 MB	500 MB
LARGE	33%	200 MB	1 GB
XLARGE	50%	500 MB	2 GB

如果您在安装期间未创建服务器，或者在初始服务器之后首次更改 AUTO_TUNE_SERVER_SIZE 配置参数的值，则新值将仅影响以下特性的大小：

- 缓冲池的大小（如果 BUFFERPOOL 配置参数的设置包括 memory='auto' 选项。）
- 在服务器停止为了提升性能而自动添加逻辑日志之前的逻辑日志文件大小的最大值。

相关链接

BUFFERPOOL 配置参数 在第78页

AUTO_LLOG 配置参数 在第68页

AUTO_LRU_TUNING 配置参数

使用 AUTO_LRU_TUNING 配置参数来启用自动 LRU 调整，以自动地为页替换维持足够的干净页。

onconfig.std 值

未设置。AUTO_TUNE 配置参数设置为 1，则启动自动 LRU 调整。

值

0 = 关闭

1 = 开启

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

如果当前 onconfig 文件中未设置 AUTO_LRU_TUNING 值，编辑 AUTO_TUNE 配置参数并重启数据库服务器。

用法

自动 LRU 调整更改影响所有缓冲池，并且调整 BUFFERPOOL 配置参数中的 lru_min_dirty 和 lru_max_dirty 值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_TUNE 配置参数 在第73页

《*SinoDB* 性能指南》: 自动检查点、*LRU* 调整以及 *AIO* 虚拟处理器调整

BUFFERPOOL 配置参数 在第78页

AUTO_READAHEAD 配置参数

使用 AUTO_READAHEAD 配置参数来更改自动预读模式或禁用一个查询的自动预读操作。

onconfig.std 值

未设置。如果 AUTO_TUNE 配置参数设置为 1，则在标准模式下自动执行预读。

值

一个 0 - 2 的整数指定模式，可选地跟随著一个逗号和一个整数指定自动请求预读的页数。例如，1,4096 表示在标准模式下启动自动预读，每次预读 4096 页。

0 = 禁用自动预读请求。

1 = 在标准模式下启用自动预读请求。数据库服务器仅在查询等待 I/O 时自动处理预读请求。

2 = 在积极模式下启动自动预读请求。数据库服务器在查询开始时自动处理预读请请，并在查询期间持续进行。

number_of_pages = 4 - 4096，指示自动请求预读的页数。默认为 128 页。

分隔符

使用逗号分隔模式与页数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

如果当前 onconfig 文件中未设置 AUTO_READAHEAD 值，编辑 AUTO_TUNE 配置参数并重启数据库服务器。

用法

当数据库服务器侦测到查询正在遇到 I/O 时，自动预读操作通过发出异步页请求来帮助提高查询性能。异步页请求可以通过将查询处理与存磁盘检索数据并将其放入缓冲池所需的处理相叠加从而提高查询性能。

通常，默认值 1 适用于大多数生产环境。

虽然没有积极预读操作明显优于标准预读操作的特定情况，但积极预读可能略微有效些：

- 一些读取少量数据的扫描
- 在小扫描关闭预读和较大扫描打开预读之间切换的情况
- 对于仅查看少量行数的扫描，因为服务器立即执行预读操作而不是等待扫描遇到 I/O。

对于因为命中高速缓存数据而可能关闭和开启预读操作的扫描，积极预读操作不关闭预读操作。

仅在测试了两种设置并知道积极预读操作更有效的情况下，使用积极预读操作。如果不确定积极预读操作更有效的话，则不要使用。

可以使用 SQL 的 SET ENVIRONMENT 语句的 AUTO_READAHEAD 环境选项来启用或禁用会话的 AUTO_READAHEAD 配置参数的值。

预读设置的优先级如下：

1. 会话的 A SET ENVIRONMENT AUTO_READAHEAD 语句。
2. AUTO_READAHEAD 配置参数值为 1 或 2。
3. 如果 onconfig 文件中没有 AUTO_READAHEAD 配置参数的值，则服务器完成查询时在 128 个数据页上执行预读（等同于将 AUTO_READAHEAD 设置为 1）。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_TUNE 配置参数 在第73页

《*SinoDB* 性能指南》: 顺序扫描
《*SinoDB* 管理员指南》: 预读操作
《*SinoDB SQL* 指南: 语法》: *AUTO_READAHEAD* 会话环境选项

AUTO_REPREPARE 配置参数

使用 AUTO_REPREPARE 配置参数控制是否数据库服务器在 SPL 例程或已准备就绪对象引用的表的模式被更改之后，自动重新优化 SPL 例程和重新准备已准备就绪的对象。

onconfig.std 值

未设置。如果 AUTO_TUNE 配置参数设置为 1，则自动重新优化 SPL 例程和自动重新准备已准备就绪的对象。

值

0 = 在修改直接引用表或间接引用表的模式之后，禁止已准备就绪的对象的自动重新准备。也禁止在间接引用表的模式被修改之后 SPL 例程的自动重新优化。

1 = 启用自动重新准备。

3 = 在乐观模式下启用自动重新准备。

5 = 对更新统计启用自动重新准备。

7 = 对乐观模式和更新统计启用自动重新准备。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

如果当前 onconfig 文件中未设置 AUTO_REPREPARE 值，编辑 AUTO_TUNE 配置参数并重启数据库服务器。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

启用 AUTO_REPREPARE 配置参数，以减少在修改动态 SQL 语句或 SPL 例程中 DML 语句所引用表的模式之后必须明显地执行的重新准备的次数。

例如，某些修改表模式的 DDL 语句，诸如 CREATE INDEX、DROP INDEX、DROP COLUMN 和 RENAME COLUMN。如果这些 DDL 语句运行时禁用 AUTO_REPREPARE 配置参数，则用户可能会收到 -710 错误。您下次运行以下项目时，这些错误还会发生：

- 直接引用或间接引用被 DDL 语句修改的表的SPL 例程
- 引用被 DDL 语句修改的表的已准备就绪对象

乐观模式通过不检查在不到一秒之前成功执行的语句来提供更快的性能 。万一在这过渡期间表被修改了，则可能会发生一些 -710 错误。

如果您要避免数据库服务器使用较旧、次优的执行计划，请在更新统计上设置自动重新准备。

限制:

对于引用被 DDL 更改列数或列数据类型的表的已准备就绪语句或 SPL 例程，启用 AUTO_REPREPARE 可能对其没有影响。在表模式更改之后，通常必须重新发出 DESCRIBE 语句、PREPARE 语句（对已准备就绪对象）和 UPDATE STATISTICS FOR ROUTINE 语句（对例程相关联的游标）以优化引用这些表的 SPL 例程的执行计划。否则，数据库服务器可能发出 SQL 错误 -710。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
AUTO_TUNE 配置参数 在第73页
《*SinoDB SQL* 指南: 语法》: *IFX_AUTO_REPREPARE* 会话环境选项

《*SinoDB SQL* 指南: 语法》: *PREPARE* 语句
《*SinoDB SQL* 指南: 语法》: *SET ENVIRONMENT* 语句
《*SinoDB SQL* 指南: 语法》: *UPDATE STATISTICS* 语句

AUTO_STAT_MODE 配置参数

使用 AUTO_STAT_MODE 配置参数来启用或禁用仅选择性更新 UPDATE STATISTICS 操作中过时或丢失的数据分布，而非更新所有数据分布的统计。

onconfig.std 值

未设置。如果 AUTO_TUNE 配置参数设置为 1，则有选择地更新统计。

值

0 = 禁用有选择的 UPDATE STATISTICS 操作。

1 = 启用有选择的 UPDATE STATISTICS 操作。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

如果当前 onconfig 文件中未设置 AUTO_STAT_MODE 值，设置 AUTO_TUNE 配置参数。

用法

当 AUTO_STAT_MODE 配置参数或 AUTO_STAT_MODE 会话环境变量启用有选择地更新 UPDATE STATISTICS 操作中过时或丢失的数据分布的自动模式时，数据库服务器使用 STATCHANGE 配置参数值来识别需要更新的表或分段分布统计。

在 AUTO_STAT_MODE 配置参数和 AUTO_STAT_MODE 会话环境变量具有不同设置的会话中，于会话期间或直到 AUTO_STAT_MODE 会话环境变量重置之前，将以会话环境变量为优先。

相关链接

STATCHANGE 配置参数 在第179页
AUTO_TUNE 配置参数 在第73页
《*SinoDB SQL* 指南: 语法》: *CREATE TABLE* 语句的统计信息选项
《*SinoDB SQL* 指南: 语法》: *AUTO_STAT_MODE* 会话环境选项
STATCHANGE 配置参数 在第179页

AUTO_TUNE 配置参数

使用 AUTO_TUNE 配置参数来启用或禁用值不在 onconfig 文件中的所有自动调整配置参数。

onconfig.std 值

AUTO_TUNE 1

值

0 = 禁用

1 = 启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果某个自动调整配置参数未在当前 onconfig 文件中设置，那么数据库服务器使用 AUTO_TUNE 配置参数中指定的值作为该配置参数的值。

自动调整参数：

- AUTO_AIOVPS
- AUTO_CKPTS
- AUTO_LRU_TUNING
- AUTO_READAHEAD
- AUTO_REPREPARE
- AUTO_STAT_MODE

如果在当前 onconfig 文件中设置了自动调整配置参数，那么数据库服务器使用 onconfig 文件中的设置值。AUTO_TUNE 配置参数不会更改那个值。

您的 onconfig 文件位于 %INFORMIXDIR%\etc 或 $INFORMIXDIR/etc 目录。

示例

示例 1：假设某些自动调整配置参数未设置，但其他的有值：

```
AUTO_LRU_TUNING （值未设置）
AUTO_STAT_MODE （值未设置）
AUTO_LRU_CKPTS （值未设置）
AUTO_AIOVPS 0
AUTO_REPREPARE 1
AUTO_READAHEAD 0
```

如果 AUTO_TUNE 配置参数设置为 1，数据库服务器自动将未设置的自动调整配置参数设置为 1。前面已设置的值将保持一样。现在自动调整配置参数值如下：

```
AUTO_LRU_TUNING 1
AUTO_STAT_MODE 1
AUTO_CKPTS 1
AUTO_AIOVPS 0
AUTO_REPREPARE 1
AUTO_READAHEAD 0
```

示例 2：假设所有自动调整配置参数设置如下值：

```
AUTO_LRU_TUNING 1
AUTO_STAT_MODE 1
AUTO_LRU_CKPTS 1
AUTO_AIOVPS 0
AUTO_REPREPARE 1
AUTO_READAHEAD 0
```

此示例中，AUTO_TUNE 配置参数不会改变任何值。

示例 3：假设您从 onconfig 文件中移除了自动调整配置参数但现在要使用它们。您可将 AUTO_TUNE 设置为 1 来重新启用所有自动调整配置参数。

相关链接

AUTO_AIOVPS 配置参数 在第67页
AUTO_CKPTS 配置参数 在第67页
AUTO_LRU_TUNING 配置参数 在第70页
AUTO_REPREPARE 配置参数 在第72页

AUTO_STAT_MODE 配置参数 在第73页
AUTO_READAHEAD 配置参数 在第71页

AUTOLOCATE 配置参数

使用 AUTOLOCATE 配置参数来启用数据户、索引和表的自动定位，以及启用自动表分段。

onconfig.std 和默认值

AUTOLOCATE 0

值

0 = 禁用自动定位和分段。

1 - 32 = 启用自动定位和分段。数字表示初始分配给一个表的循环分段数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

使用 AUTOLOCATE 配置参数来控制是否数据库服务器控制新数据户、新索引和新表的位置，以及这些表的分段。如果 AUTOLOCATE 配置参数设置为正整数，数据库服务器执行以下任务：

- 将未指定位置的新数据库存储在最佳数据库空间而非存储在根数据库空间。默认情况下，除了专属于租户数据库的数据库空间外，所有数据库空间都是可用的。然而，您可控制可用数据库空间列表。
- 通过循环方式分段新表。分段数等于 AUTOLOCATE 配置参数的值。
- 随着表增长，添加更多表分段。

如果 AUTOLOCATE 配置参数设置为 0，新数据库将默认地创建在根数据库空间中。新表和新索引创建在与数据库同样的数据库空间中，且不分段。

自动定位不适用于租户数据库或租户数据库中的表、分段和索引。

通过在 CREATE DATABASE 语句中使用 IN 子句来指定数据库空间，可以覆盖数据库的自动定位。同样地，通过使用 IN 子句指定数据库空间来覆盖表的自动定位和分段，或通过在 CREATE TABLE 语句中使用 FRAGMENT BY 子句来覆盖分段策略。

当起用此配置参数时，您可以使用 admin() 或 task() 函数的 autolocate database 参数来：

- 管理自动定位和分段的数据库空间列表。可用数据库空间列表是在 sysautolocate 系统目录表中。
- 对于指定数据库禁用自动定位和分段。

可以使用 SQL 的 SET ENVIRONMENT 语句的 AUTOLOCATE 环境选项来启用或禁用会话的 AUTOLOCATE 配置参数值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
autolocate database 参数: 指定自动定位和分段的数据库空间 *(SQL* 管理 *API)* 在第591页
autolocate database add 参数: 添加数据库空间到数据库空间列表 *(SQL* 管理 *API)* 在第590页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 *(SQL* 管理 *API)* 在第593页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 *(SQL* 管理 *API)* 在第591页
autolocate database off 参数: 禁用数据库的自动分段 *(SQL* 管理 *API)* 在第592页
《*SinoDB SQL* 指南: 语法》: *AUTOLOCATE* 会话环境选项
《*SinoDB* 管理员指南》: 管理自动定位和分段化

BATCHEDREAD_INDEX 配置参数

使用 BATCHEDREAD_INDEX 配置参数可使优化器对索引执行轻扫描。这会减少读取缓冲区的次数，进而提高性能。

onconfig.std 值

BATCHEDREAD_INDEX 1

值

0 = 禁用索引的轻扫描。

1 = 启用索引的轻扫描。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

在 IFX_BATCHEDREAD_INDEX 配置参数和 IFX_BATCHEDREAD_INDEX 会话环境变量具有不同设置的会话中，于会话期间或直到 IFX_BATCHEDREAD_INDEX 会话环境变量重置之前，将以会话环境变量为优先。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

BATCHEDREAD_TABLE 配置参数

使用 BATCHEDREAD_TABLE 配置参数在压缩表、行大于一页的表以及具有 VARCHAR、LVARCHAR 和 NVARCHAR 数据的表上启用或禁用轻扫描。

onconfig.std 值

BATCHEDREAD_TABLE 1

值

0 = 在可变记录长度的表上禁用轻扫描

1 = 在可变记录长度的表上启用轻扫描

此处压缩表和行大于一页的表视为可变记录长度。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

除了压缩表、行大于一页的表以及可变记录长度的表，BATCHEDREAD_TABLE 设置对于查询优化器是否选择包含轻扫描的执行路径没有影响。

数据库服务器不在索引上和系统表上执行轻扫描，对于行包含具有以下任何存储属性的大对象的用户表，也不执行轻扫描：

- BLOB 空间
- 智能 BLOB 空间
- 分区 BLOB

可以使用 SET ENVIRONMENT 语句的 IFX_BATCHEDREAD_TABLE 环境选项来覆盖当前会话的 BATCHEDREAD_TABLE 配置参数的值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
《*SinoDB SQL* 指南: 语法》: *SET ENVIRONMENT* 语句
《*SinoDB* 性能指南》: 轻度扫描

BLOCKTIMEOUT 配置参数

使用 BLOCKTIMEOUT 配置参数来指定线程或数据库服务器挂起的秒数。超时之后，线程或数据库服务器将继续处理或失败。

onconfig.std 值

BLOCKTIMEOUT 3600

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

BTSCANNER 配置参数

使用 BTSCANNER 配置参数来设置 B 型树扫描程序。当从具有索引的表中删除行时，B 型树扫描程序可改善带日志记录的数据库的事务处理。B 型树扫描程序移除已删除的索引条目，并且重新平衡索引节点。B 型树扫描程序自动确定删除那些索引项目。

onconfig.std 值

BTSCANNER num=1,threshold=5000,rangesize=-1,alice=6,compression=default

值范围

请参阅用法部分。

分隔符

在每个字段之间使用逗号。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -C 命令来动态地重置 onconfig 文件中的该值时。

运行带有 onmode 和 C 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

默认情况下，BTSCANNER 配置参数启动一个索引清除程序线程、优先清除有超过 5000 个删除项目的索引、动态地调整索引清除模式，以及在一个适当级别上对适度增长和更改的索引进行索引页合并。

BTSCANNER 配置参数语法

```
BTSCANNER
[ num= { 1 threads } , ]
[ threshold=thresh_size , ]
[ rangesize=100 , ]
[ alice=alice_mode , ]
[ compression= { default  | low  | med  | high } ]
```

表 61: BTSCANNER 配置参数值选项

字段	值
num	*threads* 值是一个正整数，设置系统启动时启动的 B 型树扫描程序线程数。默认为 1。
threshold	*thresh_size* 值是优先清除的索引必须达到的被删除项的最小数。默认为 5000。
rangesize	指定是否允许小索引的叶扫描： • -1 = 关。所有索引清除使用 alice 模式。 • 100 = 通过叶扫描方式扫描小索引。
alice	*alice_mode* 值控制索引清除： • 0 = 关。 • 1 = 使用正好 8 字节的内存。 • 2 = 使用正好 16 字节的内存。 • 3 - 12 = 默认为 6。设置索引清除的初始内存量。接下来，B 型树扫描程序根据之前清除操作的效率来自动地调整模式。
compression	两个部分使用的索引页的合并级别： • low = 如果您预期索引随著频繁的分裂会迅速地增长，则使用这个设置值。 • med 或 default = 默认。如果索引是适度增长或更改，则使用这个设置值。 • high = 如果索引的 90% 或更多都是仅读，或没有太多更改，则使用这个设置值。

在清除高于阈值的所有索引之后，低于阈值的索引将加到待清除索引的优先列表中。频繁更新的系统应以 10 倍或 100 倍来增加这个值。

相关链接

onmode -C: 控制 B 型树扫描程序 在第332页

onmode 和 C 参数: 控制 B 型树扫描程序 (SQL 管理 API) 在第653页

《*SinoDB 性能指南*》: 配置 *B* 型树扫描程序信息来改进事务处理

BUFFERPOOL 配置参数

使用 BUFFERPOOL 配置参数来配置在共项内存中高速缓存的数据页数，和在检查点之间将那些页清空到磁盘的频率。对于许多系统而言，BUFFERPOOL 配置参数的默认值是适当的。然而，您可以更改这些值来调整您的系统性能。

onconfig.std 值

2 KB 默认页大小的操作系统：

```
BUFFERPOOL default,buffers=10000,lrus=8,lru_min_dirty=50.00,
lru_max_dirty=60.50
BUFFERPOOL size=2k,buffers=50000,lrus=8,lru_min_dirty=50,
lru_max_dirty=60
```

4 KB 默认页大小的操作系统：

```
BUFFERPOOL default,buffers=10000,lrus=8,lru_min_dirty=50.00,
```

```
lru_max_dirty=60.50
BUFFERPOOL size=4k,buffers=10000,lrus=8,lru_min_dirty=50,
lru_max_dirty=60
```

默认值（如果安装期间创建服务器）

```
BUFFERPOOL default,memory='auto'
BUFFERPOOL size=page_size,memory=memory_size
```

page_size 值为默认页大小。缓冲池的初始大小为 32 MB。由 memory 字段的值指定作为 auto 或 *memory_size* 的大小的最大值，取决于 AUTO_TUNE_SERVER_SIZE 配置参数的值。

值

请参阅用法部分。

分隔符

使用逗号分隔字段。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onparams -b 命令在 onconfig 文件中动态地添加条目时。

当通过运行 onspaces -c -d 命令添加具有不同页大小的数据库空间来动态地添加条目时。

通过运行带有 add bufferpool 参数的 SQL 管理 API task() 或 admin() 函数，在 onconfig 文件中动态地添加条目之后

用法

高速缓存的数据页保存在缓冲区。缓冲区在缓冲池中。用作存储空间的每个页大小需要一个缓冲池。当数据库服务器将新数据页移入共享内存中时，最近最少使用的数据页将从共享内存中移出。The BUFFERPOOL configuration parameter controls the size of the buffer pool and how frequently data pages are flushed to disk.BUFFERPOOL 配置参数控制缓冲池大小和数据页清空到磁盘的频率。

BUFFERPOOL 配置参数在 onconfig.std 文件或在安装期间创建服务器时生成的 onconfig 文件中有两个条目：

- 第一个条目为具有非默认页大小的数据库空间指定缓冲池的默认值。
- 第二个条目指定基于系统默认页大小的缓冲池的默认值。

包含 size 字段的 BUFFERPOOL 配置参数条目优先于包含 default 字段的 BUFFERPOOL 配置参数条目。

BUFFERPOOL 配置参数有两种格式：

- 如果要以 MB 或 GB 等内存单位来指定缓冲池的大小，请将 BUFFERPOOL 配置参数与 memory 字段一起使用。
- 如果要以页单位来指定缓冲池的大小或保留先前版本的设置，请将 BUFFERPOOL 配置参数与 buffers 字段一起使用。

您可使用任一格式使数据库服务器根据需要来扩展缓冲池的大小以提高性能。

限制： 不可在 onconfig 文件中同时有两种格式。onconfig 文件中 BUFFERPOOL 配置参数的所有条目必须有相同的格式，否则数据库服务器不会启动，并且显示以下错误：

```
ERROR: Cannot mix buffer arguments with memory arguments.  (BUFFERPOOL)
```

BUFFERPOOL 条目中的字段不区分大小写，并且可以按任何顺序罗列。

memory 字段语法

```
BUFFERPOOL
{ default  | size =  page_size k }
[ , lrus = number_lrus ]
[ , lru_min_dirty = min_percentage ]
```

[, lru_max_dirty = *max_percentage*]
[, extendable = { 0 | 1 [, cache_hit_ratio = *ratio*] }]
[, start_memory = { auto | *start_size* [{ kb | mb | gb }] }]
, memory = { auto | *max_size* [{ kb | mb | gb }] }
buffers 字段语法
BUFFERPOOL
{ default | size = *page_size* [k] }
[, lrus = *number_lrus*]
[, lru_min_dirty = *min_percentage*]
[, lru_max_dirty = *max_percentage*]
, buffers = *number_buffers*
[, extendable = { 0 | 1 <extendable options> }]
可扩展的选项
可扩展的选项
[, max_extends = *extends*]
[, next_buffers = *number_buffers*]
[, cache_hit_ratio = *ratio*]

表 62: BUFFERPOOL 配置参数值的选项

字段	值
buffers	默认为 1000。 *number_buffers* 值是一个大于等于 1000 的整数，指定共享内存缓冲区的最大数。允许的最大缓冲区数取决于操作系统、位大小和页大小： • UNIX™、32-bit、2 KB 页大小：1000 - 1843200 • UNIX™、32-bit、4 KB 页大小：1000 - 921600 • Windows™、32-bit：100 - 524288 • 64-bit：100 - (2^{31}-1)。有关 64-bit 平台的实际值，请参阅机器说明。例如，Solaris 平台上缓冲区的最大数目为 536,870,912。 buffers 字段值设置为每个用户至少四个缓冲区。如果您的系统处理超过 500 并发用户，则指定至少 2000 个缓冲区。 每个缓冲区是操作系统页的大小。因此，数据库需求的缓冲区数取决于物理内存量和应用程序使用的内存量。例如，如果 90% 的时间数据库服务器访问 15% 的应用数据，则分配能持有 15% 数据的缓冲区。增加缓冲区数可提高系统性能。缓冲区数对性能有显著影响并使用大量物理内存。 有关更多信息，请参阅《*SinoDB*® 性能指南》中的 *BUFFERPOOL 配置参数和内存利用率*。
cache_hit_ratio	默认为 90。 *ratio* 值是一个 0 - 100 的整数，表示阈值，当低于阈值时扩充缓冲池。当平均读取高速缓冲命中率低于 *ratio* 值持续将近五分钟时，数据库服务器扩展缓冲池。 仅当设置了 extendable=1，cache_hit_ratio 字段才有效。
extendable	如果设置了 memory 字段，则默认为 1。 如果设置了 buffers 字段，则默认为 0。 是否数据库服务器可以扩展缓冲池的大小： • 0 = 禁用。缓冲池不可增长。 • 1 = 启用。缓冲池可增长。

字段	值
lru_max_dirty	默认为 60.00。 *max_percentage* 值是一个 0 - 100.00 的十进制的数，设置 LRU 队列中已修改页的百分比，到达此百分比时将清除该队列。 如果启用了 AUTO_LRU_TUNING 配置参数，这个值将按需要自动更新。
lru_min_dirty	默认为 50.00。 *min_percentage* 值是一个 0 - 100.00 十进制的数，设设置 LRU 队列中已修改页的百分比，到达此百分比时将不再强制清除页。 在一些情况下，页清除程序可能在超过指定百分比时继续清除。 如果启用了 AUTO_LRU_TUNING 配置参数，这个值将按需要自动更新。
lrus	默认为 8。如果启用 MULTIPROCESSOR 配置参数，则默认大于 8 或 CPU VP 的数目。 *number_lrus* 值是正整数，指定缓冲池中 LRU（least recently used）队列数 值范围取决于操作系统的位大小： • 32-bit 平台：8 - 128 • 64-bit 平台：8 - 512 指定越多 LRU 队列，就有越多页清除程序并行工作。然而，lrus 字段的值设置的太高可能导致过度的页清除程序活动。 lrus 字段的值与 lru_min_dirty 和 lru_max_dirty 字段一起控制共享内存缓冲区清空到磁盘的频率。 有关更多信息，请参阅《*SinoDB*® 性能指南》中的 *BUFFERPOOL* 及其对页清除的影响。
max_extends	默认为 8。 *extends* 值表示数据库服务器可以扩展缓冲池的最大次数。*extends* 的值是从 0 到最大段数，取决于操作系统和位大小： • 32 bit = 16 • UNIX™ 64 bit = 24 • Windows™ 64 bit = 8 仅当设置了 buffers 和 extendable=1，max_extends 字段才有效。
memory	默认为 auto。 *max_size* 值表示最大的缓冲池数。*max_size* 的值范围是： • 一个表示 32 MB - 4 TB 的整数。可以指定 KB、MB 或 GB 的大小单位。如果未指定单位，默认单位为 KB。 • auto = 数据库服务器决定分配给缓冲池的共享内存的最大量。如果设置，则 AUTO_TUNE_SERVER_SIZE 配置参数值控制缓冲池大小的最大值。
next_buffers	默认为 1000。 *number_buffers* 值是一个大于等于 1000 的整数，指定数据库服务器用于扩展缓冲池的共享内存缓冲区数。*number_buffers* 的最大值受限于虚拟共享内存量。 每四次扩展，*number_buffers* 值翻倍。 仅当设置了 buffers 和 extendable=1，next_buffers 字段才有效。

字段	值
size	*page_size* 值指定缓冲区的页大小，以 KB 计。T页大小必须是 2 - 16 KB 且必须是默认页大小的倍数。例如，如果默认页大小为 2 KB，那么页大小可以为 2、4、6、8、10、12、14 或 16。If the default page size is 4 KB, the page size can be 4, 8, 12, or 16. 如果默认页大小为 4 KB，那么页大小可以是 4、8、12 或 16。默认值取决于系统默认页大小： • 2 KB 默认页大小：size=2k • 4 KB 默认页大小：size=4k k 为可选的。
start_memory	默认为 32 MB。 *start_size* 值为数据库服务器启动时缓冲池的初始大小： • 一个 32 MB 直到可用共享内存的最大量的整数。可指定大小单位为 KB、MB 或 GB。如果没有指定单位，默认单位为 KB。由于缓冲池初始大小必须是共享内存段大小的倍数，因此可能大于 *start_size* 的值。 • auto = 数据库服务器决定分配给缓冲池的共享内存的初始量。 如果未设置 start_memory 字段，缓冲池的初始大小等于 memory 字段的值。 仅当设置了 memory 字段，start_memory 字段才有效。

使用内存格式的缓冲池的大小

如果使用内存格式，默认情况下，缓冲池大小按需要增长。当平均高速缓存读取命中率低于阈值时，会向缓冲池添加共享内存段。您可设置缓冲池的初始大小和大小的最大值，或允许数据库服务器决定最佳大小。

如果 extendable 字段设置为 0，则缓冲池不会增长。如果设置了值，则该大小等于 start_memory 字段的值，否则等于 memory 字段的值。

当重启服务器时，缓冲池大小将重置为 start_memory 字段的值。

使用缓冲区格式的缓冲池的大小

如果使用 buffers 格式，默认情况下，缓冲池的大小不会增长。该大小等于 buffers 字段的值。

如果 extendable 设置为 1，当平均高速缓存读取命中率低于阈值时，会向缓冲池添加共享内存段。您必须在 buffers 字段中设置初始的缓冲区数。您可以选择地设置扩展缓冲池的缓冲区数、缓冲池可扩展的最大次数，以及高速缓存命中率。每四次扩展，添加到缓冲池的缓冲区数翻倍。

示例：添加具有 memory 字段的 BUFFERPOOL 条目

以下条目创建一个有 10 KB 页大小的缓冲池：

```
BUFFERPOOL size=10k,start_memory=auto,memory=4gb
```

该缓冲池可扩展到 4 GB。数据库服务器决定缓冲池的初始大小和向缓冲池扩展的大小。

示例：添加具有 buffers 字段的 BUFFERPOOL 条目

以下条目创建一个有 2 KB 页大小的缓冲池：

```
BUFFERPOOL size=2k,extendable=1,buffers=1000,next_buffers=2000,max_extends=8
```

该缓冲池可扩展八次。该缓冲池从 1000 个缓冲区开始。首三次扩展添加 2000 个缓冲区到缓冲池。第四次到第七次扩展添加 4000 个缓冲区。第八次扩展添加 8000 个缓冲区。

示例：通过添加具有不同页大小的数据库空间来添加 BUFFERPOOL 条目

当使用 onspaces 实用程序来添加具有不同页大小的数据库空间或使用 onparams 实用程序来添加缓冲池时，onconfig 文件中会添加一个 BUFFERPOOL 配置参数条目。以下示例显示了第三个条目：

```
BUFFERPOOL default,buffers=10000,lrus=8,lru_min_dirty=50.00,lru_max_dirty=60.50
BUFFERPOOL size=2k,buffers=10000,lrus=8,lru_min_dirty=50,lru_max_dirty=60
BUFFERPOOL size=6k
```

当您创建具有非默认页大小的数据库空间时，如果现有的 BUFFERPOOL 条目存在，数据库服务器讲将使用现有的 BUFFERPOOL 条目的页大小。否则使用 BUFFERPOOL 默认行的值。

相关链接

onparams -b: 添加缓冲池 在第354页

add bufferpool 参数*:* 添加缓冲池 *(SQL* 管理 *API)* 在第583页

AUTO_LRU_TUNING 配置参数 在第70页

AUTO_TUNE_SERVER_SIZE 配置参数 在第69页

《*SinoDB* 性能指南》*: The BUFFERPOOL* 配置参数和内存利用率

《*SinoDB* 性能指南》*: BUFFERPOOL* 及其对页清除的影响

《*SinoDB* 管理员指南》*:* 共享内存的缓冲池部分

《*SinoDB* 管理员指南》*: FIFO/LRU* 队列

onstat -g buf 命令*:* 显示缓冲池的概要文件信息 在第429页

CHECKALLDOMAINSFORUSER 配置参数

使用 the CHECKALLDOMAINSFORUSER 配置参数来检查所有用户的所有域。

onconfig.std 值

不在 onconfig.std 文件中

值

0 = 禁用

1 = 启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 管理员指南》*: Windows* 网络域

CKPTINTVL 配置参数

使用 CKPTINTVL 配置参数来指定数据库服务器检查是否需要检查点的频率（以秒计）。当发生检查点时，共享内存缓冲池中的所有页写入磁盘。

onconfig.std 值

CKPTINTVL 300

值

任何大于或等于 0 的值

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

RTO_SERVER_RESTART 和 CKPTINTVL 配置参数是互斥的。如果启用了 RTO_SERVER_RESTART 配置参数，则会触发检查点并忽略 CKPTINTVL 值。否则，使用 CKPTINTVL 值来触发检查点。

如果 CKPTINTVL 配置参数设置的时间间隔太短，则系统将花费过多时间执行检查点，并且其他工作的性能会受影响。如果 CKPTINTVL 配置参数设置的时间间隔太长，则快速恢复可能需要很长时间。

实际上，30 秒是数据库服务器最小的检查间隔。如果指定检查点间隔为 0，则数据库服务器不检查检查点间隔是否已过。然而，数据库服务器仍执行检查点。在其他状况下，比如物理日志满至 75%，也会导致数据库服务器执行检查点。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

RTO_SERVER_RESTART 配置参数 在第153页

《*SinoDB* 管理员指南》: 检查点

《*SinoDB* 性能指南》

CLEANERS 配置参数

使用 CLEANERS 配置参数来指定数据库服务器操作期间可用的页清除程序线程数。在默认情况下，数据库服务器总是运行一个页清除程序线程。通用指导原则是每个磁盘驱动一个页清除程序。指定的值不影响共享内存大小。

基于服务器工作负载，服务器自动地尝试优化 AIO VP 和页清除程序线程，并在需要时向上调整 AIO VP 和页清除程序线程数。可使用环境变量 IFX_NO_AIOVP_TUNING 或 onmode -wm 实用程序选项来禁用 AIO VP 和页清除程序线程的自动调整。

onconfig.std 值

CLEANERS 8

值

1 - 128

单位

页清除程序线程数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -F 命令: 显示计数 在第420页

《*SinoDB* 管理员指南》: 将数据清空到磁盘

CLUSTER_TXN_SCOPE 配置参数

设置 CLUSTER_TXN_SCOPE 配置参数来配置高可用性集群，以便在客户端会话发出提交时，服务器将阻塞该会话直到其事务应用于辅助服务器上或跨集群中。

onconfig.std 值

CLUSTER_TXN_SCOPE SERVER

值

- SESSION = 当客户端会话发出提交时，数据库服务器会阻塞该会话直到该事务提交的影响返回该会话为止。在控制返回到该会话之后，同一数据库服务器上或集群中其他数据库服务器上的其他会话可能不知道该事务提交与该事务的结果。
- SERVER（默认行为）= 当客户端会话发出提交时，数据库服务器会阻塞该会话直到事务应用于该客户端会话发出提交的那个数据库服务器。在那个数据库服务器上的其他会话会知道该事务提交与其结果。集群中其他数据库服务器上的会话可能不知道该事务的提交与其结果。此为高可用性集群的默认行为。
- CLUSTER = 当客户端会话发出提交时，数据库服务器会阻塞该会话直到该事务应用于高可用性集群中所有数据库服务器，除了使用 DELAY_APPLY 或 STOP_APPLY 的 RS 辅助服务器以外。高可用性集群中的任何数据库服务器上（除了使用 DELAY_APPLY 或 STOP_APPLY 的 RS 辅助服务器以外）的其他会话都会知道该事务提交与其结果。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

在运行带有 -wf CLUSTER_TXN_SCOPE=*value* 或 -wm CLUSTER_TXN_SCOPE=*value* 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

设置 CLUSTER_TXN_SCOPE 配置参数来控制从高可用性集群到客户端应用程序的事务提交返回。集群事务协调可延迟事务提交到客户端应用程序的返回，直到该事务应用于一个辅助服务器或高可用性集群中所有辅助服务器。此过程防止由于异步日志处理导致的操作失败，并确保多步骤进程的步骤按照顺序发生。

集群事务协调不能应用于 DELAY_APPLY 或 STOP_APPLY 配置参数值不是 0 的 RS 辅助服务器。在客户端应用程序接收到提交之前事务不需要应用在 RS 辅助服务器上。

CLUSTER_TXN_SCOPE 影响只读辅助服务器和可更新辅助服务器上的会话。

> 示例 1：高可用性集群服务器间的事务协调
>
> 此示例中，客户端应用程序启动一个两步骤进程。客户端应用程序在主数据库服务器上插入数据，然后在一个 HDR 辅助服务器上启动数据处理。
>
> 在日志从主服务器应用到 HDR 辅助服务器之前，如果在 HDR 辅助服务器上尝试对插入的数据执行 SELECT，则操作会失败。要防止这个失败，将主服务器的 CLUSTER_TXN_SCOPE 配置参数设置为 CLUSTER，以便客户端应用程序不会收到提交，并且在数据插入也应用于 HDR 辅助服务器之前不会启动数据处理。

> 示例 2：数据库服务器上的事务协调
>
> 此示例中，您有一个分成数个处理阶段的客户端应用程序。每个处理阶段使用不同的 SQL 会话来连接到数据库服务器。应用程序更新数据，然后应用程序的另一部分在不同 SQL 会话中处理更新的数据。
>
> 如果 CLUSTER_TXN_SCOPE 设置为 SESSION，则处理更新数据的应用程序部份可能不知道更新的结果并可能发生失败。要防止这个失败，将数据库服务器的 CLUSTER_TXN_SCOPE 配置参数设置为 SERVER，以便客户端应用程序不会收到提交，并且在数据库服务器的更新完成之前不会启动数据处理。

相关链接

DELAY_APPLY 配置参数 在第95页
STOP_APPLY 配置参数 在第182页
《*SinoDB SQL* 指南: 语法》: *SET ENVIRONMENT* 语句

《SinoDB SQL 指南: 语法》: CLUSTER_TXN_SCOPE 会话环境选项
《SinoDB 管理员指南》: 集群事务协调

CONSOLE 配置参数

使用 CONSOLE 配置参数来指定控制台消息文件的路径和名称。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/tmp/online.con

在 Windows™ 上：online.con

值

pathname = online.con 文件的完整路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

CONVERSION_GUARD 配置参数

使用 CONVERSION_GUARD 配置参数来指定升级过程中发生错误时 SinoDB® 是停止还是继续升级到新版本。

onconfig.std 值

CONVERSION_GUARD 2

值

0 = 禁用。

1 = 启用恢复点作为升级过程的一部分，并且在发生与捕获恢复点数据相关的错误时停止升级。

2 = 启用恢复点作为升级过程的一部分，并且即使发生与捕获恢复点数据相关的错误仍继续升级。

单位

整数

生效

当数据库服务器重启时。

用法

默认情况下：

- 开启 CONVERSION_GUARD 配置参数（设置为 2）。如果升级到新版本失败时，您可以使用 onrestorept 实用程序来恢复您的数据。
- 服务器将恢复点数据存储在 $INFORMIXDIR/tmp 目录中。

如果 CONVERSION_GUARD 配置参数设置为 1 或 2，当升级到新版本失败时，您可以使用 onrestorept 实用程序来恢复您的数据。

如果 CONVERSION_GUARD 配置参数设置为 2，当转换防护操作失败（例如服务器没有足够空间来存储恢复点数据）且升级失败时，您无法使用 onrestorept 实用程序来恢复您的数据。

在启动服务器初始话升级到新版本之前，您可以更改 CONVERSION_GUARD 配置参数的值或 RESTORE_POINT_DIR 中指定的目录。在升级期间，您不可以更改 CONVERSION_GUARD 或 RESTORE_POINT_DIR 的值。

相关链接

RESTORE_POINT_DIR 配置参数 在第150页
《SinoDB 迁移指南》: onrestorept 实用程序

DATASKIP 配置参数

使用 DATASKIP 配置参数来控制在事务处理期间数据库服务器是否跳过不可用的数据库空间。

onconfig.std 值

未设置。不跳过任何数据库空间。

值

请参阅用法部分。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onspaces -f 命令来动态地重置 onconfig 文件中的该值时。

在运行带有 set dataskip 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

在查询处理期间，无论何时数据库服务器跳过数据库空间时，就会返回警告。

请谨慎启用 DATASKIP 配置参数，因为结果总是存疑的。仅在以下状况启用该参数：

- 您可以接受受损的事务完整性。
- 您可以确定事务完整性未受损，这是困难且耗时的。

DATASKIP 配置参数语法

DATASKIP { ALL | OFF | ON *dbspace_name* }

表 63: DATASKIP 配置参数值的选项

字段	描述
ALL	跳过所有不可用的分段。
OFF	处理所有分段，包括不可用分段。
ON	*dbspace_name* 指定一个或多个要跳过的数据库空间，以逗号分隔。

应用程序可以使用 SQL 语句 SET DATASKIP 来覆盖 DATASKIP 配置参数的值。

对于 SinoDB® ESQL/C，先前保留的 SQLCA 警告标志 sqlwarn.sqlwarn7 设置为 W。

相关链接

onspaces -f: 指定 DATASKIP 参数 在第379页

onstat -f 命令: 打印受 dataskip 影响的数据库空间信息 在第419页

set dataskip 参数: 启动或停止跳过数据库空间 (SQL 管理 API) 在第680页

《SinoDB 性能指南》: DATASKIP 如何影响表 I/O

《SinoDB SQL 指南: 语法》: SET DATASKIP 语句

DBCREATE_PERMISSION 配置参数

使用 DBCREATE_PERMISSION 配置参数来限制所指定用户创建数据库的权限。

用户 informix 总有创建数据库的权限。要限制用户 informix 创建数据库的能力，将 DBCREATE_PERMISSION 配置参数设置为 informix。

onconfig.std 值

在 UNIX™ 上：未设置。任何用户都可创建数据库。

在 Windows™ 上：#DBCREATE_PERMISSION informix

默认值

任何用户都可创建数据库。

单位

用户名

分隔符

逗号。还可在 onconfig 文件中包含 DBCREATE_PERMISSION 配置参数的多个副本，以授权多个用户创建数据库。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

DBCREATE_PERMISSION 配置参数不提供创建租户数据库的权限。用户必须有 TENANT 权限才可创建租户数据库。通过运行带有 grant admin 参数的 admin() 或 task() SQL 管理 API 函数授于 TENANT 权限。

相关链接

grant admin 参数: 授予运行 *SQL* 管理 *API* 命令的权限 在第630页

DB_LIBRARY_PATH 配置参数

使用 DB_LIBRARY_PATH 配置参数来指定以逗号分隔的有效目录前缀位置列表，供数据库服务器从中加载外部模块，比如 DataBlade® 模块。还可在该列表中包括服务器环境变量，比如 $INFORMIXDIR。

您必须完全按照向数据库服务器注册的路径来指定外部模块的路径。相对路径或包含双句号（..）的路径都是无效的。无法加载文件系统中未通过此参数指定的外部模块。在加载 C 语言模块之前扫描此列表。

如果设置了此配置参数，还必须包含字符串 $INFORMIXDIR/extend 作为其值的一部分。如果 DB_LIBRARY_PATH 中未包含字符串 $INFORMIXDIR/extend，则不加载内置扩展、DataBlade® 模块和 BladeManager 实用程序。

onconfig.std 值

未设置

如不存在

数据库服务器可从任何位置加载外部模块

值

路径名列表（最多 512 个字节）

分隔符

逗号

生效

在编辑 onconfig 文件并重启数据库服务器之后。

DBSERVERALIASES 配置参数

使用 DBSERVERALIASES 配置参数来指定数据库服务器的别名或唯一别名列表。DBSERVERALIASES 配置参数定义的每个别名可以在不同连接中使用，由 sqlhosts 信息中的条目指定。

onconfig.std 值

未设置。没有定义任何别名。

值

1 至 32 个以逗号分隔的别名。每个别名可以选择地后跟一个减号和一个 1 - 50 的整数指定用于 onimcsoc 或 onsoctcp 协议的多监听器线程数。例如，以下两个别名，每个有四个监听器线程：alias_a-4,alias_b-4。其他协议忽略监听器线程数。

一个别名的最大长度为 128 字节。超过 32 的额外别名将被忽略。DBSERVERALIASES 条目的最大长度为 512 字节。可以在 onconfig 文件中包括多个 DBSERVERALIASES 配置参数行。

别名必须以字母开头并可包括任何可打印字符，除了下列：

- 大写字符
- 字段分隔符（空格或制表符）
- 换行符
- 注释符（#）
- 连字符或减号（ = ASCII 45）
- @ 字符
- 空格

分隔符

使用逗号分隔条目。不包括空格。

生效

在编辑 onconfig 文件并重启数据库服务器，并更新每个数据库服务器的 sqlhosts 信息之后。

用法

如果 SinoDB® 支持一种以上的通信协议（例如，IPC 机制和 TCP 网络协议），，则您必须使用 sqlhosts 信息中的条目来描述与数据库服务器的每一有效连接。例如，假设您的服务器具有由 DBSERVERNAME 配置参数设置定义的名称 sanfrancisco，并且您为不同连接设置 了 menlo 的 DBSERVERALIASES 值。您必须在 sqlhosts 信息中为 sanfrancisco 和 menlo 服务器指定信息。同样地，如果数据库服务器需要支持标准 SinoDB® 协议和分布式关系数据库体系结构（DRDA®）协议，则给 DRDA® 数据库服务器指定一个别名，并在 sqlhosts 文件中为该别名添加一个条目。

数据库服务器为 DBSERVERALIASES 配置参数中罗列的每一个别名启动一个附加的监听器线程。如果有许多客户端应用程序连接到该数据库服务器，则可以在数个监听器线程间分布连接请求与加速连接时间。要利用替代的连接，请将某些客户端应用程序编程为连接到数据库服务器别名而非数据库服务器名。

使用共享内存连接的高可用性集群服务器必须也有服务器对服务器通信的 TCP 连接别名。如果高可用性集群服务器的 DBSERVERNAME 与一个共享内存 sqlhosts 文件条目相关联，则您必须通过设置 DBSERVERALIASES 值和将 HA_ALIAS 配置参数设置为 DBSERVERALIASES 值来为服务器创建 TCP 别名，然后为该别名创建一个 TCP sqlhost 文件条目。

相关链接

DBSERVERNAME 配置参数 在第90页
《*SinoDB* 管理员指南》: 连接相关的配置参数
《*SinoDB* 管理员指南》: 多种连接类型
《*SinoDB* 管理员指南》: 添加侦听线程
DBSERVERNAME 配置参数 在第90页
HA_ALIAS 配置参数 在第115页
NETTYPE 配置参数 在第135页
NUMFDSERVERS 配置参数 在第138页
onmode -d: 设置数据复制类型 在第334页
onmode -d: 设置高可用性服务器的特性 在第335页
onmode -d 命令: 使用数据复制来复制索引 在第336页

DBSERVERNAME 配置参数

使用 DBSERVERNAME 配置参数来指定要与数据库服务器关联的唯一名称。在安装数据库服务器时指定此配置参数。

onconfig.std 值

未设置。未定义数据库服务器名。

如果未出现

在 UNIX™ 上：*hostname*

在 Windows™ 上：ol_*hostname*

hostname 变量是主机计算机名。

值

数据库服务器名最长为 128 字节。数据库服务器名可选择地后跟一个减号和一个 1 - 50 的整数来指定用于 onimcsoc 或 onsoctcp 协议的多监听器线程数。默认监听器线程数为 1 。例如，以下数据库服务器名有四个监听器线程：ifxserver-4。其他协议忽略监听器线程数。

数据库服务器名必须以字母开头，并且包括任何可打印字符，除了下列之外：

- 大写字符
- 字段分隔符（空格或制表符）
- 换行符
- 注释符（#）
- 连字符或减号（ = ASCII 45）
- @ 字符
- 空格

生效

在编辑 onconfig 文件并重启数据库服务器，并更新每个数据库服务器的 sqlhosts 文件或注册之后。此外，所有用户的 INFORMIXSERVER 环境变量可能需要更改。

用法

数据库服务器名与 sqlhosts 文件或注册中指定的一个通信协议相关联。如果数据库服务器使用多个通信协议，则使用 DBSERVERALIASES 配置参数来定义每个通信协议关连的数据库服务器名。

客户端应用程序使用 INFORMIXSERVER 环境变量和 SQL 语句（比如与数据库服务器建立连接的 CONNECT 和 DATABASE）中的数据库服务器名。

> 重要： 为避免与同一计算机或节点上 SinoDB® 数据库服务器的其他实例发生冲突，应使用 DBSERVERNAME 配置参数来明确地指定数据库服务器名。

使用共享内存连接的高可用性集群服务器必须也有服务器对服务器通信的 TCP 连接别名。如果高可用性集群服务器的 DBSERVERNAME 与一个共享内存 sqlhosts 文件条目相关联，则您必须通过设置 DBSERVERALIASES 值和将 HA_ALIAS 配置参数设置为 DBSERVERALIASES 值来为服务器创建 TCP 别名，然后为该别名创建一个 TCP sqlhost 文件条目。

相关链接

DBSERVERALIASES 配置参数 在第88页

DBSERVERALIASES 配置参数 在第88页

《*SinoDB* 管理员指南》：配置参数中设置的连接信息

《*SinoDB* 管理员指南》：多种连接类型

《*SinoDB* 管理员指南》：添加侦听线程

《*SinoDB SQL* 指南: 参考》：*INFORMIXSERVER* 环境变量

HA_ALIAS 配置参数 在第115页

NETTYPE 配置参数 在第135页
NUMFDSERVERS 配置参数 在第138页
onmode -d: 设置数据复制类型 在第334页
onmode -d: 设置高可用性服务器的特性 在第335页
onmode -d 命令: 使用数据复制来复制索引 在第336页

DBSPACETEMP 配置参数

使用 DBSPACETEMP 配置参数来指定数据库服务器用于全局管理临时表存储的数据库空间列表。

DBSPACETEMP 通过使数据库服务器能够跨多个磁盘有效地分布临时表的 I/O 来提高性能。数据库服务器还在备份期间使用临时数据库空间来存储在备份发生时被覆盖的数据的前映像。

onconfig.std 值

未设置。临时表存储在根数据库空间。

分隔符

逗号或冒号（无空白）

值

一个或多个数据库空间名。数据库空间可以是标准数据库空间、临时数据库空间，或两者。使用冒号或逗号来分隔数据库空间名。此列表长度不可超过 254 字节。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

DBSPACETEMP 可以包含具有非默认页大小的数据库空间，但在 DBSPACETEMP 列表中的所有数据库空间必须具有相同的页大小。

如果客户端应用程序需要指定用于临时表位置的替代数据库空间列表，客户端可以使用 DBSPACETEMP 环境变量来罗列。仅当您使用 UPDATE STATISTICS 的 HIGH 选项时，数据库服务器才使用 DBSPACETEMP 环境变量指定的存储位置。

如果在 DBSPACETEMP 配置参数或环境变量中罗列了标准和临时数据库空间，则施行以下规则：

- 如果存在足够的空间，则会在临时数据库空间中创建排序、备份、隐式和非日志记录显示临时表。
- 不使用 WITH NO LOG 选项创建的显示临时表会创建在标准（而非临时的）数据库空间中

当使用 onspaces 实用程序创建临时数据库空间时，在您设置 DBSPACETEMP 配置参数或环境变量并重启服务器之前数据库服务器不会使用新创建的临时数据库空间。

DBSPACETEMP 环境变量立即生效并覆盖 DBSPACETEMP 配置参数。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
onspaces -c -d: 创建数据库空间 在第367页
onstat -d 命令: 显示块信息 在第414页
《*SinoDB* 管理员指南》: 临时表
《*SinoDB* 性能指南》: 为临时表和排序文件配置数据库空间
《*SinoDB SQL* 指南: 参考》: *DBSPACETEMP* 环境变量
《*SinoDB SQL* 指南: 参考》: *PSORT_DBTEMP* 环境变量

使用散列联接溢出 (Hash Join Overflow) 和 DBSPACETEMP

如果您未设置 DBSPACETEMP 环境片量或 DBSPACETEMP 配置参数，则 SinoDB® 使用操作系统目录或文件来指示某些数据库操作导致的溢出。

您可以下列方式来指定操作系统目录或文件：

- 带有 GROUP BY 子句的 SELECT 语句
- 带有 ORDER BY 子句的 SELECT 语句
- 散列联接操作
- 嵌套循环联接操作
- 索引建构

排序溢出文件的位置

下表列出可用于指定排序溢出文件位置的环境变量和 ONCONFIG 配置参数。

表 64: 排序溢出文件的位置

变量或参数	排序溢出文件的位置
PSORT_DBTEMP 环境变量	在环境变量中指定的位置
DBSPACETEMP 环境变量	在环境变量中指定的位置
ONCONFIG 文件中的 DBSPACETEMP 配置参数	ONCONFIG 文件中 DBSPACETEMP 配置参数指定的数据库空间

如指定了一个以上的变量或参数，SinoDB® 决定排序溢出文件位置的优先顺序为：

1. PSORT_DBTEMP 环境变量
2. DBSPACETEMP 环境变量
3. DBSPACETEMP ONCONFIG 变量
4. DUMPDIR
5. $INFORMIXDIR/tmp

如果未设置环境变量或配置参数，则排序溢出文件将置于 $INFORMIXDIR/tmp 目录中且临时表将置于根数据库空间。

DD_HASHMAX 配置参数

使用 DD_HASHMAX 配置参数来指定在数据字典高速缓存中每个散列存储区中表的最大数。

散列存储区 是存储的单位（通常是一页），其地址是通过散列函数计算而得。一个散列存储区包含数个记录。

例如，如果 DD_HASHMAX 配置参数设置为 10 且 DD_HASHSIZE 配置参数设置为 59，那么在数据字典高速缓存中可以存储大约 590 个表的信息，并且每个散列存储区可有最多 10 个表。

使用文字编辑器来修改配置文件。

onconfig.std 值

DD_HASHMAX 10

值

正整数

单位

一个散列存储区中的最大表数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

DD_HASHSIZE 配置参数 在第93页

《*SinoDB* 性能指南》: 配置对内存利用率的影响

DD_HASHSIZE 配置参数

使用 DD_HASHSIZE 配置参数来指定散列存储区的数目或数据字典高速缓存中的列表的数目。

使用文本编辑器来修改配置文件。

onconfig.std 值

DD_HASHSIZE 31

值

任何正整数；建议为质数。

单位

散列存储区或列表的数目

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

DD_HASHMAX 配置参数 在第92页

《*SinoDB* 性能指南》: 配置对内存利用率的影响

DEADLOCK_TIMEOUT 配置参数

使用 DEADLOCK_TIMEOUT 配置参数来指定数据库服务器线程等待获取锁的最大等待秒数。

此参数仅用于涉及远程数据库服务器的分布式查询。对于非分布式查询，请不要使用此参数。

onconfig.std 值

DEADLOCK_TIMEOUT 60

值

正整数

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果分布式事务被迫等待超过 DEADLOCK_TIMEOUT 配置参数所指定的秒数，那么该事务所属线程会假定存在多服务器死锁。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -p 命令: 显示概要文件计数 在第550页

《*SinoDB* 管理员指南》: 多阶段提交协议

《*SinoDB* 管理员指南》: 两阶段提交中使用的配置参数

DEF_TABLE_LOCKMODE 配置参数

使用 DEF_TABLE_LOCKMODE 配置参数来指定新表在页级或行级的锁模式。

onconfig.std 值

PAGE

值

PAGE = 为新表设置页的锁模式

ROW = 为新表设置行的锁模式

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

优先规则

通过在 CREATE TABLE 或 ALTER TABLE 语句中包含 LOCK MODE 子句，可以取代特定表的锁有其他锁模式设置。

客户端设置的 IFX_DEF_TABLE_LOCKMODE 环境变量优先于服务器上的该变量和 DEF_TABLE_LOCKMODE 配置参数。

服务器设置的 IFX_DEF_TABLE_LOCKMODE 环境变量优先于 DEF_TABLE_LOCKMODE 配置参数。

用法

如果 DEF_TABLE_LOCKMODE 配置参数设置为 ROW，则连接到日志记录或非日志记录数据库的所有会话创建的每个新表都将设置为行的锁模式。此参数不会影响现有表的锁模式。

如果 DEF_TABLE_LOCKMODE 配置参数设置为 PAGE，则USELASTCOMMITTED 配置参数和 SET ISOLATION 语句的 COMMITTED READ LAST COMMITTED 选项无法启用对未提交事务持有互斥锁的表中最近提交数据的访问，除非创建或修改这些表时明显地以 ROW 作为锁定粒度。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 参考》: *IFX_DEF_TABLE_LOCKMODE* 环境变量

USELASTCOMMITTED 配置参数 在第191页

《*SinoDB SQL* 指南: 参考》: *IFX_DEF_TABLE_LOCKMODE* 环境变量

《*SinoDB* 性能指南》: 配置锁模式

《*SinoDB SQL* 指南: 语法》: 优先权和缺省行为

DEFAULTESCCHAR 配置参数

DEFAULTESCCHAR 配置参数指定用于 LIKE 和 MATCHES 的默认转义字符。

onconfig.std 值

DEFAULTESCCHAR 反斜线字符（ \）。

如不存在

如果在 onconfig 文件中未设置值，则使用反斜线字符（ \）。

值

\ = 使用反斜线字符作为转义字符。

NONE = 无默认转义字符。

character = 使用任何一個字符作為轉義字符。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

在会话中，可以通过使用带有您想要使用的转义字符的 SET ENVIRONMENT DEFAULTESCCHAR 语句来取代默认值。例如：

```
SET ENVIRONMENT DEFAULTESCCHAR '\'
```

相关链接

《SinoDB SQL 指南: 语法》: DEFAULTESCCHAR 会话环境选项

DELAY_APPLY 配置参数

使用 DELAY_APPLY 配置参数来配置 RS 辅助服务器应用日志之前等待的时间。

onconfig.std 值

DELAY_APPLY 0

默认值

0

值

0 = 应用日志

整数后跟著一个时间单位：例如 1H 设置延迟一小时。

number：1-999 = 等待的天数、分钟数、时数或秒数。

time_unit：D、H、M 或 S，此处 *D* = 天，*H* = 小时，*M* = 分，和 *S* = 秒。这些值不区分大小写。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

延迟日志文件的应用，使您可以通过从 RS 辅助服务器恢复数据来快速恢复错误的数据库修改。当设置了 DELAY_APPLY 值时，您也必须设置 LOG_STAGING_DIR。如果设置了 DELAY_APPLY 但未将 LOG_STAGING_DIR 指向一个正确且安全的目录，则无法初始化数据库服务器。

您必须通过设置 LOG_STAGING_DIR 配置参数来为日志文件指定一个正确且安全的位置。在 RS 辅助服务器上处理完最后一个检查点之后，将会清除此暂存目录中的日志。

要查看有关发送到为 RS 辅助服务器设置的日志暂存目录的数据信息，请在 RS 辅助服务器上运行 onstat -g rss verbose 命令。

如果对暂存文件的写入失败，RS 辅助服务器会发出事件警报 40007。

如果远程独立辅助（RSS）服务器的 DELAY_APPLY 配置参数为一个非 0 的值，则该服务器不能使用集群事务协调。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

STOP_APPLY 配置参数 在第182页

CLUSTER_TXN_SCOPE 配置参数 在第84页

《SinoDB SQL 指南: 语法》: CLUSTER_TXN_SCOPE 会话环境选项

LOG_STAGING_DIR 配置参数 在第125页
onstat -g cluster 命令: 打印高可用性集群信息 在第440页
STOP_APPLY 配置参数 在第182页
LOG_STAGING_DIR 配置参数 在第125页
《*SinoDB* 管理员指南》: 延迟应用日志记录

DIRECT_IO 配置参数 (UNIX™)

使用 DIRECT_IO 配置参数来控制用于数据库空间块的熟文件的直接 I/O。

此参数在 UNIX™ 平台上启用直接 I/O（绕过文件系统缓存）或 在AIX® 操作系统上启用并发 I/O（绕过文件系统缓存和不必要的串行写入）

onconfig.std 值

DIRECT_IO 0

值

0 = 不使用直接 I/O 也不使用并发 I/O
1 = 如可用，则使用绕过文件系统缓冲的直接 I/O
2 = 在 AIX® 操作系统上启用并发 I/O（并发 I/O 选项包含了直接 I/O 和并发 I/O。）

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

直接 I/O 只能用于其文件系统支持页大小的直接 I/O 的数据库空间块。

通过使用直接 I/O，您可能可以减少 AIO 虚拟处理器数。

当启用了直接 I/O，如果文件系统支持 KIAO（内核异步 I/O）则会使用 KIAO。然而，如果设置了环境变量 KAIOOFF，则不会使用 KAIO。当直接 I/O 和 KAIO 都使用时，可减少 AIO 虚拟处理器的数目。如果使用直接 I/O 但未使用 KAIO，则不应减少 AIO 虚拟处理器的数目。

对用于临时数据库空间块的熟文件，SinoDB® 不会只用直接 I/O 或并发。

在 AIX® 上，如果 SinoDB® 对块使用并发 I/O，则另一个程序（例如联机外部备份程序）也必须使用并发 I/O。否则，文件打开操作将失败。

如果 SinoDB® 对块使用直接 I/O，并且另一个程序尝试不使用直接 I/O 打开块文件，则打开操作通常成功，但会有性能损失。之所发生性能损失，是因为文件系统可能试图通过在冲突的打开操作期间完全不使用直接 I/O，或通过每个直接 I/O 之前清空文件系统高速缓存并在每个直接写入之后使文件高速缓存无效来确保每个打开操作查看到相同的文件数据。

无论 DIRECT_IO 配置参数是什么值，在 Windows™ 平台上的数据库空间块使用直接 I/O。

相关链接

AUTO_AIOVPS 配置参数 在第67页
onstat -d 命令: 显示块信息 在第414页
《*SinoDB* 管理员指南》: 通过使用直接 *I/O* 提高熟文件数据库空间的性能
《*SinoDB* 性能指南》: 直接 *I/O (UNIX)*
《*SinoDB* 性能指南》: 并发 *I/O* (仅限*AIX)*

DIRECTIVES 配置参数

使用 DIRECTIVES 配置参数来启用或禁用优化器指令。这些指令指定在开发 SELECT、UPDATE 和 DELETE 查询计划中查询优化器的行为。

onconfig.std 值

DIRECTIVES 1

值

0 = 禁用优化器指令

1 = 启用优化器指令

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

环境变量

IFX_DIRECTIVES

用法

设置 DIRECTIVES 为默认值 1，使数据库服务器处理优化器指令。设置 DIRECTIVES 为 0，以禁止数据库服务器处理优化器指令。

客户端程序还可将 IFX_DIRECTIVES 环境变量设置为 ON 或 OFF 来启用或禁用数据库服务器处理指令。IFX_DIRECTIVES 环境变量的设定可覆盖 DIRECTIVES 配置参数的设置。如果未设置 IFX_DIRECTIVES 环境变量，则客户端的所有会话将继承数据库服务器处理指令的配置。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB 性能指南*》: 优化器指令

《*SinoDB SQL 指南: 参考*》: *IFX_DIRECTIVES 环境变量*

《*SinoDB SQL 指南: 语法*》: 其他语法段

DISABLE_B162428_XA_FIX 配置参数

使用 DISABLE_B162428_XA_FIX 配置参数来指定何时释放事务。

onconfig.std 值

不在 onconfig.std 文件中。

值

0 = （默认值）仅当调用 xa_rollback 时释放事务。

1 = 如果不是 xa_rollback 的事务回滚，则释放事务。

单位

整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

可以使用 IFX_XASTDCOMPLIANCE_XAEND 环境变量来覆盖客户端会话的 DISABLE_B162428_XA_FIX 配置参数设置。IFX_XASTDCOMPLIANCE_XAEND 设置为 1，则仅当调用 xa_rollback 时才释放事务。IFX_XASTDCOMPLIANCE_XAEND 设置为 0，则在不是 xa_rollback 的事务回滚时才释放事务。

相关链接

《SinoDB SQL 指南: 参考》: IFX_XASTDCOMPLIANCE_XAEND 环境变量

DRDA_COMMBUFFSIZE 配置参数

使用 DRDA_COMMBUFFSIZE 配置参数来指定 DRDA® 通信缓冲区的大小。

当建立一个 DRDA® 会话时，为会话分配等于当前缓冲区大小的通信缓冲区。如果后来更改了缓冲区大小，现有的连接不受影响，但新的 DRDA® 连接将使用新的大小。SinoDB® 自动地将大于 2MB 的值重置为 2MB，并将小于 4KB 的值重至为 默认值 32KB。

onconfig.std 值

不在 onconfig.std 文件中。

如果未出现

32K

值

Minimum = 4 KB

Maximum = 2 MB

生效

当初始化共享内存时

用法

通过为值添加 'M' 或 'K'，用户可以 MB 或 KB 为单位来指定 DRDA_COMMBUFFSIZE 的值。字母不区分大小写。默认为 KB。例如，可以下列方法之一来指定一个 1 MB 的缓冲区：

- DRDA_COMMBUFFSIZE 1M
- DRDA_COMMBUFFSIZE 1m
- DRDA_COMMBUFFSIZE 1024K
- DRDA_COMMBUFFSIZE 1024k
- DRDA_COMMBUFFSIZE 1024

相关链接

《SinoDB 管理员指南》: 使用 DRDA_COMMBUFFSIZE 配置参数指定 DRDA 通信缓冲区的大小

DRAUTO 配置参数

设置 DRAUTO 配置参数来指定高可用系统的 HDR 故障转移方式。

onconfig.std 值

DRAUTO 0

值范围

值	描述
0	禁用自动故障转移。当主服务器故障或失去网路连接时，HDR 辅助服务器成为只读。
1	Automatic failover is enabled. 启用自动故障转移。当主服务器故障或失去网路连接时，将 HDR 辅助服务器转为标准服务器。HDR 辅助服务器逐渐地结束客户端连接并关机，然后重启成为标准服务器。

值	描述
	当故障的主服务器重启或重新连接到网路时，将标准服务器转回 HDR 辅助服务器。 如果您配置了连接管理器来执行故障转移，则步使用此设置。
2	启动自动故障转移。当主服务器故障或失去网路连接时，将 HDR 辅助服务器转为主服务器。HDR 辅助服务器将保持客户端连接并且不关机。 当故障的主服务器重启或重新连接网路时，将其转为 HDR 辅助服务器。 如果您配置了连接管理器来执行故障转移，则步使用此设置。
3	通过连接管理器来控制故障转移。必须配置连接管理器，并且使其为活动状态以进行自动故障转移。

生效

当初始化共享内存时。

用法

高可用性集群的所有服务器必须有相同的 DRAUTO 配置参数设置。

DRAUTO 配置参数不控制 SDS 辅助服务器或 RS 辅助服务器的故障转移。要设置有 SD 辅助服务器或 RS 辅助服务器的高可用性集群的自动故障转移，请配置连接管理器。

重要： 如果使用连接管理器来控制故障转移，所有集群服务器上的 DRAUTO 配置参数必须设置为 3。在连接管理器是活动的情况下，不可执行手动故障转移。

相关链接

onstat -g dri 命令: 显示高可用性数据复制信息 在第454页

《*SinoDB* 管理员指南》: *HDR* 复制的完全同步模式

《*SinoDB* 管理员指南》: *HDR* 复制的异步模式

《*SinoDB* 管理员指南》: *HDR* 复制的接近同步模式

《*SinoDB* 管理员指南》: 将主服务器数据复制到辅助服务器

DRIDXAUTO 配置参数

使用 DRIDXAUTO 配置参数来指定是否在 HDR 辅助服务器检测到损坏的索引时主高可用性数据复制（HDR）服务器自动启动索引复制。

onconfig.std 值

DRIDXAUTO 0

值

0 = 关

1 = 开

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

要更改活动的服务器实例的 DRIDXAUTO 配置参数的值，请使用 onmode -d idxauto 命令。不需要重启服务器实例。然而，onmode -d idxauto 命令不会更改 onconfig 文件中的 DRIDXAUTO 配置参数的值。

相关链接

onstat -g dri 命令: 显示高可用性数据复制信息 在第454页

onmode -d 命令: 使用数据复制来复制索引 在第336页

DRINTERVAL 配置参数

使用 DRINTERVAL 配置参数来指定两次清空数据复制缓冲区间隔的最大秒数，无论使用 HDR SYNC 模式或使用 HDR_TXN_SCOPE 配置参数所指定的同步模式。

onconfig.std 值

DRINTERVAL 0

值

-1 = 使用 HDR SYNC 模式。如果主服务器使用无缓冲的日志记录，则复制是同步的。

0 = HDR_TXN_SCOPE 配置参数的值，决定 HDR 数据复制的同步模式。

正整数 = 使用 HDR ASYNC 模式。正整数是两次清空数据库缓冲区间隔的最大秒数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

DRINTERVAL 配置参数控制复制延迟，并用于设置复制同步。

如果与无缓冲日志记录一起使用，HDR SYNC 模式与通过 HDR_TXN_SCOPE 配置参数设置的近似同步模式是一样的。

表 65: DRINTERVAL、HDR_TXN_SCOPE 和日志记录设置的矩阵，及其生成的 HDR 复制模式

DRINTERVAL	HDR_TXN_SCOPE	日志记录	结果
-1	无	有缓冲	异步复制
-1	无	无缓冲	近似同步复制
0	FULL_SYNC	有缓冲	完全同步复制
0	FULL_SYNC	无缓冲	完全同步复制
0	ASYNC	有缓冲	异步复制
0	ASYNC	无缓冲	异步复制
0	NEAR_SYNC	有缓冲	近似同步复制
0	NEAR_SYNC	无缓冲	近似同步复制
正整数	无	有缓冲	异步复制
正整数	无	无缓冲	异步复制

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -g dri 命令: 显示高可用性数据复制信息 在第454页

HDR_TXN_SCOPE 配置参数 在第118页

《*SinoDB SQL* 指南: 语法》: *HDR_TXN_SCOPE* 会话环境选项

《*SinoDB* 管理员指南》: *HDR* 复制的完全同步模式

《*SinoDB* 管理员指南》: *HDR* 复制的异步模式

《*SinoDB* 管理员指南》: *HDR* 复制的接近同步模式

《*SinoDB* 管理员指南》: 将主服务器数据复制到辅助服务器

《*SinoDB Enterprise Replication* 指南》：辅助服务器的复制延迟
FAILOVER_TX_TIMEOUT 配置参数 在第113页

DRLOSTFOUND 配置参数

使用 DRLOSTFOUND 配置参数来指定 HDR 失而复得文件的路径名。此文件表示，在主数据库服务器发生故障时，某些事务已在 HDR 主数据库服务器上提交，在辅助服务器上尚未提交。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/etc/dr.lostfound

在 Windows™ 上：$INFORMIXDIR\tmp

值

pathname = dr.lostfound 文件的路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当 DRINTERVAL 配置参数设置为 -1 时，如果主数据库服务器和辅助数据库服务器之间的更新同时发生，则 DRLOSTFOUND 配置参数不适用。

失而复得文件 dr.lostfound.*timestamp* 创建时在文件名后面带有时间戳，以便数据库服务器不会覆写到另一个存在的失而复得文件。您不能使用失而复得文件来重新应用丢失的事务。

相关链接

onstat -g dri 命令：显示高可用性数据复制信息 在第454页
《*SinoDB* 管理员指南》：失物招领事务

DRTIMEOUT 配置参数

使用 DRTIMEOUT 配置参数来指定高可用性数据复制对中的一台数据库服务器等待来自该对中另一台数据库服务器的传输确认的时间长度（以秒计）。 此参数仅适用于高可用性数据复制对。

onconfig.std 值

DRTIMEOUT 30

值

正整数

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

使用以下公式来计算指定给 DRTIMEOUT 配置参数的值：

```
DRTIMEOUT = wait_time / 4
```

此公式中，*wait_time* 为高可用性数据复制对中数据库服务器在假定发生高可用性数据复制失败之前必须等待的时间长度（以秒计）。

例如，您确定系统的 *wait_time* 是 160 秒。使用先前公式设置 DRTIMEOUT 如下：

```
DRTIMEOUT = 160 seconds / 4 = 40 seconds
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
onstat -g dri 命令: 显示高可用性数据复制信息 在第454页
《*SinoDB* 管理员指南》: *HDR* 复制的完全同步模式
《*SinoDB* 管理员指南》: *HDR* 复制的异步模式
《*SinoDB* 管理员指南》: *HDR* 复制的接近同步模式
《*SinoDB* 管理员指南》: 将主服务器数据复制到辅助服务器

DS_HASHSIZE 配置参数

使用 DS_HASHSIZE 配置参数来指定数据分布高速缓存和其他高速缓存中的散列存储区的数目。数据库服务器存储和访问数据分布高速缓存中 UPDATE STATISTIC 语句于 MEDIUM 或 HIGH 模式下生成的列统计信息。

onconfig.std 值
DS_HASHSIZE 31

值
任何正整数；建议质数。

单位
散列存储区或列表的数目

生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

更新 DS_HASHSIZE 和 DS_POOLSIZE 配置参数值可提升多用户环境中频繁使用的查询的性能。

DS_HASHSIZE 配置参数设置以下高速缓存的散列存储区的数目：

- 数据分布高速缓存
- 扩展类型名高速缓存
- 扩展类型标识高速缓存
- Cast 高速缓存
- 运算符类实例高速缓存
- 例程解析高速缓存
- 合计高速缓存
- 辅助瞬态高速缓存

相关链接

DS_POOLSIZE 配置参数 在第105页
onstat -g dsc 命令: 显示分布式高速缓存信息 在第457页
《*SinoDB* 性能指南》: 数据分布配置
《*SinoDB* 性能指南》: 配置和监视内存高速缓存

DS_MAX_QUERIES 配置参数

使用 DS_MAX_QUERIES 配置参数来指定并发运行的并行数据库查询（PDQ）的最大数目。

DS_MAX_QUERIES 配置参数值取决于 DS_TOTAL_MEMORY 配置参数的设置：

- 如果设置了 DS_TOTAL_MEMORY 配置参数，则 DS_MAX_QUERIES 的值是 DS_TOTAL_MEMORY / 128，向下取整到最接近的整数值。
- 如果未设置 DS_TOTAL_MEMORY 配置参数，则 DS_MAX_QUERIES 的值是 2 * num，此处 num 是 VPCLASS 配置参数中指定的 CPU 数。

onconfig.std 值

未设置。

如不存在

2* num * 128，此处 num 是 VPCLASS 配置参数中指定的 CPU 数

值

最小值 = 1

最大值 = 8,388,608（8 MB）

单位

查询数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

内存分配管理器（MGM）根据以下公式为查询保留内存：

```
memory_reserved = DS_TOTAL_MEMORY *
                  (PDQ-priority / 100) *
                (MAX_PDQPRIORITY / 100)
```

在 PDQPRIORITY 环境变量或 SQL 语句 SET PDQPRIORITY 中指定 PDQPRIORITY 的值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onstat -g mgm 命令: 显示 *MGM* 资源信息 在第480页

《*SinoDB* 性能指南》: 并行数据库查询 *(PDQ)*

《*SinoDB SQL* 指南: 参考》: *PDQPRIORITY* 环境变量

VPCLASS 配置参数 在第196页

DS_MAX_SCANS 配置参数

使用 DS_MAX_SCANS 配置参数来限定数据库可以并发执行的 PDQ 扫描线程数。

onconfig.std 值

DS_MAX_SCANS 1048576 or (1024 * 1024)

值

10 - (1024 * 1024)

单位

PDQ 扫描线程数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当用户发出查询时，数据库服务器根据下列值来分配扫描线程数：

- PDQ 优先级的值（通过环境变量 PDQPRIORITY 或 SQL 语句 SET PDQPRIORITY 设置）
- 以 DS_MAX_SCANS 设置的上限
- 以 MAX_PDQPRIORITY 设置的因子
- 要扫描的表中的分段数（公式中的 *nfrags*）

内存分配管理器（MGM）尝试根据以下公式为查询保留的扫描线程数

```
reserved_threads = min (nfrags, (DS_MAX_SCANS *
                  PDQPRIORITY / 100 *
                 MAX_PDQPRIORITY / 100) )
```

如果公式中的 DS_MAX_SCANS 部分大于或等于要扫描的表中的分段数，则查询将保留在就绪队列中，直到可用扫描线程数与表中分段数一样多。一旦进行，因为线程以并行方式扫描分段，所以查询会快速执行。

例如，如果 *nfrags* 等于 24、DS_MAX_SCANS 等于 90、PDQPRIORITY 等于 50 以及 MAX_PDQPRIORITY 等于 60，则查询会等到有 *nfrags* 个可用扫描线程时才开始执行。扫描是以并行方式进行。

如果 DS_MAX_SCANS 低于分段数，查询可能会尽快执行，但因为某些线程是循序地扫描分段，所以会执行比较久。

如果将前面示列中的 DS_MAX_SCANS 降低到 40，则查询需要更少资源（12 个扫描线程）来开始执行，但是每个线程需要循序地扫描两个分段。执行时间会更久。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onstat -g mgm 命令: 显示 MGM 资源信息 在第480页

《*SinoDB 性能指南*》: *并行数据库查询 (PDQ)*

《*SinoDB SQL 指南: 参考*》: *PDQPRIORITY 环境变量*

DS_NONPDQ_QUERY_MEM 配置参数

使用 DS_NONPDQ_QUERY_MEM 配置参数来增加非并行数据库查询（PDQ）的可用内存量。（仅当 PDQ 优先级别设为 0 时才可使用此参数。）

onconfig.std 值

DS_NONPDQ_QUERY_MEM:

在 UNIX™ 上：256

在 Windows™ 上：128

值

从默认值到 DS_TOTAL_MEMORY 值的 25%

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果为 DS_NONPDQ_QUERY_MEM 参数指定值，请根据表行的数目与大小来决定与调整其值。

提示： 通常该值设置不超过最大可用临时数据库空间大小

DS_NONPDQ_QUERY_MEM 值是在数据初始化期间根据计算得到的 DS_TOTAL_MEMORY 值计算而得的。如果在 DS_NONPDQ_QUERY_MEM 处理期间，数据库服务器更改了您设置的值，数据库服务器会按以下格式发送消息：

```
DS_NONPDQ_QUERY_MEM recalculated and changed from old_value Kb to new_value Kb.
```

在消息中，*old_value* 表示您在用户配置文件中指定给 DS_NONPDQ_QUERY_MEM 的值，而 *new_value* 表示数据库服务器决定的值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -g mgm 命令: 显示 MGM 资源信息 在第480页

DS_POOLSIZE 配置参数

使用 DS_POOLSIZE 参数来指定数据分布高速缓存和其他高速缓存中的最大条目数。数据库服务器存储和访问数据分布高速缓存中 UPDATE STATISTICS 语句在 MEDIUM 或 HIGH 模式下生成的列统计信息。

onconfig.std 值

DS_POOLSIZE 127

值

正值 127 或更大的表示高速缓存中初始最大条目数的一半。 最大值取决于共享内存配置和服务器实例的可用共享内存。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wm 命令来增加内存中的值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

使用 DS_HASHSIZE 和 DS_POOLSIZE 配置参数来提升多用户环境中频繁运行的查询的性能。

高速缓存中的初始条目数是 DS_POOLSIZE 配置参数值的两倍。例如，如果 DS_POOLSIZE 配置参数设置为 127，则在高速缓存中允许的条目数为 254。如果高速缓存中的所有条目都满了，则高速缓存的大小自动地增长 10%。要减少高速缓存的大小，请降低 onconfig 文件中的 DS_POOLSIZE 配置参数值并重启服务器。

DS_POOLSIZE 配置参数设置下列高速缓存中的条目数：

- 数据分布高速缓存
- 扩展类型名高速缓存
- 扩展类型标识高速缓存
- Cast 高速缓存
- 运算符类实例高速缓存
- 例程解析高速缓存
- 合计高速缓存
- 辅助瞬态高速缓存

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DS_HASHSIZE 配置参数 在第102页

onstat -g dsc 命令: 显示分布式高速缓存信息 在第457页

《*SinoDB 性能指南*》: 数据分布配置

《*SinoDB* 性能指南》: 配置和监视内存高速缓存

DS_TOTAL_MEMORY 配置参数

使用 DS_TOTAL_MEMORY 配置参数来指定 PDQ 查询可用的内存量。此数量应小于计算机物理内存，减去固定开销（比如操作系统大小和缓冲持大小）。

onconfig.std 值

未设置。

如果未出现

如果 SHMTOTAL=0 并设置了 DS_MAX_QUERIES，则 DS_TOTAL_MEMORY = DS_MAX_QUERIES * 128。

如果 SHMTOTAL=0 并且未设置 DS_MAX_QUERIES ，则 DS_TOTAL_MEMORY = num_cpu_vps * 2 * 128。

值

如果设置了 DS_MAX_QUERIES，则最小直是 DS_MAX_QUERIES * 128。

如果未设置 DS_MAX_QUERIES，则最小值是 num_cpu_vps * 2 * 128。

除了您机器上所使用的软件的限制，没有最大值限制。

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

不要将 DS_TOTAL_MEMORY 跟配置参数 SHMTOTAL 和 SHMVIRTSIZE 混淆。SHMTOTAL 设置指定数据库服务器的所有内存（内存的常驻、虚拟和消息部分的总和）。SHMVIRTSIZE 设置指定虚拟部分的大小。DS_TOTAL_MEMORY 是 SHMVIRTSIZE 的一个逻辑子集。

对于 OLTP 应用程序，将 DS_TOTAL_MEMORY 设为 SHMTOTAL 值的 20% 到 50% 之间（以 KB 计）。

对于涉及大的决策支持（DSS）查询的应用程序，将 DS_TOTAL_MEMORY 值增加到 SHMTOTAL 的 50% 到 80% 之间。如果仅将数据库服务器用于 DSS 查询，则将此参数设为 SHMTOTAL 的 90% 到 100% 之间。

将 DS_TOTAL_MEMORY 配置参数设置为不大于 SHMVIRTSIZE - 10 megabytes 的任何值。

有关您的平台上可用的最大内存的信息，请参阅机器说明文件。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SHMTOTAL 配置参数 在第168页

SHMVIRTSIZE 配置参数 在第169页

VPCLASS 配置参数 在第196页

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onstat -g mgm 命令: 显示 MGM 资源信息 在第480页

《*SinoDB* 性能指南》: 确定 *DS_TOTAL_MEMORY* 的算法

DS_TOTAL_MEMORY 的算法

当未设置 DS_TOTAL_MEMORY 或设置了不适当的值时，数据库服务器会派生 DS_TOTAL_MEMORY 的值。有关算法的信息，请参阅《*SinoDB*® 性能指南》中配置对内存利用率的影响。

DUMPCNT 配置参数 (UNIX™)

使用 DUMPCNT 配置参数来指定数据库服务器线程转储共享内存或调用 gcore 实用程序生成核心文件的会话中断言失败的次数。

onconfig.std 值

DUMPCNT 1

值

正整数

单位

可在会话中生成的共享内存转储或核心文件的数目

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当数据库服务器不能继续正常处理时，发生断言失败。

断言失败可生成与 DUMPCNT 配置参数值相同数量的核心文件或共享内存转储。进一步的断言失败会在消息日志中生成错误，并可能生成应用程序错误，但不保存进一步的诊断信息。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DUMPDIR 配置参数 在第108页

DUMPSHMEM 配置参数 *(UNIX)* 在第109页

《*SinoDB* 管理员指南》: 收集诊断信息

DUMPSHMEM 配置参数 *(UNIX)* 在第109页

DUMPCORE 配置参数 (UNIX™)

使用 DUMPCORE 配置参数控制断言失败是否导致虚拟处理器转储核心映像。该核心文件保留在最后调用数据库服务器的目录中。（DUMPDIR 参数对核心文件的位置没有影响。）

onconfig.std 值

DUMPCORE 0

值

0 = 不转储核心映像。

1 = 转储核心映像。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

Warning: 当 DUMPCORE 设置为 1，断言失败会导致虚拟处理器转储核心映像，从而导致数据库服务器中止。设置 DUMPCORE 仅为了在受控环境中进行调试。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
《*SinoDB* 管理员指南》: 收集诊断信息

DUMPDIR 配置参数

DUMPDIR 指定数据库服务器用于转储失败断言的共享内存、gcore 文件或的消息的目录。

由于共享内存可能很大，所以将 DUMPDIR 设置到具有大量空间的文件系统。DUMPDIR 设置的目录必须存在，才能启动服务器。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/tmp

在 Windows™ 上：$INFORMIXDIR\tmp

值

用户 informix 有写入权限的任何目录

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

DUMPCNT 配置参数 *(UNIX)* 在第107页
DUMPSHMEM 配置参数 *(UNIX)* 在第109页
《*SinoDB* 管理员指南》: 收集诊断信息
DUMPSHMEM 配置参数 *(UNIX)* 在第109页

DUMPGCORE 配置参数 (UNIX™)

使用 DUMPGCORE 配置参数指定是否转储 gcore 核心文件。在支持 gcore 的操作系统上使用此参数。

onconfig.std 值

DUMPGCORE 0

值

0 = 不转储 gcore。

1 = 转储 gcore。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果您设置了 DUMPGCORE，但操作系统不支持 gcore，那么数据库服务器消息日志中的消息会表明尝试转储核心映像，但数据库服务器找不到预期的文件。（如果操作系统不支持 gcore，请该设置 DUMPCORE。）

如设置了 DUMPGCORE，则每当虚拟处理器遇到断言失败时，数据库服务器就会调用 gcore。gcore 实用程序指示虚拟处理器将核心映像转储到 DUMPDIR 所指定目录中的 core.pid.cnt 文件，并继续处理。

pid 是虚拟处理器的进程标识号。cnt 值在每次此进程遇到断言失败时递增。cnt 值范围从 1 到 DUMPCNT 值。在 DUMPCNT 值达到之后，不再创建核心文件。如果虚拟处理器继续遇到断言失败，则向消息日志报告错误（并可能向应用程序报告），但不保存进一步诊断信息。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》: 收集诊断信息

DUMPSHMEM 配置参数 (UNIX™)

使用 DUMPSHMEM 配置参数来指出是否在断言失败时创建共享内存转储。此配置参数还指定向 DUMPDIR 配置参数所指定目录中的shmem.*pid*.*cnt* 文件写入多少内存。

onconfig.std 值

DUMPSHMEM 1

值

0 = 不创建共享内存转储。

1 = 创建数据库使用的所有共享内存的共享内存转储。

2 = 创建不包含常驻内存中缓冲池的共享内存转储。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果 DUMPSHMEM 设置为 1，转储数据库服务器使用的所有共享内存，这可能导致生成很大的文件。当空间有限时，将 DUMPSHMEM 设置为 2，因为此设置会生成比较小的共享内存转储文件。

pid 是虚拟处理器的进程标识号。*cnt* 值在每次此进程遇到断言失败时递增。*cnt* 值范围从 1 到 DUMPCNT 配置参数的值。在 DUMPCNT 值达到之后，不再创建核心文件。如果数据库服务器继续检测到不一致，则向消息日志报告错误（并可能向应用程序报告），但不保存进一步诊断信息。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
DUMPCNT 配置参数 *(UNIX)* 在第107页
DUMPDIR 配置参数 在第108页
DUMPCNT 配置参数 *(UNIX)* 在第107页
DUMPDIR 配置参数 在第108页
在共享内存转储文件上运行 *onstat* 命令 在第404页
onstat -o 命令: 共享内存内容输出到文件 在第550页
《*SinoDB* 管理员指南》: 收集诊断信息

DYNAMIC_LOGS 配置参数

使用 DYNAMIC_LOGS 配置参数来允许当有需要防止事务阻塞时动态地添加逻辑日志。

onconfig.std 值

DYNAMIC_LOGS 2

值

0 = 关闭动态日志分配。

1 = 启动 "log file required" 报警并暂停以允许手动添加逻辑日志文件。您可以在当前日志文件之后立即添加日志文件或在日志文件列表尾部添加日志文件。

2 = 开启动态日志分配。当数据库服务器动态地添加日志文件时，会发出 "dynamically added log file" 警报。

生效

对于 HDR：当数据库服务器关闭并重启时

对于 Enterprise Replication：当 Enterprise Replication 启动时

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果 DYNAMIC_LOGS 设置为 2，当下一个活动的日志文件包含打开的事务时，数据库服务器会动态地分配一个新的日志文件。动态日志分配可防止长事务从阻塞的事务回滚。

如果您要选择新逻辑日志文件的大小和位置，请将 DYNAMIC_LOGS 设置为 1。使用带有大小（-s）、位置（-d dbspace）和 -i 选项的 onparams -a 命令在当前日志文件之后添加一个日志文件。

如果 DYNAMIC_LOGS 配置参数的值为 0 且发生事务阻塞，则关闭数据库服务器、设置 DYNAMIC_LOGS 为 1 或 2，然后重启数据库服务器。

重要： 如果您使用具有动态日志分配的 Enterprise Replication，则请将 LTXEHWM 设置为低于 70 的值。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

LTXEHWM 配置参数 在第128页

LTXHWM 配置参数 在第129页

onparams -a -d dbspace: 添加逻辑日志文件 在第352页

《*SinoDB* 管理员指南》*:* 逻辑日志

EILSEQ_COMPAT_MODE 配置参数

使用 EILSEQ_COMPAT_MODE 配置参数来控制 SinoDB® 是否检查客户端应用程序插入的字符数据是否包含当前数据库语言环境不识别的代码指向序列。

onconfig.std 值

EILSEQ_COMPAT_MODE 0

值

0 = SinoDB® 以当前语言环境来验证传入的字符序列，如果有任何无效字符，则返回错误 -202。

1 = SinoDB® 不验证传入的字符序列。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

如果设置 EILSEQ_COMPAT_MODE 配置参数为 0，则仅有效的字节序列可以插入数据库。

EILSEQ_COMPAT_MODE 配置参数防止下列情况的 202 错误：

- 当从数据库检索数据时。
- 当无效字符是位于字符串尾部且为部分字符时

相关链接

《*SinoDB GLS* 用户指南》*: DB_LOCALE* 环境变量

《*SinoDB GLS* 用户指南》*: GL_USEGLU* 环境变量

ENABLE_SNAPSHOT_COPY 配置参数

使用 ENABLE_SNAPSHOT_COPY 配置参数来启用或禁用使用 ifxclone 实用程序克隆服务器。

onconfig.std 值

0

值

0 = 禁止克隆

1 = 允许克隆

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

ENABLE_SNAPSHOT_COPY 配置参数决定是否可使用 ifxclone 实用程序克隆服务器。设置 ENABLE_SNAPSHOT_COPY 配置参数为 1，则允许克隆。设置为 0，则禁止使用 ifxclone 实用程序克隆服务器。

如果您在安装期间创建服务器，则自动地启用 ENABLE_SNAPSHOT_COPY 配置参数。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

ifxclone 实用程序 在第355页

EXPLAIN_STAT 配置参数

使用 EXPLAIN_STAT 配置参数来启用或禁用在解释输出文件中包含 Query Statistics 部分。

可使用 SET EXPLAIN 语句或 onmode -Y*sessionid* 命令来生成输出文件。当启用 EXPLAIN_STAT 配置参数时，Query Statistics 部分显示查询计划中预计的行数和返回的实际行数。

onconfig.std 值

EXPLAIN_STAT 1

值

0 = 禁用在解释输出文件中包含 Query Statistics 部分。

1 = 启用在解释输出文件中包含 Query Statistics 部分。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -Y: 动态更改 SET EXPLAIN 在第349页

《SinoDB SQL 指南: 语法》: SET EXPLAIN 语句

《SinoDB 性能指南》: 查询计划示例报告

EXT_DIRECTIVES 配置参数

使用 EXT_DIRECTIVES 配置参数来启用或禁用外部查询优化器指令的使用。

onconfig.std 值

EXT_DIRECTIVES 0

值

0（默认） = 关。即使 IFX_EXTDIRECTIVES 为开，也不可使用指令。

1 = 开。如果 IFX_EXTDIRECTIVES 为开，则为会话启用指令。

2 = 开。即使未设置 IFX_EXTDIRECTIVES，也可以使用指令。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

通过将 EXT_DIRECTIVES 配置参数与客户端 IFX_EXTDIRECTIVES 环境变量一起使用来启用外部指令，如下：

IFX_EXTDIRECTIVES 的设置取代 EXT_DIRECTIVES 配置参数的设置。如果您未设置 IFX_EXTDIRECTIVES 环境变量，则客户端所有会话将继承数据库服务器处理外部指令的配置。

SQL 的 SET ENVIRONMENT EXTDIRECTIVES 语句指定的设置取代（仅对于当前用户会话）IFX_EXTDIRECTIVES 环境变量和 EXT_DIRECTIVES 配置参数的设置。

相关链接

《SinoDB 性能指南》: 外部优化器指令

《SinoDB SQL 指南: 参考》: IFX_EXTDIRECTIVES 环境变量

《SinoDB SQL 指南: 语法》: EXTDIRECTIVES 会话环境选项

《SinoDB SQL 指南: 语法》: 优化器指令

EXTSHMADD 配置参数

使用 EXTSHMADD 配置参数指定用户定义例程或 DataBlade® 例程在用户定义虚拟处理器中运行时添加的虚拟扩展段的大小。

onconfig.std 值

EXTSHMADD 8192

值

32 位操作系统：1024 - 524288

64 位操作系统：1024 - 4294967296

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

当线程在用户定义的虚拟处理器中运行时，会创建虚拟扩展段。在 onstat -g seg 命令的输出中，虚拟扩展段有一个 VX 类。如果 onconfig 文件中未设置 EXTSHMADD 配置参数，则由 SHMADD 配置参数的值来设置虚拟扩展段的大小。

相关链接

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页
SHMADD 配置参数 在第166页
《*SinoDB* 管理员指南》: 共享内存的虚拟扩展部份

FAILOVER_CALLBACK 配置参数

使用 FAILOVER_CALLBACK 配置参数指定数据库服务器从辅助服务器转成主服务器或标准服务器时执行的脚本。

onconfig.std 值

未设置。

值

pathname = FAILOVER_CALLBACK 参数指定的脚本的全路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

将 FAILOVER_CALLBACK 设置为脚本的全路径名。

FAILOVER_TX_TIMEOUT 配置参数

在高可用性集群环境中，使用 FAILOVER_TX_TIMEOUT 配置参数可以在主服务器故障转移之后完成事务。

使用 FAILOVER_TX_TIMEOUT 配置参数来指出，在故障转移之后服务器开始回滚事务之前等待的最大秒数。将高可用性集群中所有服务器上的 FAILOVER_TX_TIMEOUT 配置参数设置为相同的值。

onconfig.std 值

FAILOVER_TX_TIMEOUT 0

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当高可用性集群环境中发生故障转移时，辅助服务器之一会取代主服务器角色。成为新的主服务器的辅助服务器称为故障转移服务器。

通过将 FAILOVER_TX_TIMEOUT 配置参数设置为一个大于零的值，则可启用事务存活。当启用事务存活时，故障转移服务器必须能够联系剩余的辅助服务器来同步和恢复任何打开的事务。同样地，存活的辅助服务器必须能够与故障转移服务器建立连接以重新发送任何挂起的事务。FAILOVER_TX_TIMEOUT配置参数指定服务器开始回滚事务之前等待的时长。

在故障转移服务器上，如果超过了 FAILOVER_TX_TIMEOUT 指定的秒数，则终止并回滚任何未与存活服务器同步的打开事务。

在剩余的辅助服务器上，如果超过了 FAILOVER_TX_TIMEOUT 指定的秒数，则服务器上的任何打开的事务会返回错误。

FAILOVER_TX_TIMEOUT 设置为 0，在发生故障转移时，立即回滚所有打开的事务。

如果主服务器故障且辅助服务器取代主服务器的角色失败，则回滚任何打开的事务，并且客户端不能进行更新。例如，如果在辅助服务器上启动一个更新活动且主服务器故障，接着，故障转移处理不能完成且不能建立新的主服务器，那么在预定的时间之后，客户端请求超时，将 sqlexec 线程置于不确定状态。

在上述场景中，回滚活动的事务，但直到新的主服务器建立后物理回滚才会发生（因为主服务器管理日志）。在这些情况下，会话察觉不到在辅助服务器上执行的操作。会话察觉不到部分应用的事务的回滚，因为部分事务的回滚要到新主服务器建立时才会发生。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DRINTERVAL 配置参数 在第100页

HDR_TXN_SCOPE 配置参数 在第118页

《*SinoDB* 管理员指南》: 将主服务器数据复制到辅助服务器

《*SinoDB* 管理员指南》: *HDR* 复制的完全同步模式

《*SinoDB* 管理员指南》: *HDR* 复制的接近同步模式

FASTPOLL 配置参数

使用 FASTPOLL 配置参数来启用或禁用网络的快速轮询。FASTPOLL 是特定于平台的配置参数。

onconfig.std 值

FASTPOLL 1

值

0 = 禁用快速轮询。

1 = 启用快速轮询。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

FILLFACTOR 配置参数

使用 FILLFACTOR 配置参数指定索引页充满度。低值提供索引增长的空间。高值压缩索引。

如果索引充满了（100%），则任何新插入将导致分裂节点。还可以设置 FILLFACTOR 作为 CREATE INDEX 语句的选项。在 CREATE INDEX 语句上的设置取代 ONCONFIG 文件值。

不可将 FILLFACTOR 配置参数用于林立的树状索引。

onconfig.std 值

FILLFACTOR 90

值

1 - 100

单位

百分比

生效

当建构索引时。现有的索引不会改变。要使用新值，必须重新建构索引。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

B 型树索引页的结构 在第264页

FULL_DISK_INIT 配置参数

使用 FULL_DISK_INIT 配置参数来防止现有数据库服务器实例意外的磁盘重新初始化。此配置参数指定，当在 root 路径位置上存在页零（在第一个块位置的第一页）时，是否可以在 SinoDB® 实例上运行磁盘初始化命令（oninit -i）。

onconfig.std 值

FULL_DISK_INIT 0

值

0 = 仅当 root 路径位置没有页零时才可运行 oninit -i 命令。

1 = 在所有情况下都可运行 oninit -i 命令，并在磁盘初始化之后将 FULL_DISK_INIT 配置参数重置为 0。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

当 FULL_DISK_INIT 配置参数设置为 1 时，任何实例启动命令（例如，oninit 和 oninit -i）都将该配置参数重置为 0。

当 FULL_DISK_INIT 配置参数设置为 0 且数据库服务器发现页零时，如果您开始运行 oninit -i 命令，则 oninit -i 不会运行并且服务器会在 online.log 中报告错误。

页零是 SinoDB® 系统页，包含有关服务器实例的通用信息。当服务器实例初始化时，创建此页。

相关链接

oninit 实用程序 在第319页

HA_ALIAS 配置参数

HA_ALIAS 配置参数用于定义高可用性集群中服务器对服务器通信的网络别名。连接管理器、 ifxclone 实用程序和 onmode -d 命令也可使用指定的网路别名。

onconfig.std 值

未设置。HA_ALIAS 配置参数适用于高可用性集群服务器。

值

HA_ALIAS 配置参数值必须与 TCP sqlhosts 文件条目关联的 DBSERVERNAME 或 DBSERVERALIASES 配置参数值相符合。如果 DBSERVERNAME 或 DBSERVERALIASES 配置参数值包括了可选的监听器线程数，则 HA_ALIAS 配置参数值可省略该可选的监听器线程值。例如，如果 DBSERVERNAME 设置为 my_server-4，则 HA_ALIAS 设置为 my_server。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

对于高可用性集群中的主服务器，通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的值。此方式在高可用性集群的辅助服务器上不起作用。

对于高可用性集群中的主服务器，通过运行 onmode -wm 命令来重置内存中的值。此方式在高可用性集群的辅助服务器上不起作用。

用法

使用共享内存连接的高可用性集群服务器需要 HA_ALIAS 配置参数。

例如，如果高可用性集群服务器的 DBSERVERNAME 配置参数与共享内存 sqlhosts 文件条目关联，则请设置 DBSERVERALIAS 配置参数与相匹配的 HA_ALIAS 配置参数值，然后为该别名创建一个 TCP sqlhosts 文件条目。

onconfig 文件值：

```
DBSERVERNAME my_server
DBSERVERALIAS alias_1
HA_ALIAS alias_1
```

sqlhosts 文件值：

```
#dbservername  nettype   hostname  servicename  options
 my_server     onipcshm  host_1    port_1              #client-to-server
 alias_1       onsoctcp  host_1    port_2              #server-to-server
```

为高可用性集群中的所有服务器设置 HA_ALIAS 配置参数，还使您可将客户端/服务器通信与服务器对服务器通信分开。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DBSERVERALIASES 配置参数 在第88页

DBSERVERNAME 配置参数 在第90页

《*SinoDB* 管理员指南》*: HA_ALIAS* 配置参数中设置的连接信息

onmode -d: 设置数据复制类型 在第334页

onmode -d: 设置高可用性服务器的特性 在第335页

onmode -d 命令*:* 使用数据复制来复制索引 在第336页

HA_FOC_ORDER 配置参数

使用 HA_FOC_ORDER 配置参数为高可用性集群服务器定义单一的连接管理故障转移规则。

onconfig.std 值

HA_FOC_ORDER SDS,HDR,RSS

值

辅助服务器类型列表，以逗号分隔并按优先级顺序罗列。例如，默认值 SDS,HDR,RSS 表示主服务器故障转移到 SD 辅助服务器，然后是 HDR 辅助服务器，再然后是 RS 辅助服务器。

- HDR = 高可用性数据复制服务器
- RSS = 远程独立辅助服务器
- SDS = 共享磁盘辅助服务器

MANUAL = 对集群中所有连接管理器禁用自动故障转移。

分隔符

以逗号分隔值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

在运行带有 -wf HA_FOC_ORDER=*value* 或 -wm HA_FOC_ORDER=*value* 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

如果在高可用性集群的主数据库服务器上设置 HA_FOC_ORDER 配置参数，则连接到该主服务器的每个连接管理器都采用该设置。该值取代连接单元的 ORDER=*rule* 故障转移顺序规则。然后，高可用性集群中的每个数据库服务器都采用主服务器的 HA_FOC_ORDER 配置参数值作为自己的 HA_FOC_ORDER 配置参数。

如果主服务器上的 HA_FOC_ORDER 配置参数设置为 MANUAL，则在管理主服务器的集群的所有连接管理器上禁用自动故障转移。

如果连接管理器配置文件中的某一连接单元的 FOC ORDER 值设置为 DISABLED，则连接管理器不会对那个连接单元执行故障转移。

HA_FOC_ORDER 配置参数的语法

```
HA_FOC_ORDER
{  , { SDS  | HDR  | RSS }  | MANUAL }
```

示例

在以下示例中，您配置了两个连接管理器来管理一个有三个服务器的集群。

三个服务器为：

- server_1（主服务器）
- server_2（SD 辅助服务器）
- server_3（HDR 辅助服务器）

第一个连接管理器具有以下配置文件：

```
NAME connection_manger_1

CLUSTER cluster_1
{
   INFORMIXSERVER servers_1
   SLA sla_1 DBSERVERS=ANY
   FOC ORDER=ENABLED \
       PRIORITY=1
}
```

第二个连接管理器具有以下配置文件：

```
NAME connection_manger_2

CLUSTER cluster_1
{
   INFORMIXSERVER servers_1
   SLA sla_2 DBSERVERS=ANY
   FOC ORDER=ENABLED \
       PRIORITY=2
}
```

server_1 的 onconfig 文件有以下值：

```
HA_FOC_ORDER SDS,HDR
```

当 connection_manger_1 和 connection_manger_2 与 server_1 连接时，其配置成为：

```
NAME connection_manger_1

CLUSTER cluster_1
{
   INFORMIXSERVER servers_1
   SLA sla_1 DBSERVERS=ANY
   FOC ORDER=SDS,HDR \
       PRIORITY=1
}
```

```
NAME connection_manger_2
```

```
CLUSTER cluster_1
{
   INFORMIXSERVER servers_1
   SLA sla_2 DBSERVERS=ANY
   FOC ORDER=SDS,HDR \
       PRIORITY=2
}
```

在 server_2 和 server_3 的 onconfig 文件中的 HA_FOC_ORDER 条目的值更新为 SDS,HDR。

相关链接

《*SinoDB* 管理员指南》：高可用性集群的配置连接管理示例

《*SinoDB* 管理员指南》：*FOC* 连接管理器配置参数

HDR_TXN_SCOPE 配置参数

HDR_TXN_SCOPE 配置参数与 DRINTERVAL 配置参数一起使用，来指定高可用性集群中 HDR 复制的同步模式。

onconfig.std 值

HDR_TXN_SCOPE NEAR_SYNC

值

FULL_SYNC = 如果完全同步，则 HDR 复制。在事务可完成之前，事务需要 HDR 辅助服务器上的完成确认。

NEAR_SYNC = 如果几乎同步，则 HDR 复制。在事务可完成之前，事务需要 HDR 辅助服务器上的收到确认。如果与无缓冲日志记录一起使用，则在 DRINTERVAL 设置为 -1 时打开 SYNC 模式，其与几乎同步模式是一样的。

ASYNC = 如果完全异步，则 HDR 复制。在事务可完成之前，事务不需要 HDR 辅助服务器上的收到或完成确认。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

在运行带有 "onmode","-wf HDR_TXN_SCOPE=*value*" 或 "onmode","-wm HDR_TXN_SCOPE=*value*" 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

当 DRINTERVAL 配置参数设置为 0 时，HDR_TXN_SCOPE 三倏地值决定 HDR 复制的同步模式。

如果使用无缓冲日志记录，HDR SYNC 模式与通过 HDR_TXN_SCOPE 配置参数设置的几乎同步模式是相同的。

表 66: DRINTERVAL、HDR_TXN_SCOPE 和日志记录设置，以及其结果的 HDR 复制模式的矩阵

DRINTERVAL	HDR_TXN_SCOPE	日志记录	结果
-1	无	有缓冲的	异步复制
-1	无	无缓冲的	几乎同步复制
0	FULL_SYNC	有缓冲的	完全同步复制
0	FULL_SYNC	无缓冲的	完全同步复制
0	ASYNC	有缓冲的	异步复制

DRINTERVAL	HDR_TXN_SCOPE	日志记录	结果
0	ASYNC	无缓冲的	异步复制
0	NEAR_SYNC	有缓冲的	几乎同步复制
0	NEAR_SYNC	无缓冲的	几乎同步复制
正整数	无	有缓冲的	异步复制
正整数	无	无缓冲的	异步复制

相关链接

DRINTERVAL 配置参数 在第100页

FAILOVER_TX_TIMEOUT 配置参数 在第113页

《*SinoDB SQL* 指南: 语法》: *HDR_TXN_SCOPE* 会话环境选项

IFX_EXTEND_ROLE 配置参数

您的数据库系统管理员（DBSA），默认用户为 informix，可使用 IFX_EXTEND_ROLE 参数来控制授权哪些用户可注册 DataBlade® 模块或外部用户定义例程（UDR）。

onconfig.std 值

IFX_EXTEND_ROLE 1

值

1 或 On（默认）= 启用 EXTEND 角色的要求，以便管理员可授予用户权限来创建或删除包含 EXTERNAL 子句的 UDR。

0 或 Off = 禁用 EXTEND 角色的要求，以便拥有适当外部语言（C 或 JAVA）的 USAGE ON LANGUAGE 权限的任何用户可以注册或删除用那个语言编写的例程。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 安全指南》: 外部例程的安全性 *(UDRs)*

IFX_FOLDVIEW 配置参数

使用 IFX_FOLDVIEW 配置参数来启用或禁用视图折入。对于某些查询中涉及视图的情况，视图折入可以显著地提升查询性能。在这些情况下，视图折入父查询中，而不是将查询结果放到临时表中。

onconfig.std 值

IFX_FOLDVIEW 1

值

0 或 Off = 禁用视图折入。

1 或 On = 默认。启用视图折入。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

下列查询类型可利用视图折入：

- 包含 UNION ALL 子句且父查询包括常规联接、SinoDB® 联接、ANSI 联接或 ORDER BY 子句的视图

对于以下执行涉及视图的 UNION ALL 操作的查询类型，创建临时表且不执行视图折入：

- 视图具有下列子句之一：AGGREGATE、GROUP BY、ORDER BY、UNION、DISTINCT 或 OUTER JOIN（SinoDB® 或 ANSI 类型）。
- 父查询有 UNION 或 UNION ALL 子句。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 性能指南》：启用视图折叠以提高查询性能

IFX_XA_UNIQUEXID_IN_DATABASE 配置参数

使用 IFX_XA_UNIQUEXID_IN_DATABASE 配置参数来启用事务管理器，在相同的数据库服务器实例中使用相同的 XID 来表示不同数据库上的全局事务。

onconfig.std 值

None

默认值

0

值

0 = 禁用

1 = 启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

XID 是分布式 XA 事务的全局事务 ID。

如果 IFX_XA_UNIQUEXID_IN_DATABASE 配置参数设置为 1，数据库服务器允许事务管理器在相同的数据库服务器实例中使用相同的 XID 来表示不同数据库上的全局事务。因此，数据库可以是域，而不是服务器。

相关链接

onstat -G 命令：打印 *TP/XA* 事务信息 在第541页

《*SinoDB DataBlade API* 程序员指南》：符合 *XA* 的外部数据源

INFORMIXCONRETRY 配置参数

使用 INFORMIXCONRETRY 配置参数来指定初始连接尝试失败之后，可对每一数据库服务器进行连接尝试的最大数。这些尝试限定在 INFORMIXCONTIME 配置参数所指定的时间之内进行。

onconfig.std 值

INFORMIXCONRETRY 1

值

正整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

INFORMIXCONTIME 设置优先于 INFORMIXCONRETRY 设置。连接尝试可在超过 INFORMIXCONTIME 值之后，但在到达 INFORMIXCONRETRY 值之前结束。

要取代当前会话的 INFORMIXCONRETRY 配置参数的值，您可设置 SET ENVIRONMENT 语句的 INFORMIXCONRETRY 环境选项或客户端的 INFORMIXCONRETRY 环境变量。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 语法》: *INFORMIXCONRETRY* 会话环境选项

《*SinoDB SQL* 指南: 参考》: *INFORMIXCONRETRY* 环境变量

INFORMIXCONTIME 配置参数

使用 INFORMIXCONTIME 配置参数来指定 CONNECT 语句尝试建立到数据库服务器连接的秒数。

onconfig.std 值

INFORMIXCONTIME 60

值

正整数

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

要设置 INFORMIXCONTIME 配置参数的最佳值，请考虑节点间距离、硬件速度、流量和网络的并发级别。

INFORMIXCONTIME 值除以 INFORMIXCONRETRY 值的结果决定连接尝试之间的秒数。如果 INFORMIXCONTIME 设置为零，则数据库服务器使用默认值 60 秒。

要取代当前会话的 INFORMIXCONTIME 配置参数的值，则可设置 SET ENVIRONMENT 语句的 INFORMIXCONTIME 环境选项或客户端的 INFORMIXCONTIME 环境变量。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 语法》: *INFORMIXCONTIME* 会话环境选项

《*SinoDB SQL* 指南: 参考》: *INFORMIXCONTIME* 环境变量

LIMITNUMSESSIONS 配置参数

使用 LIMITNUMSESSIONS 配置参数来定义要连接到 SinoDB® 的最大会话数。

如果指定了最大数，您还可指定当会话数接近最大数时，SinoDB® 是否向 online.log 文件打印消息。

如果启用了 LIMITNUMSESSIONS 配置参数，且由于此限制而限制了会话，则连接到任何数据库的常规用户线程和 DBSA 用户线程都会计入此限制。然而，即使已经达到了此限制，还是允许 DBSA 用户连接到服务器。

对服务器的分布式查询也计入此限制。

未打算使用 LIMITNUMSESSIONS 配置参数作为遵守许可协议的方法。

onconfig.std 值

未在 onconfig.std 文件中设置

值

maximum_number_of_sessions = 0 到 2,097,152 (2*1024*1024). 默认是 0。

print_warning = 0（关）或 1（开）。此可选值的默认值是 0。

分隔符

逗号

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果 print_warning 设置为 1，则当会话数大于或等于 maximum_number_of_sessions 值的 95% 时，触发警告。如果 print_warning 设置为零，或未设置，则在达到 maximum_number_of_sessions 限制之后，不发出警告，并且不可打开新的用户会话。

如果 LIMITNUMSESSIONS 配置参数的 maximum_number_of_sessions 值设置为 0，或未设置，则不限制可连接服务器的会话数。

以下示例指明您想要最多 100 个会话连接到服务器，且您想要在已连接会话数接近 100 时打印警告消息。

```
LIMITNUMSESSIONS 100,1
```

上述示例的设置导致当并发连接的会话超过 94 个时，打印警告。仅 DBSA 组成员可在已有 100 个会话连接时启动新的会话。

使用 onmode -wf 或 onmode -wm，或对等的 SQL 管理 API ONMMODE 命令，来动态地增加或暂时地禁用 LIMITNUMSESSIONS 设置。如果数据库服务器达到了 maximum_number_of_sessions 限制，则使用此配置参数来允许运行管理性实用程序。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

LISTEN_TIMEOUT 配置参数

使用 LISTEN_TIMEOUT 配置参数来指定服务器等待连接的秒数。

onconfig.std 值

LISTEN_TIMEOUT 60

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

可设置 LISTEN_TIMEOUT 为一个较低的数，来防止那些可能指示拒绝服务袭击的错误连接要求。

依赖于机器保持线程（数目）的能力，您可配置 MAX_INCOMPLETE_CONNECTIONS 为一个较高的值，并依赖于网络流量，您可设置 LISTEN_TIMEOUT 为一个较低的值来减少袭击达到上限的可能性。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
MAX_INCOMPLETE_CONNECTIONS 配置参数 在第130页
MAX_INCOMPLETE_CONNECTIONS 配置参数 在第130页

LOCKS 配置参数

LOCKS 配置参数指定锁表的初始大小。

锁表为每个锁保留一个条目。如果分配的锁数超过 LOCKS 配置参数的值，则数据库服务器增加锁表的大小。锁表最多可增加 99 次。

onconfig.std 值

LOCKS 20000

值

对于 32位数据库服务器，2,000 到 8,000,000；对于 64 位数据库服务器，2,000 到 500,000,000

单位

内部锁表中的锁数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

数据库服务器通过尝试每次增加锁表一倍来增加锁表的大小。然而，每个增加期间添加的数量受最大值限制。对于 32 位平台，每个增加期间最多可添加 100,000 个锁。因此，对于 32 位平台，允许的最大总锁数是 8,000,000（最大起始锁数）+（99（最大动态锁表扩展数）x 100,000（每个锁表扩展添加的最大锁数）。对于 64 位平台，每个增加期间可最多添加 1,000,000 个锁。因此，最大总锁数是 500,000,000（最大起始锁数）+（99（最大动态锁表扩展数）x 1,000,000（每个锁表扩展添加的最大锁数）。

由于初始锁表存储在常驻内存中且每个额外锁存储在虚拟内存中，如果共享内存的数量有限，则锁会变为资源消耗。单个锁占用的存储量取决于字大小和操作系统，且易于变更。目前，存储量范围从大约 100 字节到 200 字节。通过以不同的 LOCKS 配置参数值重启服务器，和观察常驻池的 "onstat -g mem" 所示的已用内存的增加情况，您可看到支持额外锁所需的存储量。

提示： 当删除一个数据库时，数据库中的每个表将获取并保持一个锁直到数据库删除。

相关链接

onstat -k 命令: 打印活动锁的信息 在第544页
《*SinoDB* 性能指南》: *LOCKS* 配置参数和内存利用率
《*SinoDB* 性能指南》: 锁定
《*SinoDB* 管理员指南》: 共享内存
《*SinoDB SQL* 指南: 语法》: *DROP DATABASE* 语句

LOGBUFF 配置参数

使用 LOGBUFF 配置参数来指定共享内存中三个逻辑日志缓冲区的 KB 大小。

onconfig.std 值

LOGBUFF 64

单位

KB

值

32 到 (32767 * *pagesize* / 1024) 范围中的整数值，此处 *pagesize* 为默认系统页大小。该值必须可被默认系统页大小整除。如果该值无法被该页大小整除，则数据库服务器将向下取最接近可被该页大小整除的值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

在其他缓冲区之一正在清空到磁盘时，三个逻辑日志缓冲区允许用户线程写入活动的缓冲区。如果到活动的缓冲区充满时清空仍未完成，则用户线程开始写到第三个缓冲区。

如果启用 RTO_SERVER_RESTART 配置参数，则设置 LOGBUFF 配置参数值为 256 KB。如果 LOGBUFF 配置参数值少于 256 KB，则当您重启服务器时会看到警告消息。

否则，对于标准工作负载，设置 LOGBUFF 配置参数值为 32 KB；对于重工作负载，设置为 64 KB。数据库服务器使用 LOGBUFF 参数来设置恢复期间使用的内部缓冲区的大小。如果 LOGBUFF 设置的过高，则在恢复期间数据库服务器可能用光内存并关闭。

如果您将用户数据日志记录在智能大对象中，请增加日志缓冲区的大小以使系统更为高效。数据库服务器仅对更改的智能大对象页的部分做日志记录。

通过运行 onstat -l 命令，您可查看关于逻辑日志缓冲区的信息。

相关链接

onstat -l 命令: 显示物理和逻辑日志信息 在第546页

《*SinoDB* 管理员指南》: 确定数据库服务器页大小

《*SinoDB* 管理员指南》: 逻辑日志缓冲区

LOGFILES 配置参数

使用 LOGFILES 配置参数来指定在磁盘初始化期间数据库服务器创建的逻辑日志文件的数目。

onconfig.std 值

LOGFILES 6

值

3 - 32,767（仅整数）

单位

逻辑日志文件数

生效

磁盘初始化期间和您添加新日志文件时。使用 onparms 实用程序之一来添加新日志。

用法

要更改逻辑日志文件数，请添加或删除逻辑日志文件。

如果您使用 onparams 来添加或删除日志文件，则数据库服务器会自动地更新 LOGFILES。

相关链接

onparams 实用程序 在第351页

《*SinoDB* 管理员指南》: 逻辑日志文件的大小

《*SinoDB* 管理员指南》: 手动添加逻辑日志文件

《*SinoDB* 管理员指南》: 删除逻辑日志文件

LOG_INDEX_BUILDS 配置参数

使用 LOG_INDEX_BUILDS 配置参数来启用或禁用索引页日志记录。

onconfig.std 值

未设置。

值

0 = 禁用

1 = 启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果启用 LOG_INDEX_BUILDS，依赖于索引大小，增加逻辑日志文件空间消耗。这可能导致更频繁地要求备份逻辑日志文件。当索引页日志记录状态更改时，向 online.log 文件写入消息。

> RS 辅助服务器提示： 使用 onmode -wm 仅启用或禁用当前会话的索引页日志记录，并不影响 onconfig 文件中的设置。如果服务器停止并重启，则 onconfig 文件中的设置决定是否启用索引页日志记录。 因此，当使用 RD 辅助服务器时，不建议使用 onmodem -wm 启用索引页日志记录；而是使用 onmode -wf 来更新 onconfig 文件，以便在重启之后启动索引页日志记录。当使用 RS 辅助服务器时，需要索引页日志记录。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

LOG_STAGING_DIR 配置参数

使用 LOG_STAGING_DIR 配置参数来指定当在 RS 辅助服务器上配置延迟的日志文件应用时，从主服务器收到的日志文件的位置。

onconfig.std 值

未设置。

值（第一个参数）

任何有效的安全目录。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

LOG_STAGING_DIR 配置参数指定，在下列情况下，从主服务器发送来的日志文件存储的目录：

- 在 RS 辅助服务器上设置 DELAY_APPLY 来延迟日志的应用
- 在 RS 辅助服务器上设置 STOP_APPLY 配置参数来停止日志的应用
- RS 辅助服务器必须临时地缓冲日志
- 在 HDR 辅助服务器上设置 LOG_INDEX_BUILDS 参数，且 HDR 辅助服务器正在处理检查点

延迟日志文件的应用允许您通过从 RS 辅助服务器恢复数据来快速地从错误的数据库修改中恢复。

LOG_STAGING_DIR 配置参数指定的目录必须是安全的。该目录必须是用户 informix 所拥有，必须属于组 informix，且必须没有公开读、写或执行权限。

该目录应该有足够空间以保持所有暂存的逻辑日志。 选择一个至少能够存储主服务器上逻辑日志总计大小两倍的目录。要估算存储大小，请将 LOGBUFF 配置参数的值与 LOGFILES 配置参数的值相乘，然后再加倍。

要查看有关送到为 RS 辅助服务器设置的日志暂存目录的数据的信息，请在 RS 辅助服务器上运行 onstat -g rss verbose 命令。

如果写到暂存文件失败，则 RS 辅助服务器发出事件报警 40007。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
DELAY_APPLY 配置参数 在第95页
STOP_APPLY 配置参数 在第182页
《*SinoDB* 管理员指南》: 延迟应用日志记录
《*SinoDB* 管理员指南》: 远程独立辅助服务器
DELAY_APPLY 配置参数 在第95页

LOGSIZE 配置参数

使用 LOGSIZE 配置参数来指定逻辑日志文件创建时使用的大小。

onconfig.std 值

LOGSIZE 10000

单位

KB

值

整数值。

最小值 = 200

数据库服务器首次初始化时，最大值 = (ROOTSIZE - PHYSFILE - 512 - (63 * *pagesize*/1024)) / LOGFILES

pagesize 值为操作系统的默认系统页大小。

如果您扩展根数据库空间或将逻辑日志移到不同的数据库空间，则逻辑日志文件大小的最大值不能超过以下页大小相依的值：

- 1 GB，当页大小 = 2 KB
- 2 GB，当页大小 = 4 KB

此限制是日志位置可以为那些页大小描述的页的最大数目。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当您更改了 LOGSIZE 配置参数时，仅新的日志文件受影响。现有的日志文件大小不更改。逻辑日志的合计大小是 LOGSIZE 配置参数设置与 LOGFILES 配置参数值的乘积。然而，如果您更改 LOGSIZE 配置参数的值，则所有逻辑日志文件的合计大小取决于每个大小的日志文件数。

如果启用 AUTO_LLOG 配置参数，则根据需要自动地添加逻辑日志文件来提高性能，直到逻辑日志大小合计达到可配置最大值。

要验证数据库服务器在平台上使用的页大小，请运行 onstat -b 命令。

如果您声明智能大对象列的日志记录，则您必须确保此逻辑日志比插入或更新期间日志记录的数据量大许多。数据库服务器无法备份打开的事务。如果许多事务处于活动的，则合计日志记录活动不得强制打开的事务到日志备份文件。例如，如果日志大小为 1000 KB 且高水印是 60%，则不要使用超过 600 KB 的逻辑日志进行智能大对象更新。当达到高水印 600 KB 时，数据库服务器启动回滚事务

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

LTXHWM 配置参数 在第129页

onparams -p: 更改物理日志参数 在第353页

《*SinoDB* 管理员指南》*:* 确定数据库服务器页大小

《*SinoDB* 管理员指南》*:* 逻辑日志文件的大小

《*SinoDB* 管理员指南》*:* 估计日志文件的大小和数量

LOW_MEMORY_MGR 配置参数

使用 LOW_MEMORY_MGR 配置参数来启用自动低内存管理，当达到内存限制时，您可用来更改主服务器或标准服务器的默认行为。

onconfig.std 值

LOW_MEMORY_MGR 0

值

1 = 当数据库服务器启动时，启用自动低内存管理。

0 = 禁用自动低内存管理。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

如果您配置主服务器或标准服务器使用 SHMTOTAL 配置参数值的某个百分比作为自动低内存管理启动和停止阈值，则 SHMTOTAL 配置参数必须设置为正整数。

注意： 更改 SHMTOTAL 配置参数的值可导致自动低内存管理的配置变得无效，强制数据库服务器使用默认设置。

要启用自动低内存管理，请指定：

```
LOW_MEMORY_MGR 1
```

相关链接

SHMTOTAL 配置参数 在第168页

scheduler lmm enable 参数*:* 指定自动低内存管理设置 *(SQL* 管理 *API)* 在第677页

scheduler lmm disable 参数*:* 停止自动低内存管理 *(SQL* 管理 *API)* 在第679页

onstat -g lmm 命令*:* 显示低内存管理信息 在第475页

SHMTOTAL 配置参数 在第168页

《*SinoDB* 管理员指南》*:* 为关键活动保留内存

LOW_MEMORY_RESERVE 配置参数

使用 LOW_MEMORY_RESERVE 配置参数来保留一个特定的内存量，作为关键活动需要而服务器的空闲内存有限时使用。

如果通过将 LOW_MEMORY_RESERVE 配置参数设置为一个特定的 KB 值来启用新的 LOW_MEMORY_RESERVE 配置参数，则即使您收到内存用尽的错误，诸如回滚活动这样的关键活动仍可完成。

onconfig.std 值

LOW_MEMORY_RESERVE 0

值

0 或 128 - 2147483648，虽然最大值不可高于 SHMVIRTSIZE 配置参数值的 20%

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

不管如何设置 LOW_MEMORY_RESERVE 配置参数，保留的内存大小的最大值是 SHMVIRTSIZE 配置参数的值的 20%。

例如，要保留 512 KB 内存，请指定：

```
LOW_MEMORY_RESERVE 512
```

可使用 onstat -g seg 命令来查看低内存保留信息。其输出包括保留内存的大小、服务器使用保留内存的次数，和所需的最大内存。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页

SHMVIRTSIZE 配置参数 在第169页

LTXEHWM 配置参数

使用 LTXEHWM 配置参数来指定长事务、排他访问、高水印。当逻辑日志空间达到 LTXEHWM 阈值时，则当前正在回滚的长事务将获得逻辑日志的排他访问权。

onconfig.std 值

LTXEHWM 80

如不存在

90（如果 DYNAMIC_LOGS 设置为 1 或 2）60（如果 DYNAMIC_LOGS 设置为 0）

值域

LTXHWM 至 100

单位

百分比

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果当事务达到长事务高水印时，事务不提交或回滚，则这个事务是长事务。

如果在回滚完成前您的系统耗尽了日志空间，则请降低 LTXEHWM 值。

如果不想要添加过多的逻辑日志，则应将 LTXEHWM 设置为较小的值（大约 60）。如果关闭动态日志记录（DYNAMIC_LOGS = 0），则应将 LTXEHWM 设置为更低的值（大约 50）来避免耗尽逻辑空间。

> 提示: 要允许用户继续存取逻辑日志，即使在长事务回滚期间，请将 LTXEHWM 设置为 100。将 DYNAMIC_LOGS 设置为 1 或 2，以便数据库服务器可以添加足够的日志文件数来防止长事务挂起并允许长事务回滚。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DYNAMIC_LOGS 配置参数 在第109页

LTXHWM 配置参数 在第129页

《*SinoDB* 管理员指南》: 控制长事务

LTXHWM 配置参数

使用 LTXHWM 配置参数来指定长事务高水印。长事务高水印是可用日志空间的百分比，当填满时，触发数据库服务器来检查长事务。

onconfig.std 值

LTXHWM 70

如不存在

80（如 DYNAMIC_LOGS 设置为 1 或 2）50（如 DYNAMIC_LOGS 设置为 0）

值

1 - 100

单位

百分比

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当逻辑日志空间达到 LTXHWM 阈值，数据库服务器开始回滚事务。如果您降低 LTXHWM 值，则日志文件大小或数目增加来降低回滚的可能性。

如果 DYNAMIC_LOGS 设置为 1 或 2，则数据库服务器可添加足够的日志文件数来完成事务或防止长事务回滚挂起。

如果不想要添加过多的逻辑日志，则应将 LTXHWM 设置为较小值（大约 60）。如果关闭动态日志记（DYNAMIC_LOGS = 0），则应将 LTXHWM 设置为更更低的值（大约 50）以避免耗尽逻辑空间。

> 警告: 如果将 LTXHWM 和 LTXEHWM 都设置为 100，则再也不终止长事务。虽然您可为了对自己有利而使用这种配置，但为了正常的数据库服务器运作，您应将 LTXHWM 设置为低于 100。

如果将 LTXHWM 设置为 100，则数据库服务器会发出警告消息：

```
LTXHWM is set to 100%. This long transaction high water mark
will never be reached. Transactions will not be aborted automatically
by the server, regardless of their length.
```

如果事务挂起，则请遵照《*SinoDB*® 管理员指南》中管理逻辑日志文件章节的指导，从长事务挂起恢复。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DYNAMIC_LOGS 配置参数 在第109页

LTXEHWM 配置参数 在第128页

《*SinoDB* 管理员指南》：控制长事务

LOGSIZE 配置参数 在第126页

MAX_FILL_DATA_PAGES 配置参数

使用 MAX_FILL_DATA_PAGES 配置参数来控制向有变长行的页插入更多行。

onconfig.std 值

MAX_FILL_DATA_PAGES 0

值

0 或 1

单位

整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

将 MAX_FILL_DATA_PAGES 值设置为 1 ，来允许在有变长行的表中的每页插入更多行。此设置可减少磁盘空间，更有效率的使用缓冲池，和减少表扫描次数，

当 MAX_FILL_DATA_PAGES 为启用的，服务器将添加一个新行到带有现在行的最近更改的页，如果添加该行将至少保留该页的 10% 空闲作为该页中所有行的未来扩展。如果未设置 MAX_FILL_DATA_PAGES，则服务器仅当页上有足够空间时才添加新行，来允许新行增长到其最大长度。

启用 MAX_FILL_DATA_PAGES 并允许每页更多的变长行可能带来的问题，就是服务器可能以不同物理顺序来存储行。而且，当页填满时，对行中变长列的更改可能导致行扩展，以致于该行不再能够完全地置入该页。这会导致服务器将该行分裂到两个页上，增加该行的存取时间。

要利用此设置，必须重新加载带变长行的现有表，货币修改现有页，然后再进行插入。

相关链接

《*SinoDB* 性能指南》：减少具有可变长度行的表中的磁盘空间

MAX_INCOMPLETE_CONNECTIONS 配置参数

使用 MAX_INCOMPLETE_CONNECTIONS 配置参数来指定会话中未完成连接的最大数。

onconfig.std 值

MAX_INCOMPLETE_CONNECTIONS 1024

单位

未完成连接的数目

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

在达到 MAX_INCOMPLETE_CONNECTIONS 配置参数中指定的数目之后，在联机消息日志中写入错误消息，说明服务器可能遭受拒绝服务袭击。另请参阅有关 LISTEN_TIMEOUT 配置参数的信息，该参数指定服务器等待连接的秒数。

依赖于机器保持线程（数目）的能力，您可配置 MAX_INCOMPLETE_CONNECTIONS 一个较高的值。依赖于网络流量，您还可设置 LISTEN_TIMEOUT 配置参数一个较低的值来降低袭击达到最大限制的可能性，该参数指定服务器等待连接的秒数

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

LISTEN_TIMEOUT 配置参数 在第122页

LISTEN_TIMEOUT 配置参数 在第122页

MAX_PDQPRIORITY 配置参数

使用 MAX_PDQPRIORITY 配置参数来限制数据库服务器可分配给任一 DSS 查询的 PDQ 资源。

onconfig.std 值

MAX_PDQPRIORITY 100

值

0 = 关闭 PDQ。DSS 查询不使用并行。

1 = 从分段表并行取得数据（并行扫描），但不使用其他形式的并行。

2 - 100 = 设置实际分配给查询的用户要求的 PDQ 资源百分比。100 表示使用所有可用资源来并行处理查询。

生效

在编辑 onconfig 文件并重启数据库服务器之后的所有用户会话上。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

MAX_PDQPRIORITY 是用来衡量用户设置的 PDQ 优先级的值的因素。例如，假设数据库管理员设置 MAX_PDQPRIORITY 为 80。如果某个用户设置 PDQPRIORITY 环境变量为 50，然后发出查询，则数据库服务器将默默地以 PDQ 优先级 40 处理该查询。

可使用 onmode 实用程序在数据库服务器处于联机状态时更改 MAX_PDQPRIORITY 的值。

在 SinoDB® 中，PDQ 资源包含内存、CPU、磁盘 I/O 和扫描线程。MAX_PDQPRIORITY 允许数据库管理员使用 OLTP 并发地运行决策支持，而不会使 OLTP 性能降低。然而，如果 MAX_PDQPRIORITY 设置的太低，决策支持查询的性能可能会下降。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onstat -g mgm 命令*:* 显示 *MGM* 资源信息 在第480页

《*SinoDB* 性能指南》*:* 并行数据库查询 *(PDQ)*

《*SinoDB SQL* 指南: 参考》: *PDQPRIORITY* 环境变量

MIRROR 配置参数

使用 MIRROR 配置参数来启用或禁用数据库服务器的镜像。

onconfig.std 值

MIRROR 0

值

0 = 禁用镜像

1 = 启用镜像

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

建议作为初始化的一部分制作根数据库空间和关键数据的镜像。否则，禁用镜像。如果之后决定要添加镜像，您可编辑配置文件来更改参数值。

高可用性数据复制对中的两台数据库服务器上的 MIRROR 配置参数不必设置为相同的值。您可在主数据库服务器上或辅助数据库服务器上独立启用或禁用镜像。除非使用镜像，否则不要将 MIRROR 配置参数设置为 1。

相关链接

onstat -d 命令: 显示块信息 在第414页

《*SinoDB* 管理员指南》: 镜像

MIRROROFFSET 配置参数

在 SinoDB® 中，MIRROROFFSET 指定磁盘分区或设备的偏移量，以到达作为根数据库空间初始块的镜像的块。

onconfig.std 值

MIRROROFFSET 0

值

任何大于或等于 0 的值

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 管理员指南》: 初始化期间为根数据库空间制作镜像

MIRRORPATH 配置参数

使用 MIRRORPATH 配置参数来指定根数据库服务器初始块的镜像块的全路径名。

onconfig.std 值

在 UNIX™ 上: $INFORMIXDIR/tmp/demo_on.root_mirror

在 Windows™ 上: 无

值

65 个或更少的字符

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

MIRRORPATH 应是一个指向实际镜像块的块路径名的链接，与将 ROOTPATH 指定为链接的原因相同。同样地，为镜像块选择一个简短路径名。

您必须将 MIRRORPATH 所指定的文件的权限设置为 660。拥有者和组都必须是 informix。

在 UNIX™ 平台上，如果您使用原始磁盘空间作为镜像块，那么建议您定义 MIRRORPATH 为指向镜像数据库空间的初始块的链接，而不是该初始块的实际设备名。

要在未启动镜像功能的数据库服务器上启动镜像数据：

1. 将数据库服务器脱机。
2. 将 MIRROR 配置参数更改为 1 并且让 MIRRORPATH 配置参数为空白。
3. 将数据库服务器联机。
4. 为镜像块分配磁盘空间。您可以在任何时间分配该磁盘空间，然而，当您在下一步骤中指定镜像块时，该磁盘空间必须是可用的。该镜像块必须与对应的主块位于不同磁盘。
5. 使用 onspaces -m 选项来启动数据库空间、BLOB 空间或智能大对象空间的镜像。您必须从根数据库空间开始。成功地运行根数据库空间的指令之后，服务器自动地设置 MIRRORPATH 值。

相关链接

《SinoDB 管理员指南》：初始化期间为根数据库空间制作镜像

《SinoDB 管理员指南》：管理磁盘空间

MSG_DATE 配置参数

使用 MSG_DATE 配置参数来启用以 MM/DD/YY 格式将日期插入到打印至联机日志的每个消息的开头。

onconfig.std 值

不在 onconfig.std 文件中。

值

0 = 关（默认）

1 = 开

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

下面示例中，MSG_DATE 设置为 1（开）。

```
04/10/11 10:26:06  Value of MSG_DATE has been changed to 1.
04/10/11 10:27:35  Value of MSG_DATE has been changed to 1.
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

MSGPATH 配置参数

使用 MSGPATH 配置参数来指定消息日志文件的全路径名。数据库服务器在运行期间将状态消息和诊断消息写到此文件中。

onconfig.std 值

在 UNIX™ 邵：$INFORMIXDIR/tmp/online.log

在 Windows™ 上：%INFORMIXDIR%\online.log

在 Windows™ 上，如果您在安装期间创建服务器实例：%INFORMIXDIR%*server_name*.log。*server_name* 是程序组中的服务器名和 INFORMIXSERVER 环境变量的值。

值

online.log 文件的路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果 MSGPATH 指定的文件不存在，则数据库服务器会在指定的目录创建该文件。如果 MSGPATH 指定的目录不存在，则数据库服务器会向系统控制台发送消息。

如果 MSGPATH 指定的文件存在，则数据库服务器打开该文件，并将发生的消息追加到该文件。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
数据库服务器日志中的消息 在第776页

MULTIPROCESSOR 配置参数

使用 MULTIPROCESSOR 配置参数来指定数据库服务器是否以一种适合于单处理器计算机或多处理器计算机的方式执行锁定。

如果 MULTIPROCESSOR 设置为 0，则忽略设置处理器专用的参数。

onconfig.std 值

MULTIPROCESSOR 0

值

0 = 无多处理器

1 = 多处理器可用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 管理员指南》: *CPU* 虚拟处理器

NET_IO_TIMEOUT_ALARM 配置参数

使用 NET_IO_TIMEOUT_ALARM 配置参数来控制是否在网络写入操作被阻塞达 30 分钟或更久时收到通知。

阻塞的网络写入操作通常表明操作系统问题。使用 NET_IO_TIMEOUT_ALARM 配置参数来为特定网络流量类型启用事件报警 82。

onconfig.std 值

不在 onconfig.std 中

值

下列值之一，或者一个或多个下列值的总计：

- 0 = 禁用
- 1 = 对 Enterprise Replication 操作启用
- 2 = 对分布式查询启用
- 4 = 对 HDR 操作启用
- 8 = 对 SMX 操作启用
- 16 = 对其他组件操作启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

NETTYPE 配置参数

使用 NETTYPE 参数来调整 sqlhosts 信息中定义的网络协议。

onconfig.std 值´

UNIX™：ipcshm,1,50,CPU

Windows™：未设置。

默认值

connection_type,1,50,*vp_class*

默认连接类型取决于操作系统：

- UNIX™：protocol 字段的值来自 sqlhosts 文件。
- Windows™：onsoctcp

虚拟处理器类的默认类型取决于 sqlhosts 文件中的 dbservername 条目：

- CPU，如果 dbservername sqlhosts 条目是由 DBSERVERNAME 配置参数定义。
- NET，如果 dbservername sqlhosts 条目是由 DBSERVERALIASES 配置参数定义。

分隔符

以逗号分隔字段。不可包含空白。您可省略字段的值，但必须包含每个字段的逗号。然而，可省略尾部逗号。

值

请参阅 "用法" 部分。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

NETTYPE 参数为与 sqlhosts 信息中 dbservername 条目关联的协议和接口组合提供调整选项。在 onconfig 文件中的 DBSERVERNAME 配置参数或 DBSERVERALIASES 配置参数上定义 sqlhosts 信息中的每个 dbservername 条目。

NETTYPE

connection_type ,
[{ 1 *poll_threads* }]
,
[{ 50 *conn_per_thread* }]
,
[{ CPU | NET }]

表 67: NETTYPE 配置参数值的选项

字段	值
connection_type	有效协议与接口组合，带有或不带有数据库服务器前缀 on、ol 或 dr。
poll_threads	指定给该连接类型的轮询线程数。默认是 1。值域取决于操作系统和虚拟处理器类： • UNIX™：如果虚拟处理器类是 NET，则是一个大于或等于 1 的整数。每个轮询线程需要一个单独的虚拟处理器，因此当您指定接口/协议组合的轮询线程数并指定由某个网络 VP 运行它们时，您间接地指定了网络虚拟处理器数。 • UNIX™：如果虚拟处理器类是 CPU，则是一个从 1 到 CPU VP 的数目的整数 • Windows™：一个大于或等于 1的整数。 如果数据库服务器有多个连接，那么您可能可通过增加轮询线程数来提升性能。通常，每个轮询线程可处理大约 200 - 250 个连接。 Windows：如果指定 soctcp 协议，则仅创建一个轮询线程，反而通过 NETTYPE 参数指定的每个轮询线程是在其自身的 SOC VP 中创建套接字 I/O 线程（soctcpio）。 套接字 IO 线程使用 IO 完成端口来接收完成通知，为所有连接处理接收操作。这些线程在 Windows™ 平台上执行服务网络连接的大部分工作。
conn_per_thread	从 1 到 32767 的一个整数，设置 每个轮询线程的最大连接数。默认是 50。 对于共享内存连接，*conn_per_thread* 的值是每一线程的最大连接数。通常，指定预期连接数的两倍。 对于网络连接，可超出 *conn_per_thread* 的值。根据需要，轮询线程动态地重新分配资源来支持更多连接。避免设置并发连接数的值远高于您的预期。否则，您可能浪费系统资源。 如果仅有少数连接并发地使用一个协议，您可通过明确地设置连接的预估数来节省内存。
CPU	指定一个 CPU 虚拟处理器。配置要在每一 CPU 虚拟处理器中运行的共享内存连接。
NET	指定使用适当的网络虚拟处理器：SOC、STR、SHM 或 TLI。配置要在网络虚拟处理器中运行的网络连接。

您可为想要数据库服务器使用的每一协议指定一个 NETTYPE 参数。

以下示例说明到数据库服务器的两个连接类型的 NETTYPE 参数：本地客户端的共享内存连接，和使用套接字的网络连接：

```
NETTYPE ipcshm,3,,CPU
NETTYPE soctcp,8,300,NET
```

共享内存连接（ipcshm）的 NETTYPE 参数指定要在 CPU 虚拟处理器中运行的三个轮询线程。连接数未指定，于是设置为 50。对于 ipcshm，轮询线程数对应于内存段的数目。

对于套接字连接（soctcp）的 NETTYPE 参数指定此协议的每一线程预期 300 个同时发生的连接，且指定在网络虚拟处理器中运行 8 个轮询线程。

UNIX™：NETTYPE 和 NUMFDSERVERS 配置参数设置之间可有依赖性。当有多个 CPU 虚拟处理器和轮询线程，且 onstat -g ath 命令的线程状态输出表明网络共享文件（NSF）锁定时，您可增加轮询线程的 NUMFDSERVERS 值来降低 NSF 锁争用。

相关链接

DBSERVERNAME 配置参数 在第90页
DBSERVERALIASES 配置参数 在第88页
NUMFDSERVERS 配置参数 在第138页
VPCLASS 配置参数 在第196页
The number of configured inline poll threads exceeds the number of CPU virtual processors. 在第799页
Virtual processor limit exceeded. 在第813页
NUMFDSERVERS 配置参数 在第138页
VPCLASS 配置参数 在第196页
onstat -g nsc 命令: 打印当前共享内存连接信息 在第483页
onstat -g nsd 命令: 打印轮询线程的共享内存数据 在第486页
onstat -g nss 命令: 打印共享内存网络连接的状态 在第486页
VPCLASS 配置参数 在第196页
《*SinoDB* 性能指南》: *Specifying the number of connections and poll threads*
《*SinoDB* 管理员指南》: 指定连接数和轮询线程数
《*SinoDB* 管理员指南》: 指定网络虚拟处理器的数量
《*SinoDB* 管理员指南》: *NETTYPE* 配置参数中设置的连接信息
《*SinoDB* 管理员指南》: *sqlhosts* 连接信息
《*SinoDB* 管理员指南》: *CPU* 虚拟处理器
《*SinoDB* 管理员指南》: 网络虚拟处理器

NS_CACHE 配置参数

使用 NS_CACHE 配置参数来定义 SinoDB® 名称服务高速缓存中条目的最大保留时间：主机名/IP地址高速缓存、服务高速缓存、用户高速缓存，和组高速缓存。

onconfig.std 值

NS_CACHE host=900,service=900,user=900,group=900

值

每一字段的值为等于或大于 0 的整数。

host = 设置主机名或 IP 地址高速缓存中高速缓存信息的秒数。

service = 设置服务高速缓存中高速缓存信息的秒数。

user = 设置用户高速缓存中高速缓存信息的秒数。

group = 设置组高速缓存中高速缓存信息的秒数。

0 = 禁用高速缓存。服务器总是从操作系统取得信息。您可设置个别高速缓存为 0 或设置所有名称服务高速缓存为 0：NS_CACHE 0。

单位

秒

分隔符

以逗号分隔值。不可包含空白。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

为了查找和解析主机名（或 IP 地址）、服务名、用户（和密码）或组，数据库服务器使用适当的系统调用查询操作系统。可通过使用 SinoDB® 名称服务高速缓存机制，您可以避免许多的这些 OS 查找，该机制在可配置的时间量中保存并重复使用每一检索的信息条。如果您的操作系统不提供自身的高速缓存，则您应设置 NS_CACHE 配置参数。

服务器从高速缓存中取得信息比在查询操作系统时快。然而，如果您通过将保留时间设置为 0 来禁用一个或多个高速缓存，则数据库服务器将为主机、服务、用户或组信息来查询操作系统。

在操作系统级别对名称服务进行的更改不会立即反映在 SinoDB® 名称服务高速缓存：例如，更改 IP 地址、添加用户到组或从组中移除用户，或新的密码。然而，您可使用 onmode -wf 或 onmode -wm 命令来立即更改 NS_CACHE 信息。当您用 onmode -wf 或 onmode -wm 命令更改一个特别的高速缓存的值时，服务器立即终止那个高速缓存中所有现有的条目。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》：在 *NS_CACHE* 配置参数中设置的名称服务最大保留时间

《*SinoDB* 性能指南》：提高连接性能和可伸缩性

NUMFDSERVERS 配置参数

对于 UNIX™ 上的网络连接，使用 NUMFDSERVERS 配置参数来指定用于管理 SinoDB® 和虚拟处理器（VP）之间网络连接迁移的轮询线程的最大数。

如果 SinoDB®有高比率的新连接与断开请求，或者您发现网络共享文件（NSF）锁定之间有大量争用，则指定 NUMFDSERVERS 信息是很有用的。可使用 onstat -g ath 命令来显示有关所有线程的信息。该信息包含状态，例如 mutex wait nsf.lock 表示有大量的 NSF 锁争用。

onconfig.std 值

NUMFDSERVERS 4（每个 nettype 仅前四个轮询线程涉及连接迁移管理。）

值

1 - 50

实际数量取决于 NETTYPE 配置参数所指定的轮询线程数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

NUMFDSERVERS 所指定的值对于共享内存（SHM）连接无影响。

如果您使用 NUMFDSERVERS 配置参数，请视需要，检视与更改 NETTYPE 配置参数中的轮询线程数。例如，如果您有多个 CPU VP 和轮询线程，而这导致 NSF 锁定，那么您可以增加 NUMFDSERVERS 的值和轮询线程来减少 NSF 锁定争用。

相关链接

NETTYPE 配置参数 在第135页

DBSERVERNAME 配置参数 在第90页

DBSERVERALIASES 配置参数 在第88页

NETTYPE 配置参数 在第135页

onstat -g ath 命令: 显示所有线程的信息 在第426页
《*SinoDB* 性能指南》: 提高连接性能和可伸缩性

OFF_RECVRY_THREADS 配置参数

使用 OFF_RECVRY_THREADS 配置参数来指定冷恢复或快速恢复期间逻辑恢复所使用的恢复线程数。

onconfig.std 值

OFF_RECVRY_THREADS 10

值

正整数

单位

并行运行的恢复线程数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

在执行冷恢复之前，可将此参数的值设置为约等于逻辑日志中具有大量事务的表的数量。对于单处理器的计算机或节点，多于 30 到 40 个线程可能太多，因为线程管理和内存的开销会抵消并行处理所带来的好处。

每当逻辑恢复开始时，数据库服务器都会为恢复线程创建 LGR 内存池。LGR 内存池大小约等于 OFF_RECVRY_THREADS * 100 KB。这个池在快速恢复和冷恢复期间使用。勿将 OFF_RECVRY_THREADS 配置参数设置为会导致数据库服务器尝试为 LGR 内存池分配比系统上可用内存更多的内存的值。

在高可用性集群中，辅助服务器几乎总是处于快速恢复模式。在辅助服务器上，将 OFF_RECVRY_THREADS 配置参数设置为同时考虑前滚性能与内存使用状况的值。

相关链接

《*SinoDB* 性能指南》: *OFF_RECVRY_THREADS* 和 *ON_RECVRY_THREADS* 及其对快速恢复的影响
《*SinoDB* 备份和还原指南》: *onbar -r* 语法: 复原数据

ON_RECVRY_THREADS 配置参数

ON_RECVRY_THREADS 配置参数是数据库服务器联机时（热恢复期间）数据库服务器用于逻辑恢复的恢复线程的最大数。

onconfig.std 值

ON_RECVRY_THREADS 1

值

正整数

单位

并行运行的恢复线程数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

参考

- 《*SinoDB*® 备份和还原指南》
- 《*SinoDB*® 性能指南》

用法

可将 ON_RECVRY_THREADS 调整为可能要恢复的表数，因为恢复期间处理的逻辑日志记录是按表号分配线程。当恢复线程数与正在恢复的表数相匹配时发生最大程度的并行处理。

要提升热恢复性能，请使用 ON_RECVRY_THREADS 参数增加快速恢复线程数。

相关链接

《*SinoDB* 性能指南》: *OFF_RECVRY_THREADS* 和 *ON_RECVRY_THREADS* 及其对快速恢复的影响

《*SinoDB* 备份和还原指南》: *onbar -r* 语法: 复原数据

ONDBSPACEDOWN 配置参数

使用 ONDBSPACEDOWN 配置参数来定义当非关键数据库空间中的主块上发生任何禁用事件时，数据库服务器采取的操作。

onconfig.std 值

ONDBSPACEDOWN 2

值

0 = 数据库服务器将数据库空间标记为脱机并继续运行。

1 = 数据库服务器终止。

2 = 数据库服务器向日志写入块的状态并等待用户输入。如果设置此选项，但想要数据库服务器将已禁用的数据库空间标记为关闭并继续处理，请使用 onmode -O 来覆盖此 ONDBSPACEDOWN 设置。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -O: 重设 *ONDBSPACEDOWN WAIT* 方式 在第342页

《*SinoDB* 管理员指南》: 监视数据库服务器是否有禁用 *I/O* 错误

当不应用 ONDBSPACEDOWN 时的数据库服务器的行为

如果任何关键数据库空间（例如，rootdbs 或 logsdbs）内的块丢失，数据库服务器将无法联机。

ONDBSPACEDOWN 值不影响临时数据库空间。对于临时数据库空间，无论 ONDBSPACEDOWN 设置如何，数据库服务器都会继续处理。如果需要修复临时数据库空间，则您应删除并重新创建它。

对于非关键数据库内的非主块，当禁用事件发生时，数据库服务器的行为取决于块的事务状态：

- 无事务：如果未检测到那个块的事务，则单独地将该块标记为关闭。在这种情况下，后续对那个块的写入尝试失败，则回滚相关事务。您可以安全地将该块放回，然后使用 onspaces -s 实用程序将该块标记为 back online。
- 检测到事务：如果有要前滚或回滚的事务，则数据库服务器将以适当的快速恢复错误来终止。在这种情况下，您应将该块放回并重启数据库服务器。

ONLIDX_MAXMEM 配置参数

使用 ONLIDX_MAXMEM 来限制分配给单个 *preimage* 池和单个 *updator* 日志池的内存量。

onconfig.std 值

ONLIDX_MAXMEM 5120

值

16 - 4294967295

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

preimage 和 updator 日志池，pimage_*partnum* 和 ulog_*partnum*，是在执行 CREATE INDEX ONLINE 语句时创建的共享内存池。当该语句的执行完成时，将释放这些池。

如果您指定了该参数的值，然后创建一个表，向该表添加行，并开始在某列上执行 ´ CREATE INDEX ONLINE 语句，则您还可在该列上执行其他操作，例如 UPDATE STATISTICS HIGH，而不会有内存问题。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

OPTCOMPIND 配置参数

使用 OPTCOMPIND 来指定帮助优化程序选择适用于应用程序的查询计划的信息。

提示: 可以认为变量名来自 "OPTimizer（优化程序） COMPare（比较使用）INDexes（索引与其他方式的成本）。"

onconfig.std 值

OPTCOMPIND 2

值

0 = 当每个有序表对存在适当的索引时，优化程序在不考虑成本的情况下，选择索引扫描（嵌套循环联接）而不是表扫描（散列联接）。

1 = 如果隔离级别不是 Repeatable Read，则优化程序使用成本来决定执行路径。否则，优化程序选择索引扫描（与值 0 的行为相同）。建议采用此设置以优化性能。

2 = 优化程序使用成本来决定任何隔离级别的执行路径。索引扫描不会优先于表扫描；优化程序完全根据成本做出决定。如果未设置该变量，则此值为默认值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

由于散列联接的性质，因此使用 Repeatable Read 隔离级别的应用程序可能会暂时地锁定每个有序表组中涉及联接的表中的所有记录（即便那些记录不符合联接条件）。此状况导致连接之间更激烈的争用。相反地，嵌套循环联接锁定较少的记录，但当数据库服务器检索大量行时性能较差。客户端应用程序也可影响优化程序对联接方式的选择。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 参考》: *OPTCOMPIND* 环境变量

《*SinoDB SQL* 指南: 语法》: *OPTCOMPIND* 会话环境选项

OPT_GOAL 配置参数

使用 OPT_GOAL 配置参数来指定查询的优化目标。

onconfig.std 值

OPT_GOAL -1

值

0 or -1

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

值 0 将优化目标设置为 FIRST_ROWS。值 -1（默认值）将优化目标设置为 ALL_ROWS。

当将优化目标设置为 FIRST_ROWS 时，就指定想要数据库服务器针对所感觉的响应时间来优化查询。换句话说，交互式应用程序的用户所感觉到的响应时间就是数据显示到屏幕上所花费的时间。将优化目标设置为 FIRST_ROWS 会将数据库服务器配置为返回满足查询的前几行数据。

当将优化目标设置为 ALL_ROWS 时，就指定想要数据库服务器优化查询的总执行时间。设置 ALL_ROWS 优化目标将指示数据库服务器尽可能快地处理整个查询，而不管需要花多久时间将前面几行返回给应用程序。

可使用以下四种方式之一来指定优化目标：

- 通过查询（SELECT 语句）

  使用 ALL_ROWS 和 FIRST_ROWS 指令
- 通过会话

  使用 SET OPTIMIZATION 语句
- 通过环境

  设置 OPT_GOAL 环境变量
- 通过数据库服务器

  设置 OPT_GOAL 配置参数

上面的列表按优先级降序列出设置此目标的机制。 为确定优化目标，数据库服务器以上面的顺序来检查这些设置。所遇到的第一个设置决定优化目标。例如，如果查询包含 ALL_ROWS 指令但 OPT_GOAL 配置参数设置为 FIRST_ROWS，则数据库服务器按照该查询指定的 ALL_ROWS 来进行优化。

相关链接

《SinoDB SQL 指南: 参考》: OPT_GOAL 环境变量 (UNIX)

《SinoDB SQL 指南: 语法》: 优化目标指令

《SinoDB 性能指南》: 优化目标指令

PC_HASHSIZE 配置参数

使用 PC_HASHSIZE 来指定数据库服务器所使用的高速缓存中散列存储区的数量。PC_HASHSIZE 仅应用于 UDR 高速缓存。

onconfig.std 值

PC_HASHSIZE 31

值

任何正整数。建议是质数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

PC_POOLSIZE 配置参数

使用 PC_POOLSIZE 配置参数来指定存储在 UDR 高速缓存中的用户定义例程的最大数量。

onconfig.std 值

PC_POOLSIZE 127

值

127 或以上的正值，表示高速缓存中条目的初始最大数量的一半。 最大值取决于共享内存配置和服务器实例的可用共享内存。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wm 命令来增加内存中的值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

高速缓存中的初始条目数是 PC_POOLSIZE 配置参数值的两倍。例如，如果 PC_POOLSIZE 设置为 127，则高速缓存中允许有 254 个条目。如果高速缓存中的所有条目均已满，那么高速缓存大小将自动增大 10%。要降低高速缓存的大小，请减小 onconfig 文件中 PC_POOLSIZE 配置参数的值并重新启动服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

PHYSBUFF 配置参数

使用 PHYSBUFF 配置参数来指定共享内存中两个物理日志缓冲区的大小（以 KB 为单位）。

onconfig.std 值

PHYSBUFF 128

单位

KB

值

从 4 到 (32767 * 页大小 / 1024) 的整数，其中 页大小是默认系统页大小。该值必须可被默认系统页大小整除。如果该值不能被该页大小整除，那么数据库服务器向下取可被该页大小整除的最接近的值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

双缓冲允许用户线程在另一缓冲区正在清空到磁盘上的物理日志时写到活动的物理日志缓冲区中。对物理日志缓冲区的写入长度正好是一页。PHYSBUFF 参数值决定数据库服务器需要将物理日志缓冲区清空到物理日志文件的频率。

如果启用了 RTO_SERVER_RESTART 配置参数，那么请使用 PHYSBUFF 的默认值 512 KB。如果在 RTO_SERVER_RESTART 启用的状况下 PHYSBUFF 配置参数值少于 512 KB，那么重新启动服务器时将显示警告消息。

智能大对象的用户数据部分不通过物理日志缓冲区。

相关链接

onstat -l 命令: 显示物理和逻辑日志信息 在第546页

《*SinoDB* 管理员指南》: 物理日志缓冲区

PHYSFILE 配置参数

首次初始化磁盘空间并使数据库服务器联机时，请使用 PHYSFILE 配置参数指定物理日志文件的大小。

onconfig.std 值
PHYSFILE 50000
如不存在
200
值
200 或以上的整数
单位
KB
生效
在编辑 onconfig 文件并通过运行 oninit -i 命令初始化磁盘空间之后。
在运行 onparams -p -s 命令之后。

用法

在首次启动服务器之后，无法通过编辑 onconfig 文件来更改 PHYSFILE 配置参数的值。

数据库服务器在以下情况之下更新 onconfig 文件中的 PHYSFILE 配置参数的值：

- 通过运行 onparams -p -s 命令更改物理日志文件的大小。
- 物理日志空间自动扩展。如果物理日志存储在物理日志空间，数据库服务器根据需要扩展物理日志大小以提升性能，

当启用了 RTO_SERVER_RESTART 配置参数时，请确保物理日志大小至少等于缓冲池大小的 110%。当发生以下状况时，将打印警告消息到消息日志：

- PHYSFILE 配置参数的值更改为小于所有缓冲池的 110%
- 重新启动服务器
- 添加新的缓冲池

相关链接

《SinoDB 管理员指南》：用于估计物理日志大小的策略
《SinoDB 管理员指南》：更改物理日志的位置和大小
onparams -p: 更改物理日志参数 在第353页
RESTARTABLE_RESTORE 配置参数 在第149页
SDS_PAGING 配置参数 在第159页

PLOG_OVERFLOW_PATH 配置参数

PLOG_OVERFLOW_PATH 配置参数指定物理日志文件溢出时快速恢复期间所使用的文件的位置。

该文件是 plog_extend.*servernum*，缺省位置是 $INFORMIXDIR/tmp。以 PLOG_OVERFLOW_PATH 参数使用完整路径名来为该文件指定其他位置。

onconfig.std 值
在 UNIX™ 上：$INFORMIXDIR/tmp
在 Windows™ 上：无
生效
当启动数据库服务器时（初始化共享内存）

相关链接

《*SinoDB* 管理员指南》: 快速恢复期间可能的物理日志溢出

PLCY_HASHSIZE 配置参数

PLCY_HASHSIZE 配置参数指定安全策略信息高速缓存中散列存储区的数量。

onconfig.std 值

PLCY_HASHSIZE 31

值

任何正整数

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 安全指南》: 维护基于标号的访问控制实现

PLCY_POOLSIZE configuration parameter

使用 PLCY_POOLSIZE 配置参数指定安全策略信息高速缓存的每个散列存储区中条目的最大数量。

onconfig.std 值

PLCY_POOLSIZE 127

值

127 或以上的正值，表示高速缓存中初始最大条目数的一半。 最大值取决于共享内存配置和服务器实例的可用共享内存。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wm 命令增加内存中的值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

高速缓存中初始条目数是 PLCY_POOLSIZE 配置参数值的两倍。例如，如果 PLCY_POOLSIZE 配置参数设置为 127，那么高速缓存中允许的条目数是 254。如果高速缓存中所有条目已满，那么高速缓存大小自动增大 10%。要降低高速缓存大小，请减小 onconfig 文件中 PLCY_POOLSIZE 配置参数的值并重启服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 安全指南》: 维护基于标号的访问控制实现

PN_STAGEBLOB_THRESHOLD 配置参数

使用 PN_STAGEBLOB_THRESHOLD 配置参数在循环分段中为 BYTE 和 TEXT 数据保留空间。

onconfig.std 值

未设置。

如不存在

0

值

0 - 1000000

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

将此配置参数设置为表中 BYTE 或 TEXT 数据的典型大小或平均大小。

限制: 如果扩展块的数量已达允许的最大数量或数据库空间已满，则 PN_STAGEBLOB_THRESHOLD 配置参数无效。

当表到达分段的最大页数时，则通过添加新的分段来向该表添加更多的页。 然而，如果表包含 BYTE 或 TEXT 列，并且该表以循环分布模式分段，则添加新分段不会自动使新行插入新分段中。

例如，当表中的一个分段到达最大页数，添加新的分段不会扩展该表来存储更多行。 由于 BYTE 和 TEXT 数据的大小往往很大，因此该数据在平均分布到所有分段中之前是暂存在其中一个分段 中。该暂存分段必须有足够空间来存储 BYTE 或 TEXT 数据。使用 PN_STAGEBLOB_THRESHOLD 配置参数，以便数据库服务器可以将 BYTE 或 TEXT 数据暂时存储在暂存分段中，直到 INSERT 操作完成并且数据永久存储在表中。

在 UPDATE 操作期间，如果分段没有 PN_STAGEBLOB_THRESHOLD 配置参数所指定的空间，那么受到该更新操作影响的表行会搬移到另一分段中。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

PRELOAD_DLL_FILE 配置参数

PRELOAD_DLL_FILE 配置参数指定数据库服务器启动时预加载的共享库文件的路径名。

onconfig.std 值

未设置。不预加载任何共享库文件。

值

pathname = 共享库文件的完整路径名。 Can include $INFORMIXDIR.

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

使用此参数预加载 DataBlade®、内置扩展，或以 C 编程语言创建的用户定义的例程（C UDR）的共享库文件。否则，共享库会在服务器启动后第一次被使用时加载，这会影响性能。为要预加载的每个库文件添加此参数的单独条目。预加载的共享库会保持活动直到服务器停止。

限制: 您不可使用 onmode -wm 或 onmode -wf 命令来设置 PRELOAD_DLL_FILE 配置参数。

示例

以下示例预加载内置的 Basic Text Search、Spatial 和时间序列扩展：

```
PRELOAD_DLL_FILE $INFORMIXDIR/extend/bts.version/bts.bld
PRELOAD_DLL_FILE $INFORMIXDIR/extend/spatial.version/spatial.bld
PRELOAD_DLL_FILE $INFORMIXDIR/extend/TimeSeries.version/TimeSeries.bld
```

version 是扩展的版本号。要找到正确的版本号，请运行适当的函数以返回扩展的发布号或查看您的安装目录中的目录名。

> 重要： 内置扩展的版本号在修正包或发布中可能改变。在您升级后，如果扩展的版本号变更，则您必须更新 PRELOAD_DLL_FILE 配置参数的值。

相关链接

onstat -g dll 命令: 打印动态链接库文件列表 在第452页

QSTATS 配置参数

QSTATS 配置参数指定 onstat -g qst 打印队列统计信息的能力。

onconfig.std 值

QSTATS 0

值

0 = 禁用队列统计信息

1 = 启用队列统计信息

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

onstat -g qst 命令: 打印等待队列和条件队列的等待选项 在第498页

REMOTE_SERVER_CFG 配置参数

使用 REMOTE_SERVER_CFG 配置参数指定表列了可信远程主机的文件。

onconfig.std 值

未设置。使用系统文件 hosts.equiv。

值

文件名。假定路径是 $INFORMIXDIR/etc。 请考虑使用以下命名约定：

```
authfile.server_name
```

REMOTE_SERVER_CFG 配置参数指定的文件必须位于 $INFORMIXDIR/etc。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

REMOTE_SERVER_CFG 配置参数所指定的文件的格式与系统文件 hosts.equiv 的格式相同。

在未设置 REMOTE_SERVER_CFG 配置参数的情况下，使用 cdr add trustedhost 参数运行 SQL 管理 API task() 或 admin() 函数，数据库服务器会执行以下操作：

1. 将 REMOTE_SERVER_CFG 配置参数设置为 authfile.*DBSERVER*。
2. 在 $INFORMIXDIR/etc 目录下创建 authfile.*DBSERVER* 文件。
3. 将指定的可信主机信息加入 $INFORMIXDIR/etc/authfile.*DBSERVER*。
4. 如果数据库服务器是高可用性集群的一部分，则可信主机信息将传播到其他集群服务器的可信主机文件。

> 注： 如果数据库服务器的 sqlhosts 文件使用 s=6 选项，那么您还必须将 S6_USE_REMOTE_SERVER_CFG 配置参数设置为 1 以使用 REMOTE_SERVER_CFG 配置参数指定的文件。

否则，数据库服务器使用系统文件 hosts.equiv，而不是 REMOTE_SERVER_CFG 配置参数指定的文件。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
onmode -wf#-wm: 动态更改某些配置参数 在第347页
S6_USE_REMOTE_SERVER_CFG 配置参数 在第154页
《*SinoDB 管理员指南*》: 可信主机信息
《*SinoDB 管理员指南*》: *sqlhosts 文件和 SQLHOSTS 注册表键选项*
cdr add trustedhost 参数: 添加可信任主机 (SQL 管理 API) 在第595页
cdr remove trustedhost 参数: 移除可信任主机 (SQL 管理 API) 在第599页
cdr list trustedhost 参数: 罗列可信任主机 (SQL 管理 API) 在第598页

REMOTE_USERS_CFG 配置参数

使用 REMOTE_USERS_CFG 配置参数指定表列了远程主机上可信用户名的文件。

onconfig.std 值
未设置，

值
文件名。假定该路径是 $INFORMIX/etc。

生效
在编辑 onconfig 文件并重启数据库服务器之后。
当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。
当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

REMOTE_USERS_CFG 配置参数所指的文件必须位于 $INFORMIXDIR/etc。如果设置了该配置参数，则使用指定的文件而不是 ~/.rhosts 文件。如果在 $INFORMIXDIR/etc 没有指定的文件，那么身分验证将会失败。

REMOTE_USERS_CFG 配置参数所指定的文件的格式与 ~/.rhosts 文件的格式相同。

对于 REMOTE_USERS_CFG 配置参数所指定的文件，请考虑使用以下的命名约定：

```
users.server_name
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
onmode -wf#-wm: 动态更改某些配置参数 在第347页
REMOTE_USERS_CFG 配置参数 在第148页
《*SinoDB 管理员指南*》: 可信用户信息
REMOTE_USERS_CFG 配置参数 在第148页

RESIDENT 配置参数

使用 RESIDENT 配置参数指定共享内存的常驻和虚拟段是否保持驻留在操作系统物理内存中。

onconfig.std 值
RESIDENT 0

值
-1 - 99

0 = 关

1 = 仅锁定常驻段

-1 = 锁定所有常驻和虚拟段

n = 锁定常驻段和接下来的 *n* -1 虚拟段。例如，如果值指定为 99，则锁定常驻段和接下来的 98 个虚拟段。

某些平台有不同的值。有关信息，请参阅您的机器说明。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

一些系统允许您指定共享内存的常驻部分必须始终留（驻留）在内存中。如果操作系统支持强制的常驻，那么可以指定共享内存的常驻和虚拟段不交换到磁盘。

> 警告： 在决定强制常驻之前，请确认可用物理内存量足够执行所有必需的操作系统和应用程序的进程。如果内存不足，则可能导致系统挂起而必需重启。

在支持大页面内存的 AIX®、Solaris 或 Linux™ 系统上，数据库管理员可使用操作系统命令来配置大页面的池。

如果采取以下步骤，则 SinoDB® 可以在大页面上储存非消息的虚拟内存段：

- 通过设置 IFX_LARGE_PAGES 环境变量来启用大页面的大小。
- 对于要储存在大页面上的虚拟内存段，请设置 RESIDENT 参数来锁定物理内存中的那些段，以使那些段不交换到磁盘。

在大页面上储存虚拟内存段可以在大内存配置中提供显著的性能优势。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -n, -r: 更改共享内存驻留 在第342页

《*SinoDB* 管理员指南》: 共享内存常驻部分

《*SinoDB* 管理员指南》: 设置数据库服务器共享内存配置参数

《*SinoDB SQL* 指南: 参考》: *IFX_LARGE_PAGES* 环境变量

RESTARTABLE_RESTORE 配置参数

使用 RESTARTABLE_RESTORE 配置参数来控制数据库服务器是否执行可重新开始的恢复。

onconfig.std 值

RESTARTABLE_RESTORE ON

值

ON = 启用可重新开始的恢复

OFF = 禁用可重新开始的恢复

生效

在编辑 onconfig 文件并重启数据库服务器之后。

如果将 RESTARTABLE_RESTORE 设置为 ON，那么使数据库服务器能够在故障发生点重新启动失败的物理或冷逻辑恢复。要用 ON-Bar 执行可重新开始的恢复，请使用 onbar -RESTART 命令。

如果打算使用可重新开始的恢复，请增加物理日志的大小。当有许多日志需要恢复，那么尽管可重新开始的恢复会减慢逻辑恢复，但还是因不必重复整个恢复而节省大量时间。

重要： 如果数据库服务器在热逻辑恢复期间故障，那么必需重复整个恢复。如果数据库服务器仍在运行，那么使用 onbar -r -l 完成该恢复。

如果在不相同的系统上执行冷恢复，则您可以为块分配新路径名，并可以在恢复期间重命名关键块的设备。在重命名和恢复操作完成之后，必须执行 0 级的归档。

数据库服务器使用物理恢复和逻辑恢复来恢复数据，如下所示：

- 物理恢复。数据库服务器将数据页从备份介质写到磁盘中。此操作使存储空间与原本备份时保持一致。然而，每个存储空间的备份时间通常是不同的。可重新开始的恢复在存储空间级别是可重新开始的。如果恢复失败时只恢复了存储空间的某些块，那么在重新启动恢复时需要再次恢复整个存储空间。
- 逻辑恢复。数据库服务器在介质上重放逻辑日志记录，以使所有存储空间为最新的。在逻辑恢复结束时，所有存储空间与同一点保持一致。

相关链接

PHYSFILE 配置参数 在第144页

《*SinoDB* 备份和还原指南》: *onbar -RESTART* 语法: 重新启动失败的复原

RESTORE_POINT_DIR 配置参数

使用 RESTORE_POINT_DIR 配置参数来更改在服务器升级失败期间放置恢复点文件的目录的路径名。仅当启用了 CONVERSION_GUARD 配置参数， SinoDB® 才会将恢复点文件存储在指定目录中以服务器编号作为子目录名的子目录中。

onconfig.std 值

$INFORMIXDIR/tmp

值

目录的完整路径名

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

You can change the directory, for example, if you think that the $INFORMIXDIR/tmp directory does not have enough space for restore point data. If you want to change the directory, you must change it before you initiate an upgrade to a new version of the server. You cannot change the directory during an upgrade.

The directory specified in the RESTORE_POINT_DIR configuration parameter must be empty when an upgrade begins. If the directory contains any restore point files from a previous upgrade, you must remove the files before a new upgrade begins a new restore point.

重要:

The empty directory is a prerequisite before doing the upgrade, not when recovering from a failed upgrade. After a failed upgrade, do not empty the RESTORE_POINT_DIR directory before you attempt to run the onrestorept utility.

相关链接

CONVERSION_GUARD 配置参数 在第86页

《*SinoDB* 迁移指南》: *onrestorept* 实用程序

ROOTNAME 配置参数

ROOTNAME 指定数据库服务器配置的根数据库空间的名称。

该名称在数据库服务器管理的所有数据库空间中必须是唯一的。建议选择易于识别的名称作为根数据库空间名称。

onconfig.std 值

ROOTNAME rootdbs

值

最多 128 个字节。ROOTNAME 必须以字母或下划线开始，并且必须只包含字母、数字、下划线或 $ 字符。

单位

数据库空间

生效

初始化磁盘时（毁坏所有数据）

相关链接

《*SinoDB* 管理员指南》：分配磁盘空间

ROOTOFFSET 配置参数

指定磁盘空间分配（文件、磁盘分区或设备）的偏移量，该处为根数据库空间初始块的开始位置。

仅 UNIX:

在一些 UNIX™ 平台上，将 ROOTOFFSET 设置为 0 是无效的。当此参数设置不正确 时，您必须重新初始化磁盘空间，并重新加载数据，以重新继续数据库服务器的正常运行。 在配置数据库服务器之前，请始终查看机器说明以获得有关正确设置的信息。

onconfig.std 值

ROOTOFFSET 0

值

任何大于或等于 0 的值

单位

KB

生效

初始化磁盘时（毁坏所有数据）

相关链接

《*SinoDB* 管理员指南》：在 *UNIX* 上分配原始磁盘空间

ROOTPATH 配置参数

使用 ROOTPATH 配置参数来指定根数据库空间的初始块的完整路径名，其中包含设备或文件名。ROOTPATH 配置参数作为块名称存储在保留页中。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/tmp/demo_on.rootdbs
在 Windows™ 上：无

值

路径名

生效

初始化磁盘时（毁坏所有数据）

参阅

《*SinoDB*® 管理员指南》 的管理磁盘空间章节中的以下材料：

- 分配磁盘空间
- 为原始设备创建链接

用法

在 UNIX™ 上，必须将 ROOTPATH 配置参数所指定文件的权限设置为 660，且所有者和组都必须是 informix。在 Windows™ 上，Informix-Admin 组的成员必须拥有 ROOTPATH 配置参数所指定的文件。

仅 UNIX:

如果在 UNIX™ 上初始块使用未缓冲的磁盘空间，那么您应将指向根数据库空间初始块的链接作为 ROOTPATH 配置参数定义的路径名，而不是输入该初始块的实际设备名。

相关链接

《*SinoDB* 管理员指南》：分配磁盘空间

《*SinoDB* 管理员指南》：创建原始设备的符号链接 *(UNIX)*

ROOTSIZE 配置参数

使用 ROOTSIZE 配置参数来指定根数据空间的初始块的大小（以 KB 计）。您选择的大小取决于数据库服务器的即时计划。

数据库服务起仅在磁盘初始化完成期间使用 ROOTSIZE 配置参数的值。在根数据库空间的初始块创建之后，更改 ROOTSIZE 的值是无效的。

onconfig.std 值

ROOTSIZE 300000

如不存在

0

值

50,000 到存储设备的最大容量

单位

KB

生效

初始化磁盘时（毁坏所有数据）

相关链接

《*SinoDB* 管理员指南》：根数据库空间的大小

RSS_FLOW_CONTROL 配置参数

指定在至少包含一个远程独立（RS）辅助服务器的高可用性集群中何时进行流量控制。

onconfig.std 值

RSS_FLOW_CONTROL 0

值

0 = 在当前日志位置与最近响应的日志之间的差异超过日志缓冲区大小的 12 倍时激活流量控制。

-1 = 禁用流量控制。禁用流量控制可能导致日志文件覆盖和数据遗失。

start_value,*end_value* = *start_value* 和 *end_value* 决定当前日志位置和上次响应的日志页之间的迟滞量。*start_value* 必须大于 *end_value*。值必须包含下列单位之一：

- K （KB）
- M（MB）
- G（GB）

例如，设置 RSS_FLOW_CONTROL 128M,100M，则当日志之间的迟滞量为 128 MB时开始流量控制，并当迟滞量掉到 100 MB时停止流量控制。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

流量控制提供一种限制主服务器上日志活动的方法，以便集群中 RS 辅助服务器在处理事务上不会落后太多。如果服务器位于繁忙或间歇性网络，启用流量控制可确保 RS 辅助服务器上的日志保持最新。当启用了流量控制，则在当前日志位置与上次响应的日志页之间的日志大小差异超过 *start_value* 时，主服务器上的日志活动将受到限制。连接到主服务器的用户在流量控制处于活动状态时，可能遇到响应时间较慢。当日志之间的迟滞量大于 *start_value* 时则启动流量控制，并且当日志迟滞量掉到 *end_value* 时则停止流量控制。

仅在主服务器上设置 RSS_FLOW_CONTROL 配置参数。集群中的所有 RS 辅助服务器都会受到 RSS_FLOW_CONTROL 配置参数的影响。日志始终按收到的顺序发送到 RS 辅助服务器。

要检查 RS 辅助服务器的流量控制是否处于活动状态，请使用 onstat -g rss verbose 命令，并将 RSS flow control 值与 Approximate Log Page Backlog 值进行比较。如果 Approximate Log Page Backlog 高于 RSS flow control 的第一个值，则流量控制处于活动状态。如果 Approximate Log Page Backlog 低于 RSS flow control 的第二个值，则流量控制是禁用的。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SDS_FLOW_CONTROL 配置参数 在第158页

《*SinoDB* 管理员指南》：共享磁盘辅助服务器的流量控制

《*SinoDB* 管理员指南》：远程独立辅助服务器的流量控制

RTO_SERVER_RESTART 配置参数

使用 RTO_SERVER_RESTART 配置参数来指定在重启 SinoDB® 服务器并使其进入联机或静默模式之后，SinoDB® 从问题恢复的时间量（以秒计）的恢复时间目标（RTO）标准。

onconfig.std 值

RTO_SERVER_RESTART 0（禁用）

值的范围

0 = 禁用

60 - 1800

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

CKPTINTVL 配置参数 在第83页

onparams 实用程序 在第351页

《*SinoDB* 管理员指南》: *oncheck -pr* 命令

《*SinoDB* 管理员指南》: 检查点

《*SinoDB* 性能指南》: 配置对 *I/O* 活动的影响

S6_USE_REMOTE_SERVER_CFG 配置参数

使用 S6_USE_REMOTE_SERVER_CFG 配置参数来控制是否使用 REMOTE_SERVER_CFG 配置参数所指定的文件来验证服务器集群和 Enterprise Replication 的安全连接。

onconfig.std 值

S6_USE_REMOTE_SERVER_CFG 0

默认值

0

值

0 = 使用系统文件 hosts.equiv 验证通过安全端口连接的服务器。

1 = 使用 REMOTE_SERVER_CFG 配置参数所指定的文件来验证通过安全端口连接的服务器。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

REMOTE_SERVER_CFG 配置参数用于指定列出数据库服务器所在计算机信任的远程服务器主机的文件。如果一个或多个列出的服务器的配置是使用 sqlhosts 文件的连接安全选项 s=6，那么 S6_USE_REMOTE_SERVER_CFG 配置参数必须设置为 1。

如果未设置 S6_USE_REMOTE_SERVER_CFG 或设置为 0，则使用系统文件 hosts.equiv，而不是 REMOTE_SERVER_CFG 配置参数所指定的文件，来验证通过安全端口连接的服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

REMOTE_SERVER_CFG 配置参数 在第147页

《*SinoDB* 管理员指南》: *sqlhosts* 文件和 *SQLHOSTS* 注册表键选项

SB_CHECK_FOR_TEMP 配置参数

使用 SB_CHECK_FOR_TEMP 配置参数来防止临时智能大对象复制到永久表中。

onconfig.std 值

未设置。

如果值不存在

允许将临时智能大对象复制到永久表中。

值

0 = 允许将临时智能大对象复制到永久表中。等同未在 onconfig 中设置该配置参数。

1 = 防止临时智能大对象复制到永久表中。数据库服务器返回下列错误消息，而不是复制临时智能大对象的句柄：

- -9810: Smart-large-object error.
- -12246: Smart large objects: You cannot put a temporary smart large object into a permanent table

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

在默认情况下，可以将临时智能大对象复制到永久表中。智能大对象数据类型，BLOB 和 CLOB，由两部分组成：数据（存储在智能大对象空间中）和句柄（存储在表中）。当复制临时智能大对象到永久表中时，只有 BLOB 或 CLOB 句柄复制到永久表中。如果之后删除了临时智能大对象，则永久表会包含不再有效的句柄。

要防止将临时智能大对象复制到永久表中，请将 onconfig 文件中的 SB_CHECK_FOR_TEMP 配置参数设置为 1。例如，如果 SB_CHECK_FOR_TEMP 配置参数设置为 1，则复制临时智能大对象到永久表的 INSERT INTO . . . SELECT FROM . . . 语句失败。

SBSPACENAME 配置参数

使用 SBSPACENAME 配置参数指定默认智能大对象空间的名称。

onconfig.std 值

未设置。

如不存在

0

值

最多 128 字节。

SBSPACENAME 必须唯一，以字母或下划线开始，且只包含字母、数字、下划线或 $ 字符。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果数据库表包含智能大对象列，而那些列没有明显地指定存储空间，则数据会存在在 SBSPACENAME 所指定的智能大对象空间中。

您必须先使用 onspaces -c -S 实用程序创建默认的智能大对象空间，然後才能使用它。 当发生下列任一情况时，数据库服务器会验证默认的智能大对象空间的名称：

- 指定默认的智能大对象空间作为 CREATE TABLE 或 ALTER TABLE 语句的 PUT 子句中 CLOB 或 BLOB 列的存储选项。
- 当没有为列指定智能大对象空间时，数据库服务器会尝试将智能大对象写入默认的智能大对象空间。
- 在默认的智能大对象空间中存储多重表示数据。

  JAVA 语言支持:

  如果您正在使用 J/Foundation，则必须提供数据库服务器存储 Java™ 归档（JAR）文件的智能大对象。這些 JAR 文件包含您的 Java™ 用戶定義的例程（UDR）。建议，当使用 Java™ UDRs 时，创建独立的智能大对象空间来存储智能大对象。

  Warning: 当使用 Enterprise Replication 时，必须设置 CDR_QDATA_SBSPACE 参数，并且在定义复制服务器之前创建智能大对象空间。

自动创建默认的智能大对象空间

当创建 bts 索引且未显式地指定智能大对象空间名，那么即使未设置 SBSPACENAME 配置参数，也会创建默认的智能大对象空间。

在数据库服务器的根数据库服务器中创建大小为 10,000 KB 的默认智能大对象空间。当默认智能大对象空间满时，您必须手动增加其大小。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
SBSPACETEMP 配置参数 在第156页
SYSSBSPACENAME 配置参数 在第184页
智能大对象空间结构 在第270页
onspaces -c -S: 创建智能大对象空间 在第371页
《*SinoDB* 管理员指南》: 智能大对象空间
《*SinoDB* 管理员指南》: 更改智能大对象的存储特征
《*SinoDB SQL* 指南: 语法》: *PUT* 子句
《*SinoDB Enterprise Replication* 指南》: 行数据智能大对象空间

SBSPACETEMP 配置参数

使用 SBSPACETEMP 配置参数来指定默认临时智能大对象空间的列表，用于存储没有元数据或用户数据记录的临时智能大对象。如果在标准智能大对象空间中存储临时智能大对象，则会记录元数据。

onconfig.std 值

未设置。临时智能大对象存储在 SBSPACENAME 配置参数所指定的默认智能大对象空间。

分隔符

逗号

值

一个或多个智能大对象空间名。以逗号分隔。列表长度不可超过 128 字节。

每个智能大对象空间名必须是唯一，以字母或下划线开始，且只包含字母、数字、下划线或 $ 字符。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
SBSPACENAME 配置参数 在第155页
onspaces -c -S: 创建智能大对象空间 在第371页
《*SinoDB* 管理员指南》: 临时智能大对象空间
《*SinoDB* 管理员指南》: 创建临时智能大对象空间
《*SinoDB* 管理员指南》: 临时智能大对象

SDS_ALTERNATE 配置参数

使用 SDS_ALTERNATE 配置参数来定义高可用性集群中主服务器与 SD 辅助服务器之间通信的备用方法。

onconfig.std 值

无（未设置任何 SD 辅助服务器备用通信路径。）

值

作为主服务器与 SD 辅助服务器之间备用通信路径的 BLOB 空间的名称。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

设置 SDS_ALTERNATE 配置参数和创建共享 BLOB 空间，以允许高可用性集群中的主服务器和所有 SD 辅助服务器在主服务器与所有 SD 辅助服务器之间的网络不可用时使用备用通信路径。在 SD 辅助服务器即将故障转移成为主服务器，但 TCP/IP 通信不可用时，将使用 SDS_ALTERNATE 配置参数设置的共享 BLOB 空间将关闭程序传递给原主服务器。

将主服务器和所有 SD 辅助服务器上的 SDS_ALTERNATE 配置参数设置为相同的值。

在设置 SDS_ALTERNATE 配置参数之前，必须在主服务器上创建共享 BLOB 空间。例如，在主服务器上输入以下命令，以创建名为 sds_alt_comm 的 BLOB 空间：

```
onspaces -c -b sds_alt_comm -g <pagesize> -p <path> -o <offset> -s <size>
```

执行以下命令来切换到下一个逻辑日志文件，以便新创建的 BLOB 空间可用：

```
onmode -l
```

在高可用性集群中的每个 SD 辅助服务器上，将 SDS_ALTERNATE 配置参数设置指向主服务器上的 BLOB 空间。

```
SDS_ ALTERNATE sds_alt_comm
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

oninit 实用程序 在第319页

《*SinoDB* 管理员指南》*: SD* 辅助服务器

SDS_ENABLE 配置参数

使用 SDS_ENABLE 配置参数来启用 SD 辅助服务器功能。

onconfig.std 值

未设置。

如不存在

0

值

0 = 禁用

1 = 启用

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

在 SD 辅助服务器上，必须将 SDS_ENABLE 设置为 1（启用）以启用 SD 辅助服务器功能。

当运行以下命令时，SDS_ENABLE 自动设置为 1（启用）：

```
onmode -d set SDS primary
```

当运行以下命令时，SDS_ENABLE 设置为 0（禁用）：

```
onmode -d clear SDS primary
```

为防止数据损坏，在 SDS_ENABLE 设置为 1（启用）的服务器上不可使用 oninit -i 或 oninit -iy 命令来初始化磁盘空间。要初始化 SD 辅助服务器，请使用不带参数的 oninit 仅初始化共享内存。要初始化连接了一个或多个 SD 辅助服务器的主服务其，且其磁盘从未初始化过，请江 SDS_ENABLE 设置为 0 并使用 oninit -i 初始化服务器内存和磁盘。要初始化连接了 SD 辅助服务器的主服务器，且其磁盘已初始化，请将 SDS_ENABLE 设置为 1 并使用不带参数的 oninit 仅初始化共享内存。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SDS_FLOW_CONTROL 配置参数

指定至少包含一个共享磁盘（SD）辅助服务器的高可用性集群中何时进行流量控制。

onconfig.std 值

SDS_FLOW_CONTROL 0

值

0 = 当前日志位置和最近响应日志之间的差异超过日志缓冲区大小的 12 倍时，激活流量控制。

-1 = 禁用流量控制。禁用流量控制可能导致日志文件覆盖和数据遗失。

start_value,*end_value* = *start_value* 和 *end_value* 决定当前日志位置和上次响应日志页之间的迟滞量。*start_value* 必须大于 *end_value*。值必须包含下列单位之一：

- K（KB）
- M（MB）
- G（GB）

例如，设置 SDS_FLOW_CONTROL 128M,100M，当日志之间的迟滞量是 128 MB 时开始流量控制，当迟滞量降到 100 MB 时停止流量控制。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

用法

流量控制提供一种限制主服务器上日志活动的方法，以便集群中 SD 辅助服务器在处理事务上不会落后太多。当启用了流量控制，则在当前日志位置与上次响应日志页之间的日志大小差异超过 *start_value* 时，主服务器上的日志活动将受到限制。连接到主服务器的用户在流量控制处于活动状态时，可能遇到响应时间较慢。当日志之间的迟滞量大于 *start_value* 时则启动流量控制，并且当日志迟滞量掉到 *end_value* 时则停止流量控制。

仅在主服务器上设置 SDS_FLOW_CONTROL 配置参数。集群中的所有 SD 辅助服务器都会受到 RSS_FLOW_CONTROL 配置参数的影响。日志始终按收到的顺序发送到 SD 辅助服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

RSS_FLOW_CONTROL 配置参数 在第152页
《*SinoDB* 管理员指南》：远程独立辅助服务器的流量控制
《*SinoDB* 管理员指南》：共享磁盘辅助服务器的流量控制

SDS_LOGCHECK 配置参数

使用 SDS_LOGCHECK 配置参数来设置延迟辅助服务器接管主服务起角色的秒数。如果辅助服务器在延迟期间检测到主服务器正生成日志记录，则会阻止故障转移。如果主服务器与辅助服务器之间的网络通信暂时不可用，则延迟可以防止不必要的故障转移。

onconfig.std 值

SDS_LOGCHECK

在 UNIX™ 上：10
在 Windows™ 上：0

值

0 = 不检测日志活动；允许立即故障转移。

n = 最多等待 *n* 秒。如果检测到主服务器的日志活动，则阻止故障转移；否则，允许故障转移。

单位

秒

生效

当启用主服务器上的共享磁盘功能时。

用法

重要： 必须为主服务器和所有辅助服务器指定相同的值。如果值不相同，数据库服务器会自动将辅助服务器上不同的值更改为主服务器上的值。

例如，如果 SDS_LOGCHECK 配置参数设置为 10，并且主服务器故障，则 SD 辅助服务器最多等待 10 秒，以检测主服务器是否正在生成日志记录（在这种情况下会阻止故障转移），或者 SD 辅助服务器检测到主服务器没有正在生成日志记录并进行故障转移。

不必要的故障转移可能导致两个主服务都从应用程序接收输入并写入相同的块，这可能导致无法修复的数据损坏。

如果为设置 I/O 防护且系统由一个主服务器和一个或多个 SD 辅助服务器组成，则将 SDS_LOGCHECK 配置参数设置为大于零的值。

如果系统配置了 I/O 防护，并且 SD 辅助服务器成为主服务器，则 I/O 防护脚本必须阻止故障的主服务器更新任何共享磁盘。如果系统未配置 I/O 防护，则 SDS_LOGCHECK 配置参数可因为原主服务器正在生成日志记录而不进行故障转移到 SD 辅助服务器来防止发生多主服务器。

相关链接

《*SinoDB* 管理员指南》：*SD* 辅助服务器

SDS_PAGING 配置参数

SDS_PAGING 配置参数指定了两个作为缓冲区调页文件的文件的位置。

onconfig.std 值

未设置

值

文件路径

分隔符

单一逗号

默认值

无

生效

当启动 SD 辅助服务器时

用法

必须将 SDS_PAGING 配置参数设置为有效值，以确保 SD 辅助服务器启动。由于调页文件根据需要动态地增长，因此应分配足够磁盘空间来存储两倍于 PHYSFILE 配置参数所指定值的大小。

示例

在以下示例中，将文件 page1 和 page2 设置为 SD 辅助服务器的缓冲区调页文件。

```
SDS_PAGING /usr/informix/tmp/page1,/usr/informix/tmp/page2
```

相关链接

PHYSFILE 配置参数 在第144页

《*SinoDB* 管理员指南》: *SD* 辅助服务器

SDS_TEMPDBS 配置参数

使用 SDS_TEMPDBS 配置参数指定共享磁盘（SD）辅助服务器用于动态地创建临时数据库空间的信息。此配置参数仅可在 SD 辅助服务器上指定。

onconfig.std 值

未设置。不创建共享磁盘辅助服务器的临时数据库空间。

值

按以下顺序包含以下值的字串符，以逗号分隔：

dbspace = 要创建的数据库空间的名称。该名称在所有现有的数据库空间、BLOB 空间和智能大对象空间（包括那些从主服务器继承的临时空间）中必须是唯一的。名称长度不可超过 128 字节。其必须以字母或下划线开始，并且必须只包含字母、数字、下划线或 $ 字符。

dbpath = 数据库空间的路径，可以是完整路径名或相对的路径名。如果使用相对的路径名，则必须是相对于初始化数据库服务器时当前目录的目录。

pagesize = 整数，表示数据库空间的页大小，以 KB 计。页大小必须在 2 KB 到 16 KB 之间，并且必须是默认页大小的倍数。

offset = 等于或大于 0 的整数，指定磁盘分区或设备上到达数据库空间的初始块的偏移量。起始偏移量加上块大小不能超过最大的块大小。偏移量必须是页大小的倍数。最大偏移量是 2 或 4 TB，其取决于平台。在缺省情况下，该值是以 KB 计。通过对值附加单个字符修饰符来指定不同的单位：M 或 m 代表 MB、G 或 g 代表 GB，或者 T 或 t 代表 TB。

size = 等于或大于 1000 KB 的正整数，并且是指定数据库空间初始块大小的页大小的倍数。*offset* 值加上 *size* 值不能超过最大的块大小。块的最大大小等于 2,147,483,647 个页乘上页大小。在缺省情况下，该值以 KB 计。通贵对值附加单个字符修饰符来指定不同的单位：M 或 m 代表 MB、G 或 g 代表 GB，或者 T 或 t 代表 TB。

分隔符

以逗号分隔每个值。不可使用空白。

生效

在编辑 onconfig 文件并重启 SD 辅助服务器之后。

用法

当 SD 辅助服务器启动时，如果之前存在数据库空间，则创建或初始化临时数据库空间。临时数据库空间用于创建临时表。SD 辅助服务器的 onconfig 文件中必须至少有一个 SDS_TEMPDBS 配置参数才可启动 SD 辅助服务器。通过使用多个 SDS_TEMPDBS 配置参数，您可在 onconfig 文件中指定多达 16 个辅助临时数据库空间。

对于 onconfig 文件中的每一 SES_TEMPDBS 配置参数：

- 对于每个服务器，*dbsname* 值必须是唯一的，不与任何其他 SD 辅助服务器或主服务器分享。
- *dbspath*、*pagesize*、*offset* 和 *size* 的组合必须不导致与现有块重叠，或与 SDS_TEMPDBS 配置参数指定的临时数据库空间之间重叠。
- 每个 SDS_TEMPDBS 配置参数值的 *pagesize* 值必须相同。

以下示例显示 SDS_TEMPDBS 配置参数的两个条目：

```
SDS_TEMPDBS sds_space1,/dev/raw_dev1,2,0,60M
SDS_TEMPDBS sds_space2,/dev/raw_dev2,2,0,80M
```

如果高可用性集群中的主服务器故障且 SD 辅助服务器接替作为主服务器，则该 SD 辅助服务器上 SDS_TEMPDBS 配置参数的值将用于临时数据库空间直到该服务器重启。您应确保在 SD 辅助服务器上指定的 SDS_TEMPDBS 配置参数的值不同于主服务器上指定的值。在 SD 辅助服务器重启之后，使用 DBSPACETEMP 配置参数。

相关链接

《*SinoDB* 管理员指南》：共享磁盘辅助服务器

SDS_TIMEOUT 配置参数

使用 SDS_TIMEOUT 配置参数来指定高可用性集群中的主服务器将等待共享磁盘（SD）辅助服务器发送日志位置响应的时间（以秒计）。

onconfig.std 值

SDS_TIMEOUT 20

如不存在

10

值

2 - 2147483647

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的 SDS_TIMEOUT 值时。

当通过运行 onmode -wm 命令来重置内存中的 SDS_TIMEOUT 值时。

用法

如果在指定时间内未收到来自 SD 辅助服务器的日志位置响应，则主服务器将与 SD 辅助服务器断开连接并继续。如果在等待 SD 辅助服务器时页清空已超时，则在等待 SDS_TIMEOUT 配置参数设置中指定的秒数之后，主服务器将开始移除 SD 辅助服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
《*SinoDB* 管理员指南》：共享磁盘辅助服务器

SECURITY_LOCALCONNECTION 配置参数

使用 SECURITY_LOCALCONNECTION 配置参数，通过验证正在运行程序的本地用户的 ID 与正在试图访问数据库的用户的 ID 是否相同，来验证本地连接的安全性。

onconfig.std 值

未设置。

值

0 = 不进行安全性检查。

1 = SinoDB® 检查正在运行程序的用户的 ID 与正在试图连接数据库的用户的 ID 是否相符。

2 = 与 1 相同，此外，SinoDB® 从网络 API 检索同级端口号并验证该连接是来自客户端程序。如果您的系统有 SOCTCP 或 IPCSTR 网络协议，则您仅可指定 2。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 管理员指南》：设置数据库服务器共享内存配置参数

SEQ_CACHE_SIZE 配置参数

使用 SEQ_CACHE_SIZE 配置参数指定在内存中高速缓存的序列对象的最大数目。

onconfig.std 值

SEQ_CACHE_SIZE 10

值

1 - 2147483647

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当高速缓存了序列对象的最大数量时，数据库服务器会尝试移除不再引用的序列对象的条目。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SERVERNUM 配置参数

SERVERNUM 配置参数指定共享内存中的相对位置。

onconfig.std 值

SERVERNUM 0

值

0 - 255

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

本地计算机上，您为每一数据库服务器选择的值必须是唯一的。在您的网络上，该值不需要是唯一的。因为 onconfig.std 文件中包含值 0，所以建议您选择 0 以外的值以避免无意中重复 SERVERNUM 配置参数。

相关链接

《*SinoDB* 管理员指南》: 设置数据库服务器共享内存配置参数

SESSION_LIMIT_LOCKS 配置参数

SESSION_LIMIT_LOCKS 配置参数指定会话中可用锁的最大数目。此限制不适用于具有管理权限的用户，例如，用户 informix 或 DBSA 用户。

onconfig.std 值

无

如不存在

2147483647

值

500 - 2147483647

单位

在内部锁表中锁的数目

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

对于大规模锁定密集的操作，管理员可设置 SESSION_LIMIT_LOCKS 以降低并发会话中普通用户耗尽数据库服务器的锁资源的风险。

数据库服务器终止超过锁定数限制的事务，将消息放入数据库服务器消息日置中，并触发事件警报 21014。

重要:

在可重复的读隔离级别中，因为活动集中每一行需要一个锁，所以请小心将服务器上锁的限制设置过低。同样地，锁限制设置过低可能妨碍 Enterprise Replication 任务或妨碍非 DBSA 用户发出的 cdr 命令。

相关链接

《*SinoDB SQL* 指南: 语法》: *IFX_SESSION_LIMIT_LOCKS* 会话环境选项

《*SinoDB* 管理员指南》: 管理租户数据库

《*SinoDB* 管理员指南》: 限制会话资源

tenant create 参数: 创建租户数据库 *(SQL* 管理 *API)* 在第705页

tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页

SESSION_LIMIT_LOGSPACE 配置参数

SESSION_LIMIT_LOGSPACE 配置参数指定会话可用于单个事务的最大日志空间量。此限制不适用于具有管理权限的用户，例如，informix 用户或 DBSA 用户。

onconfig.std 值

0（关）

如不存在
0（关）
值
5120 - 2147483648
单位
KB
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

SESSION_LIMIT_LOGSPACE 配置参数限制会话可用于每个事务的日志空间量，并节省租户数据库环境中的系统资源使用。

数据库服务器终止超过日志空间限制的事务，将消息放入数据库服务器消息日志中，并触发事件警报 21018。

通过 tenant create 或 tenant update SQL API 命令设置的 session_limit_logspace 租户数据库属性优先于 SESSION_LIMIT_LOGSPACE 配置参数设置。

相关链接

《*SinoDB* 管理员指南》: 管理租户数据库
《*SinoDB* 管理员指南》: 限制会话资源
tenant create 参数: 创建租户数据库 *(SQL* 管理 *API)* 在第705页
tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页

SESSION_LIMIT_MEMORY 配置参数

SESSION_LIMIT_MEMORY 配置参数指定会话可以分配的最大内存量。此限制不适用于具有管理权限的用户，例如，informix 用户或 DBSA 用户。

onconfig.std 值
0（关）
如不存在
0（关）
值
20480 - 2147483648
单位
KB
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

SESSION_LIMIT_MEMORY 配置参数限制会话可以分配的内存量，并可防止单个会话独占系统资源。

数据库服务器终止超过内存限制的会话，将消息放入数据库服务器消息日志中，并触发事件警报 21016。

通过 tenant create 或 tenant update SQL API 命令设置的 session_limit_memory 租户数据库属性优先于 SESSION_LIMIT_MEMORY 配置参数设置。

相关链接

《*SinoDB* 管理员指南》: 管理租户数据库
《*SinoDB* 管理员指南》: 限制会话资源

tenant create 参数: 创建租户数据库 *(SQL* 管理 *API)* 在第705页
tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页

SESSION_LIMIT_TEMPSPACE 配置参数

SESSION_LIMIT_TEMPSPACE 配置参数指定会话可以分配的最大临时表空间量。此限制不适用于具有管理权限的用户，例如，informix 用户或 DBSA 用户。

onconfig.std 值
0（关）
如不存在
0（关）
值
20480 - 2147483648
单位
KB
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

SESSION_LIMIT_TEMPSPACE 配置参数限制会话可以分配的临时表空间量，并节省租户数据库环境中系统资源使用。

数据库服务器终止超过空间限制的会话，将消息放入数据库服务器消息日志中，并触发事件警报 21017。

通过 tenant create 或 tenant update SQL API 命令设置的 session_limit_tempspace 租户数据库属性优先于 SESSION_LIMIT_TEMPSPACE 配置参数的设置。

相关链接

《*SinoDB* 管理员指南》: 管理租户数据库
《*SinoDB* 管理员指南》: 限制会话资源
tenant create 参数: 创建租户数据库 *(SQL* 管理 *API)* 在第705页
tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页

SESSION_LIMIT_TXN_TIME 配置参数

SESSION_LIMIT_TXN_TIME 配置参数指定会话中事务可以运行的最大时间量。此限制不适用于具有管理权限的用户，例如，informix 用户或 DBSA 用户。

onconfig.std 值
0（关）
如不存在
0（关）
值
60 - 2147483647
单位
秒
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

SESSION_LIMIT_TXN_TIME 配置参数限制会话中事务可以运行的时间量，并可防止单个会话事务独占逻辑日志。

数据库服务器终止超过时间限制的事务，将消息放入数据库服务器消息日志中，并触发事件警报 21019。

通过 tenant create 或 tenant update SQL API 命令设置的 session_limit_txn_time 租户数据库属性优先于 SESSION_LIMIT_TXN_TIME 配置参数的设置。

相关链接

《SinoDB 管理员指南》：管理租户数据库

《SinoDB 管理员指南》：限制会话资源

tenant create 参数：创建租户数据库 (SQL 管理 API) 在第705页

tenant update 参数：更改租户数据库属性 (SQL 管理 API) 在第710页

SHMADD 配置参数

使用 SHMADD 配置参数来指定动态地添加到共享内存虚拟部分的段的大小。

onconfig.std 值

与平台有关

值

32 位平台：1024 - 524288

64 位平台：1024 - 4294967296

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当需要添加内存时，SHMADD 配置参数的值表示数据库服务器添加到共享内存虚拟部分的第一批段的大小。通过 SHMVIRTSIZE 配置参数设置第一个虚拟共享内存段的大小。请设置 SHMVIRTSIZE 和 SHMADD 配置参数的值，以便在数据库服务器的正常操作期间添加最少量的段。通常，较多的段会降低性能。

SinoDB® 共享内存段的最大数目是 1024。如果 SHMADD 值低，或者数据库服务器意外地有大量活动或内存使用，则可能需要许多共享内存段。为防止数据库服务器达到共享内存段的最大数目，服务器每 16 个虚拟段加倍动态添加的虚拟段的大小。在大型段中添加内存更为有效率，但如果添加的内存未被使用，则造成浪费。而且，操作系统可能要求您在几个大段中而不是许多小段中添加内存。

下表包含设置 SHMADD 初始值的建议。

表 68: 建议的 SHMADD 值

物理内存量	建议的 SHMADD 值
小于 256 MB	8192
256 - 512 MB	16,384
大于 512 MB	32,768

通过运行 onstat -g seg 命令，您可查看关于虚拟内存段的信息。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
onstat -g seg 命令: 显示共享内存段的统计信息 在第512页
《*SinoDB* 管理员指南》: 共享内存的虚拟部分
《*SinoDB* 管理员指南》: 监视共享内存段
EXTSHMADD 配置参数 在第112页

SHMBASE 配置参数

使用 SHMBASE 配置参数来指定共享内存附加到虚拟处理器的内存空间的基本地址。

onconfig.std 值
与平台有关
值
正整数
单位
地址
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

共享内存段的地址起始于 SHMBASE 值并增长直到达到上限，该上限与平台有关。

请不要更改 SHMBASE 的值。onconfig.std 中的 SHMBASE 值取决于平台，以及处理器是 32 位还是 64 位。有关要使用的 SHMBASE 值的信息，请参阅机器说明。

相关链接

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页
《*SinoDB* 管理员指南》: 设置操作系统共享内存配置参数

SHMNOACCESS 配置参数

SHMNOACCESS 配置参数指定不用于附加共享内存的虚拟内存地址范围。

onconfig.std 值
在 UNIX™ 上：无
在 Windows™ 上：#SHMNOACCESS 0x70000000-0x7FFFFFFF，且此值在 onconfig.std 模板文件中是注释掉的。
值
1 - 10 地址范围
分隔符
逗号
生效
在编辑 onconfig 文件并重启数据库服务器之后。

用法

SHMNOACCESS 配置参数用于避开特定范围的进程地址，从而避免与操作系统库冲突。

每个范围中的每个地址必须以十六机制格式起始。范围中的每个地址必须以连字符分隔，且每个范围必须以逗号分隔，如以下示例所示：

```
SHMNOACCESS 0x70000000-0x75000000,
0x7A000000-0x80000000
```

SHMTOTAL 配置参数

使用 SHMTOTAL 配置参数来指定数据库服务器用于所有内存分配的共享内存（常驻部分、虚拟部分、通信部分和虚拟扩展部分）的总量。onconfig.std 的 0 值表示不规定内存分配限制。

onconfig.std 值

SHMTOTAL 0

值

0 = （无特定限制）或者大于或等于 1 的任何整数

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

您可以使用 SHMTOTAL 配置参数来限制数据库服务器可对系统提出的内存需求。然而，如果数据库服务器需要的内存多于 SHMTOTAL 强加的限制，则应用程序可能失败。当发生这种情况时，数据库服务器将以下消息写到消息日志中：

```
size of resident + virtual segments xx + yy > zz total allowed by
configuration parameter SHMTOTAL
```

这个消息包含以下值。

xx

常驻段的当前大小

yy

虚拟段的当前大小

zz

需要的共享内存合计

如果启用了 LOW_MEMORY_MGR 配置参数，并且将服务器配置为使用 SHMTOTAL 配置参数值的百分比作为自动低内存管理启动和停止的 阈值，则 SHMTOTAL 配置参数必定不可设置为 0（无限制）。

注意： 更改 SHMTOTAL 配置参数的值，可能导致自动低内存管理的配置变成无效，强制数据库服务器使用默认设置。

仅 UNIX:

为得到最大共享内存段大小（通常是 SHMMAX、SHMSIZE 或 SHMALL），将操作系统参数设置为数据库服务器配置所需大小的合计。有关您的操作系统所允许的共享内存量的信息，请参阅机器说明。

如果您有比机器说明中指定的值更多的物理内存，且将由 SinoDB® 使用该内存，则您可增加 SHMALL 参数的值，直到计算机指定的物理内存的 90%。建议不要达到或超过可用的 RAM。

相关链接

DS_TOTAL_MEMORY 配置参数 在第106页
LOW_MEMORY_MGR 配置参数 在第127页
《*SinoDB* 管理员指南》：共享内存

《*SinoDB* 管理员指南》: 共享内存大小
scheduler lmm enable 参数: 指定自动低内存管理设置 *(SQL* 管理 *API)* 在第677页
LOW_MEMORY_MGR 配置参数 在第127页

SHMVIRT_ALLOCSEG 配置参数

使用 SHMVIRT_ALLOCSEG 配置参数来指定 SinoDB® 应分配一个新的共享内存段的阈值，以及如果服务器不能分配新的内存段时激活的事件报警级别。

onconfig.std 值

SHMVIRT_ALLOCSEG 0,3

值

一个数值，可选地后跟一个逗号和另一个数值。

threshold = 一个数值，指示数据库服务器何时应添加共享内存段：

- 0 = 默认。当需要时，数据库服务器分配共享内存段。
- .40 - .99 = 在添加一个段之前，已使用的内存百分比。
- 256 - 10000000 = 在添加一个段之前，剩馀的 KB 数。

alarm_level：可选的。一个从 1 到 5 的整数，指定发出的事件报警级别：1 = 不值的注意，2 = 信息，3 = 注意（默认），4 = 紧急，5 = 致命。事件报警的类 ID 是 24，事件 ID 是 24003。

分隔符

使用逗号分隔这些值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

设置 SHMVIRT_ALLOGSEG 配置参数以主动添加共享内存段，而不是等到数据库服务器自动地添加共享内存段。

如果不能分配新的内存段，则事件警报每三十分钟重复一次。

相关链接

事件警报参数 在第724页
《*SinoDB* 性能指南》: *SHMVIRT_ALLOCSEG* 配置参数和内存利用率

SHMVIRTSIZE 配置参数

使用 SHMVIRTSIZE 配置参数来指定虚拟共享内存段的初始大小。

onconfig.std 值

与平台有关

如不存在

如果 SHMADD 存在：SHMADD 配置参数的值。

如果 SHMADD 不存在：8192。

值

32 位平台：正整数，最大值 2 GB

64 位平台：正整数，最大值 4 TB

由于操作系统限制，在一些平台上，最大值可能较小。您的 UNIX™ 平台的实际最大值，请参阅机器说明。

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

要确定 SHMVIRTSIZE 配置参数的适当值，请使用以下算法来确定共享内存虚拟部分的大小：

```
shmvirtsize = fixed overhead + ((stack size + heap) * number of users)
```

变量	使用的值
fixed overhead	这包括 AIO 向量的大小、排序内存、数据库空间备份缓冲区、字典大小、存储过程高速缓存的大小、直方图池、其他池和其他开销。 要获得固定开销的估计，请启动数据库服务器，并查看分配了多少附加的内存段（如果有的话）。当服务器启动时，您在系统上拥有的用户数将影响内存段分配。当启动服务器时： • 如果用户数是您环境的典型用户数，那么将内存段大小添加到 SHMVIRTSIZE 配置参数的当前值并重启服务器。 • 如果用户数是远少于您环境的典型用户数，那么您必须计算使用内存段的适当开销值。您可以通过将服务器启动时分配的附加内存段数除以当时服务器上拥有的用户数来确定每个用户消耗的内存段数。将每个用户的内存段数乘以系统上典型的用户数，将此内存段计算值添加到 SHMVIRTSIZE 配置参数的当前值并重启服务器。
stack size	在 32 位系统上，使用 32 KB 堆栈大小。在 64 位系统上，典型地使用 64 KB 堆栈大小。然而，一些 64 位系统使用不同的值。
heap	每个用户使用 30 KB。
number of users	使用您在服务器上预期的最大并发用户会话数。

如果可能，请创建一个大小超出日常处理要求的共享内存的虚拟部分。

使用 onstat -g seg 命令来确定峰值用量，并相应地降低 SHRVIRTSIZE 配置参数的值。

相关链接

DS_TOTAL_MEMORY 配置参数 在第106页

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页

STACKSIZE 配置参数 在第178页

《*SinoDB* 管理员指南》: 共享内存的虚拟部分

《*SinoDB* 性能指南》: 配置对内存利用率的影响

LOW_MEMORY_RESERVE 配置参数 在第128页

SINGLE_CPU_VP 配置参数

SINGLE_CPU_VP 配置参数指定数据库服务器是否仅使用一个 CPU 虚拟处理器运行。

onconfig.std 值

SINGLE_CPU_VP 0

值

0 = 使用多个 CPU VP 运行

1 = 使用一个 CPU VP 运行

生效

当数据库服务器关闭并重启时

用法

如果您想在数据库服务器启动时自动地增加 CPU VP 的数目，请通过将 SINGLE_CPU_VP 配置参数设置为 0 来禁用它。

将 SINGLE_CPU_VP 设置为非零，允许数据库服务器基于仅一个 CPU 虚拟处理器在运行的情况，使用优化代码。其使数据库服务器能够绕过许多必须在运行多个 CPU 虚拟处理器时才使用的互斥调用。

强烈建议，在数据库服务器仅运行一个 CPU 虚拟处理器时，设置此参数。依据应用程序与工作负载，设置此参数可将性能提升高达 10%。

如果设置 SINGLE_CPU_VP 为非零并试图添加 CPU 虚拟处理器，则您会收到下列消息之一：

```
onmode: failed when trying to change the number of classname VPs by n.
onmode: failed when trying to change the number of cpu virtual processors by n.
```

如果设置 SINGLE_CPU_VP 为非零并尝试使用 *num* 值大于 1 的 VPCLASS cpu 启动数据库服务器，则您会收到以下错误消息，且数据库服务器初始化失败：

```
Cannot have SINGLE_CPU_VP non-zero and CPU VPs greater than 1.
```

相关链接

《*SinoDB* 管理员指南》：在单处理器计算机上运行

VPCLASS 值和 SINGLE_CPU_VP 配置参数

SinoDB® 将用户定义的虚拟处理器类视作 CPU 虚拟处理器。如果您设置 *SINGLE_CPU_VP* 配置参数为非零值，则不可创建任何用户定义的虚拟处理器类。

使用用户定义的 VPCLASS

如果您设置此配置参数为非零值，然后尝试使用用户定义的 VPCLASS 启动数据库服务器，则您会收到以下错误消息，且数据库服务器初始化失败：

```
oninit: Cannot have SINGLE_CPU_VP non-zero and user-defined VP classes
```

使用 ***cpu*** VPCLASS

如果您设置此配置参数为非零值，然后尝试使用 *num* 值大于 1 的 VPCLASS *cpu* 启动数据库服务器，则您会收到以下错误消息，且数据库服务器初始化失败：

```
Cannot have SINGLE_CPU_VP non-zero and CPU VPs greater than 1.
```

SMX_COMPRESS 配置参数

使用 SMX_COMPRESS 配置参数来指定数据库服务器将数据从源数据库服务器发送到目标数据库服务器之前使用的压缩级别。

在缓慢的连接上，网络压缩节省网络带宽，但使用更多的 CPU 来压缩和解压缩数据。比较两个服务器的 SMX_COMPRESS 配置参数值，并更改到较高的压缩值。

onconfig.std 值

SMX_COMPRESS 0

值

-1 = 源数据库服务器绝不压缩数据，无论目标数据库服务器是否使用压缩。

0 = 仅当目标数据库服务器期待压缩数据时，源数据库服务器才压缩数据。

1 = 数据库服务器执行最小量的压缩。

9 = 数据库服务器执行最大可能的压缩。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SMX_NUMPIPES 配置参数

SMX_NUMPIPES 配置参数设置服务器多路复用器组（SMX）连接的管道数。

onconfig.std 值

SMX_NUMPIPES 1

值

1 - 32767 = SMX 连接的网络管道数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

高可用性集群和并行分片查询使用 SMX 连接。如果服务器之间的迟滞过久，则增加 SMX 管道数。

SMX_PING_INTERVAL 配置参数

使用 SMX_PING_INTERVAL 配置参数来指定超时间隔的秒数，于该间隔中，辅助服务器在服务器多路复用器组（SMX）连接中等待来自主服务器的活动。

onconfig.std 值

SMX_PING_INTERVAL 10

值

0 = 无限期等待。

1 和 60 之间且包含 1 和 60 的正整数 = 超时间隔的秒数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

在运行带有 "onmode","-wf SMX_PING_INTERVAL=*value*" 或 "onmode","-wm SMX_PING_INTERVAL=*value*" 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

如果在 SMX_PING_INTERVAL 配置参数所指定的时间长度期间且在 SMX_PING_RETRY 配置参数所指定的间隔数之后，辅助服务器都未收到任何消息，则辅助服务器打印错误消息到 online.log 并关闭 SMX 连接。如果在 online.log 中出现 SMX 超时消息，则您可增加 SMX_PING_INTERVAL 值、SMX_PING_RETRY 值或增加两者的值。

此配置参数仅适用于辅助服务器。如果在主服务器上设置 SMX_PING_INTERVAL，则在主服务器成为辅助服务器时才会生效。

如果高可用性集群中的辅助服务器的 onconfig 文件有以下条目，则辅助服务器将总共等待主服务器的活动 180 秒。如果在 180 秒期间未收到主服务器的活动，则辅助服务器会关闭 SMX 连接并在 online 日志中写入一个错误消息。

```
SMX_PING_INTERVAL 30
SMX_PING_RETRY 6
```

相关链接

SMX_PING_RETRY 配置参数 在第173页

《*SinoDB* 管理员指南》：设置 *SMX* 连接

SMX_PING_RETRY 配置参数

使用 SMX_PING_RETRY 配置参数来指定辅助服务器在未收到主服务器响应时重复 SMX_PING_INTERVAL 配置参数所指定超时间隔的最大次数。如果达到最大次数都未收到响应，辅助服务器将打印错误消息到 online.log 文件，并且关闭服务器多路复用器组（SMX）连接。

onconfig.std 值

SMX_PING_RETRY 6

值

任何正整数 = 重复超时间隔的最大次数。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

在运行带有 "onmode","-wf SMX_PING_RETRY=*value*" 或 "onmode","-wm SMX_PING_RETRY=*value*" 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

如果在 SMX_PING_INTERVAL 配置参数所指定的时间长度期间且在 SMX_PING_RETRY 配置参数所指定的间隔数之后，辅助服务器都未收到任何消息，则辅助服务器打印错误消息到 online.log 并关闭 SMX 连接。如果在 online.log 中出现 SMX 超时消息，则您可增加 SMX_PING_INTERVAL 值、SMX_PING_RETRY 值或增加两者的值。

此配置参数仅适用于辅助服务器。如果在主服务器上设置 SMX_PING_RETRY，则在主服务器成为辅助服务器时才会生效。

如果高可用性集群中的辅助服务器的 onconfig 文件有以下条目，则辅助服务器将总共等待主服务器的活动 60 秒。如果在 60 秒期间未收到主服务器的活动，则辅助服务器会关闭 SMX 连接并在 online 日志中写入一个错误消息。

```
SMX_PING_INTERVAL 12
SMX_PING_RETRY 5
```

相关链接

SMX_PING_INTERVAL 配置参数 在第172页

《*SinoDB* 管理员指南》：设置 *SMX* 连接

SP_AUTOEXPAND 配置参数

使用 SP_AUTOEXPAND 配置参数来启用或禁用块的自动创建或扩展。

onconfig.std 值

SP_AUTOEXPAND 1

值

0 = 禁用块的自动创建或扩展。

1 = 启用块的自动创建或扩展。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当启用了 SP_AUTOEXPAND 配置参数，且存储容器（例如，数据库空间）有一个定义的非零的创建大小或扩展大小时，容器是可自动扩展的。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》：自动空间管理

SP_THRESHOLD 配置参数

使用 SP_THRESHOLD 配置参数来定义在 SinoDB® 自动地运行任务以通过扩展空间中现有块或添加新的块来扩展空间之前可在存储空间中存在的最小空闲空间的 KB 量。

onconfig.std 值

SP_THRESHOLD 0

值

0 = 无阈值。禁用在空间低于阈值时运行存储空间监视（mon_low_storage）任务以添加空间的触发器。

1 - 50 = 阈值，其为存储空间中空闲 KB 的百分比。

如果该值是 50 或低于 50，则 SinoDB® 将该值解释为百分比（例如，10 = 10%，2.84 = 2.84%）。

1000 到块大小的最大值 = 阈值，其为 1000 KB 或是当前平台上块的大小的最大值。

如果该值是 1000 或更高，则 SinoDB® 将该值解释为特定的 KB 数。

值 50 - 1000 是无效的。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当设置 SP_THRESHOLD 配置参数为大于 0 的有效值时，在数据库空间、临时数据库空间、智能大对象空间、临时智能大对象空间或 BOLB 空间中的空闲空间低于该阈值时，内置调度程序任务（mon_low_storage）将自动地运行。

假设 SP_THRESHOLD 配置参数的值是 5.5，服务器解释为 5.5%。如果空闲页上的空间不足且空闲空间百分比低于 5.5%，并且保持低于那个级别直到 mon_low_storage 任务的下一次运行，则该任务将尝试扩展该空间。如果设置 SP_THRESHOLD 配置参数为 50000 且空间的空闲 KB 低于 50000，则 mon_low_storage 任务下一次运行时将扩展该空间。

值 0 关闭 mon_low_storage 任务，并且防止服务器扩展任何空间。然而，当所有空闲页用尽并需要更多时，值 0 不影响服务器扩展空间的能力。

SP_THRESHOLD 配置参数所指定的值适用于属于服务器的所有空间。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》：自动空间管理

SP_WAITTIME 配置参数

使用 SP_WAITTIME 配置参数来指定在返回空间用尽错误之前，线程等待数据库空间、临时数据库空间、物理日志空间、智能大对象空间、临时智能大对象空间或 BLOB 空间扩展的最大秒数。

onconfig.std 值

SP_WAITTIME 30

值

0 - 2147483647

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

服务器用来自动地添加或扩展块的时间的差异很大，其取决于不同因素，诸如块的大小、相关磁盘驱动的速度，以及系统上的负载。当 SinoDB® 自动地添加或扩展块以防止空闲空间低于 SP_THRESHOLD 配置参数所指定的阈值时，SinoDB® 会强制需要该空间的线程等待直到该空间可用。如果您要更改线程将等待更多空间的最大时间量，您可更改 SP_WAITTIME 配置参数的值。

仅当存储池包含条目时，线程才会等待存储空间扩展。如果该存储池是空的，则线程将不等待。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》：自动空间管理

SQL_LOGICAL_CHAR 配置参数

使用 SQL_LOGICAL_CHAR 配置参数来启用或禁用内置字符数据类型声明中大小规格的扩展。

onconfig.std 值

SQL_LOGICAL_CHAR OFF =（以字节为单位解释大小规格）

值

OFF = 声明的大小无扩展。

1 = 声明的大小无扩展。

2 = 使用 2 作为声明的大小的扩展因子。

3 = 使用 3 作为声明的大小的扩展因子。

4 = 使用 4 作为声明的大小的扩展因子。

ON = 使用 *M* 作为扩展因子，而 *M* 为当前数据库代码集中任何逻辑字符需要的以字节计的最大长度。根据 DB_LOCALE 设置，*M*的整数范围从 1（单字节语言环境中）到 4。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

对于在单字节语言环境中开发但在多字节语言环境中部署的应用程序，此特性可降低在数据输入操作期间截断多字节逻辑字符的风险。

在诸如 UTF-8 这些多字节代码集中或一些东亚语言的多字节代码集中，单个逻辑字符可能需要多于一个字节的存储空间。此参数的设置可导引 SQL 分析器将逻辑字符语义应用到这些内置字符数据类型的声明：

- BSON
- CHAR
- CHARACTER
- CHARACTER VARYING
- JSON
- LVARCHAR
- NCHAR
- NVARCHAR
- VARCHAR
- DISTINCT 类型，声明任一这些数据类型为基本类型
- ROW 类型（命名和未命名），包含这些数据类型的字段
- 集合类型（LIST、MULTISET 或 SET），包含这些类型作为元素。

您为这个参数指定的设置必须是下列值之一：

不论 SQL_LOGICAL_CHAR 配置参数设置为启用或禁用声明的存储大小的扩展，其设置都指定 SinoDB® 实例的所有会话如何解释数据类型声明。

扩展因子的自动调整大小

当 SQL_LOGICAL_CHAR 设置为有效的数字，且当前会话创建数据库时，SinoDB® 会将 SQL_LOGICAL_CHAR 值与用于数据库代码集的逻辑字符的最大字节数进行比较。

如果 SQL_LOGICAL_CHAR 设置大于最大的字节数，数据库使用该语言环境的最大值作为新的扩展因子，取代配置文件指定的因子。配置文件中的 SQL_LOGICAL_CHAR 设置保持不变，且继续作为其他用户数据库的默认扩展因子。

同样地，如果会话的 SQL_LOGICAL_CHAR 值自动地重置为一个数字，如上所述，但同一会话接下来连接到另一个数据库，其语言环境使用的代码集的逻辑字符要求比当前扩展因子更大的存储大小，则在用户会话连接到那个数据库时，SinoDB® 使用该该新代码集的最大字节数作为新的扩展因子，而不是使用当前的 SQL_LOGICAL_CHAR 设置。

当 SQL_LOGICAL_CHAR 设置为 ON 时，也会发生扩展因子的自动重置以匹配连接时 DB_LOCALE 指定的代码集中的最大逻辑字符大小，但是当 SQL_LOGICAL_CHAR 以两种方式设置为数字（1、2、3 或 4）时，ON 设置的影响与数据库服务器行为是不相同的：

- 如果 SQL_LOGICAL_CHAR 设置是 ON，则扩展因子可自动地重置为较小的值。
- SQL_LOGICAL_CHAR = 4 和 SQL_LOGICAL_CHAR = ON 是一样的。

在当前会话连接到一个数据库，其 DB_LOCALE 代码集中的最大逻辑字符所需的字节数比当前 SQL_LOGICAL_CHAR 设置还小时，如果您要使用较小的扩展因子，则必须将 SQL_LOGICAL_CHAR 设置为 ON，而不是一个数字。有效的扩展因子将总是小于或等于语言环境的最大字符大小。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 参考》: *SYSTABLES*

《*SinoDB SQL* 指南: 语法》: 单字节和多字节字符和语言环境

《*SinoDB GLS* 用户指南》: 数据定义语句

SQLTRACE 配置参数

使用 SQLTRACE 参数来控制 SQL 跟踪的启动环境。

onconfig.std 值

在 UNIX™ 上：未设置。禁用 SQL 跟踪。

在 Windows™ 上：#SQLTRACE level=low,ntraces=1000,size=2,mode=global

值

请参阅 "用法" 部分。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

在运行带有 set sql tracing 参数的 SQL 管理 API task() 或 admin() 函数之后。

用法

从 onconfig 移除 # 符号来保持有关任何用户运行的最后 1000 个 SQL 语句的基本信息，最多到 2 KB 大小。通过调整 SQLTRACE 配置参数的字段值，您可制定 SQL 跟踪信息的范围。

SQLTRACE 配置参数语法图

SQLTRACE
[level= { low | medium | high | off } ,]
[ntraces= { 1000 *number_traces* } ,]
[size= { 2 *buffer_size* } ,]
[mode= { global | user }]

表 69: SQLTRACE 配置参数值的选项

字段	值
level	跟踪的信息量：

字段	值
	• Low = 默认。捕获语句统计信息、语句文本和语句迭代器。 • Medium = 捕获包含在 low 级别跟踪中的所有信息，加上表名称、数据库名和存储过程堆栈。 • High = 捕获在 medium 级别跟踪中的所有信息，加上主机变量。 • Off = 指定无 SQL 跟踪。
ntraces	*number_traces* 值是重新使用资源之前要跟踪的 SQL 语句数。默认是 1000。范围是 500 - 2147483647。
size	*buffer_size* 值是要存储的可变长度数据大小的最大值，以 KB 为单位。默认是 2。范围是 1 -100。如果超过这个缓冲区大小，数据库服务器丢弃保存的数据。
mode	执行跟踪的范围： • Global = 默认。所有用户。 • User = 通过 SQL 管理 API task() 或 admin() 函数启用跟踪的用户。如果要获取一小部分用户正在运行的 SQL 样例，请指定这个模式。

onstat -g his 命令显示 SQL 跟踪信息。

相关链接

onstat -g his 命令: 打印 *SQL* 跟踪信息 在第463页
set sql tracing 参数: 设置全局 *SQL* 跟踪 *(SQL* 管理 *API)* 在第685页
《*SinoDB* 管理员指南》: 使用 *SQLTRACE* 配置参数指定启动 *SQL* 跟踪信息

STACKSIZE 配置参数

使用 STACKSIZE 配置参数来指定数据库服务器用户线程的堆栈大小。

onconfig.std 值

STACKSIZE 32 用于 32 位数据库服务器

STACKSIZE 64 用于 64 位数据库服务器

值

32 直到由数据库服务器配置和可用内存量来确定的限制

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

STACKSIZE 的值没有上限，但是设置太大的值会浪费虚拟内存空间，并导致交换空间问题。

对于 32 位平台，STACKSIZE 的默认 32 KB 足够用于非递归的数据库活动。对于 64 位平台，建议的 STACKSIZE 值是 64 KB。当数据库服务器执行递归数据库任务（如一些 SPL 例程中的递归数据库任务）时，它会检查堆栈大小溢出的可能性并自动地扩展堆栈。

用户线程执行用户定义的例程。要增加特定例程的堆栈大小，请在 CREATE FUNCTION 语句上使用 stack 修饰符。

警告: 将 STACKSIZE 的值设置的太低会导致堆栈溢出，虽未定义其结果，但其结果通常是不好的。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SHMVIRTSIZE 配置参数 在第169页

《*SinoDB* 管理员指南》: 堆栈

《*SinoDB SQL* 指南: 参考》: *INFORMIXSTACKSIZE* 环境变量

《*SinoDB SQL* 指南: 语法》: *CREATE FUNCTION* 语句

STATCHANGE 配置参数

使用 STATCHANGE 配置参数来指定更改阈值的全局百分率的正整数，让服务器在 UPDATE STATISTICS 操作的自动模式启用时用于确定分布统计信息是否符合更新条件。

onconfig.std 值

STATCHANGE 10

值

0 - 100

单位

更改阈值的百分率

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当 AUTO_STAT_MODE 配置参数、AUTO_STAT_MODE 会话环境变量，或者 UPDATE STATISTICS 语句的 AUTO 关键字启用了 UPDATE STATISTICS 操作的自动模式时，数据库服务器使用 STATCHANGE 配置参数的值。

当 UPDATE STATISTICS 操作的自动模式启动时，数据库服务器使用 STATCHANGE 设置所指定的更改阈值来确定是否分布统计信息符合更新条件。当该模式启用时，UPDATE STATISTICS 语句将比较 STATCHANGE 设置与自当前数据分布计算以来每个表或分段中更改行的百分比，并仅选择性地更新在 UPDATE STATISTICS 语句范围内每个表或分段丢失或陈旧的分布统计。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

AUTO_STAT_MODE 配置参数 在第73页

AUTO_STAT_MODE 配置参数 在第73页

《*SinoDB SQL* 指南: 语法》: *CREATE TABLE* 语句的统计信息选项

STMT_CACHE 配置参数

使用 STMT_CACHE 配置参数来确定数据库服务器是否使用SQL 语句高速缓存。

onconfig.std 值

STMT_CACHE 0

值

0 = 不使用 SQL 语句高速缓存（等同于 onmode -e OFF）。

1 = 启用 SQL 语句高速缓存，但是用户会话不使用该高速缓存。仅当用户将环境变量 STMT_CACHE 设置为 1 或执行 SQL 语句 SET STATEMENT CACHE ON，才可使用该高速缓存。

2 = 开启 SQL 语句高速缓存。高速缓存所有语句。要关闭语句高速缓存，请将环境变量 STMT_CACHE 设置为 0 或执行 SQL 语句 SET STATEMENT CACHE OFF。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

可以使用下列两种方式之一来启用 SQL 语句高速缓存：

- 总是使用 SQL 语句高速缓存，除非用户明确地指示不使用。将 STMT_CACHE 配置参数设置为 2 或执行 onmode -e ON。
- 仅当用户明确地指示使用 SQL 语句高速缓存时，才使用它。将 STMT_CACHE 配置参数设置为 1 或执行 onmode -e ENABLE。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -e: 更改 *SQL* 语句高速缓存的使用 在第338页

《*SinoDB SQL* 指南*:* 参考》*: STMT_CACHE* 环境变量

《*SinoDB* 性能指南》*:* 使用 *SQL* 语句高速缓存

STMT_CACHE_HITS 配置参数

使用 STMT_CACHE_HITS 配置参数来指定在语句完全地插入 SQL 语句高速缓存之前命中（引用）该语句的次数。

onconfig.std 值

STMT_CACHE_HITS 0

值

0 = 在 SQL 语句高速缓存中完全地插入所有限定语句

>0 = 用户首次发出唯一的语句时，数据库服务器在高速缓存中插入标识该语句的 *key-only* 条目。后续同样语句增加 *key-only* 高速缓存条目的命中数。当该 *key-only* 高速缓存条目的命中数达到指定的命中数时，数据库服务器将完全地在高速缓存中插入该语句。将 命中数 设置为 1 或更高来排除即席查询进入高速缓存。

单位

整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -W: 更改 *SQL* 语句高速缓存的设置 在第346页

onstat -g ssc 命令*:* 显示 *SQL* 语句出现次数 在第529页

《*SinoDB* 性能指南》*: SQL* 语句执行次数

onstat -g ssc 命令*:* 显示 *SQL* 语句出现次数 在第529页

《*SinoDB* 性能指南》*: SQL* 语句执行次数

STMT_CACHE_NOLIMIT 配置参数

使用 STMT_CACHE_NOLIMIT 配置参数来控制是否将限定语句插入 SQL 语句高速缓存中。

onconfig.std 值

STMT_CACHE_NOLIMIT 0

如不存在

1

值

0 = 防止语句插入高速缓存。如果高速缓存中大多数语句是当前在用的，则高速缓存可增长超过大小限制，因为高速缓存清除赶不上插入的速度。如果您担心内存使用情况，请关闭 STMT_CACHE_NOLIMIT 以防止数据库服务器分配大量内存给高速缓存。

1 = 不管 SQL 语句高速缓存大小如何，總是將語句插入高速缓存中。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -W: 更改 *SQL* 语句高速缓存的设置 在第346页

onstat -g ssc 命令*:* 显示 *SQL* 语句出现次数 在第529页

《*SinoDB* 性能指南》*: SQL* 语句执行次数

STMT_CACHE_NUMPOOL 配置参数

使用 STMT_CACHE_NUMPOOL 配置参数来指定 SQL 语句高速缓存的内存池数。要获得有关这些内存池的信息，请使用 onstat -g ssc pool。

由于数据库服务器不会把所有语句插入从存内池分配内存的高速缓存中，所以高速缓存大小可能比内存池的总大小要小一些。

onconfig.std 值

STMT_CACHE_NUMPOOL 1

值

1 - 256

单位

正整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

onstat -g ssc 命令*:* 显示 *SQL* 语句出现次数 在第529页

《*SinoDB* 性能指南》*: SQL* 语句执行次数

STMT_CACHE_SIZE 配置参数

使用 STMT_CACHE_SIZE 配置参数来指定以 KB 为单位的 SQL 语句高速缓存的大小。新的高速缓存大小要在下一次添加语句到高速缓存时，才会生效。

onconfig.std 值

STMT_CACHE_SIZE 512

值

正整数

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 性能指南》：监视和调整 *SQL* 语句高速缓存的大小

STOP_APPLY 配置参数

使用 STOP_APPLY 配置参数来停止 RS 辅助服务器应用从主服务器收到的日志文件。

onconfig.std 值

STOP_APPLY 0

默认值

0

值

0 = 应用日志

1 = 立即停止应用日志

YYYY:*MM*:*DD*-*hh*:*mm*:*ss* = 在指定的时间停止日志应用，此处：

- *YYYY* = 年
- *MM* = 月
- *DD* = 日
- *hh* = 时（24 小时制）
- *mm* = 分
- *ss* = 秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

停止日志文件的应用允许您通过从 RS 辅助服务器恢复数据来迅速地从错误的数据库修改恢复。您可配置服务器立即或者在指定的时间停止日志应用。当设置了 STOP_APPLY 的值，您还必须设置 LOG_STAGING_DIR。如果配置了 STOP_APPLY 而未将 LOG_STAGING_DIR 设置为一个有效且安全的目录，则无法初始化服务器。

日志文件以二进制格式存储在 LOG_STAGING_DIR 配置参数指定的目录中。您必须为日志文件指定一个有效且安全的位置。

要查看关于发送到 RS 辅助服务器的日志暂存目录集的数据的信息，请在 RS 辅助服务器上运行 onstat -g rss verbose 命令。

如果暂存文件的写入失败，则 RS 辅助服务器会发起事件报警 40007。

假定为 STOP_APPLY 配置参数指定的时间值与 RS 辅助服务器在同一个的时区中。

dbexport 实用程序不支持在可更新辅助服务器上的写操作，除非设置了 STOP_APPLY 参数。（除了 STOP_APPLY，还必须设置 UPDATABLE_SECONDARY 和 USELASTCOMMITTED 配置参数，才可通过 dbexport 启用辅助数据复制服务器上的写操作。）

如果远程独立辅助（RSS）服务器将其 STOP_APPLY 配置参数设置为一个非零的值，则那台服务器不可使用集群事务协调。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
DELAY_APPLY 配置参数 在第95页
UPDATABLE_SECONDARY 配置参数 在第190页
CLUSTER_TXN_SCOPE 配置参数 在第84页
《*SinoDB SQL 指南: 语法*》: *CLUSTER_TXN_SCOPE 会话环境选项*
LOG_STAGING_DIR 配置参数 在第125页
onstat -g cluster 命令: 打印高可用性集群信息 在第440页
DELAY_APPLY 配置参数 在第95页

STORAGE_FULL_ALARM 配置参数

使用 STORAGE_FULL_ALARM 配置参数来配置当存储空间填满时消息和报警的频率和严重程度。

onconfig.std 值

STORAGE_FULL_ALARM 600,3

值

seconds = 0（关）或指明通知之间秒数的正整数。

severity_level = 0（无报警）或 1 - 5

单位

seconds, *severity_level*

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

当诸如数据库空间、智能大对象空间、BLOB 空间或表空间等存储空间或分区填满时，发起报警并发送消息到 online 消息日志。您可使用这个参数的第一个值指定通知之间的秒数。您可以为返回的事件报警指定最低的严重程度。设置特定的严重程度防止发起严重程度更低的事件。但会发出与指定严重程度相同或更严重的事件。通过将这个参数设置为 0 可以防止当存储空间填满时报警。

无论 STORAGE_FULL_ALARM 的值如何，当存储空间或分区填满时都将消息发送到 online 消息日志。

相关链接

事件警报参数 在第724页
《*SinoDB 管理员指南*》: 监视存储空间

SYSALARMPROGRAM 配置参数

使用 SYSALARMPROGRAM 配置参数来指定 evidence.sh 脚本的全路径名。当数据库服务器发生故障时，它会执行 evidence.sh。您可使用 evidence.sh 脚本的输出来诊断数据库服务器故障的原因。

onconfig.std 值

在 UNIX™ 上：$INFORMIXDIR/etc/evidence.sh
在 Windows™ 上：未设置。列为 $INFORMIXDIR\etc\evidence.bat

值

pathname = evidence.sh 脚本的全路径名。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

在 Windows™ 上，您必须启用 evidence.bat 的命令扩展来成功地完成。通过发出以下命令，您可启用或禁用您正在使用的命令提示的扩展：

- 启用：cmd /x
- 禁用：cmd /y

您还可从 Windows™ XP 注册表来启用和禁用命令扩展：

表 70: 从 Windows™ 注册表来启用命令扩展

属性	值
Hive	HKEY_CURRENT_USER
Key	Software\Microsoft\Command Processor
Name	EnableExtensions
Type	REG_DWORD
Values	0（禁用），1（启用）

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SYSSBSPACENAME 配置参数

使用 SYSSBSPACENAME 配置参数来指定数据库服务器将分段级别数据分布统计信息（syfragsdist 系统目录表在其 encsdist 列中将其存储为 BLOB 对象）的智能大对象空间的名称。还使用 SYSSBSPACENAME 来指定数据库服务器存储 UPDATE STATISTICS 语句为特定用户定义的数据类型收集的统计信息的智能大对象空间的名称。

onconfig.std 值

未设置。

如不存在

0

值

最多 128 字节。SYSSBSPACENAME 必须是唯一的，以字母或下划线开头，且仅包含数字、字母、下划线或 $ 字符。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

参考

- 更新统计信息，在《*SinoDB*® 性能指南》中关于个别查询性能的章节中
- 智能大对象空间特徵，在《*SinoDB*® 性能指南》中关于配置对 I/O 影响的章节中
- 写用户定义的统计信息，在《*SinoDB*® 用户自定义例程和数据类型开发者指南》中的性能章节中
- 提供列的统计信息数据，在《*SinoDB*® *DataBlade*® *API* 程序员指南》中

用法

要支持分段级别统计信息，您必须指定一个智能大对象空间的名称作为 SYSSBSPACENAME 的设置，并且必须那个智能大对象空间（通过使用 onspaces 实用程序，如下所述。对于 STATLEVEL 属性设置为 FRAGMENT 的任何表，如果未设置 SYSSBSPACENAME 或 SYSSBSPACENAME 所指定的智能大对象空间未适当分配，则数据库服务器返回错误）。

对于分段表中一列的分布统计信息，您可根据这个公式估计智能大对象空间所需的存储容量为多少字节：

```
nfrags * 1.25 * ((10000 / resolution) * ((2 * column_width) + 6))
```

此处 1.25 接近溢出箱的数目。该公式还包括这些变量：

- *column_width* 是 UPDATE STATISTICS 指定列的以字节计的宽度
- *nfrags* 是表的分段数。
- *resolution* 是 UPDATE STATISTICS 语句的 resolution 子句中计算分布的 *percent* 值。

resolution 也是 dbschema -hd *table* 命令显示的该列分布统计信息。

SYSSBSPACENAME 也 指定数据库服务器存储 UPDATE STATISTICS 语句为特定用户定义的数据类型收集的统计信息的智能大对象空间的名称。通常，数据库服务器将统计信息存储在 sysdistrib 系统目录表中。

请不要将 SYSSBSPACENAME 配置参数与 SBSPACENAME 配置参数混淆。

因为用户定义的数据类型的数据分布可以很大，所以您可以选择将它们存储在智能大对象空间，而不是 sysdistrib 系统目录表。如果将数据分布存储在智能大对象空间，请使用 DataBlade® API 或 SinoDB® ESQL/C 函数来检查统计信息。

即使以 SYSSBSPACENAME 参数指定了一个智能大对象空间，在可以使用它之前，您必须使用 onspaces 实用程序的 -c -S 选项来创建该智能大对象空间。当发生下列情况之一时，数据库服务器会验证这个智能大对象空间的名称：

- 数据库服务器在执行带有 MEDIUM 或 HIGH 关键词的 UPDATE STATISTICS 语句时，尝试将多重表示类型的数据分布写到 SYSSBSPACENAME。
- 数据库服务器在执行带有 DROP DISTRIBUTIONS 关键词的 UPDATE STATISTICS 语句时，尝试将多重表示类型的数据分布从 SYSSBSPACENAME 删除。

如果未设置 SBSSPACENAME或未分配存储给那个智能大对象空间，则数据库服务器可能无法存储分布统计信息，以致于 UPDATE STATISTICS 操作以错误 -9814 失败。

虽然您可以将智能大对象存储在 SYSSBSPACENAME 所指定的智能大对象空间，但建议将分布统计信息和智能大对象保存在个别的智能大对象空间，因为：

- 可避免当查询正在访问智能大对象，且查询优化器正在使用分布来确定查询计划时发生磁盘争用。
- 当每个智能大对象空间用于不同目的时，磁盘空间花费较长时间才填满。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

SBSPACENAME 配置参数 在第155页

智能大对象空间结构 在第270页

onspaces -c -S: 创建智能大对象空间 在第371页

TBLSPACE_STATS 配置参数

使用 TBLSPACE_STATS 配置参数来开启或关闭表空间统计信息的收集。使用 onstat -g ppf 命令来罗列表空间统计信息。

onconfig.std 值

TBLSPACE_STATS 1

值

0 = 关闭表空间统计信息的收集。onstat -g ppf 命令显示 partition profiles disabled。

1 = 开启表空间统计信息的收集。

单位

整数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

onstat -g ppf 命令: 打印分区概要文件 在第491页

TBLTBLFIRST 配置参数

如果要在根数据库空间中指定表空间 tblspace 的第一个扩展数据块大小，则使用 TBLTBLFIRST 配置参数。如果您不要数据库服务器自动地管理扩展数据块大小，则设置这个参数。

onconfig.std 值

TBLTBLFIRST 0

值

从相当于 250 页（以 KB 计），到第一个块大小减去任一系统对象所需的空间。

单位

以 KB 为单位的页大小的倍数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

您可能要指定第一个和下一个扩展数据块大小，以减少表空间 tblspace 扩展数据块的数目，并减少需要将表空间 tblspace 扩展数据块放置到非主块中的频率。（主块是数据库空间中的初始块。）

您可使用 oncheck -pt 和 oncheck -pT 来显示表空间 tblspace 的第一个和下一个扩展数据块大小。

如果想要配置非根数据库空间的第一个扩展数据块，请使用 onspaces 实用程序。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

TBLTBLNEXT 配置参数 在第187页
oncheck -pt 和 *-pT:* 显示表或分段的表空间 在第308页
onspaces 实用程序 在第362页
《*SinoDB* 管理员指南》*:* 为表空间 *tblspace* 指定第一个和下一个扩展数据块大小*e*
TBLTBLNEXT 配置参数 在第187页
onspaces -c -d: 创建数据库空间 在第367页

TBLTBLNEXT 配置参数

TBLTBLNEXT 配置参数指定根数据库中表空间 tblspace 的下一个扩展数据块大小。如果您不想要数据库服务器自动地管理扩展数据块大小，则请设置这个参数

onconfig.std 值

TBLTBLNEXT 0

值

从相当于以 KB 指定的四页，到最大块大小减去三页

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

如果在主块中没有下一个扩展数据块的足够空间，则从另一个块分配该扩展数据块。如果指定的空间不可用，则分配最近的可用空间。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
TBLTBLFIRST 配置参数 在第186页
TBLTBLFIRST 配置参数 在第186页
《*SinoDB* 管理员指南》*:* 为表空间 *tblspace* 指定第一个和下一个扩展数据块大小
onspaces -c -d: 创建数据库空间 在第367页

TEMPTAB_NOLOG 配置参数

使用 TEMPTAB_NOLOG 配置参数来在临时表上禁用日志记录。

onconfig.std 值

TEMPTAB_NOLOG 0

值

0 = 在临时表操作上启用逻辑日志记录

1 = 在临时表操作上禁用逻辑日志记录

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

由于这个参数防止 SinoDB® 在网络上传输临时表，因此它可提高应用程序中的性能。可使用 onmode -wf 实用程序动态地更新该设置。

如果启用了这个设置，则请注意，因为在使用临时表时不记录任何数据，所以回滚临时表上的事务将不再撤消临时表中的工作。

对于高可用性集群中的 HDR、RSS 和 SDS 辅助服务器，应总是通过设置 TEMPTAB_NOLOG 配置参数为 1 来禁用临时表上的逻辑日志记录。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

TENANT_LIMIT_CONNECTIONS 配置参数

TENANT_LIMIT_CONNECTIONS 配置参数指定租户数据库的最大连接数。

onconfig.std 值

0（闭）

如不存在

0（闭）

值

1 - 65536

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

当达到限制时，后续到租户数据库的连接要求将被拒绝。

通过 tenant create 或 tenant update SQL API 命令设置的 tenant_limit_connections 租户数据库属性优先于 TENANT_LIMIT_CONNECTIONS 配置参数设置。

相关链接

tenant create 参数: 创建租户数据库 *(SQL* 管理 *API)* 在第705页

tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页

TENANT_LIMIT_MEMORY 配置参数

TENANT_LIMIT_MEMORY 配置参数指定连接到租户数据库的所有会话的最大共享内存量。

onconfig.std 值

0（闭）

如不存在

0（闭）

值

102400 - 2147483648

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

当超过这个限制时，将终止使用最多共享内存的会话。值范围从 100 MB 到 2 TB，但必须指定为以 KB 表示的整数。

通过 tenant create 或 tenant update SQL 管理 API 命令设置的 tenant_limit_memory 租户数据库属性优先于 TENANT_LIMIT_MEMORY 配置参数设置。

相关链接

tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页

TENANT_LIMIT_SPACE 配置参数

TENANT_LIMIT_SPACE 配置参数指定租户数据库可用的最大存储空间量。存储空间包括所有永久数据库空间、BLOB 空间和智能大对象空间。

onconfig.std 值

0（闭）

如不存在

0（闭）

值

1048576 - 1717986918400

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

TENANT_LIMIT_SPACE 配置参数限制租户数据库可用的永久存储空间的量，并可在租户数据库环境内保存系统资源。当达到限制时，将拒绝后续需要额外磁盘空间的操作。 值范围是从 1 GB 到 200 TB，但必须指定以 KB 为单位表示的整数。

通过 tenant create 或 tenant update SQL 管理 API 命令设置的 tenant_limit_space 租户数据库属性优先于 TENANT_LIMIT_SPACE 配置参数设置。

相关链接

tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页

tenant update 参数: 更改租户数据库属性 (SQL 管理 API) 在第710页

《SinoDB 管理员指南》: 多租户

《SinoDB 管理员指南》: 管理租户数据库

《SinoDB 管理员指南》: 限制会话资源

tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页

tenant update 参数: 更改租户数据库属性 (SQL 管理 API) 在第710页

TXTIMEOUT 配置参数

使用 TXTIMEOUT 配置参数来指定两阶段提交中参与者启动参与者恢复之前等待的时间量。这个参数仅用于涉及远程数据库服务器的分布查询。

onconfig.std 值

TXTIMEOUT 300

值

正整数

单位

秒

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 管理员指南》: 两阶段提交协议如何处理故障

UNSECURE_ONSTAT 配置参数

使用 UNSECURE_ONSTAT 配置参数来移除数据库系统管理员（DBSA）用户对 onstat 命令的访问限制

onconfig.std 值

未设置。

值

1 = 所有用户可运行 onstat 命令来查看运行的 SQL 语句

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

在默认情况下，显示活动会话的 SQL 语句文本的 onstat 命令仅限于 DBSA 用户。要移除这个限制，请将 UNSECURE_ONSTAT 配置参数设置为 1。显示 SQL 语句的 onstat 命令包括 onstat -g his、onstat -g ses、onstat -g stm、onstat -g ssc 和 onstat -g sql。

UPDATABLE_SECONDARY 配置参数

使用 UPDATABLE_SECONDARY 配置参数来设置主服务器和辅助服务器之间建立的连接数。设置这个配置参数使得客户端应用程序能够在高可用性辅助服务器上执行更新、插入和删除操作。

onconfig.std 值

UPDATABLE_SECONDARY 0

值

从零（默认值）直到双倍于 CPU VP 数目的任何数。将该值设置为 0 配置辅助服务器为只读。将该值设置为 1 到两倍于 CPU VP 数目，使辅助服务器为可更新的并配置连接线程。

单位

一给定辅助服务器与其主服务器之间的网络连接数

生效

在编辑 onconfig 文件并重启数据库服务器之后。

辅助数据复制服务器的隔离级别

如果 UPDATABLE_SECONDARY 配置参数未设置或设置为零，则辅助数据复制服务器为只读。在这种情况下，在辅助服务器上仅可用 DIRTY READ 或 READ UNCOMMITTED 事务隔离级别。

如果 UPDATABLE_SECONDARY 参数设置为大于零的有效连接数，则辅助数据复制服务器可支持 COMMITTED READ、COMMITTED READ LAST COMMITTED 或 COMMITTED READ 事务隔离级别，或者 USELASTCOMMITTED 会

话环境变量。仅SQL DML 语句，诸如 INSERT、UPDATE、MERGE 和 DELETE，和 dbexport 实用程序可在可更新的辅助服务器上支持写操作。（除了 UPDATABLE_SECONDARY 以外，还必须设置 STOP_APPLY 和 USELASTCOMMITTED 配置参数才可在辅助数据复制服务器上通过 dbexport 启用写操作。）

相关链接

《*SinoDB* 管理员指南》: 辅助服务器上的数据库更新

《*SinoDB* 管理员指南》: 设置 *SMX* 连接

STOP_APPLY 配置参数 在第182页

onstat -g cluster 命令: 打印高可用性集群信息 在第440页

USELASTCOMMITTED 配置参数

使用 USELASTCOMMITTED 配置参数来指定隔离级别，COMMITTED READ 隔离级别的 LAST COMMITTED 特征是隐含有效的。

onconfig.std 值

USELASTCOMMITTED "NONE"

default value默认值

"NONE"

值

"NONE" = 位标识隔离级别。如果在 Committed Read、Dirty Read、Read Committed 或 Read Uncommitted 级别中当您的会话尝试读取一行时遇到互斥锁，则直到持有该互斥锁的并发事务提交或回滚之前您的事务无法读取那个行。

"COMMITTED READ" = 所有来自 Committed Read 隔离级别的事务都视为最后提交的事务。当数据库服务器尝试读取 Committed Read 或 Read Committed 隔离级别中的一行时遇到互斥锁，会读取该数据的最近提交版本。

"DIRTY READ" = 所有来自 Dirty Read 隔离级别的事务都视为最后提交事务。当数据库服务器尝试读取 Dirty Read 或 Read Uncommitted 隔离级别中的一行时遇到互斥锁，会读取该数据的最近提交版本。

"ALL" = 所有来自 Committed Read 和 Dirty Read 隔离级别的事务都视为最后提交事务。当数据库服务器尝试读取 Committed Read、Dirty Read、Read Committed 或 Read Uncommitted 隔离级别中的一行时遇到互斥锁，会读取该数据的最近提交版本。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

LAST COMMITTED 特征可降低在有互斥锁的表上并发事务之间的锁定冲突的风险。USELASTCOMMITTED 配置参数还可 SET TRANSACTION 语句的 READ COMMITTED 和 READ UNCOMMITTED 隔离级别启用 LAST COMMITTED 语义。

USELASTCOMMITTED 配置参数仅适用于已创建或更改为以 ROW 作为锁定粒度的表。没有以任何明显锁模式设置创建的表使用 DEF_TABLE_LOCKMODE 中的默认设置。如果 DEF_TABLE_LOCKMODE 设置为 PAGE，则 USELASTCOMMITTED 配置参数不能启用对未提交事务持有互斥锁的表中最近提交数据的访问权限，除非这些表的锁定粒度明确地变更 ROW 级别。

与共享磁盘辅助数据库服务器一起使用

在共享磁盘（SD）辅助数据库服务器上，USELASTCOMMITTED 配置参数也是有效的。下表显示 SD 辅助服务器上 USELASTCOMMITTED 配置参数的有效值及其描述。

表 71: 有效的辅助服务器 USELASTCOMMITTED 值

USELASTCOMMITTED 值	描述
NONE	COMMITTED READ LAST COMMITTED 不是会话的默认隔离级别
COMMITTED READ	COMMITTED READ LAST COMMITTED 是所有带有 Committed Read 隔离的会话的默认隔离级别
DIRTY READ	COMMITTED READ LAST COMMITTED 是所有带有 Dirty Read 隔离的会话的默认隔离级别
ALL	COMMITTED READ LAST COMMITTED 是所有带有 Committed Read 或 Dirty Read 隔离的会话的默认隔离级别

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

DEF_TABLE_LOCKMODE 配置参数 在第94页

《*SinoDB SQL 指南: 语法*》: *USELASTCOMMITTED 会话环境选项*

《*SinoDB SQL 指南: 语法*》: *SET ISOLATION 语句*

USEOSTIME 配置参数

使用 USEOTIME 配置参数来控制数据库服务器从操作系统获取当前时间时是否使用亚秒精度。

onconfig.std 值

USEOSTIME 0

值

0 = 闭

1 = 开

生效

初始化期间

参考

- 《*SinoDB®* 性能指南》
- 《*SinoDB® SQL* 指南: 语法》中使用 CURRENT 函数来返回日期时间值

用法

USEOSTIME 设置为 1 指定数据库服务器从操作系统为 SQL 语句获取当前时间时使用亚秒精度。以下示例显示日期时间值中的亚秒：

```
2001-09-29 12:50:04.612
```

如果不需要亚秒精度，则数据库服务器每秒从操作系统检索当前时间一次，使客户端应用程序的时间精度为一秒。如果 USEOSTIME 设置为 0，则当前函数对年的分数字段返回零（.000）。

当数据库服务器的主机计算机有亚秒精度的时钟时，则其 SQL 语句依赖于亚秒精度的应用程序应将 USEOSTIME 设置为 1。

与关闭 USEOSTIME 的运行系统相比，将发现 USEOSTIME 设置为非零的运行系统的性能会下降达到 4% 至 5%。

这个设置不影响任何从应用程序到 SinoDB® 内嵌语言库函数关于时间的调用。

USERMAPPING 配置参数 (UNIX™, Linux™)

使用 USERMAPPING 配置参数来设置数据库服务器是否接受映射用户的连接。

默认值

OFF

值

OFF = 仅在具有登录服务的 SinoDB® 主机计算机 OS 中注册的用户可以连接到数据库服务器。在 SinoDB® 主机计算机上没有 OS 帐号的外部认证的用户不可连接到数据库服务器资源。

BASIC = 没有 OS 帐号的用户可以连接到 SinoDB®。即使没有 OS 帐号的用户映射到服务器管理员用户或组 ID，也不能在数据库服务器上不能执行有权限用户的操作。

ADMIN = 没有 OS 帐号的用户可怜接到 SinoDB®。如果用户以一个有权限用户身份得到验证，且映射到适当的服务器管理员组 ID，则该用户可在数据库服务器上执行 DBSA、DBSSO 或 AAO 的工作。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

当通过设置该参数为 BASIC 或 ADMIN 值开启 USERMAPPING 时，在 SinoDB® 主机计算机上没有操作系统（OS）帐号的外部验证的用户可以访问数据库服务器资源。BASIC 或 ADMIN 设置还确定映射用户能否获得管理权限。

重要: 将 USERMAPPING 配置参数从 OFF 更改为 ADMIN 或 BASIC 不是为映射用户设置 SinoDB® 的唯一步骤。要以适当的用户属性映射用户，您还必须使用诸如 CREATE USER 和 ALTER USER 这样的 DDL 语句，在 SYSUSER 数据库的适当系统表中来注册这些值。根据使用的 DDL 语句和定义的表映射，将更新与填入下列表：

- SYSINTAUTHUSERS
- SYSUSERMAP
- SYSSURORGATES
- SYSSURROGATEGROUPS

USRC_HASHSIZE 配置参数

USRC_HASHSIZE 配置参数指定 LBAC 凭证内存高速缓存中散列存储区的数目。这个内存高速缓存持有有关用户 LBAC 凭证的信息。

onconfig.std 值

USRC_HASHSIZE 31

值

任何正整数

单位

KB

生效

在编辑 onconfig 文件并重启数据库服务器之后。

相关链接

《*SinoDB* 安全指南》: 维护基于标号的访问控制实现

USRC_POOLSIZE 配置参数

USRC_POOLSIZE 配置参数指定 LBAC 凭证内存高速缓存的每个散列存储区中条目的最大数。这个内存高速缓存持有有关用户 LBAC 凭证的信息。

onconfig.std 值

USRC_POOLSIZE 127

值

正值 127 或更大，表示高速缓存中初始最大条目数的一半。该最大值取决于服务器实例的共享内存配置和可用共享内存。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wm 命令来增加内存中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

高速缓存中初始条目数是 USRC_POOLSIZE 配置参数的值的两倍。例如，如果 USRC_POOLSIZE 配置参数设置为 127，则在高速缓存中允许 254 个条目。如果高速缓存中所有条目填满，则该高速缓存大小自动地增长 10%。要减小高速缓存的大小，则降低 onconfig 文件中 USRC_POOLSIZE 配置参数的值，并且重启服务器。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB* 安全指南》: 维护基于标号的访问控制实现

USTLOW_SAMPLE 配置参数

使用 USTLOW_SAMPLE 配置参数可以基于以 LOW 模式运行 UPDATE STATISTICS 语句时的采样生成索引统计信息。

对于有多过 100 K 叶子页的索引，使用采样收集统计信息可提高 UPDATE STATISTICS 操作速度。

onconfig.std 值

USTLOW_SAMPLE 1

值

0 = 禁用采样

1 = 启用采样

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode -wf#-wm: 动态更改某些配置参数 在第347页

《*SinoDB SQL* 指南: 语法》: *USTLOW_SAMPLE* 环境选项

《*SinoDB* 性能指南》: 更新统计信息操作期间的数据采样

VP_MEMORY_CACHE_KB 配置参数

使用 VP_MEMORY_CACHE_KB 参数来为每个 CPU 虚拟处理器和租户虚拟处理器创建私有内存高速缓存

onconfig.std 值

VP_MEMORY_CACHE_KB 0

值

0 = Off

所有私有内存高速缓存的合计大小，可选地后跟逗号和高速缓存模式。

大小，以 KB 计：

- 800 到 SHMTOTAL 配置参数值的 40%。

模式：

- STATIC = 默认。该指定的大小是所有私有内存高速缓存的结合大小的最大值。
- DYNAMIC = 该指定的大小是所有私有内存高速缓存 的初始大小。该高速缓存大小动态地改变，但不会超过 SHMTOTAL 配置参数的值。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

用法

私有内存高速缓存可提升 SinoDB® 服务器中线程分配的内存的性能。私有内存高速缓存不影响分配给缓冲池或共享内存通信使用的内存。

当 VP_MEMORY_CACHE_KB 配置参数的值设置为非零的数，则为每个 CPU 虚拟处理器和租户虚拟处理器创建一个私有内存高速缓存。在默认情况下，所有私有内存高速缓存结合的大小限制在指定的 KB 数。

如果您想要每个私有内存高速缓存的大小按需要自动地增加和减少，那么在大小的后面包含一个逗号和 DYNAMIC 词，例如， VP_MEMORY_CACHE_KB 1000,DYNAMIC。虽然所有私有内存高速缓存结合的最大初始大小不能超过 SHMTOTAL 配置参数的值的 40%，但使用 DYNAMIC 模式设置，高速缓存的大小可扩展超出初始限制。高速缓存的合计大小不能超过 SHMTOTAL 配置参数的值。当许多线程同时间断开或有共享内存锁争用时，DYNAMIC 模式可提升性能。可使用 onstat -g wmx 命令来显示有关互斥锁，例如共享内存锁互斥 shmcb sh_lock，以及等待互斥锁的线程的信息。

> 注意： 繁忙系统上的动态内存高速缓存可快速增长，并使用大量可用内存。因此，如果您将模式设置为 DYNAMIC，则请将 SHMTOTAL 配置参数设置为一个特定限制，而不是默认值 0，默认值 0 不限制内存量。

如果您重置 VP_MEMORY_CACHE_KB 配置参数为 0，则内存高速缓存被清空并禁用。

onstat -g vpcache 命令返回关于私有内存高速缓存的统计信息。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -g vpcache 命令: 显示 *CPU* 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息 在第536页

《*SinoDB* 性能指南》: 专用内存高速缓存

onstat -g vpcache 命令: 显示 *CPU* 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息 在第536页

scheduler lmm enable 参数: 指定自动低内存管理设置 *(SQL* 管理 *API)* 在第677页

VPCLASS 配置参数

使用 VPCLASS 配置参数来创建和配置虚拟处理器。

onconfig.std 值

UNIX™：VPCLASS cpu,num=1,noage

Windows™：

```
VPCLASS cpu,num=1,noage
#VPCLASS aio,num=1
#VPCLASS jvp,num=1
```

值

最多 128 字节字符。每个 VPCLASS 配置参数值必须是唯一的，以字母或下划线开始，并仅包含数字、字母、下划线或 $ 字符。不可包含空格。请参阅 "用法" 章节。

分隔符

使用逗号分隔每个字段。

生效

在编辑 onconfig 文件并重启数据库服务器之后。

用法

您可在 onconfig 文件中添加多个 VPCLASS 配置参数条目。每个 VPCLASS 配置参数必须描述一个不同的虚拟处理器类。一行一个定义。

VPCLASS 配置参数语法图

```
VPCLASS
{ class  <cpu class>   | aio   [autotune]  = { 0  | 1 }  | user_defined [ , noyield ] }
[ , num = number_vps ]
[ , max = maximum ]
cpu class
cpu class
cpu
[ , aff = ( { , processor  |  start  -  end [ / increment ] } ) ]
[ , noage ]
[ ,   [autotune]  = { 0  | 1 } ]
```

表 72: VPCLASS 配置参数值的选项

字段	值
class	*class* 值是虚拟处理器类的名。数据库服务器根据需要启动大多数虚拟处理器。典型地，您可能为 CPU、AIO、JVP 和用户定义的虚拟处理器类设置 VPCLASS 配置参数。 虚拟处理器类名不区分大小写。 类名的列表，请参阅《*SinoDB*® 管理员指南》中的虚拟处理器类。
user_defined	*user_defined* 值是您为用户定义的例程创建的虚拟处理器类的名称。 确保 SINGLE_CPU_VP 配置参数设置为 0。
autotune	指定数据库服务器是否根据需要为特定类添加虚拟处理器，以提升性能，如果包含 max 选项，则最多达到该选项的值。 • autotune=0 防止自动添加虚拟处理器

字段	值
	• autotune=1 启用自动添加虚拟处理器 如果类是 cpu，则自动添加的任何 CPU 虚拟处理器都不是专用。忽略 aff 选项。
cpu	指定 CPU 虚拟处理器类。
num	*number_vps* 值设置数据库服务器启动时所启动的特定类的虚拟处理器的数目。默认值是 1。cpu 和 aio 虚拟处理器累的值范围是 1 - 10000。所有其他虚拟处理器类的值范围是 0 - 10000。 您可使用 onmode -p 命令来为当前会话的类添加虚拟处理器。
max	*maximum* 值指定数据库服务器可为该类启动的虚拟处理器的最大数目。该值可以是大于零的任何整数。默认情况下，该数目是无限制的。
aff	在支持处理器专用的多处理器计算机上，aff 选项指定数据库服务器绑定 CPU 虚拟处理器的 CPU。操作系统将 CPU 编号，从 0 到 CPU 数减一。默认情况下，以循环方式将 CPU 虚拟处理器分派给可用的处理器。aff 选项为一个或多个整数： • *processor* = 绑定 CPU 虚拟处理器的 CPU 编号。CPU 编号可以任何顺序罗列。 • *start* = CPU 编号范围的起始。 • *end* = CPU 编号范围的终止。 • *increment* = 指定使用范围中哪个 CPU 编号的因子。例如，aff=(1-5/2) 指定使用 CPU 编号 1、3 和 5。
noage	如果操作系统实现优先级迟滞，则禁用 CPU 虚拟处理器的优先级迟滞。默认情况下，优先及迟滞是有效的。
noyield	指定用户定义的虚拟处理器类不让步，允许 C UDR 让步给其他需要访问用户定义的虚拟处理器类的线程。默认情况下，用户定义的虚拟处理器的线程让步。 非让步用户定义的虚拟处理器类以给予该用户定义的例程独占使用虚拟处理器类的方式来运行该例程。使用非让步虚拟处理器类的用户定义的例程串行地运行，并不将该虚拟处理器让给其他线程。 因为 UDR 在单个虚拟处理器上运行直到完成且任何其他虚拟处理器处于空闲状态，在非让步用户定义的虚拟处理器类中仅指定一个虚拟处理器。

这些选项可以任何顺序出现，以逗号分隔。

使用 onmode -p 命令来动态地添加或移除当前数据库会话的虚拟处理器。onmode -p 命令不更新 onconfig 文件。

CPU 虚拟处理器

在单个处理器计算机上，仅分配一个 CPU 虚拟处理器。在多个处理器计算机上，分配的 CPU 虚拟处理器家上用户定义的虚拟处理器的总数目最多达到该计算机上的 CPU 数。

当数据库服务器启动时，CPU 虚拟处理器的数目自动增加到该数据库服务器计算机上 CPU 处理器数的一半，除非启用了 SINGLE_CPU_VP 配置参数。

如果包含 autotune 选项，则数据库服务器根据需要添加 CPU 虚拟处理器来提升性能，最多达到计算机上的 CPU 数目。

当数据库服务器自动地添加 CPU 虚拟处理器时，不更新 CPU 类的 VPCLASS 配置参数的 num 选项的值。

您可配置处理器专用和是否允许迟滞。例如，以下条目创建绑定到 CPU 编号 7、8、9 和 10 的四个 CPU 虚拟处理器，并且不受优先级迟滞的影响：

```
VPCLASS CPU,num=4,aff=(7-10),noage
```

AIO 虚拟处理器

使用 AIO 虚拟处理器的 VPCLASS 配置参数条目来指定 AIO 虚拟处理器的确切数目，或启用数据库服务器来根据需要添加 AIO 虚拟处理器。

当未设置 AIO 虚拟处理器类的 VPCLASS 配置参数条目时，则由 AUTO_AIOVPS 配置参数的设置决定 AIO 虚拟处理器的数目，且限定到 128：

- 如果 AUTO_AIOVPS 设置为 1（开），则初始启动的 AIO 虚拟处理器的数目等于 AIO 块的数目。
- 如果 AUTO_AIOVPS 设置为 0（闭），则启动的 AIO 虚拟处理器的数目等于 6 或 两倍 AIO 块数之中较大的一个。

Java™ 虚拟处理器

如果您使用 Java™ 用户定义的例程或 Java™ 应用程序，则通过添加 JVP 虚拟处理器类的 VPCLASS 配置参数条目来创建至少一个 Java™ 虚拟处理器。如果 JVP 的数目设置为零，或者未设置 JVP 类的 VPCLASS 参数，则您不能运行 Java™ UDR。

相关链接

AUTO_AIOVPS 配置参数 在第67页
DS_MAX_QUERIES 配置参数 在第102页
DS_TOTAL_MEMORY 配置参数 在第106页
NETTYPE 配置参数 在第135页
《*SinoDB 管理员指南*》: 租户虚拟处理器类
The number of configured inline poll threads exceeds the number of CPU virtual processors. 在第799页
Virtual processor limit exceeded. 在第813页
onmode -p: 添加或删除虚拟处理器 在第343页
NETTYPE 配置参数 在第135页
AUTO_AIOVPS 配置参数 在第67页
《*SinoDB 管理员指南*》: 虚拟处理器类
《*SinoDB 管理员指南*》: 《*SinoDB 管理员指南*》: *CPU* 虚拟处理器
《*SinoDB 管理员指南*》: 用户定义的虚拟处理器类
《*SinoDB 管理员指南*》: *Java* 虚拟处理器
《*SinoDB 管理员指南*》: *AIO* 虚拟处理器
NETTYPE 配置参数 在第135页

WSTATS 配置参数

使用 WSTATS 配置参数来指定 onstat -g wst 命令是否显示系统内的线程的等待统计信息。

> 注意： 由于收集统计信息的成本，您应预期产生小的性能影响。不建议启用生产系统的 WSTATS 配置参数。

onconfig.std 值

WSTATS 0

值范围

0 = 禁用等待统计信息

1 = 启用等待统计信息

生效

在编辑 onconfig 文件并重启数据库服务器之后。

当通过运行 onmode -wf 命令来动态地重置 onconfig 文件中的该值时。

当通过运行 onmode -wm 命令来重置内存中的该值时。

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onstat -g wst 命令: 打印线程的等待统计信息 在第539页

sysmaster 数据库

本章节描述 sysmaster 数据库，并提供关于系统监视接口（SMI）的参考信息。

本章节包含以下主题：

- sysmaster 数据库的描述
- 有关如何使用 SMI 表的信息
- SMI 表的描述
- 已记载的 SMI 表的映射

有关 ON-Bar 表的信息，请参阅《*SinoDB®* 备份和还原指南》。

sysmaster 数据库

数据库服务器创建并维护 sysmaster 数据库。类似于《*SinoDB® SQL* 指南*:* 参考》中描述的数据库的系统目录。如同数据库服务器管理的每个数据库的系统目录对数据库中的对象和权限进行跟踪那样，每个数据库的 sysmaster 数据库对有关数据库服务器的信息进行跟踪。

sysmaster 数据库包含系统监视接口（SMI）表。SMI 表提供有关数据库服务器状态的信息。您可以查询这些表以识别处理瓶颈、确定资源使用状况、跟踪会话或数据库服务器活动等等。本章描述这些表，而这些表与普通表稍有不同。

> Warning: 数据库服务器依赖 sysmaster 数据库中的信息。请不要更改 sysmaster 中的任何表或表中的任何数据。更改可能导致不可预知和消弱性的结果。

数据库服务器在初始化磁盘空间时创建 sysmaster 数据库。数据库服务器创建具有未缓冲的日志记录的数据库。您不能删除该数据库或其中的任何表，并且不能关闭日志记录。

UNIX™ 上的用户 informix 或 Windows™ 上的 Informix-Admin 组的成员可以在 sysmaster 数据库中创建 SPL 例程。（还可以在 sysmaster 内的表上创建触发器，但数据库服务器从不执行那些触发器。）

sysmaster 中多个表的联接可能返回不一致的结果，是因为数据库服务器在联接期间不锁定这些表。您可以将 sysmaster 的表与其他数据库中的表进行联接。然而，要将 sysmaster 的表与非日志记录数据库中的表进行联接，首先要使非日志记录数据库成为当前数据库。

buildsmi 脚本

当首次启动数据库服务器时，它会运行 etc 目录下名为 buildsmi 的脚本。该脚本构建数据库和支持 SMI 的表。数据库服务器需要大约 1750 个逻辑日志空间可用页来构建 sysmaster 数据库。

如果接收到指示您去运行 buildsmi 脚本的错误消息，那么可能在数据库服务器构建 SMI 数据库、表和视图时发生了问题。当使用 buildsmi 时，将删除现有 sysmaster 数据库然后重新创建。

在确保数据库创建期间没有到 sysmaster 数据库的连接后，该脚本才可运行，而且在 UNIX™ 上必须以 informix 的用户身份运行，或者在 Windows™ 上必须以 Informix-Admin 组的成员身份运行。例如，如果在

buildsmi 脚本开始时有一个调度程序的任务正在运行，当该调度程序试图访问 sysmaster 的任何表时，则该脚本会失败。

在 buildsmi 脚本运行时发生的错误会写入 UNIX™ 的 /tmp/buildsmi.out 文件，或者 Windows™ 的 %INFORMIXDIR%\etc\buildsmi_out.***%INFORMIXSERVER%*** 文件，其中 ***%INFORMIXSERVER%*** 是 SinoDB® 实例的名称。

bldutil.sh 脚本

当数据库服务器首次初始化时，它将在 UNIX™ 上运行名为 bldutil.sh 的脚本，或者在 Windows™ 上运行名为 bldutil.bat 的脚本。该脚本建构 sysutils 数据库。如果建构失败，则数据库服务器在 tmp 目录中创建输出文件。该输出文件在 UNIX™ 上为 bldutil.process_id，在 Windows™ 上为 bldutil.out。该输出文件中的消息反映该脚本执行过程中发生的错误。

系统监视接口

本节描述 SMI 表，以及如何使用这些表来监视数据库服务器操作。

了解 SMI 表

SMI（系统监视接口）由数据库服务器自动维护的表和伪表组成。当 SMI 表作为表向用户显示时，它们不像普通的表那样记录在磁盘上。而是，数据库服务器视需要在内存中根据瞬时共享内存中的信息构建表。当查询 SMI 表时，数据库服务器从这些共享内存结构中读取信息。由于数据库服务器持续地更新共享内存的数据，所以 SMI 所提供的信息允许您检查数据库服务器的当前状态。

SMI 表提供关于以下主题的信息：

- 审计
- 检查点
- 块 I/O
- 块
- 数据库日志记录状态
- 数据库空间
- 磁盘使用
- 环境变量
- 扩展数据块
- 锁
- 网络
- SQL 语句高速缓存统计信息
- SQL 语句
- 系统概要文件
- 表
- 用户概要文件
- 虚拟处理器 CPU 使用

当用户访问并修改数据库服务器所管理的数据库时，SMI 表中的数据动态地更改。

访问 SMI 表

任何用户可以使用 SQL SELECT 语句来查询 SMI 表，但标准用户不能执行 SELECT 以外的语句。尝试执行其它语句的用户会导致权限错误。管理员可执行 SELECT 以外的 SQL 语句，但这类语句的结果是不可预料的。

> 提示： 要获得可预料的结果，请查询与每个表关联的视图，而不是直接查询表。

如果直接查询 systabpaghdrs 表，您必须为 pg_partnum 参数指定一个适当的值。该值是 pg_partnum > 1048576。然而，如果查询与 systabpaghdrs 表关联的视图，则不用指定 pg_partnum 参数的值。

SinoDB® 包括 sysadtinfo 和 sysaudit 表。仅 UNIX™ 上的用户 informix 或 Windows™ 上的 Informix-Admin 组的成员可以查询 sysadtinfo 和 sysaudit 表。

不可在 sysmaster 数据库中的任何表上使用 dbschema 或 dbexport 实用程序。如果使用了，数据库服务器将生成以下错误消息：

```
Database has pseudo tables - can't build schema
```

SELECT 语句
可以对普通表使用 SELECT 的地方，就可以对 SMI 表使用 SELECT 语句。

例如，您可以从 DB-Access、在 SPL 例程中、以SinoDB® ESQL/C 等等对普通表使用 SELECT 语句。

> 限制： 当查询 SMI 表时，无法有意义地引用 rowid。使用 rowid 的 SELECT 语句不会返回错误，但是结果是不可预料的。

包括表之间联接、输出排序等等的所有标准 SQL 语法，都可用于 SMI 表。例如，如果要联接 SMI 表和非 SMI 表，则使用以下标准语法来指定 SMI 表：

```
sysmaster[@dbservername]:[owner.]tablename
```

触发器和事件警报

基于对 SMI 表的更改的触发器从不运行。尽管可以在 SMI 表上定义触发器，但仅当表上发生 INSERT、UPDATE 或 DELETE 语句时才会激活触发器。对 SMI 数据的更新发生在数据库服务器中，不需使用 SQL，因此即使由 SELECT 语句返回的数据指示应该激活触发器， 也不会激活 SMI 表上的触发器。

要创建事件警报，请在预定时间间隔查询特定条件，并在警报的必需条件满足时执行 SPL 例程。

SPL and SMI 表

可以在 SPL 例程中访问 SMI 表。当引用 SMI 表时，请使用与引用标准表相同的语法。

锁定和 SMI 表

SMI 表中的信息基于数据库服务器活动而变更。然而，数据库服务器不使用 SQL 语句更新信息。当以锁定对象的隔离级别使用 SMI 表时，它会阻止其他用户访问该对象，但不会阻止数据变更。在这种情况下，所有 SMI 表都具有永久的 Dirty Read 隔离级别。

系统监视接口表

sysmaster 数据库包含许多用于监视您系统的表。

> 提示： 对于每个系统监视接口（SMI）表，都有一相同名称的相应视图。为了获得最好的结果，应查询与表相关联的视图，而不是直接查询表。

sysmaster 数据库中的许多其他表是系统监视接口的一部份但未加记载。其模式和列内容随版本而变更。如今 flags_text 表包含更多的行。要查看新的行，首先删除并重新创建 sysmaster 数据库。

下表罗列了 SMI 表。

表 73: SMI 表

表	描述	参考
sysadtinfo	审计配置信息	*sysadtinfo* 在第204页
sysaudit	审计事件掩码	*sysadtinfo* 在第204页
syscheckpoint	检查点信息	*syscheckpoint* 在第205页
syschkio	块 I/O 统计信息	*syschkio* 在第205页

表	描述	参考
syschunks	块信息	*syschunks* 在第206页
syscluster	高可用性集群信息	*syscluster* 在第208页
syscmsmsla	连接管理器信息	*syscmsmsla* 在第209页
syscmsmtab	连接管理器信息	*syscmsmtab* 在第210页
syscmsmunit	连接管理器配置文件中每个连接管理器单元的信息	*syscmsmunit* 在第210页
syscompdicts_full	压缩字典信息	*syscompdicts_full* 在第210页
sysconfig	配置信息	*sysconfig* 在第211页
sysdatabases	数据库信息	*sysdatabases* 在第212页
sysdbslocale	语言环境信息	*sysdbslocale* 在第212页
sysdbspaces	数据库空间信息	*sysextents* 在第215页
sysdri	数据复制信息	*sysdri* 在第214页
sysdual	是单行表	*sysdual* 在第214页
sysenv	服务器启动环境	*sysenv* 在第214页
sysenvses	会话级别环境变量	*sysenvses* 在第214页
sysextents	扩展数据块分配信息	*sysextents* 在第215页
sysextspaces	外部空间信息	*sysextspaces* 在第215页
sysha_lagtime	辅助服务器延滞时间信息	*sysha_lagtime 表* 在第216页
sysha_type	关于已连接服务器的信息	*sysha_type* 在第217页
sysha_workload	辅助服务器工作负载统计信息	*sysha_workload* 在第217页
sysipl	索引页日志记录信息	*sysipl* 在第218页
syslocks	活动锁信息	*syslocks* 在第218页
syslogs	逻辑日志文件信息	*syslogs* 在第219页
syslogfil	系统日志文件信息	*syslogfil 表* 在第220页
sysmgminfo	内存分配管理器/并行数据查询信息	*sysmgminfo* 在第221页
sysnetclienttype	客户端类型网络活动	*sysnetclienttype* 在第221页
sysnetglobal	全居网络信息	*sysnetglobal* 在第222页
sysnetworkio	网络 I/O	*sysnetworkio 表* 在第222页
sysonlinelog	联机日志信息	*sysonlinelog* 在第223页
sysprofile	系统概要文件信息	*sysprofile* 在第223页
sysproxyagents	关于所有代理线程的信息	*sysproxyagents* 在第225页
sysproxydistributors	代理分发器信息	*sysproxydistributors* 在第225页
sysproxysessions	关于使用可更新辅助服务器的会话的信息	*sysproxysessions 表* 在第225页

表	描述	参考
sysproxytxnops	关于通过每个代理分发器运行的事务的信息	*sysproxytxnops 表* 在第226页
sysproxytxns	关于通过每个代理分发器运行的当前事务的所有信息	*sysproxytxns 表* 在第226页
sysptprof	表信息	*sysptprof 表* 在第226页
sysrepevtreg	发布到连接管理器和 SinoDB® 开放管理工具 (OAT) 独事件	*sysrepevtreg 表* 在第227页
sysrepstats	发布到连接管理器和 OAT 的事件	*sysrepstats 表* 在第227页
sysrsslog	RS 辅助服务器信息	*sysrsslog* 在第230页
sysscblst	用户的内存	*sysscblst* 在第231页
syssesprof	各种用户操作计数	*syssesprof* 在第231页
syssesappinfo	分布式关系数据库结构（DRDA®）客户端会话信息	*syssesappinfo* 在第231页
syssessions	每个已连接用户的描述	*syssessions* 在第232页
syssmx	SMX（服务器多路复用器组）连接信息	*syssmx* 在第233页
syssmxses	SMX（服务器多路复用器组）会话信息	*syssmxses* 在第234页
syssqexplain	SET EXPLAIN 语句启用的 SQL 语句信息	*syssqexplain 表* 在第234页
syssqltrace	SQL 语句信息	*syssqltrace* 在第235页
syssqltrace_hvar	SQL 语句跟踪主机变量信息	*syssqltrace_hvar* 在第237页
syssqltrace_info	SQL 概要文件跟踪系统信息	*syssqltrace_info* 在第237页
syssqltrace_iter	SQL 语句迭代器	*syssqltrace_iter* 在第237页
syssrcrss	RS 辅助服务器统计信息	*syssrcrss* 在第238页
syssrcsds	SD 辅助服务器统计信息	*syssrcsds* 在第238页
systabnames	表空间 tblspace 的数据库、所有者和表名称	*systabnames* 在第239页
systabpaghdrs	页头Page headers	None
systhreads	等待统计信息	*systhreads* 在第239页
systrgrss	RS 辅助服务器统计信息	*systrgrss* 在第240页
systrgsds	SD 辅助服务器统计信息	*systrgsds* 在第240页
sysvpprof	每个虚拟处理器所使用的用户和系统 CPU	*sysvpprof* 在第240页

sysutils 表

ON-Bar 使用 sysutils 数据库中的下表。有关更多信息，请参阅《*SinoDB®* 备份和还原指南》。

bar_action

列出冷恢复期间以外，对对象尝试的所有备份与恢复操作。使用该表中的信息跟踪备份和恢复历史的记录。

bar_instance

为每个成功的备份向该表写入一条记录。 ON-Bar 可能之后将该信息用于恢复操作。

bar_object

描述每个备份对象。该表提供每个至少做过一次备份尝试的数据库服务器的所有存储空间和逻辑日志的列表。

bar_server

列出安装中的数据库服务器。该表用于确保备份对象在恢复过程中返回到正确位置。

sysadtinfo

sysadtinfo 表包含有关数据库服务器审计配置的信息。有关更多信息，请参阅《*SinoDB*® 安全指南》。要从 sysadtinfo 表检索信息，必须是 UNIX™ 上的用户 informix 或 root，或者是 Windows™ 上的 Informix-Admin 组的成员。

列	类型	描述
adtmode	integer	控制审计的级别
adterr	integer	指定数据库服务器在写入审计记录时遇到错误，其行为方式
adtsize	integer	审计文件大小的最大值
adtpath	char(256)	审计文件写入的目录
adtfile	integer	审计文件数

sysaudit

对于每个已定义的审计掩码（也就是，对于每个 *username*），sysaudit 表包含表示生成审计记录的数据库事件的标志。success 和 failure 列表示组成审计掩码的位掩码。如果一个位在 success 和 failure 列都设置了，则相应的事件无论是否成功都会生成一条审计记录。

要从 sysaudit 表中检索信息，必须是 UNIX™ 上的用户 informix 或 root，或者是 Windows™ 上 Informix-Admin 组的成员。

使用 onaudit 实用程序来列出或修改审计掩码。有关 onaudit 和审计的信息，请参阅《*SinoDB*® 安全指南》。

列	类型	描述
username	char(32)	掩码名称
succ1	integer	成功审计掩码的位掩码
succ2	integer	成功审计掩码的位掩码
succ3	integer	成功审计掩码的位掩码
succ4	integer	成功审计掩码的位掩码
succ5	integer	成功审计掩码的位掩码
fail1	integer	失败审计掩码的位掩码
fail2	integer	失败审计掩码的位掩码
fail3	integer	失败审计掩码的位掩码
fail4	integer	失败审计掩码的位掩码

列	类型	描述
fail5	integer	失败审计掩码的位掩码

syschkio

syschkio 系统监视接口表提供数据库服务器所管理的各个块的 I/O 统计信息。

列	类型	描述
chunknum	smallint	块编号
reads	integer	物理读取数
pagesread	integer	读取的页数
writes	integer	物理写入数
pageswritten	integer	写入的页数
mreads	integer	物理读取（镜像）数
mpagesread	integer	读取（镜像）的页数
mwrites	integer	物理写入（镜像）数
mpageswritten	integer	写入（镜像）的页数

syscheckpoint

syscheckpoint 表提供关于检查点的信息和其统计信息。

列	类型	描述
interval	integer	服务器启动以来的检查点数
type	char(12)	Hard 或 Interval
caller	char(10)	检查点的调用者
clock_time	integer	检查点发生的时间
crit_time	float	等待释放临界区所用的时间
flush_time	float	将页清空到磁盘所用的时间
cp_time	float	检查点暂挂到检查点完成的持续时间
n_dirty_buffs	integer	脏缓冲区数
plogs_per_sec	integer	一秒内处理的物理日志页数
llogs_per_sec	integer	一秒内处理的逻辑日志页数
dskflush_per_sec	integer	一秒内清空的缓冲池页数
ckpt_logid	integer	在检查点的逻辑日志的唯一标识
ckpt_logpos	integer	在检查点的逻辑日志的位置
physused	integer	物理日志中已用的页数
logused	integer	逻辑日志中已用的页数
n_crit_waits	integer	必须等待进入临界区的用户数
tot_crit_wait	float	在检查点临界区块等待的所有用户所用的等待持续时间

列	类型	描述
longest_crit_wait	float	最长的临界区等待
block_time	float	检查点阻塞系统的持续时间

syschunks

syschunks 表包含数据库服务器所管理的每个块的描述。

flags 和 mflags 列中，每个位的位置表示一个单独标志。因此，如果使用 HEX 函数来返回这些值，那么读取 flags 和 mflags 列中的值会容易些。

表 74: The syschunks 表

列	类型	描述
chknum	smallint	块编号
dbsnum	smallint	数据库空间编号
nxchknum	smallint	数据库空间中下一个块的编号
chksize	integer	块中的页数（以系统默认页大小为单位）
offset	integer	在设备或路径中块的页偏移量
pagesize	integer	页大小（以字节计）
nfree	integer	块中空闲页数 空闲空间的量取决于空间的类型： • 数据库空间 = 空闲页数乘以系统默认页大小 2 KB 或 4 KB。 • BLOB 空间 = 空闲页数乘以 BLOB 页大小。 • 智能大对象空间 = 空闲页数乘以智能大对象页大小（其与系统默认页大小一样大）。
is_offline	integer	1 = 块处于脱机模式 0 = 块处于联机模式
is_recovering	integer	1 = 块正在被恢复 0 = 块没有在被恢复
is_blobchunk	integer	1 = 块在 BLOB 空间中 0 = 块不在 BLOB 空间中
is_sbchunk	integer	1 = 块在智能大对象空间 0 = 块不在智能大对象空间
is_inconsistent	integer	1 = 块正在进行逻辑恢复 0 = 块没有在进行逻辑恢复
is_extendable	integer	1 = 块是可扩展的 0 = 块是不可扩展的
flags	smallint	标志有以下数值、十六进制值和含义： • 16 (0x0010) = 块是镜像块

列	类型	描述
		• 32 (0x0020) = 块处于脱机模式 • 64 (0x0040) = 块处于联机模式 • 128 (0x0080) = 块处于恢复模式 • 256 (0x0100) = 块已镜像 • 512 (0x0200) = 块是 BLOB 空间的一部份 • 1024 (0x0400) = 块正在被删除 • 4096 (0x1000) = 块是不一致的 • 8192 (0x2000) = 块是可扩展的 • 16384 (0x4000) = 块在前滚过程中被添加 • 32768 (0x8000) = 块被重新命名 • 65536 (0x10000) = 块使用大块页头 • 131072 (0x20000) = 块有一个表空间 tblspace 扩展块 • 262144 (0x40000) = 从块初始化以来，没有检查点被完成（主要供内部使用）
fname	char(256)	块的文件或设备的路径名
mdsize	integer	属于智能大对象空间的块的元数据区域的大小（以页为单位）。 -1 = 块不属于智能大对象空间。
mfname	char(256)	已镜像块（如果有）的文件或设备的路径名。
moffset	integer	已镜像块的页偏移量
mis_offline	integer	1 = 镜像处于脱机状态 0 = 镜像处于联机状态
mis_recovering	integer	1 = 镜像正在被恢复 0 = 镜像没有被恢复
mflags	smallint	已镜像块标志；其值与含义与 flags 列相同。
udfree	integer	属于智能大对象空间的块的用户数据区域中的空闲页（以页计）。 -1 = 块不属于智能大对象空间。
udsize	integer	属于智能大对象空间的块的用户数据区域的大小（以页计）。 -1 = 块不属于智能大对象空间。

sysckptinfo

sysckptinfo 系统监视接口表提供有关前 20 个检查点的历史信息。

列	类型	描述
ckpt_status	int	0x0011 = 由于物理日志耗尽了资源，所以检查点被阻止。 0x0021 = 由于逻辑日志耗尽了资源，所以检查点被阻止。 0x0041 = 由于事务太长，所以检查点被阻止。 0x1000 = 物理日志太小。

列	类型	描述
		0x2000 = 逻辑日志空间太小。 0x4000 = RTO 的物理日志太小。
plogs_per_S	int	物理日志记录活动的平均率。
llogs_per_S	int	逻辑日志记录活动的平均率。
dskF_per_S	int	页清空到磁盘的平均率。
longest_dskF	int	检查点处理过程中缓冲池清空到磁盘的最长持续时间。
dirty_pgs_S	int	页被修改的平均率。
sug_plog_sz	int	建议的物理日志大小。
sug_llog_sz	int	建议的逻辑日志空间大小。
ras_plog_sp	int	快速恢复可以恢复物理日志的速率。
ras_llog_sp	int	快速恢复可以重放逻辑日志的速率。
boottime	int	服务器启动共享内存和打开块所需的时间。
auto-ckpts	int	1 = 开，0 = 关。
auto_lru	int	1 = 开，0 = 关。
cur_intvl	int	当前检查点时间间隔。
auto_aiovp	int	1 = 开，0 = 关。

相关链接

onstat -g ckp 命令: 打印检查点历史记录与配置建议 在第434页

syscluster

syscluster 系统目录表存储有关高可用性集群中服务器的信息。syscluster 表有以下列。

表 75: syscluster 表信息

该表提供了 syscluster 表列的名称、类型和描述的信息。

列	类型	描述
name	CHAR(128)	主服务器名称。
role	CHAR(1)	标示服务器是主服务器还是辅助服务器的代码
syncmode	CHAR(8)	主服务器和辅助服务器之间的同步模式：sync 或 async。
nodetype	CHAR(8)	服务器类型：HDR、RSS 或 SDS。
supports_updates	CHAR(1)	标示客户端应用程序是否可在辅助服务器上执行更新、插入和删除操作（由 UPDATABLE_SECONDARY 配置参数所指定）。
server_status	CHAR(32)	标示辅助服务器的状态。
connection_status	CHAR(32)	标示辅助服务器的连接状态。
delayed_apply	INTEGER	标示辅助服务器在应用日志之前是否需要等待一定时间量（由 DELAY_APPLY 配置参数指定）。

列	类型	描述
stop_apply	CHAR(24)	标示辅助服务器是否停止应用从主服务器接收到的日志文件（由 STOP_APPLY 配置参数指定）。
logid_sent	INTEGER	标示从主服务器发送到辅助服务器的最新日志页的日志 ID。
logpage_sent	INTEGER	标示从主服务器发送到辅助服务器的最新日志页的页号。
logid_acked	INTEGER	标示辅助服务器确认的最新日志页的日志 ID。
logpage_acked	INTEGER	标是辅助服务器确认的最新日志页的页号。
ack_time	DATETIME YEAR TO SECOND	标示最新确认日志的日期和时间。
sdscycle	INTEGER	标示主服务器已进入的循环编号。供 SinoDB® 内部用于支持监视主服务器和辅助服务器的协调。
sdscycle_acked	INTEGER	标示共享磁盘辅助服务器确认的循环编号。供 SinoDB® 内部用于支持监视主服务器和辅助服务器的协调。

syscmsm

syscmsm 表是 syscmsmtab 和 syscmsmsla 表的视图。其包含连接管理器服务等级协议（SLA）信息。该表每 5 秒更新一次。

表 76: syscmsm 表信息

该表提供了 syscmsm 表列的名称、类型和描述的信息。

列	类型	描述
sid	integer	连接管理器会话 ID
name	char(128)	连接管理器名称
host	char(256)	主机名
unit	char(128)	单元名
type	char(128)	单元类型
servers	char(1024)	单元服务器
foc	char(128)	故障转移配置（FOC）
flag	integer	仲裁标志。值 1 表示连接管理器仲裁是激活的。值 0 表示仲裁未被激活。
sla_name	char(128)	SLA 名称
sla_define	char(128)	SLA 定义
connections	integer	通过连接管理器建立的连接的数目

syscmsmsla

syscmsmsla 表包含连接管理器服务等级协议（SLA）信息。该表每 5 秒更新一次。

表 77: syscmsmsla 表信息

该表提供了 syscmsmsla 表列的名称、类型和描述的信息。

列	类型	描述
address	int8	CMSLA 内部地址
sid	integer	连接管理器会话 ID
sla_name	char(128)	SLA 名称
sla_define	char(128)	SLA 定义
connections	integer	通过连接管理器建立的连接数

syscmsmtab

syscmsmtab 表包含连接管理器信息。

表 78: syscmsmtab 表信息

该表提供了 syscmsmtab 表列的名称、类型和描述的信息。

列	类型	说明
address	int8	连接管理器内部地址
sid	integer	连接管理器会话 ID
name	char(128)	连接管理器名称
host	char(256)	主机名
flag	integer	仲裁标志。值 1 表示连接管理器仲裁是激活的。值 0 表示仲裁未被激活。

syscmsmunit

syscmsmunit 表包含连接管理器配置文件中每个连接管理器单元的信息

表 79: syscmsmunit 表信息

该表提供了 syscmsmunit 表列的名称、类型和描述的信息。

列	类型	描述
address	int8	连接管理器内部地址
sid	integer	连接管理器会话 ID
unit	char(128)	单元名
type	char(128)	单元类型
servers	char(1024)	单元服务器
foc	char(128)	故障转移配置（FOC）
flag	integer	仲裁标志。值 1 表示连接管理器仲裁是激活的。值 0 表示仲裁未被激活。

syscompdicts_full

syscompdicts_full 表和 syscompdicts 视图提供所有压缩字典的信息。该表与该视图的唯一区别是，为了安全性目的，视图不包含 dict_dictionary 列。

仅用户 informix 可以从 syscompdicts_full 表检索信息。syscompdicts 视图不限于用户 informix。

下表显示 syscompdicts_full 表和 syscompdicts 视图为每个压缩字典提供的信息。

表 80: 压缩字典信息

列	类型	描述
dict_partnum	integer	压缩字典所适用的分区号
dict_code_version	integer	创建压缩字典的代码版本 1 是第一版。
dict_dbsnum	integer	字典所在数据库空间的编号
dict_create_timestamp	integer	字典创建的时间
dict_create_loguniqid	integer	字典创建时所激活的逻辑日志的唯一 ID
dict_create_logpos	integer	字典创建时在逻辑日志中的位置
dict_drop_timestamp	integer	字典删除的时间
dict_drop_loguniqid	integer	字典删除时所创建的逻辑日志的唯一 ID
dict_drop_logpos	integer	字典删除时在逻辑日志中的位置
dict_dictionary	byte	压缩字典二进制对象 此列未包含在 syscompdicts 视图中

syscompdicts 信息样本

syscompdicts 视图中的一行显示的列包含如下信息：

```
dict_partnum         1048939
dict_code_version    1
dict_dbsnum          1
dict_create_times+   1231357656
dict_create_logun+   11
dict_create_logpos   1695768
dict_drop_timesta+   0
dict_drop_loguniq+   0
dict_drop_logpos     0
```

可以使用 UNLOAD 语句将压缩字典卸载到压缩字典文件，如下所示：

```
UNLOAD TO 'compression_dictionary_file'
   SELECT * FROM sysmaster:syscompdicts_full;
```

sysconfig

sysconfig 表描述配置参数的有效值、原始值和默认值。有关 ONCONFIG 文件和配置参数的更多信息，请参阅数据库配置参数 在第41页。

列	类型	描述
cf_id	integer	唯一数字标识符
cf_name	char(128)	配置参数名称
cf_flags	integer	保留，供将来使用
cf_original	char(256)	启动时 ONCONFIG 文件中的值

列	类型	描述
cf_effective	char(256)	当前使用的值
cf_default	char(256)	如果 ONCONFIG 文件中没有指定值，则由数据库服务器提供的值

sysdatabases

sysdatabases 视图描述数据库服务器所管理的每个数据库。

表 81: sysdatabases 视图信息

该表提供 sysdatabases 视图列的名称、类型和描述的信息。

列	类型	描述	
name	char(128)	数据库名称	
partnum	integer	数据库的 systables 表的分区编号（表空间标识）	
owner	char(32)	数据库创建者的用户 ID	
created	date	创建日期	
is_logging	integer	如果日志记录是活动的，则为 1；如果不是，则为 0。	
is_buff_log	integer	如果是缓冲日志记录，则为 1；如果不是，则为 0。	
is_ansi	integer	如果符合 ANSI/ISO，则为 1；如果不是，则为 0。	
is_nls	integer	如果启用 GLS，则为 1；如果没有启用，则为 0。	
is_case_insens	integer	如果 NCHAR 和 NVARCHAR 列不区分大小写，则为 1； 如果不是，则为 0。	
flags	smallint	日志记录标志（十六进制值）	
		0	无日志记录
		1	未缓冲的日志记录
		2	缓冲的日志记录
		4	符合 ANSI/ISO 的数据库
		8	只读数据库
		10	GLS 数据库
		20	忽略对 syscdr 数据库日志记录模式的检查
		100	已将状态更改为缓冲日志记录
		200	已将状态更改为未缓冲日志记录
		400	已将状态更改为符合 ANSI/ISO
		800	关闭数据库日志记录
		1000	启用长 ID 支持

sysdbslocale

sysdbslocale 表列出数据库服务器所管理的每个数据库的语言环境。

表 82: sysdbslocale 表信息

该表提供 sysdbslocale 表列的名称、类型和描述的信息。

列	类型	描述
dbs_dbsname	char(128)	数据库名称
dbs_collate	char(32)	数据库的语言环境

sysdbspaces

sysdbspaces 表包含了数据库服务器所管理的每个数据库空间的描述。

flags 列中，每个位的位置代表一个单独标志。因此，如果值是使用 HEX 函数返回的，那么读取 flags 列中的值可能会更容易。

表 83: sysdbspaces 表信息

该表提供了 sysdbspaces 表列的名称、类型和描述的信息。

列	类型	描述		
dbsnum	smallint	数据库空间编号		
name	char(128)	数据库空间名称		
owner	char(32)	数据库空间所有者的用户 ID		
fchunk	smallint	数据库空间中第一个块的编号		
nchunks	smallint	数据库空间中块的数量		
create_size	decimal	使用存储池为此空间创建的块的最小容量。		
extend_size	decimal	手动或自动扩展此存储空间中的块的最小容量。		
pagesize	integer	页大小		
is_mirrored	integer	如果数据库空间已镜像，则为 1；如果不是，则为 0。		
is_blobspace	integer	如果数据库空间是 BLOB 空间，则为 1；如果不是，则为 0。		
is_sbspace	integer	如果数据库空间是智能大对象空间，则为 1；如果不是，则为 0。		
is_temp	integer	如果数据库空间是临时数据库空间，则为 1；如果不是，则为 0。		
flags	smallint	标志	十六进制值	含义
		1	0x0001	数据库空间没有镜像
		2	0x0002	数据库空间使用镜像
		4	0x0004	禁用数据库空间镜像
		8	0x0008	数据库空间最近镜像过
		16	0x0010	空间是 BLOB 空间
		32	0x0020	BLOB 空间在可移动介质上
		128	0x0080	BLOB 空间已删除
		512	0x0200	空间正在恢复
		1024	0x0400	空间已被物理恢复

列	类型	描述		
		2048	0x0800	空间在逻辑恢复中
		32768	0x8000	空间是智能大对象空间

sysdri

sysdri 表提供有关数据库服务器的高可用性数据复制的信息。

列	类型	描述
type	char(50)	高可用性数据复制类型值： • 主 • 辅助 • 标准 • 未初始化
state	char(50)	高可用性数据复制的状态值： • 关 • 开 • 正在连接 • 失败 • 只读
name	char(128)	高可用性数据复制对的另一数据库服务器的名称
intvl	integer	高可用性数据复制间隔
timeout	integer	该数据库服务器的高可用性数据复制超时值
lostfound	char(256)	丢失和找到文件的路径名

sysdual

sysdual 表仅返回一行一列。

列	类型	描述
dummy	char(1)	返回 "X" 的虚拟列

sysenv

sysenv 表显示数据库服务器的启动环境设置。

列	类型	描述
env_id	integer	标识变量编号
env_name	char(128)	环境变量名称
env_value	char(512)	环境变量值

sysenvses

sysenvses 表显示会话级别的环境变量。

列	类型	描述
envses_sid	integer	会话 ID

列	类型	描述
envses_id	integer	标识变量编号
envses_name	char(128)	会话环境变量名称
envses_value	char(512)	会话环境变量值

sysextents

sysextents 表提供有关扩展数据块的信息。

列	类型	描述
dbsname	char(128)	数据库名
tabname	char(128)	表名
chunk	integer	块编号
offset	integer	块中开始扩展数据块的页数
size	integer	扩展数据块的大小，以页计

sysextspaces

sysextspaces 表提供有关外部空间的信息。id 列和 name 列的索引只允许唯一值。

列	类型	描述
id	integer	外部空间 ID
name	char(128)	外部空间名
owner	char(32)	外部空间所有者
flags	integer	外部空间标志（保留供将来使用）
refcnt	integer	外部空间引用计数
locsize	integer	外部空间位置的大小，以字节计
location	char (256)	外部空间位置

sysfeatures

sysfeatures 视图提供有关 SinoDB® 数据库服务器实例的各种功能的一般信息。sysfeatures 视图是从永久存储在磁盘上名为 syslicenseinfo 的内部表创建的。当数据库服务器实例初始化时，该表会预分配可跟踪 260 周数据大小的固定容量。这些数据每五年更新一次。

每隔 15 分钟抽样度量标准，并且只存储周中的最高值。表中的每行包含的数据仅代表特定周。

列	类型	描述
week	smallint	记录信息的周数
year	smallint	记录信息的年份
version	char(12)	SinoDB® 服务器版本
max_cpu_vps	smallint	CPU 虚拟处理器的最大数量
max_vps	smallint	虚拟处理器的最大数量
max_conns	integer	独立或高可用性集群主服务器实例上并发物理连接的最大数量

列	类型	描述
max_sec_conns	integer	HDR 或 RS 辅助服务器实例上并发物理连接的最大数量
max_sds_clones	smallint	连接到主服务器的最大 SD 辅助服务器实例数
max_rss_clones	smallint	连接到主服务器的最大 RS 辅助服务器实例数
total_size	integer	所有块中分配的最大磁盘空间（以 MB 计）
total_size_used	integer	所有块中已使用的最大磁盘空间（以 MB 计）
max_memory	integer	所有段中分配的最大内存量（以 MB 计）
max_memory _used	integer	所有段中已使用的最大内存量（以 MB 计）
is_primary	integer	标示在特定周中服务器是否为主服务器；1 = 是，0 = 否。
is_secondary	integer	标示在特定周中服务器是否为 HDR 辅助服务器；1 = 是，0 = 否。
is_sds	integer	标示在特定周中服务器是否为 SD 辅助服务器；1 = 是，0 = 否（未实施；始终是 0）。
is_rss	integer	标示在特定周中服务器是否为 RS 辅助服务器；1 = 是，0 = 否。
is_er	integer	标示在特定周中服务器是否为 Enterprise Replication 服务器；1 = 是，0 = 否。
is_pdq	integer	标示在特定周中服务器是否使用了 PDQ 功能；1 = 是，0 = 否。

sysha_lagtime 表

sysha_lagtime 表提供在任一辅助节点上应用日志记录所用时间量的历史记录

sysha_lagtime 表包含对特定辅助服务器执行的最近 20 次采样的历史记录。

表 84: sysha_lagtime 表信息

该表提供了 sysha_lagtime 表列的名称、类型和描述的信息。

列	类型	描述
lt_secondary	CHAR(128)	辅助服务器名
lt_time_last_update	INTEGER	最近更新日志记录的时间
lt_lagtime_1	FLOAT	应用最近 5 秒间隔的日志记录所需的时间量
lt_lagtime_2	FLOAT	应用前个 5 秒间隔的日志记录所需的时间量
lt_lagtime_3	FLOAT	应用前 2 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_4	FLOAT	应用前 3 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_5	FLOAT	应用前 4 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_6	FLOAT	应用前 5 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_7	FLOAT	应用前 6 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_8	FLOAT	应用前 7 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_9	FLOAT	应用前 8 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_10	FLOAT	应用前 9 个 5 秒间隔的日志记录所需的时间量

列	类型	描述
lt_lagtime_11	FLOAT	应用前 10 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_12	FLOAT	应用前 11 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_13	FLOAT	应用前 12 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_14	FLOAT	应用前 13 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_15	FLOAT	应用前 14 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_16	FLOAT	应用前 15 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_17	FLOAT	应用前 16 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_18	FLOAT	应用前 17 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_19	FLOAT	应用前 18 个 5 秒间隔的日志记录所需的时间量
lt_lagtime_20	FLOAT	应用前 19 个 5 秒间隔的日志记录所需的时间量

sysha_type

sysha_type 表是一个单行表，用于描述连接的服务器的类型。

表 85: sysha_type 表信息

该表提供了 sysha_type 表列的名称、类型和描述的信息。

列	类型	描述
ha_type	integer	服务器类型（参见下表）
ha_primary	char(128)	服务器名（参见下表）

表 86: sysha_type 表中值的描述

该表描述了 sysha_type 表中的值。

ha_type 的值	***ha_primary*** 的值	描述
0	NULL	不是高可用性环境的组成部份
1	*<primary server name>*	主服务器
2	*<primary server name>*	HDR 辅助服务器
3	*<primary server name>*	SD 辅助服务器
4	*<primary server name>*	RS 辅助服务器

sysha_workload

sysha_workload 表包含每个辅助服务器上的工作负载统计信息。

表 87: sysha_workload 表信息

该表提供了 sysha_workload 表列的名称、类型和描述的信息。

列	类型	描述
wl_secondary	char(128)	辅助服务器名
wl_time_last_update	integer	最近更新工作负载的时间

列	类型	描述
wl_type	char(12)	此行包含了就绪队列大小、用户 CPU 时间和系统 CPU 时间
wl_workload_1	float	最近的工作负载活动
wl_workload_2	float	前个工作负载活动
wl_workload_3	float	前 2 个工作负载活动
wl_workload_4	float	前 3 个工作负载活动
wl_workload_5	float	前 4 个工作负载活动
wl_workload_6	float	前 5 个工作负载活动
wl_workload_7	float	前 6 个工作负载活动
wl_workload_8	float	前 7 个工作负载活动
wl_workload_9	float	前 8 个工作负载活动
wl_workload_10	float	前 9 个工作负载活动
wl_workload_11	float	前 10 个工作负载活动
wl_workload_12	float	前 11 个工作负载活动
wl_workload_13	float	前 12 个工作负载活动
wl_workload_14	float	前 13 个工作负载活动
wl_workload_15	float	前 14 个工作负载活动
wl_workload_16	float	前 15 个工作负载活动
wl_workload_17	float	前 16 个工作负载活动
wl_workload_18	float	前 17 个工作负载活动
wl_workload_19	float	前 18 个工作负载活动
wl_workload_20	float	前 19 个工作负载活动

sysipl

sysipl 表提供有关主服务器上索引页日志记录状态的信息。

表 88: sysipl 表信息

该表提供了 sysipl 表列的名称、类型和描述的信息。

列	类型	描述
ipl_status	integer	索引页日志记录状态
ipl_time	integer	启用索引页日志记录的时间

syslocks

syslocks 表提供有关数据库服务器中所有当前活动锁的信息。

表 89: syslocks 表信息

该表提供了 syslocks 表列的名称、类型和描述的信息。

列	类型	描述	
dbsname	char(128)	数据库名	
tabname	char(128)	表名	
rowidlk	integer	实际 rowid（如果它是索引键锁）	
keynum	smallint	索引键锁的键编号	
type	char(4)	锁类型	
		B	字节锁
		IS	意向共享锁
		S	共享锁
		XS	可重复读取器持有的共享键值
		U	更新锁
		IX	意向互斥锁
		SIX	共享意向互斥锁
		X	互斥锁
		XR	可重复读取器持有的互斥键值
owner	integer	锁所有者的会话 ID	
waiter	integer	等待锁的用户的会话 ID。如果有一个以上用户在等待，那么只有第一个会话 ID 出现。	

syslogs

syslogs 表提供有关逻辑日志文件中使用的空间的信息。flags 列中，每个位的位置代表一个单独标志。例如，对于日志文件，flags 列可具有当前日志文件和临时日志文件的标志设置。因此，如果值是使用 HEX 函数返回的，那么读取 flags 列中的值可能会更容易。

表 90: syslogs 表信息

该表提供了 syslogs 表列的名称、类型和描述的信息。

列	类型	描述
number	smallint	逻辑日志文件编号
uniqid	integer	日志文件 ID
size	integer	日志文件中的页数
used	integer	日志文件中已使用的页数
is_used	integer	如果文件已使用，其值为 1，否则为 0
is_current	integer	如果文件是当前文件，其值为 1，否则为 0
is_backed_up	integer	如果文件已备份，其值为 1，否则为 0
is_new	integer	如果自上次 0 级数据库空间备份以来添加了日志，其值为 1，否则为 0
is_archived	integer	如果文件已置于备份磁带上，其值为 1，否则为 0
is_temp	integer	如果文件标识为临时日志文件，其值为 1，否则为 0

列	类型	描述		
flags	smallint	标志	十六进制值	含义
		1	0x01	日志文件使用中
		2	0x02	文件是当前日志文件
		4	0x04	日志文件已备份
		8	0x08	文件是新添加的日志文件
		16	0x10	日志文件已写到数据库空间备份介质
		32	0x20	日志是临时日志文件

syslogfil 表

syslogfil 表提供有关逻辑日志文件的信息。

表 91: syslogfil 表中的信息

列	类型	描述
address	int8	逻辑文件结构的内存地址
number	small integer	逻辑文件编号
flags	integer	有关值和其含义的信息，请参阅下面标志值部份。
fillstamp	integer	填充日志文件的内部时间戳
filltime	integer	填充日志文件的 UNIX™ 时间
uniqid	integer	日志文件的唯一 ID
chunk	integer	包含日志文件的块的编号
offset	integer	块中日志文件开始位置的页偏移量
size	integer	日志文件的总页数
used	integer	日志文件中已使用的页数

标志值

标志值对应于 onstat -l 命令的标志值。

十六进制	onstat -l 标志值	含义
0x1	U	日志文件使用中
0x2	C	文件是当前日志文件
0x4	B	日志文件已备份
0x8	A	文件是新添加的日志文件
0x20	None	临时日志文件
0x40	D	文件归档后将删除日志文件
0x4000	L	日志文件包含写入的最近检查点

sysmgminfo

sysmgminfo 表提供内部分配管理器（MGM）和并行数据查询（PDQ）信息的概述。

表 92: sysmgminfo 表信息

该表提供了 sysmgminfo 表列的名称、类型和描述的信息。

列	类型	描述
max_query	integer	允许的活动查询的最大数目
total_mem	integer	总 MGM 内存量
avail_mem	integer	空闲 MGM 内存量
total_seq	integer	顺序扫描总数
avail_seq	integer	未使用的顺序扫描数
active	integer	活动 MGM 查询数
ready	integer	就绪 MGM 查询数
min_free_mem	integer	最低空闲 MGM 内存量
avg_free_mem	float	平均空闲 MGM 内存量
std_free_mem	float	标准空闲 MGM 内存量
min_free_seq	integer	最低空闲 MGM 顺序扫描数
avg_free_seq	float	平均空闲 MGM 顺序扫描数
std_free seq	float	标准空闲 MGM 顺序扫描数
max_active	integer	活动 MGM SQL 操作的最大数量
cnt_active	integer	活动 MGM SQL 操作数量
avg_active	float	活动 MGM SQL 操作平均数量
std_active	float	活动 MGM SQL 操作标准数量
max_ready	integer	就绪 MGM SQL 操作的最大数量
cnt_ready	integer	就绪 MGM SQL 操作数量
avg_ready	float	就绪 MGM SQL 操作平均数量
std_ready	float	就绪 MGM SQL 操作标准数量

sysnetclienttype

sysnetclienttype 表提供各客户端类型的网络活动的概述。

列	类型	描述
nc_cons_allowed	integer	是否允许连接
nc_accepted	integer	已接受的连接数
nc_rejected	integer	已拒绝的网络连接数
nc_reads	int8	该客户端类型的网络读取数
nc_writes	int8	该客户端类型的网络写入数

列	类型	描述
nc_name	char(18)	客户端类型的名称

sysnetglobal

sysnetglobal 表提供系统网络的概述。

列	类型	描述
ng_reads	int8	网络读取数
ng_writes	int8	网络写入数
ng_connects	int8	网络连接数
ng_his_read_count	int8	已断开连接的用户的网络读取数
ng_his_read_bytes	int8	已断开连接的用户传输到服务器的数据量
ng_his_write_count	int8	已断开连接的用户的网络写入数
ng_his_write_bytes	int8	已断开连接的用户传输到客户端的数据量
ng_num_netscbs	integer	网络订户数
ng_max_netscbs	integer	网络订户数的最大量
ng_free_thres	integer	缓冲表中最大已释放缓冲区数的阈值
ng_free_cnt	integer	达到 ng_free_thres 限制的次数
ng_wait_thres	integer	一个连接在缓冲表可以持有的最大缓冲区数的阈值
ng_wait_cnt	integer	达到 ng_wait_thres 限制的次数
ng_pvt_thres	integer	私有缓冲队列中最大已释放缓冲区数的阈值
ng_netbuf_size	integer	传输网络缓冲区的大小
ng_buf_alloc	integer	已分配的网络缓冲区数
ng_buf_alloc_max	integer	最大已分配网络缓冲区数
ng_netscb_id	integer	下一个 netscb ID

sysnetworkio 表

sysnetworkio 表包含有关系统网络的信息。

列	类型	描述
net_id	integer	Netscb ID
sid	integer	会话 ID
net_netscb	int8	Netscb 端口
net_client_type	integer	客户端类型
net_client_name	char(12)	客户端协议名称
net_read_cnt	int8	网络读取数
net_write_cnt	int8	网络写入数
net_open_time	integer	该会话的连接时间

列	类型	描述
net_last_read	integer	上次从网络读取的时间
net_last_write	integer	上次从网络写入的时间
net_stage	integer	连接 / 断开 / 接收
net_options	integer	sqlhosts 选项
net_protocol	integer	协议
net_type	char(10)	网络协议类型
net_server_fd	integer	服务器 fd
net_poll_thread	integer	轮询线程

sysonlinelog

sysonlinelog 表提供 online.log 文件中存储的信息的视图。

列	类型	描述
offset	int8	文件偏移量
next_offset	int8	到下一个消息的偏移量
line	char(4096)	来自文件的一行文本

sysprofile

sysprofile 表包含有关数据库服务器的概要文件信息。

列	类型	描述
name	char(13)	概要文件事件的名称（请参阅下面的事件列表）
value	integer	概要文件事件的值（请参阅下面的事件列表）

下表列出事件以及相应的值，其组成 sysprofile 表的行。

sysprofile 中的概要文件事件	描述
dskreads	磁盘的实际读取数
bufreads	共享内存的读取数
dskwrites	磁盘的实际写入数
bufwrites	共享内存的写入数
isamtot	调用总数
isopens	isopen 调用
isstarts	isstart 调用
isreads	isread 调用
iswrites	iswrite 调用
isrewrites	isrewrite 调用
isdeletes	isdelete 调用

sysprofile 中的概要文件事件	描述
iscommits	iscommit 调用
isrollbacks	isrollback 调用
ovlock	溢出锁表
ovuser	溢出用户表
ovtrans	溢出事务表
latchwts	锁存器请求等待数
bufwts	缓冲区等待数
lockreqs	锁请求数
lockwts	锁等待数
ckptwts	检查点等待数
deadlks	死锁数
lktouts	死锁超时数
numckpts	检查点数
plgpagewrites	写入的物理日志页数
plgwrites	物理日志写入数
llgrecs	逻辑日志记录数
llgpagewrites	逻辑日志写入数
llgwrites	写入的逻辑日志页数
pagreads	页读取数
pagwrites	页写入数
flushes	缓冲池清空数
compress	页压缩数
fgwrites	前台写入数
lruwrites	最近最少使用（LRU）写入数
chunkwrites	检查点过程中的写入数
btradata	通过索引叶节点读取的预先读取数据页数
btraidx	通过索引分支或根节点读取的预先读取数据页数
dpra	以预先读取功能读入内存的数据页数
rapgs_used	用户已用的预先读取数据页数
seqscans	顺序扫描数
totalsorts	排序总数
memsorts	可容于内存的排序数
disksorts	不可容于内存的排序数
maxsortspace	排序使用的最大磁盘空间

sysproxyagents

sysproxyagents 表包含有关所有代理服务器线程的信息。代理服务器线程在主服务器上运行，并接受辅助服务器处理 DML 操作的请求。主服务器还包含处理辅助服务器更新的代理分发器。辅助服务器根据其 ONCONFIG 文件中 UPDATABLE_SECONDARY 设置来决定创建的代理分发器实例的数量。

列	类型	描述
tid	integer	运行在主服务器上的代理服务器线程的事务 ID。 这个 ID 是由处理来自辅助服务器会话的工作的代理分发器所创键的。
flags	integer	代理服务器线程的标志。
proxy_id	integer	代表当前正在执行的代理服务器线程（TID）的代理分发器的 ID。
source_session_id	integer	辅助服务器上用户会话的 ID。
proxy_txn_id	integer	当前事务的编号。这些编号对于代理分发器是唯一的。
current_seq	integer	当前事务中当前操作的顺序编号。
sqlerrno	integer	任何 SQL 错误（或 0 成功）的错误编号。
iserrno	integer	任何 ISAM/RSAM 错误（或 0 成功）的错误编号

sysproxydistributors

sysproxydistributors 表包含有关代理分发器的信息。

在主服务器上，该表包含高可用性集群中所有代理分发器的信息。在辅助服务器上，该表仅包含那些被分配来处理辅助服务器更新的代理分发器的信息。

列	类型	描述
node_name	char	主服务器所识别的辅助服务器的名称（例如，INFORMIXSERVER、HA_ALIAS 等等）。
proxy_id	integer	代理分发器 ID。这些 ID 在高可用性集群中是唯一的。
transaction_count	integer	代理分发器当前处理的事务数
hot_row_total	integer	代理分发器处理过的热门行总数。热门行就是在辅助服务器上被多个客户端更新过多次的行。当一行被更新多次时，如果在辅助服务器上未重施来自不同会话的最新更新操作，则该辅助服务器通过在该行放置更新锁来从主服务器读取前映像。

sysproxysessions 表

sysproxysessions 表包含了正在使用重定向写功能的每个会话的信息。该表仅在辅助服务器上有效。

下表提供有关 sysproxysessions 表列的信息：

列	类型	描述
session_id	integer	辅助服务器上用户会话的 ID
proxy_id	integer	代表正在执行的代理服务器线程（TID）的代理分发器的 ID
proxy_tid	integer	运行在主服务器上的代理服务器线程的事务 ID。这个 ID 是由处理来自辅助服务器会话的工作的代理分发器所创键的。
proxy_txn_id	integer	当前事务的编号。这些编号对于代理分发器是唯一的。
current_seq	integer	当前事务中当前操作的顺序编号。

列	类型	描述
pending_ops	integer	缓冲在辅助服务器上还没传送到主服务器的操作数。
reference_count	integer	标示正在使用这个事务的信息的线程（例如：sqlexec、sync reply、recovery 等等）的数量。当 reference_count 等于 0 时，表示事务处理完成（成功或失败）。

sysproxytxnops 表

sysproxytxnops 表包含通过每个代理分发器正在运行的每个事务的信息。

在主服务器上，此表包含高可用性集群中所有代理分发器的信息。而在辅助服务器上，此表包含用于处理辅助服务器更新的代理分发器的信息。

下表提供有关 sysproxytxnops 表列的信息：

列	类型	描述
proxy_id	integer	代理分发器 ID。在高可用性集群中这些 ID 是唯一的。
proxy_txn_id	integer	事务编号。这些编号对于代理分发器是唯一的。
sequence_number	integer	操作的序号。
operation_type	char(10)	执行的操作类型：插入、更新、删除或其它。
rowidn	integer	应用该操作的行的 ID。
table	char	完整表名称，修整以适合合理的长度。格式：*database*:*owner*.*tablename*
sqlerrno	integer	SQL 错误（或者 0 成功）的错误代码

sysproxytxns 表

sysproxytxns 表包含通过每个代理分发器运行的所有当前事务的信息。

在主服务器上，此表包含高可用性集群中每个代理分发器的信息。而在辅助服务器上，此表仅包含用于处理辅助服务器更新的代理分发器的信息。

下表提供有关 sysproxytxns 表列的信息：

列	类型	描述
proxy_id	integer	代理分发器 ID。在高可用性集群中这些 ID 是唯一的。
proxy_txn_id	integer	事务编号。对于代理分发器这些编号是唯一的。
reference_count	integer	标示使用此事务信息的线程（例如：sqlexec、sync reply、recovery等等）的数目。当数量为 0 时，表示事务处理已经完成（成功或者失败）。
pending_ops	integer	在主服务器上：表示从辅助服务器上接收的还未处理的操作的数量。而在辅助服务器上，表示缓冲在辅助服务器上还未被发送到主服务器的操作的数量。
proxy_sid	integer	代理会话 ID

sysptprof 表

sysptprof 表列出表空间的信息。表空间对应于表。t

表的概要信息仅在表打开时可用。当打开的表的最后一个用户关闭此表时，共享内存中的表空间会被释放，而且任何概要文件统计信息都将丢失。

下表提供 sysptprof 表列的信息：

列	类型	描述
dbsname	char(128)	数据库名
tabname	char(128)	表名
partnum	integer	分区（表空间）编号
lockreqs	integer	锁请求数
lockwts	integer	锁等待数
deadlks	integer	死锁数
lktouts	integer	锁超时的次数
isreads	integer	isreads 的数量
iswrites	integer	iswrites 的数量
isrewrites	integer	isrewrites 的数量
isdeletes	integer	isdeletes 的数量
bufreads	integer	缓冲区读取数
bufwrites	integer	缓冲区写入数
seqscans	integer	顺序扫描数
pagreads	integer	页读取数
pagwrites	integer	页写入数

sysrepevtreg 表

使用 sysrepevtreg 伪表为连接管理器、SinoDB® 开放管理工具 (OAT) 或任一客户端应用程序的一组预定义事件注册。完成注册后，连接管理器、OAT 或任一客户端应用程序可以通过查询该表来接收事件数据。

下表提供 sysrepevtreg 表列的信息：

表 93: sysrepevtreg 表信息

此表提供 sysrepevtreg 表列的名称、类型和描述的信息。

列	类型	描述
evt_bitmap	integer	事件 ID 位图
evt_timeout	integer	客户端可等待事件数据的最长时间（以秒为单位）。有效值是： • 0；不等待（默认） • -1；永远等待 • *n*（其中 *n* > 0）等等 *n* 秒
evt_hwm	integer	待定事件列表的高水位标记
evt_info	char(256)	事件信息（尚未实现）

sysrepstats 表

使用 sysrepstats 表向连接管理器和 SinoDB® 开放管理工具 (OAT) 发布事件。连接管理器、OAT 和客户端应用程序可通过将事件发布到 sysrepstats 伪表来相互通信。

下表提供 sysrepstats 表列的信息：

表 94: sysrepstats 表信息

此表提供 sysrepstats 表列的名称、类型和描述的信息。

列	类型	描述
repstats_type	integer	事件 ID
repstats_subtype	integer	子事件 ID
repstats_time	integer	事件发起的时间
repstats_ver	integer	事件数据版本号
repstats_desc	lvarchar	事件数据

sysrepstats 和 sysrepevtreg 表的用户接口
客户端应用程序可通过往 sysrepstats 伪表插入事件信息来向连接管理器或其它客户端发布事件。客户端应用程序可使用 sysmaster 的 sysrepevtreg 伪表来注册事件，并可通过对 sysrepstats 发出 select 或 fetch 语句来接收事件数据

发布事件到 sysrepstats 伪表为诸如 SinoDB® 开放管理工具 (OAT) 之类的程序提供了与连接管理器通信的能力。通过发布事件到 sysrepstats，您可对连接管理器发出控制消息，而无需直接连接到连接管理器本身。

当连接管理器注册它要接收事件时，它会传递一个要接收的事件类型的位图。当接收到事件时，这些事件会被发布到请求的线程中。

事件类

下表列出每个事件类，以及其位图与描述。

表 95: 事件类

此表列出每个事件类，以及其位图与描述。

事件类名称	位值	描述
REPEVT_CLUST_CHG	0x1	高可用性集群变更的事件类
REPEVT_CLUST_PERFSTAT	0x2	高可用性集群中服务器节点的工作负载统计信息的事件类
REPEVT_CLUST_LATSTAT	0x4	高可用性集群中服务器节点的复制延迟信息的事件类
REPEVT_CM_ADM	0x8	连接管理器管理命令
REPEVT_SRV_ADM	0x10	Event class for server mode changes
REPEVT_ER_ADM	0x20	与 Enterprise Replication（ER）相关的事件的事件类
REPEVT_CLIENT	0x40	用户自定义客户端事件

REPEVT_CLUST_CHG 事件类的子事件

下表列出 REPEVT_CLUST_CHG 事件类的子事件：

表 96: REPEVT_CLUST_CHG 事件类的子事件

此表列出 REPEVT_CLUST_CHG 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_SUB_CLUST_ADD	1	添加新节点到高可用性集群	仅在高可用性集群中的主服务器上
REPEVT_SUB_CLUST_DROP	2	删除高可用性集群中的节点	仅在高可用性集群中的主服务器上
REPEVT_SUB_CLUST_CON	3	高可用性辅助节点连接到主服务器	仅在高可用性集群中的主服务器上
REPEVT_SUB_CLUST_DIS	4	高可用性辅助节点从主服务器断开	仅在高可用性集群中的主服务器上
REPEVT_SUB_CLUST_NEWPRIM	5	高可用性主服务器节点变更	仅在高可用性集群中的辅助服务器上
REPEVT_SUB_CLUST_DROFF	6	HDR 辅助节点从主服务器断开	HDR 主服务器和辅助服务器
REPEVT_SUB_CLUST_DRON	7	HDR 辅助节点连接到主服务器	HDR 主服务器和辅助服务器

REPEVT_CLUST_PERFSTAT 事件类的子事件

下表列出 REPEVT_CLUST_PERFSTAT 事件类的子事件：

表 97: REPEVT_CLUST_PERFSTAT 事件类的子事件

此表列出 REPEVT_CLUST_PERFSTAT 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_ SUB_LOCAL_PERFSTAT	1	Work 本地服务器的工作负载统计信息	高可用性集群中的所有服务器上
REPEVT_ SUB_REMOTE_PERFSTAT	2	Work 高可用性辅助服务器的工作负载统计信息	仅在高可用性集群中的主服务器上

REPEVT_CLUST_LATSTAT 事件类的子事件

下表列出 REPEVT_CLUST_LATSTAT 事件类的子事件：

表 98: REPEVT_CLUST_LATSTAT 事件类的子事件

此表列出 REPEVT_CLUST_LATSTAT 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_SUB_LOCAL_LATSTAT	1	高可用性集群中的辅助服务器的复制延迟统计信息	仅在高可用性集群中的主服务器上

REPEVT_CM_ADM 事件类的子事件

下表列出 REPEVT_CM_ADM 事件类的子事件：

表 99: REPEVT_CM_ADM 事件类的子事件

此表列出 REPEVT_CM_ADM 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_SUB_CM_ADM_REQ	1	命令请求	所有 SinoDB® 服务器实例
REPEVT_SUB_CM_ADM_ACK	2	命令响应	所有 SinoDB® 服务器实例
REPEVT_SUB_CM_REG	3	已向服务器注册的连接管理器	所有 SinoDB® 服务器实例
REPEVT_SUB_CM_DEREG	4	以向服务器注销的连接管理器	所有 SinoDB® 服务器实例
REPEVT_SUB_CM_FATAL	5	连接管理器终止而没有向服务器注销	所有 SinoDB® 服务器实例

REPEVT_SRV_ADM 事件类的子事件

下表列出 REPEVT_SRV_ADM 事件类的子事件：

表 100: REPEVT_SRV_ADM 事件类的子事件

此表列出 REPEVT_SRV_ADM 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_ SUB_SRV_BLK	1	由于 DDRBLOCK 而阻塞服务器	所有 SinoDB® 服务器实例
REPEVT_ SUB_SRV_UBLK	2	取消对服务器的阻塞；除去 DDRBLOCK	所有 SinoDB® 服务器实例

REPEVT_ER_ADM 事件类的子事件

下表列出 REPEVT_ER_ADM 事件类的子事件：

表 101: REPEVT_ER_ADM 事件类的子事件

此表列出 REPEVT_ER_ADM 事件类的子事件。

子事件名称	值	描述	可用条件：
REPEVT_SUB_ER_SPOOL_FULL	1	等待向队列数据智能大对象空间/数据库空间，或是向分组器调页智能大对象空间添加空间时，阻碍 ER。	Enterprise Replication 服务器节点

sysrsslog

sysrsslog 表捕获在主服务器上有关 RS 辅助服务器的信息。

表 102: sysrsslog 表信息

此表提供 sysrsslog 表列的名称、类型和描述的信息。

列	类型	描述
server_name	char(128)	服务器名
from_cache	integer	从日志缓冲区高速缓存读取的总页数
from_disk	integer	从磁盘读取的总页数

列	类型	描述
logpages_tossed	integer	未写入日志缓冲区高速缓存的总日志页数

sysscblst

sysscblst 表的这些列提供有关会话内存量的信息。

列	类型	描述
memtotal	integer	可用内存总量
memused	integer	已用内存总量

syssesappinfo

sysmaster 中的 syssesappinfo 表显示分布式关系数据库结构（DRDA®）客户端会话的信息。syssesappinfo 表有以下列。

表 103: syssesappinfo 表列信息

此表包含 syssesappinfo 表列信息。

列	类型	描述
sesapp_sid	INTEGER	客户端会话 ID
sesapp_name	CHAR(128)	会话应用程序名称
sesapp_value	CHAR(512)	会话值

syssesprof

syssesprof 表列出用户操作（例如：写入、删除或提交）的发生数的累计计数。

列	类型	描述
sid	integer	会话 ID
lockreqs	integer	所请求的锁数
locksheld	integer	当前持有的锁数
lockwts	integer	等待锁的次数
deadlks	integer	所检测到的死锁数
lktouts	smallint	死锁超时数
logrecs	integer	已写入的逻辑日志记录数
isreads	integer	读取数
iswrites	integer	写入数
isrewrites	integer	重新写入数
isdeletes	integer	删除数
iscommits	integer	提交数
isrollbacks	integer	回滚数
longtxs	integer	长事务数

列	类型	描述
bufreads	integer	缓冲区读取数
bufwrites	integer	缓冲区写入数
seqscans	integer	顺序扫描数
pagreads	integer	页读取数
pagwrites	integer	页写入数
total_sorts	integer	排序总数
dsksorts	integer	不适合内存的排序数
max_sortdiskspace	integer	排序所使用的最大空间
logspused	integer	当前会话事务所使用的逻辑日志空间的字节数
maxlogsp	integer	会话事务曾使用的逻辑日志空间的最大字节数

syssessions

syssessions 表提供连接到数据库服务器的每个用户的一般信息。state 列中，每个位位置代表一个单独的标志。这样，如果值是使用 HEX 函数返回的，那么读取 state 列中的值可能会更容易。

表 104: syssessions 表信息

此表提供 syssessions 表列的名称、类型和描述的信息。

列	类型	描述
sid	integer	会话 ID
username	char(32)	用户 ID
uid	smallint	用户 ID 号
pid	integer	客户端进程 ID
hostname	char(256)	客户端主机名
tty	char(16)	用户的 stderr 文件名
connected	integer	用户连接到数据库服务器的时间
feprogram	char(255)	保留，供将来使用
pooladdr	integer	会话池地址
is_wlatch	integer	如果会话主线程正在等待锁存器，那么为 1
is_wlock	integer	如果会话主线程正在等待锁，那么为 1
is_wbuff	integer	如果会话主线程正在等待缓冲区，那么为 1
is_wckpt	integer	如果会话主线程正在等待检查点，那么为 1
is_wlogbuf	integer	如果会话主线程正在等待日志缓冲区，那么为 1
is_wtrans	integer	如果会话主线程正在等待事务，那么为 1
is_monitor	integer	如果会话是特殊的监视进程，那么为 1
is_incrit	integer	如果会话主线程处于临界段中，那么为 1

列	类型	描述		
state	integer	标志	十六进制	含义
		1	0x00000001	用户结构在使用中
		2	0x00000002	等待锁存器
		4	0x00000004	等待锁
		8	0x00000008	等待缓冲区
		16	0x00000010	等待检查点
		32	0x00000020	在读调用中
		64	0x00000040	向备份磁带写入逻辑日志文件
		256	0x00000100	在临界段中
		512	0x00000200	特殊守护程序
		1024	0x00000400	归档中
		2048	0x00000800	清除死进程
		4096	0x00001000	正在等待日志缓冲区的写入
		8192	0x00002000	特殊缓冲区清空线程
		16384	0x00004000	远程数据库服务器
		32768	0x00008000	用于设置 RS_timeout 死锁超时
		65536	0x00010000	常规死锁超时
		262144	0x00040000	正在等待事务
		524288	0x00080000	会话主线程
		1048576	0x00100000	构建索引的线程
		2097152	0x00200000	B 型树清除线程

syssmx

syssmx 表提供 SMX（服务器多路复用组）连接信息。

表 105: syssmx 表列信息

此表包含 syssmx 表列信息。

列	类型	描述
address	int8	SMX 管道地址
name	char(128)	目标服务器名称
encryption_status	char(20)	保留供将来使用
buffers_sent	integer	已发送的缓冲区数
buffers_recv	integer	已接收到的缓冲区数
bytes_sent	int8	已发送的字节数
bytes_recv	int8	已接收到的字节数

列	类型	描述
reads	integer	读取调用数
writes	integer	写入调用数
retries	integer	写入调用重试数

syssmxses

syssmxses 表提供 SMX（服务器多路复用组）会话信息。

表 106: syssmxses 表列信息

此表包含 syssmxses 表列信息。

列	类型	描述
name	char(128)	目标服务器名称
address	int8	SMX 会话地址
client_type	char(20)	SMX 客户端类型
reads	integer	读取调用数
writes	integer	写入调用数

syssqexplain 表

syssqexplain 伪表存储有关 SQL 查询的信息。

存储的信息包括查询优化器的计划、返回行数的估计，和查询的相对成本。

表 107: syssqexplain 伪表

列	类型	描述
sqx_sessionid	INTEGER	与 SQL 语句相关联的会话 ID。
sqx_sdbno	INTEGER	查询在会话 ID 数组中的位置。
sqx_iscurrent	CHAR	查询是否为当前 SQL 语句。
sqx_executions	INTEGER	查询执行的总次数。
sqx_cumtime	FLOAT	运行查询的累计时间。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_bufreads	INTEGER	运行查询时执行的缓冲区读取数。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_pagereads	INTEGER	运行查询时执行的页读取数。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_bufwrites	INTEGER	运行查询时执行的缓冲区写入数。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_pagewrites	INTEGER	运行查询时执行的页写入数。 重要： 如果禁用 SQL 追踪，那么显示零。

列	类型	描述
sqx_totsorts	INTEGER	运行查询时执行的排序数。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_dsksorts	INTEGER	运行查询时执行的磁盘排序数。 重要： 如果禁用 SQL 追踪，那么显示零。
sqx_sortspmax	INTEGER	排序所需的最大磁盘空间。
sqx_conbno	SMALLINT	在 conblock 列表中的位置。
sqx_ismain	CHAR	查询是否在语句的主块中。
sqx_selflag	VARCHAR(200,0)	SQL 语句类型，例如：SELECT、UPDATE、DELETE。
sqx_estcost	INTEGER	查询的估计成本
sqx_estrows	INTEGER	查询返回的估计行数
sqx_seqscan	SMALLINT	查询所用的顺序扫描数
sqx_srtscan	SMALLINT	查询所用的排序扫描数
sqx_autoindex	SMALLINT	查询所用的自动索引扫描数
sqx_index	SMALLINT	查询所用的索引路径数
sqx_remsql	SMALLINT	查询所用的远程路径数
sqx_mrgjoin	SMALLINT	查询所用的排序合并联接数
sqx_dynhashjoin	SMALLINT	查询所用的动态哈希联接数
sqx_keyonly	SMALLINT	查询所用的键扫描数
sqx_tempfile	SMALLINT	查询所用的临时文件数
sqx_tempview	SMALLINT	查询所创建的视图的临时表数
sqx_secthreads	SMALLINT	查询所用的辅助线程数
sqx_sqlstatement	CHAR	运行的 SQL 查询

syssqltrace

syssqltrace 表提供有关单个 SQL 语句的详细信息。

列	类型	描述
sql_id	int8	唯一 SQL 执行 ID
sql_address	int8	语句在代码块中的地址
sql_sid	int	运行 SQL 语句的用户的数据库会话 ID
sql_uid	int	运行 SQL 语句的用户 ID
sql_stmttype	int	语句类型
sql_stmtname	char(40)	显示为单词的语句类型
sql_finishtime	int	语句完成的时间（UNIX™）
sql_begintxtime	int	事务开始的时间
sql_runtime	float	语句执行时间

列	类型	描述
sql_pgreads	int	SQL 语句的磁盘读取数
sql_bfreads	int	SQL 语句的缓冲区读取数
sql_rdcache	float	从缓冲池读取页的时间百分比
sql_bfidxreads	int	索引页缓冲区读取数
sql_pgwrites	int	写入磁盘的页数
sql_bfwrites	int	修改并返回缓冲池三页数
sql_wrcache	float	页写入缓冲池而不是磁盘的时间百分比
sql_lockreq	int	SQL 语句所需的锁数
sql_lockwaits	int	SQL 语句等待锁的次数
sql_lockwttime	float	SQL 语句期间系统等待锁的时间
sql_logspace	int	逻辑日志中 SQL 语句所用的空间量
sql_sorttotal	int	为语句运行的排序数
sql_sortdisk	int	在磁盘上运行的排序数
sql_sortmem	int	在内存中运行的排序数
sql_executions	int	SQL 语句运行的次数
sql_totaltime	float	运行语句所用的时间总量
sql_avgtime	float	运行语句所用的平均时间量
sql_maxtime	float	执行 SQL 语句所用的最大时间量
sql_numiowaits	int	I/O 操作必须等待的次数
sql_avgiowaits	float	SQL 语句必须等待的平均时间量
sql_totaliowaits	float	SQL 语句等待 I/O 的总时间量。这不包含任何异步 I/O。
sql_rowspersec	float	生成的平均行数（每秒）
sql_estcost	int	与 SQL 语句相关联的成本
sql_estrows	int	按优化器预测为 SQL 语句返回的估计行数
sql_actualrows	int	SQL 语句返回的行数
sql_sqlerror	int	SQL 错误号
sql_isamerror	int	RSAM/ISAM 错误号
sql_isollevel	int	SQL 语句的隔离级别
sql_sqlmemory	int	执行 SQL 语句所需的字节数
sql_numiterators	int	语句所用的迭代器数
sql_database	char(128)	数据库名
sql_numtables	int	执行 SQL 语句所用的表数
sql_tablelist	char(4096)	SQL 语句中直接引用的表名列表。如果 SQL 语句触发对其他表执行语句的触发器，那些表是不列出的。
sql_statement	char(1600)	执行的 SQL 语句

syssqltrace_hvar

syssqltrace_hvar 表描述有关 SQL 追踪主变量的信息。

列	类型	描述
sql_id	int8	SQL 执行 ID
sql_address	int8	SQL 语句块的地址
sql_hvar_id	int	SQL 主变量 ID
sql_hvar_flags	int	主变量标志
sql_hvar_typeid	int	主变量类型 ID
sql_hvar_xtypeid	int	主变量 xtype ID
sql_hvar_ind	int	主变量索引
sql_hvar_type	char(128)	主变量类型
sql_hvar_data	char(8192)	主变量值

syssqltrace_info

syssqltrace_info 表描述有关 SQL 概要文件跟踪系统的信息。

列	类型	描述
flags	integer	SQL 跟踪标志
ntraces	integer	要跟踪的项数
tracesize	integer	为每个 SQL 跟踪项存储的文本的大小
duration	integer	跟踪缓冲区（以秒为单位）
sqlseen	int8	启动或重调大小后跟踪的 SQL 项数
starttime	integer	跟踪的启用时间
memoryused	int8	SQL 跟踪所用内存的字节数

syssqltrace_iter

syssqltrace_iter 表列出 SQL 语句迭代器。

列	类型	描述
sql_id	int8	SQL 执行 ID
sql_address	int8	SQL 语句块的地址
sql_itr_address	int8	迭代器的地址
sql_itr_id	int	迭代器 ID
sql_itr_left	int	向左的迭代器 ID
sql_itr_right	int	向右的迭代器 ID
sql_itr_cost	int	迭代器成本
sql_itr_estrows	int	迭代器预估行数
sql_itr_numrows	int	迭代器实际处理的行数

列	类型	描述
sql_itr_type	int	迭代器类型
sql_itr_misc	int	迭代器杂项标志
sql_it_info	char(256)	显示为文本的迭代器杂项标志

syssrcrss

syssrcrss 表提供了主服务器上 RS 辅助服务器的相关统计信息。

表 108: syssrcrss 表列信息

此表包含 syssrcrss 表列信息。

列	类型	描述
address	int8	RS 辅助服务器控制块地址
server_name	char(128)	数据库服务器名称
server_status	char(20)	静默、活动或不活动
connection_status	char(20)	已连接或已断开
log_transmission_status	char(20)	活动或阻塞
next_page_tosend_log_uniq	integer	要发送的下一页的唯一日志 ID
next_page_tosend_log_page	integer	要发送的下一页的页号
seq_tosend	integer	上次发送的缓冲区的顺序 ID
last_seq_acked	integer	上次应答的缓冲区的顺序 ID

syssrcsds

syssrcsds 表提供了主服务器上 SD 辅助服务器的相关统计信息。

syssrcsds 表包含下表显示的列。

在下表中，每一行包含了 syssrcsds 表列的名称、及其数据类型和描述。

列	类型	描述
address	int8	SD 辅助服务器控制块地址
source_server	char(128)	主数据库服务器名称
connection_status	char(20)	已连接或已断开
last_received_log_uniq	integer	上次接收的日志页的唯一日志 ID
last_received_log_page	integer	上次接收的日志页的页号
next_lpgtoread_log_uniq	integer	要读取的下一个日志页的唯一日志 ID
next_lpgtoread_log_page	integer	要读取的下一个日志页的页号
last_acked_lsn_uniq	integer	上次应答的 LSN 的唯一日志 ID
last_acked_lsn_pos	integer	上次应答 LSN 的日志位置
last_seq_received	integer	上次接收到的缓冲区的顺序 ID
last_seq_acked	integer	上次应答的缓冲区的顺序 ID

列	类型	描述
cur_pagingfile	char(640)	当前调页文件名称
cur_pagingfile_size	int8	当前调页文件大小
old_pagingfile	char(640)	以前的调页文件名称
old_pagingfile_size	int8	以前的调页文件大小

systabnames

systabnames 表描述数据库服务器所管理的每个表。

列	类型	描述
partnum	integer	表空间标识
dbsname	char(128)	数据库名称
owner	char(32)	所有者的用户 ID
tabname	char(128)	表名称
collate	char(32)	与支持 GLS 的数据库相关联的对照

systhreads

systhreads 表提供有关每个线程的信息。

列	类型	描述
th_id	INTEGER	线程的标识
th_addr	INTEGER	线程控制块的内存地址
th_joinlist	INTEGER	如果有一个列表的线程正在等待该线程退出，那么 th_joinlist 列显示该列表中第一个线程的地址。
th_joinnext	INTEGER	如果有一个列表的线程正在等待该线程退出，那么 th_joinnext 列显示该列表中下一个线程的地址。
th_joinee	INTEGER	该线程正在等待其退出线程的地址。
th_name	CHAR(12)	线程的名称。
th_state	INTEGER	线程状态的代码.
th_priority	INTEGER	线程的等级
th_class	INTEGER	运行线程的虚拟处理器的类的代码
th_vpid	INTEGER	上次计划运行线程的虚拟处理器的 ID
th_mtxwait	INTEGER	此线程正在等待的互斥的地址。
th_conwait	INTEGER	此线程正在等待的条件的地址
th_waketime	INTEGER	最近休眠的到期时间。该时间是由内部时钟计算的。值为 -1 表示时间值是不定的。
th_startwait	INTEGER	最近等待开始的时间。该时间是由内部时钟计算的。
th_startrun	INTEGER	最近执行开始的时间。该时间是由内部时钟计算的。

systrgrss

systrgrss 表提供了 RS 辅助服务器上 RS 辅助服务器的相关统计信息。

在下表中，每一行包含了 systrgrss 表列的名称，及其数据类型和描述。

列	类型	描述
address	int8	RS 辅助服务器控制块地址
source_server	char(128)	服务 RS 辅助服务器的源服务器
connection_status	char(20)	已连接或已断开
last_received_log_uniq	integer	上次接收到的日志页的唯一日志 ID
last_received_log_page	integer	上次接收到的日志页的页号
last_seq_received	integer	上次接收到的缓冲区的顺序 ID
last_seq_acked	integer	上次应答的缓冲区的顺序 ID

systrgsds

systrgsds 表提供了 SD 辅助服务器上 SD 辅助服务器的相关统计信息。

systrgsds 表包含了这些列：

在下表中，每一行包含了 systrgsds 表列的名称，及其数据类型和描述。

列	类型	描述
address	int8	SD 辅助服务器控制块地址
source_server	char(128)	服务 SD 辅助服务去的源服务器
connection_status	char(20)	已连接或已断开
last_received_log_uniq	integer	上次接收到的日志页的唯一日志 ID
last_received_log_page	integer	上次接收到的日志页的页号
next_lptoread_log_uniq	integer	要读取的下一个日志页的唯一日志 ID
next_lptoread_log_page	integer	要读取的下一个日志页的页号
last_acked_lsn_uniq	integer	上次应答的 LSN 的唯一日志 ID
last_acked_lsn_pos	integer	上次应答的 LSN 的日志位置
last_seq_received	integer	上次接收到的缓冲区的顺序 ID
last_seq_acked	integer	上次应答的缓冲区的顺序 ID
cur_pagingfile	char(640)	当前的调页文件名称
cur_pagingfile_size	int8	当前的调页文件大小
old_pagingfile	char(640)	以前的调页文件名称
old_pagingfile_size	int8	以前的调页文件大小

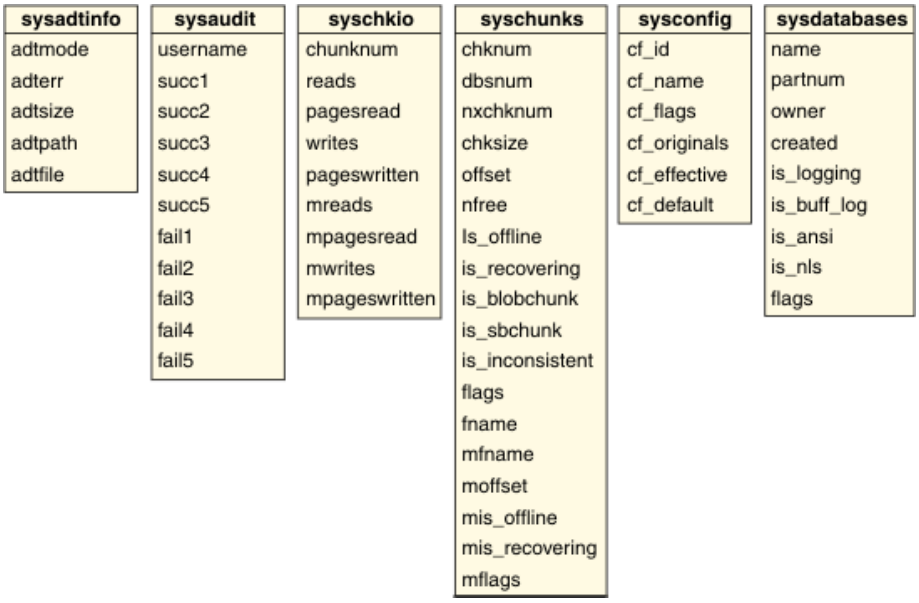
sysvpprof

sysvpprof 表列出每个虚拟处理器的用户和系统 CPU 时间。

列	类型	描述
vpid	integer	虚拟处理器 ID
	char(50)	虚拟处理器类型： • cpu • adm • lio • pio • aio • tli • soc • str • shm • opt • msc • adt
usercpu	float	用户时间（微秒数）
syscpu	float	系统时间（微秒数）

SMI 表映射

图 8: SMI 表中的列 在第241页 显示某些 SMI 表中的列。



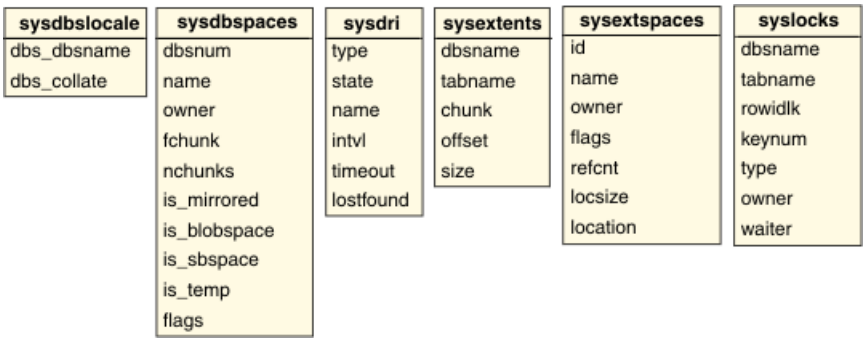
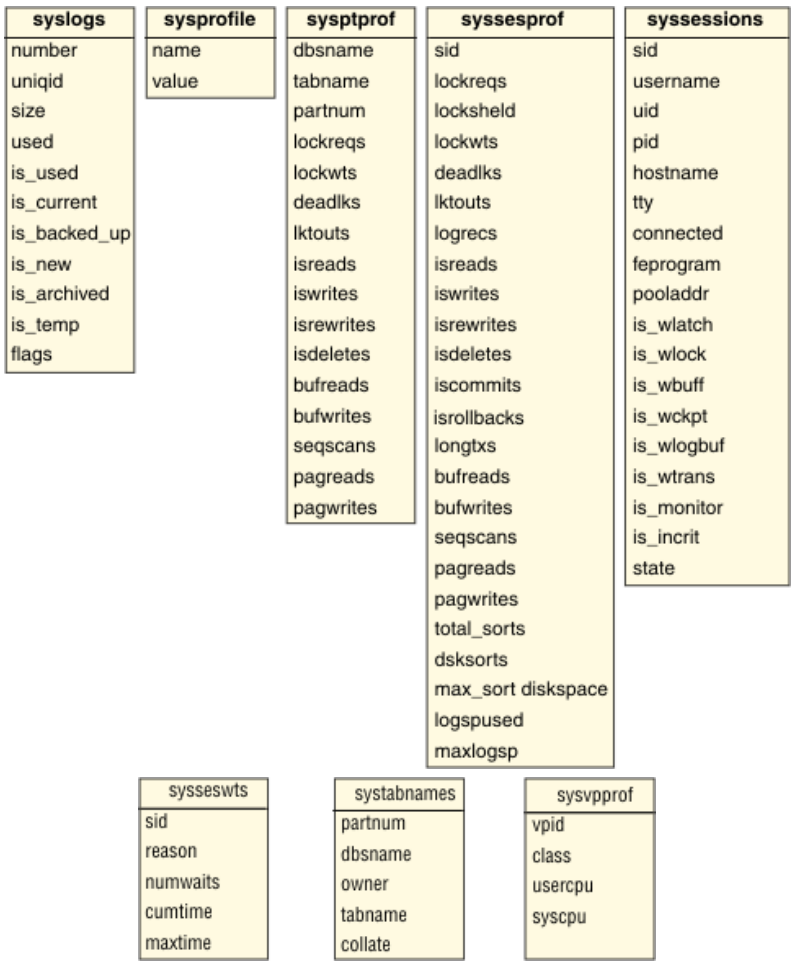


图 8: SMI 表中的列



从 onstat 获得的 SMI 表中的信息

要获得 onstat 实用程序所提供的信息，可以使用 SQL 查询适当的 SMI 表。下表指示了要查询的 SMI 表，以获取由给定 onstat 选项提供的信息。有关 onstat 选项的描述，请参阅监视数据库服务器状态 在第400页。

onstat 选项	要查询的 SMI 表	不在 SMI 表中的 onstat 字段
-d	sysdbspaces syschunks	address bpages
-D	sysdbspaces syschkio	
-F	sysprofile	address flusher snoozer state data

onstat 选项	要查询的 SMI 表	不在 SMI 表中的 onstat 字段
-g ath	systhreads	
-g dri	sysdri	Last DR CKPT (id/pg)
-g glo	sysvpprof	按类列出的虚拟处理器列表
-g ipl	sysipl	
-g rss	sysrsslog systrgrss syssrcrss	
-g his	syssqltracing	
-g sds	syssrcsds systrgsds	
-g smx	syssmx	
-g smx ses	syssmxses	
-k	syslocks	address lklist tblsnum
-l	syslogs sysprofile	所有物理日志字段（numpages 和 numwrits 除外）所有逻辑日志缓冲区字段（numrecs、numpages 和 numwrits 除外）address begin % used
-p	sysprofile	
-u	syssessions syssesprof	address wait nreads nwrites

sysadmin 数据库

sysadmin 数据库包含了含有并组织调度程序任务和传感器的表，存储传感器所收集的数据的表，以及记录调度程序作业和 SQL 管理 API 函数结果的表。

缺省情况下，仅授予用户 informix 访问 sysadmin 数据库的权限；也可授予其他用户访问 sysadmin 数据库的权限。

请不要删除或更改 sysadmin 数据库，因为有些重要数据库服务器组件使用它。然而，如果根数据库空间没有足够空间来存储任务属性和命令历史信息，那么您可以将 sysadmin 从默认的根数据库空间位置移出。要移动 sysadmin 数据库，请使用带有 reset sysadmin 参数的 SQL 管理 API admin() 或 task() 函数。

> 重要： 移动 sysadmin 数据库将重置数据库回到最初创建时的状态；将丢失所有数据、命令历史和结果表。只有内置的任务、传感器和阈值会保留在 sysadmin 的表中

相关链接

reset sysadmin 参数: 移动 sysadmin 数据库 (SQL 管理 API) 在第675页

调度程序表

调度程序在 sysadmin 数据库的以下五个表中存储有关任务和传感器的信息：ph_task、ph_run、ph_group、ph_alert 和 ph_threshold。

调度程序是一个管理工具，它使数据库服务器可以在预定的时间或由服务器内部确定的时间执行数据库函数和过程。调度程序所使用的五个表包含了自动运行的内置任务和传感器。您可以通过在这些表中插入行来添加自己的任务和传感器。这些表在其列之间有着关系，如下图所示。

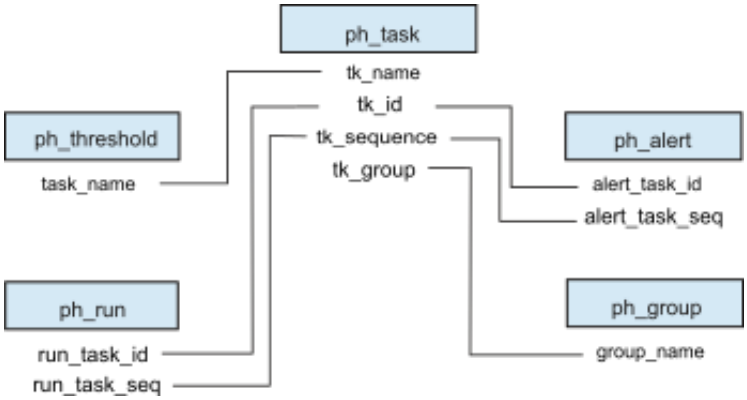


图 9: 调度程序表之间的关系

有关使用调度程序的详细信息，请参阅《*SinoDB*® 管理员指南》。

相关链接

《*SinoDB* 管理员指南》: 调度程序表

ph_task 表

ph_task 表包含有关调度程序任务和传感器的信息。ph_task 表包含调度自动运行的内置任务和传感器。

表 109: ph_task 表

列	类型	描述
tk_id	serial	顺序任务 ID。 由系统更新；不要修改。 在 ph_alert 表的 alert_task_id 列和 ph_run 表的 run_task_id 列中引用。
tk_name	char(36)	任务名称。在该列上的唯一索引确保没有两个名称是相同的。 在 ph_threshold 表的 task_name 列中引用。
tk_description	lvarchar	有关该任务或传感器的功能的描述。
tk_type	char(18)	任务类型： • TASK：在特定时间和频率调用任务 • SENSOR：（默认）收集、存储和清除结果表中数据的任务。 • STARTUP TASK：仅在服务器启动时运行的任务。 • STARTUP SENSOR：仅在服务器启动时运行的传感器。
tk_sequence	integer	当前的数据收集号。 由系统更新；不要修改。 在 ph_alert 表的 alert_task_id 列和 ph_run 表的 run_task_seq 列中引用。
tk_result_table	varchar	结果表名称，用于存储传感器所收集的数据。该表由 tk_create 列中的 CREATE TABLE 语句创建。
tk_create	lvarchar	该 CREATE TABLE 语句用于创建结果表以存储传感器所收集的数据。

列	类型	描述
		该表中的其中一个列必须命名为 ID 以保存 tk_sequence 值。这个值指示行的年龄，可用于清除该行。
tk_dbs	varchar(250)	任务所运行的数据库。 默认是 sysadmin。
tk_execute	lvarchar	要执行的 SQL 语句。 命令长度不可超过 2048 字节。
tk_delete	interval day(2) to second	从结果表中删除早于该时间间隔的数据。 默认是 1:00:00（一天）。
tk_start_time	datetime hour to second	任务或传感器的开始时间。 默认是 08:00:00。
tk_stop_time	datetime hour to second	在该时间之后，不可安排运行任务或传感器。数据库在下一个有效日安排下一次执行。 默认是 19:00:00。可以是 NULL 值，表示没有停止时间。
tk_frequency	interval day(2) to second	该任务或传感器运行的频率。 默认是 1（一天一次）。
tk_next_execution	datetime year to second	该任务或传感器的下次运行时间。 启动任务或传感器运行后，此值为 NULL。 任务或传感器启用后，数据库服务器将根据 tk_start_time、tk_stop_time 和 tk_frequency 列的值，以及任务或传感器是一周那几天启用（根据 tk_monday、tk_tuesday、tk_wednesday、tk_thursday、tk_friday、和 tk_sunday 等列的值决定），来计算该时间。 例如：*new_next_execution_time = current_next_execution_time +* tk_frequency，其中 *new_next_execution_time* 大于 *current_next_execution_time*。如果没有 tk_frequency，则该任务运行一次。
tk_total_executions	integer	任务或传感器运行的次数。 由系统更新；不要修改。 默认是 0。
tk_total_time	float	执行该任务或传感器所使用的总时间。 由系统更新；不要修改。 默认是 0.0 秒。
tk_monday	boolean	该任务或传感器是否在星期一运行。 默认是 T （true）。
tk_tuesday	boolean	该任务或传感器是否在星期二运行。

列	类型	描述
		默认是 T （true）。
tk_wednesday	boolean	该任务或传感器是否在星期三运行。 默认是 T （true）。
tk_thursday	boolean	该任务或传感器是否在星期四运行。 默认是 T （true）。
tk_friday	boolean	该任务或传感器是否在星期五运行。 默认是 T （true）。
tk_saturday	boolean	该任务或传感器是否在星期六运行。 默认是 T （true）。
tk_sunday	boolean	该任务或传感器是否在星期日运行。 默认是 T （true）。
tk_attributes	integer	标志 由系统更新；不要修改。
tk_group	varchar(128)	组名称。 必须与 ph_group 表中的 group_name 列的值相同。 默认是 MISC。
tk_enable	boolean	是否启用该任务或传感器。 默认是 T（启用该任务）。
tk_priority	integer	作业优先级，数值范围为 0 - 5，越高的数值表示越优先。如果有好几个作业要同时执行，则具有最高优先级的作业最先执行。 默认是 0（低优先级）。

ph_run 表

ph_run 表包含关于每个已运行的调度程序任务或传感器的运行方式和运行时间的信息。

表 110: ph_run 表

列	类型	描述
run_id	serial	执行期间生成的顺序 ID。 由系统更新；不要修改。
run_task_id	integer	作业 ID。 引用 ph_task 表的 tk_id 列。
run_task_seq	integer	任务或传感器的唯一序号。 引用 ph_task 表的 tk_sequence 列。

列	类型	描述
run_retcode	integer	存储过程函数或用户定义的例程的返回代码，或者是 SQL 语句的 SQL 代码。
run_time	datetime year to second	该任务或传感器运行的时间。
run_duration	float	运行该任务或传感器所使用的时间（以秒为单位）。
run_ztime	integer	服务器统计（onstat -z 命令）上次运行的时间。
run_btime	integer	服务器的启动时间。
run_mttime	integer	服务器的内部计数器。

ph_group 表

ph_group 表包含关于调度程序组名称的信息。ph_group 表包含几个用于分类内置任务和传感器的组，以及默认组 MISC。

表 111: ph_group 表

列	类型	描述
group_id	serial	组 ID。 由系统更新；不要修改。
group_name	varchar(128)	该组的唯一名称。 在 ph_task 表的 tk_group 列中引用。
group_description	lvarchar	该组的描述。

ph_alert 表

ph_alert 表包含关于数据库服务器生成的事件警报或调度程序生成的警报的信息。与内置任务和传感器关联的警报会自动添加到 ph_alert 表中。

表 112: ph_alert 表

列	类型	描述
id	serial	警报 ID。 由系统生成；不要修改。
alert_task_id	serial	任务或传感器 ID。 必须与 ph_task 表中的 tk_id 列的值相同。 事件警报的任务 ID 是 15。
alert_task_seq	integer	指示任务的哪个调用创建了警报。 由系统生成；不要修改。 引用 ph_task 表中的 tk_sequence 列。
alert_type	char(8)	警报类型： • INFO（默认） • WARNING

列	类型	描述
		• ERROR 警报或事件警报的严重性由警报类型和警报颜色的组合指示。请参阅表 113: 警报类型和颜色 在第249页。
alert_color	char(15)	警报的颜色： • GREEN（默认） • YELLOW • RED 警报或事件警报的严重性由警报类型和警报颜色的组合指示。请参阅表 113: 警报类型和颜色 在第249页。
alert_time	datetime year to second	警报生成的时间。 由系统更新；不要修改。
alert_state	char(15)	指示对象所处状态： NEW （默认）该警报是新添加的，并且对该警报尚未发生任何操作。 IGNORED DBA 已确认该警报且未执行任何操作。 ACKNOWLEDGED DBA 已确认该警报。 ADDRESSED DBA 已处理该警报。
alert_state_changed	datetime year to second	上次更改状态的时间。 由系统更新；不要修改。
alert_object_type	char(15)	警报对象的类型： • ALARM • CHUNK • DATABASE • DBSPACE • INDEX • MISC (Default) • SERVER • SQL_STATEMENT • TABLE • USER
alert_object_name	varchar(255)	警报对象的名称或事件警报类ID。
alert_message	lvarchar	警报或事件警报的详细信息。
alert_action_dbs	lvarchar(256)	修正操作所使用的数据库的名称。 默认为 sysadmin。
alert_action	lvarchar	修正操作。

列	类型	描述
		要调用以修正问题的 SQL 脚本。该脚本必须符合所有多语句准备规则。 如果没有可用的操作，可以是 NULL 值。
alert_object_info	bigint	对于 ALARM 类型的警报，其为事件警报的事件 ID。

下表定义了三种类型的消息和事件警报的警报颜色。

表 113: 警报类型和颜色

消息类型	绿色	黄色	红色
信息	指示组件的操作状态的状态消息。 严重性为 1（不值得注意）的事件警报。	重要的状态消息。 严重性为 2（信息）的事件警报。	需要操作的状态消息。
警告	来自数据库的警告。数据库已自动处理发生的状况。	需要处理的未来事件。 严重性为 3（注意）的事件警报。	即将发生预测的故障。需要立即执行操作。
错误	组件中的故障，由组件自行修正的。	组件中的故障，由组件自行修正的，但可能需要 DBA 的操作。	组件中的故障，需要 DBA 的操作。 严重性为 4（紧急）或 5（致命性）的事件警报。

ph_alerts 视图显示警报信息，以及相关联的任务或传感器信息。

相关链接

ph_alert 表中的事件 在第723页

ph_threshold 表

ph_threshold 表包含关于调度程序任务的阈值的信息。

ph_threshold 表包含与内置任务和传感器相关联的阈值。例如：名为 COMMAND HISTORY RETENTION 确定 command_history 表中的行的保留时间长度。

表 114: ph_threshold 表

列	类型	描述
id	integer	阈值 ID。
name	char	阈值名称。
task_name	varchar	与该阈值相关联的调度程序任务名称。 必须 ph_task 表中的 tk_name 列的值相同。
value	lvarchar	阈值的值。
value_type	char	该值列的数据类型： • STRING

列	类型	描述
		• NUMERIC • NUMERIC(p,s)
description	lvarchar	该阈值的描述。

结果表

结果表包含关于调度程序所运行的传感器的历史数据。

大部分传感器创建一个新的表来存储它们的结果。该表的名称列在 ph_task 表的 tk_result_table 列里。该表的结构是由 ph_task 表的 tk_create 列中的 CREATE TABLE 语句定义的。

当运行以 mon_ 为开头的内置传感器时，这些传感器会自动创建结果表。

表 115: 结果表

列	类型	描述
ID	integer	传感器的迭代序号。必须设置为 $DATA_SEQ_ID。 引用 ph_run 表的 run_task_seq 列。
user columns	any	可以指定任何类型的列来保存传感器返回的信息。

command_history 表

command_history 表包含在过去 30 天内运行的所有 SQL 管理 API 函数的列表和结果。

command_history 表显示每个已运行的 SQL 管理 API 函数，并显示关于执行的用户的信息、执行的时间、主参数，以及数据库服务器执行完成时返回的消息。

表 116: command_history 表

列	数据类型	描述
cmd_number	serial	每行的唯一 ID。
cmd_exec_time	datetime year-to-second	函数开始的时间。
cmd_user	varchar	执行的用户。
cmd_hostname	varchar	运行函数的主机的名称。
cmd_executed	varchar	所运行的函数的主参数。
cmd_ret_status	integer	返回代码。
cmd_ret_msg	lvarchar	返回消息。

下表显示 command_history 表中的示例参数和相关的结果消息。

表 117: command_history 表中的示例信息

参数 (cmd_executed)	返回消息 (cmd_ret_msg)
set sql tracing on	SQL tracing on with 1000 buffers of 2024 bytes.
create dbspace	Space 'space12' added.

参数 (cmd_executed)	返回消息 (cmd_ret_msg)
checkpoint	Checkpoint completed.
add log	Added 3 logical logs to dbspace logdbs.

要显示命令历史，请从 sysadmin 数据库运行以下 SQL 语句：

```
SELECT * FROM command_history;
```

command_history 表的大小

根据 SQL 管理 API 函数运行的数量，command_history 表可能变得相当大。可以通过更改 ph_threshold 表中 COMMAND HISTORY RETENTION 行的 value 字段，改变信息在 command_history 表中保留的时间。

还可以使用 DELETE 或 TRUNCATE TABLE 之类的 SQL 语句来手动地从该表移除数据。

相关链接

《SinoDB 管理员指南》：查看 SQL 管理 API 历史记录

storagepool 表

sysadmin 数据库中的 storagepool 表包含有关 SinoDB® 实例中存储池中所有条目的信息。每个条目表示服务器自动扩展存储空间时可使用的空闲空间。

表 118: storagepool 表

列	类型	描述
entry_id	SERIAL	存储池条目的 ID。
path	VARCHAR (255)	服务器在需要额外存储空间时可以使用的文件、目录或设备的路径。
beg_offset	BIGINT	设备上以 KB 为单位的初始偏移量，服务器从此处开始分配空间。 如果存储池信息是目录的，则结束偏移量是 0。
end_offset	BIGINT	设备上以 KB 为单位的初始偏移量，服务器从此处开始停止分配空间。 如果存储池信息是目录的，则结束偏移量是 0。
chunk_size	BIGINT	从此条目分配的块的初始大小。
status	VARCHAR (255)	存储池条目的状态。状态值为： Active = 功能存储池条目。服务器可以从此条目分配块。 Full = 该存储池条目没有任何空闲空间。服务器无法从此条目分配任何块。 Error = 当服务器尝试从该条目分配块时，存储池条目生成错误。
priority	INT	服务器在存储池中搜寻空间时目录、文件或设备的优先级。服务器在从较低优先级条目分配空间之前，会先从高优先级条目分配空间。

列	类型	描述
		1 = 高优先级 2 = 中优先级 3 = 低优先级
last_alloc	DATETIME (year to second)	从该条目最近一次分配的日期和时间。
logid	INT	上次使用此条目时当时的日志 ID。服务器在相同优先级条目之间进行选择时，将此标志与 logused 值一起使用。
logused	INT	上次使用此条目时当时在日志中的位置。服务器在相同优先级条目之间进行选择时，将此标志与 logid 值一起使用。

tenant 表

sysadmin 数据库中的 tenant 表包含有关租户数据库的信息。

表 119: tenant 表

列	Type	Description
tenant_id	int	租户数据库的唯一 ID
tenant_dbsname	varchar(128)	租户数据库的名称
tenant_resources	bson	租户数据库的属性和租户的状态。 将此列转换成 JSON 以查看信息。
tenant_last_updated	datetime year to second	租户数据库上次配置变更的时间戳。
tenant_comment	lvarchar(2048)	租户数据库的描述。

磁盘结构和存储

本章内容

数据库服务器通过管理自己的 I/O 获得其高性能。数据库服务器管理存储、搜索和检索。数据库服务器在存储数据时，创建之后它搜索和检索数据时需要的结构。数据库服务器磁盘结构还存储和跟踪管理日志记录和备份所需的控制信息。数据库服务器结构包含确保数据一致性（物理和逻辑）所需的所有信息。

在阅读本章之前，请先熟悉《*SinoDB*® 管理员指南》中的“数据存储”章节中的磁盘空间术语和定义。

本章讨论以下与磁盘数据结构相关的主题：

- 数据库空间结构和存储
- 简单大对象的存储
- 智能大对象空间结构
- 时间戳记
- 数据库和表的创建：磁盘上发生什么

数据库空间结构和存储

此节探究数据库服务器用于在数据库空间中存储数据的磁盘结构和存储技术。

相关链接

《*SinoDB* 性能指南》：森林树索引

根数据库空间的结构

作为磁盘空间初始化的一部分，数据库服务器初始化根数据库空间初始块中的特定结构。

初始化以下结构：

- 12 个保留页
- 第一个块可用列表页
- 表空间 tblspace
- 物理日志
- 逻辑日志文件
- 数据库表空间

ROOTNAME、ROOTOFFSET、ROOTPATH 和 ROOTSIZE 配置参数指定根数据库空间初始块的大小和位置。如果根数据库空间已镜像，MIRROROFFSET 和 MIRRORPATH 配置参数指定镜像块位置。有关这些参数的更多信息，请参阅数据库配置参数 在第41页。

要查看根块的结构，请使用 oncheck -pe 命令。有关更多信息，请参阅 *oncheck -ce, -pe:* 检查空闲块列表 在第301页。

保留页

根数据库空间初始块的前 12 页是保留页。每个保留页包含数据库服务器所使用的特定控制和跟踪信息。

要获取保留页内容的列表，请执行命令 oncheck -pr。要同时列出关于物理日志和逻辑日志页（包括活动物理日志页）的信息，请运行 oncheck -pR 命令。

以下示例显示了时间间隔检查点的示例 oncheck -pr 输出：

```
 Time of checkpoint      10/25/2005 17:05:20
 Checkpoint Interval    1234
```

数据库服务器还在名为 PAGE_CONFIG 的保留页中存储了当前配置信息。如果您从命令行变更配置参数并运行 oncheck -pr 命令而不关闭并重启数据库服务器，则该命令输出中的配置值与保留页中的当前值不匹配。oncheck 实用程序返回警告消息。

以下示例显示了 PAGE_CONFIG 保留页内容的样本输出。

```
...

Validating Sinoregal DS database server reserved pages - PAGE_CONFIG
    ROOTNAME                      rootdbs
    ROOTPATH                      /home/dyn_srv/root_chunk
    ROOTOFFSET                    4
    ROOTSIZE                      8000
    MIRROR                        0
    MIRRORPATH
    MIRROROFFSET                  0
    PHYSFILE                      1000
    LOGFILES                      5
    LOGSIZE                       500
    MSGPATH                       /home/dyn_srv/online.log
    CONSOLE                           /dev/ttyp5
...                               ...
```

常规数据库空间的结构

磁盘空间初始化后，可添加新的数据库空间。创建数据库空间时，至少向数据库空间指定一个块（原始或熟磁盘空间）。这个块称为数据库空间的初始块。图 *10:* 常规数据库空间初始块 在第254页说明了常规（非根）数据库空间初始块的结构。

首次创建数据库空间时，它包含以下结构：

- 2 个保留页
- 块中的第一个块可用列表页
- 该数据库空间的表空间 tblspace
- 未使用页

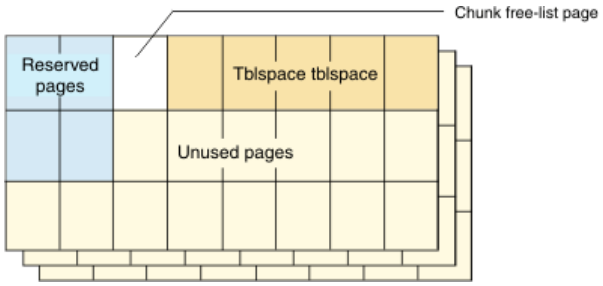


图 10: 常规数据库空间初始块

附加数据库空间块的结构

可以创建包含多于一个块的数据库空间。数据库空间的初始块包含了该数据库空间的表空间 tblspace 。附加块则不包含。首次创建附加块时，它包含以下结构：

- 2 个保留页
- 第一个块可用列表页
- 未使用页

图 *11:* 附加数据库空间块 在第254页说明了数据库空间中所有附加块的结构。（该结构也适用于根数据库空间中的附加块。）

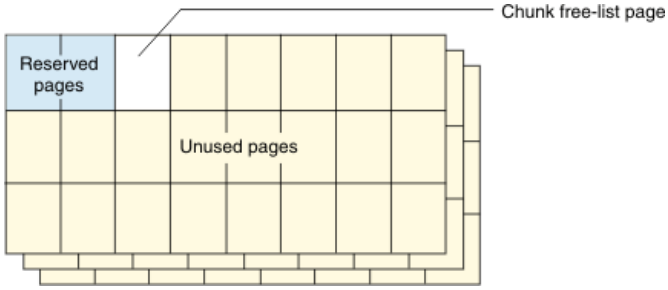


图 11: 附加数据库空间块

镜像块的结构

每个镜像块的大小都必须与其主块的大小相同。创建镜像块时，数据库服务器立即将主块的内容写入镜像块。

镜像块包含与主块相同的控制结构。在数据库服务器将 BLOB 空间、智能大对象空间或数据库空间块的镜像联机后，它们包含与其主对应对象相同的物理内容。

图 *12:* 镜像块结构 在第255页说明了块创建之后所显示的镜像块结构。

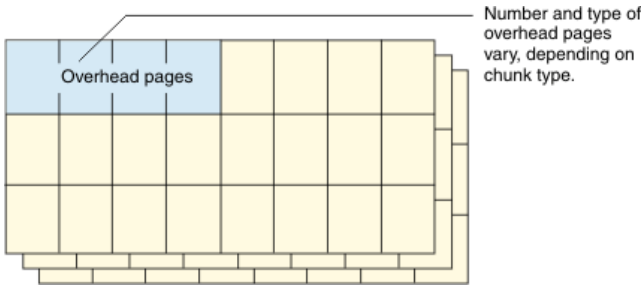


图 12: 镜像块结构

由于镜像块的所有空间都保留用于镜像，所以镜像块结构总是显示没有可用空间。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“镜像”章节。

块可用列表页的结构

在每个块中，最后一个保留页之后的页是跟踪块中可用空间的块可用列表页的第一页。对于非根块，可用空间的初始长度等于该块大小减去 3 页。如果需要附加块可用列表页以容纳新的条目，则会在该块中的一个可用页中创建新的块可用列表页。图 *13:* 可用列表页 在第255页说明了可用列表页的位置。

使用 oncheck -pe 来获取块中页的物理布局。有关更多信息，请参阅 *oncheck -ce, -pe:* 检查空闲块列表 在第301页。

图 13: 可用列表页

表空间 Tblspace 的结构

每个数据库空间包含一个名为 *tblspace* 的表空间，该 ***tblspace*** 描述数据库空间中的所有表空间。当数据库服务器创建一个表空间时，它会在表空间 tblspace 中放置一个条目，该条目描述新创建的表空间的特征。您不能删除或移动包含表空间 tblspace 的块。

一个数据库空间最多可以有 2**20 个表空间。

第一个和下一个扩展数据块的默认大小取决于数据库空间是否为根数据库空间，如下表所示。

表 120: 每个扩展数据块的默认大小和数据库空间类型

数据库空间类型	第一个扩展数据块的默认大小	下一个扩展数据块的默认大小
根	• 500 KB（2 KB 页的系统） • 1000 KB（4 KB 页的系统）	• 100 KB（2 KB 页的系统） • 200 KB（4 KB 页的系统）
非根	• 100 KB（2 KB 页的系统） • 200 KB（4 KB 页的系统）	• 100 KB（2 KB 页的系统） • 200 KB（4 KB 页的系统）

可以通过以下方式为表空间 tblspace 的第一个和下一个扩展数据块指定非默认的大小：

- 对于根数据库空间，设置 TBLTBLFIRST 和 TBLTBLNEXT 配置参数。
- 对于非根数据库空间，在创建数据库空间时使用 onspaces 实用程序的 -ef 和 -en 选项。

表空间 tblspace 条目
表空间 tblspace 描述表空间特性。

要在表空间上显示信息，请使用 oncheck -pt 命令。

表 121: 表空间 tblspace 条目

组件	描述
页头	24 字节，标准页头信息
页结束时间戳记	4 字节
表空间头	136 字节，一般表空间信息
表空间名称	*database.owner.tablename* 或 *database.owner.indexname* 通常 30-40 字节，但可以更长，取决于名称长度。
列信息	每个特殊列为 8 字节 特殊列定义为 VARCHAR、BYTE、TEXT 或用户自定义数据类型。
索引信息	对于附加索引，分区中的每个索引都有一个 20 字节的头，该头包含关于索引的一般信息，后跟随索引中每个列的 4 字节条目。 对于拆离的索引，索引中每个列为 4 字节条目。
扩展数据块信息	对于分配给表空间的每个扩展数据块有一个 10 字节条目加上 10 字节信息。 在表空间碎片整理过程中，可能使用更多的字节。

相关链接

oncheck -pt 和 -pT: 显示表或分段的表空间 在第308页

表空间编号

表空间 tblspace 中描述的每个表空间有一个表空间编号。该表空间编号与存储在 systables 系统目录表中的 partnum 字段和 sysfragments 系统目录表中的 partn 字段相同。

以下 SQL 查询检索数据库中每个表的 partnum（它们可位于几个不同的数据库空间），并以表名称和 partnum 的十六进制表示式显示它：

```
SELECT tabname, tabid, partnum, HEX(partnum) hex_tblspace_name FROM systables
```

如果输出包括具有表名称的行但 partnum 为 0，那么该表由两个或两个以上的表分段组成，每个分段位于自己的表空间中。例如：图 *14: 带有 partnum 值的 systables 查询的输出* 在第256页显示名为 account 但 partnum 为 0 的表。

```
tabname         tabid   partnum          hex_tblspace_name

sysfragments     25     1048611          0x00100023
branch          100     1048612          0x00100024
teller          101     1048613          0x00100025
account         102           0          0x00000000
history         103     1048615          0x00100027
results         104     1048616          0x00100028
```

图 14: 带有 partnum 值的 systables 查询的输出

要获取组成表的每一分段的实际表空间编号，您必须查询同一数据库的 sysfragments 表。图 *15: 带有 partn 值的 sysfragments 表的输出* 在第257页 显示了图 *14: 带有 partnum 值的 systables 查询的输出* 在第256页 中的 account 表有三个表分段和三个索引分段。

```
tabid  fragtype     partn          hex_tblspace_name

  102  T            1048614        0x00100026
  102  T            2097154        0x00200002
  102  T            3145730        0x00300002
  102  I            1048617        0x00100029
  102  I            2097155        0x00200003
  102  I            3145731        0x00300003
```

图 15: 带有 partn 值的 sysfragments 表的输出

表空间编号元素

表空间中的第一页是逻辑页 0。（物理页编号引用块中页的地址）根空间中的表空间 tblspace 总是包含在第一个数据库空间中，并且在表空间 tblspace 中的逻辑页 1 上。（位图页是页 0。）

表空间 Tblspace 大小

当初始化数据库空间时，这些表空间 tblspace 页分配为一个扩展数据块。如果数据库服务器尝试创建一个表，但是表空间 tblspace 已满，则数据库服务器会分配下一个扩展数据块给该表空间。

从数据库空间移除一个表时，也会删除该表在表空间 tblspace 中的相应条目。

表空间 Tblspace 位图页

表空间 tblspace 的第一页与任何初始块的第一页一样，是描述后续页的充满度的位图。后续的每一页在位图页上都有一个条目。如果需要，附加位图页位于分配给表空间的整个相邻空间中，这样排列使得每个位图仅描述后跟的页，直到下一个位图或数据库空间的结束。位图页在表空间页内以不同的间隔出现。每个位图页描述了后跟它的固定页数。

数据库表空间的结构

数据库表空间仅出现在根数据库空间的初始块中。对于数据库服务器所管理的每个数据库，数据库表空间都包含一个条目。图 *16: 数据库表空间在根数据库空间中的位置* 在第257页 说明了数据库表空间的位置。

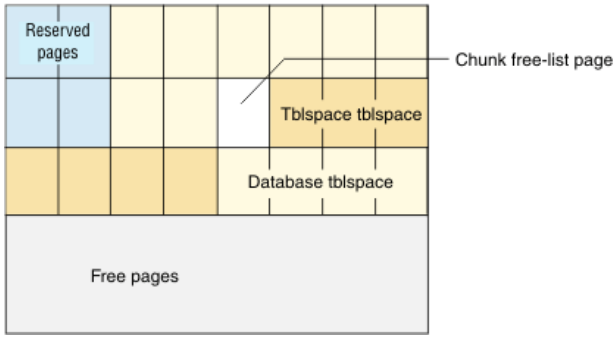


图 16: 数据库表空间在根数据库空间中的位置

数据库表空间编号

数据库表空间的表空间编号始终是 0x100002。如果数据库表空间是活动的，则该表空间编号出现在 onstat -t 列表中。

数据库表空间条目

每个数据库表空间条目包括以下 5 个组成部分：

- 数据库名
- 数据库所有者
- 数据库的创建日期和时间
- 该数据库的 systables 系统目录表的表空间编号
- 标识日志记录方式的标志

数据库表空间包含在数据库名上的唯一索引，以确保每个数据库的名称是唯一的。对于任何数据库，systables 表描述数据库中的每个永久表。因此，数据库表空间只指向位于任何地方的详细数据库信息。

在初始根数据库空间时，分配数据库表空间的缔约国扩展数据块。数据库表空间的初始扩展数据块大小和下一个扩展数据块大小为 4 页。不可修改这些值。

扩展数据块的结构和分配

本节涵盖以下主题：

- 扩展数据块结构
- 下一个扩展数据块分配

扩展数据块结构

扩展数据块是数据库空间内相邻页的集合。每个永久数据库表有两个扩展数据块大小与其关联。初始扩展数据块大小是创建表时分配给表的 KB 数。下一个扩展数据块大小是当初始扩展数据块和之后的每一扩展数据块变满时分配给表的 KB 数。

BLOB 空间不使用扩展数据块。

有关如何指定和计算扩展数据块的大小的特定信息，请参阅《*SinoDB*® 性能指南》。

扩展数据块大小
第一个和下一个扩展数据块的默认大小为 16 KB。如果在特定数据库空间中转换为少于 4 页，那么数据库服务器使用最小扩展数据块大小为 4 页。如果数据库空间的大小为 8 KB（转换为 2 页），那么数据库服务器将扩展数据块大小增加到 32 KB。

扩展数据块的最大大小是 2**31 页，相当于最大块大小。

如果块小于最大大小，那么最大扩展数据块大小取决于块中可用的相邻空间。

保存索引分段的表空间对于扩展数据块大小遵循不同的规则。数据库服务器根据相应的表分段的扩展数据块大小来决定这些表空间的扩展数据块大小。数据库服务器使用行大小对索引键大小的比率来为索引表空间指定适当的扩展数据块大小（请参阅《*SinoDB*® 性能指南》中的估算索引页大小和分段表索引的部分）。

分区的最大扩展数据块数为 32767。

表扩展数据块中的页类型

扩展数据块内，个别页包含不同类型的数据。表的扩展数据块页可分成以下类别：

- 数据页

  数据页包含表的数据行。
- 位图页

  位图页包含监视扩展数据块中每页充满度的控制信息。
- BLOB 页

  BLOB 页包含与数据行一起存储在数据库空间中的 TEXT 和 BYTE 数据。存在于 BLOB 空间中的 TEXT 和 BYTE 数据存储在 BLOB 页（一种与数据库空间 BLOB 页结构完全不同的结构）
- 可用页

可用页为扩展数据块中分配给表空间使用的页，但其功能还未定义。可用页可用于存储任何类型的信息：数据（包含 TEXT 或 BYTE 数据类型）、索引或位图。

图 17: 表的扩展数据块结构 在第259页 说明了非分段的表（其初始块大小为 8 页和下一个扩展数据块大小为 16 页）的可能结构。

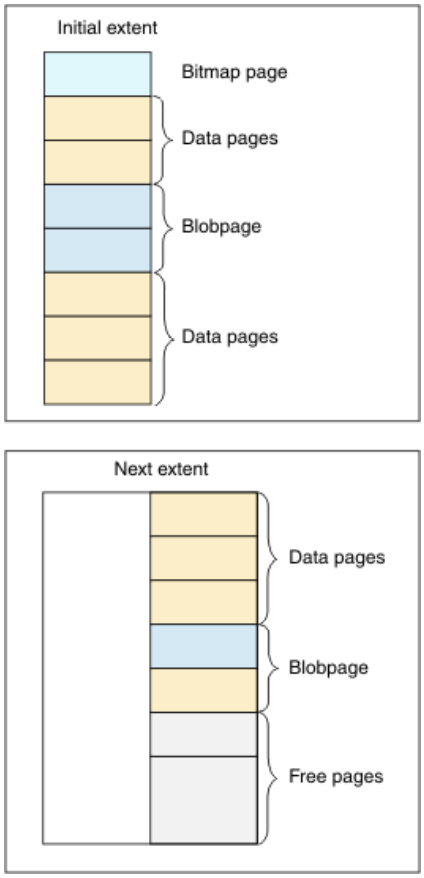


图 17: 表的扩展数据块结构

索引扩展数据块中的页类型

数据库服务器将索引页存储到与相关联的表不同的表空间中。扩展数据块内，个别索引页包含不同类型的数据。索引业可分成以下类别：

- 索引页（根、分支和叶页）

  索引页包含表的索引信息。
- 位图页

  位图页包含监视扩展数据块中每页充满度的控制信息。
- 可用页

  可用页是扩展数据块中分配给表空间使用的页，但其功能还未定义。可用页可用于存储任何类型的信息：数据、索引、TEXT 或 BYTE 数据，或者位图。

除非明显地指定了附加索引，否则所有索引都是拆离的。

> 重要: 分配给表分段的扩展数据块不包含索引页。分段表的索引页总是存在单独的表空间。有关更多信息，请参阅《*SinoDB*® 管理员指南》中表分段和 PDQ 章中的分段表索引。

图 18: 索引的扩展数据块结构 在第260页 说明了索引的扩展数据块结构。

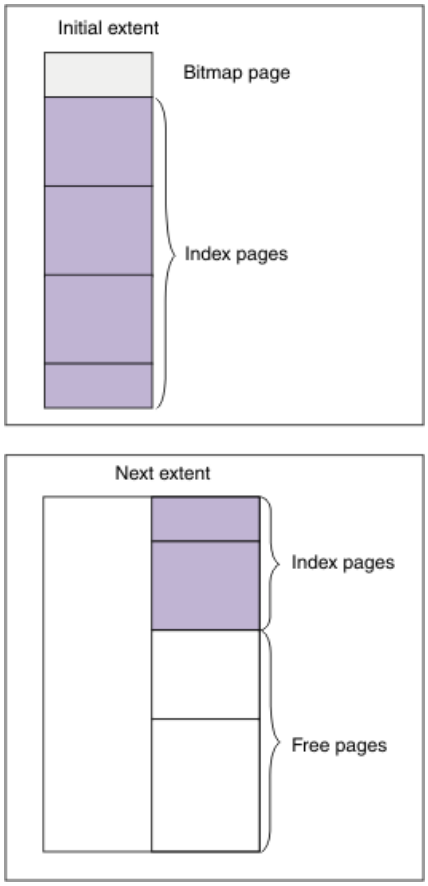


图 18: 索引的扩展数据块结构

下一扩展数据块分配

初始扩展数据块填满后，数据库服务器尝试分配相邻磁盘空间的另一扩展数据块。数据库服务器所遵循的过程称为下一扩展数据块分配。

表空间的扩展数据块作为表的表空间 tblspace 信息的一个组成进行跟踪。分配给任何表空间的最大扩展数据块数依赖于应用程序和机器，因为它随表空间 tblspace 条目上可用的空间量而变化。

下一扩展数据块大小

数据库服务器分配给下一扩展数据块的 KB 数，一般等于 SQL 语句 CREATE TABLE 所指定的下一扩展数据块的大小。然而，下一扩展数据块分配的实际大小可能与指定的大小有差异，因为分配过程考虑以下三个因素：

- 该表空间的现有扩展数据块的数目
- 块和数据库空间中相邻空间的可用性
- 现有表空间扩展数据块的位置

在以下段落和图 *19:* 下一扩展数据块分配策略 在第261页中说明这些因素中的每一个对下一扩展数据块分配的影响。

扩展数据块大小加倍

对于永久表或用户定义的临时表，每次分配的下一扩展数据块大小是自动地加倍。大小加倍最高可达 128 KB。例如：如果创建 NEXT SIZE 等于 15 KB 的表，则数据库服务器将第一个扩展数据块分配成 15 KB。下一扩展数据块分配成 30 KB，且再下一个扩展数据块分配成 60 KB。当扩展数据块大小达到 128 KB 时，则仅当表中剩余空间小于表中分配的总空间的 10% 时才会加倍大小。

对于系统创建的临时表，下一扩展数据块大小在添加了 4 个扩展数据块后开始加倍。

缺少相邻空间

如果数据库服务器无法在第一个块中找到等于指定的下一扩展数据块大小的可用的相邻空间，那么它将搜索延伸到数据库空间中的下一块。扩展数据块是不允许跨块的。

如果数据库服务器无法在数据库空间的任何地方找到足够的相邻空间，那么它将最大可用的相邻空间分配给表。（最小分配是 4 页。默认值是 8 页。）如果分配是可能的，那么即使所分配的量小于所请求的量，也不会返回任何错误消息。

同一表的扩展数据块的合并

如果分配给下一扩展数据块的磁盘空间与分配给同一表的磁盘空间是物理相邻的，那么数据库服务器分配磁盘空间但不将新分配视为单独的扩展数据块。相反地，数据库服务器是延伸现有相邻扩展数据块的大小。之后，所有磁盘空间报告将反映该分配为现有扩展数据块的的扩展。也就是说，所报告的扩展数据块的数目始终是物理上不同的扩展数据块的数目，而不是已分配下一扩展数据块的次数加 1（初始扩展数据块）。图 19: 下一扩展数据块分配策略 在第261页 说明扩展数据块分配策略。

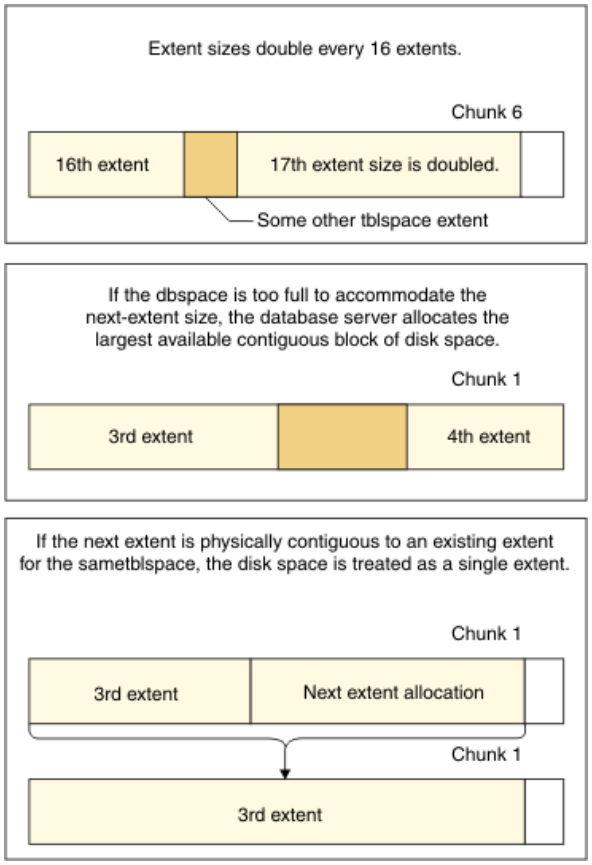


图 19: 下一扩展数据块分配策略

在磁盘空间作为扩展数据块的一部分分配给表空间之后，该空间保持专用于该表空间，即使其中包含的数据已删除也是如此。有关回收该空磁盘空间的替代方法，请参阅《*SinoDB®* 性能指南》。

数据库空间页的结构和存储

数据库服务器 I/O 的基本单位是页。页大小可因计算机而异。

在 SinoDB® 中，页大小取决于操作系统。

非分段表中的行

数据库服务器可以存储长于一页的行。数据库服务器还支持 VARCHAR 数据类型，这会导致可变长度的行。因此，行并不遵循单一格式。

如果表包含一个或多个的类型 VARCHAR 的列，那么表中的行的长度不一定相同。此外，当最终用户修改 VARCHAR 列中的数据时，行的长度可能会改变。

行的长度可能会大于一页。

TEXT 和 BYTE 数据并不存储在数据行中。而是，数据行包含指向数据位置的 56 字节的描述符。该描述符可指向一个数据库空间页。

描述符可指向 BLOB 空间的 BLOB 页。

有关如何估计固定长度和可变长度的数据行的长度的指示信息，请参阅《*SinoDB®* 性能指南》。

Rowid 的定义

SinoDB® 使用两种不同的 rowid 类型来标识表中的数据：

- 顺序 *rowid*

  这些 rowid 是表中的字段，并且指定给使用 WITH ROWID 选项所创建的表。
- 内部 *rowid*

  数据库服务器使用唯一的内部 rowid 来标识表中的每一数据行。该 rowid 标识行在数据库空间内的位置。

  要获取表的内部 rowid，请使用 oncheck -pD 选项。有关更多的信息，请参阅 *oncheck -cd* 和 *oncheck -cD* 命令: 检查页 在第300页。

在非分段表中，术语 *rowid* 是指定义行在表中的物理位置的唯一 4 字节整数。包含数据行第一字节的页是由 rowid 所指定的页。该页称为数据行的主页。

分段表也可有 rowid，但它们是以另一种方式实现的。有关该主题的更多信息，请参阅分段表中的行 在第262页。

Rowid 的使用

非分段表中的每个数据行都由不变的 rowid 唯一标识。当创建非分段表的索引时，rowid 存储在与数据行所属的表相关联的索引页中。当数据库服务器要求一个数据行时，它搜寻索引以查找键值，并使用相应的 rowid 来定位所请求的行。如果该表未经索引，那么数据库服务器可能顺序地读取该表中的所有行。

最后，行可能增长到超出原始的存储位置。如果发生了这样的情况，那么指向数据行新位置的转发指针将留在 rowid 所定义的位置。转发指针本身就是 rowid，其定义了数据行所存储的页和页上的位置。

分段表中的行

与非分段表中的行不同，数据库服务器不向分段表中的行指定 rowid。如果您要使用 rowid 访问数据，那么您必须显示创建 rowid 列，如《*SinoDB®* 性能指南》所描述的那样。如果用户应用程序尝试引用未包含显示创建的 rowid 的分段表的 rowid，那么数据库服务器向应用程序返回相应的错误代码。

使用 rowid 访问分段表中的数据

从应用程序的角度来看，分段表中 rowid 列的功能与非分段表的 rowid 是一样的。然而，与非分段表的 rowid 不同的是数据库服务器使用索引将 rowid 映射到物理位置。

当数据库服务器使用 rowid 列访问分段表中的行时，它在尝试访问该行前使用该索引查找该行的物理地址。对于非分段表，数据库服务器不使用索引查找而使用直接物理访问。结果，使用 rowid 访问分段表中的行所花时间比使用 rowid 访问非分段表中的行所花时间稍长。由于维护分段表的 rowid 索引的成本，您还应期望对插入和删除的处理产生小的性能影响。

在许多情况下，主键访问可显著提升性能，尤其在并行访问时。

Rowid 的使用建议

建议应用程序开发者使用主键而不是 rowid 作为访问方式。由于主键是在 SQL 的 ANSI 规范中定义的，所以使用它们访问数据可使应用程序的可移植性更好。

有关如何定义与使用主键访问数据的完整描述，请参阅《*SinoDB® SQL* 指南: 参考》和《*SinoDB® SQL* 指南: 教程》。

数据行格式与存储

数据行的可变长度对于行存储产生以下结果：

- 一页可能包含一个或多个整行。
- 一页可能包含一行或多行的一部分。
- 一页可能包含整行与行的一部分的组合。
- 行的大小可能在更新后增大，并变得太长而无法返回到行中原本的存储位置。

以下段落描述数据存储过程中数据库服务器所遵循的准则。

行的存储

为使检索时间最小，除非必要，否则不要跨页边界断行。短于一页的行总是以整行进行存储。当页中的可用字节数少于存储最大大小的行所需的字节数，则该页视为已满。

行的位置

当数据库服务器接收长于一页的行时，该行将存储在多个整页中。然后，数据库服务器以小于一个满页的长度存储拖尾部分。

包含该行的第一个字节的页是该行的主页。主页的编号成为包含在 rowid 中的逻辑页号。主页后跟的每个满页称为大剩余页。如果行的拖尾部分小于一个满页，那么它将存储在剩余页上。

在数据库服务器创建剩余页以容纳长行之后，它可使用该页中的剩余空间来存储其他行。

图 *20: 剩余页* 在第263页说明了主页、大剩余页和剩余页的概念。

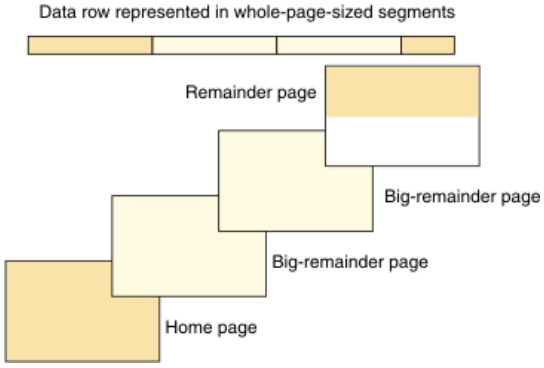


图 20: 剩余页

页压缩

随着时间流逝，页上的可用空间可能变得分段的。当数据库服务器尝试存储数据时，它首先对照页上的可用字节数检查行长度，以确定该行是否适合。如果有足够可用空间，那么数据库服务器检查是否该页包含足够相邻可用空间以保存该行（或行的一部分）。如果可用空间不是连续的，那么数据库服务器调用页压缩。

分段表的结构

尽管表分段对于应用程序是透明的，但作为数据库服务器管理员，您应知道数据库服务器是如何将磁盘空间分配用于表分段以及数据库服务器是如何标识那些分段中的行。

每个表分段都自己的具有唯一 *tblspace_id* 或 *fragment_id* 的表空间。图 *21: 分段表的磁盘结构* 在第264页显示了位于相同数据库空间已命名分段中的分段表的磁盘分配。

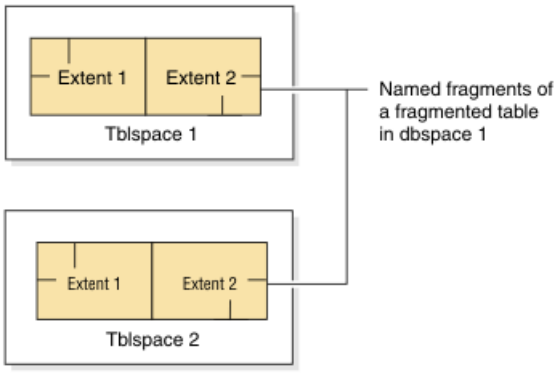


图 21: 分段表的磁盘结构

附加索引

如果使用附加索引，索引和数据的分段方式是一样的。您可以决定是否将索引页和相应的数据页存储在相同的数据库空间，还是将它们存储在不同的数据库空间。有关选择分段存储策略的信息，请参阅《*SinoDB*® 性能指南》。

拆离的索引

对于拆离的索引，表分段和索引分段是存储在不同数据库空间的表空间中。

B 型树索引页的结构

本节提供有关 B 型树索引页的结构的一般信息。它是为对此结构感兴趣的读者编写的概述。有关 B 型树索引的更多信息，请参阅《*SinoDB*® 性能指南》。

相关链接

FILLFACTOR 配置参数 在第114页

B 型树术语的定义

数据库服务器使用 B 型树结构来组织索引信心。

图 *22: 完整 B 型树结构* 在第265页显示由以下三种不同类型的索引页或节点组成的完全发展的 B 型树索引：

- 一个根节点

  根节点包含指向分支节点的节点指针。
- 两个或更多个分支节点

  分支节点包含指向叶节点或其他分支节点的指针。
- 多个叶节点

  叶节点包含索引项和指向其他叶节点的水平指针。

每个节点有不同功能。以下各节描述每个节点及其在索引建立中所扮演的角色。

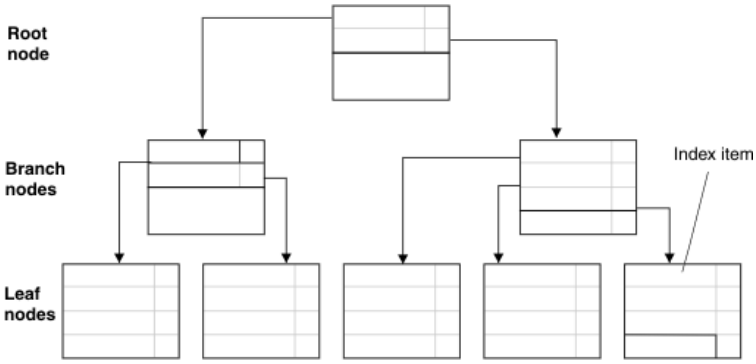


图 22: 完整 B 型树结构

索引项

索引的基本单位是 索引项。索引项包含代表特定行的索引列值的键值。索引项还包含数据库服务器用于在数据页中定位行的 rowid 信息。

索引节点

节点是存储一组索引项的索引页。

树林索引作为传统 B 型树索引的替代方式

与传统 B 型树索引不同，树林索引是一个大的 B 型树索引，它被分成具有多个根节点和较少层级的小子树。当想要减少根节点争用并允许更多并发用户无需等待就可访问索引时，您可以创建树林索引作为 B 型树的替代方式。

相关链接

《*SinoDB* 性能指南》：森林树索引

索引的逻辑存储

本节提供数据库服务器如何创建与填充索引的概述。

根节点与叶节点的创建

创建空表的索引时，数据库服务器分配单个索引页。该页代表根节点，并在插入数据到表中之前保持为空。

起初，根节点的功能与叶节点相同。对于插入表中的每一行，数据库服务器在根节点中创建并插入索引项。图 *23:* 根节点 在第265页说明了根节点在填充之前的样子。

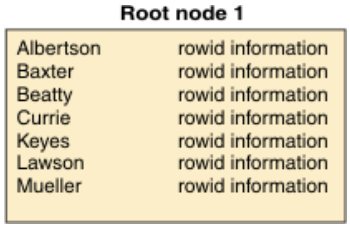


图 23: 根节点

当根节点填满索引项时，数据库服务器通过执行以下步骤分割根节点：

- 创建两个叶节点
- 将大约一半的根节点条目移动到每个新创建的叶叶节点中
- 在根节点中放入指向叶节点的指针

当向表添加新的行时，数据库服务器向叶节点添加索引项。当叶节点填满时，数据库服务器创建新的叶节点，将已满的索引节点的部分内容移动到新节点，并在根节点添加指向新叶节点的节点指针。

例如：假设图 *24: 在叶节点 3 填满后创建叶节点 4* 在第266页的叶节点 3 已满。当这种情况发生时，数据库服务器添加另一叶节点。数据库服务器将部分记录从叶节点 3 搬移到新的叶节点，如图 *24: 在叶节点 3 填满后创建叶节点 4* 在第266页所示。

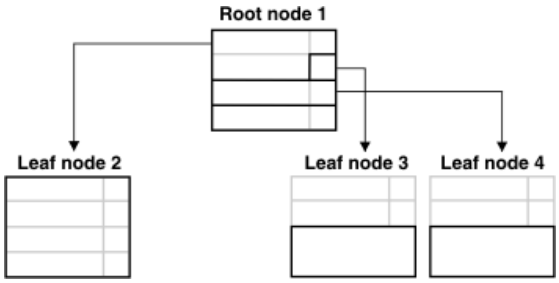


图 24: 在叶节点 3 填满后创建叶节点 4

分支节点的创建

最终，当向表添加行时，数据库服务器一指向所有现有叶节点的节点指针填充根节点。当数据库服务器分割另一叶节点，而根节点已没有空间可用于额外节点指针时，那么发生以下过程。

数据库服务器分割根节点，并将其内容分到两个新创建的分支节点上。随着索引项的添加，越来越多的叶节点被分割，这导致数据库服务器添加更多分支节点。最终，根节点充满指向这些分支节点的指针。当这种情况发生时，数据库服务器再次分割根节点。然后，数据库服务器在根节点和较低的分支级之间创建另一分支级别。这个过程生成一个 4 级树，其带有一个根节点，两个分支级，和一个叶级。B 型树结构可以继续以这个方式增长到最大 20 层。

分支节点可以指向其下面的其他分支节点（对于 4 级或更多级的大索引来说）或叶节点。在图 *25: 分支节点的典型内容* 在第266页，分支节点仅指向叶节点。左分支节点的第一项包含与最左叶节点中最大项相同的键值，以及指向它的节点指针。第二项包含下一叶节点中的最大项和指向它的节点指针。分支节点中的第三项仅包含指向下一较高叶节点的指针。根据索引的增长，在索引的生命周期中的之后的某个时间点，除了指针外，此第三项可能包含实际键值。

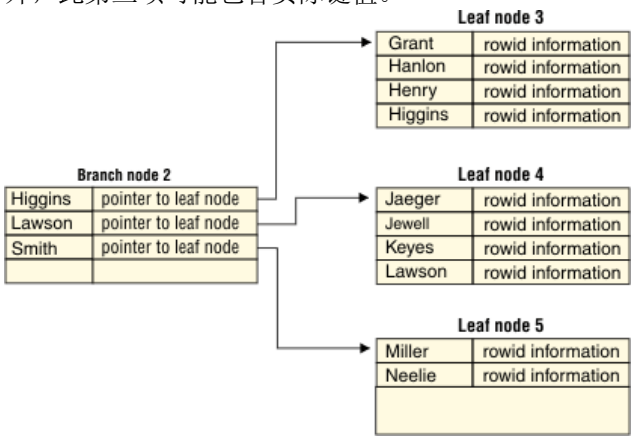


图 25: 分支节点的典型内容

重复的键值

当多行的索引列的值完全相同时，发生重复键值。例如：假设 B 型树结构的第三和第四叶节点包含键值 Smith。进一步假设该值重复了 6 次，如图 *26: 叶节点 3 和 4* 在第267页显示的.

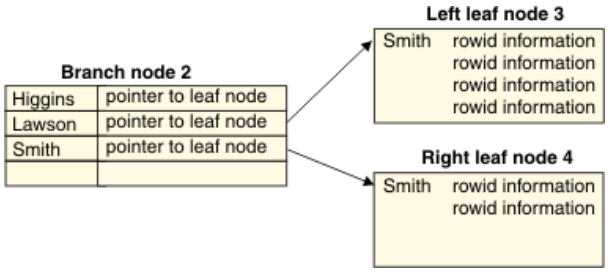


图 26: 叶节点 3 和 4

第三叶页的第一项包含重复键值 Smith 和包含重复键值的表中的第一物理行的 rowid 信息。为节省空间，第二项不重复键值 Smith 但包含 rowid 信息。这个过程在整个页上继续；叶上不存在其他键值，只有 rowid 信息。

第四叶页的第一项也包含重复键值和 rowid 信息。后续项只包含 rowid 信息。

现在考虑分支节点。分支节点中的第三项包含与第三叶节点中最大项相同的键值和 rowid，以及指向它的节点指针。第四项只包含指向第四叶节点的节点指针，因此节省了附加重复键值的空间。

键值锁定

为增加并发性，数据库服务器支持 B 型树索引中的键值锁定。键值锁定只锁定键值而非 B 型树索引中的物理位置。

键值锁定的一个最重要用途是确保唯一键直到删除它的事务结束时都保持唯一。如果没有这个保护机制，那么用户 A 可能在事务中删除唯一键，用户 B 可能在该事务提交前插入具有相同键的行。该场景使得用户 A 不可能回滚。键值锁定可防止用户 B 在用户 A 的事务结束之前插入行。

相邻键锁定

Repeatable Read 隔离级别要求数据库服务器保护读取集。读取集由符合查询的 WHERE 字句中的过滤条件的行组成。为保证这些行不改变，数据库服务器获取与读取集最右项相邻的索引项上的锁。

释放的索引页

当数据库服务器从节点中物理删除索引项并释放索引页，那么释放的页是可再用的。

填充索引

创建索引时，您可指定索引填充的密度或稀疏度。索引填充因子是索引建构过程中填充每个索引页的百分比。使用 CREATE INDEX 语句的 FILLFACTOR 选项或 FILLFACTOR 配置参数来设置填充因子。该选项对于不预期在建构后会增长的索引特别有用。有关 CREATE INDEX 语句的 FILLFACTOR 选项的其他信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。

计算索引项的长度

对于非 VARCHAR 的数据类型，索引项长度的计算方式是：键值长度加上与键值相关联的每个 rowid 信息的 5 个字节。

索引中的键值通常都是固定长度。如果索引保存 VARCHAR 数据类型的一列或多列的值，那么键值的长度至少是键中每一 VARCHAR 值的长度加一的总合。

在 SinoDB® 中，键值的最大长度是 390 字节。构成键的 VARCHAR 列的组合大小必须小于 390 减去每个 VARCHAR 列的一个附加字节。例如：数据库服务器为以下语句建构的索引键的长度等于 390 或 ((255+1) + (133+1))：

```
CREATE TABLE T1 (c1 varchar(255, 10), c2 varchar(133, 10));
CREATE INDEX I1 on T1(c1, c2);
```

函数的索引

函数的索引是所有键都从一个函数的结果得到。例如：如果您有一列图片和一个用于识别主要颜色的函数，那么您可以在函数的结果上创建索引。这样的索引使您能够迅速地检索所有具有相同主要颜色的图片，而不需要重新执行该函数。

函数的索引使用与其他 B 型树索引相同的 B 型树结构。唯一的区别是，在插入或更新过程中只要作为函数参数的列变更，就会应用该确定函数。有关函数的索引的特性的更多信息，请参阅《*SinoDB®* 性能指南》。

要创建函数的索引，请使用 CREATE FUNCTION 和 CREATE INDEX 语句。有关这些语句的更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。

R 型树索引页的结构

依赖于键值的一维排序的索引结构，不适用于空间数据；例如：二维几何形状，诸如圆形、正方形和三角形。对于诸如地理信息系统（GIS）和计算机辅助设计（CAD）等的空间数据的有效检索，需要一种处理多维度数据的存取方式。数据库服务器实现 R 型树索引以有效地存取空间数据。有关索引页的结构的信息，请参阅《*SinoDB® R-Tree* 索引用户指南》。

简单大对象的存储

本节说明数据库服务器用于存储简单大对象（TEXT 或 BYTE 数据）所使用的结构和存储技术。

BLOB 空间的结构

创建 BLOB 空间时，可以指定数据页（这称为 BLOB 页）的有效大小。BLOB 空间的 BLOB 页大小在创建 BLOB 空间时指定。BLOB 页大小必须是页大小的倍数。（有关决定数据库服务器页大小的信息，请参阅《*SinoDB®* 管理员指南》中管理磁盘空间的那一章。）BLOB 空间中的所有 BLOB 页大小是相同的，但 BLOB 页大小在 BLOB 空间之间是可变化的。由于 BLOB 页大小可能大于页大小，因此，存储在 BOLB 空间的数据从不写入共享内存中大小为页的缓冲区。

定制 BLOB 页大小的好处是存储效率。在 BLOB 空间内，TEXT 和 BYTE 数据是存储在一个或多个 BOLB 页，但简单大对象不共享 BLOB 页。当 TEXT 或 BYTE 数据等于或略小于 BLOB 页大小时，存储是最有效率的。

BLOB 空间自由图页和位图页被指定作为数据库服务器页的大小，这使它们能够被读入共享内存和被记录下来。

首次创建 BOLB 空间时，它包含以下结构：

- BLOB 空间自由图页
- 用于跟踪自由图页的 BLOB 空间位图
- 未使用的 BLOB 页

数据库空间 BLOB 页的结构

存储在数据库空间中的 TEXT 或 BYTE 数据是存储在 BLOB 页中的。数据库空间 BLOB 页的结构与数据库空间数据页的结构类似。唯一区别是随着 TEXT 或 BYTE 数据一起存储在数据区域中的额外 12 字节。

如果多个简单大对象或简单大对象的多个拖尾部分可以放入单个页中，那么简单大对象可以共享数据库空间 BLOB 页。

有关如何估计特定表所需的数据库空间 BLOB 页的数目的讨论，请参阅《*SinoDB®* 性能指南》。

存储在数据库空间页中的每段 TEXT 或 BYTE 数据前面可能有多达 12 字节的信息，这些信息不出现在任何其他数据库空间页上。这些额外字节是开销。

简单大对象存储和描述符

包含 TEXT 或 BYTE 数据的数据行在行本身中不包含数据。而是包含一个有转发指针（rowid）的 56 字节的描述符，其指向数据第一个分段的存储位置。

描述符可以指向以下项之一：

- 页（如果数据存储在数据库空间）
- BLOB 页（如果数据存储在 BLOB 空间）

简单大对象的创建

当插入包含 TEXT 或 BYTE 数据的行时，首先创建简单大对象。简单大对象写入磁盘后，使用描述符更新行，并插入行。

简单大对象的删除或插入

数据库服务器不可修改简单大对象。只可插入或删除它们。删除简单大对象表示数据库服务器释放已删除对象所耗用的空间以再用。

当更新 TEXT 或 BYTE 数据时，创建新的简单大对象，并以新的 BLOB 描述符更新数据行。行的旧映像包含指向简单大对象的废旧值的描述符。在提交更新后，将释放废旧简单大对象所消耗的空间以供再使用。当删除包含其 BLOB 描述符的行时，则会自动删除简单大对象。（存储已删除简单大对象的 BLOB 页不可复用，直到备份了包含已删除简单大对象的原始 INSERT 记录的逻辑日志。有关更多信息，请参阅《*SinoDB*®管理员指南》中“逻辑日志”一章中“备份日志文件以释放 BLOB 页”。）

简单大对象的大小限制

BLOB 描述符可容纳的最大简单大对象是 $(2^{31} - 1)$ 或大约 2 GB。

BLOB 空间页类型

每个 BLOB 空间块包含三种类型的页：

- BLOB 空间自由图页
- 位图页
- BLOB 页

BLOB 空间自由图页

BLOB 空间自由图页标识未使用的 BLOB 页，以便数据库服务器可以将它们作为简单大对象创建的一部分来进行分配。分配 BLOB 页时，将更新该页的自由图条目。单个简单大对象的所有条目都是链接的。

BLOB 空间自由图页是一个数据库服务器页的大小。自由图页上每个条目是 8 字节，以两个 32 位字存储，如下所示：

- 第一字中的第一位指定 BLOB 页是可用的还是已使用的。
- 第一字中接下来的 31 位标识写入此 BLOB 页时的当前逻辑日志文件。（该信息是记录 TEXT 或 BYTE 数据时所必需的。）
- 第二字包含与存储在此页上的简单大对象相关联的表空间编号。

可放在自由图页上的条目数取决于计算机的页大小。BLOB 空间块中自由图页的数目取决于块中 BLOB 页的数目。

BLOB 空间位图页

BLOB 空间位图页跟踪块中 BLOB 空间自由图页的充满度和数目。每个 BLOB 空间位图页可跟踪一定数量的自由图页。BLOB 空间位图页的大小取决于系统页的大小。如果系统页是 2K，那么 BLOB 空间位图页可跟踪 2,032,128 BLOB 页。如果系统页是 4K，那么 BLOB 空间位图页可跟踪 8,258,048 BLOB 页。

BLOB 页

BLOB 页包含 TEXT 或 BYTE 数据。BLOB 页大小由创建 BLOB 空间的数据库服务器管理员指定。BLOB 页大小指定为页大小的倍数。

BLOB 空间 BLOB 页的结构

用于存储 BLOB 空间中简单大对象的存储策略与数据库空间存储策略不同。数据库服务器不在单个 BLOB 空间 BLOB 页上组合整个简单大对象或简单大对象的某些部分。例如：如果每个 BLOB 空间 BLOB 页是 24 KB，那么 26 KB 的简单大对象存储在两个 24 KB 的页上。多出的 22 KB 空间保持为未使用。

BLOB 页的结构包含 BLOB 页头、TEXT 或 BYTE 数据和页结束时间戳记。BLOB 页头，除其他信息外，还包含页头时间戳记和与数据行中转发指针相关联的 BLOB 时间戳记。如果简单大对象存储在多个 BLOB 页上，那么 BLOB 页头还包含指向下一 BLOB 页的转发指针和另一 BLOB 时间戳记。

智能大对象空间结构

智能大对象空间类型于 BLOB 空间，除了它是保存智能大对象外。

当在数据库中创建时，它包含一个智能大对象空间描述符。每个智能大对象空间块包含以下结构：

- 智能大对象空间块描述符
- 块可用页列表
- 智能大对象空间元数据区域（每块最多一个）
- 保留数据区域（每块最多两个）
- 用户数据区域（每块最多两个）

为了获得最佳性能，建议元数据区域位于智能大对象空间的中间。数据库服务器自动将元数据区域置于正确的位置。然而，要指定元数据区域的位置，请在 onspaces 命令指定 -Mo 标志。

如果未在 onspaces 命令中的 -Ms 标志指定元数据区域的大小，那么数据库服务器使用 AVG_LO_SIZE（默认为 8 KB）的值来计算元数据区域的大小。I有关更多信息，请参阅使用 *-Df* 选项创建智能大对象空间 在第373页。

正常情况下，可以让系统为您计算元数据大小。如果您想要估计元数据区域的大小，请参阅《*SinoDB*® 性能指南》中的“表性能的注意事项”一章。

图 *27: 单个智能大对象空间块* 在第270页说明智能大对象空间的块结构，它是在智能大对象空间创建后立即显示的。每个保留区域可分配给用户数据或元数据区域。保留区域总是在块的用户数据区域中。

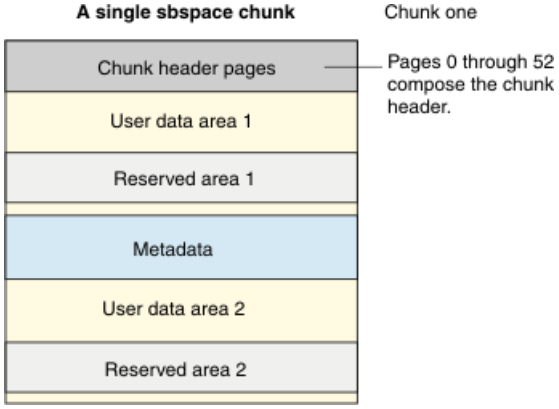


图 27: 单个智能大对象空间块

由于图 *27: 单个智能大对象空间块* 在第270页中的块是智能大对象空间中的第一个块，所以它包含智能大对象空间描述符。在 chunk one 中的块描述符表空间包含有关 chunk one 和其后加入智能大对象空间的所有块的信息

相关链接

SBSPACENAME 配置参数 在第155页

SYSSBSPACENAME 配置参数 在第184页

元数据区域的结构

智能大对象空间包含智能大对象空间中每个块的元数据区域。

与块头页一起，四个区域专用于智能大对象空间中的第一块：智能大对象空间描述符表空间、块附属表空间，以及 1 级和 2 级归档表空间。表空间头部分包含每个表空间（除表空间 tblspace 以外）的表空间头。图 28: 单块智能大对象空间的元数据区域的结构 在第271页显示了单块智能大对象空间中元数据的布局。

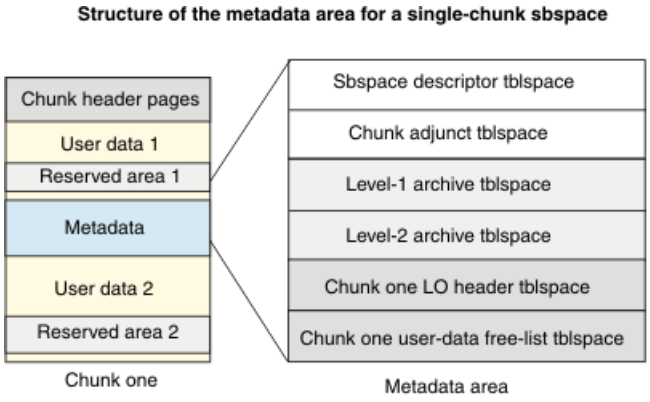


图 28: 单块智能大对象空间的元数据区域的结构

在 oncheck -ps 选项中指定智能大对象空间名称时，可以显示元数据区域中每个表空间已分配和已使用的页数。

以下项描述元数据区域是如何增长的：

- 智能大对象空间描述符表空间不增长。
- 块附属表空间在添加块时增长。
- LO 头表空间在向块添加大对象时增长。
- 如果块中可用空间已严重分段，则用户数据可用列表的表空间增长。

智能大对象页结构

每个智能大对象页由三个元素组成：智能大对象页头、实际用户数据本身，和智能大对象页尾。图 29: 智能大对象页结构 在第271页显示智能大对象页的结构。智能大对象页头由标准页头构成。智能大对象页尾用于检测页上不完整的写入和页损坏。

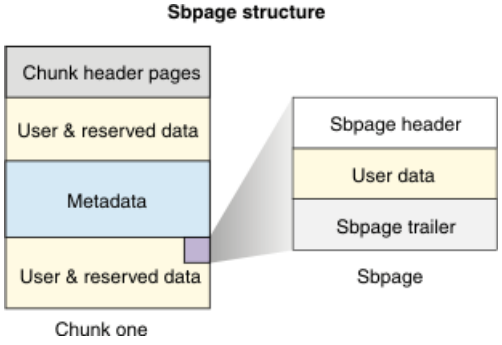


图 29: 智能大对象页结构

时间戳记

数据库服务器使用时间戳记来标识事件相对于同类型的其他事件发生时的时间。时间戳记不是引用特定时、分或秒的字面时间。它是数据库服务器循序指定的 4 字节整数。

数据库和表的创建：磁盘上发生什么

本节说明数据库服务器如何存储与数据库或表的创建相关的数据，以及分配数据存储所需的磁盘结构。

数据库的创建

在根数据库空间存在之后，用户可以创建数据库。后续段落描述当数据库服务器添加新的数据库时在磁盘上所发生的主要事件。

对系统目录表的磁盘空间分配

数据库服务器搜索数据库空间中的块可用列表页，以查找可创建系统目录表的可用空间。数据库服务器依序为每个系统目录表分配 8 个相邻页，即每个系统目录表的初始扩展数据块的大小。这些表是分别创建的，且不需要以彼此相邻方式位于数据库空间中。它们可位于不同块。当为每个表的初始扩展数据块找到足够空间时，则分配这些页，并更新相关联的块可用列表页。

对系统目录表的跟踪

数据库服务器跟踪数据库表空间（位于根数据库空间中）中新创建的数据库。描述该数据库的条目添加到根数据库空间中的数据库表空间中。（请参阅数据库表空间的结构 在第257页。）对于每个系统目录表，数据库服务器向数据库所在数据库空间中的表空间 tblspace 添加一个一页条目。（请参阅表空间 *Tblspace* 的结构 在第255页。）图 *30: 新数据库* 在第272页说明数据库表空间条目与该数据库的 systables 系统目录表位置之间的关系。

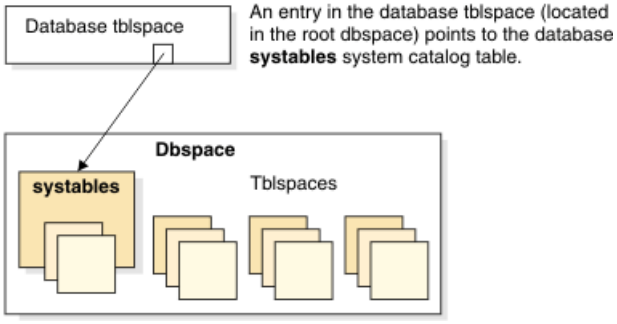


图 30: 新数据库

有关列出所创建的数据库的指示信息，请参阅《*SinoDB*® 管理员指南》中“管理数据库日志记录模式”一章中的监视数据库的部分。

表的创建

在根数据库空间存在并创建数据库之后，则具有必要 SQL 权限的用户可以创建数据库表。当用户创建表时，数据库服务器以扩展数据块（请参阅《*SinoDB*® 管理员指南》中“数据存储”一章中的“扩展数据块”）为单位向该表分配磁盘空间。后续段落描述数据库服务器创建表和分配磁盘空间的初始扩展数据块时所发生的主要事件。

磁盘空间分配

数据库服务器搜索数据库空间中块可用列表页，以查找等于表的初始扩展数据块大小的连续可用空间。当找到足够的空间时，分配这些页，并更新相关联的块可用列表页。

如果数据库服务器无法在数据库空间中找到足够的连续空间，那么它将最大可用的连续空间分配给该表。如果分配是可能的，那么即使所分配的空间量小于所请求的量，也不会返回错误消息。如果无法分配最小的扩展数据块大小，则返回错误。（扩展数据块不可跨两个块。）

表空间 Tblspace 中的条目

数据库服务器向此数据库空间中的表空间 tblspace 添加一个此表的一页条目。将从描述该表的表空间 tblspace 中的逻辑页编号派生出来的表空间编号指定给该表。请参阅表空间编号 在第256页。

表空间编号标识表空间所位于的数据库空间。表空间扩展数据块可位于任何数据库空间块中。

如果必须确切知道表空间扩展数据块的位置，请执行 oncheck -pe 命令，以获得按块排序的数据库空间布局列表

系统目录表中的条目

表本身是在数据库系统目录表中所存储的条目中完全描述的。每个表都指定了表标识号或 *tabid*。数据库中第一个用户定义的表对象的 tabid 值总是为 100。（tabid = 100 的对象也可能是一个视图、同义词或序列。）有关系统目录的全面讨论，请参阅《*SinoDB® SQL* 指南: 参考》。

表可位于与包含数据库的数据库空间不同的数据库空间中。表空间本身是所分配的扩展数据块的总和，而不是空间的单个连续分配。数据库服务器对表空间的跟踪与数据库无关。

临时表的创建

创建临时表所涉及的任务类似于数据库服务器添加新永久表所执行的任务。主要区别在于临时表不接收数据库系统目录中的条目。有关更多信息，请参阅《*SinoDB®* 管理员指南》中“数据存储”一章中“临时表”一节。

解释逻辑日志记录

本章内容

要显示逻辑日志文件所包含的逻辑日志记录，请使用 onlog 实用程序。

本章提供以下信息：

- 阅读逻辑日志记录的简要指南
- 不同逻辑日志记录类型的列表

一般来说，您无需阅读和解释逻辑日志文件。然而在调试情况下，onlog 输出是有用的。例如：您可能想要使用 onlog 跟踪特定的事务或查看数据库服务器对特定表空间做了什么更改。您还可使用 onlog 调查前滚过程中错误发生的原因。有关更多信息，请参阅 *onlog*#显示逻辑日志内容 在第327页。

关于逻辑日志记录

大多数 SQL 语句生成多个逻辑日志记录。当数据库服务器在逻辑日志中记录以下事件时，解释逻辑日志记录就更为复杂：

- 删除表或索引的事务
- 回滚事务
- 事务仍处于活动的检查点
- 分布式事务

以下各节讨论这些事件的逻辑日志记录。

删除表或索引的事务

一旦数据库服务器从数据库中删除表或索引，它无法回滚该删除操作。如果事务包含 DROP TABLE 或 DROP INDEX 语句，那么数据库服务器如下处理该事务：

1. 数据库服务器完成该事务的所有其他部分，并写下相关的逻辑日志记录。
2. 数据库服务器将 BEGCOM 记录写入逻辑日志，并写下与 DROP TABLE 或 DROP INDEX 的相关联的记录（例如：DINDEX）。
3. 数据库服务器写入 COMMIT 记录。

如果事务在数据库服务器将 BEGCOM 记录写入逻辑日志之后意外终止，那么数据库服务器将在恢复过程中前 滚该事务，因为它无法回滚该删除操作。

回滚的事务

当发生回滚时，数据库服务器为逻辑日志中每个回滚的记录生成一个补偿日志记录（CLR）。如果在回滚过程中发生系统故障，那么数据库服务器使用 CLR。CLR 提供向数据库服务器提供有关在故障发生前回滚进度的信息。换句话说，数据库服务器使用 CLR 记录回滚。

如果 CLR 包含短语 includes next record，那么打印的下一条日志记录会包含在 CLR 日志记录内作为补偿操作。否则，必须假定补偿操作是 CLR 的 link 字段所指向的日志记录的逻辑撤销。

带有活动事务的检查点

如果任何事务在检查点时是活动的，那么检查点记录包含使用以下列来描述每个活动事务的子条目：

- 日志开始（十进制格式）
- 事务 ID（十进制格式）
- 唯一日志号（十进制格式）
- 日志位置（十六进制格式）
- 用户名

分布式事务

分布式事务（跨多个数据库服务器的事务）与非分布式事务生成的日志记录稍有不同。如果事务正在提交时发生故障，那么可能需要阅读并解释它们，以确定事务在这两个数据库服务器上的状态。

分布式事务涉及以下日志记录：

- BEGPREP
- ENDTRANS
- HEURTX
- PREPARE
- TABLOCKS

有关这种类型的逻辑日志记录的更多信息，请参阅《*SinoDB®* 管理员指南》中“两阶段提交和逻辑日志记录”部分的材料。

如果使用 TP/XA 执行分布式事务，那么数据库服务器使用 XAPREPARE 记录而不是 PREPARE 记录。

逻辑日志记录结构

每个逻辑日志记录都有头信心。根据记录类型，输出中也会出现其他列信息，如逻辑日志记录类型和附加列在第275页中的解释。

逻辑日志记录头

表 *122: onlog* 的样本输出 在第274页包含了说明逻辑日志记录的头列的样本输出。

表 122: onlog 的样本输出

addr	len	type	xid	id	link
2c018	32	BEGIN	6	3	0
2c038	140	HDELETE	6	0	2c018
2c0c4	64	DELITEM	6	0	2c038
2c104	40	DELITEM	6	0	2c0c4

addr	len	type	xid	id	link
2c12c	72	HDELETE	6	0	2c104
2c174	44	DELITEM	6	0	2c12c
2c1a0	72	HDELETE	6	0	2c174
2c1e8	44	DELITEM	6	0	2c1a0
2c214	64	HDELETE	6	0	2c1e8
2c254	56	DELITEM	6	0	2c214
2c28c	48	DELITEM	6	0	2c254
2c2bc	24	PERASE	6	0	2c28c
2c2d4	20	BEGCOM	6	0	2c2bc
2c2e8	24	ERASE	6	0	2c2d4
2c300	28	CHFREE	6	0	2c2e8
2c31c	24	COMMIT	6	0	2c300

表 *123: onlog* 头列的定义 在第275页定义了每个头列的内容。

表 123: onlog 头列的定义

头字段	内容	格式
addr	日志记录地址（日志位置）	十六进制
len	以字节计的记录长度	十进制
type	记录类型的名称	ASCII
xid	事务号	十进制
id	逻辑日志号	十进制
link	到事务中的前一记录的链接	十六进制

逻辑日志记录类型和附加列

除了每个记录显示的六个头列外，一些记录类型还显示信息的附加列。信息的显示根据记录类型而变化。

下表列出了所有的记录类型和其附加列。

Action 列标识生成日志条目的数据库服务器操作类型。Additional Columns 和 Format 列描述除逻辑日志记录头 在第274页中所描述的头以外每个记录类型还出现什么信息。

表 124: 逻辑日志记录类型

此表提供记录类型、操作，以及逻辑日志记录类型的附加列和格式信息。

记录类型	操作	附加列和格式
ADDCHK	添加块。	• 块号 - 十进制 • 块名称 - ASCII
ADDDBS	添加数据库空间。	• 数据库空间名称 - ASCII
ADDITEM	向索引添加条目。	• 表空间标识 - 十六进制

记录类型	操作	附加列和格式
		• rowid - 十六进制 • 逻辑页 - 十进制 • 键号 - 十进制 • 键长 - 十进制
ADDLOG	添加日志。	• 日志号 - 十进制 • 日志大小（页） - 十进制 • pageno - 十六进制
ALLOCGENPG	分配类属页。	• 表空间标识 - 十进制 • rowid - 十进制 • 槽标志和长度 - 十进制 • 页版本（如果删除的话） - 十进制 • 标志，vimage 记录 - 十进制 • 上一个 rowid - 十进制 • 数据 - ASCII
ALTERDONE	分段更改完成。	• 表空间标识 - 十六进制 • 上一页的物理页号 - 十六进制 • 逻辑页号 - 十进制 • 更改的版本 - 十进制
ALTSPCOLSNEW	变更表中的已更改列。	• 列数 - 十进制 • 特殊列列表 - 数组
ALTSPCOLSOLD	变更表中的已更改列。	• 列数 - 十进制 • 特殊列列表 - 数组
BADIDX	损坏索引	• 表空间标识 - 十六进制
BEGCOM	开始提交。	• （无） - （无）
BEGIN	开始工作。	• 日期 - 十进制 • 时间 - 十进制 • SID - 十进制 • 用户 - ASCII
BEGPREP	由协调者数据库服务器所写入，以记录两阶段提交协议的开始。	• 标志 - 十进制（在分布式事务中值为 0。） • 参与者数 - 十进制
BEGWORK	开始事务。	• 开始事务时间 - 十进制 • 用户标识 - 十进制 • 进程标识 - 十进制
BFRMAP	简单大对象可用图更改。	• 表空间标识 - 十六进制 • bpageno - 十六进制 • 状态 USED/FREE 日志标识 - 十进制 • 上一页 - 十六进制
BLDCL	建构表空间。	• 表空间标识 - 十六进制

记录类型	操作	附加列和格式
		• fextsize - 十进制 • nextsize - 十进制 • 行大小 - 十进制 • ncolumns - 十进制 • 表名称 - ASCII
BMAPFULL	准备用于变更的已修改位图。	• 表空间标识 - 十六进制 • 位图页号 - 十进制
BMAP2TO4	2 位位图更改成 2 个 4 位位图。	• 表空间标识 - 十六进制 • 2 位位图页号 - 十进制 • 标志 - 十进制
BSPADD	添加 BLOB 空间。	• BLOB 空间名称 - ASCII
BTCPYBCK	将子键复制回父。	• 表空间标识 - 十六进制 • 父逻辑页 - 十进制 • 子逻辑页 - 十进制 • 槽 - 十进制 • rowoff - 十进制 • 键号 - 十进制
BTMERGE	合并 B 型树节点。	• 表空间标识 - 十六进制 • 父逻辑页 - 十进制 • 左逻辑页 - 十进制 • 右逻辑页 - 十进制 • 左槽 - 十进制 • 左 rowoff - 十进制 • 右槽 - 十进制 • 右 rowoff - 十进制 • 键号 - 十进制
BTSHUFFL	拖曳 B 型树节点。	• 表空间标识 - 十六进制 • 父逻辑页 - 十进制 • 左逻辑页 - 十进制 • 右逻辑页 - 十进制 • 左槽 - 十进制 • 左 rowoff - 十进制 • 键号 - 十进制 • 标志 - 十六进制
BTSPLIT	分割 B 型树节点。	• 表空间标识 - 十六进制 • rowid - 十六进制 • 父逻辑页 - 十进制 • 左逻辑页 - 十进制 • 右逻辑页 - 十进制 • 无限逻辑页 - 十进制 • 根左逻辑页 - 十进制 • midsplit - 十进制 • 键号 - 十进制

记录类型	操作	附加列和格式
		• 键长 - 十进制
CDINDEX	创建拆离的索引。	• 数据库名 - ASCII • 所有者 - ASCII • 表名 - ASCII • 索引名 - ASCII

CDR	捕获由更新语句修改过的表列组，例如：*bitvector*。此日志记录允许 Enterprise Replication 仅捕获更改过的数据，以避免将未更改过的列传输给目标站点。
	示例中，表的前 6 列未更改过（bitvector 中的最左 6 位是 0），第 7 和 8 列更新过（第 7 和 9 位是 1）等等。onlog 输出显示了与输出单行中可显示的位数一样多的 bitvector 位。要查看以十六进制显示的整个 bitvector，请使用 onlog -l 命令。
		• name of CDR record - ASCII
		• partition number - 十六进制
		• bitvector - 二进制
		CDR 日志记录的 onlog 输出样本：
		`adr len type xid id link`
		`40 36 CDR 14 0 18`
		`name partno bitvector`
		`UPDCOLS 10009a 000000110100110100`

CHALLOC	块扩展数据块分配。	• pageno - 十六进制 • 大小 - 十六进制
CHCOMBINE	块扩展数据块结合。	• pageno - 十六进制
CHFREE	块扩展数据块释放。	• pageno - 十六进制 • 大小 - 十六进制
CHKADJUP	更新磁盘上的块附件。当数据库服务器将空间从保留区域搬移到元数据或用户数据区域，或者当用户添加智能大对象空间块时，数据库服务器写入此记录。	• 块号 - 整数 • ud1_start_page - 整数 • ud1_size - 整数 • md_start_page - 整数 • md_size - 整数 • ud2_start_page - 整数 • ud2_size - 整数 • 标志 - 十六进制
CHPHYLOG	更改物理日志位置。	• pageno - 十六进制 • 大小（KB） - 十六进制 • 数据库空间名 - ASCII
CHRESERV	保留用于元数据窃取的扩展数据块。该记录在添加智能大对象空间块时写入。	• 块号 - 整数 • 页号 - 整数 • 长度 - 整数
CHSPLIT	块扩展数据块分割。	• pageno - 十六进制
CINDEX	创建索引。	• 表空间标识 - 十六进制

记录类型	操作	附加列和格式
		• 低的 rowid - 十进制 • 高的 rowid - 十进制 • 索引描述符 - ASCII
COARSELOCK	粗粒度锁定	• 表空间标识 - 十六进制 • 旧的粗粒度锁定标志值 - 十进制 • 新的粗粒度锁定标志值 - 十进制
CKPOINT	检查点。	• 最大用户数 - 十进制 • 活动的事务数 - 十进制
CLR	补偿日志记录；回滚过程中创建。	• （无）-（无）
CLUSIDX	创建群集索引。	• 表空间标识 - 十六进制 • 键号 - 十进制
COLREPAI	调整 BYTE、TEXT 或 VARCHAR 列。	• 表空间标识 - 十六进制 • 已调整的列数 - 十进制
COMMIT	提交工作。	• 日期 - 十进制 • 时间 - 十进制
COMTAB	压缩页上的槽表。	• 逻辑页号 - 十进制 • 已移动的槽数 - 十进制 • 压缩的槽对 - ASCII
COMWORK	结束事务与提交工作。	• 结束事务时间 - 十进制 • 开始事务时间 - 十进制
DELETE	删除前映像。	• 表空间标识 - 十六进制 • rowid - 十六进制
DELITEM	从索引中删除项目。	• 表空间标识 - 十六进制 • rowid - 十六进制 • 逻辑页 - 十进制 • 键号 - 十进制 • 键长 - 十进制
DERASE	删除关闭数据库空间中的表空间。	• 表空间编号 - 十六进制 • 表锁编号 - 十进制
DFADDEXT	添加新扩展数据块。	• partnum - 十六进制 • 列表中扩展数据块条目的偏移量 - 十六进制 • 扩展数据块大小（页） - 十进制 • 扩展数据块的物理地址 - 偏移量和块号 - 十六进制
DFDRPEXT	删除原扩展数据块。	• partnum - 十六进制 • 列表中扩展数据块条目的偏移量 - 十六进制

记录类型	操作	附加列和格式
		• 该扩展数据块的原始大小 - 十进制 • 物理地址 - 偏移量和块号 - 十六进制
DFEND	取消分段操作结束。	• partnum - 十六进制
DFMVPG	页从旧的扩展数据块搬移到新的扩展数据块。	• partnum - 十六进制 • 新扩展数据块的偏移量 - 十六进制 • 源逻辑页编号 - 十六进制 • 目标物理地址 - 偏移量和块号 - 十六进制 • 源物理地址 - 偏移量和块号 - 十六进制
DFREMDUM	删除虚拟条目。	• partnum - 十六进制
DFSTART	取消分段操作开始。	• partnum - 十六进制
DINDEX	删除索引。	• 表空间标识 - 十六进制 • 键号 - 十进制
DRPBSP	删除 BLOB 空间。	• BLOL 空间名 - ASCII
DRPCHK	删除块。	• 块号 - 十进制 • 块名 - ASCII
DRPDBS	删除数据库空间。	• 数据库空间名 - ASCII
DRPLOG	删除日志。	• 日志编号 - 十进制 • 日志大小（页） - 十进制 • pageno - 十六进制
ENDTRANS	由协调者和参与者数据库服务器写入以记录事务的结束。ENDTRANS 指示数据库服务器从其共享内存事务表中移除事务条目，并关闭事务。 在协调者逻辑日志中，每个导致提交事务的 BEGPREP 是与 ENDTRANS 记录配对。如果协调者的最终决定是回滚事务，则不写入 ENDTRANS 记录。 在参与者逻辑日志中，每个 ENDTRANS 记录是与相应的 HEURTX 记录配对。	• （无）-（无）
ERASE	删除表空间。	• 表空间标识 - 十六进制
FREE_RE	将保留扩展数据块的扩展数据块分配给智能大对象空间块的元数据或用户数据区域。	• 块号 - 整数 • 页号 - 整数 • 长度 - 整数 • 标志 - 十六进制

记录类型	操作	附加列和格式
HDELETE	删除主行。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制
HEURTX	由参与者数据库服务器写入以记录回滚事务的试探性决策。其应与指示事务回滚的标准 ROLLBACK 记录相关联。I	• 标志 - 十六进制(Value is always 1.)
HINSERT	主行插入。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制
HUPAFT	主行更新，后映像。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制
HUPBEF	主行更新，前映像。 此外，HUPBEF 头的标志字段可能包含以下值： LM_PREVLSN 确认存在 LSN。 LM_FIRSTUPD 确认这是该事务对此行标识的首次更新。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制 • LSN（可选）- 十进制
HUPDATE	如果主行更新前映像和后映像可一起放入单页中，那么数据库服务器写入单一 HUPDATE 记录。 此外，HUPDATE 日志的标志字段可能包含以下值： LM_PREVLSN 确认存在 LSN。 LM_FIRSTUPD 确认这是该事务对此行标识的首次更新。	• 表空间标识 - 十六进制 • rowid - 十六进制 • 向前 ptr rowid - 十六进制 • 旧的 slotlen - 十进制 • 新的 slotlen - 十进制 • 块的数量 - 十进制 • LSN（可选）- 十进制
IDXFLAGS	索引标志。	• 表空间标识 - 十六进制 • 键号 - 十六进制
INSERT	插入后映像。	• 表空间标识 - 十六进制 • rowid - 十六进制
ISOSPCOMMIT	记录隔离的保存点提交。	• 结束事务时间 - 十进制 • 开始事务时间 - 十进制
LCKLVL	锁定模式（页或行）。	• 表空间标识 - 十六进制 • 旧的锁定模式 - 十六进制 • 新的锁定模式 - 十六进制

记录类型	操作	附加列和格式
LG_ADDBPOOL	联机添加缓冲池。	• 页大小（字节） - 十进制 • 池中的缓冲区数 - 十进制 • lru 队列数 - 十进制 • lru_max_dirty 百分比 - 十进制 • lru_min_dirty 百分比 - 十进制
PTRUNCATE	表示意图截断表。根据指定的命令选项，将分区标记为删除或复用。	• 表空间标识 - 十六进制
TRUNCATE	TRUNCATE 释放扩展数据块并提交事务。	• 表空间标识 - 十六进制
MVIDXND	移动索引节点以允许 2 位至 4 位的位图转换。	• 表空间标识 - 十六进制 • 旧的页号 - 十进制 • 新的页号 - 十进制 • 父页号 - 十进制 • 父槽号 - 十进制 • 父槽偏移量 - 十进制 • 键号 - 十进制
PBDELETE	删除表空间 BLOB 页。	• bpageno - 十六进制 • 状态 USED/FREE 唯一标识 - 十进制
PBINSERT	插入表空间 BLOB 页。	• bpageno - 十六进制 • 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制 • pbrowid - 十六进制
PDINDEX	预删除索引。	• 表空间标识 - 十六进制
PGALTER	原位更改的页。	• 表空间标识 - 十六进制 • 物理页号 - 十六进制
PGMODE	位图中修改的页模式。	• 表空间标识 - 十六进制 • 逻辑页号 - 十进制 • 旧模式 - 十六进制 • 新模式 - 十六进制
PERASE	预擦除旧文件。标记要删除的表。 数据库服务器释放提交的空间。	• 表空间标识 - 十六进制
PNGPALIGN8	使用该表空间中的页作为类属页。	• 无
PNLOCKID	更改表空间的 lockid.	• 表空间标识 - 十六进制 • 旧的锁标识 - 十六进制 • 新的锁标识 - 十六进制
PNSIZES	设置表空间扩展数据块大小。	• 表空间标识 - 十六进制 • fextsize - 十进制

记录类型	操作	附加列和格式
		• nextsize - 十进制
PREPARE	由参与者数据库服务器写入，以记录参与者提交事务的能力（如果这样指示过）。	• 协调者的 DBSERVERNAME - ASCII
PTADESC	添加更改描述信息。	• 表空间标识 - 十六进制 • 上一页的物理页号 - 十六进制 • 逻辑页号 - 十进制 • 添加的列数 - 十进制
PTALTER	分段更改开始。	• 表空间标识 - 十六进制 • 上一页的物理页号 - 十六进制 • 逻辑页号 - 十进制 • 更改描述页号 - 十进制 • 添加的列数 - 十进制 • 更改的版本 - 十进制 • 添加的行大小 - 十进制
PTALTNEWKEYD	在更改表命令之后更新表空间头中的键描述符。	• 键描述符的字节数 - 十进制 • 键描述符中的数据 - ASCII
PTALTOLDKEYD	在更改表命令之后更新键描述符。	• 键描述符的字节数 - 十进制 • 键描述符中的数据 - ASCII
PTCOLUMN	向分段添加特殊列。	• 表空间标识 - 十六进制 • 列数 - 十进制
PTEXTEND	表空间扩展数据块。	• 表空间标识 - 十六进制 • 最后逻辑页 - 十进制 • 第一物理页 - 十六进制
PTRENAME	重命名表。	• 表空间标识 - 十六进制 • 旧表名 - ASCII • 新表名 - ASCII
RDELETE	剩余页删除。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制 • hrowid（可选） - 十进制 • poffset（可选）- 十进制
RENDBS	重命名数据库空间。	• 新数据库空间名 - ASCII
REVERT	记录对较早版本的数据库空间的数据库空间复原	• 复原事件类型 - 十进制 • arg1 - 十进制 • arg2 - 十进制 • arg3 - 十进制
RINSERT	剩余页插入。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制

记录类型	操作	附加列和格式
		• hrowid（可选）- 十进制 • poffset（可选）- 十进制
ROLLBACK	回滚工作。	• 日期 - 十进制 • 时间 - 十进制
ROLWORK	结束事务并回滚工作。	• 结束事务时间 - 十进制 • 开始事务时间 - 十进制
RSVEXTEND	记录对保留页的扩展。	• 页数 - 十进制 • 扩展数据块的物理页号 - 十六进制
RTREE	记录 R 型树索引页的插入和删除。（以物理方式记录 R 型树索引上的其他操作。）记录子类型： • LEAFINS - 在叶页中插入项 • LEAFDEL - 删除叶页中的项	• 记录子类型 - ASCII • 索引页行标识 - 十六进制 • 元组长度 - 十进制 • 基本表行标识 - 十进制 • 基本表分段标识 - 十进制 • 删除标志 - 十进制
RUPAFT	剩余页更新，后映像。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制
RUPBEF	剩余页更新，前映像。	• 表空间标识 - 十六进制 • rowid - 十六进制 • slotlen - 十进制 • hrowid（可选）- 十进制 • poffset（可选）- 十进制
RUPDATE	如果剩余页更新前映像和后映像可一起放入单页中，那么数据库服务器写入单一 RUPDATE 记录。	• 表空间标识 - 十六进制 • rowid - 十六进制 • 向前 ptr rowid - 十六进制 • 旧 slotlen - 十进制 • 新 slotlen - 十进制 • 块数 - 十进制 • hrowid（可选）- 十进制 • poffset（可选）- 十进制
SBLOB	指示智能大对象的子系统日志记录。 各种记录子类型为： CHALLOC CHCOMBINE CHFREE CHSPLIT CREATE DELETES EXTEND HDRUPD PDELETE	• 变化 有关更多信息，请参阅智能大对象的日志记录类型 在第286页。 变化

记录类型	操作	附加列和格式
	PTRUNC REFCOUNT UDINSERT UDINSERT_LT UDUPAFT UDUPAFT_LT UDUPAFT UDUPAFT_LT UDWRITE UDWRITE_LT	
SYNC	如果日志文件是空的且管理者指示数据库服务器切换到下一日志文件，那么写入逻辑日志文件。	• （无）-（无）
TABLOCKS	由协调者或参与者数据库服务器写入。其与 BEGPREP 或 PREPARE 记录相关联，并包含事务所持有已锁定表空间的列表（按表空间编号）。（在分布式事务中，事务显示为锁定的所有者。）	• 锁数 - 十进制 • 表空间编号 - 十六进制
UDINSERT	附加新用户数据。	• 锁数 - 十进制 • 表空间编号 - 十六进制
UDUPAFT	如果 UDWRITE 代价太高，那么更新用户数据后映像	• 块 - 十进制 • 块中的页 - 十六进制 • 页中的偏移量 - 十六进制 • 数据长度 - 十六进制
UDUPBEF	如果 UDWRITE 代价太高，那么更新用户数据前映像。	• 块 - 十进制 • 块中的页 - 十六进制 • 页中的偏移量 - 十六进制 • 数据长度 - 十六进制
UDWRITE	更新用户数据（差分映像）。	• 块 - 十进制 • 块中的页 - 十六进制 • 块中的偏移量 - 十六进制 • 写入前长度 - 十六进制 • 写入后长度 - 十六进制
UNDO	将要回滚的一系列事务的头记录。	• 计数 - 十进制
UNDOBLDC	如果应回滚 CREATE TABLE 语句但由于相关块已关闭，那么写入此记录。当重放日志文件时，将删除该表。	• 表空间编号 - 十六进制
UNIQID	当向行指定新的 SERIAL 值时记录。	• 表空间标识 - 十六进制 • 唯一标识 - 十进制

记录类型	操作	附加列和格式
UNIQ8ID	当向行指定新的 SERIAL8 时记录。	• 表空间标识 - 十六进制 • 唯一标识 - 十进制
UPDAFT	更新后映像。	• 表空间标识 - 十六进制 • rowid - 十六进制
UPDBEF	更新前映像。	• 表空间标识 - 十六进制 • rowid - 十六进制
XAPREPARE	参与者可以提交此 XA 事务。	• （无） -（无）

智能大对象的日志记录类型

所有智能大对象日志记录都是 SBLOB 类型。每个智能大对象记录包含 6 个头列（在逻辑日志记录头 在第274页中描述）、记录子类型和其他信息。信息的显示会根据记录子类型而变化。

表 125: 智能大对象的记录子类型 在第286页列出所有智能大对象记录类型。Subtype 列描述智能大对象记录类型。Action 列标识生成日志条目的数据库服务器操作的类型。Additional Columns 和 Format 列描述每个记录类型显示什么信息。

表 125: 智能大对象的记录子类型

此表提供智能大对象的记录子类型、操作、附加列和格式。

记录子类型	操作	附加列	格式
CHALLOC	分配块扩展数据块。	扩展数据块 [chk, page, len]	十进制
		flags	Hexadecimal
CHCOMBINE	组合用户数据扩展数据块列表中的两页。	块号	十进制
		第一页	十进制
		第二页	十进制
CHFREE	释放块扩展数据块。	扩展数据块 [chk, page, len]	十进制
CHSPLIT	分割用户数据扩展数据块列表中的页。	块号	十进制
		UDFET 要分割的页	十进制
CREATE	创建智能大对象。	智能大对象标识 [sbs, chk, page, oid]	十进制
		lomaphdr 中的扩展数据块的数量	十进制
DELETE	提交时删除智能大对象。	智能大对象标识 [sbs, chk, page, oid]	十进制
EXTEND	向智能大对象的扩展数据块列表添加扩展数据块。	智能大对象标识 [sbs, chk, page, oid]	十进制
		扩展数据块 [chk, page, len]	十进制
		lomap 溢出页号	十进制
HDRUPD	更新智能大对象头页。	智能大对象标识 [sbs, chk, page, oid]	十进制
		旧 EOF 偏移量	字符串

记录子类型	操作	附加列	格式
		新 EOF 偏移量	字符串
		旧的时间	十进制
		新的时间	十进制
PDELETE	对提交时要删除的智能大对象进行队列。	智能大对象标识 [sbs, chk, page, oid]	十进制
PTRUNC	对提交时要截断的智能大对象进行队列。	智能大对象标识 [sbs, chk, page, oid]	十进制
		旧偏移量	字符串
		新偏移量	字符串
REFCOUNT	增加或减少智能大对象的引用计数。	智能大对象标识 [sbs, chk, page, oid]	十进制
		如果增加，则为 1；如果减少，则为 0。	十进制
UDINSERT,	附加新用户数据。	块	十进制
UDINSERT_LT		块中的页	十进制
		页中的偏移量	十进制
		数据长度	十进制
UDUPAFT,	如果 UDWRITE 代价太高，那么更新用户数据后映像。	块	十进制
UDUPAFT_LT		块中的页	十进制
		页中的偏移量	十进制
		数据长度	十进制
UDUPBEF,	如果 UDWRITE 代价太高，那么更新用户数据前映像。	块	十进制
UDUPBEF_LT		块中的页	十进制
		页中的偏移量	十进制
		数据长度	十进制
UDWRITE,	更新用户数据（差分映像）。	块	十进制
UDWRITE_LT		块中的页	十进制
		页中的偏移量	十进制
		写入前长度	十进制
		写入后长度	十进制
		不同映像块的数量	十进制

图 *31: onlog* 输出中的智能大对象记录 在第287页显示 onlog 输出中的智能大对象记录示例。前两笔记录显示扩展数据块已释放。下一组记录（两侧为 BEGIN 和 COMMIT）显示存储分配和智能大对象的创建。

```
addr     len   type    xid       id  link     subtype  specific-info

4e8428    40   SBLOB     8        0  4e7400   CHFREE   (2,53,421)
4e8450    40   SBLOB     8        0  4e8428   CHFREE   (2,579,421)
```

```
c8018    40   BEGIN    8        3  0        07/13/98 10:23:04 34       informix
c8040    264  SBLOB    8        0  c8018    CREATE   [2,2,1,900350517]  10
c8148    44   SBLOB    8        0  c8040    CHALLOC  (2,53,8) 0x1
c8174    68   SBLOB    8        0  c8148    EXTEND   [2,2,1,900350517] (2,53,8) -1
c81b8    264  SBLOB    8        0  c8174    CREATE   [2,2,2,900350518]  10
c82c0    44   SBLOB    8        0  c81b8    CHALLOC  (2,61,1) 0x1
c82ec    68   SBLOB    8        0  c82c0    EXTEND   [2,2,2,900350518] (2,61,1) -1
c8330    56   SBLOB    8        0  c82ec    REFCOUNT [2,2,1,900350517] 1
c8368    56   SBLOB    8        0  c8330    REFCOUNT [2,2,2,900350518] 1
c83a0    36   COMMIT   8        0  c8368    07/13/98 10:23:05

c83c4    40   BEGIN    8        3  0        07/13/98 10:23:05 34       informix
c83ec    264  SBLOB    8        0  c83c4    CREATE   [2,2,3,900350519]  10
c84f4    44   SBLOB    8        0  c83ec    CHALLOC  (2,62,1) 0x1
c8520    68   SBLOB    8        0  c84f4    EXTEND   [2,2,3,900350519] (2,62,1) -1
c8564    56   SBLOB    8        0  c8520    REFCOUNT [2,2,3,900350519] 1
c859c    36   COMMIT   8        0  c8564    07/13/98 10:23:05
```

图 31: onlog 输出中的智能大对象记录

第 II 部分

管理实用程序

实用程序概述

SinoDB® 数据库服务器实用程序允许您直接从命令行执行管理任务。

数据库服务器实用程序支持多字节命令行参数。有关支持多字节命令行参数的实用程序的完整列表，请参阅*Locale-specific support for utilities*。

在执行实用程序之前，数据库服务器必须是联机的，以下情况除外：

- oninit
- 一些 onlog 选项
- 一些 oncheck 选项

  注： 在使用实用程序时，请不要使用 UNIX™ 命令 CTRL-C 向进程发送中断信号，因为这可能引起错误。

获取实用程序版本信息

许多 SinoDB® 命令行实用程序允许您使用 -V 和 -version 选项来获取版本，主要用于调试。

-V 选项显示软件版本号和序列号。

-version 选项扩展 -V 选项，显示关于构建操作系统、构建号和构建日期的其他信息。

utility { *utility specific options* | -V | -version }

-V 和 -version 选项不可与任何其他实用程序选项一起使用。例如：onstat -version 命令可能显示以下输出。

```
onstat -version

        Program:          onstat
    Build Version:    12.10.FC7
    Build Host:       connla
    Build OS:         SunOS 5.6
    Build Number:     009
    Build Date:       Fri Dec 15 03:38:27 CDT 2017
    GLS Version:      glslib-4.50.xC2
```

onstat -V 命令可能显示以下信息：

```
SinoDB Version 12.10.FC7   Software Serial Number
 RDS#N000000
```

设置实用程序的本地环境变量

在 UNIX™ 操作系统上，您可以启动某些实用程序而不需要在 shell 环境中设置本地环境变量。可在 onconfig 文件中设置本地环境变量。在运行命令以启功实用程序时，请使用 -FILE 选项指向 onconfig 文件。

开始之前，请确保满足以下先决条件：

- 实用程序的可执行程序的路径是现有 shell 环境的一部分。
- 如果要在远程计算机上执行命令，请配置远程 shell 实用程序，例如：SSH。

1. 将一个或多个环境变量的值添加到 onconfig 文件。对于每个指令，使用以下格式：
   #$*variable_name value*
2. 在执行命令以启动实用程序时，使用 -FILE 选项指定 onconfig 的完整路径或相对路径。
   查看 -FILE 选项的参考信息中的语法、用法和示例。

实用程序读取并设置 onconfig 文件中指定的环境变量，且这些值优先于本地 shell 环境中设置的值。

相关链接

数据库服务器文件 在第715页
-FILE 选项 在第322页
onconfig 文件 在第41页

finderr 实用程序

使用 finderr 实用程序查看 SinoDB® 错误消息上的其他信息。在 UNIX™ 和 Linux™ 平台上，信息显示在命令行上。在 Windows™ 平台上，信息显示在 Error Messages 程序中。

语法

finderr [{ - + }] *error_number*

表 126: finderr 元素

元素	目的	主要注意事项
error_number	用于提供其他信息的错误消息编号。	UNIX™ 或 Linux™ 上：如果您不包含减号 (-) 或加号 (+)，并且错误消息的正负版本都存在，那么显示消息的负版。要显示有关错误消息编号为正的信息，请在错误编号前加上加号。 在 Windows™ 上：如果您不包含减号 (-) 或加号 (+)，并且错误消息的正负版本都存在，那么您必须选择要在 Error Messages 程序中查看那种消息。

用法

打印在消息日志中的错误消息包括消息编号和短的消息描述。使用消息编号与 finderr 命令来查找错误原因的更多详细描述和可能的用户操作以修正或防止错误。

在 Windows™ 上，从数据库服务器程序组中选取 Error Messages 可直接打开 Error Messages 程序。

示例

在 UNIX™ 或 Linux™ 平台上，以下命令显示有关错误消息 -201 的信息：

```
finderr 201

-201  A syntax error has occurred.

This general error message indicates mistakes in the form of an SQL
statement. Look for missing or extra punctuation (such as missing or
extra commas, omission of parentheses around a subquery, and so on),
keywords misspelled (such as VALEUS for VALUES), keywords misused (such
as SET in an INSERT statement or INTO in a subquery), keywords out of
sequence (such as a condition of "value IS NOT" instead of "NOT value
IS"), or a reserved word used as an identifier.

Database servers that provide full NIST compliance do not reserve any
words; queries that work with these database servers might fail and
return error -201 when they are used with earlier versions of Sinoregal DS
database servers.

The cause of this error might be an attempt to use round-robin syntax with
CREATE INDEX or ALTER FRAGMENT INIT on an index. You cannot use round-robin
indexes.

The error may also occur if an SQL statement uses double quotation marks
around input strings and the environment variable DELIMIDENT is set.
If DELIMIDENT is set, strings that are surrounded by double quotation
marks are regarded as SQL identifiers rather than string literals. For
more information on the usage of DELIMIDENT, see the SinoDB Guide to
SQL: Reference.
```

以下命令显示有关 错误消息 100（对应于 SQLCODE 值 100）的信息：

```
finderr +100

100 No matching records found.

The database server did not find any more data. This message is an ANSI-
standard SQLCODE value. If you attempted to select or fetch data, you
encountered the end of the data, or no data matched the criteria in the
WHERE clause. Check for an empty table. Use this SQLCODE value to determine
when a statement reaches the end of the data. For more information, see the
discussion of SQLCODE in the Sinoregal ESQL/C Programmer's Manual. The
database server can return this SQLCODE value to a running program.
```

genoncfg 实用程序

用 genoncfg 实用程序可加速将默认 SinoDB® 配置文件（onconfig.std）定制到主机环境和数据库服务器实例的过程。

语法

genoncfg

{ *input_fileinformixdir* | -h | -V | -version }

元素	目的	Key 注意事项
input_file	包含参数设置的输入文件的名称。	
informixdir	要配置的 SinoDB® 安装的路径。	如果设置了 INFORMIXDIR 环境变量，那么可以省略安装路径。如果已设置 INFORMIXDIR 变量，而又在命令行输入安装路径，那么实用程序使用命令行上的路径来运行。
-h	有关 genoncfg 实用程序的帮助信息。	
-V	显示短版本信息并退出命令行实用程序。	
-version	显示扩展的版本信息并退出命令行实用程序。	

用法

运行该实用程序之前，以 root 或 用户 informix 登录主机。

在运行 genoncfg 实用程序之前，必须在输入文件中为主机环境设置有效的参数。所有环境的输入文件都需要有参数 disk。还可以在输入文件中放入指令。运行该实用程序不需要这些指令，但在某些状况下是有帮助的。

该实用程序不读取或修改任何现有的配置文件。如果主机环境中有预先存在的 ONCONFIG 文件，在运行该实用程序时不会变更其中任何参数值。因此，您可以先检查建议的配置值，然后使它们在数据库服务器实例上生效。

要使用 genoncfg 实用程序，请执行以下操作：

1. 使用文本编辑器创建输入文件，其包含 genoncfg 实用程序处理的参数的值。
2. 使用该输入文件运行该实用程序。生成配置文件（名为onconfig）并保存在工作目录中。
3. 可选#重命名生成的配置文件。
4. 如果要使用生成的配置文件运行数据库服务器实例，那么将该文件复制到 $INFORMIXDIR/etc 并更新 ONCONFIG 环境变量。

genoncfg 实用程序的输入文件

使用输入文件指定以下有关数据库服务器实例的xinx：

- 参与联机事务处理（OLTP）的连接数
- 参与决策支持系统（DSS）的连接数
- 磁盘空间
- CPU 使用率
- 网络连接设置
- 恢复时间

输入文件是 ASCII 文本文件。参数没有必需的顺序。以下是输入文件的示例：

```
cpus 1
memory 1024 m
connection name demo_on onsoctcp 9088
servernum 1
oltp_connections 10
dss_connections 2
disk /opt/IBM/informix/demo/server/online_root 0 k 300 m
directive one_crit
directive debug
```

表 127: genoncfg 实用程序的输入文件的参数

参数	描述
connection	服务器连接参数： • name 或 alias，取决于连接是使用特定的服务器名（配置文件中的 DBSERVERNAME 参数）还是替代服务器名（配置文件中的 DBSERVERALIASES 参数） • 连接的名称 • 服务器连接类型（相当于配置文件中的 NETTYPE） • 服务的端口号 示例：connection name demo_on onsoctcp 9088
cpus	分配给实例的中央处理单位（CPU）数。示例：cpus 1
directive	与 genoncfg 实用程序一起使用的指令。 • one_crit：配置数据库服务器仅在根数据库空间中存储物理日志、逻辑日志和数据。 • debug：实时显示有关主机环境和在配置文件上执行的操作的信息。 示例：directive one_crit 此信息有助于解决数据库服务器配置的问题。其中一种情况就是 debug 指令可以节省时间。在这种情况中，您阅读显示的信息并注意到该实用程序正在创建不需要或不起作用的 onconfig 文件。在实用程序仍在运行时停止该实用程序，调整输入文件设置，然后使用修改后的输入文件重新运行该实用程序。
disk	实例的磁盘存储空间设置： • 根数据库空间的位置 • 偏移量大小，以 MB（m）或 KB（k）为单位。 • 根数据库空间大小，以 MB（m）或 KB（k）为单位。 示例： UNIX™：/opt/IBM/dbspace/rootdbs Windows™：d:\INFXDATA\rootdbs 重要： 如果输入的路径位置是工作实例的根数据库空间，则该实例将被覆盖并变为不可用。
dss_connections	实例上估计的决策支持系统（DSS）连接数。例如：获取商业智能结果集的查询客户端或其他应用程序可以是 DSS 连接。示例：dss_connections 2
memory	实例的内存量，以 MB（m）为单位。示例：memory 1024 m
oltp_connections	实例上估计的联机事务处理（OLTP）连接数。通常，修改实例中数据库状态的应用程序是 OLTP 连接。示例：oltp_connections 10

参数	描述
rto_server_restart	指定重启 SinoDB® 并使其进入联机或静默模式后数据库必须从问题中恢复的时间（以秒为单位）。可以将该值设置为 0 以禁用配置参数，或者设置为介于 60 和 1800 之间的值以启用参数并指示秒数。示例：rto_server_restart 100 指定恢复时间目标为 100 秒。
servernum	数据库服务器实例的唯一标识。示例：servernum 1

相关链接

更改 *onconfig* 文件 在第42页

oncheck 实用程序

使用 oncheck 实用程序检查指定的磁盘结构的不一致性、修复不一致索引结构，以及显示有关磁盘结构的信息。

oncheck 实用程序检查索引时需要排序空间。所需的排序空间量与构建索引所需的量是一样的。有关计算所需的临时空间量的信息，请参阅《*SinoDB*® 性能指南》中的估算用于索引构建的临时空间。如果接收到错误 "no free disk space for sort"，那么您必须估计所需的临时空间量并使其可用。

还可使用等同某些 oncheck 命令的 SQL 管理 API 命令。

oncheck 检查并修复

oncheck 实用程序修复磁盘结构。

oncheck 实用程序可以修复以下类型的磁盘结构：

- 分区页统计信息
- 位图页
- 分区 BLOB 页
- BLOB 空间 BLOB 页
- 索引
- 智能大对象空间页
- 智能大对象空间的元数据分区

如果 oncheck 检测到其他结构中的不一致性，向您发送这些不一致性的消息警告，但不能解决问题。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“一致性检查”一章和磁盘结构和存储 在第252页。

每个选项做些什么？

oncheck 选项分为三类：检查、修复，和显示。

显示或打印选项（以字母 p 为前缀的那些选项）的功能与 -c 选项是相同的，但 -p 选项除外，其显示 oncheck 实用程序执行时受检查数据的其他信息。您不可以组合 oncheck 选项标志，以下段落描述的除外。

通常，-c 选项检查一致性，并且只在发现错误或不一致性时才在屏幕上显示消息。

任何用户都可执行这些检查选项。在 UNIX™ 平台上，必须是用户 informix 或 root 才可显示数据库数据或启动修复选项。在 Windows™ 上，必须是 Informix-Admin 组的成员才可显示数据库数据或启动修复选项。

表 128: *oncheck* 选项和其功能 在第295页将 oncheck选项与其功能相关联。还显示功能等同于 oncheck -c 选项的 SQL 管理 API 命令字符串。

表 128: oncheck 选项和其功能

对象	检查	SQL 管理 API 命令字符串	修复	显示
BLOB 空间智能大对象				-pB
系统目录表	-cc			-pc
数据行，无简单大对象或智能大对象	-cd			-pd
数据行，简单大对象但无智能大对象	-cD			-pD
具有用户定义存取方式的表	-cd, -cD	CHECK DATA		
块和扩展数据块	-ce	CHECK EXTENTS		-pe
索引（键值）	-ci, -cix		-ci -y -pk -y, -pkx -y	-pk
索引（键加行标识）	-cI, -cIx		-cI -y -pK -y, -pKx -y	-pK
具有用户定义存取方式的索引	-ci, -cI			
索引（叶键值）			-pl -y, -plx -y	-pl
索引（叶键加行标识）			-pL -y, -pLx -y	-pL
页（按表或分段）				-pp
页（按块）				-pP
根保留页	-cr, -cR			-pr, -pR
智能大对象的元数据	-cs, -cS			-ps, -pS
空间使用（按表或分段）		CHECK PARTITION PRINT PARTITION		-pt
空间使用（按表，带有索引）				-pT

使用 -y 选项执行修复

使用 -y 选项指示 oncheck 自动执行修复。

如果不使用 -y 选项，那么 oncheck 会在遇到不一致性时提示您，并允许您请求修复。如果指定选项 -n，那么 oncheck 不会提示您，因为该选项指示 oncheck 不要执行修复。

以下示例显示 oncheck 实用程序的自动修复命令：

```
oncheck -cd -y
oncheck -cD -y
oncheck -ci -y
oncheck -cI -y
```

修复智能大对象空间和外部空间中的索引

如果智能大对象空间或外部空间中的索引是使用支持 oncheck -y 选项的存取方式创建的，那么 oncheck 实用程序可修复这些索引。

尽管 oncheck 实用程序不修复分段索引，但用户定义存取方式可以修复它们。有关存取方式所支持的 oncheck 选项的更多信息，请参阅《*SinoDB® DataBlade® API* 程序员指南》或 《*SinoDB®* 虚表索引程序员指南》。

锁定与 oncheck

oncheck 实用程序在表上放置了共享锁，因此在检查完成之前没有任何其他用户可以执行更新、插入或删除。

以下操作期间，oncheck 实用程序在表上放置了共享锁：

- 检查数据时
- 检查索引（使用 -ci、-cI、-pk、-pK、-pl 或 -pL）且表使用页锁定时
- 当 -x 选项与 -ci、-cI、-pk、-pK、-pl 或 -pL 一起指定且表使用行锁定时

如果表不使用页锁定，那么使用 oncheck -ci、-cI、-pk、-pK、-pl 或 -pL 检查索引时，数据库服务器不在表上放置共享锁。在索引检查过程中表上没有共享锁时，其他用户可以在检查过程中更新行。

如果在索引检查过程中使用行锁的表上未放置共享锁，那么 oncheck 实用程序无法进行准确的索引检查。要绝对确保完整的索引检查，请使用 -x 选项执行 oncheck。使用 -x 选项，oncheck 会在表上放置共享锁，那么其他用户在检查完成之前就无法执行更新、插入或删除。

如果在高可用性集群中的辅助服务器上运行 oncheck 实用程序， 其将返回不可信赖的结果。

有关 -x 选项的更多信息，请参阅使用 *-x* 打开锁定 在第311页。有关共享锁和意向共享锁的信息，请参阅《*SinoDB®* 性能指南》。

oncheck 实用程序检查系统目录表时，在其上放置共享锁。在执行修复选项时，它在表上放置互斥锁。

oncheck 实用程序语法

oncheck 实用程序检查指定磁盘结构的不一致性，修复不一致索引结构，和显示磁盘结构信心。

```
oncheck
[  < -FILE option >  ]
{  {  -ce  | -pe  | -cr  | -pr  | -cR  | -pR }  | {  -ci  | -cl  | -pk  | -pK  | -pl  | -pL }
[ -x] database [ : [ owner. ] table [ #index_name ] ]  | {  -cd  | -cD } database [ : [ owner.
] table [{  ,frag_dbs  | %frag_part }] ]  | {  -cc  | -pc } database  | {  -pB  | -pt  | -pT }
database [ : [ owner. ] table [ ,frag_dbs ] ]  | {  -pd  | -pD } {  database [ : [ owner. ] table
[ ,frag_dbs ] rowid ]  | tblspacenumlogical pagenum }  |  -pp {  database  : [ owner. ] table [
,frag_dbs ] rowid  | tblspacenumlogical pagenum }  |  -pPchunknumlogical pagenum  | {  -cs  | -cS }
[ sbspace ]  | {  -ps  | -pS } [ sbspacepartnum pagenum ]  |  -uoption [ (arg_string ) ]  | [{  -V
|  -version }] }
[{  -n  | -y }]
[  -q ]
```

元素	目的	主要注意事项
-cc	检查指定数据库的系统目录表。	请参阅*oncheck -cc* 和 *-pc:* 检查系统目录表 在第300页。
-cd	从指定数据库的表空间、表或分段读取简单大对象之外的所有页，并检查每页的一致性。 还检查使用用户定义存取方法的表。	不检查简单或智能大对象。 请参阅*oncheck -cd* 和 *oncheck -cD* 命令: 检查页 在第300页。

元素	目的	主要注意事项
-cD	与 -cd 相同，但还读取每个 BLOB 页的头并检查其一致性。	检查简单大对象，但不检查智能大对象。 请参阅*oncheck -cd 和 oncheck -cD 命令: 检查页* 在第300页。
-ce	检查每个块可用列表和相应的可用空间，以及每个表空间扩展数据块。还检查智能大对象扩展数据块和智能大对象空间元数据。	oncheck 进程验证磁盘上的扩展数据块是否与描述它们的当前控制信息相对应。 请参阅*oncheck -ce, -pe: 检查空闲块列表* 在第301页。有关背景信息，请参阅下一扩展数据块分配 在第260页。
-ci	检查键值顺序，以及与指定表相关联的所有索引的水平和垂直节点链接的一致性。 还检查使用用户定义存取方式的索引。	请参阅*oncheck -ci 和 -cI: 检查索引节点链接* 在第302页。
-cI	与 -ci 相同，但还检查索引中行标识关联的键值是否与行中的键值相同。	请参阅*oncheck -ci 和 -cI: 检查索引节点链接* 在第302页。
-cr	检查每个根数据库空间保留页的几种情况。	请参阅*oncheck -cr 和 -cR: 检查保留页* 在第303页。
-cR	检查根数据库空间保留页、物理日志页和逻辑日志页。	请参阅*oncheck -cr 和 -cR: 检查保留页* 在第303页。
-cs	检查智能大对象空间的简单大对象和智能大对象空间元数据。	请参阅*oncheck -cs, -cS, -ps, -pS: 检查并显示智能大对象空间* 在第303页。
-cS	检查智能大对象空间的智能大对象和智能大对象空间元数据，以及其扩展数据块。	请参阅*oncheck -cs, -cS, -ps, -pS: 检查并显示智能大对象空间* 在第303页。
sbspace	指示可选的智能大对象空间名称。 如果未提供，那么检查所有智能大对象空间。	无。
-n	指示不执行索引修复，即使检测到错误也是如此。	与索引修复选项（-ci、-cI、-pk、-pK、-pl 和 -pL）一起使用。
-pB	显示描述指定表中 BLOB 空间 BLOB 页的平均充满度的统计信息。	这些统计信息可以衡量数据库或表中个别简单大对象的存储效率。如果未指定表或分段，那么显示整个数据库的统计信息。 请参阅*oncheck -pB: 显示 BLOB 空间统计信息* 在第304页。有关优化 BLOB 空间 BLOB 页的信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章。
-pc	与 -cc 相同，但在检查系统目录表时还显示系统目录信息，包括每个表的扩展数据块的使用情况。	无。
-pd	有十六进制格式显示行。	请参阅*oncheck -pd 和 pD: 以十六进制格式显示行* 在第304页。

元素	目的	主要注意事项
-pD	以十六进制格式显示行，并显示存储在表空间中的简单大对象值或存储在智能大对象空间智能大对象页中的智能大对象和存储在 BLOB 空间 BLOB 页中的简单大对象的头信息。	请参阅*oncheck -pd 和 pD:* 以十六进制格式显示行 在第304页。
-pe	与 -cd 相同，但在检查块可用列表、相应可用空间和每个表空间扩展数据块时还显示块和表空间扩展数据块信息。	请参阅*oncheck -ce, -pe:* 检查空闲块列表 在第301页。
-pk	与 -ci 相同，但在检查指定表上所有索引时还显示其键值。	请参阅*oncheck -pk, -pK, -pl, -pL:* 显示索引信息 在第305页。
-pK	与 -cI 相同，但在检查索引时还显示其键值和行标识。	请参阅*oncheck -pk, -pK, -pl, -pL:* 显示索引信息 在第305页。
-pl	与 -ci 相同，但显示键值。仅检查叶节点索引页。	请参阅*oncheck -pk, -pK, -pl, -pL:* 显示索引信息 在第305页。
-pL	与 -cI 相同，但还显示叶节点索引页的键值。	请参阅*oncheck -pk, -pK, -pl, -pL:* 显示索引信息 在第305页。
-pp	显示逻辑页的内容。	请参阅*oncheck -pp and -pP:* 显示逻辑页的内容 在第306页。
-pP	与 -pp 相同，但需要输入块号和逻辑页号或内部行标识。	请参阅*oncheck -pp and -pP:* 显示逻辑页的内容 在第306页。
-pr	与 -cr 相同，但在检查保留页时还显示保留页信息。	请参阅*oncheck -pr 和 pR:* 显示保留页信息 在第308页。
-pR	与 -cR 相同，但还显示保留页、物理日志页和逻辑日志页的信息。	请参阅*oncheck -pr 和 pR:* 显示保留页信息 在第308页。
-ps	检查和显示智能大对象空间的智能大对象和智能大对象空间元数据。	请参阅*oncheck -cs, -cS, -ps, -pS:* 检查并显示智能大对象空间 在第303页。
-pS	检查和显示智能大对象和智能大对象空间元数据。列出个别智能大对象的扩展数据块和头信息。	请参阅*oncheck -cs, -cS, -ps, -pS:* 检查并显示智能大对象空间 在第303页。
-pt	显示表或分段的表空间信息。	请参阅*oncheck -pt 和 -pT:* 显示表或分段的表空间 在第308页。
-pT	与 -pt 相同，但还显示数据库空间特定于索引的信息和按页类型排列的页分配信息。	请参阅*oncheck -pt 和 -pT:* 显示表或分段的表空间 在第308页。
-q	不显示索引检查和确认消息。	无。
-x	检查和打印索引时在表上放置共享锁。	与 -ci、-cI、-pk、-pK、-pl 或 -pL 选项一起使用。有关完整的信息，请参阅使用 *-x* 打开锁定 在第311页。
-y	检测到错误时修复索引。	无。
-V	显示软件版本号和序列号。	请参阅获取实用程序版本信息 在第289页。
-version	显示构建版本、主机、操作系统、编号、日期，以及 GLS 版本。	请参阅获取实用程序版本信息 在第289页。

元素	目的	主要注意事项
chunknum	指定用于指示特定块的十进制值。	值必须是大于 0 的无符号整数。块必须存在。 执行 -pe 选项可得知哪些块号是与特定数据库空间、BLOB 空间或智能大对象空间相关联。
database	指定要检查其一致性的数据库的名称。	语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南*:* 语法》。
db1	指定包含要检查的数据类型的本地数据库。	（可选）使用 db1@server1 格式指定本地数据库服务器名称。
db2	指定包含要检查的数据类型的远程数据库。	（可选）使用 db2@server2 格式指定远程数据库服务器名称。
frag_dbs	指定包含要检查其一致性的分段的数据库空间名称。	数据库空间必须存在且包含要检查其一致性的分段。语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南*:* 语法》。
index_name	指定要检查其一致性的索引的名称。	索引必须存在指定的表上和数据库中。 语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南*:* 语法》。
logical pagenum	指定用于指示表空间中特定页的整数值。	值必须是介于 0 和 16,777,215（包含 0 和 16,777,215） 之间的无符号整数。值可以表示为无符号整数或以 0x 标识符开始的十六进制数。
object	指定要检查的 DataBlade®、强制转型、运算符、用户定义的数据类型或 UDR 的名称	如果未指定对象名称，那么数据库服务器将比较相同类型（具有相同名称和所有者）的所有对象。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Owner Name 段；有关更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。
pagenum	标识要检查和显示的智能大对象空间元数据部分的页号	无。
partnum	标识要检查和显示的智能大对象元数据部分	无。
rowid	标识要显示其内容的行的 rowid。rowid 是作为 oncheck -pD 输出的一部分来显示。	值必须是介于 0 和 4,277,659,295（包含 0 和 4,277,659,295）之间的无符号整数。值可表示为无符号整数或以 0x 标识符开始的十六进制数。
sbspace	指定要检查其一致性的智能大对象的名称	无。
server	指定数据库服务器名称	如果省略数据库服务器名称，那么 oncheck 使用 INFORMIXSERVER 指定的名称。
table	指定要检查其一致性的表的名称	执行实用程序时表存在。语法必须符合 Table Name 段；有关更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。
tblspacenum	标识要显示其内容的表空间	值必须是介于 0 和 208,666,624（包含 0 和 208,666,624）之间的无符号整数。值可以表示为无符号整数或以 0x 开始的十六进制数。

oncheck -cc 和 -pc: 检查系统目录表

-cc 选项检查有关数据库表、列、索引、视图、约束、存储过程和权限等的信息的系统目录表。
语法:
oncheck { { -cc | -pc } *database* }

oncheck -cc 命令检查下列表:

- systables
- syscolumns
- sysindices
- systabauth
- syscolauth
- sysdepend
- syssyntable
- sysviews
- sysconstraints
- sysams

如果未在 oncheck -cc 中指定数据库名称，那么该命令将对所有数据库检测上面所列的系统目录表。

-pc 选项对系统目录表执行相同的检查，并还会显示系统目录信息，其包括每个表的物理地址、所用的锁定类型、行大小、主键数、扩展数据块使用情形、已分配和使用的页数、表空间分区号，以及索引使用情形。

在执行 oncheck -cc 或 oncheck -pc 之前，请执行 SQL 语句 UPDATE STATISTICS 以确保检查的准确。为检查表，oncheck 将每个系统目录表与其在表空间中的相应条目做比较。

oncheck -cd 和 oncheck -cD 命令: 检查页

使用 oncheck -cd 和 oncheck -cD 命令检查每页的一致性。使用 oncheck -cd -y 或 oncheck -cD -y 命令修复不一致。
语法:
oncheck
{ { -cd | -cD } [-y] *database* [: [*owner.*] *table* [{ *,frag_dbs* | *%frag_part* }]] }

oncheck -cd 命令从指定的数据库、表、分段或多个分段（fragparts）的表空间中读取 BLOB 页和智能大对象页以外的所有页，并检查每页的一致性。该命令将位图页中的条目与页进行比较以验证映射。

oncheck -cD 命令执行与 oncheck -cd 命令相同的检查，并还检查每个 BLOB 页头的一致性。oncheck -cD 命令不会将存储在 BLOB 页头中的开始时间戳记和存储在 BLOB 末尾的结束时间戳记进行比较。使用 oncheck -cD -y 命令清除 BLOB 空间中孤立的简单大对象，这些孤立的简单大对象可能在跨几个日志文件的回滚之后产生。

如果数据库包含分段的表，但未指定分段，那么 oncheck -cd 命令会检查表中的所有分段。如果未指定表，那么该命令检查数据库中的所有表。相较之下，oncheck -pd 命令显示指定页的十六进制转储，但不检查一致性。

对于 oncheck -cd 和 oncheck -cD 命令，oncheck 实用程序在检查表的索引时锁定每个表。要修复这些页，请使用 oncheck -cd -y 或 oncheck -cD -y。

如果表在同一数据库空间中的多个分区上分段，那么 oncheck -cd 和 oncheck -cD 命令将显示分区名称。以下示例显示了在同一数据库空间中的多个分区上有分段的表的典型输出:

```
TBLspace data check for multipart:informix.t1
              Table fragment partition part_1 in DBspace dbs1
              Table fragment partition part_2 in DBspace dbs1
              Table fragment partition part_3 in DBspace dbs1
              Table fragment partition part_4 in DBspace dbs1
```

```
                Table fragment partition part_5 in DBspace dbs1
```

当使用 oncheck -cd 或 oncheck -cD 命令时，可指定 *frag_dbs* 或 *%frag_dbs* 选项之一：

- 当使用 *frag_dbs* 选项时，实用程序会检查数据库空间 frag_dbs 中的所有分段。
- 当使用 *%frag_dbs* 选项时，实用程序仅会检查名为 *frag_part* 的分段（如果创建分段或表时使用 PARTITION 语法）。

虽然可以使用 PARTITION 语法对索引进行分段，但不可能将索引检查限制为一个分段或分区。例如：可以指定 oncheck -cDI my_db:my_tab,data_dbs1 或 oncheck -cDI my_db:my_tab%part1。D（数据）部分检查根据规范被限制，然而 I（索引）不受限制。

示例

以下示例检查 catalog 表中的数据行，其包含简单大对象和智能大对象：

```
oncheck -cD superstores_demo:catalog
```

如果指定单个分段，那么 oncheck 实用程序显示该分段的单个头。对于分段表，每个分段显示一个头：

```
TBLspace data check for stores_demo:informix.tab1
 Table fragment in DBspace db1
```

消息

如果 oncheck 实用程序发现不一致，那么每个找到的表显示一个头。例如：

```
TBLSPACE data check for stores_demo:informix.customer
```

如果 oncheck 实用程序发现不一致，则会显示一条消息。例如：

```
BAD PAGE 2:28: pg_addr 2:28 != bp-> bf_pagenum 2:69
```

物理地址 2:28 表示块号 2 的页 28。

如果使用 DataBlade® 模块提供的存取方法的索引找不到存取方法，那么您会接收到以下消息：

```
-9845 Access method access_method_name does not exist in database.
Ensure that the DataBlade installation was successful.
```

参考

要监视 BLOB 空间 BLOB 页，请参阅 *oncheck -pB: 显示 BLOB 空间统计信息* 在第304页。

相关链接

check data 参数: 检查数据一致性 (SQL 管理 API) 在第600页

oncheck -ce, -pe: 检查空闲块列表

语法：

```
oncheck
{  {  -ce  | -pe } }
```

-ce 选项检查每个空闲块列表，以及相应的可用空间和每个表空间扩展数据块。有关更多信息，请分别参阅下一扩展数据块分配 在第260页和块可用列表页的结构 在第255页。oncheck 进程验证磁盘上扩展数据块与描述它们的当前控制信息相符合。

-pe 选项执行相同的检查，并还在检查过程中显示块和表空间扩展数据块信息。-ce 和 -pe 选项还检查 BLOB 空间、智能大对象扩展数据块，以及智能大对象空间块中的用户数据和元数据信息。

有关使用 oncheck -ce 和 -pe 的信息，请参阅《*SinoDB®* 管理员指南》中的管理磁盘空间。

使用 CHECK EXTENTS 作为 oncheck -ce 的 SQL 管理 API 命令字串符。

相关链接

check extents 参数: 检查扩展数据块一致性 *(SQL* 管理 *API)* 在第601页

oncheck -ci 和 -cI: 检查索引节点链接

使用 oncheck -ci 和 oncheck -cI 命令检查键值顺序，以及与指定表相关联的所有索引的水平和垂直节点链接的一致性。

oncheck -cI 命令还会检查索引中与 rowid 关联的键值是否与行中的键值相同。-cI 选项不在功能索引上交叉检查数据。

语法：

```
oncheck
{ { -ci | -cI } database [ : [ owner . ] table [ #index ] ] }
```

如果不指定索引，那么该选项检查所有索引。如果不指定表，那么该选项检查数据库中的所有表。

相同的 -ci 修复选项与 -cI 是一起可用的。如果 oncheck -ci 或 oncheck -cI 检测到不一致，那么它会提示您确认修复问题索引。如果指定 -y（是）选项，那么将会自动修复索引。如果指定 -n（否）选项，那么将会报告问题但不进行修复；不出现任何提示。

如果 oncheck 未发现任何不一致，那么以下消息出现：

```
validating indexes......
```

消息显示 oncheck 正在检查的索引的名称。

> 注: 使用 oncheck 重新建购索引可能会消耗很长时间。如果使用 SQL 语句 DROP INDEX 和 CREATE INDEX 删除并创建索引，那么处理速度通常比较快。

以下示例检查 customer 表上的所有索引：

```
oncheck -cI -n stores_demo:customer
```

以下示例检查 customer 表上的 zip_ix 索引：

```
oncheck -cI -n stores_demo:customer#zip_ix
```

如果索引在同一数据库空间中的多个分区上分段，那么 oncheck -ci 和 oncheck -cI 命令将显示分区名称。以下示例显示在同一数据库空间中的多个分区上有分段的索引的典型输出：

```
Validating indexes for multipart:informix.t1...
                Index idx_t1
                  Index  fragment partition part_1 in DBspace dbs1
                  Index  fragment partition part_2 in DBspace dbs1
                  Index  fragment partition part_3 in DBspace dbs1
                  Index  fragment partition part_4 in DBspace dbs1
                  Index  fragment partition part_5 in DBspace dbs1
```

在默认情况下，当使用 oncheck -ci 或 oncheck -cI 命令检查索引时，数据库服务器不会在表上放置共享锁，除非表使用页锁定。为了绝对确保完全的索引检查，您可以在执行 oncheck -cior oncheck -cI 时使用 -x 选项。使用 -x 选项，oncheck 会在表上放置共享锁，这样在检查完成之前其他用户都不能执行更新、插入或删除。有关使用带有 -x 选项的 oncheck -ci 和 oncheck -cI 命令的信息，请参阅使用 *-x* 打开锁定 在第311页。

当在外部索引上执行 oncheck 时，用户定义的存取方法负责索引的检查和修复。如果使用用户定义的存取方法的索引找不到该存取方法时，数据库服务器会报告错误。oncheck 实用程序不修复外部索引的不一致。对于包含多种索引类型的表，不应使用 oncheck -cI。

oncheck 实用程序在检查索引时需要排序空间。所需的排序空间量与建构该索引所需的空间量是相同的。有关所需的临时空间量的计算的信息，请参阅《*SinoDB*® 性能指南》中的估算用于索引构建的临时空间。如果您接收到错误消息 "no free disk space for sort"，那么您必须估算所需的临时空间量，并使该空间可用。

有关索引的更多信息，请参阅 *B* 型树索引页的结构 在第264页。

oncheck -cr 和 -cR: 检查保留页

语法：

```
oncheck
{ { -cr | -cR } }
```

-cr 选项按以下方式检查每个根数据库空间保留页：

- 以 PAGE_CONFIG 保留页验证 ONCONFIG 文件的内容。
- 确保所有块可以打开、块不重叠，且块大小是正确的。

-cR 选项执行相同的检查和验证，并还检查所有的逻辑日志和物理日志页的一致性。-cr 选项是比较快的，因为它不检查日志文件页。

如果您已更改了配置参数的值（通过onparams、onmonitor、onspaces 或通过编辑配置文件），但还未重新初始化共享内存，那么oncheck -cr 和 oncheck -cR 将会检测到不一致并返回错误消息。

如果 oncheck -cr 在执行之后不显示任何错误消息，那么您可以认为前述列表中的所有三项都已成功通过了检查。

有关保留页的更多信息，请参阅保留页 在第253页。

oncheck -cs, -cS, -ps, -pS: 检查并显示智能大对象空间

语法：

```
oncheck
{ { -cs | -cS } [ sbspace ] | { -ps | -pS } [ sbspacepartnum pagenum ] }
```

-cs 选项检查智能大对象空间。-ps 选项检查智能大对象空间和扩展数据块。

-cS 选项验证并显示智能大对象空间的元数据。

-ps 选项检查智能大对象空间和扩展数据块。如果未指定智能大对象空间名称，那么这些选项检查所有的智能大对象空间。

-pS 选项验证并显示智能大对象空间的元数据，并且还会列出智能大对象的扩展数据块和头信息。

如果未指定智能大对象空间名称，那么将会检查所有智能大对象空间。以下示例检查并显示 test_sbspace 的元数据：

```
oncheck -ps test_sbspace
```

如果使用 -cs 或 -ps 选项将 rootdbs 指定为智能大对象空间的名称，那么 oncheck 将检查根数据库空间。

有关使用 -cs、-cS、-ps 和 -pS 选项的更多信息，请参阅《*SinoDB*® 管理员指南》。

oncheck -pB: 显示 BLOB 空间统计信息

语法：
oncheck { -pB } *database* [: [*owner.*] *table* [,*frag_dbs*]]
{ }

-pB 选项显示描述指定表中 BLOB 空间 BLOB 页的平均充满度的统计信息。这些统计信息衡量数据库或表中的个别简单大对象的存储效率。如果未指定表或分段，那么该选项将显示整个数据库的统计信息。有关更多信息，请参阅《*SinoDB*® 管理员指南》中管理磁盘空间中的优化 BLOB 空间 BLOB 页大小的部分。

oncheck -pd 和 pD: 以十六进制格式显示行

语法：
oncheck
{ { -pd | -pD } { *database* [: [*owner.*] *table* [,*frag_dbs*] [,*frag_part*] *rowid*] |
tblspacenumlogical pagenum } }

-pd 选项将数据库、表、分段、分段分区（fragpart），和特定的 rowid 或表空间编号以及逻辑页编号作为输入。在每种情况中，-pd 会打印页头信息并显示以十六进制和 ASCII 格式指定的数据库对象（数据库、表、分段、内部 rowid 或页号）的指定行。不执行任何一致性检查。

元素	用途	关键注意事项
database	指定要检查其一致性的数据库的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。
frag_dbs	指定包含想要检查其一致性的分段的数据库空间的名称	数据库空间必须存在，并且包含想要检查其一致性的分段。 语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。
frag_part	指定分段分区	对于使用基于表达式或循环分布模式的分段表或索引，您可以在单个数据库空间内创建多个分区（这些分区是表或索引的页的集合）。此分区被称为分段分区或 *fragpart*。
logical pagenum	指定用于标识表空间中特定页的整数值。	值可以表示为无符号整数或以 0x 标识符开始的十六进制数。 值必须是介于 0 和 16,777,215 之间（包含 0 和 16,777,215）的无符号整数。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Owner Name 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。
rowid	标识要显示其内容的行的 rowid。rowid 显示为 oncheck -pD 输出的一部分。	值必须是介于 0 和 4,277,659,295 之间（包含 0 和 4,277,659,295）的无符号整数。 值可以表示为无符号整数或以 0x 标识符开始的十六进制值。
table	指定要检查其一致性的表的名称	当执行该实用程序时，表应存在。 语法必须符合 Table Name 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。

元素	用途	关键注意事项
tblspacenum	标识要显示其内容的表空间。	值必须是介于 0 和 208,666,624 之间（包含 0 和 208,666,624）的无符号整数。 值可以表示为无符号整数或以 0x 标识符开始的十六进制值。

如果指定了内部 rowid（以十六进制值表示），那么该 rowid 映射到特定页，且打印该页中的所有行。

如果指定了逻辑页号（以十进制表示），那么打印具有该逻辑页号的表空间号的所有行。

如果指定了分段，那么打印该分段中的所有行，以及它们的 rowid、转发指针和页类型。

如果指定了表，那么打印该表中的所有行，以及它们的 rowid、转发指针和页类型。

如果指定了数据库，那么打印该数据库中所有表的所有行。打印存储在数据行中的 TEXT 和 BYTE 列描述符，但不打印 TEXT 和 BYTE 数据本身。

-pD 选项打印与 -pd 相同的信息。此外，-pD 打印存储在表空间中的 TEXT 和 BYTE 值或存储在 BLOB 空间 BLOB 页中简单大对象的头信息。以下示例显示 oncheck -pd 和 oncheck -pD 命令的不同选项：

```
oncheck -pd stores_demo:customer,frgmnt1
oncheck -pd stores_demo:customer
oncheck -pD stores_demo:customer 0x101
```

以下示例显示 oncheck -pD 命令的部分输出：

```
oncheck -pD multipart:t1 :

TBLspace data check for multipart:informix.t1
                  Table fragment partition part_1 in DBspace dbs1
page_type  rowid    length fwd_ptr
HOME       101      24     0
    0:  0  0  0  a 47 48 49 20 20 20 20 20 20 20 20 20   ....GHI
   16: 20 20 20 20 20 20 20 20
                                      ........
```

oncheck -pk, -pK, -pl, -pL: 显示索引信息

语法：
oncheck
{ { -pk | -pK | -pl | -pL } [x] *database* [: [*owner.*] *table* [#*index_name*]] }

-pk 选项执行与 -ci 选项相同的检查，此外，在检查时，还显示所指定表上所有索引的键值。

-pK 选项执行与 -cI 选项相同的检查，此外，在检查时还显示键值和 rowid。

-pl 选项执行与 -ci 相同的检查，并显示键值，但仅检查叶节点索引页，忽略根和分支节点页。

-pL 选项执行与 -cI 选项相同的检查，并显示键值和 rowid，但仅检查叶节点索引页，忽略根和分支节点页。.

元素	用途	关键注意事项
database	指定要检查其一致性的数据库的名称	语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南*:* 语法》。
index_name	指定要检查其一致性的索引的名称	索引必须存在所指定的数据库和表中。

元素	用途	关键注意事项
		语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Owner Name 段；请参阅《*SinoDB® SQL* 指南: 语法》。
table	指定要检查其一致性的表的名称	执行实用程序时，表应存在。 语法必须符合 Table Name 段；请参阅《*SinoDB® SQL* 指南: 语法》。
-x	检查和打印索引时在表上放置共享锁	请参阅使用 *-x* 打开锁定 在第311页，以获取全部信息。

如果任一 oncheck 选项检测到不一致，那么将会提示您确认修复问题索引。如果指定 -y（是）选项，那么自动修复索引。如果指定 -n（否）选项，那么将会报告问题但不修复；不出现任何提示。

以下示例显示有关 customer 表上所有索引的信息：

```
oncheck -pl -n stores_demo:customer
```

以下示例显示有关 customer 表上所创建的索引 zip_ix 的信息：

```
oncheck -pl -n stores_demo:customer#zip_ix
```

缺省情况下，当使用 oncheck -pk、-pK、-pl 或 -pL 检查索引时，数据库服务器不在表上放置共享锁，除非该表使用页锁定。为绝对确保完全索引检查，可以使用 -x 选项执行 oncheck -pk、oncheck -pK、oncheck -pl 或 oncheck -pL。 使用 -x 选项，oncheck 在表上放置共享锁，这样在检查完成之前其他用户都不能执行更新、插入或删除。有关使用 -x 选项的更多信息，请参阅使用 *-x* 打开锁定 在第311页。

有关 oncheck -ci 的更多信息，请参阅 *oncheck -ci* 和 *-cI*: 检查索引节点链接 在第302页。有关索引页的更多信息，请参阅 *B* 型树索引页的结构 在第264页。

oncheck -pp and -pP: 显示逻辑页的内容

语法：
oncheck
{ -pp { *database* : [*owner* .] *table* [{ , *frag_dbs* | % *frag_part* }] *rowid* | *tblspacenum* *logical pagenum* } | -pP *chunknum* *logical pagenum* }

元素	用途	关键注意事项
database	指定要检查一致性的数据库的名称	语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。
chunknum	指定用于指示特定块的十进制值	值必须是大于 0 的无符号整数。块必须存在。
frag_dbs	指定包含要检查其一致性的分段的数据库空间的名称	数据库空间必须存在，并且包含要检查一致性的分段。 语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。
frag_part	指定要检查的分段的分区名称。对于在同一数据库空间中创建表的多个分段的情况是很有用的。	对于分段表或使用基于表达式或循环分布模式的索引，您可以在单个数据库空间中创建多个分区（表

元素	用途	关键注意事项
		或索引的页的集合）。这种分区称为分段分区 或 *fragpart*。
logical pagenum	指定用于指示表空间中特定页的整数值。	值可以表示为无符号整数或以 0x 标识符开始的十六进制数。 值必须是介于 0 和 16,777,215 之间（包含 0 和 16,777,215）的无符号整数。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Owner Name 段；请参阅《*SinoDB® SQL* 指南: 语法》。
rowid	标识要显示其内容的行的 rowid。rowid 显示为 oncheck -pD 输出的一部分。	值必须是介于 0 和 4,277,659,295 之间（包含 0 和 4,277,659,295）的无符号整数。 值可表示为无符号整数或以 0x 标识符开始的十六进制数。
table	指定要检查其一致性的表的名称	执行实用程序时表应存在。 语法必须符合 Table Name 段；请参阅《*SinoDB® SQL* 指南: 语法》。
tblspacenum	标识要显示其内容的表空间	值必须是介于 0 和 208,666,624 之间（包含 0 和 208,666,624）的无符号整数。 值可表示为无符号整数或以 0x 标识符开始的十六进制数。

-pp 选项有以下语法变化：

调用	说明
oncheck -pp tblspc lpn <pages>	显示使用了表空间号和逻辑页号的逻辑页的内容。您还可指定一个可选参数，指定要打印的页数。
oncheck -pp ***tblspc lpn -h***	只显示使用了表空间号和逻辑页号的逻辑页的头。
oncheck -pp ***database:table rowid***	显示使用了数据库名称、表名称和 SinoDB® 内部 rowid的逻辑页的内容。您可使用 oncheck -pD 获取这个内部的 rowid。这个内部的 rowid 不是在使用 CREATE TABLE tabname WITH ROWIDS 语句创建的表中指定的序列 rowid。有关更多信息，请参阅 *Rowid* 的定义 在第262页。

页内容以 ASCII 格式显示。显示还包含了页上槽表条目数。以下示例显示 oncheck -pp 命令的不同调用：

```
oncheck -pp stores_demo:orders 0x211  # database:owner.table,   # fragment rowid
oncheck -pp stores_demo:informix.customer,frag_dbspce1 0x211
oncheck -pp 0x100000a 25 # specify the tblspace number and # logical page number
```

-pP 选项提供以下语法变化：

调用	说明
oncheck -pP chunk# offset pages	使用块号和偏移量，显示逻辑页的内容。还可指定一个可选参数，指定要打印的页数。
oncheck -pP ***chunk# offset -h***	使用块号和偏移量，只显示逻辑页的头。

注: 块页的输出以十进制格式显示 start 和 length 字段。

以下示例显示了使用 onstat -pP 命令的典型输出:

```
oncheck -pP 1 5 2
addr          stamp          nslots          flag       type          frptr       frcnt       next
      prev
stamp           100005           250181           2          1000       ROOTRSV           320
 1716       0
0       250181       slot       ptr       len       flg
...
addr       stamp          nslots          flag       type          frptr       frcnt       next
  prev
stamp       100005    6       250182       2       1000       ROOTRSV       128       1908       0
   0
250182       slot       ptr       len       flg       1       24       56       0
2       80       48         0
```

oncheck -pr 和 pR: 显示保留页信息

语法:

```
oncheck
{ { -pr | -pR } }
```

-pr 选项执行与 oncheck -cr 命令相同的检查，并显示保留页信息。

-pR 选项执行与 onchdeck -cR 命令相同的检查，并显示保留页信息，以及有关逻辑日志和物理日志页的详细信息（标记活动物理日志页的开始和结束）。

如果您已通过编辑配置文件更改了配置参数的值，但尚未重新初始化共享内存，那么 oncheck -pr 和 oncheck -pR 命令将检测到不一致并返回错误消息。

有关 oncheck -pr 输出的列表和解释，请参阅保留页 在第253页。有关 -cr 选项的描述，请参阅 *oncheck -cr 和 -cR: 检查保留页* 在第303页。

oncheck -pt 和 -pT: 显示表或分段的表空间

oncheck -pt 和 oncheck -pT 选项打印特定表或分段的表空间报告。唯一区别在于 oncheck -pT 打印更多信息，包括一些特定于索引的信息。

语法

```
oncheck
{ { -pt | -pT } database [ : [ owner . ] table [ , frag_dbs ] ] }
```

表 129: oncheck -pt 和 oncheck -pT 命令的选项

元素	用途	关键注意事项
database	指定要检查其一致性的数据库的名称	语法必须符合 Identifier 段；请参阅 *Identifier*。
frag_dbs	指定包含要检查其一致性的分段的数据库空间的名称	数据库空间必须存在，并且包含要检查其一致性的分段。 语法必须符合 Identifier 段；请参阅 *Identifier*。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Owner Name 段；请参阅 *Owner name*。
table	指定要检查其一致性的表的名称	表必须存在。

元素	用途	关键注意事项
		语法必须符合 Identifier 段；请参阅 *Identifier*。

-pt 选项打印指定数据库和名称的表或分段的表空间报告。如未指定表，该选项显示数据库中所有表的这个信息。该报告包含一般分配信息，包括最大行大小、键数量、扩展数据块数量和其大小、每个扩展数据块所分配和使用的页、当前序列值，以及表的创建日期。-pt 输出打印表空间的页大小、逻辑页的页数（分配的页、使用的页和数据页）。

TBLspace Flags 字段显示有关表空间配置的信息，包括表空间是用于 Enterprise Replication 还是时间序列数据。

Extents 字段列出表的表空间 tblspace 条目的物理地址和第一个扩展数据块的第一页的地址。扩展数据块列表显示每个扩展数据块中逻辑和物理页的数量。

-pT 选项打印与 -pt 选项相同的信息。此外，-pT 选项显示：

- 特定于索引的信息
- 按页类型排列的页分配信息（对于数据库空间）
- 表或表分段中任何压缩行的数量，以及表或表分段行的压缩百分比。

  如果表或分段行未压缩，那么在输出中不会显示 "Compressed Data Summary" 节。

计划何时运行 -pT 选项，因为它会强制分区的完整扫描。

-pt 和 -pT 的输出包含 Number of pages used 的列表。这个字段在输出中显示的值永远不会递减，因为分配给表空间作为扩展数据块一部份的磁盘空间，即使在通过删除行而释放空间之后，仍然保持专用于该扩展数据块。有关当前使用页数的精确计算，请参阅 -pT 选项提供的有关表空间使用情况（按页类型组织）的详细信息。

oncheck -pt 输出示例

以下示例显示 oncheck -pt 命令的输出：

```
TBLspace Report for testdb:tab1

Physical Address              2:10
Creation date                 10/07/2004 17:01:16
TBLspace Flags                801        Page Locking
                                             TBLspace use 4 bit bit-maps
Maximum row size              14
Number of special columns     0
Number of keys                0
Number of extents             1
Current serial value          1
Pagesize (k)                  4
First extent size             4
Next extent size              4
Number of pages allocated     340
Number of pages used          337
Number of data pages          336
Number of rows                75806
Partition partnum             2097154
Partition lockid              2097154

Extents
       Logical Page     Physical Page        Size Physical Pages
                  0             2:106         340        680
```

oncheck -pT 输出示例

以下示例显示 oncheck -pT 命令的输出：

```
TBLspace Report for database_a:nilesh.table_1a

                Table fragment partition dbspace1 in DBspace dbspace1

    Physical Address              3:5
    Creation date                 03/21/2009 15:35:47
    TBLspace Flags                8000901    Page Locking
                                             TBLspace contains VARCHARS
                                             TBLspace use 4 bit bit-maps
                                             TBLspace is compressed
    Maximum row size              80
    Number of special columns     1
    Number of keys                0
    Number of extents             1
    Current serial value          100001
    Current SERIAL8 value         1
    Current BIGSERIAL value       1
    Current REFID value           1
    Pagesize (k)                  2
    First extent size             8
    Next extent size              8
    Number of pages allocated     24
    Number of pages used          22
    Number of data pages          14
    Number of rows                500
    Partition partnum             3145730
    Partition lockid              3145730

    Extents
         Logical Page     Physical Page        Size Physical Pages
                    0          3:16053          24         24

Type                  Pages      Empty  Semi-Full       Full  Very-Full
    ----------------- ---------- ---------- ---------- ---------- ----------
    Free                       9
    Bit-Map                    1
    Index                      0
    Data (Home)               14
    Data (Remainder)           0          0          0          0          0
                      ----------
    Total Pages               24

    Unused Space Summary

        Unused data bytes in Home pages               1177
        Unused data bytes in Remainder pages             0

    Home Data Page Version Summary

                Version                                 Count

                      0 (current)                         14

    Compressed Data Summary

    Number of compressed rows and percentage of compressed rows  500 100.00
```

相关链接

TBLTBLFIRST 配置参数 在第186页

check partition 参数: 检查分区一致性 *(SQL* 管理 *API)* 在第601页
print partition 参数: 打印分区信息 *(SQL* 管理 *API)* 在第672页
print partition 参数: 打印分区信息 *(SQL* 管理 *API)* 在第672页
表空间 *tblspace* 条目 在第256页
《*SinoDB* 性能指南》: *DDL* 操作的定点变更性能
《*SinoDB* 迁移指南》: 解决未完成的定点变更操作
《*SinoDB* 管理员指南》: 使用 *oncheck -pT* 监视数据库空间中的简单大对象

使用 -x 打开锁定

-x 选项可附加到 -ci、-cI、-pk、-pK、-pl 和 -pL 选项后，用于在受影响的表上放置共享锁。在表被锁时，当 oncheck 检查或打印索引时，其他用户都无法执行插入、更新和删除。对于带有行锁定的表，在未使用 -x 选项时，oncheck 仅在该表上放置 IS（意向共享）锁，该锁会阻止在检查过程中删除表或索引之类的操作。

例如：以下样本命令指示 oncheck 在验证键值顺序、验证水平链接和确认索引中没有节点出现两次时，锁定 customer 表的索引：

```
oncheck -cix stores_demo:customer
```

当指定 -x 选项时，oncheck 锁定使用行锁定的表的索引。如果 oncheck 检测到页锁定方式，它会显示告警消息，并在表上放置共享锁。

使用 -u 将特殊参数发送给存取方法

您可以使用 -u 选项将特殊参数发送给存取方式。可能的参数视存取方式而定。例如：R 树存取方式支持 display 选项，如以下示例所示：

```
oncheck -pl -u "display"
```

使用逗号分隔参数字串符中的多个参数。

有关存取方式的有效参数的信息，请参考您的存取方式的使用手册。

退出时的返回码

oncheck 实用程序在退出时返回以下代码。

```
GLS failures:-1
Invalid srial/key:2
Onconfig access error:2
Invalid onconfig settings:2
Invalid arguments to oncheck:2
Error connecting database server:1
Warning reported by oncheck:1
error detected by oncheck:2
no errors detected by oncheck:0
```

仅 Windows™：

```
Not properly installed:1
Authentication error:2
```

onclean 实用程序

当使用 onmode 正常关闭数据库服务器失败或无法重新启动数据库服务器时，使用 onclean 实用程序强制关闭数据库服务器。onclean 实用程序尝试清理共享内存、信号量，并停止数据库服务器虚拟进程。

在 UNIX™ 和 Linux™ 上，您必须以 root 或 informix 用户身份来运行 onclean 命令。在 Windows™ 上，您必须是 Informix-Admin 组的成员才可运行该命令。

语法

```
onclean
{ [  < -FILE option >   ] | [  -k ] [  -y ]  | [{  -V  | -version }] }
```

表 130: onclean 命令的语法元素

元素	用途
-k	通过停止数据库服务器虚拟进程并尝试清理剩余的信号量和共享内存段（即使它们仍在运行）来关闭联机的服务器。
-V	显示简短版本信息。
-version	显示完整版本信息。
-y	无确认提示。

用法

仅当 onmode 实用程序无法关闭数据库服务器或您无法重新启动数据库服务器时，使用 onclean 实用程序来停止数据库服务器。可能数据库服务器以不受控的方式关闭并且无法恢复，或者挂起。如果数据库服务器重新启动失败，数据库服务器的先前实例仍然附加在共享内存段。检查消息日志以查看数据库服务器是否异常关闭。onclean 实用程序停止所有关 oninit 进程并尝试移除所以共享内存段和记录在 $INFORMIXDIR/etc/.conf.$INFORMIXSERVER 文件中的信号量。

> 注意: 请谨慎使用 onclean 实用程序。当执行 onclean 时，任何搁置的事务和进程都无法完成，并且将用户会话突然断开。当数据库服务器重新启动时，它会回滚事务。

INFORMIXDIR、INFORMIXSERVER、INFORMIXSQLHOSTS 和 ONCONFIG 环境变量必须设置为有效值才可运行该实用程序。

onclean 命令的使用取决于具体情况：

- 如果您不确定数据库服务器是否脱机，那么使用不带选项的 onclean 命令。如果数据库服务器仍联机，那么将出现消息指示您执行 onlcean -k 命令。
- 如果数据库服务器是脱机的，那么使用 onclean 命令。
- 如果数据库服务器是联机的，并且您确定想要强制关闭它，那么使用 onclean -k 命令。

您仅可使用 onclean 命令关闭本地数据库服务器，不能使用它关闭远程数据库服务器。onclean 实用程序不应用于关闭整个高可用性集群或远程数据库服务器。

onclean 实用程序可能无法清除在每种情况中数据库服务器所使用的共享内存段。onclean 实用程序尝试仅终止 oninit 进程。在以下情况中，onclean 实用程序不会成功：

- 如果在运行 onclean 命令之前非数据库服务器进程附加到共享内存段，那么 onclean 实用程序不会停止该进程以移除共享内存段。
- 当应用程序湖数据库服务器实用程序连接到网络端口时，onclean 可能无法保证干净的服务器启动。如果用户试图在相同的网络端口初始化数据库服务器实例，那么数据库服务器无法启动监听线程并启动失败。onclean 实用程序不会停止应用程序释放网络端口。

您可以使用 onshutdown 脚本自动关闭数据库服务器，如需要该脚本会调用 onclean -ky 命令。

返回码

0

成功

1

由于以下问题之一而失败：

- 不正确的环境变量设置
- 以不正确的权限运行 onclean 命令
- 不正确的命令语法
- 损坏的信息
- 在仍然联机的服务器上未使用 -k 选项来运行 onclean 命令

2

由于 onclean 使用的一个或多个 OS 系统调用返回错误而失败。

相关链接

使用 *-k* 选项使数据库服务器处于脱机模式 在第340页

onshutdown 脚本

使用 onshutdown 脚本自动关闭数据库服务器。该脚本会尝试正常关闭服务器。如果服务器在指定时间后未关闭，那么该脚本会强制该服务器关闭。

onshutdown 脚本先运行 onmode -ky 命令。在指定的等待时间之后，该脚本运行 onclean -ky 命令。

在 UNIX™ 和 Linux™ 上，您必须是 root 或 informix 用户才可执行 onsutdown 脚本。在 Windows™ 上，您必须是 Informix-Admin 组的成员才可执行 onshudown 脚本。

语法

{ onshutdown.sh | onshutdown.bat }
timeout

表 131: onshutdown 脚本的语法元素

元素	用途
timeout	在 onmode -ky 命令运行之后，onclean -ky 命令运行之前的秒数。 必须是从 10 到 60 之间的正整数。默认值为 30 秒。

用法

仅在强制关闭数据库服务器时才使用 onshutdown 脚本。

> 注意： 请谨慎使用 onshutdown 脚本。如果该脚本需要运行 onclean -ky 命令，那么任何搁置的事务和进程都无法完成，并且将用户会话突然断开。

INFORMIXDIR、INFORMIXSERVER、INFORMIXSQLHOSTS 和 ONCONFIG 环境变量必须设置为有效值才可运行此实用程序。

您可以仅使用 onshutdown 脚本关闭本地数据库服务器。您不可以使用该脚本关闭远程数据库服务器。onshutdown 脚本不应用于关闭整个高可用性集群或远程数据库服务器实例。

onshutdown 脚本有 10 秒的时间，在这个时间中间可以被中止。

相关链接

使用 *-k* 选项使数据库服务器处于脱机模式 在第340页

oncmsm 实用程序

使用 oncmsm 实用程序启动或关闭连接管理器、将新的配置文件加载到连接管理器以更改连接管理器的设置，或者更新配置文件的格式。

语法

UNIX 语法图：

oncmsm
{ [-c*configuration_file*] [-n*new_configuration_file*] | { -r | -k }
connection_manager_name }

Windows 语法图：

oncmsm
{ { -i -c *configuration_file* | [-c*configuration_file*] -n*new_configuration_file* | { -r |
-k | -u } *connection_manager_name* | *connection_manager_name* } | }

元素	用途	关键注意事项
-c	启动连接管理器或将配置文件转换成当前连接管理器的格式。	
connection_manager_name	指定连接管理器实例的名称。	
-i	将连接管理器安装为一个 Windows™ 服务。	此选项仅在 Windows™ 平台上有效。
-k	关闭特定的连接管理器实例。	
-n	指定转换的配置文件的名称。	
new_configuration_file	输出到 $INFORMIXDIR/etc 目录的文件名，作为格式转换过程的一部分。	
configuration_file	位于 $INFORMIXDIR/etc 目录中的配置文件的名称。	如未指定配置文件，那么连接管理器将尝试加载 $INFORMIXDIR/etc/cmsm.cfg。
-r	重新加载连接管理器设置而无需停止与重新启动连接管理器。	
-u	卸载连接管理器 Windows™ 服务。	此选项仅在 Windows™ 平台上有效。

用法

从命令行执行 oncmsm 实用程序，以初始化连接管理器。您可在连接管理器正在运行时添加、更改或删除服务级别协议（SLA），然后重新加载配置文件。

仅 UNIX: 以下用户可执行 oncmsm 实用程序：

- 用户 informix
- 用户 root，如果该用户有连接 sysadmin 数据库的权限
- DBSA 组成员，如果该用户有连接 sysadmin 数据库的权限

仅 Windows: 以下用户可执行 oncmsm 实用程序：

- Informix-Admin 组成员
- 用户 administrator，如果该用户有连接 sysadmin 数据库的权限
- DBSA 组成员，如果该用户有连接 sysadmin 数据库的权限

在启动 oncmsm 实用程序之前，必须将其安装成为一个服务。

oncmsm 实用程序可以两种方式启动：

- 执行 oncmsm 命令。
- 单击 开始 >控制面板 >管理工具 >服务，然后启动 oncmsm。

如果您使用多个连接管理器，那么您可以执行 onstat -g cmsm 来显示各连接管理器实例的名称。

示例 1：启动连接管理器（UNIX™）

对于以下示例，连接管理器的 configuration_file_1 存在于 $INFORMIXDIR/etc 目录中。要启动连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -c configuration_file_1
```

连接管理器启动。

示例 2：启动连接管理器（Windows™）

对于以下示例，连接管理器的 configuration_file_1 存在于 $INFORMIXDIR/etc 目录中。要启动连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -i -c configuration_file_2
oncmsm connection_manager_2
```

名为 connection_manager_2 的连接管理器启动。

示例 3：停止连接管理器

要停止连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -k connection_manager_3
```

名为 connection_manager_3 的连接管理器停止。

示例 4：重新加载连接管理器设置

对于以下示例，名为 connection_manager_4 的连接管理器的 $INFORMIXDIR\etc \configuration_file_4 已更改。要更新连接管理器的设置，请在 connection_manager_4 所安装的计算机上执行以下命令：

```
oncmsm -r connection_manager_4
```

示例 5：将连接管理器配置文件转换成当前格式

对于以下示例，名为 cmsm.cfg 的连接管理器配置文件存在于 $INFORMIXDIR/etc 目录中。要启动连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -n configuration_file_5
```

oncmsm 实用程序将 cmsm.cfg 转换成当前配置文件格式，并将名为 configuration_file_5 的文件输出到 $INFORMIXDIR/etc/ 目录中。

示例 6：将特定的连接管理器配置文件转换成当前格式

对于以下示例，名为 configuration_file_4 的连接管理器文件存在于 $INFORMIXDIR/etc 目录中。要启动连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -c configuration_file_6 -n configuration_file_7
```

oncmsm 实用程序将 configuration_file_6 转换为当前配置文件格式，并将名为 configuration_file_7 的文件输出到 $INFORMIXDIR/etc/ 目录中。

示例 7：卸载连接管理器（Windows™）

对于以下示例，您已安装了名为 connection_manager_4 的连接管理器作为一个 Windows™ 服务。要卸载连接管理器，请在连接管理器所安装的计算机上执行以下命令：

```
oncmsm -u connection_manager_4
```

oncmsm 实用程序卸载连接管理器。

相关链接

《SinoDB SQL 指南: 参考》: CMCONFIG 环境变量
《SinoDB 管理员指南》: 通过连接管理器进行连接管理
《SinoDB 管理员指南》: 启动 UNIX 和 Linux 上的连接管理器
《SinoDB 管理员指南》: 在 Windows上启动连接管理器

onconfig_diff 实用程序

使用 onconfig_diff 实用程序对两个 onconfig 文件进行比较。

语法

onconfig_diff
{ -d | [-c -f*filepath_1* -s*filepath_2*] }

元素	描述
-d	将当前 onconfig 设置与默认设置进行比较。
-c	将一个 onconfig 文件与另一个进行比较。
-f *filepath_1*	指定要进行比较的第一个文件的名称。如果该文件不存在 $INFORMIXDIR/bin 目录中，请提供文件的路径。
-s *filepath_2*	指定要进行比较的第二个文件的名称。如果该文件不存在 $INFORMIXDIR/bin 目录中，请提供文件的路径。

用法

执行 onconfig_diff 实用程序来比较两个不同的 onconfig 文件。onconfig_diff 实用程序位于 $INFORMIXDIR/bin 目录中。

要比较的两个文件必须在相同的目录中。

以下为可以使用该实用程序的一些方式：

- 将当前 onconfig 与相同版本的 onconfig.std 进行比较。

- 将当前 onconfig 与新版本的 onconfig.std 进行比较。
- 将不同服务器上的两个 onconfig 文件进行比较。

示例

此示例中，将 onconfig.std 文件与 onconfig.production 进行比较：

```
$ onconfig_diff -c -f onconfig.std -s onconfig.production
```

此为该命令的输出：

```
==========================================
File 1: onconfig.std
File 2: onconfig.production
==========================================
Parameters Found in File 1, not in File 2
==========================================

FULL_DISK_INIT  0

NETTYPE         ipcshm,1,50,CPU

NUMFDSERVERS    4
...
==========================================
Parameters Found in File 2, not in File 1
==========================================

JVPJAVAHOME    $INFORMIXDIR/extend/krakatoa/jre

...
==========================================
Parameters Found in both files, but different
==========================================
ROOTPATH

File 1: $INFORMIXDIR/tmp/demo_on.rootdbs
File 2: /usr2/support/grantf/g1150fc8/rootdbs

LOGFILES

File 1: 6
File 2: 10

LOGSIZE

File 1: 10000
File 2: 3000
...
```

相关链接

更改 *onconfig* 文件 在第42页

ondblog 实用程序

使用 ondblog 实用程序更改一台或多台数据库的日志记录模式。

ondblog
{ buf | unbuf | nolog | ansi | cancel | -V | -version }

[{ , *db_list* | -f *dbfile* }]

元素	用途	关键注意事项
buf	设置此日志记录模式，以便在事务信息写入逻辑日志之前先将其写入缓冲区	无。
unbuf	设置此日志记录模式，以便在数据写入逻辑日志之前不将其写入缓冲区	无。
nolog	设置此日志记录模式，以便不记录任何数据库事务	无。
ansi	将数据库日志记录更改为符合 ANSI 的模式	一旦创建或将数据库转换为 ANSI 模式，那么就不能将其更改回任何其他的日志记录模式。
cancel	在下一个 0 级备份发生之前取消日志记录模式更改请求	无。
-f *dbfile*	更改文本文件中列出的数据库（每行一个）的日志记录状态，该文本文件的路径名由 *dbfile* 指定。	如果数据库列表很长或经常使用，那么此命令是很有用的。
db_list	指定要更改其日志记录状态的数据库列表的名称，该列表中数据库以空格分隔	如果未指定，那么数据库服务器所管理的所有数据库都会被更改。

用法

如果打开数据库的事务日志记录，那么在更改生效之前，您必须对数据库中包含数据的所有存储空间创建 0 级备份。

有关日志记录模式的更多信息和示例，请参阅《*SinoDB*® 管理员指南》中的使用 *ondblog* 更改数据库日志记录模式。

或者，您可以使用带有 alter logmode 参数的 SQL 管理 API 命令更改日志记录模式。

不可以在高可用性数据复制（HDR）辅助服务器、远程单机（RS）辅助服务器或共享磁盘（SD）辅助服务器上使用 ondblog 实用程序。

返回码

ondblog 实用程序将消息记录在 BAR_ACT_LOG 文件中。

对于许多返回码，您可以检查 ON-Bar 日志，以找出问题源：

1. 检查 BAR_ACT_LOG 文件以获取附带消息。
2. 将 BAR_DEBUG 配置参数设置为正整数并重试该操作。
3. 检查 ON-Bar 调试日志文件。

表 132: ondblog 实用程序的返回码

返回码	描述	用户操作
1	读取 onconfig 文件错误。	检查 $INFORMIXDIR/etc/$ONCONFIG 目录中的 onconfig 文件。如果设置了 BAR_ACT_LOG 和 BAR_DEBUG_LOG 配置参数，那么确认该文件是有效的。
2	链接列表错误。	查看附带的消息。

返回码	描述	用户操作
3	用户无权运行该命令。	以用户 root、用户 informix、Windows 管理员或数据库服务器所有者身份执行 ondblog 命令。
4	INFORMIXSHMBASE 环境变量设置为 -1 失败。	联系 星瑞格® 软件支持。
5	数据库服务器未联机。	启动数据库服务器。
6	命令选项无效。	确认选项的拼写。
7	与备份实用程序通信失败。	检查数据库服务器是否联机，并且已配置 ON-Bar。
9	分配内存失败。	检查 ON-Bar 日志以获取更多信息。您可能需要要求系统管理员增加交换空间或在系统中安装更多的内存。
16	打开文件失败。	检查 ON-Bar 日志以获取更多信息。
17	无法更改到指定的日志记录模式。	检查 ON-Bar 日志以获取更多信息。
18	更改日志记录模式失败。	检查 ON-Bar 日志以获取更多信息。
19	发生 SQL 错误。	检查 ON-Bar 日志以获取更多信息。
20	发生空列表问题。	检查 ON-Bar 日志以获取更多信息。

oninit 实用程序

oninit 实用程序启动数据库服务器。

在 UNIX™、Linux™ 上，您必须作为用户 root、用户 informix 或非根数据库服务器所有者，才可执行 oninit 实用程序。用户 informix 应是 informix 组中的唯一成员。从命令行执行 oninit 命令。您可以允许属于 DBSA 组的用户执行 oninit 命令。请参阅允许 *DBSA* 组用户运行 *oninit* 命令*#UNIX#* 在第322页。

在 Windows™ 上，SinoDB® 作为 Windows™ 服务运行。任何具有启动 Windows™ 服务权限的用户都可以启动 SinoDB® 服务。服务 控制应用程序使用您提供的任何选项运行 oninit 实用程序。

语法

```
oninit
[  < -FILE option >   ]
[{  <Other options for starting the server>   <Initialize disk space>   | -PHY }]
```

启动服务器的其他选项

启动服务器的其他选项

```
{   [ -j]  [  -w { 600 max_seconds } ]  | {  -s  | -S } }
[{   -SDS=alias  | -D }]
[  -U {  , username  | " " } ]
[ -p]  [ -y]   [ -v]
```

初始化磁盘空间
初始化磁盘空间
-i
[{ -j | -s }]
[-y] [-v]
[-w { 600 *max_seconds* }]
[-U { , *username* | " " }]

表 133: oninit 命令元素

元素	用途	关键注意事项
-D	使用 Enterprise Replication 启动数据库服务器，并禁用高可用性集群复制。	
-i	初始化根数据库空间的磁盘空间，以便数据库服务器可使用它，并启动数据库服务器。	磁盘空间只需初始化一次，以便准备服务器的数据存储。 在默认情况下，为防止数据丢失，您不可重新初始化磁盘空间。要重新初始化现有根数据库空间的磁盘空间，您必须将 FULL_DISK_INIT 配置参数设置为1，然后执行 oninit -i 命令。 请参阅初始化根数据库空间的磁盘空间 在第321页。
-j	在管理模式中启动服务器。	请参阅以管理模式启动服务器 在第321页。
-p	启动数据库服务器而不删除临时表。	如果使用此选项，那么数据库服务器会启动更快，但留在磁盘上临时表使用的空间不会被收回。

-PHY	从最近的检查点启动服务器。-PHY 选项用于告诉服务器仅执行物理恢复而不进行逻辑恢复。	此选项通常用于启动辅助服务器。您必须执行以下命令之一将辅助服务器连接到主服务器：
		`onmode -d secondary`
		`onmode -d RSS`
		如果未在辅助服务器上执行主服务器上的最近的检查点，那么辅助服务器到主服务器的连接会失败。

-s	在静默模式中启动服务器。	当使用此选项时，数据库服务器必须关闭。 数据库服务器处于静默模式时，仅用户 informix 可以存取数据库服务器。
-S	以静默模式将数据库服务器启动为标准服务器，并禁用高可用性数据复制。	数据库服务器处于静默模式时，仅用户 informix 可以存取数据库服务器。
-SDS=*alias*	对于共享磁盘服务器，启动当前服务器并以别名指定主服务器。	当主服务器和所有 SDS 服务器都处于关闭时，使用 -SDS=*alias* 选项将指定的 SDS 服务器启动为主服务器。-SDS=*alias* 标志不可与 -i 标志一起使用。
-U *username*	指定哪些用户可在当前会话的管理模式下存取服务器。	用户 informix 和 DBSA 组成员始终是管理模式的用户。 请参阅以管理模式启动服务器 在第321页。
-v	当服务器正在启动时显示详细的参考性的消息。	
-w *max_seconds*	启动数据库服务器并等待直到服务器在联机模式下完全启动	默认等待秒数为 600。 此选项在高可用性集群中的辅助服务器上是无效的。

元素	用途	关键注意事项
	或 *max_seconds* 所指示的秒数过去，以指示启动成功或失败。	请参阅使用脚本启动服务器 在第321页。
-y	抑制验证提示。	对于所有验证提示，-y 选项自动回应“是”。

用法

默认情况下，oninit 实用程序在服务器启动期间显示验证提示。您可通过 -y 选项抑制验证提示。可通过 -v 选项查看详细参考性消息。在 UNIX™、Linux™ 上，oninit 输出显示到标准输出。在 Windows™ 上，可通过 ONINIT_STOUT 环境变量的设置将 oninit 的输出保存到文件中。

您可在不同操作模式下启动服务器。在默认情况下，如果执行不带选项的 oninit 命令，那么服务器以联机模式启动。当数据库服务器处于联机模式时，所有授权的用户可以存取服务器。

如果您执行 oninit -FILE 命令，那么您在启动数据库服务器之前不需要设置本地环境变量。数据库服务器自动使用 onconfig 文件中的值作为环境变量的值。

以管理模式启动服务器

管理模式是仅限管理员的模式，您可用于执行维护操作，包含要求执行 SQL 或 DDL 命令的操作。当处于管理模式时，数据库服务器仅接受以下用户的连接请求：

- informix 用户
- DBSA 组的成员
- oninit -U 或 onmode -j -U 命令为当前会话指定的用户。-U 选项覆盖 onconfig 文件中 ADMIN_MODE_USERS 配置参数所列出的所有用户。
- ADMIN_MODE_USERS 配置参数所指定的用户

将 -U 选项与逗号分隔的用户名列表一起使用以添加管理模式用户，例如：Karin,Sarah,Andrew。

使用 -U ″ ″ 选项以移除除了 informix 用户和 DBSA 组的成员以外的所有管理模式用户：oninit -U ″ ″。

初始化根数据库空间的磁盘空间

第一次在系统上安装 SinoDB®，您需要初始化数据库服务器的根数据库空间的磁盘空间。根数据库空间是由 ROOTPATH 配置参数指定。

如果您执行典型安装并选择创建数据库服务器，或者执行客户安装，磁盘空间会自动初始化。否则，您必须通过执行 oninit -i 命令来初始化磁盘空间。

如果需要，您可以重新初始化磁盘空间。重新初始化磁盘空间会毁掉数据库服务器所管理的所有现有数据。重新初始化时数据库服务器必须是脱机状态。

在默认情况下，您不可重新初始化数据库服务器正在使用的根数据库空间。如果页零位于根路径位置（第一块位置的第一页），那么磁盘初始化会失败。可通过将 FULL_DISK_INIT 配置参数设置为 1 来允许现有根数据库空间的磁盘初始化。

使用脚本启动服务器

可子在自定义的启动脚本中使用 oninit -w 命令已自动启动。-w 选项强制服务器等到启动完全成功后，然后通过将返回码 0 返回到 shell 提示来指示服务器处于联机模式。如果服务器在逾时前未处于联机模式，那么服务器会将返回码 1 返回到 shell 提示并在 online 日志中写入告警消息。

默认逾时是 600 秒（10 分钟）,可更改为任何整数值。

在运行以下命令之后，如果服务器在 60 秒内无法启动，那么 1 代码会返回到提示符：

```
oninit -w 60
```

要确认服务器无法启动的原因，请查看 online 日志。您可能需要增加逾时值。当在脚本中使用 oninit -w 命令时，您可以使用 onstat - 命令（打印输出头）来检查服务器是否联机。

允许 DBSA 组用户运行 oninit 命令（UNIX™）

要允许 DBSA 组中用户 informix 以外的用户允许 oninit 命令，请以用户 root 身份登录将 $INFORMIXDIR/bin 目录中 oninit 实用程序的权限从 6754 更改为 6755。

相关链接

ADMIN_MODE_USERS 配置参数 在第63页
FULL_DISK_INIT 配置参数 在第115页
《*SinoDB* 管理员指南》：初始化过程
数据库服务器文件 在第715页
SDS_ALTERNATE 配置参数 在第156页
《*SinoDB* 管理员指南》：数据库服务器运行模式

-FILE 选项

在 UNIX™ 上，您可以使用 -FILE 选项使用您在 onconfig 文件中设置的本地环境变量来允许某些 SinoDB® 实用程序。在运行命令以启动实用程序之前，不是一定得设置本地环境变量。

启动以下实用程序时，可以使用 -FILE 选项：oninit、oncheck、onclean、onload、onunload、onlog、onmode、onparams、onspaces、onstat 和 ontape。

语法

-FILE 选项
-FILE 选项
[-FILE=] *file_name*

表 134: -FILE 选项

元素	用途	关键注意事项
-FILE=*file_name*	指定包含环境信息的 onconfig 文件的完整路径或相对路径。	-FILE=*file_name* 选项必须是命令中的第一个参数。

用法

在使用 -FILE 选项运行命令之前，您必须按以下格式将命令添加到 onconfig 文件：

```
#$variable_name value
```

onconfig 文件中设置的环境变量优先于系统中或 shell 中设置的相同环境变量。

使用 -FILE 选项启动实用程序时，请指定 onconfig 文件的完整路径或相对路径。例如：以下两个示例使用 onconfig.serv1 文件中的环境信息来启动数据库服务器：

完整路径

```
oninit -FILE=/opt/IBM/inf/etc/onconfig.serv1
```

相对路径

```
oninit -FILE=etc/onconfig.serv1
```

如果未在用户系统、shell 或 onconfig 文件中设置 INFORMIXDIR 环境变量，那么 INFORMIXDIR 的值会被设置为可执行程序的 PATH，并假设可执行程序是在 INFORMIXDIR 的子目录。例如：当 oninit 命令是在 /opt/IBM/informix/bin 目录时，那么您可运行 oninit -FILE=etc/onconfig.myserv 命令。如果未在 shell 或 onconfig.myserv 文件中设置 INFORMIXDIR 环境变量，那么 INFORMIXDIR 的值会被设置为 /opt/IBM/informix。

如果您使用远程执行的形式，例如：ssh，那么在远程计算机上使用 -FILE 选项来指定onconfig 文件的路径。

示例

假设您在 js_3 实例的 onconfig 文件中指定 INFORMIXSERVER、DBDATE 和 SERVER_LOCALE 环境变量的值：

```
#onconfig.js_3
#
# *** Start environment settings for js_3
#
#$INFORMIXSERVER server3
#$DBDATE MDY4/
#$SERVER_LOCALE en_us.utf8
#
# *** End environment settings for js_3
```

运行该实用程序的其他重要环境变量（INFORMIXDIR、INFORMIXSQLHOSTS、ONCONFIG）是在用户环境中指定。oninit 可执行程序的路径是用户环境的一部分，且 onconfig 位于当前目录中。

您可从当前目录运行 oninit -FILE=onconfig.js_3 命令来启动数据库服务器，并自动设置 NFORMIXSERVER、DBDATE 和 SERVER_LOCALE 环境变量的值。

相关链接

设置实用程序的本地环境变量 在第290页

oninit 实用程序的返回码

如果 oninit 命令遇到错误，那么数据库服务器返回错误消息和返回码值。

下表包含 oninit 实用程序的返回码、消息文本和用户操作。

表 135: oninit 实用程序的返回码

返回码	消息文本	用户操作
0	The database server was initialized successfully.	数据库服务器启动。
1	Server initialized has failed. Look at any error messages written to stderr or the online message log.	基于写到 stderr 或联机消息日志的错误消息采取适当的操作。
87	The database server has detected security violations or certain prerequisites are missing or incorrect.	（仅UNIX™）检查 informix 用户和组是否存在。检查服务器配置文件（onconfig）和 sqlhosts 文件是否存在并有正确的权限。检查环境变量 INFORMIXDIR、ONCONFIG 和 SQLHOSTS 是否具有有效值且长度不超过255 字符。检查环境变量 INFORMIXDIR 是否指定了绝对路径，并且没有空格、制表符、换行符或其他不正确的字符。检查 $INFORMIXDIR 目录下与角色分离相关的子目录，诸

返回码	消息文本	用户操作
		如：aaodir 和 dbssodir，具有正确的所有权。执行 onsecurity 实用程序来检测与修复任何问题。
170	The database server failed to initialize the dataskip structure.	释放系统上的一些物理内存，并尝试再启动数据库服务器。
172	The database server failed to initialize the listener threads.	释放一些系统资源，当数据库服务器启动时检查监听线程启动数量的配置参数值，并且尝试再次启动数据库服务器。
173	The database server failed to initialize data replication.	释放系统中的一些物理内存，并尝试再次启动数据库服务器。
174	The database server failed to start fast recovery threads.	释放系统中的一些物理内存，并尝试再次启动数据库服务器。
175	The database server failed to initialize the root dbspace.	检查服务器配置文件（onconfig）中与根数据库空间相关的参数以确保根数据库空间的路径是有效的。
176	Shared disk secondary server initialization failed.	检查 sqlhosts 文件（UNIX™）中的条目或 SQLHOSTS 注册表键（Windows™）以确保正确地使用主服务器的 DBSERVERNAME 配置值。检查服务器配置文件（onconfig）中的 SDS_PAGING 配置参数值是否正确。释放一些系统资源，并尝试再次启动数据库服务器。
177	The database server failed to start the main_loop thread.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
178	The database server failed to initialize the memory required for page conversion.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
179	The database server was unable to start CPU VPs.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
180	The database server was unable to start the ADM VP.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
181	The database server failed to initialize kernel AIO.	F释放系统上的一些物理内存，并尝试再次启动数据库服务器。
182	The database server was unable to start IO VPs.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
183	The database server failed to initialize the memory required for asynchronous I/O operations.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
184	The database server failed to initialize memory required for parallel database queries. (PDQ)	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
185	The database server failed to initialize various SQL caches.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
186	The database server failed to initialize the Global Language Support (GLS) component.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
187	The database server failed to initialize the Associated Service Facility (ASF) components.	检查 sqlhosts 文件中的条目。

返回码	消息文本	用户操作
188	The database server was unable to start the CRYPTO VP.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
189	The database server was unable to initialize the alarm program.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
190	The database server failed to initialize the auditing component.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
192	The database server failed to restore the Window station and desktop.	（仅Windows™）在释放了一些系统资源后，尝试关闭数据库服务器。
193	The database server failed to create daemon processes.	（仅UNIX™）释放一些系统资源，并尝试再次启动数据库服务器。
194	The database server failed to redirect the file descriptors properly.	（仅UNIX™）检查 /dev/null 设备的可用性，并尝试再次启动数据库服务器。
195	The database server failed to initialize the current directory for use.	检查数据库服务器初始化的当前工作目录的有效性。
196	The database server failed to initialize the /dev/null device.	（仅AIX®）检查 /dev/null 设备的有效性。
197	The database server failed to find the password information for the user trying to initialize the database server.	验证用户密码是有效的。
198	The database server failed to set the resource limits.	（仅UNIX™）验证并增加（如必要）主计算机上进程的资源限制。
200	The database server did not have enough memory to allocate structures during initialization.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
206	The database server could not allocate the first resident segment.	检查服务器配置文件（onconfig）中 BUFFERPOOL 和 LOCKS 配置参数值，以确保主计算机上的可用内存可提供。
207	The database server failed to initialize shared memory and disk space.	释放系统中的一些物理内存，检查数据库服务器中所有块的有效性，并尝试再次启动数据库服务器。
208	The database server failed to allocate structures from shared memory.	释放一些系统资源，并尝试再次启动数据库服务器。
209	The database server encountered a fatal error during the creation of shared memory.	释放系统上的一些物理内存，并尝试再次启动数据库服务器。
210	The database server requested memory for the resident segment that exceeded the maximum allowed.	通过降低 BUFFERPOOL 和 LOCKS 配置参数值来减少驻留段的大小。
220	The database server failed to read the audit configuration file.	检查审计配置文件（adtcfg）存在并有效。
221	The database server could not detect the default directory for DUMPDIR.	如果 $INFORMIXDIR/tmp 目录不存在就创建该目录。

返回码	消息文本	用户操作
	Usually it is the $INFORMIXDIR/tmp directory.	
222	The database server detected an error in the value of the DBSERVERALIASES configuration parameter in the server's configuration file.	验证 DBSERVERALIASES 配置参数值有效，并在 sqlhosts 文件（UNIX™）或 SQLHOSTS 注册表键（Windows™）中具有相应条目。
223	The database server detected an error with the value of the DBSERVERNAME configuration parameter in the server's configuration file.	验证 DBSERVERNAME 配置参数值有效，并在 sqlhosts 文件（UNIX™）或 SQLHOSTS 注册表键（Windows™）中具有相应条目。
224	The database server detected an error with the value of the HA_ALIAS configuration parameter in the server's configuration file.	更正服务器配置文件（onconfig）中的 HA_ALIAS 配置参数值。
225	The database server detected too many entries for the NETTYPE configuration parameter or the DBSERVERALIASES configuration parameter in the server's configuration file.	减少服务器配置文件（onconfig）中 NETTYPE 或 DBSERVERALIASES 配置参数的实例数，并尝试再次启动数据库服务器。
226	The database server could not find an entry for the DBSERVERNAME configuration parameter in the sqlhosts file or the contents of the sqlhosts file are not valid.	检查 sqlhosts 文件中的条目。
227	Incorrect serial number.	重新安装数据库服务器。
228	The user does not have the necessary DBSA privileges to invoke the executable.	用户必须有 DBSA 权限，或者是 Informix-Admin 组的成员（Windows™）。
229	The database server could not initialize the security sub-system.	（仅 Windows™）用户在主机上没有必要的用户权限，或者不是 Informix-Admin 组的成员。
230	The database server, if started as a process on Windows™ platform , timed out while trying to build the required system databases during initialization. (Windows™ only)	检查主机上的事件日志，以确认服务不能打开或启动的原因。数据库服务器在尝试建构系统数据库时可能已逾时。释放一些系统资源，并尝试再次启动数据库服务器。
231	SinoDB® service startup failed when the "oninit -w" command was run as a process on the command line.	（仅Windows™）检查主机上的事件日志，以确认服务启动失败的原因。
233	The database server failed to initialize the Pluggable Authentication Module (PAM).	检查系统上 PAM 库的配置。
235	The database server detected errors for certain configuration parameter values in the server's configuration file.	检查服务器配置文件（onconfig）是否有任何错误。
236	The database server detected an error while trying to restrict the allowable values for the SinoDB® edition in use.	检查服务器配置文件（onconfig）中的 SDS_ENABLE 配置参数是否设置为 1。检查使用 oninit -SDS 命令指定的服务器名称是否与 HA_ALIAS 或 DBSERVERNAME配置参数

返回码	消息文本	用户操作
		值相符。检查已用的共享磁盘是否为现有共享磁盘集群的一部分。
237	The database server could not find the server configuration file.	确保服务器配置文件存在且有效。
238	The database server detected an incorrect value for the INFORMIXSERVER environment variable or the value did not match the value of the DBSERVERNAME configuration parameter in the server's configuration file.	（仅Windows™）检查 INFORMIXSERVER 环境变量值和其在注册表中的相应条目。
239	The database server detected an incorrect or non-existent value for the INFORMIXDIR environment variable.	（仅Windows™）检查 INFORMIXDIR 环境变量值。
240	Incorrect command-line options were issued to the database server.	更正启动时发到数据库服务器的命令行选项。
248	The database server failed to create the SinoDB® loader domain file.	（仅AIX®）检查 /var/adm/ifx_loader_domain 是否存在。
249	The database server failed to dynamically load the PAM library.	PAM 库不可用于数据库服务器。安装 PAM 库。
250	The database server failed to dynamically load the ELF library.	ELF 库不可用于数据库服务器。安装 libelf 包。
255	There was an internal error during server initialization. Look at any error messages written to stderr or to the online message log.	基于写到 stderr 或联机消息日志的错误消息采取适当操作。

onlog 实用程序

onlog 实用程序显示在磁盘或备份上的逻辑日志文件的内容。

onlog：显示逻辑日志内容

在您想要追踪特定事务或查看对特定表空间做过什么更改的调试情况下，onlog 输出是有用的。（有关解释逻辑日志文件内容的信息，请参阅解释逻辑日志记录 在第273页。）

任何用户可运行 -l 选项以外的所有 onlog 选项。仅 UNIX™ 上的用户 informix 或 Windows™ 上的 Informix-Admin 组的成员可以运行 -l 选项。

如果执行 onlog 时数据库服务器是处于脱机模式，那么仅会读取磁盘上的文件。如果数据库服务器是处于静默或联机模式，那么 onlog 还会读取共享内存中逻辑日志缓冲区内的逻辑日志记录（在读取了磁盘上的所有记录之后）

当数据库服务器在联机模式下从磁盘中读取具有状态 U 的逻辑日志文件时，数据库服务器拒绝逻辑日志文件的访问，有效地停止所有会话的数据库活动。（有关更多信息，请参阅 *onstat -l* 命令: 显示物理和逻辑日志信息 在第546页。）基于这个原因，建议您等到文件备份之后从备份读取逻辑日志文件的内容。

onlog 实用程序没有功能等同的 SQL 管理 API *command* 字串符。

onlog 语法
onlog
[< -FILE option >]
<Log-Record Read Filters> <Log-Record Display Filters>
[-q]
[{ -V | -version }]

元素	用途	关键注意事项
-q	不显示初始头和默认情况下每 18 个记录出现一次的单行头	无
-V	显示软件版本号和序列号	请参阅获取实用程序版本信息 在第289页
-version	显示构建版本、主机、操作系统编号、日期，以及 GLS 版	请参阅获取实用程序版本信息 在第289页

当 onlog 搜索要显示的记录时，您指示它读取逻辑日志的以下部分：

- 存储在磁盘上的记录
- 存储在备份介质上的记录
- 所指定逻辑日志文件中的记录

默认情况下，onlog 显示描述事务编号和记录类型的逻辑日志记录头。记录类型标识所有执行操作的类型。

除头之外，您可以使用读取过滤器指示 onlog 显示以下信息：

- 逻辑日志记录头和数据（包含存储在数据库空间或表空间中的简单大对象的副本）
- BLOB 空间中的 BLOB 页的副本

  其仅可从逻辑日志备份复制。其不可从磁盘得到。

您可以显示每个逻辑日志记录头，或者基于以下条件指定输出：

- 与特定表相关联的记录
- 特点用户启动的记录
- 与特定事务相关联的记录

如果 onlog 在日志文件中检测到错误（例如无法识别的日志类型），那么它会以十六进制格式显示整个日志并终止。

日志记录读取过滤器

onlog 实用程序使用存储在根数据库空间保留页中的路径名来定位逻辑日志文件。如果您使用 ON-Bar 备份逻辑日志，那么 onlog 请求存储管理器从备份介质检索相应的逻辑日志记录。

语法
语法
[-d*device* [-b]]
[-n*starting_uniqid - ending_uniquid*]

元素	用途	关键注意事项
-b	显示与 BLOB 空间 BLOB 页相关联的逻辑日志记录	数据库服务器将这些记录作为 BLOB 空间日志记录的一部分存储在逻辑日志备份介质上。
-d *device*	所需逻辑日志备份安装到的存储设备的路径名	如果使用 ontape，那么所指定的设备必须与指定给配置参数 LTAPEDEV 的设备路径名相同。如果不使用 -d 选项，那么 onlog 读取存储在磁盘上的逻辑日志文件，从具有最低的 *logid* 的逻辑日志文件开始。

元素	用途	关键注意事项
		如果使用 ON-Bar 备份逻辑日志，请使用 onbar -P 命令来查看逻辑日志文件的内容。请参阅《*SinoDB*® 备份和还原指南》。 For pathname syntax, see your operating-system documentation.
-n *starting_uniqid-ending_uniqid*	指示 onlog 读取包含在所指定从 *starting uniqid* 到 *ending uniqid* 的日志文件中的逻辑日志记录。	*starting_uniqid* 与 *ending_uniqid* 是逻辑日志的唯一标识号。要确定特定逻辑日志文件的 *uniqid*，请使用 onstat -l 命令。 如果不使用 -n 选项，那么 onlog 将读取磁盘或磁带上所有可用的逻辑日志文件。 有关 onstat 实用程序的信息，请参阅监视数据库服务器状态 在第400页。

日志记录显示过滤器
unique_469_Connect_42_note002

语法
语法
[{ | -l [] | -t *tblspace_num* [] | -u *username* [] | -x *transaction_id* [] | -c *compression_dictionary_file* [] }]

元素	用途	关键注意事项
-l	显示逻辑日志记录的长列表。	日志记录的长列表包含整个日志记录的复合十六进制和 ASCII 转储。该列表不是供随意使用的。
-t*tblspace_num*	显示与您指定表空间相关联的记录。	无符号整数。编号（大于 0）必须在 systables 系统目录表的 partnum 列中。 将该值指定为整数或十六进制值。（如果不使用 0x 前缀，那么该值视为整数。）要确定特定表空间的表空间编号，请照 表空间编号 在第256页 中说明查询 systables 系统目录表。
-u *username*	显示特定用户的记录。	用户名必须是现有的登录名。用户名必须符合特定于操作系统的登录名规则。
-x *transaction_id*	只显示与所指定事务相关联的记录。	值必须是介于 0 和 TRANSACTIONS - 1（包含 0 和 TRANSACTIONS - 1）之间的无符号整数。 只在不太可能发生的前滚过程中生成错误的情况下才必须使用 -x 选项。当发生此情况时，数据库服务器将向消息日志发送包含出错事务的事务标识符的消息。可将该事务标识符与 onlog 的 -x 选项一起使用，以调查错误原因。
-c *compression_dictionary_file*	使用压缩字典来展开压缩数据，并显示未压缩数据。	如果 onlog 命令包含 -l 选项和 -c 选项，并且日志记录中有压缩图像，那么 onlog 实用程序将使用压缩字典来展开日志记录中所有可展开的图像。 仅当压缩字典文件中存在该日志记录的正确压缩字典时，压缩图像才可展开。如果未指定 -c 或压缩字典文件未包含该压缩图像的正确压缩字典，那么 onlog 实用程序将以压缩形式显示该行图像。

如果没有压缩字典文件，那么可使用 UNLOAD 语句将包含在 sysmaster 数据库的 syscompdicts_full 表中的压缩字典卸载到压缩字典文件中，如下所示：

```
UNLOAD TO 'compression_dictionary_file'
   SELECT * FROM sysmaster:syscompdicts_full;
```

如果未指定任何选项，那么 onlog 将显示日志中所有记录的简短列表。可以将这些选项与任何其他选项组合使用，以生成更有选择性的过滤器。例如：如果同时使用 -u 和 -x 选项，那么 onlog 将仅显示指定用户在指定事务过程中启动的活动。如果同时使用 -u 和 -t 选项，那么 onlog 将仅显示由指定用户启动并与指定表空间相关联的活动。

相关链接

alter logmode 参数: 更改数据库日志记录模式 (SQL 管理 API) 在第588页

onmode 实用程序

使用 onmode 实用程序更改数据库服务器操作模式，并在共享内存、会话、事务、参数和段上执行各种其他操作。

这些主题显示如何使用 onmode 选项。如果不使用任何选项，那么数据库服务器返回使用语句。

在 UNIX™ 上，您必须是用户 root 或用户 informix 才可执行 onmode 实用程序。

在 Windows™ 上，您必须是 Informix-Admin 组或 Administrators 组的成员才可执行 onmode 实用程序。

有关 onmode -b 的信息，该命令仅在升级到新版 SinoDB® 并需要将数据库还原到以前版本时使用，请参阅《*SinoDB®* 迁移指南》中的 *onmode -b* 命令的语法部分。

除 onmode -b、onmode -BC 和 onmode -R 外，所有 onmode 命令选项都具有等效的 SQL 管理 API 命令字串符。

onmode 命令语法

使用 onmode 实用程序命令执行各种数据库服务器操作。

以下语法图显示了可与 onmode 命令一起使用的所有选项。该语法图未显示可与每个命令选项一起使用的所有元素。有关每个命令的完整语法，请参阅该命令的主题。

unique_515_Connect_42_note021a

```
onmode
[  < -FILE option >  ]
{  -a  | -b  []  | -BC  | -C  | -c  | -cache surrogates  | -e  | -d  | { -D  | -M  |
-Q  | -S }  | -F  | -I  | { -j  | -k  | -m  | -s  | -u }  | -l  | { -n  | -r }  | -O  |
-P  | -p  | -R  | -W  | { -wf  | -wm }  | { -we  | -wi }  | -Y  | -Z  | -z }
[{ -V  | -version }]
[ -y ]
```

元素	用途	关键注意事项
-y	使数据库服务器自动对所有提示响应“是”	无
-V	显示软件版本号和序列号	请参阅获取实用程序版本信息 在第289页。
-version	显示构建版本、主机、操作系统、编号、日期，以及GLS 版本	请参阅获取实用程序版本信息 在第289页。

onmode -a：添加共享内存段

语法
onmode -a*seg_size*

元素	用途	关键注意事项
-a ***seg_size***	允许您添加新的虚拟共享内存段。大小以 KB 为单位。	限制： ***seg_size*** 的值必须是正整数。不可超过操作系统对共享内存大小的限制。

通常，您无需向共享内存的虚拟部分添加段，因为数据库服务器会在需要时自动添加段。然而，随着段的添加，数据库服务器可能在获得其需要的内存之前达到了操作系统对最大段数的限制。这个情况通常在 SHMADD 配置参数设置的太小，以致数据库服务器在获得某些操作所需的内存之前耗尽可用段的数量时发生。

可以使用此命令添加比 SHMADD 配置参数所指定的大小还大的段。通过使用此命令添加段，可在满足数据库服务器对额外内存的需要的同时避免超过操作系统的限制。

此命令有等效的 SQL 管理 API 函数。

相关链接

add memory 参数: 增加共享内存 *(SQL* 管理 *API)* 在第586页
onmode 和 *a* 参数: 添加共享内存段 *(SQL* 管理 *API)* 在第651页

onmode -BC：允许大块方式

语法：
onmode { -BC 1 | -BC 2 }

元素	用途	关键注意事项
-BC 1	启用对于大块、大于 2 GB 的大偏移量的支持，并允许每个实例最多 32,768 个块。	此选项允许创建大块。如果没有块大于 2 GB，那么可以在不删除数据库空间下进行回复。没有大于 2 GB 块的数据库空间和 BLOB 空间保持旧格式。在向数据库空间或 BLOB 空间添加了大于 2 GB 的块之后，随后在该数据库空间或 BLOB 空间添加或更改的所有块都是新格式。 请参阅《*SinoDB*® 管理员指南》。
-BC 2	对所有数据库空间允许“仅大块”方式。	复原是不可能的。对所有数据库空间和 BLOB 空间启用 9.4 大块功能。所添加或修改的任何块或偏移量都是新格式。未更改的现有块保持旧格式。 请参阅《*SinoDB*® 管理员指南》。

如果您已经从 Informix® 9.4（小块方式）转换到 SinoDB® 12.10，onmode -BC（向下兼容）命令 是非常有用的。当首次初始化 SinoDB® 12.10 时（使用 oninit -iyv 命令），在缺省情况下，它联机的同时完全启用大块方式。复原是不可能的。如果 SinoDB® 12.10 实例是最近初始化的 ，那么 onmode -BC 命令将返回错误。

注： 在执行 onmode -BC 命令之后，请执行一次完全的系统 0 级备份。

onmode -c: 强制检查点

语法:
onmode -c [{ block [{ 15 *timeout* }] | unblock }]

元素	用途	关键注意事项
-c	强制检查点用于将缓冲区清空到磁盘。	如果逻辑日志中最新的检查点记录是阻止释放逻辑日志文件（状态 U-B-L），那么可以使用 -c 选项强制同步检查点。
block	阻塞数据库服务器进行任何事务。	数据库服务器阻塞时，用户可以只读方式访问它。使用此选项在 SinoDB® 上执行外部备份。 有关更多信息，请参阅《*SinoDB®* 备份和还原指南》。
timeout	指定在返回命令提示符之前等待检查点清除的秒数。	仅在配置了 DELAY_APPLY 配置参数时，*timeout* 选项才适用（请参阅 *DELAY_APPLY 配置参数* 在第95页）。如果启用了 DELAY_APPLY 配置参数，那么主服务器请求的检查点在一个延伸时间内可能不会到达辅助服务器。还可能没有其他检查点暂存在暂存目录中。默然超时值是 15 秒，且允许的最大超时值是 10 分钟（600 秒）。请参阅《*SinoDB®* 备份和还原指南》。
unblock	不阻塞数据库服务器。	当数据库服务器未阻塞时，可以继续数据事务和正常的数据库服务器操作。请在完成了 SinoDB® 的外部备份之后使用此选项。 有关更多信息，请参阅《*SinoDB®* 备份和还原指南》。

此命令有等效的 SQL 管理 API 函数。

相关链接

checkpoint 参数: 强制检查点 (SQL 管理 API) 在第602页

onmode 和 c 参数: 强制检查点 (SQL 管理 API) 在第652页

onmode -C: 控制 B 型树扫描程序

使用 onmode -C 命令来控制 B 型树扫描程序并指定有关 B 型树扫描程序线程的信息。
语法:
onmode -C { (yielding syntax) | start *count* | stop*count* | kill *count* | threshold*size* | duration*num* | rangesize*size* | alice *mode* | compression*value* }

元素	用途	关键注意事项
-C	控制 B 型树扫描程序以清除已删除项的索引	可运行的线程数没有限制。然而，一次可启动的线程数限制为 128 个。假设，想要运行 150 个线程，那么可执行以下两个命令：onmode -C 100 和 onmode -C 50。
start *count*	启动额外 B 型树扫描程序线程。	如果未指定 *count*，那么假定 *count* 为 1。可指定的扫描线程的数目是没有限制的。
stop *count* kill *count*	停止 B 型树扫描程序线程。	如果未指定 *count*，那么假定 *count* 为 1。停止所有索引扫描程序以防止所有索引清除。 这些命令的任一个都会停止 B 型树扫描程序。

元素	用途	关键注意事项
threshold *sizecount*	设置将索引放入热列表之前索引必须遇到的已删除项的最小数量。	一旦超过阈值的所有索引已清除且 B 型树扫描程序没有其他工作，那么低于阈值的索引将加入热列表。
duration *num*	热列表有效的秒数。	超过此秒数之后，下一个可用的 B 型树扫描程序线程将重建热列表，无论该表上是否有未处理的项目。不会中断扫描程序当前正在处理的请求。
rangesize *size*	在启动索引范围清除之前确定索引大小。	使用 -1 的大小来禁用范围扫描。
alice *num*	设置系统的 alice 方式。	有效的 *num* 值范围为 0（OFF）到 12。
compression *value*	对于数据库服务器实例，修改两个部分使用的索引页的合并级别。如果这些页上的数据总计达到设置的级别，则合并这些页。	级别的有效值为 low、med（中等）、high 和 default。 系统默认值为 med。

B 型树扫描程序有追踪索引效率以及索引当前放置在服务器上的额外工作量的统计信息。基于索引因提交的删除索引项而完成的额外工作量，B 型树扫描程序生成使服务器做了额外工作的排序列表。该列表称为热列表。首先清除导致最高额外工作量的索引，然后按降序顺序来清除其余索引。DBA 可以动态地分配清除线程，因此运行可配置的工作负载。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode 和 C 参数: 控制 B 型树扫描程序 (SQL 管理 API) 在第653页
set index compression 参数: 更改索引页压缩 (SQL 管理 API) 在第681页
BTSCANNER 配置参数 在第77页

onmode -cache surrogates: 高速缓存 allowed.surrogates 文件

语法：

```
onmode    -cache surrogates
```

元素	用途	关键注意事项
-cache surrogates	读取 /etc/informix/allowed.surrogates 文件并将用户标识符和组标识符的值存于共享内存高速缓存中。allowed.surrogates 文件中的用户名和组名必须是有效的操作系统用户和组。名称将转换为相应的 UID 和 GID。	可在会话期间使用 onmode -cache surrogates 加载 allowed.surrogates 文件。allowed.surrogates 文件用于指定可作为映射用户代理的用户和组。在与数据库服务器建立新连接或创建或更改用户之前，将自动检查 allowed.surrogates 文件。 如果高速缓存刷新失败，则清除现有的代理高速缓存，以有效地禁用映射用户。服务器上的现有连接不受共享内存高速缓存更改的影响。共享内存高速缓存更改影响新的会话。

相关链接

《SinoDB 安全指南》: allowed.surrogates 文件中的映射用户代理人 (UNIX, Linux)
《SinoDB 安全指南》: 指定映射用户的代理人 (UNIX, Linux)
《SinoDB SQL 指南: 语法》: 代理用户属性 (UNIX, Linux)

onmode -d: 设置数据复制类型

语法：

onmode -d { standard | { primary | secondary } *ha_alias* }

元素	用途	关键注意事项
-d	用于设置服务器的数据复制类型。	当数据库服务器是处于静默、联机或只读模式时，可使用 -d standard 选项。
ha_alias	标识主数据库服务器或辅助数据库服务器的高可用性别名。	高可用性别名是服务器的 HA_ALIAS 配置参数值。 重新初始化共享内存之后，将保留数据复制对中另一数据库服务器的 *ha_alias* 参数与数据库服务器类型（标准、主或辅助）。

使用 -d standard 选项

-d standard 选项删除数据复制对中数据库服务器之间的连接，并将当前数据库服务器的数据库服务器类型设置为标准。此选项不会更改对中另一数据库服务器的模式。

onmode -d standard 命令仅断开主服务器与 HDR 辅助服务器的连接。运行此命令将 HDR 辅助服务器转换为独立服务器。不应执行 onmode -d standard 命令来断开主服务器与 RS 辅助服务器的连接。要断开主服务器与 RS辅助服务器的连接，请执行以下命令：

在 RS 辅助服务器上：

```
onmode -d standard
```

在主服务器上：

```
onmode -d delete RSS rss_ha_alias
```

使用 -d primary 选项

-d primary 选项将数据库服务器类型设置为主，并尝试与 *dbservername* 所指定的数据库服务器连接。如果连接成功，将打开数据复制。主数据库服务器变成联机模式，而辅助数据库服务器变成只读模式。如果连接失败，数据库服务器处于联机模式，但不打开数据复制。

使用 -d secondary 选项

-d secondary 选项将数据库服务器类型设置为辅助，并尝试与 *ha_alias* 所指定的数据库服务器连接。如果连接成功，将打开数据复制。主数据库服务器变成联机模式，而辅助数据库服务器变成只读模式。如果连接失败，数据库服务器处于只读模式，但不打开数据复制。

此命令有等效的 SQL 管理 API 函数。

相关链接

ha set primary 参数: 定义 HDR 主服务器 (SQL 管理 API) 在第637页
ha set secondary 参数: 定义 HDR 辅助服务器 (SQL 管理 API) 在第638页
ha set standard 参数: 将 HDR 服务器转换成为标准服务器 (SQL 管理 API) 在第638页
onmode 和 d 参数: 设置数据复制类型 (SQL 管理 API) 在第654页
HA_ALIAS 配置参数 在第115页
DBSERVERALIASES 配置参数 在第88页
DBSERVERNAME 配置参数 在第90页
onmode -d: 设置高可用性服务器的特性 在第335页

onmode -d 命令: 使用数据复制来复制索引 在第336页
《*SinoDB* 管理员指南》: *HA_ALIAS* 配置参数中设置的连接信息

onmode -d: 设置高可用性服务器的特性

语法:
unique_241_Connect_42_secondaryonly unique_241_Connect_42_primaryonly unique_241_Connect_42_primaryorsecondary

onmode -d
{ make primary *ha_alias* [] [force] | <RS Secondary server commands> | <SD Secondary server commands> }
RS 辅助服务器命令
RS 辅助服务器命令
{ { add RSS *rss_ha_alias* [] | RSS *primary_ha_alias* [] } [*password*] | { change RSS *rss_ha_alias password* [] | delete RSS *rss_ha_alias* [] } }
SD 复制服务器命令
SD 复制服务器命令
{ set SDS primary *ha_alias* [] [force] | clear SDS primary *primary_ha_alias* [] }

元素	用途	关键注意事项
-d	用于在高可用性配置中创建、修改或删除辅助服务器	
add RSS	添加 RS 辅助服务器	此命令应在主数据库服务器上运行。
rss_ha_alias	标识 RS 辅助数据库服务器的高可用性别名。	该值可以是 HA_ALIAS 值或 ER 组名。
password	指定辅助服务器的密码	该密码仅在第一次尝试连接时使用。在主服务器与辅助服务器连接之后，不可更改该密码。
RSS	设置 FS 辅助服务器类型	该命令应在辅助数据库服务器上运行。
pri_ha_alias	标识主服务器的名称	
change RSS	更改 RS 辅助服务器	该命令应在主数据库服务器上运行。
delete RSS	移除 RS 辅助服务器定义	该命令应在主数据库服务器上运行。
set SDS primary	将该服务器定义为共享磁盘主服务器	
ha_alias	数据库服务器的高可用性别名	当与 set SDS 或 make primary 一起使用时，此为角色正在改变的服务器的名称。
force	用于强制更改	如果指定了 force 选项，则执行操作时辅助服务器无需连接到当前的主服务器。如果未指定 force 选项，则操作必须与当前主服务器协调。仅当 DBA 确定当前的主服务器不在活动时使用 force 选项；否则会损坏共享磁盘子系统。
clear SDS primary	禁用共享磁盘环境。指定的服务器名称不再作为 SD 主服务器	
make primary	创建主服务器	可在任何类型的辅助服务器上发出 make primary 命令。这些辅助服务器包括 HDR、RS 和 SD 辅助服务器。如果在以下服务器上执行 make primary: • HDR 辅助服务器：关闭当前主服务器并将该辅助服务器变为主服务器。 • RS 辅助服务器：该服务器变成标准服务器。

元素	用途	关键注意事项
		• SD 辅助服务器：该服务器变成新的主服务器。

还可使用等效的 SQL 管理 API 命令 来设置数据复制特性。有关更多信息，请参阅 *SQL 管理 API 概述* 在第571页和《*SinoDB*® 管理员指南》。

有关 onmode -d 的其他信息，请参阅 *onmode -d: 设置数据复制类型* 在第334页和 *onmode -d 命令: 使用数据复制来复制索引* 在第336页。

相关链接

ha make primary 参数: 更改辅助服务器的模式 (SQL 管理 API) 在第631页
ha rss 参数: 创建 RS 辅助服务器 (SQL 管理 API) 在第631页
ha rss add 参数: 添加 RS 辅助服务器到主服务器 (SQL 管理 API) 在第632页
ha rss change 参数: 更改 RS 辅助服务器的密码 (SQL 管理 API) 在第633页
ha rss delete 参数: 删除 RS 辅助服务器 (SQL 管理 API) 在第633页
ha sds clear 参数: 停止共享磁盘复制 (SQL 管理 API) 在第634页
ha sds set 参数: 创建共享磁盘主服务器 (SQL 管理 API) 在第635页
ha sds primary 参数: 将 SD 辅助服务器转换为主服务器 (SQL 管理 API) 在第635页
onmode -d: 设置数据复制类型 在第334页
DBSERVERALIASES 配置参数 在第88页
DBSERVERNAME 配置参数 在第90页
HA_ALIAS 配置参数 在第115页
onmode -d 命令: 使用数据复制来复制索引 在第336页

onmode -d 命令: 使用数据复制来复制索引

语法：
onmode -d { idxauto { on | off } | index *database*: [*owner*.] *table*#*index* }

元素	用途	关键注意事项
-d	指定当辅助服务器上的索引损坏时索引如何将数据复制到高可用性数据复制（HDR）辅助服务器。	当服务器处于联机模式时，您可以使用 onmode -d idxauto 和 onmode -d index 命令。
idxauto	当辅助服务器上的索引毁坏时启用自动索引复制	使用 onmode -d idxauto 命令来覆盖会话中 DRIDXAUTO 配置参数的值。
index	将索引从主服务器复制到辅助服务器	如果在辅助服务器上检测到损坏索引，请使用 onmode -d index 命令开始将数据从主服务器复制到辅助服务器。
database	指定包含要复制的索引的数据库	语法必须符合 Identifier 段。请参阅《*SinoDB*® 管理员指南》。
index	指定要复制的索引的名称	指定的表和数据库中必须存在索引。 语法必须符合 Identifier 段；请参阅《*SinoDB*® 管理员指南》。
owner	指定表的所有者	必须指定表的当前所有者。 语法必须符合 Table Name 段；请参阅《*SinoDB*® 管理员指南》。

元素	用途	关键注意事项
table	指定建立索引的表的名称	语法必须符合 Table Name 段；请参阅《*SinoDB*® 管理员指南》。

onmode -d idxauto 和 onmode -d index 命令提供了将索引复制到包含毁坏索引的辅助服务器的方法。在传送索引期间，基本表将被锁定。使用这些选项的另一种方式是在主服务器上删除并重建毁坏的索引。

如果是分段索引的一个分段毁坏，那么 onmode -d idxauto 命令将只传送单个受影响的分段，而 onmode -d index 命令则是传送整个索引。

相关链接

DRIDXAUTO 配置参数 在第99页
ha set idxauto 参数: 复制索引到辅助服务器 *(SQL* 管理 *API)* 在第636页
onmode -d: 设置数据复制类型 在第334页
onmode -d: 设置高可用性服务器的特性 在第335页
DBSERVERALIASES 配置参数 在第88页
DBSERVERNAME 配置参数 在第90页
HA_ALIAS 配置参数 在第115页

onmode -D, -M, -Q, -S: 更改决策支持参数

语法:
onmode { -D*max_priority* | -M*kilobytes* | -Q *queries* | -S*scans* }

元素	用途	关键注意事项
-D ***max_priority***	更改 MAX_PDQPRIORITY 的值	该值必须是 0 到 100 之间的无符号整数。 指定 *max_priority* 作为调节用户对 PDQ 资源的请求的因素。 有关用于控制 PDQ 的参数的信息，请参阅 *MAX_PDQPRIORITY* 配置参数 在第131页和《*SinoDB*® 性能指南》。
-M ***kilobytes***	更改 DS_TOTAL_MEMORY 的值	该值的上限取决于平台。如果输入一个对于平台而言太大的值，您将收到一条消息说明您的平台的值的范围。 以 *kilobytes* 来指定可用于并行查询的最大内存量。 有关更多信息，请参阅 *DS_TOTAL_MEMORY* 配置参数 在第106页和《*SinoDB*® 性能指南》。
-Q ***queries***	更改 DS_MAX_QUERIES 的值	该值必须是 1 到 8,388,608 之间的无符号整数。 指定并发执行的并行查询的最大的查询数。 有关用于控制 PDQ 的参数的信息，请参阅 *DS_MAX_QUERIES* 配置参数 在第102页和《*SinoDB*® 性能指南》。
-S ***scans***	更改 DS_MAX_SCANS 的值	该值必须是 10 到 1,048,576 之间的无符号的整数。 指定并发执行的并行扫描的最大扫描数。 有关用于控制 PDQ 的参数的信息，请参阅 *DS_MAX_SCANS* 配置参数 在第103页和《*SinoDB*® 性能指南》。

这些选项允许您在数据库服务器处于联机时更改配置参数。新的值只影响数据库服务器的当前实例；这些值不记录在 ONCONFIG 文件中。如果关闭并重新启动数据库服务器，那么这些参数的值将还原到 ONCONFIG 文件中的值。有关这些配置参数的更多信息，请参阅数据库配置参数 在第41页。

要检查 MAX_PDQPRIORITY、DS_TOTAL_MEMORY、DS_MAX_SCANS、DS_MAX_QUERIES 和 DS_NONPDQ_QUERY_MEM 配置参数的当前值，请使用 onstat -g mgm。请参阅 *onstat -g mgm* 命令: 显示 *MGM* 资源信息 在第480页。

此命令有等效的 SQL 管理 API 函数。

相关链接

DS_MAX_QUERIES 配置参数 在第102页
DS_MAX_SCANS 配置参数 在第103页
MAX_PDQPRIORITY 配置参数 在第131页
DS_TOTAL_MEMORY 配置参数 在第106页
onmode 和 *D* 参数: 设置 *PDQ* 优先级 *(SQL* 管理 *API)* 在第655页
onmode 和 *M* 参数: 临时更改决策支持内存 *(SQL* 管理 *API)* 在第658页
onmode 和 *Q* 参数: 设置决策支持查询的最大数目 *(SQL* 管理 *API)* 在第661页
onmode 和 *S* 参数: 设置决策支持扫描的最大数目 *(SQL* 管理 *API)* 在第662页

onmode -e: 更改 SQL 语句高速缓存的使用

语法:
onmode -e*mode*

元素	用途	关键注意事项
onmode -e ENABLE	启用 SQL 语句高速缓存。有关更多信息，请参阅《*SinoDB*® 性能指南》中关于提升查询性能的部分。	用户会话只在执行以下操作时才使用高速缓存: • 将环境变量 STMT_CACHE 设置为 1 • 执行 SQL 语句 SET STATEMENT CACHE ON
onmode -e FLUSH	清空 SQL 语句高速缓存中不在使用的语句	onstat -g ssc ref_cnt 字段显示 0。
onmode -e OFF	关闭 SQL 语句高速缓存	不高速缓存任何语句。
onmode -e ON	打开 SQL 语句高速缓存	高速缓存所有语句，除非用户使用以下操作关闭它: • 将环境变量 STMT_CACHE 设置为 0 • 执行 SQL 语句 SET STATEMENT CACHE OFF

onmode -e 的梗概只对当前数据库服务器会话生效。当数据库服务器重新启动时，它使用 ONCONFIG 文件中缺省的 STMT_CACHE 参数值。

此命令有等效的 SQL 管理 API 函数。

相关链接

STMT_CACHE 配置参数 在第179页
onmode 和 *e* 参数: 更改 *SQL* 语句高速缓存的使用 *(SQL* 管理 *API)* 在第655页

onmode -F：释放未使用的内存段

使用 onmode -F 命令来释放进程不可用或不再需要的共享内存段。

语法：
onmode -F

元素	用途	关键注意事项
-F	释放未使用内存段	无。

当执行 onmode -F 时，内存管理器检查每个内存池的未使用的内存。内存管理器发现未使用内存块时，它立即释放该内存。内存管理器检查每个内存池之后，它开始检查内存段并释放数据库服务器不再需要的所有段。

建议从操作系统调度工具定期运行 onmode -F，并且在数据库服务器执行了任何创建其他内存段的功能（包括大索引构建、排序或备份）之后运行。

运行 onmode -F 会造成执行该实用程序时处于活动状态的所有用户的性能的显著降级。尽管执行时间很短（1 到 2 秒），但对于单用户数据库服务器的降级可达到 100%。具有多个 CPU 虚拟处理器的系统经历的降级成比例减少。

要确认 onmode 已释放了未使用的内存，请检查消息日志。如果内存管理器释放一个或多个段，它将显示一条消息指示已释放了多少内存段和字节。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode 和 F 参数: 释放未使用的内存段 (SQL 管理 API) 在第656页

onmode -I：控制诊断收集

使用 onmode -I 选项来启动和停止诊断收集。

遇到错误时，可以指定 onmode -I *iserrno* 选项以开始收集诊断信息。还可以指定会话标识符以便只收集特定会话的信息。

要停止诊断收集，请使用没有其他参数的 onmode -I 选项。

onmode
-I
[*iserrno* [, *sid*]]

元素	用途	关键注意事项
iserrno	要收集诊断信息的错误消息码。	无。
sid	要为其收集诊断信息的会话的会话标识符。	无。

在《*SinoDB*® 管理员指南》有诊断收集程序的描述。

onmode -k, -m, -s, -u, -j：更改数据库服务器模式

语法：
onmode { -k | -m | -s | -u | -j }

元素	用途	关键注意事项
-k	使数据库服务器处于脱机模式，并移除共享内存	要重新初始化共享软件内存，请关闭并重新启动数据库服务器。 使用 *-k* 选项使数据库服务器处于脱机模式 在第340页。
-m	将数据库服务器从静默模式或管理模式更改为联机模式	请参阅 使用 *-m* 选项使数据库服务器处于联机模式 在第340页。
-s	正常关闭数据库服务器	在数据库服务器变成静默模式之前，允许正在使用数据库服务器的用户完成，但不允许新的连接。 当所有处理完成时，-s 使数据库服务器变成静默模式。-s 选项使共享内存保持不变。 请参阅使用 *-s* 选项以正常关闭数据库服务器 在第340页。
-u	立即关闭数据库服务器	该选项使数据库服务器变成静默模式，而不等待用户完成其会话。回滚他们的当前事务，并终止他们的会话。 请参阅使用 *-u* 选项立即关闭数据库服务器 在第341页。
-j	使数据库服务器进入管理模式	该选项使数据库服务器变成管理模式，其允许 informix 用户执行所有功能，包括 SQL 和 DDL 命令。-j -U 选项使 DBSA 能够指定特定用户（除了 informix 用户以外的用户）来访问数据库服务器。 请参阅《*SinoDB*® 管理员指南》。

以下各节描述了使数据库服务器从一个模式更改为另一个模式的选项。

相关链接

onmode 和 *j* 参数: 切换数据库服务器到管理模式 *(SQL* 管理 *API)* 在第657页

onmode 和 *m* 参数: 切换到多用户模式 *(SQL* 管理 *API)* 在第658页

使用 -k 选项使数据库服务器处于脱机模式

onmode -k 选项使数据库服务器变为脱机模式，并移除数据库服务器共享内存。

请求确认的提示。在数据库服务器变为脱机模式之前，另一个提示请求对杀掉用户线程进行确认。如果要消除这些提示，请执行 -y 选项与 -s 选项。

此选项不会杀掉所有客户端会话。使用 -u 选项避免挂起客户端会话或虚拟服务器进程。

> 重要： 当使用 onmode -k 命令关闭数据库服务器时，正等待用户响应的实用程序可能不终止。例如：ontape 可能正在等待另一个磁带，onstat -i 可能正在等待用户响应，或 onspaces 可能正在等待 y 或 n 以继续。如果发生这些种问题，请使用 onmode -uk 或 -uky 而不是在移除共享内存之前回滚工作。有关更多信息，请参阅其他选项的描述。

相关链接

onclean 实用程序 在第312页

onshutdown 脚本 在第313页

使用 -m 选项使数据库服务器处于联机模式

-m 选项使数据库服务器从静默模式变成联机模式。

使用 -s 选项以正常关闭数据库服务器

-s 选项导致正常关闭。在数据库服务器变成静默模式之前，允许正在使用数据库服务器的用户完成，但不允许新的连接。当所有处理完成时，-s 使数据库服务器成为静默模式。-s 选项使共享内存保持不变。

请求确认的提示。如果要消除此提示，请执行 -y 选项与 -s 选项。

使用 -u 选项立即关闭数据库服务器

-u 选项导致立即关闭。此选项使数据库服务器成为静默模式，而不等待用户完成其会话。回滚他们的事务，并终止他们的会话。

请求确认的提示。在数据库服务器变成静默模式之前，另一个提示请求对杀死用户线程进行确认。如果要消除这些提示，请执行 -y 选项和 -s 选项。

使用 -j 选项使数据库服务器更改为管理模式

-j 选项使数据库服务器进入管理模式，并只允许 DBSA 组和用户 informix 可连接服务器。-j 选项允许 DBSA 使服务器在完全功能模式来执行维护。

-j -U 选项使 DBSA 能够授权个别用户以管理模式访问数据库服务器。一旦连接了，这些个别用户可执行任何 SQL 或 DDL 命令。当服务器更改为管理模式时，用户 informix、DBSA 组用户和 onmode -j -U 命令中标识的那些用户以外的所有用户的会话将丢失与数据库服务器的连接。

以下示例使三个用户连接到数据库服务器，并可访问数据库服务器直到数据库服务器模式变成脱机模式、静默模式或联机模式：

```
onmode -j -U karin,sarah,andrew
```

可通过执行 onmode -j -U 和从该命令的新名单中删除个别用户的名字来删除他们的访问。例如：在以下命令中，第一个命令只授权 Karin 访问，第二个命令授权 Karin 和 Sarah 访问，第三个命令只授权 Sarah 访问（删除了 Karin 的访问）。

```
onmode -j -U karin
onmode -j -U karin,sarah
onmode -j -U sarah
```

要允许用户 informix 和 DBSA 组用户在管理模式下保持数据库服务器访问，并删除所有单个用户访问数据库服务器，请使用以下命令：

```
onmode -j -U ' '
```

有关使用配置参数在管理模式下指定单个用户的信息，请参阅 *ADMIN_MODE_USERS* 配置参数 在第63页。

相关链接

ADMIN_MODE_USERS 配置参数 在第63页

onmode -l: 切换逻辑日志文件

语法：

```
onmode   -l
```

元素	用途	关键注意事项
-l	将当前逻辑日志文件切换为下一个逻辑日志文件	必须使用 onmode 切换到下一个逻辑日志文件。 有关切换到下一个逻辑日志文件的信息，请参阅《*SinoDB*® 管理员指南》中有关管理逻辑日志文件的章节。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode 和 *l* 参数: 切换到下一个逻辑日志 *(SQL* 管理 *API)* 在第657页

onmode -n, -r: 更改共享内存驻留

语法:
onmode { -n | -r }

元素	用途	关键注意事项
-n	结束共享内存的驻留部分的强制驻留	此命令不影响 RESIDENT（ONCONFIG 文件中强制驻留参数）的值。
-r	启动共享内存的驻留部分的强制驻留	此命令不影响 RESIDENT（ONCONFIG 文件中强制驻留参数）的值。

重要: 在使用 onmode -r 或 -n 选项之前，请将 RESIDENT 参数设置为 1 。

有关使用强制驻留参数以在下一次重新启动数据库服务器时打开或关闭驻留的信息，请参阅《*SinoDB*® 管理员指南》中有关管理共享内存的章节。

此命令有等效的 SQL 管理 API 函数。

相关链接

RESIDENT 配置参数 在第148页

onmode 和 *n* 参数: 解锁驻留内存 *(SQL* 管理 *API)* 在第659页

onmode 和 *r* 参数: 强制共享内存的驻留 *(SQL* 管理 *API)* 在第662页

onmode -O: 重设 ONDBSPACEDOWN WAIT 方式

语法:
onmode -O

元素	用途	关键注意事项
-O	重设 ONDBSPACEDOWN 配置参数的 WAIT 方式	无。

只在以下情况时使用 onmode -O 选项:

- ONDBSPACEDOWN 设置为 WAIT。
- 发生禁用的 I/O 错误，导致数据库服务器阻挡所有正在更新的线程。
- 无法或不想更正导致禁用 I/O 错误的问题。
- 想要使数据库服务器将禁用的数据库空间标识为关闭并继续处理。

当执行此选项时，数据库服务器会将导致禁用 I/O 错误的数据库空间标识为关闭、完成检查点，并释放阻塞的线程。然后，onmode 提示您以下消息:

```
This will render any dbspaces which have incurred disabling I/O errors unusable
and require them to be restored from an archive.
Do you wish to continue?(y/n)
```

当运行 -O 选项时，如果 onmode 在非临界数据库空间上未找到任何 I/O 错误，那么它会通知您以下消息:

```
There have been no disabling I/O errors on any noncritical dbspaces.
```

此命令有等效的 SQL 管理 API 函数。

相关链接

ONDBSPACEDOWN 配置参数 在第140页

onmode 和 *O* 参数: 标记禁用的数据库空间为 *down (SQL* 管理 *API)* 在第660页

onmode -p：添加或删除虚拟处理器

使用 onmode -p 命令以动态地添加或删除数据库服务器实例的虚拟处理器。onmode -p 命令不会更新 onconfig 文件。

语法：

onmode -p { + *number* { ADT | AIO | BTS | CPU | DWAVP | JVP | LIO | MSC | PIO | SOC | STR *vpclass* } | - *number* { BTS | CPU | JVP *vpclass* } }

元素	用途	关键注意事项
-p *number*	添加或删除虚拟处理器。*number* 参数指示要添加或删除的虚拟处理器的数量。 如果该值是负整数，那么删除处理器。如果该值是正整数，那么添加处理器。	仅当数据库服务器处于联机模式时，才可使用 -p 选项，且一次只能添加一类虚拟处理器。 有关详细信息，请参阅添加和删除虚拟处理器的规则 在第344页。 如果正在删除虚拟处理器，那么最多不能超过指定类型的实际处理器的数量。如果正在添加虚拟处理器，那么最大数量取决于操作系统。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“使用虚拟处理器”一章。
ADT	运行审计进程	当通过 ONCONFIG 文件中的 ADTMODE 参数打开审计模式时，数据库服务器启动一个 audit 类的虚拟处理器。
AIO	对熟磁盘空间执行非日志记录磁盘 I/O	如果未使用内核异步 I/O（KAIO），那么对原磁盘空间也执行非日志记录 I/O。
BTS	运行 Basic Text Search 索引操作和查询。	BTS 虚拟处理器是非让步虚拟处理器。如果想要更多 Basic Text Search 查询同时运行，那么指定更多的 BTS 虚拟处理器。使用 onconfig 文件中的 VPCLASS 参数创建至少一个永久的 BTS 虚拟处理器。 有关 Basic Text Search 的更多信息，请参阅《*SinoDB*® 数据库扩展用户指南》。
CPU	运行所有会话线程和某些系统线程	建议 CPU VP 的数量不要大于物理处理器的数量。如果使用了 KAIO，那么对原磁盘空间执行 I/O，包括对物理日志和逻辑日志的 I/O。在可用情况时运行 KAIO 线程或运行单个轮询线程。数据库服务器将 CPU VP 的数量用于为并行数据库查询（PDQ）分配资源。如果删除 CPU VP，查询运行将显著减慢。onstat -g mgm 输出的 Reinit 字段显示有关在 onmode -p 命令之后等待正在运行的查询完成的查询数量的信息。另请参阅《*SinoDB*® 性能指南》。
DWAVP	在连接到仓库加速器的数据库服务器上运行仓库加速器的管理功能和程序	单个 DWAVP 可处理大部分的仓库加速器操作，而没有拖延。然而，在具有重要活动的系统中，您可以定义最多两个 DWAVP 虚拟处理器，以避免数据市集正在加载时拖延到其他管理命令。
JVP	在 Java™ 虚拟机（JVM）中执行 Java™ 用户定义的例程	如果正在运行多个 Java™ UDR，请指定更多 JVP。
LIO	写入逻辑日志文件，如果其位于熟磁盘空间。	仅当逻辑日志位于镜像数据库空间时才使用两个 LIO 虚拟处理器。 数据库服务器最多允许两个 LIO 虚拟处理器。
MSC	管理需要大量堆栈的系统调用请求	用于其他内部任务。

元素	用途	关键注意事项
PIO	写入物理日志，如果其位于熟磁盘空间	仅当物理日志位于镜像数据库空间时才使用两个 PIO 虚拟处理器。数据库服务器最多允许两个 PIO 虚拟处理器。
SOC	使用套接字执行网络通信	仅当数据库服务器配置为通过套接字进行网络连接时才使用 SOC 虚拟处理器。
STR	执行流管道连接	
vpclass	给出用户定义的虚拟处理器类的名称	在 onconfig 中使用 VPCLASS 参数定义用户定义的虚拟处理器类。如果正在运行多个 UDR，请指定更多用户定义的虚拟处理器。 在 Windows™ 上，一次仅可有一个用户定义的虚拟处理器类。在 onmode -p *vpclass* 命令中省略 *number* 参数。 For more information on extension classes, see *VPCLASS* 配置参数 在第196页.

相关链接

onmode 和 *p* 参数: 添加或移除虚拟处理器 *(SQL* 管理 *API)* 在第660页

VPCLASS 配置参数 在第196页

添加和删除虚拟处理器的规则

您可添加或删除虚拟处理器。

以下规则适用:

- 不可删除最终的虚拟处理器。至少必须保留一个虚拟处理器。
- 不可添加或删除 ADM 或 OPT。
- 仅 Windows™ ：可添加任何累的支持虚拟处理器，但是不可删除虚拟处理器。

可添加或删除这些虚拟处理器:

虚拟处理器名称	添加	删除
ADT	是	否
AIO	是	否
BTS	是	是
CPU	是	是
JVP	是	是
LIO	是[1] 在第344页	否
MSC	是	否
PIO	是[1] 在第344页	否
SOC	是	否
STR	是	否
vpclass	是	是

表注释:

1. 可以再添加一个虚拟处理器。

使用 onstat 实用程序监视轮询线程

当数据库服务器处于联机状态时，不可删除正在运行轮询线程的 CPU 虚拟处理器。要确认在 CPU 虚拟处理器上运行的轮询线程，请使用以下命令：

```
onstat -g ath | grep 'cpu.*poll'
```

以下 onstat -g ath 输出显示具有轮询线程的两个 CPU 虚拟处理器。在此情形下，不可降到低于两个 CPU 虚拟处理器。

```
tid  tcb      rstcb  prty  status    vp-class  name

8     a362b90  0     2     running   1cpu      tlitcppoll
9     a36e8e0  0     2     cond wait arrived   3cpu
```

status 字段包含信息，诸如：running、cond wait、IO Idle, IO Idle、sleeping secs: *number_of_seconds* 或 sleeping forever。要提升性能，可除去或减少标识为 sleeping forever 的线程。

有关虚拟处理器类型的信息，请参阅《*SinoDB*® 管理员指南》中“虚拟处理器和线程”一章。

此命令有等效的 SQL 管理 API 函数。

onmode -P：动态地启动、停止或重启监听线程

使用 onmode -P 命令，在不中断现有连接的情况下，启动、停止或重启 SOCTCP 或 TLITCT 网络协议的现有监听线程。

语法：
onmode -P { start | stop | restart } *server_name*

元素	用途	关键注意事项
start	在不中断现有连接的情况下，为 SOCTCP 或 TLITCP 网络协议启动新的监听线程。	监听线程的定义必须存在服务器的 sqlhosts 文件中。如果监听线程的定义不存在 sqlhosts 文件中，那么您必须在可以动态地启动监听线程之前添加它。
stop	在不中断现有连接的情况下，停止 SOCTCP 或 TLITCP 网络协议的现有监听线程。	监听线程的定义必须存在服务器的 sqlhosts 文件中。
restart	在不中断现有连接的情况下，停止并启动 SOCTCP 或 TLITCP 网络协议的现有监听线程。	监听线程的定义必须存在服务器的 sqlhosts 文件中。
server_name	要在其上启动，停止或重启监听线程的数据库服务器的名称。	

这些命令不会更新 sqlhosts 文件。

这些命令等同于具有 start listen、stop listen 或 restart listen 参数的 SQL 管理 API 函数。

示例

以下命令停止然后启动名为 ids_serv1 的服务器上的监听线程：

```
onmode -P restart ids_serv1
```

相关链接

start listen 参数: 动态地启动监听线程 *(SQL* 管理 *API)* 在第690页

stop listen 参数: 动态地停止监听线程 *(SQL* 管理 *API)* 在第692页
restart listen 参数: 动态地停止并启动监听线程 *(SQL* 管理 *API)* 在第675页

onmode -R: 重新生成 .infos.***dbservername*** 文件

数据库服务器在共享内存初始化时创建 .infos.*dbservername* 文件，并在其关闭时移除该文件。该文件位于 $INFORMIXDIR/etc or %INFORMIXDIR%\etc 目录中。该文件的名称是从 ONCONFIG 配置文件中的 DBSERVERNAME 参数衍生而来。

当数据库服务器访问实用程序时，它使用 .infos.*dbservername* 文件中的信息。如果意外删除了该文件，那么必须重新创建该文件，或者关闭并重启数据库服务器。

语法：
onmode -R

元素	用途	关键注意事项
-R	重新创建 .infos.*dbservername* 文件	在使用 -R 选项之前，将 INFORMIXSERVER 环境变量设置与 ONCONFIG 文件中的 DBSERVERNAME 参数相匹配。如果 INFORMIXSERVER 环境变量设置为 DBSERVERALIASES 名称之一，则不要使用 -R 选项。

onmode -W: 更改 SQL 语句高速缓存的设置

语句：
onmode -W { STMT_CACHE_HITS*hits* | STMT_CACHE_NOLIMIT*value* }

元素	用途	关键注意事项
STMT_CACHE_HITS ***hits***	指定语句完全插入到 SQL 语句高速缓存之前，对它=的命中（引用）数。将 ***hits*** 设置为 1 或更大可防止特殊查询进入高速缓存。	仅可增加或重设 STMT_CACHE_HITS. 的值。新值显示在 onstat -g ssc 输出的 #hits 字段中。如果 ***hits*** = 0，那么数据库服务器将在高速缓存中插入所有合格的语句及其内存结构。如果 ***hits*** > 0 且 SQL 语句执行次数小于 STMT_CACHE_HITS，那么在高速缓存中插入仅键条目。在语句达到指定的命中次数之后，它会将合格的语句插入高速缓存中。ONCONFIG 参数: STMT_CACHE_HITS
STMT_CACHE_NOLIMIT ***value***	控制语句是否插入 SQL 语句高速缓存中。	如果 *value* = 0，那么数据库服务器不会在高速缓存中插入任何语句。如果 *value* = 1，那么数据库服务器总在高速缓存中插入语句。如果没有共享任何查询，那么关闭 STMT_CACHE_NOLIMIT 可防止数据库服务器为高速缓存分配大量内存。ONCONFIG 参数: STMT_CACHE_NOLIMIT

相关链接

STMT_CACHE_HITS 配置参数 在第180页
STMT_CACHE_NOLIMIT 配置参数 在第181页
onmode 和 *W* 参数: 重设语句高速缓存属性 *(SQL* 管理 *API)* 在第663页

SQL 语句高速缓存示例

以下为用于更改 SQL 语句高速缓存（SSC）设置的 onmode -W 命令的示例。更改只对当前数据库服务器会话生效，且不更改 ONCONFIG 值。当重新启动数据库服务器时，如果在 ONCONFIG 中没有指定，则使用缺省 SSC 设置，否则使用 ONCONFIG 设置。要做出永久更改，请设置适当的配置参数。

```
onmode -W STMT_CACHE_HITS 2  # number of hits before statement is
     # inserted into SSC
onmode -W STMT_CACHE_NOLIMIT 1  # always insert statements into
     # the cache
```

此命令有等效的 SQL 管理 API 函数。

onmode -we：导出包含当前配置参数的文件

使用 onmode -we 命令来创建并导出配置文件，该文件是您当前配置参数的快照。

语法：
onmode -we *path_name*

元素	描述	关键注意事项
path_name	配置文件的完整或相对路径名	不添加副档名。

用法

onmode -we 命令自动创建 ASCII 文件，并为其指定命令中指定的名称。文件格式与 onconfig.std 文件的格式相同。

如果在当前会话中动态地更改了任何值，那么导出文件将包含已更改的值，而不是永久保存在 onconfig 文件中的值。

导出配置文件之后，可以将其导入并将其作为配置文件使用。

如果执行 onmode -we 命令并指定先前导出的文件，那么该命令将导出新版文件，覆盖掉先前的导出文件。

onmode -we 命令等同具有 onmode 和 export 参数的 SQL 管理 API 函数。

示例

以下命令导出所有配置参数及其值到 /tmp 目录中的 onconfig3 文件：

```
onmode -we /tmp/onconfig3
```

相关链接

更改 *onconfig* 文件 在第42页
onmode -wi: 导入配置参数文件 在第348页
export config 参数*:* 导出配置参数值 *(SQL* 管理 *API)* 在第628页

onmode -wf，-wm：动态更改某些配置参数

使用 onmode -wf 或 onmode -wm 命令可以动态地更改一些特定的配置参数。
语法：
onmode { -wf *config_param=value* | -wm *config_param=value* }

元素	用途	关键注意事项
-wf	更新 onconfig 文件中指定的配置参数的值。	DBA 用户必须有包含 onconfig 文件的目录的写入权限。
-wm	动态地设置内存中指定的配置参数的值。	当服务器重新启动时，不会保留指定的值。
config_param=value	指定配置参数及其新值。	请参阅数据库配置参数 在第41页。

要查看可以使用 onmode -wm 或 -wf command 动态地调整的配置参数的列表，请执行 onstat -g cfg tunable 命令。

onmode -wf 和 onmode -wm 命令有等效的 SQL 管理 API 函数。

相关链接

更改 *onconfig* 文件 在第42页

onstat -g cfg 命令: 显示配置参数的当前值 在第438页

onmode 和 *wf* 参数: 永久地更新配置参数 *(SQL* 管理 *API)* 在第664页

onmode 和 *wm* 参数: 临时地更新配置参数 *(SQL* 管理 *API)* 在第664页

set onconfig memory 参数: 临时地更改配置参数 *(SQL* 管理 *API)* 在第682页

set onconfig permanent 参数: 永久地更改配置参数 *(SQL* 管理 *API)* 在第683页

onmode -wm: 更改 LRU 调整状态

在不更改 onconfig 的情况下，可使用 onmode -wm 选项更改 LRU 调整状态。

语法:

```
onmode   -wmAUTO_LRU_TUNING { { 0  | 1 } }
```

元素	用途	关键注意事项
-wm	为当前会话动态地设置指定的配置参数的值。	无。
0	关闭当前会话的自动 LRU 调整。	无。
1	打开当前会话的自动 LRU 调整。	无。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode、*wm* 和 *AUTO_LRU_TUNING* 参数: 更改 *LRU* 调整状态 *(SQL* 管理 *API)* 在第665页

onmode -wi: 导入配置参数文件

使用 onmode -wi 命令导入包含多个配置参数的新值的文件。如果参数是可调整的，这意味着可以使用 onmode -wm 命令单独更新它们，则数据库服务器将应用新值。

语法:

```
onmode   -wi path_name
```

元素	用途	关键注意事项
path_name	先前捣珠的配置文件的完整或相对路径名。	

用法

使用 onmode -wi 导入配置文件比在多个可调配置参数上运行单独的 onmode -wm 要更快，更方便。

导入操作忽略文件中不可调的配置参数。该操作还忽略与实例当前使用的值匹配的新参数值。

导入文件之后，可以修改导入的配置参数的值。

导入操作只更改内存中配置参数的值。该操作不影响 $INFORMIXDIR/etc/$ONCONFIG 文件中的值。

onmode -wi 命令等同于具有 onmode 和 wi 参数，或者具有 import 参数的 SQL 管理 API 函数。

示例

以下命令导入 /tmp 目录中名为 onconfig3 的文件中的配置参数：

```
onmode -wi /tmp/onconfig3
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
更改 *onconfig* 文件 在第42页
onmode -we: 导出包含当前配置参数的文件 在第347页
import config 参数: 导入配置参数值 *(SQL* 管理 *API)* 在第640页

onmode -Y: 动态更改 SET EXPLAIN

语法：

onmode -Y*session_id* { 0 | { 2 | 1 } *file_name* }

元素	用途	关键注意事项
file_name	解释输出文件的名称。	如果未包含文件的绝对路径，那么将在缺省的解释输出文件位置创建解释输出文件。如果文件已存在，那么解释输出会附加到该文件。如果出于 SET EXPLAIN 语句的文件存在，那么在动态解释关闭之前不会使用该文件。
session_id	指定特定的会话。	无。
-Y	动态地更改 SET EXPLAIN 语句的值。	无。

您可以使用 SET EXPLAIN 语句显示优化器的查询计划、对返回行数的估计，和查询的相对开销。当使用 onmode -Y 命令打开 SET EXPLAIN 时，输出显示在解释输出文件中。

onmode -Y 命令动态地更改单独会话的 SET EXPLAIN 语句的值。使用该命令时，以下调用是有效的：

调用	解释
onmode -Y *session_id* 2	打开对 *session_id* 的 SET EXPLAIN。
onmode -Y *session_id* 1	打开对 *session_id* 的 SET EXPLAIN，并在解释输出文件显示查询统计。
onmode -Y *session_id* 1 /tmp/myexplain.out	打开对 *session_id* 的 SET EXPLAIN，并将解释输出写入 /tmp/myexplain.out。
onmode -Y *session_id* 0	关闭对 *session_id* 的 SET EXPLAIN。

此命令有等效的 SQL 管理 API 函数。

相关链接

EXPLAIN_STAT 配置参数 在第111页
onmode 和 *Y* 参数: 更改会话的查询计划度量 *(SQL* 管理 *API)* 在第666页
onmode 和 *Y* 参数: 更改会话的查询计划度量 *(SQL* 管理 *API)* 在第666页
《*SinoDB SQL* 指南: 语法》: *SET EXPLAIN* 语句
《*SinoDB SQL* 指南: 语法》: 使用 *FILE TO* 选项
《*SinoDB SQL* 指南: 语法》: *UNIX* 上说明输出文件的缺省名称和位置
《*SinoDB SQL* 指南: 语法》: *Windows* 上输出文件的缺省名称和位置
《*SinoDB* 性能指南》: 显示由优化器选择的查询计划的报告
《*SinoDB* 性能指南》: 说明输出文件
《*SinoDB* 性能指南》: 查询统计信息部分提供性能调试信息

onmode -z: 杀死数据库服务器会话

语法:
onmode -z*sid*

元素	用途	关键注意事项
-z ***sid***	杀死 *sid* 所指定的会话	该值必须是大于 0 的无符号整数，且必须是当前正在运行的会话的会话标识号。

要使用 -z 选项，首先使用 onstat -u 获取会话标识号（*sessid*），然后执行 onmode -z，以该会话标识号替换 *sid*。

当使用 onmode -z 时，数据库服务器将尝试杀死指定的会话。如果数据库服务器成功，那么它会释放该会话占用的所有资源。如果数据库服务器无法释放这些资源，那么它不杀死该会话。

如果会话不退出扇区或释放锁存器，那么数据库服务管理员使数据库服务器脱机（如使用 *-k* 选项使数据库服务器处于脱机模式 在第340页所述）以关闭所有会话。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode 和 *z* 参数: 终止用户会话 *(SQL* 管理 *API)* 在第667页

onmode -Z: 杀死分布式事务

语法:
onmode -Z*address*

元素	用途	关键注意事项
-Z ***address***	杀死与共享内存地址 *address* 相关联的分布式事务	该参数必须是已超出 TXTIMEOUT 所指定时间量的正在进行的分布式事务的地址。该地址必须符合特定于=操作系统的对共享内存寻址的规则。（该地址可从 onstat -x 中获得。） 直到超过 ONCONFIG 参数 TXTIMEOUT 所指定的时间量之后，该选项才有效。-Z 选项应少使用，并只由分布式事务所涉及的数据库服务器的管理员使用。 有关启动两阶段提交协议中的独立操作的信息，请参阅《*SinoDB*® 管理员指南》中的“多阶段提交协议”一章。

分布式事务 提供查询不同数据库服务器上的数据的能力。

注意： 如果应用程序正在执行分布式事务，那么杀掉分布式事务之一将使客户端/服务器数据库系统处于不一致状态。请设法避免这种情况。

此命令有等效的 SQL 管理 API 函数。

相关链接

onmode 和 Z 参数: 终止分布式事务 (SQL 管理 API) 在第667页

onparams 实用程序

使用 onparams 实用程序添加或删除逻辑日志文件、更改物理日志参数，以及添加新缓冲池。

本章中

本章显示如何使用以下 onparams 选项：

- *onparams -a -d dbspace:* 添加逻辑日志文件 在第352页
- *onparams -d -l lognum:* 删除逻辑日志文件 在第352页
- *onparams -p:* 更改物理日志参数 在第353页
- *onparams -b:* 添加缓冲池 在第354页

如果正在进行存储空间备份，那么任何 onparams 命令都会失败。如果不使用任何选项，那么 onparams 返回用法说明。

在高可用性数据复制（HDR）辅助服务器、远程独立（RS）辅助服务器或共享磁盘（SD）辅助服务器上不可使用 onparams 实用程序。

您还使用等同于 onparams 命令的 SQL 管理 API 函数来添加或删除逻辑日志文件、更改物理日志参数，以及添加新缓冲池。

在 UNIX™ 上，您必须作为用户 root 或用户 informix 登录才可执行 onparams。仅允许用户 informix 执行 SQL 管理 API 命令字串符。

在 Windows™ 上，您必须是 Informix-Admin 组的成员才可执行 onparams。

相关链接

LOGFILES 配置参数 在第124页

RTO_SERVER_RESTART 配置参数 在第153页

onparams 语法

使用onparams 实用程序修改逻辑日志或物理日志的配置。

onparams
[< -FILE option >]
[{ -a -d*dbspace* | -d -l*lognum* | -p | -b | -V | -version }]

元素	用途	关键注意事项
-V	显示软件版本号和序列号	请参阅获取实用程序版本信息 在第289页。
-version	显示构建版本、主机、操作系统、编号、日期以及 GLS 版本	请参阅获取实用程序版本信息 在第289页.

onparams -a -d *dbspace*：添加逻辑日志文件

语法：

```
onparams -a -ddbspace
[  -ssize ]
[  -i ]
```

元素	用途	关键注意事项
-a -d *dbspace*	在指定的 *dbspace* 的日志文件列表末尾添加逻辑日志文件	仅当数据库服务器具有足够的连续空间时才可向数据库空间添加日志文件。新添加的日志文件具有状态 A，并立即可以使用。可在备份过程中添加日志文件。最多可有 32,767 个逻辑日志文件。使用 onstat -l 查看逻辑日志文件的状态。建议尽早对根数据库空间和包含日志文件的数据库空间采用 0 级备份。 不可向 BLOB 空间或智能大对象空间添加日志文件。 语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。
-i	在当前日志文件的后面插入逻辑日志文件	当“请求日志文件”警告提示您添加逻辑日志文件时使用此选项。
-s *size*	指定新 逻辑日志文件的大小（以 KB 为单位）	该值必须是大于或等于 200 KB 的无符号整数。 如果未使用 -s 选项指定大小，那么使用数据库服务器磁盘空间初始化时 ONCONFIG 文件中 LOGSIZE 参数的值作为日志文件的大小。 有关更改 LOGSIZE 的信息，请参阅《*SinoDB®* 管理员指南》“管理逻辑日志文件”一章。

此命令有等效的 SQL 管理 API 函数。

相关链接

DYNAMIC_LOGS 配置参数 在第109页
add log 参数: 添加新逻辑日志 *(SQL* 管理 *API)* 在第585页

onparams -d -l *lognum*：删除逻辑日志文件

语法：

```
onparams -d -llognum
[  -y ]
```

元素	用途	关键注意事项
-d -l *lognum*	允许您删除用日志文件号指定的	限制：*lognum* 值必须是大于或等于 0 的无符号整数。 可以从 onstat -l 的 number 字段获得 ***lognum***。***lognum*** 的顺序可能是无序的。
-y	使得数据库服务器自动地对所有提示响应“是”	无。

用法

一次仅可删除一个日志文件。

任何时候，数据库服务器最少需要三个逻辑日志文件。如果逻辑日志仅由三个日志文件组成，那么就不能删除日志文件。

重要： 在您删除首三个逻辑日志文件中任一个之前，您必须添加新的逻辑日志文件并执行逻辑日志文件备份。必须使用 ontape -a 命令或 ontape -c 命令执行备份。在添加新的逻辑日志文件并执行备份之后，可以使用 onparams -d -l*lognum* 来删除首三个逻辑日志文件。

日志文件的状态决定日志文件是否可以删除，以及删除日志文件时数据库服务器可以执行的操作：

- 如果删除尚未写入过且状态为“新添加”（A）的日志文件，那么数据库服务器会删除该日志文件并立即释放其空间。
- 如果删除状态为“已使用”（U）或“空闲”（F）的日志文件，那么数据库服务器将标记该日志文件为“已删除”（D）。在对包含日志文件和根数据库空间的数据库空间执行了 0 级备份之后，数据库服务器将删除该日志文件并释放其空间。
- 无法删除当前正在使用（C）或包含最后检查点记录（L）的日志文件。

此命令有等效的 SQL 管理 API 函数。

当您将逻辑日志文件移动到另一个数据库空间时，请使用这些 onparams 命令来添加和删除逻辑日志文件。请参阅《*SinoDB*® 管理员指南》中“管理逻辑日志文件”一章中的“移动逻辑日志文件”。

相关链接

drop log 参数: 删除逻辑日志 (SQL 管理 API) 在第624页

onparams -p: 更改物理日志参数

语法：

```
onparams -p
-ssize
[  -ddbspace ]
[  -y ]
```

元素	用途	关键注意事项
-p	更改物理日志	当使用 onparams -p 命令时必须包含 -s 参数。此外，可以指定 -d 和 -y 参数。数据库服务器必须处于 管理、联机或静默模式，才可指定 -p 参数。无需重新启动数据库服务器，更改就可生效。
-s *size*	更改物理日志的大小（以 KB 为单位）	该值必须是大于或等于 200 KB 的无符号整数。 注意： 如果您将日志移动到不具有足够连续空间的数据库空间或将日志大小增至超过可用的连续空间，那么该操作将失败，并且物理日志不会更改。
-d *dbspace*	将物理日志的位置更改到指定的 *dbspace*	分配给物理日志的空间必须连续。 语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。
-y	史地数据库服务器自动对所有提示响应“是”	无。

相关链接

PHYSFILE 配置参数 在第144页

alter plog 参数: 更改物理日志 (SQL 管理 API) 在第589页

LOGSIZE 配置参数 在第126页

更改物理日志大小或位置之后进行备份

当您更改物理日志时，数据库服务器必须是处于联机或静默模式。无需重新启动数据库服务器即可使更改生效。

更改物理日志大小或位置之后，立即创建根数据库空间的 0 级备份。此备份对于正确恢复数据库服务器是极为重要的。

更改物理日志的大小和使用非缺省页大小

如果您使用非缺省页大小，那么您可能需要增加物理日志的大小。如果您对非缺省页执行了多次更新，那么您可能需要将物理日志大小增加 150% 到 200%。可能需要一些试验来调整物理日志。您可以根据填充物理日志从而触发检查点的频率来按需要调整物理日志的大小。

onparams -b: 添加缓冲池

使用 onparams -b 命令来创建对应于数据库空间的页大小的缓冲池。

语法

onparams -b -g*size*

元素	用途	关键注意事项
-b	创建缓冲池	您可以在数据库服务器正在运行的同时添加缓冲池。
-g *size*	指定要创建的缓冲页的大小，以 KB 计	缓冲页的大小必须是 2 - 16 KB且为缺省页大小的倍数。

您创建的缓冲池的所有其他特性都将设置为 BUFFERPOOL 配置参数的缺省行中的字段的值。

使用非缺省页大小创建的每个数据库空间必须有一个具有对应页大小的缓冲池。如果使用页大小创建的数据库空间没有缓冲池，那么系统会自动基于 BUFFERPOOL 参数的缺省行中的字段自动创建缓冲池。

当添加一个缓冲池时，将会在 onconfig 文件中为 BUFFERPOOL 配置参数添加一个新的条目。

此命令有等效的 SQL 管理 API 函数。

相关链接

add bufferpool 参数: 添加缓冲池 *(SQL* 管理 *API)* 在第583页

BUFFERPOOL 配置参数 在第78页

onparams 命令的示例

以下为 onparams 命令的示例：

```
onparams -a -d rootdbs -s 1000  # adds a 1000-KB log file to rootdbs

onparams -a -d rootdbs -i       # inserts the log file after the current log

onparams -d -l 7                # drops log 7

onparams -p -d dbspace1 -s 3000 # resizes and moves physical-log to dbspace1

onparams -b -g 6 -n 3000 -r 2 -x 2.0 -m 1.0 # adds 3000 buffers of size
6K bytes each with 2 LRUS with maximum dirty of 2% and minimum dirty of 1%
```

ifxclone 实用程序

使用 ifxclone 实用程序从现有的数据库服务器的快照创建服务器克隆。

语法

```
ifxclone { <Mandatory parameters> <Optional parameters> | help }
必要参数
必要参数
--source = source_name --sourceIP = source_IP --sourcePort = source_port
--target = target_name --targetIP = target_IP --targetPort = target_port
可选参数
可选参数
[ --configparm parameter = value ]
[ --autoconf ]
[ --disposition = { ER | HDR | RSS | SDS } ]
[ --useLocal ]
[ --size = size ]
[ --trusted ]
[ --createchunkfile ]
```

元素	用途	关键注意事项
disposition	指定新的服务器实例的最终部署。	如果未指定 --disposition (-d) 参数，则创建标准服务器。
ER	指定将新的服务器实例创建为复制服务器。	
HDR	指定将新的服务器实例创建为 HDR 辅助服务器。	
parameter=value	在目标服务器上指定可选配置参数和其要设置的值。	源服务器上的某些配置参数必须与目标服务器上的匹配。请参阅克隆 *RS* 辅助服务器的先决条件 在第359页.
RSS	指定将新的服务器实例创建为远程单独的辅助服务器。	
SDS	指定将新的服务器实例创建为共享磁盘辅助服务器。	ifxclone 实用程序设置目标服务器的 SDS_PAGING 和 SDS_TEMPDBS 配置参数，但完整配置超出了 ifxclone 实用程序的范围。 如果在命令中指定 --disposition=SDS 但未指定 --useLocal，那么就必须将 SD 辅助服务器的 ROOTPATH 配置参数设置成与主服务器的 ROOTPATH 配置参数相同的值。
size	只定目标服务器的大小。有效值是 tiny、 small、 medium 和 large。	如果未设置该参数，那么使用源实例的大小参数。
source_name	指定源实例的名称。	源服务器必须是主服务器，不可是辅助服务器。
source_IP	指定源服务器实例的 TCP/IP 地址。	

元素	用途	关键注意事项
source_port	指定源服务器实例的 TCP/IP 端口或与该端口相关联的服务名称。	
target_name	指定目标服务器实例的名称。	
target_IP	指定目标服务器实例的 TCP/IP 地址。	
target_port	指定目标服务器实例的 TCP/IP 端口或与该端口相关联的服务名称。	

下表描述 ifxclone 实用程序的选项。

长格式	短格式	意义
--autoconf	-a	在新克隆的服务器与高可用性集群或 Enterprise Replication 域中的其他服务器之间自动配置连接信息。如果该选项用于创建复制服务器，那么 --autoconf 选项可以自动配置复制。 --autoconf 选项有以下要求： • 在目标服务器、源服务器和所有其他集群或复制服务器上， CDR_AUTO_DISCOVER 配置参数必须设置为 1。 • 在所有集群和复制服务器上必须设置 REMOTE_SERVER_CFG。 • 目标服务器的主机信息必须存在源服务器的可信主机文件中。 • 如果与 --disposition=ER 一起使用，且主服务器是 Enterprise Replication 中的一部份，那么域中的所有其他复制服务器必须处于活动的。
--configParm	-c	指定目标服务器上要设置的配置参数的名称和值。
--createchunkfile	-k	在目标服务器上自动创建与源服务器相同的熟块文件。 此选项不创建原始块。然而，如果在源服务器上有原始块请在使用此选项之前不在目标服务器上创建匹配的原始块，那么 ifxclone 实用程序将在目标服务器上创建与源服务器上原始块匹配的块。
--disposition	-d	指定新服务器实例的部署。
--help	-h	显示用法信息。
--size	-s	指定目标实例的大小。
--source	-S	指定源服务器实例的名称。
--sourceIP	-I	指定源服务器实例的 TCP/IP 地址。
--sourcePort	-P	指定源服务器实例的 TCP/IP 端口或与该端口相关联的服务名称。
--target	-t	指定目标服务器实例的名称。
--targetIP	-i	指定目标服务器实例的 TCP/IP 地址。

长格式	短格式	意义
--targetPort	-p	指定目标服务器实例的 TCP/IP 端口或与该端口相关联的服务名称。
--trusted	-T	指定服务器是可信的，且访问该服务器不需要使用用户标识和密码。
--useLocal	-L	指定源服务器 onconfig 文件中的配置信息应与目标服务器 onconfig 文件合并。 源服务器上的某些配置参数必须与目标服务器上的匹配。请参阅See 所有服务器的先决条件 在第358页.

用法

使用 ifxclone 实用程序以最少设置或配置来克隆服务器，或者快速地添加新节点到现有 ER 复制域。当 ifxclone 实用程序运行时，大多数设置信息是从正在被克隆的服务器节点获得的。成功克隆服务器可能仍需要一些后配置步骤以实现更好的运行系统。

源服务器是包含您想要克隆的数据的服务器。目标服务器是要从源服务器加载数据的服务器。您必须从目标服务器来运行 ifxclone 实用程序。

要在 UNIX™ 计算机上运行 ifxclone 实用程序，您必须以用户 root、用户 informix、或 informix 组的成员的身份在目标服务器上运行该命令。您还必须是源服务器上的 DBSA。

要在 Windows™ 上运行 ifxclone 实用程序，您必须以本地管理员组的成员身份在目标服务器上运行该命令。您还必须是服务器上的 DBSA，并且是源服务器上 Informix-Admin 组的成员。

ifxclone 实用程序使用源服务器的 onconfig 和 sqlhosts 配置文件来配置目标服务器。ifxclone 实用程序还配置一些仅配置克隆服务器所需的其他配置设置。--autoconf 选项提供配置 sqlhosts 文件记录的附加能力，并将 sqlhosts 和可信主机文件信息传播到高可用性集群的服务器或 Enterprise Replication 域。--createchunkfile 选项在目标系统上创建与源服务器上相同的熟块和熟镜像块。ifxclone 实用程序不是要配置所有可能的配置选项，而是提供足够的配置以克隆源服务器。

目标服务器上的 CPU VP 和缓冲池的数量可使用 size 参数来配置。 表 *136: size* 参数值列表 在第357页列出目标服务器上为每个 size 选项创建的 CPU VP 和缓冲池的数量。生成配置的其他细部优化应在创建目标系统之后执行。如果省略 size 配置，那么使用原服务器上的参数配置。

表 136: size 参数值列表

Size	CPU VP 的数量	缓冲池的数量
tiny	1	50,000
small	2	100,000
medium	4	250,000
large	8	500,000

可以使用 -c 选项在目标服务器上指定配置参数和其值。您还可以使用现有的配置文件。如果目标服务器上的配置文件与源服务器上的配置文件中的内容不同，ifxclone 并不会覆盖掉该文件而是修改克隆过程中必须匹配源服务器的那些参数。

如果目标服务器与源服务器位于相同主机上，则需要 useLocal (-L) 参数。

如果指定了 useLocal 参数，那么 ifxclone 实用程序会将源服务器 onconfig 文件与目标服务器的 onconfig 文件合并。ifxclone 会覆盖掉在所有服务器的先决条件 在第358页中所列的配置参数，其余参数不受影响。

如果未将 useLocal 参数指定为输入参数，那么 ifxclone 实用程序将以源服务器的 onconfig 文件作为目标服务器的 onconfig 文件，并使用 ifxclone 实用程序的输入参数中的服务器名称。

如果未指定 useLocal 参数，那么 ifxclone 实用程序将使用目标服务器的条目来更新主机服务器上的 sqlhosts 文件，并将两边的条目复制到目标服务器的 sqlhosts 文件。

ifxclone 参数的优先顺序如下：

- --configParm (-c) 参数优先于原服务器上的配置文件。
- --size (-s) 参数优先于合并的配置参数或本地配置文件中的设置。
- --configParm (-c) 参数优先于 --size (-s) 参数。
- 在每个服务器上必须相同的参数优先于所有其他选项。

所有服务器的先决条件

在克隆服务器之前，请执行以下先决条件：

- 服务器的硬件和软件要求通常与 HDR 辅助服务器的相同（对于特定支持平台请参阅机器说明）。
- 源服务器与目标服务器必须是可信网络环境的一部份。有关配置可信任环境的信息，请参阅《*SinoDB*® 管理员指南》中的“网络安全性文件”。
- 如果目标服务器的配置指定为 ER 或 RSS，那么您必须提供用户对原服务器上 sysadmin 数据库的连接权限。在缺省情况下，对 sysadmin 数据库的连接权限仅限于用户 informix。
- 一次只能有一个服务器克隆进程。在第一个克隆进程完成之前，请不要开始克隆第二个服务器。
- 必须将源服务器的 onconfig 文件中的 ENABLE_SNAPSHOT_COPY 配置参数设置为 1。
- 目标服务器不能有任何旧的 ROOTPATH 页。如果目标服务器有旧的 ROOTPATH 页，那么创建零长度的 ROOTPATH 文件或将目标服务器的 onconfig 文件中的 FULL_DISK_INIT 配置参数设置为 1。

克隆服务器时不允许归档操作，例如 ontape 和 ON-Bar 命令。在开始克隆服务器之前执行数据归档活动。

在克隆服务器之前，必须在目标服务器上设置以下环境变量：

- INFORMIXDIR
- INFORMIXSERVER
- INFORMIXSQLHOSTS
- ONCONFIG

以下配置参数的值在源服务器和目标服务器上必须相同：

- DRAUTO
- DRINTERVAL
- DRTIMEOUT
- LOGBUFF
- LOGFILES
- LOGSIZE
- LTAPEBLK
- LTAPESIZE
- ROOTNAME
- ROOTSIZE
- PHYSBUFF
- PHYSFILE
- STACKSIZE
- TAPEBLK
- TAPESIZE

如果启用了目标服务器上的 MIRROR 配置参数，那么以下配置参数在源服务器和目标服务器之间也必须匹配：

- MIRRORPATH
- MIRROROFFSET

数据库服务器仅允许源服务器和目标服务器上 MIRROR 配置参数的某些组合。请参阅表 137: 源服务器和目标服务器上允许的 MIRROR 配置参数的设置 在第359页。

表 137: 源服务器和目标服务器上允许的 MIRROR 配置参数的设置

MIRROR 配置参数在原服务器上的设置	MIRROR 配置参数在目标服务器上的设置	允许或不允许
No	No	允许
Yes	Yes	允许
Yes	No	允许
No	Yes	不允许。如果配置此设置，那么服务器会发出警告，并禁用目标服务器 onconfig 文件中的 MIRROR 参数。

克隆 RS 辅助服务器的先决条件

1. 在目标服务器上设置以下环境变量：
   - INFORMIXDIR
   - INFORMIXSERVER
   - ONCONFIG
   - INFORMIXSQLHOSTS
2. 在目标服务器上，创建存在源服务器上的所有块和镜像块。如果目标服务器正在使用镜像，那么镜像块路径必须与源服务器的匹配，且块必须存在。您可以使用 --createchunkfile 选项 (-k) 自动在目标服务器上创建熟块。采用这些步骤为目标服务器创建块和镜像块（如果需要）：
   a. 在源服务器上，运行 onstat -d 命量显示块和镜像块的列表：

      ```
      onstat -d
      ```

   b. 在目标服务器上，以用户 informix 身份登录，并使用 touch、chown 和 chmod 命令来创建 onstat -d 命令报告的块和镜像块。例如：要创建名为 /usr/informix/chunks/rootdbs.chunk 的块，请遵循这些步骤：

      ```
      $ su informix
      Password:
      $ touch /usr/informix/chunks/rootdbs.chunk
      $ chown informix:informix /usr/informix/chunks/rootdbs.chunk
      $ chmod 660 /usr/informix/chunks/rootdbs.chunk
      ```

   c. 对 onstat -d 命令报告的每个块，重复前步骤的所有命令。
3. 在目标系统上使用适当参数运行 ifxclone 实用程序。
4. 可选地，在目标服务器上创建 onconfig 和 sqlhosts 文件。

> 示例 1：使用原服务器配置克隆 RS 辅助服务器
>
> 此示例显示如何使用源服务器的 onconfig 和 sqlhosts 配置文件来克隆服务器。
>
> 此示例中，省略 -L 选项导致 ifxclone 实用程序从源服务器取得需要的配置信息。该配置文件用作为样板以创建目标服务器配置。使用 ifxclone 实用程序为您创建配置文件，既省时也减少将错误引入配置文件的机会。
>
> -k 选项在目标服务器上创建需要的熟块。
>
> 此示例，假设源服务器（Amsterdam）配置了如下所示的 sqlhosts 文件：
>
> ```
> #Server    Protocol HostName        Service Group
> ```

```
Amsterdam onsoctcp 192.168.0.1 123     -
```

您还需要目标服务器的名称、IP 地址和端口号。此示例使用以下的值：

- 源服务器名称：Amsterdam
- 源 IP 地址：192.168.0.1
- 源端口：123
- 目标服务器名称：Berlin
- 目标 IP 地址192.168.0.2
- 目标端口：456

1. 在目标服务器上，创建存在于原服务器上的所有块。您可以使用 --createchunkfile 选项 (-k) 自动在目标服务器上创建熟块。以用户 informix 身份登录，并使用 touch、chown 和 chmod 命令来创建块。
2. 在目标服务器上运行 ifxclone 实用程序：

   ```
   ifxclone -T -S Amsterdam -I 192.168.0.1 -P 123 -t Berlin
            -i 192.168.0.2 -p 456 -d RSS -k
   ```

   ifxclone 实用程序修改源服务器上的 sqlhosts 文件，并在新的目标服务器上创建该文件的副本。目标服务器上的 sqlhosts 文件与源服务器上的相同：

   ```
   #Server   Protocol HostName        Service Group
   Amsterdam onsoctcp 192.168.0.1 123     -
   Berlin    onsoctcp 192.168.0.2 456
   ```

示例 2：通过合并源服务器配置来克隆 RS 辅助服务器

使用 -L (--useLocal) 选项在远程主计算机上创建服务器克隆： -L 选项用于将源 onconfig 文件的配置信息与目标 onconfig 文件合并。此选项也将源 sqlhosts 文件复制到目标服务器。此示例使用以下值：

- 源服务器名称：Amsterdam
- 源 IP 地址：192.168.0.1
- 源端口：123
- 目标服务器名称：Berlin
- 目标 IP 地址：192.168.0.2
- 目标端口：456

1. 在目标计算机上，创建 onconfig 和 sqlhosts 文件，并设置环境变量。
2. 在目标服务器上，创建存在于源服务器上的所有块。您可以使用 --createchunkfile 选项 (-k) 在目标服务器上自动创建熟块。以用户 informix 身份登录，并使用 touch、chown 和 chmod 命令创建块。
3. 在目标服务器上运行 ifxclone 实用程序：

   ```
   ifxclone -T -L -S Amsterdam -I 192.168.0.1 -P 123 -t Berlin
            -i 192.168.0.2 -p 456 -d RSS -k
   ```

示例 3：添加 RS 辅助服务器到集群

此示例显示如何添加 RS 辅助服务器到现有 SinoDB® 高可用性集群。此示例使用以下值：

- 源服务器名称：Amsterdam
- 源 IP 地址：192.168.0.1
- 源端口：123

- 目标服务器名称：Berlin
- 目标 IP 地址：192.168.0.2
- 目标端口：456

1. 在目标计算机上创建 onconfig 和 sqlhosts 文件，并设置环境变量。
2. 在目标服务器上，创建存在源服务器上的所有块。 您可以使用 --createchunkfile 选项 (-k) 在目标服务器上自动创建熟块。以用户 informix 身份登录，并使用 touch、chown 和 chmod 命令来创建块。
3. 在目标服务器上运行 ifxclone 实用程序：

```
ifxclone -T -L -S Amsterdam -I 192.168.0.1 -P 123 -t Berlin
         -i 192.168.0.2 -p 456 -s medium -d RSS -k
```

克隆 ER 服务器的先决条件

在尝试克隆 ER 服务器之前，请完成以下先决条件。

1. 源服务器（即正被克隆的服务器）必须已配置 ER 且处于活动状态。
2. 对于指定目录名称的配置参数，其目录名称必须存在于目标服务器上。例如：如果 CDR_LOG_STAGING_DIR 配置参数设置为源服务器上的目录名称，那么该目录也必须存在目标服务器上。
3. 如果在源服务器上启用了 ATS 或 RIS，那么相应的 ATS 或 RIS 目录必须存在目标服务器上。请参阅《*SinoDB® Enterprise Replication* 指南》的 启用 *ATS* 和 *RIS* 文件生成和创建 *ATS* 和 *RIS* 目录。如果目录不存在，那么 ATS/RIS 假脱机将失败。
4. 如果源服务器设置了 CDR_SERIAL 配置参数，那么在要克隆的服务器上您必须将 CDR_SERIAL 的值设置为不同的值。CDR_SERIAL 的值在所有复制服务器上都必须不同。可以在 ifxclone 命令行使用 --configParm (-c) 参数为 CDR_SERIAL 配置参数指定唯一值。
5. 在新 ER 克隆上的时钟必须相应地同步。请参阅《*SinoDB® Enterprise Replication* 指南》的时间同步.
6. 源服务器（即正在被克隆的服务器）不可有任何停止或暂停的复制，也不可定义影子复制。

避免执行更改复制集的 ER 管理任务，该复制集为 ifxclone 在运行时目标服务器所参与的复制集。

示例：创建 ER 服务器的克隆

假设您有五个名为 S1、S2、S3、S4 和 S5 的 ER 服务器，目前配置为 ER 域中的根服务器。您想要在名为 machine6 的新计算机上添加一个新的服务器 S6，并且想要其有与服务器 S3 相同的数据。

1. 在 machine6 上安装和配置 SinoDB® 数据库软件。您可以使用部署实用程序部署预先配置的数据库服务器实例。
2. 将 sqlhosts 文件从服务器 S3 复制到服务器 S6，并在其中为新服务器添加条目。例如：假设新服务器的 ER 组名为 g_S6，ID 为 60，则 sqlhosts 文件中的行将看起来如下。

```
g_S6    group     -           -         i=60
S6      onsoctcp  machine6    service6  g=g_S6
```

3. 在其他五个服务器上（S1 到 S5）的 sqlhosts 文件中添加前步骤的两行。
4. 将 onconfig 文件从服务器 S3 复制到服务器 S6，并将 DBSERVERNAME 配置参数更改为 S6。除了路径信息外，不要修改任何存储或块参数。
5. 在服务器 S6（machine6）上设置块路径和与服务器 S3 相同大小的其他存储。确保 S6 有足够内存和磁盘空间资源。您可以使用 --createchunkfile 选项 (-k) 在目标服务器上自动创建熟块。
6. 以用户 informix 身份执行以下命令：

```
ifxclone -L -S S3 -I machine3 -P service3 -t S6 -i machine6 -p service6 -d
 ER -k
```

提示时，请输入用户名 informix，然后输入用户 informix 的密码。

7. 监视服务器 S6 和 S3 的服务器日志。当克隆进程完成时，您可以在服务器 S3 和 S6 上执行以下命令检查服务器的状态：

```
cdr list server
```

您应看到新 ER 服务器 g_S6 连接到其他五个服务器。此外，ER 节点 g_S6 将参与 ER 节点 g_S3 所参与的所有复制。

相关链接

ENABLE_SNAPSHOT_COPY 配置参数 在第111页

《SinoDB Enterprise Replication 指南》：CDR_AUTO_DISCOVER 配置参数

onspaces 实用程序

使用 onspaces 实用程序管理数据库的存储空间。

相关链接

TBLTBLFIRST 配置参数 在第186页

onspaces 语法

执行 onspaces 实用程序命令来管理存储空间。

```
onspaces
[ < -FILE option > ]
{ -a | -c -b | -c -d | -c -P | -c -S | -c -x | -ch | -cl | -d | -f | -m | -r | -
ren | -s | -V | -version }
```

元素	用途	关键注意事项
-V	显示软件版本号和序列号	请参阅获取实用程序版本信息 在第289页
-version	显示构建版本、主机、操作系统、构建编号、日期和 GLS 版本	请参阅获取实用程序版本信息 在第289页

onspaces -a：向数据库空间或 BLOB 空间添加块

语法：

unique_588_Connect_42_note021

```
onspaces -a
{ dbspace blobspace }
{ -p pathname | -p \ \ . \ drive [] }
-o offset
-s size
{ -m pathname offset | -m \ \ . \ drive offset [] }
```

使用 onspaces -a 添加块到数据库空间或 BLOB 空间。

元素	用途	关键注意事项
-a	指示要添加块	一个实例中可以有多达 32766 个块。您可以将所有这些块放在一个存储空间，或分散于多个存储空间。
drive	指定要分配作为未缓冲磁盘空间的 Windows™ 驱动盘。格式可以是 \\.\<***drive***>（***drive*** 是分配给磁盘分区的驱动盘号）或 \\.***PhysicalDrive<number>***（***PhysicalDrive*** 是常数值，***number*** 是物理驱动盘编号）。	有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“在 *Windows* 上分配原始磁盘空间”。 示例：\\.\F: 有关路径名语法，请参阅您的操作系统文件。
-m ***pathname offset***	指定镜像新块的块的可选路径名和偏移量。另见此表中 *pathname* 和 *offset* 的条目。	有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“向数据库空间和 *BLOB* 空间添加块”。
-o ***offset***	在 -a 选项之后，*offset* 指示到达新 BLOBL 空间或数据库空间初始块的磁盘分区或设备的偏移量（以 KB 计）。	无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大的块大小。最大偏移量是 4 TB。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“在 *UNIX* 上分配原始磁盘空间”。
-p ***pathname***	指示正在添加 BLOB 空间或数据库空间的初始块的磁盘分区或未缓冲设备。 块必须是现存的未缓冲设备或已缓冲文件。	块路径名最多可有 256 字节。当您指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于初始化数据库服务器时的那个当前目录。 UNIX™ 示例（未缓冲设备）： /dev/rdsk/c0t3d0s4 UNIX™ 示例（已缓冲设备）：/ix/ids9.2/db1chunk Windows™ 示例：c:\Ifmxdata\ol_icecream\mychunk1.dat 有关路径名语法，请参阅您的操作系统文件。
-s ***size***	指示新 BLOB 空间或数据库空间块的大小（以 KB 计）	无符号整数。大小必须等于或大于 1000 KB，且是页大小的倍数。起始偏移量加上块大小不可超过最大的块大小。最大偏移量是 4 TB。
blobspace	给出块将添加至的 BLOB 空间的名称	请参阅《*SinoDB*® 管理员指南》中的“向数据库空间和 *BLOB* 空间添加块”。 语法必须符合《*SinoDB*® *SQL* 指南*:* 语法》中的“*Identifier*”。
dbspace	给出块将添加至的数据库空间的名称	请参阅《*SinoDB*® 管理员指南》中的“向数据库空间和 *BLOB* 空间添加块”。 语法必须符合《*SinoDB*® *SQL* 指南*:* 语法》中的“*Identifier*”。

此命令有等效的 SQL 管理 API 函数。

相关链接

避免覆盖块 在第384页
add chunk 参数*:* 添加新块 *(SQL* 管理 *API)* 在第584页
create chunk 参数*:* 创建块 *(SQL* 管理 *API)* 在第605页

onspaces -a：向智能大对象空间添加块

语法：
onspaces -a
sbspace
-p*pathname*
-o*offset*
-s*size*
[-m*pathname offset*]
[-Ms*mdsize*]
[-Mo*mdoffset*]
[-U]

使用 onspaces -a 向智能大对象空间添加块。

元素	用途	关键注意事项
-a	指示要添加块	一个实例可以有多达 32766 个块。您可以将所有这列块放在一个存储空间，或者分散在多个存储空间。
-m ***pathname offset***	指定镜像新块的块的可选路径名和偏移量。另见此表中的 ***pathname*** 和 ***offset*** 的条目。	有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“向智能大对象空间添加块”。
-Mo ***mdoffset***	指示应存储元数据的磁盘分区或设备的偏移量	值可以是 0 到块大小之间的整数。不可指定导致元数据空间结束处超过块结束处的偏移量。 有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“调整智能大对象空间元数据的大小”。
-Ms ***mdsize***	指定在初始块中分配的元数据区域的大小（以 KB 计）。其余为用户数据空间。	值可以是 0 到 块大小之间的整数。 有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“调整智能大对象空间元数据的大小”。
-o ***offset***	在 -a 选项之后，*offset* 指示到达新 BLOB 空间或数据库空间初始块的磁盘分区或未缓冲设备的偏移量。	无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。最大偏移量是 2 或 4 TB，其取决于平台。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“在 UNIX™ 上分配原始磁盘空间”。
-p ***pathname***	指示正在创建的智能大对象空间初始块的磁盘分区或未缓冲设备的偏移量。 块必须是现有的未缓冲设备或已缓冲文件。	块路径名最多可以有 256 字节。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于初始化数据库服务器时的当前目录的目录。 有关路径名语法，请参阅您的操作系统文件。
-U	指定应用于存储用户数据的整块	-M 合 -U 选项是互斥的。 有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“向智能大对象空间添加块”。

元素	用途	关键注意事项
-s ***size***	指示新智能大对象空间块的大小（以 KB 计）	无符号整数。大小必须等于或大于 1000 KB，且是页大小的倍数。起始偏移量加上块大小不可超过最大的块大小。 最大偏移量是 4 TB。
sbspace	给出正在添加块的智能大对象空间的名称	请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“向智能大对象空间添加块”。 语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。

此命令有等效的 SQL 管理 API 函数。

相关链接

避免覆盖块 在第384页
add chunk 参数: 添加新块 *(SQL* 管理 *API)* 在第584页
create chunk 参数: 创建块 *(SQL* 管理 *API)* 在第605页

onspaces -c -b: 创建 BLOB 空间

语法：
unique_593_Connect_42_blob01

```
onspaces -c
{   -b blobspace  -g pageunit }
{   -p pathname  |  -p \ \ . \ drive  []  }
-o offset
-s size
[{   -m pathname offset  |  -m \ \ . \ drive offset  []  }]
```

使用 onspaces -c -b 创建 BLOB 空间。

元素	用途	关键注意事项
-b ***blobspace***	Names the blobspace to be created	BLOB 空间名必须唯一且不超过 128 字节。必须以字母或下划线开始，且必须只包含字母、数字、下划线或 $ 字符。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“创建 BLOB 空间”。语法必须符合 Identifier 段。有关更多信息，请参阅《*SinoDB*® *SQL* 指南: 语法》。
-c	创建数据库空间、BLOB 空间、智能大对象空间或外部空间。 最多可创建 2047 个任意类型的存储空间。	在创建了存储空间之后，必须备份该存储空间和根数据库空间。如果创建了与已删除存储空间相同名称的存储空间，请执行另一个 0 级备份以确保以后的恢复不会混淆新的存储空间与旧的存储空间。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“创建数据库空间、BLOB 空间或外部空间”。

元素	用途	关键注意事项
drive	指定分配作为未缓冲磁盘空间的 Windows™ 驱动盘。 格式可以是 \\.\<***drive***>（***drive*** 是指定给磁盘分区的驱动盘）或 \\.***PhysicalDrive***<***number***>（***PhysicalD...*** 是常数值，***number*** 是物理驱动盘号）。	有关分配未缓冲磁盘空间的信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“在 Windows™ 上分配未缓冲磁盘空间”。示例： ``` \\.\F: \\.\PhysicalDrive2 ``` 有关路径名语法，请参阅您的操作系统文件。
-g ***pageunit***	根据 ***page_unit*** 指定 BLOB 空间 BLOB 页大小。***page_unit*** 是实例基本页大小（2K 或 4K）的数量。	无符号整数。值必须大于 0。 BLOB 空间可包含的最大页数为 2147483647。因此，BLOB 空间的大小限制为 BLOB 页大小 x 2147483647。这个包括构成 BLOB 空间的所有块中的 BLOB 页。 有关更多信息，请参阅《*SinoDB*® 性能指南》中的“I/O 活动”一章中的“BLOB 页大小注意事项”。
-m ***pathname offset***	指定镜像新 BLOB 空间或数据库空间初始块的块的可选路径名和偏移量。 另见此表中的 -p ***pathname*** 和 -o ***offset*** 的条目。	有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“创建数据库空间或 BLOB 空间”。
-o ***offset***	指示到达新 BLOB 空间、数据库空间或智能大对象空间初始块的磁盘分区或设备的偏移量（以 KB 计）。	无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。最大偏移量是 2 TB 或 4 TB，其取决于平台。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“分配原始磁盘空间”。
-p ***pathname***	指示 正在创建的 BLOB 空间或数据库空间初始块的磁盘分区或设备	块必须是现有的未缓冲设备或已缓冲文件。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，那么必须是相对于初始化数据库服务器时的当前目录的目录。UNIX™ 示例（未缓冲设备）：/dev/rdsk/c0t3d0s4。UNIX™ 示例（已缓冲设备）：/ix/ids9.2/db1chunk。Windows™ 示例：c:\Ifmxdata\ol_icecream\mychunk1.dat 有关路径名语法，请参阅您的操作系统文件。
-s ***size***	指示新 BLOB 空间或数据库空间初始块的大小（以 KB 计）。	无符号整数。大小必须等于或大于 1000 KB，且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小是 2 TB 或 4 TB，取决于平台。

此命令有等效的 SQL 管理 API 函数。

相关链接

避免覆盖块 在第384页
create blobspace 参数: 创建 *BLOB* 空间 *(SQL* 管理 *API)* 在第603页

onspaces -c -d: 创建数据库空间

使用 onspaces -c -d 命令创建数据库空间或临时数据库空间。

语法
unique_243_Connect_42_note020

```
onspaces -c
{    -d dbspace }
{   -p pathname  |  -p \ \ . \ drive }
-o offset
-s size
[{ [  -ef extentsize  -en extentsize ]  | -t }]
[{   -m pathname offset  |  -m \ \ . \ drive offset  []  }]
[  -k pagesize ]
```

元素	用途	关键注意事项
-c	创建数据库空间 最多可以创建 2047 个任意类型的存储空间。	创建了存储空间之后，必须备份该存储空间和根数据库空间。如果创建了与已删除存储空间相同名称的存储空间，请执行另一个 0 级备份，以确保以后的恢复不会混淆新存储空间和旧存储空间。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理数据库空间”。
drive	指定分配作为未缓冲磁盘空间的Windows™ 驱动盘 格式可以是 \\.*drive*（*drive* 是指定给磁盘分区的驱动盘符）或 \\.*PhysicalDrivenumber*（*Physic* 是常数值，*number* 是物理驱动盘号）。	有关分配未缓冲磁盘空间的信息，请参阅《*SinoDB*® 管理员指南》中的“在 *Windows* 上分配原始磁盘空间”。 示例： \\.\F: \\.\PhysicalDrive2 有关路径名语法，请参阅您的操作系统文件。
-d *dbspace*	给出要创建的数据库空间的名称	数据库空间名称必须唯一且不可超过 128 字节。必须使用字母或下划线为开始，且必须只包含字母、数字、下划线和 $ 字符。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理数据库空间”。语法必须符合 Identifier 段。有关更多信息，请参阅《*SinoDB*® *SQL* 指南: 语法》中的“*Identifier*”。
-ef *extentsize*	指示表空间 tblspace 第一个扩展数据块的大小（以 KB 计）	非根数据库空间的表空间 tblspace 的第一个扩展数据块的最小和缺省大小等于 50 个数据库空间页（以 KB 为单位）。例如：对于页大小为 2 KB 的数据库空间，该大小为 100 KB；对于页大小为 4 KB 的数据库空间，该大小为 200 KB；对于页大小为 8 KB 的数据库空间，该大小为 400 KB。 表空间 tblspace 扩展数据块最大的大小是 1048575 个页减去所有系统对象所需的空间。在页大小为 2 KB 的系统上，最大的大小约等于 2 GB。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“为表空间 *tblspace* 指定第一个和下一个扩展数据块大小”。

元素	用途	关键注意事项
-en *extentsize*	指示表空间 tblspace 中下一个扩展数据块的大小（以 KB 计）	非根数据库空间的表空间 tblspace 的下一个扩展数据块的最小大小等于 4 个数据库空间页（以 KB 为单位）。例如：对于页大小为 2 KB 的数据库空间，该大小为 8 KB；对于页大小为 4 KB 的数据库空间，该大小为 16 KB；对于页大小为 8 KB 的数据库空间，该大小为 32 KB。 下一个扩展数据块的缺省大小为 50 个数据库空间页。 表空间 tblspace 的最大大小是 1048572 个页。在页大小为 2 KB 的系统上，最大大小约等于 2 GB。 如果在主要块中没有足够空间分配给下一个扩展数据块，那么扩展数据块将从另一个块分配。如果指定的空间不可用，那么将分配最接近的可用空间。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“为表空间 *tblspace* 指定第一个和下一个扩展数据块大小”。
-k *pagesize*	指示新数据库空间的非缺省页大小（以 KB 为单位）。 对于具有足够空间的系统，使用较大的页大小具有以下性能优点： • 可降低 B 型树的深度，即使是对于较小的索引 • 可将跨越多个页（页大小为缺省页大小）的长行组合到同一页中 • 检查点时间通常随着页增大而减少 • 可以为临时表定义不同的页大小，这样他们将具有独立的缓冲池。	页大小必须在 2 KB 到 16 KB 之间，且是缺省页大小的倍数。例如：如果缺省页大小为 2 KB，那么 *pagesize* 可以是 2、4、6、8、10、12、14 或 16。如果缺省页大小为 4 KB（Windows™），那么 *pagesize* 可以是 4、8、12 或 16。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“创建具有非缺省页大小的数据库空间”。
-m *pathname offset*	指定镜像新数据库空间初始块的块的可选路径名和偏移量 另见此表中的 -p *pathname* 和 -o *offset* 的条目。	有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理数据库空间”。
-o *offset*	指示到达新数据库空间初始块的磁盘分区或设备的偏移量（以 KB 为单位）	无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。偏移量必须是页大小的倍数。 最大偏移量是 2 或 4 TB，取决于平台。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“在 *Windows*上分配原始磁盘空间”。
-p *pathname*	指示正在创建的数据库空间初始块的磁盘分区或设备	块必须是现有的未缓冲设备或已缓冲文件。当指定路径名时，可使用完整路径名或相对路径名。然而，如果使用相对路径名，那么必须相对于初始化数据库服务器时的当前目录的目录。UNIX™ 示例（未缓冲设备）：/dev/rdsk/c0t3d0s4；UNIX™ 示例（已缓冲

元素	用途	关键注意事项
		设备）：/ix/ids9.2/dblchunk；Windows™ 示例：c:\Ifmxdata\ol_icecream\mychunk1.dat 有关路径名语法，请参阅您的操作系统文件。
-s *size*	指示新数据库空间初始块的大小（以 KB 为单位）	无符号整数。大小必须等于或大于 1000 KB，且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小为 2 或 4 TB，取决于平台。
-t	创建临时数据库空间用于临时表的存储	不可镜像临时数据库空间。不可为临时数据库空间的表空间 tblspace 指定第一个和下一个扩展数据块大小。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“临时数据库空间”。

数据库空间的最大大小等于最大块数乘最大块大小。（一个实例的最大块数是 32766。最大块大小等于 2147483647 个页乘页大小。）

此命令有等效的 SQL 管理 API 函数。

数据库空间的页大小在创建之后就不可更改。

您不能将逻辑日志或物理日志存储在非缺省平台页大小的数据库空间中。

如果具有该页大小的的缓冲池不存在时创建数据库空间，那么 SinoDB® 将使用 BUFFERPOOL 参数的 default 行的字段值来创建缓冲池。

临时数据库空间

当使用 onspaces 创建临时数据库空间时，在将新临时数据库空间的名称添加到 DBSAPCETEMP 配置参数、DBSPACETEMP 环境变量或两者中的临时数据库空间列表并重新启动服务器之后，数据库服务器将使用该新创建的临时数据库空间。

不可指定临时数据库空间的第一个和下一个扩展数据块的大小。对于 2 KB 页大小的系统，临时数据库空间的扩展数据块大小为 100 KB；对于 4 KB 页大小的系统，其为 200 KB。

如果您不想要数据库服务器自动管理大小，您可以指定根数据库空间中的表空间 tblspace 的第一个和下一个空间的大小。要指根表空间 tblspace 的第一个和下一个扩展数据块的大小，请在首次启动数据库服务器创建根数据库空间之前，使用 TBLTBLFIRST 和 TBLTBLNEXT 配置参数。

相关链接

DBSPACETEMP 配置参数 在第91页
避免覆盖块 在第384页
TBLTBLFIRST 配置参数 在第186页
TBLTBLNEXT 配置参数 在第187页
《*SinoDB 管理员指南*》：为表空间 *tblspace* 指定第一个和下一个扩展数据块大小
create dbspace 参数: 创建数据库空间 (SQL 管理 API) 在第608页
create tempdbspace 参数: 创建临时数据库空间 (SQL 管理 API) 在第616页

onspaces -c -P: 创建物理日志空间

使用 onspaces -c -P 命令创建存储物理日志记录的物理日志空间。

语法

unique_597_Connect_42_1182note020

```
onspaces -c
{    -P plogspace }
{   -p pathname  |  -p \ \ . \ drive }
-o offset
-s size
[{   -m pathname  offset  |  -m \ \ . \ drive offset  [] }]
```

元素	用途	关键注意事项
-c	创建物理日志空间。	一个实例只可有一个物理日志空间。如果存在物理日志空间，那么创建新的物理日志空间会将物理日志搬移到新的空间并删除旧物理日志空间。
-m *pathname offset*	指定镜像新物理日志空间块的块的可选路径名和偏移量。 请查看该表中的 -p *pathname* 和 -o *offset*。	如果镜像了物理日志空间，那么物理日志空间块无法扩展。

-m \\.*drive*	指定镜像新物理日志空间块的块的Windows™ 驱动盘。
	drive 为分配给磁盘分区的驱动盘符，或常数值和物理驱动盘号。
		示例：
		`\\.\F:`
		`\\.\PhysicalDrive2`
		有关驱动盘名称的语法，请参阅您的操作系统文件。

-o *offset*	指示到达新物理日志空间块的磁盘分区或设备的偏移量（以 KB 计）。	无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大的块大小。 偏移量必须是页大小的倍数。最大偏移量是 2 或 4 TB，取决于平台。
-P *plogspace*	给出要创建的物理日志空间的名称。	物理日志空间名称必须是唯一的，且不可超过 128 字节。必须以字母或下划线为开始，且只可包含字母、数字、下划线或 $ 字符。 语法必须符合 Identifier 段。有关更多信息，请参阅《*SinoDB® SQL* 指南: 语法》中的“*Identifier*”。
-p *pathname*	指示正在创建的物理日志空间块的磁盘分区或设备。	块必须是现有的未缓冲设备或已缓冲文件。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，那么它必须是相对于数据库服务器初始化时的当前目录的目录。 UNIX™ 示例（未缓冲设备）： /dev/rdsk/c0t3d0s4 UNIX™ 示例（缓冲设备）： /ix/ifmx/db1chunk Windows™ 示例： c:\Ifmxdata\ol_icecream\mychunk1.dat

-p \\.*drive*	指定要分配作为物理日志空间的未缓冲磁盘空间的 Windows™ 驱动盘。
	drive 是指定给磁盘分区的驱动盘符，或常数值和物理驱动盘号。
		示例：
		`\\.\F:`
		`\\.\PhysicalDrive2`
		有关驱动盘名称的语法，请参阅您的操作系统文件。

元素	用途	关键注意事项
-s *size*	指示新物理日志空间块的大小。	无符号整数。该大小必须等于或大于 1000 KB 且是页大小的倍数。起始偏移量加上块大小不可超过最大的块大小。 最大的块大小是 2 或 4 TB，取决于平台。

示例

以下示例创建名为 plogdbs 的物理日志空间，其大小为 40000 KB，偏移量为 0：

```
onspaces -c -P plogdbs -p /dev/chk1 -o 0 -s 40000
```

以下示例创建名为 pdbs1 的镜像物理日志空间，其大小为 60000 KB，偏移量为 500 KB：

```
onspaces -c -P pdbs1 -p /dev/pchk1 -o 500 -s 60000 -m /dev/mchk1 0
```

相关链接

《SinoDB 管理员指南》: 物理日志空间

《SinoDB 管理员指南》: 管理物理日志空间

create plogspace: 创建物理日志空间 *(SQL 管理 API)* 在第611页

onspaces -c -S: 创建智能大对象空间

使用 onspaces -c -S 选项创建智能大对象空间或临时智能大对象空间。

语法：

onspaces -c
-S*sbspace*
[-t]
-p*pathname*
-o*offset*
-s*size*
[-m*pathname offset*]
[-Ms*mdsize*]
[-Mo*mdoffset*]
[-Df*default list*]

元素	用途	关键注意事项
-S ***sbspace***	给出要创建的智能大对象空间的名称	智能大对象空间名称必须是唯一的且不可超过 128 字节。必须以字母或下划线为开始，且必须只包含字母、数字、下划线，或 $ 字符。 语法必须符合 Identifier 段；请参阅《*SinoDB® SQL 指南: 语法*》。
-c	创建智能大对象空间 最多可创建 32767 个任何类型的存储空间。	无。
-m ***pathname offset***	指定镜像新智能大对象空间初始块的块的可选路径名和偏移量。	有关更多信息，请参阅《*SinoDB®* 管理员指南》中的“数据存储”一章中的“智能大对象空间”和“管理磁盘空间”一章的“创建智能大对象空间”。

元素	用途	关键注意事项
	另见该表中的 -p ***pathname*** 和 -o ***offset***的条目。	
-Mo ***mdoffset***	指示存储元数据的磁盘空间或设备的偏移量（以 KB 计）。	限制：值可以是 0 到块大小之间的整数。不可指定导致元数据空间末尾超出块末尾的偏移量。 限制：有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章的“缩放智能大对象空间元数据大小”。
-Ms ***mdsize***	指定初始块中分配给元数据区的大小（以 KB 计） 其余为用户数据空间。	限制：值可以是 0 到块大小之间的整数。
-o ***offset***	指示到达智能大对象空间初始块的磁盘分区或设备的偏移量（以 KB 计）	限制：无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。对于具有 2 KB 页大小的系统，最大块大小是 4 TB；对于具有 4 TB 页大小的系统，最大块大小是 8 TB。 限制：有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章的“在 UNIX™ 上分配原始磁盘空间”。
-p ***pathname***	指示智能大对象空间初始块的磁盘分区或未缓冲设备	块必须是现有的未缓冲设备或已缓冲文件。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于数据库服务器初始化时的当前目录的目录 限制：有关路径名语法，请参阅您的操作系统文件。
-s ***size***	指示心智能大对象空间初始块的大小（以 KB 计）	限制：无符号整数。大小必须等于或大于 1000 KB 且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小是 2 TB 或 4 TB，取决于平台。
-t	创建连石智能大对象空间用于存储临时智能大对象。可以指定元数据区的大小和偏移量。	限制：不可镜像临时智能大对象空间。可以指定任何 -Df 选项，但 LOGGING=ON 选项除外，其无效。 限制：有关更多信息，请参阅使用 *-t* 选项创建临时智能大对象空间 在第373页。
-Df ***default list***	列出存储在智能大对象空间中的智能大对象的缺省规范	限制：用逗号分隔标记。如果没有标记，那么优先使用系统缺省值。在命令行上，该列表必须括在双引号中（″）。 限制：有关标记和其参数的列表，请参阅 *表 138: -Df 缺省规范* 在第373页。

相关链接

SBSPACENAME 配置参数 在第155页
SBSPACETEMP 配置参数 在第156页
SYSSBSPACENAME 配置参数 在第184页
避免覆盖块 在第384页
create sbspace 参数: 创建智能大对象空间 (SQL 管理 API) 在第612页
create tempsbspace 参数: 创建临时智能大对象空间 (SQL 管理 API) 在第617页

create sbspace with log 参数: 创建具有事务日志记录的智能大对象空间 *(SQL* 管理 *API)* 在第615页

使用 -t 选项创建临时智能大对象空间

此示例创建 1000 KB 的临时智能大对象空间：

```
onspaces -c -S tempsbsp -t -p ./tempsbsp -o 0 -s 1000
```

可以选择性地在 SBSPACETEMP 配置参数中指定临时智能大对象空间的名称。重新启动数据库服务器以便它可以使用该临时智能大对象空间。

使用 -Df 选项创建智能大对象空间

当使用 -Df 选项创建智能大对象空间时，可以指定几个缺省规范，这些规范会影响智能大对象空间中智能大对象的行为。缺省规范必须用逗号区隔的列表表示。该列表不需要包含所有标记。标记列表必须括在双引号中（″）。表 *138: -Df* 缺省规范 在第373页 中的表描述标记描述了标记及其缺省值。

智能大对象空间特定的四个继承级别是系统、智能大对象空间、列以及智能大对象。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“数据存储位置”一章中的“智能大对象”。

表 138: -Df 缺省规范

标记	值	缺省	描述
ACCESSTIME	ON 或 OFF	OFF	设置为 ON 时，数据库服务器跟踪对存储在智能大对象空间中所有智能大对象的访问时间。 有关更改智能大对象的存储特性，请参阅《*SinoDB*® *DataBlade*® *API* 程序员指南》。
AVG_LO_SIZE	Windows™：4 到 2**31 UNIX™：2 到 2**31	8	指定存储在智能大对象空间中的智能大对象的平均大小（以 KB 为单位） 数据库服务器使用该值来计算元数据区的大小。不同时指定 AVG_LO_SIZE 和 -Ms。可以同时指定 AVG_LO_SIZE 和元数据偏移量（-Mo）。 如果智能大对象的大小超过 2**31，请指定 2**31。如果智能大对象大小在 UNIX™ 小于 2 或在 Windows™ 上小于 4，那么请指定 2 或 4。 如果耗尽智能大对象空间中元数据和保留区中的空间，那么返回错误 131。要将额外块分配给仅由元数据区构成的智能大对象空间，请使用 -Ms 选项。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中“管理磁盘空间”一章中的“创建智能大对象”。
BUFFERING	ON 或 OFF	ON	指定存储在智能大对象空间中的智能大对象的缓冲模式 如设置为 ON，那么对于智能大对象 I/O 操作，数据库服务器使用共享内存常驻部份中的缓冲池。如果设置为 OFF，那么数据库服务器使用共享内存虚拟部份中的轻量级 I/O 缓冲区（轻量级 I/O 操作）。 BUFFERING = OFF 与 LOCK_MODE = RANGE 不兼容，会产生冲突 有关更多信息，请参阅《*SinoDB*® 性能指南》中“内存上配置影响”一章中“轻量级 I/O”。
LOCK_MODE	RANGE 或 BLOB	BLOB	指定存储于智能大对象空间中的智能大对象的锁定模式。

标记	值	缺省	描述
			如果设置为 RANGE，那么只锁定智能大对象中一定范围内的字节。如果设置为 BLOB，那么锁定整个智能大对象。 LOCK_MODE = RANGE 与 BUFFERING = OFF 不兼容，且会产生冲突。 有关更多信息，请参阅《*SinoDB*® 性能指南》中的“锁定”一章中的“智能大对象”。
LOGGING	ON 或 OFF	OFF	指定存储于智能大对象空间中的智能大对象的日志记录状态 如果设置为 ON，那么数据库服务器将更改记录到智能大对象空间的用户数据区中。当打开智能大对象空间的日志记录时，对智能大对象空间执行 0 级备份。 当关闭日志记录时，显示以下消息：You are turning off smart large object logging。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“数据存储和日志记录”一章中的“智能大对象”。有关 onspaces -ch 消息的信息，请参阅数据库服务器日志中的消息 在第776页。
EXTENT_SIZE	4 到 2**31	无	指定在创建表时首次分配给存储于智能大对象空间中的智能大对象的磁盘空间的大小 让系统选择 EXTENT_SIZE 值。要减少智能大对象中的阔染数据块数，请使用 mi_lo_specset_estbytes（DataBlade® API）或 ifx_lo_specset_estbytes（SinoDB® ESQL/C）对系统指示智能大对象的总计大小。系统尝试为智能大对象分配单个扩展数据块。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“数据存储位置”一章中的“智能大对象”。有关更改智能大对象的存储特性，请参阅《*SinoDB*® *DataBlade*® *API* 程序员指南》或《*SinoDB*® *ESQL/C* 程序员指南》。
MIN_EXT_SIZE	2 到 2**31	Windows™：4，UNIX：2	指定要分配给每个智能大对象的最小空间量（以 KB 为单位） 显示以下消息：Changing the sbspace minimum extent size: old value *value1* new value *value2*. 有关调整此值的信息，请参阅《*SinoDB*® 性能指南》中的“配置对 I/O 使用率的影响”一章中的“智能大对象”。有关 onspaces -ch 消息的信息，请参阅数据库服务器日志中的消息 在第776页。
NEXT_SIZE	4 到 2**31	无	指定在智能大对象空间初始扩展数据块已满时，下次分配给智能大对象的磁盘空间扩展数据块的大小（以 KB 为单位）。让系统选择 NEXT_SIZE 值。要减少智能大对象中的扩展数据块数，请使用 mi_lo_specset_estbytes 或 ifx_lo_specset_estbytes 对系统指示智能大对象的总计大小。系统尝试为智能大对象分配单个扩展数据块。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的 “数据存储位置”一章中的“智能大对象”。有关获取智能大对象的大小的信息，请参阅《*SinoDB*® *DataBlade*® *API* 程序员指南》或《*SinoDB*® *ESQL/C* 程序员指南》。

此示例使用以下规范创建 20 MB 镜像智能大对象空间 eg_sbsp：

- 主块和镜像块的偏移量为 500 KB
- 元数据区的偏移量为 200 KB
- 平均期望智能大对象大小为 32 KB
- 将更改记录到智能大对象空间用户数据区中的智能大对象中

仅 UNIX:

```
% onspaces -c -S eg_sbsp -p /dev/raw_dev1 -o 500 -s 20000
   -m /dev/raw_dev2 500 -Mo 200 -Df "AVG_LO_SIZE=32,LOGGING=ON"
```

更改 -Df 设置

作为数据库服务器管理员，您可以下列方式之一重设或更改 -Df 缺省设置：

- 要更改智能大对象空间的缺省设置，请使用 onspaces -ch 选项。有关更多信息，请参阅*onspaces -ch:* 更改智能大对象空间缺省规范 在第376页。
- 要重设特定表的以下 -Df 缺省设置，请使用 SQL 语句 CREATE TABLE 或 ALTER TABLE：
  - LOGGING
  - ACCESSTIME
  - EXTENT_SIZE
  - NEXT_SIZE

  有关 ALTER TABLE 和 CREATE TABLE 语句的更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。

程序员可以使用 DataBlade® API 和 SinoDB® ESQL/C 函数重设这些 -Df 缺省设置。有关更改智能大对象的存储特性的信息，请参阅《*SinoDB® DataBlade® API* 程序员指南》和《*SinoDB® ESQL/C* 程序员指南》。

使用 onspaces -g 选项

onspaces -g 选项不用于智能大对象空间。数据库服务器使用对于智能大对象空间使用不同于 BLOB 空间的方法确定在 I/O 操作中要传送的页数。数据库服务器可自动确定在智能大对象的 I/O 操作中要传送的块大小。有关更多信息，请参阅《*SinoDB®* 性能指南》中的“I/O 活动”一章中的“智能大对象空间扩展数据块大小”。

此命令有等效的 SQL 管理 API 函数。

onspaces -c -x: 创建外部空间

使用 onspaces -c -x 选项创建外部空间。
语法：
onspaces -c
-x*extspace* -l*location*
-o*offset*
-s*size*

元素	用途	关键注意事项
-c	创建数据库空间、BLOB 空间、智能大对象空间或外部空间 最多可创建 2047 个任何类型的存储空间。	在创建了存储空间之后，必须备份该存储空间和根数据库空间。如果创建与已删除存储空间相同名称的存储空间，请执行另一个 0 级备份以确保以后的恢复不混淆新存储空间和旧存储空间。 有关更多信息，请参阅《*SinoDB®* 管理员指南》中的“管理磁盘空间”一章的“创建数据库空间、BLOB 空间或外部空间”。

元素	用途	关键注意事项
-l ***location***	指定外部空间的位置 访问方式决定该字符的格式。	限制：字串符。值不得超过 255 字节。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“创建外部空间”。
-o ***offset***	指示到达新的 BLOB 空间、数据库空间或智能大对象空间初始块的磁盘分区或设备的偏移量（以 KB 为单位）	限制：无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。最大偏移量是 2 或 4 TB，取决于平台。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“分配原始磁盘空间”。
-s ***size***	指示新的 BLOB 空间或数据库空间初始块的大小（以 KB 为单位）	限制：无符号整数。大小必须等于或大于 1000 KB 且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小是 2 或 4 TB，取决于平台。
-x ***extspace***	给出要创建的外部空间的名称	限制：外部空间名称最多可有 128 字节。其必须是唯一的，以字母或下划线为开始，且只包含字母、数字、下划线或 $ 字符。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“外部空间”。

onspaces -ch: 更改智能大对象空间缺省规范

使用 onspaces -ch 选项更改智能大对象空间缺省规范。

语法：

onspaces -ch
sbspace
-Df*default list*

元素	用途	关键注意事项
-ch	指示要更改一个或多个智能大对象空间缺省规范	无。
sbspace	给出要更改缺省规范的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关后台信息，请参阅《*SinoDB*® 性能指南》中的“使用 onspaces 更改智能大对象空间缺省规范”。
-Df ***default list***	列出存储于智能大对象空间中的智能大对象的新缺省规范	用逗号分隔标记。如果未提供标记，那么优先使用系统缺省值。在命令行上，该列表必须括在双引号（″）中。 有关标记及其参数的列表，请参阅 表 138: -Df 缺省规范 在第373页。

使用 onspaces -ch 选项可以更改任何 -Df 标记。数据库服务器将更改应用于缺省规范更改之前创建的每个智能大对象。

例如：要关闭在使用 *-Df* 选项创建智能大对象空间 在第373页中创建的智能大对象空间的日志记录，请使用以下命令：

```
onspaces -ch eg_sbsp -Df "LOGGING=OFF"
```

注： 在打开智能大对象空间的日志记录之后，请对智能大对象空间执行 0 级备份以创建恢复点。

相关链接

set sbspace accesstime 参数: 控制访问时间跟踪 *(SQL* 管理 *API)* 在第683页

set sbspace avg_lo_size 参数: 设置智能大对象的平均大小 *(SQL* 管理 *API)* 在第684页

set sbspace logging 参数: 更改智能大对象空间的日志记录 *(SQL* 管理 *API)* 在第685页

onspaces -cl: 清除智能大对象空间中的游离智能大对象

使用 onspaces -cl 选项清除智能大对象空间中的游离智能大对象。

语法：

onspaces -cl

sbspace

元素	用途	关键注意事项
-cl	清除智能大对象空间中的游离智能大对象	要查找出任何游离智能大对象，请在没有用户连接到数据库服务器时使用 oncheck -pS 命令。引用计数为 0 的智能大对象是游离对象。
sbspace	给出要清除的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。

在正常运作过程中，不应存在任何未引用（游离）智能大对象。当删除智能大对象时，释放该空间。如果正在删除智能大对象时数据库服务器失败或耗尽系统内存，那么该智能大对象可能保留为游离对象。

以下是 onspaces -cl 命令的示例：

```
onspaces -cl myspace
```

查找智能大对象引用计数的最好方式是从 C 程序中调用 mi_lo_stat 或 ifx_lo_stat 函数。虽然 mi_lo_increfcount 和 mi_lo_decrefcount 函数返回引用计数，但它们会增加或减少该引用计数。有关这些函数的更多信息，请参阅《*SinoDB® DataBlade® API* 函数参考》。

此命令有等效的 SQL 管理 API 函数。

相关链接

clean sbspace 参数: 释放未引用的智能大对象 *(SQL* 管理 *API)* 在第603页

onspaces -d: 删除数据库空间、BLOB 空间或智能大对象空间中的块

使用 onspaces -d 选项删除数据库空间、BLOB 空间或智能大对象空间中的块。

语法：

onspaces -d

{ *dbspace blobspace* | [-f] *sbspace* }

-p*pathname*

-o*offset*

[-y]

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-d	删除块	当数据库服务器处于联机或静默模式时，您可以删除数据库空间、临时数据库空间或智能大对象空间的块。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章。 仅当数据库服务器处于静默模式时，才可删除 BLOB 空间的块。
-f	删除包含用户数据但不包含元数据的智能大对象空间块。如果块包含智能大对象空间的元数据，那么必须删除整个智能大对象空间。	只对智能大对象空间使用 -f 选项。如果省略 -f 选项，则无法删除包含数据的智能大对象空间。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“使用 onspaces 删除智能大对象空间块”。
-o ***offset***	指示到达正在删除的数据库空间、BLOB 空间或智能大对象空间初始块的磁盘分区或未缓冲设备的偏移量（以 KB 为单位）	限制：无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。 最大偏移量是 4 TB。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“在 UNIX™ 上分配原始磁盘空间”。
-p ***pathname***	指示正在删除的数据库空间或智能大对象空间初始块的磁盘分区或未缓冲设备的偏移量	块必须是现有的未缓冲设备或已缓冲文件。当您指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于数据库服务器初始化时的当前目录的目录。 有关路径名语法，请参阅您的操作系统文件。
-y	致使数据库服务器自动对所有提示响应“是”	无。
blobspace	给出要删除其块的 BLOB 空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“从 BLOB 空间删除块”。
dbspace	给出要删除其块的数据库空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“使用 onspaces 从数据库空间删除块”。
sbspace	给出要删除其块的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“使用 onspaces 从数据库空间删除块”。

重要： 必须指定路径名以指示数据库服务器您正在删除块。

相关链接

drop chunk 参数*:* 删除块 *(SQL* 管理 *API)* 在第621页
drop plogspace: 删除物理日志空间 *(SQL* 管理 *API)* 在第625页

onspaces -d: 删除空间

使用 onspaces -d 选项删除数据库空间、BLOB 空间、物理日志空间、智能大对象空间或外部空间。

语法：

onspaces -d
{ *dbspace blobspace plogspace* | [-f] *sbspace extspace* }
[-y]

元素	用途	关键注意事项
-d	指示要删除的存储空间	当数据库服务器处于联机或静默模式时，您可以删除数据库空间、BLOB 空间、物理日志空间、智能大对象空间或外部空间。在删除存储空间之后，必须对其进行备份，以确保 sysutils 数据库和保留页是最新的。 执行 oncheck -pe 来验证当前没有表正向数据库空间、BLOB 空间或智能大对象空间存储数据。
-y	致使数据库服务器自动地对所有提示响应“是”	无。
-f	删除包含用户数据和元数据的智能大对象空间	必须使用 -f（强制）选项来删除包含数据的智能大对象空间。 限制： 仅对智能大对象空间使用 -f 选项。 警告： 如果使用 -f 选项，那么数据库服务器中的表可能含有指向已用此选项删除的智能大对象的死指针。
blobspace	给出要删除的 BLOB 空间的名称	在删除 BLOB 空间之前，请删除包含引用 BLOB 空间的 TEXT 或 BYTE 列的所有表。
dbspace	给出要删除的数据库空间的名称	在删除数据库空间之前，请删除先前在数据库空间中创建的所有数据库和表。
extspace	给出要删除的外部空间的名称	如果外部空间与现有表或索引相关联，则无法删除。
plogspace	给出要删除的物理日志空间的名称	物理日志空间必须是空的才可删除。
sbspace	给出要删除的智能大对象空间的名称	在删除智能大对象空间之前，请删除包含引用智能大对象空间的 BOLB 或 CLOB 列的所有表。

重要： 当删除这些存储空间时不要指定路径名。

此命令有等效的 SQL 管理 API 函数。

相关链接

drop blobspace 参数: 删除 Blob 空间 (SQL 管理 API) 在第620页
drop dbspace 参数: 删除数据库空间 (SQL 管理 API) 在第623页
drop sbspace 参数: 删除智能大对象空间 (SQL 管理 API) 在第626页
drop tempdbspace 参数: 删除临时数据库空间 (SQL 管理 API) 在第627页

onspaces -f: 指定 DATASKIP 参数

使用 onspaces -f 选项在数据库空间级别或所有数据库空间上指定 DATASKIP 配置参数的值。

语法：

onspaces -f

{ OFF | ON }
[*dbspace-list*]
[-y]

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-f	向数据库服务器指示您想要更改指定数据库空间或所有数据库空间的 DATASKIP 缺省值	DATASKIP 状态的所有变更都记录在消息日志。
-y	致使数据库服务器自动地对所有提示响应“是”	无。
dbspace-list	指定将打开（ON）或关闭（OFF）其 DATASKIP 的一个或多个数据库空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB® SQL* 指南: 语法》。有关更多信息，请参阅 *DATASKIP* 配置参数 在第87页和《*SinoDB®* 性能指南》。
OFF	关闭 DATASKIP	如果使用 OFF 时未使用 *dbspace-list*，那么对所有分段关闭 DATASKIP。如果使用 OFF 时也使用 *dbspace-list*，那么仅指定的分段被设置为 DATASKIP 关闭。
ON	打开 DATASKIP	如果使用 ON 时未使用 *dbspace-list*，那么对所有分段打开 DATASKIP。如果使用 ON 时也使用 *dbspace-list*，那么仅指定的分段被设置为 DATASKIP 打开。

相关链接

set dataskip 参数: 启动或停止跳过数据库空间 *(SQL* 管理 *API)* 在第680页
DATASKIP 配置参数 在第87页

onspaces -m: 启动镜像

使用 onspaces -m 选项来启动数据库空间、BLOB 空间或智能大对象空间的镜像。
语法：

onspaces -m
{ *dbspace blobspace sbspace* }
{ , | -p*pathname* -o*offset* -m*pathnameoffset* | -f*filename* }
[-y]

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-f ***filename***	指示块位置信息是在 *filename* 所指定的文件中	该文件必须是现有已缓冲文件。路径名必须符合特定于操作系统的路径名规则。 有关更多信息，请参阅以 *-f* 选项使用文件来指定块位置信息 在第381页。
-m	为现有数据库空间、BLOB 空间或智能大对象空间添加镜像	镜像智能大对象空间中的用户数据块不需要进行镜像。 镜像块应在不同磁盘上。必须同时镜像所有块。
-m ***pathname offset***	*pathname* 第二次出现在语法图中，其指示执行镜像的数据库空间、BLOB 空间或智能大对象空间	无。

元素	用途	关键注意事项
	的初始块的磁盘分区或未缓冲设备。 *offset* 第二次出现在语法图中，其指示到达新镜像数据库空间、BLOB 空间或智能大对象空间的镜像块的偏移量。另见此表中的 *pathname* 和 *offset*。	
-o ***offset***	*offset* 第一次出现在语法图中，其指示到达新镜像数据库空间、BLOB 空间或智能大对象空间的初始块的磁盘分区或未缓冲设备的偏移量（以 KB 为单位）。	限制：无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。 最大偏移量是 4 TB。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“在 UNIX™ 上分配原始磁盘空间”。
-p ***pathname***	*pathname* 第一次出现在语法图中，其指示要镜像的数据库空间、BLOB 空间或智能大对象空间的初始块的磁盘分区或未缓冲设备。	该块必须是现有未缓冲设备或已缓冲文件。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于数据库服务器初始化时的当前目录的目录。 有关路径名语法，请参阅您的操作系统文件。
-y	致使数据库服务器自动地对所有提示响应“是”	无。
blobspace	给出要镜像的 BLOB 空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章。
dbspace	给出要镜像的数据库空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关后台信息，请参阅 《*SinoDB*® 管理员指南》中的“使用镜像”一章。
sbspace	给出要镜像的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关后台信息，请参阅 《*SinoDB*® 管理员指南》中的“使用镜像”一章。

相关链接

add mirror 参数: 添加镜像块 (SQL 管理 API) 在第586页
start mirroring 参数: 启动存储空间镜像 (SQL 管理 API) 在第690页

以 -f 选项使用文件来指定块位置信息

您可以创建包含块位置信息的文件。然后，在执行 onspaces 时使用 -f 选项向数据库服务器指示该信息位于 ***filename*** 所指定名称的文件中。

文件内容应遵循以下格式，选项以空格隔开，且每组主块和镜像块都在单独行上：

```
primary_chunk_path offset mirror_chunk_path offset
```

如果要镜像的数据库空间包含多个块，那么必须为要镜像的数据库空间中的每个主块指定镜像块。有关对于多块数据库空间启用镜像的示例，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章的“使用 onspaces 为未镜像的数据库空间启动镜像”。

onspaces -r: 停止镜像

使用 onspaces -r 选项结束数据库空间、BLOB 空间或智能大对象空间的镜像。
语法：

```
onspaces -r
{ dbspace blobspace sbspace }
[  -y ]
```

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-r	指示数据库服务器应结束现有数据库空间、BLOB 空间或智能大对象空间的镜像。	有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章。
-y	致使数据库服务器自动地对所有提示响应“是”	无。
blobspace	给出要结束镜像的 BLOB 空间的名称。	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章。
dbspace	给出要结束镜像的数据库空间的名称。	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章。
sbspace	给出要结束镜像的智能大对象空间的名称。	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“使用镜像”一章。

相关链接

stop mirroring 参数: 停止存储空间镜像 (SQL 管理 API) 在第692页

onspaces -ren: 重命名数据库空间、BLOB 空间、智能大对象空间或外部空间

使用 onspaces -ren 选项来重命名数据库空间、BLOB 空间、智能大对象空间或外部空间。
语法：

```
onspaces -ren
{ dbspace blobspace sbspace extspace }
-nname
```

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-ren	致使数据库服务器重命名指定的 BLOB 空间、数据库空间、外部空间或智能大对象空间	限制：当数据库服务器处于静默模式时，可以重命名 BLOB 空间、数据库空间、外部空间或智能大对象空间。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章。
-n *name*	指定 BLOB 空间、数据库空间、外部空间或智能大对象空间的新名称	限制：BLOB 空间、数据库空间、外部空间或智能大对象空间的名称必须是唯一的，且不能超过 128 字节。必须以字母或下划线为开始，且必须只包含字母、数字、下划线或 $ 字符。

元素	用途	关键注意事项
		有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章。语法必须符合 Identifier 段。有关更多信息，请参阅《*SinoDB*® *SQL* 指南: 语法》。
blobspace	给出要重命名的 BLOB 空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章。
dbspace	给出要重命名的数据库空间的名称	限制：不可重命名关键数据库空间，如根数据库空间或包含物理日志的数据库空间。 其他信息：如果对包含在 DATASKIP 列表中的数据库空间进行重命名，那么请使用 onspaces -f 命令以新名称更新 DATASKIP 配置参数。 语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“重命名空间”。
extspace	给出要重命名的外部空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“重命名空间”。
sbspace	给出要重命名的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南: 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“重命名空间”。

相关链接

rename space 参数: 重命名存储空间 *(SQL* 管理 *API)* 在第673页

在 Enterprise Replication 处于活动时重命名数据库空间、BLOB 空间、智能大对象空间或外部空间

在 Enterprise Replication 处于活动时重命名空间（数据库、BLOB 空间、智能大对象空间或外部空间）。

当您将数据库服务器置于静默模式来重命名空间时，Enterprise Replication 将断开连接。然后您就可以重命名空间了。再将数据库服务器置于联机模式之后，服务器将会重新同步。

如果要在另一个服务器上重命名相同空间，那么就必须将那个服务器置于静默模式并单独重命名该空间。在不同的 ER 服务器上的重命名空间之间不传播强制关系；相同的表可在不同空间中。

如果 Enterprise Replication 服务器参与了高可用性数据复制（HDR），那么可以重命名主服务器上的数据库空间，它会自动地传播到辅助服务器。（辅助服务器不可参与 Enterprise Replication。）

重命名空间之后执行归档

在重命名任一空间（外部空间或临时空间除外）之后，请对已重命名的空间和根数据库执行 0 级归档。这将确保您能够将空间恢复到包含重命名数据库空间操作或该操作之后的状态。这也是在执行任何其他类型归档之前所必须的。

onspaces -s: 更改镜像块的状态

使用 onspaces -s 选项来更改非关键数据库空间、BLOB 空间或智能大对象空间中的非主块的镜像块的状态。

语法：

```
onspaces -s
```

{ *dbspace blobspace sbspace* }
-p*pathname*
-o*offset*
{ -D | -O }
[-y]

此命令有等效的 SQL 管理 API 函数。

元素	用途	关键注意事项
-D	指示要关闭块	无。
-o ***offset***	指示到达块的磁盘分区或未缓冲设备的偏移量（以 KB 为单位）	限制：无符号整数。起始偏移量必须等于或大于 0。起始偏移量加上块大小不可超过最大块大小。偏移量必须是页大小的倍数。 最大偏移量是 4 TB。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“管理磁盘空间”一章中的“在 UNIX™ 上分配原始磁盘空间”。
-O	指示要恢复的块并使其联机	无。
-p ***pathname***	指示块的磁盘分区或未缓冲设备	块必须是未缓冲设备或已缓冲文件。当指定路径名时，可以使用完整路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于数据库服务器初始化时的当前目录的目录。 有关路径名语法，请参阅您的操作系统文件。
-s	指示想要更改块状态	限制：可以只更改镜像对中的块或非关键数据库空间中非主块的状态。 有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“更改镜像状态”。
-y	致使数据库服务器自动地对所有提示响应“是”	无。
blobspace	给出想要更改状态的 BLOB 空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“更改镜像状态”。
dbspace	给出想要更改状态的数据库空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关更多信息，请参阅《*SinoDB*® 管理员指南》中的“更改镜像状态”。
sbspace	给出想要更改状态的智能大对象空间的名称	语法必须符合 Identifier 段；请参阅《*SinoDB*® *SQL* 指南*:* 语法》。有关后台信息，请参阅《*SinoDB*® 管理员指南》中的“更改镜像状态”。

相关链接

alter chunk 参数: 更改块状态为联机或脱机 (SQL 管理 API) 在第587页
set chunk 参数: 更改块的状态 (SQL 管理 API) 在第680页

避免覆盖块

其他 SinoDB® 实例不知道与每个 SinoDB® 实例相关联的块。有可能不经意地在被分配作为另一 SinoDB® 实例的块的文件或设备上创建块，从而导致数据损坏。

如果您尝试初始化实例，其中 ROOTPATH 配置参数指定了作为另一实例根块的文件或设备，那么该指令将失败并在 online.log 显示以下消息：

```
DISK INITIALIZATION ABORTED: potential instance overwrite detected.
```

要禁止这个初始化检查，请将配置文件中的 FULL_DISK_INIT 配置参数设置为 1，并试着再次初始化该实例。然而，这初始化检查仅限于根块。即使文件或设备已分配给其他实例，添加数据库空间或块也会成功。

相关链接

onspaces -a: 向数据库空间或 *BLOB* 空间添加块 在第362页
onspaces -a: 向智能大对象空间添加块 在第364页
onspaces -c -b: 创建 *BLOB* 空间 在第365页
onspaces -c -d: 创建数据库空间 在第367页
onspaces -c -S: 创建智能大对象空间 在第371页
create blobspace 参数*:* 创建 *BLOB* 空间 *(SQL* 管理 *API)* 在第603页
create chunk 参数*:* 创建块 *(SQL* 管理 *API)* 在第605页
create dbspace 参数*:* 创建数据库空间 *(SQL* 管理 *API)* 在第608页
create sbspace 参数*:* 创建智能大对象空间 *(SQL* 管理 *API)* 在第612页

onstat 实用程序

onstat 实用程序读取共享内存结构，并在命令执行时提供有关数据库服务器的统计信息。

可以在单个命令中组合多个 onstat 选项标志。显示 onstat 输出时，共享内存的内容可能更改了。onstat 实用程序不会在共享内存上放置任何锁，因此执行该实用程序不会影响性能。

可以使用与 onstat 命令等效的 SQL 管理 API 函数。

相关链接

onstat 参数*:* 监视数据库服务器 *(SQL* 管理 *API)* 在第669页

onstat Portal: 按功能类别排序的 onstat 实用程序命令

本主题中列出按功能类别排序的 onstat 命令。

每个类别代表不同的 SinoDB® 功能，onstat 命令为此提供的故障排除和性能增强的信息是很有用的。以 粗体 显示的命令对于提供故障排除信息是特别有用的。某些 onstat 命令特定于一个类别，而其他提供更多一般信息的命令列于多个类别中。

类别列表

从以下列表中确定适当类别，然后单击指向该类别的 onstat 选项的链结。

- *onstat* 实用程序归档信息选项 在第386页
- *onstat* 实用程序高速缓存信息选项 在第386页
- *onstat* 实用程序压缩选项 在第387页
- *onstat* 实用程序调试选项 在第387页
- *onstat* 实用程序 *Enterprise Replication* 选项 在第388页
- *onstat* 实用程序高可用性复制选项 在第389页
- *onstat* 实用程序 仓库加速器选项 在第390页
- *onstat* 实用程序 *I/O* 选项 在第390页
- *onstat* 实用程序锁和锁存器选项 在第391页

- *onstat* 实用程序日志选项 在第391页
- *onstat* 实用程序内存选项 在第392页
- *onstat* 实用程序网络选项 在第393页
- *onstat* 实用程序性能检查#第一层# 在第393页
- *onstat* 实用程序性能检查#第二层# 在第394页
- *onstat* 实用程序表选项 在第395页
- *onstat* 实用程序线程选项 在第396页
- *onstat* 实用程序用户/会话选项 在第397页
- *onstat* 实用程序虚拟处理器选项 在第397页
- *onstat* 实用程序等待选项 在第398页
- 其他有用的 *onstat* 实用程序选项 在第398页

onstat 实用程序归档信息选项

使用以下 onstat 选项来显示有关归档和恢复的信息。

表 139: onstat 实用程序归档信息选项

命令	参考
onstat -D	打印块 I/O 活动。打印数据库空间读取/写入活动以监视恢复过程。 *onstat -D* 命令: 打印页读取和页写入的信息 在第419页
onstat -g arc	打印每个数据库空间最后提交与任何正在进行的备份。 *onstat -g arc* 命令: 打印归档状态 在第425页

onstat 实用程序高速缓存信息选项

使用以下 onstat 选项来显示有关高速缓存和高速缓存数据的信息，包括缓冲池。

表 140: onstat 实用程序高速缓存信息选项

命令	参考
onstat -b	打印使用中的缓冲页。 *onstat -b* 命令: 显示正在使用的缓冲区的信息 在第405页
onstat -B	打印已使用缓冲区的信息。 *onstat -B* 命令: 显示已用缓冲区的信息 在第406页
onstat -F	打印缓冲队列清除程序和 I/O 的状态。 *onstat -F* 命令: 显示计数 在第420页
onstat -g cac	打印有关所有内存高速缓存或指定高速缓存的摘要与详细信息。 *onstat -g cac* 命令: 显示有关高速缓存的信息 在第432页
onstat -g dic	打印数据字典高速缓存，其包含表的系统目录数据。打印共享内存字典中高速缓存的每个表的信息，一行一个表。 有关更多信息，请参阅《*SinoDB*® 性能指南》。

命令	参考
	onstat -g dic 命令: 打印表信息 在第451页
onstat -g dsc	打印优化器的表分布统计信息 *onstat -g dsc* 命令: 显示分布式高速缓存信息 在第457页.
onstat -g prc	打印存储过程（SPL）例程高速缓存。打印有关 SPL 例程高速缓存的信息。 *onstat -g prc* 命令: 显示使用 *UDR* 或 *SPL* 例程的会话 在第493页
onstat -g ssc	打印数据库服务器在高速缓存中读取 SQL 语句的次数。显示与 onstat -g cac 相同的输出。 有关更多信息，请参阅《*SinoDB*® 性能指南》中的“提升查询性能”。 *onstat -g ssc* 命令: 显示 *SQL* 语句出现次数 在第529页
onstat -g vpcache	打印 CPU 虚拟处理器内存高速缓存。 *onstat -g vpcache* 命令: 显示 *CPU* 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息 在第536页
onstat -h	打印缓冲哈希链信息。 *onstat -h* 命令: 打印缓冲头哈希链信息 在第543页
onstat -p	打印有关缓冲池高速缓存的有效性全局（服务器）信息。 *onstat -p* 命令: 显示概要文件计数 在第550页
onstat -X	打印等待缓冲区的线程。 *onstat -X* 命令: 显示线程信息 在第567页

onstat 实用程序压缩选项

使用以下 onstat 选项打印压缩信息。

表 141: onstat 实用程序压缩选项

命令	参考
onstat -g dsk	打印正在运行的压缩操作的进度。 *onstat -g dsk* 命令: 显示当前正在运行的压缩操作的进度 在第458页
onstat -g ppd	打印分区压缩字典信息。 *onstat -g ppd* 命令: 显示分区压缩字典信息 在第490页

onstat 实用程序调试选项

使用以下 onstat 选项来显示对调试服务器问题有用的信息。

表 142: onstat 实用程序调试选项

命令	参考
onstat -g dmp	打印在指定地址的若干给定字节的原内存。 *onstat -g dmp* 命令: 打印原内存 在第453页
onstat -g src	搜寻共享内存中的模式。注意，在 Intel™ 平台上，内存是字节交换的。 *onstat -g src* 命令: 共享内存中的模式 在第529页
onstat -o	将共享内存内容打印到文件。 *onstat -o* 命令: 共享内存内容输出到文件 在第550页.

onstat 实用程序 Enterprise Replication 选项

使用以下 onstat 选项跟踪 Enterprise Replication 统计信息，并提供故障排除信息。有关 Enterprise Replication 的其他信息，请参阅《*SinoDB® Enterprise Replication* 指南》中说明的 cdr view 和 cdr view profile 命令。

表 143: onstat 实用程序 Enterprise Replication 选项

命令	参考
onstat -g cat	打印 Enterprise Replication 全局目录中的信息。全局目录包含了有关已定义服务器、复制、Enterprise Replication 内每个服务器的复制集的摘要信息。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g cat:* 打印 *ER* 全局目录信息”
onstat -g cdr	打印所有 Enterprise Replication 统计命令的输出。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g cat:* 打印 *ER* 统计信息”
onstat -g cdr config	打印 Enterprise Replication 配置参数和环境变量。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g cdr config:* 打印 *ER* 设置”
onstat -g ddr	打印读取和处理日志记录的 Enterprise Replication 组件的状态。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g ddr:* 打印 *ER* 日志阅读器的状态”
onstat -g dss	打印个别数据同步（事务处理）线程的活动。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g dss:* 打印数据同步线程的统计信息”
onstat -g dtc	打印删除表清除程序活动。放置于删除表中的已删除或已更新行会每隔一段时间清除一次。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g dtc:* 打印删除表清除程序的统计信息 ”

命令	参考
onstat -g grp	打印 Enterprise Replication 分组器统计信息。分组器评估日志记录、将单独的日志记录重构建到原始事务中、打包事务，以及对事务进行排队以便进行传输。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g grp:* 打印分组器统计信息”
onstat -g nif	打印网络接口统计信息。显示网络接口、服务器和服务器间数据传输的状态。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g nif:* 打印网络接口的统计信息”
onstat -g que	打印高级别队列接口（其通用于 Enterprise Replication 的所有队列）的统计信息。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g que:* 打印所有 *ER* 队列的统计信息”
onstat -g rcv	打印接收管理器的统计信息。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g rcv:* 打印接收管理器的统计信息”
onstat -g rep	打印调度管理器队列中的事件。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g rep:* 打印调度管理器队列”
onstat -g rqm	打印 Reliable Queue Manager (RQM) 所管理的低级别队列（发送队列、接收队列、应答发送队列、同步发送队列和控制发送队列）的统计信息和内容。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g rqm:* 打印 *RQM* 队列的统计信息”
onstat -g sync	打印同步状态。 《*SinoDB® Enterprise Replication* 指南》中的“*onstat -g sync:* 打印同步的统计信息”

onstat 实用程序高可用性复制选项

使用以下 onstat 选项来监视高可用性集群环境和连接管理器。

表 144: onstat 实用程序高可用性复制选项

命令	参考
onstat -g cluster	打印高可用性集群信息。 *onstat -g cluster* 命令: 打印高可用性集群信息 在第440页
onstat -g cmsm	打印连接管理器信息。 *onstat -g cmsm command:* 显示连接管理器信息 在第443页
onstat -g dri	打印数据复制信息。

命令	参考
	请参阅《*SinoDB*® 管理员指南》中的监视高可用性数据复制状态。 *onstat -g dri* 命令: 显示高可用性数据复制信息 在第454页.
onstat -g ipl	打印索引页日志记录状态。 *onstat -g ipl* 命令: 打印索引页日志记录状态信息 在第472页
onstat -g laq	打印辅助服务器上队列的信息。 *onstat -g laq* 命令: 打印辅助服务器队列 在第474页
onstat -g proxy	打印高可用性的代理分发器。 *onstat -g proxy* 命令: 打印代理分发器信息 在第494页
onstat -g rss	打印远程独立服务器（RSS）信息。 *onstat -g rss* 命令: 打印 *RS* 辅助服务器信息 在第501页
onstat -g sds	打印共享磁盘辅助服务器（SDS）信息。 *onstat -g sds* 命令: 打印 *SD* 辅助服务器信息 在第509页
onstat -g smx	打印高可用环境中服务器多路复用器组（SMX）连接。 打印数据传输统计信息。 *onstat -g smx* 命令#显示多路复用器组信息 在第524页

onstat 实用程序 仓库加速器选项

使用以下 onstat 选项来显示数据库服务器和 仓库加速器 之间交换的信息。

表 145: onstat 实用程序 仓库加速器选项

命令	参考
onstat -g aqt	打印数据市集及其相关联的加速查询表（AQT）的信息。 *onstat -g aqt* 命令: 打印数据市集和加速查询表的信息 在第422页.

onstat 实用程序 I/O 选项

使用以下 onstat 选项跟踪输入和输出（读和写）活动。

表 146: onstat 实用程序 I/O 选项

命令	参考
onstat -D	打印块 I/O 活动。 *onstat -D* 命令: 打印页读取和页写入的信息 在第419页
onstat -g cpu	打印每个线程的运行时间的统计信息。 *onstat -g cpu*: 显示运行时统计信息 在第447页

命令	参考
onstat -g ioa	打印 onstat -g ioq （队列）、 onstat -g iov（虚拟处理器）和 onstat -g iob（大缓冲区）的合并信息. *onstat -g ioa* 命令: 打印合并的 *onstat -g* 信息 在第467页
onstat -g iob	打印大缓冲区的使用摘要。 *onstat -g iob* 命令: 打印大缓冲区的使用摘要 在第469页
onstat -g iof	打印文件或块的 I/O 统计信息。此选项类似于 onstat -D 选项，但它还显示有关非块、临时和排序工作文件的信息。 *onstat -g iof* 命令: 显示异步 *I/O* 统计信息 在第469页
onstat -g iog	打印 AIO 全局信息。 *onstat -g iog* 命令: 打印 *AIO* 全局信息 在第470页
onstat -g ioq	打印队列读/写统计信息和队列长度。 *onstat -g ioq* 命令: 显示 *I/O* 队列信息 在第471页。另见《*SinoDB*® 性能指南》。
onstat -g iov	打印每个虚拟处理器的异步 I/O 统计信息。 *onstat -g iov* 命令: 显示 *AIO VP* 统计信息 在第472页
onstat -p	打印全局磁盘活动，包括顺序扫描。 *onstat -p* 命令: 显示概要文件计数 在第550页

onstat 实用程序锁和锁存器选项

使用以下 onstat 选项显示有关锁的信息。

表 147: onstat 实用程序锁和锁存器选项

命令	参考
onstat -k	打印有关活动锁的信息。 *onstat -k* 命令: 打印活动锁的信息 在第544页
onstat -L	打印锁可用列表上的锁数。 *onstat -L* 命令: 显示可用锁的数量 在第549页
onstat -p	打印锁请求、锁等待和锁存器等待的全部统计信息。 *onstat -p* 命令: 显示概要文件计数 在第550页
onstat -s	打印锁存器（互斥）的信息。 *onstat -s* 命令: 打印锁存器的信息 在第559页

onstat 实用程序日志选项

使用以下 onstat 选项监视逻辑日志和物理日志。

表 148: onstat 实用程序日志选项

命令	参考
onstat -g ipl	打印高可用性环境中索引页日志记录的信息。 *onstat -g ipl* 命令: 打印索引页日志记录状态信息 在第472页
onstat -l	打印物理日志和逻辑日志的状态，以及日志缓冲。 *onstat -l* 命令: 显示物理和逻辑日志信息 在第546页

onstat 实用程序内存选项

使用以下 onstat 选项监视服务器内存分配与使用的各个方面。

表 149: onstat 实用程序内存选项

命令	参考
onstat -g afr	打印指定会话或共享内存池的已分配内存分段。要获得池名称，请查看 onstat -g mem 选项。 *onstat -g afr* 命令: 打印已分配内存分段 在第421页
onstat -g ffr (*pool name session ID*)	打印会话或共享内存池的可用分段。 *onstat -g ffr* 命令: 打印可用分段 在第461页
onstat -g lmm	打印有关自动低内存管理设置和最近的活动： *onstat -g lmm* 命令: 显示低内存管理信息 在第475页
onstat -g mem	打印会话或池的虚拟共享内存的统计信息。 *onstat -g mem* 命令: 显示池内存统计信息 在第479页
onstat -g mgm	打印内存分配管理器（并行和排序操作）的资源的信息 *onstat -g mgm* 命令: 显示 *MGM* 资源信息 在第480页。另见《*SinoDB*® 性能指南》。
onstat -g nbm	打印非常驻段的块映射。 *onstat -g nbm* 命令: 打印块位图 在第482页
onstat -g rbm	打印常驻段的块映射。 *onstat -g rbm* 命令: 打印共享内存的块映射 在第500页
onstat -g seg	打印内存段的统计信息。 *onstat -g seg* 命令: 显示共享内存段的统计信息 在第512页。另见《*SinoDB*® 管理员指南》。
onstat -g ses	打印会话的信息，包括内存解析。有关详细信息，请使用：onstat -g ses *session_id*。 *onstat -g ses* 命令: 显示会话相关信息 在第513页。另见《*SinoDB*® 性能指南》。

命令	参考
onstat -g stm	打印 SQL 语句的内存使用。 *onstat -g stm* 命令: 打印 *SQL* 语句的内存使用 在第532页
onstat -g stq	打印流队列缓冲区。 *onstat -g stq* 命令: 打印队列信息 在第532页
onstat -g ufr	大印会话或使用中共享内存池的内存池分段。 *onstat -g ufr* 命令: 打印内存池分段 在第535页
onstat -R	打印缓冲池队列及其状态。 *onstat -R* 命令: 显示 *LRU*、*FLRU* 和 *MLRU* 队列信息 在第557页

onstat 实用程序网络选项

使用以下 onstat 选项监视共享内存和网络连接服务。

表 150: onstat 实用程序网络选项

命令	参考
onstat -g nsc	按 *client id* 打印共享内存状态。如果省略 *client id*，那么显示所有客户端状态区域。此命令打印与 nss 命令相同的状态数据。 *onstat -g nsc* 命令: 打印当前共享内存连接信息 在第483页
onstat -g nsd	打印轮询线程的共享内存数据。 *onstat -g nsd* 命令: 打印轮询线程的共享内存数据 在第486页
onstat -g nss	通过 *session id* 打印网络共享内存状态。如果省略 *session id*，那么显示所有会话的状态区域。此命令打印与 onstat -g nsc 命令相同的状态数据。 *onstat -g nss* 命令: 打印共享内存网络连接的状态 在第486页
onstat -g ntd	按服务打印网络的统计信息。 *onstat -g ntd* 命令: 打印网络统计信息 在第487页
onstat -g ntm	打印网络邮件统计信息。 *onstat -g ntm* 命令: 打印网络邮件统计信息 在第488页
onstat -g ntt	打印网络用户次数。 *onstat -g ntt* 命令: 打印网络用户时间 在第488页
onstat -g ntu	打印网络用户统计信息。 *onstat -g ntu* 命令: 打印网络用户统计信息 在第488页

onstat 实用程序性能检查（第一层）

使用以下 onstat 选项监视性能并检查性能障碍。使用第二层 onstat 选项（和其他 onstat 命令）以进一步缩小问题。

表 151: onstat 实用程序性能检查（第一层）

命令	参考
onstat -c	打印服务器信息。 *onstat -c* 命令: 显示 *ONCONFIG* 文件内容 在第408页
onstat -D	打印块 I/O。 *onstat -D* 命令: 打印页读取和页写入的信息 在第419页
onstat -g ath	打印所有线程的状态和统计信息。sqlexec 线程是客户端会话线程。rstcb 值对应于 onstat -u 命令的用户字段。 *onstat -g ath* 命令: 显示所有线程的信息 在第426页。有关使用 onstat -g ath 打印 Enterprise Replication 线程的信息，请参阅《*SinoDB® Enterprise Replication* 指南》。
onstat -g ckp	打印检查点历史记录，并显示建议的配置。 *onstat -g ckp* 命令: 打印检查点历史记录与配置建议 在第434页
onstat -g cpu	打印每个线程运行时间的统计信息。 *onstat -g cpu:* 显示运行时统计信息 在第447页
onstat -g ioq	打印 *queue name* 暂挂的 I/O 操作。 *onstat -g ioq* 命令: 显示 *I/O* 队列信息 在第471页
onstat -p	打印全局服务器性能概要文件。 *onstat -p* 命令: 显示概要文件计数 在第550页
onstat -u	打印用户线程的状态和统计信息。如果线程正在等待资源， 那么该命令会识别资源的类型（flags字段）和地址（wait字段）。 *onstat -u* 命令: 显示用户活动的概要文件 在第562页

onstat 实用程序性能检查（第二层）

使用以下 onstat 选项识别性能障碍。

表 152: onstat 实用程序性能检查（第二层）

命令	参考
onstat -b	打印活动的缓冲区。. *onstat -b* 命令: 显示正在使用的缓冲区的信息 在第405页
onstat -g act	打印活动的线程。 *onstat -g act* 命令: 显示活动线程 在第421页
onstat -g glo	打印虚拟处理器及其操作系统进程（oninit 进程）。打印虚拟处理器 CPU 的使用。在 Windows™ 上，虚拟处理器是操作系统线程，pid 字段的值是线程 ID。

命令	参考
	onstat -g glo 命令: 显示全局多线程信息 在第461页
onstat -g mgm	打印内存分配管理器资源信息。 *onstat -g mgm* 命令: 显示 *MGM* 资源信息 在第480页
onstat -g rah	打印预读请求信息。 *onstat -g rah* 命令: 显示预读请求统计信息 在第499页
onstat -g rea	打印就绪队列中正在等待 CPU 资源的线程。 *onstat -g rea* 命令: 显示准备就绪的线程 在第501页
onstat -g seg	打印共享内存段统计信息。此选项显示已分配给数据库服务器的共享内存段的数量与大小。 *onstat -g seg* 命令: 显示共享内存段的统计信息 在第512页.
onstat -g wai	打印正在等待的线程；所有正在等待互斥或条件或正在让步的线程。 *onstat -g wai* 命令: 打印等待队列线程列表 在第538页
onstat -k	打印活动的锁。 *onstat -k* 命令: 打印活动锁的信息 在第544页

onstat 实用程序表选项

使用以下 onstat 选项显示有关表状态及其统计信息。

表 153: onstat 实用程序表选项

命令	参考
onstat -g buf	打印缓冲池概要文件信息。 *onstat -g buf* 命令: 显示缓冲池的概要文件信息 在第429页
onstat -g lap	打印有关当前活动的轻量附加（绕过缓冲池的写入）的状态信息。 *onstat -g lap* 命令: 显示轻量追加状态信息 在第474页
onstat -g opn	打印打开的分区（表）。 *onstat -g opn* 命令: 打印打开的分区 在第489页
onstat -g ppf	打印指定分区号的分区概要文件（活动数据）或打印所有分区的概要文件。 *onstat -g ppf* 命令: 打印分区概要文件 在第491页
onstat -g scn	基于在压缩表、具有比页大的行的表，和具有 VARCHAR、LVARCHAR 和 NVARCHAR 数据的表上的行扫描，打印有关扫描进度的信息，并标识扫描为轻扫描或缓冲池扫描。 *onstat -g scn* 命令: 显示扫描信息 在第506页
onstat -P	打印缓冲池中按分区（表）列出的表和 B 型树页。

命令	参考
	onstat -P 命令: 打印分区信息 在第554页
onstat -t onstat -T	打印活动的（t）或所有的（T）表空间的基础表空间（分区）信息。 *onstat -t and onstat -T* 命令: 打印表空间信息 在第560页

onstat 实用程序线程选项

使用以下 onstat 选项显示线程的状态和活动。

表 154: onstat 实用程序线程选项

命令	参考
onstat -g act	打印活动的线程。此输出包括在 onstat -g ath 输出中。 *onstat -g act* 命令: 显示活动线程 在第421页
onstat -g ath	打印所有线程。 *onstat -g ath* 命令: 显示所有线程的信息 在第426页。有关使用 onstat -g ath 打印 Enterprise Replication 线程的信息，请参阅《*SinoDB® Enterprise Replication* 指南》。
onstat -g bth	显示阻塞的线程和等待的线程之间的依赖关系。 *onstat -g bth* 和 *-g BTH*: 显示被阻挡和等待中的线程 在第427页
onstat -g BTH	显示阻塞线程的会话和堆栈信息。 *onstat -g bth* 和 *-g BTH*: 显示被阻挡和等待中的线程 在第427页
onstat -g cpu	打印每个线程的运行时统计信息。 *onstat -g cpu*: 显示运行时统计信息 在第447页
onstat -g rea	打印就绪线程（正在等待 CPU 资源的线程）。此输出包括在 onstat -g ath 输出中。 *onstat -g rea* 命令: 显示准备就绪的线程 在第501页.
onstat -g sle	打印已休眠一段指定时间的线程的信息。不包括永远休眠的线程。 *onstat -g sle* 命令: 打印所有休眠的线程 在第522页
onstat -g stk	打印指定线程的堆栈或打印锁有线程的堆栈。 *onstat -g stk* 命令: 打印线程堆栈 在第531页
onstat -g sts	打印每个线程最大和当前堆栈的使用。 *onstat -g sts* 命令: 打印每个线程的堆栈使用 在第533页
onstat -g tpf	打印线程活动统计信息。 *onstat -g tpf* 命令: 打印线程概要文件 在第534页
onstat -g wai	打印等待（空闲、休眠和等待）线程。包括在 onstat -g ath 输出中。

命令	参考
	onstat -g wai 命令: 打印等待队列线程列表 在第538页
onstat -g wst	打印线程的等待统计信息。 *onstat -g wst* 命令: 打印线程的等待统计信息 在第539页

onstat 实用程序用户/会话选项

使用以下 onstat 选项显示有关用户环境和活动的会话的信息。

表 155: onstat 实用程序用户/会话选项

命令	参考
onstat -g env	打印数据库服务器正在使用的环境变量的值。 *onstat -g env* 命令: 打印环境变量值 在第459页
onstat -g his	打印 SQL 跟踪信息。 *onstat -g his* 命令: 打印 *SQL* 跟踪信息 在第463页
onstat -g pqs	打印当前运行的 SQL 查询中使用的运算符。 *onstat -g pqs* 命令: 打印所有 *SQL* 查询的运算符 在第492页
onstat -g ses	打印所有活动的会话的摘要信息或单独会话的详细信息。 *onstat -g ses* 命令: 显示会话相关信息 在第513页
onstat -g spf	打印所有活动的会话的已就绪语句的概要文件。 *onstat -g spf*: 打印已就绪语句的概要文件 在第528页
onstat -g sql	打印所有活动的会话的 SQL 信息或单独会话的详细 SQL 信息。 *onstat -g sql* 命令: 打印与 *SQL* 有关的会话信息 在第526页
onstat -G	打印全局事务。 *onstat -G* 命令: 打印 *TP/XA* 事务信息 在第541页
onstat -u	打印用户线程的状态及其全局读/写统计信息。 *onstat -u* 命令: 显示用户活动的概要文件 在第562页
onstat -x	打印有关事务的信息。 *onstat -x* 命令: 显示数据库服务器事务信息 在第564页

onstat 实用程序虚拟处理器选项

使用以下 onstat 选项显示虚拟处理器的信息和统计信息。

表 156: onstat 实用程序虚拟处理器选项

命令	参考
onstat -g glo	打印全局多线程信息和虚拟处理器类和单独虚拟处理器的全局统计信息。在 Windows™ 上，虚拟处理器是操作系统线程，pid 字段的值是线程 ID。 *onstat -g glo* 命令: 显示全局多线程信息 在第461页
onstat -g sch	打印每个虚拟处理器的信号量操作、自旋和忙等待的数量。在 Windows™ 上，虚拟处理器是操作系统线程，且 pid 字段的值是线程 ID。 *onstat -g sch* 命令: 打印 *VP* 信息 在第506页

onstat 实用程序等待选项

使用以下 onstat 选项显示有关线程的等待条件的信息。

表 157: onstat 实用程序等待选项

命令	参考
onstat -g con	打印等待条件的线程的 ID。 onstat -g ath 打印线程信息。请参阅 *onstat -g con* 命令: 打印条件和线程信息 在第446页。
onstat -g lmx	打印所有已锁定的互斥。 *onstat -g lmx* 命令: 打印所有锁定的互斥 在第477页
onstat -g qst	打印互斥队列和条件队列的队列等待统计信息。 *onstat -g qst* 命令: 打印等待队列和条件队列的等待选项 在第498页
onstat -g rwm	打印读/写互斥。 *onstat -g rwm* 命令: 打印读取和写入互斥 在第505页
onstat -g spi	打印使用长自旋的自旋锁及其统计信息。 *onstat -g spi* 命令: 显示使用长自旋的自旋锁 在第525页
onstat -g wai	打印等待线程；所有正在等待互斥或条件或正在让步的线程。 *onstat -g wai* 命令: 打印等待队列线程列表 在第538页
onstat -g wmx	打印具有等待者的所有互斥。 *onstat -g wmx* 命令: 打印具有等待者的所有互斥 在第539页

其他有用的 onstat 实用程序选项

表 158: 其他有用的 onstat 实用程序选项

命令	参考
onstat -	打印 onstat 头；包括引擎版本、状态（联机、静默等等）、初始化后经过的时间，以及内存占用。

命令	参考
	onstat - 命令: 打印输出头 在第403页
onstat -	打印 onstat 用法选项。 *onstat --* 命令: 打印 *onstat* 选项和功能 在第404页
onstat *options infile*	使用共享内存转储（infile）作为输入打印 onstat 输出。 在共享内存转储文件上运行 *onstat* 命令 在第404页
onstat -a	打印 onstat 集体输出。 *onstat -a* 命令: 打印数据库服务器的整体状态 在第405页
onstat -c	打印服务器配置文件。 *onstat -c* 命令: 显示 *ONCONFIG* 文件内容 在第408页
onstat -C	打印 B 型树索引扫描程序信息（显示有关索引清除的统计信息）。 *onstat -C* 命令: 打印 *B* 型树扫描程序信息 在第408页
onstat -d	打印块信息。 *onstat -d* 命令: 显示块信息 在第414页
onstat -f	打印为 dataskip 配置的数据库空间。 *onstat -f* 命令: 打印受 *dataskip* 影响的数据库空间信息 在第419页
onstat -g all	打印诊断信息。 *onstat -g all* 命令: 打印诊断信息 在第422页
onstat -g cfg	打印配置参数当前值列表。 *onstat -g cfg* 命令: 显示配置参数的当前值 在第438页
onstat -g dbc	打印有关 dbScheduler 和 dbWorker 线程的统计信息。 *onstat -g dbc* 命令: 打印 *dbScheduler* 和 *dbWorker* 线程统计信息 在第448页
onstat -g dis	打印数据库服务器列表，及其状态、目录位置、配置信息和主机名。 *onstat -g dis* 命令: 打印数据库服务器信息 在第451页
onstat -g dll	打印加载的动态库列表。 *onstat -g dll* 命令: 打印动态链接库文件列表 在第452页
onstat -g osi	打印有关操作系统资源和参数的信息。 *onstat -g osi:* 打印操作系统信息 在第489页
onstat -g pos	打印 $INFORMIXDIR/etc/.infos.servernum 文件中的值，这些值由客户端（如 onmode）用于与服务器建立共享内存连接。 onmode -R 重新构建 $INFORMIXDIR/etc/.infos.servernum 文件。 *onstat -g pos* 命令: 打印文件中的值 在第489页
onstat -g smb	打印有关智能大对象空间的详细信息。

命令	参考
	onstat -g smb 命令: 显示智能大对象空间信息 在第523页
onstat -g sym	打印 oninit 实用程序的符号表信息。 *onstat -g sym* 命令: 打印 *oninit* 实用程序的符号表信息 在第533页
onstat -i	将 onstat 模式更改为交互。 *onstat -i* 命令: 启动 交互模式 在第544页
onstat -m	打印消息日志 内容。 *onstat -m* 命令: 打印最近的系统消息日志信息 在第550页
onstat -r	打印重复的 onstat 执行。 *onstat -r* 命令: 重复打印选择的统计信息 在第555页
onstat -z	将积累统计信息重新设置为零。 *onstat -z* 命令: 清除统计信息 在第569页

监视数据库服务器状态

要监视数据库服务器状态，请查看 onstat 命令的标题。

每当数据库服务器阻塞时，onstat 在标题行后面显示以下行：

```
Blocked: reason
```

变量 *reason* 可以是以下一个或多个值。

原因	描述
ADMINISTRATION	数据库处于管理模式
ARCHIVE	正在备份存储空间
ARCHIVE_EBR	由于外部备份和复原阻塞
CHG_PLOG	正在变更物理日志时阻塞
CKPT	检查点
CKPT INP	正在进行间隔检查点
DBS_DROP	正在删除数据库空间
DDR	离散数据复制
DYNAMIC_LOG	正在动态地添加日志文件
DYNAMIC_LOG_FOR_ER	正在 ER 设置中动态添加日志文件
FREE_LOG	正在释放日志文件
HA_CONV_STD	高可用性服务器正在转换为标准服务器时阻塞
HA_FAILOVER	正在处理高可用性服务器故障转移时阻塞
HANG_SYSTEM	数据库服务器故障

原因	描述
LAST_LOG_RESERVED4BACKUP	等待备份最后可用的日志
LBU	日志已满高水印
LOG_DROP	正在删除日志文件
LONGTX	长事务
MEDIA_FAILURE	介质故障
OVERRIDE_DOWN_SPACE	由于 ONDBSPACEDOWN 配置参数设置为 WAIT，因此在等待重写数据库空间设置

此表中，值 CHKP INP 并不表示数据库服务器被阻塞，而是在清空缓冲池时，正在进行非阻塞间隔检查点。此 CHKP INP 值出现在 onstat 输出的状态行中，直到共享内存缓冲池中所有页都写入磁盘。有关设置间隔检查点以清空缓冲池的信息，请参阅 *CKPTINTVL* 配置参数 在第83页。

onstat 命令语法

完整的 onstat 命令语法包括交互模式和如何重复执行选项的信息。

```
onstat
[ < -FILE 选项 > ]
-pu
{ [{ | -a | -b | -B | -c | -C | -d | -D | -f | -F | -g monitoring_option
| -G | -h | -i | -k | -l | -m | -o [{ nobuffs | full }] { outfile } | -O | -p
| -P | -r seconds | -R | -s | -t | -T | -u | -x | -X | -z }] [ infile ] | -
| -- | -V | -version }
```

元素	用途	关键注意事项
-	只显示输出头	请参阅 *onstat -* 命令: 打印输出头 在第403页.
--	显示所有 onstat 选项及其功能的列表	请参阅 *onstat --* 命令: 打印 *onstat* 选项和功能 在第404页。 此选项不可与任何其他 onstat 选项组合使用。
-a	解释为 onstat -cuskbtdlp。以该顺序显示输出。	请参阅 *onstat -a* 命令: 打印数据库服务器的整体状态 在第405页。
-b	显示当前使用中的缓冲区的信息，其包括缓冲池中常驻页的数量。	请参阅 *onstat -b* 命令: 显示正在使用的缓冲区的信息 在第405页。
-B	获取有关所有数据库服务器缓冲区（不仅仅是当前使用中的缓冲区）的信息。	请参阅 *onstat -B* 命令: 显示已用缓冲区的信息 在第406页。
-c	显示 ONCONFIG 文件： • $INFORMIXDIR/etc/$ONCONFIG（对于 UNIX™） • %INFORMIXDIR%\etc\ %ONCONFIG %（对于 Windows™）	请参阅 *onstat -c* 命令: 显示 *ONCONFIG* 文件内容 在第408页。
-C	打印 B 型树扫描程序信息	请参阅 *onstat -C* 命令: 打印 *B* 型树扫描程序信息 在第408页。
-d	显示每个存储空间中的块的信息	请参阅 *onstat -d* 命令: 显示块信息 在第414页。

元素	用途	关键注意事项
-D	显示每个数据库空间中前 50 个块的页读取和页写入信息	请参阅 *onstat -D 命令: 打印页读取和页写入的信息* 在第419页。
-f	列出当前受 DATASKIP 功能影响的数据库空间	请参阅 *onstat -f 命令: 打印受 dataskip 影响的数据库空间信息* 在第419页。
-F	显示将页青空到磁盘的每种类型的写入的计数	请参阅 *onstat -F 命令: 显示计数* 在第420页。
-g ***option***	打印监视选项	请参阅 *onstat -g 监视选项* 在第421页。
-G	打印全局事务 ID	请参阅 *onstat -G 命令: 打印 TP/XA 事务信息* 在第541页。
-h	提供缓冲区头散列链的信息	请参阅 *onstat -h 命令: 打印缓冲头哈希链信息* 在第543页。
-i	使 onstat 实用程序成为交互魔式	请参阅 *onstat -i 命令: 启动 交互模式* 在第544页。
-k	显示有关活动锁的信息	请参阅 *onstat -k 命令: 打印活动锁的信息* 在第544页。
-l	显示有关物理日志和逻辑日志（包括页地址）的信息	请参阅 *onstat -l 命令: 显示物理和逻辑日志信息* 在第546页。
-m	显示数据库服务器消息日志的最新的 20 行	此选项的输出列出消息日志文件的完整路径名和 20 个文件条目。日期时间头分隔每天的条目。每天的单个条目前面都有一个时间戳。消息日志的名称以 ONCONFIG 文件中的 MSGPATH 指定。 请参阅 *onstat -m 命令: 打印最近的系统消息日志信息* 在第550页。
-o	将共享内存段的副本保存到 *outfile* 中	请参阅 *onstat -o 命令: 共享内存内容输出到文件* 在第550页。
-p	显示概要文件计数。	请参阅 *onstat -p 命令: 显示概要文件计数* 在第550页。
-P	显示所有分区的分区号和属于该分区的缓冲池页的拆离	请参阅 *onstat -P 命令: 打印分区信息* 在第554页。
-pu	如果调用没有任何选项的 onstat，那么该命令将被解释为 onstat -pu（-p 选项和 -u 选项）。显示概要文件计数并打印用户活动的概要文件	请参阅 *onstat -p 命令: 显示概要文件计数* 在第550页和 *onstat -u 命令: 显示用户活动的概要文件* 在第562页。
-r ***seconds***	在每次执行之间等待指定的 *seconds* 秒之后重复伴随的 onstat 选项	请参阅 *onstat -r 命令: 重复打印选择的统计信息* 在第555页。
-R	显示关于 LRU 队列、FLRU 队列和 MLRU 队列的详细信息	请参阅 *onstat -R 命令: 显示 LRU、FLRU 和 MLRU 队列信息* 在第557页。
-s	显示一般锁存器信息	请参阅 *onstat -s 命令: 打印锁存器的信息* 在第559页。
-t	显示活动表空间的表空间信息（包括驻留状态）	请参阅 *onstat -t and onstat -T 命令: 打印表空间信息* 在第560页。
-T	显示所有表空间的表空间信息	请参阅 *onstat -t and onstat -T 命令: 打印表空间信息* 在第560页。

元素	用途	关键注意事项
-u	打印用户活动的概要文件	请参阅 *onstat -u* 命令: 显示用户活动的概要文件 在第562页。
-V	显示软件版本号和序列号。此选项不可与任何其他 onstat 选项组合使用。	请参阅 获取实用程序版本信息 在第289页。
-version	显示构建版本、主机、操作系统、编号和日期，以及 GLS 版本。此选项不可与任何其他 onstat 选项组合使用。	请参阅 获取实用程序版本信息 在第289页。
-x	显示关于事务的信息	请参阅 *onstat -x* 命令: 显示数据库服务器事务信息 在第564页。
-X	获取关于正在共享和等待缓冲区的线程的确谢信息	请参阅 *onstat -X* 命令: 显示线程信息 在第567页。
-z	将概要文件计数设置为 0	请参阅 *onstat -z* 命令: 清除统计信息 在第569页。
infile	指定 onstat 命令的源文件	此文件必须包括先前存储的使用 onstat -o 命令所创建的共享内存段。 有关如何使用 onstat -o 创建 *infile* 的说明，请参阅 *onstat -o* 命令: 共享内存内容输出到文件 在第550页。 有关在源文件上运行 onstat 的信息，请参阅 在共享内存转储文件上运行 *onstat* 命令 在第404页。

交互式执行

要将 onstat 实用程序进入交互模式，请使用 -i 选项。交互模式允许您输入多个选项（一个接一个）而不用退出程序。有关使用交互模式的信息，请参阅 *onstat -i* 命令: 启动 交互模式 在第544页。

连续 onstat 命令执行

onstat -r 与其他 onstat 选项组合使用，将致使其他选项在指定时间间隔重复执行。有关信息，请参阅 *onstat -r* 命令: 重复打印选择的统计信息 在第555页。

onstat 命令: 等同于 onstat -pu 命令

如果调用没有任何选项的 onstat，该命令解释为 onstat -pu（-p 选项和 -u 选项）。
语法:
onstat

onstat - 命令: 打印输出头

所有 onstat 输出都包括一个头。onstat - 命令仅显示输出头和此命令的返回值，该返回值指示数据库服务器模式。
语法:
onstat -

头具有以下格式:

```
Version--Mode (Type)--(Checkpnt)--Up Uptime--Sh_mem Kbytes
```

Version

是产品名和版本号

Mode

是当前的运行模式

(Type)

如果数据库服务器使用高可用性数据复制，那么指示类型是主还是辅助。

如果数据库服务器未涉及数据复制，那么此字段不出现。如果类型是主，那么显示值 P。如果类型是辅助，那么显示值 S。

(Checkpnt)

是检查点标志

如果设置，那么在适当时间点，头可能在模式后面显示两个其他字段：

(CKPT REQ)

指示用户线程请求检查点

(CKPT INP)

指示检查点正在处理中。在检查点过程中，访问权将限制为只读。值到检查点结束前，数据库服务器不可写入或更新数据。

Uptime

指示数据库服务器已运行了多长时间

如果手动将系统时间更改为过去时间且服务器的启动时间是晚于当前系统时间，那么就此字段不可用。在这个情况下，头将显示 Uptime Unavailable。

Sh_mem

是数据库服务器共享内存的大小，以 KB 表示

以下为数据库服务器头的样本：

```
Sinoregal DSV12.10.UC1--On-Line--Up 15:11:41--9216 Kbytes
```

如果数据库服务器阻塞，那么 onstat 头的输出包括额外的行。有关该行的状态码的信息，请参阅监视数据库服务器状态 在第400页。

返回码

当退出 onstat 实用程序时，会显示几个有用的码。请参阅退出 *onstat* 实用程序时的返回码 在第570页。

onstat -- 命令：打印 onstat 选项和功能

使用 onstat -- 命令显示所有 onstat -- 选项及其功能的列表。不可将此选项与任何其他标志组合。
语法：
onstat --

在共享内存转储文件上运行 onstat 命令

可以对共享内存转储文件运行 onstat 命令。可以通过使用 onstat -o 命令显式生成共享内存转储文件。如果 DUMPSHMEM 配置参数设置为 1 或 2，那么在断言失败时自动创建转储文件。
语法：
onstat *optionsinfile*

当使用命令行时，最终参数输入源文件。以下示例打印包含在名为 onstat.out 文件中的共享内存转储的所有线程，而不是尝试连接到正在运行的服务器的共享内存。

```
onstat -g ath onstat.out
```

有关如何使用 onstat -o 创建内存转储文件的说明，请参阅 *onstat -o* 命令: 共享内存内容输出到文件 在第550页。

在共享内存转储文件上交互地运行 onstat 命令

使用 onstat -i（交互模式） 对转储文件运行多个 onstat 命令。由于交互方式只读取一次文件，因此可以节省时间。

以下示例读取共享内存转储文件并进入交互模式。可以在正常的交互模式中对转储文件执行其他 onstat 命令。

```
onstat -i source_file
```

有关交互模式的信息，请参阅 *onstat -i* 命令: 启动 交互模式 在第544页。

在以无缓冲池创建的共享内存转储文件上运行 onstat 命令

在以无缓冲池创建的转储文件（使用 onstat -o nobuffs 创建或将 DUMPSHMEM 配置参数设置为 2 创建）上运行某些 onstat 命令时，它们会有不同的输出：

- 如果在以无缓冲池创建的转储文件上运行 onstat -B，那麽输出将会在 memaddr、nslots 和 pgflgs 列显示 0。
- 如果在以无缓冲池创建的转储文件上运行 onstat -g seg，那么输出将会显示原始驻留段大小和无缓冲的驻留段大小。
- 如果在无缓冲池的共享内存转储文件上运行 onstat -P，那么输出将会是：

```
Nobuffs dumpfile -- this information is not available
```

相关链接

DUMPSHMEM 配置参数 *(UNIX)* 在第109页

onstat -g seg 命令: 显示共享内存段的统计信息 在第512页

onstat -a 命令: 打印数据库服务器的整体状态

使用 onstat -a 命令显示有关数据库服务器状态的信息。此命令不显示所有 onstat 选项的信息，仅显示用于初始故障排除的那些 onstat 选项的信息。

语法:

onstat -a

onstat -b 命令: 显示正在使用的缓冲区的信息

使用 onstat -b 选项显示当前正在使用中的缓冲区的信息，包括缓冲池中常驻页的总数。

语法:

onstat -b

可用的最大缓冲区数以 ONCONFIG 文件中 BUFFERPOOL 配置参数的 buffers 字段进行指定。

onstat -b 命令还提供有关已修改缓冲区数、缓冲池中常驻页的总数、可用缓冲区的总数、可用散列存储区数，和以字节表示的缓冲区大小（页大小）的摘要信息。

```
123 modified, 23 resident, 2000 total, 2048 hash buckets, 2048 buffer size.
```

有关显示所有缓冲区信息的信息，请参阅 *onstat -B* 命令: 显示已用缓冲区的信息 在第406页。

示例输出

以下是 onstat -b 命令的样本输出。有关输出的说明，请参阅 *onstat -B* 命令: 显示已用缓冲区的信息 在第406页。

```
Buffer pool page size: 4096

address         userthread flgs pagenum memaddr         nslots pgflgs xflgs owner waitlist
70000001097e9e8 0          c07 1:47841 7000000118e0000 10     1      0     0     0
700000010982188 0          807 1:47827 700000011939000 225    90     10    0     0
2011 modified, 50000 total, 65536 hash buckets, 4096 buffer size
```

图 98: onstat -b 命令输出

onstat -B 命令: 显示已用缓冲区的信息

使用 onstat -B 选项显示不在可用列表上的缓冲区的信息。
语法:
onstat -B

onstat -B 和 onstat -b 显示类似的信息，除了 onstat -b 命令仅显示当前正被用户线程访问的缓冲区外。onstat -B 命令显示不在可用列表上的所有缓冲区的信息。

有关在以无缓冲池创建的转储文件上运行 onstat -B 命令的信息，请参阅在共享内存转储文件上运行 *onstat* 命令 在第404页。

示例输出

输出描述

Buffer pool page size

以字节表示的缓冲池页的大小

address

缓冲区表中缓冲区头的地址

userthread

访问缓冲区表的最新用户线程的地址。许多用户线程可能正在并发读取相同的缓冲区。

flgs

使用以下标志位描述缓冲区:

0x01

已修改数据

0x02

数据

0x04

LRU

0x08

错误

pagenum

磁盘上的物理页号

memaddr

缓冲区内存地址

nslots

页中槽表条目的数量

此字段指示存储在页上的行（行的一部份）的数量。

pgflgs

使用以下值（单独或组合）来描述页类型：

1
: 数据页

2
: 表空间页

4
: 可用列表页

8
: 块可用列表页

9
: 剩余数据页

b
: 分区常驻 BLOB 页

c
: BLOB 空间常驻 BLOB 页

d
: Blob 块可用列表位页

e
: Blob 块 Blob 图页

10
: B 型树节点页

20
: B 型树根节点页

40
: B 型树分支节点页

80
: B 型树叶节点页

100
: 逻辑日志页

200
: 逻辑日志最后一页

400
: 逻辑日志的同步页

800
: 物理日志

1000
: 保留根页

2000

无需物理日志

8000

具有默认标志的 B 型树

xflgs

使用以下标志位描述缓冲区访问：

0x10

共享锁

0x80

互斥锁

owner

设置 xflgs 缓冲区标志的用户线程

waitlist

等待访问该缓冲区的第一个用户线程的地址

有关等待缓冲区的所有线程的完整列表，请参阅 *onstat -X* 命令: 显示线程信息 在第567页。

onstat -c 命令: 显示 ONCONFIG 文件内容

使用 onstat -c 命令显示 ONCONFIG 文件的内容。

语法：

```
onstat   -c
```

数据库服务器首先检查您是否已经指定了值给环境变量 ONCONFIG。可以在数据库服务器处于任何模式（包括脱机）时使用 onstat -c 选项。

仅 UNIX:

在 UNIX™ 上，如果设置了 ONCONFIG，那么 onstat -c 将显示 $INFORMIXDIR/etc/$ONCONFIG 文件的内容。如果未设置，缺省情况下，onstat -c 显示 $INFORMIXDIR/etc/onconfig 的内容。

仅 Windows:

在 Windows™ 上，如果设置了 ONCONFIG，那么 onstat -c 显示 %INFORMIXDIR%\etc\%ONCONFIG% 文件的内容。如果未设置，缺省情况下，onstat -c 显示 %INFORMIXDIR%\etc\onconfig 的内容。

相关链接

显示 *onconfig* 文件中的设置 在第43页

onstat -g cfg 命令: 显示配置参数的当前值 在第438页

数据库服务器文件 在第715页

onstat -C 命令: 打印 B 型树扫描程序信息

使用 -C 命令显示关于 B 型树扫描程序子系统和每个 B 型树扫描程序线程的信息。

语法：

```
onstat   -C { prof  | hot  | part  | clean  | range  | map  | alice  | all }
```

以下选项可用于 onstat -C 命令，并可组合：

prof

打印系统和每个 B 型树扫描程序线程的概要文件信息。此为缺省选项。

hot

以要清除的顺序打印热列表索引键

part

打印具有索引统计信息的所有分区

clean

打印已清除或要清初的所有分区的信息

range

打印通过使用索引范围扫描处理所节约的页

map

显示通过 alice 清除方法正在清除的每个索引的当前位图

alice

显示 alice 清除方法选项的效率

all

打印所有 onstat -C 选项

使用 prof 选项的示例输出

```
Btree Cleaner Info
BT scanner profile Information
==============================
Active Threads                              1
Global Commands                       2000000   Building hot list
Number of partition scans               11003
Main Block                        0xc000000003c9dc68
BTC Admin                         0xc0000000024bc208


BTS info             id   Prio    Partnum      Key     Cmd
0xc000000003c9dee8    0   High   0x00000000     0       40  Yield N
    Number of leaves pages scanned                          77
    Number of leaves with deleted items                      6
    Time spent cleaning (sec)                                0
    Number of index compresses                               0
    Number of deleted items                                113
    Number of index range scans                              0
    Number of index leaf scans                               0
    Number of index alice scans                              2
```

图 102: 使用 prof 选项的 onstat -C 命令输出

使用 prof 选项的输出描述

Id

BTSCANNER ID

Prio

BTSCANNER 的当前优先级

Partnum

该线程当前正在处理的索引的分区号

Cmd

该线程当前正在处理的命令

使用 hot 选项的示例输出

```
Btree Cleaner Info

Index Hot List
==============
    Current Item           5     List Created          15:29:47
    List Size              4     List expires in              0 sec
    Hit Threshold        500     Range Scan Threshold        -1


Partnum        Key            Hits
0x00100191       1              14 *
0x00A00022       1              13 *
0x00100191       2               8 *
0x00100150       2               7 *
```

图 103: 使用 hot 选项的 onstat -C 命令输出

使用 hot 选项的输出描述

Partnum

索引的分区号

Key

索引键

Hits

Hit 计数器的当前值

*

指示在该热列表持续期间已清除此分区

使用 part 选项的示例输出

```
Btree Cleaner Info


Index Statistics
================
 Partnum  Key     Positions   Compress      Split
0x00100002  1           146          0          0
0x00100004  1             4          0          0
0x00100004  2            13          0          0
0x00100005  1             1          0          0
0x00100005  2             0          0          0
0x00100006  1             1          0          0
0x00100006  2             0          0          0
0x00100007  2             1          0          0
0x00100008  2             1          0          0
0x0010000a  1             0          0          0
0x0010000e  3             1          0          0
0x00100011  1             1          0          0
0x00100013  2             2          0          0
```

图 104: 使用 part 选项的 onstat -C 命令输出

使用 part 选项的输出描述

Partnum

索引的分区号

Key

索引键

Positions

读取索引的次数

Compress

已压缩的页数

Split

已发生的分割数

C

指示分区正在被清除

N

索引分区不再符合清除资格

使用 clean 选项的示例输出

```
Btree Cleaner Info

Index Cleaned Statistics
========================
 Partnum  Key      Dirty Hits  Clean Time  Pg Examined  Items Del  Pages/Sec
0x00100013  2              2           0            0          0       0.00
0x0010008b  3              1           0            0          0       0.00
0x001000c7  1              2           0            0          0       0.00
0x00100150  2              7           0            0          0       0.00
0x0010016f  2              2           0            0          0       0.00
0x00100191  1             14           0            0          0       0.00
0x00100191  2              8           0            0          0       0.00
0x00a00011  2              6           0            0          0       0.00
0x00a00013  1              0           0           24          0      24.00
0x00a00019  1              0           0          470        225     470.00
0x00a00022  1             13           0            0          0       0.00
0x00a00022  2              5           0            0          0       0.00
```

图 105: 使用 clean 选项的 onstat -C 命令输出

使用 clean 选项的输出描述

Partnum

索引的分区号

Key

索引键

Dirty Hits

已扫描脏页的次数

Clean Time

总耗时，以秒为单位

Pg Examined

btscanner 线程检查的页数

Items Del

从该索引除去的项目数

Pages/Sec

每秒检查的页数

C

指示分区正在被清除

N

索引分区不再符合清除资格

示例输出

```
Btree Cleaner Info

Cleaning Range Statistics
=========================
 Partnum  Key             Low          High          Size        Saving
0x001001bc  2               36            69            96      65.6 %
0x001001be  1               16            20            48      91.7 %
0x001001cd  1                8            21            32      59.4 %
0x001001cd  2               24            25            32      96.9 %
```

图 106: onstat -C range

输出描述

Partnum

分区号

Key

索引键

Low

范围扫描的下界

High

索引扫描的上界

Size

以页表示的索引大小

Saving

相对完整扫描节约的时间百分比

C

指示分区正在被清除

N

索引分区不再符合清除资格

示例输出

```
Btree Cleaner Info

ALICE Bitmap of Deleted Index Items
===================================
Partnum   Key        Map
0x00100013  2 0000: 80000000 00000000
0x0010008b  3 0000: 80000000 00000000
```

```
0x001000c7  1 0000: 80000000 00000000
0x00100150  2 0000: 80000000 00000000
0x0010016f  2 0000: 80000000 00000000
0x00100191  1 0000: 80000000 00000000
0x00100191  2 0000: 80000000 00000000
0x00a00011  2 0000: 80000000 00000000
0x00a00013  1 0000: 00000000 00000000
0x00a00019  1 0000: 00000000 00000000
0x00a00022  1 0000: 80000000 00000000
0x00a00022  2 0000: 80000000 00000000
```

图 107: onstat -C map

输出描述

Partnum

分区号

Key

索引键

Map

Alice 位图

示例输出

```

Btree Cleaner Info

ALICE Cleaning Statistics
=========================

System ALICE Info: Mode =    6, Eff =    30 %, Adj =     5

Partnum    Mode BM_Sz  Used_Pg  Examined  Dirty_Pg  # I/O  Found    Eff   Adj
0x00100013    6    64       97         0         0      0      0   0.0 %    0
0x0010008b    6    64        5         0         0      0      0   0.0 %    0
0x001000c7    6    64        2         0         0      0      0   0.0 %    0
0x00100150    6    64       91         0         0      0      0   0.0 %    0
0x0010016f    6    64       91         0         0      0      0   0.0 %    0
0x00100191    6    64       26         0         0      0      0   0.0 %    0
0x00100191    6    64       26         0         0      0      0   0.0 %    0
0x001001bc    0     0       91         0         0      0      0   0.0 %    0
0x001001cd    0     0       26         0         0      0      0   0.0 %    0
0x001001cd    0     0       26         0         0      0      0   0.0 %    0
0x00a00011    6    64       91         0         0      0      0   0.0 %    0
0x00a00013    6    64       25        24         3      3      1  33.3 %    1
0x00a00019    6    64      470       470         3      3      2  66.7 %    1
0x00a00022    6    64       26         0         0      0      0   0.0 %    0
0x00a00022    6    64       26         0         0      0      0   0.0 %    0
```

图 108: onstat -C alice

输出描述

Partnum

索引的分区号

Mode

当前分区的 alice 方式

BM_Sz

已分配的位图大小

Used_Pg

索引大小（已使用页数）

Dirty_Pg

脏页数

 # I/O

读取的页数

Found

读取中发现的脏页数

Eff

位图的效率

Adj

对于该分区 alice 效率级别不足且被调整的次数

onstat -d 命令：显示块信息

使用 onstat -d 命令显示每个存储空间中的块的信息。
语法：
onstat -d [update]

update 选项更新共享内存以获取可用页的准确计数。

对智能大对象空间使用 onstat -d

有关使用 onstat -d 以确认智能大对象空间、用户数据区域和元数据区域的大小的信息，请参阅《*SinoDB*®管理员指南》中的“监视智能大对象空间”。

对 BLOB 空间使用 onstat -d

如果在具有 BLOB 空间块的服务器上运行 onstat -d 命令，那么数据库服务器将显示以下消息：

```
NOTE: For BLOB chunks, the number of free pages shown is out of date.
 Run 'onstat -d update' for current stats.
```

要获取 BLOB 空间块的当前统计信息，请运行 onstat -d update 命令。onstat 实用程序使用每个 BLOB 空间块的可用页的准确计数更新共享内存。数据库服务器显示以下消息：

```
Waiting for server to update BLOB chunk statistics ...
```

示例输出

```
Sinoregal DS Version 12.10.F       -- On-Line -- Up 00:01:27 -- 133540 Kbytes

Dbspaces
address   number  flags     fchunk  nchunks  pgsize  flags  owner    name
48750028  1       0x60001   1       1        2048    N  BA  informix rootdbs
4a0bee00  2       0x60001   2       1        2048    N  BA  informix dbspace2
 2 active, 2047 maximum
```

```
Chunks
address   chunk/dbs  offset  size    free   bpages flags pathname
487501c8  1      1   0       1000000 923615        PO-B-- /dev/raw/raw1
49f1bda0  2      2   0       5000    4972          PO-BED /work2/dbspaces/dbs2
 2 active, 32766 maximum

NOTE: The values in the "size" and "free" columns for DBspace chunks are
      displayed in terms of "pgsize" of the DBspace to which they belong.

Expanded chunk capacity mode: always
```

图 110: onstat -d 命令输出

对于数据库空间的输出描述

输出的第一部份描述存储空间：

address

是共享内存空间表中存储空间的地址

number

是创建存储空间时所指定的唯一 ID

flags

使用十六进制值描述每个存储空间。可以累积单个标志值以显示数据库空间的累积属性。下表描述每个十六进制值：

表 159: 每个十六进制值的描述

标志值	描述
0x0001	允许镜像且数据库空间未镜像。
0x0002	允许镜像且数据库空间已镜像。
0x0004	数据库空间包含禁用镜像的块。
0x0008	新镜像的
0x0010	BLOB 空间
0x0200	正在恢复空间。
0x0400	空间已物理恢复。
0x0800	正在恢复逻辑日志。
0x2000	临时数据库空间
0x4000	正在备份 BLOB 空间。
0x8000	智能大对象空间
0x10000	已更改物理日志或逻辑日志。
0x20000	已更改数据库空间或块表。
0x040000	BLOB 空间包含大块。
0x080000	此数据库空间中的块已重命名。
0x00100000	仅供共享磁盘辅助服务器使用的临时数据库空间。它是 SD 辅助服务器上 SDS_TEMPDBS 配置参数中所列的数据库空间之一。

标志值	描述
0x00200000	SD 辅助服务器的临时数据库空间。列于共享磁盘辅助服务器上的 DBSPACETEMP 配置参数中。
0x00400000	已在外部备份该数据库空间。
0x00800000	数据库空间正在进行碎片整理。
0x01000000	物理日志空间

fchunk

首块的 ID

nchunks

存储空间中的块数

pgsize

以字节为单位的数据库空间页的大小

flags

使用以下字母代码描述每个存储空间：

位置 1：

标志	描述
M	已镜像
N	未镜像

位置 2：

标志	描述
X	新镜像的
P	已物理恢复，等待逻辑恢复
L	逻辑恢复
R	恢复
D	关闭

位置 3：

标志	描述
B	BLOB 空间
P	物理日志空间
S	智能大对象空间
T	临时数据库空间
U	临时智能大对象空间
W	主服务器上的临时数据库空间（此标志仅在 SD 辅助服务器上显示。）

位置 4：

标志	描述
B	数据库空间具有大于 2 GB 的大块。

位置 5：

标志	描述
A	数据库空间是可自动扩展的，因为启用了 SP_AUTOEXPAND 配置参数，并数据库空间配置了不为零的创建大小或扩展大小。

owner

存储空间的所有者

name

存储空间的名称

在紧随存储空间列表的行中，active 为数据库服务器实例中存储空间（包括根数据库空间）的当前数量，maximum 为该数据库服务器实例总的允许空间。

输出描述 - 块

onstat -d 命令输出的第二部分描述块：

address

块的地址

chk/dbs

块编号和相关联的空间编号

offset

基页大小中文件或原设备的偏移量e

size

以块所属数据库空间的页大小为单位表示的块大小。

free

以块所属数据库空间的页大小为单位表示的块的可用数量。值 0 指示块中所有空间被分配给表，但未指示表中有多少可用空间。例如：假设创建了具有 200 MB 块的数据库空间，并创建了具有 200 MB 扩展数据块大小的表。那么 free 字段值为 0，表示该块无可用空间，然而，新的空表有 200 MB 可用空间。

对于 BLOB 空间，颚化符号指示可用 BLOB 页的大约数量。

对于智能大对象空间，指示用户数据空间和总用户数据空间的可用页数。

bpages

是 BLOB 页中块的大小

BLOB 页可大于磁盘页；因此，bpages 值可小于 size 值。

对于智能大对象空间，是智能大对象页中块的大小。

flags

提供如下的块状态信息：

位置 1：

标志	描述
P	主

标志	描述
M	镜像

位置 2：

标志	描述
N	已重命名，且是关机或不一致的
O	联机
D	关机
X	新镜像的
I	不一致的

位置 3：

标志	描述
-	数据库空间
B	Blob 空间
S	智能大对象空间

位置 4：

标志	描述
B	数据库空间具有大于 2 GB 的大块。

位置 5：

标志	描述
E	标示块是可扩展的
-	标示块是不可扩展的

位置 6：

标志	描述
-	此熟文件块未启用直接 I/O 或并发 I/O 选项
C	在 AIX® 上，为此熟文件块启用并发 I/O 选项
D	为此熟文件块启用直接 I/O 选项

pathname

物理设备的路径名

在紧随块列表的行中，active 显示活动块（包括根块）的数量，maximum 显示块的总数。

有关页读取和页写入的信息，请运行 onstat -D 命令。

相关链接

DBSPACETEMP 配置参数 在第91页

onstat -D 命令: 打印页读取和页写入的信息 在第419页

DIRECT_IO 配置参数 (UNIX) 在第96页
MIRROR 配置参数 在第132页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页

onstat -D 命令: 打印页读取和页写入的信息

使用 onstat -D 命令显示每个空间中首 50 块的页读取和页写入的信息。
语法:
onstat -D

示例输出

```
Dbspaces
address  number   flags      fchunk   nchunks pgsize  flags    owner    name
a40d7d8  1        0x1        1        1       2048    N        informix rootdbs
 1 active, 2047 maximum

Chunks
address  chunk/dbs  offset     page Rd  page Wr  pathname
a40d928  1      1   0          0        0        /work/11.1/dbspaces/stardbs3
 1 active, 2047 maximum

Expanded chunk capacity mode: disabled
```

图 112: onstat -D 命令输出

输出描述

onstat -D 输出与 onstat -d 输出几乎相同。以下列是 onstat -D 独有的。有关其他输出列的信息，请参阅 *onstat -d 命令: 显示块信息* 在第414页。

page Rd
是页读取的数量

page Wr
是页写入的数量

相关链接

onstat -d 命令: 显示块信息 在第414页

onstat -f 命令: 打印受 dataskip 影响的数据库空间信息

使用 -f 命令列出当前受 dataskip 功能影响的数据库空间。
语法:
onstat -f

-f 选项列出设置了 DATASKIP 配置参数与 onspaces 的 -f 选项的数据库空间。当执行 onstat -f 时，数据库服务器显示以下三种输出之一:

- Dataskip 对所有数据库空间都是 OFF。
- Dataskip 对所有数据库空间都是 ON。
- Dataskip 对以下数据库空间是 ON:

```
dbspace1 dbspace2...
```

相关链接

DATASKIP 配置参数 在第87页

onstat -F 命令：显示计数

使用 onstat -F 命令显示将页清空到磁盘上的每种类型的写操作的计数。
语法：
onstat -F

示例输出

```
Fg Writes     LRU Writes    Chunk Writes
0             330           7631

address  flusher  state  data  # LRU  Chunk  Wakeups  Idle Time
c7c8850  0        I      0     9      29     16116    16093.557
      states: Exit Idle Chunk Lru
```

图 115: onstat -F 命令输出

输出描述

可以如下解释此选项的输出：

Fg Writes

是发生前台写入的次数

LRU ***Writes***

是发生 LRU 写入的次数

Chunk Writes

是发生块写入的次数

address

是指定给该页清除程序线程的用户结构的地址

flusher

是页清除程序号

state

使用以下代码指示当前页清除程序活动：

C

块写入

E

退出

I

清除程序处于空闲状态

L

LRU 队列

退出代码指示数据库服务器正在关机，或者页清除程序在特定时间量中还未从其写操作中返回。当操作未能在分配的时间内完成时，此状况称为超时状况。数据库服务器不知道清除程序发生了什么，因此标记为退出。无论是那种情况，清除程序最终退出。

data

提供与 state 字段相呼应的其他信息。

如果 state 是 C，那么 data 是页清除程序正在将缓冲区写入的块号。如果 state 是 L，那么 data 是页清除程序正从其写入的 LRU 队列。data 值显示为十进制，后跟等号，并以十六进制进行重复。

#LRU

对应于 onstat -g ath 线程 ID 输出

Chunk

已清除的块数

Wakeups

清空程序线程被唤醒的次数

Idle Time

清空程序线程的空闲时间（以秒计）

相关链接

CLEANERS 配置参数 在第84页

onstat -g 监视选项

onstat -g 命令使用的选项只提供用于支持与调试目的。onstat -g 命令只能包含这些选项之一。

相关链接

《*SinoDB Enterprise Replication* 指南》: 用于 *Enterprise Replication* 的 *onstat -g* 命令

onstat -g act 命令: 显示活动线程

使用 onstat -g act 命令显示有关活动线程的信息。

语法:

onstat -gact

以下是 onstat -g act 命令样本输出。有关输出的描述，请参阅 *onstat -g ath 命令: 显示所有线程的信息* 在第426页。

示例输出

```
Running threads:
tid  tcb        rstcb  prty  status    vp-class    name
2    b3132d8    0      1     running   *2adm       adminthd
40   c5384d0    0      1     running   *1cpu       tlitcppoll
```

图 117: onstat -g act 命令输出

onstat -g afr 命令: 打印已分配内存分段

使用 onstat -g afr 命令显示有关指定会话或共享内存池的已分配内存段的信息。向每个会话分配一个共享内存池。

语法:

onstat -gafr { *pool_name sessionid pool_address* }

此命令需要一个额外参数来指定池名称、会话 ID 或池地址。为每个会话分配一个与会话 ID 相同名称的内存池。

pool_name 是共享内存池的名称。可运行 onstat -g mem 命令以标识池的名称。

sessionid 是会话 ID。可运行 onstat -g ses 命令标识会话 ID。

pool_address 是共享内存池的地址。可运行 onstat -g mem 命令或 onstat -g ses 命令来标识池的地址。

示例输出

```
Allocations for pool name global:
addr             size       memid        fileid   location
4b231000         3288       overhead     306      mtshpool.c:617
4b231cd8         72         mcbmsg       1637     rldmsg.c:92
4b231d20         160        mcbmsg       1637     rldmsg.c:92
4b231dc0         64         osenv        2909     osenv.c:1164
4b231e00         64         osenv        2909     osenv.c:1971
4b231e40         64         osenv        2909     osenv.c:1164
4b231e80         64         osenv        2909     osenv.c:1971
```

图 119: onstat -g afr 命令输出

输出描述

addr（十六进制）
池分段的内存地址。

size（十进制）
以字节表示的池分段的大小。

memid（字串符）
池分段的内存 ID。

fileid（十进制）
仅限内部使用。分配的代码文件标识符。

location（字串符）
仅限内部使用。分配代码中的行号。

onstat -g all 命令: 打印诊断信息

如果星瑞格®支持建议收集诊断信息，请使用 onstat -g all 命令进行收集。为了能够正常管理，请以个别选项使用 onstat -g。

语法：

```
onstat   -gall
```

onstat -g aqt 命令: 打印数据市集和加速查询表的信息

使用 onstat -g aqt 命令显示有关数据市集以及相关联的加速查询表（AQT）的信息。

语法：

```
onstat   -gaqt [ aqt_name ]
```

示例输出

```
AQT Dictionary Cache for database school:

mart: school
accelerator: DWAFINAL
last load: 2011/07/29 07:00:39

AQT name                                  FactTab #tab #matched     address
------------------------------------------------------------------------------
aqt4d11b552-7d41-4b0c-824b-7714b6cb580a    103    1      328         0x4d187e08
aqt61498fab-3617-4c8c-ab40-fd8af4253998    103    2       42         0x4d84a448
aqtbc2da77c-bca8-4ce7-9191-8180a860da34    103    2      768         0x4d187f60
aqt88757e9d-81ee-43b4-87b2-0bf48c98fa55    103    3       15         0x4d84a190
aqta786d0dc-8e95-4de0-a1bd-773aa03a52db    103    3     1475        0x4d84a650
aqt8dd61c80-2c1c-4f0e-8f0c-91babe789f41    103    4      632         0x4d84a908

mart: school2
```

```
accelerator: DWAFINAL
last load: 2011/07/29 07:01:04

AQT name                                    FactTab #tab #matched     address
------------------------------------------------------------------------------
aqt56d5aea7-32f4-44e6-8d98-02a7af37630f     103     1      845        0x4d84ac70
aqt03ec4c20-7ba8-4c3a-ae56-4134b005269d     103     2       27        0x4d95c298
aqt4ae7c2fd-5b94-423d-bc49-9ca3f5f38799     103     2     3912        0x4d84adc8
aqt5ed69a75-15e3-45cc-9892-4f5386257895     103     3       83        0x4d95c4a0
aqtdf314aa6-177d-4443-9f6d-f14ba766995a     103     3       37        0x4d95c028
aqt7e36b1f2-4646-4075-ac0b-5fdee475cd7e     103     4      518        0x4d95c758

mart: school3
accelerator: DWAFINAL
last load: 2011/07/29 07:01:50

AQT name                                    FactTab #tab #matched     address
------------------------------------------------------------------------------
aqt92b36a8a-1567-4146-833c-385cd103f5d4     103     1      678        0x4d95cac0
aqt3189bec1-b6c9-417d-b969-92c687ef2e44     103     2       59        0x4d95cc18
aqt8d3b3dc8-59b6-4e34-822b-75b06b99c900     103     2     4487        0x4d90c0d8
aqt5f9c2a05-9131-4738-a929-036fcf77f65c     103     3       71        0x4d90c2e0
aqtee08ed16-6a5c-4478-ac57-fc4f99539c74     103     3      795        0x4d95ce20
aqt04d1c96a-022b-4ed7-938d-caf765bc9926     103     4      367        0x4d90c598

18 entries
```

图 122: onstat -g aqt 命令输出

如果可选参数 *aqt_name* 使用 AQT 名称，那么该命令打印有关特定 AQT 的信息。

```
AQT: aqt6de1afdd-f10a-45b0-93e9-0c208405fefd
database: iwadb
AQT tabid: 125
Fact table: 111
Number of times matched: 8947

Join structure: alias(tabid)[colno,...] = alias(tabid)[colno,...] {u:unique}
0(111)[1] = 1(110)[1] u
            1(110)[2] = 2(109)[1] u
                        2(109)[5] = 3(101)[1] u
                                    3(101)[3] = 4(100)[1] u
0(111)[2] = 5(106)[1] u
            5(106)[2] = 6(103)[1] u
            5(106)[3] = 7(104)[1] u
            5(106)[4] = 8(105)[1] u
                        8(105)[3] = 9(101)[1] u
                                    9(101)[3] = 10(100)[1] u
            5(106)[5] = 11(102)[1] u
0(111)[2,3] = 15(108)[1,2] u
              15(108)[1] = 16(106)[1] u
                           16(106)[2] = 17(103)[1] u
                           16(106)[3] = 18(104)[1] u
                           16(106)[4] = 19(105)[1] u
                                        19(105)[3] = 20(101)[1] u
                                                     20(101)[3] = 21(100)[1] u
                           16(106)[5] = 22(102)[1] u
              15(108)[2] = 23(107)[1] u
                           23(107)[2] = 24(101)[1] u
                                        24(101)[3] = 25(100)[1] u
0(111)[3] = 12(107)[1] u
            12(107)[2] = 13(101)[1] u
```

```
                    13(101)[3] = 14(100)[1] u
```

图 123: onstat -g aqt aqt_name 命令输出

输出描述

AQT 按其所属数据市集分组。这些组先按加速程序名称排序，再按数据市集名称排序。在数据市集组中，AQT 按以下顺序排序：实际表的 tabid (FactTab)、表的数量 (#tab) 和 AQT 名称。

来自字典高速缓存中条目的输出参考了数据市集的 AQT。输出仅在 AQT加载到字典高速缓存时显示，这通常在查询与 AQT 匹配时发生。

在服务器尝试将查询与 AQT 进行匹配之前，AQT 在字典高速缓存中没有任何条目。onstat -g aqt 命令不会在输出中显示任何条目。当在数据库服务器启动过程中初始化字典高速缓存时，#matched 和 address 列会得到新值。

onstat -g aqt 命令打印以下信息：

mart
  数据市集的名称

accelerator
  加速程序实例的名称

last load
  上一次加载数据市集的时间戳

AQT name
  系统生成的唯一 AQT 名称

FactTab
  AQT 实际表的 tabid

#tab
  为 AQT 一部份的表的数量

#matched
  已发生与 AQT 匹配的查询计数

address
  AQT 的内部数据库服务器内存地址T

onstat -g aqt ***aqt_name*** 命令打印以下信息：

AQT
  The unique system-generated name of the AQT

database
  AQT 所属的数据库的名称

AQT tabid
  数据库服务器的 systables 系统目录表中构成 AQT 条目的 tabid

Fact table
  AQT 的实际表的 tabid

Number of times matched
  已发生与 AQT 匹配的查询计数

AQT 信息后跟随着数据市集星型模式的文本表示。文本表示显示星型联结中表的列是如何互相关联的。

有关仓库加速器的信息，请参阅《*SinoDB*® 仓库加速器管理指南》。

onstat -g arc 命令: 打印归档状态

使用 onstat -g arc 命令显示有关每个数据库空间最后提交的归档的信息，以及有关任何正在进行的归档的信息。

语法：

```
onstat   -garc
```

示例输出

```
Dbspaces - Ongoing archives
number   name            Q Size Q Len  buffer partnum    size     Current-page
1        rootdbs         100    3      100    0x1001c9   0        1:128
3        datadbs01       0      0
4        datadbs02       0      0

Dbspaces - Archive Status
name               number level date             log        log-position
rootdbs            1      0     07/30/2009.09:59 28         0x320018
datadbs01          3      0     07/30/2009.09:59 28         0x320018
datadbs02          4      0     07/30/2009.09:59 28         0x320018
```

图 125: onstat -g arc 命令输出

输出描述 - 正在进行的归档

此输出部分表示归档的当前信息。如果系统中没有活动的归档，将不显示此部分。

列	描述
Number	数据库空间的编号
Name	数据库空间的名称
Q Size	前映像队列大小。该信息主要是供 星瑞格®支持使用。
Q Len	前映像队列长度。该信息主要是供 星瑞格®支持使用。
Buffer	前映像缓冲区使用的页数
Partnum	前映像存储区的分区号
Size	前映像存储区中的页数
Current-page	正在归档的当前页

注： 前映像存储区是创建在临时数据库空间或根数据库空间（如果您没有临时数据库空间）中的临时表。如果前映像存储区变小，其可扩展到其他分区，这种情况下，输出将会显示相同的数据库空间的多个 Partnum 和 Size 字段。

输出描述 - 归档状态

此输出部分包含每个数据库空间最后一次备份的信息。

列	描述
Name	数据库空间的名称
Number	数据库空间编号
Level	归档级别

列	描述
Date	最后一次归档的日期和时间
Log	用于启动归档的检查点的唯一 ID (UNIQID)
Log-position	用于启动归档的检查点的日志位置 (LOGPOS)

onstat -g ath 命令: 显示所有线程的信息

使用 onstat -g ath 命令显示所有线程的信息。

语法:

```
onstat   -gath
```

示例输出

```
Threads:
 tid     tcb             rstcb        prty status             vp-class      name
 2       10bbf36a8        0            1    IO Idle              3lio         lio vp 0
 3       10bc12218        0            1    IO Idle              4pio         pio vp 0
 4       10bc31218        0            1    running              5aio         aio vp 0
 5       10bc50218        0            1    IO Idle              6msc         msc vp 0
 6       10bc7f218        0            1    running              7aio         aio vp 1
 7       10bc9e540        10b231028    1    sleeping secs: 1     1cpu         main_loop()
 8       10bc12548        0            1    running              1cpu         tlitcppoll
 9       10bc317f0        0            1    sleeping forever     1cpu         tlitcplst
 10      10bc50438        10b231780    1    IO Wait              1cpu         flush_sub(0)
 11      10bc7f740        0            1    IO Idle              8aio         aio vp 2
 12      10bc7fa00        0            1    IO Idle              9aio         aio vp 3
 13      10bd56218        0            1    IO Idle             10aio         aio vp 4
 14      10bd75218        0            1    IO Idle             11aio         aio vp 5
 15      10bd94548        10b231ed8    1    sleeping forever     1cpu         aslogflush
 16      10bc7fd00        10b232630    1    sleeping secs: 34    1cpu         btscanner 0
 32      10c738ad8        10b233c38    1    sleeping secs: 1     1cpu         onmode_mon
 50      10c0db710        10b232d88    1    IO Wait              1cpu         sqlexec
```

图 127: onstat -g ath 命令输出

输出描述

tid
: 线程 ID

tcb
: 线程控制块访问地址

rstcb
: RSAM 线程控制块访问地址

prty
: 线程优先级

status
: 线程状态

vp-class
: 虚拟处理器类

name
: 线程名称。对于参与并行存储优化操作的线程，它是操作名称和线程编号。

- compress.*number* = 线程正在压缩数据
- repack.*number* = 线程正在重新打包数据

- uncompress.*number* = 线程正在解压缩数据
- update_ipa.*number* = 线程正在移除未完成的就地 alter 操作

相关链接

onstat -g wst 命令: 打印线程的等待统计信息 在第539页
NUMFDSERVERS 配置参数 在第138页
table 或 *fragment* 参数: 压缩数据和优化存储 *(SQL* 管理 *API)* 在第699页
《*SinoDB Enterprise Replication* 指南》: *onstat -g ath* 命令显示线程信息

onstat -g bth 和 -g BTH: 显示被阻挡和等待中的线程

使用 onstat -g bth 命令显示阻塞线程与等待线程之间的依赖关系。使用 onstat -g BTH 命令显示阻塞线虫病的会话和堆栈信息。

语法:

```
onstat   -g { bth  | BTH }
```

onstat -g bth 的示例输出

```
This command attempts to identify any blocking threads.

Highest level blocker(s)
 tid      name                session
 48       sqlexec             26

Threads waiting on resources
 tid      name                blocking resource           blocker
 49       sqlexec             MGM                         48
 13       readahead_0         Condition (ReadAhead)        -
 50       sqlexec             Lock (0x4411e578)           49
 51       sqlexec             Lock (0x4411e578)           49
 52       sqlexec             Lock (0x4411e578)           49
 53       sqlexec             Lock (0x4411e578)           49
 57       bf_priosweep()      Condition (bp_cond)          -
 58       scan_1.0            Condition (await_MC1)        -
 59       scan_1.0            Condition (await_MC1)        -

Run 'onstat -g BTH' for more info on blockers.
```

图 129: onstat -g bth 命令输出

onstat -g bth 的输出描述

tid
  线程 ID

name
  线程名称

session
  会话 ID

blocking resource
  所列线程正在等待的资源的类型

blocker
  阻塞所列线程的线程 ID

onstat -g BTH 的示例输出

```
Stack for thread: 48 sqlexec
```

```
 base: 0x00000000461a3000
  len:   69632
   pc: 0x00000000017b32c3
  tos: 0x00000000461b2e30
state: ready
   vp: 1

0x00000000017b32c3 (oninit) yield_processor_svp
0x00000000017bca6c (oninit) mt_wait
0x00000000019d4e5c (oninit) net_buf_get
0x00000000019585bf (oninit) recvsocket
0x00000000019d1759 (oninit) tlRecv
0x00000000019ce62d (oninit) slSQIrecv
0x00000000019c43ed (oninit) pfRecv
0x00000000019b2580 (oninit) asfRecv
0x000000000193db2a (oninit) ASF_Call
0x0000000000c855dd (oninit) asf_recv
0x0000000000c8573c (oninit) _iread
0x0000000000c835cc (oninit) _igetint
0x0000000000c72a9e (oninit) sqmain
0x000000000194bb38 (oninit) listen_verify
0x000000000194ab8a (oninit) spawn_thread
0x0000000001817de3 (oninit) th_init_initgls
0x00000000017d3135 (oninit) startup
```

该命令尝试标识任何阻塞的线程。

```
Highest level blocker(s)
 tid      name                session
 48       sqlexec             26

session          effective                      #RSAM    total   used   dynamic
id      user     user      tty   pid   hostname threads  memory  memory explain
26      informix -         45    31041 mors     2        212992  186568 off

Program :
/work3/JC/VIEWS/jc_dct_phase2.view/.s/00055/80003fd351f804d3dbaccess

tid      name     rstcb            flags    curstk   status
48       sqlexec  448bc5e8         ---P---  4560     ready-
58       scan_1.0 448bb478         Y------  896      cond wait  await_MC1 -

Memory pools    count 2
name         class addr           totalsize  freesize   #allocfrag #freefrag
26           V     45fcc040       208896     25616      189        16
26*00        V     462ad040       4096       808        1          1

name           free       used           name           free       used
overhead       0          6576           mtmisc         0          72
resident       0          72             scb            0          240
opentable      0          7608           filetable      0          1376
log            0          33072          temprec        0          17744
blob           0          856            keys           0          176
ralloc         0          55344          gentcb         0          2240
ostcb          0          2992           sqscb          0          21280
sql            0          11880          xchg_desc      0          1528
xchg_port      0          1144           xchg_packet    0          440
xchg_group     0          104            xchg_priv      0          336
hashfiletab    0          1144           osenv          0          2520
sqtcb          0          15872          fragman        0          1024
shmblklist     0          416            sqlj           0          72
rsam_seqscan   0          368
```

```
sqscb info
scb             sqscb          optofc   pdqpriority optcompind  directives
4499c1c0        461c1028       0        100         2           1

Sess       SQL            Current       Iso Lock       SQL  ISAM F.E.
Id         Stmt type      Database      Lvl Mode       ERR  ERR  Vers  Explain
26         SELECT         jc            CR  Not Wait   0    0    9.24  Off

Current statement name : unlcur

Current SQL statement (5) :
  select * from systables,syscolumns,sysfragments

Last parsed SQL statement :
  select * from systables,syscolumns,sysfragments
```

onstat -g BTH 的输出描述

tid

线程 ID

name

线程名称

session

会话 ID

会话信息部分包含的信息与 onstat -g ses 命令输出的信息相同。请参阅 *onstat -g ses* 命令: 显示会话相关信息 在第513页。

剩下的信息显示线程的堆栈信息。

相关链接

《*SinoDB* 性能指南》: 使用 *onstat -g bth* 和 *onstat -g BTH* 命令监视块线程

onstat -g buf 命令: 显示缓冲池的概要文件信息

使用 onstat -g buf 命令显示每个缓冲池的概要文件信息。

语法:

```
onstat   -gbuf
```

示例输出

onstat -g buf 命令的输出根据 BUFFERPOOL 配置参数设置是否包含 memory 字段或 buffers 字段而有些许不同。输出显示 memory 设置。buffers 设置的输出包含 max extends 和 next buffers 字段，而不是 max memory 和 next memory 字段。

```
Profile

Buffer pool page size: 2048
dskreads   pagreads   bufreads   %cached dskwrits   pagwrits   bufwrits   %cached
1190       1773       661359     99.82   16863      83049      185805     90.92
bufwrits_sinceckpt  bufwaits   ovbuff     flushes
11243               115        0          42

Fg Writes     LRU Writes    Avg. LRU Time Chunk Writes  Total Mem
0             0             nan           10883         32Mb

                                    cache
# extends  max memory  next memory  hit ratio  last
0          128Mb       32Mb         90         11:31:17

Bufferpool Segments
```

```
id segment      size     # buffs
0  0x449f0000   32Mb     13025

----------------------------------

Buffer pool page size: 8192
dskreads   pagreads   bufreads   %cached dskwrits   pagwrits   bufwrits   %cached
0          0          11         100.00  4          16         4          0.00
bufwrits_sinceckpt  bufwaits   ovbuff     flushes
0                   0          0          1

Fg Writes     LRU Writes    Avg. LRU Time Chunk Writes  Total Mem
0             0             nan           4             128Mb

                                    cache
# extends  max memory  next memory  hit ratio  last
0          1280Mb      128Mb        90         11:31:41

Bufferpool Segments
id segment      size     # buffs
0  0x4928e000   128Mb    14988

----------------------------------

Fast Cache Stats
gets       hits       %hits   puts
246854     244407     99.01   111147
```

图 131: onstat -g buf 对于 memory 设置的输出

输出描述

Buffer pool page size
缓冲池中页的字节数

dskreads
将页带入缓冲池中所执行的磁盘读取操作数。每个读取操作读取一个或多个页。

pagreads
从磁盘读入缓冲池的页数

bufreads
从该缓冲池读取页的内存映像的次数

%cached
缓冲池从高速缓冲页映像执行页读取（而不是执行磁盘读取）的百分比。计算为 (bufreads - dskreads) / bufreads x 100。百分比越高表示高速缓存的性能越好。

dskwrits
将已更改页从缓冲池写入磁盘所执行的磁盘写入操作数。每个写入操作写入一个或多个页。

pagwrits
从缓冲池写入磁盘的页数

bufwrits
写入该缓冲池的页内存映像的次数

%cached
该缓冲池向高速缓冲页映像执行页写入（而不是执行磁盘写入）的百分比。计算为 (bufwrits - dskwrits) / bufwrits x 100。

bufwrits_sinceckpt
自上个检查点之后页内存映像写入该缓冲池的次数

bufwaits

线程必须等待该缓冲池中缓冲区锁的次数。 数字越大表示多个线程之间对于相同页的互不兼容锁的争用越多。

ovbuff

为了创建空闲的缓冲区来读取另一请求的页而将已更改缓冲区从该缓冲池写入磁盘的次数。如果 ovbuff 值很大，那么该缓冲池可能不足以容纳应用程序所需的页的工作集。不足的缓冲池可导致性能下降。

flushes

服务器对缓冲池中所有脏缓冲区执行清空的次数。大量清空可能由于不同原因而发生，包括作为检查点处理的一部分，或缓冲池已用尽干净的缓冲区（尽管 LRU 清除活动正常）。

Fg Writes

该缓冲池通过访问缓冲区的非 I/O 清除程序写入磁盘的已更改的缓冲区数。该数字是 ovbuff 字段值的超集。除了在 ovbuff 字段中写入服务页错误外，该值还包括维持数据库日志和保留页一致以确保正确恢复的前台写入。

LRU Writes

该缓冲池通过 LRU 清除程序线程写入磁盘的已更改缓冲区的数量。如果缓冲池超过 BUFFERPOOL 配置参数的 lru_max_dirty 字段中指定的值或如果由于缓冲池溢出而发生前台写入，那么将激活 LRU 清除程序。

Avg. LRU Time

LRU 清除程序线程清除单个 LRU 链所用的平均时间量

Chunk Writes

通过块清除操作写入磁盘的已更改缓冲区的数量。块清除程序将缓冲池中块的所有已更改的缓冲区写出。在诸如检查点处理过程中和快速恢复过程中，该操作进行块清除以快速清除大量缓冲区。

Total Mem

缓冲池的大小

extends

缓冲池扩展的次数

max memory (memory setting)

缓冲次的目标最大大小。缓冲池的实际大小可超过此值，但不可超过一个段大小。

max extends (buffers setting)

缓冲池扩展的最多次数T。（此字段没有在示例输出中显示。）

next memory (memory setting)

缓冲池的下一个扩展的大小

next buffers (buffers setting)

缓冲池的下一个扩展的缓冲区数。（此字段没有在示例输出中显示。）

cache hit ratio

读高速缓存命中率，低于该值时扩展缓冲池。

last

缓冲池上次扩展的时间

id

缓冲池段的 ID

segment

缓冲池段的内部地址

size

缓冲池段的大小

buffs

缓冲池段中的缓冲区数

Fast Cache Stats
快速缓存（减少访问缓冲池所需时间的一种高速缓存）的统计信息

gets
服务器在快速缓存中寻找缓冲区的次数

hits
服务器在快速缓存中找到所要缓冲区的次数

%hits
命中百分比，其为 hits*100/gets.

puts
服务器向快速缓存插入缓冲区的次数

相关链接

BUFFERPOOL 配置参数 在第78页
《*SinoDB* 管理员指南》: 监视缓冲区

onstat -g cac 命令: 显示有关高速缓存的信息

使用 onstat -g cac 命令查看有关所有高速缓存或单个高速缓存的摘要和详细信息。

语法:

onstat -gcac [{ agg | aqt | am*access_method* | cast | dic | dsc | ed | lbacplcy | lbacusrc | poci | prc | prn | rr | ssc | ttype | typei*type_id* | typen*type_name* }]

使用不带任何选项的 onstat -g cac 命令查看所有高速缓存的信息。

使用以下选项查看特定高速缓存的信息:

agg
打印聚合高速缓存的信息

aqt
打印 AQT 字典高速缓存的信息。打印与 onstat -g aqt 命令相同的输出。请参阅 *onstat -g aqt* 命令: 打印数据市集和加速查询表的信息 在第422页。

am
打印有关访问方法高速缓存的信息。要查看特定访问方法的信息，请包括访问方法名称。

cast
打印有关 cast 高速缓存的信息

dic
打印有关数据字典高速缓存的信息。打印与 onstat -g dic 命令相同的输出。请参阅 *onstat -g dic* 命令: 打印表信息 在第451页。

dsc
打印有关数据分布高速缓存的信息。打印与 onstat -g dsc 命令相同的输出。请参阅 *onstat -g dsc* 命令: 显示分布式高速缓存信息 在第457页。

ed
打印有关外部指令高速缓存的信息。

lbacplcy
打印有关 LBAC 安全策略信息高速缓存的信息。

lbacusrc
打印有关 LBAC 证书内存高速缓存的信息。

opci
打印有关操作符类实例高速缓存的信息

prc
打印有关UDR 高速缓存的信息。打印与 onstat -g prc 命令相同的输出。请参阅 *onstat -g prc 命令: 显示使用 UDR 或 SPL 例程的会话* 在第493页。

prn
打印有关程序名称高速缓存的信息

rr
打印有关例程解析高速缓存的信息

ssc
打印有关 SQL 语句高速缓存的信息。打印与 onstat -g ssc 命令相同的输出。请参阅 *onstat -g ssc 命令: 显示 SQL 语句出现次数* 在第529页。

ttype
打印有关辅助瞬态高速缓存的信息

typei
按 ID 高速缓存打印扩展类型的信息。要查看特定扩展类型的信息，请包括扩展类型 ID。

typen
按名称高速缓存打印扩展类型的信息。要查看特定扩展类型的信息，请包括扩展类型名称。

示例输出

大多数 onstat -g cac 命令的输出包含类似的格式和信息。

以下输出是 onstat -g cac lbacplcy 命令的示例：

```
Security Policy Info Cache:
    Number of lists          : 31
    PLCY_POOLSIZE            : 127


Security Policy Info Cache Entries:

list id   ref  drop hits       heap_ptr         item
-------------------------------------------------------------

9    2    0    0    0          65f1b8d0         test@informix: : secpolicyid 2

15   1    0    0    0          65f1b4d0         test@informix: : secpolicyid 1


    Total number of entries  : 2
    Number of entries in use : 0
```

输出描述

大多数 onstat -g cac 命令的输出包含以下字段：

Number of lists
分布式高速缓存中列表数

configuration parameter name
一次可以高速缓存的条目数

list
分布式高速缓存散列链 ID

id
指定给该高速缓存条目的唯一 ID

ref
引用高速缓存条目的语句的数量
drop
指示该条目被添加到高速缓存后是否被删除
hits
访问高速缓存条目的次数
heap_ptr
用于存储该条目的堆地址
item name
高速缓存中的项的名称
Total number of entries
高速缓存中的条目数
Number of entries in use
正在使用的条目数

相关链接

《*SinoDB* 性能指南》：配置和监视内存高速缓存

onstat -g ckp 命令：打印检查点历史记录与配置建议

使用 onstat -g ckp 命令打印检查点历史记录，并显示配置建议（如果检测到次优的配置）。

语法：

```
onstat   -gckp
```

示例输出

```
Auto Checkpoints=On   RTO_SERVER_RESTART=60 seconds   Estimated recovery time 7 seconds
```

```
                                                          Critical Sections
         Clock                          Total  Flush  Block  #      Ckpt  Wait  Long  #Dirty
Interval Time     Trigger  LSN          Time   Time   Time   Waits  Time  Time  Time  Buffers
1        18:41:36 Startup  1:f8         0.0    0.0    0.0    0      0.0   0.0   0.0   4
2        18:41:49 Admin    1:11c12cc    0.3    0.2    0.0    1      0.0   0.0   0.0   2884
3        18:42:21 Llog     8:188        2.3    2.0    2.0    1      0.0   2.0   2.0   14438
4        18:42:44*User     10:19c018    0.0    0.0    0.0    1      0.0   0.0   0.0   39
5        18:46:21 RTO      13:188       54.8   54.2   0.0    30     0.6   0.4   0.6   68232
```

```
          Physical Log    Logical Log
  Dskflu  Total   Avg     Total   Avg
  /Sec    Pages   /Sec    Pages   /Sec
  4       3       0       1       0
  2884    1966    163     4549    379
  7388    318     10      65442   2181
  39      536     21      20412   816
  1259    210757  1033    150118  735
```

```
Max Plog   Max Llog    Max Dskflush    Avg Dskflush   Avg Dirty   Blocked
pages/sec  pages/sec   Time            pages/sec      pages/sec   Time
8796       6581        54              43975          2314         0
```

图 134: onstat -g ckp 命令输出

输出描述

Auto Checkpoints

指示 AUTO_CKPTS 配置参数是 on 或 off

RTO_SERVER_RESTART

显示以秒为单位的 RTO 时间。零（0）表示 RTO 是关闭的。

Estimated recovery time ## seconds

指示数据服务器停止响应时的估计恢复时间。此值仅在 RTO_SERVER_RESTART 处于活动时才会出现。

Interval

检查点间隔 ID

Clock Time

检查点发生时的时钟时间

Trigger

触发检查点的事件。星号（*）表示请求的检查点是事务阻塞的检查点。

触发器名称	描述
Admin	管理相关任务。例如： • 创建、删除或重命名数据库空间 • 添加或删除块 • 添加或删除日志文件 • 更改物理日志大小或位置 • 分区上 "shrink" 操作之后 • 打开或关闭镜像
Backup	备份相关操作。例如： • 假备份 • 开始归档 • 物理复原完成之后
CDR	第一次启动 ER 子系统，或在移除所有复制参与者之后重新启动 ER 子系统。
CKPTINTVL	检查点间隔过期时。检查点间隔是在 onconfig 文件中 CKPTINTVL 参数所指定的值。
Conv/Rev	转换复原检查点。在转换检查阶段后磁盘结构实际转换前。恢复完成后也将触发检查点。
HA	高可用性。例如： • 添加新的 RSS 或 SDS 节点到高可用性集群 • 辅助服务器提升为主服务器 • 辅助服务器上的物理日志文件很少
HDR	高可用性数据复制。例如： • 更改服务器的模式 • 安装 HDR 之后开始的第一次传输 • 主服务器或辅助服务器上潜在的物理日志溢出
IPL	触发检查点以减少辅助服务器上的物理日志使用。索引页日志记录可能导致辅助服务器上的前台写入和大量物理日志使用。

触发器名称	描述
Lightscan	在分区上关闭后备前。
Llog	耗尽逻辑日志资源。
LongTX	长事务。如果发现长事务但未停止，那么会发起检查点以停止该事务。如果在长事务终止之后未发生检查点，那么会在回滚阶段发起检查点。
Misc	杂项事件。例如： • 由于 I/O 错误关闭数据库空间或块 • 在回滚过程中，撤销块的添加时。例如，删除块时。 • 在群集索引创建过程中或更改索引为群集索引时
Plog	物理日志具有以下之一的状况： • 物理日志已 75% 满 • 已用的物理日志量加上脏分区数已超过物理日志大小的 90%。
Restore Pt	复原点。在复原点开始和结束时的检查点。CONVERSION_GUARD 配置参数启用的点为复原点（由转换保护程序使用），且 RESTORE_POINT_DIR 配置参数指定临时目录。
Recovery	在恢复过程中，快速恢复开始时。
Reorg	在联机索引建构开始时
RTO	维护恢复时间目标（RTO）策略。在正常操作过程中，当系统崩溃后重新启动的时间可能超过 RTO_SERVER_RESTART 配置参数设置的值时。
Stamp Wrap	检查点时间戳。如果新检查点时间戳出现在上次写入检查点之前，则该时间戳被推前超出检查点之间的间隔。这样会触发另一检查点。
Startup	在数据库服务器启动时。
Uncompress	在表或分区上使用解压缩命令。这个只应用于表上的检查点或没有日志记录的数据库。
User	用户提出检查点请求。

LSN

检查点记录在逻辑日志中的位置

Total Time

以秒表示从请求开始到检查点完成的检查点的总持续时间

Flush Time

以秒表示清空缓冲池的时间

Block Time

事务被由于某些必需资源不足而触发的检查点阻塞的时间（以秒为单位）。例如：耗尽物理日志，或逻辑日志环绕。

Waits

由于正在等待检查点而阻塞的事务数

Ckpt Time
所有事务识别到请求的检查点的时间（以秒为单位）
Wait Time
事务等待检查点的平均时间（以秒为单位）
Long Time
事务等待检查点的最长时间量（以秒为单位）
Dirty Buffers
检查点期间清空到磁盘的脏缓冲区的数量
Dskflu/sec
每秒清空的缓冲区数
Physical Log Total Pages
检查点间隔期间物理日志记录的总页数
Physical Log Avg/Sec
检查点间隔期间平均物理日志活动率
Logical Log Total Pages
检查点间隔期间逻辑日志记录的总页数
Logical Log Avg/Sec
检查点间隔期间平均逻辑日志活动率
Max Plog pages/sec
检查点间隔期间最大物理日志活动率
Max Llog pages/sec
检查点间隔期间最大逻辑日志活动率
Max Dskflush Time
将缓冲池清空到磁盘的最大时间（以秒为单位）
Avg Dskflush pages/sec
将缓冲池清空到磁盘的平均速率
Avg Dirty pages/sec
检查点间的平均脏页速率
Blocked Time
自上次数据库服务器启动以来最长的阻塞时间（以秒为单位）

性能建议消息

如果 SinoDB® 数据服务器检测到一个次优的配置，那么在检查点历史记录下面会出现性能建议消息，以及调整建议。该性能建议消息也会在消息日志中出现。以下是性能建议消息的示例：

```
Physical log is too small for bufferpool size. System performance may be
less than optimal.
Increase physical log size to at least %ldKb

Physical log is too small for optimal performance.
Increase the physical log size to at least $ldKb.

Logical log space is too small for optimal performance.
Increase the total size of the logial log space to at least %ld Kb.

Transaction blocking has taken place. The physical log is too small.
Please increase the size of the physical log to %ldKb

Transaction blocking has taken place. The logical log space is too small.
```

```
Please increase the size of the logical log space to %ldKb
```

相关链接

sysckptinfo 在第207页

onstat -g cfg 命令: 显示配置参数的当前值

使用 onstat -g cfg 命令打印配置参数列表及其当前值。可以使用更多命令选项来打印有关配置参数的更多信息。

语法:

onstat -gcfg [{ full | diff | tunable | msg }] [*config_parameter_name*]

onstat -g cfg 命令具有以下格式:

onstat -g cfg 命令选项

命令	描述
onstat -g cfg	显示配置参数列表，以及其当前值。
onstat -g cfg *config_parameter_name*	只显示指定的配置参数的当前值。
onstat -g cfg full	显示有关每个配置参数的所有信息，包括参数的当前值、缺省值、onconfig 文件中的值，和描述。
onstat -g cfg full *config_parameter_name*	显示有关指定参数的所有信息。
onstat -g cfg diff	显示配置参数当前值不同于 onconfig 文件中的永久值的信息。
onstat -g cfg tunable	显示所有可调整参数的缺省值、原始值和当前值。星号表示可动态调整该配置参数。
onstat -g cfg msg	显示任何消息，例如：与配置参数相关联的警告或调整。

示例输出

以下是 onstat -g cfg 命令的部分样本输出，其显示 DEADLOCK_TIMEOUT 配置参数值在数据库服务器启动后动态更改为 90 秒:

```
id   name                 type   units   rsvd  tunable
26   DEADLOCK_TIMEOUT     INT4   Seconds       *

     min/max : 0,2147483647
     default : 60
     onconfig:
     current : 90

     Description:
     Use the DEADLOCK_TIMEOUT configuration parameter to specify the
     maximum number of seconds that a database server thread can wait
     to acquire a lock.
```

```
ROOTNAME                rootdbs
```

以下是 onstat -g cfg diff 命令的部分样本输出，其显示 TBLTBLFIRST 和 TBLTBLNEXT 配置参数的缺省值、当前值和在 onconfig 文件中的值:

```
id   name                    type   units   rsvd  tunable
53   TBLTBLFIRST             INT4   KB            *
```

```
     default : 500
     onconfig: 0
     current : 250


id   name                      type   units   rsvd  tunable
54   TBLTBLNEXT                INT4   KB            *

     default : 100
     onconfig: 0
     current : 150
```

以下部分样本输出显示 MSGPATH 配置参数的信息。此处，该配置参数没有缺省值，并且其 onconfig 文件中的值与当前值是相同的。

```
id   name                      type   units   rsvd  tunable
10   MSGPATH                   CHAR           *     *

     default :
     onconfig: /work2/JC/online.log
     current : /work2/JC/online.log
```

以下是 onstat -g cfg msg 命令的部分样本输出，其显示标识已更改值的配置参数的消息：

```
Configuration Parameters With Messages

name             message
TBLTBLFIRST      Parameter's user-configured value was adjusted.
TBLTBLNEXT       Parameter's user-configured value was adjusted.
BUFFERPOOL       Parameter's user-configured value was adjusted.
STACKSIZE        Parameter's user-configured value was adjusted.
VPCLASS          Parameter's user-configured value was adjusted.
```

输出描述

name
  配置参数的名称

type
  该值的数据类型

units
  表示值的单位

rsvd
  指示（以星号）该配置参数和其值存储在配置保留页

  如果没有星号，那么该配置参数及其值不是存储在配置保留页。

tunable
  指示（以星号）该配置参数可动态调整。例如：使用 onmode -wm 或 -wf 命令。

  如果没有星号，那么该配置参数不能动态调整。

min/max
  配置参数的最小值与最大值

default
  配置参数内建于服务器的缺省值

onconfig
  onconfig.std 文件中的配置参数值（如果有）

current
配置参数的当前值
如果动态地修改了该值，例如：使用 onmode -wm 命令修改，那么当前值是不同的。

Description
配置参数的描述

message
标识更改的配置参数值的消息

相关链接
显示 *onconfig* 文件中的设置 在第43页
onstat -c 命令: 显示 *ONCONFIG* 文件内容 在第408页
onmode -wf#-wm: 动态更改某些配置参数 在第347页
数据库服务器文件 在第715页

onstat -g cluster 命令: 打印高可用性集群信息

使用 onstat -g cluster 命令显示高可用性集群环境中服务器的信息。
语法:
onstat -gcluster [{ verbose }]

onstat -g cluster 命令合并了 onstat -g dri、onstat -g sds 和 onstat -g rss 的功能。onstat -g cluster 命令的输出根据该命令是在主服务器上运行还是在其他任一辅助服务器上运行而稍有不同。

示例输出（主服务器）

以下是 onstat -g cluster 命令的样本输出。该样本显示在主服务器上运行该命令时的输出。

```
Primary Server:serv1
Current Log Page:16,476
Index page logging status: Enabled
Index page logging was enabled at: 2013/12/11 14:05:17


Server ACKed Log    Applied Log  Supports     Status
       (log, page)  (log, page)  Updates
serv2  16,476       16,476       Yes          SYNC(SDS),Connected,Active
serv3  16,476       16,476       Yes          ASYNC(HDR),Connected,On
serv4  16,476       16,476       Yes          ASYNC(RSS),Connected,Active
```

图 137: onstat -g cluster 命令输出（在主服务器上运行）

输出描述（主服务器）

Primary server
指定给主服务器的名称

Current log page
当前日志页的日志 ID 和页号

Index page logging status
指示索引页日志记录是启用还是禁用

Index page logging was enabled at
启用索引页日志记录的日期和时间

Server
辅助服务器的名称

ACKed Log (log, page)

最后应答的日志传输的日志 ID 和页号

Applied Log (log, page)

最后应用的日志传输的日志 ID 和页号

Supports Updates

显示客户端应用程序是否可在辅助服务器上执行更新、插入和删除操作（由 UPDATABLE_SECONDARY 配置参数指定）

Status

显示辅助服务器的连接状态

示例输出（主服务器，verbose 输出）

以下是 onstat -g cluster verbose 命令的样本输出。该样本显示在主服务器上使用 verbose 选项运行该命令时的输出。

```
Primary Server:serv1
 Current Log Page:16,479
 Index page logging status: Enabled
 Index page logging was enabled at: 2013/12/11 14:05:17


 ------------------------------------------------------------

 server name: serv3
 type: ASYNC (HDR)
 control block: 0x4b673018
 server status: On
 connection status: Connected
 Last log page sent (log id, page): 16,479
 Last log page acked (log id, page); 16,479
 Last log page applied (log id, page); 16,479
 Approximate log page backlog: 0
 SDS cycle not used
 Delayed Apply Not Used
 Stop Apply Not Used
 Time of last ack: 2013/12/11 14:09:12
 Supports Updates: Yes


 ------------------------------------------------------------

 server name: serv2
 type: SYNC (SDS)
 control block: 0x4c2de0b8
 server status: Active
 connection status: Connected
 Last log page sent (log id, page): 16,479
 Last log page acked (log id, page); 16,479
 Last log page applied (log id, page); 16,479
 Approximate log page backlog: 0
 SDS cycle current: 20 ACKed: 20
 Delayed Apply Not Used
 Stop Apply Not Used
 Time of last ack: 2013/12/11 14:09:13
 Supports Updates: Yes
```

图 138: onstat -g cluster verbose 命令输出（在主服务器上运行）

输出描述（主服务器，verbose 输出）

Primary server

主服务器的名称

Current log page

当前日志页的日志 ID 和页号

Index page logging status

指示索引页日志记录是启用还是禁用

Index page logging was enabled at

启用索引页日志记录的日期和时间

Server name

辅助服务器的名称

type

显示辅助服务器是同步（SYNC）还是异步（ASYNC）连接。还显示辅助服务器的类型：HDR、SDS 或 RSS。

control block

线程控制块在内存的地址

server status

显示辅助服务器的当前状态

connection status

显示辅助服务器的当前网络连接状态

Last log page sent (log id, page)

主服务器最近送到辅助服务器的日志页的日志 ID 和页号

Last log page acked (log id, page)

辅助服务器最近应答的日志页的日志 ID 和页号

Last log page applied (log id, page)

辅助服务器最近应用的日志页的日志 ID 和页号

Approximate log page backlog

指示辅助服务器还未处理的大约日志页数

SDS cycle

指示主服务器前进的循环号和共享磁盘辅助服务器应答的循环号。由 星瑞格® 支持内部用于监视主服务器与辅助服务器的协调性。

Delayed Apply

指示辅助服务器在应用日志前是否等待指定的时间量（由 DELAY_APPLY 配置参数指定）

Stop Apply

指示辅助服务器是否已停止应用从主服务器接收到的日志文件（由 STOP_APPLY 配置参数指定）

Time of last ack

最后应答日志的日期和时间

Supports Updates

显示客户端应用程序是否可以在辅助服务器上执行更新、插入和删除操作（由 UPDATABLE_SECONDARY 配置参数指定）

示例输出（辅助服务器）

以下是 onstat -g cluster 命令的样本输出。该样本显示在辅助服务器上运行该命令时的输出。

```
Primary Server:serv1
```

```
Index page logging status: Enabled
Index page logging was enabled at: 2010/01/11 14:05:17


Server ACKed Log    Applied Log   Supports     Status
       (log, page)  (log, page)   Updates
serv2  16,479       16,479        Yes          SYNC(SDS),Connected,Active
```

图 139: onstat -g cluster 命令输出（在辅助服务器上运行）

输出描述（辅助服务器）

Primary server

主服务器的名称

Index page logging status

指示索引页日志记录是启用还是禁用

Index page logging was enabled at

启用索引页日志记录的日期和时间

Server

辅助服务器的名称

ACKed Log (log, page)

最后应答的日志的日志 ID 和页号

Applied Log (log, page)

最后应用的日志传输的日志 ID 和页号

Supports Updates

显示客户端应用程序是否可以在辅助服务器上执行更新、插入和删除操作（由 UPDATABLE_SECONDARY 配置参数指定）

Status

显示辅助服务器的连接状态

相关链接

DELAY_APPLY 配置参数 在第95页

STOP_APPLY 配置参数 在第182页

UPDATABLE_SECONDARY 配置参数 在第190页

onstat -g cmsm command: 显示连接管理器信息

使用 onstat -g cmsm 命令显示连接到运行该命令的数据库服务器的特定的连接管理器或所有连接管理器的信息。

语法：

onstat -gcmsm*connection_manager_name*

用法

onstat -g cmsm 显示有关连接管理器连接的连接单元、每个连接管理器服务等级协议处理的连接数、SLA 定义，故障转移顺序规则、故障转移仲裁，以及主服务器状态的信息。

使用 *connection_manager_name* 显示特定连接管理器实例的xinx。如果未指定 *connection_manager_name*，那么 onstat -g cmsm 显示连接到该数据库服务器的所有连接管理器实例的信息。

示例输出 1：特定连接管理器的输出

在以下示例中，onstat -g cmsm connection_manager_1 在 my_cluster_1 的主服务器上运行。

```
Unified Connection Manager: connection_manager_1               Hostname: my_host_1

CLUSTER         my_cluster_1      LOCAL
   SLA          Connections   Service/Protocol   Rule
   oltp_1            35     19910/onsoctcp   DBSERVERS=primary
   report_1          33     19810/onsoctcp   DBSERVERS=(HDR,SDS,RSS)

   Failover Arbitrator: Active Arbitrator, Primary is up
   ORDER=SDS,HDR,RSS PRIORITY=1
```

该命令显示 connection_manager_1 的输出。connection_manager_1 管理一个 CLUSTER 连接单元，并且是主故障转移仲裁者。

示例输出 2：高可用性集群的输出

在以下示例中，onstat -g cmsm 在 my_cluster_2 的主服务器上运行。

```
Unified Connection Manager: connection_manager_2               Hostname: my_host_2

CLUSTER         my_cluster_2      LOCAL
   SLA          Connections   Service/Protocol   Rule
   sla_1            1535     19910/onsoctcp   DBSERVERS=primary
   sla_2            2133     19810/onsoctcp   DBSERVERS=(HDR,SDS,RSS)

        Failover Arbitrator: Active Arbitrator, Primary is up
        ORDER=SDS,HDR,RSS PRIORITY=1

CLUSTER         my_cluster_3
   SLA          Connections   Service/Protocol   Rule
   sla_3             730     19930/onsoctcp   DBSERVERS=primary
   sla_4             901     19830/onsoctcp   DBSERVERS=(HDR,SDS,RSS)

   Failover Arbitrator: Active Arbitrator, Primary is up
   ORDER=SDS,HDR,RSS PRIORITY=1

Unified Connection Manager: connection_manager_3               Hostname: my_host_3

CLUSTER         my_cluster_2      LOCAL
   SLA          Connections   Service/Protocol   Rule
   sla_5             614     19920/onsoctcp   DBSERVERS=primary
   sla_6             483     19820/onsoctcp   DBSERVERS=(HDR,SDS,RSS)

   Failover Arbitrator: Failover is enabled
   ORDER=SDS,HDR,RSS PRIORITY=2

CLUSTER         my_cluster_3
   SLA          Connections   Service/Protocol   Rule
   sla_7             678     19940/onsoctcp   DBSERVERS=primary
   sla_8             270     19840/onsoctcp   DBSERVERS=(HDR,SDS,RSS)

        Failover Arbitrator: Failover is enabled
        ORDER=SDS,HDR,RSS PRIORITY=2
```

该命令显示连接到该集群的主服务器的两个连接管理器的输出。connection_manager_2 和 connection_manager_3 安装在不同的主机上，并一起管理两个 CLUSTER 连接单元。connection_manager_2 是这两个 CLUSTER 连接单元的主故障转移仲裁者。

示例 3：复制集的输出

在以下示例中，onstat -g cmsm 在 my_replicate_set_1 中的复制服务器上运行。

```
Unified Connection Manager: connection_manager_4               Hostname: my_host_4

REPLSET         my_replicate_set_1
```

```
   SLA          Connections   Service/Protocol   Rule
   sla_1                160     19810/onsoctcp   DBSERVERS=ANY

Unified Connection Manager: connection_manager_5               Hostname: my_host_5

REPLSET         my_replicate_set_1
   SLA          Connections   Service/Protocol   Rule
   sla_2                240     19820/onsoctcp   DBSERVERS=ANY
```

该命令显示连接到该复制服务器的两个连接管理器的输出。 connection_manager_4 和 connection_manager_5 是安装在不同的主机上，并且一起管理复制服务器。

示例 4：网格的输出

在以下示例中，onstat -g cmsm 在 my_grid_1 的节点上运行。

```
Unified Connection Manager: connection_manager_6               Hostname: my_host_6

GRID    my_grid_1
   SLA          Connections   Service/Protocol   Rule
   sla_1                456     19830/onsoctcp   DBSERVERS=(group_name_1,group_name_2)
 POLICY=FAILURE

Unified Connection Manager: connection_manager_7               Hostname: my_host_7

GRID    my_grid_1
   SLA          Connections   Service/Protocol   Rule
   sla_2                785     19840/onsoctcp   DBSERVERS=(group_name_1,group_name_2)
 POLICY=FAILURE
```

该命令显示连接到该网络的两个连接管理器的输出。该命令显示连接到该节点的两个连接管理器的输出。connection_manager_6 和 connection_manager_7 是安装在不同主机上，并且一起管理该网格。

示例 5：服务器集的输出

在以下示例中，onstat -g cmsm 在服务器集中的独立服务器上运行。

```
Unified Connection Manager: connection_manager_8                    Hostname: my_host_8

SERVERSET       server_1,server_2
   SLA          Connections   Service/Protocol   Rule
   sla_1                 63     19810/onsoctcp   DBSERVERS=(server_1,server_2)
 POLICY=ROUNDROBIN

Unified Connection Manager: connection_manager_9                    Hostname: my_host_9

SERVERSET       server_1,server_2
   SLA          Connections   Service/Protocol   Rule
   sla_2                 63     19810/onsoctcp   DBSERVERS=(server_1,server_2)
 POLICY=ROUNDROBIN
```

该命令显示连接到该服务器集的两个连接管理器的输出。connection_manager_8 和 connection_manager_9 是安装在不同主机上，并且一起管理该服务器集。

输出描述

onstat -g cmsm 命令的输出包含每个连接管理器的部分。每个部分显示连接管理器实例名称和主机名，后跟的子部分包含连接管理器所连接的每个连接单元的信息。

Unified Connection Manager

连接管理器实例的名称

Hostname
连接管理器的主机的名称

SLA
定义在连接管理器配置文件中的服务等级协议的名称

Connections
连接管理器启动后每个 SLA 处理的连接数

Service/Protocol
与 SLA 相关联的端口号或服务名称，以及连接协议类型

Rule
SLA 定义

Failover Arbitrator:
指定连接管理器是否为主故障转移仲裁者（如果主服务器是活动的，且启用了故障转移）。仅用于显示 CLUSTER 连接单元。

ORDER
指定集群的故障转移顺序。仅用于显示 CLUSTER 连接单元。

PRIORITY
指定连接管理器与集群的主服务器之间连接的优先级。仅用于显示 CLUSTER 连接单元。

相关链接

《*SinoDB* 管理员指南》：通过连接管理器进行连接管理

《*SinoDB* 管理员指南》：连接管理的监视和故障排除

onstat -g con 命令：打印条件和线程信息

使用 onstat -g con 命令显示条件和正在等待这些条件的线程的信息。

语法：

```
onstat   -gcon
```

示例输出

```
Conditions with waiters:
cid      addr             name              waiter   waittime
271      c63d930          netnorm           1511     6550
```

图 142: onstat -g con 命令输出

输出描述

cid
条件标识符

addr
条件控制块地址

name
线程正在等待的条件的名称

waiter
正在等待条件的线程的 ID

waittime
线程已经等待此条件的时间（以秒为单位）

onstat -g cpu: 显示运行时统计信息

使用 onstat -g cpu 命令显示正在服务器上运行的每个线程的运行时统计信息。

语法：

```
onstat   -gcpu
```

示例输出

```
onstat -g cpu

Thread CPU Info:
tid    name              vp       Last Run           CPU Time     #scheds    status
2      lio vp 0          3lio*   07/18 08:35:35        0.0000           1    IO Idle

3      pio vp 0          4pio*   07/18 08:35:36        0.0102           2    IO Idle

4      aio vp 0          5aio*   07/18 08:35:47        0.6876          68    IO Idle

5      msc vp 0          6msc*   07/18 11:47:24        0.0935          14    IO Idle

6      main_loop()       1cpu*   07/18 15:02:43        2.9365       23350    sleeping secs: 1

7      soctcppoll        7soc*   07/18 08:35:40        0.1150           1    running

8      soctcpio          8soc*   07/18 08:35:40        0.0037           1    running

9      soctcplst         1cpu*   07/18 11:47:24        0.1106          10    sleeping forever

10     soctcplst         1cpu*   07/18 08:35:40        0.0103           6    sleeping forever

11     flush_sub(0)      1cpu*   07/18 15:02:43        0.0403       23252    sleeping secs: 1

12     flush_sub(1)      1cpu*   07/18 15:02:43        0.0423       23169    sleeping secs: 1

13     flush_sub(2)      1cpu*   07/18 15:02:43        0.0470       23169    sleeping secs: 1

14     flush_sub(3)      1cpu*   07/18 15:02:43        0.0407       23169    sleeping secs: 1

15     flush_sub(4)      1cpu*   07/18 15:02:43        0.0307       23169    sleeping secs: 1

16     flush_sub(5)      1cpu*   07/18 15:02:43        0.0323       23169    sleeping secs: 1

17     flush_sub(6)      1cpu*   07/18 15:02:43        0.0299       23169    sleeping secs: 1

18     flush_sub(7)      1cpu*   07/18 15:02:43        0.0314       23169    sleeping secs: 1

19     kaio              1cpu*   07/18 14:56:42        1.4560     2375587    IO Idle

20     aslogflush        1cpu*   07/18 15:02:43        0.0657       23166    sleeping secs: 1

21     btscanner_0       1cpu*   07/18 15:00:53        0.0484         784    sleeping secs:
 61
37     onmode_mon        1cpu*   07/18 15:02:43        0.3467       23165    sleeping secs: 1

43     dbScheduler       1cpu*   07/18 14:58:14        1.6613         320    sleeping secs:
 31
44     dbWorker1         1cpu*   07/18 13:48:10        0.4264         399    sleeping forever

45     dbWorker2         1cpu*   07/18 14:48:11        1.9346        2936    sleeping forever
```

```
94     bf_priosweep()    1cpu*   07/18 15:01:42        0.0431         77    cond wait
 bp_cond
```

图 144: onstat -g cpu 命令输出

输出描述

tid
  线程 ID

name
  线程名称

vp
  正在运行线程的虚拟处理器的 ID

Last Run
  线程上次运行的时间戳

CPU Time
  直到现在线程运行的时间

#scheds
  线程被调度运行的次数

status
  线程的状态。可能的状态值如下：

- cond wait
- IO Idle
- join wait
- mutex wait
- ready
- sleeping
- terminated
- running
- yield

onstat -g dbc 命令: 打印 dbScheduler 和 dbWorker 线程统计信息

使用 onstat -g dbc 命令显示当前正在运行的调度程序任务（由 dbWorker 线程处理），或安排要运行的调度程序任务（由 dbScheduler 线程处理）的统计信息。
语法：
onstat -gdbc

示例输出

```
Worker Thread(0)    46fa6f10
====================================
Task:                47430c18
Task Name:           mon_config_startup
Task ID:             3
Task Type:           STARTUP SENSOR
Last Error
    Number               -310
    Message              Table (informix.mon_onconfig) already exists in database.
    Time                 09/11/2007 11:41
    Task Name            mon_config_startup

Task Execution:      onconfig_save_diffs
```

```
WORKER PROFILE
    Total Jobs Executed            10
    Sensors Executed                8
    Tasks Executed                  2
    Purge Requests                  8
    Rows Purged                     0
```

```
Worker Thread(1)    46fa6f80
====================================
Task:                4729fc18
Task Name:           mon_sysenv
Task ID:             4
Task Type:           STARTUP SENSOR
Task Execution:      insert into mon_sysenv select 1, env_name, env_value FROM
                     sysmaster:sysenv


WORKER PROFILE
    Total Jobs Executed             3
    Sensors Executed                2
    Tasks Executed                  1
    Purge Requests                  2
    Rows Purged                     0


Scheduler Thread    46fa6f80
====================================
Run Queue
    Empty
Run Queue Size        0
Next Task             7
Next Task Waittime    57
```

图 146: onstat -g dbc 命令输出

输出描述

Worker Thread

共享内存中工作程序线程的地址

Task

上次执行的任务的名称

Task ID

该任务的任务 ID，其来自 sysadmin:ph_task 表中的 tk_id 列

Task Type

任务类型

Last Error

dbWorker 线程遭遇的最后错误的错误号、错误消息、时间（以秒为单位）和任务名称。其可能来自于先前执行的任务或数天前执行的任务。

Task Execution

作为任务一部分所执行的 SQL 语句或 SPL 过程或例程

WORKER PROFILE

dbWorker 线程概要文件数据显示总执行工作量、已执行的传感器数、已执行的任务数、清除请求数，以及从该 dbWorker 线程所执行的所有传感器的结果表中清除的行数。

Scheduler Thread

共享内存中调度程序线程的地址

Run Queue
下一个调度的任务的任务 ID。如果没有要调度的任务，那么其值为 Empty。
Run Queue Size
正在等待 dbWorker 线程执行的任务数
Next Task
下一个要调度执行的任务的任务 ID
Next Task Waittime
在 Next Task 被调度执行之前等待的秒数

相关链接
《SinoDB 管理员指南》: 监视调度程序
scheduler 参数: 停止或启动调度程序 (SQL 管理 API) 在第677页

onstat -g defragment 命令: 显示分区扩展数据块取消分段信息

使用 onstat -g defragment 命令显示有关分区扩展数据块取消分段的活动请求的信息。
语法:
onstat -gdefragment

示例输出

```
Defrag info
id  table name                    tid  dbsnum  partnum  status                substatus
 errnum
15  stores_demo:informix.stdtab2  49   2       2097155  SEARCHING_FOR_EXTENT  0          0
```

图 148: onstat -g defragment 命令输出

注: 此命令显示活动的取消分段请求的信息。如果没有活动的取消分段请求，那么仅返回列标题。

输出描述

id
取消分段请求的 ID
table name
正在进行取消分段的表的全名
tid
线程 ID
dbsnum
正在进行取消分段的数据库空间编号
partnum
正在进行碎片重整的分区编号
status

- SEARCHING_FOR_EXTENT
- MERGING_EXTENTS
- DEFRAG_COMPLETED
- DEFRAG_FAILED

substatus
详细状态号（如果有）

errnum

取消分段请求返回的最后错误号

onstat -g dic 命令: 打印表信息

使用 onstat -g dic 命令显示高速缓存在共享内存字典中的每个表的一行信息。如果指定表名称，那么此命令打印有关该表的内部 SQL 信息。

语法:

```
onstat   -gdic
```

示例输出

```
Dictionary Cache:  Number of lists: 31, Maximum list size: 10
list#  size  refcnt  dirty?  heapptr        table name
--------------------------------------------------------
   1     3      1      no     14b5d890   wbe@oninit_shm:informix.t0010url
                1      no     14cbb820   wbe@oninit_shm:informix.t9051themeval
                0      no     14b63c20   wbe@oninit_shm:informix.t0060hits

   2     2      0      no     14b97420   wbe@oninit_shm:informix.t0120import
                1      no     14b6c820   wbe@oninit_shm:informix.t9110domain

   3     3      0      no     14bce020   wbe@oninit_shm:informix.t0150url
                0      no     14d3d820   contact@oninit_shm:informix.wbtags
                0      no     14c87420   wbe@oninit_shm:informix.wbtags

   4     1      0      no     14b7a420   drug@oninit_shm:abcdef.product      ..
Total number of dictionary entries: 36
```

图 150: onstat -g dic 命令输出

输出描述

list#

数据字典散列链 ID

size

此散列中的条目数

refcnt

当前引用高速缓存条目之一的 SQL 语句数

dirty?

条目自上次写入磁盘以来是否已被修改

heapptr

用于存储此表的堆的地址

table name

高速缓存中表的名称

onstat -g dis 命令: 打印数据库服务器信息

使用 onstat -g dis 命令显示数据库服务器列表、每个服务器的状态，以及有关每个服务器的信息，其包括 INFORMIXDIR 目录的位置、sqlhosts 文件和 ONCONFIG 文件。您可以在数据库服务器处于任何模式（包括脱机）时使用此命令。

语法:

```
onstat   -gdis
```

示例输出

```
There are 2 servers found
Server        : ol_tuxedo
Server Number : 53
Server Type   : IDS
Server Status : Up
Server Version: IBM Informix Version 11.50.UC1
Shared Memory : 0xa000000
INFORMIXDIR   : /local1/engines/ol_tuxedo/dist
ONCONFIG      : /local1/engines/ol_tuxedo/dist/etc/onconfig.ol_tuxedo
SQLHOSTS      : /local1/engines/ol_tuxedo/dist/etc/sqlhosts
Host          : avocet

Server        : ol_9next
Server Number : 0
Server Type   : IDS
Server Status : Down
Server Version:
Shared Memory : 0
INFORMIXDIR   : /local1/engines/ol_9next/dist
ONCONFIG      :
SQLHOSTS      :
Host          :
```

图 152: onstat -g dis 命令输出

输出描述

Server
  服务器名称

Server Number
  服务器编号

Server Type
  服务器类型

Server Status
  Up 表示服务器处于联机，Down 表示服务器脱机。

Server Version
  服务器版本

Shared Memory
  共享内存地址的位置

INFORMIXDIR
  在 UNIX™ 上 $INFORMIXDIR/ 目录的位置，在 Windows™ 上在 %INFORMIXDIR%\ 目录中。

ONCONFIG
  ONCONFIG 文件的位置

SQLHOSTS
  sqlhosts 文件的位置

Host
  服务器的主机名称

onstat -g dll 命令: 打印动态链接库文件列表

使用 onstat -g dll 命令显示已加载的动态链接库（DLL）文件的列表和状态。

语法:

```
onstat   -gdll
```

示例输出

每个进程组，输出仅显示一次库文件的名称。标志指示当服务器启动时，库是否已加载。

```
addr            slot  vp baseaddr        flags   filename
0x4af55310      15    1  0x2a985e3000    PM      /finance/jeffzhang/mylib.udr
0x4b6f2310            2  0x2a985e3000
0x4b71b310            3  0x2a985e3000
0x4c09f310      16    1  0x2a985e3000    M       /deptxyz/udrs/geodetic.bld
0x4c0c0310            2  0x2a985e3000
0x4c0f1310            3  0x2a985e3000
0x4c112310      17    1  0x7a138e9000            /home/informix/extend/blade.so
0x4c133310            2  0x3a421e1000
0x4c133310            3  0x3a421e1000
```

图 154: onstat -g dll 命令输出

输出描述

addr
DLL 文件的地址

slot
在库表中的槽编号条目

vp
虚拟处理器的 ID

baseaddr
共享内存的基地址

flags

- M 表示呼叫 UDR 的线程可以从一个 CPU 虚拟处理器转移到另一 CPU 虚拟处理器。
- P 表示当数据库服务器启动时加载共享库。

filename
DLL 文件的名称

相关链接

PRELOAD_DLL_FILE 配置参数 在第146页

onstat -g dmp 命令: 打印原内存

使用 onstat -g dmp 命令显示在给定地址的若干给定字节的原内存的信息。
语法:
onstat -gdmp*addresslength*

每个地址和长度必须在 onstat -g seg 输出中显示的分配内存之内。指定的地址格式可以是十进制或十六进制。十六进制地址必须以 0x 开头。可以指定地址为十进制，但这样做需要在使用它作为命令行参数之前，将 onstat -g seg 显示的内存转换为十进制。

示例输出

```
%onstat -g dmp 0x700000011a19d48 100

address           bytes in mem
0700000011a19d48: 07000000 118e0fa8 07000000 11942b40 ........ ......+@
0700000011a19d58: 07000000 10137120 00000000 00000000 ......q ........
0700000011a19d68: 00000000 00000000 00000000 00000000 ........ ........
0700000011a19d78: 07000000 11a19d48 07000000 11a19d48 .......H .......H
0700000011a19d88: 00000000 00000000 00000000 00000000 ........ ........
0700000011a19d98 *
```

```
0700000011a19da8: 00000000 ....
```

图 156: onstat -g dmp 命令输出

输出描述

address

原内存的内存地址

bytes in mem

内存内容的十六进制值和 ASCII 表示

该命令的输出分为三列：内存地址、内存中字节的十六进制值和内存中字节的 ASCII 表示。内存（中间）部分所显示的字节是从命令行所指定地址开始的内存的前 16 个字节。第三列显示十六进制数据的 ASCII 表示。对于所有没有对等的 ASCII 字符的十六进制值都会显示成句号。显示 ASCII 值以便更容易搜寻纯文本。

在所示的示例输出中，数据的第五行显示零，第六行包含星号。星号指示前一行未知次数的重复，其表示第四行之后没有数据了。

onstat -g dri 命令: 显示高可用性数据复制信息

使用 onstat -g dri 命令（单独使用，或者与 ckpt 或 que 选项一起使用），来打印当前服务器上有关高可用性数据复制统计的信息。

使用 onstat -g dri 命令打印有关 HDR 服务器状态和 HDR 相关配置参数的信息。

语法：

```
onstat   -gdri [{ ckpt  | que }]
```

onstat -g dri 的示例输出和输出描述

```
Data Replication at 0x4d676028:
  Type           State        Paired server        Last DR CKPT (id/pg)    Supports Proxy
 Writes
  primary        on           my_server                     4 / 5          NA

  DRINTERVAL   5
  DRTIMEOUT    30
  DRAUTO       3
  DRLOSTFOUND  /etc/dr.lostfound
  DRIDXAUTO    0
  Backlog      0
  Last Send    2013/12/11 16:39:48
  Last Receive 2013/12/11 16:39:48
  Last Ping    2013/12/11 16:39:44
  Last log page applied(log id,page): 4,6
```

图 158: onstat -g dri 命令输出

Type

服务器的当前类型：主服务器、辅助服务器或标准服务器

State

on 或 off

Paired server

与该服务器配对的主服务器或辅助服务器的名称

Last DR CKPT

最后检查点 ID 和页

Supports Proxy Writes
显示该服务器是否配置允许辅助服务器更新。Y = 支持辅助服务器更新，N = 不支持辅助服务器更新。
DRINTERVAL
onconfig 文件中该配置参数的值
DRTIMEOUT
onconfig 文件中该配置参数的值
DRAUTO
onconfig 文件中该配置参数的值
DRLOSTFOUND
onconfig 文件中该配置参数的值
DRIDXAUTO
onconfig 文件中该配置参数的值
Backlog
HDR 数据复制缓冲区中尚未发送到 HDR 辅助服务器的日志页数
Last Send
最后消息发送到对等节点的时间
Last Receive
从对等节点收到最后消息的时间
Last Ping
最后 ping 的时间
Last log page applied(log id,page)
最后应用的日志的日志 ID 和页号

onstat -g dri ckpt 的示例输出和输出描述

使用 onstat -g dri ckpt 命令打印 HDR 服务器中未阻塞检查点的信息。

```
Data Replication:
  Type           State        Paired server        Last DR CKPT (id/pg)   Supports Proxy
 Writes
  primary        on           BB_1                        554 / 558       Y

  DRINTERVAL   30
  DRTIMEOUT    30
  DRAUTO       0
  DRLOSTFOUND  /vobs/tristarm/sqldist/etc/dr.lostfound
  DRIDXAUTO    0

DR Checkpoint processing:
  Save State              N
  Pages Saved             0
  Save Area               none
  Received log id, page   17,68
  Saved log id, page      0,0
  Drain log id, page      0,0
  Processed log id, page  17,68
  Pending checkpoints     0
```

图 159: onstat -g dri ckpt 命令输出

Save State
B (buffering) 服务器正向暂存区添加日志
D (draining) 服务器正从暂存区删除日志

N (normal) 服务器运作正常，表示没有保存日志

Pages Saved

显示保存在暂存区尚未应用的日志页的数量

Save Area

显示暂存日志文件的位置

Received log id, page

显示从主服务器收到的最后一个日志的 ID 和页号

Processed log id, page

显示排队等待恢复管道的最后一个日志的 ID 和页号

Saved log id, page

显示存储在暂存区（如果暂存状态时 B 或 D）的最后一个日志的 ID 和页号

Drain log id, page

显示最后一个从暂存区移除的日志的 ID 和页号

Pending checkpoints

显示暂存但未应用的检查点的数量

Pending ckpt log id, page

显示任何待处理的检查点记录的位置

onstat -g dri que 的示例输出和输出描述

使用 onstat -g dri que 命令打印与几近同步的 HDR 复制相关的信息。

```
Pending Msg to Send 1
ACK QUEUE  5199:1256fff
thread 0x893de6c8 (85) 5199:1258018
thread 0x893a16b8 (83) 5199:1258048
thread 0x89229968 (72) 5199:1258078
thread 0x89381508 (82) 5199:12580a8
thread 0x87e81658 (69) 5199:12580d8
thread 0x89215968 (71) 5199:1259018
thread 0x89336bc8 (80) 5199:1259048
thread 0x89370018 (81) 5199:12590f8
thread 0x892eb018 (77) 5199:125a018
thread 0x89308018 (78) 5199:125b018
thread 0x89290138 (75) 5199:125b048
thread 0x893c1658 (84) 5199:125c018
thread 0x891fe8e8 (70) 5199:125c048
thread 0x89325018 (79) 5199:125d018
thread 0x893ff738 (86) 5199:125d048
thread 0x894207a8 (87) 5199:125d078

Applied QUEUE  5199:1251018
----------------------------------------
```

图 160: onstat -g dri que 命令输出

Pending message to send

排队到 drprsend 线程的未处理数据复制缓冲区的数量

ACK QUEUE

最新分页日志的日志唯一值、页号和值 0xfff

thread

线程控制块（TCB）的指针、括号中为该线程 ID，以及该线程执行提交的日志序列号（LSN）

Applied QUEUE

正在等待 HDR 辅助服务器接收到提交的 LSN 的应答

相关链接

DRAUTO 配置参数 在第98页

DRIDXAUTO 配置参数 在第99页

DRINTERVAL 配置参数 在第100页

DRLOSTFOUND 配置参数 在第101页

DRTIMEOUT 配置参数 在第101页

《*SinoDB* 管理员指南》: *HDR* 复制的完全同步模式

《*SinoDB* 管理员指南》: *HDR* 复制的异步模式

《*SinoDB* 管理员指南》: *HDR* 复制的接近同步模式

《*SinoDB* 管理员指南》: 将主服务器数据复制到辅助服务器

《*SinoDB SQL* 指南: 语法》: *HDR_TXN_SCOPE* 会话环境选项

onstat -g dsc 命令: 显示分布式高速缓存信息

使用 onstat -g dsc 命令显示分布式高速缓存的信息。

语法

```
onstat   -gdsc
```

示例输出

```
Data Distribution Cache:
    Number of lists         : 31
    DS_POOLSIZE             : 127

Distribution Cache Entries:

list id   ref  drop hits       heap_ptr       distribution name
-------------------------------------------------------------

0    0    0    0    300        d1567a438      testdb:informix.holding.h_t_id
0    0    0    0    6298       d0db27c38      testdb:informix.trade.th_t_id
0    0    0    0    4499       d089d6838      testdb:informix.security.s_co_id
0    0    0    0    4050       d086d9438      testdb:informix.customer.ca_c_id

1    0    0    0    900        d0c01e038      testdb:informix.compet.cp_comp_id
1    0    0    0    4049       cf99894d0      testdb:informix.customer.ca_id

3    0    0    0    900        d15674038      testdb:informix.industry.in_name
3    0    0    0    1794       d0db0f038      testdb:informix.news_xref.nx_id

4    0    0    0    1800       d12663838      testdb:informix.history.hh_h_id

5    0    0    0    900        d08cb1838      testdb:informix.watch_item.wi_id
5    0    0    0    1800       d08b95c38      testdb:informix.address.ad_zc_code

6    0    0    0    1050       d0b68d038      testdb:informix.ctaxrate.cx_c_id
6    0    0    0    1050       d0b683038      testdb:informix.taxrate.tx_id

...

Total number of distribution entries: 58
    Number of entries in use        : 0
```

图 162: onstat -g dsc 命令输出

输出描述

Number of lists
分布式高速缓存中的列表的数量

DS_POOLSIZE
可一次高速缓存的条目数

list
分布式高速缓存散列链 ID

id
散列条目数

ref
引用高速缓存条目的语句数

drop
是否在该条目添加到高速缓存后删除

hits
高速缓存条目被访问的次数

heap_ptr
用于存储该条目的堆地址

distribution name
高速缓存中分布式的名称

Total number of distribution entries
分布式高速缓存中的条目数

Number of entries in use
使用中的条目数

相关链接

DS_HASHSIZE 配置参数 在第102页
DS_POOLSIZE 配置参数 在第105页

onstat -g dsk 命令: 显示当前正在运行的压缩操作的进度

使用 onstat -g dsk 命令打印当前正在运行的压缩操作（例如：压缩、重打包和收缩）的进度信息。

语法:

```
onstat   -g dsk
```

示例输出

```
                    Processed        Remaining Duration Remaining
Partnum   OP        Rows      Blobs  Rows      Time(s)  Time(s)  Table Name
400002    Compress  6325      1752   1497      00:00:00 00:00:00 db:sl:t1
```

图 164: onstat -g dsk 命令的压缩操作输出

```
                         Processed        Remaining Duration Remaining
Partnum  OP     Pass     Rows      Blobs  Rows      Time(s)  Time(s)  Table Name
400002   Repack 1        6325      1752   1497      00:00:00 00:00:00 db:sl:t1
```

图 165: onstat -g dsk 命令的重打包操作输出

输出描述

partnum
表或分段的分区号

OP
压缩操作，例如：压缩、重打包或收缩

Pass
对于重打包操作，1 表示第一遍读取行，2 表示第二遍读取。

Processed Rows
到目前为止该指定操作已处理的行数

Blobs
操作的简单大对象的数量

Remaining Rows
要处理的剩余行数。对于重打包操作，其为当前这遍处理中剩余的行数。

Duration Time(s)
操作开始至今的时间量

Remaining Time(s)
该操作大约剩余的时间量。对于重打包操作，其为当前这遍处理所剩余的时间量。

onstat -g env 命令：打印环境变量值

使用 onstat -g env 命令显示数据库服务器当前使用的环境变量的值。

语法：
onstat -genv [{ all *variable* | *sessionidvariable* }]

可以指定以下调用之一。

调用	解释
onstat -g env	显示数据库服务器启动时环境变量的设置 不显示还未显示地设置的环境变量。
onstat -g env all	显示所有会话使用的设置 此显示与 onstat -g env 和 onstat -g env***sessionid*** 的输出相同（对所有当前会话重复）。
onstat -g env ***variable***	显示指定环境变量的缺省值 此 ***variable*** 参数使得不必将输出以管道方式运送到 grep（或某些其他实用程序）以便在多个可能设置的变量中找出某个环境变量。
onstat -g env ***sessionid***	显示特定会话使用的设置。此显示包括以下值： • 会话环境中的设置 • 由数据库服务器所指定，如 onstat -g env 所显示
onstat -g env ***sessionid variable***	显示指定会话所使用的指定环境变量的值 ***sessionid*** 和 ***variable*** 参数使得不必将输出以管道方式运送到 grep（或某些其他实用程序）以便在多个可能设置的变量中找出某个环境变量。

onstat -g env 命令显示环境变量的当前设置和每次在环境中设置此变量时的值的完整列表。例如：PDQPRIORITY 在 .informx.rc 文件中设置为 10，而在 shell 环境中设置为 55，那么 onstat -g env 命令会显示两个值。

然而，如果使用 onmode -q pdqpriority ***sessionid*** 命令更改 PDQPRIORITY，那么 onstat -g env 命令不会显示该会话的新值。onstat -g env 命令仅显示环境中环境变量集的值。它不会显示会话正在运行时修改的值。

在以下情况下，您可能想要显示环境变量的值：

- 数据库服务器实例已运行了数月，而您无法记得环境变量的设置（例如：服务器语言环境设置 SERVER_LOCALE）。
- 您想要显示环境变量的完整列表，以标识环境变量何时在多处进行了设置。
- 在这期间磁盘上的环境文件可能已更改或已丢失。
- 支持工程师想要知道特定环境变量的设置。

示例输出

以下显示 onstat -g env 命令的输出。

```
Variable            Value [values-list]
DBDATE              DMY4/
DBDELIMITER         |
DBPATH              .
DBPRINT             lp -s
DBTEMP              /tmp
INFORMIXDIR         /build2/11.50/tristarm/sqldist
                    [/build2/11.50/tristarm/sqldist]
                    [/usr/informix]
INFORMIXSERVER      parata1150
INFORMIXTERM        termcap
LANG                C
LC_COLLATE          C
LC_CTYPE            C
LC_MONETARY         C
LC_NUMERIC          C
LC_TIME             C
LD_LIBRARY_PATH     /usr/openwin/lib:/lib:/usr/lib
LKNOTIFY            yes
LOCKDOWN            no
NODEFDAC            no
NON_M6_ATTRS_OK     1
PATH                /build2/11.50/tristarm/sqldist/bin:.:
                    /root/bin:/opt/SUNWspro/bin:/usr/ccs/bin:
                    /usr/openwin/bin:/usr/sbin:/usr/bin:/usr
                    /local/bin
SERVER_LOCALE       en_US.819
SHELL               /bin/ksh
SINGLELEVEL         no
SUBQCACHESZ         10
TBCONFIG            onconfig
TERM                xterm
                    [xterm]
                    [dumb]
TERMCAP             /etc/termcap
TZ                  GB
```

图 167: onstat -g env 命令输出

onstat -g ffr 命令: 打印可用分段

使用 onstat -g ffr 命令显示有关指定会话或共享内存池的可用内存分段的信息。

此命令需要额外参数来指定要显示内存池信息的池名称或会话 ID。每个会话以其会话 ID 为名称分配一个内存池。使用 onstat -g mem 命令标识池名称，并使用 onstat -g ses 命令标识会话 ID。

语法：

onstat -gffr { *pool name* | sessionid }

示例输出

```
Free lists for pool name aio:
addr      size      idx
165dcfa0  96        10
1659cf68  152       17
165b2f20  224       26
165c7f20  224       26
1666ec38  968       79
149f2ba0  1120      84
```

图 169: onstat -g ffr aio 命令输出

输出描述

addr (hexadecimal)

池分段的内存地址

size (decimal)

池分段的大小（以字节为单位）

idx (decimal)

供内部使用。可用列表指针数组中的索引

onstat -g glo 命令: 显示全局多线程信息

使用 onstat -g glo 命令显示有关多线程、每个正在运行的虚拟处理器，和每个虚拟处理器类的累积统计的信息。此信息包括虚拟处理器的 CPU 使用信息、会话总数，以及其他多线程的全局计数。

语法：

onstat -gglo

示例输出

```
MT global info:
sessions threads  vps      lngspins time
0        23       14       0        142

          sched calls     thread switches yield 0   yield n   yield forever
total:    85240           70451           16956     868       37319
per sec:  0               0               0         0         0

Virtual processor summary:
 class       vps       usercpu   syscpu    total
 cpu         1         92.12     0.59      92.71
 aio         1         0.05      0.08      0.13
 lio         1         0.00      0.00      0.00
 pio         1         0.00      0.00      0.00
 adm         1         0.00      0.01      0.01
 soc         4         0.01      0.01      0.02
 msc         1         0.00      0.00      0.00
 jvp         1         0.00      0.00      0.00
```

```
 fifo        1         0.00      0.00      0.00
 nyevp       1         0.00      0.00      0.00
 yevp        1         0.00      0.00      0.00
 total       14        92.18     0.69      92.87

Individual virtual processors:
 vp    pid       class       usercpu   syscpu    total     Thread    Eff
 1     26328     cpu         92.12     0.59      92.71     122.65    75%
 2     26330     adm         0.00      0.01      0.01      0.00      0%
 3     26331     lio         0.00      0.00      0.00      0.00      0%
 4     26332     pio         0.00      0.00      0.00      0.00      0%
 5     26333     aio         0.05      0.08      0.13      0.28      45%
 6     26334     msc         0.00      0.00      0.00      0.19      0%
 7     26335     fifo        0.00      0.00      0.00      0.00      0%
 8     26336     nyevp       0.00      0.00      0.00      0.00      0%
 9     26337     yevp        0.00      0.00      0.00      0.00      0%
 10    26338     jvp         0.00      0.00      0.00      0.00      0%
 11    26339     soc         0.00      0.00      0.00      NA        NA
 12    26340     soc         0.00      0.00      0.00      NA        NA
 13    26341     soc         0.01      0.01      0.02      NA        NA
 14    26342     soc         0.00      0.00      0.00      NA        NA
                 tot         92.18     0.69      92.87
```

图 171: onstat -g glo 命令输出

输出描述

下表解释示例输出中全局信息部分的每个列。

表 160: 虚拟处理器摘要中列的描述

列名	描述
sessions	会话数
threads	线程总数
vps	虚拟处理器总数
lngspins	线程必须旋转超过 10,000 次以获得资源上锁存器的次数。
time	生成统计信息的秒数。服务器启动时开始统计或通过运行 onstat -z 命令重置统计信息。
sched calls	排定呼叫的总数
thread switches	从一个线程切换到另一线程的切换总次数
yield	线程让步（当线程在某些条件发生前无法继续其任务时发生）的统计信息。

下表解释示例输出中虚拟处理器摘要部分的每个列。

表 161: 虚拟处理器摘要中的列的描述

列名	描述
class	虚拟处理器的类型
vps	虚拟处理器类的实例数
usercpu	虚拟处理器类在 CPU 上运行的总用户时间（秒）

列名	描述
syscpu	虚拟处理器类在 CPU 上运行的总系统时间（秒）
total	虚拟处理器的总 CPU 时间，其为用户时间加上系统时间的总和

下表解释示例输出中各个虚拟处理器部分的每个列。

表 162: 各个虚拟处理器的列的描述

列名	描述
vp	虚拟处理器号。在 Windows™ 上，该值是线程 ID。
pid	oninit 进程的进程 ID
class	虚拟处理器的类型
usercpu	虚拟处理器在 CPU 上运行的总用户时间（秒）
syscpu	虚拟处理器在 CPU 上运行的总系统时间（秒）
total	虚拟处理器的总 CPU 时间，其为用户时间加上系统时间的总和
Thread	线程在虚拟处理器上运行的总时间
Eff	效率。总 CPU 时间占线程在虚拟处理器上运行的总时间的比率

onstat -g his 命令: 打印 SQL 跟踪信息

使用 onstat -g his 命令显示 sysmaster 数据库中 syssqltrace 表（syssqltrace、syssqltrace_info、syssqltrace_hvar 和 syssqltrace_itr）收集的 SQL 跟踪信息。

SQLTRACE 配置参数的 level 设置会影响 syssqltrace 表存储和显示的 SQL 跟踪信息，以及 onstat -g his 显示的信息。syssqltrace 表的每行描述一条先前执行的 SQL 语句。在缺省情况下，只有 DBSA 可以查看 onstat -g his 命令的 syssqltrace 信息。然而，当 UNSECURE_ONSTAT 配置参数设置为 1 时，所有用户都可以查看该信息。

语法:
onstat -ghis

示例输出

输出的内容根据跟踪设置而定。

输出中的 Statement history 部分提供有关跟踪的当前设置的信息。

```
Statement history:

Trace Level                Low
Trace Mode              Global
Number of traces          1000
Current Stmt ID              2
Trace Buffer size         2008
Duration of buffer         293 Seconds
Trace Flags          0x00001611
Control Block        0x4c2f0028
```

The following table describes this output:

信息	描述
Trace Level	跟踪的信息量。有效值为 LOW、MED、HIGH 和 OFF。
Trace Mode	执行的跟踪类型。Global 指系统上所有用户。User 仅指通过 SQL 管理 API 函数启用跟踪的用户。
Number of traces	跟踪的 SQL 语句数。该值在 onconfig 文件中设置，除非通过 SQL 管理 API 函数动态地更改 ntraces 参数。值范围从 500 到 2147483647。如果您有 100,000 个跟踪缓冲区，您的组织每秒运行 1000 条 SQL 语句并跟踪所以语句，那么这些缓冲区在被重写之前将可持续 100 秒。
Current® Stmt ID	当前 SQL 语句的 ID。每个被跟踪的语句都有一个唯一 ID。
Trace Buffer size	每个跟踪缓冲区可捕捉的数据量（以字节表示）。如果大小设置为 2 KB,而有一条 12 KB 的 SQL 语句，那么该语句将会被截断至少 10 KB。根据其他被跟踪的内容，可能更多数据会被截断。
Duration of buffer	当前跟踪缓冲区中的跟踪数据跨越的时间量（以秒表示）。它不是 sqltrace 功能已运行的时间。在上述示例中，Duration of buffer 是 293 秒，其表示跟踪的第一条与最后一条 SQL 语句之间的秒数。
Trace Flags	当前设置的 SQL 跟踪标志
Control Block	SQL 跟踪控制块的内存地址

每运行一条语句，以下信息就重复显示一次。在此示例中，调用了两个变量。

```
Statement # 2:     @ 0x4c2f3028

Database:        sysmaster
 Statement text:
  select count(*) from systables,syscolumns where systables.tabid > ? and
    systables.nrows < ?

  SELECT using tables [ systables syscolumns ]
```

下表是该输出的描述：

信息	描述
Database	数据库名称或数据库的 systables 条目的分区号
Statement text	该 SQL 语句的语句文本。如果该语句是存储过程，那么语句文本将显示过程的堆跟踪。如果语句和数字统计信息超出跟踪缓冲区，那么该语句文本可能会被截断。

```
 Iterator/Explain
 ================
 ID   Left  Right   Est Cost   Est Rows   Num Rows   Partnum   Type
```

```
  3       0       0          17          42         146    1048579   Index Scan
  4       0       0        5249        2366        2366    1048580   Seq Scan
  2       3       4        5266       99372      345436          0   Nested Join
  1       2       0           1           1           1          0   Group
```

下表描述该输出：

信息	描述
ID	SQL 迭代 ID
Left	迭代左侧输入的 ID
Right	迭代右侧输入的 ID
Est Cost	该迭代的估计成本
Est Rows	该迭代的估计行数
Num Rows	该迭代的实际行数
Partnum	表或索引的分区号
Type	操作类型

如果 SQL 语句包含一个或多个变量，并且您正在跟踪主变量，那么在输出中会显示 Host Variables 部分。

```
Host Variables
==============
1 integer    100
2 float      1000.0000000000000000
```

下表描述该输出：

信息	描述
Column 1	语句中变量的位置
Column 2	变量的数据类型
Column 3	变量的值

```
Statement information:
  Sess_id  User_id  Stmt Type  Finish Time   Run Time   TX Stamp  PDQ
  5        2053     SELECT     01:08:48      0.4247     340a6e9    0
```

下表描述该输出：

信息	描述
Sess_id	会话 ID
User_id	操作系统用户 ID
Stmt Type	SQL 语句类型
Finish Time	SQL 语句结束当天的时间

信息	描述
Run Time	用于处理语句的虚拟处理器或线程所花费的总时间量。例如：如果 Finish Time 是 1:15:00 且 Run Time 是 9 分钟，但开始时间不一定是 1:06:00，因可能有多个虚拟处理器或线程并行处理语句的各个部分。
TX Stamp	在该事务中记录 BEGIN WORK 语句的时间
PDQ	SQL 语句的 PDQ 级别

输出中的 Statement Statistics 部分提供有关该语句的特定信息。

```
Statement Statistics:
Page        Buffer      Read        Buffer      Page        Buffer      Write
Read        Read        % Cache     IDX Read    Write       Write       % Cache
1285        19444       93.39       0           810         17046       95.25

Lock        Lock        LK Wait     Log         Num         Disk        Memory
Requests    Waits       Time (S)    Space       Sorts       Sorts       Sorts
10603       0           0.0000      60.4 KB     0           0           0

Total       Total       Avg         Max         Avg         I/O Wait    Avg Rows
Executions  Time (S)    Time (S)    Time (S)    IO Wait     Time (S)    Per Sec
1           30.8660     30.8660     30.8660     0.0141      29.2329     169.8959

Estimated   Estimated   Actual      SQL         ISAM        Isolation   SQL
Cost        Rows        Rows        Error       Error       Level       Memory
102         1376        5244        0           0           CR          32608
```

信息	描述
Page Read	该 SQL 语句从磁盘读取的页数
Buffer Read	该 SQL 语句从缓冲池读取而不是从磁盘读取页的次数
Read % Cache	从缓冲池读取页的次数的百分比
Buffer IDX Read	目前尚未实施
Page Write	写入磁盘的页数
Buffer Write	修改并送回缓冲池的页数
Write % Cache	页写入缓冲池而不是写入磁盘的次数的百分比
Lock Requests	该语句所需的锁的总数
Lock Waits	该 SQL 语句等待锁的次数
LK Wait Time (S)	该语句等待锁的时间量（以秒表示）
Log Space	该 SQL 语句在逻辑日志中使用的存储空间量
Num Sorts	用于执行该语句的总排序数
Disk Sorts	需要磁盘空间来执行该 SQL 语句排序的排序数
Memory Sorts	完全在内存中执行该 SQL 语句排序的排序数
Total Executions	该语句已执行的总次数，或该游标重使用的次数

信息	描述
Total Time (S)	执行该语句的总时间（以秒为单位）
Avg Time (S)	执行该语句所需的平均时间（以秒为单位）
Max Time (S)	运行该 SQL 语句的总时间（以秒为单位），不包含应用程序使用的任何时间。如果准备查询然后运行该查询 5 次，每次查询运行时都将跟踪添加到跟踪缓冲区，那么 Max Time 是任一次执行所花费的最大时间。
Avg IO Wait	该语句等待 I/O 的平均时间量，不包括任何异步I/O。
I/O Wait Time (S)	该语句等待 I/O 的时间量，不包括任何异步 I/O（以秒为单位）
Avg Rows Per Sec	该语句每秒生成的平均行数
Estimated Cost	与该 SQL语句相关联的查询优化程序成本
Estimated Rows	由查询优化程序估计的该语句返回的行数
Actual Rows	该语句返回的行数
SQL Error	SQL 错误号
ISAM Error	RSAM 或 ISAM 错误号
Isolation Level	该语句运行使用的隔离级别
SQL Memory	该 SQL 语句需要的字节数

有关系统监视接口表 syssqltrace 的完整结构，请参阅 *syssqltrace* 在第235页。

有关设置 SQLTRACE 配置参数的详细信息，请参阅 *SQLTRACE* 配置参数 在第177页。

相关链接

SQLTRACE 配置参数 在第177页

onstat -g ioa 命令: 打印合并的 onstat -g 信息

使用 onstat -g ioa 命令显示来自 onstat -g iob、onstat -g iof、onstat -g ioq 和 onstat -g iov 命令的合并信息。

语法

```
onstat   -gioa
```

示例输出

```
AIO global info:
   9 aio classes
   9 open files
  64 max global files

AIO I/O queues:
q name/id    len maxlen totalops  dskread dskwrite  dskcopy
 fifo    0    0      0        0        0        0        0
drda_dbg 0    0      0        0        0        0        0
sqli_dbg 0    0      0        0        0        0        0
  adt    0    0      0        0        0        0        0
  msc    0    0      1      231        0        0        0
  aio    0    0      5    13069    10895        0        0
  pio    0    0      1     1580        0     1580        0
```

```
  lio    0    0      1    37900        0    37900        0
  gfd    3    0     87    42115    15806    26309        0
  gfd    4    0      4        5        1        4        0
  gfd    5    0     12       35       22       13        0
  gfd    6    0     11       33       21       12        0
  gfd    7    0      1        4        3        1        0
  gfd    8    0      1        4        3        1        0

AIO I/O vps:
class/vp/id s  io/s totalops dskread dskwrite dskcopy wakeups io/wup errors tempops
 fifo  7  0 i   0.0        0       0        0       0       1  0.0      0        0
  msc  6  0 i   0.0      231       0        0       0     221  1.0      0      231
  aio  5  0 i   0.0    39285   26358    10793       0   37531  1.0      0        5
  aio  9  1 i   0.0     5770    3795     1944       0    5926  1.0      0        0
  aio 10  2 i   0.0     2308     717     1585       0    1953  1.2      0        0
  aio 11  3 i   0.0     1463     166     1295       0    1166  1.3      0        0
  aio 12  4 i   0.0     1219      46     1172       0     943  1.3      0        0
  aio 13  5 i   0.0     1041      34     1007       0     805  1.3      0        0
  aio 15  6 i   0.0      425       2      423       0     438  1.0      0        0
  aio 16  7 i   0.0      342       5      337       0     395  0.9      0        0
  pio  4  0 i   0.0     1580       0     1580       0    1581  1.0      0     1580
  lio  3  0 i   0.0    37900       0    37900       0   29940  1.3      0    37900

AIO global files:
gfd pathname         bytes read     page reads  bytes write    page writes io/s
3   ./rootdbs        85456896       41727       207394816      101267      572.9
 op type     count          avg. time
 seeks       0              N/A
 reads       13975          0.0015
 writes      51815          0.0018
 kaio_reads  0              N/A
 kaio_writes 0              N/A

4   tempsbs.chunk    2048           1           8192           4           113.6
 op type     count          avg. time
 seeks       0              N/A
 reads       1              0.0131
 writes      3              0.0074
 kaio_reads  0              N/A
 kaio_writes 0              N/A

5   sbs1.chunk       45056          22          26624          13          173.4
 op type     count          avg. time
 seeks       0              N/A
 reads       22             0.0063
 writes      6              0.0038
 kaio_reads  0              N/A
 kaio_writes 0              N/A

6   sbs2.chunk       43008          21          24576          12          76.1
 op type     count          avg. time
 seeks       0              N/A
 reads       21             0.0148
 writes      6              0.0072
 kaio_reads  0              N/A
 kaio_writes 0              N/A

7   qhdr.chunk       6144           3           2048           1           550.5
 op type     count          avg. time
 seeks       0              N/A
 reads       3              0.0019
 writes      1              0.0016
 kaio_reads  0              N/A
 kaio_writes 0              N/A
```

```
8   ./dbs1          6144          3          2048          1          403.0
 op type     count          avg. time
 seeks       0              N/A
 reads       3              0.0027
 writes      1              0.0018
 kaio_reads  0              N/A
 kaio_writes 0              N/A


AIO big buffer usage summary:
class                reads                                    writes
       pages    ops  pgs/op  holes  hl-ops hls/op      pages    ops  pgs/op
fifo       0      0   0.00      0       0   0.00          0      0   0.00
drda_dbg   0      0   0.00      0       0   0.00          0      0   0.00
sqli_dbg   0      0   0.00      0       0   0.00          0      0   0.00
 kio       0      0   0.00      0       0   0.00          0      0   0.00
 adt       0      0   0.00      0       0   0.00          0      0   0.00
 msc       0      0   0.00      0       0   0.00          0      0   0.00
 aio  228709  20228  11.31   1005     203   4.95     213272  18556  11.49
 pio       0      0   0.00      0       0   0.00      19672   1580  12.45
 lio       0      0   0.00      0       0   0.00      55287  37900   1.46
```

输出描述

有关每个输出列的描述，请分别参阅 *onstat -g iob* 命令: 打印大缓冲区的使用摘要 在第469页、*onstat -g ioq* 命令: 显示 *I/O* 队列信息 在第471页 和 *onstat -g iov* 命令: 显示 *AIO VP* 统计信息 在第472页 命令。

onstat -g iob 命令: 打印大缓冲区的使用摘要

使用 onstat -g iob 命令显示大缓冲区的使用摘要。

语法:

```
onstat   -giob
```

示例输出

```
AIO big buffer usage summary:
                reads                                    writes
       pages    ops  pgs/op  holes  hl-ops hls/op      pages    ops  pgs/op
fifo       0      0   0.00      0       0   0.00          0      0   0.00
 kio       0      0   0.00      0       0   0.00          0      0   0.00
 adt       0      0   0.00      0       0   0.00          0      0   0.00
 msc       0      0   0.00      0       0   0.00          0      0   0.00
 aio       0      0   0.00      0       0   0.00        607    607   1.00
 pio       0      0   0.00      0       0   0.00          0      0   0.00
 lio       0      0   0.00      0       0   0.00          0      0   0.00
```

图 175: onstat -g iob 命令输出

onstat -g iof 命令: 显示异步 I/O 统计信息

使用 onstat -g iof 命令显示按块或文件的异步 I/O 统计信息。

此命令与 onstat -D 命令相类似，除了 onstat -g iof 还显示非块文件的信息。它包含临时文件和排序工作文件的信息。

语法:

```
onstat  -giof
```

示例输出

```
AIO global files:
gfd pathname      bytes read   page reads   bytes write   page writes  io/s
3   rootdbs       1918976      937          145061888     70831        36.5

      op type     count         avg. time
      seeks       0             N/A
      reads       937           0.0010
      writes      4088          0.0335
      kaio_reads  0             N/A
      kaio_writes 0             N/A
```

图 177: onstat -g iof 命令输出

输出描述

gfd
该块或文件的全局文件描述符编号

pathname
块或文件的路径名

bytes read
对块或文件执行的字节读取的次数

page reads
对块或文件执行的页读取的次数

bytes write
对块或文件执行的字节写入的次数

page writes
对块或文件执行的页写入的次数

io/s
每秒可执行的 I/O 操作数。该值代表块或文件的 I/O 性能

op type
操作的类型

count
此操作发生的次数

avg time
该操作完成所花的平均时间

onstat -g iog 命令: 打印 AIO 全局信息

使用 onstat -g iog 命令显示 AIO 的全局信息。
语法:

```
onstat   -giog
```

示例输出

```
AIO global info:
   8 aio es
   5 open files
  64 max global files
```

图 179: onstat -g iog 命令输出

onstat -g ioq 命令: 显示 I/O 队列信息

使用 onstat -g ioq 命令显示有关由 I/O 队列执行的操作的类型和数量的统计信息。

语法:

onstat -gioq [*queue_name*]

如果给定 *queue_name*，那么只显示具有该名称的队列。如果没有给定 *queue_name*，那么显示所有队列的信息。

示例输出

```
AIO I/O queues:
q name/id    len maxlen totalops  dskread dskwrite  dskcopy
sqli_dbg   0      0      0        0        0        0        0
 fifo   0      0      0        0        0        0        0
  adt   0      0      0        0        0        0        0
  msc   0      0      1      537        0        0        0
  aio   0      0      3     6537      238     5777        0
  pio   0      0      2     1103        0     1102        0
  lio   0      0      2    11795        0    11794        0
  gfd   3      0     17    17489     1526    15963        0
  gfd   4      0     17    18347     2384    15963        0
  gfd   5      0     16      220       41      179        0
  gfd   6      0      4        4        0        4        0
  gfd   7      0      4        4        0        4        0
  gfd   8      0      4        4        0        4        0
  gfd   9      0      9       54       24       30        0
  gfd  10      0     16      149       40      109        0
  gfd  11      0     16      621      128      493        0
  gfd  12      0     16     1953     1146      807        0
  gfd  13      0     16      409       71      338        0
  gfd  14      0     16      378       60      318        0
```

图 181: onstat -g ioq 命令输出

输出描述

q name/id

I/O 队列的名称和编号。名称指示队列的类型。编号用来区分具有相同名称的队列。

以下是可能的队列名称列表和每种类型的队列处理的内容:

sqli_dbg

处理星瑞格®技术支持的 SQL 接口调试功能的 I/O

fifo

处理 FIFO VP 的 I/O

adt

处理审计 I/O

msc

处理杂项 I/O

aio

处理 SinoDB® 异步 I/O

kio

处理内核 AIO

pio

处理物理日志记录 I/O

lio
处理逻辑日志记录 I/O
gfd
全局文件描述符 - 为每个主块和镜像块分配单独的全局文件描述符。每个 gfd 队列的使用取决于 kaio 是否开启和相关联的块是格式化的还是原始的。

len
队列中暂挂 I/O 请求数

maxlen
队列中同时存在的 I/O 请求的最大数量

totalops
队列中已完成的 I/O 操作的总数

dskread
队列中已完成的读操作的总数

dskwrite
队列中已完成的写操作的总数

dskcopy
队列中已完成的复制操作的总数

onstat -g ipl 命令: 打印索引页日志记录状态信息

使用 onstat -g ipl 命令显示有关索引页日志记录状态的信息。
语法:
onstat -gipl

示例输出

```
Index page logging status: Enabled
Index page logging was enabled at: 2008/12/20 16:01:02
```

图 183: onstat -g ipl 命令输出

输出描述

Index page logging status
索引页日志记录状态：启用或禁用

Index page logging was enabled at
索引页日志记录启用的日期和时间

onstat -g iov 命令: 显示 AIO VP 统计信息

使用 onstat -g iov 命令显示每个虚拟处理器的异步 I/O 统计信息。
语法:
onstat -giov

示例输出

```
AIO I/O vps:
class/vp/id  s  io/s  totalops  dskread dskwrite  dskcopy  wakeups  io/wup  errors tempops
fifo  7  0   i   0.0       0        0         0         0        1     0.0       0       0
msc   6  0   i   0.1    9988        0         0         0     7833     1.3       0    9988
aio   5  0   i   0.0    4894     3341      1426         0     4393     1.1       0       0
aio   9  1   i   0.0      41        0        41         0       33     1.2       0       0
```

```
pio   4  0   i   0.0     199       0      199          0       200     1.0       0     199
lio   3  0   i   0.0    6344       0     6344          0      6344     1.0       0    6344
```

图 185: onstat -g iov 命令输出

输出描述

class
  虚拟处理器类

vp
  虚拟处理器在类中的 ID 编号

s
  AIO 虚拟处理器的当前状态

  f
    派生

  i
    空闲

  s
    搜索

  b
    正忙

  o
    打开

  c
    关闭

io/s
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），虚拟处理器的平均 I/O 速度（以每秒的操作数衡量）。

totalops
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），虚拟处理器执行的 I/O 操作总数。

dskread
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），虚拟处理器执行的读操作的总数。

dskwrite
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），虚拟处理器执行的写操作的总数

dskcopy
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），虚拟处理器执行的复制操作的总数。

wakeups
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），AIO VP 处于空闲状态的次数。

io/wup
  自数据库服务器启动以来或自 onstat -z 命令上次运行以来（看这两者哪个发生在后），AIO VP 每次唤醒执行的 I/O 操作的平均数量

errors
  KAIO 耗尽资源的错误总数

tempops（十进制）

仅内部使用。用于确定何时添加新的 AIO VP 的 I/O 操作计数。仅当启用了 AUTO_AIOVPS 配置参数时才应用。

onstat -g lap 命令：显示轻量追加状态信息

使用 onstat -g lap 命令显示有关系统中发生的轻量追加的状态信息。

语法：

```
onstat   -glap
```

示例输出

```
Light Append Info
session id  address   cur_ppage   la_npused   la_ndata    la_nrows    bufcnt
31          b60a5e8   ffbff494    2938        2937        93990       4
```

图 187: onstat -g lap 命令输出

输出描述

Session id (decimal)

执行轻量追加操作的会话 ID

address (hexadecimal)

轻量追加缓冲区的地址

cur_ppage (hexadecimal)

当前物理页地址

la_npused (decimal)

已分配的页数

la_ndata (decimal)

追加的数据页数

la_nrows (decimal)

追加的行数

bufcnt (decimal)

轻量追加缓冲区数

onstat -g laq 命令：打印辅助服务器队列

使用 onstat -g laq 命令打印有关应用从主服务器接收的日志信息的辅助服务器上的队列的信息。

在高可用性集群中，主服务器通过网络向一个或多个辅助服务器发送日志记录。每个辅助服务器连续重播来自主服务器的事务日志，以确保数据复制到辅助服务器上。主服务器上每个 表空间 都在辅助服务器上分配一个队列，用于接收日志记录。称作 *apply thread* 的线程将队列中的日志记录应用到辅助服务器上。按照日志接收顺序来应用。

使用 onstat -g laq 命令监视辅助服务器上队列的性能。如果怀疑由于辅助服务器上日志重播不够快导致主服务器性能降低，那么可使用此命令。Avg Depth (average depth) 列指示上次队列插入操作发生时队列中日志的平均数量。

onstat -g laq 命令仅在辅助服务器上有效。在主服务器上运行该命令，只会返回 onstat 标题输出。

语法：

```
onstat   -g  laq
```

示例输出

```
Log Apply Info:
Thread          Queue    Total      Avg
                 Size   Queued    Depth
xchg_1.0            0        9     5.00
xchg_1.1            0        0     0.00
xchg_1.2            0        7     3.43
xchg_1.3            0        8     1.12
xchg_1.4            0        4     1.50
xchg_1.5            0        4     1.50
xchg_1.6            0        6     1.33
xchg_1.7            0       47     3.11
xchg_1.8            0       37     9.46
xchg_1.9            0       13     3.69

Secondary Apply Queue:         Total Buffers:12 Size:64K Free Buffers:11
Log Recovery Queue:            Total Buffers:4 Size:32K Free Buffers:2
Log Page Queue:                Total Buffers:32 Size:2K Free Buffers:32
Log Record Queue:              Total Buffers:50 Size:64K Free Buffers:49
```

图 189: onstat -g laq 命令输出

输出描述

Thread

指定接收日志记录的应用线程的名称

Queue Size

排队等待应用线程的日志记录数

Total Queued

队列中等待给定应用线程的日志记录的总数

Avg Depth

上次队列插入操作发生时队列中日志的平均数量

Secondary Apply Queue

辅助应用队列接收来自主服务器的日志缓冲区。显示的值代表分配接收日志缓冲区记录的缓冲区总数、队列大小，以及未使用缓冲区的数量。

Log Recovery Queue

日志恢复队列接收来自辅助应用队列的输出。日志缓冲区转换成兼容 ontape 实用程序的格式。显示的值代表恢复队列中流缓冲区总数、流缓冲区的大小，以及未使用缓冲区的数量。

Log Page Queue

日志页队列接收来自日志恢复队列的输出。移除 ontape 格式并将数据分成单独日志页。显示的值代表队列中日志页总数、队列的大小，以及未使用缓冲区的数量。

Log Record Queue

日志记录队列接收来自日志页队列的输出。日志页被分成单独日志记录。显示的值代表恢复队列中日志记录总数、队列的大小，以及未使用缓冲区的数量。

onstat -g lmm 命令: 显示低内存管理信息

使用 onstat -g lmm 命令显示有关自动低内存管理设置及其近期活动的信息。

语法:

```
onstat   -g lmm
```

示例输出

```
Low Memory Manager
```

```
Control Block      0x4cfca220
Memory Limit       300000 KB
Used               149952 KB
Start Threshold    10240 KB
Stop Threshold     10 MB
Idle Time          300 Sec
Internal Task      Yes
Task Name          'Low Memory Manager'
Low Mem TID        0x4cfd7178
# Extra Segments   0

Low Memory Manager Tasks

Task                Count    Last Run
Kill User Sessions  267      04/04/2011.16:57
Kill All Sessions   1        04/04/2011.16:58
Reconfig(reduce)    1        04/04/2011.16:59
Reconfig(restore)   1        04/04/2011.17:59

Last 20 Sessions Killed

Ses ID Username Hostname PID   Time
194    sfisher  host01   13433 04/04/2011.16:57
201    sfisher  host01   13394 04/04/2011.16:57
198    sfisher  host01   13419 04/04/2011.16:57
190    sfisher  host01   13402 04/04/2011.16:57
199    sfisher  host01   13431 04/04/2011.16:57

Total Killed 177
```

图 191: onstat -g lmm 命令输出

输出描述

Control Block
　自动低内存管理的内部控制结构的地址

Memory Limit
　服务器尝试保持的内存量

Used
　服务器当前使用的内存量

Start Threshold
　自动低内存管理的启动阈值

Stop Threshold
　自动低内存管理的停止阈值

Idle Time
　自动低内存管理认为会话为空闲的时间量

Internal Task
　Yes = 使用 SinoDB® 程序
　No = 使用用户自定义程序

Task Name
　用户自定义程序的名称

Low Mem TID
　自动低内存管理线程的地址

Task
Kill = 自动进程运行并终止会话
Reconfig(reduce) = 自动进程运行并释放未使用内存的区块
Reconfig(restore) = 自动进程运行并复原服务和配置
Count
任务运行的次数
Last Run
上次运行任务的日期和时间
Ses ID
（使用 onmode -z 命令）终止的会话的 ID
Username
会话的所有者的用户名称
Hostname
启动会话的主机的名称
PID
进程 ID
Time
会话终止的日期和时间

使用 LOW_MEMORY_MGR 配置参数启用自动低内存管理。

相关链接

scheduler lmm enable 参数: 指定自动低内存管理设置 *(SQL* 管理 *API)* 在第677页
scheduler lmm disable 参数: 停止自动低内存管理 *(SQL* 管理 *API)* 在第679页
LOW_MEMORY_MGR 配置参数 在第127页
《*SinoDB* 管理员指南》: 为关键活动保留内存

onstat -g lmx 命令: 打印所有锁定的互斥

使用 onstat -g lmx 命令显示有关所有锁定的互斥的信息。
语法:

```
onstat   -glmx
```

示例输出

```
Locked mutexes:
mid      addr             name             holder   lkcnt  waiter   waittime
119006   7000001e684b928  td_mutex         298      0
134825   7000002043a9148  free_lock        11009    0      200      22921
                                                             11010    22918
587817   70000022ddb3268  sync_lock1       200      0
593614   700000239ce7b68  SB_LTH_LATCH     875      0

Number of mutexes on VP free lists: 49
```

图 193: onstat -g lmx 命令输出

Output description

mid
内部互斥标识符

addr
锁定的互斥的地址

name
互斥的名称

holder
持有互斥的线程的线程 ID
0 = 共享模式下持有的读/写互斥

lkcnt
对于读/写互斥，是当前在共享模式下锁定互斥的线程数。对于可重新锁定互斥，是互斥被持有互斥的线程锁定或重新锁定的次数。

waiter
正等待该互斥的线程的 ID 的列表

waittime
线程等待的时间量（以秒为单位）

onstat -g lsc 命令：打印活动的轻扫描状态（弃用）

onstat -g lsc 命令已被 onstat -g scn 命令取代。
语法：
onstat -glsc

示例输出

```
Light Scan Info
descriptor  address           next_lpage  next_ppage        ppage_left  bufcnt  look_aside
3           474b74b0          4a0         7e2c80            416         1           N
```

图 195: onstat -g lsc 命令输出

输出描述

descriptor (decimal)
轻扫描 ID

address (hex)
轻扫描描述符的内存地址

next_lpage (hex)
要扫描的下一个逻辑页地址

next_ppage (hex)
要扫描的下一个物理页地址

ppage_left (decimal)
当前扩展数据块中剩余要扫描的物理页数

#bufcnt (decimal)
用于该轻扫描的清扫描缓冲区数

#look_aside (char)
该轻扫描是否需要观察周边（Y = 是，N = 否）。当线程需要检查缓冲池中现有页以获得正在进行轻扫描的页的最新映像时发生观察周边操作。

使用 onstat -g scn 命令基于在压缩表上、在行大于一页的表上，和在具有 VARCHAR、LVARCHAR 和 NVARCHAR 数据的表上的扫描行显示当前扫描的状态。有关更多的信息，请参阅 *onstat -g scn* 命令: 显示扫描信息 在第506页。

onstat -g mem 命令: 显示池内存统计信息

使用 onstat -g mem 命令显示池的内存统计信息。

如果您运行从 PER_STMT_EXEC 和 PER_STMT_PREP 内存持续池分配内存的 SQL 查询，那么 onstat -g mem 命令将显示有关 PRP.sessionid.threadid 池和 EXE.sessionid.threadid 池的信息。

语法：
onstat -gmem [*pool namesession id*]

会话池以会话编号命名。如果没有给参数，那么显示所有池的信息。

示例输出

```
Pool Summary:
name                 addr             totalsize freesize #allocfrag #freefrag
resident     R       10a001028        2420736   7960     2          2
res-buff     R       10a250028        8269824   7960     2          2
global       V       10aac0028        9351168   32648    650        11
...
...
...
onmode_mon   V       10b983028        20480     2752     108        1
13           V       10bd5d028        16384     5200     12         2
Blkpool Summary:
name                 addr             size      #blks      pre-hint   szavail|
global       V       10aac8920        0         0          0          0
xmf_msc_pl   V       10ac84ca0        954368    73         0          0
```

图 197: onstat -g mem 命令输出

输出描述

name
池名称

class
用于创建池的共享内存段类型

addr
池内存地址

totalsize
池的大小，以字节为单位

freesize
池中可用内存量

#allocfrag
池中已分配的分段

#freefrag
池中可用分段

name
池名称

class
用于创建池的共享内存段类型

addr
池内存地址

size
池的大小，以字节为单位

#blks
池中的块数

相关链接

《SinoDB DataBlade API 程序员指南》: PER_STMT_EXEC 内存持续期间

《SinoDB DataBlade API 程序员指南》: PER_STMT_PREP 内存持续期间

onstat -g mgm 命令: 显示 MGM 资源信息

使用 onstat -g mgm 命令显示有关内存分配管理器（MGM）的资源信息。

可以使用 onstat -g mgm 命令监视 MGM 如何协调内存使用和扫描线程。此命令读取共享内存结构并提供在命令执行时的精确统计信息。

语法：

```
onstat   -gmgm
```

onstat -g mgm 输出显示称作量子的内存单位。内存量子代表内存的单位，如下所示：

```
memory quantum = DS_TOTAL_MEMORY / DS_MAX_QUERIES
```

以下算式显示 onstat -g mgm 输出的值的内存量子：

```
内存量子 = 4000 KB / 31
                = 129 KB
```

数据库服务器分配内存时根据需要调整量子的值。因此，onstat -g mgm 命令显示的量子的值并非一直精确。

scan thread quantum 总是等于 1。

示例输出

```
Memory Grant Manager (MGM)
--------------------------
MAX_PDQPRIORITY:  100
DS_MAX_QUERIES:    31
DS_MAX_SCANS:      1048576
DS_NONPDQ_QUERY_MEM: 128 KB
DS_TOTAL_MEMORY:   4000 KB

Queries:   Active     Ready   Maximum
                        0         0        31
Memory:     Total      Free   Quantum
(KB)         4000      4000       128

Scans:      Total      Free   Quantum
               1048576   1048576         1

Load Control:     (Memory)       (Scans)  (Priority)  (Max Queries)    (Reinit)
                        Gate 1       Gate 2      Gate 3         Gate 4     Gate 5
(Queue Length)            0            0           0              0          0

Active Queries:  None
Ready Queries:  None
Free Resource        Average #        Minimum #
--------------     ---------------     ---------
Memory                0.0 +- 0.0           500
```

```
Scans                0.0 +- 0.0          1048576

Queries              Average #        Maximum #    Total #
--------------    ---------------     ---------    -------
Active               0.0 +- 0.0             0          0
Ready                0.0 +- 0.0             0          0

Resource/Lock Cycle Prevention count:  0
```

图 199: onstat -g mgm 命令输出

输出描述

输出的第一部分显示 PDQ 配置参数的值。

输出的第二部分描述 MGM 内部控制信息。其包含四组信息。第一组是 Queries：

Active
  当前正在运行的 PDQ 查询数

Ready
  已准备好运行但数据库服务器由于装入控制原因而延迟执行的用户查询的数量

Maximum
  数据库服务器允许处于活动状态的查询的最大数量。反映 DS_MAX_QUERIES 配置参数的当前值。

下一组是 Memory：

Total
  PDQ 查询可使用的内存量，以 KB 为单位（DS_TOTAL_MEMORY 指定此值）

Free
  PDQ 查询可用内存中当前未使用的内存量，以 KB 为单位

Quantum
  内存量子中大约的内存量，以 KB 表示

下一组是 Scans：

Total
  DS_MAX_SCANS 配置参数指定的扫描线程总数

Free
  当前可用于决策支持查询的扫描线程的数量

Quantum
  扫描线程量子中的扫描线程的数量

该部分输出的最后一组描述 MGM Load Control：

Memory
  正在等待内存的查询的数量

Scans
  正等待扫描的查询的数量

Priority
  正等待具有更高 PDQ 优先级的查询运行的查询的数量

Max Queries
  正等待查询槽的查询的数量

Reinit
  在 onmode -M 或 -Q 命令之后，正等待正在运行的查询完成的查询的数量

输出的下一部分 Active Queries 描述 MGM 的活动队列与就绪队列。这部分输出显示正在每个入口等待的查询的数量：

Session
启动查询的会话的会话 ID

Query
与查询关联的内部控制块的地址

Priority
分配给查询的 PDQ 优先级

Thread
向 MGM 注册查询的线程

Memory
当前分配给查询的内存量或保留给查询的内存量（单位是 MGM 页，即 8 KB。）

Scans
查询当前使用的扫描线程的数量或分配给查询的扫描线程的数量

Gate
查询正在该处等待的入口编号

输出的下一部分 Free Resource 提供 MGM 可用资源的统计信息。此部分和最后部分的数字反映系统初始化以来或在上次 onmode -Q、-M 或 -S 命令以来的统计信息。这部分输出包含以下信息：

Average
平均内存量和扫描数

Minimum
最小可用内存量和扫描数

输出的下一部分 Queries 提供 MGM 查询的统计信息：

Average
活动的和就绪的队列的平均长度

Maximum
活动的和就绪的队列的最大长度

Total
活动的和就绪的队列的总长度

Resource/Lock Cycle Prevention count
系统为避免潜在的死锁而立即激活查询的次数。（数据库服务器可以检测到如果队列中某些查询没有立即运行的话，其可能何时会造成死锁。）

相关链接

DS_MAX_QUERIES 配置参数 在第102页
DS_MAX_SCANS 配置参数 在第103页
MAX_PDQPRIORITY 配置参数 在第131页
DS_NONPDQ_QUERY_MEM 配置参数 在第104页
DS_TOTAL_MEMORY 配置参数 在第106页

onstat -g nbm 命令：打印块位图

使用 onstat -g nbm 命令显示非常驻段的块位图。

位图的每个位代表一个 4 KB 块。如果该块被使用，那么该位设置为 1。如果块是空闲的，那么该位设置为 0。位图以一系列十六进制数字显示。位和块都是从 0 开始编号，因此，第一块是块 0，第二块是块 1，依此类推。

语法：
onstat -gnbm

示例输出

此示例显示位于 0x10CC00000 的虚拟内存段的位图。位图本身位于 0x10CC00290。该段的所有 1792 块都可用，除了块 0 和块 1023。

```
Block bitmap for virtual segment address 0x10cc00000:
 address = 0x10cc00290, size(bits) = 1792
 used = 1, largest_free = -1
      0:8000000000000000 0000000000000000 0000000000000000 0000000000000000
    256:0000000000000000 0000000000000000 0000000000000000 0000000000000000
    512:0000000000000000 0000000000000000 0000000000000000 0000000000000000
    768:0000000000000000 0000000000000000 0000000000000000 0000000000000001
   1024:0000000000000000 0000000000000000 0000000000000000 0000000000000000
   1280:0000000000000000 0000000000000000 0000000000000000 0000000000000000
   1536:0000000000000000 0000000000000000 0000000000000000 0000000000000000
```

图 201: onstat -g nbm 命令输出

输出描述

address
位图的起始地址

size
位图中位的数量。这也是内存段中4 KB 块的数量。

used
位图中设置为 1 的位的总数。这也是内存段中的使用中的 4 KB 块的数量。

largest free
如果这是 -1 以外的值，那么它是连续可用位的最大数量。它也是内存段中最大邻近块集合的 4 KB 块的数量。

值 -1 表示还未计算的最大可用空间。数据库服务器只在尝试分配从 *lastalloc* 块开始的块集合却没有足够可用空间时计算最大可用空间。一旦在段中分配了另一块，这个值就再次设置为 -1。

onstat -g nsc 命令：打印当前共享内存连接信息

使用 onstat -g nsc 命令显示有关所有当前连接或指定连接 ID 的共享内存连接的信息。
语法：
onstat -gnsc [*client_id*]

如果没有提供 *client_id*，那么显示连接到数据库服务器的所有当前共享内存连接的信息。如果提供 *client_id*，那么此命令显示有关该 ID 的共享内存连接的更多详细信息。

示例输出

这是不带有 *client_id* 的 onstat -g nsc 输出。其显示当前仅一位用户通过共享内存连接到数据库服务器。该连接的 ID 是 0。

```
  clientid  clientPID      state #serverbufs #clientbufs    #rdwrts
         0       6031  Connected           4           4         12
```

图 203: onstat -g nsc 命令输出

此示例显示以 *client_id* 为 0 运行命令的输出。

```
Network Shared Memory Status for Client: 0

    clientid    clientPID       state  #serverbufs  #clientbufs      #rdwrts
           0        18949   Connected            4            4       447048

     needbuf        segid        semid       semnum     be_semid    be_semnum
           0         1303       851969            0       851969           10

  be_curread  be_curwrite   fe_curread  fe_curwrite
          -1            1            0            2

 be_nextread be_nextwrite  fe_nextread fe_nextwrite
           2            2            4            3

readyqueue
  -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1

     Server Buffers                            Client Buffers
  i: bufid     status   offset  fe_addr         bufid     status   offset  fe_addr
  0:     4      inuse     4474   804474             0      avail     3424   803424
  1:     5      inuse     4888   804888             1      avail     3838   803838
  2:     6      avail     4c9c   804c9c             2      inuse     3c4c   803c4c
  3:     7      avail     50b0   8050b0             3      avail     4060   804060
  4:    -1       free        0        0            -1       free        0        0
  5:    -1       free        0        0            -1       free        0        0
```

图 204: 带有 client id 的 onstat -g nsc 命令输出

输出描述

clientid
  服务器分配的 ID

clientPID
  客户端进程 ID

state
  连接状态

  Connected
    客户端与服务器已建立连接

  Con1
    服务器已成功建立与客户端的连接，但客户端尚未得到通知

  Waiting
    服务器正处于与客户端建立连接的过程中

  Reject
    服务器已拒绝客户端连接，通常因为服务器正在关闭或还未处于联机模式

  Closed
    服务器关闭与客户端的连接。客户端可能还不知道这个情况

  Not connected
    服务器正在初始化连接的内部结构

  Unknown
    连接已关闭且客户端知道这个情况。服务器正在清除内部结构。

#serverbufs
  当前分配的数据库服务器缓冲区

#clientbufs
当前分配的客户端缓冲区
#rdwrts
自该连接创建以来通过该连接所执行的读和写的总数

仅当使用 *client_id* 运行 onstat -g nsc 命令时，以下项才显示在输出中：

needbuf
指示服务器是否正在等待缓冲区被释放
0
否
1
是
segid
共享内存段 ID
semid
信号量 ID
semnum
信号量 ID 中的信号量编号
be_semid
后端信号量 ID
be_semnum
信号量 ID 中后端信号量编号
be_curread
正读取的后端缓冲区的 ID
be_curwrite
正写入的后端缓冲区的 ID
fe_curread
正读取的前端缓冲区的 ID
fe_currwrite
正写入的前端缓冲区的 ID
be_nextread
要读取的下一个后端缓冲区的 ID
be_nextwrite
要写入的下一个后端缓冲区的 ID
fe_nextread
要读取的下一个前端缓冲区的 ID
fe_nextwrite
要写入的下一个前端缓冲区的 ID
readyqueue
共享内存缓冲区标识的队列
i
消息缓冲区的内部位置键
bufid
消息缓冲区 ID

status
消息缓冲区的状态

offset
共享内存段中内存缓冲区的偏移量

fe_addr
消息缓冲区的前端地址

相关链接
NETTYPE 配置参数 在第135页

onstat -g nsd 命令：打印轮询线程的共享内存数据

使用 onstat -g nsd 命令显示有关轮询线程的共享内存数据的信息。
语法：
onstat -gnsd

示例输出

```
Network Shared Memory Data for Poll Thread: 0
Free Message Buffer Bitmap
(bitmap address = 10b9eef80, bitmap size 480)
000000010b9eef80:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
000000010b9eefa0:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
Free Message Buffer Status Bitmap
(bitmap address = 10ca0a9b0, bitmap size 50)
000000010ca0a9b0:ffffffff ffffff
Message Buffer Table
bufid   clientid            addr
Message Buffer Status Table
clientid      netscb addr            addr           offset
```

图 206: onstat -g nsd 命令输出

相关链接
NETTYPE 配置参数 在第135页

onstat -g nss 命令：打印共享内存网络连接的状态

使用 onstat -g nss *sessionid* 命令显示有关共享内存网络连接的状态的信息。
语法：
onstat -gnss [*sessionid*]

如果未提供 *sessionid*，那么为每个共享内存连接列出一行摘要。

示例输出

```
 clientid  clientPID   state    #serverbufs #clientbufs    #rdwrts
       1      14018  Connected        4           4        331
       0      12398  Connected        4           4        294
       2      14036  Connected        4           4         59
```

图 208: onstat -g nss 命令输出

输出描述

clientid (decimal)
服务器为查找分配的值

clientPID (decimal)
客户端进程 ID
state (string)
连接的当前状态

- Connected
- Con1
- Waiting
- Reject
- Bedcover
- Closed
- Not connected
- Unknown

#serverbufs (dec)
当前分配的数据库服务器缓冲区的数量
#clientbufs (dec)
当前分配的客户端缓冲区的数量
#rdwrts (dec)
使用中的缓冲区的总数

相关链接
NETTYPE 配置参数 在第135页

onstat -g ntd 命令: 打印网络统计信息

使用 onstat -g ntd 命令按服务显示网络统计信息。
语法:

```
onstat   -gntd
```

示例输出

```
global network information:
  #netscb connects     read    write  q-limits  q-exceed alloc/max
   4/   5       11        0     3546 3549/  10   10/   0    0/   0

Client Type     Calls   Accepted   Rejected       Read      Write
sqlexec         yes           11          0       3531       3540
srvinfx         yes            0          0          0          0
onspace         yes            0          0          4          9
onlog           yes            0          0          0          0
onparam         yes            0          0          0          0
oncheck         yes            0          0          0          0
onload          yes            0          0          0          0
onunload        yes            0          0          0          0
onmonitor       yes            0          0          0          0
dr_accept       yes            0          0          0          0
cdraccept       no             0          0          0          0
ontape          yes            0          0          0          0
srvstat         yes            0          0          0          0
asfecho         yes            0          0          0          0
listener        yes            0          0         11          0
crsamexec       yes            0          0          0          0
onutil          yes            0          0          0          0
drdaexec        yes            0          0          0          0
smx             yes            0          0          0          0
safe            yes            0          0          0          0
```

```
Totals                          11          0       3546       3549
```

图 210: onstat -g ntd 命令输出

onstat -g ntm 命令: 打印网络邮件统计信息

使用 onstat -g ntm 命令显示有关网络邮件的统计信息。

语法:

```
onstat   -gntm
```

示例输出

```
global network information:
  #netscb connects     read    write  q-limits  q-exceed alloc/max
   4/   5       11        0     3546 3549/  10   10/   0    0/   0

Network mailbox information:
 box   netscb thread name     max received   in box   max in box full signal
   5  f07e8b0 soctcppoll       10       24        0        1        0    yes
   6  f0b6ad8 soctcplst        10        0        0        0        0     no
   7  f0e8b18 soctcplst        10        0        0        0        0     no
```

图 212: onstat -g ntm 命令输出

onstat -g ntt 命令: 打印网络用户时间

使用 onstat -g ntt 命令显示有关网络用户时间的信息。

语法:

```
onstat   -gntt
```

示例输出

```
global network information:
  #netscb connects     read    write    q-limits  q-exceed alloc/max
   3/   3        0        0        0    135/  10    0/   0    2/   0
Individual thread network information (times):
 netscb   thread name   sid  open     read     write     address
c76ea28   ontape        61   14:34:48 14:34:50 14:34:50
c63e548   tlitcplst      4   14:30:43 14:34:48           server.ibm.com|5006|tlitcp
c631028   tlitcppoll     3   14:32:32
```

图 214: onstat -g ntt 命令输出

onstat -g ntu 命令: 打印网络用户统计信息

使用 onstat -g ntu 命令显示有关网络用户统计信息。

语法:

```
onstat   -gntu
```

示例输出

```
global network information:
  #netscb connects     read    write    q-free  q-limits  q-exceed alloc/max
   2/   3       16     2611     2603    1/   1  135/  10    0/   0    1/   1

Individual thread network information (basic):
  netscb type   thread name    sid   fd poll    reads   writes q-nrm q-pvt q-exp
 d1769f0 soctcp soctcplst        3    1    5       16        0  0/ 0  0/ 0  0/ 0
```

```
 d1199f0 soctcp soctcppoll       2    0    5     2595        0  0/ 0  0/ 0  0/ 0
```

图 216: onstat -g ntu 命令输出

onstat -g opn 命令: 打印打开的分区

使用 onstat -g opn 命令按线程 ID 显示系统中当前打开的分区（表和索引）的列表。

使用 ***thread_id*** 选项将列表限制为指定的 ID。

语法:
onstat -gopn [*thread_id*]

输出描述

由星瑞格®软件支持使用这个信息。输出可能随着时间而变化且取决于您的产品版本或修正包。

onstat -g osi: 打印操作系统信息

使用 onstat -g osi 命令显示操作系统上资源和参数的信息，包括共享内存和信号量参数、计算机上当前配置的内存量，以及未使用的内存量。

onstat -g osi 命令还显示计算机上硬件处理程序的统计信息。

当服务器不是处于联机模式时使用此命令。

示例输出

```
Machine Configuration....
OS Name                             Linux
OS Release                          2.6.9-34.ELsmp
OS Node Name                        idas
OS Version                          #1 SMP
OS Machine                          x86_64
Number of processors                4
Number of online processors         4
System memory page size             4096 bytes
System memory                       7970 MB
System free memory                  1536 MB
Number of open files per process    1024
shmmax            33554432
shmmin            1
shmids            4096
shmNumSegs        2097152
semmap            << UnSupported >>
semids            128
semnum            32000
semundo           << UnSupported >>
semNumPerID       250
semops            32
semUndoPerProc    << UnSupported >>
semUndoSize       20
semMaxValue       32767
```

图 218: onstat -g osi 命令输出

onstat -g pos 命令: 打印文件中的值

使用 onstat -g pos 命令显示 $INFORMIXDIR/etc/.infos.DBSERVERNAME 文件中的值。
语法:
onstat -gpos

示例输出

```
1   7   0 infos ver/size 3 264
2   1   0 snum     0 52564801 44000000 4139 demo_on
3   4   0 onconfig path /opt/IBM/informix/etc/onconfig.demo_on
4   5   0 host informixva
5   6   0 oninit ver IBM Informix Dynamic Server Version 11.70.UC2DE
6   8   0 sqlhosts path /data/IBM/informix/etc/sqlhosts.demos
7   3 -32767 sema 32769
8   2 -32768 shm  32768 52564801 44000000 114176000 R
9   2   1 shm      1 52564802 4ace3000 67108864 V
```

图 220: onstat -g pos 命令输出

onstat -g ppd 命令: 显示分区压缩字典信息

使用 onstat -g ppd 命令显示有关为压缩表和表分段或压缩 B 型树索引创建的活动压缩字典的信息。可以选择打印特定编号分区或所有打开分区的信息。

onstat -g ppd 命令打印的信息与 sysmaster 数据库中 syscompdicts_full 表和 syscompdicts 视图显示的信息相同。唯一的区别是 syscompdicts_full 表和 syscompdicts 视图显示有关所有压缩字典的信息，而不是只有活动字典。

语法:
onstat -gppd { [*partition number*] | 0 }

如果指定分区编号，那么 onstat -g ppd 打印该分区的分歧概要文件。如果指定 0，该选项打印所有分区的概要文件。

示例输出

```
partnum    ColOffset  DbsNum    CrTS        CrLogID    CrLogPos  DrTS     DrLogID  DrLogPos
0x1001d5   -1         1         1393371661  4          16339024  0        0        0
0x1001d5   4          1         1393371661  4          16355408  0        0        0
```

图 222: onstat -g ppd 输出

输出描述

partnum
应用压缩字典的分区号

ColOffset
压缩分区 Blob 列的字节偏移量。-1 表示仅压缩该行。

DbsNum
字典驻留的数据库空间号

CrTS
显示创建该字典的时间戳

CrLogID
创建字典时创建的逻辑日志的唯一 ID

CrLogPos
创建字典在逻辑日志中的位置

DrTS
显示字典清除时的时间戳

DrLogID
清除字典时创建的逻辑日志的唯一 ID

DrLogPos
清除字典在逻辑日志中的位置

onstat -g ppf 命令: 打印分区概要文件

使用 onstat -g ppf ***partition_number*** 命令显示指定分区号的分区概要文件。

使用 onstat -g ppf 或 onstat -g ppf 0 命令显示所有分区的概要文件。如果 TBLSPACE_STATS 配置参数设置为 0，那么 onstat -g ppf 命令显示：Partition profiles disabled。

有关 onstat -g ppf 命令的更多信息，请参阅《*SinoDB*® 性能指南》。

语法：
onstat -gppf { *partition_number* | 0 }

示例输出

```
Partition profiles
partnum    lkrqs lkwts dlks  touts isrd  iswrt isrwt isdel bfrd  bfwrt seqsc rhitratio
0x100001   0     0     0     0     0     0     0     0     0     0     0     0
0x100002   1506  0     0     0     416   4     0     4     1282  20    0     97
0x100003   15    0     0     0     5     0     0     0     20    0     0     75
0x1000a5   0     0     0     0     0     0     0     0     12    0     0     67
0x1000e3   4     0     0     0     1     0     0     0     4     0     0     25
0x200001   0     0     0     0     0     0     0     0     0     0     0     0
0x300001   0     0     0     0     0     0     0     0     0     0     0     0
0x400001   0     0     0     0     0     0     0     0     0     0     0     0
```

图 224: onstat -g ppf 命令输出

输出描述

partnum (hex)
分区号
lkrqs (decimal)
分区的锁请求数
lkwts (decimal)
分区的锁等待数
dlks (decimal)
分区的死锁数
touts(decimal)
分区的远程死锁超时的次数
isrd (decimal)
分区的读操作的数量
iswrt (decimal)
分区的写操作的数量
isrwt (decimal)
分区的重写或更新的数量
isdel (decimal)
分区的删除操作的数量
bfrd (decimal)
缓冲区读操作数，以页为单位

bfwrt (decimal)
缓冲区写操作数，以页为单位

seqsc (decimal)
分区的顺序扫描数

rhitratio (percentage)
磁盘读操作和缓冲区读操作的比率

相关链接

TBLSPACE_STATS 配置参数 在第186页

onstat -g pqs 命令：打印所有 SQL 查询的运算符

使用 onstat -g pqs 命令显示当前运行的所有 SQL 查询中使用的运算符的信息。

可以使用此命令对应用程序进行故障排除，并查出查询所运行的运算符，和每个运算符的运行时间和返回行数。EXPLAIN 文件包含的信息将使您对查询计划有概括性了解，onstat -g pqs 命令显示查询和查询计划的运行时运算符的信息。

语法：

onstat -gpqs [*sessionid*]

可以指定以下之一的调用：

表 163: 每个 onstat -g pqs 命令调用的描述

调用	解释
onstat -g pqs	显示每个会话的一行摘要
onstat -g pqs *sessionid*	显示指定会话的信息

示例输出

以下示例显示在不同会话中分别运行的三个 SQL 语句的结果。这些语句是：

```
select * from syscolumns;
select * from systables a, systables b;
update t1 set rowsize = rowsize +100;
```

```
Query Operators:
addr     ses-id   opname    phase   rows    time          in1         in2    stmt-type
ae50b3a  23       scan      open    0       00:00.00         0           0    SELECT
af269d0   5       nljoin    next    224717  00:01.82    af26a90     aeb4478    SELECT
af26a90   5       scan      next    472     00:00.20         0           0    SELECT
aeb4478   5       scan      next    50      00:01.63         0           0    SELECT
ad3c530  26       scan      open    0       00:00.00         0           0    UPDATE (all)
```

图 226: onstat -g pqs 命令输出

输出描述

addr
内存中运算符的地址。可以使用这个地址跟踪属于每个 JOIN 运算符的 SCAN 运算符。

ses-id
运行 SQL 语句的会话的 ID

opname
运算符的名称

phase
运算符使用的阶段。例如：OPEN、NEXT、CLOSE。

rows
运算符处理的行数

time
处理运算符的时间量。以毫秒显示。01:20.10 表示 1 分钟 20 秒 10 毫秒。

in1
join 中的第一个（outer）运算符

in2
join 中的第二个（inner）运算符

stmt-type
SQL 语句的类型，例如：SELECT、UPDATE、DELETE。

onstat -g prc 命令: 显示使用 UDR 或 SPL 例程的会话

使用 onstat -g prc 命令显示当前使用 UDR 或 SPL 例程的会话的数量。
语法：

```
onstat   -gprc
```

示例输出

```
UDR Cache:
    Number of lists          : 31
    PC_POOLSIZE              : 127

UDR Cache Entries:

list id   ref  drop hits      heap_ptr  udr name
--------------------------------------------------------
0    80   0    0    702       4c589020  syscdr@amsterdam:.ifx_allow_newline
0    494  1    0    3         4c1e6820  syscdr@amsterdam:.compare
0    219  0    0    2         4bfd1020  syscdr@amsterdam:.streamread
0    297  0    0    8         4bb99020  syscdr@amsterdam:.ifx_checksum
0    134  0    0    10214     4bb5f020  syscdr@amsterdam:.destroy
0    232  0    0    34        4bd62820  syscdr@amsterdam:.cdrcmd
0    364  0    0    1         4c345020  syscdr@amsterdam:.rci_insert
0    180  0    0    1         4bcba020  syscdr@amsterdam:.gist_drop
0    91   0    0    9         4bd2e020  sysha@amsterdam:.rlt_open
0    500  0    0    76        4bb9f020  sysadmin@amsterdam:.admin
0    27   0    0    1478      4c0ec020  sysadmin@amsterdam:.destroy
...

    Total number of udr entries : 254
    Number of entries in use    : 9
```

图 228: onstat -g prc 命令输出

输出描述

Number of lists
UDR 高速缓存中列表的数量

PC_POOLSIZE
一次可以高速缓存的条目数

list
UDR 高速缓存散列链 ID（存储区编号）

id
例程的唯一 ID
ref
当前从高速缓存访问 UDR 或 SPL 例程的会话数
drop
例程是否标识为已删除
hits
访问该高速缓存条目的次数
heap_ptr
用于存储该条目的堆地址
udr_name
高速缓存中 UDR 或 SPL 例程的名称
Total number of udr entries
高速缓存中的条目数
Number of entries in use
正在使用的条目数

onstat -g proxy 命令: 打印代理分发器信息

使用 onstat -g proxy 命令显示有关代理分发器的信息。onstat -g proxy 命令的输出根据该命令是在主服务器还是辅助服务器上运行会有稍有不同。
语法:
onstat -gproxy { all | [*proxy_id* [*proxy_transaction_id* [{ *sequence_number* }]]] }

调用	解释
onstat -g proxy	显示代理分发器信息
onstat -g proxy all	当在主服务器上运行时，显示有关代理分发器和代理线程的信息。当在辅助服务器上运行时，显示有关当前正在向辅助服务器执行更新的所有会话的信息。
onstat -g proxy ***proxy_id proxy_transaction_id sequence_number***	此选项只在辅助服务器上有效。显示给定代理分发器当前正在执行工作的详细信息。*proxy_transaction_id* 和 *sequence_number* 是可选参数。当提供时，第一个数字将认为是 *proxy_transaction_id*，第二个数字将认为是 *sequence_number*。如果提供的 *proxy_transaction_id* 或 *sequence_number* 不存在，那么该命令输出将与 onstat - 输出相同。

在主服务器上使用 onstat -g proxy 命令的示例输出

```
Secondary          Proxy      Reference Transaction  Hot Row
Node               ID         Count     Count        Total
nagpur_sdc1        2619       0         2            0
nagpur_c2          2632       0         1            0
nagpur_sec         2633       0         1            0    I
```

图 230: onstat -g proxy 命令输出（从主服务器上运行）

输出描述

Secondary Node
主服务器所知的辅助服务器名称

Proxy ID

主分发器的 ID。在高可用性集群中代理 ID 是唯一的。

Reference Count

指示当前事务中正在使用该信息的线程数当计数为 0 时，该事务处理已完成（成功或失败）。

Transaction Count

当前代理分发器正在处理的事务数

Hot Row Total

代理分发器已处理的热行总数

在辅助服务器上使用 onstat -g proxy 命令的示例输出

```
Primary           Proxy      Reference Transaction  Hot Row
Node              ID         Count     Count        Total
nagpur            2619       5         2            0
```

图 231: onstat -g proxy 命令输出（从辅助服务器上运行）

输出描述

Primary Node

主服务器名称

Proxy ID

代理分发器的 ID。在高可用性集群中代理 ID 是唯一的。

Reference Count

指示当前事务中正在使用该信息的线程数。当计数为 0 时，事务处理已经完成（成功或失败）。

Transaction Count

代理分发器当前正在处理的事务数

Hot Row Total

代理分发器已处理的热行总数

在主服务器上使用 onstat -g proxy all 命令的示例输出

```
Secondary         Proxy      Reference Transaction  Hot Row
Node              ID         Count     Count        Total
nagpur_sdc1       2619       0         2            0
nagpur_c2         2632       0         1            0
nagpur_sec        2633       0         1            0

TID      Flags      Proxy  Source   Proxy    Current  sqlerrno iserrno
                    ID     SessID   TxnID    Seq
94       0x00000224 2619   21       1        29       0        0
95       0x00000224 2619   22       2        68       0        0
93       0x00000224 2632   21       2        2        0        0
91       0x00000224 2633   25       1        6        0        0
```

图 232: onstat -g proxy all 命令输出（从主服务器上运行）

输出描述

Secondary Node

主服务器所知的辅助服务器名称

Proxy ID

代理分发器的 ID。在高可用性集群中代理 ID 是唯一的。

Reference Count

指示当前事务中正在使用该信息的线程数。当计数为 0 时，事务处理已经完成（成功或失败）。

Transaction Count

代理分发器当前正在处理的事务数

Hot Row Total

代理分发器已处理的热行总数

TID

主服务器上运行的代理线程的 ID。该 ID 由代理分发器创建，用于处理辅助服务器上会话的工作。

Flags

代理线程的标志

Proxy ID

代表正在运行的代理线程（TID）的代理分发器的 ID

Source SessID

辅助服务器上用户会话的 ID

Proxy TxnID

当前事务的编号。这些编号对于代理分发器是唯一的。

Current Seq

当前事务中当前操作的序号

sqlerrno

任何 SQL 错误的错误号（或者 0 如果没有错误）

iserrno

任何 ISAM 或 RSAM 错误的错误号（或者 0 如果没有错误）

在辅助服务器上使用 onstat -g proxy all 命令的示例输出

```
Primary            Proxy      Reference Transaction  Hot Row
Node               ID         Count     Count        Total
nagpur             2619       5         2            0

Session  Session  Proxy    Proxy    Proxy    Current  Pending  Reference
         Ref      Proxy_id TID      TxnID    Seq      Ops      Count
21       2        2619     94       1        29       1        1
22       2        2619     95       2        68       1        1
```

图 233: onstat -g proxy all 命令输出（从辅助服务器上运行）

输出描述

Primary Node

主服务器名称

Proxy ID

代理分发器的 ID。在高可用性集群中代理 ID 是唯一的。

Reference Count

指示当前事务中正在使用该信息的线程数。当计数为 0 时，事务处理已经完成（成功或失败）。

Transaction Count

代理分发器当前正在处理的事务数

Hot Row Total

代理分发器已处理的热行总数。热行是辅助服务器上被一个以上客户端多次修改的行。当一行被多次修改时，如果在辅助服务器上尚未重播来自不同会话的最新修改操作，那么辅助服务器通过在该行上放置更新锁从主服务器读取其前映像。

Session

会话 ID

Proxy ID

代表正在运行的代理线程的代理分发器的 ID

Proxy TID

主服务器上运行的代理线程的事务 ID。此 ID 是由代理分发器创建用于处理辅助服务器上会话的工作。

Proxy TxnID

当前事务的编号。这些编号对代理分发器是唯一的。

Current Seq

当前事务中当前操作的序号

Pending Ops

在辅助服务器上已缓冲但尚未发送到主服务器的操作数

Reference Count

指示当前事务中正在使用该信息的线程数。当计数为 0 时，事务处理已经完成（成功或失败）。

在辅助服务器上使用 ***proxy_id*** 选项的示例输出

此命令只在辅助服务器上返回信息。

```
Proxy    Reference Pending  ProxySID
TxnID    Count     Ops
1        1         1        3
2        1         1        4
```

图 234: onstat -g proxy proxy_id 命令输出（从辅助服务器上运行）

输出描述

Proxy TxnID

当前事务的编号，这些编号对代理分发器是唯一的。

Reference Count

指示当前事务中正在使用该信息的线程数。当计数为 0 时，事务处理已经完成（成功或失败）。

Pending Ops

在辅助服务器上已缓冲但尚未发送到主服务器的操作数

Proxy SID

代理会话 ID

在辅助服务器上使用 ***proxy_id proxy_transaction_id*** 选项的示例输出

此命令只在辅助服务器上返回信息。

```
Sequence Operation rowid    Table                          sqlerrno
Number   Type              Name
```

```
28       *Update    526       stores_demo:nilesho.customer    0
```

图 235: onstat -g proxy_id proxy_transaction_id 命令输出（从辅助服务器上运行）

输出描述

Sequence Number

操作的编号

Operation Type

执行的操作的类型。可以是：Insert、Update、Delete、Other。

rowid

应用该操作的行的行 ID

Table Name

完整表名称，修改为适合长度。格式：database.owner.tablename

sqlerrno

任何 SQL 错误的错误号（或者 0 如果没有错误）

在辅助服务器上使用 ***proxy_id proxy_transaction_id sequence_number*** 选项的示例输出

此命令只在辅助服务器上返回信息。

输出字段与 onstat -g proxy_id proxy_transaction_id 命令显示的输出字段相同。onstat -g proxy_id proxy_transaction_id 命令显示事务的详细信息，onstat -g proxy_id proxy_transaction_id sequence_number 显示所有事务操作的详细信息。

```
s
Proxy    Reference Pending  ProxySID
TxnID    Count     Ops
61       0         3        22

onstat -g proxy 2788 61


Sequence Operation rowid    Table                          sqlerrno
Number   Type               Name
960      Update    264      stores_demo:nilesho.customer      0
961      Update    265      stores_demo:nilesho.orders        0
962      Update    266      stores_demo:nilesho.items         0

onstat -g proxy 2788 61 962


Sequence Operation rowid    Table                          sqlerrno
Number   Type               Name
962      Update    266      stores_demo:nilesho.items         0
```

图 236: onstat -g proxy_id proxy_transaction_id sequence_number 命令输出（从辅助服务器上运行）

onstat -g qst 命令：打印等待队列和条件队列的等待选项

使用 onstat -g qst 命令显示互斥队列和条件队列（互斥或条件的等待者的队列）的等待统计信息。

QSTATS 配置参数必须设置为 1 以启用统计信息的采集。有关更多信息，请参阅 *QSTATS* 配置参数 在第147页。

语法：

```
onstat   -gqst
```

示例输出

```
Mutex Queue Statistics
name     nwaits   avg_time max_time avgq maxq nservs   avg_time

ddh chai 1        1354863  1354863  1    1    56       1690

Condition Queue Statistics
name     nwaits   avg_time max_time avgq maxq nservs   avg_time

arrived  1        110008   110008   1    1    0        0
logbf0   21       642      4431     1    2    0        0
logbf1   15       475      2519     1    2    0        0
logbf2   19       596      3274     1    2    0        0
bp_cond  1        0        0        1    1    0        0
```

图 238: onstat -g qst 命令输出

输出描述

name (string)
正在等待的互斥或条件资源的名称

nwaits (decimal)
等待该资源的次数

avg_time (decimal)
平均等待时间（以微秒为单位）

max_time (decimal)
最长等待时间（以微秒为单位）

avgq (decimal)
队列的平均长度

maxq (decimal)
队列的最大长度

nservs (decimal)
获得此资源的次数

avg_time (decimal, microsecond)
每次获得此资源时持有的平均时间（以微秒为单位）

相关链接

QSTATS 配置参数 在第147页

onstat -g rah 命令: 显示预读请求统计信息

使用 onstat -g rah 命令显示预读请求的信息。
语法:
onstat -grah

示例输出

```
Read Ahead

# Qs                1
# threads           2
# Requests          58690
# Continued         0
# Memory Failures   0
```

```
Last Thread Add      04/06/2013.14:34
Way behind           0

Partition ReadAhead Statistics

          Buffer   Disk   Hit    Data       Index        Idx/Dat     Log/PageList Last Committed
Partnum   Reads    Reads  Ratio  # Reqs Eff # Reqs  Eff  # Reqs  Eff # Pages Eff  # Reqs  Eff   #
 Resch
0x200003  4312677  110    99     0      0   0       0    0       0    0      0     12906  100   0
0x300002  23740584 1427   99     0      0   0       0    0       0    0      0     6681   100   7
0x400002  17818942 966    99     0      0   0       0    0       0    0      0     25849  100   57
```

图 240: onstat -g rah 命令输出

输出描述

Qs
: 预读请求的队列数

threads
: 预读线程数

Requests
: 预读请求数

Continued
: 持续发生预读请求的次数

Memory Failures
: 由于内存不足而请求失败的次数

Last Thread Add
: 上次添加预读线程的日期和时间

Way behind
: 由于预读程序太落后而删除的页列表请求数

Partnum
: 分区号

Buffer reads
: 已读取的缓冲池数和磁盘页数

Disk Reads
: 从磁盘读取的页数

Hit Ratio
: 分区的高速缓存命中率

Reqs
: 预读请求数。（此输出字段有 5 个部分：数据、索引、索引数据、日志页和上次提交的行）

Eff
: 预读请求的效率。它是通过预读操作请求的页数与已高速缓存而不需预读操作的页数的比率。其值位于 0 到 100 之间。值越高表示预读效果更好。（此输出字段有 5 部分：数据、索引、索引数据、日志页和上次提交的行）

Resch
: 由于对多分片行的更新不完整而要重新安排上次提交的行的请求数。

onstat -g rbm 命令: 打印共享内存的块映射

使用 onstat -g rbm 命令显示共享内存常驻段中可用与已使用块的十六进制位图。

语法:

```
onstat   -grbm
```

示例输出

```
Block bitmap for resident segment address 0x44000000:
 address = 0x440003bc, size(bits) = 3035
 used = 3031, largest_free = 4

      0:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
    256:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
    512:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
    768:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   1024:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   1280:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   1536:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   1792:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   2048:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   2304:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   2560:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
   2816:ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff fffffe00
```

图 242: onstat -g rbm 命令输出

输出描述

address (hex)
  段内已使用/可用块的内存中起始地址

size (bits)
  块位图中位的数量；每个位代表一个块。

used (blocks)
  位图中已使用的块

largest_free (blocks)
  最大可用块

Bit number (decimal): data (hex)
  位号，后跟 32 字节数据（十六进制）

onstat -g rea 命令: 显示准备就绪的线程

使用 onstat -g rea 命令显示有关当前已就绪的虚拟处理器线程的信息。
语法：

```
onstat   -grea
```

示例输出

```

Ready threads:
tid     tcb     rstcb   prty    status   vp-class   name
6       536a38  406464  4       ready       3cpu    main_loop()
28      60cfe8  40a124  4       ready       1cpu    onmode_mon
33      672a20  409dc4  2       ready       3cpu    sqlexec
```

图 244: onstat -g rea 命令输出

onstat -g rss 命令: 打印 RS 辅助服务器信息

使用 onstat -g rss 命令显示有关远程独立辅助服务器的信息。
语法：

```
onstat   -grss [{ verbose  | log server_name }]
```

onstat -g rss 命令的输出根据该命令是在主服务器还是 RS 辅助服务器上运行而稍有不同。

调用	解释
onstat -g rss	显示 RS 辅助服务器的简要信息
onstat -g rss verbose	显示 RS 辅助服务器的详细信息
onstat -g rss log	显示日志信息。此命令只可在主服务器上运行。
onstat -g rss ***server_name***	显示有关特定 RS 辅助服务器的信息。此命令只可在主服务器上运行。

示例输出（主服务器）

```
Local server type: Primary
Index page logging status: Enabled
Index page logging was enabled at: 2009/08/31 09:35:22
Number of RSS servers: 1

RSS Server information:

RSS Server control block: 0x5fdd9740
RSS server name: serv3
RSS server status: Active
RSS connection status: Connected
RSS flow control:576/528
Log transmission status: Active
Next log page to send(log id,page): 53,117632
Last log page acked(log id,page): 53,115615
Last log page applied (log id, page); 53,115615
Time of Last Acknowledgment: 2009-08-31.14:14:09
Pending Log Pages to be ACKed: 1984
Approximate Log Page Backlog:97104
Sequence number of next buffer to send: 3676
Sequence number of last buffer acked: 3612
Supports Proxy Writes: Y
```

图 246: onstat -g rss verbose 命令输出，在主服务器上运行。

输出描述（主服务器）

Local server type

主服务器类型或 RSS （远程独立辅助服务器）服务器类型

Index page logging status

显示主服务器与辅助服务器之间的索引页日志记录是启用还是禁用

Index page logging was enabled at

启用索引页日志记录的日期和时间

Number of RSS servers

连接到主服务器的 RS 辅助服务器数

RSS Server control block

RS 辅助服务器控制块

RSS Server name

RS 辅助服务器的名称

RSS Server status

显示 RS 辅助服务器是否处于活动状态

RSS flow control

以逻辑日志页数表示的值，确定流控制何时启用或禁用。

RSS Connection status
RS 辅助服务器的连接状态
Log transmission status
显示日志传输是否处于活动状态
Next log page to send (log id, page)
要发送的下一个日志页的日志 ID 和页号
Last log page acked (log id, page)
上一个应答的日志的日志 ID 和页号
Last log page applied (log id, page)
上一个应用的日志的日志 ID 和页号
Time of Last Acknowledgment
上一个日志应答的时间
Pending Log pages to be ACKed
已发送但尚未应答的日志数
Approximate Log Page Backlog
已发送日志数与最后逻辑日志数的差
Sequence number of next buffer to send
要发送的下一个缓冲区的序号
Sequence number of last buffer acked
上一个应答的缓冲区的序号
Supports Proxy Writes
显示服务器当前是否配置了允许向辅助服务器进行更新。Y = 支持向辅助服务器进行更新，N = 不支持向辅助服务器进行更新。

使用 log 选项的示例输出（主服务器）

```
Log Pages Snooped:
RSS Srv         From      From      Tossed
name             Cache     Disk     (LBC full)

cdr_ol_nag_1_c1    1368       1331      0
cdr_ol_nag_1_c2    1357       1342      0
cdr_ol_nag_1_c3    1356       1343      0
```

图 247: onstat -g rss log 命令输出，在主服务器上运行。

使用 log 选项的输出描述（主服务器）

Log Pages Snooped
每个 RS 辅助服务器的统计信息
RSS Srv name
RS 辅助服务器名称
From Cache
来自高速缓存号
From Disk
来自磁盘的日志
Tossed (LBC full)
由于 LBC 变满而丢弃的日志页数

示例输出（RS 辅助服务器）

```
Local server type: RSS
Server Status: Active
Source server name: cdr_ol_nag_1
Connection status: Connected
Last log page received(log id,page): 7,877
```

图 248: onstat -g rss 命令输出，在 RS 辅助服务器上运行。

输出描述（RS 辅助服务器）

Local server type
: 主服务器类型或 RSS（远程独立辅助服务器）服务器类型

Server Status
: 显示 RS 辅助服务器是否处于活动状态

Source server name
: 主服务器名称

Connection status
: RS 辅助服务器的连接状态

Last log page received (log id,page)
: 最近接收的日志的日志 ID 和页号

使用 verbose 选项的示例输出（RS 辅助服务器）

```
RSS Server control block: 0x45a3fe58
Local server type: RSS
Server Status: Active
Source server name: my_server
Connection status: Connected
Last log page received(log id,page): 10,1364
Sequence number of last buffer received: 489
Sequence number of last buffer acked: 489
Delay Apply: Configured (3)
Stop Apply: Not configured.
Delay or Stop Apply control block: 0x45a40ba8
    Pending pages: 7
    Last page written: (10:1372).
    Next page to read: (10:1366).
    Delay or Stop Apply thread: Running.
```

图 249: onstat -g rss verbose 命令输出，在 RS 辅助服务器上运行。

使用 verbose 选项的输出描述（RS 辅助服务器）

RSS Server control block
: 服务器控制块

Local server type
: 本地服务器类型

Server Status
: RS 辅助服务器的状态

Source server name
: 在 RS 辅助服务器的高可用性集群中的主服务器名称

Connection status
RS 辅助服务器与集群的主服务器之间的连接状态
Last log page received (log id,page)
RS 辅助服务器上次应答的日志的日志 ID 和页号
Sequence number of last buffer received
RS 辅助服务器上次接收的缓冲区的顺利号
Sequence number of last buffer acked
RS 辅助服务器上次应答的缓冲区的序号
Delay Apply
是否配置了延迟应用。延迟值，以秒表示，括在括号中。
Stop Apply
是否配置了停止应用。停止值括在括号中，其为 1 或 Unix 时间。
Delay or Stop Apply control block
延迟应用或停止应用的控制块
Pending pages
正在等待写入日志暂存目录的页数
Last page written
最近写入日志暂存目录的日志的日志 ID 和页号
Next page to read
下一个要写入日志暂存目录的日志的日志 ID 和页号
Delay or Stop Apply thread
延迟应用或停止应用线程的状态

onstat -g rwm 命令: 打印读取和写入互斥

使用 onstat -g rwm 命令显示有关读取、写入和等待互斥线程的信息，并列出这些线程获取的凭单的地址
语法:
onstat -grwm

示例输出

```
MUTEX   NAME     write/read/wait    tcb list
<address> <name>     first mutex
       Writer     ticket = <ticket address>   tcb=<thread address> <thread name>
       Readers    ticket = <ticket address>   tcb=<thread address> <thread name>
       Waiters    ticket = <ticket address>   tcb=<thread address> <thread name>
<address> <name>        second mutex
       Writer     ticket = <ticket address>   tcb=<thread address> <thread name>
       Readers     ticket = <ticket address>  tcb=<thread address> <thread name>
       Waiters     ticket = <ticket address>  tcb=<thread address> <thread name>
....
....
....
<address> <name>        last mutex
       Writer     ticket = <ticket address>   tcb=<thread address> <thread name>
       Readers    ticket = <ticket address>   tcb=<thread address> <thread name>
       Waiters    ticket = <ticket address>   tcb=<thread address> <thread name>
```

图 251: onstat -g rwm 命令输出

输出描述

tcb

线程地址列表

Writer

写线程列表

Readers

读线程列表

Waiters

等待线程列表

ticket

线程获得的凭单的地址

onstat -g sch 命令: 打印 VP 信息

使用 onstat -g sch 命令显示有关迁移，以及每个虚拟处理器的信号量操作、自旋和忙等待的数量的信息。

语法：

```
onstat   -gsch
```

示例输出

```
VP Scheduler Statistics:
 vp    pid       class       semops    busy waits  spins/wait
 1     3284      cpu         23997     0           0
 2     1340      adm         0         0           0
 3     4624      lio         2         0           0
 4     3320      pio         2         0           0
 5     6076      aio         7710      0           0
 6     4580      msc         46        0           0
 7     3428      soc         7         0           0
 8     2308      soc         1         0           0

Thread Migration Statistics:
 vp  pid      class steal-at steal-sc idlvp-at idlvp-sc inl-polls Q-ln
 1   3284     cpu   0        0        0        0        0         0
 2   1340     adm   0        0        0        0        0         0
 3   4624     lio   0        0        0        0        0         0
 4   3320     pio   0        0        0        0        0         0
 5   6076     aio   0        0        0        0        0         0
 6   4580     msc   0        0        0        0        0         0
 7   3428     soc   0        0        0        0        0         0
 8   2308     soc   0        0        0        0        0         0
```

图 253: onstat -g sch 命令输出

onstat -g scn 命令: 显示扫描信息

使用 onstat -g scn 命令显示当前扫描的状态和有关该扫描的信息。

如果您有一个长时间运行的扫描，那么可以使用此命令检查扫描的进度、确认需要多久才完成，并检视有关该扫描的信息。对于表来说，onstat -g scn 命令输出标识扫描是轻扫描还是缓冲池扫描。

语法：

```
onstat   -g scn
```

示例输出

```
Light Scan Info
descriptor  address           next_lpage  next_ppage        ppage_left  bufcnt  look_aside

RSAM batch sequential scan info

SesID Thread Partnum  Rowid    Rows Scan'd Scan Type Lock Mode  Notes
48    68     10016e   12bb09   43146       Light     Table      Look aside,
40    47     100106   101      0           Buffpool  +Test      Must copy
```

图 255: onstat -g scn 输出显示表信息

扫描运行时，有关索引扫描的信息是有效的。

```
RSAM batch index scan info

SesID Thread Partnum  Scan Type Lock Mode  Notes
136   156    100197             SLock+Test
       Start Key   GT   :-2147483648:
       Stop Key    EQ   :1500:
       Current key      :170:
       Current position: buffp 0x10a4bc0c8 pagenum 2 slot 17 rowoff 4 flags 0
```

图 256: onstat -g scn 输出显示索引扫描信息

输出描述

descriptor (decimal)
   轻扫描 ID

address (hex)
   轻扫描描述符的内存地址

next_lpage (hex)
   要扫描的下一个逻辑页地址

next_ppage (hex)
   要扫描的下一个物理页地址

ppage_left (decimal)
   当前扩展数据块中剩余要扫描的物理页数

bufcnt
   此轻扫描使用的轻扫描缓冲区数

look_aside
   此轻扫描是否需要观察周边（Y = 是，N = 否）。当线程需要在缓冲池中检查现有页以获取正在轻扫描的页的最新映像时，发生观察周边。

SesID
   会话 ID

Thread
   线程 ID

Partnum
   分区号

Rowid
   当前行的 ID

Rows Scan'd

已扫描的行数

Scan Type

对于表来说，是以下其中之一：

- Bufferpool
- Light (light scan)

对于索引来说，是以下其中之一：

- key only
- 无值，如果该扫描不是仅键值扫描。

Lock Mode

获得的锁的类型或没有锁：

- Table（获得表级别锁）
- Slock （获得共享锁）
- Ulock（获得更新锁）
- 空白（没有锁）

此列还可显示以下之一值：

- +Test（扫描测试带有指定锁类型的冲突； 没有获得锁）
- +Keep（获得的锁将持有直到会话结束而不是事务结束）

Notes

此列显示以下之一的值：

- Look aside

  轻扫描正在执行观察周边。

  轻扫描直接从磁盘读取数块页到大缓冲区，而不是从缓冲区管理器获得每个页。在某些情况下，这个过程会要求轻扫描检查来自大缓冲区的每个数据页是否存在缓冲池；此过程称为 *look aside*。如果页存在缓冲池中，那么轻扫描将使用该副本而不是使用在轻扫描大缓冲区中的那个数据页。如果页不存在缓冲池中，那么轻扫描将使用从磁盘读入大缓冲区的副本。如果轻扫描正在执行观察周边，那么扫描性能会稍微降低。

  在许多情况下，请扫描会检测到缓冲池不可能具有新版本的页。在这些情况下，请扫描不会检查缓冲池，并且不会有此值。
- Forward row lookup

  服务器正在对具有跨页的行的表执行轻扫描。该轻扫描必须访问并使用缓冲池以取得主页上任何不完整行的剩余片段。

Start key

扫描的起始键

Stop key

扫描的结束键

Current key

扫描中的当前键

Current position

扫描在索引中的当前位置。例如：页、槽和偏移量。

onstat -g sds 命令: 打印 SD 辅助服务器信息

使用 onstat -g sds 命令显示有关共享磁盘辅助服务器的信息。

语法:

onstat -g sds [{ *server_name* | verbose }]

onstat -g sds 命令的输出根据该命令是在主服务器还是 SD 辅助服务器上运行而稍有不同。

调用	解释
onstat -g sds	显示 SD 辅助服务器的简要信息
onstat -g sds verbose	显示 SD 辅助服务器的详细信息
onstat -g sds ***server_name***	显示特定 SD 辅助服务器的信息。当指定 *server_name* 时，该命令必须在主服务器上运行。

示例输出（主服务器）

```
Local server type: Primary
Number of SDS servers:1

SDS server information

SDS srv      SDS srv      Connection        Last LPG sent  Supports
name         status       status            (log id,page)  Proxy Writes
C_151162     Active       Connected             554,4998   Y
```

图 258: onstat -g sds 命令输出（在主服务器上运行）

输出描述（主服务器）

Local server type

主服务器类型或 SDS（共享磁盘辅助服务器）服务器类型

Number of SDS servers

连接到主服务器的 SD 辅助服务器数

SDS Srv name

SD 辅助服务器名称

SDS Srv status

显示 SD 辅助服务器是否处于活动状态

Connection status

显示 SD 辅助服务器是否连接

Last LPG sent (log id, page)

最近的 LPG 日志 ID和页号

Supports Proxy Writes

显示服务器当前是否配置了允许向辅助服务器进行更新。Y = 支持向辅助服务器进行更新，N = 不支持向辅助服务器进行更新。

使用 verbose 选项的示例输出（主服务器）

```
Number of SDS servers:1
Updater node alias name: server_1
SDS server control block: 0x4d6a5e08
server name: server_2
server type: SDS
server status: Active
```

```
connection status: Connected
Last log page sent(log id,page):5,1829
Last log page flushed(log id,page):5,1829
Last log page acked (log id, page):5,1829
Last LSN acked (log id,pos):5,7492024
Last log page applied(log id,page): 5,1829
Approximate Log Page Backlog:0
Current SDS Cycle:1054
Acked SDS Cycle:1054
Sequence number of next buffer to send: 84329
Sequence number of last buffer acked: 84326
Time of last ack:2013/12/12 09:13:49
Supports Proxy Writes: N
Time of last received message: 2013/12/12 09:13:49
Time of last alternate write: N/A
Time of last alternate read : N/A
```

图 259: onstat -g sds ***server_name*** 命令输出（在主服务器上运行）

使用 verbose 选项的输出描述（主服务器）

Number of SDS servers

与主服务器共享磁盘空间的 SD 辅助服务器数

Updater node alias name

主服务器名称

SDS server control block

SD 辅助服务器控制块

server name

服务器名称

server type

服务器类型

server status

显示服务器是否处于活动状态

connection status

主服务器与辅助服务器之一的连接状态

Last log page sent (log id, page)

最近发送的日志页的日志 ID 和页号

Last log page flushed (log id, page)

最近清空的日志页的日志 ID 和页号

Last log page acked (log id, pos))

最近应答的日志页

Last LSN acked (log id, pos)

最近应答的日志序列号

Last log page applied(log id,page)

上次应用的日志的日志 ID 和页号

Approximate Log Page Backlog

等待发送的日志数

Current SDS Cycle

星瑞格®支持内部使用，用于监视主服务器与 SDS 服务器之间的协调性

Acked SDS Cycle

星瑞格®支持内部使用，用于监视主服务器与 SDS 服务器之间的协调性

Sequence number of next buffer to send

要发送的下一个缓冲区的序号

Sequence number of last buffer acked

应答的下一个缓冲区的序号

Time of last ack

上次日志应答的日期和时间

Supports Proxy Writes

显示服务器当前是否配置了允许向辅助服务器进行更新。Y = 支持向辅助服务器进行更新，N = 不支持向辅助服务器进行更新。

Time of last received message:

当前服务器最近一次从另一服务器接收消息的时间戳。

Time of last alternate write

当前服务器最近一次写入 SDS_ALTERNATE 配置参数所指定的 BLOB 空间的时间戳

Time of last alternate read

当前服务器最近一次读取 SDS_ALTERNATE 配置参数所指定的 BLOB 空间的时间戳

使用 verbose 选项的示例输出（SD 辅助服务器）

```
SDS server control block: 0xb299880
Local server type: SDS
Server Status : Active
Source server name: my_source_server
Connection status: Connected
Last log page received(log id,page): 7,884
Next log page to read(log id,page):7,885
Last LSN acked (log id,pos):7,3621272
Sequence number of last buffer received: 0
Sequence number of last buffer acked: 0
Current paging file:/dbspaces/page_my_source_server_sdc1_
Current paging file size:2048
Old paging file:/dbspaces/page_my_source_server_sdc1_
Old paging file size:10240
```

图 260: onstat -g sds verbose 命令输出（在 SD 辅助服务器上运行）

使用 verbosee 选项的输出描述（SD 辅助服务器）

SDS server control block

SD 辅助服务器控制块

Local server type

主服务器类型或 SDS（共享磁盘辅助服务器）服务器类型

Server status

显示 SD 辅助服务器是否处于活动状态

Source server name

显示主服务器名称

Connection status

显示 SD 辅助服务器是否连接

Last log page received (log id, page)

最近一次接收的日志页

Next log page to read (log id,page)

顺序中下一个要读取的日志页

Last LSN acked (log id,pos)

最近一次应答的 LSN

Sequence number of last buffer received

上次接收的缓冲区的序号

Sequence number of last buffer acked

上次应答的缓冲区的序号

Current paging file

当前调页文件名称

Current paging file size

当前调页文件的大小

Old paging file

前一个调页文件的名称

Old paging file size

前一个调页文件的大小

onstat -g seg 命令: 显示共享内存段的统计信息

使用 onstat -g seg 命令显示共享内存段的统计信息。

此命令显示附加了多少段及其大小。可以在创建时无缓冲池的转储文件上运行 onstat -g seg 命令。

语法:

```
onstat   -gseg
```

示例输出

```
Segment Summary:
id         key        addr     size       ovhd     class blkused  blkfree
720914    52e44801  44000000 4390912    248812  R    1072     0
753683    52e44802  44430000 131072000  769136  V    22573    9427
819221    52e44803  4c130000 66027520  1        B    16120    0
851990    52e44804  50028000 83648512  1        B    20422    0
Total:    -          -       285138944 -        -    60187    9427
Virtual segment low memory reserve (bytes):4194304
Low memory reserve used 0 times and used maximum block size 0 bytes
```

图 262: onstat -g seg 命令输出

输出描述

id

共享内存段的 ID

key

与共享内存段 ID 关联的共享内存键

addr

共享内存段的地址

size

共享内存段的大小，以字节为单位

ovhd
共享内存段控制信息（开销）大小，以字节为单位

class
共享内存段的类（B 代表缓冲池，R 代表常驻，V 代表虚拟，VX 代表虚拟扩展，M 代表消息）

blkused
已用内存的块数

blkfree
可用内存的块数

Virtual segment low memory reserve (bytes)
当需要关键活动且服务取的可用内存有限时使用的保留内存的大小，以字节为单位（可在 LOW_MEMORY_RESERVE 配置参数中指定保留内存）

Low memory reserve used 0 times and used maximum block size 0 bytes)
服务器使用保留内存的次数和所需的最大内存

相关链接

SHMADD 配置参数 在第166页
SHMBASE 配置参数 在第167页
SHMVIRTSIZE 配置参数 在第169页
LOW_MEMORY_RESERVE 配置参数 在第128页
LOW_MEMORY_RESERVE 配置参数 在第128页
EXTSHMADD 配置参数 在第112页
在共享内存转储文件上运行 *onstat* 命令 在第404页

onstat -g ses 命令：显示会话相关信息

使用 onstat -g ses 命令显示有关会话的信息。

缺省情况下，仅 DBSA 可查看 onstat -g ses 信息。然而，当 UNSECURE_ONSTAT 配置参数设置为 1 时，所有用户都可查看此信息。

语法：
onstat -gses [*session_id*]

可指定以下调用之一。

onstat -g ses
显示每个会话的一行摘要

onstat -g ses ***session_id***
显示特定会话的信息

所有会话的示例输出

```
session                                         #RSAM    total      used       dynamic
id       user     tty      pid      hostname    threads  memory     memory     explain
24       informix -        0        -           0        12288      7936       off
23       informix -        17602    carson      1        57344      48968      off
3        informix -        0        -           0        12288      9168       off
2        informix -        0        -           0        12288      7936       off

Last 20 Sessions Terminated

Ses ID  Username  Hostname  PID    Time              Reason
46      user_1    host_1    21220  01/19/2015.15:20  session limit txn time (60s)
43      user_1    host_1    21340  01/19/2015.15:14  session limit memory (5124 KB)
61      user_1    host_1    21404  01/19/2015.15:04  session limit logspace (10242 KB)
```

```
64      user_1    host_1    21458  01/19/2015.15:02  session limit txn time (39548 KB)
```

图 264: onstat -g ses 命令输出

输出描述：会话部分

Session id
  会话 ID

user
  启动会话的用户

tty
  与此会话前端关联的 tty

pid
  与此会话前端关联的进程 ID

hostname
  此会话连接的主机名

#RSAM threads
  分配给此会话的 RSAM 线程数

total memory
  分配给此会话的内存量

used memory
  此会话实际已使用的内存量

dynamic explain
  生成会话的 SQL 语句的说明输出（on 或 off）

输出描述：Last 20 Sessions Terminated 部分

Ses ID
  会话 ID

Username
  启动会话的用户

Hostname
  此会话连接的主机名

PID
  与此会话前端关联的进程 ID

Time
  终止会话的时间

Reason
  超出限制，后跟括号括起来的限制值

特定会话的示例输出

```
session         effective                     #RSAM    total   used   dynamic
id      user    user      tty  pid   hostname threads  memory  memory explain
53      informix -        36   18638 apollo11 1        73728   63048  off

Program :
/usr/informix/bin/dbaccess

tid      name     rstcb            flags    curstk   status
77       sqlexec  4636ba20         Y--P---  4240     cond wait  sm_read   -
```

```
Memory pools    count 1
name         class addr              totalsize  freesize   #allocfrag #freefrag
53           V     4841d040          73728      10680      84         6

name           free       used           name           free       used
overhead       0          3288           scb            0          144
opentable      0          2904           filetable      0          592
log            0          16536          temprec        0          2208
gentcb         0          1656           ostcb          0          2920
sqscb          0          21296          sql            0          72
hashfiletab    0          552            osenv          0          2848
sqtcb          0          7640           fragman        0          392

sqscb info
scb              sqscb            optofc   pdqpriority optcompind  directives
481b70a0         483e2028         0        0           0           1

Sess       SQL            Current            Iso Lock       SQL  ISAM F.E.
Id         Stmt type      Database           Lvl Mode       ERR  ERR  Vers  Explain
53         -              sysmaster          CR  Not Wait   0    0    9.24  Off

Last parsed SQL statement :
  Database 'sysmaster@lxl'

Xadatasources participated in this session :
Xadatasource name                                      RMID       Active
xabasicdb@atmol10:sitaramv.xads_t3_i1                     6        YES
xabasicdb@atmol10:sitaramv.xads_t2_i1                     4        YES
xabasicdb@atmol10:sitaramv.xads_t1_i3                     3        YES
xabasicdb@atmol10:sitaramv.xads_t1_i2                     2        YES
xabasicdb@atmol10:sitaramv.xads_t1_i1                     1        YES
xabasicdb@atmol10:sitaramv.xads_t2_i2                     5        NO

DRDA client info
           Userid:
           Wrkstnname:   nemea
           Applname:     db2jcc_application
           Acctng:       JCC03510nemea
           Programid:
           Autocommit:
           Packagepath:

Session Limits
               Limit     Current
Locks          10000     1
Memory(KB)     5120      72
Temp Space(KB) 30720     0
Log Space(KB)  10240     0
Txn Time(s)    120       0
```

图 265: onstat -g ses ***session_id*** 命令输出

输出描述：program 部分

显示在会话中使用的客户端程序的完整路径。使用客户端程序信息来监视或停止对数据库的访问。

输出描述：threads 部分

尽管此部分没有标题，但以下输出显示有关线程的信息。

tid
线程 ID
name
线程名称
rstcb
RSAM 控制块
flags
使用以下代码描述线程的状态：
位置 1
B
等待缓冲区
C
等待检查点
G
等待逻辑日志缓冲区写入
L
等待锁定
S
等待互斥
T
等待事务
X
等待事务清除
Y
等待条件
位置 2
*
此位置中的星号表示线程在事务中遇到 I/O 失败
位置 3
A
归档线程
B
开始工作
P
开始准备或已准备好工作
X
XA 已准备好
C
提交中或已提交
R
正在中止或已中止
H
尝试中止或尝试回滚

位置 4

P

主线程

位置 5

R

读取

X

临界区

位置 6

R

恢复线程

位置 7

M

监视线程

D

守护进程线程

C

清除程序

F

清空程序

B

B 型树扫描程序

curstk

当前堆栈大小

status

当前线程状态

输出描述：memory pools 头部分

每个会话池重复这些信息。

name

池的名称

class

分配池的内存类。R 代表常驻，V 代表虚拟，M 代表消息。

addr

池结构的地址

totalsize

池所获得的内存总大小（以字节表示）

freesize

池中可用字节数

#allocfrag

池中已分配内存段的数量

#freefrag

池中可用段的数量

输出描述：Memory pools 部分

name
从池中分配内存的组件名称

free
释放的字节数

used
分配的字节数

输出描述：sqscb info 部分

scb
会话控制块。此为共享内存中主会话结构的地址。

sqscb
会话的 SQL 级别控制块

optofc
OPTOFC 环境变量或 ONCONFIG 配置文件设置的当前值

pdqpriority
PDQPRIORITY 环境变量或 ONCONFIG 配置文件设置的当前值

optcompind
OPTCOMPIND 环境变量或 ONCONFIG 配置文件设置的当前值

directives
DIRECTIVES 环境变量或 ONCONFIG 配置文件设置的当前值

输出描述：SQL 部分

显示指定会话的 SQL 信息。此部分包含与 onstat -g sql 命令输出相同的信息。请参阅 *onstat -g sql* 命令: 打印与 *SQL* 有关的会话信息 在第526页。

输出描述：Last parsed SQL statement 部分

Last parsed SQL statement 部分包含与 onstat -g sql 命令输出相同的信息。请参阅 *onstat -g sql* 命令: 打印与 *SQL* 有关的会话信息 在第526页。

输出描述：Xadatasources participated in this session 部分

Xadatasources participated in this session 部分显示有关会话期间可用的 XA 数据源，以及其资源管理器标识符和当前是否处于活动状态的信息。

Xdatasource name
参与会话的 XA 数据源

RMID
对应 XA 数据源的资源管理器的标识符

Active
是否 XA 数据源仍然处于活动状态

输出描述：DRDA® client info 部分

DRDA® client info 部分显示有关到客户端的分布式关联数据库体系结构（DRDA®）连接的信息。

Userid
客户端用户的用户 ID

Wrkstnname
客户端工作站名称
Applname
客户端应用程序的名称。例如：db2jcc_application。
Acctng
来自客户端的记账字串符。例如：JCC03510nemea 。
Programid
客户端程序标识符（SinoDB® 不使用）
Autocommit
SinoDB® 数据源的缺省事务自动提交模式
Packagepath
客户端程序包路径（SinoDB® 不使用）

输出描述：Session limits 部分
Locks
会话的锁数
Memory(KB)
会话的内存
Temp Space(KB)
会话的临时表空间
Log Space(KB)
单个事务的日志空间
Txn Time(s)
单个事务的持续时间

相关链接
tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页
tenant update 参数: 更改租户数据库属性 (SQL 管理 API) 在第710页

onstat -g shard 命令: 打印有关分片定义的信息

使用 onstat -g shard 命令显示有关分片定义的信息。
语法：
onstat -gshard

输出描述

onstat -g shard 的输出显示以下没有字段标签的信息。

Sharding definition name
分片定义的名称
Database name
包含跨多个分片的集合或表的数据库的名称
Table owner name
跨多个分片的集合或表的所有者
Table name
跨多个分片的集合或表的名称
Shard key
用于分布行或文档的分片键。其值可以是表的一个列、文档字段或一个表达式。

Sharding strategy
确定新行或文档存储在哪个数据库服务器的方法。其值可以是 HASH（哈希算法）、CONSISTENT HASH（一致性哈希算式）或 EXPRESSION（表达式）。

Sharding type
指定行或文档复制到目标服务器后源服务器的动作。其值可以是 DELETE、KEEP 或 INFORMATIONAL。

Shard optimization
指定是否查询可以跳过不包含相关数据的分片服务器。其值可以是 ENABLED 或 NOT ENABLED。

Version column
指定 Enterprise Replication 尝试验证源行或文档没有被更新时使用的列或键。该值是一个列或文档字段。

Sharding rule
复制数据到特定数据库服务器的规则

示例：使用基于一致性哈希分片的分片定义的输出

对于此示例，可以使用以下命令创建分片定义：

```
cdr define shardCollection collection_1 database_1:john.customers_1
   --type=delete --key=b --strategy=chash --partitions=3 --versionCol=column_3
   g_shard_server_1
   g_shard_server_2
   g_shard_server_3
```

以下示例显示在 g_shard_server_1、g_shard_server_2 或 g_shard_server_3 上运行 onstat -g shard 命令的输出。

```
collection_1 database_1:john.customers_1 key:b CONSISTENT HASH:DELETE SHARD OPTIMIZATION:NOT
 ENABLED
Matching for delete:column_3
   g_shard_server_1 (65542) (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 4019 and
 5469)
                            or (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 5719
 and 6123)
                            or (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 2113
 and 2652)
   g_shard_server_2 (65543) (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 6124 and
 7415)
                            or (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 5470
 and 5718)
                            or (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 7416
 and 7873)
   g_shard_server_3 (65544) (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 2653 and
 3950)
                            or mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) >= 7874
                            or mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) < 2113
                            or (mod(abs(ifx_checksum(b::LVARCHAR, 0)), 10000) between 3951
 and 4018)
```

图 267: 对于使用一致性哈希算式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。

每个分片服务器有三个哈希分区。

示例：使用基于哈希分片的分片定义的输出

对于此示例，可以使用以下命令创建分片定义：

```
cdr define shardCollection collection_1 database_1:josh.customers_1
   --type=delete --key=column_2 --strategy=hash --versionCol=column_3
   g_shard_server_A
   g_shard_server_B
   g_shard_server_C
   g_shard_server_D
```

以下示例显示在 g_shard_server_A、g_shard_server_B、g_shard_server_C 或 g_shard_server_D 上运行 onstat -g shard 命令时的输出。

```
collection_1 database_1:josh.customers_1 key:column_2 HASH:DELETE SHARD OPTIMIZATION:ENABLED
Matching for delete:column_3
g_shard_server_A (65545) mod(ifx_checksum(column_2::LVARCHAR, 0), 4) = 0
g_shard_server_B (65546) mod(ifx_checksum(column_2::LVARCHAR, 0), 4) in (1, -1)
g_shard_server_C (65547) mod(ifx_checksum(column_2::LVARCHAR, 0), 4) in (2, -2)
g_shard_server_D (65548) mod(ifx_checksum(column_2::LVARCHAR, 0), 4) in (3, -3)
```

图 268: 对于使用哈希算式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。

示例：使用基于表达式分片的分片定义的输出

对于此示例，可以使用以下命令创建分片定义：

```
cdr define shardCollection collection_2 database_2:john.customers_2
   --type=keep --key=state --strategy=expression --versionCol=version_column
   g_shard_server_F "IN ('AL','MS','GA')"
   g_shard_server_G "IN ('TX','OK','NM')"
   g_shard_server_H "IN ('NY','NJ')"
   g_shard_server_I REMAINDER
```

以下示例显示在 g_shard_server_F、g_shard_server_G、g_shard_server_H 或 g_shard_server_I 上运行 onstat -g shard 命令时的输出。

```
collection_2 database_2:john.customers_2 key:state EXPRESSION:KEEP SHARD
 OPTIMIZATION:ENABLED
Matching for delete:version_column
g_shard_server_F (65564) state IN ('AL','MS','GA')
g_shard_server_G (65565) state IN ('TX','OK','NM')
g_shard_server_H (65566) state IN ('NY','NJ')
g_shard_server_I (65567) not ((state IN ('AL','MS','GA')) or (state IN('TX','OK','NM'))
or (state IN ('NY','NJ')))
```

图 269: 对于使用表达式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。

示例：使用 BSON 分片键和基于表达式分片的分片定义的输出

对于此示例，可以使用以下命令创建分片定义：

```
cdr define shardCollection collection_3 database_3:susan.customers_3
   -t delete -k bson_value_lvarchar(data,'age') -s expression -v version
   g_shard_server_J "BETWEEN 0 and 20"
   g_shard_server_K "BETWEEN 21 and 62"
   g_shard_server_L "BETWEEN 63 and 100"
   g_shard_server_M  REMAINDER
```

以下示例显示在 g_shard_server_J、g_shard_server_K、g_shard_server_L 或 g_shard_server_M 上运行 onstat -g shard 命令时的输出。

```
collection_3 database_3:susan.customers_3 key:bson_value_lvarchar(data,'age')
EXPRESSION:DELETE SHARD OPTIMIZATION:ENABLED
Matching for delete:version
g_shard_server_J (65568) bson_value_lvarchar(data,'age') BETWEEN 0 and 20"
g_shard_server_K (65569) bson_value_lvarchar(data,'age') BETWEEN 21 and 62"
g_shard_server_L (65570) bson_value_lvarchar(data,'age')BETWEEN 63 and 100"
g_shard_server_M (65571) not((bson_value_lvarchar(data,'age') BETWEEN 0 and 20)
or (bson_value_lvarchar(data,'age') BETWEEN 21 and 62) or (bson_value_lvarchar
(data,'age') BETWEEN 63 and 100))
```

图 270: 对于使用 BSON 分片键和表达式将数据分布到多个分片服务器的分片定义的 onstat -g shard 命令输出。

相关链接

《SinoDB Enterprise Replication 指南》: cdr define shardCollection

《SinoDB Enterprise Replication 指南》: cdr change shardCollection

《SinoDB Enterprise Replication 指南》: cdr delete shardCollection

《SinoDB Enterprise Replication 指南》: cdr list shardCollection

《SinoDB Enterprise Replication 指南》: CDR_AUTO_DISCOVER 配置参数

onstat -g sle 命令: 打印所有休眠的线程

使用 onstat -g sle 命令打印所有休眠线程。

语法:

```
onstat   -gsle
```

示例输出

```
Current Admin VP sleep period:  10  millisecs
Sleeping threads with timeouts: 21  threads
   tid v_proc            rstcb             name     time
    49      1         b3b13a8      onmode_mon     0.02
     5      1               0  Cosvr Avail Mgr     0.05
    42      1         b3ad028      main_loop()     0.08
     9      3         b3ad6e8         xtm_svcc     0.64
    14      5               0       mgmt_thd_5     0.65
    13      4               0       mgmt_thd_4     0.65
     4      1               0       mgmt_thd_1     0.65
     6      3               0          dfm_svc     0.98
    33     13               0      mgmt_thd_13     1.54
    27     10               0      mgmt_thd_10     1.54
    21      7               0       mgmt_thd_7     1.54
    12      3               0       mgmt_thd_3     1.76
    29     11               0      mgmt_thd_11     1.76
    23      8               0       mgmt_thd_8     2.08
    31     12               0      mgmt_thd_12     2.08
    35     14               0      mgmt_thd_14     2.98
    19      6               0       mgmt_thd_6     3.00
    25      9               0       mgmt_thd_9     3.00
    37      3               0          sch_rgm     3.48
    44      5         b3af8a8      btscanner 0     7.31
    46      3         b3b0628        bum_sched    41.26
```

图 272: onstat -g sle 命令输出

onstat -g smb 命令：显示智能大对象空间信息

使用 onstat -g smb 命令显示有关智能大对象空间的详细信息。

语法：

```
onstat   -gsmb [{ c  | fdd  | lod  | s  | e  | h [{ cad  | fdd  | lod }] }]
```

命令	解释
onstat -g smb c	列出智能大对象空间中所有块
onstat -g smb e	列出所有智能大对象表类型的条目
onstat -g smb e cad	列出智能大对象块附加表的条目
onstat -g smb e fdd	列出智能大对象文件描述符表的条目
onstat -g smb e lod	列出智能大对象头表中的条目
onstat -g smb fdd	列出智能大对象文件描述符
onstat -g smb h	列出所有智能大对象表类型的头
onstat -g smb h cad	列出智能大对象块附加表的头
onstat -g smb h fdd	列出智能大对象文件描述符表的头
onstat -g smb h lod	列出智能大对象头表的头
onstat -g smb lod	列出智能大对象头表中的头和条目
onstat -g smb s	列出智能大对象空间属性（所有者、名称、页大小、-Df 标志设置）。在智能大对象空间创建过程中，不初始化值为 0 或 -1 的字段。

onstat -g smb c 命令的示例输出

使用 onstat -g smb c 命令监视每个智能大对象空间中的可用空间量，以及用户数据和元数据以页表示的大小。onstat -g smb c 命令显示每个智能大对象空间块的以下信息：

- 块号和智能大对象空间名称
- 块大小和路径名
- 总用户数据页和可用的用户数据页
- 每个用户数据和元数据区域中的页的位置和数量

在以下示例中，智能大对象空间 sbspace1 的块 2 具有 2253 个原可用页（orig fr），2253 个用户页（usr pgs）和 2245 个可用页（free pg）。对于第一个用户数据区域（Ud1），起始页偏移量为 53，页数为 1126。对于元数据区域（Md），起始页偏移量为 1179，页数为 194。对于第二个用户数据区域（Ud2），起始页偏移量为 1373，页数为 1127。

```
Chunk Summary:

sbnum 2  chunk 2
chunk:   address   flags    offset   size   orig fr   usr pgs   free pg
         303cf2a8  F------- 0        2500   2253      2253      2245
         path: /usr11/myname/sbspace1

         start pg   npages
Ud1  :   53         1126
Md   :   1179       194
Ud2  :   1373       1127
```

onstat -g smb s 命令的输出

onstat -g smb s 命令显示系统中所有智能大对象空间的存储属性：

- 智能大对象空间名称、标志、所有者
- 日志记录状态
- 平均智能大对象的大小
- 第一个扩展数据块的大小、下一个扩展数据块的大小和最小扩展数据块的大小
- 最大 I/O 访问时间
- 锁定模式

有关 onstat -g smb 命令的更多信息，请参阅《*SinoDB®* 性能指南》。

onstat -g smx 命令：显示多路复用器组信息

使用 onstat -g smx 命令显示有关使用 SMX 的服务器的服务器多路复用器组的信息。
语法：
onstat -gsmx [ses]

命令	解释
onstat -g smx	显示 SMX 连接统计信息
onstat -g smx ses	显示 SMX 会话统计信息

示例输出

```
SMX connection statistics:
SMX control block: 0x47d5e028

  Peer server name: lx1
  SMX connection address: 0x47d60d10
  Total bytes sent: 27055
  Total bytes received: 2006989
  Total buffers sent: 782
  Total buffers received: 7090
  Total write calls: 782
  Total read calls: 7090
  Total retries for write call: 0
  Data compression level: 1
  Data sent: compressed 40760 bytes by 33%
  Data received: compressed 12579324 bytes by 84%
```

图 275: onstat -g smx 命令输出

输出描述

SMX control block

SMX 控制块

Peer server name

显示同级服务器的名称

SMX connection address

显示 SMX 连接的地址

Total bytes sent

显示已发送的总字节数

Total bytes received

显示已接收的总字节数

Total buffers sent

显示已发送的总缓冲区数

Total buffers received

显示已接收的总缓冲区数

Total write calls

显示写入调用的总数

Total read calls

显示读取调用的总数

Total retries for write call

显示写入调用的重试总数

Data compression level

显示 SMX_COMPRESS 配置参数设置的 SMX 压缩级别

Data sent: compressed x bytes by y%

显示已发送数据的未压缩的字节数和压缩率

Data received: compressed x bytes by y%

显示已接收数据的未压缩的字节数和压缩率

示例输出

```
SMX session statistics:
SMX control block: 0x17c69028

Peer        SMX session        client        reads        writes
name        address            type
delhi_sec     19022050   smx Clone Send            6          183
```

图 276: onstat -g smx ses 输出

输出描述

SMX control block

SMX 控制块

Peer name

显示同级服务器的名称

SMX session address

SMX 会话地址

Client type

显示辅助服务器的类型

reads

显示会话读取的总数

writes

显示会话写入的总数

onstat -g spi 命令: 显示使用长自旋的自旋锁

使用 onstat -g spi 命令显示有关使用长自旋的自旋锁的信息。

语法:

```
onstat   -gspi
```

服务器中的许多资源由两个或更多个线程访问。在一些访问中（例如：更新共享值），服务器必须保证每次只有一个线程在访问该资源。自旋锁是为一些资源提供互斥访问的机制。使用这类锁，在第一次尝试时由于另一线程占有锁而没有成功获取锁的线程将重复尝试获取锁，直到成功为止。

自旋锁的开销成本很低，并且通常用于短期间需要互斥的资源。然而，如果自旋锁变得高度争用，那么循环重试机制将变得昂贵。

onstat -g spi 命令对于识别高度争用自旋锁导致的性能瓶颈是很有帮助的。这个选项列出带有等待的自旋锁，这些自旋锁的线程在第一次尝试获取锁时没有成功，因此必须循环重试。

示例输出

```
Spin locks with waits:

Num Waits   Num Loops   Avg Loop/Wait    Name
114         117675      1032.24          lockfr3
 87         256461      2947.83          fast mutex, lockhash[832]
  1             11        11.00          fast mutex, 1:bhash[16668]
  4          51831     12957.75          fast mutex, 1:lru-4
  1            490       490.00          fast mutex, 1:bf[994850] 0xe00002 0x14eb32000
```

图 278: onstat -g spi 命令输出

输出描述

Num Waits (decimal)

线程等待此自旋锁的总次数

Num Loops (decimal)

线程成功获取自旋锁前的尝试总数

Avg Loop/Wait (floating point)

获取自旋锁所需的平均尝试次数。以 Num Loops / Num Waits 计算。

Name (string)

使用以下代码命名自旋锁：

lockfr

锁可用列表。在 lockfr 后的数字是锁可用列表数组中的索引。

lockhash[]

锁散列存储区。方括号中的字段是锁散列存储区数组中的索引。

:bhash []

缓冲散列存储区。冒号前的字段是缓冲池索引；bhash 后方括号中的字段是缓冲散列存储区数组中的索引。

:lru-

LRU 锁存器。冒号前字段是缓冲池索引；lru- 后字段指示正在使用的缓冲区链对。

:bf[]

缓冲区锁存器。冒号前字段是缓冲池索引；bf 后方括号中的字段是缓冲区数组中的缓冲区位置。接下来的两个字段分别是以十六进制表示的缓冲区内存的分区号和页头地址。

onstat -g sql 命令: 打印与 SQL 有关的会话信息

使用 onstat -g sql 命令显示有关会话的 SQL 相关信息。

在缺省情况下，仅 DBSA 可以查看 onstat -g sql syssqltrace 信息。然而，当 UNSECURE_ONSTAT 配置参数设置为 1 时，所有用户都可以查看该信息。

语法：
onstat -gsql *sessionid*

可以指定以下调用之一。

onstat -g sql
显示每个会话的一行摘要

onstat -g sql***sessionid***
显示特定会话的 SQL 信息

```
onstat -g sql 22

Sess  SQL            Current            Iso Lock       SQL  ISAM F.E.           Current
Id    Stmt type      Database           Lvl Mode       ERR  ERR  Vers Explain    Role
22     -             test               CR  Not Wait    0    0   9.03 Off         hr
Last parsed SQL statement :
  select id, name from emp
```

图 280: onstat -g sql 命令输出

输出描述

Sess id
会话标识符

SQL Stmt type
SQL 语句类型

Current Database
会话的当前数据库的名称

ISO Lvl
隔离级别

DR
脏读

CR
已提交读取

CS
游标稳定

DRU
脏读，保留更新锁

CRU
已提交读取，保留更新锁

CSU
游标稳定，保留更新锁

LC
已提交读取，最后提交

LCU
已提交读取，最后提交，保留更新锁

RR
重复读取

NL
无事务数据库

Lock mode

当前会话的锁定模式

SQL Error

当前语句遇到的 SQL 错误号

ISAM Error

当前语句遇到的 ISAM 错误号

F.E. Version

客户端程序使用的 SQLI 协议的版本

Explain

SET EXPLAIN 设置

Current Role

当前用户的角色

onstat -g spf: 打印已就绪语句的概要文件

使用 onstat -g spf 命令显示有关 SQL 查询的当前统计信息。

可以使用统计信息确定每条语句的成本。

语法：

```
onstat   -gspf
```

如果启用了 SQL 跟踪，那么显示的信息语句所完成的工作的快照，并且它可能随着语句继续运行而改变。例如：要监视活动的语句中的缓冲区读取或写入的增长率，那么可以以 2 秒的间隔发出三次 onstat -g spf 的运行。

如果禁用了 SQL 跟踪，那么会发出警告消息："Statistics disabled"。

示例输出

```
Statement profiles
sid  sdb      tottm  execs runtm  pdq scans sorts bfrd  pgrd  bfwrt  pgwrt  lkrqs  lkwts
35   4de84028 0.01   0     0.01   0   0     0     301   352   0      512    2998   0
25   4dc0b028 0.00   0     0.00   0   0     0     0     0     0      0      0      0
...
```

图 282: onstat -g spf 命令输出

输出描述

sid

会话 ID

sdb

语句指针的最后 8 位数字

tottm

以秒表示所有语句的当前总运行时间

execs

完成运行的语句数。此值不包含正在运行的语句。

runtm

以秒表示语句的当前运行时间

pdq

当前并发数据库查询（PDQ）优先级别。PDQ 优先值可以是 0 到 100 之间的任何整数。有关更多信息，请参阅《*SinoDB*® 性能指南》中的管理 *PDQ* 查询。

scans
当前已分配的 PDQ 扫描的数量
sorts
当前完成的排序的数量
bfrd
当前的缓冲区读取数
pgrd
当前的页读取数
bfwrt
当前的缓冲区写入数
pgwrt
当前的页写入数
lkrqs
当前的锁请求数
lkwts
当前的锁等待数

相关链接

set sql tracing 参数: 设置全局 *SQL* 跟踪 *(SQL* 管理 *API)* 在第685页

onstat -g src 命令: 共享内存中的模式

使用 onstat -g src 命令在共享内存中搜索模式。

语法:

onstat -gsrc *patternmask*

示例输出

以下示例显示 onstat -g src***pattern mask*** command where *pattern* = 0x123 and *mask* = 0xffff 的输出。

```
Search Summary:
addr             contents
000000000ad17a50: 01090000 00000000 00000000 00000123   ........ .......#
000000000ad7dec0: 00000001 014e3a0c 00000000 0ade0123   .....N:. .......#
```

图 284: onstat -g src 命令输出

输出描述

addr (hexadecimal)
在共享内存种找到搜索模式的地址

contents (hexadecimal)
给定地址的内存内容

onstat -g ssc 命令: 显示 SQL 语句出现次数

使用 onstat -g ssc 命令监视数据库服务器读取高速缓存中 SQL 语句的次数。

在缺省情况下，仅 DBSA 可以查看 onstat -g ssc syssqltrace 信息。然而，当 UNSECURE_ONSTAT 配置参数设置为 1 时，所有用户都可以查看这个信息。

语法:

onstat -gssc [{ all | pool }]

all 选项显示仅键的高速缓存条目以及完整高速缓存的语句。 如果 hits 列的值比 STMT_CACHE_HITS 值小，则该条目是仅键的高速缓存条目。有关更多信息，请参阅《*SinoDB*® 性能指南》中的内存利用率。

pool 选项显示 SQL 语句高速缓存的所有内存池的使用。输出显示内存池的名称、类、地址和总大小的信息。有关更多信息，请参阅《*SinoDB*® 性能指南》中的提升查询性能。

示例输出

```
Statement Cache Summary:
#lrus   currsize  maxsize   Poolsize  #hits   nolimit
4       117640    524288    139264    0       1
Statement Cache Entries:
lru hash ref_cnt hits flag heap_ptr     database           user
---------------------------------------------------------------
  0  262       0    7   -F  aad8038      sscsi007           admin
   INSERT INTO ssc1 ( t1_char , t1_short , t1_key , t1_float , t1_smallfloat
    , t1_decimal , t1_serial ) VALUES ( ? , ? , ? , ? , ? , ? , ? )
  0  127       0    9   -F  b321438      sscsi007           admin
   INSERT INTO ssc2 ( t2_char , t2_key , t2_short ) VALUES ( ? , ? , ? )
  1  134       0   15   -F  aae0c38      sscsi007           admin
   SELECT t1_char , t1_short , t1_key , t1_float , t1_smallfloat ,
    t1_decimal , t1_serial FROM ssc1 WHERE t1_key = ?
  1  143       0    3   -F  b322c38      sscsi007           admin
   INSERT INTO ssc1 ( t1_char , t1_key , t1_short ) SELECT t2_char , t2_key
    + ? , t2_short FROM ssc2
  2   93       0    7   -F  aae9838      sscsi007           admin
   DELETE FROM ssc1 WHERE t1_key = ?
  2  276       0    7   -F  aaefc38      sscsi007           admin
   SELECT count ( * ) FROM ssc1
  2  240       1    7   -F  b332838      sscsi007           admin
   SELECT COUNT ( * ) FROM ssc1 WHERE t1_char = ? AND t1_key = ? AND
    t1_short = ?
  3   31       0    7   -F  aaec038      sscsi007           admin
   SELECT count ( * ) FROM ssc1 WHERE t1_key = ?
  3   45       0    1   -F  b31e438      sscsi007           admin
   DELETE FROM ssc1
  3  116       0    0   -F  b362038      sscsi007           admin
   SELECT COUNT ( * ) FROM ssc1
    Total number of entries: 10.
```

图 286: onstat -g ssc 命令输出

输出描述 - Statement Cache Summary 部分

#lrus

最近最少使用的队列（LRUS）的数量

currsize

当前高速缓存的大小

maxsize

总高速缓存内存量限制

Poolsize

总的池大小

#hits

插入前的命中数。此数等于 STMT_CACHE_HITS 配置参数的值。

nolimit

STMT_CACHE_NOLIMIT 配置参数的值

输出描述 - Statement Cache Entries 部分

Statement Cache Entries 部分显示完全插入高速缓存的条目。

lru

高速缓存条目所属 lru 队列的索引

hash

高速缓存条目的哈希值

ref_count

引用语句的线程的数量

hits

一个语句与高速缓存中语句匹配的次数。匹配可用于仅键或完全高速缓存条目。

flag

高速缓存条目标志 -D 表示语句已删除，-F 表示语句是完全高速缓存的，-I 表示语句正在变成完全高速缓存状态的过程中。

heap_ptr

高速缓存条目的内存堆的地址

相关链接

STMT_CACHE_HITS 配置参数 在第180页
STMT_CACHE_NOLIMIT 配置参数 在第181页
STMT_CACHE_NUMPOOL 配置参数 在第181页
STMT_CACHE_HITS 配置参数 在第180页

onstat -g stk 命令: 打印线程堆栈

使用 onstat -g stk *tid* 命令显示线程 ID 所指定的线程的堆栈。

在所有平台上都不支持这个选项，并且该选项不是总是精确的。

语法:
onstat -g stk *tid*

示例输出

```
Stack for thread: 2 adminthd
 base: 0x000000010aad5028
  len:   33280
   pc: 0x00000001002821e8
  tos: 0x000000010aadc621
state: running
   vp: 2

0x1002821e8 oninit :: yield_processor + 0x260 sp=0x10aadce20(0x10ac834d0, 0x0, 0x1,
   0x100000000, 0xc8a000, 0x100c8a000)
0x100274e38 oninit :: wake_periodic + 0xdc sp=0x10aadced0 delta_sp=176(0x41b0, 0xc7a024bc,
   0x0, 0x41c4, 0x10aacf598, 0x90)
0x100274fcc oninit :: admin_thread + 0x108 sp=0x10aadcf80 delta_sp=176(0x0, 0x2328,
   0xd26c00, 0x5, 0xc8a000, 0x156c)
0x1002484ec oninit :: startup + 0xd8 sp=0x10aadd050 delta_sp=208(0xa, 0x10aad47d0,
   0x10aad47d0, 0x100db1988, 0xd1dc00, 0x1)
```

图 288: onstat -g stk ***tid*** 命令输出

onstat -g stm 命令: 打印 SQL 语句的内存使用

使用 onstat -g stm 命令显示每个已准备好的 SQL 语句所使用的内存。

在缺省情况下，仅 DBSA 可以查看 onstat -g stm syssqltrace 信息。然而，当 UNSECURE_ONSTAT 配置参数设置为 1 时，所有用户都可以查看此信息。

语法：

```
onstat   -gstm
```

要只显示一个会话的内存，请在 onstat -g stm 命令指定会话 ID。

示例输出

```
session   65 ------------------------------------------------------------
 sdblock  heapsz  statement ('*' = Open cursor)
 aad8028   16544   SELECT COUNT ( * ) FROM ssc1 WHERE t1_char = ?
 AND t1_key = ? AND t1_short = ?
```

图 290: onstat -g stm 命令输出

输出描述

sdblock

语句描述符块的地址

heapsz

语句内存堆的大小

statement

查询文本

onstat -g stq 命令: 打印队列信息

使用 onstat -g stq 命令显示有关队列的信息。

语法：

```
onstat   -gstqsession
```

要查看特定会话的队列信息，请指定 *session* 选项。要查看所有会话的队列信息，就省略 *session* 选项。

示例输出

```
Stream Queue: (session 25 cnt 4)  0:db12400  1:db18400  2:dcf0400  3:dcf6400
 Full Queue:  (cnt 2  waiters 0)  0:0  1:db12400
 Empty Queue: (cnt 0  waiters 0)
```

图 292: onstat -g stq 命令输出

输出描述

session

会话 ID

cnt

流队列缓冲区的数量

waiters

等待流队列缓冲区的线程数

onstat -g sts 命令: 打印每个线程的堆栈使用

使用 onstat -g sts 命令显示有关每个线程最大和当前堆栈使用的信息。

语法:

```
onstat   -g sts
```

示例输出

```
Stack usage:

TID     Total   Max             Current         Thread Name
                bytes   %       bytes   %
2       32768   3124    9       3079    9       adminthd
3       32768   2870    8       2871    8       childthd
5       32768   14871   45      2871    8       Cosvr Avail Mgr
6       32768   2870    8       2871    8       dfm_svc
7       131072  3190    2       3191    2       xmf_svc
9       32768   3126    9       3127    9       xtm_svcc
10      32768   3580    10      3335    10      xtm_svcp
11      32768   3238    9       3239    9       cfgmgr_svc
12      32768   6484    19      2871    8       lio vp 0
14      32768   6484    19      2871    8       pio vp 0
16      32768   6484    19      2871    8       aio vp 0
18      131072  10391   7       2871    2       msc vp 0
20      32768   4964    15      2871    8       fifo vp 0
22      32768   4964    15      2871    8       fifo vp 1
24      32768   6028    18      2871    8       aio vp 1
26      32768   5444    16      2951    9       dfmxpl_svc
27      32768   2886    8       2887    8       sch_svc
28      32768   7812    23      5015    15      rqm_svc
29      32768   7140    21      3079    9       sm_poll
30      32768   11828   36      6439    19      sm_listen
31      32768   2870    8       2871    8       sm_discon
32      32768   14487   44      4055    12      main_loop()
33      32768   4272    13      2903    8       flush_sub(0)
34      32768   2902    8       2903    8       flush_sub(1)
35      32768   2870    8       2871    8       btscanner 0
36      32768   3238    9       3239    9       aslogflush
37      32768   3055    9       2887    8       bum_local
38      32768   3238    9       3239    9       bum_rcv
39      32768   4902    14      4903    14      onmode_mon
42      32768   4964    15      2871    8       lio vp 1
44      32768   5136    15      2871    8       pio vp 1
```

图 294: onstat -g sts command output

onstat -g sym 命令: 打印 oninit 实用程序的符号表信息

使用 onstat -g sym 命令显示 oninit 实用程序的符号表信息。

语法:

```
onstat   -gsym
```

示例输出

以下示例显示输出的前几行:

```
Table for oninit has 23378 entries
    Initial value for -base-: 0x0
        0x3451e0 _start
        0x345300 .ld_int
        0x345348 .ld_llong
```

```
        0x3453dc .ld_float
        0x345428 .ld_double
        0x3454c4 .st_int
        0x3454fc .st_llong
        0x34556c .st_float
        0x3455c0 .st_double
        0x34565c .st_float_foreff
        0x345694 .st_double_foreff
        0x345718 main
        0x34c2ac get_cfgfile
        0x34c2fc is_server_alias
```

图 296: onstat -g sym 命令输出

输出描述

onstat -g sym 命令显示 oninit 实用程序中符号（函数和变量）的相关内存地址和名称。

onstat -g tpf 命令: 打印线程概要文件

使用 onstat -g tpf 命令显示线程概要文件。
语法：
onstat -gtpf*tid*

要打印特定线程的概要文件，请指定 *tid* 线程 ID。将 *tid* 设置为 0，则会显示所有线程的概要文件。

示例输出

```
onstat -g tpf 945

Thread profiles
tid lkreqs lkw dl to lgrs isrd iswr isrw isdl isct isrb lx bfr bfw lsus lsmx seq
945 1969   0   0  0  6181 1782 2069 13   0    0    0    0  16183 7348 743580 0    6
```

图 298: onstat -g tpf 命令输出

输出描述

tid
线程 ID

lkreqs
锁请求数

lkw
锁等待数

dl
死锁数

to
远程死锁超时

lgrs
日志记录数

isrd
读取数

iswr
写入数

isrw

重新写入数

isdl

删除数

isct

提交数

isrb

回滚数

lx

长事务数

bfr

缓冲区读取数

bfw

缓冲区写入数

lsus

当前使用的日志空间

lsmx

使用的最大日志空间

seq

顺序扫描数

onstat -g ufr 命令: 打印内存池分段

使用 onstat -g ufr 命令显示指定内存池中当前使用的分段列表。

此命令需要额外参数指定要显示的内存池信息的池名称或会话 ID。每个会话都会分配一个与其会话 ID 名称相同的内存池。使用 onstat -g mem 命令标识池名称，使用 onstat -g ses 命令标识会话 ID。

语法:

onstat -gufr { *pool name* | sessionid }

内存池分为各种用途的分段。使用 onstat -g ufr 命令，有可能看到这些分段的列表，显示其各自大小（以字节为单位）和所包含的信息类型。技术支持通常使用所提供的信息来协助报告问题分析。

指定池名称的示例输出

```
Memory usage for pool name global:
size       memid
1736       overhead
23544      mcbmsg
72         messages
33112      osenv
25432      rsam
88         shmblklist
5170664    net
```

图 300: 指定池名的 onstat -g ufr global 命令输出

指定会话 ID 的示例输出

以下示例显示会话 ID 6 的输出。

```
Memory usage for pool name 6:
size       memid
3256       overhead
144        scb
2968       ostcb
18896      sqscb
3312       opentable
72         sql
808        filetable
352        fragman
552        hashfiletab
1584       gentcb
12096      log
2960       sqtcb
2928       osenv
720        keys
224        rdahead
16248      temprec
```

图 301: 指定会话 ID 的 onstat -g ufr 命令输出

输出描述

size (decimal)
以字节表示的池分段的大小

memid (string)
池分段的名称

onstat -g vpcache 命令: 显示 CPU 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息

运行 onstat -g vpcache 命令来显示有关 CPU 虚拟处理器和租户虚拟处理器专用内存高速缓存的统计信息。
语法:

```
onstat   -gvpcache
```

示例输出

每个 CPU 或租户虚拟处理器都有相同的格式。以下示例显示一个 CPU 虚拟处理器的输出。

```
CPU virtual processor memory block cache statistics - 4096 byte blocks

Number of 4096 byte memory blocks requested for each CPU virtual processor:262144
CPU virtual processor memory block cache mode : Dynamic

vpid    pid         Blocks held  Hit percentage   Free cache
1       2557540     4667202       99.2 %           100.0 %

Current total virtual processor allocations from cache: 59466799, Total frees: 60209953

  size  cur blks    tgt blks    alloc       miss      free       drain  draintime
  1     1662023     9661        49167485    0         49816526   0      Thu Apr 11 09:43:35 2013
  2     130         52428       7609556     297043    7609612    0      Thu Jan  1 00:00:00 1970
  3     329160      9           905094      0         943256     0      Thu Apr 11 09:43:36 2013
  4     424         9           306637      16192     306506     0      Thu Apr 11 09:43:33 2013
  5     10          9           119313      122607    119315     0      Thu Apr 11 09:43:36 2013
  6     20790       9           55305       0         57700      0      Thu Apr 11 09:43:23 2013
  7     9877        9           31164       0         31942      0      Thu Apr 11 09:43:14 2013
  8     2816        5242        6500        0         6537       0      Thu Jan  1 00:00:00 1970
  9     234         9           606575      8323      605525     0      Thu Apr 11 09:43:36 2013
  10    1130        9           5597        0         5679       0      Thu Apr 11 09:43:18 2013
```

```
   11     231          5242         1808         0            1753         0       Thu Jan  1 00:00:00 1970
   12     1068         9            5667         0            5666         0       Thu Apr 11 09:43:28 2013
   13     65           5242         7114         175          7110         0       Thu Jan  1 00:00:00 1970
   14     28           5242         26200        172          26185        0       Thu Jan  1 00:00:00 1970
   15     30           5242         13562        553          13547        0       Thu Jan  1 00:00:00 1970
   16     2627136      34           349124       0            408425       0       Thu Apr 11 09:43:35 2013
   17     1309         9            59           0            107          0       Thu Apr 11 09:27:33 2013
   18     198          5242         7            0            6            0       Thu Jan  1 00:00:00 1970
   19     190          5242         5            0            1            0       Thu Jan  1 00:00:00 1970
   20     60           5242         30           19           19           0       Thu Jan  1 00:00:00 1970
   21     462          5242         38           0            43           0       Thu Jan  1 00:00:00 1970
   22     22           5242         3            0            1            0       Thu Jan  1 00:00:00 1970
   23     69           5242         141          15           135          0       Thu Jan  1 00:00:00 1970
   24     4944         35           189509       2078         185347       0       Thu Apr 11 09:43:35 2013
   25     75           5242         1            0            1            0       Thu Jan  1 00:00:00 1970
   26     0            9            364          220          361          0       Thu Apr 11 09:39:17 2013
   27     27           5242         1            0            2            0       Thu Jan  1 00:00:00 1970
   28     56           5242         415          33           410          0       Thu Jan  1 00:00:00 1970
   29     319          5242         7101         735          7088         0       Thu Jan  1 00:00:00 1970
   30     3240         5242         174          0            223          0       Thu Jan  1 00:00:00 1970
   31     279          11           51994        2515         50682        0       Thu Apr 11 09:43:36 2013
   32     800          5242         256          0            243          0       Thu Jan  1 00:00:00 1970
```

图 303: onstat -g vpcache 命令输出

输出描述

vpid

虚拟处理器的 ID

pid

操作系统分配的虚拟处理器的进程 ID

Blocks held

专用内存高速缓存中可用的 4096-字节块的数量

Hit percentage

发生请求时块可用的时间百分比

Free cache

块被释放以供重复使用而不被调出的时间百分比

Current VP total allocations from cache

从高速缓存拿取一个块或一组块的次数

Total frees

一个块或一组块被添加到高速缓存的次数

size

以 4096-字节表示的内存块的大小

cur blks

当前已分配的4096-字节块的数量（是 size 的倍数）

tgt blks

在高速缓存被调出之前高速缓存条目的目标块数

alloc

请求者接收该大小的块的次数

miss

请求块时却没有可用块的次数

free

内存块放入高速缓存中的次数

drain

将到期块强制调出，为其他块让出空间的次数

draintime

上次内存块被调出的时间

相关链接

VP_MEMORY_CACHE_KB 配置参数 在第195页

VP_MEMORY_CACHE_KB 配置参数 在第195页

《*SinoDB* 性能指南》：专用内存高速缓存

onstat -g wai 命令：打印等待队列线程列表

使用 onstat -g wai 命令显示系统中当前在等待队列中未执行的线程列表。输出是按线程 ID 排序。

语法：

```
onstat   -gwai
```

示例输出

```
Waiting threads:
 tid     tcb             rstcb           prty status               vp-      name
 2       46b1ea40         0               1    IO Idle                5lio        lio vp 0
 3       46b3dc58         0               1    IO Idle                6pio        pio vp 0
 4       46b5dc58         0               1    IO Idle                7aio        aio vp 0
 5       46b7cc58         0               1    IO Idle                8msc        msc vp 0
 6       46b1ed10         460f5028        1    sleeping secs: 1       3cpu
 main_loop()
 9       46d0d6e0         0               1    sleeping forever       1cpu
 soctcplst
 10      46d70b48         0               1    sleeping forever       3cpu
 sm_listen
 11      46e5d9a0         0               1    sleeping secs: 1       3cpu
 sm_discon
 12      46e5dc70         460f5820        1    sleeping secs: 1       3cpu
 flush_sub(0)
 13      46e8a5a8         460f6018        1    sleeping secs: 1       3cpu
 aslogflush
 14      46fe8148         460f6810        1    sleeping secs: 41      3cpu
 btscanner_0
 15      46fe84a8         0               1    IO Idle               10aio        aio vp 1
 16      46fe8778         460f7008        1    sleeping secs: 1       1cpu
 onmode_mon
 36      47531960         460f7ff8        1    sleeping secs: 253     3cpu
 dbScheduler
 37      47531c30         460f87f0        1    sleeping forever       4cpu
 dbWorker1
 38      47491028         460f7800        1    sleeping forever       4cpu
 dbWorker2
```

图 305: onstat -g wai 命令输出

输出描述

tid (decimal)

线程 ID

tcb (hex)

线程控制块的内存地址

rstcb (hex)
RSAM 线程控制块的内存地址

prty (decimal)
线程优先级。数字越大代表优先级越高。

status (string)
Current线程的当前状态

vp- (decimal and string)
上次运行线程的 VP 的虚拟处理器整数 ID，与运行线程的 VP 类的名称连接。

name (string)
线程名称

onstat -g wmx 命令: 打印具有等待者的所有互斥

使用 onstat -g wmx 命令显示具有等待者的所有互斥。

语法:

```
onstat   -gwmx
```

示例输出

```
Mutexes with waiters:
mid      addr              name              holder   lkcnt  waiter   waittime
134825   7000002043a9148   free_lock         11009    0      200      22921
                                                               11010    22918
```

图 307: onstat -g wmx 命令输出

输出描述

mid
内部互斥标识符

addr
已锁定的互斥的地址

name
互斥的名称

holder
持有互斥的线程的线程 ID
0 = 在共享模式中持有读/写互斥

lkcnt
对于读/写互斥，当前在共享模式下正锁定互斥的线程数。对于可重新锁定的互斥，互斥被正持有该互斥的线程锁定或重新锁定的次数。

waiter
正在等待该互斥的线程的 ID 的列表

waittime
以秒表示线程正在等待的时间量

onstat -g wst 命令: 打印线程的等待统计信息

使用 onstat -g wst 命令显示系统中线程的等待统计信息。

WSTATS 配置参数必须设置为 1 才可启用等待统计信息收集。有关更多信息，请参阅 *WSTATS* 配置参数 在第198页 。

语法：
onstat -gwst

示例输出

```
Version 12.10.F  -- On-Line -- Up 18:52:59 -- 78856 Kbytes
name   tid    state            n         avg(us)    max(us)
msc vp 0 5    ready            6              9         17
msc vp 0 5    run              6           1107       2215
msc vp 0 5    IO Idle          5         2985.9s    1496.1s

main_loo 7    IO Wait          55            6496      16725
main_loo 7    yield time       44929          1.2s     343.1s
main_loo 7    ready            44998       206085      343.1s
main_loo 7    run              44985            5        436

...

sqlexec  63   IO Wait          2             1118       2165
sqlexec  63   other cond       6            34237     204142
sqlexec  63   ready            9               7         16
sqlexec  63   run              7             1.1s       7.7s
```

输出描述

name (string)
线程名称

tid (decimal)
线程 ID

state (string)
在此输出行内等待线程的状态。如果单个线程在多个不同状态中等待，那么它可以有多行输出。出现在 state 字段的值包括：

chkpt cond：线程等待检查点条件
cp mutex：线程等待检查点互斥可用
deadlock mutex：线程等待死锁互斥可用
empty Q：线程等待队列中的空缓冲区
fork：线程等待子线程运行
full Q：线程等待队列中的满缓冲区
IO Idle：I/O 线程闲置
IO Wait：线程在等待 I/O 时让出
join wait：线程等待另一线程退出
lock mutex：线程等待锁定互斥可用
lockfree mutex：线程等待锁定释放互斥可用
logflush：发生逻辑日志清空
log mutex：线程等待逻辑日志互斥可用
logcopy cond：线程等待逻辑日志复制条件
logio cond：线程等待逻辑日志条件
lrus mutex：线程等待缓冲区 LRU 互斥可用
misc：线程等待杂项理由
other cond：线程等待内部条件
other mutex：线程等待内部系统互斥可用
other yield：线程因此内部理由让出
OS read：线程等待操作系统读调用完成

OS write：线程等待操作系统写调用完成
ready：线程就绪可运行
run: 线程已运行
sort io：线程等待排序 I/O 完成
vp mem sync：线程等待虚拟处理器内存的同步
yield bufwait：线程在等待缓冲区可用时让出
yield 0：线程因立即超时而让出
yield time：线程因超时而让出
yield forever：线程让出并保持这个方式直到被唤醒

n (decimal)
线程在此状态等待的次数

avg(us) (floating point)
每次等待发生时，线程在此状态下等待的平均用户时间。时间的单位是微秒，值后的 s 表示以秒计算用户时间。

max(us) (floating point)
等待发生时，线程在此状态下等待的最大用户时间。 时间的单位是微秒，值后的 s 表示以秒计算用户时间。

相关链接

onstat -g ath 命令: 显示所有线程的信息 在第426页
WSTATS 配置参数 在第198页

onstat -G 命令：打印 TP/XA 事务信息

使用 onstat -G 命令显示有关通过 TP/XA 库生成的全局事务的信息。
语法：
onstat -G

示例输出

```
Global Transaction Identifiers
address    flags  isol    timeout  fID       gtl  bql  data      dbpartnum
45cb0318   -LH-G  COMMIT  0        4478019   2    2    30323032  100163
```

图 310: onstat -G 命令输出

对于紧耦合事务，所有分支共用 address 列中显示的相同事务地址。

输出描述

address
事务地址

flags
位置 1 的标志代码（当前事务状态）：

A
事务所属的用户线程

S
TP/XA 暂挂的事务

C
TP/XA 正等待回滚

位置 2 的标志代码（事务方式）：

T

紧耦合方式（MTS）

L

松耦合方式（缺省方式）

位置 3 的标志代码（事务阶段）：

B

开始工作

P

已准备提交的分布式查询

X

TP/XA 已准备提交

C

提交中或已提交

R

回滚中或已回滚

H

正尝试回滚或已回滚

位置4 的标志代码：

X

XA 数据源全局事务

位置 5 的标志代码（事务类型）：

G

全局事务

C

分布式查询协调者

S

分布式查询从属者

B

分布式查询协调者和从属者

M

重定向的全局事务

isol

事务隔离级别

timeout

事务锁定超时

fID

格式 ID

gtl

全局事务 ID 长度

bql

分支限定符长度

data

特定于事务的数据

dbpartnum

启动事务的数据库标识符

相关链接

IFX_XA_UNIQUEXID_IN_DATABASE 配置参数 在第120页

onstat -h 命令：打印缓冲头哈希链信息

使用 onstat -h 命令显示用于访问每个缓冲池中页的缓冲区头散列链（有时称为“散列存储区”）的信息。

语法：

```
onstat   -h
```

示例输出

输出以链长度的数字柱状图的形式显示，并带有每个缓冲池的摘要信息。输出中的所有数字值都是十进制。散列链越短，服务器越快找到请求的缓冲区，因为一般来说，在目标链上找到目标缓冲区所需检查的缓冲区头较少

每个缓冲池输出中，最先显示以字节表示的缓冲池的页大小。接着显示该缓冲池的柱状图和摘要信息。

```
Buffer pool page size: 2048

buffer hash chain length histogram
   # of chains         of len
          3423              0
          4546              1
           223              2
          8192   total chains
          4992   hashed buffs
          5000    total buffs

Buffer pool page size: 4096

buffer hash chain length histogram
   # of chains         of len
           707              0
           315              1
             2              2
          1024   total chains
           319   hashed buffs
          1000    total buffs
```

图 312: onstat -h 命令输出

输出描述

Histogram Information on Hash Chains

柱状图对于当前存在系统中的每个缓冲区散列链长度都有一行。每行有两列：

# of chains

给定长度的散列链的数量

of len

这些链的长度

Summary Information Per Buffer Pool

total chains

该缓冲池中存在的散列链的数量

hashed buffs

当前散列到该缓冲池的散列链中的缓冲区头的数量

total buffs

该缓冲池中缓冲区的总数

onstat -i 命令：启动 交互模式

使用 onstat -i 命令将 onstat 实用程序置于交互模式。
语法：
onstat -i [{ r*seconds* | rz*seconds* }]

在交互模式中，可以在每个会话输入多个 onstat 选项，但一次只能输入一个。一个 onstat 提示符允许输入一个选项。

重要： 在交互模式中，不要在选项前加连字符。

其他选项

两个其他选项（onstat r ***seconds*** 和 onstat rz ***seconds***）可用于交互模式。onstat r ***seconds*** 选项类似于当前 onstat -r ***seconds*** 选项，重复生成显示。如果管理员在交互模式提示处执行 onstat r ***seconds***，那么该提示更改以反映指定的时间间隔（以秒为单位）并重新出现，等待下一个命令。以下示例中，由下一个命令生成的显示每 3 秒重复一次：

```
onstat> r 3
onstat[3]>
```

onstat rz ***seconds*** 选项使您可以重复下一个指定的命令，并在每个执行之间将所有概要文件计算器归零。

终止交互模式或重复顺序

要终止交互模式，请按 CTRL-d。

要终止重复顺序，请按 CTRL-c。

onstat -k 命令：打印活动锁的信息

使用 onstat -k 命令打印有关活动锁的信息，其包括锁表中的锁的地址。
语法：
onstat -k

可用锁的最大数量由 onconfig 文件中 LOCKS 配置参数的值指定。

示例输出

```
Locks
address  wtlist   owner    lklist   type     tblsnum  rowid    key#/bsiz
a095f78  0        a4d9e68  0        HDR+S    100002   203        0
 1 active, 2000 total, 2048 hash buckets, 0 lock table overflows
```

图 315: onstat -k 命令输出

在以下输出中，最后一行的数字 2 显示的是一个 Enterprise Replication 伪锁：

```
Locks
address  wtlist    owner    lklist   type    tblsnum  rowid    key#/bsiz
a1993e8  0         5c2f03d0 a19be30     S         2   1c05a           0
```

输出描述

address
锁表中锁的地址

如果用户线程正在等待该锁，那么锁的地址出现在 onstat -u（用户）输出的 wait 字段中。

wtlist
正等待锁的用户线程（如果有）列表的第一个条目

owner
正持有锁的线程的共享内存地址

此地址对应于 onstat -u（用户）输出的 address 字段中的地址。当 owner 值显示在括号中时，它代表事务结构的共享内存地址。只有锁是为全局事务而分配时，才会出现这个情况。该地址对应于 onstat -G 输出的 address 字段。

lklist
由所列所有者持有的锁链接列表中的下一个锁

type
使用以下代码指示锁的类型：

HDR
头

B
字节

S
共享

X
互斥

I
意向

U
更新

IX
意向-互斥

IS
意向-共享

SIX
共享的意向-互斥

tblsnum
锁定资源的表空间号。如果数字小于 10000，那么它表示 Enterprise Replication 伪锁。

rowid
行标识号

rowid 提供以下锁信息：

- 如果 rowid 等于 0，那么该锁是表锁。

- 如果 rowid 以 2 个 0 结束，那么该锁是页锁。
- 如果 rowid 是 6 个数字或更少且不以 0 结束，那么该锁可能是行锁。
- 如果 rowid 是多于 6 个数字，那么该锁可能是索引键值锁。

key#/bsiz

索引键号，或对于 VARCHAR 锁的已锁定字节数

如果此字段包含后跟一个值的 'K-'，那么它是键锁。该值标识正锁定哪个索引。例如：K-1 指示锁定对该表定义的第一个索引。

相关链接

LOCKS 配置参数 在第123页

onstat -l 命令：显示物理和逻辑日志信息

使用 onstat -l 命令显示有关物理日志、逻辑日志和临时逻辑日志的信息。
语法：

```
onstat   -l
```

示例输出

```
Physical Logging
Buffer bufused  bufsize  numpages numwrits pages/io
  P-1  0        16       716      55       13.02
      phybegin         physize  phypos   phyused  %used
      1:263            500      270      0        0.00

Logical Logging
Buffer bufused  bufsize  numrecs  numpages numwrits recs/pages pages/io
  L-3  0        16       42169    2872     1043     14.7       2.8
       Subsystem    numrecs  Log Space used
       OLDRSAM      42169    4436496

address  number   flags    uniqid   begin               size     used    %used
a517f70  1        U-B----  1        1:763                500      500   100.00
a517fb0  2        U-B----  2        1:1263               500      500   100.00
a40daf0  3        U-B----  3        1:1763               500      500   100.00
a40db30  4        U-B----  4        1:2263               500      500   100.00
a40db70  5        U-B----  5        1:2763               500      500   100.00
a40dbb0  6        U---C-L  6        1:3263               500      372    74.40
a40dbf0  7        A------  0        1:3763               500        0     0.00
a40dc30  8        A------  0        1:4263               500        0     0.00
 8 active, 8 total
```

图 317: onstat -l 命令输出

物理日志文件的输出描述

显示的第一部分描述物理日志配置：

buffer

物理日志缓冲区的数量

bufused

已使用的物理日志缓冲区页数

bufsize

每个物理日志缓冲区的大小（以页表示）

numpages

写入物理日志的页数

numwrits

对磁盘的写入数

pages/io

计算方式是 numpages/numwrits

该值指示正在缓冲的物理日志写入的效率。

phybegin

日志开始处的物理页号

physize

物理日志的大小（以页表示）

phypos

日志中要发生下一个日志记录写入的当前位置

phyused

日志中已使用页数

%used

已使用页的百分比

onstat -l 命令输出的第二部分描述逻辑日志配置：

buffer

逻辑日志缓冲区的数量

bufused

逻辑日志缓冲区中已使用的页数

bufsize

每个逻辑日志缓冲区的大小（以页为单位）

numrecs

已写入记录的数量

numpages

已写入页数

numwrits

对逻辑日志的写入数

recs/pages

计算方式是 numrecs/numpages

您不能影响这个值。不同类型的操作生成不同类型（和大小）的记录。

pages/io

计算方式是 numpages/numwrits

可通过更改逻辑日志缓冲区的大小（由 ONCONFIG 文件中的 LOGBUFF 指定）或通过更改数据库的日志记录模式（从已缓冲更改为未缓冲，或反之亦然）来影响此值。

以下字段将对每个逻辑日志文件重复：

address

日志文件描述符的地址

number

逻辑日志文件的日志标识号

日志标识号可能是无序的，因为数据库服务器或管理员可以直接插入日志文件。

flags

提供每个日志的状态，如下所示：

A

新添加（且就绪可用）

B

已备份

C

当前逻辑日志文件

D

标记为删除

要删除日志文件并释放空间以再利用，那么必须对所有存储空间执行 0 级备份。

F

可用的

L

最近检查点记录

U

已使用

uniqid

日志的唯一 ID 号

begin

日志文件的开始页

size

日志的大小（以页为单位）

used

已使用的页数

%used

已使用页的百分比

active

活动逻辑日志的数量

total

逻辑日志的总数

临时逻辑日志文件的输出描述

在热复原过程中，由于永久日志不可用，因此数据库服务器使用临时逻辑日志。以下字段将对每个临时逻辑日志文件重复：

address

日志文件描述符的地址

number

逻辑日志文件的逻辑标识号

flags

提供每个日志的状态，如下所示：

B

已备份

C
当前逻辑日志文件
F
可用的
U
已使用

uniqid
日志的唯一 ID 号

begin
日志文件的开始页

size
日志的大小（以页为单位）

used
已使用的页数

%used
已使用页的百分比

active
活动的临时逻辑日志的数量

相关链接
LOGBUFF 配置参数 在第123页
PHYSBUFF 配置参数 在第143页

onstat -L 命令：显示可用锁的数量

使用 onstat -L 命令打印锁可用列表中可用锁的数量。
语法：
onstat -L

示例输出

```
num   list head     available locks
0     10a143b70     19996
1     101010101     200
3     020202020     300
```

图 319: onstat -L 输出

输出描述

num
列表号

list head
列表的起始地址

available locks
列表上的锁数

onstat -m 命令：打印最近的系统消息日志信息

使用 onstat -m 命令显示系统消息日志的最近 20 个消息。

可以在数据库服务器处于任何模式（包括脱机）时使用 onstat -m 命令选项。

语法：
onstat -m

示例输出

此命令的输出列出消息日志文件的全路径名和 20 个文件条目。日期-时间头分隔每天的条目。每天中的单个条目的开始处都有时间戳。消息日志的名称是由 ONCONFIG 文件中的 MSGPATH 指定。

```
Message Log File: /work/11.50/dbspaces/star3.log
11:26:33  Checkpoint Completed:  duration was 0 seconds.
11:26:33  Checkpoint loguniq 1, logpos 0x23c408, timestamp: 0x2cc2 Interval: 9
```

图 321: onstat -m 命令输出

onstat -o 命令：共享内存内容输出到文件

使用 onstat -o 命令将共享内存的内容写到指定文件，以便日后分析。如果不指定输出文件，那么将在当前目录中创建名为 onstat.out 的文件。
语法：
onstat -o
[{ nobuffs | full }]
{ *outfile* }

使用 nobuffs 选项将共享内存常驻段中的缓冲池从输出文件中排除。这会生成较小的输出文件。

使用 full 选项创建与 SinoDB® 实例的共享内存段同样大小的输出文件。文件系统中必须有足够空间来处理该输出。

如果不指定 nobuffs 也不指定 full 选项，那么输出是由数据库服务器的 DUMPSHMEM 配置参数设置来控制：

- 如果 DUMPSHMEM 设置为 0 或 1，那么 onstat -o 命令将生成一个完整的共享内存转储文件。
- 如果 DUMPSHMEM 设置为 2，那么 onstat -o 命令将生成一个排除常驻段中缓冲池的 nobuffs 共享内存转储文件。

通过对此文件运行其他的 onstat 命令，您可以从先前保存的共享内存转储中收集信息。使用 onstat -o 命令创建的 *outfile* 是用作为运行其他 onstat 命令的源文件 *infile*。有关更多信息，请参阅在共享内存转储文件上运行 *onstat* 命令 在第404页。

相关链接

DUMPSHMEM 配置参数 *(UNIX)* 在第109页

onstat -p 命令：显示概要文件计数

使用 onstat -p 命令显示自数据库服务器启动以来或自运行 onstat -z 命令以来的概要文件计数的信息。
语法：
onstat -p

示例输出

```
Profile
dskreads  pagreads   bufreads   %cached dskwrits   pagwrits   bufwrits   %cached
16934     47321      203600361  99.99   103113     158697     950932     89.16

isamtot   open     start    read     write   rewrite  delete  commit  rollbk
139214865 9195777  12257208 94191268 362691  55696    38134   128294  24

gp_read   gp_write   gp_rewrt   gp_del     gp_alloc   gp_free   gp_curs
39        2          27         51         0          0         16

ovlock    ovuserthread   ovbuff   usercpu  syscpu   numckpts   flushes
0         0              0        1551.59  144.82   1822       1822

bufwaits  lokwaits  lockreqs   deadlks  dltouts  ckpwaits  compress  seqscans
176       1         195872383  0        0        1         39331     1259170

ixda-RA   idx-RA     da-RA    logrec-RA  RA-pgsused lchwaits
0         7594       2124     0          2002       18848
```

图 324: onstat -p 命令输出

输出描述

输出的第一部分描述读取和写入。

读取和写入表列成三类：从磁盘、从缓冲区，以及页数（读取或写入）。

第一个 %cached 字段是缓冲区的读取数与磁盘读取数的相比衡量。第二个 %cached 字段是缓冲区的写入数与磁盘写入数的相比衡量。

数据库服务器以页为单位来缓冲信息和将信息写入磁盘。基于这个原因，显示为 dskwrits 的磁盘写入数通常小于个别用户运行的写入数：

dskreads

实际的磁盘读取数

pagreads

已读取页的数量

bufreads

共享内存的读取数

%cached

缓冲池中已高速缓存的读取数百分比。

如果 bufreads 超过最大整数（或长整数）值，那么其内部表示变成负数，但值显示为 0.0。

dskwrits

实际的磁盘物理写入数

此数字包含 onstat -l 输出中的物理日志和逻辑日志的写入数。

pagwrits

已写入页的数量

bufwrits

共享内存的写入数

%cached

缓冲池中已高速缓存的写入数百分比。

-p 显示的下一部分表列不同 ISAM 调用执行的次数。这些调用发生在操作的最底层，并且不需要与 SQL 语句执行一一对应。一个查询可能生成多个 ISAM 调用。这些统计信息是跨数据库服务器收集的，且不能用于监视单个数据库上的活动，除非只有一台数据库处于活动状态或只有一台数据库存在：

isamtot

调用的总次数

open

当表空间打开时的增量

start

索引内指针的增量

read

当调用读取函数时的增量

write

每次写入调用的增量

rewrite

当发生更新时的增量

delete

当删除一行时的增量

commit

每次调用 iscommit() 时的增量

此值与执行的显示 COMMIT WORK 语句次数之间不存在一对一的对应关系。

rollbk

当回滚事务时的增量

onstat -p 命令输出的下一部分显示有关一般页的信息。一般页管理器提供一个 API 让 SinoDB® 管理数据库服务器缓冲池中的非标准页。下表描述 onstat -p 命令输出中的一般页管理器字段。

gp_read

一般页的读取数

gp_write

一般页的写入数

gp_rewrt

一般页的更新数

gp_del

一般页的删除数

gp_alloc

一般页的分配数

gp_free

已释放并返还表空间的一般页数

gp_curs

用于一般页的游标数

onstat -p 命令输出的下一部分显示请求资源时无资源可用的次数：

ovlock

会话尝试超过最大锁数的次数

有关更多信息，请参阅第 1-56 页的“LOCKS”。

ovuserthread

用户尝试超过最大用户线程数的次数

ovbuff

数据库服务器找不到可用共享内存缓冲区的次数

当没有可用的缓冲区时，数据库服务器会将脏缓冲区写入磁盘，然后设法找到可用缓冲区。

usercpu

所有用户线程使用的总用户 CPU 时间（以秒表示）

此条目每 15 秒更新一次。

syscpu

所有用户线程使用的总系统 CPU 时间（以秒表示）

此条目每 15 秒更新一次。

numckpts

自启动时间以来的检查点数

flushes

缓冲池清空到磁盘的次数

onstat -p 命令输出的下一部分包含杂项信息，如下所示：

bufwaits

每次用户线程必须等待缓冲区时增量

lokwaits

每次用户线程必须等待锁时增加

lockreqs

每次请求锁时增量

deadlks

每次检测到并防止潜在死锁时增量

dltouts

每次用户线程正等待锁时超过分布式死锁超时值时增量

ckpwaits

检查点等待数

compress

每次压缩数据页时增量

seqscans

对每个顺序扫描增量

*

onstat -p 命令输出的最后部分包含以下信息：

ixda-RA

索引叶到数据页的预读的计数

idx-RA

遍历索引叶的预读的计数

da-RA

仅数据路径扫描的计数

logrec-RA

数据库服务器预读的日志记录

RA-pgsused

已使用的数据库服务器预读页数

lchwaits

线程需要等待共享内存锁存器的次数

许多锁存器等待通常是由大量处理活动而造成，而数据库服务器正这些活动中记录大多数事务。

相关链接

DEADLOCK_TIMEOUT 配置参数 在第93页

onstat -P 命令：打印分区信息

使用 onstat -P 命令显示所有分区的分区号和属于分区的缓冲池中的页。

语法：

```
onstat   -P
```

有关在无缓冲池的转储文件上运行 onstat -P 的信息，请参阅在共享内存转储文件上运行 *onstat* 命令 在第404页.

示例输出

```
Buffer pool page size: 2048
partnum  total    btree    data     other    dirty
0        36       1        8        27       0
1048577  2        0        0        2        0
1048578  4        1        1        2        0
1048579  23       10       12       1        0
1048580  68       31       36       1        0
4194309  3        0        1        2        0

Totals:  3000     786      1779     435      0
Percentages:
Data  59.30
Btree 26.20
Other 14.50

Buffer pool page size: 8192
partnum  total    btree    data     other    dirty
0        999      0        0        999      0
5242881  1        0        0        1        0

Totals:  1000     0        0        1000     0
Percentages:
Data  0.00
Btree 0.00
Other 100.00
```

图 326: onstat -P 命令输出

输出描述

Buffer pool page size

以字节表示缓冲池页大小

partnum

分区号

total

分区总数

btree

分区中 B 型树页的数量

data

分区中数据页的数量

other

分区中其他页的数量

dirty

分区中脏页的数量

onstat -r 命令：重复打印选择的统计信息

使用 onstat -r 命令反复在指定时间间隔打印命令中指定的其他选项的统计信息。
语法：
onstat -r [{ *secondsother_options* | *other_options* }]

使用 onstat -r ***seconds other_options*** 命令指定重复其他选项的时间间隔。

使用 onstat -r ***other_options*** 命令使该选项每 5 秒重复一次，这将使其他选项与 -r 选项连接。例如：onstat -rFh。

onstat -r 命令可在命令模式和交互模式中使用，且对于重复命令输出以监视系统资源使用率是很有帮助的

每隔 5 秒运行 onstat -r 的示例输出

```
onstat -r

Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:05:25 -- 1067288 Kbytes


Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:05:30 -- 1067288 Kbytes

Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:05:35 -- 1067288 Kbytes
```

图 328: 命令输出

每隔 10 秒运行 onstat -r 的示例输出

```
onstat -r 10

Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:06:58 -- 1067288 Kbytes


Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:07:08 -- 1067288 Kbytes


Sinoregal DS Version 12.10.F       -- On-Line -- Up 20:07:18 -- 1067288 Kbytes
```

图 329: 命令输出

每隔 1 秒运行带有 -h 选项的 onstat -r 命令的示例输出

```
onstat -r 1 -h
```

```
Buffer pool page size: 2048

buffer hash chain length histogram
   # of chains         of len
          3841              0
          3767              1
           522              2
            62              3
          8192   total chains
          4351   hashed buffs
          5000    total buffs


Buffer pool page size: 2048

buffer hash chain length histogram
   # of chains         of len
          4020              0
          3392              1
           735              2
            43              3
             2              4
          8192   total chains
          4172   hashed buffs
          5000    total buffs
```

图 330: onstat -r 1 -h 命令输出

每隔 5 秒运行带有 -Fh 选项的 onstat -r 命令的示例输出

```
onstat -rFh


Fg Writes     LRU Writes    Chunk Writes
0             0             21

address           flusher  state    data     # LRU    Chunk    Wakeups  Idle Tim
460e6820         0        I        0        0        2        5        9.820
      states: Exit Idle Chunk Lru

Buffer pool page size: 2048

buffer hash chain length histogram
   # of chains         of len
          6342              0
          1850              1
          8192   total chains
          1850   hashed buffs
          5000    total buffs


Fg Writes     LRU Writes    Chunk Writes
0             0             21

address           flusher  state    data     # LRU    Chunk    Wakeups  Idle Tim
460e6820         0        I        0        0        2        10       22.755
      states: Exit Idle Chunk Lru

Buffer pool page size: 2048

buffer hash chain length histogram
   # of chains         of len
          4396              0
```

```
          3796              1
          8192   total chains
          3796   hashed buffs
          5000    total buffs
```

图 331: onstat -rFh 命令输出

onstat -R 命令: 显示 LRU、FLRU 和 MLRU 队列信息

使用 onstat -R 命令显示有关 LRU 队列、FLRU 队列和 MLRU 队列的详细信息。对于每个队列，onstat -R 命令显示队列中的缓冲区数，以及已修改缓冲区的数量和百分比。

有关三种队列的深入讨论，请参阅《*SinoDB*® 管理员指南》的“共享内存”一章中的 LRU 队列。

语法：
onstat -R

示例输出

```
    Buffer pool page size: 2048
    8 buffer LRU queue pairs             priority levels
# f/m   pair total     % of    length       LOW      HIGH
 0 f        375     100.0%      375        375         0
 1 m                  0.0%        0          0         0
 2 f        375     100.0%      375        375         0
 3 m                  0.0%        0          0         0
 4 f        375     100.0%      375        375         0
 5 m                  0.0%        0          0         0
 6 F        375     100.0%      375        375         0
 7 m                  0.0%        0          0         0
 8 f        375     100.0%      375        375         0
 9 m                  0.0%        0          0         0
10 f        375     100.0%      375        375         0
11 m                  0.0%        0          0         0
12 f        375     100.0%      375        375         0
13 m                  0.0%        0          0         0
14 f        375     100.0%      375        375         0
15 m                  0.0%        0          0         0
0 dirty, 3000 queued, 3000 total, 4096 hash buckets, 2048 buffer size
start clean at  60.000% (of pair total) dirty, or 226 buffs dirty, stop at
  50.000%
Buffer pool page size: 8192
    4 buffer LRU queue pairs             priority levels
# f/m   pair total     % of    length       LOW      HIGH
 0 F        250     100.0%      250        250         0
 1 m                  0.0%        0          0         0
 2 f        250     100.0%      250        250         0
 3 m                  0.0%        0          0         0
 4 f        250     100.0%      250        250         0
 5 m                  0.0%        0          0         0
 6 f        250     100.0%      250        250         0
 7 m                  0.0%        0          0         0
0 dirty, 1000 queued, 1000 total, 1024 hash buckets, 8192 buffer size
start clean at  60.000% (of pair total) dirty, or 150 buffs dirty, stop at
  50.000%
```

图 333: onstat -R 命令输出

输出描述

Buffer pool page size

以字节表示的缓冲池的页大小

#

队列号

每个 LRU 队列是由两个子队列组成：FLRU 队列和 MLRU 队列。（有关 FLRU 和 MLRU 队列的定义，请参阅《*SinoDB*® 管理员指南》的“共享内存”一章中的 LRU 队列。）因此，队列 0 和 1 属于第一个 LRU 队列，队列 2 和 3 属于第二个 LRU 队列，依此类推。

f/m

标识队列类型

此字段有 4 个可能值：

f

可用 LRU 队列

可用意味着未修改过。尽管 LRU 队列中几乎所有缓冲区是可用的，但数据库服务器会尝试使用 FLRU 队列而不是 MLRU 队列的缓冲区。（在数据库服务器使用已修改的缓冲区之前，必须将其写入磁盘。）

F

具有最少元素的可用 LRU

数据库服务器使用此估计来确定接下来将未修改（可用）缓冲区置于何处。

m

MLRU 队列

M

清空程序正在清除的 MLRU 队列

length

跟踪队列的长度（以缓冲区度量）

% of

显示组成 LRU 队列的子队列的百分比

例如：假设 LRU 队列有 50 个缓冲区，其中 30 个在 MLRU 队列中，20 个在 FLRU 队列中。% of 列将分别列出 60.00 和 40.00 的百分率。

pair total

提供 LRU 队列中缓冲区的总数

priority levels

显示优先级级别：LOW、MED_LOW、 MED_HIGH、HIGH

onstat -R 命令也列出优先级级别。

摘要信息在单独的 LRU 队列信息后面。可如下解释摘要信息：

dirty

所以 LRU 队列中已修改缓冲区的总数

queued

LRU 队列中缓冲区的总数

total

缓冲区的总数

hash buckets

散列存储区的数量

buffer size

每个缓冲区的大小

start clean

在 BUFFERPOOL 配置参数的 lru_max_dirty 字段中指定的值

stop at

在 BUFFERPOOL 配置参数的 lru_min_dirty 字段中指定的值

priority downgrades

已降级为较低优先级的 LRU 队列的数量

priority upgrades

已升级为较高优先级的 LRU 队列的数量

onstat -s 命令：打印锁存器的信息

使用 onstat -s 命令显示一般锁存器信息，其包括锁存器控制的资源。
语法：
onstat -s

示例输出

```
Sinoregal DS Version 12.10.F         -- On-Line (CKPT REQ)
-- Up 00:19:21 -- 167608 Kbytes
Blocked:CKPT

Latches with lock or userthread set
name     address           lock wait userthread
physlog  4410d5a8          0    0    45a616b8
```

图 335: onstat -s 命令输出

输出描述

name

使用以下缩写标识锁存器所控制的资源：

archive

存储空间备份

bf

缓冲区

bh

散列缓冲区

chunks

块表

ckpt

检查点

dbspace

数据库空间表

flushctl

页清空程序控制

flushr

页清除程序

locks
锁表
loglog
逻辑日志
LRU
LRU 队列
physb1
第一个物理日志缓冲区
physb2
第二个物理日志缓冲区
physlog
物理日志
pt
表空间 tblspace
tblsps
表空间表
users
用户表

address
锁存器地址
如果线程正在等待锁存器，那么这个地址出现在 onstat -u（用户）命令输出的 wait 字段中。

lock
指示锁存器是否已锁定并设置
指示锁状态的代码（1 或 0）取决于计算机。

wait
正在等待锁存器的用户线程的共享内存地址。如果没有用户线程正在等待，那么将显示空白。

userthread
持有锁存器的用户线程的共享内存地址
该地址对应到 onstat -g ath 输出的 tcb 列中的值。可将该地址与 onstat -u 输出中的用户地址比较以获取用户进程标识号。

onstat -t and onstat -T 命令：打印表空间信息

使用 onstat -t 命令显示活动表空间的表空间信息。使用 onstat -T 命令显示所有表空间的表空间信息。

onstat -t 命令也列出活动表空间的数量和表空间的总数。

语法：
onstat { -t | -T }

示例输出

```
Tblspaces
 n address  flgs ucnt tblnum   physaddr         npages nused  npdata nrows  nextns
62 a40dc70  0    1    100001   1:14             250    250    0      0      1
195 ac843e0  0    1    1000df   1:236             16     9      4      53     2
```

```
 2 active, 221 total
```

图 337: onstat -t 命令输出

输出描述

n

打开的表空间的计数器

address

共享内存表空间表中表空间的地址

flgs

使用以下标志位描述标志：

0x00000002

在下一个检查点清空分区信息

0x00000004

正在删除分区

0x00000008

分区是伪分区（sysmaster）

0x00000020

ALTER TABLE 正在进行中

0x00000040

已删除分区

0x00000800

Partition is the system temp table.

0x00001000

分区是用户临时表

0x00008000

正在进行联机索引创建或删除

0x00400000

请求单个用户对分区访问

0x00800000

已完成删除分区

0x40000000

清空分区信息。分区清空可延迟到检查点进程后期。

ucnt

使用计数，其指示当前访问表空间的用户线程数

tblnum

以十六进制值表示的表空间号

等价的整数显示为 systables 系统目录表中的 partnmn 值。

physaddr

表空间在磁盘上的物理地址

npages

分配给表空间的页数

nused

表空间中已使用页的数量

npdata

已使用数据页的数量

nrows

已使用数据行的数量

nextns

已分配非连续扩展数据块的数量

该数字与已分配下一个扩展数据块的次数不相同。

onstat -u 命令：显示用户活动的概要文件

使用 onstat -u 命令显示用户活动的概要文件。
语法：

```
onstat   -u
```

示例输出

```
Userthreads
address  flags   sessid   user     tty      wait     tout locks nreads   nwrites
a4d8018  ---P--D 1        informix -        0        0    0     58       4595
a4d8628  ---P--F 0        informix -        0        0    0     0        2734
a4d8c38  ---P--- 5        informix -        0        0    0     0        1
a4d9248  ---P--B 6        informix -        0        0    0     40       0
a4d9858  ---P--D 7        informix -        0        0    0     0        0
a4d9e68  Y--P--- 21       niraj    -        a65e5a8  0    1     0        0
 6 active, 128 total, 7 maximum concurrent
```

图 339: onstat -u 命令输出

输出描述

-u 选项对每个用户线程提供以下输出。

address

用户表中用户线程的共享内存地址

比较该地址与显示在 -s 选项（锁存器）、-b、-B 和 -X 选项（缓冲区），以及 -k 选项（锁）输出的地址，以了解该线程正持有或等待什么资源。

flags

提供会话的状态

位置 1 的标志代码：

B
: 正在等待缓冲区

C
: 正在等待检查点

G
: 正在等待逻辑日志缓冲区的写入

L
: 正在等待锁

S
: 正在等待互斥

T
正在等待事务
Y
正在等待条件
X
正在等待事务清除（回滚）
DEFUNCT
该线程已引起严重断言失败，并已暂挂以允许其他线程继续其工作。

位置 2 的标志代码：

*
I/O 失败过程中事务处于活动状态

位置 3 的标志代码：

A
数据库空间备份线程

有关此处显示的其他值，请参阅 -x 选项的位置 3 的标志代码。

位置 4 标志代码：

P
会话的主线程

位置 5 的标志代码：

R
正在读取
X
临界段中的线程

位置 6 的标志代码：

R
用于恢复（例如：物理或逻辑恢复）的线程
-
不是用户恢复的线程

位置 7 的标志代码：

B
B 型树清除程序线程
C
已终止正在等待清除的用户线程
D
守护程序线程
F
页清除程序线程

sessid

会话标识号

在操作（例如：并行排序和并行索引构建）过程中，会话可能有许多与其关联的用户线程。对于这样的原因，会话 ID 用来标识每个唯一的会话。

user

用户登录名称（由操作系统派生）

tty

用户正在使用的标准错误（stderr）文件的名称（由操作系统派生）

在 Windows™ 上,此字段为空白。

wait

如果用户线程正在等待特定锁存器、锁、互斥或条件，那么这个字段显示该资源的地址。使用该地址来映射到 -s（锁存器）选项或 -k（锁）选项输出中提供的信息。如果是用于持久条件，请对 onstat -a 命令输出中的地址运行 grep。

tout

当前等待中的剩余秒数

如果值是 0，那么用户线程没有在等待锁存器或锁。如果值是 -1，那么用户线程是处于无限期等待中。

locks

用户线程正持有的锁数

-k 输出应包含每个持有锁的列表。

nreads

用户线程已执行的磁盘读取数

nwrites

用户线程已执行的写调用数

所有写调用都写入共享内存缓冲区高速缓存。

onstat -u 命令输出的最后一行显示自数据库服务器初始化以来分配的并发用户线程的最大数量。例如：样本 onstat -u 命令输出的最后一行如下：

```
4 active, 128 total, 17 maximum concurrent
```

该行的最后部分 17 maximum concurrent 指示自数据库服务器初始化以来并发运行的用户线程的最大数为 17。

该输出还指示活动用户数和允许的最大用户数。

onstat -x 命令：显示数据库服务器事务信息

使用 onstat -x 命令显示数据库服务器上事务信息。
语法：
onstat -x

仅在以下情况下，才需要事务信息：

- X/Open 环境
- 数据库服务器参与分布式查询
- 数据库服务器使用 Microsoft™ Transaction Server（MTS）事务管理器

示例输出

```
Transactions

address   flags userthread  locks  begin    current     isol   est.     retrys coord
                              logpos   logpos            rb_time
a6d8028   A---- a695028     0      -        -         COMMIT  -        0
a6d8348   A---- a695878     0      -        -         COMMIT  -        0
a6d8668   A---- a6960c8     0      -        -         COMMIT  -        0
a6d8988   A---- a696918     0      -        -         COMMIT  -        0
```

```
a6d8fc8   A---- a698208     0      -          -          COMMIT  -        0
a6d92e8   A---- a6979b8     0      -          -          COMMIT  -        0
a6d9608   A---- a698a58     0      -          -          COMMIT  -        0
a6d9928   A---- a6992a8     1      -          -          DIRTY   -        0
a6d9c48   A---- a6992a8     0      -          -          NOTRANS -        0
a6d9f68   A---- a69a348     0      -          -          COMMIT  -        0
a6da288   A---- a69ab98     0      -          -          COMMIT  -        0
a6da5a8   A---- a69b3e8     0      -          -          COMMIT  -        0
a6da8c8   A---- a69bc38     0      -          -          COMMIT  -        0
a6dabe8   A---- a69c488     0      -          -          COMMIT  -        0
a6daf08   A---- a699af8     0      -          -          COMMIT  -        0
a6db228   A---- a6992a8     0      -          -          COMMIT  -        0
a6db548   A---- a69ccd8     1      -          -          DIRTY   -        0
a6db868   A---- a69d528     1      -          -          DIRTY   -        0
a6dbb88   A---- a69ccd8     0      -          -          COMMIT  -        0
a6dbea8   A---- a69dd78     0      -          -          COMMIT  -        0
a6dc1c8   A---- a69e5c8     0      -          -          COMMIT  -        0
a6dc4e8   A-B-- a69ee18   502    33:0x25018  34:0x486fc COMMIT  0:07     0
 22 active, 128 total, 23 maximum concurrent
```

图 341: onstat -x 命令输出

输出描述

可以如下解释 onstat -x 输出：

address

事务结构的共享内存地址

flags

位置 1 的标志代码（当前事务状态）：

A

连接事务的用户线程

S

TP/XA 暂挂事务

C

TP/XA 正等待回滚

位置 2 的标志代码（事务方式）：

T

紧耦合方式（MTS）

L

松耦合方式（缺省方式）

位置 3 的标志代码（事务阶段）：

B

开始工作

P

已准备好提交的分布式查询

X

已准备好提交的 TP/XA

C

正在提交或已提交

R

正在回滚或已回滚

H
正尝试回滚或已回滚

位置 4 的标志代码：

X
XA 事务

位置 5 的标志代码（事务类型）：

G
全局事务

C
分布式查询协调者

S
分布式查询从属者

B
分布式查询协调者和从属者

M
重定向全局事务

userthread
拥有事务的线程（rstcb 地址）

locks
事务持有的锁数

begin logpos
BEGIN WORK 记录在日志中的位置

current logpos
事务写入的最近记录的当前日志位置（当事务回滚时，当前日志位置将回退到起始日志位置。当到达起始日志位置时，回滚完成。）

isol
隔离级别

est. rb time
服务器回滚事务所需的估计时间。随着事务前进，该时间增长。如果事务回滚，那么该时间随着事务回退而减少。

retrys
尝试启动分布区查询的恢复线程的次数

coord
从属者正在执行事务时事务协调者的名称

此字段告诉您哪个数据库服务器正在协调两阶段提交。

onstat -x 命令输出的最后一行指示自数据库服务器初始化以来最大并发事务数是 8。

```
8 active, 128 total, 8 maximum concurrent
```

确定逻辑日志记录的位置

使用 onstat -x 命令确定逻辑日志记录的位置。

curlog 和 logposit 字段提供逻辑日志记录的确切位置。如果事务没在回滚，那么 curlog 和 logposit 描述最新写入的日志记录的位置。当事务正在回滚时，这些字段描述最新“撤销”日志记录的位置。随着事务回滚，curlog 和 logposit 的值下降。在长事务中，curlog 和 logposit 的聚集率可帮助您估计回滚还将花费多久时间。

有关 onstat -x 命令示例，请参阅《*SinoDB*® 管理员指南》的“多阶段提交协议”一章中的“监视全局事务”。

确定全局事务的方式

onstat -x 命令对于确定全局事务是以松耦合方式还是紧耦合方式执行很有用。

onstat -x 输出中 flags 列的第二个位置显示全局事务的标志。T 标志指示紧耦合方式，L 标志指示松耦合方式。

- 松耦合方式 意味着不同数据库服务器协调事务但不共享锁。全局事务中的每个分支都有独立事务 XID. 在逻辑日志中，所有分支的记录都显示为独立事务。
- 紧耦合方式 意味着不同数据库服务器协调事务并共享诸如锁定和日志记录之类的资源。在全局事务中，访问同一数据库的所有分支共享同一事务 XID。具有相同 XID 的分支的日志记录显示在同一会话 ID 下面。MTS 使用紧耦合方式。

onstat -X 命令：显示线程信息

使用 onstat -X 命令获取有关正等待缓冲区的线程的确切信息。

对于每个使用中的缓冲区，onstat -X 命令显示一般的缓冲区信息，这些信息也可使用 onstat -b 或 onstat -B 命令获得。有关更多信息，请参阅 *onstat -b* 命令: 显示正在使用的缓冲区的信息 在第405页中的 onstat -b 命令。

语法：
onstat -X

示例输出

```
    Buffers (Access)
address  owner    flags pagenum          memaddr  nslots pgflgs scount   waiter
    Buffer pool page size: 2048
 0 modified, 3000 total, 4096 hash buckets, 2048 buffer size
    Buffer pool page size: 8192
 0 modified, 1000 total, 1024 hash buckets, 8192 buffer size
```

图 343: onstat -X 命令输出

输出描述

onstat -X 命令的 waiter 字段列出正在等待缓冲区的所有用户线程，而 onstat -b 和 onstat -B 包含 waitlist 字段显示正在等待缓冲区的第一个用户线程的地址。共享缓冲区的最大数量以 ONCONFIG 文件中 BUFFERPOOL 配置参数的 buffers 字段进行指定。

Buffer pool page size
: 以字节表示缓冲池页的大小

address
: 缓冲区表中缓冲区头的地址

flags
: 标识缓冲区页的当前状态的标志：

  0x01
  : 已修改的数据

  0x02
  : 数据

0x04
LRU
0x08
错误
0x10
共享锁
0x20
正在进行 LRU AIO 写入
0x40
正在进行块写入
0x80
互斥锁
0x100
分配给 LRU 的清除程序
0x200
缓冲区应避免 bf_check 调用
0x400
在写入页之前执行日志清空
0x800
缓冲区已进行缓冲区检查
0x8000
缓冲区已固定

pagenum
磁盘上物理页号

memaddr
缓冲区内存地址

nslots
页中槽表条目的数量

此字段指示存储于页的行（或行的部分）的数量。

pgflgs
使用以下值（单独或组合）来描述页类型：

1
数据页
2
表空间页
4
可用列表页
8
块可用列表页
9
剩余数据页
b
分区常驻 Blob 页

c
Blob 空间常驻 Blob 页
d
Blob 块可用列表位页
e
Blob 块 blob 图页
10
B 型树节点页
20
B 型树根节点页
40
B 型树分支节点页
80
B 型树叶节点页
100
逻辑日志页
200
逻辑日志的最后一页
400
逻辑日志的同步页
800
物理日志
1000
保留根页
2000
无需物理日志
8000
具有缺省标志的B 型树的叶

scount
显示正等待该缓冲区的线程数

waiter
列出正在等待该缓冲区的所有用户线程的地址

onstat -z 命令：清除统计信息

使用 onstat -z 命令清除数据库服务器统计信息（包括与 Enterprise Replication 有关的统计信息），并将概要文件计数设置为 0。

如果使用 onstat -z 命令来重设并监视一些字段的计数，那么应了解概要文件计数对于数据库服务器管理的任何数据库中发生的所有活动都是增加的。任何用户可以重设概要文件计数，从而干扰另一用户正在执行的监视。

语法：
```
onstat  -z
```

退出 onstat 实用程序时的返回码

当退出 onstat 实用程序时，该程序会显示一组返回码。

示例

以下是当您退出 onstat 实用程序时，显示的返回码和消息的几行示例：

```
GLS failures: -1
Failed to attach shared memory: -1
Failed to attach shared memory when running 'onstat -': 255
All other errors detected by onstat: 1
No errors detected by onstat: 0
Administration mode: 7
```

返回码值

下表列出与退出 onstat 实用程序时显示的返回码相应的数据库服务器模式。

值	解释
-1	GLS 本地语言环境初始化失败或 SinoDB® 连接共享内存失败
0	初始化模式
1	静默模式
2	恢复模式
3	备份模式
4	关闭模式
5	联机模式
6	中止模式
7	用户模式
255	脱机模式

第 III 部分

SQL 管理 API

SQL 管理 API 函数

这些主题描述 SQL 管理 API admin() 和 task() 函数。

SQL 管理 API 概述

使用 SQL 管理 API 来通过 SQL 语句远程地管理 SinoDB®。

SQL 管理 API 由两个函数组成：admin() 和 task()。这些函数执行相同的操作，但返回不同格式的结果。这些函数使用一个或多个定义操作的参数。许多操作也可使用命令行实用程序完成。使用 SQL 管理 API 函数的好处是可以远程地从其他数据库服务器运行它们；反之，您必须直接连接到要运行命令行实用程序命令的数据库服务器。

您可在 SQL 语句内调用 admin() 和 task()，该语句可包含表达式，或者您可使用 EXECUTE FUNCTION 语句来调用它们。在不包含任何其他语句的事务内运行 admin() 或 task() 函数。

SQL 管理 API 函数定义在 sysadmin 数据库中。您必须直接或远程地连接到 sysadmin 数据库以运行这些函数。

仅以下用户可以运行 SQL 管理 API 函数：

- 用户 informix
- root 用户，如果将 sysadmin 数据库上的 Connect 权限授予该用户
- DBSA 组成员，如果将 sysadmin 数据库上的 Connect 权限授予该角色
- 通过带有 grant admin 参数的 admin() 和 task() 函数授予 SQL 管理 API 命令权限的用户。

您可以生成用于复制文件中存在的存储空间、块和日志的 SQL 管理 API 命令。要执行这个，请运行带有 -c 选项的 dbschema 实用程序。

相关链接

《SinoDB 迁移指南》：存储空间、块和日志的创建

admin() 和 task() 函数语法行为

admin() 和 task() 函数带有一个或多个参数，这些参数是以逗号分隔的带引用号的字符串。

admin() 和 task() 函数的语法包括以下规则：

- 每个参数必须用一对单引号（'）或双引号（"）界定。
- 参数必须用逗号分隔。
- 最多 28 个参数。
- 大多数参数不区分大小写，但以下例外：

- 紧跟在初始 onmode 参数后的参数区分大小写：

```
EXECUTE FUNCTION task("onmode","D","50");
```

- 与 cdr 包含在一起的参数区分大小写。

  例如：

```
EXECUTE FUNCTION task("cdr define server",
   "-c=g_amsterdam","--init g_amsterdam");
```

- 如果不是直接连接到 sysadmin 数据库，那么根据标准“数据库对象名称”语法必须包含 sysadmin 数据库名称和服务器名称。例如：假设服务器名称是 ids_server，则可运行下面的语句：

```
EXECUTE FUNCTION sysadmin@ids_server:admin("add bufferpool","2",
"50000","8","60.0","50.0");
```

  有关“数据库对象名称”语法的更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。

admin() 和 task() 参数大小规范

在缺省情况下，在 admin() 和 task() 函数中指定大小的参数的单位是 KB。您可以指定其他单位。

您可以在 admin() 和 task() 函数的大小参数中使用以下单位：

B
: 字节

K
: KB（缺省）

M
: MB

G
: GB

T
: TB

P
: PB

这些字符的字母大小写是忽略的。

参数中分隔大小规范和单位缩写的任何空白是忽略的。例如：规范 "128M" 和 "128 m" 都解释为 128 MB。

当省略大小参数时，则根据配置参数的设置或根据系统缺省（如果未设置参数）来应用对象的缺省大小。例如：存储空间的缺省大小是 100 MB。

admin() 和 task() 函数返回码

admin() 和 task() 函数执行相同任务但产生不同的返回码类型。如果您想要一个整数返回码，则使用 admin() 函数，或者如果您想要一个文本返回码，则使用 task() 函数。

当运行 admin() 或 task() 函数时，函数：

- 执行指定的操作。
- 返回一个表示函数成功还是失败的值。
- 在 sysadmin 数据库的 command_history 表中插入一行。

admin() 和 task() 的返回码以不同格式指示函数是成功还是失败：

- task() 函数返回一个文本消息。该消息也会插入到 task() 插入 command_history 表中的那个新行的 cmd_ret_msg 列中。

- admin() 函数返回一个整数。该数字也会插入到 admin() 插入 command_history 表的那个新行的 cmd_number 列中。
  - 如果该值大于 0，则函数是成功的，并且在 command_history 表中插入一新行。
  - 如果该值是 0，则函数是成功的，但是 SinoDB® 不会在 command_history 插入一新行。
  - 如果该值小于 0，则函数是失败的，但会在 command_history 表中插入一新行。

admin() 或 task() 函数指定的操作发生在一个单独事务中，与在 command_history 表中插入新行的事务分开。如果命令执行成功，但 command_history 表的插入失败，那么命令会生效，但在 online.log 会有一条错误条目指明问题。

当调用函数时，如果 command_history.cmd_number 序列计数器是 200，且命令成功，那么 SinoDB® 会执行该命令并返回整数 201。如果命令失败，则此示例返回值 -201。

假设 task() 函数已执行了同一的命令：

```
EXECUTE FUNCTION task("check extents");
```

这个命令指示数据库服务器检查扩展数据块，并返回一个消息表示此命令是成功还是失败。

当调用函数时，如果 command_history.cmd_number 序列计数器是 201，且该命令失败，则返回值 -202。假设 DBSA 下一个调用的 SQL 管理 API 函数是：

```
EXECUTE FUNCTION admin('create dbspace',
   'dbspace2', '/work/CHUNKS/dbspace2', "20M");
```

如果此示例中的命令成功，则返回值 203。DBSA 使用以下查询来检查这些调用 admin() 函数时插入 command_history 表中的两行：

```
SELECT * FROM command_history WHERE cmd_number IN (202,203);
```

这个查询返回两行：

```
cmd_number      202
cmd_exec_time   2009-04-17 16:26:14
cmd_user        informix
cmd_hostname    olympia
cmd_executed    create dbspace
cmd_ret_status  -1
cmd_ret_msg     Unable to create file /work/dbspace2

cmd_number      203
cmd_exec_time   2009-04-17 16:26:15
cmd_user        informix
cmd_hostname    olympia
cmd_executed    create dbspace
cmd_ret_status  0
cmd_ret_msg     created dbspace number 2 named dbspace2
```

SQL 管理 API 门户：按权限组划分参数

您可查看 admin() 和 task() 函数按权限组排序的参数列表，并带有有关参数信息的链接。

权限组标识用户可运行哪些 SQL 管理 API 命令。某些函数参数在多个权限组中。将权限组授予用户，以便他们可以运行工作所需的命令。在缺省情况下，仅用户 informix 或 DBSA 可运行 SQL 管理 API 命令。

使用 grant admin 参数授予权限，和使用 revoke admin 参数撤销权限。

- ADMIN：用户可运行所有 SQL 管理 API 函数。

- *BAR 权限组* 在第574页：用户可运行备份和复原函数。
- *FILE 权限组* 在第574页：用户可管理消息日志并显示文件信息。
- *GRANT 权限组* 在第574页：用户有授予和撤销权限的权限。
- *HA 权限组* 在第575页：用户可运行高可用性函数。
- *MISC 权限组* 在第575页：用户可管理数据库服务器。
- *MONITOR 权限组* 在第578页：用户可运行仅显示信息的所有 SQL 管理 API 函数。
- OPERATOR：用户可运行所有 SQL 管理 API 函数，除了 GRANT 权限组中函数。
- *REPLICATION 权限组* 在第579页：用户可运行 Enterprise Replication cdr 实用程序函数。
- *SQL 权限组* 在第579页：用户可运行与管理数据库的 SQL 语句有关的函数。
- *SQLTRACE 权限组* 在第579页：用户可运行 SQL 跟踪函数。
- *STORAGE 权限组* 在第580页：用户可运行空间相关的函数。
- *TENANT 权限组* 在第583页：用户可运行租户数据库函数。
- WAREHOUSE：用户可运行仓库加速器管理工具。请参阅 《*SinoDB*® 仓库加速器管理指南》中的管理 *SinoDB*® 仓库加速器的许可。

BAR 权限组

BAR 权限组包括备份数据库的 SQL 管理 API 参数

表 164: 备份和复原的 admin() 和 task() 函数参数

参数
archive fake 参数: 执行无记录的备份 (SQL 管理 API) 在第589页
ontape archive 参数: 备份数据库上的数据 (SQL 管理 API) 在第670页
onbar 参数: 备份存储空间 (SQL 管理 API) 在第650页
onsmsync 参数: 与存储管理器目录同步 (SQL 管理 API) 在第668页

FILE 权限组

FILE 权限组包括管理消息日志和显示文件信息的 SQL 管理 API 参数。

表 165: 消息日志命令的 admin() 和 task() 函数参数

参数
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页
print file info 参数: 显示目录或文件信息 (SQL 管理 API) 在第671页

GRANT 权限组

GRANT 权限组包括将运行 SQL 管理 API 命令的权限授予其他用户或撤销的 SQL 管理 API 函数。

表 166: 授予和撤销权限的 admin() 和 task() 函数参数

参数
grant admin 参数: 授予运行 SQL 管理 API 命令的权限 在第630页

参数
revoke admin 参数: 撤销运行 *SQL* 管理 *API* 命令的权限 在第676页

HA 权限组

HA 权限组包括管理高可用性集群的 SQL 管理 API 函数。

表 167: 高可用性集群命令的 admin() 和 task() 函数参数

参数
ha make primary 参数: 更改辅助服务器的模式 *(SQL* 管理 *API)* 在第631页
ha rss 参数: 创建 *RS* 辅助服务器 *(SQL* 管理 *API)* 在第631页
ha rss add 参数: 添加 *RS* 辅助服务器到主服务器 *(SQL* 管理 *API)* 在第632页
ha rss change 参数: 更改 *RS* 辅助服务器的密码 *(SQL* 管理 *API)* 在第633页
ha rss delete 参数: 删除 *RS* 辅助服务器 *(SQL* 管理 *API)* 在第633页
ha sds clear 参数: 停止共享磁盘复制 *(SQL* 管理 *API)* 在第634页
ha sds primary 参数: 将 *SD* 辅助服务器转换为主服务器 *(SQL* 管理 *API)* 在第635页
ha sds set 参数: 创建共享磁盘主服务器 *(SQL* 管理 *API)* 在第635页
ha set idxauto 参数: 复制索引到辅助服务器 *(SQL* 管理 *API)* 在第636页
ha set ipl 参数: 在主服务器上构建日志索引 *(SQL* 管理 *API)* 在第637页
ha set primary 参数: 定义 *HDR* 主服务器 *(SQL* 管理 *API)* 在第637页
ha set secondary 参数: 定义 *HDR* 辅助服务器 *(SQL* 管理 *API)* 在第638页
ha set standard 参数: 将 *HDR* 服务器转换成为标准服务器 *(SQL* 管理 *API)* 在第638页
ha set timeout 参数: 更改 *SD* 辅助服务器超时 *(SQL* 管理 *API)* 在第639页
onmode 和 *d* 参数: 设置数据复制类型 *(SQL* 管理 *API)* 在第654页

MISC 权限组

MISC 权限组包括管理数据库服务器的 SQL 管理 API 函数参数:

- *onstat* 在第575页
- 配置参数 在第576页
- 数据、分区和扩展数据块 在第576页
- 监听线程 在第576页
- 消息日志 在第577页
- 内存 在第577页
- *PDQ* 在第577页
- 服务器模式 在第578页
- *SQL* 语句高速缓存 在第578页
- 其他管理任务 在第578页

onstat

通过运行 onstat 命令监视数据库服务器的 SQL 管理 API 函数参数。

表 168: onstat 命令的 admin() 和 task() 函数参数

参数
onstat 参数: 监视数据库服务器 (SQL 管理 API) 在第669页

配置参数

更新配置参数的 SQL 管理 API 函数参数。

表 169: 配置参数命令的 admin() 和 task() 函数参数

参数
export config 参数: 导出配置参数值 (SQL 管理 API) 在第628页
import config 参数: 导入配置参数值 (SQL 管理 API) 在第640页
modify config 参数 : 更改配置参数 (SQL 管理 API) 在第647页
onmode 和 wf 参数: 永久地更新配置参数 (SQL 管理 API) 在第664页
onmode 和 wm 参数: 临时地更新配置参数 (SQL 管理 API) 在第664页
onmode、wm 和 AUTO_LRU_TUNING 参数: 更改 LRU 调整状态 (SQL 管理 API) 在第665页
reset config 参数: 恢复配置参数值 (SQL 管理 API) 在第674页
reset config all 参数: 恢复所有动态地可更新的配置参数值 (SQL 管理 API) 在第674页
set onconfig memory 参数: 临时地更改配置参数 (SQL 管理 API) 在第682页
set onconfig permanent 参数: 永久地更改配置参数 (SQL 管理 API) 在第683页

数据、分区和扩展数据块

管理数据、分区和扩展数据块的 SQL 管理 API 函数参数。

表 170: 数据、分区和扩展数据块命令的 admin() 和 task() 函数参数

参数
check data 参数: 检查数据一致性 (SQL 管理 API) 在第600页
check extents 参数: 检查扩展数据块一致性 (SQL 管理 API) 在第601页
check partition 参数: 检查分区一致性 (SQL 管理 API) 在第601页
checkpoint 参数: 强制检查点 (SQL 管理 API) 在第602页
create dbaccessdemo 参数: 创建演示数据库 (SQL 管理 API) 在第608页
onmode 和 C 参数: 控制 B 型树扫描程序 (SQL 管理 API) 在第653页
onmode 和 c 参数: 强制检查点 (SQL 管理 API) 在第652页
print partition 参数: 打印分区信息 (SQL 管理 API) 在第672页
set dataskip 参数: 启动或停止跳过数据库空间 (SQL 管理 API) 在第680页
set index compression 参数: 更改索引页压缩 (SQL 管理 API) 在第681页

监听线程

不中断现有的连接而控制 SOCTCP 或 TLITCP 网络协议的监听线程的 SQL 管理 API 函数参数。

表 171: 监听线程命令的 admin() 和 task() 函数参数

参数
restart listen 参数: 动态地停止并启动监听线程 (SQL 管理 API) 在第675页
start listen 参数: 动态地启动监听线程 (SQL 管理 API) 在第690页
stop listen 参数: 动态地停止监听线程 (SQL 管理 API) 在第692页

消息日志

管理消息日志的 SQL 管理 API 函数参数。

表 172: 消息日志命令的 admin() 和 task() 函数参数

参数
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页

内存

管理内存的 SQL 管理 API 函数参数。

表 173: 内存命令的 admin() 和 task() 函数参数

参数
add bufferpool 参数: 添加缓冲池 (SQL 管理 API) 在第583页
add memory 参数: 增加共享内存 (SQL 管理 API) 在第586页
onmode 和 a 参数: 添加共享内存段 (SQL 管理 API) 在第651页
onmode 和 F 参数: 释放未使用的内存段 (SQL 管理 API) 在第656页
onmode 和 n 参数: 解锁驻留内存 (SQL 管理 API) 在第659页
onmode 和 r 参数: 强制共享内存的驻留 (SQL 管理 API) 在第662页
scheduler lmm enable 参数: 指定自动低内存管理设置 (SQL 管理 API) 在第677页
scheduler lmm disable 参数: 停止自动低内存管理 (SQL 管理 API) 在第679页

PDQ

管理 PDQ 的 SQL 管理 API 函数参数。

表 174: PDQ 命令的 admin() 和 task() 函数参数

参数
onmode 和 D 参数: 设置 PDQ 优先级 (SQL 管理 API) 在第655页
onmode 和 M 参数: 临时更改决策支持内存 (SQL 管理 API) 在第658页
onmode 和 Q 参数: 设置决策支持查询的最大数目 (SQL 管理 API) 在第661页

参数
onmode 和 S 参数: 设置决策支持扫描的最大数目 (SQL 管理 API) 在第662页

服务器模式

更改服务器模式的 SQL 管理 API 函数参数。

表 175: 服务器模式命令的 admin() 和 task() 函数参数

参数
onmode 和 j 参数: 切换数据库服务器到管理模式 (SQL 管理 API) 在第657页
onmode 和 m 参数: 切换到多用户模式 (SQL 管理 API) 在第658页

SQL 语句高速缓存

管理 SQL 语句高速缓存的 SQL 管理 API 函数参数。

表 176: SQL 语句高速缓存命令的 admin() 和 task() 函数参数

参数
onmode 和 e 参数: 更改 SQL 语句高速缓存的使用 (SQL 管理 API) 在第655页
onmode 和 W 参数: 重设语句高速缓存属性 (SQL 管理 API) 在第663页

其他管理任务

管理其他管理任务的 SQL 管理 API 函数参数。

表 177: 其他管理任务命令的 admin() 和 task() 函数参数

参数
alter logmode 参数: 更改数据库日志记录模式 (SQL 管理 API) 在第588页
create dbaccessdemo 参数: 创建演示数据库 (SQL 管理 API) 在第608页
onmode 和 e 参数: 更改 SQL 语句高速缓存的使用 (SQL 管理 API) 在第655页
onmode 和 l 参数: 切换到下一个逻辑日志 (SQL 管理 API) 在第657页
onmode 和 p 参数: 添加或移除虚拟处理器 (SQL 管理 API) 在第660页
onmode 和 Y 参数: 更改会话的查询计划度量 (SQL 管理 API) 在第666页
onmode 和 z 参数: 终止用户会话 (SQL 管理 API) 在第667页
onmode 和 Z 参数: 终止分布式事务 (SQL 管理 API) 在第667页
print error 参数: 打印错误消息 (SQL 管理 API) 在第671页
reset sysadmin 参数: 移动 sysadmin 数据库 (SQL 管理 API) 在第675页
scheduler 参数: 停止或启动调度程序 (SQL 管理 API) 在第677页

MONITOR 权限组

MONITOR 权限组包括监视消息日志、Enterprise Replication 和压缩估算的 SQL 管理 API 函数参数。

表 178: 监视消息日志、Enterprise Replication 或压缩估算的 admin() 和 task() 函数参数

参数
cdr error, cdr finderr, cdr list repair, cdr list replicate, cdr list replicateset, cdr list server, cdr list template, cdr stats recv, and cdr stats rqm arguments
cdr 参数: 管理 Enterprise Replication (SQL 管理 API) 在第594页
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页
index estimate_compression 参数: 估计索引压缩 (SQL 管理 API) 在第642页
print error 参数: 打印错误消息 (SQL 管理 API) 在第671页
onstat 参数: 监视数据库服务器 (SQL 管理 API) 在第669页
table estimate_compression and fragment estimate_compression arguments
table 或 fragment 参数: 压缩数据和优化存储 (SQL 管理 API) 在第699页

REPLICATION 权限组

REPLICATION 权限组包括管理 Enterprise Replication 的 SQL 管理 API 函数参数。

表 179: Enterprise Replication 命令的 admin() 和 task() 函数参数

参数
cdr 参数: 管理 Enterprise Replication (SQL 管理 API) 在第594页

SQL 权限组

SQL 权限组包括创建和删除数据库以及查看错误消息的 SQL 管理 API 函数参数。

表 180: 数据库和错误消息的 admin() 和 task() 函数参数

参数
create database 参数: 创建数据库 (SQL 管理 API) 在第607页
create dbaccessdemo 参数: 创建演示数据库 (SQL 管理 API) 在第608页
drop database 参数: 删除数据库 (SQL 管理 API) 在第623页
print error 参数: 打印错误消息 (SQL 管理 API) 在第671页

SQLTRACE 权限组

SQLTRACE 权限组包括管理 SQL 跟踪的 SQL 管理 API 函数参数。

表 181: SQL 跟踪命令的 admin() 和 task() 函数参数

参数
set sql tracing 参数: 设置全局 SQL 跟踪 (SQL 管理 API) 在第685页
set sql tracing database 参数: 更改数据库跟踪 (SQL 管理 API) 在第686页
set sql tracing session 参数: 控制对会话的跟踪 (SQL 管理 API) 在第687页
set sql tracing user 参数: 控制对用户的跟踪 (SQL 管理 API) 在第688页

参数
set sql user tracing 参数: 设置对用户会话的全局 *SQL* 跟踪 *(SQL* 管理 *API)* 在第688页

STORAGE 权限组

STORAGE 权限组包括管理以下存储方面的 SQL 管理 API 函数参数：

- 自动表存储位置参数 在第580页
- 压缩 在第580页
- 逻辑和物理日志 在第580页
- 镜像 在第581页
- 存储空间 在第581页
- 存储供应 在第582页

自动表存储位置参数

管理存储自动分配分段的数据库空间列表的 SQL 管理 API 函数参数。

表 182: 表存储命令的 admin() 和 task() 函数参数

参数
autolocate database add 参数: 添加数据库空间到数据库空间列表 *(SQL* 管理 *API)* 在第590页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 *(SQL* 管理 *API)* 在第591页
autolocate database 参数: 指定自动定位和分段的数据库空间 *(SQL* 管理 *API)* 在第591页
autolocate database off 参数: 禁用数据库的自动分段 *(SQL* 管理 *API)* 在第592页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 *(SQL* 管理 *API)* 在第593页

压缩

管理数据压缩和优化存储的 SQL 管理 API 函数参数。

表 183: 压缩命令的 admin() 和 task() 函数参数

参数
index compress repack shrink 参数: 优化 *B* 型树索引的存储 *(SQL* 管理 *API)* 在第640页
index estimate_compression 参数: 估计索引压缩 *(SQL* 管理 *API)* 在第642页
table 或 *fragment* 参数: 压缩数据和优化存储 *(SQL* 管理 *API)* 在第699页
purge compression dictionary 参数: 移除压缩字典 *(SQL* 管理 *API)* 在第704页

有关压缩和存储优化命令的概述，请参阅表和分段压缩和解压缩操作 *(SQL* 管理 *API)* 在第698页。

逻辑和物理日志

管理逻辑和物理日志的 SQL 管理 API 函数参数。

表 184: 日志命令的 admin() 和 task() 函数参数

参数
add log 参数: 添加新逻辑日志 *(SQL* 管理 *API)* 在第585页

参数
alter logmode 参数: 更改数据库日志记录模式 *(SQL* 管理 *API)* 在第588页
alter plog 参数: 更改物理日志 *(SQL* 管理 *API)* 在第589页
drop log 参数: 删除逻辑日志 *(SQL* 管理 *API)* 在第624页

镜像

管理镜像的 SQL 管理 API 函数参数。

表 185: 镜像命令的 admin() 和 task() 函数参数

参数
add mirror 参数: 添加镜像块 *(SQL* 管理 *API)* 在第586页
start mirroring 参数: 启动存储空间镜像 *(SQL* 管理 *API)* 在第690页
stop mirroring 参数: 停止存储空间镜像 *(SQL* 管理 *API)* 在第692页

存储空间

管理块、Blob 空间、数据库空间和智能大对象空间的 SQL 管理 API 函数参数。

表 186: 空间命令的 admin() 和 task() 函数参数

参数
add chunk 参数: 添加新块 *(SQL* 管理 *API)* 在第584页
alter chunk 参数: 更改块状态为联机或脱机 *(SQL* 管理 *API)* 在第587页
clean sbspace 参数: 释放未引用的智能大对象 *(SQL* 管理 *API)* 在第603页
create blobspace 参数: 创建 *BLOB* 空间 *(SQL* 管理 *API)* 在第603页
create chunk 参数: 创建块 *(SQL* 管理 *API)* 在第605页
create dbaccessdemo 参数: 创建演示数据库 *(SQL* 管理 *API)* 在第608页
create dbspace 参数: 创建数据库空间 *(SQL* 管理 *API)* 在第608页
create sbspace 参数: 创建智能大对象空间 *(SQL* 管理 *API)* 在第612页
create sbspace with accesstime 参数: 创建跟踪访问时间的智能大对象空间 *(SQL* 管理 *API)* 在第614页
create sbspace with log 参数: 创建具有事务日志记录的智能大对象空间 *(SQL* 管理 *API)* 在第615页
create tempdbspace 参数: 创建临时数据库空间 *(SQL* 管理 *API)* 在第616页
create tempsbspace 参数: 创建临时智能大对象空间 *(SQL* 管理 *API)* 在第617页
drop blobspace 参数: 删除 *Blob* 空间 *(SQL* 管理 *API)* 在第620页
drop chunk 参数: 删除块 *(SQL* 管理 *API)* 在第621页
drop dbspace 参数: 删除数据库空间 *(SQL* 管理 *API)* 在第623页
drop sbspace 参数: 删除智能大对象空间 *(SQL* 管理 *API)* 在第626页
drop tempdbspace 参数: 删除临时数据库空间 *(SQL* 管理 *API)* 在第627页
onmode 和 *O* 参数: 标记禁用的数据库空间为 *down (SQL* 管理 *API)* 在第660页

参数
print error 参数: 打印错误消息 *(SQL* 管理 *API)* 在第671页
rename space 参数: 重命名存储空间 *(SQL* 管理 *API)* 在第673页
set chunk 参数: 更改块的状态 *(SQL* 管理 *API)* 在第680页
set sbspace accesstime 参数: 控制访问时间跟踪 *(SQL* 管理 *API)* 在第683页
set sbspace avg_lo_size 参数: 设置智能大对象的平均大小 *(SQL* 管理 *API)* 在第684页
set sbspace logging 参数: 更改智能大对象空间的日志记录 *(SQL* 管理 *API)* 在第685页

存储供应

管理来自存储池的块、Blob 空间、数据库空间和智能大对象空间的 SQL 管理 API 函数参数。

表 187: 存储供应空间命令的 admin() 和 task() 函数参数

参数
create blobspace from storagepool 参数: 从存储池创建 *BLOB* 空间 *(SQL* 管理 *API)* 在第604页
create chunk from storagepool 参数: 从存储池创建块 *(SQL* 管理 *API)* 在第606页
create dbspace from storagepool 参数: 从存储池创建数据库空间 *(SQL* 管理 *API)* 在第610页
create plogspace: 创建物理日志空间 *(SQL* 管理 *API)* 在第611页
create sbspace from storagepool 参数: 从存储池创建智能大对象空间 *(SQL* 管理 *API)* 在第613页
create tempdbspace 参数: 创建临时数据库空间 *(SQL* 管理 *API)* 在第616页
create tempsbspace from storagepool 参数: 从存储池创建临时智能大对象空间 *(SQL* 管理 *API)* 在第618页
create tempdbspace from storagepool 参数: 从存储池创建临时数据库空间 *(SQL* 管理 *API)* 在第617页
drop blobspace to storagepool 参数: 从空 *BLOB* 空间归还空间到存储池 *(SQL* 管理 *API)* 在第621页
drop chunk to storagepool 参数: 将空块的空间归还到存储池 *(SQL* 管理 *API)* 在第622页
drop dbspace to storagepool 参数: 从空数据库空间归还空间到存储池 *(SQL* 管理 *API)* 在第624页
drop plogspace: 删除物理日志空间 *(SQL* 管理 *API)* 在第625页
drop sbspace to storagepool 参数: 从空智能大对象空间归还空间到存储池 *(SQL* 管理 *API)* 在第626页
drop tempdbspace to storagepool 参数: 从空临时数据库空间归还空间到存储池 *(SQL* 管理 *API)* 在第627页
drop tempsbspace to storagepool 参数: 从空临时智能大对象空间归还空间到存储池 *(SQL* 管理 *API)* 在第628页
modify chunk extend 参数: 扩展块的大小 *(SQL* 管理 *API)* 在第645页
modify chunk extendable off 参数: 标记块为不可扩展 *(SQL* 管理 *API)* 在第647页
modify chunk extendable 参数: 标记块为可扩展的 *(SQL* 管理 *API)* 在第646页
modify space expand 参数: 扩大空间的大小 *(SQL* 管理 *API)* 在第648页
modify space sp_sizes 参数: 更改可扩展的存储空间的大小 *(SQL* 管理 *API)* 在第649页
storagepool add 参数: 添加存储池条目 *(SQL* 管理 *API)* 在第693页
storagepool modify 参数: 更改存储池条目 *(SQL* 管理 *API)* 在第696页
storagepool delete 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第695页

参数
storagepool purge 参数: 删除存储池条目 (SQL 管理 API) 在第697页

TENANT 权限组

TENANT 权限组包括管理租户数据库的 SQL 管理 API 函数参数。

表 188: 租户数据库命令的 admin() 和 task() 函数参数

参数
tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页
tenant drop 参数: 删除租户数据库 (SQL 管理 API) 在第710页
tenant update 参数: 更改租户数据库属性 (SQL 管理 API) 在第710页

add bufferpool 参数: 添加缓冲池 (SQL 管理 API)

admin() 或 task() 函数使用 add bufferpool 参数来创建缓冲池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"add bufferpool", "page_siz"e
);
```

元素	描述	关键注意事项
page_size	以 KB 表示页大小。	页大小必须是缺省页大小的整数倍，且不可大于 16 KB。在 Windows™ 上，页大小总是 4 KB。

用法

使用 add bufferpool 参数为还没有缓冲池的页大小创建缓冲池。所创建的缓冲池的所有其他特性将设置为 BUFFERPOOL 配置参数的默认行中的字段值。

这个函数等同于 onparams -b -g 命令和 BUFFERPOOL 配置参数。

以下示例添加页大小为 8 KB 的缓冲池：

```
EXECUTE FUNCTION task("add bufferpool","8");
```

相关链接

onparams -b: 添加缓冲池 在第354页

BUFFERPOOL 配置参数 在第78页

add chunk 参数: 添加新块 (SQL 管理 API)

admin() 或 task() 函数使用 add chunk 参数来添加块到数据库空间或 Blob 空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"add chunk", "space_nam"e, "path_nam"e
[ , "disk_siz"e [ ,"offse"t [ ,"mirror_pat"h [ ,"mirror_offse"t ] ] ] ]
);
```

元素	描述	关键注意事项
disk_size	以 KB 表示要添加的磁盘空间量。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
mirror_offset	镜像块的位置。	
mirror_path	镜像块的路径。	如果您正添加块到镜像存储空间，那么您还必须添加镜像块。
offset	新块的位置。	
path_name	已添加的磁盘空间的路径。	
space_name	您正添加磁盘空间到其上的数据库空间、Blob 空间或智能大对象空间的名称。	

用法

块的大小必须等于或大于 1000 KB，且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。最大偏移量是 4 TB。

这个函数等同于 onspaces -a 命令。

示例

以下示例将偏移量 5200 KB 处的 5 MB 裸磁盘空间添加到名为 dbspc3 的数据库空间：

```
EXECUTE FUNCTION task("add chunk", "dbspc3","\\.\e:","5120","5200");
```

以下示例将 10 MB 镜像块添加到名为 blobsp3 的 Blob 空间，其主块和镜像块的偏移量都是 200 KB：

```
EXECUTE FUNCTION task("add chunk","blobsp3","/dev/raw_dev1","10240",
"200","/dev/raw_dev2","200");
```

相关链接

onspaces -a: 向数据库空间或 BLOB 空间添加块 在第362页

onspaces -a: 向智能大对象空间添加块 在第364页

add log 参数：添加新逻辑日志 (SQL 管理 API)

admin() 或 task() 函数使用 add log 参数来添加逻辑日志到数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"add log","dbspac"e
[ ,"siz"e [ , { 1 "coun"t } [ ,after_current_flag ] ] ]
);
```

元素	描述	关键注意事项
after_current_flag	是在当前日志之后还是在最后的逻辑日志（缺省）之后添加新的日志。	可能值为： • 1 = 在当前日志之后添加新的日志。 • 0 = 在最后的日志之后添加新的日志。
count	要创建的文件数。缺省为 1。	该数字必须不会导致逻辑日志文件的总数超过 32,767。
dbspace	要插入逻辑日志文件的数据库空间的名称。	仅当数据库服务器有足够的连续空间时，才可添加日志文件到数据库空间。 可在备份过程中添加日志文件。 不可添加日志文件到 Blob 空间或智能大对象空间。
size	以 KB 为单位的新逻辑日志文件大小。缺省为 LOGSIZE 配置参数所指定的大小。	该值必须是大于或等于 200 KB 的无符号整数。 另请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

新添加的日志文件具有状态 A 且立即可用。使用 onstat -l 来查看逻辑日志文件的状态。建议在运行这个函数之后，请尽速对包含该日志文件的根数据库空间和数据库空间进行 0 级备份。

在缺省情况下，新日志文件是添加在最后的逻辑日志之后。请包括 1 作为第五个参数来将逻辑日志文件插在当前日志文件之后。

这个函数类似于 onparams -a -d 命令，可添加单个逻辑日志文件。然而，您可单次调用这个函数来添加多个逻辑日志文件到指定的数据库空间。

示例

以下示例中命令在当前日志之后添加三个逻辑日志，每个的大小是 5 MB：

```
EXECUTE FUNCTION task ("add log","logdbs","5M",3,1);
```

相关链接

onparams -a -d dbspace: 添加逻辑日志文件 在第352页

add memory 参数: 增加共享内存 (SQL 管理 API)

admin() 或 task() 函数使用 add memory 参数来添加共享内存的虚拟部分。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"add memory", "memory_siz"e
);
```

元素	描述	关键注意事项
memory_size	以 KB 为单位的新虚拟共享内存段的大小。	该值必须不超过操作系统对于共享内存段大小的限制。 另请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

该大小预设置为 SHMADD 配置参数。

这个函数等同于 onmode -a 命令。

示例

以下示例添加 500 KB 的虚拟共享内存：

```
EXECUTE FUNCTION task("add memory","500");
```

相关链接

onmode -a: 添加共享内存段 在第331页

add mirror 参数: 添加镜像块 (SQL 管理 API)

admin() 或 task() 函数使用 add mirror 参数来添加镜像块到数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"add mirror", "space_nam"e, "path_nam"e,
"offse"t,"mirror_pat"h ,"mirror_offse"t
);
```

元素	描述	关键注意事项
mirror_path	执行镜像的数据库空间、Blob 空间或智能大对象空间的初始块的磁盘分区或无缓冲的设备。	
mirror_offset	到达新镜像数据库空间、Blob 空间或智能大对象空间的镜像块的偏移量。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

元素	描述	关键注意事项
offset	磁盘分区或无缓冲设备内，以 KB 为单位到达新镜像数据库空间、Blob 空间或智能大对象空间的初始块的偏移量。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
path_name	您想要镜像的数据库空间、Blob 空间或智能大对象空间的初始块的磁盘分区或无缓冲的设备。	
space_name	要镜像的数据库空间、Blob 空间或智能大对象空间的名称。	

用法

这个函数等同于 onspaces -m 命令。

示例

以下示例添加镜像块到名为 blobsp3 的 Blob 空间：

```
EXECUTE FUNCTION task("add mirror","blobsp3","/dev/raw_dev1",
"10240","/dev/raw_dev2","200");
```

相关链接

onspaces -m: 启动镜像 在第380页

alter chunk 参数：更改块状态为联机或脱机 (SQL 管理 API)

admin() 或 task() 函数使用 alter chunk 参数使数据库空间、Blob 空间或智能大对象空间中的块为联机或脱机。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "alter chunk offline"  | "alter chunk online" }
, "space_nam"e,  "path_ nam"e, "offse"t
);
```

元素	描述	关键注意事项
space_name	Blob 空间、数据库空间或智能大对象空间的名称。	
path_name	块的磁盘分区或无缓冲设备。	
offset	磁盘分区或无缓冲设备内，到达块的偏移量（以 KB 为单位）。缺省是 0。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

块必须在镜像对中，或者是非关键数据库空间中的非主块。

使用 alter chunk online 参数将块状态更改为联机。

使用 alter chunk offline 参数将块状态更改为脱机。

这个函数等同于 onspaces -s 命令。

示例

以下示例使名为 dbspace4 空间内的块为联机：

```
EXECUTE FUNCTION task("alter chunk online","dbspace4","/dev/raw_dev1","0");
```

相关链接

onspaces -s: 更改镜像块的状态 在第383页

alter logmode 参数：更改数据库日志记录模式 (SQL 管理 API)

admin() 或 task() 函数使用 alter logmode 参数来更改数据库日志记录模式为 ANSI、缓冲、无日志记录，或无缓冲。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"alter logmode", "database_nam"e,
{ "a"  | "b"  | "n"  | "u" }
);
```

元素	描述	关键注意事项
database_name	您想要更改其日志记录模式的数据库的名称。	

用法

与使用 ondblog 或 ontape 实用程序更改数据库日志记录模式时不同，当使用这个函数时，数据库保持可访问的，通常不需要进行 0 级备份。在运行这个函数之前，请确保没有其他会话是活动的，否则会失败。

使用 "a" 参数来更改数据库日志记录为 ANSI 兼容。创建或转换数据库为 ANSI 模式之后，则无法将其更改为任一其他日志记录模式。

使用 "b" 参数来更改数据库日志记录为缓冲，那么事务信息在写入逻辑日志之前是写入缓冲区。

使用 "n" 参数来更改数据库日志记录为无日志记录，那么所有数据库事务都不会记录日志。在使用这个参数之前，您必须执行 0 级备份。

使用 "u" 参数来更改数据库日志记录为无缓冲，那么数据在写入逻辑日志之前不写入缓冲区。

示例

以下示例更改名为 employee 数据库的日志记录模式为无缓冲日志记录：

```
EXECUTE FUNCTION task("alter logmode","employee","u");
```

相关链接

onlog 实用程序 在第327页

alter plog 参数: 更改物理日志 (SQL 管理 API)

admin() 或 task() 使用 alter plog 参数来更改物理日志的位置和大小。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"alter plog","dbspac"e [ , "phys_log_siz"e ]
);
```

元素	描述	关键注意事项
dbspace	物理日志的位置。	分配给物理日志的空间必须是连续的。
phys_log_size	以 KB 为单位的物理日志大小。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

要只更改大小，请指定物理日志的当前数据库空间。

这个函数等同于 onparams -p 命令。

示例

以下示例将物理日志移动到名为 phsdbs 的数据库空间：

```
EXECUTE FUNCTION task ("alter plog","physdbs","49 M");
```

相关链接

onparams -p: 更改物理日志参数 在第353页

archive fake 参数: 执行无记录的备份 (SQL 管理 API)

admin() 或 task() 函数使用 archive fake 参数来执行备份操作以克隆服务器中的数据，而不创建可用于执行复原的永久备份。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"archive fake"
);
```

用法

使用这个函数来充填高可用性数据复制对中的辅助服务器。

这个函数等同于运行带有 -F 选项的 ontape 命令。

示例

以下示例启动无记录的备份：

```
EXECUTE FUNCTION task("archive fake");
```

autolocate database add 参数：添加数据库空间到数据库空间列表 (SQL 管理 API)

admin() 或 task() 函数使用 autolocate database add 参数来添加数据库空间到可用数据库空间列表，供指定数据库中表的自动定位和分段。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"autolocate database add","database_nam"e , "dbspac"e
);
```

元素	描述	关键注意事项
database_name	数据库名称	
dbspace	要添加到数据库空间名称列表的数据库空间名称，在这些数据库空间中数据库服务器可自动创建分段。	数据库空间必须存在。

用法

AUTOLOCATE 配置参数或会话环境变量必须设置为正整数。

可用数据库空间的列表存储在 sysautolocate 系统目录表。

示例

以下命令添加数据库空间 dbspace9 到可用数据库空间的列表，供 customer 数据库中的表自动定位和分段。

```
EXECUTE FUNCTION task("autolocate database add", "customer", "dbspace9");
```

相关链接

AUTOLOCATE 配置参数 在第75页
autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页
autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API) 在第592页
autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页

autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API)

admin() 或 task() 函数使用 autolocate database anywhere 参数来指定可使用任何非关键数据库空间进行指定数据库中表的自动定位和分段的数据库服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"autolocate database anywhere","database_name"
);
```

元素	描述	关键注意事项
database_name	数据库名称	不可是租户数据库名称。

用法

这个命令以所有可用数据库空间的列表取代任何先前的数据库空间列表。专用于租户数据库的数据库空间不可用。可用数据库空间的列表存储在 sysautolocate 系统目录表中。

AUTOLOCATE 配置参数或会话环境变量必须设置为正整数。

示例

以下命令将所有非关键数据库空间添加到可用数据库空间列表，供 potential_cust 数据库中表的自动定位和分段：

```
EXECUTE FUNCTION task("autolocate database anywhere", "potential_cust");
```

相关链接

AUTOLOCATE 配置参数 在第75页
autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页
autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API) 在第592页
autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页

autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 autolocate database 参数来指定可用于指定数据库的表自动定位和分段的可用数据库空间列表。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"autolocate database","database_name" ,"dbspace_list"
);
```

元素	描述	关键注意事项
database_name	数据库名称	不可是租户数据库名称。
dbspace_list	以逗号分隔数据库空间名称的列表，数据库服务器可于这些数据库空间中自动创建分段。	这些数据库空间必须存在。数据库空间不可专用于租户数据库。

用法

AUTOLOCATE 配置参数或会话环境变量必须设置为正整数。

在缺省情况下，所有数据库空间是可用的。可用数据库空间的列表存储在 sysautolocate 系统目录表中。

示例

以下命令限制 customer 数据库中表的自动定位和分段的可用数据库空间列表：

```
EXECUTE FUNCTION task("autolocate database", "customer",
 "dbspace1,dbspace2,dbspace4,dbspace8");
```

相关链接

AUTOLOCATE 配置参数 在第75页
autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页
autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API) 在第592页
autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页
autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页
autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API) 在第592页

autolocate database off 参数: 禁用数据库的自动分段 (SQL 管理 API)

admin() 或 task() 函数使用 autolocate database off 参数来禁用指定数据库的表的自动定位和分段。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"autolocate database off","database_nam"e
);
```

元素	描述	关键注意事项
database_name	数据库名称	

用法

您在指定数据库中创建的新表存储在与数据库相同的数据库空间中，并且未分段。随着表增长，现有自动分段的不分配新的分段。

示例

以下命令禁用 customer_old 数据库中的表的自动定位和分段：

```
EXECUTE FUNCTION task("autolocate database off", "customer_old");
```

相关链接

AUTOLOCATE 配置参数 在第75页

autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页

autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页

autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API) 在第593页

autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页

autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页

autolocate database remove 参数: 从数据库空间列表中移除数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 autolocate database remove 参数从数据库服务器可为指定数据库自动定位表和分段表的可用数据库空间列表中移除数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"autolocate database remove","database_nam"e , "dbspac"e
);
```

元素	描述	关键注意事项
database_name	数据库名称	
dbspace	要从数据库服务器可自动创建分段的数据库空间名称的列表中移除的数据库空间的名称。	数据库空间必须存在。

用法

AUTOLOCATE 配置参数或会话环境变量必须设置为正整数。

可用数据库空间的列表存储在 sysautolocate 系统目录表中。

示例

以下命令从 customer 数据库的可用数据库空间的列表中移除 dbspace1。

```
EXECUTE FUNCTION task("autolocate database remove", "customer", "dbspace1");
```

相关链接

AUTOLOCATE 配置参数 在第75页

autolocate database 参数: 指定自动定位和分段的数据库空间 (SQL 管理 API) 在第591页

autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页

autolocate database add 参数: 添加数据库空间到数据库空间列表 (SQL 管理 API) 在第590页

autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 (SQL 管理 API) 在第591页

autolocate database off 参数: 禁用数据库的自动分段 *(SQL 管理 API)* 在第592页
autolocate database 参数: 指定自动定位和分段的数据库空间 *(SQL 管理 API)* 在第591页
autolocate database anywhere 参数: 添加所有数据库空间到数据库列表 *(SQL 管理 API)* 在第591页

cdr 参数: 管理 Enterprise Replication (SQL 管理 API)

admin() 或 task() 函数使用 cdr 参数来管理 Enterprise Replication。

语法
unique_754_Connect_42_repeat6

EXECUTE FUNCTION { admin | task } ('cdr*command_name*' [{ , [] *'option_name'* }]) ;

元素	描述	关键注意事项
command_name	cdr 命令名。	您不能在 cdr 命令行实用程序所需的 *command_name* 中包含任何连字符，标志或其他约束选项。您不可以 使用缩写。
option_name	*command_name* 的 cdr 命令行选项的一个或多个元素。	元素必须以引号界定。并可依正确顺序包含任何连字符、标志或 *command_name* 所需的 cdr 命令行选项的其他元素。您可以使用缩写。

用法

使用这些函数产生的管理 Enterprise Replication 的效果与 cdr 命令行实用程序相同。

SQL 管理 API 支持 cdr 命令用于管理 Enterprise Replication。不支持下列监视 Enterprise Replication 的命令：

- cdr list grid
- cdr list replicate
- cdr list replicateset
- cdr list server
- cdr list template
- cdr stats recv
- cdr stats rqm
- cdr -V
- cdr view

第一个参数必须仅包括 cdr 命令名，与《*SinoDB® Enterprise Replication* 指南》中 cdr 实用程序附录指定的名称完全一致，例如 cdr define server。命令名是区分大小写且不支持缩写（例如以 cdr sto replset 取代 cdr stop replicateset）。SQL 管理 API 在将参数传给 cdr 实用程序之前不执行任何验证。

第二个和任何后续参数包括命令选项。可在一个或最多六个参数中指定这些选项。

以下示例说明使用 SQL 管理 API 来定义 Enterprise Replication 服务器：

```
EXECUTE FUNCTION task ( 'cdr define server', '--connect=g_amsterdam
 --ats=/local0/er/ats --ris=/local0/er/ris --init g_amsterdam' );
```

以下示例显示选项如何分为几个参数；上面语句还可写为：

```
EXECUTE FUNCTION task( 'cdr define server',
   '--connect=g_amsterdam',
   '--ats=/local0/er/ats',
```

```
    '--ris=/local0/er/ris',
    '--init g_amsterdam' );
```

以下示例显示参数内的双引号字串符：

```
EXECUTE FUNCTION task('cdr change replicate',
'-d repl_1 -"db1@server1:antonio.table1" "db2@server2:carlo.table2"');
```

cdr add trustedhost 参数: 添加可信任主机 (SQL 管理 API)

admin() 或 task() 函数使用 cdr add trustedhost 参数来为高可用性集群或 Enterprise Replication 域中数据库服务器添加可信任主机关系。对于参与高可用性集群或 Enterprise Replication 域的数据库，其主机必须罗列在其他高可用性或复制服务器的可信任主机文件中。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("cdr add trustedhost ","
, {  localhost_name [  user_name ]  | fully_qualified_domain_name [  user_name ] } "
);
```

元素	描述	关键注意事项
localhost_name	数据库服务器的 localhost 名称。	
fully_qualified_domain_name	数据库服务器的完全域名称。	
user_name	在指定主机有数据库服务器实例授权的用户帐户。	

用法

cdr add trustedhost argument with the admin() 或 task() 函数使用 cdr add trustedhost 添加值到数据库服务器的 REMOTE_SERVER_CFG 配置参数所指定的文件中。如果数据库服务器是高可用性集群的一部分，那么可信任主机信息也会传播到其他集群服务器的可信任主机文件。可信任主机值指定分片集群中其他数据库服务器的 localhost 名称或完全限定域名称。为了增加安全性，您可指定与特定主机关联的用户名。

如果未设置 REMOTE_SERVER_CFG 配置参数，并运行带有 cdr add trustedhost 参数的 SQL 管理 API task() 或 admin() 函数时，数据库服务器会执行下列动作：

1. REMOTE_SERVER_CFG 配置参数设置为 authfile.*DBSERVER*。
2. 在 $INFORMIXDIR/etc 中创建 authfile.*DBSERVER* 文件。
3. 将指定的可信任主机信息添加到 $INFORMIXDIR/etc/authfile.*DBSERVER*。

如果您在高可用性集群中的服务器上运行带有 cdr add trustedhost 参数的 admin() 或 task() 函数，则将可信任主机信息添加到集群中所有数据库服务器的可信任主机文件。

您必须是数据库服务器管理员（DBSA）才可运行带有 cdr add trustedhost 参数的 admin() 或 task() 函数。

要查看可信任主机文件中的条目，请运行带有 cdr list trustedhost 参数的 admin() 或 task() 函数。

示例 1：添加可信任主机值到可信任主机文件

下列命令添加六个可信任主机值到数据库服务器的 REMOTE_SERVER_CFG 配置参数指定的文件：

```
EXECUTE FUNCTION task("cdr add trustedhost","myhost1, myhost1.ibm.com,
   myhost2, myhost2.ibm.com, myhost3, myhost3.ibm.com");
```

该任务指定三个数据库服务器的 localhost 名称和完全限定域名称。

示例 2：添加可信任主机和可信任用户值到可信任主机文件

下列命令添加四个可信任主机和用户的组合到数据库服务器的 REMOTE_SERVER_CFG 配置参数指定的文件：

```
EXECUTE FUNCTION task("cdr add trustedhost", "myhost1 informix,
    myhost1.ibm.com informix, myhost2 user_1, myhost2.ibm.com user_1");
```

该任务指定两个数据库服务器的 localhost 名称、完全限定的域名称和用户名称。

相关链接

REMOTE_SERVER_CFG 配置参数 在第147页
《*SinoDB JSON* 兼容性指南》：准备 *shard* 服务器

cdr autoconfig serv 参数：自动配置连接性和复制 (SQL 管理 API)

admin() 或 task() 函数使用 cdr autoconfig serv 参数可自动配置高可用性集群或 Enterprise Replication 域中服务器的连接性，以及自动配置复制。

语法

EXECUTE FUNCTION
{ admin | task }
(' cdr autoconfig serv '
[,' <连接选项 > { <源选项> <目标选项> | <源选项> <目标选项> }]
) ;

连接选项

连接选项
{ -c *server* | --connect = *server* | -c *server_group* | --connect = *server_group* }

源选项

源选项
--sourcehost *host* --sourceport *port*

目标选项

目标选项
--targethost *host* --targetport *port*

元素	用途	限制
host	数据库服务器主机的名称	
port	通信的端口号	
server	要连接的数据库服务器的名称	该名称必须是数据库服务器的名称或服务器连接的名称。
server_group	包括要连接的数据库服务器的数据库服务器组的名称	该名称必须是现有数据库服务器组的名称。

下表描述 cdr autoconfig serv 的选项。

长形	短形	含意
--sourcehost	-H	发送自动配置信息的数据库服务器的主机。如果未指定 --sourcehost 和 --sourceport，则将运行函数的数据库服务器视为源数据库服务器。
--sourceport	-P	数据库服务器用于发送自动配置信息的端口。
--targethost	-h	接收自动配置信息的数据库服务器的主机。
--targetport	-p	数据库服务器用于接收机自动配置信息的端口。

用法

运行带有 cdr autoconfig serv 参数的 admin() 或 task() 函数可自动配置高可用性集群或 Enterprise Replication 域中服务器的连接性，如果您正添加数据库服务器到 Enterprise Replication 域，则还可自动配置复制。在运行带有 cdr autoconfig serv 参数的 admin() 或 task() 函数之前，必须将参与 Enterprise Replication 域的所有数据库服务器上的 CDR_AUTO_DISCOVER 配置参数设置为 1。通过 cdr autoconfig serv 参数添加到 Enterprise Replication 域的新安装数据库服务器必须有一个配置的存储池。

如果源服务器配置了 Enterprise Replication，则函数执行下列动作：

1. 源服务器传播其可信任主机文件到目标服务器。
2. 目标服务器将自身和所有其他复制服务器的条目添加到其 sqlhosts 文件。
3. 源服务器以目标服务器的条目更新其 sqlhost 文件。
4. 每个复制服务器以目标服务器的条目更新其 sqlhost 文件和可信任主机文件。
5. 目标服务器设置其 CDR_DBSPACE 配置参数，并创建 Enterprise Replication 所需的数据库空间。
6. 目标服务器设置其 CDR_QDATA_SBSPACE 配置参数，并创建 Enterprise Replication 所需的智能大对象空间。
7. 在目标服务器上创建中止事务假脱机（ATS）文件目录 $INFORMIXDIR/tmp/ats_*dbservername*。
8. 在目标服务器上创建行信息假脱机（RIS）文件目录 $INFORMIXDIR/tmp/ris_*dbservername*。
9. 启动到目标服务器的复制。

如果源服务器未配置 Enterprise Replication，则函数执行其他动作：

1. 源服务器将自身的条目添加到其 sqlhosts 文件。
2. 源服务器设置其 CDR_DBSPACE 配置参数，并创建 Enterprise Replication 所需的数据库空间。
3. 源服务器设置其 CDR_QDATA_SBSPACE 配置参数，并创建 Enterprise Replication 所需的智能大对象空间。
4. 在源服务器上创建中止事务假脱机（ATS）文件目录 $INFORMIXDIR/tmp/ats_*dbservername*。
5. 在源服务器上创建行信息假脱机（RIS）文件目录 $INFORMIXDIR/tmp/ris_*dbservername*。
6. 在开始目标服务器上的复制之前，开始源服务器上的复制。

下列限制适用于带有 cdr autoconfig serv 参数的 admin() 或 task() 函数：

- 所有复制服务器必须是活动的，否则函数失败。
- 如果您是手动地配置了可信任主机信息，而不是通过运行带有 cdr add trustedhost 参数的 admin() 或 task() 函数配置的，则请不要运行带有 cdr autoconfig serv 参数的 admin() 或 task() 函数。
- 如果复制服务器已配置了安全端口，则请不要运行带有 cdr autoconfig serv 参数的 admin() 或 task() 函数。
- 带有 cdr autoconfig serv 参数的 admin() 或 task() 函数不会将 hosts.equiv 信息复制到 REMOTE_SERVER_CFG 配置参数指定的可信任主机文件。如果您必须将 hosts.equiv 文件的信息添加到 REMOTE_SERVER_CFG 配置参数指定的可信任主机文件，则请运行带有 cdr add trustedhost 参数的 admin() 或 task() 函数。

串行地配置数据库服务器。不支持并行配置。

您可如 cdr utility 那样运行这个函数。

示例 1：在本地服务器上配置 Enterprise Replication

此示例，您有一个未配置 Enterprise Replication 的本地数据库服务器：

在本地服务器上运行下列任务函数：

```
EXECUTE FUNCTION task('cdr autoconfig server');
```

该任务函数在本地服务器上配置 Enterprise Replication。

示例 2：通过使用源语法在两个独立服务器之间配置连接性和 ER

此示例，您有两个数据库服务器：

- host_1 上的 server_1 配置了 Enterprise Replication
- host_2 上的 server_2 未配置 Enterprise Replication

```
EXECUTE FUNCTION task('cdr autoconfig server', '--connect server_2
   --sourcehost host_1 --sourceport 9020');
```

该任务函数执行下列动作：

1. 命令连接到 server_2。
2. 在 server_2 上定义 Enterprise Replication。
3. server_1 复制其数据到 server_2

示例 3：通过使用目标语法在两个独立服务器之间配置连接性和 ER

以下命令

```
EXECUTE FUNCTION task('cdr autoconfig server', '--connect server_1
   --targethost host_2 --targetport 9030');
```

该任务函数执行下列动作：

1. 命令连接到 server_1。
2. 在 server_2 上定义 Enterprise Replication。
3. server_1 复制其数据到 server_2

相关链接

《SinoDB Enterprise Replication 指南》: cdr autoconfig serv

《SinoDB Enterprise Replication 指南》: CDR_AUTO_DISCOVER 配置参数

cdr list trustedhost 参数: 罗列可信任主机 (SQL 管理 API)

admin() 或 task() 函数使用 cdr list trustedhost 参数来罗列数据库服务器 REMOTE_SERVER_CFG 配置参数所指定文件中的可信任主机信息。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("cdr list trustedhost");
```

用法

您必须是数据库服务器管理员（DBSA）才可运行这个函数。

示例

下列命令罗列数据库服务器的可信任主机文件的可信任主机条目：

```
EXECUTE FUNCTION task("cdr list trustedhost");
```

下来示例输出显示使用 cdr list trustedhost 参数的可能的结果。

```
myhost1 user_1
myhost1.example.com user_1
myhost2 user_2
myhost2.example.com user_2
```

相关链接

REMOTE_SERVER_CFG 配置参数 在第147页

《SinoDB JSON 兼容性指南》：准备 shard 服务器

cdr remove trustedhost 参数：移除可信任主机（SQL 管理 API）

admin() 或 task() 函数使用 cdr remove trustedhost 参数来从数据库服务器的可信任主机文件中移除条目。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("cdr remove trustedhost","
, { localhost_name fully_qualified_domain_name  | localhost_name user_name  |
fully_qualified_domain_name  user_name } "
);
```

元素	描述	关键注意事项
localhost_name	数据库服务器的 localhost 名称。	如果在命令中未指定 *user_name*，则移除包括指定主机名称的所有条目。
fully_qualified_domain_name	数据库服务器的完全域名称。	如果在命令中未指定 *user_name*，则移除包括指定完全限定域名称的所有条目。
user_name	在指定主机有数据库服务器实例授权的用户帐户。	

用法

cdr remove trustedhost 参数从数据库服务器 REMOTE_SERVER_CFG 配置参数指定的可信任主机文件中移除可信任主机条目。对于参与高可用性集群或 Enterprise Replication 域中的数据库，其主机必须罗列在其他高可用性或复制服务器的可信任主机文件中。当在高可用性集群的服务器上运行带有 cdr remove trustedhost 参数的 admin() 或 task() 函数时，将从所有集群服务器的可信任主机文件中移除可信任主机条目。

要查看可信任主机文件中的条目，请运行带有 cdr list trustedhost 参数的 admin() 或 task() 函数。

您必须是数据库服务器管理员才可运行带有 cdr remove trustedhost 参数的 admin() 或 task() 函数。

示例 1：从可信任主机文件移除主机条目

以下命令从数据库服务器 REMOTE_SERVER_CFG 配置参数指定的可信任主机文件中移除一个 localhost 名称值和一个完全限定域名称：

```
EXECUTE FUNCTION task("cdr remove trustedhost","myhost1, myhost1.ibm.com");
```

从数据库服务器可信任主机文件中将 myhost1 和 myhost1.ibm.com 条目移除。

示例 2：从可信任主机文件中移除主机和用户条目

以下命令从数据库服务器 REMOTE_SERVER_CFG 配置参数指定的可信任主机文件中移除 localhost 名称值、完全限定域名称和用户名称值：

```
EXECUTE FUNCTION task("cdr remove trustedhost",
   "myhost2 john,myhost2.ibm.com john,myhost3 informix,myhost3.ibm.com
 informix");
```

从数据库服务器可信任主机文件中将用户 john 的 myhost2、用户 john 的 myhost2.ibm.com、用户 informix 的 myhost3 和 用户 informix 的 myhost3.ibm.com 等条目移除。

相关链接

REMOTE_SERVER_CFG 配置参数 在第147页

《*SinoDB JSON* 兼容性指南》：准备 *shard* 服务器

check data 参数：检查数据一致性 (SQL 管理 API)

Use the check data argument with the admin() 或 task() 函数使用 check data 参数来检查或修复指定分区中所有页的一致性。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "check data"  | "check data only"  | "check data repair" }  ,"partition_numbe"r );
```

元素	描述	关键注意事项
partition_number	在其中检查数据的分区号	在 systables 系统目录表的 partnum 列中查找分区号。

用法

使用 check data 参数读取智能大对象页以外的所有页，并检查每一页的一致性。这个参数等同于 oncheck -cD 命令。

使用 check data only 参数读取 Blob 页和智能大对象页以外的所有页，并检查每一页的一致性。这个参数等同于 oncheck -cd 命令。

使用 check data repair 参数来修复不一致的页。这个参数等同于 oncheck -cD -y 命令。

示例

以下命令检查分区 1048611 中的所有页的一致性：

```
EXECUTE FUNCTION task("check data","1048611");
```

相关链接

oncheck -cd 和 oncheck -cD 命令: 检查页 在第300页

check extents 参数: 检查扩展数据块一致性 (SQL 管理 API)

admin() 或 task() 函数使用 check extents 参数来验证磁盘上扩展数据块与当前控制信息是否相符。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("check extents" { [ ,dbspace_number ] } );
```

元素	描述	关键注意事项
dbspace_number	要检查的数据库空间号	

用法

运行这个函数来检查每个可用块列表和相应的可用空间和每个表空间扩展数据块。如果不指定数据库空间号，则检查所有数据库空间。该函数检查数据库空间、Blob 空间、智能大对象扩展数据块，以及智能大对象空间块中的用户数据和元数据。

这个函数等同于 oncheck -ce 命令。

示例

以下示例检查编号 2 的数据库空间中的扩展数据块：

```
EXECUTE FUNCTION task("check extents",2);
```

相关链接

oncheck -ce, -pe: 检查空闲块列表 在第301页

check partition 参数: 检查分区一致性 (SQL 管理 API)

admin() 或 task() 函数使用 check partition 参数来打印表或分段的表空间信息。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("check partition"  ,"partition_numbe"r );
```

元素	描述	关键注意事项
partition_number	您想要检查一致性的分区号。	在 systables 系统目录表的 partnum 列中查找分区号。

用法

带有 check partition 参数的 task() 函数返回的信息等同于 oncheck -pt 命令的输出。输出包含一般信息，例如：最大行大小、键数、扩展数据块的数量和大小、每个扩展数据块已分配和使用的页、当前序列值和创建表的日期。

admin() 函数返回一个整数，您可以使用该整数在 sysadmin 数据库的 command_history 表中查找信息。

示例

以下示例打印分区 1048611 的信息：

```
EXECUTE FUNCTION task("check partition","1048611");
```

相关链接

oncheck -pt 和 -pT: 显示表或分段的表空间 在第308页

checkpoint 参数: 强制检查点 (SQL 管理 API)

admin() 或 task() 函数使用 checkpoint 参数来强制检查点。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"checkpoint",
{ "block"  | "hard"  | "norm"  | "unblock" }
);
```

用法

这个函数强制一个将缓冲区清空到磁盘的检查点。如果逻辑日志中的最新检查点记录阻止释放逻辑日志文件（状态 U-B-L），则可使用这个函数强制一个检查点。

使用 block 参数来防止数据库服务器处理任何事务。在 SinoDB® 上执行外部备份时使用此选项。数据库服务器阻塞时，除非在只读模式中，否则用户不能访问它。直到数据库服务器阻塞解除，事务才可完成。

使用 hard 参数来强制一个阻塞检查点。这个缺省情况。

使用 norm 参数来强制一个非阻塞检查点。

使用 unblock 参数来解除数据库服务器阻塞。当数据库服务器解除阻塞时，可以恢复数据事务和正常的数据库服务器操作。在完成 SinoDB® 上的外部备份之后使用此选项。

这个函数等同于 onmode -c 命令。

示例

以下示例启动一个阻塞检查点：

```
EXECUTE FUNCTION task("checkpoint","block");
```

相关链接

onmode -c: 强制检查点 在第332页

clean sbspace 参数: 释放未引用的智能大对象 (SQL 管理 API)

admin() 或 task() 函数使用 clean sbspace 参数来从智能大对象空间释放未引用的 BLOB 或 CLOB 对象。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"clean sbspace" ,"sbspac"e
) ;
```

元素	描述	关键注意事项
sbspace	要清除的智能大对象空间的名称。	

用法

这个函数等同于 onspaces -cl 命令。

示例

以下示例清除名为 sbsp1 的智能大对象空间:

```
EXECUTE FUNCTION task("clean sbspace","sbsp1");
```

相关链接

onspaces -cl: 清除智能大对象空间中的游离智能大对象 在第377页

create blobspace 参数: 创建 BLOB 空间 (SQL 管理 API)

admin() 或 task() 函数使用 create blobspace 参数来创建 Blob 空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  blobspace","blobspac"e,  "path_nam"e
[ ,  "initial_chunk_siz"e [ ,"offse"t [ ,"page_siz"e ] ] ]
);
```

元素	描述	关键注意事项
blobspace	要创建的 Blob 空间的名称。	
initial_chunk_size	以 KB 为单位的新 Blob 空间的初始块的大小。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
offset	到达新 Blob 空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
page_size	Blob 空间 Blob 页大小。	以操作系统的默认 SinoDB® 页大小的倍数来指定 Blob 页大小。

元素	描述	关键注意事项
		有关更多信息，请参阅《*SinoDB*® 性能指南》中的 Blob 页大小的注意事项。
path_name	正在创建的 Blob 空间的初始块的磁盘分区或设备。	

用法

使用 create with_check blobspace 参数来检查指定的路径名，如果路径不存在，则返回错误。

这个函数等同于 onspaces -c -b 命令。

示例

以下示例创建一个 Blob 空间，其大小为 20 MB，偏移量为 0，页大小为 2 。在 Windows™ 上（基本页大小是 4K），Blob 页是 2 * 基本页大小 = 8 K。

```
EXECUTE FUNCTION task ("create with_check blobspace","blobs3",
"$INFORMIXDIR/WORK/blobs3","20 M","0","2");
```

相关链接

onspaces -c -b: 创建 *BLOB* 空间 在第365页

避免覆盖块 在第384页

create blobspace from storagepool 参数：从存储池创建 BLOB 空间 (SQL 管理 API)

admin() 或 task() 函数使用 create blobspace from storagepool 参数来从存储池中一个条目创建一个 Blob 空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create blobspace from storagepool"
, "blobspac"e, "initial_chunk_siz"e, [ ,"blobpage_siz"e [ ,"mirroring_fla"g ] ]
);
```

元素	描述	关键注意事项
blobspace	Blob 空间名称。	Blob 空间名称必须唯一且不可超过 128 字节。必须以字母或下划线开始，且必须仅包含字母、数字、下划线或 $ 字符。
blobpage_size	Blob 页大小，根据 *page_unit* 指定，每个 Blob 页的磁盘页数。	页大小是可选的。然而，如果您为镜像指定 1，则还必须指定页大小。
initial_chunk_size	以 KB 为单位的新 Blob 空间的初始块的大小。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
mirroring_flag	二者之一： 1 = 镜像	镜像标志是可选的。

元素	描述	关键注意事项
	0 = 无镜像	

示例

下列命令创建名为 blobspace1 的镜像 Blob 空间。新 Blob 空间大小为 100 GB， Blob 页大小为 100 页。

```
EXECUTE FUNCTION task("create blobspace from storagepool", "blobspace1", "100
 GB",
 "100", "1");
```

下列命令创建名为 blobspace2 的无镜像 Blob 空间，其使用缺省 Blob 页大小，因此未指定 Blob 页大小：

```
EXECUTE FUNCTION task("create blobspace from storagepool", "blobspace2",
 "5000");
```

create chunk 参数: 创建块 (SQL 管理 API)

admin() 或 task() 函数使用 create chunk 参数在数据库空间或 Blob 空间中创建块。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  chunk", "space_nam"e, "path_nam"e
[ , "disk_siz"e [ ,"offse"t [ ,"mirror_pat"h [ ,"mirror_offse"t ] ] ] ]
);
```

元素	描述	关键注意事项
disk_size	要添加的磁盘空间量，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
mirror_offset	镜像块的位置。	
mirror_path	到镜像块的路径。如果正在添加块到镜像存储空间，那么还必须添加镜像站点块。	
offset	新块的位置。	
path_name	添加的磁盘空间的路径。	
space_name	您正在添加磁盘空间的数据库空间、Blob 空间或智能大对象空间的名称。	

用法

使用 create with_check chunk 参数来检查指定的路径名，如果该路径名不存在，则返回错误。

这个函数等同于 onspaces -a 命令。

示例

下列示例在偏移量 5200 KB 处添加 5 MB 裸磁盘空间的块到名为 dbspc3 的数据库空间：

```
EXECUTE FUNCTION task("create chunk", "dbspc3","\\.\e:","5120","5200");
```

下列示例添加 10 MB 镜像块到名为 blobsp3 的 Blob 空间，主块和镜像块的偏移量都是 200 KB：

```
EXECUTE FUNCTION task("create with_check chunk","blobsp3","/dev/
raw_dev1","10240",
"200","/dev/raw_dev2","200");
```

相关链接

onspaces -a: 向数据库空间或 *BLOB* 空间添加块 在第362页

避免覆盖块 在第384页

onspaces -a: 向智能大对象空间添加块 在第364页

create chunk from storagepool 参数：从存储池创建块 (SQL 管理 API)

admin() 或 task() 函数使用 create chunk from storagepool 参数来手动地从存储池中的条目创建块。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create chunk from storagepool"
,"space_nam"e,  "initial_chunk_siz"e
);
```

元素	描述	关键注意事项
space_name	您正在其中添加块的存储空间的名称。	
initial_chunk_size	初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

还可使用带有 modify space expand 参数的 SQL 管理 API 命令来手动地从存储池创建块，并将块添加到指定的存储空间。然而，如果该空间有可扩展的块，那么 SinoDB® 可能扩展块，而不是创建一个新块。与 modify space expand 参数不同，create chunk from storagepool 参数强制 SinoDB® 添加块。

示例

下列命令添加块到名为 logdbs 数据库空间。新块的大小为 200 MB。

```
EXECUTE FUNCTION task("create chunk from storagepool", "logdbs", "200 MB");
```

相关链接

modify space expand 参数: 扩大空间的大小 (SQL 管理 API) 在第648页

create database 参数: 创建数据库 (SQL 管理 API)

admin() 或 task() 函数使用 create database 参数来创建数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("create database
{ [ with { log  | buffered log  | log mode ansi } ] }
[ { nlscase insensitive  | nlscase sensitive } ]
"
,"database_nam"e
[ ,"dbspac"e [ ,"local"e ] ]
);
```

元素	描述	关键注意事项
database_name	数据库的名称。	
dbspace	存储该数据库的数据的数据库空间的名称。缺省为根数据库空间。	数据库空间必须在数据库服务器上已存在。
locale	与数据库关联的语言环境。	*locale* 的值与 DB_LOCALE 环境变量的值相同。 如果省略这个属性，则以 DB_LOCALE 环境变量的值设置该语言环境。缺省语言环境为 US English。

用法

这个函数等同于 CREATE DATABASE 语句。

不可使用这个函数来创建租户数据库。请使用 tenant create 参数来创建租户数据库。

示例

以下示例创建带有无缓冲日志记录的 demodbs 数据库：

```
EXECUTE FUNCTION task("create database with log","demodbs");
```

以下示例在 dataspace1 数据库空间中创建带有 ANSI 兼容日志记录且不区分大小写的 demodbs2 数据库：

```
EXECUTE FUNCTION task("create database with log mode ansi nlscase insensitive",
"demodbs2","dataspace1");
```

相关链接

tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页

《*SinoDB SQL 指南: 语法*》: *CREATE DATABASE 语句*

《*SinoDB GLS 用户指南*》: *DB_LOCALE 环境变量*

create dbaccessdemo 参数: 创建演示数据库 (SQL 管理 API)

admin() 或 task() 函数使用 create dbaccessdemo 参数来创建 stores_demo 演示数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("create dbaccessdemo "
[ ,"database_name " [ ," dbspace" [ ,"-log " ] [ ,"-nots " ] ] ]
);
```

元素	用途	关键注意事项
database_name	要创建的数据库的名称。	缺省数据库名称是 stores_demo。
dbspace	要在其中创建数据库的数据库空间的名称。	缺省数据库空间是根数据库空间。

用法

运行这个函数来创建 stores_demo 数据库。

使用 -log 选项来启用演示数据库的事务日志记录。

使用 -nots 选项来在演示数据库中创建具有时间序列数据的表。

示例

下列命令在根数据库空间中创建 stores_demo 数据库：

```
EXECUTE FUNCTION task("create dbaccessdemo");
```

下列命令在名为 dbs1 的数据库空间中创建名为 demo2 的演示数据库：

```
EXECUTE FUNCTION task("create dbaccessdemo","demo2","dbs1");
```

下列命令在名为 dbs1 数据库空间中创建带有事务日志记录的 stores_demo 数据库：

```
EXECUTE FUNCTION task("create dbaccessdemo","stores_demo","dbs1","-log");
```

相关链接

《SinoDB SQL 指南: 参考》: stores_demo 数据库映射

create dbspace 参数: 创建数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 create dbspace 参数来创建数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create  [with_check]  dbspace" , "dbspac"e , "path_nam"e
[ , "initial_chunk_siz"e [ , "offse"t [ , "page_siz"e [  <扩展数据块和镜像块>  ] ] ] ]
) ;
```

扩展数据块和镜像块

扩展数据块和镜像块

[, *"first_extsiz"e* [, *"next_extsiz"e* [, *"mirror_pat"h* [, *"mirror_offse"t*]]]]

元素	描述	关键注意事项
dbspace	要创建的数据库空间的名称。	
first_extsize	表空间 tblspace 的第一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
initial_chunk_size	新数据库空间的初始块的大小，以 KB 为单位。该大小取整到页大小的倍数。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
mirror_offset	镜像块的偏移量，以 KB 为单位。	
mirror_path	对数据库空间初始块镜像的那个块的路径名。	
next_extsize	表空间 tblspace 中下一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
offset	到达新数据库空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	
page_size	新数据库空间的非缺省页大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
path_name	正在创建的数据库空间的初始块的磁盘分区或设备。	有效的页大小取决于计算机的缺省页大小： • 2 KB 缺省页大小：2、4、6、8、10、12 或 16 KB • 4 KB 缺省页大小：4、8、12 或 16 KB

用法

使用 create with_check dbspace 参数来检查指定的路径名，如果该路径名不存在，则返回错误。

这个函数等同于 onspaces -c -d 命令。

示例

以下示例创建一个大小为 20 MB，偏移量为 0 的数据库空间。

```
EXECUTE FUNCTION task ("create dbspace", "dbspace3",
"$INFORMIXDIR/WORK/dbspace3", "20 M", "0");
```

相关链接

onspaces -c -d: 创建数据库空间 在第367页

避免覆盖块 在第384页

create dbspace from storagepool 参数: 从存储池创建数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 create dbspace from storagepool 参数来从存储池中的条目创建永久数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
"create dbspace from storagepool"
,"dbspac"e,  "initial_chunk_siz"e,
[ ,"page_siz"e [ ,"mirroring_fla"g [ ,  "first_exten"t [ ,  "next_exten"t ] ] ] ]
);
```

元素	描述	关键注意事项
dbspace	要创建的数据库空间的名称。	在数据库空间名称中必须是唯一的，且不可超过 128 字节。名称必须以字母或下划线开始，且仅可包括字母、数字、下划线（_）或 $ 字符。
first_extent	表空间 tblspace 的第一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
initial_chunk_size	新数据库空间的初始块的大小。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
mirroring_flag	二者之一： 1 = 镜像 0 = 无镜像	该镜像标志是可选的。如果未指定，则缺省是无镜像的数据库空间。
next_extent	表空间 tblspace 中下一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
page_size	新数据库空间的非缺省页大小，以 KB 为单位。	该页大小是可选的。如果镜像指定为 1，则还必须页大小。有效页大小取决于计算机的缺省页大小： • 2 KB 缺省页大小：2、4、6、8、10、12 或16 KB • 4 KB 缺省页大小：4、8、12 或16 KB

有关创建临时数据库空间的 admin() 或 task() 语法，请参阅 *create tempdbspace 参数: 创建临时数据库空间 (SQL 管理 API)* 在第616页。

示例

以下命令创建名为 dbspace3 的镜像数据库空间。新数据库空间的大小为 1 GB，页大小为 6 KB，表空间 tblspace 的第一个扩展数据块大小为 200 KB，且下一个扩展数据块大小为 400 KB。

```
EXECUTE FUNCTION task("create dbspace from storagepool",
 "dbspace3",  "1 GB", "6", "1", "200", "400");
```

以下命令创建名为 dbspace8 的无镜像数据库空间。新数据库空间的大小为 50 MB。由于未指定页大小，因此新数据库空间为缺省页大小。

```
EXECUTE FUNCTION task("create dbspace from storagepool",
 "dbspace8", "50000");
```

create plogspace: 创建物理日志空间 (SQL 管理 API)

admin() 或 task() 函数使用 create plogspace 参数来创建用于存储物理日志的物理日志空间。

语法

```
EXECUTE FUNCTION
{ admin | task }
(
"createplogspace","plogspac"e, "path_nam"e
, "chunk_siz"e[ ,"offse"t[ ,"mirror_pat"h[ ,"mirror_offse"t ] ] ]
);
```

元素	描述	关键注意事项
chunk_size	新物理日志空间的块的大小，以 KB 为单位。该大小取整到页大小的倍数。	请参阅 *admin() 和 task() 参数大小规范* 在第572页.
mirror_offset	镜像块的偏移量，以 KB 为单位。	无符号整数。该大小必须等于或大于 1000 KB 且是页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小是 2 或 4 TB，其取决于平台。
mirror_path	对物理日志空间的块镜像的块的路径名。	如果您镜像了物理日志空间，那么物理日志空间块无法扩展。
offset	到达新物理日志空间的块的磁盘分区内或设备内的偏移量，以 KB 为单位。	无符号整数。该大小必须等于或大于 1000 KB 且为页大小的倍数。起始偏移量加上块大小不可超过最大块大小。 最大块大小是 2 或 4 TB，其取决于平台。
path_name	正在创建的物理日志空间的块的磁盘分区或设备。	块必须是现有无缓冲的设备或缓冲的文件。当指定路径名时，可使用全路径名或相对路径名。然而，如果使用相对路径名，其必须是相对于初始化数据库服务器时的当前目录。 UNIX™ 示例（无缓冲的设备）： /dev/rdsk/c0t3d0s4 UNIX™ 示例（缓冲的设备）： /ix/ifmx/db1chunk Windows™ 示例：

元素	描述	关键注意事项
plogspace	要创建的物理日志空间的名称。	c:\Ifmxdata\ol_icecream\mychunk1.dat 物理日志空间名称必须是唯一的且不可超过 128 字节。其必须以字母或下划线开始，且必须仅包含字母、数字、下划线或 $ 字符。 语法必须符合 Identifier 段。有关更多信息，请参阅《*SinoDB® SQL* 指南: 语法》中的 *Identifier*。

用法

这个函数等同于 onspaces -c -P 命令。

一个实例仅可有一个物理日志空间。如果存在物理日志空间，则在创建新物理日志空间时，物理日志将搬移到新的空间，并删除旧的物理日志空间。

物理日志必须存储在单个块。缺省情况下，该块标记为可扩展的，以便如果有需要时数据库服务器可扩展该物理日志空间以扩展物理日志。如果您镜像该物理日志空间，那么该空间将因为镜像块不可扩展而无法扩展。

示例

下例创建大小为 30000 KB，偏移量为 0 的物理日志空间。

```
EXECUTE FUNCTION task ("create plogspace", "plogdbs",
"/dev/chk1", 30000, 0);
```

下例创建大小为 30000 KB，偏移量为 0 的镜像物理日志空间。

```
EXECUTE FUNCTION task ("create plogspace", "plogdbs",
"/dev/chk1", 30000, 0, "/dev/mchk1", 0);
```

相关链接

《*SinoDB* 管理员指南》: 物理日志空间

《*SinoDB* 管理员指南》: 管理物理日志空间

onspaces -c -P: 创建物理日志空间 在第369页

create sbspace 参数: 创建智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create sbspace 参数来创建智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  sbspace","sbspac"e,  "path_nam"e
[ ,"initial_chunk_siz"e [ ,"offse"t ] ]
);
```

元素	描述	关键注意事项
initial_chunk_size	新智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
offset	到达新智能大对象空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	
path_name	智能大对象空间的初始块的磁盘分区或无缓冲设备。	
sbspace	要创建的智能大对象空间的名称。	

用法

使用 create with_check sbspace 参数来检查指定的路径名，如果该路径不存在，则返回错误。

这个函数等同于 onspaces -c -S 命令。

示例

下例创建大小为 20 MB，偏移量为 0 的新智能大对象空间：

```
EXECUTE FUNCTION task ("create sbspace","sbspace2",
"$INFORMIXDIR/WORK/sbspace2","20 M","0");
```

相关链接

onspaces -c -S: 创建智能大对象空间 在第371页

避免覆盖块 在第384页

create tempsbspace 参数: 创建临时智能大对象空间 (SQL 管理 API) 在第617页

create sbspace from storagepool 参数: 从存储池创建智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create sbspace from storagepool 参数来从存储池中的条目创建智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create sbspace from storagepool"
, "sbspac"e,"initial_chunk_siz"e [ ,"logging_fla"g [ ,"mirroring_fla"g ] ]
);
```

元素	描述	关键注意事项
sbspace	智能大对象空间的名称。	智能大对象空间名称必须是唯一的，且不可超过 128 字节。名称必须以字母或下划线开始，且必须仅包含字母、数字、下划线或 $ 字符。
initial_chunk_size	新智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

元素	描述	关键注意事项
logging_flag	二者之一： 1 = 日志记录 0 = 无日志记录	该日志记录标志是可选的。然而，如果您指定镜像为 1，则还必须指定日志记录标志。
mirroring_flag	二者之一： 1 = 镜像 0 = 无镜像	该镜像标志的可选的。

示例

以下命令镜像的且日志记录的名为 sbspace1 的智能大对象空间。该新智能大对象空间的大小为 240 MB。

```
EXECUTE FUNCTION task("create sbspace from storagepool", "sbspace1",
 "240 MB", "1", "1");
```

以下命令创建无镜像且无日志记录的名为 sbspace2 的智能大对象空间。这个智能大对象空间的大小为 5 GB。

```
EXECUTE FUNCTION task("create sbspace from storagepool", "sbspace2", "5 GB");
```

create sbspace with accesstime 参数：创建跟踪访问时间的智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create sbspace with accesstime 参数来创建智能大对象空间，跟踪存储在该智能大对象空间中所有智能大对象的访问时间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  sbspace with accesstime","sbspac"e , "path_nam"e
[ , "initial_chunk_siz"e [ ,"offse"t ] ]
);
```

元素	描述	关键注意事项
initial_chunk_size	新智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
offset	到达新智能大对象空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	
path_name	智能大对象空间的初始块的磁盘分区或无缓冲设备。	
sbspace	要创建的智能大对象空间的名称。	

用法

使用 create with_check sbspace 参数来检查指定的路径名，如果该路径不存在，则返回错误。

这个函数等同于 onspaces -c -S 命令创建智能大对象空间，并使用带有 set sbspace accesstime 参数的 admin() 或 task() 函数来启动跟踪存储在该智能大对象空间中的所有智能大对象的访问时间。

示例

下例创建跟踪访问时间的新智能大对象空间。这个智能大对象空间的大小为 20 MB，偏移量为 0：

```
EXECUTE FUNCTION task ("create sbspace with accesstime","sbspace4",
"$INFORMIXDIR/WORK/sbspace4","20 M","0");
```

相关链接

set sbspace accesstime 参数: 控制访问时间跟踪 (SQL 管理 API) 在第683页

create sbspace with log 参数: 创建具有事务日志记录的智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create sbspace with log 参数来创建打开事务日志记录的智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  sbspace with log","sbspac"e,  "path_nam"e
[ ,  "initial_chunk_siz"e  [ ,"offse"t  ] ]
);
```

元素	描述	关键注意事项
initial_chunk_size	新智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
offset	到达新智能大对象空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	
path_name	智能大对象空间的初始块的磁盘分区或无缓冲设备。	
sbspace	要创建的智能大对象空间的名称。	

用法

使用 create with_check sbspace 参数来检查指定的路径名，如果该路径不存在，则返回错误。

这个函数等同于 onspaces -c -S 命令，以开启日志记录的选项来创建智能大对象空间。

示例

下例创建打开事务日志记录的新智能大对象空间。这个智能大对象空间的大小为 20 MB，偏移量为 0：

```
EXECUTE FUNCTION task ("create sbspace with log","sbspace2",
"$INFORMIXDIR/WORK/sbspace2","20 M","0");
```

相关链接

onspaces -c -S: 创建智能大对象空间 在第371页
《*SinoDB* 管理员指南》: 智能大对象空间日志记录

create tempdbspace 参数: 创建临时数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 create tempdbspace 参数来创建临时数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  tempdbspace", "tempdbspac"e, "path_nam"e
[ , "initial_chunk_siz"e [ ,"offse"t [ ,"pag"e [ ,"firs"t [ ,"nex"t ] ] ] ] ]
);
```

元素	描述	关键注意事项
first	表空间 tblspace 的第一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
initial_chunk_size	新临时数据库空间的初始块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
next	表空间 tblspace 中下一个扩展数据块的大小，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
offset	到达新临时数据库空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	请参阅 *admin()* 和 *task()* 参数大小规范 在第572页。
page	新临时数据库空间的非缺省页大小，以 KB 为单位。	有效的页大小取决于计算机的缺省页大小： • 2 KB 缺省页大小：2、4、6、8、10、12 或 16 KB • 4 KB 缺省页大小：4、8、12 或 16 KB
path_name	正在创建的临时数据库空间的初始块的磁盘分区或设备的路径	
tempdbspace	要创建的临时数据库空间的名称。	不可超过 128 字节。名称必须以字母或下划线开始，且仅可包括字母、数字、下划线（_）符号或 $ 字符。

用法

使用 create with_check tempdbspace 参数来检查指定的路径名，如果该路径不存在，则返回错误。

这个函数等同于 onspaces -c -d -t 命令。

示例

下例创建大小为 20 MB，偏移量为 0 的临时数据库空间：

```
EXECUTE FUNCTION task("create tempdbspace","tempdbspace3",
"$INFORMIXDIR/WORK/tempdbspace3","20 M","0");
```

有关从存储池创建永久数据库空间的 admin() 或 task() 语法，请参阅 *create dbspace from storagepool 参数: 从存储池创建数据库空间 (SQL 管理 API)* 在第610页。

相关链接

onspaces -c -d: 创建数据库空间 在第367页

create tempdbspace from storagepool 参数: 从存储池创建临时数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 create tempdbspace from storagepool 参数来从存储池中的条目创建临时数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create tempdbspace from storagepool "
, "tempdbspac"e, "initial_chunk_siz"e,"page_siz"e
);
```

元素	描述	关键注意事项
initial_chunk_size	新临时数据库空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
page_size	新临时数据库空间的非缺省页大小，以 KB 为单位。	该页大小是可选的。
tempdbspace	临时数据库空间的名称。	

示例

以下命令创建名为 tempdbspace1 的临时数据库空间。该新数据库空间的大小为 1 GB，页大小为 12 KB。

```
EXECUTE FUNCTION task("create tempdbspace from storagepool", "tempdbspace1",
 "1 GB", "12");
```

create tempsbspace 参数: 创建临时智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create sbspace 参数来创建智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create [with_check]  tempsbspace", "tempsbspac"e,"path_nam"e
[ , "initial_chunk_siz"e [ ,"offse"t ] ]
);
```

元素	描述	关键注意事项
initial_chunk_size	新临时智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
offset	到达新临时智能大对象空间的初始块的磁盘分区内或设备内的偏移量，以 KB 为单位。	
path_name	临时智能大对象空间的初始块的磁盘分区或无缓冲设备。	
tempsbspace	要创建的临时智能大对象空间的名称。	

用法

使用 create with_check sbspace 参数来检查指定的路径名，如果该路径不存在，则返回错误。

这个函数等同于带有用于创建临时智能大对象空间的 -t 选项的 onspaces -c -S 命令。

示例

下例创建大小为 20 MB，偏移量为 0 的临时智能大对象空间：

```
EXECUTE FUNCTION task ("create tempsbspace","tempsbspace3",
"$INFORMIXDIR/WORK/tempsbspace3","20 M","0");
```

相关链接

create sbspace 参数: 创建智能大对象空间 (SQL 管理 API) 在第612页

onspaces -c -S: 创建智能大对象空间 在第371页

《*SinoDB 管理员指南*》: 临时智能大对象空间

create tempsbspace from storagepool 参数: 从存储池创建临时智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 create tempsbspace from storagepool 参数来从存储池中的条目创建临时智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"create tempsbspace from storagepool "
, "tempsbspac"e, "initial_chunk_siz"e
);
```

元素	描述	关键注意事项
initial_chunk_size	新智能大对象空间的初始块的大小，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。
tempsbspace	临时智能大对象空间的名称。	临时智能大对象空间名称必须是唯一的，且不可超过 128 字节。该名称必须以字母或下划线开始，且必须仅包含字母、数字、下划线或 $ 字符。

示例

以下命令创建名为 tempsbspace5 的临时智能大对象空间。该临时智能大对象空间的大小为 240 MB。

```
EXECUTE FUNCTION task("create tempsbspace from storagepool",
 "tempsbspace5", "240 MB");
```

defragment 参数：动态地对分区扩展块取消分段 (SQL 管理 API)

admin() 或 task() 函数使用 defragment 参数来对表或索引进行取消分段以合并不连续的扩展数据块。

对表取消分段可使数据行更紧密，以避免分区头页溢出问题，并可提升性能。

在对分区取消分段之前，请先查阅《*SinoDB*® 管理员指南》中的“对分区取消分段”。

语法

您可使用以下语法指定 defragment 参数或 defragment partnum 参数：

```
EXECUTE FUNCTION
{ admin | task }
( " defragment " , { " database : owner . table " } ) ;
EXECUTE FUNCTION
{ admin | task }
( " defragment partnum " , , partition_number " ) ;
```

元素	描述	关键注意事项
database	包括想要取消分段的表或索引的数据库的名称。	
owner	表的所有者的用户 ID。	
table	要取消分段的表的名称。	
partition_number	要取消分段的一个或多个分区号。	使用以逗号分隔的分区号列表来指定多个分区。

用法

使用 defragment 参数来对特定表进行取消分段。使用 defragment partnum 参数来对一个或多个特定磁盘分区进行取消分段。

有关取消分段的信息存储在共享内存中。使用 *oncheck -pt and -pT: Display tblspaces for a Table or Fragment* 命令来显示特定表或分段的块数量的信息。使用 *onstat -g defragment* 命令: 显示分区扩展数据块取消分段信息。

如果取消分段请求减少了至少 1 个扩展数据块，则该请求返回 0（成功），即使分区中还有多个扩展数据块。

如果分区有单个扩展数据块，那么即使没有合并任何扩展数据块，取消分段请求还是返回 0 以表示请求成功。

示例

要对 stores_demo 数据库中 custmer 表进行取消分段，请使用下列函数之一：

```
EXECUTE FUNCTION task("defragment","stores_demo:informix.customer");
```

```
EXECUTE FUNCTION admin("defragment","stores_demo:informix.customer");
```

要对索引取消分段，您必须指定该索引的分区号，如以下两个示例所示：

```
EXECUTE FUNCTION task("defragment partnum","2097154");
```

```
EXECUTE FUNCTION admin("defragment partnum","2097154");
```

要对分区列表取消分段，使用下列函数之一：

```
EXECUTE FUNCTION task("defragment partnum", "16777217,28477346");
```

```
EXECUTE FUNCTION admin("defragment partnum", "16777217,28477346");
```

drop blobspace 参数: 删除 Blob 空间 (SQL 管理 API)

admin() 或 task() 函数使用 drop blobspace 参数来删除指定的 Blob 空间。.

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop blobspace"
, "blobspac"e
);
```

元素	描述	关键注意事项
blobspace	要删除的 Blob 空间的名称。	必须是现有的 Blob 空间。 在删除 Blob 空间之前，请删除包括引用该 Blob 空间的 TEXT 或 BYTE 列的所有表。

用法

这个函数等同于 onspaces -d 命令。

示例

下列删除名为 blobspace3 的 Blob 空间：

```
EXECUTE FUNCTION task("drop blobspace","blobspace3");
```

相关链接

onspaces -d: 删除空间 在第379页

drop blobspace to storagepool 参数: 从空 BLOB 空间归还空间到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop blobspace to storagepool 参数来从空 Blob 空间归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"drop blobspace to storagepool"
, "blobspac"e
);
```

元素	描述	关键注意事项
blobspace	空 Blob 空间的名称。	

示例

下列命令删除名为 blob2 的空 Blob 空间并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop blobspace to storagepool", "blob2");
```

drop chunk 参数: 删除块 (SQL 管理 API)

admin() 或 task() 函数使用 drop chunk 参数来删除数据库空间、Blob 空间或智能大对象空间中的指定块。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop chunk"
,"space_nam"e [ , "path_nam"e  ,"offse"t ]
);
```

元素	描述	关键注意事项
offset	磁盘分区内或无缓冲设备内，到达正在删除的数据库空间、Blob 空间或智能大对象空间的初始块的偏移量，以 KB 为单位。	起始偏移量为无符号整数，必须是等于或大于 0。起始偏移量加上块大小不可超过最大块大小。最大偏移量是 4 TB。 另参阅 *admin() 和 task() 参数大小规范* 在第572页。
path_name	正删除的数据库空间、Blob 空间或智能大对象空间的初始块的磁盘分区或无缓冲设备。	该块必须是现有的无缓冲设备或缓冲的文件。指定路径名时，可使用全路径名或相对路径名。然而，如果使用相对路径名，则其必须相对于初始化数据库服务器时的当前目录。

元素	描述	关键注意事项
space_name	要从其中删除块的数据库空间、Blob 空间或智能大对象空间的名称。	当数据库服务器处于联机或静默模式时，您可从数据库空间、临时数据库空间或智能大对象空间删除块。 仅当数据库服务器处于静默模式时，您才可从 Blob 空间删除块。

用法

这个函数等同于 onspaces -d 命令。

示例

以下示例从名为 dbspc3 的数据库空间删除位于偏移量 5200 KB 处的块：

```
EXECUTE FUNCTION task("drop chunk", "dbspc3","\\.\e:","5200");
```

相关链接

onspaces -d: 删除数据库空间、*BLOB* 空间或智能大对象空间中的块 在第377页

drop chunk to storagepool 参数：将空块的空间归还到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop chunk to storagepool 参数来从空块归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"drop chunk to storagepool"
,"space_nam"e, "path_nam"e,"offse"t
);
```

元素	描述	关键注意事项
space_name	块所在其中的存储池空间的名称。	
path_name	块的路径	
offset	块的偏移量，以 KB 为单位。	

示例

以下命令在名为 bigdbs 数据库空间中删除空块，并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop chunk to storagepool", "bigdbs", "/dev/
rawdisk23",
 "100 KB");
```

drop database 参数: 删除数据库 (SQL 管理 API)

admin() 或 task() 函数使用 drop database 参数来删除数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop database"
,"database_nam"e
);
```

元素	描述	关键注意事项
database_name	数据库名称。	

用法

这个函数等同于 DROP DATABASE 语句。这个函数删除整个数据库，包括系统目录表、对象和数据。

示例

以下示例删除名为 demodbs 的数据库：

```
EXECUTE FUNCTION task("drop database","demodbs");
```

相关链接

《SinoDB SQL 指南: 语法》: DROP DATABASE 语句

drop dbspace 参数: 删除数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 drop dbspace 参数来删除指定的数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop dbspace"
, "dbspac"e
);
```

元素	描述	关键注意事项
dbspace	要删除的数据库空间的名称。	数据库空间必须存在。 在删除数据库空间之前，请删除所有以前在该数据库空间中创建的数据库和表。

用法

这个函数等同于 onspaces -d 命令。

示例

以下示例删除名为 dbspace4 的数据库空间：

```
EXECUTE FUNCTION task("drop dbspace","dbspace4");
```

相关链接

onspaces -d: 删除空间 在第379页

drop dbspace to storagepool 参数：从空数据库空间归还空间到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop dbspace to storagepool 参数来从空数据库空间归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"drop dbspace to storagepool"
, "dbspac"e
);
```

Element	Description	Key Considerations
dbspace	空数据库空间的名称。	

示例

以下命令删除名为 dbs5 的空数据库空间，并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop dbspace to storagepool", "dbs5");
```

drop log 参数：删除逻辑日志 (SQL 管理 API)

admin() 或 task() 函数使用 drop log 参数来删除指定的逻辑日志。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop log"
,"log_numbe"r
);
```

元素	描述	关键注意事项
log_number	逻辑日志文件编号。	该编号必须是大于或等于 0 的无符号整数。

用法

使用这个函数来删除单个逻辑日志文件。

数据库服务器随时都要求最少三个逻辑日志文件。如果数据库服务器只有三个逻辑日志文件，则不可删除日志文件。

> 重要： 在删除前三个逻辑日志文件的任何一个之前，必须添加新逻辑日志文件并执行逻辑日志文件备份。必须使用 ontape -a 命令或 ontape -c 命令执行备份。在添加新逻辑日志文件并执行备份之后，才可使用 onparams -d -l*lognum* 删除前三个逻辑日志文件。

日志文件的状态决定能否删除日志文件，并决定删除日志文件时数据库服务器采取的动作：

- 如果删除状态为“新添加”（A）还没有写过的日志文件，则数据库服务器删除该日志文件并立即释放该空间。
- 如果删除状态为“用户”（U）或“释放”（F）的已使用日志文件，则数据库服务器将该日志文件名标记为“删除”（D）。在您对包含日志文件和根数据库空间的数据库空间执行了 0 级备份之后，数据库服务器删除该日志文件名并释放该空间。
- 不可删除当前正在使用（C）或包含最后检查点记录（L）的日志文件。

可从 onstat -l 命令的 number 字段获得日志编号。日志编号可能是无序的。

这个函数等同于 onparams -d -l *lognum* 命令。

示例

以下示例删除文件号 2 的逻辑日志：

```
EXECUTE FUNCTION task("drop log","2");
```

以下示例通过基于块号查找日志号来删除指定块的日志：

```
SELECT task("drop log", number) FROM sysmaster:syslogfil WHERE chunk = 1;
```

相关链接

onparams -d -l lognum: 删除逻辑日志文件 在第352页

drop plogspace: 删除物理日志空间 (SQL 管理 API)

admin() 或 task() 函数使用 drop plogspace 参数来删除物理日志空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop plogspace"
);
```

用法

要删除的物理日志空间必须是空的。例如：如果您将物理日志移出物理日志空间并通过运行 onparams -p 命令移入数据库空间，则可删除该物理日志空间。或者，通过创建新的物理日志空间将物理日志空间移到不同的块。旧的物理日志空间将自动地被移除。

这个函数等同于 onspaces -d 命令。

示例

以下示例删除物理日志空间：

```
EXECUTE FUNCTION task("drop plogspace");
```

相关链接

onspaces -d: 删除数据库空间、*BLOB* 空间或智能大对象空间中的块 在第377页

drop sbspace 参数：删除智能大对象空间 (SQL 管理 API)

admin() 或 task() 函数使用 drop sbspace 参数来删除指定的智能大对象空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop sbspace"
, "sbspac"e
);
```

元素	描述	关键注意事项
sbspace	要删除的智能大对象空间的名称。	该智能大对象空间必须存在。 在删除智能大对象空间之前，请删除包含引用该智能大对象空间的 BLOB 或 CLOB 列的所有表。

用法

这个函数等同于 onspaces -d 命令。

示例

以下示例删除名为 sbspace3 的智能大对象空间：

```
EXECUTE FUNCTION task("drop dbspace","sbspace3");
```

相关链接

onspaces -d: 删除空间 在第379页

drop sbspace to storagepool 参数：从空智能大对象空间归还空间到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop sbspace to storagepool 参数来从空智能大对象空间归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
,"drop sbspace to storagepool "
"sbspac"e
);
```

元素	描述	关键注意事项
sbspace	空智能大对象空间的名称。	

示例

以下命令删除名为 sbspace8 的空智能大对象空间，并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop sbspace to storagepool", "sbspace8");
```

drop tempdbspace 参数: 删除临时数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 drop tempdbspace 参数来删除指定的临时数据库空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("drop tempdbspace"
, "tempdbspac"e
);
```

元素	描述	关键注意事项
tempdbspace	要删除的临时数据库空间的名称。	该临时数据库空间必须存在。 在删除临时数据库空间之前，请删除以前在临时数据库空间中创建的所有数据库和表。

用法

这个函数等同于 onspaces -d 命令。

示例

以下示例删除名为 tdbspace2 的临时数据库空间：

```
EXECUTE FUNCTION task("drop tempdbspace","tdbspace2");
```

相关链接

onspaces -d: 删除空间 在第379页

drop tempdbspace to storagepool 参数: 从空临时数据库空间归还空间到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop tempdbspace to storagepool 参数来从空临时数据库空间归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"drop tempdbspace to storagepool"
, "tempdbspac"e
);
```

元素	描述	关键注意事项
tempdbspace	空临时数据库空间的名称。	

示例

以下命令删除名为 tempdbs1 的空临时数据库空间，并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop tempdbspace to storagepool", "tempdbs1");
```

drop tempsbspace to storagepool 参数：从空临时智能大对象空间归还空间到存储池 (SQL 管理 API)

admin() 或 task() 函数使用 drop tempsbspace to storagepool 参数来从空智能大对象空间归还空间到存储池。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"drop tempsbspace to storagepool"
, "tempsbspac"e
);
```

元素	描述	关键注意事项
tempsbspace	空临时智能大对象空间的名称。	

示例

以下命令删除名为 tempsbspace3 空临时智能大对象空间，并将所有释放的空间添加到存储池。

```
EXECUTE FUNCTION task("drop tempsbspace to storagepool", "tempsbspace3");
```

export config 参数：导出配置参数值 (SQL 管理 API)

admin() 或 task() 函数使用 export config 参数来导出包含所有配置参数及其当前值的文件。

语法

```
EXECUTE FUNCTION { admin  | task }
(
"export config"
, "file_pat"h
);
```

表 189: export config 命令元素

元素	描述	关键注意事项
file_path	文件的全路径名	不要添加扩展名。

用法

该 SQL 管理 API export 命令自动地创建 ASCII 文件，并赋予您在命令中指定的名称。该文件的格式与 onconfig.std 文件的格式相同。

您必须指定全路径名。不可指定相对路径。

这个命令等同于 onmode -we 命令。

示例

以下命令将所有配置参数及其当前值导出到 /tmp 目录中名为 cfg_12 的文件中：

```
EXECUTE FUNCTION task("export config", "/tmp/cfg_12");
```

相关链接

更改 *onconfig* 文件 在第42页

import config 参数: 导入配置参数值 *(SQL* 管理 *API)* 在第640页

onmode -we: 导出包含当前配置参数的文件 在第347页

file status 参数: 显示消息日志文件的状态 (SQL 管理 API)

admin() 或 task() 函数使用 file status 参数来指定联机、ON-Bar 活动或 ON-Bar 调试的消息日志文件的状态。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("file status"," file_path"
);
```

元素	用途	关键注意事项
file_path	联机、ON-Bar 活动或 ON-Bar 调试的消息日志文件的全路径名。	

示例

以下示例展示可用于显示 /usr/informix/online.log 文件状态的参数：

```
execute function task("file status", "/usr/informix/online.log");
```

服务器显示以下信息：

```
(expression)  File name          = /tmp/x
              Is File            = 1
              Is Directory       = 0
              Is Raw Device      = 0
              Is Block Device    = 0
              Is Pipe            = 0
              File Size          = 554
              Last Access Time   = 11/29/2010 21:55:02
              Last Modified Time = 11/29/2010 21:51:45
              Status Change Time = 11/29/2010 21:51:45
              User Id            = 200
              Group id           = 102
```

```
                File Flags          = 33206
```

相关链接

message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页

grant admin 参数: 授予运行 SQL 管理 API 命令的权限

admin() 或 task() 函数使用 grant admin 参数来授予运行 SQL 管理 API 命令的权限。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"grant admin"
,"user_name" ,"privilege_group"
);
```

元素	描述	关键注意事项
user_name	授予权限的用户名。	
privilege_group	权限组名。	有关权限组的列表，请参阅 *SQL 管理 API 门户: 按权限组划分参数* 在第573页。

用法

可授予个别用户通过运行 SQL 管理 API 命令来管理数据库服务器的权限。具有这些权限的用户可通过使用 SinoDB® 开放管理工具 (OAT) 或直接连接方式，以他们的用户名连接到数据库服务器运行 SQL 管理 API 命令。

仅用户 informix 或具有 SQL 管理 API 命令的 ADMIN 或 GRANT 权限的用户可以使用 grant admin 参数。

示例

以下命令授予用户 Bob 运行备份和复原 SQL 管理命令的权限：

```
EXECUTE FUNCTION task("grant admin", "Bob", "BAR");
```

相关链接

DBCREATE_PERMISSION 配置参数 在第87页

ha make primary 参数: 更改辅助服务器的模式 (SQL 管理 API)

admin() 或 task() 函数使用 ha make primary 参数来将指定的辅助服务器更改为主服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "ha make primary"  | "ha make primary force" }
, "database_serve"r);
```

元素	描述	关键注意事项
database_server	数据库服务器的名称。	该名称必须在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中定义，或者将其定义为 Enterprise Replication 组名。

用法

根据辅助服务器类型，这个函数有不同的结果：

- HDR 辅助服务器：关闭当前主服务器并使 HDR 辅助服务器成为主服务器。
- RS 辅助服务器：将 RS 辅助服务器更改为标准服务器。
- SD 辅助服务器：使 SD 辅助服务器成为新的主服务器。

当主服务器与不活动辅助服务器之间有连接时，可使用 ha make primary 参数来将不活动的辅助服务器更改为主服务器。

无论辅助服务器是否连接到主服务器，都可使用 ha make primary force 参数来将不活动辅助服务器更改为主服务器。如果连接是活动的，则函数成功，然而，如果您在 SD 辅助服务器上使用 force 参数运行该函数，则共享磁盘子系统可能遭损坏。

这个函数等同于 onmode -d make primary 命令。

示例

以下示例将名为 ids_stores2 的 HDR 辅助服务器转换为主服务器：

```
EXECUTE FUNCTION task("ha make primary","ids_stores2");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha rss 参数: 创建 RS 辅助服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha rss 参数来创建远程独立（RS）辅助服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha rss" , "primary_serve"r [ ,"passwor"d ]
);
```

元素	描述	关键注意事项
password	要设置或更改的密码。	仅在第一次连接尝试期间使用密码。在主服务器和辅助服务器连接之后，不可更改该密码。
primary_server	主数据库服务器名。	该名称必须在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中定义，或者定义为 Enterprise Replication 组名。

用法

在标准服务器或静默 HDR 辅助服务器上运行这个函数来将其转换成为 RS 辅助服务器。

这个函数等同于 onmode -d RSS 命令。

示例

以下示例将标准服务器转换为 RS 辅助服务器，主服务器名为 ids_stores：

```
EXECUTE FUNCTION task("ha rss","ids_stores");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha rss add 参数：添加 RS 辅助服务器到主服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha rss add 参数来关联主服务器与远程独立（RS）辅助服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha rss add"  ,  "ha_alia"s [ , "passwor"d ] ) ;
```

元素	描述	关键注意事项
password	要设置或更改的密码。	仅在第一次连接尝试期间使用密码。在主服务器与辅助服务器连接之后，不可更改该密码。
ha_alias	要转换为 RS 辅助服务器的数据库服务器的高可用性别名。	该名称必须在 ALIAS 配置参数中定义，或者定义作为 Enterprise Replication 组名。

用法

从已建立的主服务器运行这个函数来创建 RS 辅助服务器，并在 sysha 数据库中注册该 RS 辅助服务器名。

这个函数等同于 onmode -d add RSS 命令。

示例

以下示例将高可用性别名为 ids_stores2 的作为 RS 辅助服务器与主服务器关联：

```
EXECUTE FUNCTION task("ha rss add","ids_stores2");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha rss change 参数：更改 RS 辅助服务器的密码 (SQL 管理 API)

admin() 或 task() 函数使用 ha rss change 参数来更改指定的 RS 辅助服务器的连接密码。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha rss change","secondary_serve"r  , "passwor"d
);
```

元素	描述	关键注意事项
password	要设置或更改的密码。	仅在第一次连接尝试期间使用密码。在主服务器与辅助服务器连接之后，不可更改该密码。
secondary_server	要转换为 RS 辅助服务器的数据库服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中，或者定义为 Enterprise Replication 组名。

用法

在已建立的主服务器上运行这个函数来更改主服务器与辅助服务器之间连接的密码。

这个函数等同于 onmode -d change RSS 命令。

示例

以下示例将 RS 辅助服务器的密码更改为 secure：

```
EXECUTE FUNCTION task("ha rss change","ids_stores2","secure");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha rss delete 参数：删除 RS 辅助服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha rss delete 参数来停止复制并删除 RS 辅助服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha rss delete","secondary_serve"r
```

);

元素	描述	关键注意事项
secondary_server	要转换为 RS 辅助服务器的数据库服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数，或者定义为 Enterprise Replication 组名。

用法

从已建立的主服务器运行这个函数来将 RS 辅助服务器转换为标准服务器并删除该 RS 辅助服务器。

这个函数等同于 onmode -d delete RSS 命令。

示例

以下示例删除名为 ids_stores2 的 RS 辅助服务器：

```
EXECUTE FUNCTION task("ha rss delete","ids_stores2");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha sds clear 参数：停止共享磁盘复制 (SQL 管理 API)

admin() 或 task() 函数使用 ha sds clear 参数来停止复制到共享磁盘（SD）辅助服务器并将主服务器转换成为标准服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha sds clear"  ,"primary_serve"r );
```

元素	描述	关键注意事项
primary_server	要转换成为标准服务器的主服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数，或者定义为 Enterprise Replication 组名。

用法

在已建立地主服务器上运行这个函数来停止复制到 SD 辅助服务器。

这个函数等同于 onmode -d clear SDS primary 命令。

示例

以下示例停止从名为 ids_stores 的主服务器复制到 SD 辅助服务器：

```
EXECUTE FUNCTION task("ha sds clear","ids_stores");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha sds primary 参数：将 SD 辅助服务器转换为主服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha sds primary 参数来将共享磁盘（SD）辅助服务器更改为主服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "ha sds primary"  | "ha sds primary force" }
,"secondary_serve"r
);
```

元素	描述	关键注意事项
secondary_server	要设置为主服务器的 SD 辅助服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中，或者定义为 Enterprise Replication 组名。

用法

在已建立的 SD 辅助服务器上运行这个函数以转换成为主服务器。

如果 SD 辅助服务器连接到主服务器，则使用 ha sds primary 参数将不活动的 SD 辅助服务器转换为主服务器。

无论 SD 辅助服务器是否连接到主服务器，都可使用 ha sds primary force 参数将不活动的 SD 辅助服务器转换成为主服务器。如果会话是活动的，则调用成功，但共享磁盘子系统可能损坏。

这个函数等同于 onmode -d make primary 命令。

示例

以下示例将名为 ids_stores3 的 SD 辅助服务器转换成为主服务器：

```
EXECUTE FUNCTION task("ha sds primary","ids_stores3");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha sds set 参数：创建共享磁盘主服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha sds set 参数来定义要复制到共享磁盘（SD）辅助服务器的主服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "ha sds set"  | "ha sds set force" }
,"primary_serve"r
);
```

元素	描述	关键注意事项
primary_server	要设置作为主服务器的数据库服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中，或者

元素	描述	关键注意事项
		定义作为 Enterprise Replication 组名。

用法

在标准服务器上运行这个函数以将其定义为 SD 辅助服务器的主服务器。

如果 SD 辅助服务器连接到不活动的标准服务器，可使用 ha sds set 参数将该不活动的标准服务器定义为主服务器。

无论是否有 SD 辅助服务器连接到不活动的标准服务器，可使用 ha sds set force 参数将该不活动的标准服务器定义为主服务器。如果会话是活动的，则调用成功，但共享磁盘子系统可能损坏。

这个函数等同于 onmode -d set SDS primary 命令。

示例

以下示例将名为 ids_stores 的标准服务器转换为主服务器：

```
EXECUTE FUNCTION task("ha sds set","ids_stores");
```

相关链接

onmode -d: 设置高可用性服务器的特性 在第335页

ha set idxauto 参数：复制索引到辅助服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha set idxauto 参数来控制索引是否自动地复制到辅助服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "ha set idxauto off"  | "ha set idxauto on" }
);
```

用法

在已建立的主服务器上运行这个函数来启用或禁用向辅助服务器的自动索引复制。

您可在任何类型的主服务器上运行这个函数。

这个函数等同于 onmode -d idxauto 命令。

示例

以下示例启用自动索引复制：

```
EXECUTE FUNCTION task("ha set idxauto on");
```

相关链接

onmode -d 命令：使用数据复制来复制索引 在第336页

ha set ipl 参数: 在主服务器上构建日志索引 (SQL 管理 API)

admin() 或 task() 函数使用 ha set ipl 参数来控制是否在主服务器上构建日志索引。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "ha set ipl off"  | "ha set ipl on" }
);
```

用法

在已建立的主服务器上运行这个函数来启用或禁用索引构建的日志记录。这个函数重置 ONCONFIG 文件中的 LOG_INDEX_BUILDS 配置参数的值。

您可在任何类型的主服务器上运行这个函数。

这个函数等同于 onmode -wf LOG_INDEX_BUILDS 命令。

示例

以下示例启用索引构建的日志记录：

```
EXECUTE FUNCTION task("ha set ipl on");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

ha set primary 参数: 定义 HDR 主服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha set primary 参数来定义高可用性数据复制（HDR）主服务器，并指定辅助服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha set primary","secondary_serve"r
);
```

元素	描述	关键注意事项
secondary_server	要连接到的 HDR 辅助服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中，或者定义为 Enterprise Replication 组名。

用法

在标准服务器上运行这个函数以将其转换成为 HDR 主服务器，并连接到指定的 HDR 辅助服务器。如果连接成功，则开始复制。

这个函数等同于 onmode -d primary 命令。

示例

以下示例将名为 ids_stores 的标准服务器转换成为 HDR 主服务器：

```
EXECUTE FUNCTION task("ha set primary","ids_stores");
```

相关链接

onmode -d: 设置数据复制类型 在第334页

ha set secondary 参数：定义 HDR 辅助服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha set secondary 参数来定义高可用性数据复制（HDR）辅助服务器并指定主服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha set secondary","primary_serve"r
);
```

元素	描述	关键注意事项
primary_server	要连接到的 HDR 主服务器的名称。	该名称必须定义在 DBSERVERNAME 或 DBSERVERALIASES 配置参数中，或者定义为 Enterprise Replication 组名。

用法

在标准服务器上运行这个函数以将其转换成为 HDR 辅助服务器，并连接到指定的 HDR 主服务器。如果连接成功，则开始复制。

这个函数等同于 onmode -d secondary 命令。

示例

以下示例将标准服务器转换成为 HDR 辅助服务器，主服务器名为 ids_stores：

```
EXECUTE FUNCTION task("ha set secondary","ids_stores");
```

相关链接

onmode -d: 设置数据复制类型 在第334页

ha set standard 参数：将 HDR 服务器转换成为标准服务器 (SQL 管理 API)

admin() 或 task() 函数使用 ha set standard 参数来将高可用性数据复制（HDR）主服务器或辅助服务器转换成为标准服务器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha set standard"
```

);

用法

在 HDR 主服务器或辅助服务器上运行这个函数来将其转换成为标准服务器。删除主服务器与辅助服务器之间的连接并停止复制。不更改 HDR 对中的其他服务器模式。

这个函数等同于 onmode -d standard 命令。

示例

以下示例将 HDR 辅助服务器转换成为标准服务器：

```
EXECUTE FUNCTION task("ha set standard");
```

相关链接

onmode -d: 设置数据复制类型 在第334页

ha set timeout 参数：更改 SD 辅助服务器超时 (SQL 管理 API)

admin() 或 task() 函数使用 ha set timeout 参数来更改主服务器等待共享磁盘（SD）辅助服务器响应的时间量（以秒为单位）。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"ha set timeout" ,"second"s
) ;
```

元素	描述	关键注意事项
seconds	主服务器与 SD 辅助服务器断开之前等待的秒数。	该值必须是以下范围内的正整数： 从 2 到 2 147 483 647

用法

在已建立的共享磁盘主服务器上运行这个函数，以指定主服务器等待来自 SD 辅助服务器的日志位置响应的秒数。如果在指定的时间内未接收到 SD 辅助服务器发出的日志位置响应，则主服务器与 SD 辅助服务器断开并继续。如果在等待 SD 辅助服务器时页清空已经超时，则在等待指定的秒数之后，主服务器将开始移除 SD 辅助服务器。

这个函数重置 ONCONFIG 文件中的 SDS_TIMEOUT 配置参数的值。

这个函数等同于 onmode -wf SDS_TIMEOUT 命令。

示例

以下示例设置超时期间为 5 秒：

```
EXECUTE FUNCTION task("ha set timeout","5");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

import config 参数: 导入配置参数值 (SQL 管理 API)

admin() 或 task() 函数使用 import config 参数来导入包含一个或多个可动态地更新的配置参数的文件，并应用这些新值。

语法

```
EXECUTE FUNCTION { admin  | task }
(
"import config"
, "file_pat"h
);
```

表 190: import config command elements

元素	描述	关键注意事项
file_path	以前导出包含一个或多个可动态地更新的配置参数的名称和值的文件的全路径名。	

用法

可动态地更新的配置参数是那些您可使用 onmode -wf 或 onmode -wm 命令为会话更改的参数。

必须指定全路径名。不可指定相对路径。

这个命令等同于 onmode -wi 命令。

示例

以下命令导入 /tmp 目录中名为 cfg_12 的文件：

```
EXECUTE FUNCTION task("import config", "/tmp/cfg_12");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
更改 *onconfig* 文件 在第42页
export config 参数: 导出配置参数值 (SQL 管理 API) 在第628页
onmode -wi: 导入配置参数文件 在第348页

index compress repack shrink 参数: 优化 B 型树索引的存储 (SQL 管理 API)

admin() 或 task() 函数使用 index compress repack shrink 参数来压缩分离的 B 型树索引、合并空闲空间（重新打包），以及归还分区中空闲空间（收缩）

语法：索引压缩命令参数

```
EXECUTE FUNCTION { admin  | task }
("index
{  compress  [repack]  [shrink]  [parallel]  | repack [shrink]  [parallel]  | shrink } "
,"index_nam"e,  "database_nam"e,"owne"r
);
```

命令参数

The following table contains a brief explanation of each argument. 下表包含每个参数的简要说明

表 191: 索引压缩操作的参数

参数	描述
compress	压缩索引。
parallel	并行运行压缩或重新打包操作。为表或分段列表的每个分段启动一个线程，并跨越这些分段并行地运行该操作。
repack	通过将数据搬移到索引的前部来合并空闲空间。
shrink	将索引末端的空闲空间归还到数据库空间，以减少索引的总大小。

命令元素

下表显示可用来压缩、重新打包和收缩索引的元素

表 192: 索引压缩命令元素

元素	描述	关键注意事项
index_name	想要压缩的索引的名称。	必要的。 必须使用与系统目录表中相同的大写或小写字母。
database_name	包含想要压缩的索引的数据库名。	可选的 如果未指定数据库，则数据库服务器使用当前的数据库。 如果指定数据库名，则必须使用与系统目录表中相同的大写或小写字母。
owner	包含想要压缩的索引的数据库的所有者名。	可选的。 如果不指定所有者，数据库服务器使用当前的所有者。 如果指定所有者名，则必须使用与系统目录表中相同的大写或小写字母。

用法

您可压缩分段表或未分段表上分离的 B 型树索引。不可压缩附加索引。

要压缩，则索引必须有至少 2000 个键。如果索引内的分段没有至少 2000 个键， 则当数据库服务器创建索引时不压缩索引或分段。即使有新的键加到索引，索引还是保持不压缩。如果您要压缩索引，请运行带有 index compress 参数的 SQL 管理 API task() 或 admin() 函数。

要确定索引是否包含最小键数，请运行 oncheck -pT 命令并查看 Number of keys 字段中的信息。

通常，在压缩操作之后执行重新打包操作，并在重新打包操作之后执行收缩操作。

压缩操作仅压缩索引的叶子（底层）。

您可取消操作，例如：在 DB-Access 中按下 CTRL-C。

您不可解压缩索引。如果您要解压缩索引，您可删除压缩的索引并重新创建它。

示例

以下命令并行地压缩、重新打包和收缩索引。

```
EXECUTE FUNCTION task("index compress repack shrink parallel",
```

```
"ind5", "customer", "jayson");
```

index estimate_compression 参数: 估计索引压缩 (SQL 管理 API)

admin() 或 task() 函数使用 index estimate_compression 参数来估计您是否可通过压缩 B 型树索引节省磁盘空间。

语法: index estimate_compression 命令参数

```
EXECUTE FUNCTION { admin  | task }
("index estimate_compression "
, "index_nam"e, "database_nam"e,"owne"r
);
```

命令元素

下表显示您可用来估计索引压缩的元素。

表 193: Index estimate_compression 命令元素

元素	描述	关键注意事项
index_name	您想要估计压缩成效的索引的名称。	必要的。 您必须使用与系统目录表中相同的大写或小写字母。
database_name	包含该索引的数据库名。	可选的。 如果未指定数据库，则数据库服务器使用当前的数据库。 如果指定数据库名，则必须使用与系统目录表中相同的大写或小写字母。
owner	包含该索引的数据库的所有者名。	索引的可选项。 如果未指所有者，则数据库服务器使用当前的所有者。 如果指定所有者，则必须使用与系统目录表中相同的大写或小写字母。

用法

您仅可估计分段的或非分段的表上分离的 B 型树索引的压缩。

估计压缩操作显示索引名、可归档的估计压缩比率、当前压缩比率，以及百分率损益的估计。如果索引未压缩，则当前比率为 0.0 百分比。

示例

The following command estimates compression benefits for an index named ind4 in the customer database for which anjul is the owner.以下命令估计 customer 数据库中名为 ind4 的索引压缩成效，数据库所有者是 anjul。

```
EXECUTE FUNCTION task("index estimate_compression","ind4",
"customer","anjul");
```

相关链接

估计压缩操作的输出 *(SQL 管理 API)* 在第703页

message log delete 参数: 删除消息日志文件 (SQL 管理 API)

admin() 或 task() 函数使用 message log delete 参数或 file delete 参数来指定要删除的特定联机、ON-Bar 活动或 ON-Bar 调试消息日志。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "message log delete"  | "file delete" } ," file_path"
);
```

元素	用途	关键注意事项
file_path	特定联机、ON-Bar 活动或 ON-Bar 调试消息日志的全路径名。	

示例

以下示例显示您可用来删除 /usr/informix/online.log 文件的参数:

```
execute function task("message log delete", "/usr/informix/online.log");
```

```
execute function task("file delete", "/usr/informix/online.log");
```

相关链接

message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API) 在第644页
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页

message log rotate 参数: 轮换消息日志文件 (SQL 管理 API)

admin() 或 task() 函数使用 message log rotate 参数或 file rotate 参数来指定要轮换的特定联机、ON-Bar 活动或 ON-Bar 调试消息日志文件，并指明要轮换的最大消息日志数。

当消息日志文件轮换时，数据库服务器切换到新的联机消息日志文件，并将先前日志文件的 ID 号增加 1。当到达最大日志文件数时，删除最高 ID 的日志文件。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "message log rotate"  | "file rotate" } ," file_path"
[ maximum_version ]
);
```

元素	用途	关键注意事项
file_path	服务器要轮换的联机、ON-Bar 活动或 ON-Bar 调试消息日志文件的全路径名，例如：/usr/informix/online.log。	
maximum_version	最高 ID 的日志文件。这是服务器将要轮换的最大消息日志版本。	

示例

以下示例显示您可用来轮换最大 52 个 /usr/informix/online.log 文件的参数：

```
execute function task("message log rotate", "/usr/informix/online.log",52);
```

```
execute function task("file rotate", "/usr/informix/online.log",52);
```

当数据库服务器轮换这些文件时，服务器删除该文件的版本 52。版本 51 成为版本 52，版本 50 成为版本 51，依此类推。新的联机日志成为版本 1。

相关链接

message log truncate 参数: 删除消息日志文件的内容 *(SQL 管理 API)* 在第644页
message log delete 参数: 删除消息日志文件 *(SQL 管理 API)* 在第643页
file status 参数: 显示消息日志文件的状态 *(SQL 管理 API)* 在第629页
message log truncate 参数: 删除消息日志文件的内容 *(SQL 管理 API)* 在第644页
message log delete 参数: 删除消息日志文件 *(SQL 管理 API)* 在第643页
file status 参数: 显示消息日志文件的状态 *(SQL 管理 API)* 在第629页

message log truncate 参数: 删除消息日志文件的内容 (SQL 管理 API)

admin() 或 task() 函数使用 message log truncate 参数或file truncate 参数来指定要删节的特定联机、ON-Bar 活动或 ON-Bar 调试消息日志文件。数据库服务器删节消息日志文件时，其删除日志文件中的消息，但保留日志文件。.

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "message log truncate"  | "file truncate" } ," file_path"
);
```

元素	用途	关键注意事项
file_path	联机、ON-Bar 活动或 ON-Bar 调试消息日志文件的全路径名。	

示例

以下示例显示您可用来删节 /usr/informix/online.log 文件的参数：

```
execute function task("message log truncate", "/usr/informix/online.log");
```

```
execute function task("file truncate", "/usr/informix/online.log");
```

相关链接

message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log rotate 参数: 轮换消息日志文件 (SQL 管理 API) 在第643页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页
message log delete 参数: 删除消息日志文件 (SQL 管理 API) 在第643页
file status 参数: 显示消息日志文件的状态 (SQL 管理 API) 在第629页

modify chunk extend 参数: 扩展块的大小 (SQL 管理 API)

admin() 或 task() 函数使用 modify chunk extend 参数来以指定的最小量扩展块的大小。块必须标记为可扩展的。

```
EXECUTE FUNCTION
{ admin  | task }
("modify chunk extend "  ,"chunk_numbe"r,"extend_amoun"t ) ;
```

元素	描述	关键注意事项
chunk_number	The number of the chunk.	
extend_amount	要添加到块的最小空间量，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

在手动地或自动地扩展块之前，必须将块标记为可扩展的。使用带有 modify chunk extendable 参数的 admin() 或 task() 函数来将块标记为可扩展的。

modify chunk extend SQL 管理 API 命令是 adm_add_storage 任务的替代方式，当包含块的空间的可用页不足或耗尽时，服务器将运行该任务来自动地扩展块的大小。

不可扩展镜像空间中的块，且如果您在运行 modify chunk extend SQL 管理 API 命令时提供了镜像块的编号，则您将收到错误。

要识别镜像空间中的主块和镜像块，请在 onstat -d 命令输出中 flags 字段的位置 1 中查找 P（主块）或 M（镜像）。

根据页大小和该空间配置的创建大小和扩展大小，服务器可能取整到需要的大小。

示例

假设您的 onstat -d 命令输出显示块号 3 是镜像块，块号 4 不是镜像块。您不可扩展块号 3 的大小。然而，您可以修改块号 4。以下命令扩展块号 4 的大小 10000 KB：

```
EXECUTE FUNCTION task("modify chunk extend", "4", "10000");
```

相关链接

modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页
modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页
modify space sp_sizes 参数: 更改可扩展的存储空间的大小 (SQL 管理 API) 在第649页
modify space expand 参数: 扩大空间的大小 (SQL 管理 API) 在第648页
onstat -d 命令: 显示块信息 在第414页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页
modify space expand 参数: 扩大空间的大小 (SQL 管理 API) 在第648页

modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API)

admin() 或 task() 函数使用 modify chunk extendable 参数来指定未镜像数据库空间或临时数据库空间中的特定块可扩展。

```
EXECUTE FUNCTION
{ admin  | task }
("modify chunk extendable "  ,"chunk_number" ) ;
```

元素	描述	关键注意事项
chunk_number	块号。	

用法

如果块标记为可扩展，则二者之一：

- 当包含该块的未镜像数据库空间或临时数据库空间的可用页不足或耗尽时，服务器可自动地扩展该块。
- 您可使用带有 modify chunk extend 参数的 admin() 或 task() 函数来扩展该块的大小。

然而，如果数据库空间或临时数据库空间的扩展大小设置为 0，则服务器在那个空间里不可自动地扩展可扩展的块。在这种情况下，您仍可手动地扩展该块。

服务器自动地将那些从可扩展存储池条目分配的块标记为可扩展的。因此，您不需要将这些块标记为可扩展的。有关可扩展的存储池条目的信息，请参阅 *storagepool add 参数: 添加存储池条目 (SQL 管理 API)* 在第693页。

镜像空间中的块是不可扩展的。如果您尝试使镜像块成为可扩展的，您会收到错误。

要识别镜像空间中的主块和镜像块，请在 onstat -d 命令输出中的 flags 字段的位置 1 查找 P（主）或 M（镜像）。

示例

以下 onstat -d 输出的片段显示块号 3 是镜像块：

```
Chunks
address      chunk/dbs  offset   size     free     bpages   flags pathname
451191c8     1      1   0        225000   101572            PO-B-- /reg1/
rootchunk
451197d0     2      2   0        1250     1149              PO-B-- /reg1/dbs1
451199d0     3      3   0        1250     1149              PO-B-- /reg1/dbs2
46a36638     3      3   0        1250     0                 MO-B-- /reg1/
chunk2
45119bd0     4      4   0        1250     1149              PO-B-- /reg1/dbs3
```

因此，您不可扩展块号 3 的大小。然而，您可指定块号 4 为可扩展的，如下所示：

```
EXECUTE FUNCTION sysadmin:task("modify chunk extendable", "4");
```

相关链接

modify chunk extendable off 参数: 标记块为不可扩展 (SQL 管理 API) 在第647页
modify space sp_sizes 参数: 更改可扩展的存储空间的大小 (SQL 管理 API) 在第649页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页
modify space expand 参数: 扩大空间的大小 (SQL 管理 API) 在第648页
modify chunk extendable off 参数: 标记块为不可扩展 (SQL 管理 API) 在第647页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页

modify chunk extendable off 参数: 标记块为不可扩展 (SQL 管理 API)

admin() 或 task() 函数使用 modify chunk extendable off 参数来指定特定的块为不可扩展的。

```
EXECUTE FUNCTION
{ admin  | task }
("modify chunk extendable off"  ,"chunk_number" ) ;
```

元素	描述	关键注意事项
chunk_number	块号。	

用法

块的缺省状态为不可扩展的。如果您先前标记块为可扩展的，则您可更改其状态为不可扩展的。

如果块标记为不可扩展的：

- 当包含该块的空间的可用页不足或耗尽时，服务器不可自动地扩展该块。
- 您不可手动地扩展该块的大小。

如果存储池包含条目，服务器可通过添加另一块到存储空间来扩展该存储空间。

示例

以下示例指定您或服务器不可扩展块 9：

```
EXECUTE FUNCTION task("modify chunk extendable off", "9");
```

相关链接

modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页
modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页

modify config 参数 : 更改配置参数 (SQL 管理 API)

admin() 或 task() 函数使用 modify config 参数来更改内存中配置参数的值，直到您重新启动数据库服务器。使用 modify config persistent 参数来更改内存中配置参数的值，并在重新启动服务器之后在 onconfig 文件中保留该值。

语法

```
EXECUTE FUNCTION { admin  | task }
(
{  "modify config"  | "modify config persistent " }
,"configuration_parameter_nam"e
,"new_valu"e
);
```

表 194: modify config 命令元素

元素	描述	关键注意事项
configuration_parameter_name	想要更改的配置参数名。	
new_value	配置参数的新值。	有关配置参数的有效值的信息，请参阅数据库配置参数 在第41页。

用法

这个 SQL 管理 API 命令等同于使用 onmode -wm 或 -wf 命令来更改配置参数的值。

示例

以下命令将内存中 DYNAMIC_LOGS 配置参数的值更改为 2 作为当前使用：

```
EXECUTE FUNCTION task("modify config","DYNAMIC_LOGS",
"2");
```

以下命令更改 DYNAMIC_LOGS 配置参数的值以供当前使用。在重新启动服务器之后，更改的值保留在 onconfig 文件中。

```
EXECUTE FUNCTION task("modify config persistent","DYNAMIC_LOGS",
"2");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页
更改 *onconfig* 文件 在第42页

modify space expand 参数：扩大空间的大小 (SQL 管理 API)

当您不想等待 SinoDB® 自动地扩展空间时，可使用带有 modify space expand 参数的 admin() 或 task() 函数来立即扩展空间的大小。

```
EXECUTE FUNCTION
{ admin  | task }
(
"modify space expand"
,"space_nam"e,  "minimum_siz"e
);
```

元素	描述	关键注意事项
space_name	存储空间的名称。	
minimum_size	想要扩展空间的最小的大小。	请参阅 *admin() 和 task()* 参数大小规范 在第572页。

用法

modify space expand SQL 管理 API 命令通过扩展空间中的可扩展块或添加新块来立即扩展存储空间。空间的创建大小和扩展大小设置不影响这个操作。

添加到空间的实际 KB 数可能超过您需要的大小，其取决于一些因素，例如空间的页大小和存储池中可用条目的块大小设置。

存储池必须包含服务器可用来扩展空间的条目（例如裸设备、熟文件或目录）。

在您运行了 modify space expand SQL 管理 API 命令之后，SinoDB® 将通过扩展空间内可扩展块来首次尝试扩展空间。如果空间不包含任何可扩展块，则服务器使用存储池中的条目来扩展空间。

不可扩展镜像存储空间。

示例

以下命令将 dbspace5 扩展 10 MB：

```
EXECUTE FUNCTION task("modify space expand", "dbspace5", "10 MB");
```

相关链接

modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页

modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页

modify space sp_sizes 参数: 更改可扩展的存储空间的大小 (SQL 管理 API) 在第649页

modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页

create chunk from storagepool 参数: 从存储池创建块 (SQL 管理 API) 在第606页

modify space sp_sizes 参数: 更改可扩展的存储空间的大小 (SQL 管理 API)

admin() 或 task() 函数使用 modify space sp_sizes 参数来更改与扩展存储空间相关联的创建大小、扩展大小和大小的最大值。更改大小以控制 SinoDB® 如何使用特定存储空间的存储池条目。

```
EXECUTE FUNCTION
{ admin  | task }
(
"modify space sp_sizes" ,"space_name"
,"new_create_size" ,"new_extend_size"
[ , "max_size" ]
);
```

元素	描述	关键注意事项
space_name	存储空间的名称。	
max_size	存储空间大小的最大值，以 KB 为单位。	大小的缺省最大值是 0，其表示不受限。
new_create_size	当服务器使用存储池自动地扩展这个空间时，其可创建的新块的大小的最小值。您可以 KB 数或以总空间的百分比来定义该大小。	缺省创建大小设置为该空间总大小的 10%。 您指定的大小影响服务器自动地创建的块。其不影响您为关联空间手动地创建的任何块。
new_extend_size	服务器可用来在未镜像数据库空间或临时数据库空间中自动地扩展块的最小的大小值。该大小可以KB数或总空间的百分比来指定。	缺省扩展大小是 10 MB。 您所指定的大小影响服务器自动地扩展的块。其不影响您为关联的空间进行的手动地块扩展。

用法

如果创建或扩展大小值是 100 或较低的值，则 SinoDB® 将该值解释为百分比（例如：10 = 10%，2.84 = 2.84%）。如果该值是 1000 或更高的值，则服务器将该值解释为特定的 KB 数。值 100 - 1000 是无效的。

如果您将创建大小和扩展大小设置为 0，则即使当该空间变满 SinoDB® 也不自动地扩展该空间。此外，如果您将扩展大小设置为 0，那么您还需从那个空间中的所有块移除“Extentable”标志。这是一种使用一个操作将空间中的所有块标记为不可扩展的简单方法。

创建大小和扩展大小是大小的最小值。空间扩展的实际大小可能比较大，其取决于服务器所使用的存储池条目的块大小或服务器在那个特定时间所需的空间量。

例如：假设您创建存储池条目以在必要时扩展存储空间。然后假设名为 logdbs 的数据库空间用尽了可用页并需要额外 500 MB 做新日志。如果 logdbs 中没有块可扩展，则 SinoDB® 添加一个块，其大小为 logdbs 数据库空间的创建大小所指定的最小值。如果 logdbs 数据库空间的创建大小小于或等于 500 MB，则服务器会尝试找一个最小 500 MB 的空间。如果 logdbs 的创建大小是 1 GB，则服务器忽略所需的大小并添加一个 1 GB 块。

如果服务器无法找到所请求的最小空间量，则服务器返回空间不足错误且日志创建失败。

如果您将存储空间大小的最大值设置为非 0 值，则无论新扩展大小如何，存储空间大小都不可超过该最大值。当空间大小达到最大值之前剩余可扩展空间量小于新扩展大小时，将截断该扩展大小并将空间大小扩展到最大值。当空间大小达到最大值时，触发事件报警 86001。当剩余可扩展空间量小于存储池的最小块大小时，则不扩展该空间并返回错误。

示例

以下命令设置名为 dbspace3 数据库空间的最小创建大小为 60 MB，最小扩展大小为 10 MB，和大小的最大值为 100 MB：

```
EXECUTE FUNCTION task("modify space sp_sizes", "dbspace3", "60000",
     "10000", "100000");
```

以下命令设置名为 dbspace8 的数据库空间的最小创建大小为 20% 和最小扩展大小为 1.5%。

```
EXECUTE FUNCTION task("modify space sp_sizes", "dbspace8", "20", "1.5");
```

相关链接

modify chunk extendable 参数: 标记块为可扩展的 (SQL 管理 API) 在第646页
modify chunk extend 参数: 扩展块的大小 (SQL 管理 API) 在第645页
modify space expand 参数: 扩大空间的大小 (SQL 管理 API) 在第648页

onbar 参数：备份存储空间 (SQL 管理 API)

admin() 或 task() 函数使用 onbar 参数来备份存储空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("onbar backup
{  [ whole system ] [ levellevel ]  | [ logs ] }
"
)
;
```

元素	用途	关键注意事项
onbar backup	执行完全的存储空间备份。	如果未指定级别，则执行 0 级备份。
whole system	执行全系统备份。	这个等同于从命令行执行带有 -w 选项的 onbar 命令。如果未指定级别，则执行 0 级备份。
level *level*	指定在存储空间上执行的备份级别： • 0 代表完全备份。这是缺省值。	如果您请求增量备份但特定存储空间还未执行级别备份，则这个函数将以前一级别来备份那个存储空间。 例如：如果您请求 1 级备份，但该函数发现没有 0 级备份，则改为进行 0 级备份。

元素	用途	关键注意事项
	• 1 代表自从最后 0 级备份以来的更改。 • 2 代表自从最后 1 级备份以来的更改。	这个等同于从命令行执行带有 -L *level* 选项的 onbar 命令。
logs	执行逻辑日志文件的备份。	这个等同于从命令行执行带有 -l 选项的 onbar 命令。

用法

这个函数等同于调用 onbar 命令的特定选项来创建存储空间和逻辑日志文件的备份。

示例

以下示例创建存储空间的 0 级备份：

```
EXECUTE FUNCTION task("onbar backup");
```

以下示例创建存储空间的 1 级备份：

```
EXECUTE FUNCTION task("onbar backup level 1");
```

以下示例创建逻辑日志文件的 1 级备份：

```
EXECUTE FUNCTION task("onbar backup logs");
```

以下示例创建存储空间的全系统 0 级备份：

```
EXECUTE FUNCTION task("onbar backup whole system");
```

以下示例创建存储空间的全系统 2 级备份：

```
EXECUTE FUNCTION task("onbar backup whole system level 2");
```

相关链接

《SinoDB 备份和还原指南》: 使用 ON-Bar 进行备份

《SinoDB 备份和还原指南》: onbar -b 语法: 备份

onmode 和 a 参数: 添加共享内存段 (SQL 管理 API)

admin() 霍 task() 函数使用 onmode 和 a 参数来添加共享内存段。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","a" ,  "memory_siz"e
);
```

元素	描述	关键注意事项
memory_size	新虚拟共享内存段的大小，以 KB 为单位。	*size* 的值必须是正整数，且不超过操作系统对共享内存段大小的限制。

用法

通常，数据库服务器按需要自动地添加段，所以您不需要添加段到共享内存的虚拟部分。 然而，随着段的添加，数据库服务器可能在获得所需的内存之前已达到操作系统对于最大段数的限制。这种情况通常发生在 SHMADD 配置参数设置太小，以至于在数据库服务器获得某些操作所需的内存之前就耗尽了可用段的数量。

您可使用这个函数来添加大小比 SHMADD 配置参数所指定的大小还要大的段。通过使用这个函数来添加段，您可遵守操作系统对段的限制，同时满足数据库服务器对于更多内存的需求。

这个函数等同于 onmode -a 命令。

示例

以下示例添加 500 KB 虚拟共享内存：

```
EXECUTE FUNCTION task("onmode","a","500");
```

相关链接

onmode -a: 添加共享内存段 在第331页

onmode 和 c 参数：强制检查点 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 c 参数来强制检查点。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","c" ,
[{ "block"  | "hard"  | "norm"  | "unblock" }]
);
```

用法

这个函数强制检查点，清空缓冲区到磁盘的。如果逻辑日志中的最近检查点记录阻止释放逻辑日志文件（状态 U-B-L），则您可使用 c 选项来强制检查点。

使用 block 参数来防止数据库服务器处理任何事务。使用这个选项在 SinoDB® 上执行外部备份。数据库服务器阻塞时，除了处于只读模式，否则用户不可访问它。直到数据库服务器阻塞解除之前，不能完成任何事务。

使用 hard 参数来强制阻塞检查点。这是缺省值。

使用 norm 参数来强制非阻塞检查点。

使用 unblock 参数来解除数据库服务器阻塞。当解除数据库服务器阻塞时，数据事务和正常的数据库服务器操作可以重新开始。请在 SinoDB® 上完成外部备份之后使用这个选项。

这个函数等同于 onmode -c 命令。

示例

以下示例启动阻塞的检查点：

```
EXECUTE FUNCTION task("onmode","c","hard");
```

相关链接

onmode -c: 强制检查点 在第332页

onmode 和 C 参数：控制 B 型树扫描程序 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 C 参数来控制 B 型树扫描程序清除已删除条目的索引。

语法

```
EXECUTE FUNCTION
{ admin | task }
(
"onmode" ,"C",
{ "start", { "1" "start_coun"t } | "stop", { "1" "stop_coun"t } | "threshold","thresh_siz"e
| "duration","dur_nu"m | "rangesize","range_siz"e | "alice","alice_mod"e |
"compression","compression_leve"l }
);
```

元素	描述	关键注意事项
alice_mode	系统的 alice 模式。	从 0（OFF）到 12 范围内的有效整数值。
compression_level	对于数据库服务器实例，两个部分使用地索引页合并的级别。如果在这些页上的数据总计达到设置的级别，则合并页。	级别的有效值是 low、med（中级）、high 和 default。系统缺省值为 med。
dur_num	热列表有效的秒数。	超过这个秒数之后，即使该表上有未处理的项目，也将由下一个可用 B 型树扫描程序重新构建热列表。正在处理请求的扫描程序不中断。
range_size	在启用索引范围清除之前的索引大小。	可用大小 -1 来禁用范围扫描。
start_count	要启动的 B 型树扫描程序线程数。	如果未指定 *start_count*，则再多启动一个线程。一次最多启动 32 个线程。但同时运行的扫描程序线程数没有限制。
stop_count	要停止的 B 型树扫描程序线程数。	如果未指定 *stop_count*，则停止单个线程。停止所有索引扫描程序防止所以索引清除。 如果指定的 *stop_count* 值大于正在运行的线程数，不会发出错误，但会停止所有扫描程序线程。
thresh_size	索引置于热列表上之前，索引必须遇到的已删除条目的最小数。	在閾值之上的所有索引都已清除且没有 B 型树扫描程序要做的其他工作之后，低于閾值的索引将被添加到热列表。

用法

B 型树扫描程序有追踪索引效率和索引在服务器上放置多少额外工作的统计信息。基于索引因已提交的删除索引条目而完成的额外工作量，B 型树扫描程序会生成一个导致服务器额外工作的索引的有序列表，称为热列表。首先清除导致最高额外工作量的索引，并以递减的顺序清除其余索引。DBA 可动态地分配清除线程以配置工作负载。

这个函数等同于 onmode -C 命令。

示例

以下命令启动 60 个 B 型树扫描程序线程：

```
EXECUTE FUNCTION admin("onmode","C","start","30");
EXECUTE FUNCTION admin("onmode","C","start","30");
```

以下命令停止所有这些线程：

```
EXECUTE FUNCTION admin("onmode","C","stop","30000");
```

当 *stop_count* 值大于运行中的线程数时，不发出错误。

相关链接

onmode -C: 控制 B 型树扫描程序 在第332页
BTSCANNER 配置参数 在第77页

onmode 和 d 参数：设置数据复制类型 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 d 参数来更改参与高可用性数据复制（HDR）的服务器的模式。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode",
"d",
{  "standard"  | "primary" , "ha_alia"s  | "secondary" , "ha_alia"s }
);
```

元素	描述	关键注意事项
ha_alias	主数据库服务器或辅助数据库服务器的高可用性别名。	*ha_alias* 必须对应于预期的辅助数据库服务器的 ONCONFIG 文件中的 HA_ALIAS 配置参数。

用法

使用这个函数来设置高可用性数据复制类型为 standard、primary 或 secondary。当数据库服务器是处于静默、联机或只读模式时，您可使用 standard 参数。

在共享内存重新初始化之后，将保留数据库服务器的类型（standard、primary 或 secondary）和数据复制对中另一数据库服务器的 *ha_alias* 参数。

standard 参数删除数据复制对中数据库服务器之间的连接（如果存在），并将当前数据库服务器的类型设置为 standard。这个选项不更改对中另一数据库服务器的模式或类型。

primary 和 *ha_alias* 参数将数据库服务器类型设置为 primary，并尝试连接 *ha_alias* 指定的数据库服务器。如果连接成功，则开启数据复制。主数据库服务器进入联机模式，辅助数据库服务器进入只读模式。如果连接不成功，则数据库服务器进入联机模式，但不开启数据复制。

secondary 和 *ha_alias* 参数将数据库服务器类型设置为 secondary，并尝试连接 *ha_alias* 指定的数据库服务器。如果连接成功，则开启数据复制。主数据库服务器进入联机模式，辅助数据库服务器进入只读模式。如果连接不成功，则数据库服务器进入只读模式，但不开启数据复制。

这个函数等同于 onmode -d 命令。

示例

以下示例将名为 ids_stores 的服务器设置为 HDR 主服务器：

```
EXECUTE FUNCTION task("onmode","d","primary","ids_stores");
```

相关链接

onmode -d: 设置数据复制类型 在第334页

onmode 和 D 参数：设置 PDQ 优先级 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 D 参数来临时重置数据库服务器可分配给任一决策支持查询的 PDQ 资源。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode"  , "D","max_priority"
);
```

元素	描述	关键注意事项
max_priority	实际分配给查询的用户请求的PDQ资源的百分比。	该值必须是从 0 到 100 之间的无符号整数。

用法

当数据库服务器处于联机时，使用这个函数来覆盖 MAX_PDQPRIORITY 配置参数设置的限制。这些新值仅影响数据库服务器的当前实例，不会记录在 onconfig 文件中。如果关闭并重新启动数据库服务器，则这些值会恢复到 onconfig 文件中的值。

这个函数等同于 onmode -D 命令。

示例

以下示例设置可分配给查询的 PDQ 资源百分比为 50%：

```
EXECUTE FUNCTION task("onmode","D","50");
```

相关链接

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onmode 和 e 参数：更改 SQL 语句高速缓存的使用 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 e 参数来临时更改 SQL 语句高速缓存的模式。

语法

```
EXECUTE FUNCTION
```

```
{ admin  | task }
(
"onmode","e",
{ "enable"  | "flush"  | "off"  | "on" }
);
```

用法

如果禁用了 SQL 语句高速缓存，则使用 enable 参数来启用它。单个用户会话仅在执行以下任一操作之后可使用语句高速缓存：

- 设置环境变量 STMT_CACHE 为 11。
- 执行 SQL 语句 SET STATEMENT CACHE ON。

使用 flush 参数来清空 SQL 语句高速缓存中未使用的语句，并保持 SQL 语句高速缓存启用。在高速缓存清空之后，onstat -g ssc ref_cnt 字段显示 0。

使用 off 参数来关闭 SQL 语句高速缓存，以便不高速缓存任何语句。

使用 on 参数来高速缓存所有语句，除了用户通过以下操作之一关闭的那些以外：

- 使用这个命令指定 OFF 模式。
- 设置环境变量 STMT_CACHE 为 0。
- 执行 SQL 语句 SET STATEMENT CACHE OFF。

这个函数不可更改 ONCONFIG 文件中 STMT_CACHE 配置参数设置，但最后的参数覆盖那个设置（或如果未设置 STMT_CACHE，则为缺省值）。您使用这个命令进行的任何语句高速缓存行为的变更都仅影响当前数据库服务器会话。当您重新启动数据库服务器时，它使用 ONCONFIG 文件中 STMT_CACHE 参数的设置。如果在 ONCONFIG 文件中未定义 STMT_CACHE 配置参数，则服务器不使用语句高速缓存。

这个函数等同于 onmode -e 命令。

示例

以下示例启用 SQL 语句高速缓存：

```
EXECUTE FUNCTION task("onmode","e","enable");
```

相关链接

onmode -e: 更改 SQL 语句高速缓存的使用 在第338页

onmode 和 F 参数：释放未使用的内存段 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 F 参数来释放未使用的内存段。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"F"
) ;
```

用法

当您执行这个函数时，内存管理器检查未使用内存的每个内存池。内存管理器立即释放找到的未使用的内存块。内存管理器在检查每个内存池之后，开始检查内存段并释放数据库服务器不再需要的内存段。

运行这个命令导致当下处于活动的用户的性能显著的下降。尽管执行时间很短（1 到 2 秒），但对单个用户数据库服务器的降级可达 100%。具有多个 CPU 虚拟处理器的系统经历的降级会成比例的减少。

要确认未使用内存已释放，请检查消息日志。如果内存管理器释放一个或多个段，其将显示消息指示有多少段和字节的内存已释放。

提示： 定期地从操作系统调度工具运行这个命令，并在数据库服务器执行了任何创建多个内存段的函数（包括大索引构建、排序或备份）之后运行这个命令。

这个函数等同于 onmode -F 命令。

示例

以下示例释放未使用的内存块：

```
EXECUTE FUNCTION task("onmode","F");
```

相关链接

onmode -F: 释放未使用的内存段 在第339页

onmode 和 j 参数：切换数据库服务器到管理模式（SQL 管理 API）

admin() 或 task() 函数使用 onmode 和 j 参数来更改数据库服务器为管理模式。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"j"
) ;
```

用法

当服务器更改为管理模式时，所有会话都会丢失到该数据库服务器的连接，除了以下用户的会话：

- 用户 informix
- DBSA 组中的用户
- ADMIN_MODE_USERS 设置中标识的用户

这个函数等同于 onmode -j 命令。

示例

以下示例更改服务器为管理模式：

```
EXECUTE FUNCTION task("onmode","j");
```

相关链接

onmode -k, -m, -s, -u, -j#更改数据库服务器模式 在第339页

onmode 和 l 参数：切换到下一个逻辑日志（SQL 管理 API）

admin() 或 task() 函数使用 onmode 和 l 参数来切换当前逻辑日志文件到下一个逻辑日志文件。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","l"
```

);

用法

这个函数等同于 onmode -l 命令。

有关切换到下一个逻辑日志文件的信息，请参阅《*SinoDB*® 管理员指南》中管理逻辑日志文件的章节。

示例

以下示例将逻辑日志移出 root 块：

```
SELECT task("onmode", "l") FROM sysmaster:syslogfil
   WHERE chunk = 1 AND sysmaster:bitval(flags,"0x02")>0;
```

相关链接

onmode -l: 切换逻辑日志文件 在第341页

onmode 和 m 参数：切换到多用户模式 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 m 参数来更改数据库服务器为多用户模式。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"m"
) ;
```

用法

使用这个函数使数据库服务器从静默模式或管理模式改为联机模式。

这个函数等同于 onmode -m 命令。

示例

以下示例更改服务器为多用户模式：

```
EXECUTE FUNCTION task("onmode","m");
```

相关链接

*onmode -k, -m, -s, -u, -j#*更改数据库服务器模式 在第339页

onmode 和 M 参数：临时更改决策支持内存 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 M 参数来临时更改并行查询的可用内存大小。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","M", "memory_siz"e
);
```

元素	描述	关键注意事项
memory_size	并行查询可用的内存的最大量的新大小限制，以 KB 为单位。	对于 32 位平台，最大值是 2 GB。 对于 64 位平台，最大值是 4 GB。

用法

当数据库服务器处于联机时，使用这个函数来覆盖 DS_TOTAL_MEMORY 配置参数设置的限制。新值仅影响数据库服务器当前的实例；这些值不记录在 ONCONFIG 文件中。如果您关闭并重新启动数据库服务器，则参数的值恢复为 ONCONFIG 文件中的值。

这个函数等同于 onmode -M 命令。

示例

以下示例设置并行查询的大小限制为 50 MB：

```
EXECUTE FUNCTION task("onmode","M","50000");
```

相关链接

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onmode 和 n 参数：解锁驻留内存 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 n 参数来结束共享内存驻留部分的强制驻留。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"n"
) ;
```

用法

在您可运行这个函数之前，ONCONFIG 文件中的 RESIDENT 配置参数必须设置为 1。

这个函数不影响 ONCONFIG 文件中强制驻留参数 RESIDENT 配置参数的值。

这个函数等同于 onmode -n 命令。

示例

以下示例解锁驻留内存：

```
EXECUTE FUNCTION task("onmode","n");
```

相关链接

onmode -n, -r: 更改共享内存驻留 在第342页

onmode 和 0 参数：标记禁用的数据库空间为 down (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 0 参数来标记禁用的数据库空间为 down，以便被禁用数据库空间阻塞的检查点可以继续并释放任何被阻塞的线程。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","0"
);
```

用法

这个函数覆盖 ONDBSPACEDOWN 配置参数的 WAIT 模式。仅在下列情况使用这个命令：

- ONDBSPACEDOWN 设置为 WAIT.
- 发生禁用 I/O 错误，导致数据库服务器阻塞所有更新线程。
- 您无法或不想纠正导致禁用 I/O 错误的问题。
- 您想要数据库服务器标记禁用的数据库空间为 down 并继续处理。

这个函数等同于 onmode -0 命令。

示例

以下示例标记禁用的数据库空间为 down：

```
EXECUTE FUNCTION task("onmode","0");
```

相关链接

onmode -O: 重设 ONDBSPACEDOWN WAIT 方式 在第342页

onmode 和 p 参数：添加或移除虚拟处理器 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 p 参数来动态地为当前数据库服务器会话添加或移除虚拟处理器。这个函数不更新 onconfig 文件。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" , "p",
{  "numbe"r, { "aio"  | "bts"  | "cpu"  | "jvp"  | "lio"  | "po"  | "shm"  | "soc"  | "tli"
"vpclas"s }  |  "-number" , { "bts"  | "cpu"  | "jvp" "vpclas"s } }
);
```

元素	描述	关键注意事项
number	要添加或移除的虚拟处理器数。	正数则添加虚拟处理器。可添加的虚拟处理器的最大数目取决于操作系统。 UNIX™：负数则移除虚拟处理器。要移除的虚拟处理器数不可超过指定类型的处理器的实际数目。

元素	描述	关键注意事项
vpclass	用户定义的虚拟处理器类的名称。	Windows™：由于您仅可创建一个用户定义虚拟处理器实例，因此，*number* 参数必须设置为 1。

用法

仅当数据库服务器处理联机模式时，您才可使用这个函数。

CPU VP 的数目不应超过您的系统上物理处理器的数目，但如果超过也不发出任何错误。数据库服务器使用 CPU VP 的数量来为并行数据库查询（PDQ）分配资源。如果您删除 CPU VP，则您的查询可能明显地运行变慢。在您更改了 CPU VP 的数目之后，从 onstat -g mgm 命令输出中的 Reinit 字段显示有多少查询正在等待其他查询完成。

有关 CPU VP 类的性能影响的更多信息，请参阅《*SinoDB*® 性能指南》。

有关每个虚拟处理器类的说明，请参阅《*SinoDB*® 管理员指南》。

这个函数等同于 onmode -p 命令。

示例

以下示例添加一个 CPU 虚拟处理器：

```
EXECUTE FUNCTION task("onmode","p","1","cpu");
```

以下示例移除一个 Java™ 虚拟处理器：

```
EXECUTE FUNCTION task("onmode","p","-1","jvp");
```

相关链接

onmode -p: 添加或删除虚拟处理器 在第343页

onmode 和 Q 参数：设置决策支持查询的最大数目 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 Q 参数来更改并发执行的决策支持查询的最大数量。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"Q","querie"s
);
```

元素	描述	关键注意事项
queries	并发执行的并行查询的最大数量。	该数字必须是从 1 到 8,388,608 的无符号整数。

用法

当数据库服务器处于联机时，使用这个函数覆盖 DS_MAX_QUERIES 配置参数设置的限制。新值仅影响数据库服务器的当前实例；这些值不记录在 ONCONFIG 文件中。如果您关闭并重新启动数据库服务器，则参数的值恢复为 ONCONFIG 文件中的值。

有关用于控制 PDQ 的参数的信息，请参阅《*SinoDB*® 性能指南》。

这个函数等同于 onmode -Q 命令。

示例

以下示例设置并发执行的并行查询的最大数目为 8：

```
EXECUTE FUNCTION task("onmode","Q","8");
```

相关链接

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onmode 和 r 参数：强制共享内存的驻留 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 r 参数来启动共享内存驻留部分的强制驻留。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"r"
) ;
```

用法

在您可运行这个函数之前，ONCONFIG 文件中的 RESIDENT 配置参数必须设置为 1。

这个函数不影响 ONCONFIG 文件中强制内存参数 RESIDENT 配置参数的值。

这个函数等同于 onmode -r 命令。

示例

以下示例启动共享内存的强制驻留：

```
EXECUTE FUNCTION task("onmode","r");
```

相关链接

onmode -n, -r: 更改共享内存驻留 在第342页

onmode 和 S 参数：设置决策支持扫描的最大数目 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 S 参数来更改当前会话并发执行的决策支持扫描的最大数目。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"S","scan"s
);
```

元素	描述	关键注意事项
scans	并发执行的并行扫描的最大数目。	该数字必须是从 10 到 1,048,576 的无符号整数。

用法

当数据库服务器处于联机时，使用这个函数覆盖 DS_MAX_SCANS 配置参数设置的限制。新值仅影响数据库服务器的当前实例；这些值不记录在 ONCONFIG 文件中。如果您关闭并重新启动数据库服务器，则参数的值恢复为 ONCONFIG 文件中的值。

有关用于控制 PDQ 的参数的信息，请参阅《*SinoDB*® 性能指南》。

这个函数等同于 onmode -S 命令。

示例

以下示例设置并发执行的并行扫描的最大数目为 2000：

```
EXECUTE FUNCTION task("onmode","S","2000");
```

相关链接

onmode -D, -M, -Q, -S: 更改决策支持参数 在第337页

onmode 和 W 参数：重设语句高速缓存属性 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 W 参数来更改语句是否以及何时插入 SQL 高速缓存中。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode" ,"W",
{  "STMT_CACHE_HITS","hit"s  | "STMT_CACHE_NOLIMIT","valu"e }
);
```

元素	描述	关键注意事项
hits	在语句完全地插入 SQL 语句高速缓存之前，命中（引用）该语句的次数。	可能值为： • 0 = 在高速缓存中插入所有符合条件的语句，及其内存结构。 • 1 或更大 = 排除即席查询进入高速缓存中。
value	语句是否插入 SQL 语句高速缓存中。	可能值为： • 0 = 数据库服务器不向高速缓存插入语句。 • 1 = 数据库服务器始终将语句插入高速缓存中。

用法

当数据库服务器处于联机时，使用这个函数重设 STMT_CACHE_HITS 或 STMT_CACHE_NOLIMIT 配置参数的值。新值仅影响数据库服务器的当前实例；这些值不记录在 ONCONFIG 文件中。如果您关闭并重新启动数据库服务器，则参数的值恢复为 ONCONFIG 文件中的值。

如果您设置 STMT_CACHE_HITS 的值等于 0，则数据库服务器将所有符合条件的语句及其内存结构插入高速缓存中。如果该值大于 0 且 SQL 语句执行的次数少于 STMT_CACHE_HITS 的值，则数据库服务器将仅键条目插入高速缓存中。当语句发生次数达到了指定的命中数，则数据库服务器将符合条件的语句插入高速缓存中。STMT_CACHE_HITS 的新值显示在 onstat -g ssc 输出的 #hits 字段中。

如果没有任何查询共享，则将 STMT_CACHE_NOLIMIT 设置为 0 以防止数据库服务器为语句高速缓存分配大量内存。

这个函数等同于 onmode -W 命令。

示例

以下示例防止即席查询进入 SQL 语句高速缓存：

```
EXECUTE FUNCTION task("onmode","W","STMT_CACHE_HITS","1");
```

相关链接

onmode -W: 更改 SQL 语句高速缓存的设置 在第346页

onmode 和 wf 参数：永久地更新配置参数 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 wf 参数来动态地更新 onconfig 文件中的配置参数的值。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","wf",  "configuration_parameter_name= new_valu"e );
```

元素	描述	关键注意事项
configuration_parameter_name	配置参数名	该配置参数必须是您可动态地更新的。 您可动态地更新的配置参数列表与 onmode -wf 命令的相同。
new_value	配置参数的新值或值。	该值对于配置参数必须是有效的。 新值的格式必须严格遵守那个配置参数的语法。

用法

使用这个函数永久地更新配置参数的值。新值立即生效，并在服务器重新启动之后保存在 ONCONFIG 文件中。

这个函数等同于 onmode -wf 命令。

示例

以下示例设置 onconfig 文件中的 DYNAMIC_LOGS 配置参数的值为 2：

```
EXECUTE FUNCTION task("onmode","wf","DYNAMIC_LOGS=2");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode 和 wm 参数：临时地更新配置参数 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 wm 参数来动态地更新内存中配置参数的值。

语法

```
EXECUTE FUNCTION
{ admin  | task }
```

(
"onmode","wm", "*configuration_parameter_name= new_valu*"e);

元素	描述	关键注意事项
configuration_parameter_name	配置参数名。	您指定的配置参数必须是您可动态地更新的。 您可动态地更新的配置参数列表与 onmode -wf 命令的相同。
new_value	配置参数的新值或值。	该值对于配置参数必须是有效的。 新值的格式必须严格遵守那个配置参数的语法。

用法

使用这个函数来临时地更新可动态更新的配置参数的值。新值立即生效。新值不写入 ONCONFIG 文件，并在服务器重新启动之后丢失。

这个函数等同于 onmode -wm 命令。

示例

以下示例为当前会话设置 DYNAMIC_LOGS 配置参数的值为 2：

```
EXECUTE FUNCTION task("onmode","wm","DYNAMIC_LOGS=2");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

onmode、wm 和 AUTO_LRU_TUNING 参数: 更改 LRU 调整状态 (SQL 管理 API)

Use the onmode, wm, and AUTO_LRU_TUNING arguments with the admin() 或 task() 函数使用 onmode、wm 和 AUTO_LRU_TUNING 参数来更改 LRU 调整状态而不更新 onconfig 文件。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","wm",
{ "AUTO_LRU_TUNING=0  | "AUTO_LRU_TUNING=1 }
");
```

用法

使用 AUTO_LRU_TUNING =1 参数来启用自动 LRU 调整。

使用 AUTO_LRU_TUNING=0 参数来禁用自动 LRU 调整。

这个函数等同于 onmode -wm AUTO_LRU_TUNING 命令。

示例

以下示例启用自动 LRU 调整：

```
EXECUTE FUNCTION task("onmode","wm","AUTO_LRU_TUNING=1");
```

相关链接

onmode -wm: 更改 LRU 调整状态 在第348页

onmode 和 Y 参数：更改会话的查询计划度量 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 Y 参数来更改单个会话的查询计划度量的输出。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","Y""session_i"d ,
{ 0  | { 2  | 1 } [ ,file_name ] }
);
```

元素	描述	关键注意事项
file_name	说明输出文件名。	如果未包括文件的绝对路径，则在缺省示例输出文件位置中创建示例输出文件。如果该文件已存在，则说明输出附加到该文件。如果从 SET EXPLAIN 语句已存在文件，则在动态说明关闭之前不使用那个文件。
session_id	标识特定会话。	无。
-Y	动态地更改 SET EXPLAIN 语句的值。	无。

用法

可以使用这个函数来模拟 SET EXPLAIN 语句。

最后一个参数确定是否记录查询度量，包括查询优化器的计划、返回行数的估计以及查询的相对成本。

使用 2 参数来使数据库服务器发送查询计划到说明输出文件。

使用 1 参数来使数据库服务器发送查询计划和统计到说明输出文件。这个设置等同于特定会话的 SET EXPLAIN ON 语句。

使用 0 参数以使查询度量不输出到当前会话的说明输出文件。这个设置等同于 SET EXPLAIN OFF 语句。

这个函数等同于 onmode -Y 命令。

示例

以下示例禁止 ID 为 32 的用户会话的查询度量输出：

```
EXECUTE FUNCTION task("onmode","Y","32","0");
```

相关链接

onmode -Y: 动态更改 SET EXPLAIN 在第349页

《SinoDB SQL 指南: 语法》: SET EXPLAIN 语句

《*SinoDB SQL* 指南: 语法》: 使用 *FILE TO* 选项
《*SinoDB SQL* 指南: 语法》: *UNIX* 上说明输出文件的缺省名称和位置
《*SinoDB SQL* 指南: 语法》: *Windows* 上输出文件的缺省名称和位置
《*SinoDB* 性能指南》: 显示由优化器选择的查询计划的报告
《*SinoDB* 性能指南》: 说明输出文件
《*SinoDB* 性能指南》: 查询统计信息部分提供性能调试信息
onmode -Y: 动态更改 *SET EXPLAIN* 在第349页

onmode 和 z 参数: 终止用户会话 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 z 参数来终止指定的用户会话。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","z", "session_i"d
);
```

元素	描述	关键注意事项
session_id	会话 ID。	该值必须是大于 0 的无符号整数，且必须适当前正在运行的会话的会话标识号。

用法

这个函数等同于 onmode -z 命令。

示例

以下示例终止 ID 为 14 的用户会话:

```
EXECUTE FUNCTION task("onmode","z","14");
```

相关链接

onmode -z: 杀死数据库服务器会话 在第350页

onmode 和 Z 参数: 终止分布式事务 (SQL 管理 API)

admin() 或 task() 函数使用 onmode 和 Z 参数来终止指定的分布式事务。仅当参与的数据库服务器之间已失去通信才使用这个函数。如果应用程序正在执行分布式事务，则终止分布式事务之一会使客户端/服务器数据库系统处于不一致状态。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"onmode","Z", "addres"s
);
```

元素	描述	关键注意事项
address	与分布式事务关联的共享内存地址。	这必须是正在进行且已超过 TXTIMEOUT 配置参数指定的时间量的分布式事务的地址。 *address* 必须符合特定于操作系统对共享内存寻址的规则。这个地址可从 onstat -x 输出找到。

用法

仅当分布式事务超过 TXTIMEOUT 配置参数指定的时间量，这个函数才会成功。

这个函数等同于 onmode -Z 命令。

示例

以下示例终止地址为 0xa509018 的分布式事务：

```
EXECUTE FUNCTION task("onmode","Z","0xa509018");
```

相关链接

onmode -Z: 杀死分布式事务 在第350页

onsmsync 参数：与存储管理器目录同步 (SQL 管理 API)

admin() 或 task() 函数使用 onsmsync 参数来将 sysutils 数据库和紧急启动文件与存储管理器目录同步。

语法

```
EXECUTE FUNCTION
{ admin | task }
( "onsmsync"
[ , " { [{ -g number_of_generations | -t date_time | -i interval }] [ -s ] [ -O ][{
  -f file_name | dbspace }] | -b } " ]
) ;
```

元素	用途	关键注意事项
no options	将 sysutils 数据库和紧急启动文件与存储管理器目录同步。	无。
-b	彼此相互重新生成紧急启动文件（ixbar.*servernum*）和 sysutils 数据库。	如果 ixbar 文件为空或不存在，则 onsmsync -b 重新创建 ixbar 文件并从 syutils 数据库中的表来充填它。 如果 ixbar 不为空且包含对象数据，则 onsmsync -b 更新 sysutils 数据库和 ixbar 文件，以便它们同步。 如果 ixbar 文件有条目且 sysutils 数据库已重新构建，但因为其不包含数据而为空，则 onsmsync -b 选项从 ixbar 文件重新创建 sysutils 的数据。 -b 元素不与其他 onsmsync 选项一起使用。此外，它不与存储管理器同步。
dbspace	指定要到期的一个或多个存储空间。	如果您输入一个以上的存储空间，请使用空白来分隔名称。

元素	用途	关键注意事项
-f *file_name*	指定包含要到期的存储空间列表的文件的路径名。	使用这个选项可避免输入一个存储空间长列表。该文件名是任何有效的 UNIX™ 或 Windows™ 文件名。
-g *number_of_generations*	保留每一 0 级备份的一定数目的版本。	保留备份的最近一版，使所有较早的版本到期。
-i *interval*	使早于某一时间期间的所有备份到期。	保留晚于这个间隔的备份。如果需要从那个间隔之后的其他备份恢复，则不使早于间隔的备份到期。*interval* 使用 ANSI 或 GLS 格式：YYYY-MM 或 DD HH:MM:SS
-s	跳过存储管理器已到期的备份。	使用这个选项来跳过与存储管理器中已到期的对象同步。如果提供了 -s 选项，根据其他参数确定对象到期。
-O	严格地实施到期策略	如果与 -t、-g 或 -i 选项一起使用，则使备份的所有级别到期，即使其中某些需要从到期日之后发生的备份来恢复。-O 选项不影响逻辑日志到期。请参阅《*SinoDB*® 备份和还原指南》中的“使所有备份到期”。
-t *date_time*	是特定日期和时间之前的所有备份到期。	保留晚于这个 *datetime* 的备份。如果需要从那个 *datetime* 之后的其他备份恢复，则不使早于 *datetime* 的备份到期。*datetime* 使用 ANSI 或 GLS_DATETIME 格式。

用法

这个函数调用 onsmsync 实用程序来将 sysutils 数据库和紧急启动文件与存储管理器目录同步。

示例

以下示例调用 onsmsync 实用程序，并指定保留的备份版本数为 1 且使所有较早的备份版本到期：

```
EXECUTE FUNCTION task("onsmsync", "-g 1");
```

相关链接

《*SinoDB* 备份和还原指南》：*onsmsync* 实用程序

onstat 参数：监视数据库服务器 (SQL 管理 API)

admin() 或 task() 函数使用 onstat 参数来监视数据库服务器。

语法

EXECUTE FUNCTION { admin | task } ("onstat","*option_name*") ;

元素	描述	关键注意事项
option_name	onstat 命令选项。	该选项必须包括连字符且符合 onstat 选项语法。有关 onstat 选项，请参阅 *onstat* 实用程序 在第385页。

用法

使用这些命令来显示与运行 onstat 实用程序命令相同的关于数据库服务器的信息。

示例

以下示例运行 onstat -g ses 命令：

```
EXECUTE FUNCTION task("onstat","-g ses");
```

相关链接

onstat 实用程序 在第385页

ontape archive 参数：备份数据库上的数据 (SQL 管理 API)

admin() 或 task() 函数使用 ontape archive 参数来创建您数据库数据的备份。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("ontape archive
[{ file  | directory  | tape }]
[ level { 0  | 1  | 2 } ]
"
,"locatio"n
[ ,"block_size" ]
);
```

元素	描述	关键注意事项
location	到文件或目录或磁带设备的路径。	
block_size	在存储空间备份期间，ontape 写到的设备的块大小，以 KB 为单位。	缺省块大小为 512 KB。

用法

这个函数调用 ontape 实用程序来创建备份。

对于备份位置，您有三种设备可选：

file
现有文件。这是缺省值。

directory 或 dir
location 指定的现有目录路径。

tape
现有的磁带设备。

示例

这个函数在目录路径 /local/informix/backup/ 中创建 0 级归档：

```
EXECUTE FUNCTION task("ontape archive","/local/informix/backup/");
```

这个函数在目录路径 /local/informix/backup/ 中创建块大小为 256 KB 的 0 级归档：

```
EXECUTE FUNCTION task("ontape archive directory level 0",
"/local/informix/backup/","256");
```

相关链接

《*SinoDB* 备份和还原指南》：使用 *ontape* 进行备份

print error 参数：打印错误消息 (SQL 管理 API)

admin() 或 task() 函数使用 print error 参数来打印与指定错误号关联的消息。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"print error" ,"error_numbe"r
) ;
```

元素	描述	关键注意事项
error_number	错误号，无减号。	*error_number* 必须是现有的错误号。

用法

这个函数等同于 finderr 实用程序。

示例

以下示例打印错误号 -105 的消息文本：

```
EXECUTE FUNCTION task("print error","105");
(expression)  ISAM error:  bad isam file format.
```

print file info 参数：显示目录或文件信息 (SQL 管理 API)

admin() 或 task() 函数使用 print file info 参数来显示关于目录或文件的信息。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("print file info", " file_path"
);
```

元素	用途	关键注意事项
file_path	目录或文件的路径	

示例：文件信息

以下示例显示您要用来打印关于 /tmp 目录中 x 文件的信息的参数：

```
execute function task("print file info","/tmp/x");
```

返回下列信息：

```
(expression)  File name          = /tmp/x
              Is File            = 1
              Is Directory       = 0
              Is Raw Device      = 0
              Is Block Device    = 0
              Is Pipe            = 0
              File Size          = 554
              Last Access Time   = 11/29/2010 21:55:02
              Last Modified Time = 11/29/2010 21:51:45
              Status Change Time = 11/29/2010 21:51:45
              User Id            = 200
              Group id           = 102
              File Flags         = 33206
```

示例：目录信息

以下示例显示您要用来打印关于 /tmp 目录的信息的参数：

```
execute function task("print file info","/tmp");
```

返回下列信息：

```
(expression)  File name          = /tmp
              Is File            = 0
              Is Directory       = 1
              Is Raw Device      = 0
              Is Block Device    = 0
              Is Pipe            = 0
              File Size          = 32768
              Last Access Time   = 12/06/2010 11:53:00
              Last Modified Time = 12/06/2010 12:05:53
              Status Change Time = 12/06/2010 12:05:53
              User Id            = 0
              Group id           = 0
              File Flags         = 17407
```

print partition 参数: 打印分区信息 (SQL 管理 API)

admin() 或 task() 函数使用 print partition 参数来打印指定分区的头部。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "print partition"  | "print partition full" } ,"partition_numbe"r ) ;
```

元素	描述	关键注意事项
partition_number	分区号。	请在 systables 系统目录表的 partnum 列中查找分区号。

用法

使用这个函数来打印指定分区的表空间报告。

使用 full 参数运行这个函数，以包括按数据库空间的页类型排列的特定索引信息和页分配信息。

使用 print partition 参数运行这个函数等同于 oncheck -pt 命令。

使用 print partition full 参数运行这个函数等同于 oncheck -pT 命令。

示例

以下示例打印分区号为 1048611 的分区的头部：

```
EXECUTE FUNCTION task("print partition","1048611");
```

相关链接

oncheck -pt 和 -pT: 显示表或分段的表空间 在第308页

oncheck -pt 和 -pT: 显示表或分段的表空间 在第308页

rename space 参数: 重命名存储空间 (SQL 管理 API)

admin() 或 task() 函数使用 rename space 参数来重命名数据库空间、Blob 空间、智能大对象空间或外部空间。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"rename space", "space_nam"e,"new_nam"e
);
```

元素	描述	关键注意事项
new_name	空间的新名称。	
space_name	您想要重命名的数据库空间、Blob 空间、智能大对象空间或外部空间的名称。	

用法

这个函数等同于 onspaces -ren 命令。

示例

以下示例将名为 dbsp1 的数据库空间重命名为 dbsp2：

```
EXECUTE FUNCTION task("rename space","dbsp1","dbsp2");
```

相关链接

onspaces -ren: 重命名数据库空间、*BLOB* 空间、智能大对象空间或外部空间 在第382页

reset config 参数: 恢复配置参数值 (SQL 管理 API)

admin() 或 task() 函数使用 reset config 参数来将可动态更新的配置参数的值恢复为其 onconfig 文件中的值。可动态更新的配置参数为您可用 onmode 或 SQL 管理 API 命令来为会话变更的那些参数。

语法

```
EXECUTE FUNCTION { admin  | task }
(
"reset config"
,"configuration_parameter_nam"e
);
```

表 195: reset config 命令元素

元素	描述	关键注意事项
configuration_parameter_name	您想要恢复其值的配置参数的名称。	

用法

reset config 参数将配置参数的值恢复到 onconfig 文件中最后保存的值，即使在数据库服务器启动之后更改了该值。

示例

以下命令将 DYNAMIC_LOGS 配置参数的值恢复为 onconfig 文件中的值。

```
EXECUTE FUNCTION task("reset config","DYNAMIC_LOGS");
```

相关链接

更改 *onconfig* 文件 在第42页

reset config all 参数: 恢复所有动态地可更新的配置参数值 *(SQL* 管理 *API)* 在第674页

reset config all 参数: 恢复所有动态地可更新的配置参数值 (SQL 管理 API)

admin() 或 task() 函数使用 reset config all 参数来将所有可动态更新的配置参数的值恢复为其在 onconfig 文件中的值。可动态更新的配置参数为您可用 onmode 或 SQL 管理 API 命令来为会话变更的那些参数。

语法

```
EXECUTE FUNCTION { admin  | task }
(
"reset config all"
);
```

用法

reset config all 参数将所有可动态更新的配置参数的值恢复到 onconfig 文件中最后保存的值，即使在数据库服务器启动之后更改了这些值。

示例

以下命令恢复所有可动态地更新的配置参数的值。

```
EXECUTE FUNCTION task("reset config all");
```

相关链接

更改 *onconfig* 文件 在第42页
reset config 参数: 恢复配置参数值 *(SQL* 管理 *API)* 在第674页

reset sysadmin 参数: 移动 sysadmin 数据库 (SQL 管理 API)

admin() 或 task() 函数使用 reset sysadmin 参数来将 sysadmin 数据库搬移到指定的数据库空间。搬移 sysadmin 数据库将使该数据库重置回到首次创建它时的原始状态；所有数据、command history 和结果表都会丢失。仅内建任务、传感器和阈值保留在 sysadmin 的表中。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"reset sysadmin" [ ,"dbspac"e ]
);
```

元素	描述	关键注意事项
dbspace	The name of the dbspace.	

用法

这个函数没有等同的实用程序命令。

如果未指定 *dbspace* 作为最后的参数，则这个命令删除 sysadmin 数据库，然后在 rootdbs 中重新创建它。删除所有 ph_* 表和 command_history 行，并删除所有结果表。

示例

以下示例删除现有 sysadmin 数据库，并在名为 dbsp1 数据库空间中创建新的 sysadmin 数据库：

```
EXECUTE FUNCTION task("reset sysadmin","dbs1");
```

下一个示例删除 sysadmin 数据库，然后在 rootdbs 中重新创建它。

```
EXECUTE FUNCTION admin("reset sysadmin");
```

除了内建任务、传感器和阈值，这个函数调用从 ph_ 表中删除所有数据行和从 sysadmin 中删除所有结果表。在该函数完成执行之后，command_history 表为空。

相关链接

sysadmin 数据库 在第243页

restart listen 参数: 动态地停止并启动监听线程 (SQL 管理 API)

admin() 或 task() 函数使用 restart listen 参数来停止然后启动现有的 SOCTCP 或 TLITCP 网络协议的监听线程，而不中断现有连接。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"restart listen" ,"server_nam"e
```

) ;

元素	描述	关键注意事项
server_name	您想要停止并重新启动监听线程的数据库服务器的名称。	

用法

在 sqlhosts 文件中必须存在监听线程的定义。

如有需要，请在重新启动监听线程之前修订 sqlhosts 条目。例如：如果正在运行的监听线程绑定到端口 7777，则您可更改 sqlhosts 文件中的该端口，然后重新启动该线程。

这个函数等同于 onmode -P restart *server_name* 命令。

这个函数不更新 sqlhosts 文件。

示例

以下命令停止并再启动名为 ids_serv1 服务器的监听线程：

```
EXECUTE FUNCTION task("restart listen","ids_serv1");
```

相关链接

onmode -P: 动态地启动、停止或重启监听线程 在第345页

start listen 参数: 动态地启动监听线程 (SQL 管理 API) 在第690页

stop listen 参数: 动态地停止监听线程 (SQL 管理 API) 在第692页

revoke admin 参数: 撤销运行 SQL 管理 API 命令的权限

admin() 或 task() 函数使用 revoke admin 参数来撤销运行 SQL 管理 API 命令的权限。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"revoke admin"
,"user_name"
[ ,"privilege_group " ]
);
```

元素	描述	关键注意事项
user_name	要撤销其权限的用户名。	
privilege_group	权限组名。	关于权限组的列表，请参阅 *SQL 管理 API 门户: 按权限组划分参数* 在第573页 。

用法

仅用户 informix 或对于 SQL 管理 API 命令有 ADMIN 或 GRANT 权限的用户，可以使用 revoke admin 参数。

使用 revoke admin 参数来撤销个别用户运行 SQL 管理 API 命令的权限。您可撤销特定权限组的权限，或撤销所有权限。

示例

以下命令撤销用户 Bob 运行备份和复原 SQL 管理命令的权限：

```
EXECUTE FUNCTION task("revoke admin", "Bob", "BAR");
```

以下命令撤销用户 Bob 运行任何 SQL 管理命令的所有权限：

```
EXECUTE FUNCTION task("revoke admin", "Bob");
```

scheduler 参数: 停止或启动调度程序 (SQL 管理 API)

admin() 或 task() 函数使用 scheduler 参数来启动或停止调度程序。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "scheduler shutdown"  | "scheduler start" }
);
```

用法

使用 scheduler shutdown 参数来停止调度程序并解除其资源。

使用 scheduler start 参数来启动调度程序。

这个函数没有等同的实用程序命令。

您可用 onstat -g dbc 命令来查看调度程序线程的状态。

示例

以下示例在调度程序关闭之后，重新启动它：

```
EXECUTE FUNCTION task("scheduler start");
```

相关链接

onstat -g dbc 命令: 打印 dbScheduler 和 dbWorker 线程统计信息 在第448页

scheduler lmm enable 参数: 指定自动低内存管理设置 (SQL 管理 API)

admin() 或 task() 函数使用 scheduler lmm enable 参数来启动自动的低内存管理与更新低内存阈值设置。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("scheduler lmm enable",
[ LMM START THRESHOLD, "start_threshold_siz"e, ]
[ LMM STOP THRESHOLD, "stop_threshold_siz"e, ]
[ LMM IDLE TIME, "minimum_amount_of_tim"e, ]
);
```

元素	描述	关键注意事项
start_threshold_size	您想要数据库分为完全保持的空闲内存量。如果内存量低于 *start_threshold_size*，则服务器自动地释放内存并终止应用程序。	该值可表示为 SHMTOTAL 配置参数值的百分比表示或特定数量。如果该值小于 50，则认为是百分比。输入参数的结果值必须大于 5 MB 且小于 95 MB。 缺省值为 5 MB。 LMM START THRESHOLD 值和 LMM STOP THRESHOLD 值之间必须至少差 5 MB。
stop_threshold_size	在数据库服务器停止自动地释放内存与终止应用程序之前，您想要其拥有的空闲内存量。	该值可表示为 SHMTOTAL 配置参数值的百分比表示或特定数量。如果该值小于 50，则认为是百分比。输入参数的结果值必须大于 10 MB 且小于 100 MB。该值还必须比 LMM START THRESHOLD 多至少 5 MB。 缺省值为 10 MB。
minimum_amount_of_time	定义会话为空闲的时间量，以秒为单位。	该值必须在 1 与 86400 之间。 缺省值为 300 秒。

用法

您可用带有 scheduler lmm disable 参数的 admin() 或 task() 函数来在主数据库服务器或标准数据库服务器中停止当前和后续的低内存管理进程。当触发低内存管理时，数据库服务器按顺序执行下列这些任务：

1. 数据库服务器从空闲时间量最大的会话开始，一次终止一个会话，如有需要，将继续进行到空闲时间量最小的会话（其时间量超过 LMM IDLE TIME 设置中指定的时间量）。当达到 LMM STOP THRESHOLD 时，服务器停止终止会话。
2. 数据库服务器从使用最多内存的会话开始终止，如有需要，将继续进行的使用最少内存的会话，直到达到 LMM STOP THRESHOLD。
3. 数据库服务器通过将 VP_MEMORY_CACHE 配置参数设置为 0 与运行 onmode -F 命令释放未使用的共享内存段，来执行内存重新配置。

当低内存管理操作完成时，低内存管理器返回监视模式，并通过将 VP_MEMORY_CACHE 配置参数设置回其原始值来恢复数据库服务器的内存配置。

数据库服务器在 ph_threshold 表中存储自动低内存管理设置。

您可用 onstat -g lmm 命令查看低内存管理设置与最近活动。

注意： 如果您启用自动低内存管理并配置数据库服务器使用 SHMTOTAL 配置参数指定值的百分比作为启动与停止阈值，请在更改 SHMTOTAL 配置参数的值时使用告警。更改 SHMTOTAL 配置参数的值可能导致自动低内存管理的配置变成无效，强制 SinoDB® 使用缺省设置。

设置低内存管理阈值的示例

以下示例指定当数据库服务器的空闲内存为 10 MB 或更少时，服务器将启动自动低内存管理以停止应用程序与释放内存。该示例还指定如果会话有 300 秒未运行，则将其视为空闲，并且该示例指定当服务器有 20 MB 或更多的空闲内存时，其将停止自动低内存管理。

```
EXECUTE FUNCTION task("scheduler lmm enable",
 "LMM START THRESHOLD", "10MB",
 "LMM STOP THRESHOLD", "20MB",
```

```
"LMM IDLE TIME", "300");
```

SHMTOTAL 配置参数影响低内存管理阈值设置的示例

假设您设置 SHMTOTAL 配置参数为 1000000（1000 MB 或 1 GB），LMM START THRESHOLD 为 2 且 LMM STOP THRESHOLD 为 3。由于小于 50 的任何值都是 SHMTOTAL 的百分比，因此实际的 LMM START THRESHOLD 是 20000（20 MB）且实际的 LMM STOP THRESHOLD 是 30000（30 MB）。

当剩余的空闲内存为 20 MB 或更少时，数据库服务器开始管理低内存，并当空闲内存量为 30 MB 或更多时，停止管理内存。

假定，您知道您现在不需要这么多内存，并想要让操作系统使用内存，因此，您决定更改 SHMTOTAL 配置参数的值。您将 SHMTOTL 的值设置为 250000（250 MB）。这将实际的 LMM START THRESHOLD 变更为 5000（5 MB）并将实际的 LMM STOP THRESHOLD 变更为 7500（7.5 MB）。LMM STOP THRESHOLD 现在是无效的，因为 LMM START THRESHOLD 和 LMM STOP THRESHOLD 值之间必须至少差 5 MB。LMM STOP THRESHOLD 值也必须至少是 10 MB。

您可能决定差 10 MB 是适合您系统的量。但是差 5 MB 时，数据库服务器可能会在低内存管理进程上花费太多时间，而可能导致性能问题。

相关链接

scheduler lmm disable 参数: 停止自动低内存管理 (SQL 管理 API) 在第679页
onstat -g lmm 命令: 显示低内存管理信息 在第475页
LOW_MEMORY_MGR 配置参数 在第127页
SHMTOTAL 配置参数 在第168页
VP_MEMORY_CACHE_KB 配置参数 在第195页
《*SinoDB 管理员指南*》: 配置内存严重过低时的服务器响应
scheduler lmm disable 参数: 停止自动低内存管理 (SQL 管理 API) 在第679页

scheduler lmm disable 参数: 停止自动低内存管理 (SQL 管理 API)

admin() 或 task() 函数使用 scheduler lmm disable 参数来停止当前和后续的自动低内存管理的调用。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("scheduler lmm disable",
);
```

用法

如果其用量自动低内存管理，则您可通过下列指定来禁用它：

```
EXECUTE FUNCTION task("scheduler lmm disable");
```

您可用带有 scheduler lmm enable 参数的 admin() 或 task() 函数来启动自动低内存管理与更新阈值设置。

您可用 onstat -g lmm 命令来查看关于自动低内存管理设置与最近活动的信息。

相关链接

scheduler lmm enable 参数: 指定自动低内存管理设置 (SQL 管理 API) 在第677页
scheduler lmm enable 参数: 指定自动低内存管理设置 (SQL 管理 API) 在第677页
onstat -g lmm 命令: 显示低内存管理信息 在第475页

LOW_MEMORY_MGR 配置参数 在第127页

set chunk 参数：更改块的状态 (SQL 管理 API)

admin() 或 task() 函数使用 set chunk 参数来将 Blob 空间、数据库空间或智能大对象空间的状态更改为联机或脱机。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "set chunk offline"  | "set chunk online" }
, "space_nam"e, "path_nam"e,"offse"t
);
```

元素	描述	关键注意事项
space_name	Blob 空间、数据库空间或智能大对象空间的名称。	
path_name	块的磁盘分区或无缓冲设备。	
offset	磁盘分区内或无缓冲设备内到达块的偏移量，以 KB 为单位。	请参阅 *admin() 和 task() 参数大小规范* 在第572页。

用法

块必须在镜像对中，或必须是非关键数据库空间内的非主块。

使用 set chunk offline 参数来将块的状态更改为脱机。

使用 set chunk online 参数来将块的状态更改为联机。

这个函数等同于 onspaces -s 命令。

示例

以下示例将块的状态更改为联机：

```
EXECUTE FUNCTION task("set chunk online","dbs1","/dev/raw_dev1","0");
     Database selected.

(expression)  Chunk status successfully changed.
              Chunk number 2 "/dev/raw_dev1" -- Online

1 row(s) retrieved.
```

相关链接

onspaces -s: 更改镜像块的状态 在第383页

set dataskip 参数：启动或停止跳过数据库空间 (SQL 管理 API)

admin() 或 task() 函数使用 set dataskip 参数来指定数据库服务器在事务处理期间是否跳过不可用的数据库空间。

语法

```
EXECUTE FUNCTION
```

```
{ admin | task }
(
{ "set dataskip on" | "set dataskip off" }
, "dbspac"e
);
```

元素	描述	关键注意事项
dbspace	要开始或要停止跳过的数据库空间名。	

用法

运行这个函数来更新 DATASKIP 配置参数的值，其指定数据库服务器在事务处理过程中是否跳过不可用的数据库空间（例如：由于介质故障）。

使用 set dataskip on 参数来开始在指定的数据库空间停止工作时跳过它。

使用 set dataskip off 参数来停止跳过指定的数据库空间。

这个函数等同于 onspaces -f 命令。

示例

如果名为 dbsp1 的数据库空间停止工作，则以下示例跳过它：

```
EXECUTE FUNCTION task("set dataskip on","dbsp1");
```

相关链接

onspaces -f: 指定 *DATASKIP* 参数 在第379页

DATASKIP 配置参数 在第87页

set index compression 参数: 更改索引页压缩 (SQL 管理 API)

admin() 或 task() 函数使用 set index compression 参数来更改合并两个部分使用的索引页的级别。

语法

```
EXECUTE FUNCTION
{ admin | task }
(
"set index compression" ,"partition_numbe"r, [{ "med" | "default" | "high" | "low" }] );
```

元素	描述	关键注意事项
partition_number	分区号。	在系统目录表 systables 的 partnum 列中找分区号。

用法

使用这个函数来调整索引页压缩。如果在这些页上的数据总计达到设置的级别，则合并这些页。如果您的索引快速地增长，则您可降低压缩级别以优化空间和事务处理。如果您的索引有一些删除和插入操作或如果执行了批量更新，则您可提高级别。

如果您预期索引会以频繁分裂而快速地增长，则请使用 low 参数。

如果索引的增长或更改是适度的，则请使用 med 或 default 参数。

如果索引 90% 或更多为只读，或者没有太多更改，则请使用 high 参数。

这个函数等同于 onmode -C 命令和 BTSCANNER 配置参数的 compression 选项。

示例

以下示例设置分区的索引压缩为 high：

```
EXECUTE FUNCTION task("set index compression","1048611","high");
```

相关链接

onmode -C: 控制 B 型树扫描程序 在第332页

set onconfig memory 参数：临时地更改配置参数 (SQL 管理 API)

admin() 或 task() 函数使用 set onconfig memory 参数来动态地更新内存中配置参数的值。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"set onconfig memory", "configuration_parameter_nam"e, "new_valu"e);
```

元素	描述	关键注意事项
configuration_parameter_name	配置参数名。	该配置参数必须是您可动态地更新的。 您可动态地更新的配置参数列表与 onmode -wf 命令的相同。
new_value	配置参数的新值或值。	配置参数的新值或值必须是有效的。 新值的格式必须严格地遵守该配置参数的语法。

用法

使用这个函数来临时地更新可动态更新的配置参数的值。新值立即生效。新值不写入 onconfig 文件，且当数据库服务器重新启动时就会丢失。

这个函数等同于 onmode -wm 命令。

示例

以下示例设置当前会话的 DYNAMIC_LOGS 配置参数的值为 2：

```
EXECUTE FUNCTION task("set onconfig memory","DYNAMIC_LOGS","2");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

set onconfig permanent 参数: 永久地更改配置参数 (SQL 管理 API)

admin() 或 task() 函数使用 set onconfig permanent 参数来动态地更新 onconfig 文件中配置参数的值。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"set onconfig permanent", "configuration_parameter_nam"e, "new_valu"e);
```

元素	描述	关键注意事项
configuration_parameter_name	配置参数名。	该配置参数必须是您可动态地更新的。 您可动态地更新的配置参数列表与 onmode -wf 命令的相同。
new_value	配置参数的新值或值。	配置参数的新值或值必须是有效的。 新值的格式必须严格地遵守该配置参数的语法。

用法

使用这个函数来永久地更新配置参数的值。新值立即生效，并在服务器重新启动之后保存在 onconfig 文件中。

这个函数等同于 onmode -wf 命令。

示例

以下示例在 onconfig 文件中设置 DYNAMIC_LOGS 配置参数的值为 2：

```
EXECUTE FUNCTION task("set onconfig permanent","DYNAMIC_LOGS","2");
```

相关链接

onmode -wf#-wm: 动态更改某些配置参数 在第347页

set sbspace accesstime 参数: 控制访问时间跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sbspace accesstime 参数来启动或停止对智能大对象空间中所有智能大对象的访问时间的跟踪。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "set sbspace accesstime off"  | "set sbspace accesstime on" }
,"sbspac"e
);
```

元素	描述	关键注意事项
sbspace	智能大对象空间的名称。	

用法

使用 set sbspace accesstime off 参数关闭访问时间跟踪。

使用 set sbspace accesstime on 参数打开对存储在智能大对象空间中所有智能大对象的访问时间跟踪。

这个函数等同于 onspaces -ch 命令。

示例

以下示例关闭名为 sbsp1 智能大对象空间的访问时间的跟踪：

```
EXECUTE FUNCTION task("set sbspace accesstime off","sbsp1");
```

相关链接

onspaces -ch: 更改智能大对象空间缺省规范 在第376页

create sbspace with accesstime 参数: 创建跟踪访问时间的智能大对象空间 *(SQL 管理 API)* 在第614页

set sbspace avg_lo_size 参数: 设置智能大对象的平均大小 (SQL 管理 API)

admin() 或 task() 函数使用 set sbspace avg_lo_size 参数来指定在指定的智能大对象空间中的智能大对象的预期平均大小，以便数据库服务器可以计算元数据区的大小。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"set sbspace avg_lo_size","sbspac"e ,  "average_siz"e
);
```

元素	描述	关键注意事项
sbspace	智能大对象空间的名称。	
average_size	存储在智能大对象空间中的智能大对象的平均大小，以 KB 为单位。	Windows™: 4 到 2**31 UNIX™: 2 到 2**31

用法

这个函数等同于 onspaces -ch 命令。

示例

以下示例设置名为 sbsp1 的智能大对象空间中的智能大对象的预期平均大小为 8 KB:

```
EXECUTE FUNCTION task("set sbspace avg_lo_size","sbsp1","8");
```

相关链接

onspaces -ch: 更改智能大对象空间缺省规范 在第376页

set sbspace logging 参数: 更改智能大对象空间的日志记录 (SQL 管理 API)

admin() 或 task() 函数使用 set sbspace logging 参数来指定数据库服务器是否日志记录对智能大对象空间用户数据区的更改。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
{ "set sbspace logging on"  | "set sbspace logging off" }
,"sbspac"e
);
```

元素	描述	关键注意事项
sbspace	智能大对象空间的名称。	

用法

使用 set sbspace logging on 参数，则记录对智能大对象空间用户数据区的更改。

使用 set sbspace logging off 参数，则不记录对智能大对象空间用户数据区的更改。

这个函数等同于 onspaces -ch 命令。

示例

以下示例为智能大对象空间 sbsp1 启动智能大对象空间日志记录：

```
EXECUTE FUNCTION task("set sbspace logging on","sbsp1");
```

相关链接

onspaces -ch: 更改智能大对象空间缺省规范 在第376页

set sql tracing 参数: 设置全局 SQL 跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sql tracing 参数来设置全局 SQL 跟踪。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "set sql tracing info"  | "set sql tracing off"  | "set sql tracing on"  , "number_trace"s [
,"trace_siz"e [ ,"leve"l [ ,"mod"e ] ] ]  | "set sql tracing resume"  | "set sql tracing suspend"
}
);
```

元素	描述	关键注意事项
level	跟踪级别。缺省值为 low。	可能的值为： • low • med • high
mode	跟踪所有用户还是选定的用户。	可能的值为： • global

元素	描述	关键注意事项
		• user
number_traces	要跟踪的 SQL 语句数。缺省值为 1000。	
trace_size	跟踪缓冲区的 KB 数。如果超过这个缓冲区大小，则数据库服务器丢弃保存的数据。缺省大小为 2 KB。	

用法

使用这个函数来重置 SQLTRACE 配置参数的值。

使用set sql tracing info 参数来显示全局 SQL 跟踪的状态。

使用 set sql tracing off 参数来关闭全局 SQL 跟踪。

使用 set sql tracing on 来打开全局 SQL 跟踪。您可选择地指定跟踪级别和模式，或更改跟踪缓冲区的大小。

- 使用 low 参数来捕获语句统计、语句文本和语句迭代器。
- 使用 med 参数来捕获包括在 low 级跟踪中的所有信息，加上表名称、数据库名称和存储过程堆栈。
- 使用 high 参数来捕获包括在 med 级跟踪中的所有信息，加上主机变量。
- 使用 global 参数来启用对所有用户的跟踪。
- 使用 user 参数来对那些通过 set sql tracing user 参数启用跟踪的用户启用跟踪。

当暂停 SQL 跟踪时，使用 set sql tracing resume 参数来重新启动它。

使用 set sql tracing suspend 参数来暂停 SQL 跟踪而不重新分配任何资源。

示例

以下示例在 4 KB 跟踪缓冲区内对 1500 个 SQL 语句启动 high 级的全局跟踪：

```
EXECUTE FUNCTION task("set sql tracing on","1500","4","high","global");
```

以下示例暂停 SQL 跟踪：

```
EXECUTE FUNCTION task("set sql tracing suspend");
```

相关链接

SQLTRACE 配置参数 在第177页
onstat -g spf: 打印已就绪语句的概要文件 在第528页

set sql tracing database 参数: 更改数据库跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sql tracing database 参数来启动或停止对数据库的跟踪，或表列跟踪的数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( {  "set sql tracing database add" [  ,"database_nam"e ]  | "set sql tracing database clear"
| "set sql tracing database list"  | "set sql tracing database remove", "database_nam"e }
);
```

元素	描述	关键注意事项
database_name	数据库名	指定一个数据库名。

用法

使用 set sql tracing database add 参数来指定跟踪一个或多个数据库，而不是所有数据库。缺省为所有数据库。在单个 admin() 或 task() 函数中指定最多 6 个参数。数据库名的最大数目可设置为 16。

使用 set sql tracing database clear 参数来从正被跟踪的数据库列表中清除所有数据库。跟踪返回到缺省的所有数据库。

使用 set sql tracing database list 参数来罗列正被跟踪的数据库。

使用 set sql tracing database remove 参数来从正被跟踪的数据库列表中移除单个数据库。

当您使用 set sql tracing database 参数时，仅可指定一个数据库名。虽然可有最多 16 个数据库名，然而您必须在分别的函数调用中指定每个数据库名。每次调用该函数时，该函数添加一个数据库到该列表，直到该列表包含 16 个数据库。

示例

以下示例为名为 db1、db2 和 db3 的三个数据库设置 SQL 跟踪：

```
EXECUTE FUNCTION task("set sql tracing database add","db1");
EXECUTE FUNCTION task("set sql tracing database add","db2");
EXECUTE FUNCTION task("set sql tracing database add","db3");
```

set sql tracing session 参数: 控制对会话的跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sql tracing session 参数来更改当前会话的 SQL 跟踪。

语法

```
EXECUTE FUNCTION
{ admin  | task }
("set sql tracing session", { "clear"  | "off"  | "on" } [{  ,"current_session_i"d  | ,"session_i"d
}]
);
```

元素	描述	关键注意事项
current_session_id	当前会话的 ID。 这是缺省会话 ID。	
session_id	这个命令应用的会话的 ID。	

用法

使用 clear 参数来清除任何全局跟踪覆盖。该会话将符合全局跟踪策略。

使用 off 参数来关闭对该会话的跟踪，即使全局跟踪策略设置为启用跟踪。

使用 on 参数来打开对该会话的跟踪，即使全局跟踪策略设置为禁用跟踪。

示例

以下示例停止对当前会话的跟踪：

```
EXECUTE FUNCTION task("set sql tracing session","off");
```

set sql tracing user 参数: 控制对用户的跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sql tracing user 参数来更改对用户的 SQL 跟踪。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( {  "set sql tracing user add", "user_nam"e  | "set sql tracing user clear"  | "set sql tracing user list"  | "set sql tracing user remove","user_nam"e }
);
```

元素	描述	关键注意事项
user_name	用户名。	

用法

使用 set sql tracing user add 参数来指定对特定用户的跟踪。

使用 set sql tracing user clear 参数来从跟踪列表中移除所有用户。

使用 set sql tracing user list 来罗列正被跟踪的用户。

使用 set sql tracing user remove 参数来从正被跟踪的用户列表中移除单个用户。

示例

以下示例停止对名为 fred 的用户的 SQL 语句跟踪：

```
EXECUTE FUNCTION task("set sql tracing user remove","fred");
```

set sql user tracing 参数: 设置对用户会话的全局 SQL 跟踪 (SQL 管理 API)

admin() 或 task() 函数使用 set sql user tracing 参数来对指定的用户会话设置全局 SQL 跟踪的模式。

语法

```
EXECUTE FUNCTION
{ admin  | task }
( { "set sql user tracing clear"  | "set sql user tracing off"  | "set sql user tracing on" }
,"session_i"d
) ;
```

元素	描述	关键注意事项
session_id	会话 ID。	

用法

使用 set sql user tracing clear 来对指定的用户会话清除用户跟踪标志，以便其遵守全局跟踪策略。

使用 set sql user tracing off 来禁用对用户会话的 SQL 跟踪，即使全局模式为 ON。

使用 set sql user tracing on 来对用户会话启用用户 SQL 跟踪。即使全局跟踪模式为 OFF，也跟踪这个用户会话的 SQL 语句。

示例

以下示例启动对 ID 18 会话的跟踪：

```
EXECUTE FUNCTION task("set sql user tracing on","18");
```

start json listener 参数：启动 MongoDB API 有线监听器

admin() 或 task() 函数使用 start json listener 参数来启动 MongoDB API 有线监听器。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"start json listener"," [ ,"property_fil"e [ ,"listener_argument"s ] ] "
);
```

元素	描述	关键注意事项
property_file	要使用的有线监听器配置文件的名称，而不是缺省的。	*property_file* 是可选的。缺省有线监听器配置文件在 $INFORMIXDIR/etc/jsonListener.properties 中。
listener_arguments	传递给有线监听器的命令行参数。	

用法

start json listener 参数启动 MongoDB API 有线监听器。

示例

此示例中，通过使用 mycustom.properties 文件而不是缺省 jsonListener.properties 文件来启动 MongoDB API 有线监听器：

```
EXECUTE FUNCTION task("start json listener", "mycustom.properties");
```

在这个示例中，通过使用 mycustom.properties 文件而不是缺省 jsonListener.properties 文件来启动 MongoDB API 有线监听器，端口指定为 27018，且日志记录级别设置为 debug：

```
EXECUTE FUNCTION task("start json listener", "mycustom.properties",
   "-port 27018 -loglevel debug");
```

相关链接

《*SinoDB JSON* 兼容性指南》：启动有线监听器

《*SinoDB JSON* 兼容性指南》：有线监听器命令行选项

《*SinoDB JSON* 兼容性指南》：启动有线监听器

《*SinoDB JSON* 兼容性指南》: 有线监听器命令行选项

start listen 参数: 动态地启动监听线程 (SQL 管理 API)

admin() 或 task() 函数使用 start listen 参数来为 SOCTCP 或 TLITCP 网络协议启动现有的监听线程，而不中断现有的连接。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"start listen" ,"server_nam"e
) ;
```

元素	描述	关键注意事项
server_name	您想要为其启动监听线程的数据库服务器的名称。	

用法

服务器的 sqlhosts 文件中必须存在监听线程的定义。如果 sqlhosts 中不存在监听线程的定义，则您必须在可动态地启动监听线程之前添加它。有关添加监听线程的信息，请参阅《*SinoDB*® 管理员指南》。

这个函数不更新 sqlhosts 文件。

这个函数等同于 onmode -P start *server_name* 命令。

示例

以下命令为名为 ids_serv2 的服务器启动一个新的监听线程：

```
EXECUTE FUNCTION task("start listen","ids_serv2");
```

相关链接

onmode -P: 动态地启动、停止或重启监听线程 在第345页

stop listen 参数: 动态地停止监听线程 *(SQL* 管理 *API)* 在第692页

restart listen 参数: 动态地停止并启动监听线程 *(SQL* 管理 *API)* 在第675页

start mirroring 参数: 启动存储空间镜像 (SQL 管理 API)

admin() 或 task() 函数使用 start mirroring 参数来为指定的数据库空间、Blob 空间或智能大对象空间启动镜像。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"start mirroring", "space_nam"e);
```

元素	描述	关键注意事项
space_name	Blob 空间、数据库空间或智能大对象空间的名称。	

用法

这个函数等同于 onspaces -m 命令。

示例

以下示例启动名为 dbsp1 的数据库空间的镜像：

```
EXECUTE FUNCTION task("start mirroring","dbsp1");
```

相关链接

onspaces -m: 启动镜像 在第380页

stop json listener 参数: 停止有线监听器

Use the stop json listener argument with the admin() or task() function to stop the wire listener.

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"stop json listener"," [ ,"property_file" [ ,"listener_argument"s ] ] "
);
```

元素	描述	关键注意事项
property_file	要使用的有线监听器配置文件的名称，而不是缺省的。	*property_file* 是可选的。缺省有线监听器配置文件在 $INFORMIXDIR/etc/jsonListener.properties 中。
listener_arguments	传递给有线监听器的命令行参数。	

用法

stop json listener 参数停止有线监听器。

示例

此下列示例中，通过使用 mycustom.properties 文件而不是缺省 jsonListener.properties 文件来停止有线监听器：

```
EXECUTE FUNCTION task("stop json listener", "mycustom.properties");
```

此下列示例中，通过使用 mycustom.properties 文件而不是缺省 jsonListener.properties 文件来停止有线监听器，并将命令行参数传递到有线监听器：

```
EXECUTE FUNCTION task("stop json listener", "mycustom.properties", "-port
 27018");
```

相关链接

《SinoDB JSON 兼容性指南》: 有线监听器命令行选项

《SinoDB JSON 兼容性指南》: 停止有线监听器

stop listen 参数: 动态地停止监听线程 (SQL 管理 API)

admin() 或 task() 函数使用 stop listen 参数来为 SOCTCP 或 TLITCP 网络协议停止现有的监听线程，而不中断现有的连接。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"stop listen" ,"server_nam"e
) ;
```

元素	描述	关键注意事项
server_name	您想要为其停止监听线程的数据库服务器的名称。	

用法

sqlhosts 文件中必须存在监听线程的定义。

这个函数不更新 sqlhosts 文件。

这个函数等同于 onmode -P stop *server_name* 命令。

示例

以下命令为名为 ids_serv3 的服务器停止监听线程：

```
EXECUTE FUNCTION task("stop listen","ids_serv3");
```

相关链接

onmode -P: 动态地启动、停止或重启监听线程 在第345页

start listen 参数: 动态地启动监听线程 (SQL 管理 API) 在第690页

restart listen 参数: 动态地停止并启动监听线程 (SQL 管理 API) 在第675页

stop mirroring 参数: 停止存储空间镜像 (SQL 管理 API)

admin() 或 task() 函数使用 stop mirroring 参数来停止指定的数据库空间、Blob 空间或智能大对象空间的镜像。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"stop mirroring", "space_nam"e);
```

元素	描述	关键注意事项
space_name	Blob 空间、数据库空间或智能大对象空间的名称。	

用法

这个函数等同于 onspaces -r 命令。

示例

以下示例停止名为 dbsp1 的数据库空间的镜像：

```
EXECUTE FUNCTION task("stop mirroring","dbsp1");
```

相关链接

onspaces -r: 停止镜像 在第382页

storagepool add 参数：添加存储池条目 (SQL 管理 API)

admin() 或 task() 函数使用 storagepool add 参数来添加条目到 存储池（SinoDB® 可用于自动地添加空间到现有的存储空间的裸设备、熟文件或目录的集合）。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"storagepool add", "path_nam"e
,"begin_offse"t,"total_siz"e ,"chunk_siz"e,"priorit"y
);
```

元素	描述	关键注意事项
path_name	当需要额外存储空间时，服务器可使用的文件、目录或设备的路径。	目录名无需添加结尾斜杠（"/"）。 如果运行 oninit 命令时变量位于您的环境中，则您可以使用路径中的环境变量。
begin_offset	SinoDB® 可以开始分配空间处的设备偏移量，以 KB 为单位。	如果您指定了到目录的路径，则您必须指定 0 作为偏移量。
total_size	在这个条目中 SinoDB® 可用的总计空间。服务器可从这个空间量中分配多个块。	请确保指定目录的总大小为 0。 如果您为目录指定了非零值，则 SQL 管理 API 命令返回错误。 如果您为文件或设备指定了 0，则服务器将从该条目分配一个可扩展的块。
chunk_size	从设备、文件或目录中可分配的块的大小的最小值。	您可创建的最小块为 1000 K。因此，您可指定的最小块为 1000 K。 请参阅 *admin() 和 task() 参数大小规范* 在第572页。
priority	当服务器通过存储池查找空间时目录、文件或设备的优先级。 1 = 高优先级 2 = 中优先级 3 = 低优先级	在服务器从较低优先级条目分配空间之前，会试图从高优先级条目分配空间。

用法

如有需要，服务器使用存储池中的条目来添加新块到存储空间。

当您添加条目到存储池时，您可能想要对那个条目的使用方式进行一些控制。例如：为了减少实例中的块数，您可能仅想从特定裸设备分配大的块空间，并且不想这些块是可扩展的。在这种情况下，请将那个存储池条目的块大小配置为大型的。

您可添加下列类型的条目到存储池：

- 定长裸设备
- 定长熟文件
- 可扩展的裸设备（以扩展块的大小）
- 可扩展的熟文件（以扩展块的大小）
- 目录

存储池条目为目录，则通常归类为可扩展的，因为其没有总计大小。如果在目录中自动地创建新块，则服务器标记这些块为可扩展的。 当您添加目录存储池条目时，则您可能想要小的块大小， 因为服务器可扩展在目录中创建的任何块，且较小的块大小可减少实例中浪费的空间量。

如果存储池条目是在高可用性数据复制（HDR）主服务器上，则该条目中的相同路径在 HDR 集群中的所有辅助服务器上必须是可用的。

大小和偏移量的缺省单位为 KB。然而，您可以下列示例中显示的任一方式指定信息：

- "100000"
- "100000 K"
- "100 MB"
- "100 GB"
- "100 TB"

示例：添加目录的存储池条目

以下命令添加名为 /region2/dbspaces 的目录，起始偏移量为 0，总计大小为 0，初始块大小为 20 MB，且为高优先级：

```
DATABASE sysadmin;
EXECUTE FUNCTION task("storagepool add", "/region2/dbspaces", "0", "0",
"20000", "1");
```

Example: 为定长裸设备添加存储池条目

下列命令将路径名为 /dev/raw/device1 且总计 500 MB 空间的定长裸设备添加到存储池。该命令指定起始偏移量为 50 MB，总计大小为 10 GB，分配到块的最小的大小为 100 MB，且为低优先级。

```
EXECUTE FUNCTION task("storagepool add", "/dev/rawdevice1", "50 MB",
 "10 GB", "100 MB", "3");
```

Example: 为定长熟文件添加存储池条目

以下命令添加定长熟文件 1 GB 空间到存储池。该命令指定起始偏移量为 0，总计大小为 1000000 KB，分配到块的最小的大小为 50000 KB，且为中优先级：

```
EXECUTE FUNCTION task("storagepool add", "/ifmx_filesystem/storage/cooked7",
 "0", "1000000", "50000", "2");
```

当添加这个条目时，服务器试图将 cooked7 文件的大小增加到 1 GB。如果由于文件系统已满，服务器无法增加大小，则服务器返回错误消息，且不添加该条目到存储池。

SinoDB® 起初使用熟文件的一部分，但随着空间变满，如有需要可使用更多设备。

Example: 为可扩展的熟文件添加存储池条目

以下命令添加路径名为 /ifmx/CHUNKFILES/cooked2 的熟文件。如果服务器使用这个条目，则服务器创建一个初始大小为 1 GB 的块，并且自动地标记该块为可扩展的。

```
EXECUTE FUNCTION task("storagepool add", "/ifmx/CHUNKFILES/cooked2",
 "0", "0", "1 GB", "2");
```

Example: 使用路径中环境变量添加存储池条目

以下示例包括路径中的环境变量。该变量在运行 oninit 命令时出现在服务器环境中。

```
EXECUTE FUNCTION task("storagepool add", "$DBSDIR/chunk1",
 "0", "100000", "20000", "2");
```

相关链接

storagepool modify 参数: 更改存储池条目 *(SQL* 管理 *API)* 在第696页
storagepool delete 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第695页
storagepool purge 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第697页
《*SinoDB* 管理员指南》: 自动空间管理
storagepool modify 参数: 更改存储池条目 *(SQL* 管理 *API)* 在第696页
storagepool delete 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第695页
storagepool purge 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第697页

storagepool delete 参数: 删除存储池条目 (SQL 管理 API)

admin() 或 task() 函数使用 storagepool delete 参数来从存储池删除条目。

```
EXECUTE FUNCTION
{ admin  | task }
(
"storagepool delete",
"entry_id"
);
```

元素	描述	关键注意事项
entry_id	存储池条目 ID。	sysadmin 数据库中的 storagepool 表包含一个显示存储池中每一条目 ID 的列。

用法

如果您不想在扩展存储空间时服务器继续使用该条目，则删除存储池条目。

要删除所有存储池条目、状态为 Full 的存储池条目或状态为 Error 的存储池条目，请使用 SQL 管理 API storagepool purge 命令。（sysadmin 数据库中的 storagepool 表包含一个显示存储池中每一条目状态的列。）

示例

以下命令删除条目 ID 为 13 的存储池条目：

```
EXECUTE FUNCTION task("storagepool delete", "13");
```

相关链接

storagepool add 参数: 添加存储池条目 (SQL 管理 API) 在第693页
storagepool modify 参数: 更改存储池条目 (SQL 管理 API) 在第696页
storagepool add 参数: 添加存储池条目 (SQL 管理 API) 在第693页
storagepool modify 参数: 更改存储池条目 (SQL 管理 API) 在第696页
storagepool purge 参数: 删除存储池条目 (SQL 管理 API) 在第697页
《*SinoDB 管理员指南*》: 自动空间管理
storagepool purge 参数: 删除存储池条目 (SQL 管理 API) 在第697页

storagepool modify 参数: 更改存储池条目 (SQL 管理 API)

admin() 或 task() 函数使用 storagepool modify 参数来更改在需要额外存储空间时 SinoDB® 可使用的目录、熟文件或裸设备的条目。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
"storagepool modify", "entry_id",
"new_total_siz"e,"new_chunk_siz"e ,"new_priorit"y
);
```

元素	描述	关键注意事项
entry_id	存储池条目 ID。	sysadmin 数据库中的 storagepool 表包含一个显示存储池中每一条目 ID 的列。
new_total_size	在这个条目中 SinoDB® 可用的新总空间量。服务器可从这个空间量分配多个块。	请确保指定目录的总大小为 0。如果您为目录指定非零值，则 SQL 管理 API 命令返回错误。 如果您为文件或设备指定 0，则服务器从该条目分配一个可扩展的块。
new_chunk_size	可从设备、文件或目录分配的块大小的最小值。	您可创建的最小块为 1000 K。因此，您可指定的块大小的最小值为 1000 K。 请参阅 *admin() 和 task() 参数大小规范* 在第572页。
new_priority	当服务器通过存储池查找空间时目录、文件或设备的优先级。 1 = 高优先级 2 = 中优先级 3 = 低优先级	在服务器从较低优先级条目分配空间之前，会试图从高优先级条目分配空间。

用法

有时您可能想要更改存储池条目。例如：当存储池用尽空间时，您可能想要增加存储池的总大小，或者您可能想要更改块大小或优先级。当更改条目时，即使您不想更改所有这些值，也要包括总大小、块大小和优先级。

不可更改存储池条目的路径或起始偏移量。如果您想要更改这些值中的任何一个，则您必须删除该存储池条目，并以新路径或起始偏移量添加条目。

如果存储池条目在高可用性数据复制（HDR）主服务器上，则该条目中的相同路径在 HDR 集群中的所有辅助服务器上必须是可用的。

存储池大小和偏移量的缺省单位为 KB。然而，您可以下列示例中显示的任一方式指定信息：

- "100000"
- "100000 K"
- "100 MB"
- "100 GB"
- "100 TB"

示例

以下命令将 ID 为 4 的存储池条目的总大小、块大小和优先级更改为 10 GB、10 MB和中优先级。

```
EXECUTE FUNCTION task("storagepool modify", "4", "10 GB", "10000", "2");
```

假定，您添加一个条目到存储池，且该条目的路径为 /dev/IDS/chunk2，偏移量为 0，总大小为 100 MB，最小的块大小为 100 MB，且优先级为 2。在 SinoDB® 从这个条目分配任何空间之前，您使用 onspaces 来手动地添加带有相同路径（/dev/IDS/chunk2）的 50 MB 的块，且偏移量为 50 MB。当服务器试图使用这个条目来自动地创建块时，服务器仅检测到重叠。此时，服务器将标记该条目为“Error”状态，并试图使用另一条目来创建块。

您可通过将存储池条目（例如：条目 2）的总大小更改为 50 MB，且通过将该条目的最小的块大小更改为 50 MB 来修正该问题，如下所示：

```
EXECUTE FUNCTION task("storagepool modify", "2", "50 MB", "50 MB", "2");
```

相关链接

storagepool add 参数: 添加存储池条目 (SQL 管理 API) 在第693页
storagepool add 参数: 添加存储池条目 (SQL 管理 API) 在第693页
storagepool delete 参数: 删除存储池条目 (SQL 管理 API) 在第695页
storagepool purge 参数: 删除存储池条目 (SQL 管理 API) 在第697页
《*SinoDB* 管理员指南》: 自动空间管理
storagepool delete 参数: 删除存储池条目 (SQL 管理 API) 在第695页
storagepool purge 参数: 删除存储池条目 (SQL 管理 API) 在第697页

storagepool purge 参数: 删除存储池条目 (SQL 管理 API)

admin() 或 task() 函数使用 storagepool purge 参数来删除所有存储池条目、状态为 Full 的存储池条目，或状态为 Error 的存储池条目。

```
EXECUTE FUNCTION
{ admin  | task }
( { "storagepool purge all"  | "storagepool purge full"  | "storagepool purge errors"  | ); }
```

用法

使用 storagepool purge all 参数来删除所有存储池中的条目。

使用 storagepool purge full 参数来删除状态为 Full 的所有存储池条目。

使用 storagepool purge errors 参数来删除状态为 Error 的所有存储池条目。

sysadmin 数据库中的 storagepool 表包含一个显示存储池中每一条目状态的列。

示例

以下命令删除状态为 Full 的所有存储池条目：

```
EXECUTE FUNCTION task("storagepool purge full");
```

相关链接

storagepool add 参数: 添加存储池条目 *(SQL* 管理 *API)* 在第693页

storagepool modify 参数: 更改存储池条目 *(SQL* 管理 *API)* 在第696页

storagepool delete 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第695页

storagepool delete 参数: 删除存储池条目 *(SQL* 管理 *API)* 在第695页

storagepool add 参数: 添加存储池条目 *(SQL* 管理 *API)* 在第693页

storagepool modify 参数: 更改存储池条目 *(SQL* 管理 *API)* 在第696页

《*SinoDB* 管理员指南》: 自动空间管理

表和分段压缩和解压缩操作 (SQL 管理 API)

您可使用 SQL 管理 API admin() 或 task() 函数和参数来压缩和解压缩表中或表分段中的数据。压缩操作仅适用于数据行的内容和那些出现在逻辑日志记录中的数据行的映像。

内建 SQL 管理 API admin() 或 task() 函数定义在每个 SinoDB® 实例的 sysadmin 数据库中。在缺省情况下，仅用户 informix 可以调用这些函数。如果将 sysadmin 数据库的 Connect 权限授予用户 root 或 DBSA 组成员，则当他们直接地或远程地连接到 sysadmin 数据库时，他们也可调用 SQL 管理 API admin() 或 task() 函数。

您可用于表和表分段的压缩和解压缩操作的 SQL 管理 API admin() 或 task() 命令参数为：

table ***compression parameters***

对指定表的所有分段执行各种压缩操作。有关更多信息，请参阅 *table* 或 *fragment* 参数: 压缩数据和优化存储 *(SQL* 管理 *API)* 在第699页。

fragment ***compression parameters***

对于属于特定表的单个分段或指定分段集合执行各种压缩操作。有关更多信息，请参阅 *table* 或 *fragment* 参数: 压缩数据和优化存储 *(SQL* 管理 *API)* 在第699页。

compression purge_dictionary

删除所有不活动的压缩字典或在您指定日期之前创建的所有不活动的压缩字典。有关更多信息，请参阅 *purge compression dictionary* 参数: 移除压缩字典 *(SQL* 管理 *API)* 在第704页。

表和分段压缩操作包括创建压缩字典、估计压缩率、压缩表和表分段中的数据、合并空闲空间（重新打包）、归还空闲空间到数据库空间（收缩）、解压缩数据，以及删除个别的表和分段压缩字典。

当您运行 SQL 管理 API 压缩和解压缩命令时，压缩和解压缩数据库空间中的行数据和简单大对象。您还可指定是否仅压缩或解压缩数据库空间中的行数据或简单大对象。

admin() 命令返回一个整数；task() 命令返回一个字串符。

有关您可压缩的数据类型、压缩率、压缩估计和压缩字典，以及使用压缩命令参数的步骤的信息，请参阅《*SinoDB*® 管理员指南》中的“压缩”。有关显示压缩信息的实用程序、sysmaster 表和视图的信息，请参阅 *syscompdicts_full* 在第210页。

您还可压缩、优化存储和估计对 B 型树索引 的压缩收益。请参阅 *index compress repack shrink* 参数: 优化 *B* 型树索引的存储 *(SQL* 管理 *API)* 在第640页 和 *index estimate_compression* 参数: 估计索引压缩 *(SQL* 管理 *API)* 在第642页。

table 或 fragment 参数: 压缩数据和优化存储 (SQL 管理 API)

使用带有 table 或 fragment 参数的 SQL 管理 API 函数来创建压缩字典、估计压缩率、压缩表和表分段中的数据、合并空闲空间（重新打包）、归还空闲空间到数据库空间（收缩）、解压缩数据，和删除压缩字典。

当您运行 SQL 管理 API 压缩和解压缩命令时，压缩和解压缩数据库空间中的行数据和简单大对象。您还可指定是否仅压缩或解压缩数据库空间中的行数据或简单大对象。

语法：表数据压缩命令参数

```
EXECUTE FUNCTION { admin  | task }
( " table  <命令参数>  "
, "table_nam"e [ , "database_nam"e [ , "owne"r ] ]
) ;
```

语法：分段数据压缩命令参数

```
EXECUTE FUNCTION { admin  | task }
( " fragment  <命令参数>  "
, "partition_numbe"r
) ;
```

表和分段命令参数

表和分段命令参数

```
{ create_dictionary  |  compress [ { rows  | blobs } ]  [repack]  [shrink]  [parallel]
| repack [shrink]  [parallel]  | shrink  | estimate_compression  | repack_offline  |
uncompress [ { rows  | blobs } ]  [parallel]  | uncompress_offline  | purge_dictionary  |
update_ipa [parallel]  }
```

命令参数

下表描述每个参数。

表 196: 压缩和解压缩操作的参数

参数	描述
blobs	指定您仅想要压缩或解压缩数据库空间中的简单大对象，而不是行数据。
compress	就地压缩所有现有的行，而不移动它们（不重新打包表）。 这个选项自动地压缩数据库空间中的行数据和简单大对象。要仅压缩数据库空间中的行数据或简单大对象，还要使用 rows 或 blobs 元素。 如果目标表或分段的压缩字典不存在，则压缩操作还创建该字典。
create_dictionary	构建压缩字典。压缩字典是频繁地发生的模式和在压缩行中取代它们的符号数的库。 在创建字典之后，如果任何新插入或更新的行是可压缩的，则会压缩它们。不压缩现有的行。
estimate_compression	估计新压缩率和当前的压缩率。如果该表未压缩，则当前压缩率为 0.0%。
parallel	并行运行压缩、重新打包、update_ipa 或解压缩操作。 为表或分段类表的每个分段启动一个线程，并且跨那些分段并行地运行该操作。
purge_dictionary	在您解压缩表或分段之后，删除不活动的压缩字典。
repack	通过将数据移动到分段或表的前部来合并空闲空间。 由于重新打包操作在分段处于联机时移动数据，因此，访问使用隔离级别低于 Repeatable Read 的分段的其他查询可能偶尔地找同一行两次或找

参数	描述
	不到行。要避免这个可能性，请对并发查询使用 Repeatable Read 隔离级别；或者，使用 repack_offline 参数，而不使用 repack 参数。
repack_offline	在表或分段上持有互斥锁时，通过将数据移动到表或分段的前部，合并空闲的空间。这个操作会阻止所有其他对数据的访问，直到该操作完成。
rows	指定您仅想要压缩或解压缩数据库空间中的行数据，而不是简单大对象。
shrink	将分段或表的尾端的空闲空间归还到数据库空间，以减少该分段或该表的总大小。
uncompress	停止对新的 INSERT 和 UPDATE 操作压缩、解压缩所有压缩的行，并停用压缩字典。这个操作还会为分段分配新页，并将不再适合其原始页的解压缩行移动到新页。 由于这个操作在分段处于联机时移动行，因此，访问使用隔离级别低于 Repeatable Read 的分段的其他查询可能偶尔地找到同一行两次或找不到行。要避免这个可能性，请对并发查询使用 Repeatable Read 隔离级别；或者，使用 uncompress_offline 参数，而不使用 uncompress 参数。 这个选项自动地解压缩数据库空间中的行数据和简单大对象。要仅解压缩数据库空间中的行数据或简单大对象，还要使用 rows 或 blobs 元素。
uncompress_offline	在分段上持有互斥锁时，停止对新的 INSERT 和 UPDATE 操作压缩、解压缩所有压缩的行，并停用压缩字典。这个操作还会为分段分配新页，并将不再适合其原始页的解压缩行移动到新页。这个操作会阻止所有其他对分段数据的访问，直到该操作完成。 这个操作还会为分段分配新页，并将不再适合其原始页的解压缩行移动到新页。
update_ipa	移除指定表或分段未完成的就地更改操作。

命令元素

下表显示您可在命令中使用的元素。

表 197: 表压缩和存储优化命令元素

元素	描述	关键注意事项
database_name	包含指定表的数据库名。	可选的。 如果您未指定 *database*，则 SinoDB® 使用当前数据库。 如果您输入数据库名，则必须使用与系统目录表中相同的大写或小写字母。
owner	包含指定表的数据库所有者的授权标识符。	可选的。 如果您未指定 *owner*，则 SinoDB® 使用当前所有者。 如果您输入所有者名，则必须使用与系统目录表中相同的大写或小写字母。

元素	描述	关键注意事项
table_name	包含数据可表名。	您必须使用与系统目录表中相同的大写或小写字母。

表 198: 分段压缩和存储优化命令元素

元素	描述	关键注意事项
partition_number	属于同一表的用空格分隔的分区号列表。	

用法

SinoDB® 使用压缩字典来压缩数据。

您在表或分段上运行 compress 命令之后，SinoDB® 自动地压缩您添加到该表或分段的任何新行。当您运行 compress 命令时，如果表或分段包含的行数多于 2000，则构建压缩字典并压缩所有的行。当您运行压缩命令时，如果表或分段包含的行数少于 2000，则启用该表或分段的自动压缩。在插入 2000 行之后，创建压缩字典并压缩最初 2000 行之后的所有行。要压缩最初的 2000 行，请再次运行 compress 命令。

如果您显著地更改数据，则压缩字典可能无效。在这种情况之下，请解压缩，然后再次压缩。

您可取消带有 compress 或 uncompress 参数的命令，例如：在 DB-Access 中键入 CTRL-C。在先前中断的命令之后，您可重新发出带有 repack、repack_offline、uncompress 和 uncompress_offline 参数的命令。

在表或分段上已发生 compress、repack、repack_offline、shrink、uncompress 或 uncompress_offline 操作之一时，您不可在该表或分段上执行这些操作。

当您在单个命令中指定多个操作时，服务器按以下顺序执行这些操作：

- create_dictionary
- compress
- repack
- shrink

compress、repack、repack_offline、uncompress 和 uncompress_offline 操作可消耗大量日志文件。如果您预期运行的任何工作负载（包括但不限于这些压缩操作）消耗的日志文件快于每 30 秒一个，则请将日志配置的更大些。

日志记录 compress、repack 和 uncompress 操作，但以小部分运行。

如果您在执行压缩操作之后更改表的分段策略，则该表丢失其压缩状态并且需要重新压缩。

在 repack_offline 或 uncompress_offline 操作完成之前删除或禁用索引，可减少服务器完成该操作所需的时间。随后，您可重新创建或启用索引，最好利用 PDQ。删除或禁用索引，然后再次创建或启用它们，与不这么做比起来，可更快地完成 repack_offline 或 uncompress_offline 操作。

如果数据库空间曾包含压缩表，则请不要删除正在使用 Change Data Capture（CDC）API 的数据库空间，因为这可能删除 CDC 仍需要的压缩字典。

重新打包

压缩操作通常在个别数据和剩余页上创建一些空闲空间，但该空间在表或分段的末尾不合并。相反，该空间可用于保存新插入的行，直到该空间填满之前，该表不会增大。

仅在联机时发生的压缩操作就地压缩表的行。重新打包操作移动这些行。您可在联机或脱机时执行重新打包操作。联机时的操作允许在表上发生并发的活动。然而，这可导致幻象行。（幻象行是后来回滚的事务，在事务期间一开始修改或插入的行。）

要避免幻象行，那么在您可使其他用户不访问表或分段时，您可能想要脱机重新打包。例如：您可能在日间执行带有并发活动的 compress 操作，然后在夜间（此时表上没有预期的并发活动）执行 repack_offline 操作。

不可将脱机操作与联机操作一起执行。例如：您可执行 compress 与 repace 的组合操作，但您不可执行 compress 与 repack_offline 的组合操作。如果您想要执行脱机重新打包，则您必须分两步来做：

1. 执行 compress 操作。
2. 执行 repack_offline 操作。

同样地，您不可执行 repack_offline 与 shrink 的组合操作。

如果正在发生重新打包操作的同时表或分段中发生轻量级追加（无缓冲、无日志记录的插入操作），那么重新打包操作不会完成表或分段末尾的空间合并。重新打包操作不会完成是因为新扩展数据块添加到已发生过重新打包操作的位置上，因此空间无法归还到数据库空间。要完成重新打包过程，您必须在轻量级追加活动完成之后运行第二次重新打包操作。这个第二次重新打包操作是构建在第一次重新打包操作的结果之上。

收缩

通常在重新打包操作之后执行收缩操作。

您可稳妥地收缩整个表而不会损及表的分配策略。例如：如果您有一个分段表，一周的每一天一个分段，并为未来使用预先分配了许多分段，则您可收缩该表而不会损及这个分配策略。如果该表为空，则 SinoDB® 将该表收缩到该表创建时指定的初始的扩展数据块大小。

当您启动收缩操作时，SinoDB® 如下地缩小扩展数据块：

- 除了第一个扩展数据块以外，尽可能的缩小所有扩展数据块。
- 如果表是整个地在第一个扩展数据块中（例如，该表是空表），则 SinoDB® 不会将第一个扩展数据块缩小到比使用 CREATE TABLE 语句创建该表是指定的扩展数据块大小还小。

您可使用 ALTER TABLE 语句的 MODIFY EXTENT SIZE 子句来减小当前的扩展数据块大小。在执行了这个操作之后，您可再次运行收缩操作，将第一个扩展数据块收缩到新的扩展数据块大小。

解压缩

对于任何未压缩的表或分段，解压缩操作无效。

在您解压缩表或分段之后，您可执行 purge_dictionary 操作来删除那个表或分段的字典。

清除

在对表或分段执行 purge_dictionary 操作之前，您必须：

- 解压缩表和分段。

  当解压缩表或分段时，SinoDB® 将表或分段的字典标记为不活动。
- 确保 Enterprise Replication 函数不需要较早日志的压缩字典。
- 归档包含带有压缩字典的表或分段的任何数据库空间，即使已经解压缩了表或分段中的数据且字典不再是活动的。

您还可删除所有压缩字典或在指定日期之前以及那天所创建的所有压缩字典。有关信息，请参阅 *purge compression dictionary 参数: 移除压缩字典 (SQL 管理 API)* 在第704页。

示例

以下命令压缩、重新打包和收缩 insurance 数据库中名为 auto 的表中的行数据和该数据库空间中的简单大对象。该数据库的所有者为 tjones。

```
EXECUTE FUNCTION task("table compress repack shrink","auto",
"insurance","tjones");
```

以下命令仅并行地压缩名为 dental 表中的行数据。

```
EXECUTE FUNCTION task("table compress rows parallel","dental");
```

以下命令解压缩分区号为 14680071 的分段。

```
EXECUTE FUNCTION task("fragment uncompress","14680071");
```

以下命令仅并行地解压缩分区号为 14680071 的分段中的行数据。

```
EXECUTE FUNCTION task("fragment uncompress rows parallel","14680071");
```

以下命令估计压缩 insurance 数据库中名为 home 的表的收益。该数据库的所有者为 fgomez。

```
EXECUTE FUNCTION task("table estimate_compression","home",
"insurance","fgomez");
```

以下命令并行地移除名为 auto 的表上挂起的就地更改操作。

```
EXECUTE FUNCTION task("table update_ipa parallel","auto");
```

在您运行该命令之后，数据库服务器显示可达成的压缩率估计，以及当前已达成的压缩率（如果存在的话）。有关该命令的输出的信息，请参阅 估计压缩操作的输出 *(SQL 管理 API)* 在第703页。

相关链接

估计压缩操作的输出 *(SQL 管理 API)* 在第703页
onstat -g ath 命令: 显示所有线程的信息 在第426页
《*SinoDB* 管理员指南》: 压缩

估计压缩操作的输出 (SQL 管理 API)

在您运行估计压缩率的命令之后，数据库服务器显示可达成的压缩率估计和当前已达成压缩率（如果存在的话）的信息。

表 199: estimate_compression 命令显示的信息

列	显示的信息
est	这是使用新压缩字典可达成的压缩率估计。该估计是相较于不压缩所节省的空间百分比。
curr	这是当前达成的压缩率的估计。该估计是相较于不压缩所节省的空间百分比。对于未压缩的分段或表，总是出现 0.0%。
change	这是通过切换到新压缩字典，您可达成的压缩率中所赢得的百分点（或可能损失的百分点，虽然应该很罕见）的估计。这是 est 和 curr 之间的差异。 如果表或分段未压缩，则您可用压缩参数来创建压缩字典。如果分段已压缩，则在可压缩之前，您必须执行 uncompress 或 uncompress_offline 操作。
partnum	分段的分区号。
coloff	该值定义估计是对数据库空间中的行数据还是简单大对象，如下所示： -1 表示是对行数据的估计 正数值表示是对该值标识的偏移量处的分区简单大对象的估计。该偏移量是表中列偏移量，以字节为单位。

列	显示的信息
table	分段所属的表的全名，格式为 *database:owner.tablename* 如果您正在估计索引的压缩收益，则该列出现索引的全名。

示例

以下输出显示如果您重新压缩第一个分段，则节省的空间可增加 .4%。如果您压缩未压缩的第二个分段，则可发生 75.7% 的增长。 coloff 列中的值 -1 表示压缩行数据。

```
est   curr  change  partnum     coloff  table
----- ----- ------ ------------  ------ ----------------------------
75.7% 75.3%   +0.4  0x00200003     -1  insurance:bwilson.auto
75.7%  0.0%  +75.7  0x00300002     -1  insurance:pchang.home
```

以下输出显示对行数据的压缩估计（在第一行中），和对偏移量 4 和 60 的简单大对象的压缩估计（在第二和第三行中）：

```
est   curr  change  partnum     coloff  table
----- ----- ------ ------------  ------ --------------------
75.4% 71.5%  +3.9 0x00200002       -1 test:mah.table1
 5.0% 75.0%  +0.0 0x00200002        4 test:mah.table1
75.0% 75.0%  +0.0 0x00200002       60 test:mah.table1
```

对表和分段的压缩估计的输出看上去相同，除了表的输出总是显示该表中的所有分段，而分段的输出仅显示指定分段的信息。

相关链接

index estimate_compression 参数: 估计索引压缩 (SQL 管理 API) 在第642页
table 或 fragment 参数: 压缩数据和优化存储 (SQL 管理 API) 在第699页

purge compression dictionary 参数: 移除压缩字典 (SQL 管理 API)

调用带有 compression purge_dictionary 初始命令的 admin() 或 task() 函数来删除所有不活动的压缩字典或在指定日期之前为压缩表或分段创建的所有不活动的压缩字典。在删除为表和分段创建的任何压缩字典之前，您必须解压缩这些表和分段以使字典处于不活动状态。

语法：Compression Purge_Dictionary

```
EXECUTE FUNCTION { admin  | task } ("compression purge_dictionary"  ,"date" ) ;
```

用法

在对表和分段执行 purge_dictionary 操作之前，您必须：

- 解压缩表和分段。

  当您解压缩表或分段时，SinoDB® 标记该表或分段的字典为不活动的。
- 确保 Enterprise Replication 函数不需要该压缩字典。
- 归档包含带有压缩字典的表或分段的任何数据库空间，即使您已解压缩该表或分段中的数据且该字典不再是活动的。

compression purge_dictionary 命令删除所有压缩字典。

带有日期作为第二参数的 compression purge_dictionary 命令删除在指定日期之前和那一天创建的所有压缩字典。您可使用能够根据语言环境转换为 DATE 数据类型的任何格式的日期。例如：您可指定 03/29/2009、03/29/09 或 Mar 29, 2009。

您还可通过调用以 table 或 fragment 为初始命令，且以 purge_dictionary 为下一参数的 admin() 或 task() 函数来删除特定的压缩字典。

您不可删除为索引创建的压缩字典。当索引删除时，数据库服务器会移除这些压缩字典。

以下命令告诉 SinoDB® 移除在 2009 年 7 月 8 日之前和那一天创建的所有字典：

```
EXECUTE FUNCTION task("compression purge_dictionary", "07/08/2009");
```

以下命令告诉 SinoDB® 移除 insurance 数据库中名为 auto 的表的不活动的字典，该数据库的所有者为 tjones。

```
EXECUTE FUNCTION task("table purge_dictionary",
"auto", "insurance", "tjones");
```

tenant create 参数：创建租户数据库 (SQL 管理 API)

admin() 或 task() 函数使用 tenant create 参数来创建租户数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
'tenant create',  'database_name','
{dbspace: "  , name "
[{  ,blobspace:"  , name "  | ,case:" { INSENSITIVE  | SENSITIVE } "  | , catalogs: "
name "  | ,dbspacetemp:"  , name "  | ,logmode:" { UNBUFFERED  | ANSI  | BUFFERED  |
NONE } "  | ,locale:"definition "  | ,sbspace:"  , name "  | ,sbspacetemp:"  , name "  |
,session_limit_locks:" number"  | ,session_limit_logspace:" number [{ KB  | MB  | GB }] "  |
,session_limit_memory:" number [{ KB  | MB  | GB }] "  | ,session_limit_tempspace:" number [{
KB  | MB  | GB }] "  | ,session_limit_txn_time:" number"  | ,tenant_limit_space:" number [{
KB  | MB  | GB  | TB }] "  | , tenant_limit_memory: " number [{ KB  | MB  | GB  | TB }] "  | ,
tenant_limit_connections: " number "  | ,vpclass:"  name [ ,num= number ] " }]
}');
```

元素	描述	关键注意事项
blobspace	以逗号分隔的一个或多个分配给租户数据库的 Blob 空间的列表。	如果租户数据库包含简单大对象，则至少需要一个 Blob 空间。 Blob 空间必须为空才能分配给租户数据库。 Blob 空间在分配给租户数据库之前必须存在。 在租户数据库以外创建的简单大对象不可存储在租户数据库的 Blob 空间中。
case	数据库对大写和小写字母的敏感性： INSENSITIVE 不区分大 小写。 SENSITIVE 区分大小写。这是缺省值。	如果省略这个属性，则数据库区分大小写。

元素	描述	关键注意事项
catalogs	存储租户数据库目录的数据库空间。	该数据库空间必须罗列在 dbspace 属性中。 如果省略这个属性，则 dbspace 属性的第一个数据库空间包含租户数据库目录。
database_name	租户数据库名。	该数据库名必须在该数据库服务器上。 现有的非租户数据库不可变为租户数据库。
dbspace	分配给租户数据库的一个或多个数据库空间的列表。用逗号分隔数据库空间名。	数据库空间必须为空才能分配给租户数据库。 数据库空间分配给租户数据库之前必须存在。 在租户数据库以外创建的对象不可存储在租户数据库的数据库空间中。
dbspacetemp	分配给租户数据库的一个或多个临时数据库空间的列表。用逗号分隔临时数据库空间名。	您可通过将 DBSPACETEMP 环境变量设置为 dbspacetemp 属性指定的数据库空间的子集来覆盖会话的 dbspacetemp 属性。 如果省略 v属性，则临时表存储在 DBSPACETEMP 配置参数或环境变量指定的临时数据库空间中。
locale	数据库的语言环境。	locale 的值与 DB_LOCALE 环境变量的值相同。 缺省语言环境为 en_US.819。
logmode	日志模式定义： UNBUFFERED 无缓冲数据库日志记录。这是缺省值。 ANSI 符合 ANSI 的数据库日志记录。 BUFFERED 缓冲的数据库日志记录。 NONE 无数据库日志记录。	如果省略这个属性，则日志记录模式为无缓冲的。
sbspace	分配给租户数据库的一个或多个智能大对象空间的列表。用逗号分隔智能大对象空间名。	如果租户数据库包含智能大对象，则至少需要一个智能大对象空间。智能大对象包括 BLOB 或 CLOB 数据，以及过大而无法放入一行的数据和表统计信息。 智能大对象空间必须为空才能分配给租户数据库。 智能大对象空间分配给租户数据库之前必须存在。 在租户数据库以外创建的智能大对象不可存储在租户数据库的智能大对象空间。 某些 SinoDB® 功能，例如： Enterprise Replication 和基本文本搜索，需要智能大对象空间。
num	要运行的虚拟处理器数。	如果未包括 num 属性，则启动一个虚拟处理器。
sbspacetemp	分配给租户数据库的一个或多个临时智能大对象空	如果省略这个属性，则临时智能大对象存储在 SBSPACETEMP 配置参数指定的临时智能大对象空间中。

元素	描述	关键注意事项
	间的列表。用逗号分隔临时智能大对象空间名。	
session_limit_locks	会话可用的最大锁数。	该值必须是 500 - 2147483648。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_locks 属性的值优先于 SESSION_LIMIT_LOCKS 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_LOCKS 配置参数设置锁数。如果也未设置 SESSION_LIMIT_LOCKS，则会话的最大锁数为 2147483648。 您可通过将 IFX_SESSION_LIMIT_LOCKS 环境选项设置为低于 session_limit_locks 属性值的值来覆盖会话的 session_limit_locks 属性。
session_limit_logspace	会话可用于单个事务的最大日志空间量。	该值必须为 5120 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_logspace 属性的值优先于 SESSION_LIMIT_LOGSPACE 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_LOGSPACE 配置参数设置日志空间量。如果也未设置 SESSION_LIMIT_LOGSPACE，则会话可用于单个事务的最大日志空间量为 2147483648 KB。
session_limit_memory	会话可分配的最大内存量。	该值必须是 20480 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_memory 属性的值优先于 SESSION_LIMIT_MEMORY 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_MEMORY 配置参数设置内存空间量。如果也未设置 SESSION_LIMIT_MEMORY，则会话可分配的最大内存空间量为 2147483648 KB。
session_limit_tempspace	会话可分配的临时表空间的最大量。	该值必须是 20480 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_tempspace 属性的值优先于 SESSION_LIMIT_TEMPSPACE 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_TEMPSPACE 配置参数设置临时表空间量。如果也未设置 SESSION_LIMIT_TEMPSPACE，则会话可分配的最大临时表空间量为 2147483648 KB。
session_limit_txn_time	事务在会话中可运行的最大时间量。	该值必须是 60 - 20000000000。 以秒为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。

元素	描述	关键注意事项
		session_limit_txn_time 属性的值优先于 SESSION_LIMIT_TXN_TIME 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_TXN_TIME 配置参数设置时间量。如果也未设置 SESSION_LIMIT_TXN_TIME，则事务在会话中可运行的最大时间量为 20000000000 秒。
tenant_limit_space	租户数据库在磁盘上的最大存储空间量。 当达到该限制时，则拒绝后续需要更多磁盘空间的操作。	该值必须是 1048576 - 1717986918400 KB（1 GB - 200 TB）。以 KB、MB、GB 或 TB 为单位。 tenant_limit_space 属性的值优先于 TENANT_LIMIT_SPACE 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_SPACE 配置参数设置空间量。如果也未设置 TENANT_LIMIT_SPACE，则租户用户可用的最大存储空间量为 1717986918400 KB。
tenant_limit_memory	连接到租户数据库的所有会话的最大共享内存量。当超过该限制时，则终止使用最多共享内存的会话。	该值必须是 102400 - 2147483648 KB（100 MB - 2 TB）。以 KB、MB、GB 或 TB 为单位。 tenant_limit_memory 属性的值优先于 TENANT_LIMIT_MEMORY 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_MEMORY 配置参数设置内存量。如果也未设置 TENANT_LIMIT_MEMORY，则租户会话可用的最大内存量为 2147483648 KB。
tenant_limit_connections	到租户数据库的最大连接数。当达到该限制时，则拒绝后续的连接请求。	该值必须是 1 - 65536。 tenant_limit_connections 属性的值优先于 TENANT_LIMIT_CONNECTIONS 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_CONNECTIONS 配置参数设置连接数。如果也未设置 TENANT_LIMIT_CONNECTIONS，则租户数据库的最大连接数为 65536。
vpclass	运行租户数据库会话线程的虚拟处理器类名。	如果省略这个属性，则会话线程在 CPU 虚拟处理器上运行。 值必须是 8 个或更少的字符。最多可创建 200 个租户虚拟处理器类。 如果虚拟处理器类名是唯一的，则创建新的租户虚拟处理机类。如果虚拟处理器类名存在，则该租户数据库与其他租户数据库共享该类。 当删除租户虚拟处理器时，则直到数据库服务器重新启动才会释放虚拟处理器类 ID 资源。

用法

您必须有 DBA 权限或被授予 TENANT 权限来运行这个命令。只有每个属性的第一次出现才有效。

运行带有 tenant create 参数的 admin() 或 task() 来创建租户数据库。授予 DBA 权限的用户才可创建数据库。您可在 sysadmin 数据库的 tenant 表中查看租户数据库属性。

以下语句创建名为 company_A 的租户数据库：

```
EXECUTE FUNCTION task('tenant create', 'companyA',
   '{dbspace:"company_A_dbs1,company_A_dbs2,company_A_dbs3",
     sbspace:"company_A_sbs",
```

```
      vpclass:"tvp_A,num=6",
      dbspacetemp:"company_A_tdbs",
      session_limit_locks:"1000",
      session_limit_memory:"100MB",
      session_limit_tempspace:"25MB",
      session_limit_logspace:"30MB",
      session_limit_txn_time:"120",
      tenant_limit_space:"2TB",
      tenant_limit_memory:"1GB",
      tenant_limit_connections:"1000",
      logmode:"ansi"}'
);
```

租户数据库有下列属性：

- 三个专用的数据库空间
- 一个专用的智能大对象空间
- 六个租户虚拟处理器
- 一个专用的临时数据库空间
- 每个会话限定 1000 个锁
- 每个会话限定 100 MB 的内存分配
- 每个会话限定 25 MB 的临时表空间
- 每个会话限定 30 MB 的日志空间
- 事务时间限定为 120 秒
- 租户数据库可使用的总存储空间量限制为 2 TB
- 连接到租户数据库的所有会话的总共享内存量限制为 1 GB
- 连接数限定为 1000
- ANSI 日志记录模式
- 区分大小写
- 临时智能大对象存储于数据库服务器 SBSPACETEMP 配置参数指定的智能大对象空间。
- 无 Blob 空间

相关链接

create database 参数: 创建数据库 *(SQL* 管理 *API)* 在第607页
TENANT_LIMIT_SPACE 配置参数 在第189页
tenant update 参数: 更改租户数据库属性 *(SQL* 管理 *API)* 在第710页
tenant drop 参数: 删除租户数据库 *(SQL* 管理 *API)* 在第710页
SESSION_LIMIT_LOCKS 配置参数 在第163页
SESSION_LIMIT_MEMORY 配置参数 在第164页
SESSION_LIMIT_TEMPSPACE 配置参数 在第165页
SESSION_LIMIT_LOGSPACE 配置参数 在第163页
SESSION_LIMIT_TXN_TIME 配置参数 在第165页
TENANT_LIMIT_SPACE 配置参数 在第189页
TENANT_LIMIT_MEMORY 配置参数 在第188页
TENANT_LIMIT_CONNECTIONS 配置参数 在第188页
onstat -g ses 命令: 显示会话相关信息 在第513页
《*SinoDB* 管理员指南》: 多租户
《*SinoDB GLS* 用户指南》: *DB_LOCALE* 环境变量

tenant drop 参数: 删除租户数据库 (SQL 管理 API)

admin() 或 task() 函数使用 tenant drop 参数来删除租户数据库。

语法

```
EXECUTE FUNCTION
{ admin  | task }
(
'tenant drop',  'database_name'
);
```

元素	描述	关键注意事项
database_name	租户数据库名。	必须是现有的租户数据库。

用法

您必须有 DBA 权限或被授予 TENANT 权限来运行这个命令。不可打开到该数据库的其他连接。

删除数据库中的表和数据。释放专用于租户数据库的存储空间。从 sysadmin 数据库的 tenant 表中移除数据库租户属性。如果相关的虚拟处理器未与任何其他租户数据库关联，则删除。

以下语句删除 companyA 租户数据库：

```
EXECUTE FUNCTION task('tenant drop', 'companyA');
```

相关链接

tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页
tenant update 参数: 更改租户数据库属性 (SQL 管理 API) 在第710页
《*SinoDB 管理员指南*》: 多租户

tenant update 参数: 更改租户数据库属性 (SQL 管理 API)

admin() 或 task() 函数使用 tenant update 参数来更改租户数据库的属性。

Syntax

```
EXECUTE FUNCTION
{ admin  | task }
(
'tenant update',  'database_name','
{   [{  ,blobspace:" , name "  | dbspace:" , name "  | ,dbspacetemp:" , name "  |
,sbspace:" , name "  | ,sbspacetemp:" , name "  | ,session_limit_locks:" number"  |
,session_limit_logspace:" number [{ KB  | MB  | GB }] "  | ,session_limit_memory:" number
[{ KB  | MB  | GB }] "  | ,session_limit_tempspace:" number [{ KB  | MB  | GB }] "  |
,session_limit_txn_time:" number"  | [{ KB  | MB  | GB  | TB }] "  | , tenant_limit_memory: "
number [{ KB  | MB  | GB  | TB }] "  | , tenant_limit_connections: " number "  | ,vpclass:"
name [ ,num= number ] " }]  }
);
```

元素	描述	关键注意事项
blobspace	分配给租户数据库的一个或多个 Blob 空间的	指定的 Blob 空间追加到租户数据库现有的 Blob 空间列表。

元素	描述	关键注意事项
	列表。用逗号分隔 Blob 空间名。	Blob 空间必须为空才可分配给租户数据库。 Blob 空间在分配给租户数据库之前必须存在。
database_name	租户数据库名。	数据库名必须位于数据库服务器上。
dbspace	分配给租户数据库的一个或多个数据库空间的列表。用逗号分隔数据库空间名。	指定的数据库空间追加到租户数据库现有的数据库空间列表。 数据库空间必须为空才可分配给租户数据库。 数据库空间在分配给租户数据库之前必须存在。
dbspacetemp	分配给租户数据库的一个或多个临时数据库空间的列表。用逗号分隔临时数据库空间名。	取代现有的 dbspacetemp 属性值。 您可通将 DBSPACETEMP 环境变量设置为 dbspacetemp 属性指定的数据库空间的子集来覆盖会话的 dbspacetemp 属性。 如果省略 dbspacetemp 属性，则临时表存储于 DBSPACETEMP 配置参数或环境变量指定的临时数据库空间中。
sbspace	分配给租户数据库的一个或多个智能大对象空间的列表。用逗号分隔智能大对象空间名。	指定的智能大对象空间追加到租户数据库现有的智能大对象空间列表。 智能大对象空间必须为空才可分配给租户数据库。 智能大对象空间在分配给租户数据库之前必须存在。
num	要运行的虚拟处理器数。	如果未包括 num 属性，则启动一个虚拟处理器。
sbspacetemp	分配给租户数据库的一个或多个临时智能大对象空间的列表。用逗号分隔临时智能大对象空间名。	取代现有 sbspacetemp 属性值。
session_limit_locks	对于没有 DBA 权限的用户，会话的最大锁数。	取代现有的 session_limit_locks 属性值。 该值必须是 500 - 2147483648。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 如果未设置这个属性，则由 SESSION_LIMIT_LOCKS 配置参数设置锁数。如果也未设置 SESSION_LIMIT_LOCKS，则会话的最大锁数为 2147483648。 您可通过将 IFX_SESSION_LIMIT_LOCKS 环境选项设置为低于 session_limit_locks 属性值的值来覆盖会话的 session_limit_locks 属性。
session_limit_logspace	会话可用于单个事务的最大日志空间量。	取代现有的 session_limit_logspace 属性值。 该值必须是 5120 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。

元素	描述	关键注意事项
		session_limit_logspace 属性的值优先于 SESSION_LIMIT_LOGSPACE 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_LOGSPACE 配置参数设置日志空间量。如果也未设置 SESSION_LIMIT_LOGSPACE，则会话可用于单个事务的最大日志空间量为 2147483648 KB。
session_limit_memory	会话可分配的最大内存量。	取代现有的 session_limit_memory 属性值。 该值必须是 20480 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_memory 属性的值优先于 SESSION_LIMIT_MEMORY 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_MEMORY 配置参数设置内存空间量。如果也未设置 SESSION_LIMIT_MEMORY，则会话可分配的最大内存空间量为 2147483648 KB。
session_limit_tempspace	会话可分配的临时表空间的最大量。	取代现有的 session_limit_tempspace 属性值。 该值必须是 20480 - 2147483648 KB。以 KB、MB 或 GB 为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_tempspace 属性的值优先于 SESSION_LIMIT_TEMPSPACE 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_TEMPSPACE 配置参数设置临时表空间量。如果也未设置 SESSION_LIMIT_TEMPSPACE，则会话可分配的最大临时表空间量为 2147483648 KB。
session_limit_txn_time	事务在会话中可运行的最大时间量。	取代现有的 session_limit_txn_time 属性值。 该值必须是 60 - 20000000000。 以秒为单位。 这个限制不适用于具有管理权限的用户，例如：用户 informix 或 DBSA 用户。 session_limit_txn_time 属性的值优先于 SESSION_LIMIT_TXN_TIME 配置参数的值。如果省略这个属性，则由 SESSION_LIMIT_TXN_TIME 配置参数设置时间量。如果也未设置 SESSION_LIMIT_TXN_TIME，则事务在会话中可运行的最大时间量为 20000000000 秒。
tenant_limit_space	租户数据库在磁盘上的最大存储空间量。 当达到该限制时，则拒绝后续需要更多磁盘空间的操作。	取代现有的 tenant_limit_space 属性值。 该值必须是 1048576 - 1717986918400 KB（1 GB - 200 TB）。以 KB、MB、GB 或 TB 为单位。 tenant_limit_space 属性的值优先于 TENANT_LIMIT_SPACE 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_SPACE 配置参数设置空间量。如果也未设置 TENANT_LIMIT_SPACE，则租户用户可用的最大存储空间量为 1717986918400 KB。

元素	描述	关键注意事项
tenant_limit_memory	连接到租户数据库的所有会话的最大共享内存量。当超过该限制时，则终止使用最多共享内存的会话。	取代现有的 tenant_limit_memory 属性值。 该值必须是 102400 - 2147483648 KB（100 MB - 2 TB）。以 KB、MB、GB 或 TB 为单位。 tenant_limit_memory 属性的值优先于 TENANT_LIMIT_MEMORY 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_MEMORY 配置参数设置内存量。如果也未设置 TENANT_LIMIT_MEMORY，则租户会话可用的最大内存量为 2147483648 KB。
tenant_limit_connections	到租户数据库的最大连接数。当达到该限制时，则拒绝后续的连接请求。	取代现有的 tenant_limit_connections 属性值。 该值必须是 1 - 65536。 tenant_limit_connections 属性的值优先于 TENANT_LIMIT_CONNECTIONS 配置参数的值。如果省略这个属性，则由 TENANT_LIMIT_CONNECTIONS 配置参数设置连接数。如果也未设置 TENANT_LIMIT_CONNECTIONS，则租户数据库的最大连接数为 65536。
vpclass	运行租户数据库会话线程的虚拟处理器类名。	取代 vpclass 属性值。 如果省略这个属性，则会话线程在 CPU 虚拟处理器上运行。 值必须是 8 个或更少的字符。最多可创建 200 个租户虚拟处理器类。 如果虚拟处理器类名是唯一的，则创建新的租户虚拟处理机类。如果虚拟处理器类名存在，则该租户数据库与其他租户数据库共享该类。 当删除租户虚拟处理器时，则直到数据库服务器重新启动才会释放虚拟处理器类 ID 资源。

用法

您必须是用户 informix 或 DBSA 用户，或者您必须有 TENANT 权限来运行这个命令。

数据库属性的变更对新会话生效。

以下语句更新名为 company_A 的租户数据库的属性：

```
EXECUTE FUNCTION task('tenant update', 'company_A',
                      '{dbspace:"company_A_dbs4,company_A_dbs5",
                        sbspace:"company_A_sbs3",
                        vpclass:"tvp_B",
                        session_limit_txn_time:"120"}'
);
```

该租户数据库获得两个数据库空间和一个智能大对象空间。更改虚拟处理器类。事务的时间限制变更为 120 秒。

相关链接

TENANT_LIMIT_SPACE 配置参数 在第189页
tenant create 参数: 创建租户数据库 (SQL 管理 API) 在第705页
tenant drop 参数: 删除租户数据库 (SQL 管理 API) 在第710页

SESSION_LIMIT_LOCKS 配置参数 在第163页
SESSION_LIMIT_MEMORY 配置参数 在第164页
SESSION_LIMIT_TEMPSPACE 配置参数 在第165页
SESSION_LIMIT_LOGSPACE 配置参数 在第163页
SESSION_LIMIT_TXN_TIME 配置参数 在第165页
TENANT_LIMIT_SPACE 配置参数 在第189页
TENANT_LIMIT_CONNECTIONS 配置参数 在第188页
onstat -g ses 命令: 显示会话相关信息 在第513页
《*SinoDB* 管理员指南》: 多租户
《*SinoDB GLS* 用户指南》: *DB_LOCALE* 环境变量

附录

A

附录

数据库服务器文件

数据库服务器文件创建于缺省目录下，或相关配置参数指定的目录下。数据库管理员可能需要编辑或检查数据库服务器使用的文件的内容。

- 表 *200:* 您可使用的数据库服务器文件 在第715页罗列您可能需要查看、复制、编辑、搬移或删除的数据库服务器文件（除非另有说明）。
- 表 *201:* 仅供内部使用的数据库服务器文件 在第718页罗列仅内部使用的数据库服务器文件。您不可编辑、搬移或删除这些文件。

表 200: 您可使用的数据库服务器文件

这个表罗列您配置和使用数据库服务器时可能参考或使用的文件。

文件名	目录	用途	创建者
af.*xxx* *xxx* 标识特定的断言失败	$INFORMIXDIR/tmp（UNIX™） %INFORMIXDIR%\tmp（Windows™） 由 DUMPDIR 配置参数指定	断言失败信息	数据库服务器
ac_msg.log	/tmp（UNIX™） %INFORMIXDIR%\etc（Windows™）	archecker 实用程序的消息日志	数据库服务器
ac_config.std	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR%\etc（Windows™）	archecker参数值的样本	数据库服务器
bar_act.log	/tmp（UNIX™） %INFORMIXDIR%\etc（Windows™） 由 BAR_ACT_LOG 配置参数指定	ON-Bar 活动日志	ON-Bar
bar_dbug.log	/usr/informix/（UNIX™） \usr\informix\（Windows™）	ON-Bar 调试日志	ON-Bar

文件名	目录	用途	创建者
	由 BAR_DEBUG_LOG 配置参数指定		
bldutil.*process_id*（UNIX™） bldutil.out（Windows™）	/tmp（UNIX™） \tmp（Windows™）	关于构建 sysutils 数据库的错误消息	数据库服务器
buildsmi.out（UNIX™） buildsmi_out.*%INFORMIXSERVER%*（Windows™）	/tmp（UNIX™） %INFORMIXDIR%\etc（Windows™）	关于构建 sysmaster 数据库的错误消息	数据库服务器
concdr.sh	$INFORMIXDIR /etc/conv（UNIX™） %INFORMIXDIR% \etc \conv（Windows™）	升级过程中转换 syscdr 数据库	数据库服务器
core（UNIX™）	数据库服务器的启动目录	核心转储	数据库服务器
gcore.*xxx*（UNIX™）	$INFORMIXDIR/tmp（UNIX™） %INFORMIXDIR% \tmp（Windows™） 由 DUMPDIR 配置参数指定	断言失败信息	数据库服务器
.informix（UNIX™）	用户主目录	设置个人环境变量	用户
informix.rc（UNIX™）	$INFORMIXDIR/etc	设置所有用户的缺省环境变量	数据库管理员
InstallServer.log（Windows™）	C:\tmp	数据来服务器安装日志	数据库服务器
ixbar.*servernum*	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR% \etc（Windows™）	冷恢复中使用的紧急启动文件	ON-Bar
jvp.log	/urs/informix 由 JVPLOGFILE 配置参数指定	Java™ 虚拟处理器的消息	数据库服务器
.jvpprops	urs/informix/extend/krakatoa 由 JVPPROPFILE 配置参数指定	Java™ VP 属性的样本	安装过程中
oncfg_*servername.servernum*	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR% \etc（Windows™）	通过 ON-Bar 全系统恢复的配置信息	数据库服务器
online.log	$INFORMIXDIR/tmp（UNIX™） %INFORMIXDIR% （Windows™）	数据库服务器消息日志，包含错误消息和状态信息	数据库服务器

文件名	目录	用途	创建者
	由 MSGPATH 配置参数指定		
onconfig	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR%\etc（Windows™）	配置信息	数据库管理员或数据库服务器管理员
onconfig.std	$INFORMIXDIR/etc（UNIX™）	配置参数值的样本 重要： 不要搬移、更改或删除 onconfig.std 文件，除非星瑞格®软件支持指示您去做。然而，您可以复制 onconfig.std 文件以创建自定义配置文件，并将该副本搬移到其他位置。	安装过程中
psm_act.log	/tmp（UNIX™） %INFORMIXDIR%\etc（Windows™） 由 PSM_ACT_LOG 配置参数指定	SinoDB® 主要存储管理器的日志文件	ON-Bar
pua.map	$INFORMIXDIR/gls/etc（UNIX™） %INFORMIXDIR%\gls\etc\(Windows™）	用于显示 Unicode 私人使用区域（PUA）范围内字符的映射文件。	用户
revcdr.sh（UNIX™） revcdr.bat（Windows™）	$INFORMIXDIR/etc/conv（UNIX™） %INFORMIXDIR%\etc\conv(Windows™）	将 syscdr 数据库还原到较早的格式	数据库服务器
shmem.*xxx*	$INFORMIXDIR/tmp（UNIX™） %INFORMIXDIR%\tmp（Windows™） 由 DUMPDIR 配置参数指定	断言失败信息	数据库服务器
sm_versions.std	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR%\etc（Windows™）	标识使用中的存储管理器	安装过程中
sqlhosts.*servername*	$INFORMIXDIR/etc %INFORMIXDIR%\etc（Windows™）	连接信息	安装过程中；由数据库服务器管理员变更

文件名	目录	用途	创建者
			该文件扩展为服务器名（缺省扩展为 ol_informix*version*）
sqlhosts.std	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR%\etc（Windows™）	连接信息样本	安装过程中

表 201：仅供内部使用的数据库服务器文件

该表罗列数据库服务器需要的文件。

重要： 不要搬移、更改或删除这些文件，除非星瑞格®软件支持指示您去做。

文件名	目录	用途	创建者
illlsrra.*xx*	$INFORMIXDIR/lib（UNIX™） %INFORMIXDIR%\lib（Windows™）	数据库服务器和一些实用程序的共享库	安装程序
INFORMIXTMP	/INFORMIXTMP（UNIX™） \%INFORMIXDIR%（Windows™）	内部文件的临时目录	数据库服务器
.inf.servicename	/INFORMIXTMP（UNIX™） drive:\INFORMIXTMP（Windows™）	连接信息	数据库服务器
.infos.*dbservername*	$INFORMIXDIR/etc（UNIX™） %INFORMIXDIR%\etc（Windows™）	连接信息	数据库服务器
JVM_*vpid*	由 JVPLOG 配置参数指定	Java™ 虚拟机产生的消息	Java™ 虚拟机
servicename.exp	/INFORMIXTMP（UNIX™） *drive*:\INFORMIXTMP（Windows™）	连接信息	数据库服务器
servicename.str	/INFORMIXTMP（UNIX™） *drive*:\INFORMIXTMP（Windows™）	连接信息	数据库服务器
VP.*servername.nnx*	/INFORMIXTMP（UNIX™） *drive*:\INFORMIXTMP（Windows™）	连接信息	数据库服务器

相关链接

onstat -c 命令：显示 *ONCONFIG* 文件内容 在第408页
onstat -g cfg 命令：显示配置参数的当前值 在第438页

onconfig 门户: 按功能类别划分的配置参数 在第43页
《*SinoDB* 管理员指南》: 数据库服务器配置
设置实用程序的本地环境变量 在第290页
oninit 实用程序 在第319页
《*SinoDB* 管理员指南》: 数据库服务器配置

故障排除错误

有时，一系列事件导致数据库服务器返回非预期的错误代码。

您可使用以下诊断工具来获取故障排除错误的信息：

- onmode -I
- 跟踪点
- ifxcollect 工具

使用 onmode -I 收集诊断信息

要收集额外诊断信息，请使用 onmode -I 来指示数据库服务器执行《*SinoDB*® 管理员指南》所描述的诊断收集程序。 要在遭遇错误号时使用 onmode -I，请提供 *iserrno* 和可选的会话 ID。有关 onmode 的更多信息，请参阅 *onmode* 实用程序 在第330页。

创建跟踪点

跟踪点在调试用 C 编写的用户定义例程是很有用。您可创建用户定义跟踪点来发送关于用户定义例程的当前执行状态的特殊信息。

每个跟踪点有以下部分：

- 跟踪组将相关跟踪点归在一起，以便可同时打开或关闭它们。

  您可使用名为 _myErrors 的内置跟踪或创建您自己的跟踪。要创建您自己的跟踪，请在 systracees 系统目录表中插入行。
- 跟踪消息 是数据库服务器发送到跟踪输出文件的文本。

  您可将国际化的跟踪消息存储在 systracemsgs 系统目录表中。
- 跟踪点阈值 确定何时执行跟踪点。

在缺省情况下，数据库服务器将所有跟踪消息放置在 tmp 目录下具有以下文件名的跟踪输出文件中：

```
session_num.trc
```

有关跟踪用户定义例程的更多信息，请参阅《*SinoDB*® *DataBlade*® *API* 程序员指南》。

使用 ifxcollect 工具收集数据

如有必要，您可使用 ifxcollect 工具来收集诊断数据，以排除特定问题，例如：断言失败。您还可指定选项以通过文件传输协议（FTP）传送收集的数据。

ifxcollect 工具位于 $INFORMIXDIR/bin 目录。ifxcollect 命令产生的输出文件位于 $INFORMIXDIR/isa/data 目录。

每个类和子类收集的数据类型是预定义在 $INFORMIXDIR/isa/ 目录中的 XML 文件中。可修改这些 XML 文件以添加或移除特定的命令。

> 重要： XML 文件可包含覆盖选项（为数据收集所指定的选项）的命令。例如：一个 XML 文件可能包含休眠命令，该命令以较短的秒数覆盖 -d 选项；或者，一个 XML 文件可能包含 onstat -z 的调用。

语法

```
ifxcollect -c category -s subcategory
[ -r number_of_times ]
[ -d seconds ]
[ -y ]
[{ -V | -version }]
[{ -f | -e | -p | -m | -l | -u | -w }]
```

表 202: 数据收集选项

元素	描述	关键注意事项
-c *category*	告诉服务器收集指定类的数据。	必须指定要收集的数据的类。
-s *category*	告诉服务器收集指定的子类的数据。	必须指定要收集的数据的子类。
-r *number of times*	指定重复数据收集的次数。	可选的。缺省值为 1。
-d *number of seconds*	指定在收集操作之间暂停的次数。	可选的。缺省值是 0。
-y	Causes the database server to automatically respond yes to all prompts.使数据库服务器自动地对所有提示响应“yes”	可选的。
-V	显示软件版本号和序列号。	可选的。 请参阅 获取实用程序版本信息 在第289页。
-version	显示构建版本、主机操作系统、编号和日期，以及 GLS 版本。	可选的。 请参阅 获取实用程序版本信息 在第289页。

表 203: FTP 选项（如果还传输数据）

元素	描述	关键注意事项
-f	FTP 整个收集	传输数据所必需的。
-e *email address*	电子邮件地址	传输数据所必需的。
-p *the PMR number*	PMR 编号	传输数据所必需的。
-m *machine name*	要连接的机器	传输数据所必需的。
-l *directory*	包含数据的目录	传输数据所必需的。
-u *user name*	FTP 的用户名	传输数据所必需的。
-w *password*	FTP 的密码	传输数据所必需的。

用法

下表显示命令中可使用的类和子类。

表 204: 类和子类的组合

类和子类	解释
-c ids -s general	收集与所有 SinoDB® 产品相关的问题的一般数据

类和子类	解释
-c af -s general	收集断言失败的一般数据
-c er -s general	收集 Enterprise Replication 的一般数据
-c er -s init	收集 Enterprise Replication 初始化问题的一般数据
-c performance -s general	收集性能问题的数据
-c performance -s cpu	收集 CPU 利用率问题的数据
-c onbar -s archive_failure	收集 onbar 归档失败的数据
-c onbar -s restore_failure	收集 onbar 恢复失败的数据
-c ontape -s archive_failure	收集 ontape 归档失败的数据
-c ontape -s restore_failure	收集 ontape 恢复失败的数据
-c connection -s failure	收集连接失败的数据
-c connection -s hang	收集连接挂起的数据
-c cust -s prof	收集客户概要文件信息

要查看所有 ifxcollect 实用程序命令选项，请在命令提示输入 ifxcollect。

示例

要收集一般断言失败的信息，请运行这个命令：

```
ifxcollect -c af -s general
```

要收集与 CPU 利用率相关的性能问题的信息，请运行这个命令：

```
ifxcollect -c performance -s cpu
```

要包括 FTP 信息，请指定如这个示例显示的附加信息：

```
-f -e user_name@company_name.org -p 9999.999.999
-f -m machine -l /tmp -u user_name -w password
```

事件警报

数据库服务器提供根据数据库服务器环境中发生的事件自动地触发管理操作的机制。这个机制就是事件警报功能。

事件可以是信息性的（例如：Backup Complete）或指示需要您的注意的错误状况（例如：Unable to Allocate Memory）。

相关链接

ALARMPROGRAM 配置参数 在第65页

ALRM_ALL_EVENTS 配置参数 在第66页

使用 ALARMPROGRAM 捕获事件

在 UNIX™ 上使用 alarmprogram.sh 与在 Windows™ 上使用 alarmprogram.bat shell 脚本来处理事件警报与自动自动日志备份。关于设置指示，请参阅 *ALARMPROGRAM* 配置参数 在第65页。

要仅自动执行逻辑日志备份，提供了两个现成脚本：log_full.[sh|bat] 和 no_log.[sh|bat]。将 ALARMPROGRAM 设置为脚本的完整路径名。有关信息，请参阅 *ALARMPROGRAM* 配置参数 在第65页。

设置 ALRM_ALL_EVENTS

您可设置 ALRM_ALL_EVENTS 来指定 ALARMPROGRAM 是对记录到 MSGPATH 的所有事件还是仅对指定的值得的事件（事件严重性大于 1 的事件）运行。

编写自己的警报脚本

另一方式是您可编写包含事件警报参数的自己的 shell 脚本、batch 文件或二进制程序。当事件发生时，数据库服务器调用这个可执行的文件并将事件警报参数（请参阅表 *205:* 事件警报参数 在第724页）传给该文件。例如：当发生表失败时，您的脚本使用 _id 和 _msg 参数来执行管理操作。将 ALARMPROGRAM 设置为这个可执行文件的完整路径名。

相关链接

ALARMPROGRAM 配置参数 在第65页

定制 ALARMPROGRAM 脚本

您可以根据您的环境定制 ALARMPROGRAM 脚本。

mail 实用程序必须已存在。

请按照这些步骤来定制 alarmprogram.[sh|bat] 脚本。您可使用 alarmprogram.[sh|bat] 而非 log_full.[sh|bat] 来自动执行日志备份。

定制 ALARMPROGRAM 脚本：

1. 将 ADMINMAIL 的值更改为数据库服务器管理员的邮件地址。
2. 将 PAGERMAIL 的值更改为传呼服务邮件地址。
3. 设置 MAILUTILITY 参数的值。
   - UNIX™：/usr/bin/mail
   - Windows™：$INFORMIXDIR/bin/ntmail.exe
   - Linux™：/usr/lib/sendmail -t
4. 当逻辑日志满时要自动地执行备份，请将 BACKUP 更改为 yes。

   要停止自动的日志备份，请将 BACKUP 更改为 yes 以外的任何其他值。
5. 在 ONCONFIG 文件中，将 ALARMPROGRAM 设置为 alarmprogram.[sh|bat] 的完整路径名。
6. 重新启动数据库服务器。

严重性 1 或 2 的警报不会写任何消息到消息日志，也不会发送电子邮件。严重性 3 或更高的警报发送电子邮件给数据库管理员。严重性 4 和 5 的警报还会通过电子邮件通知传呼机。

相关链接

ALARMPROGRAM 配置参数 在第65页

警报脚本的前台操作的注意事项

要确保连续的服务器可用性，请不要在警报脚本中运行某些前台操作。

当服务器调用警报脚本时，服务器有时会等待脚本完成，然后再继续。例如：

- 当由于致命错误而调用警报时，服务器会等待该脚本完成向错误日志写入信息。在某些情况下，警报事件 5 和 6 会在前台运行。
- 某些 Enterprise Replication 事件警报在前台运行，例如：事件警报 31、34、37 和 39。

由于服务器可能需要等待警报程序脚本完成，所以不要在警报脚本中的前台运行以下操作：

- 强制用户与服务器断开连接的 onmode 命令，例如：onmode -u 或 onmode -yuk。这些 onmode 命令可能在服务器和该警报脚本之间导致死锁，因为服务器可能在等待警报脚本完成，而执行 onmode 命令的警报脚本在等待用户会话关闭且那些会话之一正在运行该警报脚本。
- 可能需要花费长时间才可完成的操作或具有高可变运行时间的操作。 需要花费长时间才可完成的操作可能导致服务器在该操作运行时显示其为无响应。

如果您需要在警报脚本中运行上述操作，则使用以下操作系统实用程序之一在后台中运行它们：

在 UNIX™ 上：使用 nohup 实用程序。例如：nohup onmode -yuk & 指示 nohup 继续运行该命令，即使它的父命令终止，并且 & 符号使该命令在后台运行而不会阻塞该警报程序脚本本身的执行。

在 Windows™ 上：使用带有 /B 标志的 start 实用程序。例如：start /B onmode -yuk。

解释事件警报消息
数据库服务器报告给消息日志的某些事件会触发警报程序。类消息指示数据库服务器报告的事件。

数据库服务器在消息日志中报告一个非零的退出代码。在警报程序中，将 EXIT_STATUS 变量设置为 0 表示成功完成，设置为另一个数表示失败。

例如：如果线程试图获取一个锁，但最大的锁数已经达到，则数据库服务器向消息日志写入以下消息：

```
10:37:22  Checkpoint Completed:  duration was 0 seconds.
10:51:08  Lock table overflow - user id 30032, rstcb 10132264
10:51:10  Lock table overflow - user id 30032, rstcb 10132264
10:51:12  Checkpoint Completed:  duration was 1 seconds.
```

当数据库服务器运行 alarmprogram.sh 或 alarmprogram.bat 程序，或您的警报程序，则数据库服务器产生描述事件严重性和类的消息。如果严重性大于 2，则该消息使用以下格式：

操作	消息
一个合理严重的服务器事件	Severity: 3 Class ID: 21 Class msg: Database server resource overflow: 'Locks'. Specific msg: Lock table overflow - user id 30032, rstcb 10132264 See Also: # optional message Event ID: 21005
该消息出现在每个邮寄的消息末尾	This e-mail was generated by the server ALARMPROGRAM script on *servername* because something untoward just happened to *eventname*.

ph_alert 表中的事件

产生的所有事件警报都会插入 sysadmin 数据库中的 ph_alert 表中。

您可以在本地或远程服务器上查询 ph_alert 表来查看那个服务器的最近的事件警报。您可根据 ph_alert 表编写 SQL 脚本来处理事件警报而不是使用 ALARMPROGRAM 配置参数控制的脚本。

在缺省情况下，警报在 ph_alert 表中保留 15 天，之后将会清除。

示例

以下示例显示 ph_alert 表中的一个事件警报：

```
SELECT * FROM ph_alerts WHERE alert_object_type=ALARM;
```

```
id                  34
alert_task_id       18
alert_task_seq      10
alert_type          INFO
alert_color         YELLOW
alert_time          2010-03-08 12:05:48
alert_state         NEW
alert_state_chang+  2010-03-08 12:05:48
alert_object_type   ALARM
alert_object_name   23
alert_message       Logical Log 12 Complete, timestamp: 0x8e6a1.
alert_action_dbs    sysadmin
alert_action
alert_object_info   23001
```

相关链接

ph_alert 表 在第247页
事件警报参数 在第724页

事件警报参数

事件警报有 5 个参数来描述每个事件。

下表罗列事件警报的部分参数。

表 205: 事件警报参数

参数	描述	数据类型
severity	事件严重性。	integer
class_id	数字标识符用于分类已发生事件的类型。	integer
class_msg	描述事件类型的简要消息。	string
specific_msg	描述已发生事件的特定消息。	string
see_also	包含关于事件其他信息的文件引用。	string
uniqueid	特定消息的唯一事件标识符。	bigint

事件严重性

事件严重性代码是一个事件严重性的数字指标。消息日志中的每个事件都包含一个严重性代码。事件严重性代码是发送到警报程序的第一个参数。在 ph_alert 表中，事件严重性通过警报颜色和警报类型的组合来表示。事件严重性代码罗列在下表。

表 206: 事件严重性代码

严重性	描述
1	不值得注意。事件（例如：消息日志中的日期更改）不报告给警报程序，除非启用了 ALRM_ALL_EVENTS 配置参数。 在 ph_alert 表中，警报颜色是 GREEN，警报类型是 INFO。
2	信息。没有发生错误，但某些例程事件成功地完成（例如：检查点或日志备份完成）。 在 ph_alert 表中，警报颜色是 YELLOW，警报类型是 INFO。
3	注意。事件不危及数据或阻碍系统的使用；然而，该事件值得您的注意。例如：镜像对当中的一个块关闭。向系统管理员发送邮件。

严重性	描述
	在 ph_alert 表中，警报颜色是 YELLOW，警报类型是 WARNING。
4	紧急。发生意外事件，可能危及数据或访问数据。例如：断言失败或 oncheck 报告数据损坏。立即采取行动。当此类事件严重性发生时，将传呼系统管理员。 在 ph_alert 表中，警报颜色是 RED，警报类型是 ERROR。
5	致命。发生意外事件且导致数据库服务器故障。当此类事件严重性发生时，将传呼系统管理员。 在 ph_alert 表中，警报颜色是 RED and the alert type is ERROR.

类 ID

类 ID 是一个整数，标识导致数据库服务器运行警报程序的事件。类 ID 是数据库服务器显示在警报程序中的第二个参数。

类 ID 存储在 ph_alert 表的 alert_object_name 列中。

类消息

类消息是简要描述或分类导致数据库服务器运行警报程序的事件的文本消息。类消息是数据库服务器显示在警报程序中的第三个参数。

特定消息

特定消息是更详细描述导致数据库服务器运行警报程序的事件的文本消息。特定消息是数据库服务器显示在警报程序中的第四个参数。对于许多警报，这个消息的文本与该事件写入消息日志的消息是相同的。

特定消息存储在 ph_alert 表的 alert_message 列中。

另见途径

对于某些事件，数据库服务器在事件发生时会向文件写入其他信息。此内容的路径名是指数据库服务器写入其他信息的文件的路径名。

事件 ID

事件 ID 是每个特定消息的唯一编号。您可以在定制的警报处理脚本中使用事件 ID，以创建对特定事件的响应。

事件 ID 存储在 ph_alert 表的 alert_object_info 列中。

相关链接

STORAGE_FULL_ALARM 配置参数 在第183页
ph_alert 表中的事件 在第723页
SHMVIRT_ALLOCSEG 配置参数 在第169页

事件警报 ID

事件警报的类 ID 标识事件的类型。 事件 ID 指示特定的事件。

下表列出事件警报 ID 和消息或从哪里可以获得更多的信息。许多警报有其他解释和用户操作。许多触发事件警报的问题也会导致联机消息日志中的消息。消息日志的位置由 MSGPATH 配置参数指定。

表 207: 事件警报

ID	严重性	消息	解释
类 ID: 1 事件 ID: 1001	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: Page allocation error on '*object*'	数据库服务器在对表或索引的页分配过程中检测到不一致。 联机日志:断言失败或断言警告,带有问题详细信息。 服务器状态:取决于发现的问题的性质。 用户操作:查看 online.log 文件以获得合适的操作。您需要在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。数据库服务器有时会自动地解决了问题并在 online.log 文件中标识解决方法。
类 ID: 1 事件 ID: 1002	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: Row allocation error on '*object*'	数据库服务器在对表或索引的行分配过程中检测到不一致。 联机日志:断言失败或断言警告,带有问题详细信息。 服务器状态:取决于发现的问题的性质。 用户操作:查看 online.log 文件以获得合适的操作。您需要在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。数据库服务器有时会自动地解决了问题并在 online.log 文件中标识解决方法。
类 ID: 1 事件 ID: 1003	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: Slot allocation error for '*object*'	数据库服务器行处理过程中检测到不一致。 联机日志:断言失败或断言警告,带有问题详细信息。 服务器状态:取决于发现的问题的性质。 用户操作:查看 online.log 文件以获得合适的操作。您需要在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。数据库服务器有时会自动地解决了问题并在 online.log 文件中标识解决方法。
类 ID: 1 事件 ID: 1004	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: An internal error prevented the database server to find the next possible data page in this tblspace.	数据库服务器在对表或索引的行分配过程中检测到位图页不一致。 联机日志:断言失败或断言警告,带有问题详细信息。 服务器状态:取决于发现的问题的性质。 用户操作:查看 online.log 文件以获得合适的操作。在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。

ID	严重性	消息	解释
类 ID： 1 事件 ID： 1005	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： Dropping wrong TBLSpace, requested *tblspace_name* != actual *tblspace_name*	数据库服务器在试图删除表时检测到请求的表和现有的表不匹配。没有表被删除。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作：查看 online.log 文件以获得合适的操作。在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。
类 ID： 1 事件 ID： 1006	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error which may have been caused due to data corruption prevented the database server from altering the bitmap pages for this partition.	在表或索引更改位图页过程中，数据库服务器遭遇可能的数据损坏。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作：查看 online.log 文件以获得合适的操作。在消息中标识的 *'dbname":owne"r.tabname'* 上运行 oncheck 实用程序。
类 ID： 1 事件 ID： 1007	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： [3] An internal error which may have been caused due to corrupted bitmap pages as the database server is still in the process of converting them.	在表或索引更改位图页过程中，数据库服务器遭遇内部位图页的不完整修改。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作： 记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1008	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error which may have been caused due to unconverted bitmap pages.	在表或索引更改位图页过程中，数据库服务器遭遇内部位图页修改未完成的情况。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作： 记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1009	4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： Page Check Error in *object*	数据库服务器检查正读取到内部缓冲区的页时检测到不一致。 联机日志：断言失败或断言警告，带有问题描述。 服务器状态：联机或脱机，取决于问题的严重程度。 用户操作：遵循联机日志中的建议。通常，对类消息或数据库提到的表执行 oncheck -cD 命令。
类 ID：	4	类消息：	数据库服务器检测到无效的行 ID。

ID	严重性	消息	解释
1 事件 ID: 1010		Table failure: *'dbname":owne"r.tabname'* 特定消息: Bad rowid *rowid*	联机日志：断言警告，带有发现该问题的位置描述。 服务器状态： 联机 用户操作：通过对类消息或数据库提到的表执行 oncheck -cI 命令以修复索引。
类 ID: 1 事件 ID: 1011	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: Closing TBLSpace *tblspace_name*	数据库服务器确定表或索引已关闭。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作：无。数据库服务器会自动地纠正该问题。
类 ID: 1 事件 ID: 1012	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: Cannot recreate index *index_name* for partnum *partition_number*, iserrno = *error_number*	数据库服务器遭遇阻止重新创建索引的错误。 联机日志：断言失败或断言警告，带有问题详细信息。 服务器状态：联机 用户操作：查看 online.log 文件的索引信息，然后手动地删除并重建索引。
类 ID: 1 事件 ID: 1013	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: An internal error occurred while the database server was trying to initialize the type of set read operation.	数据库服务器无法初始化集读取操作的内部数据结构。 联机日志：断言警告，带有数据库和表的详细信息。 服务器状态：联机 用户操作：查看 online.log 文件的 ISAM 错误代码，以及表和数据库信息。再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID: 1 事件 ID: 1014	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息: An internal error occurred while the database server was trying to read records from the tblspace's pages	当数据库服务器从表或索引读取记录时遭遇内部错误。 联机日志：断言警告，带有数据库和表的详细信息。 服务器状态：联机 用户操作：查看 online.log 文件的表和数据库信息。再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID: 1 事件 ID: 1015	3	类消息: Table failure: *'dbname":owne"r.tabname'* 特定消息:	当数据库服务器从表或索引读取记录时遭遇内部错误。 联机日志：断言警告，带有数据库和表的详细信息。 服务器状态：联机

ID	严重性	消息	解释
		An internal error occurred while the database server was trying to read the current record.	用户操作：查看 online.log 文件的表和数据库信息。再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1016	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was trying to initialize the set read buffer.	当数据库服务器试图初始化集读取缓冲区时触发内部错误。 联机日志：断言警告，带有数据库和表的详细信息。 服务器状态：联机 用户操作：查看 online.log 文件的表和数据库信息。再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1017	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was trying to set the new mode on the bitmap page.	位图从较早版本的数据库服务器转换过程中发生内部错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：无
类 ID： 1 事件 ID： 1018	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was attempting to convert bitmap pages to the correct format.	数据库服务器无法纠正位图页转换过程中发生的错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：记录所有情况，查看 online.log 文件的其他信息，并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1019	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was trying to modify the bitmap pages during light append operation.	数据库服务器在轻量级追加操作过程中遭遇内部错误，且无法定位需要的位图页。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1020	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was trying to perform light scan operation.	数据库服务器执行轻量级扫描操作时遭遇内部错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机

ID	严重性	消息	解释
			用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1021	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while the database server was trying to perform light scan I/O operation.	数据库服务器执行轻量级扫描操作时遭遇内部错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1022	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to validate light append buffer.	数据库服务器执行轻量级扫描操作时遭遇内部错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1023	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to write the next record to the page in the light append buffer.	数据库服务器执行轻量级扫描操作时遭遇内部错误。 联机日志：断言警告，带有问题的详细信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1024	4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to open a light append for a tblspace.	数据库服务器在表空间上的轻量级追加操作过程中遭遇内部错误。 联机日志：断言失败 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1025	4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to load the first bitmap page for a light append operation.	数据库在轻量级追加操作过程中遭遇内部错误。 联机日志： Assertion Failure : Light Append(Redo/Undo): Can't find bitmap page 服务器状态：联机

ID	严重性	消息	解释
			用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1026	4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to write the cached bitmap pages for a light append operation.	数据库在轻量级追加操作过程中遭遇内部错误。 联机日志：断言失败 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1027	2	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal deadlock database condition was caught by the Lock Manager in the database server.	数据库服务器检测到内部死锁数据库状况。 联机日志：断言警告标识涉及死锁的数据库 和表。 服务器状态：联机 用户操作：无
类 ID： 1 事件 ID： 1028	2	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal deadlock database condition was caught by the Lock Manager in the database server.	数据库服务器检测到内部死锁数据库状况。 联机日志：断言警告标识涉及死锁的数据库 和表。 服务器状态：联机 用户操作：无
类 ID： 1 事件 ID： 1029	4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to map the logical page number in the tblspace to its physical location in the chunk.	数据库服务器无法访问表，因为物理页和其逻辑页号的不一致。 联机日志：断言失败，带有页信息。 服务器状态：联机 用户操作：对类消息或数据库中提到的表上运行 oncheck -cDI 命令，修复任何报告的问题，然后尝试再执行该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1030	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to allocate the alter information.	数据库服务器试图读取内部磁盘结构时遭遇内部错误。 联机日志：断言警告，带有问题的详细信息以及表和数据库信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1	3	类消息： Table failure: *'dbname":owne"r.tabname'*	当数据库服务器尝试创建内部操作列表把行从压缩版本变换为最近行的解压缩版本时，遭遇内部错误。

ID	严重性	消息	解释
事件 ID： 1031		特定消息： An internal error was reported by the database server when it tried to prepare the list of operations to be performed on a compressed row.	联机日志：断言警告，带有问题的详细信息以及表和数据库信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1032	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to insert an operation in the list of operations based on the offset in the row where the new operation points to.	当数据库服务器尝试创建内部操作列表把行从压缩版本变换为最近行的解压缩版本时，遭遇内部错误。 联机日志：断言警告，带有问题的详细信息以及表和数据库信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1033	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it detected an inconsistency with the operation list.	当数据库服务器尝试创建内部操作列表把行从压缩版本变换为最近行的解压缩版本时，遭遇内部错误。 联机日志：断言警告，带有问题的详细信息以及表和数据库信息。 服务器状态：联机 用户操作：再次尝试该操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1034	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to free the partition header page.	当数据库服务器试图释放分区的头页时遭遇内部错误。数据库服务器不释放该头页。 联机日志：断言警告带有问题的详细信息、表和数据库信息，以及要运行的特定的 oncheck 命令。 服务器状态：联机 用户操作：查看 online.log 文件的信息，并运行该特定的 oncheck 命令。
类 ID： 1 事件 ID： 1035	3 or 4	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to validate the partition header page.	数据库服务器无法访问表，因为表空间页的验证错误。 联机日志：声明关于该表的详细信息。 服务器状态：联机 用户操作：检查联机日志以获取有关指定表的信息。对该表或该数据库运行 oncheck -pt 命令，并纠正找到的任何错误。再尝试原操作。如果该操作再次失败，请记录所有情况并联系星瑞格®软件支持。

ID	严重性	消息	解释
类 ID： 1 事件 ID： 1036	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to update the special columns list during an alter table command processing.	正在更改表时，数据库服务器处理与该表相关的特殊列列表时遭遇内部错误。 联机日志：断言警告带有问题的详细信息、表和数据库信息，以及要运行的特定的 oncheck 命令。 服务器状态：联机 用户操作：查看 online.log 文件以获取信息并运行指定的 oncheck 命令。
类 ID： 1 事件 ID： 1037	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to log the completion of the alter and remove the associated version information from the tblspace's header page.	数据库服务器试图更改表时遭遇内部错误。 联机日志：断言警告带有问题的详细信息、表和数据库信息，以及要运行的特定的 oncheck 命令。 服务器状态：联机 用户操作：查看 online.log 文件以获取信息并运行指定的 oncheck 命令。
类 ID： 1 事件 ID： 1038	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it detected a buffer inconsistency.	数据库服务器在对它所操作的内部缓冲区进行一致性检查的过程中遭遇内部错误。 联机日志：断言警告带有问题的详细信息、表和数据库信息，以及要运行的特定的 oncheck 命令。 服务器状态：联机 用户操作：再尝试原操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1039	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to construct a forwarded row into a single tuple.	当数据库服务器处理行时遭遇内部错误。 联机日志：断言警告带有问题的详细信息，以及表和数据库信息。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息。再尝试原操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1040	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to read the data from a partition into the set read buffer.	数据库服务器在读取数据的过程中遭遇损坏的记录，并且无法取回该数据。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件。该错误的一些实例需要星瑞格®软件支持的关注。

ID	严重性	消息	解释
类 ID： 1 事件 ID： 1041	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to read the data row for a given rowid.	数据库服务器在从索引读取数据的过程中遭遇损坏的记录，并且无法取回该数据。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取信息，并运行建议的 oncheck 命令。此错误的一些实例需要星瑞格®软件支持的关注。
类 ID： 1 事件 ID： 1042	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to alter the row in memory to the latest schema.	数据库服务器尝试在已更改的表中将记录从较旧版本转换最新版本时遭遇内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息。再尝试原操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 1 事件 ID： 1043	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to undo the alter of a bitmap page.	数据库服务器尝试还原更改内部位图页的操作时，遭遇内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 1 事件 ID： 1044	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to undo the addition of special column descriptors from the tblspace's header page.	数据库服务器尝试还原添加信息到用于追踪表的内部结构的操作时，遭遇内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 1 事件 ID： 1045	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to undo the addition of the new version to a partition.	数据库服务器尝试还原添加信息到用于追踪表的内部结构的操作时，遭遇内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。

ID	严重性	消息	解释
类 ID： 1 事件 ID： 1046	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to allocate the file descriptor for a partition number.	数据库服务器尝试为表或索引创建新的文件描述符时，遭遇内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 1 事件 ID： 1047	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： An internal error was reported by the database server when it tried to free the file descriptor for a partition number.	数据库服务器尝试释放与表或索引相关的内部数据结构时，遇到内部错误。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：无。数据库服务器会内部地纠正该问题。
类 ID： 1 事件 ID： 1048	3	类消息： Table failure: *'dbname":owne"r.tabname'* 特定消息： Error updating table record.	数据库服务器无法更新就地更改的表的数据库记录。它无法写入该记录的新版本。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2001	3 or 4	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： Fragid *fragment_id*, Rowid *rowid* not found for delete in partnum *partition_number*	数据库服务器无法删除记录，因为它在索引中找不到该记录。 联机日志：声明指示 DELETE 操作失败，并附有发生该问题的表和索引的详细信息。 服务器状态：联机 用户操作：对指定的表和索引，或数据库运行 oncheck -cI 命令，并纠正找到的任何错误。再尝试原操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 2 事件 ID： 2002	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error was raised due to an inconsistency in the index which is preventing the database server to position on the first record in that index.	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。

ID	严重性	消息	解释
类 ID： 2 事件 ID： 2003	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error was raised due to an inconsistency in the index which is preventing the database server to read ahead pages in that index.	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2004	4	类消息： Index failure: *'dbname":owne"r.tabname-idxname'* 特定消息： Page Check Error in *object*	数据库服务器检测到不一致的索引。 联机日志：各种消息，取决于在哪里检测到该问题。例如：Possible inconsistencies in a DBSpace TBLSpace Run 'oncheck -cD' on all DBSpace TBLSpaces 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并在数据库上运行建议的 oncheck -cD 命令。
类 ID： 2 事件 ID： 2005	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error occurred during batched index read because the database server had an invalid index key item.	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2006	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： *index_page* log record too large to fit into the logical log buffer. Recommended minimum value for LOGBUFF is *number*.	服务器检测到索引页的日志记录对于配置的逻辑日志缓冲区大小而言太大。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并将 onconfig 的 LOGBUFF 的值更新为建议的值。
类 ID： 2 事件 ID： 2007	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： Comparison based on locale '*locale_name*' failed	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。

ID	严重性	消息	解释
类 ID： 2 事件 ID： 2008	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： Comparison failed	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2009	4	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error occurred while the database server was trying to add a new item to the index.	数据库服务器无法向索引插入记录。 联机日志：声明指定的索引和建议运行的 oncheck 命令。 服务器状态：联机 用户操作：检查联机日志文件并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2010	4	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error was raised due to an inconsistency in the index which is preventing the database server to position at the correct item in the index.	数据库服务器由于索引中的不一致而无法检索该索引中正确的项 联机日志：声明指定的索引和建议运行的 oncheck 命令。 服务器状态：联机 用户操作：检查联机日志文件并运行建议的 oncheck 命令。
类 ID： 2 事件 ID： 2011	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： Cannot drop index *index_name* for partnum *partition_number*, iserrno = *error_number*	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，并运行建议的 oncheck 命令。再尝试原操作。如果该操作再次失败，则联系星瑞格®软件支持。
类 ID： 2 事件 ID： 2012	3	类消息： Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error occurred while the database server was trying to mark an index key descriptor as bad.	数据库服务器检测到不一致的索引，并将其标记为无法使用。 联机日志：断言警告，带有所遇到的错误的详细信息和涉及的 *database:table*。 服务器状态：联机 用户操作：查看 online.log 文件以获取更多信息，运行建议的 oncheck 命令，修复检测到的任何问题，然后重新启用该索引。
类 ID：	4	类消息：	数据库服务器无法从索引中删除记录。

ID	严重性	消息	解释
2 事件 ID： 2013		Index failure: *'dbname":owne"r.tabname:idxname'* 特定消息： An internal error occurred while the database server was trying to delete an item from the index.	联机日志：声明指定的索引和建议运行的 oncheck 命令。 服务器状态：联机 用户操作：检查联机日志文件并运行建议的 oncheck 命令。
类 ID： 3 事件 ID： 3001	3	类消息： Blob failure: *'dbname":owne"r.tabname'* 特定消息： tb_sockid in blob descriptor is corrupted. Current® table is *'dbname":owne"r.tabname'*	
类 ID： 3 事件 ID： 3002	3	类消息： Blob failure: *'dbname":owne"r.tabname'* 特定消息： Incorrect BLOB stamps.	
类 ID： 3 事件 ID： 3003	4	类消息： Blob failure: *'dbname":owne"r.tabname'* 特定消息： BLOB Page Check error at *dbspace_name*	数据库服务器检查在磁盘和内存间移动的页时，发生失败。 联机日志：声明描述该错误。 服务器状态：联机 用户操作：再尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 3 事件 ID： 3004	3	类消息： Blob failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while trying to read a blob from a table.	
类 ID： 3 事件 ID： 3005	3	类消息： Blob failure: *'dbname":owne"r.tabname'* 特定消息： An internal error occurred while trying to copy a blob from a table.	
类 ID： 4 事件 ID：	4	类消息： Chunk is offline, mirror is active: *chunk_number* 特定消息：	从块读取或写入块时发生错误。数据库服务器使该块脱机并切换到活动的镜像块执行所有 I/O 操作。 联机日志：声明描述该发生的错误。 服务器状态：联机

ID	严重性	消息	解释
4001		I/O error, *error_number* Chunk '*chunk_number*' -- Offline	用户操作：检查联机日志以获取信息并修复该错误。运行 onspaces -s 命令以恢复该脱机的块。再尝试原操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 4 事件 ID： 4002	3 or 4	类消息： Chunk is offline, mirror is active: *chunk_number* 特定消息： An internal error occurred during physical I/O because the chunk was not opened.	数据库服务器无法访问块，并切换到活动的镜像块执行所有 I/O 操作。 联机日志：声明描述该错误和关于发生问题的块的信息。 服务器状态：联机 用户操作：检查联机日志，修复任何错误，并使用 onspaces 实用程序恢复镜像。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 4 事件 ID： 4003	3	类消息： Chunk is offline, mirror is active: *chunk_number* 特定消息： I/O error, *error_number* Chunk '*chunk_number*' -- Offline (sanity)	
类 ID： 4 事件 ID： 4004	3	类消息： Chunk is offline, mirror is active: *chunk_number* 特定消息： Chunk failed sanity check	
类 ID： 4 事件 ID： 4005	3	类消息： Chunk is offline, mirror is active: *chunk_number* 特定消息： Mirror Chunk *chunk_number* added to space '*space_number*'. Perform manual recovery.	
类 ID： 5 事件 ID： 5001	4	类消息： Dbspace is offline: *'dbspace_name'* 特定消息： Chunk *chunk_number* is being taken OFFLINE.	由于访问块发生错误，数据库服务器使数据库空间脱机。 联机日志：如果换数据库空间是重要的数据库空间（例如：rootdbs），那么断言失败。如果数据库空间不重要，那么断言警告。两者都提供于脱机的块和数据库空间的信息。 服务器状态：如果是非重要介质故障，则服务器处于联机。如果是重要介质故障，则服务器处于脱机。

ID	严重性	消息	解释
			用户操作：检查联机日志文件，并修复导致数据库空间脱机的问题。您可能需要恢复数据库空间。
类 ID： 5 事件 ID： 5002	4	类消息： Dbspace is offline: *'dbspace_name'* 特定消息： WARNING! Chunk *chunk_number* is being taken OFFLINE for testing.	onmode 命令导致数据库服务器脱机。 联机日志：断言警告指示数据库空间已脱机 。 服务器状态：联机 用户操作：无
类 ID： 6 事件 ID： 6016	3	类消息： Internal subsystem failure: *'message'* 特定消息： Pool not freed. pool name:*pool_name*, address:*address*	
类 ID： 6 事件 ID： 6017	4	类消息： Internal subsystem failure: *'message'* 特定消息： CDR Grouper FanOut thread is aborting	Enterprise Replication 发生了问题。 联机日志：声明描述该问题。 服务器状态：联机 用户操作：遵循联机日志中的指示操作。
类 ID： 6 事件 ID： 6018	4	类消息： Internal subsystem failure: *'message'* 特定消息： CDR Pager: Paging File full: Waiting for additional space in CDR_QDATA_SBSPACE	Enterprise Replication 队列存储空间已满。 联机日志：声明描述该问题。 服务器状态：联机 用户操作：向 CDR_QDATA_SBSPACE 配置参数指定的一个或多个智能大对象空间添加块。
类 ID： 6 事件 ID： 6021	3	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was reported by the database server during conversion when it found some indices in the old format.	
类 ID： 6 事件 ID： 6022	3	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was reported by the database server when it checks for any new in-place alter pending in the current server during reversion.	

ID	严重性	消息	解释
类 ID： 6 事件 ID： 6023	3	类消息： Internal subsystem failure: *'message'* 特定消息： Cannot open index '*dbname:index_name*', iserrno = *error_number*	
类 ID： 6 事件 ID： 6024	3	类消息： Internal subsystem failure: *'message'* 特定消息： Cannot drop index '*dbname:index_name*', iserrno = *error_number*	
类 ID： 6 事件 ID： 6025	3	类消息： Internal subsystem failure: *'message'* 特定消息： Cannot open table '*dbname:table_name*', iserrno = *error_number*	
类 ID： 6 事件 ID： 6026	3	类消息： Internal subsystem failure: *'message'* 特定消息： Cannot drop table '*dbname:table_name*', iserrno = *error_number*	
类 ID： 6 事件 ID： 6027	3	类消息： Internal subsystem failure: *'message'* 特定消息： An error was reported by the database server when it tried to drop the sysmaster database during reversion	
类 ID： 6 事件 ID： 6030	3	类消息： Internal subsystem failure: *'message'* 特定消息： Invalid or missing name for Subsystem Staging BLOBspace	
类 ID： 6	5	类消息： Internal subsystem failure: *'message'* 特定消息：	数据库服务器在读取内部高速缓存遇到错误之后关闭。 联机日志：断言失败

ID	严重性	消息	解释
事件 ID： 6033		Cache read error	服务器状态：脱机 用户操作：启动数据库服务器并再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6034	3	类消息： Internal subsystem failure: *'message'* 特定消息： Could not start remote server	
类 ID： 6 事件 ID： 6035	3	类消息： Internal subsystem failure: *'message'* 特定消息： An error was reported by the database server during the handling of audit trail files.	
类 ID： 6 事件 ID： 6036	3	类消息： Internal subsystem failure: *'message'* 特定消息： Archive on *dbspaces_list* ABORTED	
类 ID： 6 事件 ID： 6037	3	类消息： Internal subsystem failure: *'message'* 特定消息： Waiting on BLOBSpace to appear for Logical Recovery	
类 ID： 6 事件 ID： 6038	3	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error reported by the database server. Users may need to look at the specific message which accompanies with this id.	
类 ID： 6 事件 ID： 6039	3	类消息： Internal subsystem failure: *'message'* 特定消息： Wrong page for cleaning deleted items	
类 ID：	3	类消息：	

ID	严重性	消息	解释
6 事件 ID： 6040		Internal subsystem failure: *'message'* 特定消息： Buffer in wrong state for cleaning deleted items	
类 ID： 6 事件 ID： 6041	5 or 3	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was detected by the Buffer Manager in the database server.	对于严重性 5，数据库服务器缓冲区管理器遇到内部错误，则关闭或纠正该问题。 联机日志：断言警告或断言失败，带有错误发生时执行的操作的描述。通常，断言警告显示该错误已内部纠正。 服务器状态：如果错误不可恢复，则处于脱机。如果错误已纠正，则处于联机。 用户操作：如果错误不可恢复，则启动数据库服务器并再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。如果该错误已被数据库服务器内部纠正，则不需采取任何操作。
类 ID： 6 事件 ID： 6042	5 or 2	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was reported by the database server when it detected an inconsistency with the internal buffer queues.	对于严重性 5，数据库服务器在内部缓冲队列处理过程中检测到不一致，则关闭或纠正该问题。 联机日志：断言警告或断言失败，带有错误发生时执行的操作的描述。通常，断言警告显示该错误已内部纠正。 服务器状态：如果错误不可恢复，则处于脱机。如果错误已纠正，则处于联机。 用户操作：如果错误不可恢复，则启动数据库服务器并再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。如果该错误已被数据库服务器内部纠正，则不需采取任何操作。
类 ID： 6 事件 ID： 6043	3	类消息： Internal subsystem failure: *'message'* 特定消息： Internal file error	
类 ID： 6 事件 ID： 6044	3	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was corrected automatically by the database server when it tried to save the log buffer into a system log buffer.	
类 ID：	5	类消息：	数据库服务器由于处理逻辑日志时发生的错误而关闭。

ID	严重性	消息	解释
6 事件 ID： 6045		Internal subsystem failure: *'message'* 特定消息： Logical logging error for '*object*' in *'space'*	联机日志：断言失败，带有该问题的描述和逻辑日志信息。 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6046	4	类消息： Internal subsystem failure: *'message'* 特定消息： Page Check Error in *object*	数据库服务器检测到数据的不一致。 联机日志：各种输出，取决于检测到问题的位置。例如：Possible inconsistencies in a DBSpace TBLSpace Run 'oncheck -cD' on all DBSpace TBLSpaces 服务器状态：Online. 用户操作：检查联机日志文件并在数据库上运行建议的 oncheck -cD 命令。
类 ID： 6 事件 ID： 6047	3	类消息： Internal subsystem failure: *'message'* 特定消息： Errors occurred while recreating indexes	
类 ID： 6 事件 ID： 6049	5	类消息： Internal subsystem failure: *'message'* 特定消息： Lock types *lock_type* and *lock_type* should never be merged	数据库服务器在试图合并不兼容的锁后关闭。 联机日志：断言失败，带有数据库服务器试图合并的锁类型。 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6050	5	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was reported by the database server when it detected some corruption in the lock free list chain.	数据库服务器在检测到用于管理可用锁内部列表的内部结构损坏后关闭。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6051	3	类消息： Internal subsystem failure: *'message'* 特定消息： ERROR - NO 'waitfor' locks in Critical Section!!!	

ID	严重性	消息	解释
类 ID： 6 事件 ID： 6052	3	类消息： Internal subsystem failure: *'message'* 特定消息： Internal Tblspace error	
类 ID： 6 事件 ID： 6053	3	类消息： Internal subsystem failure: *'message'* 特定消息： Session does not have exclusive access to partition *partition_name*. Request to drop the partition ignored.	
类 ID： 6 事件 ID： 6054	3	类消息： Internal subsystem failure: *'message'* 特定消息： Error building 'sysmaster' database.	
类 ID： 6 事件 ID： 6055	3	类消息： Internal subsystem failure: *'message'* 特定消息： Setread error on SMI Table, partnum *partition_number*	
类 ID： 6 事件 ID： 6056	3	类消息： Internal subsystem failure: *'message'* 特定消息： Comparison based on locale '*locale_name*' failed	
类 ID： 6 事件 ID： 6057	2	类消息： Internal subsystem failure: *'message'* 特定消息： DBSPACETEMP internal list not initialized, using default	数据库服务器不创建保存 DBSPACETEMP 信息的所必需的结构。 联机日志：消息显示该内部 DBSPACETEMP 列表未初始化。 服务器状态：联机 用户操作：无
类 ID： 6 事件 ID：	3	类消息： Internal subsystem failure: *'message'* 特定消息：	

ID	严重性	消息	解释
6058		A data source accessed using a gateway (*gateway_name*) might be in an inconsistent state	
类 ID: 6 事件 ID: 6059	3	类消息: Internal subsystem failure: *'message'* 特定消息: Prepared participant site *site_name* not responding	
类 ID: 6 事件 ID: 6060	5	类消息: Internal subsystem failure: *'message'* 特定消息: Thread exited with *number* buffers held	数据库服务器检测到一个线程持有一个或多个缓冲区后关闭。 联机日志：断言失败，带有该线程持有的缓冲区数。 服务器状态：脱机 用户操作：使数据库服务器处于联机。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6061	3	类消息: Internal subsystem failure: *'message'* 特定消息: An internal error was automatically corrected by the database server when it detected that the undo log for the transaction was not applicable.	
类 ID: 6 事件 ID: 6062	3	类消息: Internal subsystem failure: *'message'* 特定消息: Internal Error - Freeing transaction entry that still holds locks!	当数据库服务器释放与事务相关的资源时检测到该事务正持有锁。在多数情况下，数据库服务器会释放这些锁。 联机日志：声明该事务的警告，并声明数据库服务器已内部纠正该问题。 服务器状态：联机 用户操作：如果数据库服务器关闭，则启动数据库服务器。
类 ID: 6 事件 ID: 6063	3	类消息: Internal subsystem failure: *'message'* 特定消息: User thread not on TX wait list	
类 ID: 6 事件 ID:	3	类消息: Internal subsystem failure: *'message'* 特定消息:	

ID	严重性	消息	解释
6064		Due to a heuristic decision, the work done on behalf of the specified transaction branch might have been heuristically completed or committed or rolled back or partially committed and partially rolled back.	
类 ID: 6 事件 ID: 6065	3	类消息: Internal subsystem failure: *'message'* 特定消息: Errors occurred while recreating indexes	
类 ID: 6 事件 ID: 6066	3	类消息: Internal subsystem failure: *'message'* 特定消息: An internal error is reported by the database server when it has checked all sites to see if a heuristic rollback was the reason for the failure.	
类 ID: 6 事件 ID: 6067	5	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Recursive exception) has caused the database server processes to terminate unexpectedly.	数据库服务器检测到异常处理的递归调用，则立即关闭以避免无限循环。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6068	5	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Internal exception) has caused the database server processes to terminate unexpectedly.	数据库服务器由于不可恢复的内部错误而关闭。 联机日志：断言失败，带有关于导致问题的异常的信息。 服务器状态：脱机 用户操作：启动数据库服务器。查看断言失败文件中的该异常信息。如果该异常与用户定义例程有关，则检查并纠正用户定义例程。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6069	5	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Master daemon died) has caused the database server processes to terminate unexpectedly.	主守护 oninit 进程停止且数据库服务器关闭。该错误可能由操作系统进程终止所导致。 联机日志：断言失败 服务器状态：脱机

ID	严重性	消息	解释
			用户操作：启动数据库服务器。终止操作系统进程时，请谨慎。
类 ID： 6 事件 ID： 6070	5	类消息： Internal subsystem failure: *'message'* 特定消息： A fatal internal error (VP died) has caused the database server processes to terminate unexpectedly.	oninit 进程停止且数据库服务器关闭。该错误可能由操作系统进程终止所导致。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。终止操作系统进程时，请谨慎。
类 ID： 6 事件 ID： 6071	5	类消息： Internal subsystem failure: *'message'* 特定消息： ERROR: can not fork secondary Server thread (MACH11 Shutdown)	辅助服务器关闭，但无法创建线程来正常关闭。 联机日志： DR: Shutting down the server. ERROR: can not fork secondary Server thread (MACH11 Shutdown) Can not run onmode -ky PANIC: Attempting to bring system down. 服务器状态：脱机 用户操作：无
类 ID： 6 事件 ID： 6072	3	类消息： Internal subsystem failure: *'message'* 特定消息： Generic unique event id when the server failed to fork a new thread.	
类 ID： 6 事件 ID： 6073	3	类消息： Internal subsystem failure: *'message'* 特定消息： An error was reported by the database server when it could not initialize GLS for starting a session.	
类 ID： 6 事件 ID： 6074	3	类消息： Internal subsystem failure: *'message'* 特定消息： WARNING: mt_aio_wait: errno == EINVAL	
类 ID： 6 事件 ID： 6075	5	类消息： Internal subsystem failure: *'message'* 特定消息： A fatal internal error (KAIO) has caused the database server processes to terminate unexpectedly.	数据库服务器由于 KAIO 子系统中的错误而关闭。 联机日志：断言失败，带有该失败的特定操作。 服务器状态：脱机

ID	严重性	消息	解释
			用户操作：启动数据库服务器。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6100		一般事件，当数据库服务器隐式地提出断言警告。	发生一般内部错误。 联机日志：断言警告，带有该问题的详细信息。 服务器状态：联机 用户操作：查看联机日志并采取建议的纠正操作。数据库服务器可能自动地纠正该问题。再尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6300		一般事件，当数据库服务器隐式地提出断言失败。	发生一般内部错误。 联机日志：断言警告，带有该问题的详细信息。 服务器状态：联机 用户操作：查看联机日志并采取建议的纠正操作。再尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6500		一般事件，当数据库服务器由于内部错误情况而意外地终止。	发生内部错误且数据库服务器关闭。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。检查断言失败文件以获取更多信息。如果可能的话，修复任何标识的问题，并再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格® 软件支持。
类 ID： 7 事件 ID： 7001	3	类消息： Database server initialization failure 特定消息： TABLOCKS log record too large to fit into the logical log buffer. Recommended minimum value for LOGBUFF is *size*. I-STAR(C) begins prepare log record too large to fit into the logical log buffer. Recommended minimum value for LOGBUFF is *size*. Partition blob log record too large to fit into the logical log buffer. Recommended minimum value for LOGBUFF is *size*.	

ID	严重性	消息	解释
		Alter table special column desc log record too large to fit into the logical log buffer. Recommended minimum value for LOGBUFF is *size*.	
类 ID: 7 事件 ID: 7002	4	类消息: Database server initialization failure 特定消息: Unable to extend *number* reserved pages for checkpoint in ROOT chunk. Unable to extend *number* reserved pages for log in ROOT chunk.	数据库服务器由于无法分配更多空间给初始根块中的内部结构而无法启动。 联机日志：断言 服务器状态：脱机 用户操作：再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格® 软件支持。
类 ID: 7 事件 ID: 7003	4	类消息: Database server initialization failure 特定消息: An internal error occurred during conversion. Users may need to take a look at the specific messages for further action.	数据库服务器在升级过程中无法启动，因为在转换过程中发生内部错误。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：查看联机日志和特定消息，并采取必需的纠正操作。再次尝试该操作。如果该操作再次失败，则记录所有情况并联系星瑞格® 软件支持。
类 ID: 7 事件 ID: 7004	4	类消息: Database server initialization failure 特定消息: An internal error occurred while trying to convert the database tblspace.	数据库服务器由于尝试转换数据库表空间时发生内部错误而无法启动。该表空间保存了关于实例中数据库的信息。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：请联系星瑞格® 软件支持。
类 ID: 7 事件 ID: 7005	4	类消息: Database server initialization failure 特定消息: An internal error occurred while trying to convert blob free map pages.	数据库服务器由于尝试转换 Blob 空间可用映射页时发生内部错误而无法启动。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：请联系星瑞格® 软件支持。
类 ID: 7 事件 ID: 7006	4	类消息: Database server initialization failure 特定消息: Cannot Open Logical Log.	数据库服务器无法启动，因为它仍在恢复物理日志或逻辑日志。如果在恢复完成之前就运行了 onmode -m 或 onmode -s 命令，那么就会发生这种情况。 联机日志：声明描述该错误。 服务器状态：恢复并启动。 用户操作：在恢复完成后运行 onmode -m 或 onmode -s 命令。

ID	严重性	消息	解释
类 ID： 7 事件 ID： 7007	4	类消息： Database server initialization failure 特定消息： Logical Log File not found.	数据库服务器由于逻辑日志文件丢失而无法启动。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：从备份中恢复数据库服务器。
类 ID： 7 事件 ID： 7008	3	类消息： Database server initialization failure 特定消息： WARNING! LTXHWM is set to 100%. This long transaction high water mark will never be reached. Transactions will not be aborted automatically by the server, regardless of their length.	
类 ID： 7 事件 ID： 7009	4	类消息： Database server initialization failure 特定消息： A Physical or Logical Restore is active.	数据库服务器无法启动，因为它仍在恢复物理日志或逻辑日志。如果在恢复完成之前就运行了 onmode -m 或 onmode -s 命令，那么就会发生这种情况。 联机日志：声明描述该错误。 服务器状态：恢复并启动。 用户操作：在恢复完成后运行 onmode -m 或 onmode -s 命令。
类 ID： 7 事件 ID： 7010	4	类消息： Database server initialization failure 特定消息： *root_dbspace* has not been physically recovered.	数据库服务器无法启动，因为在 rootdbs 物理恢复之前中断了恢复。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：恢复 rootdbs。
类 ID： 7 事件 ID： 7011	4	类消息： Database server initialization failure 特定消息： *dbspace* has not been physically recovered.	数据库服务器无法启动，因为数据库空间未进行物理复原。如果数据库服务器尝试在复原完成之前启动，就会发生这种情况。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：启动数据库服务器之前等待直到复原完成。
类 ID： 7 事件 ID：	4	类消息： Database server initialization failure 特定消息：	数据库服务器由于数据库空间还原失败而无法启动。 联机日志：声明描述该错误。 服务器状态：脱机

ID	严重性	消息	解释
7012		*dbspace* not recovered from same archive backup as *dbspace*.	用户操作：从备份中恢复数据库空间并前滚必需的日志，以使数据库空间回到正确的时间点。
类 ID： 7 事件 ID： 7013	4	类消息： Database server initialization failure 特定消息： Log *log_number* not found.	数据库服务器无法启动，因为复原未完成。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：启动数据库服务器之前等待直到复原完成。
类 ID： 7 事件 ID： 7014	4	类消息： Database server initialization failure 特定消息： Logical restore cannot be skipped. Perform a logical restore.	数据库服务器无法启动，因为复原未完成。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：执行逻辑复原（例如：通过使用 onbar -r -l 命令）并以静默或联机模式启动数据库服务器。
类 ID： 7 事件 ID： 7015	4	类消息： Database server initialization failure 特定消息： Cannot change to On-Line or Quiescent mode.	数据库服务器无法启动，因为在快速恢复或完整恢复过程中发生错误。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：检查联机日志以获取更多信息。再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7016	4	类消息： Database server initialization failure 特定消息： Cannot Open Primary Chunk '*chunk_number*'.	数据库服务器无法启动，因为其无法访问主块。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：检查联机日志以获取更多信息。再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7017	4	类消息： Database server initialization failure 特定消息： The chunk '*chunk_number*' must have owner-ID "owner_id" and group-ID "*group_id*".	数据库服务器无法启动，因为块路径的所有者和群组不正确。 联机日志：声明描述该问题。 服务器状态：脱机 用户操作：纠正特定消息中提到的块路径的权限。 再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID：	4	类消息：	数据库服务器无法启动，因为块路径的权限不正确。

ID	严重性	消息	解释
7 事件 ID： 7018		Database server initialization failure 特定消息： The chunk '*chunk_number*' must have READ/WRITE permissions for owner and group (660).	联机日志：声明描述该问题。 服务器状态：脱机 用户操作：纠正特定消息中提到的块路径的权限。 再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7019	4	类消息： Database server initialization failure 特定消息： Memory allocation error.	数据库服务器无法启动，因为其无法分配足够的内存。 联机日志： 服务器状态：脱机 用户操作：请确保有足够可用的内存可用于您指定的数据库服务器的配置。再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7020	4	类消息： Database server initialization failure 特定消息： The chunk '*chunk_number*' will not fit in the space specified.	数据库服务器无法启动，因为没有足够空间来创建指定的块。 联机日志：声明描述该问题。 服务器状态：脱机 用户操作：指定较小的块大小或释放额外空间给块。
类 ID： 7 事件 ID： 7021	4	类消息： Database server initialization failure 特定消息： *device_name*: write failed, file system is full.	数据库服务器无法启动，因为文件系统没有可用空间。 联机日志：声明描述该问题。 服务器状态：脱机 用户操作：请确保特定消息中提到的文件系统有足够空间。重试原操作。 如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7022	3	类消息： Database server initialization failure 特定消息： An error occurred while the database server was creating the SMI database.	
类 ID： 7 事件 ID： 7023	4	类消息： Database server initialization failure 特定消息： Unable to create boot strap config file - '*file_name*'	数据库服务器由于无法创建配置文件而无法启动。 联机日志：声明描述该错误。 服务器状态：脱机 用户操作：检查联机日志以获取更多信息并修复该问题。该问题可能是目录的权限不正确。重试原操作。 如果该操作再次失

ID	严重性	消息	解释
			败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 7 事件 ID： 7024	3	类消息： Database server initialization failure 特定消息： 'sysmaster' database will not be built/checked	
类 ID： 7 事件 ID： 7025	3	类消息： Database server initialization failure 特定消息： WARNING! Physical Log size *size* is too small. Physical Log overflows may occur during peak activity. Recommended minimum Physical Log size is *number* times maximum concurrent user threads.	
类 ID： 7 事件 ID： 7026	3	类消息： Database server initialization failure 特定消息： WARNING! Logical log layout may cause __ISN__ to get into a locked state. Recommended smallest logical log size is *number* times maximum concurrent user threads.	
类 ID： 7 事件 ID： 7027	3	类消息： Database server initialization failure 特定消息： WARNING! Buffer pool size may cause __ISN__ to get into a locked state. Recommended minimum buffer pool size is *number* times maximum concurrent user threads.	
类 ID： 7 事件 ID： 7028	3	类消息： Database server initialization failure 特定消息： Checkpoint log record may not fit into the logical log buffer.	

ID	严重性	消息	解释
		Recommended minimum value for LOGBUFF is *size*.	
类 ID: 7 事件 ID: 7029	3	类消息: Database server initialization failure 特定消息: Temp transaction not NULL.	
类 ID: 9 事件 ID: 9001	4	类消息: Physical recovery failure 特定消息: Physical log recovery error	数据库服务器的物理恢复失败。 联机日志：断言失败，带有该问题的描述。 服务器状态：联机 用户操作：重试该操作或从备份中复原。
类 ID: 10 事件 ID: 10001	3 or 4	类消息: Logical recovery failure 特定消息: Rollback error *error_number*	逻辑恢复失败，因为数据库服务器无法回滚事务。 联机日志：声明该错误的详细信息，以及该问题发生的日志或日志记录。 服务器状态：联机或脱机，取决于该错误。 用户操作：检查联机日志文件以获取更多信息，并运行建议的命令，例如：oncheck 命令。重试原操作。 如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 10 事件 ID: 10002	4	类消息: Logical recovery failure 特定消息: Logical Recovery ABORTED.	数据库服务器的逻辑恢复失败。 联机日志：断言警告，带有关于日志记录的信息。如果该失败与重要数据库空间相关，则断言失败带有关于该日志记录的信息。 服务器状态：如果不是重要的数据库空间，则服务器处于联机。如果是重要的数据库空间，则服务器处于脱机。 用户操作：检查联机日志以确认合适的操作，例如：您可能需要重新启动热复原。
类 ID: 10 事件 ID: 10003	4	类消息: Logical recovery failure 特定消息: Log record (*log_subsystem*:*log_type*) in log *log_number*, offset *log_position* was not rolled back	逻辑恢复在回滚事务时遇到内部错误。 联机日志：消息描述该日志记录。 服务器状态：联机 用户操作：检查联机日志并确认合适的操作，例如：重新提交该事务。
类 ID:	3	类消息: Logical recovery failure	

ID	严重性	消息	解释
10 事件 ID： 10004		特定消息： Logical Logging error for '*log_subsystem*:*log_type*' in '*object*'	
类 ID： 10 事件 ID： 10005	4	类消息： Logical recovery failure 特定消息： An internal error occurred while trying to apply the log records during logical log recovery.	逻辑恢复失败。 联机日志：断言警告，带有关于该日志记录的信息。 服务器状态：取决于该失败。 用户操作：检查联机日志，并确认合适的操作，liru：重新启动热复原。
类 ID： 10 事件 ID： 10006	3 or 4	类消息： Logical recovery failure 特定消息： An internal error occurred when the database server tried to find the file descriptor for the tblspace.	逻辑恢复失败，因为数据库服务器无法找到分区的内部文件描述符。 联机日志：声明指示发生错误的表和运行 oncheck 命令的指示操作。 服务器状态：联机 用户操作：对联机日志提到的表或对数据库运行 oncheck -cDI 命令。
类 ID： 11 事件 ID： 11001	3	类消息： Cannot open chunk: *'pathname'* 特定消息： Cannot Open Mirror Chunk '*chunk_number*', errno = *error_number*	
类 ID： 11 事件 ID： 11002	3	类消息： Cannot open chunk: *'pathname'* 特定消息： Cannot Open Primary Chunk '*chunk_number*', errno = *error_number*	
类 ID： 12 事件 ID： 12001	3	类消息： Cannot open dbspace: *'dbspace_name'* 特定消息： ERROR: DBspace *dbspace_name* not found among table *table_name* fragments.	
类 ID： 13 事件 ID： 13001	2	类消息： Performance improvement possible 特定消息： The number of configured CPU poll threads exceeds number of CPU VPs specified in 'VPCLASS cpu'. NETTYPE	数据库服务器检测到 CPU 虚拟处理器数与服务器初始化期间请求的 CPU 轮询线程数不匹配。 联机日志：关于配置不匹配的性能警告。数据库服务器将使用 NET 虚拟处理器。 服务器状态：联机

ID	严重性	消息	解释
		'*protocol*' poll threads started on NET VPs.	用户操作：检查服务器的配置。
类 ID： 13 事件 ID： 13002	2	类消息： Performance improvement possible 特定消息： Transaction table overflow due to parallel recovery.	内部结构不够大，无法处理逻辑日志。数据库服务器将推迟日志处理直到该结构中出现了更多的空间。 联机日志：警告消息指示事务处理推迟。 服务器状态：联机 用户操作：无
类 ID： 14 事件 ID： 14001	3	类消息： Database failure. *'dbname'* 特定消息： '*'dbname'*' - Error *error_number* during logging mode change.	
类 ID： 15 事件 ID： 15001	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Turned off on secondary server	
类 ID： 15 事件 ID： 15002	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Turned off on primary server	
类 ID： 15 事件 ID： 15003	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Cannot connect to secondary server	
类 ID： 15 事件 ID： 15004	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Received connection request from remote server when DR is not Off [Local type: *type*, Current® state: *state*] [Remote type: *type*]	

ID	严重性	消息	解释
类 ID： 15 事件 ID： 15005	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Received connection request before physical recovery completed.	
类 ID： 15 事件 ID： 15006	3	类消息： High-Availability Data-Replication failure 特定消息： DR: Local and Remote server type and/or last change (LC) incompatible [Local type: *type*, LC: *type*] [Remote type: *type*, LC: *type*]	
类 ID： 16 事件 ID： 16001	2	类消息： Backup completed: *'dbspace_list'* 特定消息： Archive on *dbspace_list* completed without being recorded.	归档完成，但在归档过程中服务器检测到损坏的页。 联机日志：消息指示备份完成但检测到损坏的页。 服务器状态：联机 用户操作：不使用这个备份。使用较早的备份来立即复原损坏的块。
类 ID： 16 事件 ID： 16002	2	类消息： Backup completed: *'dbspace_list'* 特定消息： Archive on *dbspace_list* Completed with *number* corrupted pages detected.	归档完成，但在归档过程中服务器检测到损坏的页。 联机日志：消息指示备份完成但检测到损坏的页。 服务器状态：联机 用户操作：不使用这个备份。使用 0 损坏页的较早备份来立即复原损坏的块。
类 ID： 16 事件 ID： 16003	2	类消息： Backup completed: *'dbspace_list'* 特定消息： Archive on *dbspace_list* Completed	完成所列的数据库空间的归档。 联机日志：消息指示完成所列的数据库空间的归档。 服务器状态：联机 用户操作：无
类 ID： 17 事件 ID： 17001	4	类消息： Backup aborted: *'dbspace_list'* 特定消息： Archive detects that page *chunk_number*:*page_offset* is corrupt.	数据库服务器检测到损坏并停止备份。 联机日志：声明描述该问题。 服务器状态：联机 用户操作：检查联机日志以获取关于损坏的信息。再尝试该操作。如果该操作再次

ID	严重性	消息	解释
			失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 17 事件 ID： 17002	3	类消息： Backup aborted: *'dbspace_list'* 特定消息： Page %d:%d of partition *partition_number* not archived.	
类 ID： 18 事件 ID： 18001	2	类消息： Log backup completed: *log_number* 特定消息： Logical Log *log_number* - Backup Completed	逻辑日志已备份。 联机日志：消息标识备份的逻辑日志的日志编号。 服务器状态：联机 用户操作：无
类 ID： 19 事件 ID： 19001	3	类消息： Log backup aborted: *log_number* 特定消息： Logical Log *log_number* - Backup Aborted *message*	
类 ID： 20 事件 ID： 20001	3	类消息： Logical logs are full—backup is needed 特定消息： Logical Log Files are Full -- Backup is Needed	
类 ID： 20 事件 ID： 20002	3	类消息： Logical logs are full—backup is needed 特定消息： Waiting for Next Logical Log File to be Freed	
类 ID： 20 事件 ID： 20003	3	类消息： Logical logs are full—backup is needed 特定消息： Logical Log Files are almost Full -- Backup is Needed. In Data replication scenario, this could block failure-recovery of the paired server.	

ID	严重性	消息	解释
类 ID： 21 事件 ID： 21001	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Archive arcbu_next_tbuf() - Buffer Overflow	
类 ID： 21 事件 ID： 21002	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Archive tcp_logbu_hdr() - Buffer Overflow	
类 ID： 21 事件 ID： 21003	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Archive tcp_logbu_trl() - Buffer Overflow	
类 ID： 21 事件 ID： 21004	2 or 5	类消息： Database server resource overflow: *'resource_name'* 特定消息： Physical log file overflow	对于严重性 5，物理日志文件已满并需要溢出。如果在恢复过程中发生这个情况，则数据库服务器会尝试扩展物理日志。 联机日志：如果数据库服务器不在恢复状态或无法扩展物理日志，那么断言失败。如果数据库服务器在恢复状态并且扩展物理日志，那么断言警告。 服务器状态：脱机 用户操作：无
类 ID： 21 事件 ID： 21005	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Lock table overflow - user id %d, session id %d	
类 ID： 21 事件 ID： 21006	5	类消息： Database server resource overflow: *'resource_name'* 特定消息： Logical log buffer overflow detected	数据库服务器关闭，因为逻辑日志缓冲区已满。 联机日志：断言失败，带有日志记录大小和缓冲区大小。 服务器状态：脱机

ID	严重性	消息	解释
			用户操作：增加 onconfig 文件中 LOGBUFF 配置参数的值。启动数据库服务器。
类 ID： 21 事件 ID： 21007	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Llog logbu_logfile() - Buffer Overflow	
类 ID： 21 事件 ID： 21008	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Llog logbu_bpage() - Buffer Overflow	
类 ID： 21 事件 ID： 21009	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Unable to allocate a user thread for user id *user_ID*	
类 ID： 21 事件 ID： 21010	3	类消息： Database server resource overflow: *'resource_name'* 特定消息： Unable to allocate a transaction for user id *user_ID*, session id *session_ID*	
类 ID： 21 事件 ID： 21014	3	类消息： OnLine resource overflow: Locks; 特定消息： SID='*session id*' User ID='*uid*' NAME='*user name*' PID='*process id*'	已达到该会话中的锁限制。终止当前事务。 联机日志：Session *session id* has exceeded the session limit of *number* locks. 服务器状态：联机 用户操作：减少您的事务大小。
类 ID： 21 事件 ID： 21016	3	类消息： OnLine resource overflow: Memory; 特定消息： Session info: '*session id username@hostname*' pid '*pid*'	已达到该会话中的内存限制。终止当前会话。 联机日志：Session *session id* has exceeded the session limit of *size* of memory. 服务器状态：联机

ID	严重性	消息	解释
			用户操作：减少会话内存消耗。
类 ID： 21 事件 ID： 21017	3	类消息： OnLine resource overflow: Tempspace; 特定消息： Session info: '*session id username@hostname*' pid '*pid*'	已达到该会话中的临时空间限制。终止当前会话。 联机日志：Session *session id* has exceeded the session limit of *size* of temporary space. 服务器状态：联机 用户操作：减少您的数据大小。
类 ID： 21 事件 ID： 21018	3	类消息： OnLine resource overflow: Log Space; 特定消息： Session info: '*session id username@hostname*' pid '*pid*'	已达到该会话中的日志空间限制。终止当前事务。 联机日志：Session *session id* has exceeded the session limit of *size* of log space. 服务器状态：联机 用户操作：减少您的事务大小。
类 ID： 21 事件 ID： 21019	3	类消息： OnLine resource overflow: Transaction Time; 特定消息： Session info: '*session id username@hostname*' pid '*pid*'	已达到该会话中的事务时间限制。终止当前会话。 联机日志：Session *session id* has exceeded the session limit of *seconds* of transaction time. 服务器状态：联机 用户操作：减少打开事务的空闲时间。
类 ID： 22 事件 ID： 22001	3	类消息： Long transaction detected 特定消息： Blocking on XA transaction, tx *transaction_number*, till it is cleaned up.	
类 ID： 22 事件 ID： 22002	3	类消息： Long transaction detected 特定消息： Continuing Long Transaction (for COMMIT): tx:	
类 ID： 22 事件 ID： 22003	3	类消息： Long transaction detected 特定消息： Aborting Long Transaction: tx:	

ID	严重性	消息	解释
类 ID： 23 事件 ID： 23001	2	类消息： Logical log *'number'* complete 特定消息： Logical Log *log_number* Complete, timestamp: *timestamp*.	逻辑日志已满，无法写入任何事务。 联机日志：消息指示逻辑日志已满。 服务器状态：联机 用户操作：无
类 ID： 24 事件 ID： 24001	3	类消息： Unable to allocate memory 特定消息： Generic unique event id when the server failed to allocate memory for starting a new thread.	
类 ID： 24 事件 ID： 24002	3	类消息： Unable to allocate memory 特定消息： Warning: unable to allocate requested big buffer of size *size*	
类 ID： 24 事件 ID： 24003	3	类消息： Unable to allocate memory 特定消息： The database server tried to allocate a shared memory virtual segment before it was actually needed, in accordance with the setting of the SHMVIRT_ALLOCSEG configuration parameter - but the segment could not be added. Next failure message will be printed in 30 minutes.	
类 ID： 24 事件 ID： 24004	3	类消息： Unable to allocate memory 特定消息： out of message shared memory	
类 ID： 24 事件 ID： 24005	3	类消息： Unable to allocate memory 特定消息： out of message shared memory	
类 ID：	3	类消息： Unable to allocate memory	

ID	严重性	消息	解释
24 事件 ID： 24006		特定消息： out of virtual shared memory	
类 ID： 24 事件 ID： 24007	3	类消息： Unable to allocate memory 特定消息： No memory available for page cleaners	
类 ID： 24 事件 ID： 24008	3	类消息： Unable to allocate memory 特定消息： kysearch(): Memory allocation error	
类 ID： 24 事件 ID： 24009	3	类消息： Unable to allocate memory 特定消息： Lock table overflow - user id *user_ID*, session id *session_ID*	
类 ID： 24 事件 ID： 24010	3	类消息： Unable to allocate memory 特定消息： Unable to allocate a user thread for user id *user_ID*	
类 ID： 24 事件 ID： 24011	3	类消息： Unable to allocate memory 特定消息： Unable to allocate a transaction for user id *user_ID*, session id *session_ID*	
类 ID： 26 事件 ID： 26001	3	类消息： Dynamically added log file *logid* 特定消息： Dynamically added log file *logid* to DBspace *dbspace_name*	
类 ID： 27	4	类消息： Log file required	数据库服务器需要额外的日志文件以继续处理。

ID	严重性	消息	解释
事件 ID: 27001		特定消息: ALERT: The oldest logical log (*log_number*) contains records from an open transaction (*transaction_number*). Logical logging will remain blocked until a log file is added. Add the log file with the onparams -a command, using the -i (insert) option, as in: onparams -a -d *dbspace* -s *size* -i Then complete the transaction as soon as possible.	联机日志: ALERT: The oldest logical log (log_number) contains records from an open transaction (transaction_number). Logical logging will remain blocked until a log file is added. Add the log file with the onparams -a command, using the -i (insert) option, as in: onparams -a -d dbspace -s size -i Then complete the transaction as soon as possible. 服务器状态: 联机 用户操作: 添加新的逻辑日志。
类 ID: 28 事件 ID: 28001	4	类消息: No space for log file 特定消息: ALERT: Because the oldest logical log (*log_number*) contains records from an open transaction (*transaction_number*), the server is attempting to dynamically add a log file. But there is no space available. Please add a DBspace or chunk. Then complete the transaction as soon as possible.	数据库服务器无法动态地添加额外的逻辑日志文件, 因为没有足够可用的空间。 联机日志: 断言警告指示没有足够可用空间可用于额外的逻辑日志文件 服务器状态: 联机 用户操作: 添加新的逻辑日志文件或额外的空间。
类 ID: 28 事件 ID: 28002	4	类消息: No space for log file 特定消息: Warning - Enterprise Replication is attempting to dynamically add a log file. But there is no space available. The replay position may overrun.	数据库服务器无法动态地添加额外的逻辑日志文件, 因为没有足够可用的空间。 联机日志: 断言警告指示没有足够可用空间可用于额外的逻辑日志文件 服务器状态: 联机 用户操作: 添加新的逻辑日志文件或额外的空间。
类 ID: 29 事件 ID: 29001	2	类消息: Internal subsystem: *subsystem* 特定消息: Skipped existing audit trail files *file_name* to *file_name*.	审计子系统需要更改到一个新的输出文件。 联机日志: 消息指示该审计文件变更跳过现有文件。 服务器状态: 联机 用户操作: 无
类 ID: 30 - 39	2, 3, or 4	Enterprise Replication 事件。请参阅《*SinoDB® Enterprise Replication* 指南》中的 *Enterprise Replication* 事件警报。	
类 ID: 40	3	类消息: RSS alarm	

ID	严重性	消息	解释
事件 ID： 40001		特定消息： RSS *server_name* added	
类 ID： 40 事件 ID： 40002	3	类消息： RSS alarm 特定消息： Password for RSS Source *server_name* changed	
类 ID： 40 事件 ID： 40003	3	类消息： RSS alarm 特定消息： RSS *server_name* deleted	
类 ID： 40 事件 ID： 40004	3	类消息： RSS alarm 特定消息： RSS *server_name* log replay position is falling too far behind RSS Source	
类 ID： 40 事件 ID： 40005	3	类消息： RSS alarm 特定消息： RSS *server_name* is not acknowledging log transmission	
类 ID： 40 事件 ID： 40006	3	类消息： RSS alarm 特定消息： Error receiving a buffer from RSS *server_name* - shutting down	
类 ID： 40 事件 ID： 40007	3	类消息： RSS alarm 特定消息： Delay or Stop Apply: I/O write error: *error_number error_description*.	
类 ID： 40	3	类消息： RSS alarm 特定消息：	

ID	严重性	消息	解释
事件 ID： 40008		Delay or Stop Apply: Thread exiting due to error.	
类 ID： 41 事件 ID： 41001	3	类消息： SDS alarm 特定消息： ERROR: Removing SDS Node *server_name* has timed out - removing	
类 ID： 42 事件 ID： 42001	1	类消息： Event occurred	数据库服务器在验证表空间页时遇到了错误。 联机日志：断言警告，带有该表的详细信息。 服务器状态：联机 用户操作：检查联机日志以确定发生问题的 *database.owner.tablename*。对该表运行 oncheck -pt 命令。纠正 oncheck 实用程序标识的错误，并重试该操作。如果该操作再次失败，则请记录所有情况，并联系星瑞格®软件支持。
类 ID： 43 事件 ID： 43001	3	类消息： Connection Manager alarm 特定消息： CM:Session for Connection manager *name* terminated abnormally	
类 ID： 43 事件 ID： 43002	3	类消息： Connection Manager alarm 特定消息： The FOC setting *FOC_String* for Connection Manager *CM_Name* does not match the FOC setting for the other Connection Managers that are configured to arbitrate failover for the cluster. If this Connection Manager becomes the active arbitrator, its FOC will not match the previous FOC policy.	
类 ID： 44 事件 ID： 44001	3	类消息： DBSpace is full: *dbspace_name* 特定消息： WARNING: *dbspace_type dbspace_name* is full	

ID	严重性	消息	解释
类 ID： 45 事件 ID： 45001	3	类消息： partition *'partition_name'*: no more extents 特定消息： Partition *'partition_name'*: No more extents	
类 ID： 46 事件 ID： 46001	3	类消息： partition *'partition_name'*: no more pages 特定消息： Partition *'partition_name*: No more pages	
类 ID： 47 - 71	3 or 4	Enterprise Replication 事件。请参阅《*SinoDB® Enterprise Replication* 指南》中的 *Enterprise Replication* 事件警报。	
类 ID： 72 事件 ID： 72001	2	类消息： Audit trail is switched to a new file. 特定消息： Audit trail switched to *file_name*	审计子系统切换到新的输出文件。 联机日志：消息提供新输出文件的文件名。 服务器状态：联机 用户操作：无
类 ID： 73-77	3 or 4	Enterprise Replication 事件。请参阅《*SinoDB® Enterprise Replication* 指南》中的 *Enterprise Replication* 事件警报。	
类 ID： 78 事件 ID： 78001	3	类消息： The storage pool is empty. 特定消息： Warning: The storage pool is out of space.	
类 ID： 79 事件 ID： 79001	3	类消息： Dynamically added chunk *chunk_name* to space 特定消息： Dynamically added chunk *chunk_name* to space '*space_name*' Path: *path*, offset *offset_number* kilobytes Size: *size* kilobytes	

ID	严重性	消息	解释
类 ID： 80 80001	2	类消息： A new fragment for table *table_name* has been added in DBspace *dbspace_name*.	自动地添加新分段到表，因为该表增长的大小大于现有分段的大小。 联机日志：消息提供表名称和数据库空间名。 服务器状态：联机 用户操作：无
类 ID： 81 事件 ID： 81001	4	类消息： Logical log file or dbspace corruption detected during backup. Loguniq or Dbspace id: *ID*. 特定消息： Log Backup detected a corrupted logical log file. Expected loguniq:pagenum *log_number*:*page_number* Actual loguniq:pagenum *log_number*:*page_number* Log backup continuing but the log backup cannot be used to restore a server. You should run oncheck and take a level 0 archive.	备份失败，因为数据库服务器在逻辑日志文件中或数据库空间中检测到损坏。 联机日志：断言警告 服务器状态：联机 用户操作：执行新 0 级备份。
类 ID： 82 事件 ID： 82001	3	类消息： session *ID* (*thread*) network write operation has been blocked for at least 30 minutes, which might indicate an operating system problem 特定消息： session *ID* (*thread*) network write operation has been blocked for at least 30 minutes, which might indicate an operating system problem	
类 ID： 83 事件 ID： 83001	3	类消息： SDS: Failover aborted - detected primary server is still active. 特定消息： SDS: Failover aborted - detected primary server is still active.	
事件 ID： 84001	3	类消息： Generic network failure alarm 特定消息：	主机名或 IP 地址、服务名或端口号可能不正确。该端口可能在使用中。 服务器状态：联机，服务器启动过程中。 联机日志：断言警告

ID	严重性	消息	解释
		Unable to bind to the port (port number or service name) on the host (IP address or host name) for the server (*dbservername*).	用户操作：检查主机名或 IP 地址、服务名与 sqlhosts 文件中的端口条目。请确认该端口没有在使用中。进行必需的更改并重新启动服务器。
事件 ID：86001	3	类消息： Space has reached its maximum configured size. Specific msg: Warning: Space *space_name* has reached its maximum configured size (*size* MB).	可扩展的存储空间处于配置的最大的大小，无法进一步扩展。 服务器状态：联机 联机日志： 用户操作：没有需要的操作。如果您想要增加存储空间的最大的大小，请运行带有 modify space sp_sizes 参数的 admin() 或 task() SQL 管理 API 函数并指定新的最大的大小。

严重性 5 事件警报
严重性 5 事件警报指示数据库服务器故障。

表 208: Severity 5 event alarms

ID	消息	解释
类 ID： 6 事件 ID： 6033	类消息： Internal subsystem failure: *'message'* 特定消息： Cache read error	数据库服务器在读取内部高速缓冲遇到错误后关闭。 联机日志：断言失败 服务器状态：脱机 用户操作： 启动数据库服务器并再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 6 事件 ID： 6041	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was detected by the Buffer Manager in the database server.	数据库服务器缓冲区管理器遇到内部错误，则数据库服务器关闭或纠正该问题。 联机日志： 断言警告或断言失败，并附有发生错误时执行的操作的描述。通常，断言警告显示该错误已内部更正。 服务器状态：如果该错误不可恢复，则处于脱机。如果该错误已更正，则处于联机。 用户操作： 如果该错误不可恢复，则启动数据库服务器并再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。如果该错误已有数据库服务器内部更正，则无需采取任何操作。
类 ID： 6 事件 ID： 6042	类消息： Internal subsystem failure: *'message'* 特定消息： An internal error was reported by the database server when it detected an	数据库服务器在内部缓冲区的处理过程中检测到不一致，则关闭或纠正该问题。 联机日志： 断言警告或断言失败，并附有发生错误时执行的操作的描述。通常，断言警告显示该错误已内部更正。 服务器状态：如果该错误不可恢复，则处于脱机。如果该错误已更正，则处于联机。

ID	消息	解释
	inconsistency with the internal buffer queues.	用户操作： 如果该错误不可恢复，则启动数据库服务器并再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。如果该错误已有数据库服务器内部更正，则无需采取任何操作。
类 ID: 6 事件 ID: 6045	类消息: Internal subsystem failure: *'message'* 特定消息: Logical logging error for '*object*' in *'space'*	数据库服务器由于处理逻辑日志时发生错误而关闭。 联机日志：断言失败，并附有该操作的描述和逻辑日志信息。 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6049	类消息: Internal subsystem failure: *'message'* 特定消息: Lock types *lock_type* and *lock_type* should never be merged	数据库服务器尝试合并不兼容的锁后关闭。 联机日志：断言失败，并附有数据库服务器尝试合并的锁类型。 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6050	类消息: Internal subsystem failure: *'message'* 特定消息: An internal error was reported by the database server when it detected some corruption in the lock free list chain.	数据库服务器在检测到用于管理可用锁的内部列表的内部结构的损坏后关闭。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6060	类消息: Internal subsystem failure: *'message'* 特定消息: Thread exited with *number* buffers held	数据库服务器检测到线程持有一个或多个缓冲区后关闭。 联机日志：断言失败，并附有该线程持有的缓冲区数。 服务器状态：脱机 用户操作：使数据库处于联机。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6067	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Recursive exception) has caused the database server processes to terminate unexpectedly.	数据库服务器检测到异常处理的递归调用，立即关闭以避免无限循环。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。

ID	消息	解释
类 ID: 6 事件 ID: 6068	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Internal exception) has caused the database server processes to terminate unexpectedly.	数据库服务器由于不可恢复的内部错误而关闭。 联机日志：断言失败，并附有关于导致该问题的异常的信息。 服务器状态：脱机 用户操作：启动数据库服务器。查看断言失败文件中的异常信息。如果该异常与用户定义例程相关，则调查并纠正该用户定义例程。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID: 6069	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (Master daemon died) has caused the database server processes to terminate unexpectedly.	主守护 oninit 进程停止且数据库服务器关闭。该错误由操作系统进程终止而导致。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。终止操作系统进程时，请谨慎。
类 ID: 6 事件 ID: 6070	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (VP died) has caused the database server processes to terminate unexpectedly.	oninit 进程停止且数据库服务器关闭。该错误由操作系统进程终止而导致。 联机日志：断言失败 服务器状态：脱机 用户操作：启动数据库服务器。终止操作系统进程时，请谨慎。
类 ID: 6 事件 ID: 6071	类消息: Internal subsystem failure: *'message'* 特定消息: ERROR: cannot fork secondary Server thread (MACH11 Shutdown)	辅助服务器关闭但无法创建正常关闭的线程。 联机日志： DR: Shutting down the server. ERROR: can not fork secondary Server thread (MACH11 Shutdown) Can not run onmode -ky PANIC: Attempting to bring system down. 服务器状态：脱机 用户操作：无
类 ID: 6 事件 ID: 6075	类消息: Internal subsystem failure: *'message'* 特定消息: A fatal internal error (KAIO) has caused the database server processes to terminate unexpectedly.	数据库服务器由于 KAIO 子系统中的错误而关闭。 联机日志：断言失败，并附有特定失败的操作。 服务器状态：脱机 用户操作：启动数据库服务器。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID: 6 事件 ID:	Generic event for when the database server terminates unexpectedly due to an internal error condition.	发生内部错误并且数据库服务器关闭。 联机日志：断言失败 服务器状态：脱机

ID	消息	解释
6500		用户操作：启动数据库服务器。检查断言失败文件以获取更多信息。如果可能的话，修复标识的问题并再次尝试该操作。如果该操作再次失败，则请记录所有情况并联系星瑞格®软件支持。
类 ID： 21 事件 ID： 21004	类消息： Database server resource overflow: *'resource_name'* 特定消息： Physical log file overflow	物理日志文件已满且需要溢出。如果在恢复过程中发生这个情况，则数据库服务器尝试扩展该物理日志。 联机日志：如果换数据库服务器不在恢复状态中或无法扩展该物理日志，则断言失败。如果换数据库服务器在恢复状态中且扩展该物理日志，则断言警告。 服务器状态：脱机 用户操作：无
类 ID： 21 事件 ID： 21006	类消息： Database server resource overflow: *'resource_name'* 特定消息： Logical log buffer overflow detected	数据库服务器由于逻辑日志缓冲区已满而关闭。 联机日志：断言失败，并附有日志记录大小和缓冲区大小。 服务器状态：脱机 用户操作：增加 onconfig 文件中 LOGBUFF 配置参数的值。启动数据库服务器。

连接管理器事件警报标识符

事件警报的类 ID 指示事件的类型。事件 ID 指示特定的事件。

下表列出连接管理器的事件警报 ID 和消息。

您可以设置您自己的警报程序脚本来捕获连接管理器类 ID 和消息，并启动纠正操作或通知。

当您编写连接管理器的警报处理程序时，可以使用 INFORMIXCMNAME 和 INFORMIXCMCONUNITNAME 环境变量中设置的值。如果连接管理器发出事件警报，连接管理器实例名存储于 INFORMIXCMNAME 环境变量，连接管理器连接单位名存储于 INFORMIXCMCONUNITNAME 环境变量。

事件警报消息写入连接管理器日志文件。

表 209: 连接管理器事件警报

ID	严重性	消息	解释
类 ID： 1 事件 ID： 1001	3	类消息： Connection Manager generic alarm 特定消息： Connection Manager stopped	连接管理器停止运行。 联机日志消息： Connection Manager shut down successfully. 用户操作： Restart the 连接管理器, if necessary.
类 ID： 1 事件 ID：	3	类消息： Connection Manager generic alarm 特定消息：	连接管理器初始化失败。 联机日志消息： Failed to switch to daemon mode, Connection Manager stopped.

ID	严重性	消息	解释
1002		Connection Manager fatal error	Error: Initialize failed, Connection Manager stopped. Error: SLA listener failed, Connection Manager can not start. 用户操作：检查消息文件以获取失败的详细信息。纠正错误百出，然后重新启动连接管理器。
类 ID: 1 事件 ID: 1003	3	类消息: Connection Manager generic alarm 特定消息: Connection Manager received signal	连接管理器停止或崩溃。 联机日志消息： Connection Manager process received signal, shutting down 用户操作：如果连接管理器是被信号 9 杀死，则无需采取任何操作。否则，向系统管理员报告该问题。
类 ID: 2 事件 ID: 2001	3	类消息: Failover Arbitrator alarm 特定消息: Failover in progress	连接管理器故障转移仲裁者启动故障转移事件。 联机日志消息： Failover Arbitrator automated failover in progress.
类 ID: 2 事件 ID: 2002	3	类消息: Failover Arbitrator alarm 特定消息: Failover completed	连接管理器故障转移仲裁者完成故障转移。 联机日志消息： Failover Arbitrator automated failover completed.
类 ID: 2 事件 ID: 2003	3	类消息: Failover Arbitrator alarm 特定消息: Failover disabled	连接管理器的自动故障转移被禁用。 联机日志消息： Failover Arbitrator automated failover is disabled. 用户操作： N/A
类 ID: 2 事件 ID: 2004	3	类消息: Failover Arbitrator alarm 特定消息: Failover Arbitrator aborting automated failover	故障转移处理失败。 联机日志消息： Failover Arbitrator aborting automated failover. 用户操作：检查消息日志文件，然后手动地启动主服务器或手动地执行故障转移。
类 ID: 2 事件 ID: 2005	3	类消息: Failover Arbitrator alarm 特定消息: Failover processing is in manual mode	故障转移处理不是自动模式。 联机日志消息： Failover processing is in manual mode
类 ID:	3	类消息:	连接管理器无法连接到主服务器。

ID	严重性	消息	解释
3 事件 ID： 3001		Connection to the primary 特定消息： Cannot connect to primary server	联机日志消息： Unable to connect to SinoDB® server. 用户操作：纠正设置问题。然后，连接管理器会自动地连接到该服务器。
类 ID： 3 事件 ID： 3002	3	类消息： Connection to the primary 特定消息： Lost connection to primary server	连接管理器与主服务器断开。 联机日志消息： Detected lost connection to SinoDB® server. 用户操作：纠正设置或网络问题。然后，连接管理器会自动地连接到该主服务器。
类 ID： 3 事件 ID： 3003	4	类消息： Connection to the primary 特定消息： CM detected multiple primary servers in the cluster	连接管理器连接到多个主服务器。 联机日志消息： Detected multiple primary servers in the cluster. 用户操作：纠正设置或网络问题，以便连接管理器仅连接到一个主服务器。
类 ID： 4 事件 ID： 4001	3	类消息： Connection to ER node 特定消息： Cannot connect to ER node	The 连接管理器无法连接到 Enterprise Replication 服务器。 用户操作：纠正设置问题。然后，连接管理器会自动地连接到 Enterprise Replication 服务器。
类 ID： 4 事件 ID： 4002	3	类消息： Connection to ER node 特定消息： Lost connection to ER node	连接管理器与 Enterprise Replication 服务器断开。 联机日志消息： Detected lost connection to SinoDB® server 用户操作：纠正设置或网络问题。然后，连接管理器会自动地连接到 Enterprise Replication 服务器。
类 ID： 5 事件 ID： 5001	3	类消息： Connection to generic server 特定消息： Cannot connect to server	连接管理器无法连接到高可用性集群中的服务器。 联机日志消息： Unable to connect to SinoDB® server. 用户操作：纠正设置问题。然后，连接管理器会自动地连接到高可用性服务器。
类 ID： 5 事件 ID： 5002	3	类消息： Connection to generic server 特定消息： Lost connection to server	连接管理器与辅助服务器断开。 联机日志消息： Detected lost connection to SinoDB® server. 用户操作：纠正设置或网络问题。然后，连接管理器会自动地连接到辅助服务器。

相关链接

《SinoDB SQL 指南: 参考》: INFORMIXCMNAME 环境变量

《*SinoDB SQL 指南: 参考*》: *INFORMIXCMCONUNITNAME 环境变量*

数据库服务器日志中的消息

未编号的消息打印在数据库服务器消息日志（online.log）中。错误消息包括纠正操作。

要获得错误信息描述，请使用 finderr 实用工具。

某些消息可以需要您联系星瑞格®软件支持。

相关链接

MSGPATH 配置参数 在第134页

《*SinoDB 备份和还原指南*》: *ON-Bar 消息和返回代码*

本章中消息是如何排列的

本章中的数据库服务器消息日志的消息以字母顺序排列，按以下附加规则排序：

- 忽略每个消息前面的时间戳。
- 按字母顺序的排列中忽略字母大小写。
- 忽略空格。
- 忽略引号。
- 忽略前导省略号。
- 如果 *the* 是消息中的第一个字词，则忽略它。
- 以数字或标点符号开始的消息显示在列表的末尾标记为“消息: 符号 在第815页”的特殊部分中。
- 某些相关消息归在一起，如下所示：
  - 转换和复原错误消息 在第815页
  - *Conversion and Reversion Messages for Enterprise Replication* 在第818页
  - 动态日志消息 在第820页
  - 智能大对象空间元数据消息 在第821页
  - 截断表消息 在第822页

消息文本后跟消息或消息组的原因和建议的纠正操作。

如何查看这些消息

使用以下方式之一来查看这些消息：

- 联机消息日志

  要查看发生时显示的消息，请使用 tail -f *online.log* 命令。
- onstat -m 命令

  有关更多信息，请参阅 *onstat -l 命令: 显示物理和逻辑日志信息* 在第546页。

要查看与这些未编号的消息相关的错误号，请查看 sysmaster 数据库中的 logmessage 表：

```
SELECT * FROM logmessage;
```

消息类别

未编号的消息存在四个一般类别，但一些消息归入多个类别：

- 常规信息
- 断言失败消息
- 需要的管理操作
- 检测到不可恢复错误

技术支持实用断言失败消息来辅助故障排除和诊断。其报告的信息通常归入意外事件类别，这些事件可能发展为或不发展为由其他错误代码捕获的问题。并且，这些消息是简洁且通常是极为技术性的。它们可能报告一个或两个孤立的统计信息，但不提供发生了什么的全面描述。此信息可建议技术支持可能的研究途径。

消息: A-B

Aborting Long Transaction: ***tx 0xn.***

原因

事务跨越事务高水位标志（LTXHWM）所指定的日志空间，且正在回滚违例长事务。

操作

不需要额外操作。共享内存中事务结构的地址显示为十六进制值。

Affinitied VP ***mm*** to phys proc ***nn***.

原因

数据库服务器成功地将 CPU 虚拟处理器绑定到物理处理器。

操作

不需要任何操作。

Affinity not enabled for this server.

原因

您尝试将 CPU 虚拟处理器绑定到物理处理器，但您所运行的数据库服务器不支持处理器专用。

操作

从 VPCLASS 配置参数移除专用设置。

Assert Failed: Error from SBSpace cleanup thread.

原因

智能大对象空间清除线程清除游离智能大对象时遇到错误。

操作

查看消息日志文件中建议的操作。

大多数时候，在失败的智能大对象空间上运行 onspaces -cl sbspacename 可以成功地清除游离的智能大对象。如果您遇到不可恢复错误，请联系技术支持。

Assert Failed: Short description of what failed Who: Description of user/session/thread running at the time Result: State of the affected database server entity Action: What action the database administrator should take See Also: DUMPDIR/af.uniqid containing more diagnostics.

原因

这个消息指示内部错误。

操作

ONCONFIG 的 DUMPDIR 参数所指定目录中的 af.uniqid 文件包含发送到消息日志的断言失败消息的副本，以及当前相关的结构和/或数据缓冲区的内容。该消息中包含的信息是供技术支持使用的。

Begin re-creating indexes deferred during recovery.

原因

恢复过程中，延迟索引创建直到恢复完成后。此消息指示数据库服务器推迟重新创建索引且目前正在创建索引。数据库服务器重新创建索引过程中，其使用共享锁锁定受影响的表。

操作

不需要任何操作。

Building 'sysmaster' database requires ***~mm*** pages of logical log. Currently there are ***nn*** pages available. Prepare to back up your logs soon.

原因

您当前没有完成 sysmaster 数据库建构所需的可用日志空间的接近量。

操作

备份日志。

Building 'sysmaster' database...

原因

数据库服务器正在建构 sysmaster 数据库。

操作

不需要任何操作。

消息: C

Cannot Allocate Physical-log File, ***mm*** wanted, ***nn*** available.

原因

数据库服务器尝试增加物理日志大小，但它需要比数据库空间中可用连续空间还大的连续空间。空间量以 KB 表示。

操作

您必须指定较小的物理日志大小（使用 PHYSFILE 配置参数）或将物理日志的位置更改到包含适当连续空间的数据库空间以容纳较大的物理日志。

Cannot alter a table which has associated violations table.

原因

用户尝试添加、删除或修改具有相关联违例表的表中的列。

操作

不要更改用户表中的列。

Cannot change to mode.

原因

在快速或完全恢复过程中的某些错误阻止了系统更改为联机或静默模式。

操作

请参阅日志文件中的以前的消息以获取信息。

Cannot Commit Partially Complete Transactions.

原因

直到 COMMIT 语句处理之后，删除表或索引的事务才会执行删除（少数例外）。在这些情况下，写入 *beginning commit* 日志记录，后跟一般的提交日志记录。如果数据库服务器在这两个操作之间失败，那么快速恢复进程尝试在您下一次初始化数据库服务器时完成该提交。

如果此提交的完成失败，那么数据库服务器产生前述消息。

操作

要确认您是否需要采取操作，请查看解释逻辑日志记录 在第273页中描述的逻辑日志。

Cannot create a user-defined VP class with 'SINGLE_CPU_VP' non-zero.

原因

SINGLE_CPU_VP 设置为非零，且使用 onmode 创建用户定义 VP 类。

操作

如果用户定义 VP 类是必需的，那么停止数据库服务器，将 SINGLE_CPU_VP 更改为零，并重新启动数据库服务器。

Cannot create violations/diagnostics table.

原因

用户发出目标表的 START VIOLATIONS TABLE 语句数据库服务器无法创建该目标表的违例表。任何以下情况可能是此失败的原因：

- 目标表已有违例表。
- 在 START VIOLATIONS TABLE 语句中指定了无效的违例表名称。例如：如果您省略了语句中的 USING 子句，且如果目标表中的字符数加上 4 个字符长超过最大的标识符长度，那么产生的违例表名称超过最大标识符长度。
- 在 START VIOLATIONS TABLE 语句中指定的违例表的名称与数据库中现有表的名称匹配。
- 目标表包含名称为 informix_tupleid、informix_optype 或 informix_recowner 的列。由于这些列名称与违例表中的 informix_tupleid、informix_optype 或 informix_recowner 列重复，因此数据库无法创建违例表。
- 目标表是临时表。
- 目标表是其他某些表的违例表。
- 目标表是系统目录表。

操作

要解决此错误，请执行下列操作之一：

- 如果违例表名称无效，那么请在 START VIOLATIONS TABLE 语句的 USING 子句中指定违例表的唯一名称。
- 如果目标表包含名称为 informix_tupleid、 informix_optype 或 informix_recowner 的列，那么请将这些列重命名为其他名称。
- 选择不是系统目录表也不是其他表的违例表的永久目标表。

Cannot insert from the violations table to the target table.

原因

用户发出了试图将行从违例表插入目标表的语句。例如：用户输入以下无效语句：

```
INSERT INTO mytable SELECT * FROM mytable_vio;
```

如果目标表具有过滤方式约束，那么您也会接收到此错误。

操作

要从此错误恢复，请执行以下操作：

- 不要使用过滤约束。
- 停止违例表。
- 将行从违例表插入临时表，然后，将行从临时表插入目标表。

Cannot modify/drop a violations/diagnostics table.

原因

用户尝试更改或删除作为另一表的违例表的表。

操作

不要更改或删除违例表。

Cannot Open Dbspace ***nnn***.

原因

数据库服务器无法访问指定的数据库空间。此消息指示打开表空间的问题或数据库空间的初始块中的损坏。

操作

验证构成该数据库空间的块的设备正常运行，并且已指定给它们正确的操作系统权限（rw-rw----）。您可能需要执行数据复原。

Cannot Open Logical Log.

原因

数据库服务器无法访问逻辑日志文件。由于数据库服务器不能访问逻辑日志就无法运行，因此您必须解决这个问题。

操作

验证逻辑日志文件驻留的块设备是否正常运行且具有正确的操作系统权限（rw-rw----）。

Cannot Open Mirror Chunk ***pathname***, errorno = ***nn***.

原因

数据库服务器无法打开镜像对的镜像块。返回块 *pathname* 和操作系统错误。

操作

有关纠正操作的更多信息，请参阅您的操作系统文件。

Cannot Open Primary Chunk ***pathname***, errorno = ***nnn***.

原因

无法打开镜像对的主块。返回块 *pathname* 和操作系统错误。

操作

有关纠正操作的更多信息，请参阅您的操作系统文件。

Cannot Open Primary Chunk ***chunkname.***

原因

无法打开数据库空间的 *initial* 块。

操作

验证块设备正常运行且具有正确的操作系统权限（rw-rw----）。

Cannot open sysams in database ***name***, iserrno ***number***.

原因

数据库服务器打开 sysams 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot open sysdistrib in database ***name***, iserrno ***number***.

原因

数据库服务器访问 sysdistrib 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot open ***system_table*** in database ***name***, iserrno ***number***.

原因

数据库服务器打开指定的系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot open systrigbody in database ***name***, iserrno ***number***.

原因

数据库服务器访问 systrigbody 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot open systriggers in database ***name***, iserrno ***number***.

原因

数据库服务器访问 systriggers 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot open sysxtdtypes in database ***name***, iserrno ***number***.

原因

访问 sysxtdtypes 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Cannot Perform Checkpoint, shut system down.

原因

正在尝试复原镜像块的线程请求了检查点，但无法执行该检查点。

操作

关闭数据库服务器。

Cannot Restore to Checkpoint.

原因

数据库服务器无法恢复物理日志从而无法执行快速恢复。

操作

如果数据库服务器未联机，那么请从数据库空间备份执行数据复原。

Cannot Rollback Incomplete Transactions.

原因

在快速恢复或数据复原过程中，逻辑日志记录是最先前滚的。然后，回滚尚未提交的打开事务。打开事务可能在回滚中失败，从而使打开事务的某些修改留在原地。此错误不会阻止数据库服务器变成静默模式或联机模式，但可能指示数据库不一致。

操作

要确认是否需要任何操作，请使用 onlog 实用程序检查逻辑日志。

Cannot update pagezero.

原因

数据库服务器在回复过程中尝试重写保留页时发生故障。

操作

请参阅日志文件中以前的消息以获得信息，或联系技术支持。

Cannot update syscasts in database ***name***. Iserrno ***number***.

原因

向 syscasts 系统表插入数据时发生内部错误。

操作

联系技术支持。

Can't affinity VP ***mm*** to phys proc ***nn***.

原因

数据库服务器支持进程专用，但将虚拟处理器绑定到物理处理机的系统调用失败。

操作

请参阅您的操作系统文件。

Changing the sbspace minimum extent value: old value ***value1***, new value ***value2***.

原因

当您发出以下命令时发生这个参考性消息：

```
onspaces -ch sbspace -Df "MIN_EXT_SIZE=value1" -y
```

操作

无。有关更多信息，请参阅 *onspaces -ch:* 更改智能大对象空间缺省规范 在第376页。

Checkpoint blocked by down space, waiting for override or shutdown.

原因

数据库空间在检查点间隔期间已关闭。当发生这种情况是时，数据库服务器配置为等待重置。

操作

关闭数据库服务器或发出 onmode -O 命令来重置已关闭的数据库空间。有关 onmode 实用程序的更多信息，请参阅 *onmode* 实用程序 在第330页。

Checkpoint Completed: duration was ***n*** seconds.

原因

检查点成功地完成。

操作

不需要任何操作。

Checkpoint Page Write Error.

原因

数据库服务器在尝试将检查点信息写入磁盘时检测到错误。

操作

有关解决该情况的其他帮助，请联系技术支持。

Checkpoint Record Not Found in Logical Log.

原因

逻辑日志或包含逻辑日志的块损坏。数据库服务器无法初始化。

操作

从数据库空间备份执行数据复原。

Chunk ***chunkname*** added to space ***spacename.***

原因

此消息中的变量具有以下值：

chunkname
　数据库服务器管理员正在添加的块的名称。
spacename
　数据库服务器管理员正在添加该块的存储空间的名称。

操作

不需要任何操作。

Chunk ***chunkname*** dropped from space ***spacename.***

原因

数据库服务器管理员从空间 *spacename* 中删除块 *chunkname*。

操作

不需要任何操作。

Chunk ***number nn pathname*** -- Offline.

原因

镜像对中所指示的块标记为状态 D 并脱机。镜像对中的另一块正在成功地运行。

操作

立即采取步骤修复该块设备并复原该块。将显示块 *number* 和块设备 *pathname*。

Chunk ***number nn pathname*** -- Online.

原因

镜像对中所指示的块已恢复并已联机（标记为状态 O）。将显示该块 *number* 和块设备 *pathname* 。

操作

不需要任何操作。

The chunk ***pathname*** must have READ/WRITE permissions for owner and group.

原因

块 *pathname* 没有正确的所有者和群组权限。

操作

请确保对该块所在设备指定了正确的权限（-rw-rw---）。

The chunk ***pathname*** must have ***owner-ID*** and ***group-ID*** set to informix.

原因

块 *chunkname* 没有正确的所有者和组 ID。

操作

确保该块所在的设备具有所有权。在 UNIX™ 上，所有者和群组应该是 informix。在 Windows™ 上，所有者必须是 Informix-Admin 群组的成员。

The chunk ***pathname*** will not fit in the space specified.

原因

块 *pathname* 无法放入所指定的空间。

操作

选择较小的块大小，或释放空间以在其中创建块。

Cleaning stray LOs in sbspace ***sbspacename***.

原因

数据库服务器管理员正在运行 onspaces -cl sbspacename。

操作

不需要任何操作。

Completed re-creating indexes.

原因

数据库服务器已完成推迟索引的重新创建。

操作

不需要任何操作。

Configuration has been grown to handle up to ***integer*** chunks.

原因

数据库服务器管理员通过更改 CONFIGSIZE 或将 MAX_CHUNKS 设置为较高的值，将块数增加到指定的值。

操作

不需要任何操作。更改成功。

Configuration has been grown to handle up to ***integer*** dbslices.

原因

数据库服务器管理员通过更改 CONFIGSIZE 或将 MAX_DBSLICES 设置为较高的值，将数据库片的数量增加到指定的值。

操作

不需要任何操作。更改成功。

Configuration has been grown to handle up to ***integer*** dbspaces.

原因

数据库服务器管理员通过更改 CONFIGSIZE 或将 MAX_DBSPACES 设置为较高的值，将数据库空间的数量增加到指定的值。

操作

不需要任何操作。更改成功。

Continuing Long Transaction (for COMMIT): ***tx 0xn***.

原因

逻辑日志已填充到超过长事务高水位标志（LTXHWM），当违例长事务正在提交中。在这种情况下，允许该事务继续写入逻辑日志且不回滚。共享内存中事务结构的地址显示为十六进制值 *tx 0xn*。

操作

不需要任何操作。

Could not disable priority aging: errno = ***number***.

原因

当正在尝试禁用 CPU 虚拟处理器优先级迟滞时，操作系统调用失败。返回与该失败相关联的系统错误 *number*。

操作

请参阅操作系统文件。

Could not fork a virtual processor: errno = ***number***.

原因

虚拟处理器派生失败。数据库服务器返回与该失败相关联的操作系统错误 *number*。

操作

有关确定每个用户可用进程和系统整体可用进程的最大数量的信息，请参阅操作系统文件。

Create_vp: cannot allocate memory.

原因

数据库服务器无法分配新的共享内存。

操作

数据库服务器管理员必须使更多的共享内存可用。这种情况可能需要增加 SHMTOTAL 或重新配置操作系统。此消息通常伴随着给出其他信息的其他信息一起出现。

消息: D-E-F

Dataskip is OFF for all dbspaces.

原因

参考性。

操作

不需要任何操作。

Dataskip is ON for all dbspaces.

原因

参考性。

操作

不需要任何操作。

Dataskip is ON for dbspaces: ***dbspacelist***.

原因

参考性；指定的数据库空间的 DATASKIP 为 ON。

操作

不需要任何操作。

Dataskip will be turned {ON|OFF} for ***dbspacename***.

原因

参考性；指定的数据库空间的 DATASKIP 为 ON 或 OFF。

操作

不需要任何操作。

DBSERVERALIASES exceeded the maximum limit of 32

原因

达到 32 个别名的限制。

操作

无。将只使用前 32 个别名。

DBSPACETEMP internal list not initialized, using default.

原因

当初始化用户指定的 DBSPACETEMP 列表时发生错误。通常，这种情况是因为内存分配失败。

操作

检查伴随的错误消息。

The DBspace/BLOBspace ***spacename*** is now mirrored.

原因

您已成功地将镜像添加到所指示的存储空间。

操作

不需要任何操作。

The DBspace/BLOBspace ***spacename*** is no longer mirrored.

原因

您已结束了所指示的存储空间的镜像。

操作

不需要任何操作。

devname: write failed, file system is full.

原因

由于文件系统 *devname* 已满，所以写入失败。

操作

释放 *devname* 中的一些空间。

Dropping temporary tblspace ***0xn***, recovering ***nn*** pages.

原因

共享内存初始化过程中，数据库服务器例行地搜索遗留下而未适当清除的临时表。如果数据库服务器找到临时表，则删除该表并恢复该空间。数据库服务器定位指定的临时表空间并删除它。值 *0xn* 是表空间号的十六进制表示。

操作

不需要任何操作。

Dynamically allocated new shared memory segment (size ***nnnn***).

原因

此状态消息通知您数据库服务器已成功地分配了大小为 *nnnn* 的新共享内存段。

操作

不需要任何操作。

ERROR: NO "wait for" locks in Critical Section.

原因

数据库服务器不允许线程位于临界段时拥有必须等待的锁。任何此类锁请求都遭到拒绝，并向用户返回一条 ISAM 错误消息。

操作

所报告的错误是内部错误。请联系星瑞格®技术支持。

Error building sysmaster database. See ***outfile***.

原因

构建 sysmaster 数据库时遇到了错误。文件 *outfile* 包含 buildsmi 脚本的运行结果。

操作

请参阅文件 *outfile*。

Error in dropping system defined type.

原因

更新 sysxtdtypes、sysctddesc 或 sysxtdtypeauth 系统表时发生内部错误。

操作

联系技术支持。

Error in renaming systdist.

原因

尝试查找并重命名 SPL 例程 informix.systdist 时发生内部错误。

操作

联系技术支持。

Error removing sysdistrib row for tabid = ***tabid***, colid = ***colid*** in database ***name***. iserrno = ***number***

原因

更新 sysdistrib 系统表时发生错误。

操作

记录错误号 *number* 并联系技术支持。

Error writing ***pathname*** errno = ***number***.

原因

操作系统无法写入 *pathname*。*Number* 是返回的操作系统错误编号。

操作

调查操作系统错误的原因。通常，它表示没有可用空间可用于该文件。也可能表示目录不存在或没有写入权限。

Error writing shmem to file ***filename*** (***error***). Unable to create output file ***filename*** errno=***mm.***Error writing ***filename*** errno=***nn***.

原因

数据库服务器在尝试将共享内存写入 *filename* 时检测到错误。第一个消息后跟着下两条消息中的一条消息。该尝试失败的原因是无法创建输出文件，或者是无法写入共享内存的内容。该错误指的是提示共享内存尝试写入文件的操作系统错误。值 *nn* 是操作系统错误。

操作

请参阅操作系统文件。

Fail to extend physical log space.

原因

扩展物理日志空间的尝试失败。路径不存在或权限不正确。

操作

使用存在的路径。检查当前工作目录的权限。您或系统管理员必须给予组在当前工作目录上的执行权限。在组拥有权限后，重试产生此消息的操作。

Fatal error initializing CWD string. Check permissions on current working directory. Group ***groupname*** must have at least execute permission on '.'.

原因

组 *groupname* 没有当前工作目录的执行权限。

操作

检查当前工作目录的权限。您或系统管理员必须给予组在当前工作目录上的执行权限。在组拥有权限后，重试产生此消息的操作。

Fragments ***dbspacename1 dbspacename2*** of table ***tablename*** set to non-resident.

原因

tablename 的指定分段已由 SET TABLE 语句设置为非常驻。

操作

不需要任何操作。

Forced-resident shared memory not available.

原因

您计算机的数据库服务器端口不支持强制常驻共享内存。

操作

不需要任何操作。

Freed ***mm*** shared-memory segment(s) ***number*** bytes.

原因

在您运行 onmode 实用程序的 -F 选项以释放未使用的内存后，数据库服务器发送此消息到消息日志。该消息通知您数据库服务器成功地释放的段数和字节数。

操作

不需要任何操作。

消息: G-H-I

gcore ***pid***; mv core.***pid*** dir/core.***pid***.ABORT.

原因

在数据库服务器失败过程中，此状态消息提供与虚拟处理器相关联的每个核心文件的名称和位置。

操作

不需要任何操作。

I/O ***function*** chunk ***mm***, pagenum ***nn***, pagecnt ***aa*** --> errno = ***bb***.

原因

在尝试从磁盘空间访问数据的过程中发生操作系统错误。*function* 为失败的操作系统功能。发生错误的页的块编号和物理地址显示为整数。*pagecnt* 值指的是线程尝试读取或写入的页数。如果显示 *errno* 值，它是操作系统错误编号，可能解释该失败。如果 *function* 被指为 *bad request*，那么某些意外事件已导致在无效块或页上的 I/O 尝试。

操作

如果块状态更改为 D 或关闭，则从其镜像恢复该块或修复该块。否则，执行数据恢复。

I/O error, ***primary/mirror*** Chunk ***pathname*** -- Offline (***sanity***).

原因

数据库服务器在具有 *pathname* 的主块或镜像块上检测到 I/O 错误。该块已脱机。

操作

检查该块所存储于的设备是否正常运行。

已删除索引 idx1 和 idx 2 错误消息。

SinoDB® ***database_server*** Initialized - Complete Disk Initialized.

原因

已初始化磁盘空间和共享内存。初始化之前存在于磁盘上的任何数据库现在都是不可访问的。

操作

不需要任何操作。

SinoDB® ***database_server*** Initialized - Shared Memory Initialized.

原因

共享内存已初始化。

操作

不需要任何操作。

SinoDB® ***database_server*** Stopped.

原因

数据库服务器已从静默模式变为脱机模式。数据库服务器是处于脱机模式。

操作

不需要任何操作。

In-Place Alter Table. Perform EXECUTE FUNCTION sysadmin:task('table update_ipa', 'table_name','database');

原因

对表进行定点 alter 操作时，尝试还原到以前版本的数据库服务器。以前版本的数据库服务器无法处理具有多个行模式的表。

操作

在尝试还原到以前的数据库服务器版本之前，通过更新受影响的表中的行来强制执行定点 alter 操作。运行带有 table update_ipa 参数的 SQL 管理 API task() 或 admin() 命令来解决该表上所有暂挂的定点 alter 操作。

ERROR: Insufficient available disk in the root dbspace to increase the entire Configuration save area.

原因

用户尝试通过更改 CONFIGSIZE 或将 MAX_DBSPACES、MAX_DBSLICES 或 MAX_CHUNKS 设置为较高的值来将存储对象的数量增加到特定值，但数据库服务器没有足够的根空间用于增加的存储对象。存储对象可能是数据库空间、数据库片或块。

操作

增加根数据库空间的大小，或将 CONFIGSIZE、 MAX_DBSPACES、MAX_DBSLICES 或 MAX_DBSLICES 重新设置为较低的值，并重新启动数据库服务器。例如：如果将 MAX_CHUNKS 设置为 32,768，但根数据库空间没有足够空间，那么将 MAX_CHUNKS 设置为较低的值。

Insufficient available disk in the root dbspace for the CM save area. Increase the size of the root dbspace in the ONCONFIG file and reinitialize the server.

原因

The cause might be one of the following:

- 用户尝试通过更改 CONFIGSIZE 或将 MAX_DBSPACES、MAX_DBSLICES 或 MAX_CHUNKS 设置为较高的值来增加存储对象数量到特定值，但数据库服务器没有足够的根空间用于增加的存储对象。存储对象可能是数据库空间、数据库片或块。
- 用户已转换到需要稍多根空间的数据库服务器版本，但它不可用（不太可能有这种情况）。

操作

采取下列操作之一：

- 增加根数据库空间的大小或将 CONFIGSIZE、 MAX_DBSPACES、MAX_DBSLICES 或 MAX_DBSLICES 重新设置为较低的值，并重新启动数据库服务器。例如：如果将 MAX_DBSPACES 设置为 32,768，但根数据库空间没有足够空间，那么将 MAX_DBSPACES 设置为较低的值。
- 增加根数据库空间的大小并重新初始化数据库服务器。

Internal overflow of shmid's, increase system max shared memory segment size.

原因

数据库服务器正在初始化共享内存，此时它耗尽了与该段相关联的共享内存 ID 的内部存储。

操作

增加最大核心共享内存段大小的值，通常是 SHMMAX。有关更多信息，请参阅操作系统文件。

消息: J-K-L-M

Listener-thread err = ***error_number***: ***error_message.***

原因

监听器线程遇到错误。此消息显示错误号和消息文本。

操作

要获得错误信息描述，请使用 finderr 实用工具。

Lock table overflow - user id ***mm*** session id ***nn.***

原因

当无锁可用时，线程尝试获取锁。显示用户 ID 和会话 ID。

操作

增大 LOCKS 配置参数，并初始化共享内存。

Logical-log File not found.

原因

根数据库空间保留页中的检查点记录损坏。

操作

从数据库空间备份执行数据复原。

Logical Log ***nn*** Complete.

原因

日志 ID 号 *nn* 标识的逻辑日志文件已满。数据库服务器自动地切换到序列中的下一个逻辑日志文件。

操作

不需要任何操作。

Logical logging ***vberror*** for ***type:subtype*** in (***failed_system***).

原因

日志记录失败。导致该错误的日志记录如下标识：

type
  是逻辑日志记录类型。
subtype
  是日志记录子系统。
failed_system
  指示系统无法日志记录的内部函数的名称。

操作

联系技术支持。

Log Record: log = ***ll***, pos = ***0xn***, type = ***type:subtype(snum),*** trans = ***xx***

原因

数据库服务器在快速恢复或逻辑日志复原的前滚部分中检测到错误。

导致该错误的日志记录如下标识：

ll
　是存储该记录的逻辑日志 ID。

0xn
　是日志中的十六进制地址位置。

type
　是逻辑日志记录类型。

subtype
　是日志记录子系统。

snum
　是子系统编号。

xx
　是出现在逻辑日志中的事务编号。

操作

联系技术支持。

Log record (***type:subtype***) at log ***nn***, ***0xn*** was not undone.

原因

日志撤销因日志损坏而失败。

导致该错误的日志记录如下标识：

type
　是逻辑日志记录类型。

subtype
　是日志记录子系统。

nn
　是存储该记录的逻辑日志 ID。

0xn
　是日志中的十六进制地址位置。

操作

要确定是否需要任何操作，请使用 onlog 实用程序检查逻辑日志。联系技术支持。

Log record (***type:subtype***) failed, partnum ***pnum row rid*** iserrno ***num.***

原因

发生日志记录失败。

导致该错误的日志记录如下标识：

type
是逻辑日志记录类型。
subtype
是日志记录子系统。
pnum
是部件号。
rid
是行 ID。
num
是 iserror 号。

操作

联系技术支持。

Log record (***type:subtype***) in log ***nn***, offset ***0xn*** was not rolled back.

原因

日志撤销因日志损坏而失败。

导致该错误的日志记录如下标识：

type
是逻辑日志记录类型。
subtype
是日志记录子系统。
log
是存储该记录的逻辑日志 ID。
offset
是日志中的十六进制地址位置。

操作

要确定是否需要任何操作，请使用 onlog 实用程序检查逻辑日志。联系技术支持。

Logical Recovery allocating ***nn*** worker threads ***thread_type***.

原因

数据库服务器确定用于并行恢复的工作程序线程的数量。变量 *thread_type* 可以采用值 ON_RECVRY_THREADS 或 OFF_RECVRY_THREADS。

操作

此状态消息不需要任何操作。如果您想要为并行恢复分配不同的工作程序线程的数量，请更改 ONCONFIG 配置参数 ON_RECVRY_THREADS 或 OFF_RECVRY_THREADS 的值。

Logical Recovery Started.

原因

逻辑恢复开始。

操作

此状态消息不需要任何操作。

Maximum server connections ***number***.

原因

每个检查点消息的输出指示在上一次数据库服务器重新启动以来到数据库服务器的最大并发连接数。

操作

此消息帮助客户跟踪许可证使用，以确定何时需要购买更多的许可证。有关协助，请联系技术支持。

Memory allocation error.

原因

数据库服务器耗尽共享内存。

操作

采取下列操作之一：

1. 增加计算机上交换空间。
2. 检查核心共享内存参数的共享内存限制。
3. 使用 BUFFERPOOL 配置参数中的 buffers 字段减少分配的内存的大小。
4. 增加虚拟内存大小（SHMVIRTSIZE）、添加的段的大小（SHMADD）或总共享内存大小（SHMTOTAL）。

Mirror Chunk ***chunkname*** added to space ***spacename***. Perform manual recovery.

原因

快速恢复、完全恢复或 HDR 辅助服务器已恢复镜像块的添加。然而，它不执行自动镜像恢复。管理员必须执行这一操作。

操作

使用 onspaces 实用程序尝试恢复镜像块。

Mixed transaction result. (***pid***=***nn*** user=***userid***).

原因

仅当事务涉及多个数据库服务器时才接收到这个消息。此消息指示数据库服务器在准备提交事务后尝试回滚该事务，且全局事务完成时不一致。*pid* 值是协调者进程的用户进程标识号。*user* 值是与协调者进程相关联的用户 ID。

操作

请参阅《*SinoDB*® 管理员指南》中有关从失败的两阶段提交手动恢复的信息。

mt_shm_free_pool: pool ***0xn*** has blocks still used (id ***nn***).

原因

在池取消分配过程中由于块仍与该池关联而发生内部错误。

操作

联系技术支持。

mt_shm_init: can’t create ***resident/virtual*** segment.

原因

创建常驻段或虚拟段失败的原因为：（1）段大小小于最小的段大小；（2）段大小大于最大的段大小；（3）分配另一段将超过允许的总共享内存大小；或（4）数据库服务器尝试分配段时失败。

操作

如果怀疑此错误是由前段中的第 1 或 2 项生成，请联系技术支持。要纠正第 3 项，请增加 ONCONFIG 配置文件中的 SHMTOTAL 值。有关第 4 项生成的错误的其他信息，请参阅逻辑日志文件。

mt_shm_remove: WARNING: may not have removed all/correct segments.

原因

当操作系统尝试移除与数据库服务器相关联的共享内存段时，最后一段不等于内部注册的最后一段。这种情况可能是由于数据库服务器的意外失败。

操作

移除未清除的所有段。

消息: N-O-P

Newly specified value of ***value*** for the pagesize in the configuration file does not match older value of ***value***. Using the older value.

原因

此消息在数据库服务器重新启动时显示。在数据库服务器初始化后更改了 ONCONFIG 文件中的 PAGESIZE 值。

操作

数据库服务器使用较旧的 PAGESIZE 值。

Not enough main memory.

原因

数据库服务器在尝试从操作系统获得更多内存空间时检测到错误。

操作

有关共享内存配置和管理的更多信息，请参阅操作系统文件。

Not enough logical-log files, Increase LOGFILES.

原因

在数据恢复过程中，LOGFILES 配置参数的值必须始终大于或等于逻辑日志文件的总数。在恢复过程中的某个时刻，逻辑日志文件数超过 LOGFILES 的值。

操作

增加 ONCONFIG 中的 LOGFILES 的值。

The number of configured inline poll threads exceeds the number of CPU virtual processors.

原因

当 NETTYPE 配置参数指定的内联轮询线程数超过 VPCLASS 配置参数指定的 CPU 虚拟处理器数时生成此消息。配置在 CPU 虚拟处理器上运行的轮询线程被视为内联轮询线程。

操作

更改 VPCLASS 配置参数以增加 CPU 虚拟处理器数，或更改 NETTYPE 配置参数以减少内联轮询线程数。

相关链接

NETTYPE 配置参数 在第135页
VPCLASS 配置参数 在第196页

onconfig parameter ***parameter*** modified from ***old_value*** to ***new_value.***

原因

当数据库服务器共享内存重新初始化时，此消息记载自上一次初始化以来所发生的所有更改。

操作

不需要任何操作。

oninit: Cannot have SINGLE_CPU_VP non-zero and number of CPU VPs greater than 1.

原因

ONCONFIG 文件包含 num= 值大于 1 的 VPCLASS cpu，且 SINGLE_CPU_VP 的值非零。当有多个 CPU VP 时，SINGLE_CPU_VP 必须为 0（或省略）。

操作

更正 ONCONFIG 文件并重新启动数据库服务器。

oninit: Cannot have SINGLE_CPU_VP non-zero and user-defined VP classes.

原因

ONCONFIG 文件包含用户定义的 VPCLASS 以及非零值的 SINGLE_CPU_VP。当 ONCONFIG 包含用户定义的 VPCLASS 时，SINGLE_CPU_VP 必须为 0（或省略）。

操作

更正 ONCONFIG 文件并重新启动数据库服务器。

oninit: Fatal error in initializing ASF with 'ASF_INIT_DATA' flags asfcode = '25507'.

原因

在数据库服务器的 sqlhosts 文件或注册表中指定的 nettype 值无效或不受支持，或在数据库服务器的 sqlhosts 文件或注册表中指定的 servicename 无效。

操作

检查每个 DBSERVERNAME 和每个 DBSERVERALIASES 的 sqlhosts 文件或注册表中的 nettype 和 servicename 值。检查 ONCONFIG 文件中每个 NETTYPE 参数的 nettype 值。

Cannot alter a table which has associated violations table.

原因

用户尝试添加、删除或修改具有相关联违例表的表中的列。

操作

不要更改用户表中的列。

oninit: Too many VPCLASS parameters specified.

原因

ONCONFIG 文件中指定了太多的 VPCLASS 参数。

操作

如果可能的话，减少 VPCLASS 行数。如果不可能的话，请联系技术支持。

oninit: VPCLASS ***classname*** bad affinity specification.

原因

VPCLASS 行的专用规范不正确。专用作为范围进行指定：

对于 *m*，使用处理器 *m*。

对于 *m* 到 *n*，使用范围 *m* 到 *n*（包括 *m* 和 *n*） 的处理器，其中 *m* <= *n*，*m* >= 0 且 *n* >= 0。

操作

更正 ONCONFIG 文件中的 VPCLASS 参数并重新启动数据库服务器。

oninit: VPCLASS ***classname*** duplicate class ***name***.

原因

ONCONFIG 文件中的 VPCLASS *classname* 有重复的名称。VP 类名称必须是唯一的。

操作

更正重复的名称，并重新启动数据库服务器。

oninit: VPCLASS ***classname*** illegal option.

原因

VPCLASS *classname* 参数中的字段之一是不合法的。

操作

更正 ONCONFIG 文件中的参数并重新启动数据库服务器。

oninit: VPCLASS ***classname*** maximum number of VPs is out of the range 0-10000.

原因

VPCLASS 参数行指定的最大 VP 数必须在范围 1 到 10，000 中。

操作

更正该值并重新启动数据库服务器。

oninit: VPCLASS ***classname*** name is too long. Maximum length is ***maxlength***.

原因

VPCLASS *classname* 中的名称字段的长度太长。

操作

选择较短的类名称，更正 ONCONFIG 文件，并重新启动数据库服务器。

oninit: VPCLASS ***classname*** number of VPs is greater than the maximum specified.

原因

VPCLASS 参数所指定的 VP 初始数量大于同一 VPCLASS 参数所指定的最大值。

操作

更正 VPCLASS 参数并重新启动数据库服务器。

oninit: VPCLASS ***classname*** number of VPs is out of the range 0-10000.

原因

VPCLASS 参数行所指定的 VP 初始数量必须是在范围 1 到 10,000 中。

操作

更正该值并重新启动数据库服务器。

onmode: VPCLASS ***classname*** name is too long. Maximum length is ***maxlength***.

原因

动态地添加的 VP 类名（由 onmode -p 指定）太长。

操作

选择较短的名称，并重试 onmode -p 命令。

Online Mode.

原因

数据库服务器处于联机模式。用户可以访问所有数据库。

操作

此状态消息不需要任何操作。

onspaces: unable to reset dataskip.

原因

此错误消息来自 onspaces 实用程序。因某种原因，该实用程序无法跨数据库服务器实例中所有数据库空间更改 DATASKIP 的规范（ON 或 OFF）。

操作

您不太可能接收到此消息。如果在重新启动数据库服务器后该错误仍存在，那么联系技术支持。

Open transaction detected when changing log versions.

原因

数据库服务器在尝试从数据库服务器的以前版本转换数据时检测到打开的事务。

操作

不允许转换，除非日志中的最后一个记录是检查点。您必须恢复数据库服务器的以前版本，强制检查点，然后重试转换。

Out of message shared memory.

原因

数据库服务器无法分配更多的内存给指定的段。

操作

有关更多信息，请参阅日志文件。

Out of resident shared memory.

原因

数据库服务器无法分配更多的内存给指定的段。

操作

有关更多信息，请参阅日志文件。

Out of virtual shared memory.

原因

数据库服务器无法分配更多的内存给指定的段。

操作

有关更多信息，请参阅日志文件。

PANIC: Attempting to bring system down.

原因

发生致命的数据库服务器错误。

操作

查看引起紧急情况的错误，并尝试该错误消息建议的纠正操作。有关可解释该失败的更多信息，还请参阅消息日志文件中的其他消息。

Participant site ***database_server*** heuristically rolled back.

原因

远程站点在达到“准备好提交”阶段之后回滚事务。

操作

您可能需要在其他站点回滚该事务，然后重新启动它。

Physical recovery complete: ***number*** pages examined, ***number*** pages restored.

原因

此消息在快速恢复过程中显示。*number of pages examined* 指示存在于物理日志中的页映像的数量。*number of pages restored* 指示从物理日志恢复的实际页数。恢复的页数总是少于或等于检查到的数量。

数据库服务器可能在检查点之间多次物理地记录页映像。物理恢复只恢复第一个记录的页映像。

如果页留在内存缓冲池中，那么数据库服务器在每个检查点记录它一次，并在物理日志中存储一个页映像。如果缓冲池太小，那么被更新多次的页可能会被强制从缓冲池移到磁盘，然后带回到内存供下一次更新。每次该页被带回到内存时，将再次物理记录它，因而导致在物理日志中有重复的页映像。

操作

如果 *number of pages examined* 远远大于 *number of pages restored*，则增加缓冲池的大小以降低重复前映像的数量。有关更多信息，请参阅《*SinoDB*® 性能指南》。

Physical recovery started at page (***chunk***:***offset***).

原因

此消息在快速恢复过程中显示。*Chunk* 是包含物理日志的块的编号。*Offset* 是物理日志条目起始的页偏移量。物理恢复从那个点开始恢复页。

操作

不需要任何操作。有关快速恢复的信息，请参阅《*SinoDB*® 管理员指南》。

Portions of partition partnum of table tablename in database dbname were not logged. This partition cannot be rolled forward.

原因

自上次备份以来发生于操作表的轻量级追加。

操作

如果您想要完全访问该表中的数据，那么需要将该表更改为行，然后更改为想要的表类型。该更改操作除去了由重放未记录操作（例如：轻量级追加）所导致的表中的不一致。

Possible mixed transaction result.

原因

此消息指示返回错误 -716。与此消息相关联的是其上事务结果未知的数据库服务器的列表。

操作

有关确定事务是否实现不一致的信息，请参阅《*SinoDB*® 管理员指南》。

Prepared participant site ***server_name*** did not respond.

原因

联系远程站点 *server_name* 的尝试太多。在几个超过时间间隔之后，可判定该站点已关闭。

操作

验证远程站点处于联机，且针对分布式事务正确配置。一旦远程站点就绪，重新启动事务。

Prepared participant site ***server_name*** not responding.

原因

数据库服务器尝试联系远程站点 *server_name*。因某种未知原因，数据库服务器无法联系该远程站点。

操作

验证远程站点处于联机，并针对分布式事务正确配置。

消息: Q-R-S

Quiescent Mode.

原因

数据库服务器从某个其他状态进入静默模式。在 UNIX™ 上，只有以 informix 或 root 登录的用户可以与数据库服务器交互。在 Windows™ 上，只有 Informix-Admin 组的成员可以与数据库服务器交互。没有用户可以访问数据库。

操作

不需要任何操作。

Read failed. Table ***name***, Database ***name***, iserrno = ***number***

原因

读取指定的系统表时发生错误。

操作

记下错误编号并联系技术支持。

Recovery Mode.

原因

数据库服务器进入恢复模式。直到恢复完成之前，没有用户可以访问数据库。

操作

不需要任何操作。

Recreating index: '***dbsname***:"***owner***".***tabname-idxname***'.

原因

在 DDL 语句隐式或显式创建一个或多个新索引之后，但在下一个检查点之前数据库服务器异常终止，那么新索引的重新创建延迟到逻辑恢复之后，而不是逐行添加每个索引项。在逻辑恢复结束之后，服务器开始并行索引构建以重新创建它们。此消息显示每个延迟的索引开始重建的时间。（但是如果索引在异常关闭之前已删除，那么在逻辑恢复之后不会重建它，并且不会输出有关该索引的消息。）

操作

不需要任何操作。

Rollforward of log record failed, iserrno = ***nn***.

原因

在快速恢复或数据恢复过程中，如果数据库服务器无法前滚特定的逻辑日志记录，则显示该消息。数据库服务器可能变更为静默或联机模式，但可能导致某些不一致。有关进一步信息，请参阅恰在这条消息前面的那条消息。*iserrno* 值是错误编号。

操作

联系星瑞格®技术支持。

Root chunk is full and no additional pages could be allocated to chunk descriptor page.

原因

Root 块已满。

操作

要释放 Root 块中的空间，请采取下列操作之一：

- 删除并重建 sysmaster 数据库。
- 将用户表从根数据库空间搬移到另一数据库空间。
- 对表进行重新分段。

scan_logundo: subsys ***ss***, type ***tt***, iserrno ***ee.***

原因

日志撤销因日志类型 *tt* 损坏而失败。

此消息中的变量具有以下值：

ss
是子系统名称。

tt
是逻辑日志记录类型。

ee
是 iserror 编号。

操作

使用 onlog 实用程序检查逻辑日志以确定是否需要任何操作。联系技术支持。

Session completed abnormally. Committing ***tx*** id ***0xm***, flags ***0xn***.

原因

仅当数据库服务器尝试提交没有当前所有者的事务，且该事务发展成长事务时，才发生异常会话结束。数据库服务器派生一个线程来完成该提交。

操作

不需要任何操作。

Session completed abnormally. Rolling back ***tx*** id ***0xm***, flags ***0xn.***

原因

仅当数据库服务器尝试提交没有当前所有者的分布式事务，且该事务发展成长事务时，才发生异常会话结束。数据库服务器派生一个线程来回滚该事务。

操作

不需要任何操作。

semctl: errno = ***nn.***

原因

当数据库服务器初始化信号量时发生错误。返回操作系统错误。

操作

请参阅操作系统文件。

semget: errno = ***nn.***

原因

信号量集的分配失败。返回操作系统错误。

操作

请参阅操作系统文件。

shmat: ***some_string os_errno***: ***os_err_text***.

原因

附加到共享内存段的尝试失败。返回系统错误编号和建议的更正操作。

操作

查看更正操作（如果给出的话），并决定是否值得尝试。有关更多信息，请参阅操作系统文件。

shmctl: errno = ***nn.***

原因

当数据库服务器尝试除去或锁定共享内存段时发生错误。返回操作系统错误编号。

操作

请参阅操作系统文件。

shmdt: errno = ***nn.***

原因

当数据库服务器尝试从共享内存段拆离时发生错误。返回操作系统错误编号。

操作

请参阅操作系统文件。

shmem sent to ***filename.***

原因

数据库服务器把共享内存副本写入指定的文件作为断言失败的结果。

操作

无。

shmget: ***some_str os_errno***: key ***shmkey***: ***some_string.***

原因

共享内存段的创建失败，或获取与特定键相关联的共享内存 ID 的尝试失败。返回系统错误编号和建议的更正操作。

操作

请参阅操作系统文件。

Shutdown (onmode -k) or override (onmode -O).

原因

数据库空间在检查点间隔过程中已关闭。当发生这种情况时，数据库服务器配置为等待重设。

当实际发生检查点时，出现以下消息： Checkpoint blocked by down space, waiting for override or shutdown。

操作

关闭数据库服务器或发出 onmode -O 命令以重设关闭的数据库空间。有关 onmode 实用程序的更多信息，请参阅 *onmode* 实用程序 在第330页。

Shutdown Mode.

原因

数据库服务器正在从联机模式转换为静默模式。

操作

不需要任何操作。

Space ***spacename*** added.

原因

数据库服务器管理员添加新存储空间 *spacename* 到数据库服务器。

操作

不需要任何操作。

Space ***spacename*** dropped.

原因

数据库服务器管理员从数据库服务器删除存储空间 *spacename*。

操作

不需要任何操作。

Space ***spacename*** -- Recovery Begins(***addr***).

原因

此参考消息指示数据库服务器尝试恢复存储空间。

此消息中的变量具有以下值：

spacename
　是数据库服务器正在恢复的存储空间的名称。
addr
　是控制块的地址。

操作

不需要任何操作。

Space ***spacename*** -- Recovery Complete(***addr***).

原因

此参考消息指示数据库服务器已恢复存储空间。

此消息中的变量具有以下值：

spacename
　是数据库服务器已恢复的存储空间的名称。
addr
　是控制块的地址。

操作

不需要任何操作。

Space ***spacename*** -- Recovery Failed(***addr***).

原因

此参考消息指示数据库服务器无法恢复存储空间。

此消息中的变量具有以下值：

spacename
是数据库服务器恢复失败的存储空间的名称。

addr
是控制块的地址。

操作

不需要任何操作。

sysmaster database built successfully.

原因

数据库服务器已成功构建 sysmaster 数据库。

操作

不需要任何操作。

Successfully extend physical log space

原因

物理日志空间已成功地扩展到指定路径下的文件 plog_extend.*servernum*。

操作

不需要任何操作。

消息: T-U-V

This ddl operation is not allowed due to deferred constraints pending on this table and dependent tables.

原因

当尝试启动违例表且约束处于推迟方式时，返回此错误。

注：如果启动违例表，然后将约束设置为推迟， 那么不返回任何错误。然而，违例立即得到撤销，而不是写入推迟的约束缓冲区。有关更多信息，请参阅《*SinoDB® SQL* 指南*:* 语法》。

操作

如果要启动违例表，则必须将约束方式更改为立即，或提交该事务。

This type of space does not accept log files.

原因

不允许添加逻辑日志文件到 Blob 空间或智能大对象空间。

操作

添加逻辑日志文件到数据库空间。有关更多信息，请参阅 *onparams -a -d dbspace:* 添加逻辑日志文件 在第352页。

TIMER VP: Could not redirect I/O in initialization, errno = ***nn.***

原因

操作系统无法打开 null 设备或复制与打开该设备相关联的文件描述符。返回系统错误编号。

操作

请参阅操作系统文件。

Too Many Active Transactions.

原因

在数据恢复过程中有太多活动事务。在恢复过程中的某个时刻，活动事务的数量超过了 32 KB。

操作

无。

Too many violations.

原因

诊断表中的违例数超过 START VIOLATIONS TABLE 语句的 MAX VIOLATIONS 子句中所指定的限制。当目标表上的单个语句（）向违例表插入记录的数量超过 MAX VIOLATIONS 子句所指定的限制时，向在目标表上发出该语句的用户返回此错误。

操作

要解决此错误，请执行下列操作之一：

- 当启动违例表时，省略 START VIOLATIONS TABLE 语句的 MAX VIOLATIONS 字句。此处，您指定违例表中的行数没有限制。
- 将 MAX VIOLATIONS 设置为较高的值。

Transaction Not Found.

原因

逻辑日志损坏。这个情况发生在启动新事务时，该事务的第一个逻辑日志记录不是 BEGWORK 记录。

操作

联系技术支持。

Transaction heuristically rolled back.

原因

在事务完成两阶段提交的第一阶段之后，发生了回滚该事务的试探性决策。

操作

不需要任何操作。

Transaction table overflow - user id ***nn***, process id ***nn.***

原因

当共享内存表中没有可用条目时，线程尝试在事务表中分配一个条目。显示请求线程的用户 ID 和进程 ID。

操作

稍后重试。

Unable to create output file ***filename*** errno = ***nn.***

原因

操作系统无法创建输出文件 *filename*。*errno* 是返回的操作系统错误编号。

操作

验证目录存在且具有写入权限。

Unable to extend ***nn*** reserved pages for ***purpose*** in root chunk.

原因

操作系统无法在根块中扩展到 *nn* 保留页用于 *purpose*。（值 *purpose* 可以是检查点/日志、数据库空间、块或镜像块。）

操作

减少所指示资源的 ONCONFIG 参数；启动数据库服务器并释放主根块中的一些空间。然后，再尝试相同的操作。

Unable to start SQL engine.

原因

数据库服务器遇到内存耗尽的情况。

操作

不需要任何操作。

Unable to open tblspace ***nn***, iserrno = ***nn.***

原因

数据库服务器无法打开指定的表空间。（值 *nn* 是表空间编号的十六进制的表示。）

操作

请参阅 ISAM 错误消息编号 *nn*，其应能说明为何无法访问该表空间。

The value of pagesize ***pagesize*** specified in the config file is not a valid pagesize. Use 2048, 4096 or 8192 as the value for PAGESIZE in the onconfig file and restart the server.

原因

此消息在磁盘初始化时显示。ONCONFIG 文件中指定的 PAGESIZE 值是无效值。

操作

以有效的 PAGESIZE 值重新启动数据库服务器。

Violations table is not started for the target table.

原因

如果您发出 STOP VIOLATIONS TABLE 语句而没有启动违例表，那么您会接收到此消息。

操作

要从此错误中恢复，则必须启动目标表的违例表。

Violations table reversion test completed successfully.

原因

当 revtestviolations.sh 脚本成功完成（未找到打开的违例表）时，在 sysmaster 数据库的 logmessage 表中记录此消息。

操作

不需要任何操作。

Violations table reversion test failed.

原因

当数据库服务器找到打开的违例表时，它会在 sysmaster 数据库的 logmessage 表中报告错误 16992 和 16993，并终止该复原进程。

操作

当此消息出现时，您必须对每个打开的违例表发出 STOP VIOLATIONS TABLE FOR *table_name* 命令。在关闭所有打开的违例表之后，可以重新启动复原进程。

Violations table reversion test start.

原因

当执行 revtestviolations.sh 脚本时，在 sysmaster 数据库的 logmessage 表中记录此消息。

操作

不需要任何操作。

Violations tables still exist.

原因

当找到打开的违例表时，在 sysmaster 数据库的 logmessage 表中记录此消息。

操作

当出现此消息时，您必须对每个打开的违例表发出 STOP VIOLATIONS TABLE FOR *table_name* 命令。在关闭所有打开的违例表之后，可以重新启动复原进程。

Virtual processor limit exceeded.

原因

配置数据库服务器的虚拟处理器数超过所允许的最大数（1000）。

操作

修改 VPCLASS 配置参数的值、NETTYPE 配置参数的值，或者两个都修改。

相关链接

NETTYPE 配置参数 在第135页
VPCLASS 配置参数 在第196页

VPCLASS ***classname*** name is too long. Maximum length is ***maxlength***.

原因

此消息指示内部错误。

操作

联系技术支持。

VPCLASS ***classname*** duplicate class name.

原因

此消息指示内部错误。

操作

联系技术支持。

VPCLASS ***classname*** Not enough physical procs for affinity.

原因

VP 类 *classname* 的专用规范中的物理处理器不存在或脱机。

操作

确定指定的处理器是联机的。更正指定的 VP 类的专用规范。重新启动数据库服务器。

消息: W-X-Y-Z

WARNING: aio_wait: errno = ***nn.***

原因

当数据库服务器正在等待 I/O 请求完成时，它在其正在尝试执行的操作上生成错误号 *nn*。

操作

请联系技术支持，以获得帮助。

WARNING: Buffer pool size may cause database server to get into a locked state. Recommended minimum buffer pool size is ***num*** times maximum concurrent user threads.

原因

缓冲池中没有足够的缓冲区。数据库服务器可能使用了所有可用的缓冲区，并导致死锁发生。

操作

将 ONCONFIG 文件中的 BUFFERPOOL 参数的 buffers 字段更改为此消息建议的数量。有关 BUFFERPOOL 参数的更多信息，请参阅 *BUFFERPOOL* 配置参数 在第78页。

Warning: Chunk time stamps are invalid.

原因

在系统初始化第一次打开块时执行块的稳定情况检查。指定的块未通过检查并将变成脱机。

操作

从数据库空间备份或该块的镜像中恢复该块。

Warning: ***name_old*** is a deprecated onconfig parameter. Use ***name_new*** instead. See the release notes and the SinoDB® Administrator's Reference for more information.

原因

使用了已弃用的 ONCONFIG 参数。您第一次使用弃用的参数时显示此消息。之后显示此消息的短版形式。

操作

使用建议的替代 ONCONFIG 参数。

Warning: ***name_old*** is a deprecated onconfig parameter. Use ***name_new*** instead.

原因

使用了弃用的 ONCONFIG 参数。

操作

使用建议的替代 ONCONFIG 参数。

Warning: Unable to allocate requested big buffer of size ***nn.***

原因

对大缓冲区的内部内存分配失败。

操作

增加虚拟内存大小（SHMVIRTSIZE）、添加段的大小（SHMADD），或总共享内存大小（SHMTOTAL）。

You are turning off smart large object logging.

原因

这些更改将成为新的智能大对象空间缺省值。已对智能大对象空间进行了更改。onspaces 实用程序将一次读取和更新 100 个智能大对象，并将每个 100 个智能大对象的块作为单个事务提交。该实用程序可能要花费长时间才能完成。

操作

当您发出以下命令时出现此参考信息：

```
onspaces -ch sbspace -Df “LOGGING=OFF“ -y
```

有关更多信息，请参阅 *onspaces -ch:* 更改智能大对象空间缺省规范 在第376页。

消息: 符号

HH:MM:SS SinoDB® database server Version ***R.VV.PPPPP*** Software Serial Number RDS#***XYYYYYY***.

原因

共享内存初始化之后，此消息指示数据库服务器的启动。

操作

不需要任何操作。

argument: invalid argument.

原因

此内部错误指示向内部例程传递了无效参数。

操作

联系技术支持。

function_name: cannot allocate memory.

原因

数据库服务器无法从共享内存池分配内存。

操作

增加虚拟内存大小（SHMVIRTSIZE）、添加段的大小（SHMADD），或总共享内存大小（SHMTOTAL）。

转换和复原错误消息

如果转换或复原不成功，那么在 online.log 文件中存储错误消息以帮助您辨识失败的原因和要修复该问题应采取的操作。

Cannot revert new fragment expression for index ***index***, tabid ***id***.

原因

索引分段定义在比要复原的版本更新的版本中。

操作

删除有问题的索引分段模式并重试复原。

Cannot revert new table fragment expression for ***table*** with id ***id***.

原因

该表的分段定义在比要复原的版本更新的版本中。

操作

删除有问题的表分段模式并重试复原。

The conversion of the database ***name*** has failed.

原因

指示指定的数据库的转换失败。

操作

连接到该数据库。此操作触发数据库的转换。如果失败，则出现相关的错误消息。联系技术支持。

Database ***name*** is not revertible...

原因

数据库无法通过复原检查之一，且是不可复原的。

操作

采取操作以更正作为单独消息显示的错误。

Database ***name***: Must drop trigger (id = ***id_number***) before attempting reversion.

操作

数据库包含的触发器是在比要转换到的版本更新的版本中创建的。

操作

删除带有指定触发器标识号的触发器，然后尝试复原。

The dummy updates failed while converting database ***name***. This may imply data corruption in the database. If so, restore the original database with the tape backup. For more information, see ***output_file***.

原因

在从比 9.2 版更早版本的数据库的转换过程中，虚拟更新语句运行在转换的数据库中的系统表上。此消息指示运行这些更新语句之一时失败。

操作

要重试虚拟更新，请运行您旧数据库服务器版本的虚拟更新脚本。有关指示信息，请参阅《*SinoDB*® 迁移指南》。

如果发生数据损坏，请使用磁带备份来恢复原数据库。有关更多信息，请参阅《*SinoDB*® 备份和还原指南》。

Error in slow altering a system table.

原因

当执行复原时发生内部错误。

操作

联系技术支持。

Internal server error.

原因

数据库复原过程中发生意外错误。

操作

联系技术支持。

Must drop long identifiers in table ***name*** in database ***name***

原因

正在复原到的数据库服务器版本不支持长度大于 18 字节的标识符。

操作

在尝试复原之前，确认已删除或重命名了系统中所有长标识符。

Must drop new database (***name***) before attempting reversion. Iserrno ***error_number***

原因

系统包含创建于较新数据库服务器版本的数据库。

操作

删除新数据库并尝试复原。

Must drop new user defined statistics in database ***name***, iserrno ***number***

原因

sysdistrib 系统表中的某些分布使用用户定义的统计信息。正在复原到的版本不支持此功能。

操作

确保系统中不存在或未使用用户定义的统计信息，然后尝试复原。

Reversion canceled.

原因

复原进程因遇到错误而取消。

操作

更正错误原因，并重新启动复原。

There is a semi-detached index in this table, which cannot be reverted.

原因

无法复原此表上半拆离索引。

操作

要查看所有半拆离索引的列表，请参阅数据库服务器消息日志。删除所有半拆离索引，并重试复原。在复原完成之后，您可能需要重新创建那些索引。

WARNING: Target server version must have a certified Storage Manager installed after conversion/reversion and before bringing up server.

原因

ON-Bar 正在转换或复原。用户必须确保已安装了以目标数据库服务器版本认证的存储管理器

操作

无。

Conversion and Reversion Messages for Enterprise Replication

在转换和复原过程中，通过 concdr、revcdr 和 revtestcdr 脚本记录有关 Enterprise Replication 的特定消息。

缺省情况下，这些脚本将消息写到标准输出。这些消息存储在 $INFORMIXDIR/etc 目录（在 UNIX™ 上）或 %INFORMIXDIR%\etc 目录（在 Windows™ 上）中的 concdr.out、revcdr.out 和 revtestcdr.out 文件中。

CDR reversion test failed; for details look in $INFORMIXDIR/etc/revtestcdr.out.

原因

Enterprise Replication 是不可复原的。

操作

有关更多信息，请查看 revtestcdr.out 中的消息。在尝试复原之前，修复已报告的问题。

将 revcdr.sh 或 revcdr.bat 的输出打印到标准输出。

Enterprise Replication is not ready for conversion. The Control and TRG send queues should be empty for conversion/reversion to proceed.

原因

在控制和事务发送队列（也称为 TRG）发送队列中存在元素。数据库服务器将复制的数据发送到目标系统之前发送到 TRG 队列。

操作

在尝试转换或复原之前等待这些队列清空。有关更多信息，请参阅《*SinoDB® Enterprise Replication* 指南》。

在转换过程中将打印此消息到 concdr.out 或在复原过程中打印此消息到 revcdr.out。

Enterprise Replication should be in a stopped state for conversion/reversion to proceed.

原因

Enterprise Replication 应处于停止状态，以继续进行转换或复原。

操作

停止 Enterprise Replication。有关更多信息，请参阅《*SinoDB® Enterprise Replication* 指南》。

在转换过程中将此消息打印到 concdr.out 或在复原过程中将此消息打印到 revcdr.out。

... 'syscdr' reversion failed; for details look in $INFORMIXDIR/etc/revcdr.out.

原因

syscdr 数据库复原失败。

操作

在 revcdr.out 文件中查找失败的原因，然后在尝试复原之前修正该问题。

将 revcdr.sh 或 revcdr.bat 脚本的输出打印到标准输出。

'syscdr' conversion failed. For details, look in $INFORMIXDIR/etc/concdr.out.

原因

syscdr 数据库转换失败。

操作

如果转换失败，请解决 concdr.out 中报告的问题。从备份中复原 syscdr 数据库并重新尝试转换。

将 concdr.sh 或 concdr.bat 脚本的输出打印到标准输出。

Syscdr should NOT contain new replicate sets for reversion to succeed.

原因

syscdr 数据库中的新复制集与较旧的版本不兼容。

操作

使用 cdr delete replicateset 命令删除复制集。然后重新运行 revcdr.sh 或 revcdr.bat 脚本以重新尝试复原。

将此消息打印到 revtestcdr.out。

Syscdr should not contain replicates defined with the --floatieee option for reversion to succeed.

原因

已使用 --floatieee 选项定义了复制。您无法将这些复制复原到较旧的版本。

操作

使用 cdr delete replicateset 命令删除使用 --floatieee 选项定义的复制，然后重新尝试复原。

将此消息打印到 revtestcdr.out。

动态日志消息

Dynamically added log file ***logid*** to DBspace ***dbspace_number***.

原因

下一个活动日志文件包含打开事务的记录。每当数据库服务器动态地添加日志时，它就会记录此消息。例如：Dynamically added log file 38 to DBspace 5。

操作

尽快完成事务。

Log file ***logid*** added to DBspace ***dbspace_number***.

原因

每当管理员手动地添加日志文件时，数据库服务器就会记录此消息。例如：Log file 97 added to Dbspace 2.

操作

不需要任何操作。

Log file number ***logid*** has been dropped from DBspace ***dbspace_number***.

原因

当您删除新添加的日志文件时，数据库服务器就会记录此消息。例如：Log file number 204 has been dropped from DBspace 17.

操作

不需要任何操作。

Log file ***logid*** has been pre-dropped.

原因

当您删除已使用的日志文件时，它将被标记为删除（状态 D）且不能再使用。在您执行了 0 级备份之后，数据库服务器会删除该日志文件并可重新使用该空间。例如：Log file 12 has been pre-dropped.

操作

要删除日志文件，请对所有存储空间执行 0 级备份。

Pre-dropped log file number ***logid*** has been deleted from DBspace ***dbspace_number***.

原因

在备份之后，数据库服务器删除预先删除的日志文件并记录此消息。例如：Pre-dropped log file number 12 has been deleted from DBspace 3.

操作

不需要任何操作。

ALERT: Because the oldest logical log (***logid***) contains records from an open transaction (***transaction_address***), the server is attempting to dynamically add a log file. But there is

no space available. Please add a DBspace or chunk. Then complete the transaction as soon as possible.

原因

如果数据库服务器因实例已耗尽空间而无法动态地添加日志文件，则它记录此消息。

操作

添加数据库空间或块到现有数据库空间。然后尽快完成该事务。

ALERT: The oldest logical log (***logid***) contains records from an open transaction (***transaction_address***). Logical logging will remain blocked until a log file is added. Add the log file with the onparams -a command, using the -i (insert) option, as in: onparams -a -d ***dbspace*** -s ***size*** -i. Then complete the transaction as soon as possible.

原因

如果 DYNAMIC_LOGS 参数设置为 1，那么数据库服务器将会提示管理员在需要时手动地添加日志文件。

操作

使用带有 -i 选项的 onparams -a 命令来在当前日志文件后面添加日志文件。然后尽快完成该事务。

Log file ***logid*** has been pre-dropped. It will be deleted from the log list and its space can be reused once you take level-0 archives of all BLOBspaces, Smart BLOBspaces and non-temporary DBspaces.

原因

当您删除已使用的日志文件时，它会被标记为已删除（状态 D）且无法再使用，且 onparams 打印此消息。

操作

要删除日志文件，请对所有存储空间执行 0 级备份。

智能大对象空间元数据消息

Allocated ***number*** pages to Metadata from chunk ***number***.

原因

数据库服务器从保留区释放指定数量的页，并将它们移到块 *number* 的元数据区。

操作

不需要任何操作。

Allocated ***number*** pages to Userdata from chunk ***number***.

原因

数据库服务器从保留区释放指定数量的页，并将它们移到块 *number* 的用户数据区。

操作

不需要任何操作。

Freeing reserved space from chunk ***number*** to Metadata.

原因

块 *number* 中的元数据区已满。数据库服务器正尝试从保留区释放空间给元数据区。

操作

不需要任何操作。

Freeing reserved space from chunk ***number*** to Userdata.

原因

块 *number* 中的用户数据区已满。数据库服务器正尝试从保留区释放空间给用户数据区。

操作

不需要任何操作。

截断表消息

The table cannot be truncated if it has an open cursor or dirty readers.

原因

您必须对表具有互斥访问权。

操作

等待脏阅读器完成或关闭所有打开的游标并重新发出 TRUNCATE TABLE 命令。

The table cannot be truncated. It has at least one non-empty child table with referential constraints.

原因

如果表含有带有参考约束和至少一行的子表，那么您不能截断它。

操作

在截断该表之前，请清空子表。

SinoDB® 中的限制

本节列出 SinoDB® 的系统级和表级参数限制、系统缺省值，以及存取能力。

- 系统级参数限制 在第823页
- 表级参数限制 在第823页
- *SinoDB* 系统缺省值 在第824页
- 存取能力 在第824页

系统级参数限制

表 210: 系统级参数限制

系统级参数	每个计算机系统的最大容量
每台计算机的 SinoDB® 系统（取决于可用系统资源）	255
可访问的远程站点的最大数量	依机器而定
虚拟共享内存段的最大值（SHMVIRTSIZE）	2GB（32 位平台）或 4TB（64 位平台）
SinoDB® 共享内存段的最大数量	1024
最大的地址空间	UNIX：依机器而定 Windows：2.7 GB 如果启用了 4 GB 调整： • Windows™ 2003 之后的所有 Windows™ 版本 • Windows™ 2003 和更早的版本，如果 boot.ini 文件包含 /3GB 开关 1.7 GB 对于 Windows™ 2003 和更早的版本，如果 boot.ini 文件不包含 /3GB 开关

表级参数限制

表 211: 表级参数限制

表级参数（页大小为 2K 除非另有说明）	每个表的最大容量
每页的数据行	255
每个分段的数据行	4,277,659,295
每个分段的数据页	16,775,134
每个分段的数据字节数（不包括在 BLOB 空间中创建的智能大对象（BLOB、CLOB）和 简单大对象（BYTE、TEXT））	2K 页大小 = 33,818,670,144 4K 页大小 = 68,174,144,576 8K 页大小 = 136,885,093,440 12K 页大小 = 205,596,042,304 16K 页大小 = 274,306,991,168
二进制大对象 BLOB/CLOB 页数	4 TB
二进制大对象 TEXT/BYTE 字节数	4 TB
行长度	32,767
列数	32K
每个索引分段的最大页数	2,147,483,647
每个索引的键部分	16
每个功能索引的列	102（对于 C UDR）341（对于SPL 或 Java™ UDR）
每个索引键的最大字节数：	2K 页大小 = 387 4K 页大小 = 796 8K 页大小 = 1615 12K 页大小 = 2435

表级参数（页大小为 2K 除非另有说明）	每个表的最大容量
	16K 页大小 = 3254
SQL 语句大小的最大值	仅受限于可用内存

SinoDB® 系统缺省值

表 212: 系统缺省值

下表中的每一行罗列一个数据库特性，后跟该特性的 SinoDB® 系统缺省值。

数据库特性	SinoDB® 系统缺省值
表锁模式	页
初始扩展数据块大小	8 页
下一个扩展数据块大小	8 页
Read-only 隔离级别（带有数据库事务）	Committed Read
Read-only 隔离级别（ANSI兼容数据库）	Repeatable Read

存取能力

表 213: 存取能力

存取能力	每个系统的最大容量
每个 SinoDB® 系统的数据库最大值	2100 万
每个 SinoDB® 系统的最大表数	477 102 080
Maximum active users 每个 SinoDB® 的最大活动用户数（减去系统线程的最小数）	32K 用户线程
每个数据库和表的最大活动用户数（也受可用锁数和可调参数限制）	32K 用户线程
会话中打开数据库的最大数量	UNIX：32 个数据库 Windows：8 个数据库
每个 SinoDB® 系统的打开表的最大数量	动态分配
每个用户和联接的打开表的最大数量	动态分配
每个实例的打开事务的最大数量	32 767
每个 SinoDB® 系统和数据库的最大锁数	动态分配
页清除程序的最大数量	128
每个数据库空间的最大分区数	4K 页大小：1048445，2K 页大小：1048314（基于 4 位的位图）
递归同义词映射的最大数量	16
每个用户使用 LOCK TABLE 锁定的表的最大数量	32
每个用户的游标的最大数量	依机器而定
Enterprise Replication 事务大小的最大值	4 TB

存取能力	每个系统的最大容量
Maximum dbspace size	131 PB
智能大对象空间大小的最大值	131 PB
块大小的最大值	4 TB
块的最大数量	32 766
每个块的 2K 页的最大数量	20 亿
打开的简单大对象的最大数量（仅适用于 TEXT 和 BYTE 数据类型）	20
B 型树级别的最大数量	20
决策支持内存的最大量	依机器而定
实用程序对于大文件的支持	170 亿 GB
存储空间（数据库空间、Blob 空间、智能大对象空间或外部空间）的最大数量	2047