



参考手册：过程

Adaptive Server[®] Enterprise

15.7

文档 ID: DC37417-01-1570-01

最后修订日期: 2011 年 9 月

版权所有 © 2011 by Sybase, Inc. 保留所有权利。

本出版物适用于 Sybase 软件 and 任何后续版本, 除非在新版本或技术声明中另有说明。此文档中的信息如有更改, 恕不另行通知。此处说明的软件按许可协议提供, 其使用和复制必须符合该协议的条款。

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

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

Sybase 商标可在位于 <http://www.sybase.com/detail?id=1011207> 的 “Sybase 商标页” (Sybase trademarks page) 处进行查看。Sybase 和列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

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

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

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

IBM 和 Tivoli 是 International Business Machines Corporation 在美国和 / 或其它国家 / 地区的注册商标。

提到的所有其它公司和产品名均可能是与之相关的相应公司的商标。

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

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

目录

| | | |
|--------------|-----------------------------|----------|
| 第 1 章 | 系统过程 | 1 |
| | 系统过程简介 | 1 |
| | 系统过程的返回值 | 2 |
| | 系统过程的权限 | 2 |
| | 审计系统过程 | 2 |
| | 执行系统过程 | 3 |
| | 输入参数值 | 3 |
| | 消息 | 4 |
| | 系统过程表 | 5 |
| | sp_activeroles | 6 |
| | sp_add_qpgroup | 8 |
| | sp_add_resource_limit | 9 |
| | sp_add_time_range | 13 |
| | sp_addalias | 16 |
| | sp_addauditrecord | 18 |
| | sp_addauditable | 20 |
| | sp_addengine | 22 |
| | sp_addexceclass | 24 |
| | sp_addextendedproc | 26 |
| | sp_addexternlogin | 28 |
| | sp_addgroup | 31 |
| | sp_addlanguage | 32 |
| | sp_addlogin | 35 |
| | sp_addmessage | 36 |
| | sp_addobjectdef | 38 |
| | sp_addremotelogin | 41 |
| | sp_addsegment | 44 |
| | sp_addserver | 46 |
| | sp_addthreshold | 50 |
| | sp_addtype | 54 |
| | sp_addumpdevice | 58 |
| | sp_adduser | 60 |
| | sp_altermessage | 62 |
| | sp_audit | 63 |
| | sp_autoconnect | 69 |

| | |
|---------------------------|-----|
| sp_autoformat | 71 |
| sp_bindcache | 76 |
| sp_bindefault | 80 |
| sp_bindexeclass | 83 |
| sp_bindmsg | 86 |
| sp_bindrule | 88 |
| sp_cacheconfig | 90 |
| sp_cachestrategy | 99 |
| sp_changedbowner | 102 |
| sp_changegroup | 103 |
| sp_checknames | 105 |
| sp_checkreswords | 108 |
| sp_checksouce | 120 |
| sp_chgattribute | 122 |
| sp_cleanpwdchecks | 127 |
| sp_clearpsexex | 128 |
| sp_clearstats | 129 |
| sp_client_addr | 131 |
| sp_clusterlockusage | 133 |
| sp_cluster | 134 |
| sp_cmp_all_qplans | 156 |
| sp_cmp_qplans | 158 |
| sp_commonkey | 160 |
| sp_companion | 162 |
| sp_compatmode | 165 |
| sp_configure | 166 |
| sp_copy_all_qplans | 173 |
| sp_copy_qplan | 174 |
| sp_countmetadata | 175 |
| sp_cursorinfo | 177 |
| sp_dbextend | 180 |
| sp_dboption | 187 |
| sp_dbrecovery_order | 201 |
| sp_dbremap | 203 |
| sp_defaultloc | 204 |
| sp_deletesmobj | 207 |
| sp_depends | 209 |
| sp_deviceattr | 215 |
| sp_diskdefault | 217 |
| sp_displayaudit | 219 |
| sp_displaylevel | 223 |
| sp_displaylogin | 225 |
| sp_displayroles | 232 |
| sp_downgrade | 235 |

| | |
|------------------------------|-----|
| sp_dropalias | 242 |
| sp_drop_all_qplans | 244 |
| sp_drop_qpgroup | 245 |
| sp_drop_qplan | 246 |
| sp_drop_resource_limit | 247 |
| sp_drop_time_range | 250 |
| sp_dropdevice | 251 |
| sp_dropengine | 252 |
| sp_dropexeclss | 254 |
| sp_dropextendedproc | 255 |
| sp_dropexternlogin | 256 |
| sp_dropglockpromote | 257 |
| sp_dropgroup | 258 |
| sp_dropkey | 259 |
| sp_droplanguage | 261 |
| sp_droplogin | 262 |
| sp_dropmessage | 263 |
| sp_dropobjectdef | 264 |
| sp_dropremotelogin | 266 |
| sp_droprowlockpromote | 267 |
| sp_dropsegment | 268 |
| sp_dropserver | 270 |
| sp_dropthreshold | 271 |
| sp_droptype | 272 |
| sp_dropuser | 273 |
| sp_dumpoptimize | 274 |
| sp_encryption | 279 |
| sp_engine | 291 |
| sp_estspace | 295 |
| sp_export_qpgroup | 300 |
| sp_extendsegment | 301 |
| sp_extengine | 303 |
| sp_extrapwdchecks | 304 |
| sp_familylock | 305 |
| sp_find_qplan | 307 |
| sp_fixindex | 309 |
| sp_flushstats | 311 |
| sp_forceonline_db | 312 |
| sp_forceonline_object | 314 |
| sp_forceonline_page | 316 |
| sp_foreignkey | 318 |
| sp_freedll | 320 |
| sp_getmessage | 321 |
| sp_grantlogin | 322 |

| | |
|------------------------------|-----|
| sp_ha_admin | 324 |
| sp_help | 326 |
| sp_help_resource_limit | 334 |
| sp_help_qpgroup | 337 |
| sp_help_qplan | 339 |
| sp_helpapptrace | 341 |
| sp_helppartition | 342 |
| sp_helpcache | 346 |
| sp_helpcomputedcolumn | 348 |
| sp_helpconfig | 349 |
| sp_helpconstraint | 354 |
| sp_helpdb | 358 |
| sp_helpdevice | 362 |
| sp_helpextendedproc | 365 |
| sp_helpexternlogin | 367 |
| sp_helpgroup | 369 |
| sp_helpindex | 371 |
| sp_helpjava | 374 |
| sp_helpjoins | 377 |
| sp_helpkey | 379 |
| sp_helplanguage | 381 |
| sp_helplog | 382 |
| sp_helpobjectdef | 383 |
| sp_helpremotelogin | 384 |
| sp_helpprotect | 385 |
| sp_helpsegment | 390 |
| sp_helpserver | 394 |
| sp_helpsort | 396 |
| sp_helptext | 398 |
| sp_helpthread | 403 |
| sp_helpthreshold | 405 |
| sp_helpuser | 406 |
| sp_hidetext | 408 |
| sp_import_qpgroup | 410 |
| sp_indsuspect | 412 |
| sp_jreconfig | 413 |
| sp_ldapadmin | 423 |
| sp_listener | 430 |
| sp_listsuspect_db | 434 |
| sp_listsuspect_object | 435 |
| sp_listsuspect_page | 436 |
| sp_lmconfig | 437 |
| sp_lock | 441 |
| sp_locklogin | 445 |

| | |
|-----------------------------------|-----|
| sp_logdevice | 448 |
| sp_loginconfig | 450 |
| sp_logininfo | 452 |
| sp_logiosize | 454 |
| sp_logintrigger | 457 |
| sp_maplogin | 458 |
| sp_merge_dup_inline_default | 459 |
| sp_metrics | 462 |
| sp_modify_resource_limit | 464 |
| sp_modify_time_range | 467 |
| sp_modifylogin | 469 |
| sp_modifystats | 470 |
| sp_modifythreshold | 473 |
| sp_monitor | 477 |
| sp_monitorconfig | 485 |
| sp_monitor_server | 492 |
| sp_object_stats | 493 |
| sp_opt_querystats | 496 |
| sp_options | 498 |
| sp_passthru | 507 |
| sp_password | 509 |
| sp_passwordpolicy | 510 |
| sp_pciconfig | 524 |
| sp_placeobject | 528 |
| sp_plan_dbccdb | 530 |
| sp_poolconfig | 532 |
| sp_post_xpload | 537 |
| sp_primarykey | 539 |
| sp_processmail | 540 |
| sp_procxmode | 542 |
| sp_querysmobj | 544 |
| sp_recompile | 546 |
| sp_refit_admin | 548 |
| sp_remap | 550 |
| sp_remoteoption | 551 |
| sp_remotesql | 553 |
| sp_rename | 555 |
| sp_rename_qpgroup | 557 |
| sp_renamedb | 558 |
| sp_reportstats | 560 |
| sp_revokelogin | 562 |
| sp_role | 563 |
| sp_securityprofile | 565 |
| sp_sendmsg | 569 |

| | |
|---------------------------------|-----|
| sp_serveroption | 571 |
| sp_set_qplan | 575 |
| sp_setlangalias | 576 |
| sp_setpglockpromote | 577 |
| sp_setpsexex | 579 |
| sp_setrowlockpromote | 581 |
| sp_setsuspect_granularity | 583 |
| sp_setsuspect_threshold | 586 |
| sp_setup_table_transfer | 588 |
| sp_show_options | 589 |
| sp_showcontrolinfo | 590 |
| sp_showexecclass | 592 |
| sp_showoptstats | 593 |
| sp_showplan | 597 |
| sp_showpsexex | 599 |
| sp_spaceusage | 601 |
| sp_spaceused | 613 |
| sp_ssladmin | 616 |
| sp_syntax | 620 |
| sp_sysmon | 622 |
| sp_tab_suspectptn | 628 |
| sp_tempdb | 629 |
| sp_tempdb_markdrop | 638 |
| sp_thresholdaction | 639 |
| sp_tran_dumpable_status | 641 |
| sp_transactions | 642 |
| sp_unbindcache | 648 |
| sp_unbindcache_all | 651 |
| sp_unbinddefault | 652 |
| sp_unbindexecclass | 654 |
| sp_unbindmsg | 656 |
| sp_unbindrule | 657 |
| sp_version | 659 |
| sp_volchanged | 662 |
| sp_webservices | 665 |
| sp_who | 673 |
| sp_xmlschema | 677 |

| | | |
|--------------|----------------------------|------------|
| 第 2 章 | 目录存储过程 | 679 |
| | 概述 | 679 |
| | 指定可选参数 | 680 |
| | 模式匹配 | 681 |
| | 系统过程表 | 681 |
| | ODBC 数据类型 | 681 |
| | sp_column_privileges | 683 |
| | sp_columns | 685 |
| | sp_databases | 687 |
| | sp_datatype_info | 688 |
| | sp_fkeys | 689 |
| | sp_pkeys | 691 |
| | sp_server_info | 692 |
| | sp_special_columns | 694 |
| | sp_sproc_columns | 696 |
| | sp_statistics | 698 |
| | sp_stored_procedures | 700 |
| | sp_table_privileges | 701 |
| | sp_tables | 702 |
| | | |
| 第 3 章 | 系统扩展存储过程 | 705 |
| | 概述 | 705 |
| | 系统 ESP 的权限 | 706 |
| | 与系统 ESP 关联的 DLL | 706 |
| | 使用系统 ESP | 706 |
| | xp_cmdshell | 707 |
| | xp_deletemail | 710 |
| | xp_enumgroups | 711 |
| | xp_findnextmsg | 712 |
| | xp_logevent | 713 |
| | xp_readmail | 714 |
| | xp_sendmail | 717 |
| | xp_startmail | 720 |
| | xp_stopmail | 721 |

| | | |
|-----------------|----------------------------------|------------|
| 第 4 章 | dbcc 存储过程 | 723 |
| | 概述 | 723 |
| | 指定对象名和日期 | 724 |
| | 指定对象名 | 724 |
| | 指定日期 | 724 |
| | sp_dbcc_alterws | 726 |
| | sp_dbcc_configreport | 727 |
| | sp_dbcc_createws | 728 |
| | sp_dbcc_deletedb | 730 |
| | sp_dbcc_deletehistory | 731 |
| | sp_dbcc_differentialreport | 733 |
| | sp_dbcc_evaluatedb | 735 |
| | sp_dbcc_exclusions | 737 |
| | sp_dbcc_faultreport | 739 |
| | sp_dbcc_fullreport | 744 |
| | sp_dbcc_help_fault | 745 |
| | sp_dbcc_patch_finishtime | 746 |
| | sp_dbcc_recommendations | 748 |
| | sp_dbcc_runcheck | 749 |
| | sp_dbcc_statisticsreport | 750 |
| | sp_dbcc_summaryreport | 753 |
| | sp_dbcc_updateconfig | 756 |
| 索引 | | 759 |

系统过程

本章介绍各个系统过程，它们是 Sybase[®] 提供的、用于更新和获取系统表报告的存储过程。

涉及的主题包括：

| 主题 | 页码 |
|---------|----|
| 系统过程简介 | 1 |
| 系统过程的权限 | 2 |
| 审计系统过程 | 2 |
| 执行系统过程 | 3 |
| 输入参数值 | 3 |
| 消息 | 4 |
| 系统过程表 | 5 |

系统过程简介

系统过程是在安装时由 `installmaster` 创建的。它们位于 `sybssystemprocs` 数据库中，并由系统管理员拥有。使用 `sp_version` 可以确定最近运行的 `installmaster` 版本。

某些系统过程只能在特定数据库中运行，但多数系统过程都能在任意数据库中运行。可以自行创建从任意数据库中都能执行的系统过程。有关详细信息，请参见《系统管理指南》。

所有系统过程都在隔离级别 1 上执行。

所有系统过程都会报告返回状态。以下示例表示过程已成功执行：

```
return status = 0
```

本书中的示例不包括返回状态。

从 Adaptive Server[®] 12.5.1 版开始，在存储过程中可以声明多达 10,000 个变量。在早期的 Adaptive Server 版本中，该限值为 2,000。

系统过程的返回值

存储过程将报告一个**返回状态**，该返回状态指示存储过程是否成功完成，如果没有，将报告失败原因。有关系统过程返回值的详细信息，请参见《Transact-SQL 用户指南》的第 16 章“使用存储过程”中的“返回值”。

系统过程的权限

系统过程的权限在 `sybsystemprocs` 数据库中设置。

某些系统过程只能由数据库所有者来运行。这些过程可以确保执行过程的用户是从中执行这些过程的数据库的所有者。

其它系统过程（例如，所有 `sp_help` 过程）可由任何被授予权限的用户来执行，只要该权限是在 `sybsystemprocs` 中授予的权限。用户要么在所有数据库中都具有执行某个系统过程的权限，要么在任何数据库中都不具有这一权限。

未在 `sybsystemprocs..sysusers` 中列出的用户在 `sybsystemprocs` 中被视为“`guest`”用户，并被自动授予多个系统过程的权限。

要拒绝向用户授予某个系统过程的权限，系统管理员必须将此用户添加到 `sybsystemprocs..sysusers` 中，并编写用于此过程的 `revoke` 语句。用户数据库的所有者不能从自己的数据库内部直接控制该系统过程的权限。

审计系统过程

通常，您可以通过启用审计选项“`exec_procedure`”来审计存储过程的执行，该选项将生成包含存储过程名称和参数的审计记录。

执行系统过程

如果从 `sybssystemprocs` 以外的数据库中执行某个系统过程，该过程则会针对执行它的数据库中的系统表进行操作。例如，如果 `pubs2` 的数据库所有者在 `pubs2` 中运行 `sp_adduser`，新用户将添加到 `pubs2..sysusers` 中。

若要在特定数据库中运行系统过程，请执行以下操作之一：

- 使用 `use` 命令打开该数据库并执行该过程，或者，
- 用数据库名限定过程名。

例如，用户定义的系统过程 `sp_foo`（执行 `db_name` 系统函数）将返回在其中执行该过程的数据库的名称。当从 `pubs2` 数据库执行时，它返回“pubs2”：

```
exec pubs2..sp_foo
-----
pubs2
(1 row affected, return status = 0)
```

而在 `sybssystemprocs` 中执行时，它返回“sybssystemprocs”：

```
exec sybssystemprocs..sp_foo
-----
sybssystemprocs
(1 row affected, return status = 0)
```

输入参数值

如果系统过程的参数值包含标点符号或嵌入的空格，或者该值是保留字，则必须用单引号或双引号引起来。如果参数是由数据库名或所有者名限定的对象名，则整个名称必须用单引号或双引号括起来。

注释 不要将分隔标识符用作系统过程参数；它们可能导致意外结果。

如果过程有多个可选参数，则可以采用以下形式提供参数，而不用提供所有参数：

```
@parametername = value
```

语法语句中的参数名称要与过程定义的参数名称匹配。

例如，`sp_addlogin` 的语法如下：

```
sp_addlogin login_name, password [, defdb  
            [, deflanguage [, fullname]]]
```

要使用 `sp_addlogin` 创建 “susan” 登录名（口令为 “wonderful”，全名为 Susan B. Anthony，使用服务器缺省数据库和语言），则可以使用：

```
sp_addlogin susan, wonderful, @fullname="Susan B. Anthony"
```

该命令提供的信息与以下指定了所有参数的命令所提供的信息相同：

```
sp_addlogin susan, wonderful, public_db, us_english, "Susan B. Anthony"
```

也可以用 “null” 作为占位符：

```
sp_addlogin susan, wonderful, null, null, "Susan B. Anthony"
```

不要用引号将 “null” 引起来。

SQL 中没有规定每一行中的单词数量或者必须折行的位置。如果在发出系统过程后发出命令，Adaptive Server 将尝试先执行系统过程，然后再执行命令。例如，如果执行以下命令，Adaptive Server 会先返回 `sp_help` 的输出结果，然后再运行 `checkpoint` 命令：

```
sp_help checkpoint
```

如果指定的参数个数比系统过程需要的参数多，Adaptive Server 会忽略多余的参数。

消息

系统过程返回信息性消息和错误消息，这些消息随本书中的各个过程一同列出。系统过程错误消息的错误号从 17000 开始。

过程中包含的函数和命令的错误消息在《故障排除和错误消息指南》中说明。

系统过程表

系统过程使用 master 数据库中的几个**系统过程表**（如 `spt_values`、`spt_committab`、`spt_monitor` 和 `spt_limit_types`），以便将内部系统值（例如，状态位）转换为人工可读的格式。

`spt_values` 从不更新。若要查看该表如何使用，请执行 `sp_helptext`，查看引用该表的系统过程之一的文本。

此外，某些系统过程会先创建临时表，然后再删除它们。

sp_activeroles

- 说明** 显示所有活动角色。
- 语法** sp_activeroles [expand_down]
- 参数** expand_down
显示您的角色中包含的所有活动角色的层次树。
- 示例** **示例 1** 显示所有活动角色。

```
sp_activeroles

Role Name
-----
sa_role
sso_role
oper_role
replication_role
```

示例 2 显示活动角色及其角色层次树：

```
sp_activeroles expand_down

Role Name          Parent Role Name          Level
-----
sa_role            NULL                      1
doctor_role       NULL                      1
oper_role          NULL                      1
```

- 用法** sp_activeroles 显示所有活动角色和这些角色中包含的所有角色。
- 权限** 任何用户都可以执行 sp_activeroles。
- 审计** sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关创建、管理和使用角色的信息，请参见《系统管理指南》。

命令 [alter role](#)、[create role](#)、[drop role](#)、[grant](#)、[revoke](#)、[set](#)

函数 [mut_excl_roles](#)、[proc_role](#)、[role_contain](#)、[role_name](#)

系统过程 [sp_displayroles](#)

sp_add_qpgroup

- 说明** 添加抽象计划组。
- 语法** `sp_add_qpgroup new_name`
- 参数** *new_name*
是新抽象计划组的名称。组名必须是有效的标识符。
- 示例** 创建名为 `dev_plans` 的新抽象计划组：
`sp_add_qpgroup dev_plans`
- 用法**
- 使用 `sp_add_qpgroup` 添加抽象计划组以用于捕获或创建抽象计划。抽象计划组必须存在，之后您才可以创建、保存计划或将计划复制到组中。
 - 不能在事务中运行 `sp_add_qpgroup`。
- 权限** 只有系统管理员或数据库所有者才能执行 `sp_add_qpgroup`。
- 审计** sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

- 另请参见
- 命令** [set](#)
 - 系统过程** [sp_help_qpgroup](#)

sp_add_resource_limit

| | |
|----|---|
| 说明 | 创建 Adaptive Server 登录和 / 或应用程序执行查询、查询批处理或事务时可以使用服务器资源的数目限制。 |
| 语法 | <pre>sp_add_resource_limit name, appname, rangename, limittype, limitvalue [, enforced [, action [, scope]]]</pre> |
| 参数 | <p><i>name</i></p> <p>是对其应用限制的 Adaptive Server 登录名。必须指定 <i>name</i> 或 <i>appname</i>，或指定两者。若要创建应用于特定应用程序所有用户的限制，请将 <i>name</i> 指定为 NULL。</p> <p><i>appname</i></p> <p>是对其应用限制的应用程序名称。必须指定 <i>name</i> 或 <i>appname</i>，或指定两者。若要创建应用于 Adaptive Server 登录名使用的所有应用程序的限制，请指定 <i>appname</i> 为 null。若要创建应用于特定应用程序的限制，请指定客户程序在登录包中传递给 Adaptive Server 的应用程序的名称。</p> <p><i>rangename</i></p> <p>是强制执行限制的时间范围。该时间范围在您创建限制时必须存在于 master 数据库的 <i>systranges</i> 系统表中。</p> <p><i>limittype</i></p> <p>是要限制的资源的类型。必须是以下值之一：</p> |

| 限制类型 | 说明 |
|--------------|-----------------------------|
| row_count | 限制一个查询可以返回的行数 |
| elapsed_time | 限制查询批处理或事务可以运行的秒数（按挂钟时间） |
| io_cost | 限制处理查询的实际开销或优化程序的开销估计 |
| tempdb_space | 限制单个会话过程中 tempdb 数据库可以具有的页数 |

limitvalue

是在 Adaptive Server 强制执行限制之前登录名或应用程序可以使用的服务器资源（I/O 开销、以秒计算的经历时间、行计数或 tempdb 空间）的最大量。这必须是一个小于或等于 2^{31} 的非零正整数。下表列出了应为每种限制类型指定的值：

| 限制类型 | 限制值 |
|--------------|-----------------------------------|
| row_count | 在强制执行限制前查询可以返回的最大行数。 |
| elapsed_time | 在强制执行限制前查询批处理或事务可以运行的秒数（按挂钟时间计算）。 |
| io_cost | 一个通过优化程序的开销计算公式得出的无单位的数字。 |
| tempdb_space | 每个会话在 tempdb 中使用的页数。 |

enforced

确定是在执行查询前还是执行查询时强制应用限制。下表列出了每种限制类型的有效值：

| 强制代码 | 说明 | 限制类型 |
|------|-------------------------------------|--------------------------------------|
| 1 | 当执行的估计 I/O 开销超过了指定的限制时执行操作。 | io_cost |
| 2 | 当执行的实际行数、经历时间或 I/O 开销超过了指定的限制时执行操作。 | row_count elapsed_time io_cost |
| 3 | 当估计开销或实际开销超过指定的限制时执行操作。 | io_cost |

如果指定 **enforced** 值为 3，Adaptive Server 执行 1 和 2 的逻辑“或”操作。例如，假定将 **enforced** 设置为 3。如果所运行查询的 **io_cost** 超过了估计的开销，则将执行指定的 **action**。如果查询在为估计开销指定的限制内但超过了实际开销，则也将执行指定的 **action**。

如果未指定 **enforced** 值，则 Adaptive Server 对 **row_count** 和 **elapsed_time** 强制执行限制 2，对 **io_cost** 强制执行限制 3。换句话说，如果限制类型是 **io_cost**，则不管查询超过了估计开销还是实际开销都会执行指定的操作。

action

是在超出限制时采取的操作。以下操作代码对所有限制类型都有效：

| action 代码 | 说明 |
|------------------|---------|
| 1 | 发出警告 |
| 2 | 中止查询批处理 |
| 3 | 中止事务 |
| 4 | 注销会话 |

如果未指定 **action** 值，则 Adaptive Server 使用缺省值 2（中止查询批处理）。

scope

是限制的作用域。指定以下适用于限制类型的代码之一：

| scope 代码 | 说明 | 限制类型 |
|-----------------|-------------------------------|----------------------|
| 1 | 查询 | io_cost row_count |
| 2 | 查询批处理（客户端发送到服务器的一个或多个 SQL 语句） | elapsed_time |
| 4 | 事务 | elapsed_time |
| 6 | 查询批处理和事务 | elapsed_time |

如果未指定 `scope` 值，则限制适用于限制类型所有可能的作用域。

示例

示例 1 创建在 `early_morning` 时间范围内应用于 `payroll` 应用程序所有用户的资源限制。如果查询批处理的执行时间超过 120 秒，则 Adaptive Server 将发出警告：

```
sp_add_resource_limit NULL, payroll, early_morning, elapsed_time, 120, 2, 1, 2
```

示例 2 创建在 `midday` 时间范围内应用于由 “`joe_user`” 运行的所有即席查询和应用程序的资源限制。当查询返回的行数超过 5000 行时，Adaptive Server 中止事务：

```
sp_add_resource_limit joe_user, NULL, midday, row_count, 5000, 2, 3, 1
```

示例 3 创建在 `midday` 时间范围内应用于由 “`joe_user`” 运行的所有即席查询和应用程序的资源限制。当优化程序估计 I/O 开销将超过 650 时，Adaptive Server 中止事务：

```
sp_add_resource_limit joe_user, NULL, midday, io_cost, 650, 1, 3, 1
```

用法

- 您必须启用 `sp_configure` "allow resource limits" 以使资源限制生效。
- 给定的用户、应用程序、限制类型、作用域和强制时间可以存在多个资源限制，但前提是这些资源限制的时间范围不重叠。
- 对登录名和 / 或应用程序名的当前活动的指定时间范围和 “所有时间” 范围的所有限制都会在登录时绑定到用户的会话。因此，如果用户独立于给定的应用程序登录到 Adaptive Server，则不适用同时限制用户与该应用程序的资源限制。若要确保对该用户的限制，请创建一个特定于该用户并独立于任何应用程序的资源限制。
- 由于确定资源限制要使用用户登录名或应用程序名（或者使用两者），所以 Adaptive Server 在扫描 `sysresourcelimits` 表以搜索适用于登录会话的限制时，遵守预定义的搜索优先级。下表说明了匹配的有序登录名和应用程序名对的优先级：

| 级别 | 登录名 | 应用程序名 |
|----|------------|---------|
| 1 | “joe_user” | payroll |
| 2 | NULL | payroll |
| 3 | “joe_user” | NULL |

如果在给定的优先级别中发现了一个或多个匹配项，则不会搜索其它级别。这样可防止用于不同登录 / 应用程序组合的类似限制之间出现冲突。

如果在所有级别都没有发现匹配项，则不对会话进行限制。

- 当添加、删除或修改资源限制时， Adaptive Server 在该登录和 / 或应用程序每个会话下一个查询批处理开始时重新绑定该会话的限制。
- 当更改当前活动的时间范围时， Adaptive Server 重新绑定该会话的限制。此重新绑定发生在下一个查询批处理开始时。
- 您不能将特定登录、应用程序或登录 / 应用程序组合的限制与重叠的指定时间范围关联（共享相同时间范围的限制除外）。

例如，如果限制某个用户在上午 9:00 到下午 1:00 间检索 50 行，则不能对同一个用户创建另一个限制他在上午 10:00 到中午 12:00 间检索 100 行的资源限制。但是您可以创建一个资源层次，例如指定用户在上午 10:00 到中午 12:00 间检索 100 行的限制，并指定应用程序（如 isql）在上午 9:00 到下午 1:00 间检索 50 行的限制。

注释 虽然在当前事务达到其时间限制时 Adaptive Server 会终止该事务，但在您发出另一个 SQL 命令或批处理之前，不会收到 1105 错误消息；换句话说，只有当您再次尝试使用该连接时才显示该消息。

权限

只有系统管理员才能执行 sp_add_resource_limit。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关资源限制的详细信息，请参见《系统管理指南》。

系统过程 [sp_configure](#)、[sp_drop_resource_limit](#)、[sp_help_resource_limit](#)、[sp_modify_resource_limit](#)

实用程序 isql

sp_add_time_range

| | |
|----|---|
| 说明 | 将指定的时间范围添加到 Adaptive Server。 |
| 语法 | <code>sp_add_time_range name, startday, endday, starttime, endtime</code> |
| 参数 | <p>name 是时间范围的名称。时间范围名称必须小于或等于 255 个字符。该名称不能已存在于 <code>master</code> 数据库的 <code>sysstimeranges</code> 系统表中。</p> <p>startday 指定时间范围从星期几开始。它必须是缺省服务器语言的完整的星期日期名称，此数据存储在 <code>master</code> 数据库的 <code>syslanguages</code> 系统表中。</p> <p>endday 指定时间范围到星期几结束。它必须是缺省服务器语言的完整的星期日期名称，此数据存储在 <code>master</code> 数据库的 <code>syslanguages</code> 系统表中。 endday 在星期内可以在 startday 之前或之后，也可以与 startday 相同。</p> <p>starttime 是时间范围开始的那一天中的时间。以 24 小时制指定 starttime，值介于“00:00”（午夜）到“23:59”（晚上 11:59）之间。使用如下格式： "HH:MM"</p> <p>endtime 是时间范围结束的那一天中的时间。以 24 小时制指定 endtime，值介于“00:00”（午夜）到“23:59”（晚上 11:59）之间。使用如下格式： "HH:MM"</p> |

注释 若要创建跨全天的时间范围，请将开始时间和结束时间都指定为“00:00”。

endtime 在一天中的发生时间必须晚于 **starttime**，除非 **endtime** 是 00:00。

示例 **示例 1** 创建 `business_hours` 时间范围，该时间范围从星期一到星期五，每天上午 9:00 到下午 5:00 处于活动状态：

```
sp_add_time_range business_hours, monday, Friday, "09:00", "17:00"
```

示例 2 创建两个时间范围 `before_hours` 和 `after_hours`，这两个时间范围一起跨越从星期一到星期五的所有非工作时间。`before_hours` 时间范围涵盖星期一到星期五从午夜 12:00 到上午 9:00 这段时间。`after_hours` 时间范围涵盖星期一到星期五从下午 6:00 到午夜 12:00 这段时间：

```
sp_add_time_range before_hours, Monday, Friday, "00:00", "09:00"
```

```
sp_add_time_range after_hours, Monday, Friday, "18:00", "00:00"
```

示例 3 创建 `weekends` 时间范围，该时间范围从星期六午夜 12:00 到星期天午夜 12:00:

```
sp_add_time_range weekends, Saturday, Sunday, "00:00", "00:00"
```

示例 4 创建 `Fri_thru_Mon` 时间范围，该时间范围包括星期五、星期六、星期天和星期一从上午 9:00 到下午 5:00 这段时间:

```
sp_add_time_range Fri_thru_Mon, Friday, Monday, "09:00", "17:00"
```

示例 5 创建 `Wednesday_night` 时间范围，该时间范围从星期三下午 5:00 到午夜 12:00:

```
sp_add_time_range Wednesday_night, Wednesday, Wednesday, "17:00", "00:00"
```

用法

- **Adaptive Server** 包括一个指定的时间范围，即“所有时间”时间范围。此时间范围涵盖所有时间，从星期中的第一天到最后一天，从 00:00 到 23:59。不能对其进行修改或将其删除。
- **Adaptive Server** 为每个指定的时间范围生成一个唯一 ID 号并将其插入到 `systemranges` 系统表中。
- 当在 `systemranges` 系统表中存储时间范围时，**Adaptive Server** 会将其 `startday` 和 `endday` 值转换为整数。对于缺省语言为美国英语 (`us_english`) 的服务器，每个星期从星期一（第 1 天）开始到星期天（第 7 天）结束。
- 可以创建与一个或多个其它时间范围重叠的时间范围。
- 范围中的每一天都是连续的，因此星期中最后一天接在星期第一天之前。换句话说，星期天和星期一是连续的两天，就像星期二和星期三一样。
- 每次查询批处理开始时，活动的时间范围都会被捆绑到一个会话。由于实际时间更改而导致的服务器活动时间范围的更改在查询批处理期间不会对会话产生影响。换句话说，如果资源限制在给定的时间范围内对查询批处理进行限制，但某个查询批处理是在此时间范围变为活动状态之前开始的，则该资源限制对已在运行的这一查询批处理不会产生影响。
- 使用系统过程添加、修改和删除时间范围不影响当前正在进行的会话的活动时间范围。
- 如果资源限制的作用域是某一事务，而在事务运行时，服务器的活动时间范围发生了更改，则新的活动时间范围不会影响当前正在进行的事务。
- 对作用域为事务的资源限制进行更改不会影响任何当前正在进行的事务。
- 有关时间范围的详细信息，请参见《系统管理指南》。

权限 只有系统管理员才能执行 `sp_add_time_range`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_add_resource_limit](#)、[sp_drop_time_range](#)、[sp_modify_time_range](#)

sp_addalias

| | |
|----|--|
| 说明 | 允许 Adaptive Server 用户在数据库中以另一个用户的名义操作。 |
| 语法 | <code>sp_addalias loginname, name_in_db</code> |
| 参数 | <p><i>loginname</i> 是在当前数据库中需要替代标识的用户的 <code>master.dbo.syslogins</code> 名称。</p> <p><i>name_in_db</i> 是要作为 <i>loginname</i> 的别名的数据库用户名。该名称必须存在于 <code>master.dbo.syslogins</code> 和当前数据库的 <code>sysusers</code> 表中。</p> |
| 示例 | <p>数据库的 <code>sysusers</code> 表中有一个名为 “albert” 的用户，而 <code>master.dbo.syslogins</code> 中有一个名为 “victoria” 的用户登录名。此命令允许 “victoria” 以 “albert” 的名义使用当前数据库：</p> <pre>sp_addalias victoria, albert</pre> |
| 用法 | <ul style="list-style-type: none">• 执行 <code>sp_addalias</code> 可以将一个用户映射为当前数据库中的另一个用户。映射关系显示在 <code>sysalternates</code> 中，其中这两个用户的 <code>suids</code>（系统用户 ID）是相连的。• 一次只能将某用户的别名指向一个数据库用户。• 使用 <code>sp_helpuser</code> 并将指定用户名作为参数，即可为映射到指定用户的所有用户生成报告。• 在用户尝试使用数据库时，Adaptive Server 会检查 <code>sysusers</code>，确认该用户是否列在其中。如果用户未列在其中，Adaptive Server 则检查 <code>sysalternates</code>。如果用户的 <code>suid</code> 列在映射到某个数据库用户的 <code>suid</code> 的 <code>sysalternates</code> 中，Adaptive Server 在使用该数据库时会将第一个用户视为第二个用户。 <p>如果在 <i>loginname</i> 中指定的用户存在于数据库的 <code>sysusers</code> 表中，Adaptive Server 不会使用该用户的别名标识，因为它会先检查 <code>sysusers</code> 并查找 <i>loginname</i>，然后才检查列有别名的 <code>sysalternates</code>。</p> |
| 权限 | 只有数据库所有者、系统管理员和系统安全员才能执行 <code>sp_addalias</code> 。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 [use](#)

系统过程 [sp_addlogin](#)、[sp_adduser](#)、[sp_dropalias](#)、[sp_helpuser](#)

sp_addauditrecord

| | |
|----|--|
| 说明 | 允许用户在审计追踪中输入用户定义的审计记录（注释）。 |
| 语法 | <code>sp_addauditrecord [text [, db_name [, obj_name [, owner_name [, dbid [, objid]]]]]]</code> |
| 参数 | <p>text 是添加到当前审计表的消息的文本。这些文本将被插入该表的 <code>extrainfo</code> 字段。</p> <p>db_name 是记录所引用的数据库的名称。该名称将被插入当前审计表的 <code>dbname</code> 字段。</p> <p>obj_name 是记录所引用的对象的名称。该名称将被插入当前审计表的 <code>objname</code> 字段。</p> <p>owner_name 是记录所引用的对象的所有者。该名称将被插入当前审计表的 <code>objowner</code> 字段。</p> <p>dbid 是 <code>db_name</code> 的数据库 ID 号。不要用引号引起这个整数值。<code>dbid</code> 将被插入当前审计表的 <code>dbid</code> 字段。</p> <p>objid 是 <code>obj_name</code> 的对象 ID 号。不要用引号引起这个整数值。<code>objid</code> 将被插入当前审计表的 <code>objid</code> 字段。</p> |
| 示例 | <p>示例 1 在当前审计表中，将 “I gave A. Smith permission to view the payroll table in the corporate database. This permission was in effect from 3:10 to 3:30 pm on 9/22/92” 添加到 <code>extrainfo</code> 字段中；将 “corporate” 添加到 <code>dbname</code> 字段中；将 “payroll” 添加到 <code>objname</code> 字段中；将 “dbo” 添加到 <code>objowner</code> 字段中；将 “10” 添加到 <code>dbid</code> 字段中，并将 “1004738270” 添加到 <code>objid</code> 字段中：</p> <pre>sp_addauditrecord "I gave A. Smith permission to view the payroll table in the corporate database. This permission was in effect from 3:10 to 3:30 pm on 9/22/92.", "corporate", "payroll", "dbo", 10, 1004738270</pre> <p>示例 2 将此记录添加到审计追踪中。本例使用带 <code>@</code> 前缀的参数名，以便将某些字段留空：</p> <pre>sp_addauditrecord @text="I am disabling auditing briefly while we reconfigure the system", @db_name="corporate"</pre> |

- 用法
- Adaptive Server 将所有审计记录都写入当前审计表。当前审计表由 `current audit table` 配置参数的值决定，而该参数是用 `sp_configure` 设置的。一个安装系统至多可有八个系统审计表，分别名为 `sysaudits_01`、`sysaudits_02` 直到 `sysaudits_08`。

注释 记录实际上最先存储在内存中的审计队列中，审计进程稍后才将这些记录从审计队列写入当前审计表。因此，不能指望审计记录会立即存储到审计表中。

- 在下列情况下可使用 `sp_addauditrecord`:
 - 已被授予针对 `sp_addauditrecord` 的执行权限 — 无需任何特殊角色
 - 已启用审计 — 系统安全员使用 `sp_configure` 启用了 `auditing` 配置参数
 - `sp_audit` 的 `adhoc` 选项被设置为 `on`

权限 只有系统安全员才能执行 `sp_addauditrecord`。`sybsecurity` 的数据库所有者（必须也是系统安全员）可以向其他用户授予执行权限。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 1 | adhoc | 用户定义的审计记录 | 用 <code>sp_addauditrecord</code> 的 <code>text</code> 参数填充 <code>extrainfo</code> |
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_audit](#)

sp_addauditable

| | |
|----|--|
| 说明 | 安装审计功能后，再添加一个系统审计表。 |
| 语法 | <code>sp_addauditable devname</code> |
| 参数 | <p><i>devname</i></p> <p>是审计表的设备名。指定一个设备名或指定“default”。如果指定“default”，Adaptive Server 会在 <code>sybsecurity</code> 数据库所在的设备上创建审计表。否则，Adaptive Server 将在指定的设备上创建该表。</p> |
| 示例 | <p>示例 1 在 <code>auditdev2</code> 上创建系统审计表。如果执行该过程时只存在一个系统审计表 (<code>sysaudits_01</code>)，Adaptive Server 会将新审计表命名为 <code>sysaudits_02</code> 并将其放在其自身的段上，该段名为 <code>aud_seg_02</code>，位于 <code>auditdev2</code> 上：</p> <pre>sp_addauditable auditdev2</pre> <p>示例 2 在 <code>sybsecurity</code> 数据库所在的设备上创建系统审计表。如果执行该过程时存在两个系统审计表 (<code>sysaudits_01</code> 和 <code>sysaudits_02</code>)，Adaptive Server 会将新审计表命名为 <code>sysaudits_03</code> 并将其放在其自身的段上，该段名为 <code>aud_seg_03</code>，位于 <code>sybsecurity</code> 数据库所在的设备上：</p> <pre>sp_addauditable "default"</pre> |
| 用法 | <ul style="list-style-type: none"> • 运行 <code>sp_addauditable</code> 时必须已经安装了审计功能。添加系统审计表： <ol style="list-style-type: none"> a 使用 <code>disk init</code> 为审计表创建设备。例如，在 UNIX 中，可以运行如下命令： <pre>disk init name = "auditdev2", physname = "/dev/rxyla", size = "5K"</pre> b 使用 <code>alter database</code> 命令将设备添加到 <code>sybsecurity</code> 数据库中。例如，若要将 <code>auditdev2</code> 添加到 <code>sybsecurity</code> 数据库中，请使用： <pre>alter database sybsecurity on auditdev2</pre> c 执行 <code>sp_addauditable</code> 创建该表。 • Adaptive Server 根据已经定义的审计表数来命名新的系统审计表和新段。例如，如果执行该过程之前已经定义了五个审计表，Adaptive Server 会将新审计表命名为 <code>sysaudits_06</code> 并将新段命名为 <code>aud_seg_06</code>。如果指定“default”，Adaptive Server 会将该段放在 <code>sybsecurity</code> 数据库所在的设备上。如果不这样指定，Adaptive Server 会将该段放在指定的设备上。 • 最多允许创建八个审计表。如果已有八个审计表还试图执行 <code>sp_addauditable</code> 添加其它审计表，Adaptive Server 会显示错误消息。 |

- 有关如何安装审计功能的信息，请参见所用平台的安装文档。有关如何使用审计功能的信息，请参见《系统管理指南》。

权限

您必须既是系统管理员又是系统安全员才能执行 `sp_addauditable`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_audit](#)

sp_addengine

| | |
|-------------|--|
| 说明 | sp_addengine 不在线程化模式中运行。 |
| 有关进程模式的考虑事项 | 向现有引擎组添加引擎，如果引擎组不存在，则创建引擎组并添加引擎。 |
| 语法 | sp_addengine engine_number, engine_group [, instance_id] |
| 参数 | <p>engine_number 是要向引擎组添加的引擎数。合法值最小为 0，最大值是配置的联机引擎数减 1。</p> <p>engine_group 是要将引擎添加到的引擎组的名称。如果 engine_group 不存在，Adaptive Server 将创建引擎组并向其添加引擎。引擎组名称必须符合标识符的规则。有关详细信息，请参见《参考手册：构件块》中的第 4 章“表达式、标识符和通配符”。</p> <p>instance_id 在集群环境中—要向其添加引擎或引擎组的实例的 ID。</p> |
| 示例 | <p>示例 1 如果没有名为 DS_GROUP 的引擎组，以下语句会建立该组。如果 DS_GROUP 已存在，以下语句会向该组添加 2 号引擎：</p> <pre>sp_addengine 2, DS_GROUP</pre> <p>示例 2 向实例 ID 8 添加 5 号引擎：</p> <pre>sp_addengine 5, 8</pre> |
| 用法 | <ul style="list-style-type: none"> • 如果 engine_group 的值尚不存在，sp_addengine 会创建新的引擎组。 • 如果 sp_cluster set system_view 设置为 cluster，则可以向集群中的任意实例添加引擎或引擎组。如果 system_view 设置为 instance，则只能向本地实例添加引擎或引擎组。 • 引擎组 ANYENGINE 和 LASTONLINE 是预定义的。ANYENGINE 包含所有现有引擎。LASTONLINE 指定引擎号最大的引擎。系统管理员可以创建更多引擎组。用户不能修改预定义引擎组。 • 一使用 sp_bindexclass 将应用程序或登录绑定到与 engine_group 关联的执行类，关联进程就可以在 engine_number 上开始运行。 • sp_engine 可以在您使用 sp_procxmode 将事务模式更改为 anymode 后在使用链式事务的会话中运行。 • 指派引擎密切连接之前，应研究环境并考虑非首选应用程序数和可用的 Adaptive Server 引擎数。有关非首选应用程序的详细信息，请参见《性能和调优指南》。 |

权限 只有系统管理员才能执行 `sp_addengine`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** `sp_addexclass`、`sp_bindexclass`、`sp_clearpsex`、`sp_dropengine`、`sp_setpsex`、`sp_showcontrolinfo`、`sp_showexclass`、`sp_showpsex`、`sp_unbindexclass`

sp_addexeclass

| | |
|-------------|--|
| 说明 | 创建或更新可以绑定到客户端应用程序、登录和存储过程的用户定义执行类。 |
| 有关进程模式的考虑事项 | 预定义的引擎组参数 ANYENGINE 和 LASTONLINE 仅在进程模式中有效。 |
| 语法 | <code>sp_addexeclass classname, priority, timeslice, engine_group [, instance_id]</code> |
| 参数 | <p>classname 是新执行类的名称。</p> <p>priority 是优先级值，客户端应用程序、登录名或存储过程与此执行类建立关联后，要依据该优先级值来运行。合法值为 HIGH、LOW 和 MEDIUM。</p> <p>timeslice 是指派给与此类关联的进程的时间单位。Adaptive Server 当前忽略此参数。</p> <p>engine_group 标识可以运行与此类关联的进程的现有引擎组。</p> <p>instance_id (在集群环境中) 要绑定用户定义执行类的实例的 ID。</p> |
| 示例 | <p>示例 1 定义名为 DS 的新执行类（此执行类的 <i>priority</i> 值为 LOW），并使其与引擎组 DS_GROUP 关联：</p> <pre>sp_addexeclass "DS", "LOW", 0, "DS_GROUP"</pre> <p>示例 2 在集群环境中— 定义名为 DS 的新执行类（此执行类的 <i>priority</i> 值为 LOW），并使其与 8 号实例上的引擎组 DS_GROUP 关联，请输入：</p> <pre>sp_addexeclass "DS", "LOW", 0, "DS_GROUP", 8</pre> |
| 用法 | <ul style="list-style-type: none"> • sp_addexeclass 创建或更新可以绑定到客户端应用程序、登录和存储过程的用户定义执行类。如果该类已经存在，类属性值将被更新为用户提供的值。 • 当您在线程化模式中运行 sp_addexeclass 时，Adaptive Server 会将 engine_group 用于线程池的名称。 • (在集群环境中) 如果 sp_cluster set system_view 设置为 cluster，则可以在集群中的任意实例上添加执行类。如果 system_view 设置为 instance，则只能向本地实例添加执行类。 • 如果不想让执行对象受限于引擎组，可以使用预定义引擎组参数 ANYENGINE。 |

- 使用 [sp_addengine](#) 可以定义引擎组。使用 [sp_showexeclass](#) 可以显示执行类属性以及与指定执行类关联的所有引擎组中的引擎。
[sp_showcontrolinfo](#) 列出了现有引擎组。

权限

只有系统管理员才能执行 [sp_addexeclass](#)。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_addengine](#)、[sp_bindexeclass](#)、[sp_clearpsexec](#)、[sp_dropengine](#)、[sp_dropexeclass](#)、[sp_setpsexec](#)、[sp_showcontrolinfo](#)、[sp_showexeclass](#)、[sp_showpsexec](#)、[sp_unbindexeclass](#)

sp_addextendedproc

| | |
|----|--|
| 说明 | 在 master 数据库中创建扩展存储过程 (ESP)。 |
| 语法 | <code>sp_addextendedproc esp_name, dll_name</code> |
| 参数 | <p>esp_name 是扩展存储过程的名称。此名称必须与实现 ESP 的过程语言函数的名称相同。esp_name 必须是一个有效的 Adaptive Server 标识符。</p> <p>dll_name 是 esp_name 指定的函数所在的动态链接库 (DLL) 文件的名称。指定 dll_name 时可以不带扩展名, 也可以带有平台特定的扩展名, 如在 Windows NT 上扩展名为 <i>.dll</i>, 在 Sun Solaris 上扩展名为 <i>.so</i>。如果指定了扩展名, 必须用引号将 dll_name 引起来。</p> |
| 示例 | <p>为名为 my_esp 的函数注册 ESP, 该函数位于 <i>sqlsrvdll.dll</i> 文件中。生成的 ESP 数据库对象的名称也是 my_esp:</p> <pre>sp_addextendedproc my_esp, "sqlsrvdll.dll"</pre> |
| 用法 | <ul style="list-style-type: none"> • 从 master 数据库中执行 sp_addextendedproc。 • 只能使用 sp_addextendedproc 添加不带参数的扩展存储过程。如果扩展存储过程需要正式的参数列表, 则必须使用带 as external name 选项的 create procedure 命令, 并提供完整的参数列表。 • esp_name 区分大小写。它必须与 DLL 中的函数名称匹配。 • dll_name 表示的 DLL 必须位于创建 ESP 的服务器计算机上, 并且 DLL 目录必须位于以下位置: <ul style="list-style-type: none"> • Windows NT — <i>\$PATH</i> • Compaq Tru64 — <i>\$LD_LIBRARY_PATH</i> • HP — <i>\$SH_LIBRARY_PATH</i> <p>如果未找到该文件, 搜索机制还会搜索 <i>\$\$SYBASE/dll</i> (在 Windows NT 上) 和 <i>\$\$SYBASE/lib</i> (在其它平台上)。</p> • (在 Windows NT 上) ESP 函数不应调用 C 运行时信号例程。这会导致 XP Server 失败, 因为 Open Server 不支持 Windows NT 上的信号处理。 |
| 权限 | 只有系统管理员才能执行 sp_addextendedproc 。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create procedure](#)

系统过程 [sp_dropextendedproc](#)、[sp_helpextendedproc](#)

sp_addexternlogin

说明 (仅适用于组件集成服务) 创建替代的登录帐号和口令, 以便在通过组件集成服务与远程服务器通信时使用。

语法 sp_addexternlogin server, loginame, externname
[, externpasswd] [rolename]

参数 server

是远程服务器的名称。remote_server 必须通过 master.dbo.syssservers 表中的条目为本地服务器所知。

loginame

是本地服务器识别的帐户。必须用 master.dbo.syslogins 表中的条目来表示 loginame。只有使用 “sa”、“sso” 和 loginame 帐户的用户有权修改给定本地用户的远程访问权限。

externname

是 server 已知的帐户, 它必须是运行 server 的节点上的有效帐户。这是用于登录 server 的帐户。

externpasswd

是 externname 的口令。

rolename

是 Adaptive Server 用户指派的角色。如果指定 rolename, login_name 将被忽略。

示例 **示例 1** 通知本地服务器当登录名 “bobj” 登录时, 通过远程名 “jordan” 和远程口令 “hitchpost” 来访问远程服务器 OMNI1012。对于登录名 “bobj”, 只有 “bobj” 帐户、“sa” 帐户和 “sso” 帐户有权添加或修改远程登录:

```
sp_addexternlogin OMNI1012, bobj, jordan, hitchpost
```

示例 2 显示多对一映射, 以便为所有需要连接 DB2 的 Adaptive Server Enterprise 用户指派相同的名称和口令:

```
sp_addexternlogin DB2, NULL, login2, password2
```

示例 3 也可以为 Adaptive Server Enterprise 角色指派远程登录。使用此功能, 可以为拥有特定角色的任何人指派给定远程服务器的相应登录名和口令:

```
sp_addexternlogin DB2, NULL, login3, password3, role
```

用法

- `sp_addexternlogin` 指派在与远程服务器通信时要使用的替代登录名和口令。它以加密的形式将口令存储在内部。

注释 只有配置了组件集成服务后才能使用 `sp_addexternlogin`。

- 映射可以采用以下形式：一对一（对于特定用户）、角色对一（基于角色）、多对一（基于服务器），或基于 TDS `loginrec` 的客户端登录名和口令。
- 登录名和口令具有多对一映射关系。这表示，可以为所有要登录远程服务器的用户指派相同的名称和口令。
- 为某个用户设置多个外部登录名时，在用户连接远程服务器时将使用以下优先顺序。1) 一对一映射；2) 如果没有一对一映射，则使用活动角色；3) 如果既没有一对一映射，也没有活动角色，则使用多对一映射；4) 如果未使用上述任一种映射，则使用 Adaptive Server Enterprise 登录名和口令。
- 可以将外部登录名指派给 Adaptive Server 角色。可以为拥有特定角色的任何人指派任意给定远程服务器的相应登录名和口令。
- 为有多个活动角色的用户建立与远程服务器的连接时，将搜索每个角色的外部登录映射，并使用找到的第一个映射建立登录。这与存储过程 `sp_activeRoles` 所显示的顺序相同。
- 如果执行角色映射，并且更改了用户的角色（使用 `set role`），则必须断开任何使用角色映射与远程服务器建立的连接。有尚未完成的事务时无法执行该操作。如果有活动事务并且存在使用角色映射的远程连接，则无法使用 `set role`。
- 运行 `sp_addexternlogin` 之前，使用 `sp_addserver` 将远程服务器添加到 Adaptive Server 中。
- `externname` 和 `externpasswd` 在运行 `server` 的节点上必须是有效的用户名和口令组合。
- 设置自动口令失效的站点需要就定期更新外部登录名的口令作出计划。
- 使用 `sp_dropexternlogin` 可以删除外部登录名的定义。
- `sp_addexternlogin` 不能在事务内部使用。
- 只有拥有“sa”帐户和 `loginame` 帐户的用户才能修改给定本地用户的远程访问权限。

权限

只有 `loginame`、系统管理员和系统安全员才能执行 `sp_addexternlogin`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addserver](#)、[sp_dropexternlogin](#)、[sp_helpexternlogin](#)、[sp_helpserver](#)

sp_addgroup

说明 向数据库添加组。在授予和撤消特权时可使用组作为集合名称。

语法 `sp_addgroup grpname`

参数 `grpname`

是组的名称。组名必须符合标识符的规则。

示例 在当前数据库中创建名为 `accounting` 的组：

```
sp_addgroup accounting
```

用法

- `sp_addgroup` 将新组添加到数据库的 `sysusers` 表中。每个组的用户 ID (`uid`) 都为 16384 或更大 (“`public`” 除外，其用户 ID 始终为 0)。
- 组与用户不能具有相同的名称。
- 一旦创建了组，即可使用 `sp_adduser` 添加新用户。若要向组添加现有用户，请使用 `sp_changegroup`。
- 每个数据库在创建后都有一个名为 “`public`” 的组。每个用户都自动成为 “`public`” 的成员。每个用户还可以是另一个组的成员。

权限 只有数据库所有者、系统管理员和系统安全员才能执行 `sp_addgroup`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[revoke](#)

系统过程 [sp_adduser](#)、[sp_changegroup](#)、[sp_dropgroup](#)、[sp_helpgroup](#)

sp_addlanguage

| | |
|----|--|
| 说明 | 定义替代语言的月份和星期日期的名称及其日期格式。 |
| 语法 | <code>sp_addlanguage language, alias, months, shortmons, days, datefmt, datefirst</code> |
| 参数 | <p><i>language</i> 是该语言正式的语言名称，必须以 7 位 ASCII 字符的形式输入。</p> <p><i>alias</i> 可用于替换替代语言的正式名称。可输入 “null”（别名与正式语言名称相同）或所需的名称。如果您的终端支持，则可以在别名中使用 8 位 ASCII 字符，例如 “français”。</p> <p><i>months</i> 是 12 个月份全名的列表，从 1 月到 12 月顺次排序，只能用逗号分隔（不允许使用空格）。月份名称最多可以有 20 个字符并可以包含 8 位 ASCII 字符。</p> <p><i>shortmons</i> 是 12 个月份缩写名称的列表，从 1 月到 12 月顺次排序，只能用逗号分隔（不允许使用空格）。月份缩写最多可以有 9 个字符并可以包含 8 位 ASCII 字符。</p> <p><i>days</i> 是每周 7 天全名的列表，从星期一到星期天顺次排序，只能用逗号分隔（不允许使用空格）。星期日期名称最多可以有 30 个字符并可以包含 8 位 ASCII 字符。</p> <p><i>datefmt</i> 是输入 <code>datetime</code>、<code>smalldatetime</code>、<code>date</code> 或 <code>time</code> 数据时日期分量 <code>month/day/year</code> 的日期顺序。有效参数是 <code>mdy</code>、<code>dmy</code>、<code>ymd</code>、<code>ydm</code>、<code>myd</code> 或 <code>dym</code>。“<code>dmy</code>”表示日期顺序为日/月/年。</p> <p><i>datefirst</i> 设置每星期第一天的数字以进行日期计算。例如，星期一是 1，星期二是 2，依此类推。</p> |
| 示例 | <p>下面的存储过程将法语添加到服务器上可用的语言中。“null”表示别名与正式名称 “french” 相同。日期顺序是 “dmy” 一日/月/年。“1”指定 <i>days</i> 列表中的第一项 <code>lundi</code> 是每星期的第一天。因为法语中只有当星期日期和月份的名称出现在句首时才将它们大写，所以此示例显示它们在被添加时为小写：</p> <pre>sp_addlanguage french, null, "janvier,fevrier,mars,avril,mai,juin,juillet, aout,septembre,octobre,novembre,decembre",</pre> |

```

"jan, fev, mars, avr, mai, juin, jui, aout, sept, oct,
nov, dec",
"lundi, mardi, mercredi, jeudi, vendredi, samedi,
dimanche",
dmy, 1

```

用法

- 通常，使用 `langinstall` 实用程序或 Adaptive Server 安装程序从某个 Adaptive Server 语言模块添加替代语言。语言模块提供该语言的日期名称和经过翻译的错误消息。但是，如果您的服务器未提供某个语言模块，请使用 `sp_addlanguage` 定义日期名称和格式。
- 使用 `sp_modifylogin` 更改用户的缺省语言。如果将用户的缺省语言设置为某种使用 `sp_addlanguage` 添加的语言，但是没有该语言的本地化文件，则用户在登录时将收到一条信息性消息，指出他们的客户端软件无法打开本地化文件。

系统表更改

- `sp_addlanguage` 在 `master.dbo.syslanguages` 中创建一个条目，在每个替代语言的 `langid` 列中插入唯一数值。`langid 0` 是为美国英语保留的。
- `language` 参数成为正式语言名称，存储在 `master.dbo.syslanguages` 的 `name` 列中。语言名称必须是唯一的。使用 `sp_helplanguage` 显示 Adaptive Server 上可用的替代语言的列表。
- 如果为 `alias` 输入了 `NULL`，则 `sp_addlanguage` 将 `master.dbo.syslanguages` 中的 `alias` 列设置为正式的语言名称，但是系统管理员可以通过 `sp_setlangalias` 更改 `syslanguage.alias` 的值。
- `sp_addlanguage` 将 `master.dbo.syslanguages` 中的 `upgrade` 列设置为 0。

通过 `sp_addlanguage` 添加的语言的日期

- 对于通过语言模块添加的替代语言，Adaptive Server 将日期值以 `datetime` 数据类型形式发送到客户端，客户端使用本地化文件以用户的当前语言显示日期。对于通过 `sp_addlanguage` 添加的日期字符串，使用 `convert` 函数将日期转换为服务器中的字符数据，其中 `pubdate` 是 `datetime` 数据，`table` 是任意表：

```
select convert(char, pubdate) from table
```

- 当用户对日期值执行数据输入并且需要使用通过 `sp_addlanguage` 创建的日期名称时，客户端必须以字符数据形式输入这些值，并将它们以字符数据形式发送到服务器。

权限

只有系统管理员才能执行 `sp_addlanguage`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [set](#)

系统过程 [sp_droplanguage](#)、[sp_helplanguage](#)、[sp_modifylogin](#)、[sp_setlangalias](#)

实用程序 [langinstall](#)

sp_addlogin

| | |
|----|---|
| 说明 | Adaptive Server 15.7 和更高版本不支持此系统过程。若要向 Adapter Server 中添加登录帐户，请使用 <code>create login</code> 命令。 |
| 语法 | 无 |
| 参数 | 无 |
| 用法 | 无 |

sp_addmessage

说明 将用户定义的消息添加到 `sysusermessages` 以供存储过程 `print` 和 `raiserror` 调用以及 `sp_bindmsg` 使用。

语法 `sp_addmessage message_num, message_text
[, language [, with_log [, replace]]]`

参数 *message_num*
是要添加的消息的消息号。用户定义的消息的消息号必须大于等于 20000。

message_text
是要添加的消息的文本。最大长度为 1024 个字节。

language
是要添加的消息的语言。这必须是 `syslanguages` 表中有效的语言名称。如果缺少此参数，Adaptive Server 便假定这些消息使用 `@@langid` 指定的缺省会话语言。

with_log
指定如果启用了记录，是否在 Adaptive Server 错误日志以及 Windows NT 服务器上的 Windows NT 事件日志中记录消息。如果 *with_log* 为 TRUE，则不管错误的严重性如何，都将记录消息。如果 *with_log* 为 FALSE，则是否记录消息取决于错误的严重性。如果未指定 *with_log* 的值，则缺省值为 FALSE。

replace
指定是否覆盖具有相同编号和 *langid* 的现有消息。如果指定了 *replace*，则将覆盖现有消息；如果省略了 *replace*，则不覆盖现有消息。如果未指定 *replace* 的值，则该参数的缺省行为指定不覆盖现有消息。

示例 **示例 1** 将编号为 20001 的消息添加到 `sysusermessages`：

```
sp_addmessage 20001, "The table '%1!' is not owned by the user '%2!'."
```

示例 2 将编号为 20002 的消息添加到 `sysusermessages`。如果启用了事件记录，则将在 Adaptive Server 错误日志以及 Windows NT 服务器上的 Windows NT 事件日志中记录此消息。如果编号为 20002 的消息使用缺省会话语言，则此消息将覆盖以前的消息：

```
sp_addmessage 20002, "The procedure '%1!' is not owned  
by the user '%2!'." , NULL, TRUE, "replace"
```

- 用法
- 除非您指定 `@replace = "replace"`，否则 `sp_addmessage` 不覆盖具有相同编号和 `langid` 的现有消息。
 - `print` 和 `raiserror` 识别要输出的消息文本中的占位符。一条消息最多可包含 20 个任意顺序的、各不相同的占位符。消息的文本发送到客户端时，这些占位符将被替换为消息后面任意参数的格式化内容。

为实现在 Adaptive Server 将消息翻译为具有不同语法结构的语言时可以对参数进行重新排序，需要对占位符进行编号。参数的占位符显示为 “%nn!”，即一个百分比符号 (%) 后跟一个从 1 到 20 的整数，然后跟感叹号 (!)。整数表示参数列表中字符串中的参数编号。“%1!” 是原始版本中的第一个参数，“%2!” 是第二个参数，依此类推。

- 只有创建消息的用户才能执行带 `replace` 选项的 `sp_addmessage` 来替换该原始消息。

权限 任何用户都可以执行 `sp_addmessage`。但是，只有创建消息的用户才能执行带 `replace` 选项的 `sp_addmessage` 来替换该原始消息。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|---------------|---|
| 15 | create | sp_addmessage | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 消息号 代理信息 — set proxy 有效时的初始登录名 |
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** `print`、`raiserror`

系统过程 `sp_altermessage`、`sp_bindmsg`、`sp_dropmessage`、`sp_getmessage`

sp_addobjectdef

| | |
|----|--|
| 说明 | (仅限组件集成服务) 指定本地表和外部存储位置之间的映射。 |
| 语法 | <code>sp_addobjectdef tablename, objectdef [, "objecttype"]</code> |
| 参数 | <p>tablename</p> <p>是对象的名称 (如本地表中所定义的)。<i>tablename</i> 可以是以下任意一种形式:</p> <pre> dbname.owner.object dbname..object owner.object object </pre> <p><i>dbname</i> 和 <i>owner</i> 是可选的。<i>object</i> 是必需的。如果未指定 <i>owner</i>, 则使用缺省名称 (当前用户名)。如果指定了 <i>dbname</i>, 则它必须是当前数据库名称, 并且您必须指定 <i>owner</i> 或用 <i>dbname..object</i> 格式的占位符标记所有者。用引号将任何由多个部分组成的 <i>tablename</i> 值引起来。</p> <p>objectdef</p> <p>是一个指定对象外部存储位置的字符串。<i>objectdef</i> 中的 <i>objecttype</i> 可以是远程服务器可以访问的表、视图或只读远程过程调用 (RPC) 结果集。对于 <i>objectdef</i>, 表、视图或 RPC 使用以下格式:</p> <pre> server_name.dbname.owner.object </pre> <p><i>server_name</i> 和 <i>object</i> 是必需的。<i>dbname</i> 和 <i>owner</i> 是可选的, 但是如果未提供, 则必须使用 <i>dbname..object</i> 格式的占位符。</p> <p>有关详细信息, 请参见《组件集成服务用户指南》中的“服务器类”。</p> <p>objecttype</p> <p>是指定 <i>objectdef</i> 所确定对象的格式的值之一。表 1-1 描述了有效的值。应将 <i>objecttype</i> 值放在引号内。</p> |

表 1-1: objecttype 允许使用的值

| 值 | 说明 |
|-------|---|
| table | 指示 <i>objectdef</i> 所指定的对象是一个远程服务器可访问的表。此值是 <i>objecttype</i> 的缺省值。 |
| view | 指示 <i>objectdef</i> 所指定的对象是一个由远程服务器管理的视图, 它被作为表进行处理。 |
| rpc | 指示 <i>objectdef</i> 所指定的对象是一个由远程服务器托管的 RPC。Adaptive Server 将 RPC 的结果集作为只读表进行处理。 |

表 1-2 总结如何使用每个 *objecttype*。表 1-2: *objecttype* 用法总结

| <i>objecttype</i> | <i>create table</i> | <i>create existing table</i> | 向表写入 | 从表读取 |
|-------------------|---------------------|------------------------------|------|------|
| table | 是 | 是 | 是 | 是 |
| view | 否 | 是 | 是 | 是 |
| rpc | 否 | 是 | 否 | 是 |

示例

示例 1 将数据库 *finance* 中的本地表 *accounts* 映射到名为 SYBASE 的远程服务器中的远程对象 *pubs.dbo.accounts*。当前数据库必须是 *finance*。随后的 *create table* 在 *pubs* 数据库中创建一个表。如果 *pubs.dbo.accounts* 是现有的表，则 *create existing table* 语句用有关远程表的信息填充表 *finance.dbo.accounts*：

```
sp_addobjectdef "finance.dbo.accounts", "SYBASE.pubs.dbo.accounts", "table"
```

示例 2 将本地表 *stockcheck* 映射到名为 *stockcheck* 的 RPC，该 RPC 在远程服务器 NEWYORK 上所有者为 “kelly” 的数据库 *wallstreet* 中。来自 RPC *stockcheck* 的结果集被视为一个只读表。通常，下一个操作将是对对象 *stockcheck* 执行 *create existing table* 语句：

```
sp_addobjectdef stockcheck, "NEWYORK.wallstreet.kelly.stockcheck", "rpc"
```

用法

- *sp_addobjectdef* 指定本地表和外部存储位置之间的映射。它标识位于该位置的对象的格式。只有当安装并配置了组件集成服务后才能使用 *sp_addobjectdef*。
- *sp_addobjectdef* 取代了 *sp_addtabledef* 命令。*sp_addobjectdef* 允许现有的脚本运行而无需修改。在内部，*sp_addtabledef* 调用 *sp_addobjectdef*。
- 只有系统管理员才能提供另一个用户的名称作为表所有者。
- 当 *objecttype* 是 *table*、*view* 或 *rpc* 时，*objectdef* 参数采用以下形式：

```
"server_name.database.owner.tablename"
```

 - *server_name* — 表示已被 *sp_addserver* 添加到 *sys.servers* 的服务器。
 - *database* — 可能不是必需的。某些服务器类不支持它。
 - *owner* — 应总是提供它来避免歧义。如果未指定 *owner*，所引用的远程对象可能有所变化，具体情况取决于外部登录名是否与远程对象所有者对应。
 - *tablename* — 是远程服务器表的名称。

- 请在发出任何 [create table](#) 或 [create existing table](#) 命令之前使用 sp_addobjectdef。但是，如果存在远程表，您就不必在执行 [create proxy_table](#) 之前使用 sp_addobjectdef。

[create table](#) 仅对 *objecttype* 值 [table](#) 和 [file](#) 有效。当使用 [create table](#) 或 [create existing table](#) 时，Adaptive Server 检查 *sysattributes* 以确定是否已为该对象指定了任何表映射。跟在 *objecttype* 值 [view](#) 和 [rpc](#) 后面的是 [create existing table](#) 语句。

- 在创建完表之后，以后对本地表名称的所有引用（通过 [select](#)、[insert](#)、[delete](#) 和 [update](#)）都将被映射到正确的位置。

权限

任何用户都可以执行 sp_addobjectdef。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create existing table](#)、[create table](#)、[drop table](#)

系统过程 [sp_addlogin](#)、[sp_addserver](#)、[sp_defaultloc](#)、[sp_dropobjectdef](#)、[sp_helpserver](#)

sp_addremotelogin

说明 通过将条目添加到 `master.dbo.sysremotelogins` 中对新远程服务器用户进行授权。

语法 `sp_addremotelogin remoteserver [, loginame [, remotename]]`

参数 *remoteserver*

是对其应用远程登录名的远程服务器的名称。此服务器必须通过 `master.dbo.sysservers` 表中的条目为本地服务器所知，该条目是通过 [sp_addserver](#) 创建的。

注释 本手册页使用“本地服务器”一词指执行在“远程服务器”中运行的远程过程的服务器。

loginame

是本地服务器上用户的登录名。*loginame* 必须已存在于 `master.dbo.syslogins` 表中。

remotename

是远程服务器在登录到本地服务器时所使用的名称。所有未与一个本地 *loginame* 显式匹配的 *remotenames* 将自动与一个本地名称匹配。在示例 1 中，本地名称是用于登录的远程名称。在示例 2 中，本地名称是“albert”。

示例 **示例 1** 在 `sysremotelogins` 表中为远程服务器 GATEWAY 创建一个条目以进行登录验证。当本地和远程服务器具有相同的用户时，这是一种将远程名称映射到本地名称的简便方法：

```
sp_addremotelogin GATEWAY
```

本示例将在 `sysremotelogins` 中创建一行，其 `suid` 列为 -1，`remoteusername` 为 NULL。

示例 2 创建一个条目，该条目将远程服务器 GATEWAY 中的所有登录名映射到本地用户名“albert”。Adaptive Server 会将一行添加到 `sysremotelogins` 中，其 `suid` 列是 Albert 的服务器用户 ID，`remoteusername` 是空值：

```
sp_addremotelogin GATEWAY, albert
```

若要让这些登录名能够在本地服务器上运行 RPC，在登录到本地服务器时，它们必须为 RPC 连接指定一个口令，或者它们必须在本地服务器上为“trusted”。若要将这些登录名定义为“trusted”，请使用 [sp_remotoption](#)。

示例 3 将远程服务器 GATEWAY 上远程用户 “pogo” 的远程登录名映射到本地用户 “ralph”。Adaptive Server 会将一行添加到 `sysremotelogins` 中，其 `suid` 列是 Ralph 的服务器用户 ID，`remoteusername` 列是 “pogo”：

```
sp_addremotelogin GATEWAY, ralph, pogo
```

用法

- 当收到远程登录名时，本地服务器以三种不同的方式尝试将远程用户映射到本地用户：
 - 首先，本地服务器在 `sysremotelogins` 中查找与远程服务器名和远程用户名匹配的行。如果本地服务器找到匹配的行，则将使用该行的本地服务器用户 ID 登录远程用户。这适用于来自指定远程用户的映射。
 - 如果未找到匹配的行，则本地服务器将搜索具有空远程名称并且本地服务器用户 ID 不为 -1 的行。如果找到这样的行，则将远程用户映射到该行中的本地服务器用户 ID。这适用于远程服务器中任何远程用户到特定本地名称的映射。
 - 最后，如果前面的尝试都已失败，则本地服务器查看 `sysremotelogins` 表中是否存在具有空远程名称并且本地服务器用户 ID 为 -1 的条目。如果找到这样的行，则本地服务器使用远程服务器提供的远程名称在 `syslogins` 表中查找本地服务器用户 ID。这适用于远程服务器和本地服务器中的登录名相同的情况。
- 本地用户的名称在远程服务器上可能有所不同。
- 如果使用 `sp_addremotelogin` 将远程服务器的所有用户映射到相同的本地名称，则可以使用 `sp_remotoption` 为这些用户指定 “trusted” 选项。例如，如果服务器 GOODSRV 中映射到 “albert” 的所有用户都将是 “trusted”，则按如下方式使用 `sp_remotoption`：

```
sp_remotoption GOODSRV, albert, NULL, trusted, true
```

未指定为 “trusted” 的登录名不能在本地服务器上执行 RPC，除非它们在登录远程服务器时指定本地服务器的口令。在 Open Client™ Client-Library™ 中，用户可以使用 `ct_remote_pwd` 例程为服务器到服务器的连接指定口令。isql 和 bcp 不允许用户为 RPC 连接指定口令。

如果用户使用 “统一登录” 登录远程服务器，则这些登录名已通过安全性机制的鉴定。这些登录名在本地服务器上也必须是受信任的，否则用户在登录远程服务器时必须指定服务器的口令。

- 每个远程登录名条目都具有一个状态。trusted 选项的缺省状态是 false（不受信任）。这意味着，当远程登录名使用该条目时，将对口令进行检查。如果您不想检查口令，请使用 `sp_remotoption` 将 trusted 选项的状态更改为 true。

权限

只有系统管理员才能执行 `sp_addremotelogin`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关为远程过程调用和为使用“统一登录”而设置服务器的详细信息，请参见《系统管理指南》。

系统过程 [sp_addlogin](#)、[sp_addserver](#)、[sp_droptremotelogin](#)、[sp_helptremotelogin](#)、[sp_helprotect](#)、[sp_helpserver](#)、[sp_remotoption](#)

实用程序 isql

sp_addsegment

说明 定义数据库中数据库设备上的段。

语法 `sp_addsegment segname, dbname, devname`

参数 *segname*

是要添加到数据库的 `syssegments` 表中的新段的名称。段名在每个数据库中都是唯一的。

dbname

指定在其中定义段的数据库的名称。 *dbname* 必须是当前数据库的名称或符合限定 `sp_addsegment` 的数据库名称。

devname

是在其中定位 *segname* 的数据库设备的名称。数据库设备可以有多个与之关联的段。

示例 **示例 1** 在名为 `dev1` 的数据库设备上为数据库 `pubs2` 创建一个名为 `indexes` 的段:

```
sp_addsegment indexes, pubs2, dev1
```

示例 2 在名为 `pubs2_dev` 的数据库设备上为数据库 `pubs2` 创建一个名为 `indexes` 的段:

```
disk init
    name = "pubs2_dev",
    physname = "/dev/pubs_2_dev",
    vdevno = 9, size = 5120
go
alter database pubs2 on pubs2_dev = 2
go
pubs2..sp_addsegment indexes, pubs2, dev1
```

用法

- 不能在已经具有排它段的设备上创建段。如果试图这样做，则会显示类似以下内容的错误消息:

```
A segment with a virtually hashed table exists on
device orders_dat.
```

- `sp_addsegment` 为数据库设备定义段名，这些数据库设备通过 `disk init` 创建并通过 `alter database` 或 `create database` 命令被分配到特定数据库。
- 在定义了段之后，在 `create table` 和 `create index` 命令以及 `sp_placeobject` 过程中使用它在段上放置一个表或索引。

当在特定段上创建了表或索引后，该表或索引所有随后的数据都位于该段上。

- 使用系统过程 `sp_extendsegment` 可以将段的范围扩展到同一个数据库所使用的另一个数据库设备。
- 如果使用 `alter database` 在数据库使用的设备上对该数据库进行扩展，则还将对映射到该设备的段进行扩展。
- `system` 和 `default` 段会被映射到 `create database` 或 `alter database` 命令中包括的每个数据库设备。`logsegment` 也会被映射到每个设备，除非您使用 `create database` 的 `log on` 扩展或使用 `sp_logdevice` 将其放在单独的设备上。有关详细信息，请参见《系统管理指南》。
- 虽然您可以在同一个设备上既有数据又有日志的数据库中使用 `sp_addsegment`（例如在不使用 `log on` 选项创建数据库时），但如果使用以下语句创建数据库，Adaptive Server 将返回一条错误消息：

```
create database dbname on devicename log on devicename with override
```

权限

只有数据库所有者或系统管理员才能执行 `sp_addsegment`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `alter database`、`create index`、`create table`、`disk init`

系统过程 `sp_dropsegment`、`sp_extendsegment`、`sp_helpdb`、`sp_helpdevice`、`sp_placeobject`

sp_addserver

说明 定义远程服务器或定义本地服务器的名称；指定使用 `host` 和 `port` 参数时用于远程过程调用 (RPC) 的服务器。

语法 `sp_addserver lname [, class [, pname]]`

仅用于组件集成服务 (CIS):

```
sp_addserver 'logical_server_name', ASEnterprise, 'host:port:filter'
```

参数

lname

是用于指系统上的服务器的名称。如果尚不存在 *lname* 的条目，则 `sp_addserver` 会在 `sys.servers` 表中添加一行。服务器名必须是唯一的并且必须符合标识符的规则。

class

标识要添加的服务器的类别。服务器 *class* 为 “null” 表示服务器为缺省类别 “ASEnterprise”。*class* 参数的允许值:

- `local` — 本地服务器（只能有一个）在启动或重新启动 Adaptive Server 后只能使用一次，以标识本地服务器名称以便它可以显示在 Adaptive Server 输出的消息中
- `null` — 未定义任何类别的远程服务器。
- `ASEnterprise` — 所有版本的 Adaptive Server Enterprise；未提供对 SQL Server 4.9 的支持。
- `ASAnywhere` — Adaptive Server Anywhere 6.0 版或更高版本。
- `ASIQ` — 具有服务器类 ASIQ 的服务器是 Adaptive Server IQ 12.0 版或更高版本。
- `direct_connect`（仅用于组件集成服务）— 一个基于 Open Server 的应用程序，符合 `direct_connect` 接口规范。
- `sds` — 符合《Adaptive Server Specialty Data Store 开发人员工具包》手册中描述的 Specialty Data Store™ 接口要求。

有关这些参数值的详细信息，请参见《组件集成服务用户指南》中第 2 章“了解组件集成服务”中的“远程服务器”。

注释 Adaptive Server 不支持服务器类 `db2`。若要使用 `db2`，请将 `db2` 服务器类迁移到 `direct_connect` 类。

pname

是名为 *lname* 的服务器在 `interfaces` 文件中的名称。这使您可以为需要与之通讯的其它 Adaptive Server 或 Backup Server 建立本地别名。如果未指定 *pname*，则将使用 *lname*。

(仅用于组件集成服务) 您可以使用 *pname* 指定主机名或 IP 地址以及要连接的服务器的端口。这使在使用 CT-Library 时无需为服务器提供目录服务 (例如 LDAP 或 `interfaces` 文件)。使用以下格式:

```
"hostname:port"
"ipaddr:port"
```

注释 您必须用单引号或双引号将主机名和端口引起来以使用此选项。

filter

在集群环境中 — 添加用于远程过程调用 (RPC) 的远程服务器。

```
filter = ssl [= 'CN = common_name']
```

使用以下格式声明 *host:port* 编号:

```
ip_address:port
```

示例

示例 1 (在集群环境中) 添加名为 "big_logical_server" 的远程服务器。

```
sp_addserver 'big_logical_server', ASEntprise,
'maynard:23954:ssl= "CN=ase1.big server 1.com"
```

用于公用名的规则与用于动态监听器和目录服务条目的规则相同。

示例 2 在 `master.dbo.sysservers` 中为名为 GATEWAY 的远程服务器添加一个条目。 *pname* 也是 GATEWAY:

```
sp_addserver GATEWAY
```

示例 3 在 `master.dbo.sysservers` 中为名为 GATEWAY 的远程服务器添加一个条目。 *pname* 是 VIOLET。如果 GATEWAY 已存在具有不同 *pname* 的 `sysservers` 条目，则服务器 GATEWAY 的 *pname* 将更改为 VIOLET:

```
sp_addserver GATEWAY, null, VIOLET
```

示例 4 为名为 PRODUCTION 的本地服务器添加一个条目:

```
sp_addserver PRODUCTION, local
```

示例 5（仅适用于组件集成服务）为具有主机名“myhost”和端口号 10224 的远程 Adaptive Server 添加一个条目：

```
sp_addserver S1, ASEnterprise, "myhost:10224"
```

注释 如果对 *pname* 使用此语法，则 Adaptive Server 站点处理器不能成功连接到此服务器；只有 CIS 连接识别 *pname* 的这种语法。

示例 6（仅适用于组件集成服务）为具有主机 IP 192.123.456.010 和端口号 11222 的远程 Adaptive Server 添加一个条目：

```
sp_addserver S3, direct_connect, "192.123.456.010:11222"
```

用法

- **syssservers** 表标识本地服务器的名称及其选项，以及本地服务器可与之通信的任何远程服务器。
若要对远程服务器执行远程过程调用，远程服务器必须存在于 **syssservers** 表中。
- 如果 *lname* 已作为 **syssservers** 表中的服务器名存在，则 **sp_addserver** 将远程服务器的 *srvnetname* 更改为 *pname* 指定的名称。当它执行此操作时，**sp_addserver** 报告它更改的服务器、以前的网络名和新网络名。
- 服务器的安装或升级过程在 **syssservers** 中为 Backup Server 添加一个条目。如果删除了此条目，就不能备份数据库了。
- Adaptive Server 要求 Backup Server 的 *lname* 为 SYB_BACKUP。如果您不想使用该名称作为 Backup Server 的名称，或者如果您的系统上运行了多个 Backup Server，则用 **sp_addserver** 修改服务器 SYB_BACKUP 的 *pname*，以便 Adaptive Server 可以与 Backup Server 进行通讯以进行数据库转储和装载。
- 如果指定的 *lname*、*pname* 和 *class* 已存在于 **syssservers** 中，则 **sp_addserver** 将输出一条错误消息并且不更新 **syssservers**。
- 使用 **sp_serveroption** 设置或清除服务器选项。

权限

只有系统安全员才能执行 **sp_addserver**。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 《组件集成服务用户指南》。

系统过程 [sp_addremotelogin](#)、[sp_droptremotelogin](#)、[sp_dropserver](#)、[sp_helpremotelogin](#)、[sp_helpserver](#)、[sp_serveroption](#)

sp_addthreshold

说明 创建一个阈值以监控数据库段上的空间。当段上的可用空间低于指定水平时， Adaptive Server 执行关联的存储过程。

语法 `sp_addthreshold dbname, segname, free_space, proc_name`

参数

dbname

是为其添加阈值的数据库。它必须是当前数据库的名称。

segname

是监控其可用空间的段。指定 “default” 段时要使用引号。

free_space

是达到阈值时的可用页数。当段中的可用空间低于此水平时， Adaptive Server 执行关联的存储过程。

proc_name

是当 *segname* 上的可用空间量降到 *free_space* 之下时执行的存储过程。该过程可以位于当前 Adaptive Server 或某个 Open Server 上的任何数据库中。阈值不能在远程 Adaptive Server 上执行过程。

示例

示例 1 为 `segment1` 创建一个阈值。当 `segment1` 上的可用空间降到 200 页以下时， Adaptive Server 执行过程 `pr_warning`：

```
sp_addthreshold mydb, segment1, 200, pr_warning
```

示例 2 为 `user_data` 段创建一个阈值。当 `user_data` 上的可用空间降到 100 页以下时， Adaptive Server 对 Open Server `mail_me` 过程执行远程过程调用：

```
sp_addthreshold userdb, user_data, 100, "o_server...mail_me"
```

示例 3 在 `pubs2` 数据库的 `indexes` 段上创建一个阈值。您可以从任何数据库发出此命令：

```
pubs2..sp_addthreshold pubs2, indexes, 100, pr_warning
```

用法

达到阈值

- 达到阈值时， Adaptive Server 执行关联的存储过程。 Adaptive Server 对阈值过程使用以下搜索路径：
 - 如果过程名称不指定数据库， Adaptive Server 将在达到阈值的数据库中查找。
 - 如果在该数据库中未找到该过程，并且过程名称以 “sp_” 开头， Adaptive Server 将在 `sybssystemprocs` 数据库中查找。

如果在两个数据库中都没有找到该过程，则 Adaptive Server 向错误日志发送一条错误消息。

- Adaptive Server 使用一个**停滞值**（即全局变量 `@@thresh_hysteresis`）来确定阈值对可用空间变化的敏感程度。一旦阈值执行其过程，阈值将失效。在段中的可用空间量超过阈值 `@@thresh_hysteresis` 页之前，阈值一直处于不活动状态。这样可防止因响应可用空间的微小波动而使阈值重复执行其过程。

最后机会阈值

- 缺省情况下，Adaptive Server 监控日志所在的段上的可用空间并在可用空间量少于允许成功转储事务日志所必需的空间时执行 `sp_thresholdaction`。这一可用空间量（称为**最后机会阈值**）是 Adaptive Server 计算出来的，用户不能对其进行更改。
- 如果在记录某个事务前达到了最后机会阈值，Adaptive Server 将挂起该事务直到日志空间被释放。使用 `sp_dboption` 为特定数据库更改此行为。`sp_dboption "abort tran on log full", true` 使 Adaptive Server 在达到最后机会阈值时回退所有尚未记录的事务。
- 包括 master 在内的所有数据库都有一个最后机会阈值。该阈值是对备份事务日志所需的可用日志页数的估计值。为日志段分配更多的空间时，Adaptive Server 会自动调整最后机会阈值。

创建附加阈值

- 每个数据库最多可以有 256 个阈值，其中包括最后机会阈值。
- 当添加阈值时，它与最近阈值的间隔至少应该是 `@@thresh_hysteresis` 页数的两倍。

创建阈值过程

- 具有 `create procedure` 权限的任何用户都可以在数据库中创建阈值过程。通常，系统管理员在 `sybsystemprocs` 数据库中创建 `sp_thresholdaction`，数据库所有者在用户数据库中创建阈值过程。
- `sp_addthreshold` 不检验指定的过程是否存在。在创建阈值执行的过程前添加阈值是可能的。
- `sp_addthreshold` 检查以确保添加阈值过程的用户已被直接授予“sa_role”。创建阈值过程时所有活动的系统角色都将被作为编写阈值的用户的有效角色输入 `systhresholds` 中。但是，触发阈值时，将只激活直接授予的系统角色。不激活间接授予的系统角色和用户定义的角色。
- Adaptive Server 向阈值过程传递四个参数：
 - `@dbname, varchar(30)`，它标识数据库
 - `@segmentname, varchar(30)`，它标识段
 - `@space_left, int`，它指示与阈值关联的可用页的数目

- `@status, int`，它具有一个值，对于最后机会阈值该值为 1，对于其它阈值该值为 0

按位置而不是按名称传递这些参数；阈值过程可为这些参数使用其它名称，但它必须按以上顺序并用正确的数据类型声明这些参数。

- 没有必要为每个阈值创建一个不同的过程。为了减少维护工作，您可以在 `sybssystemprocs` 数据库中创建一个阈值过程，为 Adaptive Server 中的所有阈值执行该过程。
- 在阈值过程中包括 `print` 和 `raiserror` 语句以可输出发送到错误日志。

执行阈值过程

- 达到阈值时启动的任务将作为后台任务执行。这些任务没有关联的终端会话或用户会话。如果您在这些任务运行时执行 `sp_who`，`status` 列将显示 “background”。
- Adaptive Server 执行阈值过程所使用的权限是用户添加阈值时所具有的权限减去从那之后被撤消的任何权限。
- 每个阈值过程使用一个用户连接，使用时间为执行过程的时间。

更改或删除阈值

- 使用 `sp_helptreshold` 可以获得有关现有阈值的信息。
- 使用 `sp_modifythreshold` 可以将阈值与新阈值过程、可用空间值或段关联起来。（您不能更改与最后机会阈值关联的可用空间值或段名。）
每次用户修改阈值时，该用户就会成为阈值所有者。当达到阈值时，Adaptive Server 执行阈值所使用的权限是所有者修改阈值时所具有的权限减去从那之后被撤消的任何权限。
- 使用 `sp_droptreshold` 可以从段中删除阈值。

禁用可用空间计数

警告！ 如果禁用可用空间计数，系统过程将无法提供有关空间分配的准确信息。

- 使用 `sp_dboption` 的 `no free space acctg` 选项可以禁用对非日志段的可用空间计数。
- 不能禁用对日志段的可用空间计数。

权限

只有数据库所有者或系统管理员才能执行 `sp_addthreshold`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关使用阈值的详细信息，请参见《系统管理指南》。

命令 [create procedure](#)、[dump transaction](#)

函数 [lct_admin](#)

系统过程 [sp_dboption](#)、[sp_droptreshold](#)、[sp_helpthreshold](#)、[sp_modifythreshold](#)、[sp_thresholdaction](#)

sp_addtype

说明

创建用户定义的数据类型。

语法

```
sp_addtype typename,  
           phystype [(length) | (precision [, scale])]  
           [, "identity" | nulltype]
```

参数

typename

是用户定义的数据类型的名称。类型名必须符合标识符的规则并且在每个数据库中必须是唯一的。

phystype

是物理数据类型或 Adaptive Server 提供的数据类型，该数据类型是用户定义的数据类型的基础。您可以指定除 `timestamp` 之外的任何 Adaptive Server 数据类型。

`char`、`varchar`、`unichar`、`univarchar`、`nchar`、`nvarchar`、`binary` 和 `varbinary` 数据类型需要一个括在小括号中的 *length*。如果未提供长度，则 Adaptive Server 将使用 1 个字符的缺省长度。

`float` 数据类型要求将二进制 *precision* 加以括号。如果未提供精度，Adaptive Server 将使用所用平台的缺省精度。

`numeric` 和 `decimal` 数据类型需要括在小括号中并用逗号分隔的十进位 *precision* 和 *scale*。如果未提供精度和标度，则 Adaptive Server 将使用缺省精度 18 和缺省标度 0。

用单引号或双引号将包含标点符号（例如小括号或逗号）的物理类型引起来。

标识

指示用户定义的数据类型具有 IDENTITY 属性。用单引号或双引号将 `identity` 关键字引起来。您只能为标度为 0 的 `numeric` 数据类型指定 IDENTITY 属性。

IDENTITY 列存储由 Adaptive Server 生成的顺序编号，例如发票编号或员工编号。IDENTITY 列的值唯一地标识表的每一行。IDENTITY 列不能进行更新，也不允许空值。

nulltype

指示用户定义的数据类型如何处理空值条目。此参数可接受的值包括 null、NULL、nonnull、NONULL、“not null”和“NOT NULL”。必须用单引号或双引号将包括空格的任何 *nulltype* 引起来。

如果既省略了 IDENTITY 属性又省略了 *nulltype*，则 Adaptive Server 将使用为数据库定义的空值模式创建数据类型。缺省情况下，未指定 *nulltype* 的数据类型将被创建为 NOT NULL（也就是说，不允许空值并且需要明确输入）。为符合 SQL 标准，使用 `sp_dboption` 系统过程将 `allow nulls by default` 选项设置为 true。这会将数据库的空值模式更改为 NULL。

示例

示例 1 创建一个名为 `ssn` 的用户定义的数据类型，该数据类型将用于保存社会保险号的列。由于未指定 *nulltype* 参数，Adaptive Server 将使用数据库的缺省空值模式创建数据类型。请注意，`varchar(11)` 引在引号中，这是因为它包含标点符号（小括号）：

```
sp_addtype ssn, "varchar(11)"
```

示例 2 创建一个名为 `birthday` 的用户定义的数据类型，该数据类型允许使用空值：

```
sp_addtype birthday, "datetime", null
```

示例 3 创建一个名为 `temp52` 的用户定义的数据类型，该数据类型用于存储温度（最多 5 个有效位数，其中小数点右侧 2 位）：

```
sp_addtype temp52, "numeric(5,2)"
```

示例 4 创建一个名为 `row_id` 的用户定义的数据类型，该数据类型具有 IDENTITY 属性，将用作唯一行标识符。使用此数据类型创建的列存储长度最多为 10 位的系统生成的值：

```
sp_addtype "row_id", "numeric(10,0)", "identity"
```

示例 5 创建具有 `sysname` 基础类型的用户定义的数据类型。虽然您不能在 `create table`、`alter table` 或 `create procedure` 语句中使用 `sysname` 数据类型，但却可以使用以 `sysname` 为基础的用户定义的数据类型：

```
sp_addtype systype, sysname
```

用法

- `sp_addtype` 创建用户定义的数据类型并将其添加到 `systypes` 系统表中。在创建了用户定义的数据类型之后，您可以在 `create table` 和 `alter table` 语句中使用它，并向其绑定缺省值和规则。
- 根据 Adaptive Server 提供的数据类型之一建立每个用户定义的数据类型，适当地指定长度或精度和标度。您不能替换 `create table` 或 `alter table` 语句中的长度、精度或标度。

- 用户定义的数据类型名在数据库中必须是唯一的，但具有不同名称的用户定义的数据类型可以具有相同的定义。
- 如果将 `nchar` 或 `nvarchar` 指定为 *phystype*，使用该新类型创建的列的最大长度是添加类型时 `sp_addtype` 中指定的长度乘以 `@@ncharsize` 的值。
- 如果将 `unichar` 或 `univarchar` 指定为 *phystype*，使用该新类型创建的列的最大长度是添加类型时 `sp_addtype` 中指定的长度乘以值 2。
- 每个系统类型都具有层次，存储在 `systypes` 系统表中。用户定义的数据类型与其基础物理类型具有相同的数据类型层次。在混合型表达式中，所有类型都会被转换为普通类型，即具有最低层次的数据类型。
使用以下查询列出数据库中每个系统提供的和用户定义的数据类型的层次：

```
select name, hierarchy
from systypes
order by hierarchy
```

具有 IDENTITY 属性的数据类型

- 如果用 `IDENTITY` 属性定义用户定义的数据类型，则从该数据类型创建的所有列都是 `IDENTITY` 列。您可以在 `create` 或 `alter table` 语句中指定 `IDENTITY`、`NOT NULL` 或一个都不指定。按照以下三种不同的方法从具有 `IDENTITY` 属性的用户定义的数据类型创建 `IDENTITY` 列：

```
create table new_table (id_col Identity)
create table new_table (id_col Identity identity)
create table new_table (id_col Identity not null)
```

- 当使用 `create table` 或 `alter table` 语句创建列时，您可以替换通过 `sp_addtype` 系统过程指定的空值类型：
 - 被指定为 `NOT NULL` 的类型可用于创建 `NULL` 或 `IDENTITY` 列。
 - 被指定为 `NULL` 的类型可用于创建 `NOT NULL` 列，但不能创建 `IDENTITY` 列。

注释 如果尝试从 `IDENTITY` 类型创建空列，则 `create` 或 `alter table` 语句将失败。

权限

任何用户都可以执行 `sp_addtype`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create default](#)、[create rule](#)、[create table](#)

数据类型 [用户定义的数据类型](#)

系统过程 [sp_bindefault](#)、[sp_bindrule](#)、[sp_dboption](#)、[sp_droptype](#)、[sp_rename](#)、[sp_unbindefault](#)、[sp_unbindrule](#)

sp_addumpdevice

| | |
|----|--|
| 说明 | 向 Adaptive Server 中添加转储设备。 |
| 语法 | <code>sp_addumpdevice {"tape" "disk"}, <i>logicalname</i>, <i>physicalname</i> [, <i>tapesize</i>]</code> |
| 参数 | <p>"tape" 代表磁带驱动器。用引号将 <code>tape</code> 引起来。</p> <p>"disk" 代表磁盘或文件设备。用引号将 <code>disk</code> 引起来。</p> <p><i>logicalname</i> 是“逻辑”转储设备名。它必须是一个有效的标识符。在您将转储设备添加到 <code>sysdevices</code> 后，可以在 <code>load</code> 和 <code>dump</code> 命令中指定其逻辑名。</p> <p><i>physicalname</i> 是设备的物理名称。可以指定绝对路径名，也可以指定相对路径名。在转储和装载期间，Backup Server 通过搜索 Adaptive Server 的当前工作目录来解析相对路径名。用引号将包含非字母数字字符的名称引起来。对于 UNIX 平台，指定非回绕磁带设备名。</p> <p><i>tapesize</i> 是磁带转储设备的容量，以兆字节为单位指定。如果指定，OpenVMS 系统将忽略 <code>tapesize</code>。其它平台要求此参数用于磁带设备，但不用于磁盘设备。<code>tapesize</code> 应该至少为 5 个数据库页大小（每页需要 2048 个字节）。Sybase 建议指定的容量应比设备的额定容量稍低。</p> |
| 示例 | <p>示例 1 添加一个 40MB 的磁带设备。转储和装载命令可以通过设备的物理名称 <code>/dev/nrmt8</code> 或设备的逻辑名称 <code>mytapedump</code> 引用该设备：</p> <pre>sp_addumpdevice "tape", mytapedump, "/dev/nrmt8", 40</pre> <p>示例 2 添加一个名为 <code>mydiskdump</code> 的磁盘设备。指定一个绝对或相对路径名和一个文件名：</p> <pre>sp_addumpdevice "disk", mydiskdump, "/dev/rxyld/dump.dat"</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_addumpdevice</code> 将一个转储设备添加到 <code>master.dbo.sysdevices</code> 表。为磁带设备分配的 <code>cntrtype</code> 是 3；为磁盘设备分配的 <code>cntrtype</code> 是 2。 • 若要将操作系统文件用作转储设备，请将设备指定为类型 <code>disk</code> 并为 <code>physicalname</code> 指定绝对或相对路径名。省略 <code>tapesize</code> 参数。如果指定相对路径名，当执行转储或装载命令时，将向当前 Adaptive Server 工作目录转储或从该目录装载。 |

- 所有权和权限问题可能影响磁盘或文件转储设备的使用。
`sp_addumpdevice` 将设备添加到 `sysdevices` 表中，但不能确保您可以将文件创建为转储设备或用户可以转储到特定的设备。
- `dump database` 和 `dump transaction` 命令的 `with capacity = megabytes` 子句可以替换通过 `sp_addumpdevice` 指定的 `tapesize`。在不能可靠地检测磁带结束标志的平台上，Backup Server 在转储完指定的兆字节数后将发出卷更改请求。
- 如果转储设备出现故障，则使用 `sp_dropdevice` 将其从 `sysdevices` 中删除。在替换设备之后，使用 `sp_addumpdevice` 将逻辑设备名称与新物理设备关联起来。这可以避免每次转储设备出现故障时都更新备份脚本和阈值过程。
- 若要将数据库设备添加到 `sysdevices` 中，请使用 `disk init` 命令。

权限

只有系统管理员才能执行 `sp_addumpdevice`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `disk init`、`dump database`、`dump transaction`、`load database`、`load transaction`

系统过程 `sp_dropdevice`、`sp_helpdevice`

sp_adduser

| | |
|----|--|
| 说明 | 将新用户添加到当前数据库。 |
| 语法 | <code>sp_adduser loginname [, name_in_db [, grpname]]</code> |
| 参数 | <p><i>loginname</i> 是 master.dbo.syslogins 中的用户名。</p> <p><i>name_in_db</i> 是当前数据库中用户的新名称。</p> <p><i>grpname</i> 将用户添加到数据库的现有组中。</p> |
| 示例 | <p>示例 1 将 “margaret” 添加到数据库。她的数据库用户名与其 Adaptive Server 登录名相同，她属于缺省组 “public”：</p> <pre>sp_adduser margaret</pre> <p>示例 2 将 “haroldq” 添加到数据库。当 “haroldq” 使用当前数据库时，他的名称为 “harold”。他属于 fort_mudge 组以及缺省组 “public”：</p> <pre>sp_adduser haroldq, harold, fort_mudge</pre> |
| 用法 | <ul style="list-style-type: none">• 数据库所有者执行 <code>sp_adduser</code> 将用户名添加到当前数据库的 <code>sysusers</code> 表中，从而使该用户可以在自己的名称下访问当前数据库。• 指定 <i>name_in_db</i> 参数可以在数据库中为新用户指定一个名称，该名称不同于其在 Adaptive Server 中的登录名。为用户分配不同名称的功能是为方便操作而提供的。它不是别名（如 <code>sp_addalias</code> 所提供的），因为不会将它映射到另一个用户的标识和特权。• 用户和组不能具有相同的名称。• 除了缺省组 “public” 之外，用户只能是一个组的成员。每个用户都是缺省组 “public” 的成员。使用 <code>sp_changegroup</code> 可以更改用户的组。• 为了访问数据库，用户必须列在 <code>sysusers</code> 中（通过 <code>sp_adduser</code>）或被映射到 <code>sysalternates</code> 中的另一个用户（通过 <code>sp_addalias</code>），或者 <code>sysusers</code> 中必须有一个 “guest” 条目。 |
| 权限 | 只有数据库所有者、系统管理员或系统安全员才能执行 <code>sp_adduser</code> 。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 `grant`、`revoke`、`use`

系统过程 `sp_addalias`、`sp_addgroup`、`sp_changegroup`、`sp_dropalias`、`sp_dropgroup`、`sp_helpuser`

sp_altermessage

说明 启用或禁用在 Adaptive Server 错误日志中记录系统定义的消息或用户定义的消息。

语法 sp_altermessage *message_id*, *parameter*, *parameter_value*

参数 *message_id*

是要更改的消息的消息号。这是在 `sysmessages` 或 `sysusermessages` 系统表的 `error` 列中记录的消息的编号。

parameter

是要更改的消息参数。最大长度为 30 个字节。唯一有效的参数是 `with_log`。

parameter_value

是在 *parameter* 中指定的参数的新值。最大长度为 5 个字节。值可以是 `true` 和 `false`。

示例 指定 `sysmessages` 中编号为 2000 的消息应该记录在 Adaptive Server 错误日志和 Windows NT 事件日志（如果启用了记录）中：

```
sp_altermessage 2000, 'with_log', 'TRUE'
```

用法

- 如果 *parameter_value* 是 `true`，则始终记录指定的消息。如果它是 `false`，则使用缺省记录行为；是否记录消息取决于错误的严重性和其它因素。将 *parameter_value* 设置为 `false` 产生的行为与未调用 `sp_altermessage` 发生的情况相同。
- 在 Windows NT 服务器上，`sp_altermessage` 也可以启用和禁用 Windows NT 事件日志中的记录。

权限 只有数据库所有者或系统管理员才能执行 `sp_altermessage`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_addmessage](#)、[sp_dropmessage](#)

sp_audit

说明 允许系统安全员配置审计选项。

语法 `sp_audit option, login_name, object_name [,setting]`

或者:

`sp_audit 'restart'`

参数 `option`
是要设置的审计选项的名称。表 1-3 列出了有效的审计选项。

表 1-3: sp_audit 的审计选项

| 选项 | 说明 |
|-----------------|--|
| adhoc | 允许用户使用 <code>sp_addauditrecord</code> 将他们自己的用户定义的审计记录添加到审计追踪中。 |
| all | 审计由特定用户或具有特定角色的用户执行的所有操作。您只能使用此选项指定系统角色。 注释 审计所有操作不影响用户是否可以添加即席审计记录。 |
| alter | 审计 <code>alter table</code> 或 <code>alter database</code> 命令的执行。 |
| bcp | 审计 <code>bcp in</code> 实用程序的执行。 |
| bind | 审计 <code>sp_bindefault</code> 、 <code>sp_bindmsg</code> 和 <code>sp_bindrule</code> 系统过程的执行。 |
| cluster | 审计集群命令。 |
| cmdtext | 审计特定用户的所有操作。 注释 为安全起见，可用星号组成的固定长度字符串替换系统存储过程和命令口令参数。有关详细信息，请参见第 68 页的“隐藏系统存储过程和命令口令参数”。 |
| create | 审计数据库对象的创建。 |
| dbaccess | 审计另一个数据库对当前数据库的访问。 |
| dbcc | 审计任何 <code>dbcc</code> 命令的执行。 |
| delete | 审计从表或视图中删除行。 |
| disk | 审计 <code>disk init</code> 、 <code>disk refit</code> 、 <code>disk reinit</code> 、 <code>disk mirror</code> 、 <code>disk unmirror</code> 和 <code>disk remirror</code> 的执行。 |
| drop | 审计数据库对象的删除。 |
| dump | 审计 <code>dump database</code> 或 <code>dump transaction</code> 的执行。 |
| encryption_key | 审计 <code>create encryption key</code> 、 <code>sp_encryption</code> 、 <code>drop encryption key</code> 和 <code>alter encryption key</code> 。 |
| errors | 审计错误（不管错误是否是致命的）。 |
| exec_procedure | 审计存储过程的执行。 |
| exec_trigger | 审计触发器的执行。 |
| func_dbaccess | 审计通过 Transact-SQL 函数对数据库的访问。 |
| func_obj_access | 审计选项审计通过 Transact-SQL 函数对数据库对象的访问。 |

| 选项 | 说明 |
|----------------|--|
| grant | 审计 grant 的执行。 |
| insert | 审计向表或视图中插入行。 |
| install | 审计 Java 类的安装。 |
| load | 审计 load database 或 load transaction 的执行 |
| login | 审计登录 Adaptive Server 的所有尝试。 |
| login_locked | 当登录帐户由于失败登录尝试次数超出配置次数而被锁定时，将审计主机名和网络 IP 地址。 |
| logout | 审计注销 Adaptive Server 的所有尝试。 |
| mount | 审计 mount database 命令。 |
| quiesce | 审计 quiesce database 命令。 |
| reference | 审计表之间的引用。 |
| remove | 审计 Java 类的删除。 |
| revoke | 审计 revoke 的执行。 |
| rpc | 审计远程过程调用的执行。 |
| security | <p>审计以下安全性相关事件：</p> <ul style="list-style-type: none"> • 启动或关闭服务器 • 激活或停用角色 • 发出下列任何命令： <ul style="list-style-type: none"> addcert connect create 和 drop login dropcert create、drop、alter、grant 和 revoke role kill online database set proxy set session authorization sp_configure • 第 65 页的表 1-4 列出了这些函数。 • 使用以下函数之一： <ul style="list-style-type: none"> valid_user proc_role （在系统过程中） • 重新生成 SSO 口令 |
| select | 审计 select 的执行。 |
| setuser | 审计 setuser 的执行。 |
| table_access | 审计特定用户对任何表的访问。 |
| transfer table | 审计 transfer table 命令的执行 |
| truncate | 审计 truncate table 的执行。 |

| 选项 | 说明 |
|-------------|---|
| unbind | 审计 <code>sp_unbindrule</code> 、 <code>sp_unbindmsg</code> 和 <code>sp_unbindefault</code> 的执行。 |
| unmount | 审计 <code>umount database</code> 命令的执行。 |
| update | 审计对表和视图中行的更新。 |
| view_access | 审计特定用户对任何视图的访问。 |

表 1-4: `sp_audit security` 审计的函数

- | | | |
|--------------------------------------|---------------------------------|-------------------------------------|
| • <code>config_admin</code> | • <code>set_password</code> | • <code>valid_user</code> |
| • <code>attr_notify</code> | • <code>ha_add_companion</code> | • <code>ha_remove_companion</code> |
| • <code>ha_check_alive</code> | • <code>ha_getversion</code> | • <code>ha_failback</code> |
| • <code>ha_rerestrictionclass</code> | • <code>ha_getrcs</code> | • <code>ha_setrcs</code> |
| • <code>ha_hacluster_verify</code> | • <code>js_wakeup</code> | • <code>unlock_admin_account</code> |
| • <code>ssl_admin</code> | • <code>ws_admin</code> | |

login_name

是一个参数，用于指定 `all`（一个系统角色）或要审计的特定登录的名称。但是，只有使用 `all` 选项才能指定系统角色。不能审计系统角色的个别选项。

object_name

是要审计的对象的名称。根据为 `option` 指定的值，有效值包括：

- 对象名称，包括所有者的名称（如果您不是该对象的所有者）。例如，如果审计 Joe 拥有的名为 `inventory` 的表，则需要为 `object_name` 指定 `joe.inventory`。
- `all`（对于所有对象）。
- `default table`、`default view`、`default procedure` 或 `default trigger` — 审计对任何新表、视图、过程或触发器的访问。

当为 `option` 参数指定 `delete`、`insert`、`select` 或 `update` 时，`default table` 和 `default view` 是 `object_name` 的有效值。当指定 `exec_procedure` 选项时，`default procedure` 是有效值。当指定 `exec_trigger` 选项时，`default trigger` 是有效值。

有关每个 `option` 值有效的 `object_name` 值的详细信息，请参见《系统管理指南》。

setting

是审计的级别。如果未指定 **setting** 的值，则 Adaptive Server 将显示选项的当前审计设置。**setting** 参数的有效值为：

- **on** — 激活对指定选项的审计。Adaptive Server 生成受此选项控制的事件的审计记录，而不管该事件是否通过权限检查。
- **off** — 失活对指定选项的审计。
- **pass** — 激活通过权限检查的事件的审计。
- **fail** — 激活未通过权限检查的事件的审计。

如果对某个选项指定 **pass**，稍后又对同一个选项指定 **fail**（或相反），则结果与指定 **on** 相同。不管事件是否通过权限检查，Adaptive Server 都将生成审计记录。

设置：

- **on** 或 **off** — 适用于所有审计选项
- **pass** 和 **fail** — 适用于除 **cmdtext**、**errors** 和 **adhoc** 之外的所有选项。对于这两个选项，只有 **on** 或 **off** 适用。所有选项的初始、缺省值都是 **off**。如果将 **cmdtext** 选项选择为 **pass** 或 **fail**，则 Adaptive Server 用 **on** 替换该值。

restart

如果审计过程因出现错误而被强制终止，则可通过输入以下命令手动重新启动 **sp_audit**：

```
sp_audit restart
```

如果当前没有运行任何审计，但该审计进程被配置为通过输入 **sp_configure "auditing" 1** 来运行，则可重新启动该审计进程。

示例

示例 1 当登录帐户由于失败登录尝试次数超出配置次数而被锁定时，将 **login_locked** 审计选项设置为启动对主机名和网络 IP 地址的审计：

```
sp_audit "login_locked", "all", "all", "ON"
```

如果审计表满了，不能记录事件，会向错误日志发送包含相关信息的信息。

监控 Locked Login 事件 (112) 的审计日志有助于标识对登录帐户的攻击。

示例 2 启动对 SSL 安全性相关事件的审计。成功和失败的事件都将进行审计：

```
sp_audit "security", "all", "all", "on"
sample records added:
```

查看 sybsecurity 中的事件：

```
select * from sybsecurity..sysaudits_01 where event=99
```

示例 3 显示 security 审计选项的设置：

```
sp_audit "security", "all", "all"
```

示例 4 启动对 master 数据库中对象创建（包括 [create database](#)）的审计。

```
sp_audit "create", "all", master, "on"
```

示例 5 审计 pubs2 数据库中的命令：

```
sp_audit "encryption_key", "all", "pubs2", "on"
```

示例 6 启动对 db1 数据库中所有对象创建的审计：

```
sp_audit "create", "all", db1, "on"
```

示例 7 启动对系统管理员所有失败执行的审计。

```
sp_audit "all", "sa_role", "all", "fail"
```

示例 8 启动对当前数据库中新表的所有更新的审计。例如，如果当前数据库是 utility，则将审计 utility 中创建的所有新表是否发生更新。对现有表的审计不受影响。

```
sp_audit "update", "all", "default table", "on"
```

示例 9 启动对为 titles 表输入的所有 transfer table 命令的审计：

```
sp_audit "transfer table", "all", "all", "titles", "on"
```

用法

- `sp_audit` 确定当启用审计时将审计的内容。在您使用 [sp_configure](#) 将 `auditing` 参数设置为 `on` 之前，不会发生任何实际的审计。设置之后，使用 `sp_audit` 配置的所有审计选项生效。有关详细信息，请参见 [sp_configure](#)。
- 如果您不是所指定对象的所有者，请用所有者的名称对 `object_name` 参数值进行限定，格式如下：

```
"ownername.objname"
```

- 您不能在 `tempdb` 数据库中激活以下选项的缺省审计：

| | | |
|--------|----------------|--------------|
| delete | insert | select |
| update | exec_procedure | exec_trigger |

- 控制审计的配置参数为：
 - `auditing` — 启用或禁用对服务器的审计。

- **audit queue size** — 确定审计队列的大小。
- **current audit table** — 设置当前审计表。 Adaptive Server 将所有审计记录写入该表。
- **suspend auditing when full** — 控制在审计设备充满时审计进程的行为。

所有审计配置参数都是动态的，将立即生效。

隐藏系统存储过程和命令口令参数

如果配置并启用了审计，且设置了 **sp_audit** 选项 'cmdtext'，则审计日志中包含的审计记录里的系统存储过程和命令口令参数就会被替换为固定长度的星号串。

例如，当启用审计并设置 **sp_audit cmdtext** 后，执行以下命令：

```
sp_password 'oldpassword', 'newpassword'
```

这将导致类似如下的输出：

```
sp_password '*****', '*****'
```

这样做可以保护口令不会被可以访问审计日志的其他人看到。

权限

只有系统安全员才能执行 **sp_audit**。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关配置 Adaptive Server 进行审计的详细信息，请参见《系统管理指南》中的 **sp_configure**。

系统过程 [sp_addauditrecord](#)、[sp_configure](#)、[sp_addauditable](#)

实用程序命令 bcp

sp_autoconnect

| | |
|----|--|
| 说明 | (仅限组件集成服务) 为特定用户定义到远程服务器的直通连接, 这样就可以使指定的用户在登录时自动进入直通模式。 |
| 语法 | <code>sp_autoconnect server, {true false} [, loginame]</code> |
| 参数 | <p><i>server</i></p> <p>是与其建立自动直通连接的服务器的名称。 <i>server</i> 必须是已通过 <code>sp_addserver</code> 添加的远程服务器的名称。此服务器不能是本地服务器。</p> <p><code>true false</code></p> <p>确定是否为 <i>server</i> 启用或禁用自动直通连接。 <code>true</code> 会启用自动连接; <code>false</code> 则会禁用自动连接。</p> <p><i>loginame</i></p> <p>指定需要自动连接的用户名称。如果未提供任何 <i>loginame</i>, 则修改当前用户的自动连接状态。</p> |
| 示例 | <p>示例 1 当前用户下次登录时, 该用户将自动连接到服务器 SYBASE。该用户的连接处于直通模式:</p> <pre>sp_autoconnect SYBASE, true</pre> <p>示例 2 禁用用户 “steve” 的自动连接功能:</p> <pre>sp_autoconnect SYBASE, false, steve</pre> |
| 用法 | <ul style="list-style-type: none">• <code>sp_autoconnect</code> 为特定用户定义到远程服务器的直通连接, 这样就可以使指定的用户在登录时自动进入直通模式。• 在执行 <code>sp_autoconnect</code> 之前, 系统管理员必须授予登录名 <code>connect to</code> 权限。• 只有在安装并配置了组件集成服务时才能使用 <code>sp_autoconnect</code>。• 不要更改 “sa” 登录帐户的自动连接状态。• 对于当前连接的用户, 对自动连接状态的更改不会立即发生。在进行更改之前, 他们必须从本地服务器断开连接, 然后重新连接。• 使用 <code>disconnect</code> 退出直通模式。 |
| 权限 | 只有系统管理员才能执行 <code>sp_autoconnect</code> 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [connect to...disconnect](#)、[grant](#)

系统过程 [sp_addlogin](#)、[sp_addserver](#)、[sp_passthru](#)、[sp_remotesql](#)

sp_autoformat

说明 可生成可读结果集数据的实用程序存储过程，`sp_autoformat` 将重新设置长度可变的字符数据的宽度格式，以便只显示非空字符。输出中将截断尾随空白。

语法 `sp_autoformat fulltablename[, selectlist, whereclause, orderby]`

参数 *fulltablename*

指定从中选择数据的表的名称。如果对象所有者不是运行该命令的用户，请使用所有者名称。

selectlist

指定要从结果集中选择的以逗号分隔的列列表。使用 `<name> = <column>` 符号可对表中的列重命名。请参见示例。如果未提供 *selectlist*，则所指定表中的所有列都将以列 ID 顺序输出。

whereclause

是搜索谓词，被指定为 `where` 子句，用于从选定的表中过滤行。

orderby

是可选的 `order by` 子句，用于指定输出结果集的显示顺序。

示例 **示例 1** 从类似 `select id, colid, name from syscolumns where id = 3` 的 `select` 语句返回结果集（其中字符列的格式将自动设置）：

```
1> sp_autoformat "syscolumns", "id, colid, name", "where id = 3"
2> go
```

| id | colid | name |
|-------|-------|-------------|
| ----- | ----- | ----- |
| 3 | 1 | id |
| 3 | 2 | number |
| 3 | 3 | colid |
| 3 | 4 | status |
| 3 | 5 | type |
| 3 | 6 | length |
| 3 | 7 | offset |
| 3 | 8 | usertype |
| 3 | 9 | cdefault |
| 3 | 10 | domain |
| 3 | 11 | name |
| 3 | 12 | printfmt |
| 3 | 13 | prec |
| 3 | 14 | scale |
| 3 | 15 | remote_type |
| 3 | 16 | remote_name |
| 3 | 17 | xstatus |
| 3 | 18 | xtype |

```

3      19 xdbid
3      21 accessrule
3      22 status2

```

示例 2 使用下列语法重命名输出列:

```
[ < AS-Name label of Column> ][ ]*=[ ]*<column name>
```

<AS-Name label of Column> 是可选的, 可在 = 分隔符两侧使用空格:

```

1> sp_autoformat syscolumns, "'Object Id' = id, 'Column Name'=name,
   'Column ID'=colid", "where id = 3"
2> go

```

| Object Id | Column Name | Column ID |
|-----------|-------------|-----------|
| 3 | id | 1 |
| 3 | number | 2 |
| 3 | colid | 3 |
| 3 | status | 4 |
| 3 | type | 5 |
| 3 | length | 6 |
| 3 | offset | 7 |
| 3 | usertype | 8 |
| 3 | cdefault | 9 |
| 3 | domain | 10 |
| 3 | name | 11 |
| 3 | printfmt | 12 |
| 3 | prec | 13 |
| 3 | scale | 14 |
| 3 | remote_type | 15 |
| 3 | remote_name | 16 |
| 3 | xstatus | 17 |
| 3 | xtype | 18 |
| 3 | xdbid | 19 |
| 3 | accessrule | 21 |
| 3 | status2 | 22 |

(1 row affected)

示例 3 使用 *orderby* 参数可以指定结果输出中的顺序:

```

sp_autoformat @fulltabname = 'syscolumns',
              @selectlist = "id, name",
              @orderby = "ORDER BY name"

```

示例 4 要在从多个表中选择时生成自动设置格式的结果，或者如果 `select` 列表中的表达式具有复杂的 SQL `select` 语句，则必须：

1 使用临时表生成结果集：

下列命令将生成在所有系统目录中具有最高列 ID 的列的列表：

```
select o.id, o.name, c.colid, c.name
from sysobjects o, syscolumns c
where o.id < 100 and o.id = c.id
      and c.colid = (select max(c2.colid) from syscolumns c2
                    where c2.id = c.id)
order by o.name
```

下列命令可生成自动设置了字符数据格式的结果集，并使用临时表来生成可读输出，其中还包括微小的更改，以便在临时表中提供列名：

```
select o.id, ObjectName = o.name, c.colid, ColumnName = c.name
into #result
from sysobjects o, syscolumns c
where o.id < 100 and o.id = c.id
      and c.colid = (select max(c2.colid) from syscolumns c2
                    where c2.id = c.id)
```

2 对临时表使用 `sp_autoformat` 可生成格式化的输出：

生成临时表时将跳过原始 `select` 语句中的 `order by` 子句，而在生成输出结果时会将该子句添加到对 `sp_autoformat` 的调用中。

```
1> exec sp_autoformat @fulltablename = #result, @orderby = "order by
      ObjectName"
2> go
```

| id | ObjectName | colid | ColumnName |
|----|-----------------|-------|------------|
| 11 | sysalternates | 2 | altsuid |
| 21 | sysattributes | 13 | comments |
| 55 | syscertificates | 6 | suid |
| 45 | syscharsets | 8 | sortfile |
| 3 | syscolumns | 22 | status2 |
| 6 | syscomments | 8 | status |
| 37 | sysconfigures | 9 | value4 |
| 17 | sysconstraints | 7 | spare2 |
| 38 | syscurconfigs | 15 | type |
| 30 | sysdatabases | 19 | status4 |
| 12 | sysdepends | 10 | readobj |
| 35 | sysdevices | 7 | mirrorname |
| 43 | sysengines | 12 | starttime |

...

```
(1 row affected)
(return status = 0)
```

可对临时表进行进一步处理，以便只报告对选定表的必需输出，如下所示：

```
1> exec sp_autoformat #result, "id, 'Object Name' = ObjectName,
   'Column Name' = ColumnName", "where id < 5"
2> go
```

| id | Object Name | Column Name |
|----|-------------|-------------|
| 1 | sysobjects | loginame |
| 2 | sysindexes | crdate |
| 3 | syscolumns | status2 |
| 4 | systypes | accessrule |

用法

- 在 Adaptive Server 15.0.3 版和更高版本中，`sp_autoformat` 接受数据类型为 `int` (`smallint`、`bigint`、`tinyint`、`unsigned int`)、`numeric`、`money`、`date/time` 以及 `float`、`real` 和 `double precision` 的列。
- `sp_autoformat` 仅在当前数据库中查找对象。要对临时表使用 `sp_autoformat`，请从 `tempdb` 发出该过程。
- `sp_autoformat` 不会验证各个参数所引用的列是否实际存在于 `fulltablename` 参数指定的表中。如果引用了任何不存在的列，`sp_autoformat` 将失败。
- 只在 `select` 列表中提供一个列的实例。

返回代码

- 0 — 成功完成
- 1 — 内部错误，或在调用中出现使用错误
- 其它 — 在执行生成的 SQL 语句时，由 Adaptive Server 引发的所有其它错误都将返回给调用方。

限制

- `sp_autoformat` 使用内部 SQL 变量来生成 SQL 语句，然后使用 `execute immediate` 执行该语句。生成的 SQL 语句的长度限制为 2K 字节。用于较大列表的自动格式化结果集或名称较长的列有时会因为缓冲区大小对于生成的 SQL 语句而言不够充足而引发错误。

- 表或列名称不支持带引号的标识符。如果您的结果集使用带引号的标识符或者需要自动设置格式，则应当：
 - a 在临时表中生成所需数据，其中临时表中的列没有带引号的标识符。
 - b 使用 `sp_autoformat` 并通过临时表来生成所需输出。
 - c 以所需的输出格式对 *selectlist* 中的列进行重命名。

权限

任何用户都可以执行 `sp_autoformat`。但是，从表中进行选择的用户必须具有相应的 `select` 特权。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

sp_bindcache

| | |
|----|--|
| 说明 | 将数据库、表、索引、text 对象或 image 对象绑定到数据高速缓存。 |
| 语法 | <pre>sp_bindcache cachename, dbname [, [ownername.]tablename [, indexname "text only"]]</pre> |
| 参数 | <p>cachename 是活动的数据高速缓存的名称。</p> <p>dbname 是要绑定到高速缓存的数据库的名称或包含要绑定到高速缓存的表、索引、text 或 image 对象的数据库的名称。</p> <p>ownername 是表所有者的名称。如果表由 “dbo” 所有，则所有者名称是可选的。</p> <p>tablename 是要绑定到高速缓存的表的名称或其索引、text 对象或 image 对象要绑定到高速缓存的表的名称。</p> <p>indexname 要绑定到高速缓存的索引的名称。</p> <p>text only 将 text 或 image 对象绑定到高速缓存。当使用此参数时，您不能同时给出索引名称。</p> |
| 示例 | <p>示例 1 将 titles 表绑定到名为 pub_cache 的高速缓存：</p> <pre>sp_bindcache pub_cache, pubs2, titles</pre> <p>示例 2 将聚簇索引 titles.title_id_cix 绑定到 pub_ix_cache：</p> <pre>sp_bindcache pub_ix_cache, pubs2, titles, title_id_cix</pre> <p>示例 3 将 pubs2 绑定到 tempdb_cache：</p> <pre>sp_bindcache tempdb_cache, pubs2</pre> <p>示例 4 将 pubs2 事务日志 syslogs 绑定到名为 logcache 的高速缓存：</p> <pre>sp_bindcache logcache, pubs2, syslogs</pre> <p>示例 5 将 au_pix 表的 image 链绑定到名为 pub_cache 的高速缓存：</p> <pre>sp_bindcache pub_cache, pubs2, au_pix, "text only"</pre> |

用法

- 只能将数据库或数据库对象绑定到一个高速缓存。您可以将数据库绑定到一个高速缓存，并将该数据库中的个别表、索引、**text** 对象或 **image** 对象绑定到其它高速缓存。数据库绑定充当该数据库中所有没有其它绑定的对象的缺省绑定。表或索引的数据高速缓存层次如下所示：
 - 如果对象被绑定到高速缓存，则使用对象绑定。
 - 如果对象未被绑定到高速缓存，但对象的数据库被绑定到高速缓存，则使用数据库绑定。
 - 如果对象及其数据库都未被绑定到高速缓存，则使用缺省数据高速缓存。
- 在您可以执行 **sp_bindcache** 之前，高速缓存和绑定到它的对象或数据库必须存在。使用 **sp_cacheconfig** 创建高速缓存，如果操作不是动态的，则请在将对象绑定到高速缓存之前重新启动 **Adaptive Server**。
- 高速缓存绑定将立即生效，不需要重新启动服务器。当将对象绑定到数据高速缓存时：
 - 将清除当前内存中该对象的所有页。
 - 当查询中使用该对象时，其页将被读入绑定的高速缓存中。
- 您可以将索引绑定到与它所引用的表不同的高速缓存中。如果将聚簇索引绑定到高速缓存，该绑定将只影响索引的根和中间页。它不影响数据页（根据定义，数据页是索引的叶页）。
- 若要绑定数据库，您必须使用 **master** 数据库。若要绑定表、索引、**text** 对象或 **image** 对象，您必须使用存储这些对象的数据库。
- 若要绑定数据库中的任何系统表，您必须使用该数据库并且该数据库必须处于单用户模式。请使用以下命令：

```
sp_dboption db_name, "single user", true
```

有关详细信息，请参见 [sp_dboption](#)。

- 将对象或数据库绑定到不同的高速缓存时，不需要解除它们的现有绑定。对已经绑定的对象发出 **sp_bindcache** 将删除旧绑定并创建新绑定。
- 当要将表或其索引绑定到高速缓存时，**sp_bindcache** 需要获得排它表锁，这样在绑定发生时，将无法读取任何页。如果某个用户持有表锁，而您对该对象发出 **sp_bindcache**，则执行绑定的任务在锁被释放之前将处于休眠状态。

- 如果绑定对象或解除对象的绑定，则引用该对象的所有存储过程在下次执行时将被重新编译。如果更改数据库的绑定，则引用绑定的数据库中对象的所有存储过程在下次执行时将被重新编译。
- 如果删除表、索引或数据库，则所有关联的高速缓存绑定将被删除。如果重新创建表、索引或数据库，则必须再次使用 `sp_bindcache` 才能将其绑定到高速缓存。
- 如果将数据库或数据库对象绑定到高速缓存，而高速缓存被删除，则高速缓存绑定被标记为无效，但仍存储在 `sysattributes` 系统表中。当 **Adaptive Server** 重新启动时，将在错误日志中输出警告。如果创建了具有相同名称的高速缓存，则当 **Adaptive Server** 重新启动时，绑定变为有效。
- 以下过程提供有关各自对象绑定的信息：`sp_helpdb` 用于数据库、`sp_help` 用于表、`sp_helpindex` 用于索引。`sp_helpcache` 提供有关绑定到特定高速缓存的所有对象的信息。
- 使用 `sp_spaceused` 查看表和索引的当前大小，使用 `sp_estspace` 估计表预计增长大小。使用 `sp_cacheconfig` 查看有关高速缓存大小和状态的信息，以及配置和重新配置高速缓存。
- 虽然仍可以在系统 `tempdb` 上使用 `sp_bindcache`，但系统 `tempdb` 的绑定现在是非动态的。在重新启动服务器之前：
 - 所做的更改不会生效。
 - 除非将系统 `tempdb` 显式绑定到缺省数据高速缓存，否则 `sp_helpcache` 报告状态 “P”（表示挂起），显式绑定到缺省数据高速缓存后，状态显示为 “V”（表示有效），这是因为缺省情况下，系统 `tempdb` 已被绑定到缺省数据高速缓存。

限制

- 不能将 `master` 数据库、`master` 中的系统表和 `master` 中系统表上的索引绑定到高速缓存。您可以将 `master` 中的非系统表及其索引绑定到高速缓存。
- 以下情况下，不能将数据库或对象绑定到高速缓存：
 - 隔离级别 0 读取在表上是活动的
 - 执行绑定的任务有在表上打开的游标
- 如果高速缓存类型为 `log only`，则只能将 `syslogs` 表绑定到该高速缓存。使用 `sp_cacheconfig` 可以查看高速缓存的类型。

权限

只有系统管理员才能执行 `sp_bindcache`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_cacheconfig](#)、[sp_configure](#)、[sp_help](#)、[sp_helpcache](#)、[sp_helpdb](#)、[sp_helpindex](#)、[sp_poolconfig](#)、[sp_unbindcache](#)、[sp_unbindcache_all](#)

sp_bindefault

| | |
|----|--|
| 说明 | 将用户定义的缺省值绑定到列或用户定义的数据类型。 |
| 语法 | <code>sp_bindefault defname, objname [, futureonly]</code> |
| 参数 | <p>defname</p> <p>是用 <code>create default</code> 语句创建的缺省值的名称，要将该缺省值绑定到特定列或用户定义的数据类型。</p> <p>objname</p> <p>是要向其绑定缺省值的表和列的名称或用户定义的数据类型。如果 objname 参数的形式不是 “<i>table.column</i>”，则它将被认为是用户定义的数据类型。如果对象名包括嵌入的空格或标点符号，或者是保留字，则用引号将它引起来。</p> <p>除非您指定 futureonly，否则用户定义的数据类型的现有列继承缺省值 defname。</p> <p>无法将缺省值绑定到计算列。</p> <p>futureonly</p> <p>防止用户定义的数据类型的现有列获得新缺省值。当您 将缺省值绑定到用户定义的数据类型时，此参数是可选的。它从不用于将缺省值绑定到列。</p> |
| 示例 | <p>示例 1 假设已用 <code>create default</code> 在当前数据库中定义了名为 today 的缺省值，此命令将其绑定到 employees 表的 startdate 列。除非 startdate 列中提供了其它值，否则添加到 employees 表的每个新行在该列中都具有缺省值 today：</p> <pre>sp_bindefault today, "employees.startdate"</pre> <p>示例 2 假设存在名为 def_ssn 的缺省值和名为 ssn 的用户定义的数据类型，此命令将 def_ssn 绑定到 ssn。当创建表时，分配有用户定义的数据类型 ssn 的所有列都将继承该缺省值。除非您指定 futureonly（它防止该用户定义的数据类型的现有列继承缺省值），或者除非列的缺省值已被更改（在这种情况下将保持更改的缺省值），否则类型为 ssn 的现有列也继承缺省值 def_ssn：</p> <pre>sp_bindefault def_ssn, ssn</pre> <p>示例 3 将缺省值 def_ssn 绑定到用户定义的数据类型 ssn。因为包括 futureonly 参数，所以不影响类型为 ssn 的任何现有列：</p> <pre>sp_bindefault def_ssn, ssn, futureonly</pre> |

- 用法
- 您可以通过两种方式创建列缺省值：通过在 `create table` 或 `alter table` 语句中将缺省值声明为列约束或通过使用 `create default` 语句创建缺省值并使用 `sp_bindefault` 将其绑定到列。使用 `create default`，您可以将该缺省值绑定到数据库中的多个列。
 - 您不能将缺省值绑定到 Adaptive Server 提供的数据类型。
 - 不能将缺省值绑定到系统表。
 - 绑定到具有 IDENTITY 属性的列或用户定义的数据类型的缺省值不会对列值产生影响。每次将行插入表中时，Adaptive Server 都会将下一个顺序号分配到 IDENTITY 列。
 - 如果将缺省值绑定到列，则以 “`table.column`” 形式提供 `objname` 参数。任何其它格式都将被认为是用户定义的数据类型的名称。
 - 如果列上已存在缺省值，那么在绑定新的缺省值之前必须将其删除。使用 `sp_unbindefault` 可以删除用 `sp_bindefault` 创建的缺省值。若要删除用 `create table` 或 `alter table` 创建的缺省值，请使用 `alter table` 替换具有 NULL 的缺省值。
 - 除非您指定 `futureonly`，否则用户定义的数据类型的现有列继承新缺省值。用户定义的数据类型的新列始终继承该缺省值。将缺省值绑定到用户定义的数据类型将替换已绑定到该类型的列的缺省值；若要恢复列绑定，请解除列缺省值的绑定，然后重新绑定。
 - 使用缺省值的语句不能与其 `sp_bindefault` 语句在同一批处理中。

权限 只有对象所有者才能执行 `sp_bindefault`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|---------------|---|
| 6 | bind | sp_bindefault | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 缺省名称 代理信息 — set proxy 有效时的初始登录名 |
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create default](#)、[create table](#)、[drop default](#)

系统过程 [sp_unbindefault](#)

sp_bindexeclass

| | |
|----|--|
| 说明 | 将执行类与客户端应用程序、登录、存储过程或缺省执行类关联起来。 |
| 语法 | <code>sp_bindexeclass "object_name", "object_type", "scope", "classname"</code> |
| 参数 | <p>object_name 是要与执行类 <code>classname</code> 关联的客户端应用程序、登录或存储过程的名称。如果 <code>object_type</code> 为 <code>df</code>，它就应该为空。</p> <p>object_type 标识 <code>object_name</code> 的类型。使用：</p> <ul style="list-style-type: none"> • <code>ap</code>（表示应用程序） • <code>df</code>（表示用户定义的缺省执行类） • <code>lg</code>（表示登录名） • <code>pr</code>（表示存储过程） • <code>sv</code>（表示服务任务，仅在线程化模式中有效） <p>scope 是客户端应用程序或登录的名称，或者对于 <code>ap</code>、<code>df</code>、<code>lg</code> 或 <code>sv</code> 对象，它可以为 <code>NULL</code>。对于对象，<code>scope</code> 是存储过程所有者的名称（用户名）。当具有 <code>object_name</code> 的对象与应用程序或登录交互时，<code>classname</code> 属性适用于您设置的 <code>scope</code>。</p> <p>classname 指定与 <code>object_name</code> 关联的类的类型。其值包括：</p> <ul style="list-style-type: none"> • <code>EC1</code>、<code>EC2</code> 或 <code>EC3</code> • 用户定义的执行类的名称 • <code>ANYENGINE</code> |
| 示例 | <p>示例 1 此语句指定对调用 <code>isql</code> 的任何登录或应用程序进程（因为 <code>scope</code> 的值为 <code>NULL</code>），除非该登录或应用程序被绑定到更高的执行类，否则 Transact-SQL 应用程序将使用 <code>EC3</code> 属性执行：</p> <pre>sp_bindexeclass 'isql', 'ap', NULL, 'EC3'</pre> <p>示例 2 此语句指定当具有系统管理员角色的登录名执行 Transact-SQL 应用程序时，登录进程使用 <code>EC1</code> 属性执行。如果已在第一个示例中执行了该语句，则调用 <code>isql</code> 的任何其它登录或客户端应用程序将使用 <code>EC3</code> 属性执行：</p> <pre>sp_bindexeclass 'sa', 'lg', 'isql', 'EC1'</pre> |

示例 3 此语句将 EC3 属性分配到用户 kundu 所有的名为 my_proc 的存储过程:

```
sp_bindexeclass 'my_proc', 'PR', 'kundu', 'EC3'
```

示例 4 此语句将 CLASS1 属性分配到所有在缺省执行属性下运行的任务:

```
sp_bindexeclass NULL, 'DF', NULL, 'CLASS1'
```

示例 5 将许可证心跳操作绑定到内核执行任务:

```
sp_bindexeclass "License Heartbeat", sv, NULL, core
```

用法

- 将执行类绑定到缺省执行类时，所有在缺省执行属性下运行的任务都将在新类的属性下执行。
- 可以将服务任务绑定到为管理用户任务而创建的现有执行类。也就是说，服务任务和用户任务可以在同一个执行类中并存。
- monServiceTask 监控表含有所有服务任务，包括其名称和当前绑定。
- sp_bindexeclass 将执行类与客户端应用程序、登录名或存储过程关联起来。它还将执行类与缺省执行类相关联。使用 [sp_addexeclass](#) 创建执行类。
- 当 scope 为 NULL 时，object_name 没有作用域。classname 的执行属性适用于它的所有交互。例如，如果 object_name 是应用程序名称，则属性适用于调用该应用程序的任何登录进程。如果 object_name 是登录名，则属性适用于登录进程调用的任何应用程序的特定登录进程。
- 当将存储过程绑定到执行类时，您必须对 scope 参数使用该存储过程所有者的名称（用户名）。当同一个数据库中有多个存储过程调用时，这可以缩小确定该存储过程的范围。
- 由于优先级和作用域规则，被绑定的执行类对名为 object_name 的对象不一定有效。该对象将自己自动绑定到另一个执行类，具体情况取决于其它绑定规范、优先级和作用域规则。如果没有其它绑定适用，该对象将绑定到缺省执行类。如果未指定用户定义的缺省执行类，则对象绑定到系统定义的执行类 EC2。
- 可以使用 sp_bindexeclass 将 RepAgent 线程绑定到将 rep agent 用作应用程序的执行类，而不会生成错误。但是，因为 Adaptive Server 中的限制，优先级属性被设置为中，绑定不起作用。
- 当尝试将活动进程绑定到无联机引擎的引擎组时，绑定失败。
- Adaptive Server 在 sysattributes 表中创建行，该表在存储绑定数据的行中包含对象 ID 和用户 ID。

- 绑定存储过程之前，该存储过程必须存在。
- 存储过程绑定必须在存储过程所在的数据库中进行。因此，当绑定系统过程时，请在 `sybsystemprocs` 数据库中执行 `sp_bindexeclass`。
- 将类绑定到存储过程时，仅使用执行类的“优先级属性”。
- 当将存储过程绑定到执行类时，必须以 `scope` 参数的形式提供存储过程的所有者的名称。当数据库中存在多个具有相同名称（但具有不同的所有者）的存储过程时，这有助于唯一确定存储过程。

权限

只有系统管理员才能执行 `sp_bindexeclass`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_addexeclass](#)、[sp_showexeclass](#)、[sp_unbindexeclass](#)
实用程序 [isql](#)

sp_bindmsg

| | |
|----|---|
| 说明 | 将用户消息绑定到参照完整性约束或检查约束。 |
| 语法 | <code>sp_bindmsg <i>constrname</i>, <i>msgid</i></code> |
| 参数 | <p><i>constrname</i></p> <p>是向其绑定消息的完整性约束的名称。使用 <code>create table</code> 命令的 <code>constraint</code> 子句或 <code>alter table</code> 命令的 <code>add constraint</code> 子句创建和命名约束。</p> <p><i>msgid</i></p> <p>是要绑定到完整性约束的用户消息的编号。在调用 <code>sp_bindmsg</code> 之前，该消息必须存在于本地数据库的 <code>sysusermessages</code> 表中。</p> |
| 示例 | <pre>sp_bindmsg positive_balance, 20100</pre> <p>将编号为 20100 的用户消息绑定到 <code>positive_balance</code> 约束。</p> |
| 用法 | <ul style="list-style-type: none"> <code>sp_bindmsg</code> 通过将消息号添加到 <code>sysconstraints</code> 表中的约束行将用户消息绑定到完整性约束。 一个约束只能绑定一条消息。若要更改约束的消息，只要绑定一条新消息即可。新消息号将替换 <code>sysconstraints</code> 表中的旧消息号。 不能将消息绑定到唯一约束，因为唯一约束在 <code>sysconstraints</code> 中没有约束行（唯一约束是唯一索引）。 使用 <code>sp_addmessage</code> 过程将用户消息插入 <code>sysusermessages</code> 表中。 <code>sp_getmessage</code> 过程从 <code>sysusermessages</code> 表中检索消息文本。 <code>sp_help tablename</code> 显示在 <code>tablename</code> 上声明的所有约束名。 |
| 权限 | 只有对象所有者才能执行 <code>sp_bindmsg</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|------|------------|--|
| 7 | bind | sp_bindmsg | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 消息 ID 代理信息 — set proxy 有效时的初始登录名 |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [alter table](#)、[create table](#)

系统过程 [sp_addmessage](#)、[sp_getmessage](#)、[sp_unbindmsg](#)

sp_bindrule

| | |
|----|--|
| 说明 | 将规则绑定到列或用户定义的数据类型。 |
| 语法 | <code>sp_bindrule rulename, objname [, futureonly]</code> |
| 参数 | <p><i>rulename</i> 是规则的名称。用 <code>create rule</code> 语句创建规则，并用 <code>sp_bindrule</code> 将规则绑定到特定的列或用户定义的数据类型。</p> <p><i>objname</i> 是要向其绑定规则的表和列或用户定义的数据类型的名称。如果 <i>objname</i> 的形式不是 “<i>table.column</i>”，则它将被认为是用户定义的数据类型。如果对象名包含嵌入的空格或标点符号，或者对象名是保留字，则用引号将它引起来。</p> <p><i>futureonly</i> 禁止用户定义的数据类型的现有列继承新规则。当您 will 规则绑定到用户定义的数据类型时，此参数是可选的。如果将规则绑定到列，则此参数无意义。</p> |
| 示例 | <p>示例 1 假设已用 <code>create rule</code> 在当前数据库中创建了名为 <code>today</code> 的规则，此命令将其绑定到 <code>employees</code> 表的 <code>startdate</code> 列。当将行添加到 <code>employees</code> 时，将根据规则 <code>today</code> 检查 <code>startdate</code> 列的数据：</p> <pre>sp_bindrule today, "employees.startdate"</pre> <p>示例 2 假设存在名为 <code>rule_ssn</code> 的规则和名为 <code>ssn</code> 的用户定义的数据类型，此命令将 <code>rule_ssn</code> 绑定到 <code>ssn</code>。在 <code>create table</code> 语句中，类型为 <code>ssn</code> 的列继承规则 <code>rule_ssn</code>。除非以前已更改过 <code>ssn</code> 的规则，否则类型为 <code>ssn</code> 的现有列也继承规则 <code>rule_ssn</code>（在这种情况下，更改后的规则仅在以后维护）：</p> <pre>sp_bindrule rule_ssn, ssn</pre> <p>示例 3 将规则 <code>rule_ssn</code> 绑定到用户定义的数据类型 <code>ssn</code>，但类型为 <code>ssn</code> 的任何现有列都不受影响。<code>futureonly</code> 禁止类型为 <code>ssn</code> 的现有列继承该规则：</p> <pre>sp_bindrule rule_ssn, ssn, futureonly</pre> |
| 用法 | <ul style="list-style-type: none"> • 使用 <code>create rule</code> 语句创建规则。然后执行 <code>sp_bindrule</code> 将其绑定到当前数据库中的列或用户定义的数据类型。 • 当尝试执行 <code>insert</code>（而不是执行 <code>sp_bindrule</code> 时），将强制执行规则。您可以将字符规则绑定到数据类型为精确数值或近似数值的列，即使这样的 <code>insert</code> 并不合法。 • 不能使用 <code>sp_bindrule</code> 为 <code>create table</code> 语句中的列绑定检查约束。 • 不能将规则绑定到 Adaptive Server 提供的数据类型，或绑定到 <code>text</code> 或 <code>image</code> 列。 |

- 不能将规则绑定到系统表。
- 无法将规则绑定到计算列。
- 如果绑定到列，`objname` 参数的形式必须是“`table.column`”。任何其它格式都将被认为是用户定义的数据类型的名称。
- 使用规则的语句不能和它们的 `sp_bindrule` 语句位于同一个批处理中。
- 可以将规则绑定到列或用户定义的数据类型，而无需解除现有规则的绑定。绑定到列的规则始终比绑定到数据类型的规则具有更高的优先级。将规则绑定到列后，将替换已绑定到该列的数据类型的规则。但是，如果将规则绑定到数据类型，则不会替换已绑定到该用户定义的数据类型的列的规则。
- 除非以前更改过列的规则或者可选的第三个参数的值是 `futureonly`，否则用户定义的数据类型的现有列将继承新规则。用户定义的数据类型的新列始终继承该规则。

权限

只有对象所有者才能执行 `sp_bindrule`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|------|--------------------------|--|
| 8 | bind | <code>sp_bindrule</code> | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 规则名称 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 [create rule](#)、[drop rule](#)

系统过程 [sp_unbindrule](#)

sp_cacheconfig

说明 创建、配置、重新配置和删除数据高速缓存，并提供有关信息。

语法

```
sp_cacheconfig [cachename [, "cache_size[P | K | M | G]"
                [, logonly | mixed | inmemory_storage][, strict | relaxed]]
                [, "cache_partition=[1 | 2 | 4 | 8 | 16 | 32 | 64]"
                [, instance instance_name]
```

参数

cachename

是要创建或配置的数据高速缓存的名称。高速缓存名必须是唯一的，其长度最多可以是 30 个字符。高速缓存名不一定是有效的 Adaptive Server 标识符，也就是说，它可以包含空格和其它特殊字符。

cache_size

是要创建的数据高速缓存的大小；如果高速缓存已存在，则是该数据高速缓存的新大小。高速缓存的最小大小是服务器逻辑页大小的 256 倍。在指定大小单位时，可以用 P 表示页，用 K 表示千字节，用 M 表示兆字节，用 G 表示千兆字节。缺省值是 K。对于兆字节和千兆字节，可以指定浮点值。高速缓存大小是逻辑页大小的倍数。

logonly | mixed | inmemory_storage

指定高速缓存的类型。inmemory_storage 指示要为内存数据库或宽松持久性数据库创建高速缓存。

strict | relaxed

指定高速缓存替换策略。

cache_partition

指定在高速缓存中创建的分区数。

instance_name

（在集群环境中）是要调整其高速缓存的实例的名称。

示例

示例 1 创建具有 10 MB 空间的数据高速缓存 pub_cache。所有空间都在缺省逻辑页大小内存池中：

```
sp_cacheconfig pub_cache, "10M"
```

示例 2 报告 pub_cache 的当前配置和高速缓存中的所有内存池：

```
sp_cacheconfig pub_cache
```

示例 3 在下次启动 Adaptive Server 时删除 pub_cache：

```
sp_cacheconfig pub_cache, "0"
```

示例 4 使用单个步骤创建 pub_log_cache 并将其类型设置为 logonly：

```
sp_cacheconfig pub_log_cache, "2000K", logonly
```

示例 5 第一个命令创建缺省类型为 `mixed` 的高速缓存 `pub_log_cache`。第二个命令将其状态更改为 `logonly`。最终配置与示例 4 中的配置相同：

```
sp_cacheconfig pub_log_cache, "2000K"
sp_cacheconfig pub_log_cache, logonly
```

示例 6 创建高速缓存并设置大小、类型、替换策略和高速缓存分区数：
`sp_cacheconfig 'newcache', '50M', mixed, strict, "cache_partition=2"`

示例 7 创建一个名为 `pubs3_imdb` 的内存存储：

```
sp_cacheconfig pubs_imdb, '500M', inmemory_storage
```

示例 8 （在集群环境中）显示实例 `blade1` 的高速缓存：

```
sp_cacheconfig 'instance blade1'
```

示例 9 （在集群环境中）将 `blade1` 上的 `Sales Cache` 大小设置为 100 MB：

```
sp_cacheconfig 'Sales Cache', '100M', 'instance blade1'
```

示例 10 （在集群环境中）将 `blade1` 上的 `Sales Cache` 大小设置为 0 MB，从而在实际中删除该高速缓存：

```
sp_cacheconfig 'Sales Cache', '0M', 'instance blade1'
```

用法

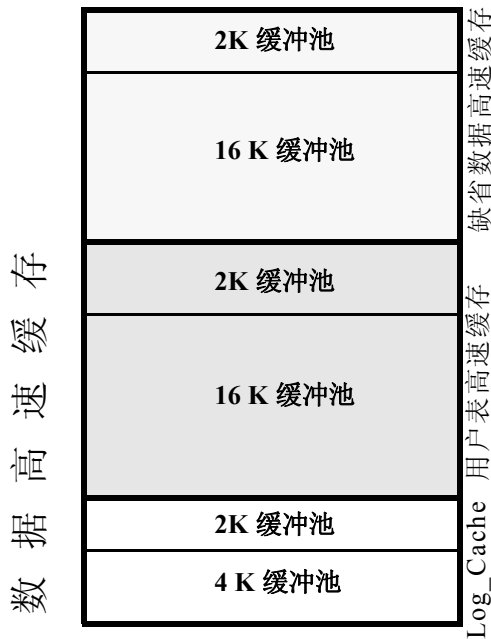
- 最小高速缓存大小是逻辑页大小的 256 倍。例如，4K 服务器的最小高速缓存大小为 1024K。
- 如果在您创建新高速缓存或将内存添加到现有高速缓存时 `Adaptive Server` 无法分配所有请求的内存，则它将分配所有可用内存。但是，这些附加内存是在下次重新启动 `Adaptive Server` 时分配的。
- 如果高速缓存（包括缺省高速缓存）上绑定有对象，则在解除对象的绑定之前不能删除该高速缓存。
- （在集群环境中）如果不指定 `instance_name`，则显示集群的高速缓存。
- 使用 `sp_cacheconfig` 执行的某些操作是动态的（不需要重新启动 `Adaptive Server`），而某些操作是静态的（需要重新启动）。表 1-5 介绍哪些操作是动态的，哪些操作是静态的：

表 1-5: 动态和静态 `sp_cacheconfig` 操作

| 动态 <code>sp_cacheconfig</code> 操作 | 静态 <code>sp_cacheconfig</code> 操作 |
|-----------------------------------|-----------------------------------|
| 添加新高速缓存 | 更改高速缓存分区的数目 |
| 将内存添加到现有高速缓存 | 减小高速缓存大小 |
| 删除高速缓存 | 更改替换策略 |
| 更改高速缓存类型 | |

- 当首次创建数据高速缓存时：
 - 所有空间都分配到逻辑页大小内存池。
 - 缺省类型是 `mixed`。
- 第 92 页的图 1-1 显示了 2K 服务器的数据高速缓存，它配置有两个用户定义的数据高速缓存以及以下缓冲池：
 - 缺省数据高速缓存包含一个 2 K 的缓冲池和一个 16 K 的缓冲池
 - 用户高速缓存包含一个 2 K 的缓冲池和一个 16 K 的缓冲池
 - 日志高速缓存包含一个 2 K 的缓冲池和一个 4 K 的缓冲池

图 1-1：带有缺省高速缓存和用户定义的高速缓存的数据高速缓存



- 缺省数据高速缓存的类型必须始终是 `default`，而任何其它高速缓存都不能具有类型 `default`。
- 在类型为 `logonly` 或具有宽松 LRU 替换策略的高速缓存中，Adaptive Server 管家任务不执行任何缓冲区清洗。

- 以下命令只执行 2K 的 I/O 操作：**disk init**、某些 **dbcc** 命令和 **drop table**。**dbcc checkdb** 和 **dbcc checktable** 命令可以对表执行大型 I/O 操作，但对索引只执行 2K 的 I/O 操作。表 1-6 显示了在绑定 / 不绑定数据库或对象时的高速缓存使用情况。

表 1-6: Transact-SQL 命令的高速缓存使用情况

| 命令 | 数据库已绑定 | 表或索引已绑定 | 数据库或对象未绑定 |
|---|---------|---------|-----------|
| create index | 绑定的高速缓存 | 不适用 | 缺省数据高速缓存 |
| disk init | 不适用 | 不适用 | 缺省数据高速缓存 |
| dbcc checkdb | 绑定的高速缓存 | 不适用 | 缺省数据高速缓存 |
| dbcc checktable 、 indexalloc 、 tablealloc | 绑定的高速缓存 | 绑定的高速缓存 | 缺省数据高速缓存 |
| drop table | 绑定的高速缓存 | 绑定的高速缓存 | 缺省数据高速缓存 |

- 恢复只使用缺省数据高速缓存的逻辑页大小缓冲池。必须回退或前进的所有事务的所有页都会被读入该缓冲池并在该缓冲池中进行更改。请确保缺省逻辑页大小缓冲池对这些事务来说足够大。
- 当使用不带任何参数的 **sp_cacheconfig** 时，它将报告有关服务器上所有高速缓存的信息。如果只指定了一个高速缓存名，则它只报告有关指定的高速缓存的信息。如果使用高速缓存名的一部分，它将报告符合 “%fragment%” 的所有名称的信息。

所有报告都包括一个报告高速缓存信息的信息区，并为每个高速缓存提供一个单独的数据区，以提供有关该高速缓存中缓冲池的信息。

以下来自使用 2K 的服务器的输出显示了下列配置：

- 缺省数据高速缓存具有两个缓冲池：一个 2K 的缓冲池和一个 16K 的缓冲池。缺省数据高速缓存具有两个分区。
- **pubs_cache** 具有两个缓冲池：2K 和 16K
- **pubs_log**，其类型设置为 **logonly**，高速缓存替换策略设置为 **relaxed**，包括一个 2 K 缓冲池和一个 4 K 缓冲池

```

Cache Name                Status      Type        Config Value  Run Value
-----
default data cache       Active     Default     0.00 Mb      26.09 Mb
pubs_cache                Active     Mixed       10.00 Mb     10.00 Mb
pubs_log                  Active     Log Only    2.40 Mb      2.40 M
-----
                                Total       12.40 Mb    38.49 Mb
=====
Cache: default data cache, Status: Active, Type: Default
      Config Size: 0.00 Mb, Run Size: 26.09 Mb
      Config Replacement: strict LRU, Run Replacement: strict LRU

```

```

        Config Partition:          2,      Run Partition:          2
IO Size  Wash Size Config Size  Run Size      APF Percent
-----
      2 Kb   3704 Kb    0.00 Mb    18.09 Mb     10
      16 Kb   1632 Kb    8.00 Mb     8.00 Mb     10
=====
Cache: pubs_cache,  Status: Active,  Type: Mixed
      Config Size: 10.00 Mb,  Run Size: 10.00 Mb
      Config Replacement: strict LRU,  Run Replacement: strict LRU
      Config Partition:          1,      Run Partition:          1
IO Size  Wash Size Config Size  Run Size      APF Percent
-----
      2 Kb   1228 Kb    0.00 Mb     6.00 Mb     10
      16 Kb    816 Kb    4.00 Mb     4.00 Mb     10
=====
Cache: pubs_log,  Status: Active,  Type: Log Only
      Config Size: 2.40 Mb,  Run Size: 2.40 Mb
      Config Replacement: relaxed LRU,  Run Replacement: relaxed LRU
      Config Partition:          1,      Run Partition:          1
IO Size  Wash Size Config Size  Run Size      APF Percent
-----
      2 Kb    206 Kb    0.00 Mb     1.01 Mb     10
      16 Kb    272 Kb    1.40 Mb     1.39 Mb     10

```

表 1-7 列出了输出中各列的含义:

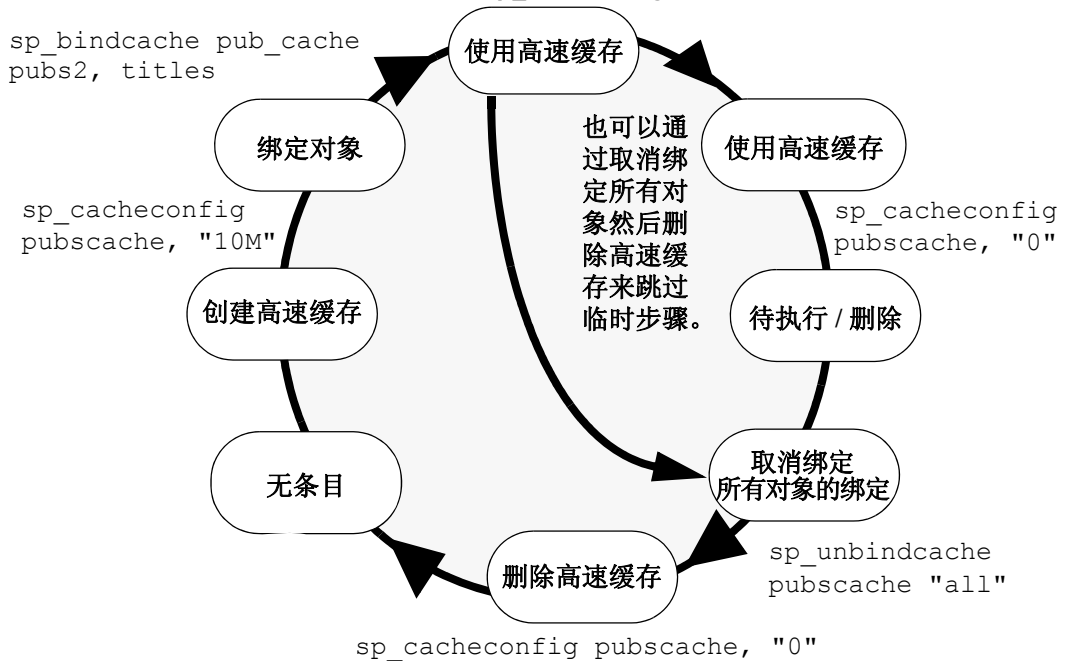
表 1-7: sp_cacheconfig 输出

| 列 | 含义 |
|----------------------------|---|
| Cache Name | 高速缓存的名称。 |
| Status | 下列值之一： <ul style="list-style-type: none"> “Active” “Pend/Act” “Pend/Del” 此表之后将对这些状态进行解释。 |
| Type | 用户定义的高速缓存为 “Mixed” 或 “Log Only”，缺省数据高速缓存为 “Default”。 |
| I/O Size | 内存池的 I/O 的大小。在显示高速缓存配置的行上，此列为空白。 |
| Wash Size | 缓冲池的清洗区的大小。当页进入高速缓存的清洗区时，它们将被写入磁盘。在显示高速缓存配置的行上，此列为空白。 |
| Config Value 或 Config Size | 高速缓存或缓冲池的大小。如果该值为 0，则表示尚未显式配置大小，将使用缺省值。 |
| Run Value 或 Run Size | 当前在 Adaptive Server 上使用的高速缓存或缓冲池的大小。 |

| 列 | 含义 |
|-------------------------|---|
| Config/ Run Replacement | 将在下次重新启动后用于高速缓存的高速缓存策略（严格或宽松），以及当前的替换策略。只有当策略在上次重新启动后被更改时，它们才会有所不同。 |
| Config/Run Partition | 将用于高速缓存的高速缓存分区数，以及当前的分区数。如果上次重新启动后使用 <code>sp_cacheconfig</code> 更改了分区数，则它们将有所不同。 |
| APF Percent | 在可容纳缓冲区的缓冲池中，已被异步预取但尚未被使用的缓冲区的百分比。 |
| Total | 数据高速缓存的总大小（如果报告包括所有高速缓存），或特定高速缓存的当前大小（如果指定了一个高速缓存名）。 |

状态“Pend”是“pending”（待执行）的缩写。它总是与“Act”（即 Active，活动的）或“Del”（即 Delete，删除）一起使用。“Pend”表示已发生配置操作，但必须重新启动服务器才能使更改生效。

图 1-2: 重新启动和 `sp_cacheconfig` 对高速缓存状态的影响



- 您还可以通过编辑配置文件配置高速缓存和缓冲池。有关详细信息，请参见《系统管理指南》。

数据高速缓存内存

- 当首次安装 Adaptive Server 时，所有数据高速缓存内存都被分配到名为 `default data cache` 的高速缓存的逻辑页大小缓冲池中。缺省数据高速缓存用于所有未用 `sp_bindcache` 显式绑定到数据高速缓存或其数据库未绑定到高速缓存的对象。
- 当创建数据高速缓存时，将根据 `max memory` 对内存分配进行验证。使用 `total logical memory` 配置参数，从分配给 Adaptive Server 的内存中分配高速缓存的内存。若要增加高速缓存可用的空间量，请增加 `total logical memory`，或减少其它使用内存的配置设置。如果 `total logical memory` 和请求的其它内存的和大于 `max memory`，则 Adaptive Server 发出一条错误并且不执行更改。

缺省高速缓存用于未绑定到其它高速缓存的所有对象（包括系统表），并且是唯一在恢复过程中使用的高速缓存。有关详细信息，请参见《系统管理指南》。

- 数据高速缓存要求将小部分开销用于管理高速缓存的结构。所有高速缓存开销都取自于可用内存。若要查看特定高速缓存大小所需的开销量，可以使用 `sp_helpcache` 并提供大小：

```
sp_helpcache "200M"
```

```
10.38Mb of overhead memory will be needed to manage
a cache of size 200M
```

这只是估计的开销。由于运行期间的问题，实际开销可能会大一些。

为内存数据库和宽松持久性数据库创建高速缓存

- 高速缓存名的长度不能大于 127 个字节。
- 内存存储高速缓存最小为 256 个逻辑页（在使用 2K 逻辑页的服务器上为 512 K）。
- 您不能：
 - 对内存存储包括 `strict` 或 `relaxed` 替换策略。缺省情况下，`sp_cacheconfig` 对内存存储高速缓存使用 `none` 替换策略。
 - 为内存存储高速缓存创建大 I/O 缓冲池（内存数据库不执行 I/O）。如果使用 `sp_poolconfig` 为内存存储高速缓存创建缓冲池，Adaptive Server 将发出错误。
 - 将高速缓存类型从 `mixed` 更改为 `logonly`（或者相反）。

更改现有的高速缓存大小

- 若要更改现有高速缓存的大小，请指定高速缓存的名称和新大小。
 - 如果增加现有高速缓存的大小，所有增加的空间都被放置在最小的缓冲池中。
 - 若要减少现有高速缓存的大小，所有空间必须在逻辑页大小缓冲池中可用。可能需要使用 `sp_poolconfig` 将空间从其它缓冲池移动到该缓冲池中。
- 如果有数据库或任何非日志对象绑定到高速缓存，则不能将其类型更改为 `logonly`。

使用高速缓存分区

- 利用高速缓存分区，可以减少高速缓存螺旋锁争用，而不需要创建单独的高速缓存并将数据库对象绑定到这些高速缓存。有关监控高速缓存螺旋锁争用的详细信息，请参见《性能和调优指南》。
- 您可以使用配置参数 `global cache partition number` 设置所有高速缓存的缺省高速缓存分区数。请参见《系统管理指南》。

删除高速缓存

- 若要删除数据高速缓存，请将其大小更改为 0，如示例 3 所示。当将高速缓存的大小设置为 0 时，会给该高速缓存标上删除标记。该高速缓存继续处于活动状态，绑定到该高速缓存的所有对象将继续使用它。

您不能删除缺省数据高速缓存。

- 如果删除数据高速缓存并且该高速缓存上有绑定的对象，则该高速缓存在内存中保持目前的状态，Adaptive Server 将发出以下消息：

```
Cache (nmc3) not deleted dynamically. Objects are bound to the cache. Use
sp_unbindcache_all to unbind all objects bound to the cache.
```

配置文件中对应于该高速缓存的条目和 `sysconfigures` 中对应于该高速缓存的条目都将被删除，而该高速缓存将在下次重新启动 Adaptive Server 时被删除。

- 不能在事务中运行 `sp_cacheconfig`。

权限

只有系统管理员才能执行 `sp_cacheconfig` 更改高速缓存配置。任何用户都可以执行 `sp_cacheconfig` 查看高速缓存配置。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_bindcache](#)、[sp_helpcache](#)、[sp_poolconfig](#)、[sp_unbindcache](#)、[sp_unbindcache_all](#)

sp_cachestrategy

说明 对表、索引、text 对象或 image 对象启用或禁用预取（大 I/O）和 MRU 高速缓存替换策略。

语法

```
sp_cachestrategy dbname, [ownername.]tablename
    [, indexname | "text only" | "table only"
    [, {prefetch | mru}, {"on" | "off"}]]
```

参数

dbname

是存储对象的数据库的名称。

ownername

是表所有者的名称。如果表由“dbo”所有，则所有者名称是可选的。

tablename

是表的名称。

indexname

是表上索引的名称。

text only

更改 text 或 image 对象的高速缓存策略。

table only

更改表的高速缓存策略。

prefetch | mru

是 prefetch 或 mru，并指定要更改的设置。

on | off

指定设置“on”或“off”（用引号引起来）。

示例

示例 1 显示有关 titles 表的高速缓存策略的信息：

```
sp_cachestrategy pubs2, titles

object name          index name          large IO  MRU
-----
dbo.titles           titleidind         ON        ON
```

示例 2 显示有关 titleind 索引的高速缓存策略的信息：

```
sp_cachestrategy pubs2, titles, titleind
```

示例 3 对 titles 表的 titleind 索引禁用预取：

```
sp_cachestrategy pubs2, titles, titleind, prefetch, "off"
```

示例 4 对 authors 表重新启用 MRU 替换策略：

```
sp_cachestrategy pubs2, authors, "table only", mru, "on"
```

示例 5 对 blurbs 表的文本页重新启用预取：

```
sp_cachestrategy pubs2, blurbs, "text only", prefetch, "on"
```

用法

- 如果为表或索引使用的高速缓存配置了大 I/O 的内存池，则优化程序可以选择通过一次执行最多 8 个数据页的大 I/O 预取数据或索引页。可以在表的数据页或非聚簇索引的叶级页上使用这种 **prefetch** 策略。缺省情况下，对所有表、索引和 **text** 或 **image** 对象启用预取。将 **prefetch** 选项设置为 “off” 会禁用指定对象的预取。
- 为了对任意大小的 I/O 执行表扫描和索引扫描，优化程序可以选择使用 **MRU 替换策略** 来读取和放弃高速缓存中的缓冲区。缺省情况下，对所有对象启用此策略。将 **mru** 设置为 “off” 会禁用此策略。如果对某个对象关闭 **mru**，则所有页都会被读入高速缓存中的 **MRU/LRU** 链，在被其它 I/O 刷新之前，它们会一直保留在高速缓存中。有关高速缓存策略的详细信息，请参见《性能和调优指南》。
- 只能更改当前数据库中对象的高速缓存策略。
- 如果在不指定策略和设置的情况下使用 **sp_cachestrategy**，它将报告对象的当前设置，如示例 1 所示。
- 若要查看服务器上所有数据高速缓存的大小、状态和 I/O 大小，请使用 **sp_cacheconfig**。
- 设置 **prefetch "on"** 对已读入只允许 2K I/O 的高速缓存的表或索引不起任何作用。无论可用 I/O 大小是多少，**mru** 策略都可用于所有高速缓存。

替换

- 如果启用了表或索引的预取，则可以用 **set prefetch "off"** 替换会话的预取。如果关闭了对象的预取，则不能替换该设置。
- **select**、**delete** 和 **update** 命令的 **prefetch**、**lru** 和 **mru** 选项建议各个语句的 I/O 大小和高速缓存策略。如果对表或索引启用了预取或 **MRU** 策略，则可以通过为 **prefetch** 指定逻辑页大小的 I/O 大小以及通过指定 **lru** 策略替换查询的预取或 **MRU** 策略。例如，以下命令对 **titles** 表强制 **LRU** 策略、逻辑页大小 I/O 和表扫描：

```
select avg(advance)
from titles (index titles prefetch 2 lru)
```

如果请求了预取大小，但没有为请求大小的 I/O 配置对象的高速缓存，则优化程序将选择最佳可用 I/O 大小。

- 如果用 `sp_cachestrategy` 为对象启用了预取，则在 `select`、`update` 或 `delete` 命令中指定逻辑页大小的 `prefetch` 将替换先前的 `set prefetch "on"` 语句。在 `select`、`update` 或 `delete` 命令中指定较大的 I/O 大小不会替换 `set prefetch "off"` 命令。

权限 只有系统管理员或对象所有者可以执行 `sp_cachestrategy`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 命令 `delete`、`select`、`set`、`update`

存储过程 `sp_cacheconfig`、`sp_poolconfig`

sp_changedbowner

| | |
|----|---|
| 说明 | 更改用户数据库的所有者。 |
| 语法 | sp_changedbowner <i>loginame</i> [, true] |
| 参数 | <i>loginame</i> 是当前数据库新所有者的登录名。 true 将别名及其权限移交给新数据库所有者。值为“true”和“TRUE”。 |
| 示例 | 使用用户“albert”成为当前数据库的所有者： sp_changedbowner albert |
| 用法 | <ul style="list-style-type: none"> 新所有者不能是已知的用户或别名（也就是说，新所有者不能已列在 sysusers 或 sysalternates 中）。执行带单个参数 <i>loginame</i> 的 sp_changedbowner 会将数据库所有权移交给 <i>loginame</i>，同时删除充当旧“dbo”的用户的别名。 在执行 sp_changedbowner 之后，新所有者便成为数据库中的数据库所有者。 sp_changedbowner 不能移交系统数据库的所有权。 新所有者必须已在 Adaptive Server 中具有登录名，但不能具有数据库中的数据库用户名或别名。若要将数据库所有权分配给这样一个用户，请在执行 sp_changedbowner 前删除用户名或别名条目。 若要将权限授予新所有者，系统管理员必须将权限授予数据库所有者，这是因为该用户在数据库中不再有任何其它名称。 |
| 权限 | 具有 sa_role 或 sso_role 特权的用户可以执行 sp_changedbowner。 sp_changedbowner 用于更改数据库的所有者。可以使用 sa_role 或 sso_role 权限来执行它。 |
| 审计 | sysaudits 表的 event 和 extrainfo 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色—当前活动角色 关键字或选项—NULL 以前值—NULL 当前值—NULL 其它信息—所有输入参数 代理信息—set proxy 有效时的初始登录名 |

另请参见

命令 create database

系统过程 sp_addlogin、sp_dropalias、sp_dropuser、sp_helpdb

sp_changegroup

| | |
|----|---|
| 说明 | 更改用户所在的组。 |
| 语法 | <code>sp_changegroup grpname, username</code> |
| 参数 | <p>grpname 是组的名称。该组必须已存在于当前数据库中。将“public”用作 grpname 时，因为“public”是关键字，所以应该用引号将它引起来。</p> <p>username 是要添加到组的用户的名称。该用户必须已存在于当前数据库中。</p> |
| 示例 | <p>示例 1 用户“albert”现在已是“fort_mudge”组的成员。“albert”以前属于什么组并不重要：</p> <pre>sp_changegroup fort_mudge, albert</pre> <p>示例 2 从“albert”所属的组中将其删除，并且不使其成为某个新组的成员（所有用户始终是“public”的成员）：</p> <pre>sp_changegroup "public", albert</pre> |
| 用法 | <ul style="list-style-type: none"> • 执行 <code>sp_changegroup</code> 可以将指定的用户添加到指定的组。先从该用户以前所属的组中将其删除，然后将其添加到 grpname 指定的组中。 • 在用 <code>sp_adduser</code> 为新数据库用户提供数据库访问权的同时，可以将这些用户添加到组中。 • 在授予和撤销特权时可使用组作为集合名称。每个用户始终是缺省组“public”的成员，除此之外仅能属于一个其它组。 • 若要将某个用户从组中删除而不使其成为新组的成员，请使用 <code>sp_changegroup</code> 将用户的组更改为“public”，如示例 2 中所示。 • 当某个用户从一个组更改到另一个组时，该用户将失去在旧组中拥有的所有权限，而获得向新组授予的权限。 |
| 权限 | 只有数据库所有者、系统管理员或系统安全员才能执行 <code>sp_changegroup</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[revoke](#)

系统过程 [sp_addgroup](#)、[sp_adduser](#)、[sp_dropgroup](#)、[sp_helpgroup](#)

sp_checknames

说明 检查当前数据库中是否存在包含 7 位 ASCII 字符集以外字符的名称。

语法 sp_checknames [help | silent]

参数 help
显示有关扫描的系统表的信息。

silent
在无提示模式下检查当前数据库，返回下列值之一：

- 0 — 如果不存在包含 7 位 ASCII 字符之外字符的名称，或者
- 1 — 如果至少存在一个包含 7 位 ASCII 字符之外字符的名称

示例 **示例 1** 检查 master 数据库中是否存在包含 7 位 ASCII 字符集以外字符的名称：

```
sp_checknames
Looking for non 7-bit ASCII characters in the system tables of database:
"master"
```

```
=====
Table.Column name: "syslogins.password"

The following logins have passwords that contain non 7-bit
ASCII characters.  If you wish to change them use "sp_password";
Remember, only the sa and the login itself may examine or change
the syslogins.password column:
```

```
  suid  name
-----
      1 sa
      2 probe
      3 bogususer
```

示例 2 显示有关扫描的系统表的信息：

```
1> sp_checknames help
2> go
```

```
sp_checknames is used to search for non 7-bit ASCII characters
several important columns of system tables. The following
columns are searched:
```

```
In "master":
  sysdatabases.name
  sysdevices.name
```

```

syslogins.name
syslogins.dbname
syslogins.password
sysremotelogins.remoteusername
syssservers.srvname
syssservers.srvnetname
    
```

```

In all databases:
syscolumns.name
sysindexes.name
sysobjects.name
syssegments.name
systypes.name
sysusers.name
    
```

```

(return status = 0)
1>
    
```

示例 3 禁止输出系统表名称，只显示返回状态：

```

1> sp_checknames silent
2> go

(return status = 1)
    
```

用法

- **sp_checknames** 检查当前数据库中的所有对象、列、索引、用户名、组名和其它元素的名称中是否包含 7 位 ASCII 字符集以外的字符。它将报告非法名称并提供使这些名称与 7 位 ASCII 字符集兼容的说明。
- 当从 SQL Server 版本 4.0.x 或 4.2.x 升级后或使用非 7 位 ASCII 的缺省字符集后，应该在服务器上的每个数据库中运行 **sp_checknames**。
- 根据 **sp_checknames** 报告中的说明更正所有非 ASCII 名称。

权限

任何用户都可以执行 **sp_checknames**。

审计

sysaudits 表的 **event** 和 **extrainfo** 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — set proxy 有效时的初始登录名 |

另请参见

命令 [update](#)

系统过程 [sp_password](#)、[sp_rename](#)、[sp_renamedb](#)

sp_checkreswords

- 说明** 检测并显示属于 Transact-SQL 保留字的标识符。检查服务器名、设备名、数据库名、段名、用户定义的数据类型、对象名、列名、用户名、登录名和远程登录名。
- 语法** sp_checkreswords [*user_name_param*]
- 参数** *user_name_param*
是当前数据库中用户的名称。如果提供 *user_name_param*，则 sp_checkreswords 将只检查指定用户拥有的对象。
- 示例** **示例 1** 如果 sp_checkreswords 在 master 数据库中执行，则显示结果：

```

1> /* executed in the master database */
2> sp_checkreswords

Reserved Words Used as Database Object Names for Database master

Upgrade renames sysobjects.schema to sysobjects.schemact.

Owner
-----
dbo

Table                                     Reserved Word Column Names
-----
authorization                             cascade

Object Type                               Reserved Word Object Names
-----
rule                                       constraint
stored procedure                          check
user table                                arith_overflow
user table                                authorization

-----
-----

Owner
-----
lemur

Table                                     Reserved Word Column Names
-----
key                                       close

Table                                     Reserved Word Index Names

```

```

-----
key                                isolation
Object Type                        Reserved Word Object Names
-----
default                            isolation
rule                               level
stored procedure                   mirror
user table                          key

Reserved Word Datatype Names
-----
identity

-----
-----

Database-wide Objects
-----

Reserved Word User Names
-----
at
identity

Reserved Word Login Names
-----
at
identity

Reserved Word as Database Names
-----
work

Reserved Word as Language Names
-----
national

Reserved Word as Server Names
-----
mirror
primary

Reserved Word ServerNetNames

```

```
-----  
mirror  
primary
```

示例 2 如果 sp_checkreswords 在用户数据库 user_db 中执行，则显示结果：

```
1> /* executed in the user database, user_db */  
2> sp_checkreswords
```

Reserved Words Used as Database Object Names for Database user_db

Upgrade renames sysobjects schema to sysobjects.schemacnt.

Owner

```
-----  
tamarin
```

| Table | Reserved Word Column Names |
|---------|----------------------------|
| ----- | ----- |
| cursor | current |
| endtran | current |
| key | identity |
| key | varying |
| schema | primary |
| schema | references |
| schema | role |
| schema | some |
| schema | user |
| schema | work |

| Table | Reserved Word Index Names |
|-------|---------------------------|
| ----- | ----- |
| key | double |

| Object Type | Reserved Word Object Names |
|------------------|----------------------------|
| ----- | ----- |
| default | escape |
| rule | fetch |
| stored procedure | foreign |
| user table | cursor |
| user table | key |
| user table | schema |
| view | endtran |

```
-----  
-----
```


Database-wide Objects

```
Found no reserved words used as names for database-wide objects.
```

用法

- **sp_checkreswords** 报告是保留字的现有对象的名称。除非使用分隔标识符，否则 Transact-SQL 不允许将任何命令语法某一部分的单词用作标识符。保留字是 SQL 语法的构成元素，当用作命令的一部分时，它们具有特殊含义。例如在 10.0 以前版本的 SQL Server 中，您可以具有一个名为 **work** 的表，并使用以下查询选择该表的数据：

```
select * from work
```

在 SQL Server 版本 10.0 中，**work** 是命令 **commit work** 的一部分，它是新增的保留字。在版本 10.0 或更高版本中发出相同的 **select** 语句会导致语法错误。**sp_checkreswords** 将查找会导致这些问题的标识符。

- **sp_checkreswords** 也会查找使用 **set quoted_identifier** 选项创建的、用作标识符的保留字。
- 在升级到新版本的 Adaptive Server 之前或紧接在升级之后使用 **sp_checkreswords**。有关在执行升级前安装和运行此过程的信息，请参见所用平台的安装文档。

在 **master** 数据库和每个用户数据库中运行 **sp_checkreswords**。另外，如果已将用户或对象添加到 **model** 和 **sysystemprocs** 中，还应在这些数据库中运行此过程。

- 返回状态将指示找到的项目数。
- 如果提供了用户名，**sp_checkreswords** 将检查所有可由用户表、索引、视图、过程、触发器、规则、缺省值和用户定义的数据类型拥有的对象。它将报告所有是保留字的标识符。
- 如果当前数据库不是 **master** 数据库且没有提供用户名，则 **sp_checkreswords** 将检查上述所有对象，并在报告中为每个用户名提供一个单独的部分。它还检查 **sysusers** 和 **syssegments** 中是否有用户名和段名是保留字。如果已添加了对象、用户或用户定义的数据类型，则只需要检查 **model** 和 **sysystemprocs**。
- 如果当前数据库是 **master** 且没有提供用户名，则 **sp_checkreswords** 将执行上述所有检查并检查 **sysdatabases**、**syslogins**、**syscharsets**、**syssservers**、**sysremotelogins**、**sysdevices** 和 **syslanguages** 中是否有保留字用作数据库、本地或远程登录、本地和远程服务器、字符集和语言的名称。

处理保留字的报告实例

- 如果 `sp_checkreswords` 报告有保留字用作标识符，则您可以执行以下两项操作之一：
 - 使用 `sp_rename`、`sp_renamedb` 或更新系统表以更改标识符的名称。
 - 如果保留字是表名、视图名或列名，则使用 `set quoted_identifier on`。如果您的大多数应用程序都使用存储过程，则可以删除然后用 `set quoted_identifier on` 重新创建这些过程并用引号将所有标识符引起来。所有用户都将能够运行这些过程，而无需对会话使用 `set quoted_identifier on`。您可以使用 `set quoted_identifier on`，创建为表或列提供替代名称的视图，然后更改您的应用程序以引用这些视图。

以下示例提供了新保留字 “key”、“level” 和 “work” 的替代名称：

```
create view keyview
as
select lvl = "level", wrk = "work"
from "key"
```

`set` 命令的语法是：

```
set quoted_identifier on
```

- 如果未更改标识符或使用分隔标识符，则任何将保留字用作标识符的查询都会报告错误（通常是语法错误）。例如：

```
select level, work from key
Msg 156, Level 15, State 1:
Server 'rosie', Line 1:
Incorrect syntax near the keyword 'level'.
```

注释 带引号的标识符选项是 SQL92 选项，许多支持其它 Adaptive Server 功能的客户端产品可能不支持该选项。例如，不能在名称为保留字的表上使用 `bcp`。

在选择带引号的标识符选项之前，应使用所有将用于访问 Adaptive Server 的工具对各种对象执行测试。使用 `set quoted_identifier on` 创建一个表名是保留字，各列名也是保留字的表。如果客户端产品生成 SQL 代码，则它就必须用双引号将标识符（如果它们是保留字）引起来，并用单引号将字符常量引起来。

- 如果过程、触发器和视图所依赖的对象已经更改了名称，则在名称更改之后，这些过程、触发器和视图可能会继续工作，但它们将在重新编译查询计划后停止工作。进行重新编译的原因有多种，并且不会通知用户。若要避免意外的功能损失，请在更改对象名之后立即在过程、触发器和视图中更改对象的名称。
- 不管是更改对象名还是使用分隔标识符，您必须更改包含保留字的所有存储过程、视图、触发器和应用程序。如果更改对象名，则必须更改标识符；如果使用分隔标识符，则必须添加 `set quoted_identifier` 选项和引号。
- 如果没有将过程、触发器、视图、规则和缺省值的文本保存在操作系统文件中，则可以使用 `defncopy` 将定义从服务器复制到文件中。请参见《实用程序指南》中的 `defncopy`。

更改标识符

- 如果更改 `sp_checkreswords` 所报告的项目的名称，则必须更改所有引用了使用保留字的对象的过程、触发器、视图和应用程序中的名称。
- 在更改标识符名称之前转储数据库。在更改标识符名称之后，运行 `dbcc` 以确定没有问题，然后再次转储数据库。
- 如果正在活动的生产数据库上更改标识符：
 - 则应在系统最空闲时执行更改，以便将对用户的干扰降到最低的程度。
 - 应通过查找使用特定标识符的所有 Open Client DB-Library™ 程序、窗口应用程序、存储过程、触发器和脚本进行精心准备。这样，您可以在源代码中进行所需的编辑，然后尽可能快地更改标识符并替换过程和代码。
- 过程 `sp_depends` 有助于查找使用表名和视图名的过程、视图和触发器。

使用 `sp_rename` 更改标识符

- 系统过程 `sp_rename` 可以重命名表、索引、视图、过程、触发器、规则、缺省值、用户定义的数据类型和列。使用 `sp_renamedb` 可以重命名数据库。
- 表 1-8 显示了可以用 `sp_rename` 更改的标识符的类型，并列出了可能需要在服务器和应用程序中进行的其它更改。

表 1-8: sp_rename 和更改标识符

| 标识符 | 务必要 |
|--------------|--|
| 表名 | <ul style="list-style-type: none"> 删除所有引用该表的过程、触发器和视图，然后用新名称重新创建它们。使用 <code>sp_depends</code> 查找依赖于该表的对象。 更改所有引用该表的应用程序或 SQL 源脚本以使用新表名。 更改使用表名执行表级检查的 <code>dbcc</code> 脚本。 |
| 索引名 | <ul style="list-style-type: none"> 删除任何创建或删除该索引的存储过程，然后用新名称重新创建它们。 更改所有创建或删除该索引的应用程序或 SQL 源脚本。 更改使用索引名执行索引级检查的 <code>dbcc</code> 脚本。 |
| 视图名 | <ul style="list-style-type: none"> 删除所有引用该视图的过程、触发器和视图，然后用新名称重新创建它们。使用 <code>sp_depends</code> 查找依赖于该视图的对象。 更改所有引用该视图的应用程序或 SQL 源脚本以使用新视图名。 |
| 过程名称 | <ul style="list-style-type: none"> 删除引用该过程的所有过程和触发器，然后用新过程名称重新创建它们。 更改执行该过程的所有应用程序或 SQL 源脚本以使用新名称。 如果另一个服务器远程调用该过程，则更改远程服务器上的应用程序以使用新过程名称。 |
| 触发器名 | <ul style="list-style-type: none"> 更改所有创建该触发器的 SQL 源脚本。 |
| 规则名 | <ul style="list-style-type: none"> 更改所有创建该规则的 SQL 源脚本。 |
| 缺省名称 | <ul style="list-style-type: none"> 更改所有创建该缺省值的 SQL 源脚本。 |
| 用户定义的数据类型的名称 | <ul style="list-style-type: none"> 删除所有使用用户定义的数据类型来创建表的过程，然后用新名称重新创建它们。 更改所有使用用户定义的数据类型创建表的应用程序。 |
| 列名 | <ul style="list-style-type: none"> 删除所有引用该列的过程、触发器和视图，然后用新列名重新创建它们。 <code>sp_depends</code> 无法找到列名引用。以下查询显示引用了名为 “key” 的列的过程、触发器和视图的名称： <pre>select distinct sysobjects.name from sysobjects, syscomments where sysobjects.id = syscomments.id and syscomments.text like "%key%"</pre> 更改所有通过名称引用该列的应用程序和 SQL 源脚本。 |

以下命令将视图 `isolation` 的名称更改为 `isolated`：

```
sp_rename "isolation", isolated
```

以下命令更改已重命名的视图 `isolated` 中某一列的名称：

```
sp_rename "isolated.key", keyname
```

- 使用 `sp_depends` 可以列出引用了将被重命名的视图、过程或表的所有视图、过程和触发器。若要在重命名对象后使用 `sp_depends`，请提供新名称。例如：

```
sp_depends new_name
```

用 `sp_renamedb` 重命名数据库

若要更改数据库的名称，请使用 `sp_renamedb`。该数据库必须处于单用户模式。删除并重新创建显式引用该数据库名称的所有过程、触发器和视图。有关详细信息，请参见 `sp_renamedb`。

更改其它标识符

- 若要更改用户名、登录名、设备名、远程服务器名、远程服务器用户名、段名、字符集名和语言名，首先需要确定是否可以删除然后重新添加或创建该对象或用户。如果不能这样做，则使用以下命令以允许对系统表进行直接更新：

```
sp_configure "allow updates to system tables", 1
```

只有系统安全员才能设置 `allow updates to system tables` 配置参数。

直接更新系统表时产生的错误可能会在 Adaptive Server 中造成严重问题。若要确定是否可以删除然后重新创建对象或用户，请参见表 1-9。

第 117 页的表 1-11 显示对这组标识符可能具有的依赖性。无论您选择的升级方法是删除再重新创建对象、使用分隔标识符还是对系统表执行直接更新，都请查看此表以了解可能的依赖性。

表 1-9：更改标识符时直接更新系统表的替代方法

| 标识符类型 | 为避免更新系统表而推荐的操作 |
|----------|---|
| 用户名和登录名 | 若要更改没有对象的用户的名称，请首先在每个数据库中使用 <code>sp_helprotect username</code> 来记录该用户的权限。然后，从所有数据库中删除该用户 (<code>sp_dropuser</code>) 并删除登录名 (<code>sp_droplogin</code>)。最后，添加新登录名 (<code>sp_addlogin</code>) 并将新用户名添加到数据库 (<code>sp_adduser</code>)，然后用 <code>grant</code> 恢复用户的权限。 |
| 设备名 | 如果此设备已被完全分配，则无需在 <code>create database</code> 命令中使用其名称，因此可以保持该名称不变。 |
| 远程服务器名 | 除非远程服务器中有大量的远程登录名，否则应删除远程服务器 (<code>sp_dropserver</code>)，然后用新名称添加该服务器 (<code>sp_addserver</code>)。 |
| 远程服务器登录名 | 用 <code>sp_dropremotelogin</code> 删除远程登录，使用 <code>sp_addremotelogin</code> 以新名称添加该登录，然后用 <code>grant</code> 恢复用户执行过程的权限。 |
| 段名 | 在段上创建对象之后，很少会使用段名。 |
| 字符集名和语言名 | 只有当系统管理员用 <code>sp_addlanguage</code> 创建替代语言后，语言和字符集才会将保留字用作标识符。用 <code>sp_droplanguage</code> 删除该语言，然后用新名称添加它。 |

警告！ 直接更新系统表可能非常危险。您可能出现严重的错误，使 Adaptive Server 无法运行或无法访问数据库中的对象。若要直接更新系统表，您应该在心平气和、精力集中，并且服务器上没有任何（或只有很少的）生产活动时进行。如果可能，请使用表 1-9 中介绍的替代方法。

- 下面的示例显示一个“安全的”用户名更新过程，在所有数据修改之前是一个 `begin transaction` 命令。系统安全员可以执行以下命令：

```
sp_configure "allow updates to system tables", 1
```

然后可以执行以下命令：

```
begin transaction
update sysusers
set name = "workerbee"
where name = "work"
```

此时运行查询，检查以确保该命令只影响要更改的行。唯一可以影响多个行的标识符更改是更改 `syslogins` 中的 `language` 名称。如果查询影响了：

- 仅正确的行 — 则使用 `commit transaction`。
- 多个行或不正确的行 — 则使用 `rollback transaction`，确定问题的原因，然后正确地执行该命令。

完成之后，系统安全员可以使用以下命令关闭 `allow updates to system tables` 配置参数：

```
sp_configure "allow updates to system tables", 0
```

警告！ 只更新每个用户定义的事务中单个数据库的系统表。不要在发出 `begin transaction` 命令后更新多个数据库中的表。这样的操作会使恢复极为困难。

表 1-10 显示了更改保留字时应该更新的系统表和列。以“`master.dbo.`”为前缀的表只能出现在 `master` 数据库中。所有其它表出现在 `master` 和用户数据库中。在尝试更新之前，请确保使用的是正确的数据库。您可以使用以下命令检查当前数据库名：

```
select db_name()
```

表 1-10: 更改标识符时要更新的系统表列

| 标识符类型 | 要更新的表 | 列名 |
|----------|---|------------------|
| 用户名 | sysusers | name |
| 登录名 | master.dbo.syslogins | name |
| 段名 | syssegments | name |
| 设备名 | sysdevices | name |
| 远程服务器名 | syssservers | srvname |
| 远程服务器网络名 | syssservers | srvnetname |
| 字符集名 | master.dbo.syscharsets | name |
| 语言名 | master.dbo.syslanguages master.dbo.syslogins | name language |

表 1-11 显示了需要在服务器和应用程序中进行的其它更改:

表 1-11: 更改标识符时的注意事项

| 标识符 | 务必要 |
|----------|--|
| 登录名 | 更改该用户在其每个数据库中的用户名。 |
| 用户名 | 删除、编辑和重新创建对该用户拥有的对象使用限定 (<i>owner_name.object_name</i>) 引用的所有过程、触发器和视图。更改使用限定对象名的所有应用程序和 SQL 源脚本以使用新用户名。您无需删除这些对象本身: <code>sysusers</code> 已按照存储用户 ID (而不是用户名) 的列链接到 <code>sysobjects</code> 。 |
| 设备名 | 更改引用该设备名的所有 SQL 源脚本或应用程序以使用新名称。 |
| 远程服务器名 | 更改远程服务器上的名称。如果 <code>sp_checkreswords</code> 报告的名称是本地服务器的名称, 则在发出或接收远程过程调用之前必须先重新启动服务器。 |
| 远程服务器网络名 | 在 <code>interfaces</code> 文件中更改服务器的名称。 |
| 远程服务器登录名 | 更改远程服务器上的名称。 |
| 段名 | 删除并重新创建所有在该段名上创建表或索引的过程。更改所有在段上创建对象的应用程序以使用新段名。 |
| 字符集名 | 无。 |
| 语言名 | 更改 <code>master.dbo.syslanguages</code> 和 <code>master.dbo.syslogins</code> 。对 <code>syslogins</code> 的更新可能涉及多个行。另外, 请更改本地化文件的名称。 |

使用分隔标识符

- 可以将分隔标识符用于表名、列名和视图名。不能将分隔标识符用于其它对象名。
- 如果选择使用分隔标识符, 则使用 `set quoted_identifier on`, 然后删除并重新创建所有使用该标识符的过程、触发器和视图。编辑这些对象的文本, 将保留字用双引号引起来, 将所有字符串用单引号引起来。

下面的示例显示为使用分隔标识符而对查询进行的更改。此示例更新名为 **work** 的表，该表包括名为 **key** 和 **level** 的列。下面列出的是 10.0 以前版本的查询（它将字符文字用双引号引起来），以及为使用分隔标识符而对该查询进行编辑后的版本：

```
/* pre-release 10.0 version of query */
update work set level = "novice"
    where key = "19-732"
/* 10.0 or later version of query, using
** the quoted identifier option
*/
update "work" set "level" = 'novice'
    where "key" = '19-732'
```

- 必须对所有将保留字用作标识符的应用程序进行如下更改：
 - 应用程序必须启用带引号标识符选项。
 - 每次使用保留字都必须用双引号引起来。
 - 在启用带引号标识符选项时，应用程序使用的所有字符文字都必须用单引号引起来。否则，Adaptive Server 会尝试将它们解释为对象名。

例如，以下查询会导致出现错误消息：

```
set quoted_identifier on
select * from titles where title_id like "BU%"
```

以下是正确的查询：

```
select * from titles where title_id like 'BU%'
```

- 在分隔标识符生效时创建的存储过程可以在不打开该选项的情况下运行。（**allow updates to system tables** 选项的工作方式也是这样。）这意味着您可以打开带引号标识符模式，删除存储过程，编辑该过程以在用作标识符的保留字两边插入引号，然后重新创建该过程。所有用户无需使用 **set quoted_identifier** 即可执行该过程。

权限

只有系统管理员才能执行 **sp_checkreswords**。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 [set](#)

系统过程 [sp_configure](#)、[sp_depends](#)、[sp_rename](#)、[sp_renamedb](#)

实用程序 [defncopy](#)

sp_checksource

说明 检查已编译对象的源文本以及计算列的源文本是否存在。编译对象可以是谓词的用户定义名称，或者如果谓词没有外部名称，则是其内部名称。

语法 sp_checksource [objname [, tabname [, username]]]

参数

objname

是要检查其源文本是否存在的已编译对象。

tabname

是要检查在其上定义的所有检查约束、缺省值和触发器是否存在的表或视图的名称。

username

是拥有要检查源文本是否存在的已编译对象的用户的名称。

示例

示例 1 检查当前数据库中所有已编译对象的源文本是否存在：

```
sp_checksource
```

示例 2 检查名为 `titleview` 的视图的源文本是否存在：

```
sp_checksource titleview
```

示例 3 检查 Mary 拥有的名为 `titls_vu` 的视图的源文本是否存在：

```
sp_checksource title_vu, @username = Mary
```

示例 4 检查自定义存储过程 `list_phone_proc` 的源文本是否存在：

```
sp_checksource list_phone_proc
```

示例 5 检查在名为 `my_tab` 的表上定义的所有检查约束、触发器和声明缺省值的源文本是否存在：

```
sp_checksource @tabname = "my_tab"
```

示例 6 检查视图 `my_vu` 和在 `my_tab` 表上定义的所有检查约束、触发器和缺省值的源文本是否存在：

```
sp_checksource @objname = "my_vu", @tabname = "my_tab"
```

示例 7 检查 Tom 拥有的所有已编译对象的源文本是否存在：

```
sp_checksource @username = "Tom"
```

示例 8 检查 “pred1” 谓词的源文本是否存在：

```
sp_checksource pred1
```

```
Msg 18404, Level 16, State 1:  
Procedure 'sp_aux_text', Line 265:  
Source text for compiled object pred1 (id = 592002109  
exists)
```

- 用法**
- `sp_checksource` 检查指定的已编译对象的源文本是否存在。如果指定的对象存在源文本，则 `sp_checksource` 返回 0。如果指定的对象不存在源文本，则 `sp_checksource` 返回 1。
 - 如果不提供任何参数，则 `sp_checksource` 检查当前数据库中所有已编译对象的源文本是否存在。
 - 若要使用不带任何参数的 `sp_checksource`，您必须是数据库所有者或系统管理员。
 - `sp_checksource` 用于对用户定义的函数的文本进行加密。

权限 只有数据库所有者或系统管理员才能执行 `sp_checksource` 来检查另一个用户拥有的已编译对象的源文本是否存在。任何用户都可以执行 `sp_checksource` 来检查其所拥有的已编译对象的源文本是否存在。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_hidetext](#)

sp_chgattribute

| | |
|----|--|
| 说明 | 为以后分配表或索引的空间而更改 <code>max_rows_per_page</code> 、 <code>fillfactor</code> 、 <code>reservepagegap</code> 或 <code>exp_row_size</code> 值；设置表的 <code>concurrency_opt_threshold</code> 。提供优化索引锁定的用户界面。 |
| 语法 | <pre>sp_chgattribute <i>objname</i>, {"max_rows_per_page" "fillfactor" "reservepagegap" "exp_row_size" "concurrency_opt_threshold" "optimistic_index_lock" "identity_burn_max" "plldegree"} , <i>value</i>, <i>optvalue</i> sp_chgattribute <i>objname</i>, {"identity_gap", <i>set_number</i> "dealloc_first_txtpg", <i>value</i>}</pre> |
| 参数 | <p><i>objname</i> 是要更改其属性的表或索引的名称。</p> <p><code>max_rows_per_page</code> 指定行大小。对具有可变长度列的表使用此选项。</p> <p><code>fillfactor</code> 指定在运行 <code>reorg rebuild</code> 命令或 <code>alter table</code> 命令更改锁定方案而导致重新创建索引或复制表页时，Adaptive Server 对每页的填充程度。 <code>fillfactor</code> 百分比仅在重建索引时才有意义。有效值为 0 到 100。</p> <p><code>reservepagegap</code> 指定填充页与空白页的比率，扩充 I/O 分配操作过程中要保留该比率的空白页。对于每个指定的 <code>num_pages</code>，都会预留一个空白页供表将来扩展之用。有效值为 0 到 255。缺省值为 0。</p> <p><code>exp_row_size</code> 为仅数据锁定表中的行保留指定数量的空间。使用此选项可以减少转移的行数，更新期间转移行是很昂贵的。有效值可以是 0、1 以及介于表的最小和最大行长度之间的任何值。0 表示应用全服务器范围的设置，1 表示完全填满数据页上的行。</p> <p><code>concurrency_opt_threshold</code> 指定表大小（以页为单位），对仅数据锁定表的访问应该从该大小值开始优化，以减小 I/O（而不是为了并发）。如果表小于 <code>concurrency_opt_threshold</code> 指定的页数，则通常使用可用的索引来为并发进行查询优化；如果表大于 <code>concurrency_opt_threshold</code> 指定的页数，则为 I/O 进行查询优化。有效值为 -1 到 32767。将该值设置为 0 将禁用并发优化。使用 -1 可以对大于 32767 页的表强制并发优化。缺省值为 15 页。</p> |

optimistic_index_lock

启用可消除索引根页中的争用的性能优化。如果由于索引拆分而必须更改根页，则会获取排它表。因此，**optimistic_index_lock** 适用于修改数较小的表。有效值包括 1 和 0，1 表示打开优化索引锁，0 表示关闭优化索引锁，0 是缺省值。

identity_burn_max

允许设置表的 identity burn 最大值。此参数使用 **varchar** 数据类型。

identity_gap

表示要更改标识间距。

value

是您在 **sp_chgattribute** 中指定的各个选项的数字输入值。

optvalue

是新值。有效值和缺省值取决于指定的参数。只有 **identity_burn_max** 参数使用此参数。对于其它参数，此值为 **NULL**。

set_number

是标识间距的新大小。

dealloc_first_txtpg

将文本或图像列更新为 **null**。释放以前引用的 **text** 或 **image** 页之后，将对应的文本指针设置为 **null**。这样将会减少对空 **text/images** 列的空间分配。有效值包括缺省值 0，它表示在 **null** 更新时不释放 **text** 或 **image** 页，有效值还包括 1，它将释放设置为开。

plldegree

指定查询优化程序可以使用的最大线程数。

示例

示例 1 为 **authors** 表将 **max_rows_per_page** 设置为 1，以用于以后的所有空间分配：

```
sp_chgattribute authors, "max_rows_per_page", 1
```

示例 2 为 **titleidind** 索引将 **max_rows_per_page** 设置为 4，以用于以后的所有空间分配：

```
sp_chgattribute "titles.titleidind", "max_rows_per_page", 4
```

示例 3 为 **title_ix** 中的页将 **fillfactor** 指定为 90%：

```
sp_chgattribute "titles.title_ix", "fillfactor", 90
```

示例 4 为 **authors** 表将 **exp_row_size** 设置为 120，以用于以后的所有空间分配：

```
sp_chgattribute "authors", "exp_row_size", 120
```

示例 5 为 titleidind 索引将 reservepagegap 设置为 16，以用于以后的所有空间分配：

```
sp_chgattribute "titles.titleidind", "reservepagegap", 16
```

示例 6 关闭 titles 表的并发优化：

```
sp_chgattribute "titles", "concurrency_opt_threshold", 0
```

示例 7 将 mytable 的标识间距设置为 20：

```
sp_chgattribute "mytable", "identity_gap", 20
```

示例 8 更改 mytable 以使用 identity burning set factor 设置而不是 identity_gap 设置：

```
sp_chgattribute "mytable", "identity_gap", 0
```

将 sp_chgattribute 的值设置为 1，打开优化索引锁定功能。

```
sp_chgattribute "mytable", "optimistic_index_lock", 1
```

将 sp_chgattribute 的值设置为 0，关闭优化索引锁定功能。

```
sp_chgattribute "mytable", "optimistic_index_lock", 0
```

示例 9 使用 dealloc_first_txdpg 打开文本和图像空间的释放：

```
sp_chgattribute "mytable", "dealloc_first_txdpg", 1
```

若要关闭此功能：

```
sp_chgattribute "mytable", "dealloc_first_txdpg", 0
```

示例 10 将 authors 表的 identity_burn_max 值更改为 5：

```
sp_chgattribute "authors", "identity_burn_max", 0, 5
```

示例 11 指示查询优化程序最多使用 4 个线程：

```
sp_chgattribute my_table, "plldegree", 4
```

如果查询优化程序未找到足够的资源，可选择使用较少的线程（即少于 4 个）。此机制同样适用于索引。例如，以下示例使用 authors 上名为 auth_ind 的索引以使用两个线程对它进行访问：

```
sp_chgattribute "authors.auth_ind", "plldegree", 4
```

必须从当前数据库中运行 sp_chgattribute。

用法

- sp_chgattribute 更改 max_rows_per_page、fillfactor、reservepagegap、exp_row_size 或 dealloc_first_txdpg 的值，以用于以后对表或索引进行空间分配或数据修改。它不影响现有数据页的空间分配。您只能为当前数据库中的对象更改这些值。

- 使用 `sp_help` 查看表的存储空间管理值。使用 `sp_helpindex` 查看索引的存储空间管理值。
- 将 `max_rows_per_page` 设置为 0 通知 Adaptive Server 填充数据或索引页且不限制行数（这是 Adaptive Server 在未设置 `max_rows_per_page` 时的缺省行为）。
- 将 `sysobjects` 中存储的 `identity_burn_max` 值和当前标识值设置为新值。
- 如果表非空，则要求 `identity_burn_max` 的新值大于或等于标识列的当前最大值。如果表为空表，则可将该值设置为有效范围内的任意正值。
- `max_rows_per` 页值较小会导致页面拆分。当需要在页中添加新的数据或索引行，但页中没有足够的空间用于新行时，就会发生页面拆分。通常，现有页上的数据会相当平均地在新分配的页和现有页之间拆分。若要大概算出非聚簇索引的最大值，可将页大小减去 32，然后将所得的值除以索引键大小。以下语句计算非聚簇索引 `titleind` 的 `max_rows_per_page` 的最大值：

```
select
    (select @@pagesize - 32) / minlen
    from sysindexes where name = "titleind"
-----
                288
```

- 如果为 `max_rows_per_page`、`fillfactor`、`reserverpagegap` 或 `exp_row_size` 指定了不正确的值，则 `sp_chgattribute` 将返回一条错误消息，其中指定了有效值。
- 有关 `max_rows_per_page`、`fillfactor`、`reserverpagegap`、`exp_row_size` 和 `concurrency_opt_threshold` 的详细信息，请参见《性能和调优指南》。
- 有关标识间隔的详细信息，请参见《Transact-SQL 用户指南》中第 7 章“创建数据库和表”的“管理表中的标识间隔”一节。
- 不能在事务中运行此存储过程。
- 只有具有 `sa_role` 特权的用户才能执行此存储过程。
- 不能为具有数据页或数据行锁定方案的表设置优化索引锁定。
- 不能为系统数据库（例如 `master` 或 `tempdb`）中的表设置优化索引锁定选项。只能在用户定义的表上设置该选项。
- 即使在执行 NULL 更新时，也会为 `text` 和 `image` 页分配空间。可以使用 `dealloc_first_txtpg` 从表中删除这些空的文本页。
对列进行新的更新会导致重新分配 `text` 或 `image` 页。

权限 只有对象所有者才能执行 sp_chgattribute。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [alter table](#)、[create index](#)、[create table](#)
系统过程 [sp_helpindex](#)

sp_cleanpwdchecks

| | |
|----|---|
| 说明 | <code>sp_cleanpwdchecks</code> 是一个自定义存储过程，允许您定义何时及如何删除存储在用户定义表中的、与登录名和口令相关的属性。 |
| 语法 | <code>sp_cleanpwdchecks, login_name</code> |
| 参数 | <i>login_name</i> 指定要清除的登录名。 |
| 用法 | <code>sp_cleanpwdchecks</code> 是用户定义过程，在删除登录名时系统会在 <code>master</code> 数据库中动态调用该过程。 |

sp_clearpsexex

| | |
|----|--|
| 说明 | 清除由 sp_setpsexex 设置的 Adaptive Server 会话的执行属性。 |
| 语法 | <code>sp_clearpsexex spid, exeattr</code> |
| 参数 | <p><i>spid</i> 是要清除执行属性的会话的进程 ID。</p> <p><i>exeattr</i> 确定要清除的执行属性。<i>exeattr</i> 的值为 “priority” 和 “enginegroup”。</p> |
| 示例 | <p>删除进程 12 的引擎组条目。</p> <pre>sp_clearpsexex 12, 'enginegroup'</pre> |
| 用法 | <ul style="list-style-type: none"> • 如果未在会话的生存期间清除执行属性，则将在会话异常退出或终止时清除它们。 • 如果关联的引擎组中没有联机引擎，则 <code>sp_clearpsexex</code> 将失败。 • 当删除引擎组条目后，会话将在类定义或缺省类所确定的引擎组上执行。 • 使用 sp_who 可以列出进程 ID (spids)。 |
| 权限 | 只有系统管理员才能执行 <code>sp_clearpsexex</code> 来清除所有用户的优先级属性。任何用户都可以执行 <code>sp_clearpsexex</code> 来清除该用户拥有的任务的优先级属性。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [文档](#) 《性能和调优指南》

系统过程 [sp_addexclass](#)、[sp_bindexclass](#)、[sp_dropexclass](#)、[sp_showexclass](#)、[sp_unbindexclass](#)

sp_clearstats

说明 为所有服务器用户或指定的用户开始新的会计周期。执行 `sp_reportstats` 可以输出上一个周期的统计信息。

语法 `sp_clearstats [loginame]`

参数 *loginame*
是用户的登录名。

示例 1 为所有用户开始新的帐户使用周期。

```
sp_clearstats
Name      Since          CPU  Percent CPU    I/O    Percent I/O
-----  -
probe    Jun 19 1990      0      0%             0      0%
julie    Jun 19 1990    10000    24.9962%      5000    24.325%
jason    Jun 19 1990    10002    25.0013%      5321    25.8866%
ken      Jun 19 1990    10001    24.9987%      5123    24.9234%
kathy    Jun 19 1990    10003    25.0038%      5111    24.865%
(5 rows affected)
Total CPU    Total I/O
-----
40006        20555
5 login accounts cleared.
```

示例 2 为用户 “kathy” 开始新的帐户使用周期。

```
sp_clearstats kathy
Name  Since          CPU  Percent CPU    I/O    Percent I/O
----  -
KATHY Jul 24 1990    498    49.8998%      483924  9.1829%
(1 row affected)
Total CPU    Total I/O
-----
998          98392
1 login account cleared.
```

用法

- `sp_clearstats` 创建会计周期，只应在周期结束时运行。
- 因为 `sp_clearstats` 将清除会计统计信息，所以在运行该过程之前必须记录这些统计信息。
- `sp_clearstats` 更新 `syslogins` 字段 `accdate` 并清除 `syslogins` 字段 `totcpu` 和 `totio`。

权限 只有系统管理员才能执行 `sp_clearstats`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_reportstats](#)

sp_client_addr

说明 显示每个具有附加客户端应用程序（包括 `spid` 和客户端主机名）的 Adaptive Server 任务的 IP（Internet 协议）地址。

语法 `sp_client_addr [spid]`

参数 `spid`

指定一个需要其 IP 地址的任务。

示例 1 列出所有任务的 IP 地址：

```
sp_client_addr
-----
spid  hostname  ipaddr
-----
11    FRED      162.66.131.36
21    BARNEY    162.66.100.233
22    WILMA     162.66.100.206
23    BETTY     162.66.100.119
24    PEBBLES  162.66.100.125
25    BAMBAM    162.66.100.124
(6 rows affected)
(return status = 0)
```

示例 2 显示 `spid 21` 的 IP 地址：

```
sp_client_addr 21
-----
spid  hostname  ipaddr
-----
21    BARNEY    162.66.100.233
(1 row affected)
(return status = 0)
```

示例 3 显示当客户端应用程序未通过 IP 连接时的结果：

```
sp_client_addr 11
-----
spid  hostname  ipaddr
-----
11    FRED      0.0.0.0
(1 row affected)
(return status = 0)
```

示例 4 显示没有附加客户端的任务（例如，管家任务）的结果：

```
sp_client_addr 9
-----
spid  hostname  ipaddr
-----
9                NULL
(1 row affected)
(return status = 0)
```

示例 5 显示当指定了不正确的 spid 时的结果：

```
sp_client_addr 99
-----
Msg 18934, Level 16, State 1:
Procedure "sp_client_addr", Line 32:
spid not found
(return status = 1)
```

用法

- 如果客户端应用程序不是通过 IP 附加的，则地址显示为 0.0.0.0。Adaptive Server 不支持显示非 IP 协议的地址。
- 如果任务未附加任何客户端（例如，管家任务），则 IP 地址显示为“NULL”。当使用不带参数的 sp_client_addr 时，将不列出没有附加客户端的任务。

权限

任何用户都可以执行 sp_client_addr。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_who](#)

sp_clusterlockusage

说明 (仅限集群环境) 报告集群中可用、已用和保留的锁。

语法 `sp_clusterlockusage`

示例 报告集群中当前使用的锁:

| Lock Usage | count | % of total |
|----------------|-------|------------|
| ----- | ----- | ----- |
| Total Locks | 95039 | n/a |
| Free Locks | 85807 | 90.29 % |
| Used Locks | 9232 | 9.71 % |
| Object Locks | 4032 | 4.24 % |
| Physical Locks | 233 | 0.25 % |
| Table Locks | 0 | 0.00 % |
| Page Locks | 0 | 0.00 % |
| Row Locks | 17 | 0.02 % |
| Others | 501 | 0.53 % |
| Retention Used | 0 | 0.00 % |

用法

- Retention Used 报告不由任何任务拥有但由于锁保留而在集群级拥有的锁数。

sp_cluster

说明

(仅限集群环境) 执行大量与集群有关的过程。

语法

将连接迁移到另一个逻辑集群或实例:

```
sp_cluster connection, migrate, lc_name, instance_name, "spid_list"
```

确定之前迁移到新实例的连接迁移是否处于挂起状态, 如果是则终止迁移:

```
sp_cluster connection, ['migrate_status' | 'migrate_cancel' ], 'spid_list'
```

修改未完成的操作, 例如取消操作或更改操作的计时:

```
sp_cluster logical, "action", lc_name, {
  cancel, action_handle |
  modify_time, action_handle, wait_option[, timeout ] |
  release, action_handle }
```

将资源或一个或多个路由添加到逻辑集群中:

```
sp_cluster logical, "add", lc_name, {
  route, route_type, key_list |
  instance, instance_list | failover, instance_list }
```

将路由从一个逻辑集群移到另一个逻辑集群:

```
sp_cluster logical, "alter", lc_name, route, route_type, key_list
```

新建逻辑集群:

```
sp_cluster logical, "create", lc_name
```

停止一个或多个实例上的逻辑集群或停止整个逻辑集群, 并将这些实例或该集群置于非活动状态:

```
sp_cluster logical, "deactivate", lc_name, {
  "cluster" | "instance", instance_list }
[, wait_option[, timeout[, @handle output ]]]
```

删除一个逻辑集群或从该逻辑集群中删除一个或多个资源:

```
sp_cluster logical, "drop", lc_name,
{cluster | instance, instance_list |
failover, instance_list | route, route_type, key_list }
```

逆向执行手动故障切换, 恢复原始基本实例:

```
sp_cluster logical, "failback", lc_name, {
  cluster[, wait_option[, timeout[, @handle output ]]] |
  instance, from_instance_list, to_instance_list[, wait_option[,
  timeout[, @handle output ]]] }
```

启动从基本实例到故障切换实例的手动故障切换。


```
sp_cluster logical, "failover", lc_name, {cluster
  [, to_instance_list], wait_option[, timeout], @handle output ]]}
| instance, from_instance_list, to_instance_list[, wait_option[,
  timeout], @handle output ]]} }
```

手动收集一组连接并将其迁移到其它逻辑集群:

```
sp_cluster logical, 'gather', lc_name
```

显示 `sp_cluster logical` 的完整语法:

```
sp_cluster logical, "help"
```

停止一个或多个实例上的逻辑集群或停止整个逻辑集群:

```
sp_cluster logical, "offline", lc_name,
  {cluster | instance, instance_list }
  [, wait_option[, timeout], [, @handle output ]]} }
```

启动一个或多个实例上的缺省逻辑集群:

```
sp_cluster logical, "online", { lc_name[, instance_list] }
```

设置逻辑集群规则: 开放式逻辑集群、故障切换模式、系统视图、启动模式和负载配置文件:

```
sp_cluster logical, "set", lc_name, { open
  | failover, failover_mode | system_view, view_mode
  | startup, { automatic | manual } | load_profile, profile_name }
  login_distribution, { affinity | "round-robin" }
```

显示有关逻辑集群的信息:

```
sp_cluster logical, "show"
  [, lc_name[, {action[, state] | route[, type[, key]}}]]
```

允许设置和管理逻辑集群的负荷配置文件:

```
sp_cluster profile, [ "show" [, profile_name ]
  | "create", profile_name | "drop", profile_name
  | "set", profile_name [, weight [, wt_metric [, wt_value ]
  | threshold [, thr_metric [, thr_value ] ] ] ] }
```

允许设置和管理逻辑集群的负荷配置文件:

```
sp_cluster profile, [ "show" [, profile_name ] | "create", profile_name |
  "drop", profile_name | "set", profile_name [, weight [, wt_metric [,
  wt_value ] | threshold [, thr_metric [, thr_value ] ] ] ] }
```

参数

sp_cluster connection, migrate, *lc_name*, *instance_name*, "*spid_list*"

其中:

- *lc_name* — 逻辑集群的名称。
- *instance_name* — 实例的名称。
- *spid_list* — 要迁移的 spid 列表。如果存在多个 spid，则以分号分隔。

sp_cluster connection, ['migrate_status' | 'migrate_cancel'], '*spid_list*'

其中:

- *spid_list* — 要调查的 spid 列表。
- migrate_cancel — 指示要终止连接迁移。
- migrate_status — 指示要调查连接迁移的状态。

sp_cluster logical, "action", *lc_name*, {cancel, *action_handle* | modify_time, *action_handle*, *wait_option*[, *timeout*]|release, *action_handle*}

其中:

- cancel — 指定要取消的操作。
- *action_handle* — 操作标识符。
- modify_time — 指定要修改的操作时间。
- *wait_option* — 如何修改操作时间。其值包括:
 - wait — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - nowait — 指示立即迁移或断开现有连接。
 - until — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 wait 一起使用时）或特定的时间（与 until 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 “11:30 p.m”（或 “11:30pm”）记录为 “23:30:00”。
- release — 指定操作完成后释放由其占有的所有资源。

```
sp_cluster logical, "add", lc_name, { route, route_type, key_list | instance,  
instance_list | failover, instance_list}
```

其中：

- *lc_name* — 逻辑集群的名称。
- *route* — 指定要将一个或多个路由添加到逻辑集群中
- *route_type* — 要添加的路由的类型。其值包括：
 - *application* — 为应用程序名指定到逻辑集群的路由。
 - *login* — 为登录名指定到逻辑集群的路由。
 - *alias* — 为服务器名别名指定到逻辑集群的路由。
- *key_list* — 应用程序、登录名或别名的列表（具体取决于路由类型）。键列表中的元素以分号分隔。
- *instance* — 指定要将一个或多个基本实例添加到逻辑集群中。
- *instance_list* — 要添加的实例列表。如果存在多个实例，则以分号分隔。
- *failover* — 指定要将一个或多个故障切换实例添加到逻辑集群中。

```
sp_cluster logical, "alter", lc_name, route, route_type, key_list
```

其中：

- *lc_name* — 逻辑集群的名称。
- *route* — 指定要更改的路由。
- *route_type* — 要更改的路由的类型。其值包括：
 - *application* — 为应用程序名指定到逻辑集群的路由。
 - *login* — 为登录名指定到逻辑集群的路由。
 - *alias* — 为服务器名别名指定到逻辑集群的路由。
- *key_list* — 应用程序、登录名或别名的列表（具体取决于路由类型）。键列表中的元素以分号分隔。

```
sp_cluster logical, "create", lc_name
```

其中：

- *lc_name* — 逻辑集群的名称。

```
sp_cluster logical, "deactivate", lc_name, { "cluster" | "instance", instance_list }  
[, wait_option[, timeout[, @handle output ]]]
```

- *lc_name* — 逻辑集群的名称。
- *cluster* — 指定整个集群。
- *instance* — 指定仅将逻辑集群中的某些实例置于非活动状态。
- *instance_list* — 逻辑集群中所选实例的列表。
- *wait_option* — 有效选项包括：
 - *wait* — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - *nowait* — 指示立即迁移或断开现有连接。
 - *until* — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 *wait* 一起使用时）或特定的时间（与 *until* 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 11:30 p.m. 记录为 23:30:00。
- *@handle output* — 指定要检索操作的操作句柄。

`sp_cluster logical, "drop", lc_name, { cluster | instance, instance_list | failover, instance_list | route, route_type, key_list }`

其中：

- *lc_name* — 逻辑集群的名称。
- *cluster* — 指定整个集群。
- *instance* — 指定仅将逻辑集群中的某些实例置于非活动状态。
- *instance_list* — 逻辑集群中所选实例的列表。
- *wait_option* — 有效选项包括：
 - *wait* — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - *nowait* — 指示立即迁移或断开现有连接。
 - *until* — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 *wait* 一起使用时）或特定的时间（与 *until* 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 11:30 p.m. 记录为 23:30:00。
- `@handle output` — 指定要检索操作的操作句柄。

sp_cluster logical, "failback", *lc_name*, {cluster[, *wait_option*[, *timeout*[, @handle output]]] | instance, *from_instance_list*, *to_instance_list*[, *wait_option*[, *timeout*[, @handle output]]] }

其中:

- *lc_name* — 逻辑集群的名称。
- cluster — 指定整个集群。
- *to_instance_list* — 预定义故障切换实例的列表。值 NULL 会激活第一个故障切换组。
- *from_instance_list* — 要使其脱机的实例的列表。
- *wait_option* — 有效选项包括:
 - wait — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - nowait — 指示立即迁移或断开现有连接。
 - until — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 wait 一起使用时）或特定的时间（与 until 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 11:30 p.m. 记录为 23:30:00。
- @handle output — 指定要检索操作的操作句柄。

```
sp_cluster logical, "failover", lc_name, {cluster[, to_instance_list[, wait_option[,
timeout[, @handle output ]]] | instance, from_instance_list, to_instance_list[,
wait_option[,timeout[, @handle output ]]] }
```

其中：

- *lc_name* — 逻辑集群的名称。
- *cluster* — 指定对整个逻辑集群进行故障切换。
- *to_instance_list* — 预定义故障切换实例的列表。值 NULL 会激活第一个故障切换组。
- *wait_option* — 如何记录操作时间。其值包括：
 - *wait* — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - *nowait* — 指示立即迁移或断开现有连接。
 - *until* — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 *wait* 一起使用时）或特定的时间（与 *until* 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 11:30 p.m. 记录为 23:30:00。
- *@handle output* — 指定要检索故障切换的操作句柄。
- *instance* — 指定仅对逻辑集群中的选定实例进行故障切换。
- *from_instance_list* — 要使其脱机的实例的列表。

```
sp_cluster logical, 'gather', lc_name
```

其中：

- *gather* — 指示要收集一组限定连接以将其一起迁移到另一个逻辑集群。
- *lc_name* — 要将连接迁移到的逻辑集群的名称。

```
sp_cluster logical, "offline", lc_name, { cluster | instance, instance_list }  
[, wait_option[, timeout[, @handle output ]]]
```

其中:

- *lc_name* — 逻辑集群的名称。
- *cluster* — 指定整个集群。
- *instance* — 指定仅使逻辑集群中的选定实例脱机。
- *instance_list* — 逻辑集群中所选实例的列表。
- *wait_option* — 如何指定操作时间。其值包括:
 - *wait* — 指示提供指定的时间量（如果未提供 *timeout*，则不限时间）来迁移或断开现有连接。
 - *nowait* — 指示立即迁移或断开现有连接。
 - *until* — 指示提供某一天的特定时间，以在该时间之前迁移或断开现有连接。
- *timeout* — 特定的时间量（与 *wait* 一起使用时）或特定的时间（与 *until* 一起使用时）。格式为 “hh:mm:ss”（按照 24 小时制）。例如，*timeout* 会将 11:30 p.m. 记录为 23:30:00。
- *@handle output* — 指定要检索操作的操作句柄。
- *from_instance_list* — 要使其脱机的实例的列表。

```
sp_cluster logical, "online", { lc_name[, instance_list]}
```

其中:

- *lc_name* — 逻辑集群的名称。
- *instance_list* — 逻辑集群中所选实例的列表。


```
sp_cluster logical, "set", lc_name, {open | failover, failover_mode | system_view,
view_mode | startup, { automatic | manual } | load_profile, profile_name |
action_release, { automatic | manual } | gather, { automatic | manual } |
login_distribution, { affinity | "round-robin" }
```

其中：

- *lc_name* — 逻辑集群的名称。
- *open* — 设置打开的逻辑集群。未路由的连接将发送至打开的逻辑集群。
- *failover failover_mode* — 设置逻辑集群的故障切换模式。
failover_mode 的值包括：
 - *instance* — 指定 1:1 故障切换策略；每当基本实例失败时，都使一项故障切换资源联机。
 - *group* — 指定仅在集群中的所有基本实例失败时才使故障切换资源联机。
- *system_view view_mode* — 为逻辑集群中运行的任务设置缺省系统视图。*view_mode* 的值包括：
 - *instance* — 指定监控和信息性工具，例如 *sp_who*、*sp_lock* 以及描述实例的监控表。
 - *cluster* — 指定监控和信息性工具，例如 *sp_who*、*sp_lock* 以及描述整个集群的监控表。
- *startup { automatic | manual}* — 设置逻辑集群的启动模式。
 - *automatic* — 指定集群启动时自动启动逻辑集群。
 - *manual* — 指定必须手动启动逻辑集群。
- *login_distribution* — 指定 Cluster Edition 在逻辑集群跨多个实例时如何分配连接。
- *action_release* — 允许或禁止这些逻辑集群操作（*online*、*offline*、*failover* 和 *failback*）在完成或取消之后自动释放和清除。
 - *automatic* — 指定自动清除逻辑集群操作。
 - *manual* — 指定在逻辑集群操作完成或取消后不对其进行清除。这是缺省值。
- *gather* — 允许或禁止在以下预定义操作之一发生时将连接组移动到另外的逻辑集群：*online*、*add route*、*alter route* 或 *drop route*。
 - *automatic* — 指定自动移动连接。

- **manual** — 指定不自动移动连接。这是缺省值。
- **@handle output** — 指定要检索操作的操作句柄。
- **from_instance_list** — 要使其脱机的实例的列表。

`sp_cluster logical, "show"[, lc_name[, {action[, state] | route[, type[, key]]}]`

其中：

- ***lc_name*** — 逻辑集群的名称。如果输入 NULL，则显示所有逻辑集群的摘要信息。
- ***action*** — 指定有关管理操作的信息：**failover**、**failback**、**online**、**offline**、**deactivate**。
- ***state*** — 以下值之一：**cancelled**、**complete** 或 **active**。
- ***route*** — 指定有关路由的信息。
- ***type*** — 以下值之一：**application**、**alias** 或 **login**。
- ***key*** — 特定的登录名、别名或应用程序名。

```
sp_cluster profile, [ "show" [, profile_name ] | "create", profile_name | "drop",
profile_name | "set", profile_name [, weight [, wt_metric [, wt_value ] | threshold [,
thr_metric [, thr_value ]]]
```

- `show` — 显示配置的负载配置文件及其设置。
- `profile_name` — 负载配置文件的名称。
- `creates` — 创建新的负载配置文件。
- `drop` — 删除负载配置文件。
- `set` — 指定负载配置文件的属性。必须单独设置各个属性。
- `weight` — 指定权值属性。
- `wt_metric` — 指定单个权值指标。其值包括：
 - `user connections` — 实例基于资源可用性接受新连接的能力。
 - `cpu utilization` — 一个实例接受新连接的容量（基于资源可用性）。
 - `run queue` — 一个实例接受新连接的容量（基于资源可用性）。
 - `io load` — 未完成的异步 I/O。
 - `engine deficit` — 集群内各实例之间的联机引擎数差异。

注释 只有集群内的各个实例具有不同数量的引擎时，才能测量 `engine deficit`。`engine deficit` 会在负载分值中添加一个衡量最大相对容量的指标。

- `user metric` — 用户提供的可选指标。
- `wt_value` — 指定权值。有效值为 0 到 255。如果权值为零 (0)，则在计算中排除指标。
- `threshold` — 指定阈值属性。
- `thr_metric` — 指定特定的阈值属性。其值包括：
 - `dynamic` — 指定动态负载分配的阈值。
 - `login` — 指定登录重定向的阈值。
 - `hysteresis` — 为任意连接重定向指定最小负载分值。

- *thr_value* — 取决于 *thr_metric* 的值。当 *thr_metric* 为：
 - *dynamic* 或 *login* — *thr_value* 是两个实例负载分值的百分比差。有效值为 0 到 100。如果权值为零 (0)，则禁用这种负载分配。
 - *hysteresis* — *thr_value* 是实现动态负载分配或登录重定向而要求目标实例必须达到的最小负载分值。

示例

示例 1 将 spid 为 73 的连接移到 SalesLC 集群：

```
sp_cluster connection, migrate, SalesLC, NULL, '73'
```

示例 2 将当前连接移到 “ase3” 实例：

```
sp_cluster connection, migrate, NULL, ase3
```

示例 3 将 spid 值为 73 和 75 的连接移入 “ase3” 实例和 SalesLC 集群：

```
sp_cluster connection, migrate, SalesLC, ase3, '73;75'
```

示例 4 确定是否正在对 spid 73 进行连接迁移；如果是，Cluster Edition 将取消迁移：

```
sp_cluster connection, 'migrate_cancel', '73'
```

示例 5 对 spid 值为 73 的连接检查已迁移连接的状态：

```
sp_cluster connection, 'migrate_status', '73'
```

```
SPID LogicalCluster Instance MigrationLogicalCluster MigrationInstance Command
-----
73 SystemLC          ase1      SalesLC          ase3      connection migrate
```

示例 6 取消对 “SalesLC” 逻辑集群的定时操作。操作句柄为 4390。

```
sp_cluster logical, "action", SalesLC, cancel, "4390"
```

示例 7 将现有操作 5364 的 wait 选项更改为 nowait。

```
sp_cluster logical, "action", SalesLC, modify_time,
"5364", nowait
```

示例 8 释放针对 “SalesLC” 逻辑集群的操作 3456。

```
sp_cluster logical, "action", SalesLC, release, "3456"
```

示例 9 释放针对 “SalesLC” 逻辑集群的所有已完成或已取消的操作。

```
sp_cluster logical, "action", SalesLC, release, "all"
```

示例 10 将实例 “ase1” 和 “ase2” 添加到 “SalesLC” 逻辑集群。

```
sp_cluster logical, "add", SalesLC, instance,
"ase1;ase2"
```

示例 11 为 “SalesLC” 创建一个包含 “ase3” 的故障切换组。

```
sp_cluster logical, "add", SalesLC, failover, ase3
```

示例 12 将登录名 “tom”、“dick” 和 “harry” 添加到 “SalesLC” 逻辑集群

```
sp_cluster logical, "add", SalesLC, route, login,
"tom;dick;harry"
```

示例 13 将 field_sales 应用程序路由到 “SalesLC” 逻辑集群。

```
sp_cluster logical, "add", SalesLC, route, application,
field_sales
```

示例 14 创建一个到逻辑集群 “lc1”（别名为 “SalesLC”）的别名类型的路由。随后将路由的逻辑集群关联从 “lc1” 更改为 “lc2”。路由用路由类型（别名）及其键 (SalesLC) 标识。

```
sp_cluster logical, "add", "lc1", "route", "alias", "SalesLC"
sp_cluster logical, "alter", "lc2", "route", "alias", "SalesLC"
```

示例 15 创建名为 “SalesLC” 的逻辑集群：

```
sp_cluster logical, "create", SalesLC
```

示例 16 立即停止 “SalesLC” 逻辑集群中的所有实例，并将 “SalesLC” 置于非活动状态：

```
sp_cluster logical, "deactivate", SalesLC, cluster, nowait
```

示例 17 停止 “ase1” 和 “ase2” 实例，并将 “SalesLC” 置于非活动状态：

```
sp_cluster logical, "deactivate", SalesLC, instance, "ase1;ase2"
```

示例 18 删除 “SalesLC” 逻辑集群：

```
sp_cluster logical, "drop", SalesLC, cluster
```

示例 19 从 “SalesLC” 逻辑集群中删除基本实例 “ase1” 和 “ase2”。

```
sp_cluster logical, "drop", SalesLC, instance, "ase1;ase2"
```

示例 20 删除从应用程序 field_sales 和 web_sales 到 “SalesLC” 逻辑集群的路由。

```
sp_cluster logical "drop", SalesLC, route, application,
"field_sales;web_sales"
```

示例 21 故障恢复 “SalesLC” 逻辑集群：

```
sp_cluster logical, "failback", SalesLC, cluster
```

示例 22 “SalesLC”将在“ase3”和“ase1”上运行。在本示例中，“ase3”故障恢复到“ase1”，而“SalesLC”继续在“ase2”上运行。此操作在两分钟后执行：

```
declare @out_handle varchar(15)

execute
sp_cluster logical, "failback", SalesLC, instance,
ase3, ase1, wait, "00:02:00", @handle = @out_handle
output
```

示例 23 将“SalesLC”逻辑集群故障切换到第一个预定义故障切换资源组。故障切换在等待 2 分钟后终止连接。

```
declare @out_handle varchar(15)

execute
sp_cluster logical, "failover", SalesLC, cluster, NULL, wait, "00:02:00",
@handle = @out_handle output

Action '2' has been issued for the 'failover cluster' command.

Logical Cluster Handle      Action          From To      State      InstancesWaiting
ConnectionsRemaining WaitType      StartTime      Deadline      CompleteTime
-----
SalesLC                      2 failover cluster      2, 4      NULL
complete                    0                                0 wait
Nov 15 2007 3:23PM      Nov 15 2007 3:25PM      Nov 15 2007 3:23PM
```

Remember to issue the 'sp_cluster logical, action, <logical cluster name>, release, <handle>' command for any cancelled or completed actions.

示例 24 “SalesLC”将在“ase1”和“ase2”上运行。在本示例中，“ase1”故障恢复到“ase3”，而“SalesLC”继续在“ase2”上运行。不指定 wait 选项，以采用缺省设置（即不限等待时间）。

```
sp_cluster logical, "failover", SalesLC, instance,
ase1, ase3

Action '1' has been issued for the 'failover instance' command.

Logical Cluster Handle      Action          From To      State      InstancesWaiting
ConnectionsRemaining WaitType      StartTime      Deadline      CompleteTime
-----
SalesLC                      1 failover instance      1 4      complete  0
0 infinite      Nov 15 2007 3:06PM      NULL      Nov 15 2007 3:06PM
```

Remember to issue the `sp_cluster logical, action, <logical cluster name>, release, <handle>` command for any cancelled or completed actions.

示例 25 收集一组连接并将其迁移到 “new_stores” 逻辑集群:

```
sp_cluster logical, 'gather', new_stores
```

示例 26 显示 sp_cluster logical 存储过程的语法。

```
sp_cluster logical, "help"
```

```
Usage for sp_cluster 'logical':
sp_cluster 'logical', 'help' [, <module>]
```

To show the logical cluster configuration:

```
sp_cluster 'logical', 'show'
sp_cluster 'logical', 'show', <lcname>
sp_cluster 'logical', 'show', <lcname> | NULL, 'action' [, <state>]
sp_cluster 'logical', 'show', <lcname> | NULL, 'route' [, <type [, <key>]]
```

To create a logical cluster:

```
sp_cluster 'logical', 'create', <lcname>
```

To add resources to a logical cluster:

```
sp_cluster 'logical', 'add', <lcname>, 'failover', <instance_list> [, <group>]
sp_cluster 'logical', 'add', <lcname>, 'instance', <instance_list>
sp_cluster 'logical', 'add', <lcname>, 'route', <route_type>, <key_list>
```

To drop resources from a logical cluster:

```
sp_cluster 'logical', 'drop', <lcname>, 'cluster'
sp_cluster 'logical', 'drop', <lcname>, 'failover', <instance_list>
sp_cluster 'logical', 'drop', <lcname>, 'instance', <instance_list>
sp_cluster 'logical', 'drop', <lcname>, 'route', <route_type>, <key_list>
```

Argument details:

```
<lcname> is a logical cluster nam
<instance_list> is a ';' separated list of instance
<route_type> is one of {'user', 'application', 'alias'}
<key_list> is a ';' separated list of keys
```

To set attributes of a logical cluster:

```
sp_cluster 'logical', 'set', <lcname>, 'open'
sp_cluster 'logical', 'set', <lcname>, 'down_routing', 'disconnect' | 'system' |
'open'
sp_cluster 'logical', 'set', <lcname>, 'failover', 'instance' | 'group'
sp_cluster 'logical', 'set', <lcname>, 'load_profile', <profile_name>
sp_cluster 'logical', 'set', <lcname>, 'startup', 'automatic' | 'manual'
sp_cluster 'logical', 'set', <lcname>, 'system_view', 'instance' | 'cluster'
```

To start and stop a logical cluster:

```
sp_cluster 'logical', 'online', <lcname>[, <instance_list>]
sp_cluster 'logical', 'offline', <lcname>, 'cluster'[, <wait_option>[, <time>[,
```

```
@handle output]]]
sp_cluster 'logical', 'offline', <lname>, 'instance',
  <instance_list>[, <wait_option>[, <time>[, @handle output]]]
```

To failover and failback a logical cluster:

```
sp_cluster 'logical', 'failover', <lname>, 'cluster'[, <instance_list>[,
  <wait_option>[, <time>[, @handle output]]]]
sp_cluster 'logical', 'failover', <lname>, 'instance', <from_instance_list>,
  <instance_list>[, <wait_option>[, <time>[, @handle output]]]]
sp_cluster 'logical', 'failback', <lname>, 'cluster'[, <instance_list>[,
  <wait_option>[, <time>[, @handle output]]]]
sp_cluster 'logical', 'failback', <lname>, 'instance', <from_instance_list>,
  <instance_list>[, <wait_option>[, <time>[, @handle output]]]
```

To work with action handles:

```
sp_cluster 'logical', 'action', <lname>, 'cancel', <handle>
sp_cluster 'logical', 'action', <lname>, 'modify_time', <handle>, <wait_option>[,
  <time>]
sp_cluster 'logical', 'action', <lname>, 'release', <handle>
```

Argument details:

```
<wait_option> is one of {'nowait', 'wait', 'until'}
<time> is a time in hh:mm:ss format
<handle> is an action handle
```

示例 27 立即停止 “SalesLC” 中的所有实例，并将 “SalesLC” 置于脱机状态。

```
sp_cluster logical, "offline", SalesLC, cluster, nowait
```

示例 28 停止 “SalesLC” 中的 “ase1” 和 “ase2” 实例，并将 “SalesLC” 置于脱机状态。

```
sp_cluster logical, "offline", SalesLC, instance, "ase1;ase2"
```

示例 29 启动 “SalesLC” 逻辑集群中的所有基本实例，并使该集群联机。

```
sp_cluster logical, "online", SalesLC
```

示例 30 启动 “SalesLC” 中的 “ase1” 实例，并使该集群联机。

```
sp_cluster logical, "online", SalesLC, ase1
```

示例 31 将 “SalesLC” 逻辑集群的负载配置文件设置为 Sybase 配置文件 sybase_profile_oltp:

```
sp_cluster logical, "set", SalesLC, load_profile,
  sybase_profile_oltp
```

示例 32 将缺省系统视图设置为 cluster:

```
sp_cluster logical, "set", SalesLC, system_view, cluster
```


示例 33 显示已配置的所有逻辑集群的摘要信息。

```
sp_cluster logical, "show", NULL
```

| ID | Name | State | Online Instances | Connections |
|----|------------|---------|------------------|-------------|
| 1 | mycluster | online | 4 | 1 |
| 2 | SalesLC | online | 2 | 0 |
| 3 | HRLC | online | 1 | 0 |
| 4 | CatchallLC | offline | 0 | 0 |

Logical cluster 'mycluster' is the system logical cluster.
 Logical cluster 'CatchallLC' is the open logical cluster.

| Logical Cluster | Instance | State | Type | Connections | Load Score |
|-----------------|----------|---------|----------|-------------|------------|
| HRLC | silk | online | base | 0 | 0.01 |
| SalesLC | cotton | offline | failover | 0 | 0.00 |
| SalesLC | linen | online | base | 0 | 0.00 |
| SalesLC | silk | offline | failover | 0 | 0.01 |
| SalesLC | wool | online | base | 0 | 0.01 |
| mycluster | cotton | online | base | 0 | 0.00 |
| mycluster | linen | online | base | 0 | 0.00 |
| mycluster | silk | online | base | 0 | 0.01 |
| mycluster | wool | online | base | 1 | 0.01 |

示例 34 显示所有未完成操作的列表。

```
sp_cluster logical, "show", NULL, action
```

示例 35 显示有关 SalesLC 逻辑集群的信息。

```
sp_cluster logical, "show", SalesLC
```

| ID | Name | State | Online Instances | Connections |
|----|---------|--------|------------------|-------------|
| 2 | OrderLC | online | 1 | 0 |

| Instance | State | Type | Connections | Load Score | Failover Gro |
|----------|--------|------|-------------|------------|--------------|
| asedem01 | online | base | 0 | 0.78 | NU |

| Attribute | Setting |
|-------------------|---------------------------|
| Down Routing Mode | system |
| Failover Mode | instance with fail_to_any |
| LC Roles | none |

```
Load Profile                sybase_profile_oltp
Login Distribution         affinity
Startup Mode              automatic
System View               cluster
```

```
Route Type                Route Key
-----
application              order_app
```

```
Logical cluster 'OrderLC' has no associated actions.
(return status = 0)
```

示例 36 创建负载配置文件 “my_profile”:

```
sp_cluster profile, "create", my_profile
```

示例 37 指定 “my_profile” 的指标权值。“user connections” 设置为零，这会将该指标从配置文件中排除:

```
sp_cluster profile, "set", my_profile, weight, "user connections", '0'
sp_cluster profile, "set", my_profile, weight, cpu utilization, '20'
sp_cluster profile, "set", my_profile, weight, runqueue, '30'
sp_cluster profile, "set", my_profile, weight, io load, '10'
sp_cluster profile, "set", my_profile, weight, engine deficit, '10'
sp_cluster profile, "set", my_profile, weight, user metric, '30'
```

示例 38 对于 “my_profile”，将登录重定向阈值设置为 80 并将停滞值设置为 10:

```
sp_cluster profile, "set", my_profile, threshold, login, '80'
sp_cluster profile, "set", my_profile, threshold, hysteresis, '10'
```

示例 39 显示有关已配置的配置文件的的信息:

```
sp_cluster profile, "show", my_profile
```

| ID | Profile | Type | Connections | CPU | Run | Queue |
|-----|------------|------|-------------|-----|-----|-------|
| 100 | my_profile | user | 0 | 20 | 30 | 10 |

| Profile | Logical Cluster |
|------------|-----------------|
| my_profile | SalesLC |

| Profile | Logical Cluster Instance |
|---------------|--------------------------|
| Load Score | Connections Score |
| CPU Score | Run Queue Score |
| IO Load Score | User Score |

```

-----
-----
-----
-----
-----
my_profile          SalesLC          ase1
                   0.028871          0.000000
                   0.028871          0.000000
                   0.000000          0.000000
                   0.000000
my_profile          ase2
                   0.029474          0.000000
                   0.029474          0.000000
                   0.000000          0.000000
                   0.000000
my_profile          ase3
                   0.019503          0.000000
                   0.019503          0.000000
                   0.000000          0.000000
                   0.000000
my_profile          ase4
                   0.582675          0.000000
                   0.290930          0.291745
                   0.000000          0.000000
                   0.000000

```

用法

sp_cluster connection

若要迁移当前 `spid`，请在 `sp_cluster connection, migrate` 中省略 `spid_list`。

sp_cluster logical, action

- 通过查询 `monLogicalClusterAction` 表或执行以下命令检索操作句柄：


```
sp_cluster logical, "show", NULL, action
```
- 任何不支持迁移的客户端将在完成 SQL 批处理且没有打开的事务时或在 `timeout` 期限到期时（以较早者为准）断开连接。
- 在 `timeout` 期限结束时，所有剩余客户端都将断开连接。
- 取消操作时不会回滚操作。可能需要执行附加任务才能将配置恢复到初始状态。
- 只有完成的操作才能获得释放。释放操作时会将该完成的操作从系统和 `monLogicalClusterAction` 表中删除。

sp_cluster logical, 'add'

- 不能向系统逻辑集群中添加基本实例或故障切换资源。
- 如果存在多个实例、故障切换资源或应用程序，则以分号分隔。

- 通过在组两侧添加小括号并用逗号分隔各组可创建多个故障切换组。如果不指定组，则创建一个新组并将实例添加到该组中。您可以指定要将实例放入的组（组号必须用引号括起来）。

例如：

```
1> sp_cluster logical, 'add', tempLC, failover, "asedemo3;asedemo2"
2> go
```

```
Added failover instance 'asedemo3' to group 1 for logical cluster 'tempLC'.
Added failover instance 'asedemo2' to group 1 for logical cluster 'tempLC'.
```

随后将实例添加到该组中：

```
1> sp_cluster logical, 'add', tempLC, failover, asedemo4, "4"
2> go
```

```
Added failover instance 'asedemo4' to group 4 for logical cluster 'tempLC'.
```

sp_cluster logical, "deactivate"

- **deactivate** 命令不能用于系统逻辑集群。
- **offline** 与 **deactivate** 相同，只是 **deactivate** 会将停止的实例或集群置于非活动状态，而 **offline** 会将它们置于脱机状态。

sp_cluster logical "drop"

- 删除实例或故障切换资源之前，必须将其置于脱机状态。
- 删除集群时还会删除与该集群关联的所有路由、资源和设置。

sp_cluster logical "failback"

若要启动故障恢复，必须先对逻辑集群进行故障切换。

sp_cluster logical "gather"

- 逻辑集群必须处于联机状态，才能手动收集连接
- 必须为逻辑集群定义路由才能收集连接

sp_cluster logical, "offline"

- **offline** 命令不能用于系统逻辑集群。
- **offline** 与 **deactivate** 相同，只是 **deactivate** 会将停止的实例或集群置于非活动状态。

sp_cluster logical "online"

online 命令不能用于系统逻辑集群。

sp_cluster logical "set"

只能有一个逻辑集群具有 open 属性。为新的逻辑集群设置 open 属性时，将从前一个打开的逻辑集群删除该 open 属性。

sp_cluster profile

- 必须对用户指标值进行规范化，才能使其与 Sybase 提供的指标值兼容。可考虑使用测量响应时间的用户指标。如果可接受的最长响应时间是 10 秒，而测得的值是 5，则指标值为 50 ($5/10 \times 100 = 50$)。
- 通过阈值指标可以配置负载不平衡导致连接从一个实例重定向到另一个实例的时点。当目标实例与负载最小的实例之间的负载分值差（以百分比表示）达到或超过该阈值时，工作负荷管理器即会重定向连接。

当负载分值差达到该阈值，但因实例的负载分值（例如 2 和 8）较低而不适于重定向时，停滞值可防止执行重定向。


```

Different query plans that have the same association key
count
-----
                1
Query plans present only in group 'dev_plans':
count
-----
                1
Query plans present only in group 'prod_plans':
count
-----
                0
    
```

示例 2 使用 `brief` 模式生成报告:

```
sp_cmp_all_qplans dev_plans, prod_plans, brief
```

用法

- 使用 `sp_cmp_all_qplans` 检查两个计划组中抽象计划的差异。
- `sp_cmp_all_qplans` 匹配每个组中具有相同用户 ID 和查询文本的计划对。计划按如下原则归类：
 - 相同的计划
 - 在两个组中具有相同关联键但具有不同抽象计划的计划。关联键是组 ID、用户 ID 和查询文本。
 - 只存在于一个组中的计划
- 若要比较两个单独的抽象计划，请使用 `sp_cmp_qplans`。若要查看抽象计划组的名称，请使用 `sp_help_qpgroup`。
- 当系统管理员或数据库所有者运行 `sp_cmp_all_qplans` 时，它将报告两个组中的所有计划。当另一个用户执行 `sp_cmp_all_qplans` 时，它只报告具有该用户的 ID 的计划。

权限

任何用户都可以执行 `sp_cmp_all_qplans`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_cmp_qplans](#)、[sp_help_qpgroup](#)

sp_cmp_qplans

说明 比较两个抽象计划。

语法 `sp_cmp_qplans id1, id2`

参数 `id1, id2`
是两个抽象计划的 ID。

示例 **示例 1** 将抽象计划 411252620 和 1383780087 进行比较：

```
sp_cmp_qplans 411252620, 1383780087
```

```
The queries are the same.
```

```
The query plans are the same.
```

示例 2 将抽象计划 2091258605 和 647777465 进行比较：

```
sp_cmp_qplans 2091258605, 647777465
```

```
The queries are the same.
```

```
The query plans are different.
```

用法

- `sp_cmp_qplans` 比较查询、抽象计划和两个抽象计划的散列键，并报告查询是否相同以及计划是否相同。对于查询，它输出以下消息之一：

- The queries are the same.
- The queries are different.
- The queries are different but have the same hash key.

它将输出抽象计划的以下消息之一：

- The query plans are the same.
 - The query plans are different.
- `sp_cmp_qplans` 也将输出返回状态，显示比较的结果。状态值 1、2 和 10 是附加值。表 1-12 中显示了这些状态值：

表 1-12: `sp_cmp_qplans` 的返回状态值

| 返回值 | 含义 |
|-----|-----------------|
| 0 | 查询文本和抽象计划相同。 |
| +1 | 查询和散列键不同。 |
| +2 | 查询不同但散列键相同。 |
| +10 | 抽象计划不同。 |
| 100 | 一个或两个计划 ID 不存在。 |

- 若要查找计划的 ID，请使用 `sp_help_qpgroup` 或 `sp_find_qplan`。计划 ID 也可以由 `create plan` 返回并包含在 `showplan` 输出中。

权限

任何用户都可以执行 `sp_cmp_qplans` 来比较他 / 她拥有的计划。只有系统管理员或数据库所有者才能比较其他用户拥有的计划。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 `sp_cmp_all_qplans`、`sp_help_qpgroup`

sp_commonkey

| | |
|----|--|
| 说明 | 定义两个表或视图之间的公用键（经常被连接的列）。 |
| 语法 | <pre>sp_commonkey <i>tabaname</i>, <i>tabbname</i>, <i>col1a</i>, <i>col1b</i> [, <i>col2a</i>, <i>col2b</i>, ..., <i>col8a</i>, <i>col8b</i>]</pre> |
| 参数 | <p><i>tabaname</i> 是要连接的第一个表或视图的名称。</p> <p><i>tabbname</i> 是要连接的第二个表或视图的名称。</p> <p><i>col1a</i> 是在表或视图 <i>tabaname</i> 中构成公用键的第一个列的名称。至少应该指定一个列对（第一个表或视图中的一个列和第二个表或视图中的一个列）。</p> <p><i>col1b</i> 是表或视图 <i>tabbname</i> 中与表或视图 <i>tabaname</i> 中的 <i>col1a</i> 连接的对列的名称。</p> |
| 示例 | <p>示例 1 定义 <code>titles.titleid</code> 和 <code>titleauthor.titleid</code> 的公用键：</p> <pre>sp_commonkey titles, titleauthor, title_id, title_id</pre> <p>示例 2 假定有两个表 <code>projects</code> 和 <code>departments</code>，每个表都包含名为 <code>empid</code> 的列。此语句在这两个列上定义常用连接：</p> <pre>sp_commonkey projects, departments, empid, empid</pre> |
| 用法 | <ul style="list-style-type: none">• 创建公用键的目的是为了使数据库设计中的隐式逻辑关系变为显式关系。此信息可供应用程序使用。<code>sp_commonkey</code> 不强制参照完整性约束；可以使用 <code>create table</code> 或 <code>alter table</code> 命令的 <code>primary key</code> 和 <code>foreign key</code> 子句强制键关系。• 执行 <code>sp_commonkey</code> 将键添加到 <code>syskeys</code> 系统表中。若要显示已定义的公用键的报告，请使用 <code>sp_helpkey</code>。• 若要定义两个表或视图之间的公用键，您必须是其中至少一个的所有者。• 第一个表或视图中的列数必须与第二个表或视图中的列数相同。在每个表或视图中，最多可以有 8 列参与公用键。公用列的数据类型也必须一致。对于可指定长度的列，长度可以不同。公用列的空值类型无需一致。• 安装进程在系统表的适当列上运行 <code>sp_commonkey</code>。• 不能将 Java 数据类型用于 <code>sp_commonkey</code>。 |

权限 只有 *tabaname* 或 *tabbname* 的所有者可以执行 `sp_commonkey`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `alter table`、`create table`、`create trigger`

系统过程 `sp_dropkey`、`sp_foreignkey`、`sp_helpjoins`、`sp_helpkey`、`sp_primarykey`

sp_companion

说明 执行集群操作，例如在高可用性系统中将 Adaptive Server 配置为辅助协同服务器，以及将协同服务器从一种故障切换模式转换为另一种故障切换模式。sp_companion 在辅助协同服务器上运行。

语法

```
sp_companion
    [server_name
    {, configure
      [, {with_proxydb | NULL}]
      [, srvlogin]
      [, server_password]
      [, cluster_login]
      [, cluspassword]}
    | drop
    | suspend
    | resume
    | prepare_failback
    | do_advisory}
    {, all
    | help
    | group_attribute_name
    | base_attribute_name}
```

参数

server_name

是执行集群操作的 Adaptive Server 的名称。

configure

将 **server_name** 指定的服务器配置为故障切换配置中的主协同服务器。

drop

将协同服务器从故障切换配置中永久删除。完成该命令后，服务器处于单服务器模式。

suspend

将协同服务器从故障切换配置中临时删除。完成该命令后，协同服务器处于挂起模式。

resume

撤消 **suspend** 命令并恢复协同服务器之间的正常协同模式。

prepare_failback

准备辅助协同服务器以释放主协同服务器的资源，以便进行故障恢复。

do_advisory

为在故障切换模式中成功执行主协同服务器的功能，验证辅助协同服务器是否兼容。

- **all** — 使 **do_advisory** 检查所有参数。
- **help** — 显示有关 **do_advisory** 参数的信息和语法。
- **group_attribute_name** — **sp_companion** 所报告的组属性的名称。
- **base_attribute_name** — 希望 **sp_companion do_advisory** 报告的基本属性的名称。

with_proxydb

如果协同服务器的初始配置中包含此参数，则将在辅助协同服务器上为除系统数据库外的所有数据库（以及随后添加的所有数据库）创建代理数据库。缺省情况下，禁用 **with_proxydb**。

srvlogin

是访问协同服务器的用户登录名。缺省情况下，**srvlogin** 的值为“sa”。

srvpassword

是访问协同服务器的用户口令。缺省情况下，**srvpassword** 的值为空。

cluster_login

是用户登录到集群时使用的登录名。缺省情况下，**cluster_login** 的值是“sa”。

cluspassword

是登录到集群中时必须提供的用户口令。缺省情况下，**cluspassword** 的值为空。

示例

示例 1 将 Adaptive Server MONEY1 配置为主协同服务器：

```
sp_companion "MONEY1", configure
```

示例 2 将 Adaptive Server MONEY1 配置为主协同服务器并在辅助协同服务器上创建代理数据库：

```
sp_companion "MONEY1", configure, with_proxydb, "sa", "sapsswd"
```

示例 3 从故障切换配置中删除 Adaptive Server PERSONEL1。完成该命令后，主协同服务器和辅助协同服务器都处于单服务器模式：

```
sp_companion "PERSONEL1", "drop"
```

示例 4 恢复协同服务器的正常协同模式（在本例中为 MONEY1）：

```
sp_companion "MONEY1", "resume"
```

示例 5 准备主协同服务器（在本例中为 PERSONEL1）以转换为正常协同模式并恢复对已执行故障切换的 Adaptive Server 的控制：

```
sp_companion "PERSONEL1", "prepare_failback"
```

示例 6 检查以确保对 PERSONEL1 协同服务器的集群操作将成功执行。因为本例中的 do_advisory 使用了 all 参数，所以它将检查 PERSONEL1 的所有 do_advisory 属性以确保所有这些属性都不会阻碍集群操作的成功执行，并确保辅助协同服务器在完成故障切换后能够成功执行主协同服务器的操作：

```
sp_companion "PERSONEL1", do_advisory, "all"
```

示例 7 检查以确保协同服务器上的组件集成服务 (CIS) 的所有属性都与本地服务器不兼容：

```
sp_companion "PERSONEL1", do_advisory, "CIS"
```

用法

- sp_companion 执行集群操作，例如在高可用性系统中将 Adaptive Server 配置为辅助协同服务器。sp_companion 还将协同服务器从一种故障切换模式转换到另一种模式（例如，从故障切换模式回到正常协同模式）。sp_companion 在辅助协同服务器中运行。
- sp_companion 用 installhasvss（在 Windows NT 上用 insthasv）而不是 installmaster 脚本来安装。installhasvss 位于 \$SYBASE_ASE 的 scripts 子目录中。
- sp_companion 自动禁用 Sybase 的镜像。Sybase 建议使用第三方镜像软件来保护数据不受磁盘故障的损坏。

有关详细信息，请参见《在高可用性系统中使用 Sybase 故障切换》。在运行 do_advisory 命令之前，请确保阅读本书中有关配置的章节和“do_advisory”一章。

权限

只有具有 ha_role 的用户才能发出 sp_companion。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

sp_compatmode

| | |
|------|---|
| 说明 | 检验是否可以使用完全兼容模式。 |
| 语法 | sp_compatmode |
| 示例 | <pre>1> sp_compatmode ----- Compatibility mode is enabled. WARNING: Compatibility mode may not be used when statement cache and literalautoparam are enabled. WARNING: The configuration option 'histogram tuning factor' is configured with value '20', which is not the default value in ASE 12.5. This may lead to different accuracy of statistics and different query plans. (return status = 0) 1></pre> |
| 用法 | 此查询报告是否启用了 compatibility mode 。如果使用 enable compatibility mode 时发生冲突，则会显示警告。 |
| 权限 | 适用于所有用户。 |
| 另请参见 | 有关详细信息，请参见《迁移技术指南》。 |

sp_configure

| | |
|----|---|
| 说明 | 按组显示配置参数、其当前值、非缺省值设置、最近设置的值以及此设置所使用的内存量。只显示其显示级别不高于用户的显示级别的参数。 |
| 语法 | <pre>sp_configure [configname [, configvalue] group_name non_unique_parameter_fragment] 'drop instance' [, instance_name] [display_nondefault_settings] sp_configure "configuration file", 0, {"write" "read" "verify" "restore"} "file_name"</pre> |
| 参数 | <p>configname 显示符合 <i>parameter</i> 的所有参数的当前值、缺省值、最近更改的值以及该设置所使用的内存量。</p> <p>configvalue 将 <i>configname</i> 重新设置为 <i>configvalue</i>，并显示当前值、缺省值、配置值和 <i>configname</i> 所使用的内存量。</p> <p><code>sp_configure configname, 0, "default"</code> 将 <i>configname</i> 重置为缺省值，并显示当前值、缺省值、配置值和 <i>configname</i> 使用的内存量。</p> <p>group_name 显示 <i>group_name</i> 中的所有配置参数、它们的当前值、缺省值、最近设置的值（如果适用）以及该设置所使用的内存量。</p> <p>non_unique_parameter_fragment 显示符合 <i>non_unique_parameter_fragment</i> 的所有参数名，它们的当前值、缺省值、配置值以及所使用的内存量。</p> <p>drop instance 可用于删除实例特定的配置设置。</p> <p>instance_name 在集群环境中 — 指示要为其设置特定于实例的选项的实例。</p> <p>display_nondefault_settings 显示配置值或运行值不同于缺省值的配置选项。</p> <p>write 从当前配置中创建 <i>file_name</i>。如果 <i>file_name</i> 已经存在，则向错误日志写入一条消息，并使用约定 <i>file_name.001</i>, <i>file_name.002</i>（依此类推）对现有文件进行重命名。如果已经更改了静态参数但没有重新启动服务器，则“write”会提供该参数的当前运行值。</p> |

read

对包含在 *file_name* 中的值执行检验检查，并且将通过检验的值读入服务器。如果 *file_name* 遗漏了任何参数，则使用这些参数的当前运行值。

verify

对 *file_name* 中的值执行检验检查。

restore

用 *sysconfigures* 中的值创建 *file_name*。在配置文件的所有副本都已丢失，而需要生成一个新的副本时，这会很有帮助。

file_name

是要在其上使用 *sp_configure* 的文件的名称。

示例

示例 1 按组显示所有配置参数、其当前值、缺省值、最近设置的值（如果适用）以及该设置所使用的内存量：

```
sp_configure
```

示例 2 显示包括 “identity” 一词的所有配置参数：

```
sp_configure "identity"
```

```
Configuration option is not unique.
```

| Parameter Name | Default | Memory Used | Config Value | Run Value | Unit | Type |
|----------------------|---------|-------------|--------------|-----------|-------|--------|
| identity burning set | 1 | 0 | 1 | 1 | id | static |
| identity grab size | 0 | 0 | 0 | 0 | id | dyna |
| size of auto identit | 10 | 0 | 10 | 10 | bytes | dyna |
| ... | | | | | | |

示例 3 将系统 *recovery interval in minutes* 设置为 3 分钟：

```
sp_configure "recovery interval in minutes", 3
```

| Parameter Name | Default | Memory Used | Config Value | Run Value | Unit | Type |
|-------------------|---------|-------------|--------------|-----------|------|------|
| recovery interval | 5 | 0 | 3 | 3 | min | dyn |

```
Configuration option changed. The SQL Server need not be rebooted since the option is dynamic.
```

示例 4 将 *number of devices* 的值重置为 Adaptive Server 缺省值：

```
sp_configure "number of device", 0, "default"
```

示例 5 若要配置同时恢复 4 个数据库，请输入：

```
sp_configure "max concurrently recovered db", 4
```

示例 6 若要启动 4 个检查点任务，请输入：

```
sp_configure "number of checkpoint tasks", 4
```

示例 7 在服务器级别捕获查询处理指标（qp 指标）：

```
sp_configure "enable metrics capture", 1
```

示例 8 对 *srv.config* 文件中的值执行检验检查，并将通过检验的参数读入到服务器中。当前的运行值会替代没有通过验证检查的值：

```
sp_configure "configuration file", 0, "read",
"srv.config"
```

示例 9 对文件 *restore.config* 中的值执行验证检查：

```
sp_configure "configuration file", 0, "restore",
"generic.config"
```

示例 10 创建文件 *my_server.config*，并将服务器正在使用的当前配置值写入该文件：

```
sp_configure "configuration file", 0, "write",
"my_server.config"
```

示例 11 对 *\$\$SYBASE/backup_config.cfg* 中的值执行检验检查：

```
sp_configure "configuration file", 0, "verify",
"$$SYBASE/backup_config.cfg"
```

用法

- 任何用户都可以执行 `sp_configure` 来显示参数及其当前值的信息（但不能修改参数）。系统管理员可以执行 `sp_configure` 来更改大多数配置参数的值。只有系统安全员才能执行某些参数。本节的“权限”部分中列出了这些参数。
- `sp_configure` 允许以单位指示符指定配置参数的值。单位指示符用 `p` 或 `P` 代表页，用 `m` 或 `M` 代表兆字节，用 `g` 或 `G` 代表千兆字节，用 `t` 或 `T` 代表太字节。如果不指定单位，并且配置了控制内存的参数，则 Adaptive Server 使用逻辑页大小作为基本单位。
- 当执行 `sp_configure` 修改动态参数时：
 - 将更新配置值和运行值。
 - 将更新配置文件。
 - 更改会立即生效。
- 当执行 `sp_configure` 修改静态参数时：
 - 将更新配置值。
 - 将更新配置文件。
 - 只有当重新启动 Adaptive Server 后更改才会生效。

- 如果发出不带任何参数的 `sp_configure`，则它将以报告形式按组显示所有配置参数、其当前值、缺省值、最近设置的值（如果适用）以及该设置所使用的内存量：
 - 报告中的 `default` 列显示随 Adaptive Server 提供的值。如果没有明确重新配置参数，该参数将保留其缺省值。
 - `memory used` 列显示参数在其当前值所使用的内存量（以千字节为单位）。某些相关参数占用的内存来自同一内存池。例如，`stack size` 和 `stack guard size` 占用的内存已被计入 `number of user connections` 占用的内存。如果将这些参数使用的内存一一相加，则总值会大于实际使用的内存量。在 `memory used` 列中，与其它参数“共享”内存的参数标有井号 (#)。
 - `config_value` 列显示最近用 `sp_configure` 将配置参数设置成的值。
 - `run_value` 列显示 Adaptive Server 当前使用的值。该值会在用 `sp_configure` 修改参数值后更改，而对于静态参数，只有在重新启动 Adaptive Server 后才会发生更改。该值存储在 `syscurconfigs.value` 中。

注释 如果服务器使用的是不区分大小写的排序顺序，则不带参数的 `sp_configure` 会按字母顺序返回所有配置参数和组的列表，而不分组显示。

- 每个配置参数都具有一个关联的显示级别。有三个显示级别：
 - “basic” 级别 — 只显示最基本的参数。它适于最一般的服务器调优。
 - “intermediate” 级别 — 在显示所有 “basic” 参数的同时显示较为复杂的参数。该级别适于中等复杂级别的服务器调优。
 - “comprehensive” 级别 — 缺省显示级别。显示包括最复杂参数在内的所有参数。该级别适于对服务器进行高精度调优的用户。
 设置为其它显示级别时可使用配置参数的子集，从而减少 `sp_configure` 显示的信息量。

用于显示当前显示级别的语法是：

```
sp_displaylevel
```

- `sp_configure` 可以在没有打开的事务时在使用链式事务模式的会话中运行：
- 有关各个配置参数的信息，请参见《系统管理指南》。

设置集群配置参数

- 如果未指定配置选项或实例名，则显示的信息取决于 `system_view` 设置。
- 如果未指定配置选项但指定了实例名， Adaptive Server 将显示指定实例的所有特定于实例的配置设置。
- 如果指定了配置选项但未指定配置值和实例名， Adaptive Server 将显示“集群”视图下所有实例的指定选项的当前设置。如果指定了实例名， Adaptive Server 将显示指定实例的配置信息。
- 如果指定了配置选项和值但未指定实例， Adaptive Server 将为选项配置全集群范围的设置。但如果指定了实例名， Adaptive Server 将仅设置该实例的配置值。语法为：

```
sp_configure configuration_name, config_value, NULL,  
instance_name
```

- 不能从本地临时数据库内部设置配置选项。
- 如果已为实例的某个配置参数设置了特定于实例的设置，可将此参数重新配置为全集群范围的设置。
- 用户只能重新配置与自己连接的实例。

max concurrently recovered db

此参数确定数据库恢复期间的并行度：

- 当 Adaptive Server 未处于恢复过程时，此配置参数以静态方式生效。但是，当 Adaptive Server 处于恢复过程时，系统管理员可以动态强制执行依次恢复。
- `max concurrently recovered db` 的效力取决于数据库布局和基础 I/O 子系统的性能。

number of checkpoint tasks

此参数配置并行检查点：

- 并行检查点取决于数据库的布局和基础 I/O 子系统的性能。根据活动数据库的数目和处理写入操作的 I/O 子系统的功能调优此参数。
- 此配置参数是动态的。当减小此参数的值时，将清除某些检查点任务；当增大该值时，将创建其它任务。

权限

任何用户都可以执行 `sp_configure` 以显示有关参数及其当前值的信息。

只有系统管理员才能执行 `sp_configure` 修改以下参数的值：

- `enable logins during recovery`
- `enable semantic partitioning`

- max concurrently recovered db
- number of checkpoint tasks

只有系统管理员和系统安全员才能执行 `sp_configure` 修改配置参数。

只有系统安全员才能执行 `sp_configure` 修改以下参数的值：

| | |
|----------------------------------|--------------------------------|
| allow procedure grouping | enable encrypted columns |
| allow select on syscomments.text | remote access |
| allow updates | restricted decrypt permission |
| auditing | suspend auditing when full |
| current audit table | systemwide password expiration |

系统管理员可以修改所有其它参数。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|------------------|---|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |
| 73 | 不受选项控制的自动被审计的事件。 | 用 <code>sp_configure</code> 打开 <code>auditing</code> 参数 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — NULL • 代理信息 — set proxy 有效时的初始登录名 |
| 74 | 不受选项控制的自动被审计的事件。 | 用 <code>sp_configure</code> 关闭 <code>auditing</code> 参数 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — NULL • 代理信息 — set proxy 有效时的初始登录名 |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------|--------------|--|
| 82 | security | sp_configure | <ul style="list-style-type: none"> • <i>角色</i>— 当前活动角色 • <i>关键字或选项</i>— 配置参数的名称 • <i>以前值</i>— 旧参数值（如果该命令设置新的值） • <i>当前值</i>— 新参数值（如果该命令设置新的值） • <i>其它信息</i>— 配置参数的数量（如果设置参数）；配置文件的名称（如果使用配置文件来设置参数） • <i>代理信息</i>— set proxy 有效时的初始登录名 |

另请参见

有关 max concurrently recovered db 和 number of checkpoint tasks 的详细信息，请参见《系统管理指南》的第 27 章“备份和恢复用户数据库”。

命令 `set`

系统过程 [sp_dboption](#)、[sp_displaylevel](#)、[sp_helpconfig](#)、[sp_monitorconfig](#)、`set`

sp_copy_all_qplans

| | |
|----|--|
| 说明 | 将一个抽象计划组中的所有计划复制到另一个组中。 |
| 语法 | <code>sp_copy_all_qplans src_group, dest_group</code> |
| 参数 | <p><i>src_group</i> 是源抽象计划组的名称。</p> <p><i>dest_group</i> 是要向其中复制计划的抽象计划组的名称。</p> |
| 示例 | 将 dev_plans 组中的所有抽象计划复制到 ap_stdin 组中： <pre>sp_copy_all_qplans dev_plans, ap_stdin</pre> |
| 用法 | <ul style="list-style-type: none"> 在复制计划之前，目标组必须已经存在。它可能包含计划。 <code>sp_copy_all_qplans</code> 为源组中的每个计划调用 <code>sp_copy_qplan</code>。每个计划都作为单独的事务复制，所以任何使 <code>sp_copy_all_qplans</code> 无法完成的问题都不会影响已复制的计划。 当 <code>sp_copy_qplan</code> 无法复制特定的抽象计划时，它将输出相应的消息。当运行 <code>sp_copy_all_qplans</code> 时，您也会看到这些消息。 如果目标组中计划的查询文本与源组中的查询文本完全匹配并且用户 ID 相同，则不复制该计划，同时向用户发送一条提供该计划 ID 的消息，但复制进程会继续处理源组中的下一个计划。 复制大量的抽象计划可能需要相当长的时间，同时还需要数据库 <code>system</code> 段上的空间以及用来记录数据库更改的空间。使用 <code>sp_spaceused</code> 可检查 <code>sysqueryplans</code> 的大小以及 <code>system</code> 和 <code>logsegment</code> 的 <code>sp_helpsegment</code> 大小，以检查可用空间。 |
| 权限 | 任何用户都可以执行 <code>sp_copy_all_qplans</code> 来复制其所拥有的抽象计划。只有系统管理员或数据库所有者才能复制其他用户拥有的计划。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** `sp_copy_qplan`、`sp_help_qpgroup`

sp_copy_qplan

| | |
|----|--|
| 说明 | 将一个抽象计划复制到抽象计划组。 |
| 语法 | <code>sp_copy_qplan src_id, dest_group</code> |
| 参数 | <p><i>src_id</i> 是要复制的抽象计划的 ID。</p> <p><i>dest_group</i> 是目标抽象计划组的名称。</p> |
| 示例 | <code>sp_copy_qplan 2140534659, ap_stdin</code> |
| 用法 | <ul style="list-style-type: none"> 在复制抽象计划之前，目标组必须已经存在。因为计划已由计划 ID 唯一标识，所以无需指定源组。 当复制计划时，会生成一个新的计划 ID。即使系统管理员或数据库所有者复制计划，该计划仍保留其创建者的用户 ID。若要分配不同的用户 ID，系统管理员或数据库所有者可以使用 sp_export_qpgroup 和 sp_import_qpgroup。 如果目标组中计划的查询文本与源组中的查询文本完全匹配并且用户 ID 相同，则不复制该计划，同时向用户发送一条提供该计划 ID 的消息。 若要复制抽象计划组中的所有计划，请使用 sp_copy_all_qplans。 |
| 权限 | 任何用户都可以执行 <code>sp_copy_qplan</code> 复制其所拥有的计划。只有系统管理员或数据库所有者才能复制其他用户拥有的计划。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_copy_all_qplans](#)、[sp_help_qpgroup](#)、[sp_help_qplan](#)、[sp_import_qpgroup](#)

sp_countmetadata

| | |
|----|---|
| 说明 | 显示 Adaptive Server 中索引、对象或数据库的数目。 |
| 语法 | <code>sp_countmetadata "configname" [, dbname]</code> |
| 参数 | <p>configname 是 “number of open databases”、“number of open objects”、“number of open indexes” 或 “number of open partitions”。</p> <p>dbname 是要在其上运行 <code>sp_countmetadata</code> 的数据库的名称。如果未提供任何数据库名称，<code>sp_countmetadata</code> 将提供所有数据库的总计数。</p> |
| 示例 | <p>示例 1 报告 Adaptive Server 中用户对象的数目。使用此值可以设置数据库中允许的对象数，以及附加对象和临时表的空间：</p> <pre>sp_configure "number of open objects", 310 sp_countmetadata "open objects" There are 283 user objects in all database(s), requiring 117.180 Kbytes of memory. The 'open objects' configuration parameter is currently set to a run value of 500.</pre> <p>示例 2 报告 Adaptive Server 中索引的数目：</p> <pre>sp_countmetadata "open indexes", pubs2 There are 21 user indexes in pubs2 database(s), requiring 8.613 kbytes of memory. The 'open indexes' configuration parameter is currently set to 600.</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_countmetadata</code> 显示 Adaptive Server 中索引、对象、数据库或分区的数目，其中包括系统数据库（例如 <code>model</code> 和 <code>tempdb</code>）的数目。 • 应避免在 Adaptive Server 高峰期运行 <code>sp_countmetadata</code>。它会导致 <code>sysindexes</code>、<code>sysobjects</code>、<code>sysdatabases</code> 和 <code>syspartitions</code> 系统表上出现争用。 • 如果需要有关特定数据库的信息，可对指定数据库运行 <code>sp_countmetadata</code>。但是，在为索引、对象、数据库或分区配置高速缓存时，应在不使用 <code>database_name</code> 选项的情况下运行 <code>sp_countmetadata</code>。 • <code>sp_countmetadata</code> 返回的内存信息会随平台不同有所变化。例如，对于 Windows NT 和 Sun Solaris，Adaptive Server 上的相同数据库会得到不同的 <code>sp_countmetadata</code> 结果。但有关用户索引、对象、数据库或分区数量的信息应该一致。 |

- sp_countmetadata 在计算中不包括临时表。可以将 open objects 值加 5%，将 open indexes 和 open partitions 值加 10% 以将临时表考虑在内。
- 如果为 configname 指定不唯一的“open indexes”、“open objects”、“open databases”或“open partitions”片断，sp_countmetadata 将返回带有配置值和当前值的匹配配置参数名的列表。例如：

```
sp_countmetadata "open"

Configuration option is not unique.
option_name                config_value  run_value
-----
currread change w/ open cursors      1           1
number of open databases              12          12
number of open indexes                500         500
number of open objects                500         500
open index hash spinlock ratio        100         100
open index spinlock ratio             100         100
open object spinlock ratio            100         100
```

权限 只有系统管理员或数据库所有者才能执行 sp_countmetadata。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_configure](#)、[sp_helpconfig](#)、[sp_monitorconfig](#)

sp_cursorinfo

说明

报告有关特定游标或为会话激活的所有执行游标的信息。

语法

```
sp_cursorinfo [{cursor_level | null}] [, cursor_name]
```

参数

cursor_level | null

是 Adaptive Server 为游标返回信息的级别。可以为 *cursor_level* 指定以下级别：

| 级别 | 游标类型 |
|----|--------------------------------------|
| N | 在特定过程嵌套级别的存储过程内声明的任何游标。可以为此级别指定任何正数。 |
| 0 | 在存储过程外声明的任何游标。 |
| -1 | 上述两个级别的任意游标。可以为此级别指定任何负数。 |

若要了解任何游标级别上具有特定 *cursor_name* 的游标，请为此参数指定 null。

cursor_name

是游标的特定名称。Adaptive Server 报告在指定的 *cursor_level* 上使用该名称的所有活动游标的信息。如果省略了此参数，Adaptive Server 将报告该级别上所有游标的信息。

示例

示例 1 显示级别 0 上名为 c 的游标的信息：

```
1> declare c cursor
2> for select au_id, au_lname, au_fname from authors
3> go
1> sp_cursorinfo
2> go

Cursor name 'c' is declared at nesting level '0'.
The cursor is declared as NON-SCROLLABLE cursor.
The cursor id is 917505.
The cursor has been successfully opened 0 times.
The cursor will remain open when a transaction is
committed or rolled back.
The number of rows returned for each FETCH is 1.
The cursor is updatable.
This cursor is using 5389 bytes of memory.

(return status = 0)
```

示例 2 显示有关游标的可滚动性和敏感性的信息，在本例中为半敏感的可滚动游标 `css`：

```
sp_cursorinfo 0, cursor_css
```

```
-----
```

```
Cursor name 'css' is declared at nesting level '0'.
The cursor is declared as SEMI_SENSITIVE SCROLLABLE cursor.
The cursor id is 786434.
The cursor has been successfully opened 1 times.
The cursor was compiled at isolation level 1.
The cursor is currently scanning at a nonzero isolation level.
The cursor is positioned on a row.
There have been 1 rows read, 0 rows updated and 0 rows deleted through this
cursor.
The cursor will remain open when a transaction is committed or rolled back.
The number of rows returned for each FETCH is 1.
The cursor is read only.
This cursor is using 19892 bytes of memory.
There are 2 columns returned by this cursor.
The result columns are:
Name = 'c1', Table = 't1', Type = INT, Length = 4 (not updatable)
Name = 'c2', Table = 't1', Type = INT, Length = 4 (not updatable)
```

用法

- 如果未指定 `cursor_level` 或 `cursor_name` 中的一个，Adaptive Server 将显示所有活动游标的信息。活动游标是由用户声明并由 Adaptive Server 分配的游标。
- Adaptive Server 报告每个游标的以下信息：
 - 游标名、嵌套级别、游标 ID 和过程名（如果在存储过程中声明）。
 - 游标已被打开的次数。
 - 对其进行编译或当前扫描（如果打开）所在的隔离级别（0、1 或 3）。
 - 游标是打开还是关闭。如果游标打开，则显示当前游标位置和读取的行数。
 - 如果游标的当前位置被删除，是否将关闭已打开的游标。
 - 如果提交或回退游标的当前事务，游标是保持打开状态还是被关闭。
 - 为该游标的每次读取返回的行数。

- 该游标是可更新的还是只读的。
- 游标返回的列数。对于每一列，都将显示列名、表名或表达式结果以及该列是否可以更新。

根据游标的状态，`sp_cursorinfo` 的输出有所不同。除了列出的信息之外，`sp_cursorinfo` 还显示游标的 `showplan` 输出。有关 `showplan` 的详细信息，请参见《性能和调优指南》。

权限 任何用户都可以执行 `sp_cursorinfo`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** `declare cursor`、`set`

sp_dbextend

说明

使您能够：

- 在数据库 / 段对和设备上安装自动数据库扩展过程。
- 为单个段和设备定义特定于站点的策略。
- 模拟数据库扩展机制的执行，在进行大容量卷装载前研究这一操作。

这些策略存储在 **master** 数据库的 **sysattributes** 表中。

所有参数都是字符串参数：

语法

```
sp_dbextend 'help' [, command]  
  
sp_dbextend [ ['set', ['threshold', dbname, segmentname, freespace |  
                'database', dbname, segmentname { [ [, growby] [, maxsize] } ] |  
                'device', devicename { [ [, growby] [, maxsize] } ] } |  
                'clear', 'threshold', dbname, segmentname  
  
sp_dbextend 'clear', 'database' [, dbname [, segmentname ] ]  
sp_dbextend 'clear', 'device' [, devicename ]  
sp_dbextend 'modify', 'database', dbname, segmentname,  
            { 'growby' | 'maxsize' }, newvalue  
sp_dbextend 'modify', 'device', devicename, { 'growby' | 'maxsize' },  
            newvalue  
  
sp_dbextend { 'list' | 'listfull' } [, 'database' [, dbname [, segmentname  
            [, order_by_clause ] ] ] ]  
sp_dbextend { 'list' | 'listfull' } [, 'device' [, devicename [, order_by_clause ] ] ]  
sp_dbextend { 'list' | 'listfull' } [, 'threshold' [, @dbname  
            [, @segmentname ] ] ]  
  
sp_dbextend 'check', 'database' [, dbname [, segmentname ] ]  
sp_dbextend { 'simulate' | 'execute' }, dbname, segmentname [, iterations ]  
sp_dbextend 'trace', { 'on' | 'off' }  
sp_dbextend 'reload [defaults]  
sp_dbextend { 'enable' | 'disable' }, 'database' [, dbname [, segmentname ] ]  
sp_dbextend 'who' [, 'spid' | 'block' | 'all' ] ]
```

参数

set

设置应引发数据库、段或设备的阈值。参数包括：

- **threshold** — 指定在指定的数据库和段上安装阈值的可用空间级别。您应该始终使用大小单位指示符（例如兆字节）指定 **freespace**。如果未指定任何大小单位，则 **freespace** 的值被认为是段中的千字节数。
- **database** — 指定数据库 / 段对的名称、数据库更改幅度以及数据库的最大大小（达到该最大大小时，扩展进程停止）。
growby — 是每次扩展尝试时数据库增长的比率，以单位指示符或百分比值表示。**maxsize** 是段的最大大小，达到最大大小后不再进行扩展。这两个参数都是可选的。
- **device** — 定义设备的增长率和可以增长到的最大大小，用单位指示符或百分比值表示。设备中的 **maxsize** 受 OS 磁盘限制。

clear

清除指定数据库和段或者指定设备以前设置的扩展规则。

modify

修改数据库和段以前设置的特定于站点的策略，例如 **growby** 和 **maxsize**。

使用 **newvalue** 指定为自动扩展设置的新值。

list

简要列出指定数据库、段、设备或指定段的阈值的现有规则，并以一种可读格式显示来自 **master.dbo.sysattributes** 的数据。允许您按每个数据库或每个设备查看规则。

显示当前有效的规则。

使用 **order_by_clause** 生成与缺省的名称、类型排序顺序不同的列表。

使用 **threshold** 可以显示当前在指定数据库中（使用 **@dbname**）和段中（使用 **@segment name**）安装的所有阈值。

listfull

完全列出特定于站点的策略规则，并在 **sysattributes** 表中包括一个 **comment** 列，该列显示的 **datetime** 戳指出规则是何时设置、何时最后修改的。

check

检查当前策略并验证它们与每个段中的当前空间布局是否一致。如果发现任何策略设置多余、无效或不正确，则将显示一条警告消息。

simulate

根据 **set** 命令实施的当前策略集，模拟执行在运行期执行的数据库或设备扩展方案。

iterations 指定模拟扩展的次数。

execute

使用当前策略集执行实际数据库 / 段或设备扩展。

reload defaults

在所有数据库、段和设备中用 *growby* 和 *maxsize* 的系统提供的缺省值重新初始化 *sysattributes*，并将数据库或设备回复到初始缺省行为。

help

提供所有命令参数（例如 **set** 或 **list**）的帮助信息或任何单个命令的帮助信息。

trace

跟踪所有扩展进程中的阈值过程执行逻辑。

enable、disable

在指定数据库段或设备上启用或禁用自动扩展过程。

who

显示当前运行的任何活动扩展进程。“<spid>”限制特定 spid 的输出。
使用：

- *block* — 显示当前导致扩展进程阻塞的任务。
- *all* — 显示所有当前活动的任务。

freespace

指定可用空间为何值时在指定的段上安装阈值过程。一定要使用大小单位指示符（例如 **MB**）来指定 *freespace*。

dbname

是在其中安装阈值的数据库的名称。

segmentname

是包含在数据库 *dbname* 中的段。

devicename

是相关设备的逻辑名。

newvalue

指定当您修改数据库 / 段或对设备的策略时为自动扩展设置的新值。

order_by_clause

生成与 *list* 命令中的缺省排序不同顺序的列表。缺省顺序是名称、类型。

iterations

指定模拟或执行扩展的次数。

growby

指定每次尝试执行阈值过程时指定的数据库段或设备增长的比率，以单位指示符或百分比值表示。

maxsize

是段 / 数据库对或设备的最大大小，当达到该大小时，自动扩展必须停止。

示例

示例 1 *set thresholds* — 在数据库 `pubs2` 中的日志段上安装 100 MB 的空间扩展阈值：

```
sp_dbextend 'set', 'thresh', pubs2, logsegment, '100m'
```

示例 2 *set database* — 为 `logsegment` 段安装策略，每次扩展尝试的增长率为 100 MB：

```
sp_dbextend 'set', 'database', pubs2, logsegment, '100m'
```

示例 3 *set device* — 扩展此设备直到达到 OS 磁盘空间限制或设备大小达到 32GB：

```
sp_dbextend 'set', 'device', pubs2-datadev1, '100m'
```

示例 4 *clear* — 说明如何清除以前在 `pubs2` 中 `logsegment` 上安装的所有空间扩展阈值：

```
sp_dbextend 'clear', 'thresh', pubs2, logsegment
```

您还可以清除在 `pubs2` 中的段 `dataseg1` 上安装的可用空间为 200 MB 的空间扩展阈值：

```
sp_dbextend 'clear', 'thresh', pubs2, dataseg1, '200m'
```

示例 5 *modify* — 将每次扩展尝试的增长率指定为当前值的 5%：

```
sp_dbextend 'modify', 'da', pubs2, logsegment, 'growby', '5%'
```

如果以前未定义 `maxsize`，则命令可能失败：

```
sp_dbextend 'modify', 'device', pubs2_log_dev, 'maxsize', '2.3g'
```

示例 6 *list* — 简要列出所有数据库和设备的规则：

```
sp_dbextend 'list'
```

以下命令列出名称类似于 “`pubs%`” 的所有数据库的规则：

```
sp_dbextend 'list', 'database', 'pubs%'
```

示例 7 listfull — 列出所有数据库和设备的规则，其中包括一个显示 `datetime` 戳的 `comment` 列：

```
sp_dbextend 'listfull'
```

示例 8 list threshold — 从 `pubs2` 数据库中发出该命令时，将列出 `pubs2` 数据库中各段的阈值设置：

```
sp_dbextend 'list', 'threshold'
```

要检查特定段上的阈值，请使用：

```
sp_dbextend 'list', 'threshold', pubs2, 'logsegment'
```

示例 9 simulate — 模拟扩展两次，两次扩展都没有达到阈值：

```
sp_dbextend 'simulate', pubs2, logsegment, '2'
```

示例 10 execute — 执行自动扩展过程：

```
sp_dbextend 'execute', pubs2, logsegment
```

示例 11 help — 获得特定命令的帮助：

```
sp_dbextend help, 'set'
```

用法

- 在任何给定的数据库 / 段对上只能设置一个自动扩展阈值。如果尝试安装阈值过程的另一个实例，即使是在不同的可用空间值上，也会引发错误。
- 您不能设置系统提供的缺省值，只能修改它们。在修改系统缺省值之后，可以通过重新运行 `installdbextend` 脚本或使用 `reload defaults` 命令重新设置它们。
- 若要在特定段中禁止自动增长，请将 `growby` 或 `maxsize` 指定为 0 或根本不安装该阈值过程。如果为此参数指定 `NULL`，则缺省使用系统提供的缺省 `growby` 率。
- 缺省情况下，如果设备的大小大于 40MB，则数据库的大小将增大 10%。如果数据库的大小小于 40 MB，则数据库的大小将增大 4 MB。但是，您可以根据您的站点需要指定数据库大小调整限制。
- `maxsize` 是段的最大大小，当达到该大小时自动扩展进程停止，该值并不是数据库的最大大小。
- 对于 `default` 数据库，没有系统指定的最大大小。如果未指定任何 `maxsize` 值，则数据库的大小仅受数据库设备物理大小的限制。
- 若要关闭特定设备上的自动增长功能，请将 `growby` 或 `maxsize` 指定为 0。如果未指定 `growby` 的值，则使用缺省扩展率。
- 可以将 `maxsize` 值设置为大于设备上的可用磁盘空间总量，但尝试进行扩展时实际扩展受可用磁盘空间的限制。

- 当使用此存储过程清除阈值时，*dbname* 和 *segmentname* 是必需的参数。
- 当使用此存储过程清除数据库时，如果未提供 *dbname* 和 *segmentname*，则当前数据库和其中所有段的所有策略规则（即 *master.dbo.sysattributes* 中的所有相关行）都将被删除。若要将所有设置恢复为缺省值并重新启动，这是一个不错的方法。
- 当使用此存储过程清除设备时，如果未提供 *devicename* 的值，则不会清除任何策略规则。您可以提供 *devicename* 来清除单个设备的策略规则或使用 “%” 清除所有设备的策略。
- 您可以使用模式指定 *dbname*、*devicename* 和 *segmentname*，以便对名称模式符合指定模式的对象执行 *clear*、*enable*、*disable* 和 *list* 操作。
- 必须先用 *set* 命令设置一个值或属性，然后才能修改它。如果先前未设置任何值，则 *modify* 失败。*growby* 和 *maxsize* 会被修改为 *newvalue* 指定的新值。
- *newvalue* 中指定的新值将在随后的扩展数据库或设备的尝试中一直有效。即使 *newvalue* 小于数据库、段或设备的当前值，对象也不会缩减。*newvalue* 仅指定以后的扩展，不会影响当前大小。
- 如果对数据库使用 *list* 并且不提供任何 *dbname* 或 *segmentname*，则将列出当前数据库中所有段的所有策略规则（即，*master.dbo.sysattributes* 中的行）。
- 提供 *dbname* 和 *segmentname* 可以获得个别数据库和其中的段的策略规则。
- 如果对设备名使用 *list* 并且不提供任何 *devicename*，则将列出所有设备的缺省策略规则。您可以通过提供 *devicename* 对列出的内容进行过滤以列出单个设备的策略规则或对 *devicename* 使用模式指示符。
- 一次只能模拟一个数据库 / 段对的扩展。*dbname* 和 *segmentname* 都是必需的参数。使用 *execute* 或 *simulate* 命令时，不能在 *dbname* 或 *segmentname* 中使用通配符模式。
- 设备的最大大小是 32 GB。
- 在使用 *modify* 和 *simulate* 后，请使用 *reload* 重新初始化您的数据库和设备。*reload* 将删除 *master.dbo.sysattributes* 中描述系统缺省行为的任何现有行，然后装载新行。
- *reload* 不删除用户指定的策略。
- *trace* 打开或关闭整个服务器的跟踪功能。如果 *trace* 打开，当触发阈值时，消息出现在服务器错误日志中。仅将 *trace* 用于故障排除。

权限 运行 `installdbextend` 脚本需要 `sa_role` 权限，只将 `sp_dbextend` 的 `execute` 权限授予 `public`。

任何用户都可以执行 `list` 参数。必须授予指定数据库上的数据库所有者或 `sa_role` 权限才能执行所有其它命令。

诸如 `clear` 这样允许对 `dbname` 参数使用模式指示符的命令需要 `sa_role` 特权。

以下命令参数需要 `sa_role` 特权：`simulate`、`execute`、`check`、`reload defaults`、`trace`。

如果不具有 `sa_role` 特权的数据库所有者在某个段上安装了自动扩展过程，则设备将不会扩展，这是因为用户不能运行 `disk resize` 命令。Sybase 建议具有 `sa_role` 特权的用户在安装阈值过程时运行 `set threshold` 命令。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **存储过程** [sp_droptreshold](#)、[sp_modifythreshold](#)
命令 [alter database](#)、[create database](#)、[disk init](#)、[disk resize](#)

sp_dboption

| | |
|----|---|
| 说明 | 显示或更改数据库选项，并启用异步日志服务功能。 |
| 语法 | <code>sp_dboption [dbname, optname, optvalue [, dockpt]</code> |
| 参数 | <p>dbname 是要在其中设置选项的数据库的名称。必须使用 master 来执行带参数的 sp_dboption（即更改数据库选项）。但是，您不能更改 master 数据库中的选项设置。</p> <p>optname 是要设置的选项的名称。Adaptive Server 可以识别属于选项名一部分的任何唯一字符串。如果选项名是关键字或含有嵌入的空格或标点符号，则请用引号将选项名引起来。</p> <p>可以一次打开多个数据库选项。不能在用户定义的事务中更改数据库选项。</p> <p>optvalue 是设置的值。true 将打开该选项，false 将关闭该选项。</p> <p>dockpt 指定 sp_dboption 是否对 dbname 执行 checkpoint 命令。缺省值为 1，表示自动执行 checkpoint。可以通过手动执行 checkpoint 命令来对 dbname 运行 checkpoint。</p> |

示例 **示例 1** 显示数据库选项列表：

```
sp_dboption
Settable database options
  database_options
-----
abort tran on log full
allow nulls by default
allow wide dol rows
async log service
auto identity
dbo use only
ddl in tran
delayed commit
enforce dump tran sequence
full logging for all
full logging for alter table
full logging for reorg rebuild
full logging for select into
identity in nonunique index
no chkpt on recovery
no free space acctg
```

```
read only
scratch database
select into/bulkcopy/pullsort
single user
trunc log on chkpt
trunc. log on chkpt.
unique auto_identity index
```

示例 2 将数据库 pubs2 设置为只读:

```
1> use pubs2
2> go
1> master..sp_dboption pubs2, "read", true
2> go
```

Database option 'read only' turned ON for database 'pubs2'.
Running CHECKPOINT on database 'pubs2' for option 'read only' to take effect.
(return status = 0)

read 字符串在所有可用数据库选项中唯一确定 read only 选项。请注意，使用引号将关键字 read 引起来。

示例 3 使数据库 pubs2 重新变为可写的，但通过为 *dochkpt* 选项指定 0，您可以看到消息 “Run the CHECKPOINT command in the database that was changed”:

```
1> use pubs2
2> go
1> master..sp_dboption pubs2, "read", false, 0
2> go
```

Database option 'read only' turned OFF for database 'pubs2'.
Run the CHECKPOINT command in the database that was changed.
(return status = 0)

若要在 pubs2 上手动执行检查点，请输入:

```
1> checkpoint
2> go
```

示例 4 允许对 pubs2 数据库中的表进行 select into、bcp 和并行排序操作。select into 字符串在所有可用数据库选项中唯一确定 select into/bulkcopy 选项:

```
use pubs2
go
master..sp_dboption pubs2, "select into", true
go
```

注释 因为有嵌入的空格，所以需要引号将选项引起来。

示例 5 自动在 mydb 中创建的新表中定义 10 位的 IDENTITY 列。如果在创建新表时未指定主键、unique 约束或 IDENTITY 列，则将在每个这样的新表中定义 IDENTITY 列 SYB_IDENTITY_COL:

```
use mydb
go
master..sp_dboption mydb, "auto identity", true
go
```

示例 6 自动在 mydb 表的索引键中包括 IDENTITY 列，但前提条件是这些表已经具有了 IDENTITY 列。在表上创建的所有索引都将是内部唯一的:

```
use master
go
sp_dboption mydb, "identity in nonunique index", true
go
use mydb
go
```

示例 7 自动为 pubs2 数据库中的新表包括带有唯一非聚簇索引的 IDENTITY 列:

```
use master
go
sp_dboption pubs2, "unique auto_identity index", true
go
use pubs2
go
```

示例 8 在指定的数据库中设置异步日志服务 (ALS)，启用用户日志高速缓存和日志写入器线程。

```
sp_dboption "mydb", "async log service", true
use mydb
```

示例 9 在指定的数据库中禁用 ALS。

```
sp_dboption "mydb", "async log service", false
use mydb
```

示例 10 强制对 big_db 数据库进行 dump tran sequence 操作:

```
sp_dboption 'big_db', 'enforce dump tran sequence',
true
```

示例 11 在 mydb 数据库中启用 `select into` 和 `alter table` 命令的完全日志记录:

create database 命令创建 mydb:

```
create database mydb on datadev=20 log on logdev=10
go
CREATE DATABASE: allocating 10240 logical pages (20.0 megabytes) on disk
'datadev' (10240 logical pages requested).
CREATE DATABASE: allocating 5120 logical pages (10.0 megabytes) on disk
'logdev' (5120 logical pages requested).
Database 'mydb' is now online.
```

在 mydb 中开启 `select into` 的完全日志记录选项:

```
sp_dboption "mydb", "full logging for select into", "true"
go
Database option 'full logging for select into' turned ON for database
'mydb'.
Running CHECKPOINT on database 'mydb' for option 'full logging for select
into' to take effect.
(return status = 0)
```

在 mydb 中开启 `alter table` 的完整日志记录选项:

```
sp_dboption "mydb", "full logging for alter table", "true"
go
Database option 'full logging for alter table' turned ON for database
'mydb'.
Running CHECKPOINT on database 'mydb' for option 'full logging for alter
table' to take effect.
(return status = 0)
```

运行 `sp_helpdb` 显示 mydb 的设置:

```
sp_helpdb mydb
go
name db_size owner dbid created durability status
-----
mydb 30.0 MB sa 5 Dec 16, 2010 full full logging for select
into/alter table

(1 row affected)
device_fragments size usage created free kbytes
-----
datadev 20.0 MB data only Dec 16 2010 6:08PM 18696
logdev 10.0 MB log only Dec 16 2010 6:08PM not applicable

-----
log only free kbytes = 10184
```



```
(return status = 0)
1>
```

用法

- 不能更改 `master` 数据库选项设置。
- 如果为 `optname` 输入了一个不明确的值，则会显示一条错误消息。例如，两个数据库选项分别为 `dbo use only` 和 `read only`。使用“only”作为 `optname` 参数的值将产生一条消息，因为它与两个名称都匹配。将显示与所给的字符串匹配的完整名称，以便可以查看如何使 `optname` 更具体。
- 若要显示数据库选项列表，请在 `master` 数据库中执行不带参数的 `sp_dboption`。
- 若要得到关于特定数据库中设置的数据库选项的报告，请执行 `sp_helpdb`。
- 如果用 `sp_dboption` 为同一个数据库同时设置了 `no chkpt on recovery` 选项和 `trunc log on chkpt` 选项，则前者将禁用后者。在 `trunc log on chkpt` 设置为缺省值 `on` 的 `tempdb` 数据库中出现这种冲突的可能性非常大。
- 数据库所有者或系统管理员可以通过在 `model` 上执行 `sp_dboption`，为所有新数据库设置或取消设置特定的数据库选项。
- 执行完 `sp_dboption` 之后，直到在为其更改选项的数据库中发出 `checkpoint` 命令之后，所进行的更改才会生效。

完全日志记录

缺省情况下，`select into`、某些类型的 `alter table` 以及 `reorg rebuild` 将在最少日志记录模式中运行。在执行这些命令之前，请先将 `select into/bulk copy` 数据库选项设置为 `true` 以允许 Adaptive Server 中断转储序列，也就是说，执行操作以阻止使用 `dump transaction` 的能力。

当您使用“full logging for [select into | alter table | reorg rebuild | all]”选项时，该命令会在完全日志记录模式中运行。对于任何立即完全日志记录命令，以前设置的任何 `select into/bulk copy` 值都会变得毫不相关。

不支持对快速 `bcp` 和并行排序使用完全日志记录，而且只有在将 `select into/bulk copy` 设置为 `true` 时才会发生。

一旦操作设置为完全日志记录，您就可以对这些操作运行 `dump transaction/load transaction` 以及恢复，就像任何其它完全日志记录操作一样。

缺省情况以最少日志记录模式记录的完全日志记录命令的语法是：

```
sp_dboption dbname, "full logging for  
[select into | alter table | reorg rebuild | all]",  
true | false
```

其中：

- **full logging for select into** — 为了让 **select into proxy table** 进行完全日志记录，请在承载实际表的远程服务器上将 “full logging for select into” 选项设置为 **true**。如果在承载实际表的远程服务器上将 **full logging for select into** 选项设置为 **false**，该命令会在该数据库中以最少日志记录模式执行，而且转储事务序列会中断。
- **full logging for alter table** — 对以下需要移动数据的 **alter table** 版本启用完全日志记录：
 - **alter table add *column* not null**
 - **alter table drop *column* not null**
 - **alter table modify *datatype* of not null *column***

alter table 的其它变体已经在完全日志记录模式中执行。

- **full logging for reorg rebuild** — 包含表数据移动。这对已经在完整日志记录模式中记录的 **reorg rebuild** 索引命令没有影响。此参数启用 **full logging for reorg rebuild** 表语句。如果不设置此选项（或者将此选项设置为 **false**），Adaptive Server 会在最少日志记录模式中执行 **reorg rebuild** 表命令。
- **full logging for all** — 启用所有上述完全日志记录选项。将 **all** 设置为 **false** 会禁用所有完全日志记录选项。

注释 此语法需要您指定想要完全记录的内容；“full logging”自身不是有效的选项。

当您使用任何 **full logging for** 选项时，该命令会在完全日志记录模式中运行。对于任何立即完全日志记录命令，以前设置的任何 **select into/bulk copy/pllsort** 值都会变得毫不相关。不支持对快速 **bcp** 和并行排序使用完全日志记录，而且只有在将 **select into/bulk copy/pllsort** 设置为 **true** 时才会发生。

一旦操作设置为完全日志记录，您就可以对这些操作运行 **dump transaction/load transaction** 以及恢复，就像任何其它完全日志记录操作一样。

dboption 是 “full logging for all”，而不仅仅是单独的 “full logging”。

注释 完全记录 `select into`、`alter table` 或 `reorg rebuild` 命令的执行可能需要大量的日志空间来容纳事务日志。

缩减日志

如果在启用完全日志记录时发出 `select into`、`alter table` 和 `reorg rebuild`，可能会极大地增加日志空间的需求，特别是对于大表。您可能需要增加日志的大小。完成该命令后，可以使用 `alter database log off` 命令删除额外的日志空间。请参见 `alter database` 以及《系统管理指南第一卷》中的“缩减日志空间”。

不能针对以下数据库设置 `select into`、`alter database` 或 `reorg rebuild` 的完全日志记录：

- master 数据库
- 内存数据库

可以更改以下数据库的设置：

- 任何混有日志和数据段的数据库，但除非数据库改变为不再混有日志和数据段等情况，否则该选项会被忽略。
- 没有完全持久性级别的数据库，但除非数据库改变为具有完全持久性，否则该选项会被忽略。

之所以有这些限制，是因为这些数据库都不允许您执行 `dump transaction` 命令。使用完成可恢复的 DDL 可启用 `dump transaction`。

允许宽行

`allow wide dol rows` 将数据库配置为允许宽的、可变长度的仅数据锁定 (DOL) 行。

- 必须为每个数据库单独启用 `allow wide dol rows`。
- 只能在用户数据库中设置 `allow wide dol rows` 数据库选项。不能为 master 数据库设置 `allow wide dol rows` 数据库选项。
- 在配置了 8K 或更少页大小的 Adaptive Server 中启用 `allow wide dol rows` 不起作用。
- 禁用 `allow wide dol rows` 会阻止 Adaptive Server 创建宽的、可变长度的 DOL 行；它不会阻止您选择包括这种行的数据。但是，在启用 `allow wide dol rows` 之前，您不能更改包含宽数据的行，除非更改生成了不再包含宽数据的行。

- 临时数据库不能使用宽 DOL 工作表，除非您启用其 `allow wide dol rows` 设置。如果使用 `tempdb` 组，则对组中的所有数据库全部启用 `allow wide dol rows` 或者全部不启用，这样，工作表和查询处理行为就会在整个组中一致，无论与特定用户会话绑定的 `tempdb` 如何都是如此。

异步日志服务 (ALS) 选项

启用 `async log service (ALS)` 将使 Adaptive Server 具有更大的可伸缩性，从而可以为高端均衡多处理器系统的记录子系统提供更高的吞吐量。

- 缺省情况下禁用 ALS 选项。
- 不能在系统数据库（例如 `master` 或 `model`）中启用 ALS 选项。
- ALS 选项是持久的；在指定的数据库上启用 ALS 之后，您可以转储和重新装载数据库而无需禁用 ALS。若要禁用此功能，您必须使用 `sp_dboption` 将参数设置为 `false`。

有关 `enforce dump tran sequence` 的考虑事项

`enforce dump tran sequence` 可防止会禁止后续转储事务的操作：

- `false` —（缺省值）不影响与转储事务交互的操作。
- `true` — 不允许会禁止转储事务的操作。

只有在数据具有以下特点时，才可以将此选项设置为 `true`：

- 是专用日志数据库。
- 不是存档数据库。
- 不是本地或全局临时数据库。
- 不是只读的。
- 未联机以用于备用访问。
- 具有完全持久性。具有 `at_shutdown` 和 `no_recovery` 持久性的数据库是不允许的。
- `select into/bulk copy/plsort` 或 `trunc log on chkpt` 设置为 `false`。如果这些选择有的为 `true`，则它们会自动重置为 `false`。
- 由于以下原因之一，不需要 `dump database`。在将此数据库选项设置为 `true` 之前执行转储数据库。
 - 部分记录的更新已经完成，例如，`select into`、`alter table modify`、`reorg rebuild`、`fast bcp` 和 `writetext`。
 - 事务日志被截断。

- 它是新建的或升级的数据库。

如果数据库选项 `enforce dump tran sequence` 为 `true`，您不能：

- 将 `select into/bulk copy/pilsort` 设置为 `true`。具有部分日志记录的命令是不允许的。
- 将 `trunc log on chkpt` 设置为 `true`。日志不能由 checkpoint 进程截断。
- 执行 `dump tran with truncate_only` 或 `dump tran with no_log`。如果不能将日志转储到存档设备中，就不能将其截断。
- 将数据库标记为只读。
- 将持久性从 `full` 更改为 `at_shutdown` 或 `no_recovery`。
- 更改为 `mixed-log-and-data` 数据库。例如 `load database` 和 `dbcc findstranded`，可以将数据库更改为日志和数据混合。

数据库选项

- 当指定数据库的日志段中超过了最后机会阈值时，`abort tran on log full` 选项将决定如何处理正在运行的事务。其缺省值是 `false`，表示事务被挂起并只有在空间释放后才被激活。如果将此设置更改为 `true`，则所有需要写入事务日志的用户查询都将被注销，直到日志中的空间被释放。
- 将 `allow nulls by default` 选项设置为 `true` 可以将列的缺省值从 `not null` 更改为 `null`，以符合 SQL 标准。列的 Transact-SQL 缺省值是 `not null`，表示除非在 `create table` 或 `alter table` 列定义中指定了 `null`，否则列值不能为空。`allow nulls by default true` 与此相反。

不能使用 `allow nulls by default` 在 `select into` 语句中更改列的可空性。而应使用 `convert` 指定结果列的可空性。

- 当 `auto identity` 选项设置为 `true`（打开）时，如果在创建新表时未指定 `primary` 键、`unique` 约束或 `IDENTITY` 列，则将在每个新表中定义一个 10 位的 `IDENTITY` 列。当使用 `select *` 语句选择所有列时，该列不可见。若要检索它，必须在选择列表中明确提供此列的名称 `SYB_IDENTITY_COL`。

若要设置自动 `IDENTITY` 列的精度，请使用 `size of auto identity column` 配置参数。

尽管可以在 `tempdb` 中将 `auto identity` 设置为 `true`，但其并不被识别或使用，且在该数据库中创建的临时表也不会自动包括 `IDENTITY` 列。

若要得到有关包含 `IDENTITY` 列的特定表中索引的报告，请执行 `sp_helpindex`。

- 当 `dbo use only` 选项被设置为 `true`（打开）时，只有数据库所有者才能使用该数据库。
- 当 `ddl in tran option` 被设置为 `true`（打开）时，可以在事务中使用某些数据定义语言命令。如果在特定数据库中 `ddl in tran` 为 `true`，则允许在该数据库的事务中使用诸如 `create table`、`grant` 和 `alter table` 等命令。如果在 `model` 数据库中 `ddl in tran` 为 `true`，则对于那些在 `model` 中设置了 `ddl in tran` 之后创建的所有数据库来说，允许在事务中使用这些命令。

警告！ 数据定义语言 (DDL) 命令在系统表（例如 `sysobjects`）上持有锁。应避免在事务中使用它们；如果必须使用的话，请保持事务尽量简短。

在事务中对 `tempdb` 使用任何 DDL 命令可能导致系统速度减慢甚至挂起。应始终将 `tempdb` 中的 `ddl in tran` 设置为 `false`。

- 当将 `ddl in tran` 选项设置为 `true` 时，以下命令可用在用户定义的事务中：

| | |
|---|-----------------------------|
| <code>alter table</code> — 允许使用除 <code>partition</code> | <code>drop default</code> |
| 和 <code>unpartition</code> 之外的子句 | <code>drop index</code> |
| <code>create default</code> | <code>drop procedure</code> |
| <code>create index</code> | <code>drop table</code> |
| <code>create procedure</code> | <code>drop trigger</code> |
| <code>create rule</code> | <code>drop view</code> |
| <code>create schema</code> | <code>grant</code> |
| <code>create table</code> | <code>revoke</code> |
| <code>create trigger</code> | |
| <code>create view</code> | |

- 在任何情况下，以下命令都不能用在用户定义的事务中：

| | |
|--|--------------------------------|
| <code>alter database</code> | <code>dump transaction</code> |
| <code>alter table...lock</code> | <code>drop database</code> |
| <code>alter table...partition</code> | <code>load transaction</code> |
| <code>alter table...unpartition</code> | <code>load database</code> |
| <code>create database</code> | <code>select into</code> |
| <code>disk init</code> | <code>truncate table</code> |
| <code>dump database</code> | <code>update statistics</code> |

另外，不能在用户定义的事务中使用创建临时表或更改 `master` 数据库的系统过程。

- **identity in nonunique index** 自动在表的索引键中包含一个 IDENTITY 列，以使在表上创建的全部索引具有唯一性。此数据库选项使逻辑上非唯一的索引在内部唯一，并使这些索引可用于处理可更新游标和隔离级别 0 读取。

表必须已经具有 IDENTITY 列，**identity in nonunique index** 选项才能工作。IDENTITY 列可用 **create table** 语句或在创建表之前通过将 **auto identity** 数据库选项设置为 **true** 来创建。

如果计划在具有非唯一索引的表上使用游标和隔离级别 0 读取，则可以使用 **identity in nonunique index**。唯一索引可确保下一次在游标处执行 **fetch** 时，游标定位于正确的行。如果打算在具有唯一索引和任何隔离级别的表上使用游标，则最好使用 **unique auto_identity index** 选项。

不要将 **identity in nonunique index** 选项与 **unique auto_identity index** 混淆，后者用于向新表添加具有唯一非聚簇索引的 IDENTITY 列。

若要得到有关包含 IDENTITY 列的特定表中索引的报告，请执行 **sp_helpindex**。

- **no free space acctg** 禁止对非日志段执行可用空间计数和阈值操作。由于不对这些段重新计算可用空间计数，因此这可以缩短恢复时间。**no free space acctg** 禁止更新为每个表存储的每页行数，因此可能导致估计空间使用情况的系统过程报告不准确。
- 当保留有数据库的最新副本时，**no chkpt on recovery** 选项被设置为 **true**（打开）。在这些情况下，将既有“主”数据库又有“辅助”数据库。最初，主数据库被转储并被装载到辅助数据库中。然后，每隔一段时间，主数据库的事务日志被转储并装载到辅助数据库中。

如果将此选项设置为 **false**（关闭）（缺省状态），则当重新启动 Adaptive Server 时，在数据库恢复之后会向其添加一个检查点记录。此检查点可以确保不会不必要地再次运行恢复机制，并且可以更改序列号，从而使主数据库中对事务日志随后进行的装载失败。

如果将辅助数据库的此选项设置为 **true**（打开），则可以防止从恢复进程中取得检查点，从而保证主数据库中随后的事务日志转储能够被装载到其中。

- **read only** 选项表示用户可以从数据库检索数据，但不能修改任何数据。
- **select into/bulkcopy/pllsort** 必须设置为 **on**，才可执行不在日志中保留事务完整记录的操作，这些操作包括：
 - 使用 **writetext** 实用程序。

- 对永久表执行 `select into`。
- 使用 `bcp` 执行“快速”**批量复制**。缺省情况下，快速 `bcp` 用于没有索引的表。
- 执行并行排序。

事务日志转储不能恢复这些最低限度记录的操作，因此禁止对转储设备执行 `dump transaction`。但是，仍可以使用 `dump transaction...with no_log and dump transaction...with truncate_only`。在执行不在记录范围内的操作后，请将 `select into/bulk copy/pllsort` 设置为 `false`（关闭）并发出 `dump database`。

在用 `select into`、批量复制或并行排序对数据库进行了未记录的更改之后发出 `dump transaction` 语句将产生一条错误消息，指出应使用 `dump database`。（`writetext` 命令没有这样的保护。）

因为从不恢复 `tempdb`，所以对临时表执行 `select into` 时无需将 `select into/bulkcopy/pllsort` 选项设置为 `true`。不必为在带有索引的表上运行 `bcp` 而将该选项设置为 `true`，这是因为带有索引的表始终采用较慢形式的批量复制进行复制，而且将记录日志。

设置 `select into/bulkcopy/pllsort` 不会阻塞日志转储，但对数据进行的最小记录的更改会阻塞常规 `dump transaction` 的使用。

缺省情况下，在新创建的数据库中 `select into/bulkcopy/pllsort` 被关闭。若要更改该缺省设置，请在 `model` 数据库中打开此选项。

- 当 `single user` 设置为 `true` 时，每次只有一个用户可以访问该数据库（单用户模式）。

不能在存储过程的用户数据库中或当用户已打开数据库时将 `single user` 设置为 `true`。不能将 `tempdb` 的 `single user` 设置为 `true`。

- `trunc log on chkpt` 选项表示如果事务日志的已提交事务超过 50 行，则每次 `checkpoint` 检查进程发生（通常每分钟超过一次）时，事务日志都会被截断（已提交的事务将被删除）。但是，当数据库所有者手动运行 `checkpoint` 时，该日志不会被截断。在进行开发工作时，为防止日志增长可能需要将此选项打开。

当 `trunc log on chkpt` 选项打开时，因为不能使用截断的事务日志转储从介质故障中恢复，所以禁止对转储设备执行 `dump transaction`。如果发出 `dump transaction` 语句，则会产生一条错误消息，指出应使用 `dump database`。

`trunc log on chkpt` 在新建的数据库中处于关闭状态。若要更改该缺省设置，请在 `model` 数据库中打开此选项。

警告！ 如果在 `model` 中将 `trunc log on chkpt` 设置为打开，而且需要将一组数据库和事务日志装载到一个新创建的数据库中，则一定要在新数据库中关闭该选项。

- 缺省情况下禁用 `delayed commit` 选项。启用此选项时，所有本地事务都将使用延迟提交。也就是说，提交时控制将返回到客户端，而不等待日志页中的 I/O 完成，并且不在最后的日志缓冲区中为延迟的提交事务发出 I/O。当为数据库同时启用 `delayed commit` 和 `ALS` 选项时，将不使用延迟提交。
- 当 `unique auto_identity index` 选项设置为 `true` 时，它向新表添加具有唯一非聚簇索引的 `IDENTITY` 列。缺省情况下，`IDENTITY` 列为 10 位的数值数据类型，但可以使用 `size of auto identity column` 配置参数更改此缺省值。同 `auto identity` 一样，当您用 `select *` 语句选择了所有列时，`IDENTITY` 列是不可见的。若要检索它，必须在选择列表中明确提供此列的名称 `SYB_IDENTITY_COL`。

如果需要使用游标或具有非唯一索引的隔离级别 0 读取，请使用 `identity in nonunique index` 选项。

尽管可以在 `tempdb` 中将 `unique auto_identity index` 设置为 `true`，但此选项并不被承认或使用，并且在这里创建的临时表不会自动包括具有唯一索引的 `IDENTITY` 列。

`unique auto_identity index` 选项提供一种机制，可用来创建具有自动 `IDENTITY` 列的表，这种列具有可与可更新游标一起使用的唯一索引。表的唯一索引可确保在 `fetch` 后游标位于正确的行。（如果正在使用隔离级别 0 读取且需要使非唯一索引在内部逻辑上唯一，以便它们可以处理可更新的游标，可使用 `identity in nonunique index` 选项。）

在某些情况下，`unique auto_identity index` 选项可避免以下原因引起的 Halloween 问题：

- 用户不能更新 `IDENTITY` 列；因此它不能用于游标更新。
- 自动创建了具有唯一非聚簇索引的 `IDENTITY` 列，以便将其用于可更新的游标扫描中。

有关 Halloween 问题、`IDENTITY` 列和游标的详细信息，请参见《Transact-SQL 用户指南》中的第 18 章“游标：访问数据”以及《性能和调优系列：查询处理和抽象计划》中的第 8 章“游标优化”。

不要将 `unique auto_identity index` 选项与 `identity in nonunique index` 选项混淆，后者用于通过在表的索引键中包括 `IDENTITY` 列来使表中的所有索引具有唯一性。

权限 只有系统管理员或数据库所有者才能执行带有参数的 `sp_dboption` 来更改数据库选项。别名为数据库所有者的用户不能执行 `sp_dboption` 来更改数据库选项。任何用户都可以执行不带参数的 `sp_dboption` 来查看数据库选项。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关数据库选项的详细信息，请参见《系统管理指南》。

命令 [checkpoint](#)、[select](#)

系统过程 [sp_configure](#)、[sp_helpdb](#)、[sp_helpindex](#)、[sp_helpjoins](#)

实用程序 [bcp](#)

sp_dbrecovery_order

说明 指定用户数据库恢复的顺序，并列出一个数据库或所有数据库的用户定义的恢复顺序。

语法 sp_dbrecovery_order [*database_name* [, *rec_order* [, *force* [, *relax* | *strict*]]]]

参数

database_name

为其指派恢复顺序的数据库的名称或要列出其用户定义的恢复顺序的数据库的名称。

rec_order

恢复数据库的顺序。若 *rec_order* 为 -1，则将从用户定义的恢复序列中删除指定的数据库。

force

允许用户将数据库插入现有恢复序列中，而不将该数据库放在末尾。

relax

指定在恢复数据库时创建数据库（缺省值）。

缺省值是 *relax*，这意味着在完成恢复后数据库立即进入联机状态。

strict

指定按恢复顺序指定数据库。

示例

示例 1 使 *pubs2* 数据库成为在系统故障后恢复的第一个用户数据库：

```
sp_dbrecovery_order pubs2, 1
```

示例 2 将 *pubs3* 数据库插入用户定义的恢复序列中的第三个位置。如果另一个数据库原先在第三个位置，则它将被移动到第四位，后面的所有数据库也相应向后移动：

```
sp_dbrecovery_order pubs3, 3, force
```

示例 3 从用户定义的恢复序列中删除 *pubs2* 数据库。这样，在恢复了具有用户指定的恢复顺序的所有数据库后才会恢复 *pubs2*：

```
sp_dbrecovery_order pubs2, -1
```

示例 4 列出具有恢复顺序（通过 *sp_dbrecovery_order* 指派）的所有数据库的当前恢复顺序：

```
sp_dbrecovery_order
```

用法

- 只有在 *master* 数据库中才能使用 *sp_dbrecovery_order* 来输入或修改用户指定的恢复顺序。可以从任何数据库列出用户定义的数据库恢复顺序。

- 若要更改数据库的用户定义的恢复位置，请使用 `sp_dbrecovery_order` 将该数据库从恢复序列中删除，然后使用 `sp_dbrecovery_order` 将其插入新位置。
- 系统数据库总是在用户数据库之前恢复。系统数据库和它们的恢复顺序是：
 - master
 - model
 - tempdb
 - sybsystemdb
 - sybsecurity
 - sybsystemprocs
- 如果没有通过 `sp_dbrecovery_order` 为任何数据库指派恢复顺序，则先恢复系统数据库，然后按用户数据库的 ID 对所有用户数据库进行恢复。
- 如果 `database_name`:
 - 已指定但没有给出 `rec_order` — `sp_dbrecovery_order` 将显示指定数据库的用户定义的恢复位置。
 - 未指定 — `sp_dbrecovery_order` 将列出具有用户指派的恢复顺序的所有数据库的恢复顺序。
- 通过 `sp_dbrecovery_order` 指派的恢复顺序必须是连续的，从 1 开始，值之间不存在间隔。必须将指派了恢复顺序的第一个数据库的 `rec_order` 指派为 1。如果有三个数据库的恢复顺序分别被指派为 1、2 和 3，则不能将下一个数据库的恢复顺序指派为 5。

权限

只有系统管理员才能执行 `sp_dbrecovery_order`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

sp_dbremap

说明 强制 Adaptive Server 识别 `alter database` 所进行的更改。只有在 Adaptive Server 消息要求这样做时才运行此过程。

语法 `sp_dbremap dbname`

参数 `dbname`

是 `alter database` 命令在其中被中断的数据库的名称。

示例 `alter database` 命令更改了数据库 `sample_db`。该命令进行了对 Adaptive Server 可见的更改：

```
sp_dbremap sample_db
```

用法

- 如果在处于转储过程中的数据库上发出的 `alter database` 语句被中断，则 Adaptive Server 将输出一条消息，指示用户执行 `sp_dbremap`。
- 在数据库或事务转储过程中对 `sysusages` 进行的任何更改在转储完成之前不会被复制到活动内存中，这样可以确保数据库映射在转储过程中不发生更改。运行 `alter database` 将立即对磁盘上的系统表进行更改。在转储完成之前，不能更改内存中的分配。这就是 `alter database` 暂停的原因。

当执行 `sp_dbremap` 时，它必须等到转储进程完成。

- 如果指示您运行 `sp_dbremap`，但是您没有这样做，则在下一次重新启动前，Adaptive Server 不能使用您通过 `alter database` 分配的空间。

权限 只有系统管理员或数据库所有者才能执行 `sp_dbremap`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** `alter database`、`dump database`、`dump transaction`

sp_defaultloc

| | |
|----|---|
| 说明 | (仅限组件集成服务) 为本地数据库中的对象定义缺省存储位置。 |
| 语法 | <code>sp_defaultloc dbname, defaultloc, defaulttype</code> |
| 参数 | <p>dbname 是要映射到远程存储位置的数据库的名称。必须已通过 create database 语句对该数据库进行了定义。不能将系统数据库映射到远程位置。</p> <p>defaultloc 是要将数据库映射到的远程存储位置。若要指示服务器删除某一数据库现有的缺省映射, 请将此参数设置为 NULL。defaultloc 的值必须以句点 (.) 结束, 如下所示:</p> <pre>server.dbname.owner.</pre> <p>defaulttype 是指定对象格式的值之一, 而对象是由 object_loc 指定的。有效值如下所示。用引号将 defaulttype 值引起来:</p> <ul style="list-style-type: none"> • table — 指示由 object_loc 指定的对象是远程服务器可以访问的表。该值是 defaulttype 的缺省值。 • view — 指示由 object_loc 指定的对象是由远程服务器管理并被作为表来处理的视图。 • rpc — 指示由 object_loc 指定的对象是由远程服务器管理的 RPC ; RPC 的结果集将作为只读表处理。 |
| 示例 | <p>示例 1 <code>sp_defaultloc</code> 在名为 SYBASE 的远程服务器中定义远程存储位置 <code>pubs.dbo</code>。它将数据库 <code>pubs</code> 映射到该远程位置。<code>create table book1</code> 语句将在远程位置上创建一个名为 <code>book1</code> 的表。用于 <code>bookN</code> 的 create existing table 语句将要求 <code>pubs.dbo.bookN</code> 已存在于远程位置, 并且有关表 <code>bookN</code> 的信息要存储在本地表 <code>bookN</code> 中:</p> <pre>sp_defaultloc pubs, SYBASE.pubs.dbo., table create table pubs.dbo.book1 (bridges char(15))</pre> <p>示例 2 删除数据库 <code>pubs</code> 到远程位置的映射:</p> <pre>sp_defaultloc pubs, NULL</pre> |

示例 3 标识出远程存储位置 `wallst.nasdaq.dbo`，其中“`wallst`”是为 `server_name` 提供的值，“`nasdaq`”是为 `database` 提供的，“`dbo`”是为 `owner` 提供的。RPC `sybase` 必须已存在于远程位置。`create existing table sybase` 语句将有关 RPC `sybase` 结果集的信息存储到本地表 `ticktape` 中。RPC `sybase` 的结果集被认为是只读表。对于 RPC，不支持插入、更新和删除：

```
sp_defaultloc ticktape, wallst.nasdaq.dbo., rpc
create existing table sybase (bestbuy integer)
```

用法

- `sp_defaultloc` 为本地数据库中的表定义一个缺省存储位置。它将数据库中的表名映射到远程位置。它允许用户为整个数据库建立一个缺省值，而不是在每个 `create table` 和 `create existing table` 命令之前发出一个 `sp_addobjectdef` 命令。

- 当 `defaulttype` 是 `table`、`view` 或 `rpc` 时，`defaultloc` 参数将采取以下形式：

```
server_name.dbname.owner.
```

- 请注意，`defaultloc` 说明必须以句点 (.) 结束。
- `server_name` 表示已经由 `sp_addserver` 添加到 `syssservers` 中的服务器。`server_name` 参数是必需的。
- `dbname` 不一定是必需的。某些服务器类不支持它。
- `owner` 一应始终提供它来避免出现歧义。如果未提供，则实际引用的远程对象可能会不同，这取决于外部登录是否与远程对象所有者对应。
- 在任何一个 `create table` 或 `create existing table` 语句之前发出 `sp_defaultloc`。无论使用了这两个语句中的哪一个，服务器都将使用 `sysattributes` 表来确定是否已经为将要创建或定义的对象指定了任何表映射。如果已指定了映射，则 `create table` 语句指示在 `object_loc` 指定的位置创建该表。`create existing table` 语句将有关现有远程对象的信息存储在本地表中。
- 如果在 `defaulttype view` 上发出了 `sp_defaultloc`，然后又发出了 `create table`，则组件集成服务将在远程服务器上创建一个新表，而不是视图。
- 更改数据库的缺省位置不会影响先前已经映射到不同缺省位置的表。
- 在数据库中创建表之后，随后对 `dbname` 中表的所有引用（通过 `select`、`insert`、`delete` 和 `update`）都将映射到正确的位置。

权限

任何用户都可以执行 `sp_defaultloc`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create existing table](#)、[create table](#)

系统过程 [sp_addobjectdef](#)、[sp_addserver](#)、[sp_helpserver](#)

sp_deletesmobj

说明 从 IBM Tivoli Storage Manager (TSM) 中删除指定的备份对象。

注释 仅在您的站点已获得 TSM 许可时才支持 `sp_deletesmobj`。

语法 `sp_deletesmobj "syb_tsm", "server_name"{, "database_name", "object_type", "dump_type", "until_time", "bs_name"}`

参数 `syb_tsm`
是调用 `libsyb_tsm.so` 模块的关键字，该模块用于实现与 TSM 的通信。

server_name
是与要删除的 TSM 备份对象关联的 Adaptive Server 的名称。

database_name
是与要删除的 TSM 备份对象关联的数据库的名称。星号 (*) 表示所有数据库。

object_name
是 `dump database` 或 `dump transaction` 命令中提供的 TSM 备份对象的名称。如果省略此参数，则会删除所有备份对象。星号 (*) 表示所有备份对象。

dump_type
是要删除的备份对象类型。其值包括：

- DB — 通过 `dump database` 命令创建的数据库备份对象。
- XACT — 通过 `dump transaction` 命令创建的数据库备份对象。
- * — 所有数据库备份对象。这是缺省值。

until_time
是日期时间戳字段。将删除符合指定标准且创建日期早于 `until_time` 的所有备份对象。

bs_name
是远程 Backup Server 的名称。如果省略 `bs_name`，则会使用缺省的 `SYB_BACKUP`。

示例 1 删除在 Adaptive Server “svr1” 中创建的所有 TSM 备份对象。

```
sp_deletesmobj "syb_tsm", "svr1"
```

示例 2 删除由 “svr1” 在 2009 年 5 月 20 日上午 10:51:43:866 之前为 testdb 数据库创建的所有备份对象。备份对象名为 “obj1.dmp”。

```
sp_deletesmobj "syb_tsm", "svr1", "testdb", "obj1.dmp",
               "*", "may 20, 2009 10:51:43:866am"
```

示例 3 中的语法是否正确？

示例 3 删除由 “svr1” 在 2009 年 5 月 21 日上午 10:51:43:866 之前为 “testdb” 数据库创建的 dump database 类型的所有备份对象。

```
sp_deletesmobj "syb_tsm", "svr1", "'testdb", "*", "DB",
               "may 21, 2009 10:51:43:866am"
```

示例 4 删除由 “svr1” 在 2009 年 5 月 21 日上午 10:51:43:866 之前为 “testdb” 数据库创建的 dump transaction 类型的所有备份对象。

```
sp_deletesmobj "syb_tsm", "svr1", "testdb",
               "/tmp/obj1.dmp", "*", "XACT", "may 21, 2009
               10:51:43:866am"
```

用法 有关 TSM 的 Sybase 支持的详细信息，请参见 THE TSM BOOK (《TSM 手册》)。

权限 只有系统管理员和具有操作员角色的用户才能执行 sp_deletesmobj。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色。 关键字或选项 — NULL。 先前值 — NULL。 当前值 — NULL。 其它信息 — 所有输入参数。 代理信息 — set proxy 有效时的初始登录名。 |

另请参见 [sp_querysmobj](#)

sp_depends

说明

显示有关数据库对象依赖性的信息，这些对象是数据库中依赖于指定的表或视图的视图、触发器、用户定义的函数和过程，以及数据库中指定的视图、触发器或过程所依赖的表和视图。

还显示有关表列依赖性的信息，如果提供了 *column_name*，则这些对象是在指定的任一列中定义的索引、缺省值、检查约束、规则和参照完整性约束；如果未提供 *column_name*，则这些对象是在表中所有列上定义的索引、缺省值、检查约束、规则和参照完整性约束。

语法

```
sp_depends objname[, column_name]
```

参数

objname

是要对其进行依赖性检查的表、视图、Transact-SQL 存储过程、SQLJ 存储过程、SQLJ 函数或触发器的名称。您不能指定数据库名。如果对对象所有者既不是运行该命令的用户也不是数据库所有者，则使用所有者名。

column_name

是要对其进行依赖性检查的列的名称。

示例

示例 1 列出依赖于表 *sysobjects* 的数据库对象：

```
sp_depends sysobjects
```

示例 2 列出依赖于 *titleview* 视图的数据库对象，以及 *titleview* 视图所依赖的数据库对象：

```
sp_depends titleview
```

```
Things that the object references in the current
database.
```

| object | type | updated | selected |
|-----------------|------------|---------|----------|
| dbo.authors | user table | no | no |
| dbo.titleauthor | user table | no | no |
| dbo.titles | user table | no | no |

```
Things inside the current database that reference the
object.
```

| object | type |
|------------|------|
| dbo.tview2 | view |

示例 3 列出依赖于用户 “mary” 所拥有的 *titles* 表的数据库对象。因为句点是特殊字符，所以需要加引号：

```
sp_depends "mary.titles"
```

示例 4 显示 sysobjects 表的所有列的列级依赖性:

```
sp_depends sysobjects
```

Things inside the current database that reference the object.

| object | type |
|------------------|------------------|
| dbo.sp_dbupgrade | stored procedure |
| dbo.sp_procxmode | stored procedure |

Dependent objects that reference all columns in the table. Use sp_depends on each column to get more information.

Columns referenced in stored procedures, views or triggers are not included in this report.

| Column | Type | Object Names or Column Names |
|----------|------------|--|
| cache | permission | column permission |
| ckfirst | permission | column permission |
| crdate | permission | column permission |
| deltrig | permission | column permission |
| expdate | permission | column permission |
| id | index | sysobjects (id) |
| id | logical RI | From syscolumns (id) To sysobjects (id) |
| id | logical RI | From syscomments (id) To sysobjects (id) |
| id | logical RI | From sysdepends (id) To sysobjects (id) |
| id | logical RI | From sysindexes (id) To sysobjects (id) |
| id | logical RI | From syskeys (depid) To sysobjects (id) |
| id | logical RI | From syskeys (id) To sysobjects (id) |
| id | logical RI | From sysobjects (id) To sysprocedures (id) |
| id | logical RI | From sysobjects (id) To sysprotects (id) |
| id | logical RI | sysobjects (id) |
| id | permission | column permission |
| indexdel | permission | column permission |
| instrig | permission | column permission |
| loginame | permission | column permission |
| name | index | ncsysobjects (name, uid) |
| name | permission | column permission |
| objspare | permission | column permission |
| schemact | permission | column permission |
| seltrig | permission | column permission |
| sysstat | permission | column permission |
| sysstat2 | permission | column permission |
| type | permission | column permission |
| uid | index | ncsysobjects (name, uid) |
| uid | logical RI | From sysobjects (uid) To sysusers (uid) |
| uid | permission | column permission |

```

updtrig      permission  column permission
userstat     permission  column permission
versionts    permission  column permission

```

示例 5 显示有关 `sysobjects` 表中 `id` 列的更详细的列级依赖性信息：

```
sp_depends sysobjects, id
```

Things inside the current database that reference the object.

```

object                                     type
-----                                     -
dbo.sp_dbupgrade                           stored procedure
dbo.sp_procxmode                            stored procedure

```

Dependent objects that reference column id.

Columns referenced in stored procedures, views or triggers are not included in this report.

| Type | Property | Object Names or Column Names Also see/Use command |
|------------|------------|--|
| index | index | sysobjects (id) sp_helpindex, drop index, |
| logical RI | primary | sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | foreign | From syskeys (id) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From syscolumns (id) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From sysdepends (id) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From sysindexes (id) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From syskeys (depid) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From syscomments (id) To sysobjects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From sysobjects (id) To sysprotects (id) sp_helpkey, sp_dropkey |
| logical RI | common | From sysobjects (id) To sysprocedures (id) sp_helpkey, sp_dropkey |
| permission | permission | column permission sp_helprotect, grant/revoke |

示例 6 显示用户创建的表 **titles** 的所有列的列级依赖性:

```
1> sp_depends titles
```

Things inside the current database that reference the object.

| object | type |
|---------------------|------------------|
| dbo.deltitle | trigger |
| dbo.history_proc | stored procedure |
| dbo.title_proc | stored procedure |
| dbo.titleid_proc | stored procedure |
| dbo.titleview | view |
| dbo.totalsales_trig | trigger |

Dependent objects that reference all columns in the table. Use sp_depends on each column to get more information.

Columns referenced in stored procedures, views or triggers are not included in this report.

| Column | Type | Object Names or Column Names |
|----------|------------|--|
| pub_id | logical RI | From titles (pub_id) To publishers (pub_id) |
| pubdate | default | datedflt |
| title | index | titleind (title) |
| title | statistics | (title) |
| title_id | index | titleidind (title_id) |
| title_id | logical RI | From roysched (title_id) To titles (title_id) |
| title_id | logical RI | From salesdetail (title_id) To titles (title_id) |
| title_id | logical RI | From titleauthor (title_id) To titles (title_id) |
| title_id | logical RI | titles (title_id) |
| title_id | rule | title_idrule |
| title_id | statistics | (title_id) |
| type | default | typedflt |

示例 7 显示用户创建的表 **titles** 的 **pub_id** 列的更详细的列级依赖性信息:

```
sp_depends titles, pub_id
```

Things inside the current database that reference the object.

| object | type |
|---------------------|------------------|
| dbo.deltitle | trigger |
| dbo.history_proc | stored procedure |
| dbo.title_proc | stored procedure |
| dbo.titleid_proc | stored procedure |
| dbo.titleview | view |
| dbo.totalsales_trig | trigger |

Dependent objects that reference column `pub_id`.

Columns referenced in stored procedures, views or triggers are not included in this report.

| Type | Property | Object Names or Column Names Also see/Use command |
|------------|----------|--|
| ----- | ----- | ----- |
| logical RI | foreign | From titles (<code>pub_id</code>) To publishers (<code>pub_id</code>) <code>sp_helpkey</code> , <code>sp_dropkey</code> |

用法

- 执行 `sp_depends` 将列出当前数据库中依赖于 *objname* 以及 *objname* 所依赖的所有对象。例如，视图可以依赖于一个或多个表，同时过程或其它视图又可以依赖于这些视图。如果某个对象引用了另一个对象，则前者将依赖于后者。将不报告对当前数据库之外对象的引用。
- 在修改或删除列之前，请使用 `sp_depends` 确定表是否包含可能受该修改影响的任何依赖对象。例如，如果修改列使用新数据类型，则可能需要重新定义与该表密切联系的对象，以与该列的新数据类型保持一致。
- `sp_depends` 过程通过查看 `sysdepends` 表确定依赖性。
如果不按顺序创建对象（例如，如果在创建视图之前就创建了使用该视图的过程），那么在 `sysdepends` 中就不存在有关依赖性的行，而且 `sp_depends` 也不报告这些依赖性。
- 如果所报告的对象是存储过程或触发器，则 `sp_depends` 产生的报告中的 `updated` 和 `selected` 列是有意义的。`updated` 列的值指示存储过程是否更新该对象。`selected` 列指示该对象是否用于读游标或数据修改语句。
- `sp_depends` 遵循这些查找对象的 Adaptive Server 规则：
 - 如果用户没有指定所有者名，并且执行该命令的用户拥有具有指定名称的对象，则使用该对象。
 - 如果用户没有指定所有者名，并且用户不拥有具有该名称的对象，但数据库所有者却拥有这样的对象，则使用数据库所有者的对象。
 - 如果用户和数据库所有者都不拥有具有该名称的对象，则即使数据库中存在具有此名称但属于其它所有者的对象，该命令也会报告一个错误状态。
 - 如果用户和数据库所有者都拥有具有指定名称的对象，并且用户想访问数据库所有者的对象，则必须指定名称，例如 *dbo.objectname*。

- 数据库用户（而不是执行该命令的用户）和数据库所有者拥有的对象必须用所有者的名称进行限定，如示例 3 中所示。
- SQLJ 函数和 SQLJ 存储过程是包装在 SQL 包装中的 Java 方法。有关详细信息，请参见《Adaptive Server Enterprise 中的 Java》。
 - SQLJ 函数和 SQLJ 存储过程是可列出依赖性的数据库对象。SQLJ 存储过程和 SQLJ 函数的唯一依赖性 is Java 类。
 - 如果 *objname* 是 SQLJ 存储过程或 SQLJ 函数，则 `sp_depends` 可列出创建语句中声明的例程外部名称中的 Java 类，而不是作为返回类型指定的类或参数列表中的数据类型。
 - SQLJ 存储过程和 SQLJ 函数可作为其它数据库对象的依赖性列出。

权限 任何用户都可以执行 `sp_depends`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** [create procedure](#)、[create table](#)、[create view](#)、[execute](#)
系统过程 [sp_help](#)

sp_deviceattr

说明 (仅限 UNIX 平台) 更改现有数据库设备文件的设备参数设置。

语法 `sp_deviceattr logicalname, optname, optvalue`

参数

logicalname

是现有数据库设备的逻辑名称。该设备既可存储在操作系统文件上，也可以存储在原始分区上，但如果存储在原始分区上，`dsync` 设置将被忽略。

optname

要更改的属性的名称。有效值为 `directio` 或 `dsync`：

- `directio` — 允许 Adaptive Server 跳过操作系统的缓冲系统，直接将数据写入磁盘。
- `dsync` — 允许对设备的更新直接在存储介质上进行，或者由 UNIX 文件系统进行缓冲。

注释 `directio` 和 `dsync` 选项是互斥的；不能同时为两者指定“true”。

optvalue

可以是“true”或“false”。

示例

将名为“file_device1”的设备的 `dsync` 设置为打开：

```
sp_deviceattr file_device1, dsync, true
```

用法

- 对于存储在 UNIX 文件上的数据库设备，`dsync` 确定对设备的更新是直接在存储介质上进行还是由 UNIX 文件系统进行缓冲。
当 `dsync` 打开时，对数据库设备的写入可以直接在物理存储介质上进行，并且 Adaptive Server 可以在系统出现故障时恢复设备上的数据。
当 `dsync` 关闭时，UNIX 文件系统可能会对数据库设备的写入进行缓冲。即使尚未修改物理介质，UNIX 文件系统仍可能将更新标记为已经完成。在系统出现故障时，不能保证物理介质上已进行了请求的数据更新，并且 Adaptive Server 可能无法恢复数据库。
- 在使用 `sp_deviceattr` 更改 `dsync` 或 `directio` 设置之后，您必须重新启动 Adaptive Server 才能使更改生效。
- 如果为文件系统上的数据库设备禁用了 `dsync` 选项，`sp_deviceattr` 将显示一条警告消息。
`directio` 设置的更改必须在重新启动 Adaptive Server 后才能生效。

- 对于主设备文件，`dsync` 始终处于打开状态。不能用 `sp_deviceattr` 更改主设备文件的 `dsync` 设置。因此，不能为主设备设置 `directio` 选项。
- 仅在出现系统故障且无需恢复设备上的数据库时，才能关闭 `dsync` 值。例如，对于只存储 `tempdb` 数据库的设备，可以考虑将 `dsync` 关闭。
- Adaptive Server 将忽略存储在原始分区上的设备的 `dsync` 设置；不管 `dsync` 设置如何，都不会缓冲对这些设备的更新。
- `dsync` 不能在 Windows NT 平台上使用。

权限

执行 `sp_deviceattr` 的用户必须具有更新 `sysdevices` 表的权限。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_helpdevice](#)

sp_diskdefault

| | |
|----|---|
| 说明 | 在用户未指定数据库设备或通过 create database 或 alter database 命令指定 <code>default</code> 的情况下，指定数据库设备是否可用于数据库存储。 |
| 语法 | <code>sp_diskdefault logicalname, {defaulton defaultoff}</code> |
| 参数 | <p><i>logicalname</i></p> <p>是 <code>master.dbo.sysdevices.name</code> 中给出的设备的逻辑名。该设备必须是一个数据库设备，而不能是转储设备。</p> <p><code>defaulton defaultoff</code></p> <p><code>defaulton</code> 指定该数据库设备为缺省数据库设备；<code>defaultoff</code> 指定所指定的数据库设备不是缺省数据库设备。</p> <p>在用 disk init 将数据库设备添加到系统之后，使用 <code>defaulton</code>。使用 <code>defaultoff</code> 更改主设备（在首次安装 Adaptive Server 时被指定为缺省设备的设备）的缺省状态。</p> |
| 示例 | 对于数据库的缺省存储，主设备不能再被 create database 或 alter database 使用： |
| 用法 | <pre>sp_diskdefault master, defaultoff</pre> <ul style="list-style-type: none">缺省数据库设备是指在用户没有指定数据库设备名或指定了关键字 <code>default</code> 的情况下，create database 或 alter database 用于数据库存储的设备。可以有多个缺省设备。将按照这些缺省设备在 <code>master.dbo.sysdevices</code> 表中出现的顺序（即字母顺序）使用它们。当第一个缺省设备被填满后，将使用第二个缺省设备，依此类推。当首次安装 Adaptive Server 时，主设备是唯一的缺省数据库设备。 <hr/> <p>注释 初始化设备来存储用户数据库后，请使用 <code>sp_diskdefault</code> 关闭主设备的缺省状态。这可以防止用户意外地在主设备上创建数据库，并简化 <code>master</code> 数据库的恢复。</p> <hr/> <ul style="list-style-type: none">若要找出哪些数据库设备是缺省数据库设备，请执行 sp_helpdevice。 |
| 权限 | 只有系统管理员才能执行 <code>sp_diskdefault</code> 。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [alter database](#)、[create database](#)、[disk init](#)
系统过程 [sp_helpdevice](#)

sp_displayaudit

- 说明** 显示审计选项的状态。
- 语法** `sp_displayaudit ["procedure" | "object" | "login" | "database" | "global" | "default_object" | "default_procedure" [, "name"]]`
- 参数**
- procedure**
显示指定的存储过程或触发器的审计选项的状态。如果未指定 *name* 的值，`sp_displayaudit` 将显示当前数据库中所有过程和触发器的活动审计选项。
- object**
显示指定的表或视图的审计选项的状态。如果未指定 *name* 的值，`sp_displayaudit` 将显示当前数据库中所有表和视图的活动审计选项。
- login**
显示指定的用户登录的审计选项的状态。如果未指定 *name* 的值，`sp_displayaudit` 将显示 `master` 数据库中所有登录的活动审计选项。
- database**
显示指定数据库的审计选项的状态。如果未指定 *name* 的值，`sp_displayaudit` 将显示服务器上所有数据库的活动审计选项。
- global**
显示指定的全局审计选项的状态。如果未指定 *name* 的值，`sp_displayaudit` 将显示当前数据库中所有过程和触发器的活动审计选项。
- default_object**
显示将用于在指定的数据库上创建的任意新表或视图的缺省审计选项。如果未指定 *name* 的值，`sp_displayaudit` 将显示所有具有活动缺省审计设置的数据库的缺省审计选项。
- default_procedure**
显示将用于在指定的数据库上创建的任意新过程或触发器的缺省审计选项。如果未指定 *name* 的值，`sp_displayaudit` 将显示所有具有活动缺省审计设置的数据库的缺省审计选项。
- name**
是指定参数的信息，如下表中的说明：

| 参数 | <i>name</i> 的值 |
|-----------|----------------|
| procedure | 过程或触发器名 |
| object | 表或视图名 |
| login | 用户登录名 |
| database | 数据库名称 |

| 参数 | name 的值 |
|-------------------|---------|
| global | 全局审计选项 |
| default_object | 数据库名称 |
| default_procedure | 数据库名称 |

示例 1 当未指定参数时，将显示每个类别的状态和所有审计选项：

```
sp_displayaudit

Procedure/Trigger      Audit Option  Value Database
-----
dbo.sp_altermessage    exec_procedure on   sybsystemprocs
dbo.sp_help            exec_procedure on   sybsystemprocs
dbo.sp_who             exec_procedure on   sybsystemprocs
No databases currently have default sproc/trigger auditing enabled.
No objects currently have auditing enabled.
No databases currently have default table/view auditing enabled.
No logins currently have auditing enabled.
No databases currently have auditing enabled.
```

```
Option Name              Value
-----
adhoc                    off
dbcc                     off
disk                     off
errors                   off
login                    off
logout                   off
keycustodian_role       off
navigator_role           off
oper_role                off
replication_role        off
rpc                      off
sa_role                  off
security                 off
sso_role                 off
```

示例 2 当未指定过程名时，将显示所有过程审计选项的状态：

```
sp_displayaudit "procedure"

Procedure/Trigger      Audit Option  Value Database
-----
dbo.sp_altermessage    exec_procedure on   sybsystemprocs
dbo.sp_help            exec_procedure on   sybsystemprocs
dbo.sp_who             exec_procedure on   sybsystemprocs
```

示例 3 当指定了过程名时，只显示该过程的状态：

```
sp_displayaudit "procedure", "sp_who"

Procedure/Trigger Audit Option      Value Database
-----
dbo.sp_who          exec_procedure on      sybsystemprocs
```

示例 4 当未指定全局审计选项时，将显示所有全局审计选项的状态：

```
sp_displayaudit "global"

Option Name                                Value
-----
adhoc                                       off
dbcc                                       off
disk                                       off
errors                                     off
login                                       off
logout                                     off
keycustodian_role                         off
navigator_role                            off
oper_role                                  off
replication_role                          off
rpc                                        off
sa_role                                    off
security                                   off
sso_role                                   off
```

用法

- sp_displayaudit 显示审计选项的状态。
- 下表显示每个参数的有效审计选项：

| 对象类型参数 | 有效审计选项 |
|-------------------|---|
| procedure | exec_procedure、exec_trigger |
| object | delete、func_obj_access、insert、reference、select、update |
| login | all、cmdtext、table_access、view_access |
| database | alter、bcp、bind、create、dbaccess、drop、dump、encryption_key、func_dbaccess、grant、load、revoke、setuser、truncate、unbind |
| global | adhoc、dbcc、disk、errors、login、logout、navigator_role、oper_role、replication_role、rpc、keycustodian_role、sa_role、security、sso_role |
| default_object | delete、func_obj_access、insert、reference、select、update |
| default_procedure | exec_procedure、exec_trigger |

- 除非先指定了对象类型参数，否则不能指定名称的值。

权限 只有系统安全人员才能执行 sp_displayaudit。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **文档** 有关设置审计的信息，请参见《系统管理指南》。

系统过程 [sp_audit](#)

实用程序 bcp

sp_displaylevel

| | |
|----|---|
| 说明 | 设置或显示哪些 Adaptive Server 配置参数显示在 <code>sp_configure</code> 输出中。 |
| 语法 | <code>sp_displaylevel [loginame [, level]]</code> |
| 参数 | <p><i>loginame</i></p> <p>是要为之设置或显示显示级别的用户的 Adaptive Server 登录名。</p> <p><i>level</i></p> <p>将显示级别设置为以下之一：</p> <ul style="list-style-type: none">• “basic” 显示级别只显示最基础的配置参数。此级别适用于最一般的服务器调优。• “intermediate” 显示级别除显示 “basic” 级别显示的参数之外，还显示稍微复杂一些的配置参数。此级别适用于中等复杂程度的服务器调优。• “comprehensive” 显示级别显示所有配置参数，包括最复杂的配置参数。此级别适用于高精度的服务器调优。 |
| 示例 | <p>示例 1 显示调用 <code>sp_displaylevel</code> 的用户的当前显示级别：</p> <pre>sp_displaylevel The current display level for login 'sa' is 'comprehensive'.</pre> <p>示例 2 显示用户 “jerry” 的当前显示级别：</p> <pre>sp_displaylevel jerry The current display level for login 'jerry' is 'intermediate'.</pre> <p>示例 3 将用户 “jerry” 的显示级别设置为 “comprehensive”：</p> <pre>sp_displaylevel jerry, comprehensive The display level for login 'jerry' has been changed to 'comprehensive'.</pre> |
| 用法 | 有关显示级别和配置参数的详细信息，请参见《系统管理指南》。 |
| 权限 | 只有系统管理员才能执行 <code>sp_displaylevel</code> 为另一个用户设置显示级别。任何用户都可以执行 <code>sp_displaylevel</code> 来设置并显示自己的显示级别。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_configure](#)

sp_displaylogin

- 说明** 显示有关登录帐户的信息。通过使用通配符表达式 (%), 还可以获取有关匹配登录名的信息。同时还会显示存储在磁盘上的登录口令的加密版本。
- 语法** `sp_displaylogin ['user_id' | '[loginame | wildcard]'`
- 参数**
- user_id*
是服务器用户 ID。
- loginame*
是需要有关其信息的用户登录帐户。您必须是系统安全员或系统管理员才能获得有关其他人的登录帐户的信息。
- wildcard*
是用于获取有关登录帐户的信息的通配符表达式。
- 示例 1** 显示有关您的服务器登录帐户的信息。口令有效期设置为“0”，指示口令永不到期。

```

1> sp_displaylogin 'sa'
2> go

Suid: 121
Loginame: sa
Fullname:
Configured Authorization:
    sa_role (default ON)
    ssO_role (default ON)
    oper_role (default ON)
    sybase_ts_role (default ON)
Locked: NO
Date of Last Password Change: Aug 10 2010 11:17AM
Password expiration interval: 0
Password expired: NO
Minimum password length: 6
Maximum failed logins: 0
Current failed login attempts:
Login password encryption: SYB-PROP, SHA-256
Last login date : Aug 17 2010 5:55PM
Login Profile :emp_lp

```

示例 2 显示有关登录帐户“susanne”的信息。显示的信息会因执行 `sp_displaylogin` 的用户角色而异。此处没有为用户“susanne”设置口令有效期，因此口令不会到期。

```

sp_displaylogin susanne

Suid: 12

```

```
Loginame: susanne
Fullname:
Configured Authorization:
    supervisor (default OFF)
Locked: NO
Date of Last Password Change: July 26 2010 10:42AM
Login Profile :emp_lp
```

示例 3 显示为登录名配置的与登录安全性相关的参数，以及指定的鉴定机制。口令在 2007 年 11 月 29 日下午 3:46 更改，并于 5 天后（即 2007 年 12 月 5 日下午 3:46）到期。

```
sp_displaylogin joe

Suid: 294
Loginame: joe
Fullname: Joseph Resu
Configured Authorization:
    intern_role (default OFF)
Locked: NO
Date of Last Password Change: Nov 24 2010 3:46PM
Password expiration interval : 5
Password expired : NO
Minimum password length:4
Maximum failed logins : 10
Current failed logins : 3
Login password encryption: SHA-256
Login Profile :emp_lp
```

示例 4 显示有关服务器用户 ID 为 1 的登录帐户的信息。

```
sp_displaylogin '1'
-----
Suid: 1
Loginame: sa
Fullname:
Configured Authorization:
    sa_role (default ON)
    sso_role (default ON)
    oper_role (default ON)
    sybase_ts_role (default ON)
Locked: NO
Date of Last Password Change: Dec 18 2010
Password expiration interval: 0
Login Profile :emp_lp
```

示例 5 可以使用通配符指示您自己的服务器登录帐户之外的任何服务器登录帐户。

```
sp_displaylogin '%'
-----

Suid  Loginname  Fullname  Locked  Date of Last Password Change
Password expiration interval  Password expired  Minimum password length  Maximum
failed logins  Current failed login attempts  Authenticate with  Login
Profile  Configured Authorization

-----
-----
-----
-----

2 probe NULL sybssystemdb NULL NULL NO Jan  8 2010 7:13AM 1 NO 6 0 0 NONE
NULL
1 sa NULL master NULL NULL NO Jan  8 2010 6:46AM 1 NO 6 0 0 NONE
```

示例 6 磁盘上的登录口令同时使用旧的 Sybase 专有加密算法和 SHA-256 算法以加密形式存储：

```
1> sp_displaylogin 'mylogin'
2> go

Suid: 121
Loginname: mylogin
Fullname:
Configured Authorization:
    sa_role (default ON)
    sso_role (default ON)
    oper_role (default ON)
    sybase_ts_role (default ON)
Locked: NO
Date of Last Password Change: Aug 10 2006 11:17AM
Password expiration interval: 0
Password expired: NO
Minimum password length: 6
Maximum failed logins: 0
Current failed login attempts:
Login password encryption: SYB-PROP, SHA-256
Last login date : Aug 17 2010 5:55PM
Login Profile :emp_lp

(return status = 0)
```

当登录口令仅使用 SHA-256 算法存储在磁盘上时，sp_displaylogin 的输出就会包含行 “Login password encryption: SHA-256”：

```
1> sp_displaylogin 'mylogin'
2> go

Suid: 121
Loginame: mylogin
...
Authenticate with: NONE
Login password encryption: SHA-256
Last login date : Aug 17 2010 5:55PM
Login Profile :emp_lp

(return status = 0)
```

如果从 Adaptive Server 15.0.2 版之前的版本升级后尚未登录，则仍使用旧的加密算法，且 sp_displaylogin 的输出会包含行 “Login password encryption: SYB-PROP”：

```
1> sp_displaylogin 'mylogin'
2> go

Suid: 121
Loginame: mylogin
...
Authenticate with: NONE
Login password encryption: SYB-PROP
Last login date : Aug 17 2006 5:55PM
(return status = 0)
```

如果登录被锁定，sp_displaylogin 会显示锁定帐号的日期、原因和登录。还会显示 lastlogindate 值：

```
1> sp_displaylogin 'mylogin'
2> go

Suid: 121
Loginame: mylogin
Fullname:
Configured Authorization:
    sa_role (default ON)
    sso_role (default ON)
    oper_role (default ON)
    sybase_ts_role (default ON)
Locked: YES
    Date when locked: Aug 18 2010 9:15AM
    Reason: Account locked by Adaptive Server due
to failed login attempts reaching max failed logins.
```

```

        Locking suid: mylogin
Date of Last Password Change: Aug 10 2010 11:17AM
Password expiration interval: 0
Password expired: NO
Minimum password length: 6
Maximum failed logins: 3
Current failed login attempts: 3
Login password encryption: SYB-PROP, SHA-256
Last login date : Aug 17 2010 5:55PM
Login Profile :emp_lp
(return status = 0)

```

示例 7 显示用于登录名的加密版本；此输出的最后一行包含有关 Adaptive Server 对磁盘上的登录口令使用的加密方式的信息：

```

sp_displaylogin sa
go
Suid: 1
Loginame: sa
Fullname:
Configured Authorization:
    sa_role (default ON)
    sso_role (default ON)
    oper_role (default ON)
    sybase_ts_role (default ON)
Locked: NO
Date of Last Password Change: Mar  8 2010 3:04PM
Password expiration interval: 0
Password expired: NO
Minimum password length: 6
Maximum failed logins: 0
Current failed login attempts:
Login Password Encryption: SHA-256
Login Profile :emp_lp

```

如果 Adaptive Server 在降级期间使用 Adaptive Server 15.0.2 版的早期版本或当前版本中的加密算法，`sp_displaylogin` 将显示早期的 Sybase 专有加密算法和新算法 SHA-256：

```

Login password encryption: SYB-PROP, SHA-256

```

示例 8 显示当前登录帐户的登录名和口令策略选项：

```

sp_displaylogin
go
Suid: 5
Loginame: tammi
Fullname:
Configured Authorization:

```

```
sa_role (default ON)
sso_role (default ON)
oper_role (default ON)
sybase_ts_role (default ON)
Locked: NO
Date of Last Password Change: Mar  8 2010 3:04PM
Password expiration interval: 0
Password expired: NO
Minimum password length: 6
Maximum failed logins: 0
Current failed login attempts:
Authenticate with: ANY
Login Password Encryption: SHA-256
Exempt inactive lock: 0
Login Profile: emp_lp
```

示例 9 显示 `suid` 为 56 的用户的登录帐号：

```
sp_displaylogin '56'
```

显示登录名以 “st” 开头的所有用户的登录帐户信息：

```
sp_displaylogin 'st%'
```

用法

- 当显示与口令有效期、最大失败登录次数及口令长度有关的登录信息时，会将 `sp_passwordpolicy` 安全性选项考虑在内。
- `sp_displaylogin` 显示用于登录的加密版本。例如，如果口令降级阶段使用了新旧两种加密，`sp_displaylogin` 的输出就会包含新行 “Password encryption”。
- `sp_displaylogin` 显示配置的角色，因此即使通过 `set` 命令使某个角色处于不活动状态，仍会显示该角色。
- 与当前登录相关联的登录触发器是通过登录配置文件指定的。有关详细信息，请参见《系统管理指南》中的 “Managing login accounts and login profiles”（管理登录帐户和登录配置文件）。
- 当使用 `sp_displaylogin` 来获得有关您自己的帐户的信息时，无需使用 `loginame` 参数。`sp_displaylogin` 显示您的服务器用户 ID、登录名、全名、授予您的所有角色、上次更改口令的日期、缺省数据库、缺省语言以及您的帐户是否被锁定。
- 如果您是系统安全员或系统管理员，则可以使用 `loginame` 参数访问有关任何帐户的信息。

权限

只有系统管理员或系统安全员才能执行带有 `loginame` 或 `suid` 的 `sp_displaylogin` 来获得有关其他用户登录帐户的信息。任何用户都可以执行 `sp_displaylogin` 来获得有关自己的登录帐户的信息。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **存储过程** [sp_activeroles](#)、[sp_displayroles](#)、[sp_helprotect](#)

sp_displayroles

说明

显示授予另一个角色的所有角色、登录名或登录配置文件、角色的表格式的整个层次树，以及其它为指定角色配置的与登录安全性有关的参数，包括锁定角色的日期、锁定原因以及锁定该角色的登录服务器用户 ID (suid)。对于受口令保护的角色，还显示角色口令加密版本。

显示通过关联的登录配置文件向登录名授予的角色。输出中的 **grantee** 列显示登录配置文件名（如果适用）。该列仅在登录名具有与授予登录名的角色相关联的登录配置文件时才会显示。登录配置文件关联可以是直接的，也可以是通过缺省登录配置文件实现的。

语法

```
sp_displayroles [grantee_name [, mode]]
```

参数

grantee_name

是要获得的角色信息所对应的用户的登录名或登录配置文件名，或者是要获得的信息所对应的角色的名称。

mode

是以下值之一：

- **expand_up** — 显示父级的角色层次树
- **expand_down** — 显示子级的角色层次树
- **display_info** — 显示为指定角色配置的与登录安全性相关的参数

示例

示例 1 显示授予发出该命令的用户的所有角色：

```
sp_displayroles

Role Name
-----
supervisor_role
```

示例 2 显示授予 supervisor_role 的所有角色：

```
sp_displayroles "supervisor_role"

Role Name
-----
clerk
```

示例 3 显示授予登录名 “susanne” 的角色，以及层次中位于该角色之下的角色：

```
sp_displayroles susanne, expand_down

Role Name          Parent Role Name      Level
-----
supervisor_role    NULL                   1
clerk_role         supervisor_role       2
```

示例 4 显示授予 `intern_role` 的角色，以及层次中位于该角色之上的角色：

```
sp_displayroles "intern_role", expand_up
```

示例 5 显示为指定角色配置的与登录安全性相关的参数：

```
sp_displayroles physician_role, "display_info"
```

```
Role name = physician_role
```

```
Locked : YES
```

```
    Date when locked: Jul 14 2007 9:15AM
```

```
    Reason: Role locked by Adaptive Server due to failed login
    attempts reaching max failed logins.
```

```
    Locking suid: dr_john
```

```
Date of Last Password Change : Oct 31 1999 3:33PM
```

```
Password expiration interval = 5
```

```
Password expired : NO
```

```
Minimum password length = 4
```

```
Maximum failed logins = 10
```

```
Current failed logins = 3
```

```
Password encryption version: SHA-256
```

示例 6 显示授予登录名 “tom” 的角色，该角色与名为 “sec_profile” 的登录配置文件相关联：

```
grant role sec_role to sec_profile
```

```
create login tom with password C0mp13x login profile sec_profile
```

```
grant role emp_role to tom
```

```
go
```

```
sp_displayroles tom
```

```
go
```

| Role Name | Grantee |
|-----------|-------------|
| emp_role | tom |
| sec_role | sec_profile |

用法

当指定了可选参数 `expand_up` 或 `expand_down` 时，将显示被指定角色名包含的或包含指定角色名的所有直接授予的角色。

`Grantee` 列仅在登录名具有关联登录配置文件或缺省登录配置文件适用于具有授予它的角色的登录名时才会显示。

权限 只有系统管理员才能执行 `sp_displayroles`，以显示有关授予任何其它用户的角色的信息。所有用户都可以执行 `sp_displayroles` 以查看向其授予的角色。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关详细信息，请参见《系统管理指南》中的“用户定义的登录安全性”。

命令 [alter role](#)、[create role](#)、[drop role](#)、[grant](#)、[revoke](#)、[set](#)

系统过程 [sp_activeroles](#)、[sp_displaylogin](#)、[sp_helprotect](#)

sp_downgrade

| | |
|----|--|
| 说明 | (仅限 master 数据库) 验证是否已准备好降级至早期的 15.0.x 版。同时降级 Adaptive Server 15.0.2 已修改的系统目录更改。 |
| 语法 | <pre>sp_downgrade @cmd = {'prepare' 'downgrade' 'help'}, @toversion = 'n'[, @verbose = 0 1][, @override = 0 1]</pre> |
| 参数 | <p>prepare 先使用该参数来验证 Adaptive Server 15.0.2 是否为降级做好了准备。</p> <p>downgrade 如果已做好准备, 则在 prepare 参数之后用于继续执行降级至以前安装的 15.x 版的 Adaptive Server 的操作。服务器必须在单用户模式下。(以 -m 选项开头)</p> <p>toversion 可以是 15.0 或 15.0.1, 分别写作 “150” 或 “15.0” 以及 “1501” 或 “15.0.1”。</p> <p>verbose specifies verbosity。有效选项为 0 (表示 “否”) 或 1 (表示 “是”)。</p> <p>override 指定是否跳过此时不可写的数据库。有效选项为 0 (表示 “否”) 或 1 (表示 “是”)。</p> |

示例 **示例 1** 以下是运行 **sp_downgrade** 的输出示例。

```
00:0006:00000:00006:2011/06/29 02:16:44.35 server   Preparing ASE downgrade from
15.7.0.0 to 15.5.0.0.
00:0006:00000:00006:2011/06/29 02:16:44.37 server   Starting downgrading ASE.
00:0006:00000:00006:2011/06/29 02:16:44.37 server   Downgrade : Marking stored
procedures to be recreated from text.
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Removing full logging
modes from sysattributes.
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Downgrading data-only
locked table rows.
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Removing full logging
modes from sysattributes.
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Removing column
sysoptions.number.
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Removing srvprincipal
column from syssservers system table
00:0006:00000:00006:2011/06/29 02:16:45.34 server   Downgrade : Removing 'automatic
master key access' configuration parameter.
00:0006:00000:00006:2011/06/29 02:16:45.35 server   Downgrade : Removing DualControl
sysattribute rows
00:0006:00000:00006:2011/06/29 02:16:45.35 server   Downgrade : Downgrading
sysattributes system table.
00:0006:00000:00006:2011/06/29 02:16:45.37 server   Downgrade : Downgrading
```

```
syscomments system table.
00:0006:00000:00006:2011/06/29 02:16:45.42 server Downgrade : Truncated role
password, locked role and removed columns locksuid, lockreason, lockdate from
sysssrvroles
00:0006:00000:00006:2011/06/29 02:16:45.43 server Downgrade : Removing catalog
changes for RSA Keypair Regeneration Period and Login Profile
00:0006:00000:00006:2011/06/29 02:16:45.43 server Downgrade : Turning on database
downgrade indicator.
00:0006:00000:00006:2011/06/29 02:16:45.43 server Downgrade : Resetting database
version indicator.
00:0006:00000:00006:2011/06/29 02:16:45.43 server ASE downgrade completed.
```

示例 2 检查数据库是否做好降级准备:

```
1> sp_downgrade 'prepare','15.5',1
2> go
Downgrade from 15.7.0.0 to 15.5.0.0 (command: 'prepare')
```

Checking databases for downgrade readiness.

There are no errors which involve encrypted columns.
sp_downgrade 'prepare' completed.
(return status = 0)

示例 3 将 Adaptive Server 从 15.7 版降级到 15.5 版:

```
1> sp_downgrade 'downgrade','15.5',1
2> go

Downgrade from 15.7.0.0 to 15.5.0.0 (command: 'downgrade')
```

Checking databases for downgrade readiness.

There are no errors which involve encrypted columns.

Executing downgrade step 2 [dbcc markprocs(@dbid)] for :

```
- Database: master (dbid: 1)
sql comman is: dbcc markprocs(@dbid)
```

DBCC execution completed. If DBCC printed error messages, contact a user with System Administrator (SA) role.

```
- Database: tempdb (dbid: 2)
sql comman is: dbcc markprocs(@dbid)
```

DBCC execution completed. If DBCC printed error messages, contact a user with System Administrator (SA) role.

```
- Database: model (dbid: 3)
sql comman is: dbcc markprocs(@dbid)
```

DBCC execution completed. If DBCC printed error messages, contact a user with System Administrator (SA) role.

```
- Database: sybssystemdb (dbid: 31513)
sql comman is: dbcc markprocs(@dbid)
```

DBCC execution completed. If DBCC printed error messages, contact a user with System Administrator (SA) role.

```
- Database: sybssystemprocs (dbid: 31514)
sql comman is: dbcc markprocs(@dbid)
```

DBCC execution completed. If DBCC printed error messages, contact a user with System Administrator (SA) role.

Executing downgrade step 17 [delete sysattributes where class = 38] for :

```
- Database: master (dbid: 1)
sql comman is: delete sysattributes where class = 38
```

```
Executing downgrade step 18 [declare @ret int select @ret =
dol_downgrade_check(':DBNAME:', @toversid) print "Database :DBNAME: table downgrade
status: %1!", @ret if @ret != 0 begin print "**** Tables in database ':DBNAME:' cannot
be downgraded." print "**** See the server error log for details." select
@exec_error_count = @exec_error_count + 1 end] for :
```

```
- Database: master (dbid: 1)
sql comman is: declare @ret int select @ret = dol_downgrade_check('master', @toversid)
print "Database master table downgrade status: %1!", @ret if @ret != 0 begin print
"**** Tables in database 'master' cannot be downgraded." print "**** See the server
error log for details." select @exec_error_count = @exec_error_count + 1 end
```

Database master table downgrade status: 0

```
- Database: tempdb (dbid: 2)
```

```
sql comman is: declare @ret int select @ret = dol_downgrade_check('tempdb', @toversid)
print "Database tempdb table downgrade status: %1!", @ret if @ret != 0 begin print
"**** Tables in database 'tempdb' cannot be downgraded." print "**** See the server
error log for details." select @exec_error_count = @exec_error_count + 1 end
```

Database tempdb table downgrade status: 0

```
- Database: model (dbid: 3)
```

```
sql comman is: declare @ret int select @ret = dol_downgrade_check('model', @toversid)
print "Database model table downgrade status: %1!", @ret if @ret != 0 begin print
"**** Tables in database 'model' cannot be downgraded." print "**** See the server
error log for details." select @exec_error_count = @exec_error_count + 1 end
```

Database model table downgrade status: 0

```
- Database: sybssystemdb (dbid: 31513)
```

```
sql comman is: declare @ret int select @ret = dol_downgrade_check('sybssystemdb',
@toversid) print "Database sybssystemdb table downgrade status: %1!", @ret if @ret !=
0 begin print "**** Tables in database 'sybssystemdb' cannot be downgraded." print
"**** See the server error log for details." select @exec_error_count =
@exec_error_count + 1 end
```

Database sybssystemdb table downgrade status: 0

```
- Database: sybssystemprocs (dbid: 31514)
sql comman is: declare @ret int select @ret = dol_downgrade_check('sybssystemprocs',
@toversid) print "Database sybssystemprocs table downgrade status: %1!", @ret if @ret
!= 0 begin print "*** Tables in database 'sybssystemprocs' cannot be downgraded."
print "*** See the server error log for details." select @exec_error_count =
@exec_error_count + 1 end
```

Database sybssystemprocs table downgrade status: 0

Executing downgrade step 19 [delete sysattributes where class = 38] for :

```
- Database: master (dbid: 1)
sql comman is: delete sysattributes where class = 38
```

Executing downgrade step 20 [delete syscolumns where id = object_id('sysoptions') and name='number'] for :

```
- Database: master (dbid: 1)
sql comman is: delete syscolumns where id = object_id('sysoptions') and name='number'
```

Executing downgrade step 21 [delete syscolumns where id = object_id('syssservers') and name = 'srvprincipal'] for :

```
- Database: master (dbid: 1)
sql comman is: delete syscolumns where id = object_id('syssservers') and name =
'srvprincipal'
```

Executing downgrade step 22 [delete sysconfigures where config = 503] for :

```
- Database: master (dbid: 1)
sql comman is: delete sysconfigures where config = 503
```

Executing downgrade step 23 [delete sysattributes where class = 25 and attribute in (2, 3)] for :

```
- Database: master (dbid: 1)
sql comman is: delete sysattributes where class = 25 and attribute in (2, 3)
```

Executing downgrade step 24 [update :DBNAME:..sysattributes set object_cinfo2 = null, object_datetime = null where object_cinfo2 is not null or object_datetime is not null delete :DBNAME:..syscolumns where id = 21 and name in ('object_cinfo2', 'object_datetime')] for :

```
- Database: master (dbid: 1)
sql comman is: update master..sysattributes set object_cinfo2 = null, object_datetime
= null where object_cinfo2 is not null or object_datetime is not null delete
master..syscolumns where id = 21 and name in ('object_cinfo2', 'object_datetime')
```

```
- Database: tempdb (dbid: 2)
```

```
sql comman is: update tempdb..sysattributes set object_cinfo2 = null, object_datetime
= null where object_cinfo2 is not null or object_datetime is not null delete
tempdb..syscolumns where id = 21 and name in ('object_cinfo2', 'object_datetime')
```



```
- Database: model (dbid: 3)
sql comman is: update model..sysattributes set object_cinfo2 = null, object_datetime
= null where object_cinfo2 is not null or object_datetime is not null delete
model..syscolumns where id = 21 and name in ('object_cinfo2', 'object_datetime')
```

```
- Database: sybssystemdb (dbid: 31513)
sql comman is: update sybssystemdb..sysattributes set object_cinfo2 = null,
object_datetime = null where object_cinfo2 is not null or object_datetime is not null
delete sybssystemdb..syscolumns where id = 21 and name in ('object_cinfo2',
'object_datetime')
```

```
- Database: sybssystemprocs (dbid: 31514)
sql comman is: update sybssystemprocs..sysattributes set object_cinfo2 = null,
object_datetime = null where object_cinfo2 is not null or object_datetime is not null
delete sybssystemprocs..syscolumns where id = 21 and name in ('object_cinfo2',
'object_datetime')
```

```
Executing downgrade step 25 [update :DBNAME:..syscomments set encrkeyid = null where
encrkeyid is not null delete:DBNAME:..syscolumns where id = 6 and name = 'version'
delete :DBNAME:..syscolumns where id = 6 and name ='encrkeyid'] for :
```

```
- Database: master (dbid: 1)
sql comman is: update master..syscomments set encrkeyid = null where encrkeyid is not
null delete master..syscolumns where id = 6 and name = 'version' delete
master..syscolumns where id = 6 and name ='encrkeyid'
```

```
- Database: tempdb (dbid: 2)
sql comman is: update tempdb..syscomments set encrkeyid = null where encrkeyid is not
null delete tempdb..syscolumns where id = 6 and name = 'version' delete
tempdb..syscolumns where id = 6 and name ='encrkeyid'
```

```
- Database: model (dbid: 3)
sql comman is: update model..syscomments set encrkeyid = null where encrkeyid is not
null delete model..syscolumns where id = 6 and name = 'version' delete
model..syscolumns where id = 6 and name ='encrkeyid'
```

```
- Database: sybssystemdb (dbid: 31513)
sql comman is: update sybssystemdb..syscomments set encrkeyid = null where encrkeyid
is not null delete sybssystemdb..syscolumns where id = 6 and name = 'version' delete
sybssystemdb..syscolumns where id = 6 and name ='encrkeyid'
```

```
- Database: sybssystemprocs (dbid: 31514)
sql comman is: update sybssystemprocs..syscomments set encrkeyid = null where encrkeyid
is not null delete sybssystemprocs..syscolumns where id = 6 and name = 'version' delete
sybssystemprocs..syscolumns where id = 6 and name ='encrkeyid'
```

```
Executing downgrade step 26 [delete statistics sysssrvroles(password) if exists
(select 1 from sysssrvroles where password is not null) begin print "Truncating
password and locking following role(s)" select name from sysssrvroles where password
is not null update sysssrvroles set password = null, status = (status | @lockrole)
```

```
where password is not null end update syscolumns set length = 30 where id =
object_id('sysssrvroles') and name = 'password' update sysssrvroles set locksuid = null,
lockreason = null, lockdate = null where locksuid is not null or lockreason is not
null or lockdate is not null delete syscolumns where id = object_id('sysssrvroles')
and name in ('locksuid', 'lockreason', 'lockdate')] for :
```

```
- Database: master (dbid: 1)
```

```
sql comman is: delete statistics sysssrvroles(password) if exists (select 1 from
sysssrvroles where password is not null) begin print "Truncating password and locking
following role(s)" select name from sysssrvroles where password is not null update
sysssrvroles set password = null, status = (status | @lockrole) where password is not
null end update syscolumns set length = 30 where id = object_id('sysssrvroles') and
name = 'password' update sysssrvroles set locksuid = null, lockreason = null, lockdate
= null where locksuid is not null or lockreason is not null or lockdate is not null
delete syscolumns where id = object_id('sysssrvroles') and name in ('locksuid',
'lockreason', 'lockdate')
```

```
Truncating password and locking following role(s)
```

```
name
```

```
-----
```

```
doctor_role
```

```
Executing downgrade step 27 [delete sysattributes where class = 35 delete
sysattributes where class = 39 update syslogins set lpid = null, crsuid = null where
lpid is not null or crsuid is not null delete syscolumns where id =
object_id('syslogins') and name in ('lpid', 'crsuid') delete syslogins where (status
& @lp_status) = @lp_status update syslogins set status = status & ~(@exempt_lock)
where (status & @exempt_lock) = @exempt_lock] for :
```

```
- Database: master (dbid: 1)
```

```
sql comman is: delete sysattributes where class = 35 delete sysattributes where class
= 39 update syslogins set lpid = null, crsuid = null where lpid is not null or crsuid
is not null delete syscolumns where id = object_id('syslogins') and name in ('lpid',
'crsuid') delete syslogins where (status & @lp_status) = @lp_status update syslogins
set status = status & ~(@exempt_lock) where (status & @exempt_lock) = @exempt_lock
```

```
Executing downgrade step 998 [declare @d int, @stat4 int select @stat4=convert(int,
dbinfo_get('master','status4')) select @d=dbinfo_update(1, 'status4', 32 | @stat4)]
for :
```

```
- Database: master (dbid: 1)
```

```
sql comman is: declare @d int, @stat4 int select @stat4=convert(int,
dbinfo_get('master','status4')) select @d=dbinfo_update(1, 'status4', 32 | @stat4)
```

```
Executing downgrade step 999 [declare @d int select @d=dbinfo_update(@dbid,
'ASEvers', 15500)] for :
```

```
- Database: master (dbid: 1)
```

```
sql comman is: declare @d int select @d=dbinfo_update(@dbid, 'ASEvers', 15500)
```

```
- Database: tempdb (dbid: 2)
```

```
sql comman is: declare @d int select @d=dbinfo_update(@dbid, 'ASEvers', 15500)
```

```
- Database: model (dbid: 3)
sql comman is: declare @d int select @d=dbinfo_update(@dbid, 'ASEvers', 15500)

- Database: sybssystemdb (dbid: 31513)
sql comman is: declare @d int select @d=dbinfo_update(@dbid, 'ASEvers', 15500)

- Database: sybssystemprocs (dbid: 31514)
sql comman is: declare @d int select @d=dbinfo_update(@dbid, 'ASEvers', 15500)

(return status = 0)
```

用法

用来还原为以前安装的 Adaptive Server 15.0.x 版。目前不能还原为 Adaptive Server 15.0.2。

角色口令和 sp_downgrade

当降级到低于 15.7 版的 Adaptive Server 时，sp_downgrade 系统过程在内部连同要降级的 Adaptive Server 版本号一起调用 sp_passwordpolicy [prepare | downgrade]。

当您执行 sp_downgrade 时，Adaptive Server 会执行以下任务：

- 截断角色口令和锁定角色。
- 删除 sysattributes 中类 35 下新添加的属性。
- 删除 sysattributes 中新添加的类 35。
- 从 sysserveroles 中删除新的 locksuid、lockreason 和 lockdate 列。

注释 当您将 Adaptive Server 15.7 版降级到 15.0.2 之前的版本时，角色和登录口令都会降级。但是，当降级到 15.0.2 版时，Adaptive Server 会仅截断并锁定角色口令。

有关降级角色口令的详细信息，请参见针对您的平台的安装指南中的降级部分。

sp_dropalias

| | |
|----|---|
| 说明 | 删除通过 <code>sp_addalias</code> 建立的别名用户名标识。 |
| 语法 | <code>sp_dropalias loginame [, force]</code> |
| 参数 | <p><i>loginame</i></p> <p>是用户的名称（在 <code>master.dbo.syslogins</code> 中），该名称是另一个用户的别名。</p> <p><i>force</i></p> <p>即使某个别名拥有数据库对象，也可以使用此参数删除该别名。</p> |
| 示例 | <p>示例 1 假设 “victoria” 在当前数据库中是别名（例如，是数据库所有者的别名），则此语句将 “victoria” 作为一个具有别名的用户从数据库中删除：</p> <pre>sp_dropalias victoria</pre> <p>示例 2 删除别名 “harry”，它拥有一个过程 <code>namelist</code>。Adaptive Server 删除此别名，但会发出一条警告消息：</p> <pre>sp_dropalias harry, force Warning: You have forced the drop of the alias for login 'harry' which owns objects in the database. This may result in errors when those objects are accessed from or contain references to another database. Alias user dropped. (return status = 0)</pre> |
| 用法 | <ul style="list-style-type: none">• 执行 <code>sp_dropalias</code> 过程将从 <code>sysalternates</code> 表中删除用户的替代 <code>suid</code> 映射。• 删除了用户的别名后，该用户将不再能够访问为其创建别名的数据库。• 可以删除在数据库中拥有对象的用户的别名。在删除登录名之前不需要先删除对象。 |
| 权限 | 只有数据库所有者、系统管理员或系统安全员才能执行 <code>sp_dropalias</code> 。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程 sp_addalias](#)、[sp_adduser](#)、[sp_droplogin](#)、[sp_dropuser](#)、[sp_helpuser](#)

sp_drop_all_qplans

| | |
|----|---|
| 说明 | 删除抽象计划组中的所有抽象计划。 |
| 语法 | <code>sp_drop_all_qplans name</code> |
| 参数 | <i>name</i> 是要从其中删除所有计划的抽象计划组的名称。 |
| 示例 | <code>sp_drop_all_qplans dev_test</code> |
| 用法 | <ul style="list-style-type: none"> 若要删除个别计划，请使用 sp_drop_qplan。 若要查看当前数据库中抽象计划组的名称，请使用 sp_help_qpgroup。 <code>sp_drop_all_qplans</code> 自动删除组中属于指定用户的所有计划，或组中的所有计划（如果该命令是由系统管理员或数据库所有者执行的）。 |
| 权限 | 任何用户都可以执行 <code>sp_drop_all_qplans</code> 删除其拥有的计划。只有系统管理员或数据库所有者才能删除其他用户拥有的计划。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色—当前活动角色 关键字或选项—NULL 以前值—NULL 当前值—NULL 其它信息—所有输入参数 代理信息—<code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_drop_qplan](#)、[sp_drop_qpgroup](#)

sp_drop_qpgroup

说明 删除抽象计划组。

语法 `sp_drop_qpgroup group`

参数 `group`
是要删除的抽象计划组的名称。

示例 删除抽象计划组 “dev_test”：

```
sp_drop_qpgroup dev_test
```

用法

- 不能删除缺省组 `ap_stdin` 和 `ap_stdout`。
- 不能删除含有计划的组。若要删除组中的所有计划，请使用 [sp_drop_all_qplans](#)。若要查看组和它们包含的计划数的列表，请使用 [sp_help_qpgroup](#)。
- `sp_drop_qpgroup` 不能在事务中运行。

权限 只有系统管理员或数据库所有者才能执行 `sp_drop_qpgroup`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_drop_all_qplans](#)、[sp_help_qpgroup](#)

sp_drop_qplan

说明 删除抽象计划。

语法 sp_drop_qplan *id*

参数 *id*
是要删除的抽象计划的 ID。

示例 删除具有指定 ID 的抽象计划：

```
sp_drop_qplan 1760009301
```

用法

- 若要查找计划的 ID，请使用 [sp_help_qpgroup](#)、[sp_help_qplan](#) 或 [sp_find_qplan](#)。create plan 还返回计划 ID，这些计划 ID 包括在 showplan 输出中。

- 若要删除组中的所有抽象计划，请使用 [sp_drop_all_qplans](#)。

权限 任何用户都可以执行 sp_drop_qplan 来删除其所拥有的计划。只有系统管理员或数据库所有者才能删除其他用户拥有的计划。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [create plan](#)

系统过程 [sp_drop_all_qplans](#)、[sp_find_qplan](#)、[sp_help_qpgroup](#)、[sp_help_qplan](#)

sp_drop_resource_limit

说明 从 Adaptive Server 中删除一个或多个资源限制。

语法 `sp_drop_resource_limit { name, appname }
[, rangename, limittype, enforced, action, scope]`

参数 *name*

是对其应用限制的 Adaptive Server 登录名。若要删除应用于特定应用程序的所有用户的资源限制，请将 *appname* 和 *name* 指定为 NULL。

appname

是对其应用限制的应用程序。若要删除应用于指定登录所使用的所有应用程序的资源限制，请将登录名和 *appname* 指定为 NULL。若要删除应用于特定应用程序的限制，请指定客户端程序在登录包中传递给 Adaptive Server 的应用程序名。

rangename

是强制执行限制的时间范围。该时间范围必须是 `systimeranges` 系统表中存储的现有时间范围或 NULL，后者表示删除具有指定的 *name*、*appname*、*limittype*、*action* 和 *scope* 的所有资源限制，而不考虑 *rangename*。

limittype

是被限制的资源的类型。必须是以下值之一：

- `row_count` — 只删除限制查询可返回行数的限制。
- `elapsed_time` — 只删除限制查询批处理或事务可以运行秒数的限制。
- `io_cost` — 只删除限制实际的或估计的查询处理开销的限制。
- `tempdb_space` — 只删除限制单个会话期间使用或可以具有的 `tempdb` 数据库页数的限制。
- NULL — 删除具有指定的 *name*、*appname*、*rangename*、强制时间、*action* 和 *scope* 的所有资源限制，而不考虑 *limittype*。

enforced

确定是在执行查询前还是执行查询时强制应用限制。下表列出了每种限制类型的有效值：

| 强制代码 | 说明 | 限制类型 |
|------|--------------------------------------|---|
| 1 | 只删除当估计的执行开销超过指定的限制时采取操作的限制。 | <code>io_cost</code> |
| 2 | 只删除当实际行计数、经历的时间或执行开销超过指定的限制时采取操作的限制。 | <code>row_count</code> <code>elapsed_time</code> <code>io_cost</code> |

| 强制代码 | 说明 | 限制类型 |
|------|--|---------|
| 3 | 只删除当估计的开销 (1) 或实际的开销 (2) 超过指定的限制时采取操作的限制。 | io_cost |
| NULL | 删除具有指定的 <i>name</i> 、 <i>appname</i> 、 <i>rangename</i> 、 <i>limittype</i> 和 <i>scope</i> 的所有资源限制，而不考虑何时强制执行 <i>action</i> 。 | |

action

是超过限制时采取的操作，必须为以下值之一：

| action 代码 | 说明 |
|-----------|--|
| 1 | 只删除发出警告的限制。 |
| 2 | 只删除中止查询批处理的限制。 |
| 3 | 只删除中止事务的限制。 |
| 4 | 只删除注销会话的限制。 |
| NULL | 删除具有指定的 <i>name</i> 、 <i>appname</i> 、 <i>rangename</i> 、 <i>limittype</i> 、强制时间和 <i>scope</i> 的所有资源限制，而不考虑采取的 <i>action</i> 。 |

scope

是限制的作用域，必须为以下值之一：

| scope 代码 | 说明 |
|----------|--|
| 1 | 只删除应用于查询的限制。 |
| 2 | 只删除应用于查询批处理的限制。 |
| 4 | 只删除应用于事务的限制。 |
| 6 | 只删除应用于查询批处理和事务的限制。 |
| NULL | 删除具有指定的 <i>name</i> 、 <i>appname</i> 、 <i>rangename</i> 、 <i>limittype</i> 、强制时间和 <i>action</i> 的所有资源限制，而不考虑其 <i>scope</i> 。 |

示例

示例 1 删除一个资源限制（当 *joe* 使用 *payroll* 应用程序在 *friday_afternoon* 时间范围内运行查询而导致过多的执行时间 I/O 开销时，该资源限制将注销会话）：

```
sp_drop_resource_limit joe, payroll, friday_afternoon, io_cost, 2, 4, 1
```

注释 如果没有资源限制符合这些选择标准，则 *sp_drop_resource_limit* 将不返回任何错误。

示例 2 删除对 *joe* 使用 *payroll* 应用程序的所有限制：

```
sp_drop_resource_limit joe, payroll
```

示例 3 删除应用于用户“*joe*”的所有限制：

```
sp_drop_resource_limit joe
```

示例 4 删除应用于 *payroll* 应用程序的所有资源限制：

```
sp_drop_resource_limit NULL, payroll
```

示例 5 删除对 *payroll* 应用程序的所有资源限制，该应用程序的操作是注销会话：

```
sp_drop_resource_limit NULL, payroll, NULL, NULL, NULL, 4, NULL
```

用法

- 使用 `sp_help_resource_limit` 系统过程可以确定哪些资源限制应用于给定的用户、应用程序或时间范围。
- 使用 `sp_droplogin` 删除 Adaptive Server 登录名时，与此登录名关联的所有资源限制也将被删除。
- 删除资源限制将导致对这一登录名和 / 或应用程序的每个会话的限制在该会话的下次查询批处理开始时重新绑定。

权限

只有系统管理员才能执行 `sp_drop_resource_limit`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关资源限制的详细信息，请参见《系统管理指南》。

系统过程 [sp_add_resource_limit](#)、[sp_droplogin](#)、[sp_help_resource_limit](#)、[sp_modify_resource_limit](#)

sp_drop_time_range

- 说明** 从 Adaptive Server 中删除用户定义的时间范围。
- 语法** `sp_drop_time_range name`
- 参数** *name*
是要删除的时间范围的名称。
- 示例** 删除 “evenings” 时间范围：
`sp_drop_time_range evenings`
- 用法**
- 不能删除 “所有时间” 时间范围。
 - 如果某个时间范围存在资源限制，则不能删除该时间范围。
 - 对于当前正在运行的会话，删除某个时间范围不会影响其活动时间范围。
- 权限** 只有系统管理员才能执行 `sp_drop_time_range`。
- 审计** sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **文档** 有关时间范围的详细信息，请参见 《系统管理指南》。

系统过程 [sp_add_resource_limit](#)、[sp_add_time_range](#)、[sp_modify_time_range](#)

sp_dropdevice

说明 删除 Adaptive Server 数据库设备或转储设备。

语法 sp_dropdevice *logicalname*

参数 *logicalname*

是 master.dbo.sysdevices.name 中列出的设备的名称。

示例 **示例 1** 从 Adaptive Server 中删除名为 `tape5` 的设备：

```
sp_dropdevice tape5
```

示例 2 从 Adaptive Server 中删除名为 `fredsdata` 的数据库设备。此时该设备一定不能被任何数据库使用：

```
sp_dropdevice fredsdata
```

用法

- sp_dropdevice 过程从 Adaptive Server 中删除设备，同时从 master.dbo.sysdevices 中删除该设备的条目。
- sp_dropdevice 不会删除被当做数据库设备删除的文件；它使该文件不能被 Adaptive Server 访问。使用 sp_dropdevice 后，请使用操作系统命令删除文件。

权限 只有系统管理员才能执行 sp_dropdevice。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [drop database](#)

系统过程 [sp_addumpdevice](#)、[sp_helpdb](#)、[sp_helpdevice](#)

sp_dropengine

| | |
|-------------|---|
| 说明 | sp_dropengine 不在线程化模式中运行。 |
| 有关进程模式的考虑事项 | 从指定的引擎组中删除引擎，如果该引擎是组中的最后一个引擎，则删除引擎组。 |
| 语法 | <code>sp_dropengine engine_number [,engine_group] [,instance_id]</code> |
| 参数 | <p><i>engine_number</i> 是要从组中删除的引擎的编号。值介于 0 和最大值之间，最大值等于配置的联机引擎数减 1。</p> <p><i>engine_group</i> 是要从其中删除引擎的引擎组的名称。</p> <p><i>instance_id</i> (仅限集群环境)要从中删除引擎或引擎组的实例的 ID。</p> |
| 示例 | <p>示例 1 从名为 DS_GROUP 的引擎组中删除编号为 2 的引擎。如果它是组中的最后一个引擎，则还将删除该组：</p> <pre>sp_dropengine 2, DS_GROUP</pre> <p>示例 2 (仅限集群环境)从 ID 为 8 的实例中删除编号为 5 的引擎：</p> <pre>sp_dropengine 5, 8</pre> |
| 用法 | <ul style="list-style-type: none">只能从 master 数据库中调用 sp_dropengine。如果 <i>engine_number</i> 是 <i>engine_group</i> 中的最后一个引擎，则 Adaptive Server 还将删除 <i>engine_group</i>。在集群环境中—如果 sp_cluster set system_view 设置为 cluster，则可以在集群中的任意实例上删除引擎或引擎组。如果 system_view 设置为 instance，则只能向本地实例中删除引擎或引擎组。sp_dropengine 可以在您使用 sp_procxmode 将事务模式更改为 anymode 后在使用链式事务的会话中运行。您指定的 <i>engine_number</i> 必须存在于 <i>engine_group</i> 中。 |
| 权限 | 只有系统管理员才能执行 sp_dropengine。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addengine](#)

sp_dropexeclass

| | |
|----|---|
| 说明 | 删除用户定义的执行类。 |
| 语法 | <code>sp_dropexeclass <i>classname</i></code> |
| 参数 | <i>classname</i> 是要删除的用户定义的执行类的名称。 |
| 示例 | 此语句删除用户定义的执行类 DECISION: <code>sp_dropexeclass 'DECISION'</code> |
| 用法 | <ul style="list-style-type: none"> 执行类有助于定义 Adaptive Server 处理任务时使用的执行优先顺序。有关执行类和执行属性的详细信息，请参见《性能和调优指南》。 不要将 <i>classname</i> 绑定到任何客户端应用程序、登录名、存储过程或缺省执行类。首先使用 sp_unbindexeclass 解除执行类的绑定，然后使用 <code>sp_dropexeclass</code> 删除执行类。 不能删除系统定义的执行类。 |
| 权限 | 只有系统管理员才能执行 <code>sp_dropexeclass</code> 。 |
| 审计 | sysaudits 表的 event 和 extrainfo 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程 sp_addexeclass](#)、[sp_bindexeclass](#)、[sp_showexeclass](#)、[sp_unbindexeclass](#)

sp_dropextendedproc

| | |
|----|--|
| 说明 | 删除扩展存储过程 (ESP)。 |
| 语法 | <code>sp_dropextendedproc esp_name</code> |
| 参数 | <i>esp_name</i> 是要删除的扩展存储过程的名称。 |
| 示例 | 删除 xp_echo: <pre>sp_dropextendedproc xp_echo</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_dropextendedproc</code> 必须从 <code>master</code> 数据库中执行。 • <i>esp_name</i> 区分大小写。它必须精确地匹配创建 ESP 时使用的名称。 |
| 权限 | 只有系统管理员才能执行 <code>sp_dropextendedproc</code> 。 |
| 审计 | sysaudits 表的 event 和 extrainfo 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [drop procedure](#)

系统过程 [sp_addextendedproc](#)、[sp_freelibrary](#)、[sp_helpextendedproc](#)

sp_dropexternlogin

说明 (仅限组件集成服务) 删除以前通过 `sp_addexternlogin` 定义的远程登录的定义。 [sp_addexternlogin](#)

语法 `sp_dropexternlogin server [, loginame [, rolename]]`

参数 *server*

是本地服务器从中删除帐户访问的远程服务器的名称。远程服务器通过 `master.dbo.sysservers` 表中的条目为本地服务器所知。

loginame

是本地服务器所知的登录帐户。如果未指定 *loginame*，则使用当前帐户。*loginame* 必须存在于 `master.dbo.syslogins` 表中。

rolename

是 Adaptive Server 用户指派的角色。

示例 **示例 1** 从“bobj”删除对远程服务器 CIS1012 的外部登录名的定义。只有“bobj”帐户和“sa”帐户可以添加或修改“bobj”的远程登录名：

```
sp_dropexternlogin CIS1012, bobj
```

示例 2 从具有 `sa_role` 的用户删除对远程服务器 SSB 的外部登录名的定义：

```
sp_dropexternlogin SSB, NULL, sa_role
```

用法

- `sp_dropexternlogin` 删除以前通过 [sp_addexternlogin](#) 定义到本地服务器的远程登录名的定义。
- 不能从事务中执行 `sp_dropexternlogin`。
- 必须通过 [sp_addserver](#) 将远程服务器定义到本地服务器。
- 若要添加和删除本地服务器用户，请使用系统过程 [sp_addalias](#) 和 [sp_droplogin](#)。

权限

只有 *loginame* 或系统管理员才能执行 `sp_dropexternlogin`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_addexternlogin](#)、[sp_helpexternlogin](#)、[sp_addlogin](#)、[sp_droplogin](#)

sp_dropglockpromote

| | |
|----|---|
| 说明 | 从表或数据库中删除锁升级值。 |
| 语法 | <code>sp_dropglockpromote {"database" "table"}, objname</code> |
| 参数 | <p>database table 指定是否从数据库或表中删除锁升级阈值。因为这些都是 Transact-SQL 关键字，所以需要使用引号。</p> <p>objname 是要从其中删除锁升级阈值的表或数据库的名称。</p> |
| 示例 | 从 <code>titles</code> 中删除锁升级值。 <code>titles</code> 的锁升级现在使用数据库或全服务器范围的值： |
| 用法 | <ul style="list-style-type: none"> 使用 <code>sp_dropglockpromote</code> 删除用 <code>sp_setpglockpromote</code> 设置的锁升级值。 当删除数据库的锁升级阈值时，没有配置锁升级阈值的表将使用全服务器范围的值。 删除表的值之后，<code>Adaptive Server</code> 将使用数据库的锁升级阈值（如果已配置），或全服务器范围的值（如果未配置）。 可以通过 <code>sp_setpglockpromote</code> 更改全服务器范围的值，但不能将这些值删除。 |
| 权限 | 只有系统管理员才能执行 <code>sp_dropglockpromote</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程 sp_setpglockpromote](#)

sp_dropgroup

说明 从数据库中删除组。

语法 sp_dropgroup *grpname*

参数 *grpname*
是当前数据库中组的名称。

示例 “purchasing”组已与“accounting”组合并。在删除组之前，这些命令将“purchasing”组的成员“martha”和“george”移动到其它组中。因为“public”是保留字，所以将组名“public”用引号引起来：

```
sp_changegroup accounting, martha
sp_changegroup "public", george
sp_dropgroup purchasing
```

用法

- 执行 sp_dropgroup 将从数据库的 sysusers 表中删除组名。
- 不能删除包含成员的组。在删除组之前，必须为每个成员执行 [sp_changegroup](#)。

权限 只有数据库所有者、系统管理员或系统安全员才能执行 sp_dropgroup。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addgroup](#)、[sp_changegroup](#)、[sp_helpgroup](#)

sp_dropkey

| | |
|----|---|
| 说明 | 从 <code>syskeys</code> 表中删除使用 <code>sp_primarykey</code> 、 <code>sp_foreignkey</code> 或 <code>sp_commonkey</code> 定义的键。 |
| 语法 | <code>sp_dropkey keytype, tablename [, deptabname]</code> |
| 参数 | <p>keytype 是要删除的键的类型。 <i>keytype</i> 必须是 <code>primary</code>、<code>foreign</code> 或 <code>common</code>。</p> <p>tablename 是包含要删除的键的键表或视图的名称。</p> <p>deptabname 如果 <i>keytype</i> 是 <code>foreign</code> 或 <code>common</code>，则指定关系中的第二个表的名称。如果 <i>keytype</i> 是 <code>primary</code>，则不需要此参数，这是因为 <code>primary</code> 键没有相关表。如果 <i>keytype</i> 是 <code>foreign</code>，则这是主键表的名称。如果 <i>keytype</i> 是 <code>common</code>，则请按照执行 <code>sp_helpkey</code> 时它们出现的顺序给出这两个表名。</p> |
| 示例 | <p>示例 1 删除 <code>employees</code> 表的主键。依赖于 <code>employees</code> 的主键的任何外键也将被删除：</p> <pre>sp_dropkey primary, employees</pre> <p>示例 2 删除 <code>employees</code> 和 <code>projects</code> 表之间的公用键：</p> <pre>sp_dropkey common, employees, projects</pre> <p>示例 3 删除 <code>titleauthor</code> 和 <code>titles</code> 表之间的外键：</p> <pre>sp_dropkey foreign, titleauthor, titles</pre> |
| 用法 | <ul style="list-style-type: none">• 执行 <code>sp_dropkey</code> 将从 <code>syskeys</code> 中删除指定的键。只有表的所有者才能从该表中删除键。• 创建键是为了使数据库设计中隐式的逻辑关系变为显式的。此信息可供应用程序使用。• 删除主键将自动删除与之关联的所有外键。而删除外键不会对该表上指定的主键产生影响。• 执行 <code>sp_commonkey</code>、<code>sp_primarykey</code> 或 <code>sp_foreignkey</code> 会将键添加到 <code>syskeys</code> 系统表中。若要显示有关已定义键的报告，请执行 <code>sp_helpkey</code>。 |
| 权限 | 只有 <i>tablename</i> 的所有者才能执行 <code>sp_dropkey</code> 。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_commonkey](#)、[sp_foreignkey](#)、[sp_helpkey](#)、[sp_primarykey](#)

sp_droplanguage

| | |
|----|--|
| 说明 | 从服务器中删除替代语言并从 <code>master.dbo.syslanguages</code> 中删除相关的行。 |
| 语法 | <code>sp_droplanguage language [, dropmessages]</code> |
| 参数 | <p><i>language</i> 是要删除的语言的正式名称。</p> <p><i>dropmessages</i> 删除 <i>language</i> 中的所有 Adaptive Server 系统消息。在没有删除与语言关联的系统消息的情况下，不能删除该语言。</p> |
| 示例 | <p>示例 1 如果没有关联的消息，此示例将从可用替代语言中删除法语：</p> <pre>sp_droplanguage french</pre> <p>示例 2 如果有关联的消息，此示例将从可用替代语言中删除法语：</p> <pre>sp_droplanguage french, dropmessages</pre> |
| 用法 | <ul style="list-style-type: none"> 执行 <code>sp_droplanguage</code> 通过从 <code>master.dbo.syslanguages</code> 表中删除某个语言的条目将该语言从替代语言列表中删除。 如果要删除具有系统消息的语言，除非提供 <code>dropmessages</code> 参数，否则请求将失败。 |
| 权限 | 只有系统管理员才能执行 <code>sp_droplanguage</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_addlanguage](#)、[sp_helplanguage](#)

sp_droplogin

| | |
|----|--|
| 说明 | Adaptive Server 15.7 和更高版本不支持此系统过程。若要删除 Adapter Server 的登录帐户，请使用 <code>drop login</code> 命令。 |
| 语法 | 无 |
| 参数 | 无 |
| 用法 | 无 |

sp_dropmessage

| | |
|----|---|
| 说明 | 从 sysusermessages 中删除用户定义的消息。 |
| 语法 | <code>sp_dropmessage message_num [, language]</code> |
| 参数 | <p><i>message_num</i> 是要删除的消息的消息号。消息号的值必须大于或等于 20000。</p> <p><i>language</i> 是要删除的消息的语言。</p> |
| 示例 | 从 sysusermessages 中删除编号为 20002 的消息的法语版本： <code>sp_dropmessage 20002, french</code> |
| 用法 | <ul style="list-style-type: none"> <i>language</i> 参数是可选的。如果包含该参数，则将只删除以指定语言表示的、具有指定的 <i>message_num</i> 的消息。如果未指定 <i>language</i>，则将删除具有指定的 <i>message_num</i> 的所有消息。 |
| 权限 | 只有数据库所有者、系统管理员或创建这条要删除的消息的用户才能执行 <code>sp_dropmessage</code> 。 |
| 审计 | sysaudits 表的 event 和 extrainfo 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addmessage](#)、[sp_getmessage](#)

sp_dropobjectdef

| | |
|----|---|
| 说明 | (仅限组件集成服务) 删除为本地对象提供的外部存储映射。 |
| 语法 | <code>sp_dropobjectdef <i>tablename</i></code> |
| 参数 | <p><i>tablename</i></p> <p>的格式为 <code>dbname.owner.object</code>，其中：</p> <ul style="list-style-type: none"> • <i>dbname</i> 是数据库的名称，该数据库包含您要删除其存储位置的对象。<i>dbname</i> 是可选的；如果存在，它必须是当前数据库，并且需要有 <i>owner</i> 或占位符。 • <i>owner</i> 是您要删除其存储位置的对象的所有者的名称。<i>owner</i> 是可选的；如果指定了 <i>dbname</i>，则该项是必需的。 • <i>object</i> 是要为其删除外部存储映射的本地表的名称。 |
| 示例 | <p>示例 1 从 <code>sysattributes</code> 中删除为服务器已知的表（<code>personnel</code> 数据库中的 <code>colleges</code> 表）提供外部存储映射的条目：</p> <pre>sp_dropobjectdef "personnel.dbo.colleges"</pre> <p>示例 2 从 <code>sysattributes</code> 中删除为 <code>andrea.fishbone</code> 对象提供外部存储映射的条目，其中 <code>andrea</code> 是 <code>owner</code>，本地表名是 <code>fishbone</code>：</p> <pre>sp_dropobjectdef "andrea.fishbone"</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_dropobjectdef</code> 删除为本地对象提供的外部存储映射。它替代了 <code>sp_drotabledef</code>。 • 在用 <code>drop table</code> 删除了远程表之后，请使用 <code>sp_dropobjectdef</code>。 • 如果映射信息是使用 <code>sp_addobjectdef</code> 添加的，则删除表不会从 <code>sysattributes</code> 表中删除映射信息。必须使用 <code>sp_dropobjectdef</code> 显式删除这些映射信息。 • <i>tablename</i> 可以是以下任一形式： <ul style="list-style-type: none"> • <i>object</i> • <i>owner.object</i> • <i>dbname..object</i> • <i>dbname.owner.object</i> |
| 权限 | 只有数据库所有者或系统管理员才能执行 <code>sp_dropobjectdef</code> 。只有系统管理员才能执行 <code>sp_dropobjectdef</code> 删除另一个用户的对象的映射信息。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create existing table](#)、[create table](#)、[drop table](#)

系统过程 [sp_addobjectdef](#)

sp_dropremotelogin

| 说明 | 删除远程用户登录名。 | | | | | | | | |
|------|--|------------|---|------------|----------------|----|----------------|------|---|
| 语法 | <code>sp_dropremotelogin remoteserver [, loginame [, remotename]]</code> | | | | | | | | |
| 参数 | <p><i>remoteserver</i> 是具有要删除的远程登录名的服务器的名称。</p> <p><i>loginame</i> 是与 <code>sysremotelogins</code> 表中远程服务器关联的本地服务器的用户名。</p> <p><i>remotename</i> 是在从远程服务器登录时，被映射为 <i>loginame</i> 的远程用户名。</p> | | | | | | | | |
| 示例 | <p>示例 1 删除名为 GATEWAY 的远程服务器的条目：</p> <pre>sp_dropremotelogin GATEWAY</pre> <p>示例 2 删除从远程服务器 GATEWAY 的远程登录名到名为 “churchy” 的本地用户的映射条目：</p> <pre>sp_dropremotelogin GATEWAY, churchy</pre> <p>示例 3 删除远程服务器 GATEWAY 上远程用户 “pogo” 的登录名，该登录名被映射到名为 “churchy” 的本地用户：</p> <pre>sp_dropremotelogin GATEWAY, churchy, pogo</pre> | | | | | | | | |
| 用法 | <ul style="list-style-type: none"> • 执行 <code>sp_dropremotelogin</code> 将从远程服务器中删除用户登录名，并同时从 <code>master.dbo.sysremotelogins</code> 中删除该用户的条目。 • 有关远程登录名更完整的讨论，请参见 sp_addremotelogin。 • 若要添加和删除本地服务器用户，请使用系统过程 sp_addlogin 和 sp_droplogin。 | | | | | | | | |
| 权限 | 只有系统管理员才能执行 <code>sp_dropremotelogin</code> 。 | | | | | | | | |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： | | | | | | | | |
| | <table border="1"> <thead> <tr> <th>事件</th> <th>审计选项</th> <th>审计的命令或访问权限</th> <th>extrainfo 中的信息</th> </tr> </thead> <tbody> <tr> <td>38</td> <td>exec_procedure</td> <td>执行过程</td> <td> <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 </td> </tr> </tbody> </table> | 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 | 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |
| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 | | | | | | |
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 | | | | | | |
| 另请参见 | <p>系统过程 sp_addlogin、sp_addremotelogin、sp_addserver、sp_droplogin、sp_helpremotelogin、sp_helpserver</p> | | | | | | | | |

sp_dropprowlockpromote

| | |
|----|---|
| 说明 | 从数据库或表中删除行锁升级阈值。 |
| 语法 | <code>sp_dropprowlockpromote {"database" "table"}, objname</code> |
| 参数 | <p><code>database table</code> 指定是否从数据库或表中删除行锁升级阈值。</p> <p><code>objname</code> 是从中删除行锁升级阈值的数据库或表的名称。</p> |
| 示例 | <p>从 <code>sales</code> 表中删除行锁升级阈值。<code>sales</code> 的锁升级现在将使用数据库或全服务器范围的值：</p> <pre>sp_dropprowlockpromote "table", "sales"</pre> |
| 用法 | <ul style="list-style-type: none"> 使用 <code>sp_dropprowlockpromote</code> 可以删除通过 <code>sp_setrowlockpromote</code> 设置的行锁升级值。 删除数据库的行锁升级阈值后，没有配置行锁升级阈值的数据行锁定表将使用全服务器范围的值。使用 <code>sp_configure</code> 检查行锁升级配置参数的值。 删除表的行锁升级值后，Adaptive Server 将使用数据库的行锁升级阈值（如果已配置），或全服务器范围的值（如果未为该数据库设置阈值）。 若要更改数据库的锁升级阈值，必须使用 <code>master</code> 数据库。若要更改数据库中表的锁升级阈值，必须使用表所在的数据库。 可以使用 <code>sp_setrowlockpromote</code> 更改全服务器范围的值。这会更改行锁升级配置参数中的值，因此 <code>sp_dropprowlockpromote</code> 没有对应的服务器选项。 |
| 权限 | 只有系统管理员才能执行 <code>sp_dropprowlockpromote</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_setrowlockpromote](#)

sp_dropsegment

| | |
|----|--|
| 说明 | 从数据库中删除段或从特定的数据库设备取消段的映射。 |
| 语法 | <code>sp_dropsegment segname, dbname [, device]</code> |
| 参数 | <p>segname 是要删除的段的名称。</p> <p>dbname 是要从其中删除段的数据库的名称。</p> <p>device 是要从其中删除段 <i>segname</i> 的数据库设备的名称。除非要从数据库设备中删除系统段 <code>system</code>、<code>default</code> 或 <code>logsegment</code>，否则此参数是可选的。</p> |
| 示例 | <p>示例 1 此命令从 <code>pubs2</code> 数据库中删除段 <code>indexes</code>。</p> <pre>sp_dropsegment indexes, pubs2</pre> <p>示例 2 此命令从数据库设备 <code>dev1</code> 取消段 <code>indexes</code> 的映射：</p> <pre>sp_dropsegment indexes, pubs2, dev1</pre> |
| 用法 | <ul style="list-style-type: none"> • 如果在指定的数据库中没有任何表、索引或分区引用段，则可以删除该段。 • 如果未提供可选参数 <i>device</i>，则将从指定的数据库中删除该段。如果提供了 <i>device</i> 名称，则该段将不再映射到指定的数据库设备，但不会删除该段。 • 删除段将删除与该段关联的所有阈值。 • 只能以单用户模式为 <code>logsegment</code> 系统段执行 <code>sp_dropsegment</code>。 <hr/> <p>注释 处理很大的数据库时，此命令可能需要很长时间才能完成。</p> <hr/> <ul style="list-style-type: none"> • 当从一个或多个设备取消段的映射时，Adaptive Server 将删除超过该段上总空间的任何阈值。当从一个或多个设备取消 <code>logsegment</code> 的映射时，Adaptive Server 将重新计算最后机会阈值。 • <code>sp_placeobject</code> 逐段更改将来用于表或索引的空间分配，并从初始段中删除引用。在使用 <code>sp_placeobject</code> 之后，可以使用 <code>sp_dropsegment</code> 删除初始段名。 • 对于系统段 <code>system</code>、<code>default</code> 和 <code>logsegment</code>，您必须指定要从其中删除段的设备名。 |
| 权限 | 只有数据库所有者或系统管理员才能执行 <code>sp_dropsegment</code> 。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_addsegment](#)、[sp_addthreshold](#)、[sp_helpsegment](#)、[sp_helpthreshold](#)、[sp_placeobject](#)

sp_dropserver

说明 从已知服务器列表中删除服务器，或在同一操作中删除远程登录名和外部登录名。

语法 `sp_dropserver server [, droplogins]`

参数 `server`
是要删除的服务器的名称。

`droplogins`
指出还应删除 `server` 的任何远程登录名。

示例 **示例 1** 此命令删除远程服务器 GATEWAY:

```
sp_dropserver GATEWAY
```

示例 2 删除远程服务器 RDBAM_ALPHA 的条目并删除该服务器的所有远程登录名和外部登录名:

```
sp_dropserver RDBAM_ALPHA, droplogins
```

用法

- 执行 `sp_dropserver` 会通过从 `master.dbo.syssservers` 表中删除服务器的条目将该服务器从已知服务器列表中删除。
- 对在 `master.dbo.sysremotelogins` 表中具有关联条目的服务器运行 `sp_dropserver` 将产生一条操作消息，指出在删除服务器之前必须删除远程用户。若要在删除服务器时删除该服务器的所有远程登录名，请使用 `droplogins`。
- 对于在 `sysattributes` 表中具有关联条目的服务器，未运行 `droplogins` 就对其运行 `sp_dropserver` 将导致一个错误。在删除该服务器之前，必须先删除远程登录名和外部登录名。
- 在配置了组件集成服务之后，将检查 `sysattributes` 看看是否存在外部登录名和对服务器的缺省映射。

权限 只有系统安全员才能执行 `sp_dropserver`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_addserver](#)、[sp_dropremotelogin](#)、[sp_helpremotelogin](#)、[sp_helpserver](#)

sp_droptreshold

| | |
|----|--|
| 说明 | 从段中删除可用空间阈值。 |
| 语法 | <code>sp_droptreshold dbname, segname, free_space</code> |
| 参数 | <p>dbname 是您从中删除阈值的数据库。它必须是当前数据库的名称。</p> <p>segname 是其可用空间受阈值监控的段。指定“default”段时要使用引号。</p> <p>free_space 是达到阈值时的可用页数。</p> |
| 示例 | <p>从 mydb 的 segment1 中删除阈值。必须指定数据库、段和可用空间量来标识该阈值：</p> <pre>sp_droptreshold mydb, segment1, 200</pre> |
| 用法 | <ul style="list-style-type: none"> 不能从日志段中删除最后机会阈值。 可以使用 sp_dboption 的 <code>no free space acctg</code> 选项作为 <code>sp_droptreshold</code> 的替代项。此选项将在非日志段上禁用可用空间计数。不能禁用对日志段的可用空间计数。 |
| 权限 | 只有数据库所有者或系统管理员才能执行 <code>sp_droptreshold</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程](#) [sp_addthreshold](#)、[sp_dboption](#)、[sp_helpthreshold](#)、[sp_thresholdaction](#)

sp_droptype

| | |
|----|---|
| 说明 | 删除用户定义的数据类型。 |
| 语法 | <code>sp_droptype <i>typename</i></code> |
| 参数 | <i>typename</i> 是您拥有的用户定义的数据类型的名称。 |
| 示例 | 删除名为 <code>birthday</code> 的用户定义的数据类型： <pre>sp_droptype birthday</pre> |
| 用法 | <ul style="list-style-type: none"> 执行 <code>sp_droptype</code> 将从 <code>systypes</code> 中删除用户定义的数据类型。 如果表或其它数据库对象引用了一个用户定义的数据类型，则不能删除该数据类型。 |
| 权限 | 只有数据库所有者或数据类型所有者才能执行 <code>sp_droptype</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **数据类型** [用户定义的数据类型](#)
系统过程 [sp_addtype](#)、[sp_rename](#)

sp_dropuser

| | |
|----|--|
| 说明 | 从当前数据库中删除用户。 |
| 语法 | <code>sp_dropuser name_in_db</code> |
| 参数 | <code>name_in_db</code> 是当前数据库的 <code>sysusers</code> 表中的用户名。 |
| 示例 | 从当前数据库中删除用户“albert”。用户“albert”不能再使用该数据库了： <pre>sp_dropuser albert</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_dropuser</code> 通过从 <code>sysusers</code> 中删除用户的行从当前数据库中删除用户。 • 不能删除以下错误类： <ul style="list-style-type: none"> • 在数据库中拥有对象的用户。 • 曾向其他用户授予权限的用户。 • 数据库中的数据库所有者。 • 如果其他用户的别名是要被删除的用户，则还将删除他们的别名。他们不能再访问该数据库了。 • 如果用户拥有一个绑定到该数据库中执行类的存储过程，则不能从数据库中删除该用户。请参见 sp_bindexeclss。 • <code>sp_dropuser</code> 会从当前数据库的 <code>sysencryptkeys</code> 中删除指定用户的所有密钥副本。如果用户在任何数据库中拥有加密密钥，<code>sp_dropuser</code> 将失败。请参见《加密列用户指南》。 |
| 权限 | 只有数据库所有者、系统管理员或系统安全员才能执行 <code>sp_dropuser</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [grant](#)、[revoke](#)、[use](#)

系统过程 [sp_addalias](#)、[sp_adduser](#)、[sp_bindexeclss](#)、[sp_droplogin](#)

sp_dumpoptimize

| | |
|----|---|
| 说明 | 指定在转储数据库操作期间 Backup Server 转储的数据量。 |
| 语法 | <pre>sp_dumpoptimize ['archive_space = {maximum minimum default }'] sp_dumpoptimize ['reserved_threshold = {nnn default }'] sp_dumpoptimize ['allocation_threshold = {nnn default }']</pre> |
| 参数 | <p>archive_space 指定要转储的数据库量。</p> <p>maximum 在未确定分配哪些页的情况下转储整个数据库。档案图像使用的总空间量等于该数据库的大小。使用此选项与使用 <code>reserved_threshold=0</code> 和 <code>allocation_threshold=0</code> 选项的作用相同。</p> <p>minimum 只转储分配的页，这将产生可能的最小档案图像。当转储到吞吐量远小于数据库设备（例如 QIC 磁带驱动器）的吞吐量的档案设备时，可以使用此选项。使用此选项与使用 <code>reserved_threshold=100</code> 和 <code>allocation_threshold=100</code> 选项的作用相同。</p> <p>default 指定应使用的缺省值。在与以下选项一起使用时：</p> <ul style="list-style-type: none">• <code>archive_space</code> — 此选项转储数据库，<code>reserved_threshold</code> 和 <code>allocation_threshold</code> 选项被设置为其缺省值。使用该选项将 Backup Server 重新设置为缺省配置。• <code>reserved_threshold</code> — <code>default</code> 指定 85%。• <code>allocation_threshold</code> — <code>default</code> 指定 40%。 <p>reserved_threshold 如果数据库磁盘中保留的页的百分比大于或等于 <code>nnn</code>，则转储该磁盘中属于该数据库的所有页。例如，如果将 <code>nnn</code> 指定为 60，并且数据库磁盘中保留页的百分比等于或大于 60%，则将转储整个磁盘，而不确定分配该磁盘中的哪些页。此选项的缺省值为 85%。</p> |

nnn

一个介于 0 到 100 之间的整数值，该值表示阈值的值。该值用于确定要转储的数据量。

当和 `reserved_threshold` 一起使用时，如果磁盘中保留页的百分比大于指定的值，则将转储数据库磁盘中该数据库的所有页。

当和 `allocation_threshold` 一起使用时，如果分配单元中已分配页的百分比大于为 `allocation_threshold` 指定的百分比，则将转储分配单元中的所有页。

`allocation_threshold`

如果分配单元中已分配页的百分比大于或等于 `nnn`，则将转储该分配单元中的所有页。例如，如果将 `nnn` 指定为 70 并且分配单元中已分配页的百分比大于或等于 70%，则将转储整个分配单元，而不确定是否分配该分配单元中的页。如果 `reserved_threshold` 设置导致转储整个磁盘，则将忽略磁盘的 `allocation_threshold` 设置。此选项的缺省值为 40%。

示例

示例 1 此语句将转储整个数据库：

```
sp_dumpoptimize 'archive_space=maximum'
```

```
Backup Server: 4.172.1.1: The value of 'reserved pages threshold' has been set to 0%.  
Backup Server: 4.172.1.2: The value of 'allocated pages threshold' has been set to 0%.
```

示例 2 此语句只转储已分配的页，因此可产生最小的档案映像：

```
sp_dumpoptimize 'archive_space=minimum'
```

```
Backup Server: 4.172.1.1: The value of 'reserved pages threshold' has been  
set to 100%.  
Backup Server: 4.172.1.2: The value of 'allocated pages threshold' has been  
set to 100%.
```

示例 3 此语句将保留阈值设置为 85%，分配阈值设置为 40%：

```
sp_dumpoptimize 'archive_space=default'
```

```
Backup Server: 4.172.1.1: The value of 'reserved pages threshold' has been  
set to 85%.  
Backup Server: 4.172.1.2: The value of 'allocated pages threshold' has been  
set to 40%.
```

示例 4 转储其保留页的百分比大于或等于 60% 的数据库中的磁盘，而不读取该磁盘上的分配页。对于其余的磁盘，将读取分配页，并且使用为 `allocation_threshold` 最后设置的值。如果在启动 Backup Server 后未设置 `allocation_threshold`，则使用 `allocation_threshold` 的缺省值 40%:

```
sp_dumpoptimize 'reserved_threshold=60'
```

```
Backup Server: 4.172.1.3: The value of 'reserved pages threshold' has been set to 60%.
```

示例 5 将保留阈值设置为 85%。它不影响分配页阈值:

```
sp_dumpoptimize 'reserved_threshold=default'
```

```
Backup Server: 4.172.1.3: The value of 'reserved pages threshold' has been set to 85%.
```

示例 6 为保留页的百分比小于 `reserved_threshold` 最后设置值的那些磁盘读取分配页，如果分配单元中已分配了 80% 或更多的页，则将转储整个分配单元:

```
sp_dumpoptimize 'allocation_threshold=80'
```

```
Backup Server: 4.172.1.4: The value of 'allocated pages threshold' has been set to 80%.
```

示例 7 将分配页阈值设置为缺省值 40%。它不影响保留页阈值:

```
sp_dumpoptimize 'allocation_threshold=default'
```

```
Backup Server: 4.172.1.4: The value of 'allocated pages threshold' has been set to 40%.
```

示例 8 转储其保留页的百分比大于或等于 60% 的数据库中的磁盘，而不读取该磁盘上的分配页。对于其余的磁盘，将读取分配页；如果分配单元中已分配了 30% 或更多的页，则将转储整个分配单元:

```
sp_dumpoptimize 'reserved_threshold=60', 'allocation_threshold=30'
```

```
Backup Server: 4.172.1.3: The value of 'reserved pages threshold' has been set to 60%.
```

```
Backup Server: 4.172.1.4: The value of 'allocated pages threshold' has been set to 30%.
```

示例 9 这显示阈值的当前值:

```
sp_dumpoptimize
```

```
Backup Server: 4.171.1.1: The current value of 'reserved pages threshold' is 60%
```

```
Backup Server: 4.171.1.2: The current value of 'allocated pages threshold' is 30%.
```

用法

- 在使用 `sp_dumpoptimize` 设置阈值时，此阈值将作用于数据库驻留的每个设备。
- 当使用 `sp_dumpoptimize` 设置值时，这些值将立即生效，而无需重新启动 Backup Server。但是，只有重新启动 Backup Server，更改才会生效。重新启动 Backup Server 时，将使用缺省值。
- 如果您多次发出 `sp_dumpoptimize`，则以后的转储使用由最后的实例指定的阈值。例如，如果您先设置了 `reserved_threshold` 值，随后发出了 `archive_space=maximum`，则该值将覆盖以前为 `reserved_threshold` 设置的值。
- 通过在每次数据库转储之前更改 `sp_dumpoptimize` 值，不同数据库的转储可以使用不同的阈值。
- 最佳阈值随数据库的不同而不同。因此，转储的性能既取决于 I/O 配置，也取决于数据库中使用的空间量。DBA 可以通过使用不同的值试验转储过程并选择使转储时间最短的值，来确定数据库的适当配置。
- 可以对本地和远程转储使用 `sp_dumpoptimize`。
- `sp_dumpoptimize` 不影响事务日志转储或装载的性能。因此，在执行 `dump transaction`、`load database` 或 `load transaction` 操作前，不需要发出该命令。
- 如果发出不带任何参数的 `sp_dumpoptimize`，则将在客户端上显示阈值的当前值。
- 在档案设备吞吐量等于或大于所有数据库磁盘累积吞吐量的配置上，使用 `archive_space=maximum` 将使转储速度更快。但是，在档案设备吞吐量小于所有数据库磁盘累积吞吐量的配置上，使用该选项将使转储速度变慢。
- 可以缩写选项名和此过程的值，使之成为标识它们的唯一子字符串。例如，`ar = ma` 足以用来唯一标识选项 `archive_space=maximum`。
- 在选项字符串中，等号 (=) 两边可以不使用空格，也可以使用多个空格。
- 选项名和它们的值不区分大小写。

阈值

阈值的缺省值为：

- 保留的页：85%
- 分配页：40%

如果数据库设备段的保留页百分比：

- 大于或等于保留的阈值 — 则转储该设备上与此数据库有关的所有块。
- 小于保留的阈值 — 则 Backup Server 开始检查该设备上每个分配单元的分配百分比。如果累积分配百分比：
 - 小于分配阈值 — 则只转储写有数据的页
 - 大于分配阈值 — 则转储整个分配单元。

权限

只有系统管理员、数据库所有者或具有操作员角色的用户才能执行 sp_dumpoptimize。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关分配页的信息，请参见 《系统管理指南》。

命令 [dump database](#)、[dump transaction](#)、[load database](#)、[load transaction](#)

sp_encryption

说明 报告加密信息。

语法

```

sp_encryption help | helpkey
sp_encryption help | helpkey [, key_name | wildcard]
    [, all_dbs | key_copy | display_cols]
sp_encryption help | 'helpkey',
    {'master'|'dual master'} [, 'display_keys' | 'all_dbs']
sp_encryption 'help'[, 'servicekeyname' [, 'display_objs']]
sp_encryption 'helpextpasswd'
sp_encryption helpcol [, table_name | column_name ]
sp_encryption helpuser [, user_name | wildcard ][, key_copy]
sp_encryption 'mkey_startup_file'[, {'new_path' | 'default_location' | 'null'}
    [, {sync_with_mem | sync_with_qrm}]]
sp_encryption 'downgrade_kek_size' [, 'true'|'false']
sp_encryption system_encr_passwd, 'newpasswd' [, 'oldpasswd']

```

参数

helpkey

列出加密密钥属性，包括：

- 数据库是否包含加密密钥。
- 当由具有 `sso_role` 的用户、密钥管理者或 DBO 运行时，包括以下属性：密钥名称、密钥所有者、密钥长度、密钥算法、密钥类型、`pad`、初始化矢量、用于加密密钥的口令类型、是否已启用密钥恢复以及密钥副本的计数。输出按 `owner.keyname` 排序。当由非特权用户运行时，此命令会列出密钥名称、密钥所有者和密钥类型。

help

包含该参数以用于向后兼容。包含与 `helpkey` 相同的输出。

key_name

要调查的密钥的名称。列出为 `key_name` 定义的属性。如果省略 `key_name`，则会列出所有密钥的属性。

wildcard

列出当前数据库中与通配符模式匹配的密钥的属性。有关使用通配符的信息，请参见《参考手册：构件块》。

all_dbs

列出所有可用数据库中加密密钥的信息。只有 SSO 可以运行 `all_dbs`。

key_copy

列出当前数据库中指定密钥的所有用户副本。输出按 **key_owner.key_name** 排序。包括以下内容的相关信息：

- 基本密钥所有者。
- 密钥副本是否为恢复密钥副本。
- 副本所属的用户。
- 是否使用用户加密口令、登录口令或系统加密口令加密副本以用于登录关联（由登录访问权限指定）。

display_keys

与 **system_encr_passwd** 一起使用，以显示使用系统加密口令加密的密钥和密钥副本。与 **master** 或 **dual master** 一起使用，以显示使用主密钥或双主密钥加密的密钥和密钥副本。

您必须是系统安全员、密钥管理者或数据库所有者，才能运行 **sp_encryption helpkey, master | 'dual master', display_keys** 来显示受主密钥和双主密钥保护的加密密钥。

display_cols

显示密钥名称、当前数据库中的所有密钥（或与通配符匹配的密钥）以及密钥加密的列。当 SSO 包括 **display_cols** 时，它会显示所有可用数据库中由密钥加密的列。当没有 **sso_role** 的用户运行 **display_cols** 时，仅显示当前数据库中由密钥加密的列。数据按 **key_name**、**key_owner**、**database_name**、**table_owner**、**table_name** 和 **column_name** 排序。

master

报告有关主密钥的信息。

dual master

报告有关双主密钥的信息。

servicekeyname

设置为 **syb_extpasswdkey** 或 **syb_syscommkey%**。与 **display_objs** 一起使用来显示用服务密钥加密的对象。

display_objs

显示对象所有者。

您必须是系统安全员、密钥管理者或数据库所有者，才能运行 **sp_encryption helpkey, keyname, display_objs** 来显示当前数据库中受 **syb_extpasswdkey** 或 **syb_syscommkey** 保护的對象。

helpxtpasswd

显示状态列中的外部口令的加密状态。加密状态为以下状态之一：

- **FIPS Encryption**（FIPS 加密）— 口令由 `syb_extpasswdkey` 服务密钥使用与 FIPS 兼容的密码术算法进行保护
- **Needs Reset**（需要重置）— 指示系统删除了口令，您必须手动重置它。
- **Legacy Encryption**（旧式加密）— 口令是使用低于 15.7 版的 Adaptive Server 中的算法保护的。

您必须是系统安全员才能运行 `sp_encryption helpxtpasswd` 以检查外部口令的状态。

helpcol *column_name*

显示列名以及用于加密列的密钥。如果 SSO 包括 `helpcol`，则即使密钥不存在于当前数据库中，它也会输出该密钥的名称。如果非 SSO 用户包括 `helpcol`，则当密钥不存在于当前数据库中时，Adaptive Server 会输出密钥的 `keyid`，而省略 `key_name`。输出包括：`owner.table.column、database.owner.keyname`。该信息按 `owner.table.column` 排序。

helpuser

显示当前数据库中用户所拥有的或分配给当前数据库中用户的密钥。

mkey_startup_file

显示或设置主密钥启动文件名和路径。`sp_encryption` 将主密钥启动文件设置为 `new_path` 或缺省位置。如果指定 `null` 或不指定任何位置，`sp_encryption` 会显示当前主密钥启动文件名和路径。

sync_with_mem

（仅在 Cluster Edition 上可用）将存在于服务器内存中的主密钥加密密钥写入主密钥启动文件中。替换当前主密钥加密密钥（如果存在）。如果 `automatic master key access` 设置为 `off`，则 `sync_with_mem` 也会被禁用。

您必须是系统安全员，才能显示、设置或同步主密钥启动文件。

sync_with_qrm

将本地主密钥启动文件更新为仲裁设备中的版本。

您必须是系统安全员，才能显示、设置或同步主密钥启动文件。

downgrade_kek_size

显示或设置 downgrade kek size 配置。 true 指示 Adaptive Server 处于 downgrade kek size 模式， false 禁用此模式。

如果不指定任何参数， sp_encryption 会显示 downgrade_kek_size 的当前值。

您必须是系统安全员或密钥管理者才能运行此命令。

system_encr_passwd

显示当前数据库中使用系统加密口令加密的密钥和密钥副本。

system_encr_passwd, all_dbs

在设置系统加密口令的每个数据库中显示该口令的属性。输出按数据库名称排序。只有系统安全员能够运行此命令。如果未为所有数据库设置系统加密口令， Adaptive Server 将生成消息 19782:

尚未为所有可用数据库设置系统加密口令

示例

示例 1 使用 helpkey 参数显示当前数据库中的密钥信息。可以获取有关所有密钥或特定密钥的信息。 sp_encryption 的第二个参数提供密钥名称，可包含与 SQL 模式匹配的字符。如果您不是数据库所有者，也不具有 sso_role 或 keycustodian_role， sp_encryption 将显示较少的列。

当由 SSO、密钥管理者或 DBO 运行时，这会显示当前数据库中所有基本加密密钥的属性：

```
sp_encryption helpkey
```

| Key Name | Key Owner | Key Length | Key Algorithm | Key Type | Pad |
|-------------|----------------------------|--------------|-----------------|-----------------------|-----|
| Init Vector | Protected By | Key Recovery | # of Key Copies | | |
| tinnap_key | tinnap | 128 | AES | symmetric key | 0 |
| 1 | system encryption password | | | 0 | 0 |
| tinnap_key1 | tinnap | 128 | AES | symmetric default key | 0 |
| 1 | user Passwd | | | 1 | 3 |
| sample_key1 | dbo | 192 | AES | symmetric key | 1 |
| 1 | login Passwd | | | 1 | 2 |

当由用户 “tinnap” 运行时，这会显示当前数据库中所有基本加密密钥的下列属性：

```
sp_encryption helpkey
```

| Key Name | Key Owner | Key Type |
|-------------|-----------|-----------------------|
| tinnap_key | tinnap | symmetric key |
| tinnap_key1 | tinnap | symmetric default key |
| sample_key1 | dbo | symmetric key |

如果您不是系统安全员，也不具有 `keycustodian_role`，查询将显示您在当前数据库中拥有的所有基本密钥。如果未将 `user_name` 指定为第二个参数，查询将显示您拥有的基本密钥。

示例 2 当由 SSO、密钥管理者或 DBO 运行时，显示当前数据库中名称与 “tinnap%” 相似的所有基本加密密钥的属性：

```
sp_encryption helpkey, "tinnap%"
```

| Key Name | Key Owner | Key Length | Key Algorithm | Key Type |
|-------------|-------------|--------------------|---------------|-----------------------|
| Pad | Init Vector | Protected By | Key Recovery | # of Key Copies |
| tinnap_key | tinnap | 128 | AES | symmetric key |
| 0 | 1 | system encr passwd | 0 | 0 |
| tinnap_key1 | tinnap | 128 | AES | symmetric default key |
| 0 | 1 | user passwd | 1 | 3 |

当由用户 “tinnap” 运行时，会显示当前数据库中名称与 “tinnap%” 相似的加密密钥的以下属性：

```
sp_encryption helpkey, "tinnap%"
```

| Key Name | Key Owner | Key Type |
|-------------|-----------|-----------------------|
| tinnap_key | tinnap | symmetric key |
| tinnap_key1 | tinnap | symmetric default key |

示例 3 当由 SSO、密钥管理者或 DBO 在当前数据库中运行时，会显示基本加密密钥 `sample_key1` 的属性：

```
sp_encryption helpkey, sample_key1
```

| Key Name | Key Owner | Key Length | Key Algorithm | Key Type |
|-------------|-------------|--------------|---------------|-----------------|
| Pad | Init Vector | Protected By | Key Recovery | # of Key Copies |
| sample_key1 | dbo | 192 | AES | symmetric Key |
| 1 | 1 | Login | 1 | 2 |

当非特权用户 “tinnap” 运行此命令时，它会显示当前数据库中基本加密密钥 `sample_key1` 的以下属性：

```
sp_encryption helpkey, sample_key1
```

| Key Name | Key Owner | Key Type |
|-------------|-----------|--------------|
| sample_key1 | dbo | ymmetric key |

示例 4 显示所有可用数据库中所有基本加密密钥的属性（只有 SSO 才能运行此命令）：

```
sp_encryption helpkey, NULL, all_dbs
```

| Db.Owner.Keyname | Key Length | Key Algorithm | Key Type |
|------------------------------|--------------|----------------|-----------------------|
| Pad Init Vector Protected By | Key Recovery | #of Key Copies | |
| keydb.dbo.cc_key | 256 | AES | symmetric default key |
| 1 1 system encr passwd | | 0 | 0 |
| keydb.dbo.sample_key1 | 128 | AES | symmetric key |
| 0 0 system encr passwd | | 1 | 4 |
| keydb1.tinnap.tinnap_key | 128 | AES | symmetric key |
| 0 1 system encr passwd | | 0 | 0 |
| keydb1.tinnap.tinnap_key1 | 128 | AES | symmetric default key |
| 0 1 user password | | 1 | 3 |
| keydb1.dbo.sample_key1 | 192 | AES | symmetric key |
| 1 1 login passwd | | 1 | 2 |

示例 5 all_dbs 指示需要所有数据库的密钥信息。您必须具有 sso_role 才能使用 all_dbs 参数。

显示所有可用数据库中与 %key 相似的所有基本加密密钥的属性：

```
sp_encryption helpkey, '%key', all_dbs
```

| Db.Owner.Keyname | Key Length | Key Algorithm | Key Type |
|------------------------------|--------------|----------------|-----------------------|
| Pad Init Vector Protected By | Key Recovery | #of Key Copies | |
| keydb.dbo.cc_key | 256 | AES | symmetric default key |
| 1 1 system encr passwd | | 0 | 0 |
| keydb1.tinnap.tinnap_key | 128 | AES | symmetric key |
| 0 1 system encr passwd | | 0 | 0 |

示例 6 通过将 key_copy 用作第三个参数显示有关密钥副本的信息。为第二个参数的 keyname 输入空值而非具体值，可查看有关所有密钥副本的信息。在 keyname 中可以使用与模式匹配的字符（请参见示例 2）：

```
sp_encryption helpkey, tinnap_key1, key_copy
```

| Owner.Keyname | Assignee | Protected by | Key Recovery |
|--------------------|----------|--------------|--------------|
| tinnap.tinnap_key1 | joesmp | user passwd | 0 |
| tinnap.tinnap_key1 | samcool | user passwd | 1 |
| tinnap.tinnap_key1 | billyg | user passwd | 0 |

当由用户“joesmp”运行时，这会显示当前数据库中分配给用户“joesmp”的所有加密密钥副本，并且如果该用户是该密钥的所有者，则还会显示该密钥名称的所有密钥副本：

```
sp_encryption helpkey, tinnap_key1, key_copy
```

| Owner.Keyname | Assignee | Protected by | Key Recovery |
|--------------------|----------|--------------|--------------|
| tinnap.tinnap_key1 | joesmp | user passwd | 0 |

示例 7 使用 `display_cols` 参数可显示所有可用数据库中由当前数据库中的密钥加密的所有加密列。如果您不具有 `sso_role`，查询仅显示当前数据库中由当前数据库中的密钥加密的加密列。可使用模式匹配的字符或 `key_name` 作为第二个参数。如果将 `key_name` 的模式匹配字符用作 `sso_role`，查询将显示所有可用数据库中由模式匹配的 `key_name` 加密的所有加密列。如果使用 `key_name` 作为第二个参数，并且您具有 `sso_role`，将显示所有可用数据库中由指定 `key_name` 加密的所有加密列：

```
sp_encryption helpkey, null, display_cols
```

| Key Name | Key Owner | Database Name | Table Owner | Table Name | Column Name |
|-------------|-----------|---------------|-------------|------------|-------------|
| tinnap_key | tinnap | testdb1 | tinnap | t3 | c3 |
| tinnap_key1 | tinnap | testdb1 | tinnap | t4 | c4 |
| sample_key1 | dbo | coldb | dbo | t1 | c1 |
| sample_key1 | dbo | coldb | billyg | t2 | c2 |

示例 8 显示当前数据库中使用系统加密口令加密的所有密钥和密钥副本。如果您不具有上述特权，查询将显示由用户拥有或分配给相应用户的使用系统加密口令加密的密钥：

```
sp_encryption helpkey, system_encr_passwd, display_keys
```

| Owner.Keyname | Assignee |
|-----------------|----------|
| dbo.cc_key | NULL |
| dbo.sample_key1 | NULL |
| dbo.sample_key1 | tinnap |

示例 9 在由数据库所有者或具有 `keycustodian_role` 或 `sso_role` 的用户运行时，`helpuser` 参数会显示当前数据库中由用户拥有的所有基本密钥。

```
sp_encryption helpuser
```

| Owner.Keyname | Protected by |
|--------------------|--------------------|
| tinnap.tinnap_key | system encr passwd |
| tinnap.tinnap_key1 | user passwd |
| dbo.sample_key1 | login passwd |

如果用户 “tinnap” 运行此命令，则会列出当前数据库中由此用户所拥有的所有基本密钥：

```
sp_encryption helpuser

Owner.Keyname          Protected by
-----
tinnap.tinnap_key     system encr passwd
tinnap.tinnap_key1    user passwd
```

示例 10 数据库所有者或具有 `keycustodian_role` 或 `sso_role` 的用户可结合使用 `key_copy` 和 `helpuser` 参数显示当前数据库中分配给一个或多个用户的密钥副本。可在 `user` 参数中使用模式匹配的字符。此查询显示当前数据库中所有用户的密钥副本：

```
sp_encryption helpuser, NULL, key_copy
Owner.Keyname          Assignee          Protected by      Key Recovery
-----
dbo.sample_key1       tinnap           login passwd     0
tinnap.tinnap_key1   joesmp          user passwd     0
dbo.sample_key1       joesmp          login passwd     1
tinnap.tinnap_key1   samcool         user passwd     1
tinnap.tinnap_key1   billyg          user passwd     0
```

如果您不是数据库所有者，也不具有 `keycustodian_role` 或 `sso_role`，此查询将显示您拥有的所有密钥的副本，以及其他密钥所有者分配给您的密钥副本。例如，当用户 “tinnap” 运行此查询时：

```
sp_encryption helpuser, NULL, "key_copy"
Owner.Keyname          Assignee          Protected by      Key Recovery
-----
dbo.sample_key1       tinnap           login passwd     0
tinnap.tinnap_key1   joesmp          user passwd     0
tinnap.tinnap_key1   samcool         user passwd     1
tinnap.tinnap_key1   billyg          user passwd     0
```

示例 11 如果您是数据库所有者或是具有 `keycustodian_role` 或 `sso_role` 的用户，`helpcol` 将显示当前数据库中的所有加密列以及用于加密列的密钥。如果您不具有上述特权，则当加密密钥位于其它数据库中时，`helpcol` 将显示 `keyid` 而非 `key_name`：

```
sp_encryption helpcol

Owner.Table.Column    Db.Owner.Keyname
-----
dbo.t1.c1             keydb1.dbo.sample_key1
```



```

billyg.t2.c2                keydb.dbo.sample_key1
tinnap.t3.c3                coldb.dbo.sample_key2

```

示例 12 如果将 `helpcol` 参数与 `table_name` 和 `column_name` 参数结合使用，将显示给定表中的所有加密列或特定加密列。当由具有 `sso_role` 的用户运行时，以下查询将显示当前数据库中表 `t3` 中的所有加密列，以及用于在所有可用数据库中加密列的密钥。在由不具有 `sso_role` 的用户运行时，如果密钥不在当前数据库中，此查询将显示该密钥的 ID 而非其名称。第二个参数可具有 `[database_name.][table_name.][column_name]` 的组合：

```

sp_encryption helpcol, t3

Owner.Table.Column          Db.Owner.Keyname
-----
tinnap.t3.c3                coldb.dbo.sample_key2

```

示例 13 显示每个数据库中的系统加密口令属性（您必须具有 `sso_role` 才能运行此查询）：

```

sp_encryption helpkey, system_encr_passwd, all_dbs
Database  Type of system_encr_passwd  Last modified by  Date
-----
master    persistent                  sa Aug 26 2008 10:05AM

```

示例 14 显示当前数据库中所有用主密钥加密的加密密钥（您必须具有 `sso_role`、`keycustodian_role` 或是数据库所有者，才能运行此查询）：

```

sp_encryption helpkey, 'master', display_keys
Owner.Keyname          Assignee
-----
user1.key_dual         NULL
user1.key_mst          NULL
user4.key_dC_pwd       NULL
user4.key_dC_pwd       user5
user4.key_dC_pwd       user6
user4.key_dC_pwd       KC_tdb1

```

示例 15 显示为当前服务器配置当前主密钥启动文件的名称和位置：

```

sp_encryption mkey_startup_file
Msg 19956, Level 16, State 1: Procedure 'sp_encryption',
Line 298: The current master key startup file
is: '/sybase/release/ASE-
150/init/ase_encrcols_mk_1157.dat'.

```

示例 16 显示用密钥 `syb_syscommkey_123456` 加密并被 `user1` 和 `user2` 拥有的三个存储过程：

```
sp_encryption helpkey, "syb_syscommkey%", display_objs
Key Name                               Key Owner       Database Name
-----                               -
Object owner                           Object Name
-----                               -
syb_syscommkey_1234567890ab            dbo             testdb
      user1                            sp_mysproc1
syb_syscommkey_abcdefghijkl123456      dbo             testdb
      user1                            sp_mysproc2_
syb_syscommkey_ABCDEF123456           dbo             testdb
      user2                            sp_mysproc3
```

用法

- 向运行 `sp_encryption` 的用户授予的特权将决定输出的内容。有关详细信息，请参见第 289 页的“使用限制”。
- 如果运行 `sp_encryption helpkey` 并且数据库中没有密钥，则会显示一条信息性消息。
- 必须指定 `key_copy` 参数，才可获取有关密钥副本的信息。如果未指定 `key_copy` 参数，则 `sp_encryption` 仅返回有关基本密钥的信息。
- 如果 `sp_encryption helpkey, keyname, key_copy` 中的 `keyname` 为 `NULL`，则它会为 `SSO`、密钥管理者或 `DBO` 列出当前数据库中的所有密钥副本。如果没有特权的用户运行此命令，则它会列出当前数据库中分配给该用户的所有密钥副本，以及当前数据库中该用户所拥有密钥的所有密钥副本。
- 对于 `sp_encryption helpcol, column_name` 使用 `name.name.name` 格式，其中：
 - `name` — 如果 `sp_encryption` 找不到此名称的表，则它会查找具有此名称的所有列。
 - `name.name` — 为 `owner.table`。如果 `sp_encryption` 找不到此名称的表，则会查找名称为 `table.column` 的单个列。
 - `name.name.name` — 为 `owner.table.name`。

对于当前数据库中由这些规则确定的所有列，`sp_encryption` 会显示列名以及用于加密列的密钥。

`sp_encryption helpcol, column_name` 的输出为 `owner.table.column` 和 `db.owner.keyname`。当由非 `SSO` 用户运行时，`keyname` 表示为 `database.keyid`，并且密钥存在于与加密列不同的数据库中。结果集按 `owner.table.column` 排序。

使用限制

- 只有 SSO 才能运行 `sp_encryption helpkey, [,keyname | wildcard], all_dbs`，以获取所有数据库中密钥的属性。如果不具有 `sso_role` 的用户运行此命令，则他们会收到“unauthorized user”的错误消息。如果密钥没有限定 `keyname` 或 `wildcard`，则 Adaptive Server 会返回一条消息，指出“`There are no encryption keys (key copies) like keyname in all databases`”。
- 当 SSO 运行 `sp_encryption helpkey, keyname, display_cols` 时，它会列出所有可用数据库中由 `keyname` 加密的所有列。如果没有特权的用户运行此命令，则它会列出当前数据库中由 `keyname` 加密的列。

如果 SSO 运行 `sp_encryption helpkey, keyname, display_cols` 并且 `keyname` 值为 NULL，则它会显示所有可用数据库中的所有加密列。当没有特权的用户运行此命令时，它会显示当前数据库中的所有加密列。

- 如果 SSO、密钥管理者或 DBO 在没有为 `helpuser` 参数指定 `user_name` 和 `key_copy` 的情况下运行 `sp_encryption helpuser, user_name, key_copy`，则它会列出当前数据库中所有用户拥有的所有基本密钥。如果没有特权的用户或未指定 `user_name` 或 `key_copy` 的情况下运行 `sp_encryption`，则它会显示当前用户所拥有的基本密钥。

如果任何用户运行 `sp_encryption helpuser, user_name`，则它会列出 `owner.keyname` 所拥有的所有基本密钥。如果没有特权的用户运行该命令并且不拥有基本密钥，则 Adaptive Server 会显示信息性消息以指出此情况。

如果 SSO、密钥管理者或 DBO 运行 `sp_encryption helpuser, user_name, key_copy`，则它会列出分配给 `user_name` 的密钥副本。如果没有特权的用户发出此命令，则它会列出当前数据库中分配给该用户的密钥副本以及该用户所拥有的密钥的密钥副本，并在结果集中包含以下列：`Owner.Keyname`、`Assignee`、`Type of Password` 和 `Key Recovery`。输出按 `Assignee` 排序。

如果 `sp_encryption helpuser user_name, key_copy` 的 `user_name` 为 NULL，则它会为 SSO、密钥管理者或 DBO 列出当前数据库中的所有密钥副本。对于没有特权的用户，它会列出当前数据库中分配给该用户的所有密钥副本，以及该用户所拥有的密钥的密钥副本。

- 当 SSO、密钥管理者或 DBO 运行 `sp_encryption helpkey, keyname, key_copy` 时，它会列出当前数据库中 `keyname` 的密钥副本。如果没有特权的用户运行此命令，则它会列出分配给该用户的该 `keyname` 的密钥副本，如果该用户是密钥所有者，则会列出该 `keyname` 的密钥副本。

- SSO、密钥管理者和 DBO 可以运行 `sp_encryption helpkey, system_encr_passwd, display_keys`，以接收当前数据库中由系统加密口令加密的所有密钥和密钥副本的相关信息。没有特权的用户会收到当前数据库中他们所拥有的或已分配到的基本加密密钥或密钥副本的相关信息。仅当密钥副本是为登录关联而创建时，才会使用系统加密口令进行加密。输出按 `owner.keyname` 排序。

权限

只有 SSO、密钥管理者和 DB 才能运行 `sp_encryption helpkey, master | 'dual master', display_keys` 来显示受主密钥和双主密钥保护的加密密钥。

只有 SSO、密钥管理者和 DB 才能运行 `sp_encryption helpkey, keyname, display_objs` 来显示当前数据库中受服务密钥 `syb_extpasswdkey` 或 `syb_syscommkey` 保护的對象。

只有 SSO 和密钥管理者才能运行 `sp_encryption downgrade_kek_size` 来设置或重置 “`downgrade_kek_size`” 选项。

只有 SSO 才能运行 `sp_encryption mkey_startup_file {'newpath' | 'default_location' | 'null'} [, {sync_with_mem | sync_with_qrm}]` 来显示、设置、同步主密钥启动文件。

只有 SSO 才能运行 `sp_encryption helpextpasswd` 来检查外部口令的状态。

sp_engine

| | |
|----|---|
| 说明 | 使您可以让引擎联机或脱机。在线程化模式下，使用 <code>alter thread pool</code> 可让引擎联机。 |
| 语法 | <pre>sp_engine {"online" [offline can_offline] [, engine_id] ["shutdown", engine_id]}</pre> |
| 参数 | <p>"online"</p> <p>使引擎联机。 <code>sp_configure "max online engines"</code> 的值必须大于当前的联机引擎数。因为 <code>"online"</code> 是保留关键字，因此必须使用引号。</p> <p>在线程化模式中， <code>online</code> 将 <code>syb_default_pool</code> 的线程计数增加 1。</p> <p>offline</p> <p>使引擎脱机。还可以使用 <code>engine_id</code> 参数指定让特定的引擎脱机。</p> <p>在线程化模式中， <code>offline</code> 将 <code>syb_default_pool</code> 的线程计数减少 1。</p> <p>can_offline</p> <p>返回有关是否可以让某个引擎脱机的信息。如果其状态是 <code>online</code>，则 <code>can_offline</code> 返回具有此引擎密切连接的 Adaptive Server 任务（例如，在 <code>Omni</code> 或 <code>java.net</code> 任务中）。如果未指定 <code>engine_id</code>，则该命令所指出的状态是 <code>sysengines</code> 中具有最高 <code>engine_id</code> 的引擎的状态。</p> <p>在线程化模式中，只有在引擎总数小于 <code>syb_default_pool</code> 中的线程总数并且 <code>syb_default_pool</code> 中的线程总数大于或等于 2 时， <code>can_offline</code> 才会成功。</p> <p>engine_id</p> <p>引擎的 ID。 <code>engine_id</code> 参数是可选的。如果未指定 <code>engine_id</code>，则 <code>sp_engine</code> 对 <code>sysengines</code> 中找到的引擎的值使用 <code>engine_id</code> 的递增或递减值。也就是说，如果系统使用引擎 0、1、2 和 3，并且您未指定引擎 ID，则 <code>sp_engine</code> 先将引擎 ID 3 脱机，然后将引擎 ID 2 脱机，依此类推。</p> <p>此参数在线程化模式中将被忽略。</p> <p>"shutdown"</p> <p>强制引擎脱机。如果任何任务具有此引擎的密切连接，则在 5 分钟等待后将注销这些任务。因为 <code>shutdown</code> 是保留关键字，所以必须使用引号。</p> |

示例

示例 1 使引擎 1 联机。消息是平台特定的（本例使用 Sun Solaris）：

```
sp_engine "online", 1
02:00000:00000:2001/10/26 08:53:40.61 kernel Network and device connection
limit is 3042.
02:00000:00000:2001/10/26 08:53:40.61 kernel SSL Plus security modules
loaded successfully.
02:00000:00000:2001/10/26 08:53:40.67 kernel engine 2, os pid 8624 online
02:00000:00000:2001/10/26 08:53:40.67 kernel Enabling Sun Kernel
asynchronous disk I/O strategy
00:00000:00000:2001/10/26 08:53:40.70 kernel ncheck: Network fc0330c8
online
```

示例 2 介绍使引擎脱机的步骤，当前正在运行具有该引擎密切连接的任务：

```
select engine, status from sysengines

engine      status
-----      -
0           online
1           online
2           online
3           online
```

如果要让引擎 1 脱机：

```
sp_engine offline, 1

The following task(s) will affect the offline process:
spid: 19 has outstanding ct-lib connections.
```

然后运行与上面相同的查询，它现在显示引擎 1 处于脱机状态：

```
select engine, status from sysengines

engine      status
-----      -
0           online
1           in offline
2           online
3           online
```

具有引擎 1 密切连接的任务一完成，Adaptive Server 就会向错误日志发出一个类似以下内容的消息：

```
02:00000:00000:2001/10/26 09:02:09.05 kernel engine 1, os pid
8623 offline
```

示例 3 确定是否可以使引擎 1 脱机：

```
sp_engine can_offline, 1
```

示例 4 使引擎 1 脱机:

```
sp_engine offline, 1
```

Adaptive Server 最终返回类似于以下内容的消息:

```
01:00000:00000:2001/11/09 16:11:11.85 kernel Engine 1 waiting for
affinitated process(es) before going offline
01:00000:00000:2001/11/09 16:11:11.85 kernel Process 917518 is preventing
engine 1 going offline
00:00000:00000:2001/11/09 16:16:01.90 kernel engine 1, os pid
21127 offline
```

示例 5 关闭引擎 1:

```
sp_engine shutdown, 1
```

用法

- 由于 `sp_engine` 仅在进程模式中工作, 因此, 如果您在线程化模式中运行 `sp_engine`, Adaptive Server 会发出错误消息。在线程化模式中使用 `alter thread pool`。
- 不能使引擎 0 变为脱机或将其关闭。
- 使用以下查询, 您可以确定引擎的状态, 以及哪些引擎当前处于联机状态:

```
select engine, status from sysengines
where status = "online"
```

- `online` 和 `shutdown` 是关键字, 必须用引号将它们引起来。
- 使引擎联机的前提条件是: `max online engines` 大于当前具有 `online` 状态的引擎的数目, 并且有足够的 CPU 可用来支持额外的引擎。
- `sp_engine` 可以在没有打开的事务时在使用链式事务模式的会话中运行:
- 如果存在某些服务器进程具有与该引擎的密切连接, 则 `engine offline` 命令可能失败或不立即生效。

使用 `sp_engine "offline"` 与 `sp_engine "shutdown"`

有时, 当您使用 `sp_engine "ffline"` 时, 引擎不会立即脱机, 而看上去是在引擎表中处于“休眠”状态。这种情况是不能将附加到引擎的进程迁移到其它引擎导致的。当出现这种情况时, 引擎不会接受新的工作, 并消耗最低的 CPU 周期。当妨碍 `engine offline` 完成的进程结束或可进行迁移时, 引擎将从休眠状态变为完全脱机状态, 并从引擎表中消失。

`sp_engine "shutdown"` 是 `offline` 命令的一个强制性更强的版本。 `sp_engine "shutdown"` 过程将主动注销妨碍引擎脱机的任何进程, 强制引擎关闭。

但是，如果对具有 Client Library™ 或 Java 连接的引擎使用 sp_engine "shutdown"，则会看到：

```
Engine has outstanding ct-lib/java connections and
cannot be offlined.
```

当出现这种情况时，请每隔几分钟就重复执行该命令，直到连接消失，这时便可以关闭引擎了。

权限

您必须是系统管理员才能使引擎联机或脱机。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

sp_estspace

- 说明** 估计表及其索引所需的空间量以及创建索引需要的时间。
- 语法** `sp_estspace table_name, no_of_rows, fill_factor, cols_to_max, textbin_len, iosec, page_size`
- 参数**
- table_name**
是表的名称。它必须已存在于当前数据库中。
- no_of_rows**
是估计的表将包含的行数。
- fill_factor**
是索引填充因子。缺省值为空值，这意味着 Adaptive Server 使用其缺省填充因子。
- cols_to_max**
是用逗号分隔的可变长度列的列表，您要使用其中的最大长度，而不是平均长度。缺省值是可变长度列的平均声明长度。
- textbin_len**
是所有 text 和 image 列每行的长度。缺省值是 0。只有当表存储 text 或 image 数据时，才需要提供值。text 和 image 列存储在单独的数据页集中，与表的其它数据的存储位置不同。实际表行存储指向 text 或 image 值的指针。sp_estspace 另外提供一行有关行的 text 或 image 页大小的信息。
- iosec**
是此计算机上每秒磁盘 I/O 的数目。缺省值是每秒 30 次 I/O。
- pagesize**
如果将表迁移到指定页大小的服务器中，此参数允许估计给定表及其所有索引需要的空间。可以指定页大小（2048、4096、8192、16384 或 2 K、4 K、8 K、16 K）或指定 NULL 以使用当前的页大小。如果不使用“K”作为单位指示符，则 pagesize 的缺省单位是字节。因为页分配为各种对象分配了相同大小的页，所以 page_size 值适用于所有页类型（index、data、text 等）。
- 示例** **示例 1** 计算 titles 表及其索引的空间要求以及创建索引需要的时间。行数为 10,000，填充因子为 50%，使用列的最大大小计算两个可变长度的列，并且磁盘 I/O 速度为每秒 25 次 I/O：

```
sp_estspace titles, 10000, 50, "title,notes", 0, 25
```

| name | type | idx_level | Pages | Kbytes |
|--------|------------|-----------|-------|--------|
| titles | data | 0 | 3364 | 6728 |
| titles | text/image | 0 | 0 | 0 |

| | | | | |
|------------|--------------|---|------|------|
| titleidind | clustered | 0 | 21 | 43 |
| titleidind | clustered | 1 | 1 | 2 |
| titleind | nonclustered | 0 | 1001 | 2002 |
| titleind | nonclustered | 1 | 54 | 107 |
| titleind | nonclustered | 2 | 4 | 8 |
| titleind | nonclustered | 3 | 1 | 2 |

Total_Mbytes

8.68

| name | type | total_pages | time_mins |
|------------|--------------|-------------|-----------|
| titleidind | clustered | 3386 | 13 |
| titleind | nonclustered | 1060 | 5 |
| titles | data | 0 | 2 |

示例 2 使用 `au_pix` 表中的现有 `image` 数据的平均长度计算具有 1000 行的表的大小。还可以将该大小作为常量提供:

```
declare @i int
select @i = avg(datalength(pic)) from au_pix
exec sp_estspace au_pix, 1000, null, null, 16, @i
```

au_pix has no indexes

| name | type | idx_level | Pages | Kbytes |
|--------|------------|-----------|-------|--------|
| au_pix | data | 0 | 31 | 63 |
| au_pix | text/image | 0 | 21000 | 42000 |

Total_Mbytes

41.08

示例 3 计算具有 50,000 行的 `titles` 表的大小，所有其它值都使用缺省值:

```
sp_estspace titles, 50000
```

| name | type | idx_level | Pages | Kbytes |
|------------|--------------|-----------|-------|--------|
| titles | data | 0 | 4912 | 9824 |
| titleidind | clustered | 0 | 31 | 61 |
| titleidind | clustered | 1 | 1 | 2 |
| titleind | nonclustered | 0 | 1390 | 2780 |
| titleind | nonclustered | 1 | 42 | 84 |
| titleind | nonclustered | 2 | 2 | 4 |
| titleind | nonclustered | 3 | 1 | 2 |

```
Total_Mbytes
```

```
-----
12.46
```

| name | type | total_pages | time_mins |
|------------|--------------|-------------|-----------|
| titleidind | clustered | 4943 | 19 |
| titleind | nonclustered | 1435 | 8 |

示例 4 此示例在将聚簇索引添加到 `blurbs` 表后运行:

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 6, null, null, 16, @i, "16k"
```

| name | type | idx_level | Pages | Kbytes |
|------------|------------|-----------|-------|--------|
| blurbs | data | 0 | 8 | 128 |
| blurbs | text/image | 0 | 6 | 96 |
| blurbs_ind | clustered | 0 | 1 | 16 |
| blurbs_ind | clustered | 1 | 1 | 16 |

```
Total_Mbytes
```

```
-----
0.25
```

| name | type | total_pages | time_mins |
|------------|-----------|-------------|-----------|
| blurbs_ind | clustered | 10 | 0 |
| blurbs | data | 6 | 0 |

此示例运行在 2 K 的服务器上，并指示在将 `blurbs` 表迁移到 16 K 服务器后，它需要 0.25 MB。下面是在 16K 服务器上运行的同一查询，它将检验 0.25MB 的空间要求：

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 6, null, null, 16, @i, "16k"
```

| name | type | idx_level | Pages | Kbytes |
|------------|------------|-----------|-------|--------|
| blurbs | data | 0 | 8 | 128 |
| blurbs | text/image | 0 | 6 | 96 |
| blurbs_ind | clustered | 0 | 1 | 16 |
| blurbs_ind | clustered | 1 | 1 | 16 |

```
Total_Mbytes
-----
0.25
```

| name | type | total_pages | time_mins |
|------------|-----------|-------------|-----------|
| blurbs_ind | clustered | 10 | 0 |
| blurbs | data | 6 | 0 |

示例 5 估计如果在 2 K 的服务器上 blurbs 表有 1000 行，它将需要 1.99 MB 的空间：

```
declare @i int
select @i = avg(datalength(copy)) from blurbs
exec sp_estspace blurbs, 1000, null, null, 16, @i, "2k"
```

| name | type | idx_level | Pages | Kbytes |
|------------|------------|-----------|-------|--------|
| blurbs | data | 0 | 16 | 32 |
| blurbs | text/image | 0 | 1000 | 2000 |
| blurbs_ind | clustered | 0 | 1 | 2 |
| blurbs_ind | clustered | 1 | 1 | 2 |

```
Total_Mbytes
-----
1.99
```

| name | type | total_pages | time_mins |
|------------|-----------|-------------|-----------|
| blurbs_ind | clustered | 18 | 0 |
| blurbs | data | 1000 | 0 |

用法

- 若要估计表及其索引要求的空间量，请执行以下操作：
 - a 创建表。
 - b 创建表上的所有索引。
 - c 根据需要给定表名、表的估计行数和可选参数，运行 `sp_estspace`。

不需要将数据插入表中。 `sp_estspace` 将使用系统表中的信息（而不是表中数据的大小）来计算表和索引的大小。

- 如果数据库中设置了 `auto identity` 选项，则对于在没有指定 `primary` 键、`unique` 约束或 `IDENTITY` 列的情况下创建的每个新表，Adaptive Server 将自动在其中定义一个 10 位数字的 `IDENTITY` 列。若要估计该列需要的额外空间的大小，请执行以下操作：
 - a 在 `master` 数据库中，使用 `sp_dboption` 打开数据库的 `auto identity` 选项。
 - b 创建表。
 - c 在表上运行 `sp_estspace` 并记录结果。
 - d 删除表。
 - e 关闭数据库的 `auto identity` 选项。
 - f 重新创建表。
 - g 在表上重新运行 `sp_estspace` 并记录结果。
- 有关表或列的信息，请使用 `sp_help tablename`。

权限

任何用户都可以执行 `sp_estspace`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `create index`、`create table`

系统过程 `sp_dboption`、`sp_help`

sp_export_qpgroup

| | |
|----|---|
| 说明 | 将指定的用户和抽象计划组的所有计划导出到用户表中。 |
| 语法 | <code>sp_export_qpgroup <i>usr</i>, <i>group</i>, <i>tab</i></code> |
| 参数 | <p><i>usr</i></p> <p>是拥有要导出的抽象计划的用户的名称。</p> <p><i>group</i></p> <p>是包含要导出的计划的抽象计划组的名称。</p> <p><i>tab</i></p> <p>是要向其中复制计划的表的名称。该表必须是当前数据库中的表。可以用 <code>dbname.tablename</code> 格式指定数据库名，但不能指定所有者名。对于长标识符，总长度不得超过 255 个字符。</p> |
| 示例 | 创建一个名为 <code>moveplans</code> 的表，该表包含 <code>ap_stdout</code> 组中由用户 “freidak” 拥有的所有计划： |

```
sp_export_qpgroup freidak, ap_stdout, "tempdb..moveplans"
```

| | |
|----|---|
| 用法 | <ul style="list-style-type: none"> <code>sp_export_qpgroup</code> 将计划从抽象计划组复制到用户表中。使用 <code>sp_import_qpgroup</code> 可以在服务器和数据库之间复制抽象计划组或将用户 ID 指派给复制的计划。 在运行 <code>sp_export_qpgroup</code> 之前，指定的用户表名不会存在。该表是用与 <code>sysqueryplans</code> 的结构相同的结构创建的。 <code>sp_export_qpgroup</code> 使用 <code>select...into</code> 创建表以存储复制的计划。若要使用 <code>sp_export_qpgroup</code> 或在 <code>tempdb</code> 中创建表，您必须使用 <code>sp_dboption</code> 来启用 <code>select into/bulkcopy/pllsort</code>。 |
|----|---|

权限 只有系统管理员或数据库所有者才能执行 `sp_export_qpgroup`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_copy_all_qplans](#)、[sp_copy_qplan](#)、[sp_dboption](#)、[sp_import_qpgroup](#)

sp_extendsegment

| | |
|----|--|
| 说明 | 将段的范围扩展到其它数据库设备。 |
| 语法 | <code>sp_extendsegment segname, dbname, devname</code> |
| 参数 | <p>segname 是以前用 <code>sp_addsegment</code> 定义的现有段的名称。</p> <p>dbname 是要在其上扩展段的数据库的名称。 <i>dbname</i> 必须是当前数据库的名称。</p> <p>devname 是要添加到当前数据库设备范围的数据库设备的名称，该范围已包括在 <i>segname</i> 中。</p> |
| 示例 | 在数据库设备 <code>dev2</code> 上扩展数据库 <code>pubs2</code> 的段 <code>indexes</code> 的范围： <pre>sp_extendsegment indexes, pubs2, dev2</pre> |
| 用法 | <ul style="list-style-type: none">• 可以将一个段扩展到多个数据库设备上。• 只能以单用户模式为 <code>logsegment</code> 系统段执行 <code>sp_extendsegment</code>。• 如果扩展 <code>logsegment</code> 段，则将删除该设备上的任何其它段，该设备专门供日志段使用。• 当扩展 <code>logsegment</code> 段时， <code>Adaptive Server</code> 将重新计算它的最后机会阈值。• 若要将段与数据库设备关联，请创建或更改引用该设备的数据库。数据库设备可以有多个与之关联的段。• 在定义了段之后，您可以在 <code>create table</code> 和 <code>create index</code> 命令中使用它将表或索引放置在段上。如果在特定段上创建表或索引，则该表或索引后续的数据也位于该段上。 |
| 权限 | 只有数据库所有者或系统管理员才能执行 <code>sp_extendsegment</code> 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [alter database](#)、[create index](#)、[create table](#)

系统过程 [sp_addsegment](#)、[sp_dropsegment](#)、[sp_helpdb](#)、[sp_helpdevice](#)、[sp_helpsegment](#)、[sp_placeobject](#)

sp_extengine

说明 启动和停止 EJB Server。显示 EJB Server 的状态信息。

语法 sp_extengine 'ejb_server', '{ start | stop | status }'

参数 *ejb_server*
是 EJB Server 的逻辑名。

start
启动 EJB Server。

stop
关闭 EJB Server。

status
显示有关 EJB Server 的状态信息。

示例 **示例 1** 通知用户 EJB Server SYB_EJB 正在运行：

```
sp_extengine 'SYB_EJB', 'status'
Enterprise java bean server is up and running.
```

示例 2 关闭 EJB Server SYB_EJB：

```
sp_extengine 'SYB_EJB', 'stop'
```

用法

- 您必须具有有效的 Adaptive Server EJB Server 站点许可证才能使用 sp_extengine。

权限 只有系统管理员才能执行 sp_extengine。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **文档** 有关详细信息，请参见《EJB Server 用户指南》。

sp_extrapwdchecks

| | |
|----|---|
| 说明 | 一个自定义存储过程，它可以包含用于口令复杂程度检查的用户定义的逻辑。您可以根据安全需要对 <code>sp_extrapwdchecks</code> 进行配置。将 <code>sp_extrapwdchecks</code> 安装在 <code>master</code> 数据库中。 |
| 语法 | <code>sp_extrapwdchecks caller_password, new_password, login_name</code> |
| 参数 | <p><i>caller_password</i> 指定当前口令。</p> <p><i>new_password</i> 指定要设置的新口令。</p> <p><i>login_name</i> 指定与要更改或要添加的口令相关联的登录名。</p> |
| 用法 | <code>sp_extrapasswordchecks</code> 必须使用 <code>raiserror</code> 向 Adaptive Server 报告错误。使用 <code>sp_addmessage</code> 为 Adaptive Server 中的此故障添加错误消息。 |

注释 不要使用 `raiserror` 获取正常行为。`raiserror` 会更新 `@@error` 全局变量。每次执行 T-SQL 语句（包括 `print` 和 `if`）时也会更新 `@@error`。如果 `raiserror` 后跟有任何 T-SQL 语句，`@@error` 将被覆盖，而且如果 `raiserror` 后跟有任何 TSQL 语句，`sp_extrapwdchecks` 将无法返回失败口令的错误。

sp_familylock

说明 报告由并行执行某个语句的一系列进程（协调进程及其工作进程）持有的所有锁的相关信息。

语法 sp_familylock [fpid1 [, fpid2]]

参数 *fpid1*
是 master.dbo.sysprocesses 表中工作进程系列的系列标识符。运行 [sp_who](#) 或 [sp_lock](#) 可以获得父进程的 *spid*。

fpid2
是其它锁的 Adaptive Server 进程 ID 号。

示例 显示 fid 为 5 的系列所有成员持有的锁的相关信息：

```
sp_familylock 5

fid spid locktype      table_id page dbname class          context
---- -
5   5   Sh_intent      176003658 0   userdb Non cursor lock Sync-pt duration request
5   5   Sh_intent-blk  208003772 0   userdb Non cursor lock Sync-pt duration request
5   6   Sh_page        208003772 3972 userdb Non cursor lock Sync-pt duration request
5   7   Sh_page        208003772 3973 userdb Non cursor lock Sync-pt duration request
5   8   Sh_page        208003772 3973 userdb Non cursor lock Sync-pt duration request
```

- 用法**
- 不带参数的 `sp_familylock` 将报告有关当前持有锁的系列中所有进程的信息。该报告与 `sp_lock` 的输出内容相同；但 `sp_familylock` 允许基于系列 ID 而不是进程 ID 生成报告。它可用于检测系列死锁。
 - 使用 `object_name` 系统函数可以通过表的 ID 号得出表名。
 - “locktype”列指示锁是共享锁（前缀为“Sh”）、排它锁（前缀为“Ex”）还是更新锁，以及锁是被表（“table”或“intent”）持有还是被页（“page”）持有。
“locktype”列的“blk”后缀表示该进程目前阻塞了需要获得锁的另一个进程。在此进程结束后，其它进程将继续执行。“demand”后缀指示进程正在尝试获得排它锁。
 - “class”列表示锁是否与游标关联。它显示下列内容之一：
 - “Non cursor lock”指示锁没有与游标关联。
 - “Cursor Id number”指示锁与该 Adaptive Server 进程 ID 的游标 ID 号相关联。
 - 游标名表示锁与游标 `cursor_name` 相关联，该游标属于执行 `sp_lock` 的当前用户。

- “fid” 列标识锁所属的系列（包括协调进程及其工作进程）。“fid” 的值为：
 - 零值指示 spid 代表的任务是串行执行的。它不参与并行执行。
 - 非零值表示持有锁的任务 (spid) 是并行执行语句的进程系列的成员（用 “fid” 标识）。如果值等于 spid，则表明该任务是并行执行查询的进程系列中的协调进程。
- “context” 列标识锁的环境。同一系列中的工作进程具有相同的环境值。“context” 的值为：
 - “NULL” 表示持有该锁的任务在执行串行查询，或者在事务隔离级别 1 上执行并行查询。
 - “FAM_DUR” 表示持有该锁的任务在完成查询之前一直持有锁。
 以下情况下，锁的环境可以是 “FAM_DUR”：锁是作为并行查询的一部分而被持有的表锁；锁在事务隔离级别 3 上被工作进程持有；锁被并行查询中的工作进程持有并且必须在事务持续期间被持有。

权限 任何用户都可以执行 sp_familylock。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 命令 `kill`、`select`

系统过程 `sp_lock`、`sp_who`

sp_find_qplan

说明 在给定查询文本或计划文本的模式的情况下查找抽象计划。

语法 `sp_find_qplan pattern [, group]`

参数 *pattern*

是要在查询或抽象计划的文本中查找的字符串。

group

是抽象计划组的名称。

示例 **示例 1** 报告在查询中具有字符串 “from titles” 的所有抽象计划：

```
sp_find_qplan "%from titles%"
gid id          text
-----
2   921054317  select count(*) from titles
2   921054317
      ( plan
      ( i_scan t_pub_id_ix titles )
      ( )
    )
  ( prop titles
    ( parallel 1 )
    ( prefetch 16 )
    ( lru )
  )
5   937054374  select type, avg(price) from titles group by type
5   937054374
      ( plan
      ( store Worktab1
        ( i_scan type_price titles )
      )
      ( t_scan ( work_t Worktab1 ) )
    )
  ( prop titles
    ( parallel 1 )
    ( prefetch 16 )
    ( lru )
  )
```

示例 2 查找包含表扫描运算符的所有计划：

```
sp_find_qplan "%t_scan%"
```

示例 3 使用范围模式匹配在 dev_plans 组的计划中查找字符串，例如 “table1”、“table2” 等：

```
sp_find_qplan "%table[0-9]%", dev_plans
```

用法

- 使用 `sp_find_qplan` 可以查找包含特定字符串的抽象计划。既可以与查询文本中的字符串匹配，也可以与抽象计划文本中的字符串匹配。
- 对于每个匹配的计划，`sp_find_qplan` 都将输出组 ID、计划 ID、查询文本和抽象计划文本。
- 如果提供组名，则 `sp_find_qplan` 将在指定的组中搜索字符串。如果未提供组名，则 `sp_find_qplan` 将搜索所有组的所有查询和计划。
- 除非要搜索的字符串位于查询或计划的开头或末尾，否则必须像示例中那样提供 “%” 通配符。可以使用任何 Transact-SQL 模式匹配语法，如示例 3 中所示。
- `sysqueryplans` 中的查询文本被拆分为 255 字节的列值。`sp_find_qplan` 可能会漏掉跨越其中某个边界的匹配项，但将找到所有少于 127 字节的匹配项（即使这些匹配项跨两行）。

权限

任何用户都可以执行 `sp_find_qplan`。除非由系统管理员或数据库所有者执行，否则该命令只报告执行它的用户所拥有的抽象计划。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_help_qpgroup](#)、[sp_help_qplan](#)

sp_fixindex

| | |
|----|--|
| 说明 | <code>sp_fixindex</code> 可在系统表损坏时修复其上的一组索引（而非单个索引）。如果目标表具有位置索引或聚簇索引， <code>sp_fixindex</code> 将重建数据层（它在处理系统表的位置索引或聚簇索引时回收数据层中未使用的空间）。 |
| 语法 | <code>sp_fixindex database_name, table_name [, index_id null] [, index_name null] [, force_option]</code> |
| 参数 | <p>dbname 是数据库名称</p> <p>tablename 是表名</p> <p>index_id 是要修复的索引的 ID</p> <p>index_name 指示需要处理的索引。如果使用 NULL 值，则会重建与 <code>index_id</code> 关联的索引。如果 <code>index_id</code> 也为 NULL 值，则将重建系统表中的所有索引</p> <p>force_option 强制 Adaptive Server 重建 tempdb 中的系统表索引。不带 <code>force_option</code> 的 <code>sp_fixindex</code> 将强制 <code>database_name</code> 指定的数据库处于单用户模式，这是 tempdb 无法实现的。虽然 <code>force_option</code> 允许重建 tempdb 中的系统目录，但不应将其用于用户数据库。</p> |
| 示例 | <p>示例 1 修复 pubs2 数据库的 sysprocedures 表上的聚簇索引：</p> <pre>sp_fixindex pubs2, sysprocedures, 1</pre> <p>示例 2 重建 testdb.sysprocedures 上索引 ID 为 2 的索引：</p> <pre>sp_fixindex 'testdb', 'sysprocedures', 2</pre> <p>示例 3 重建 testdb.sysprocedures 系统表中的索引 csysprocedures：</p> <pre>sp_fixindex 'testdb', 'sysprocedures', null, 'csysprocedures'</pre> <p>示例 4 重建 testdb 中的 sysprocedures 表上的所有可用索引。如果该表具有聚簇或位置索引，<code>sp_fixindex</code> 将通过删除数据页中存在的碎片来回收未使用的空间（即重建数据页）：</p> <pre>sp_fixindex 'testdb', 'sysprocedures'</pre> <p>示例 5 重建 tempdb.sysprocedures 上索引 ID 为 2 的索引：</p> <pre>sp_fixindex 'tempdb', 'sysprocedures', 2, null, 1</pre> |

示例 6 重建表 tempdb..sysprocedures 的索引 csysprocedures:

```
sp_fixindex 'tempdb', 'sysprocedures', null,
           'sysprocedures', 1
```

示例 7 重建 tempdb 中的 sysprocedures 上的所有索引:

```
sp_fixindex 'tempdb', 'sysprocedures', null, null, 1
```

用法

在运行 `sp_fixindex` 之前，确保您的数据库处于单用户模式并重新配置数据库以允许更新系统表。

在运行 `sp_fixindex` 之后:

- 使用 `dbcc checktable` 命令验证损坏的索引已被修复
- 使用 `sp_configure` 禁止更新系统表
- 关闭单用户模式

不要在用户表上运行 `sp_fixindex`。

使用 `sp_fixindex` 修复 `sysobjects` 上的非聚簇索引需要执行额外的步骤。

警告! 不要在 `sysobjects` 或 `sysindexes` 表或用户表的聚簇索引上运行 `sp_fixindex`。否则 `sp_fixindex` 将返回以下错误消息:

```
The index with id 1 on sysobjects cannot be recreated.
```

权限

只有 SA 才能运行 `sp_fixindex`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色—当前活动角色 • 关键字或选项—NULL • 以前值—NULL • 当前值—NULL • 其它信息—所有输入参数 • 代理信息—set proxy 有效时的初始登录名 |

另请参见

文档 有关 `sp_fixindex` 的详细信息，请参见:

- 《故障排除和错误消息指南》中的第 2 章“任务综述”。
- 《性能和调优指南: 基础知识》中的第 13 章“为改善性能建立索引”。

sp_flushstats

| | |
|----|--|
| 说明 | 将内存中的统计信息刷新到 <code>systabstats</code> 和 <code>sysstatistics</code> 系统表中。 |
| 语法 | <code>sp_flushstats [objname]</code> |
| 参数 | <i>objname</i> 是表的名称。 |
| 示例 | 刷新 <code>titles</code> 表的统计信息： <pre>sp_flushstats titles</pre> |
| 用法 | <ul style="list-style-type: none"> 如果指定表时未使用 <i>objname</i> 参数，<code>sp_flushstats</code> 将在数据库级别操作。 <code>systabstats</code> 表中的某些统计信息是在内存中的存储位置更新的，并定期刷新到 <code>systabstats</code>，以减少 <code>systabstats</code> 上的开销和争用。 如果使用 SQL 查询 <code>systabstats</code>，则执行 <code>sp_flushstats</code> 可确保内存中的统计信息被刷新到 <code>systabstats</code>。 <code>optdiag</code> 命令总是在显示输出内容之前刷新内存中的统计信息。 只能通过数据定义语言命令来更改 <code>sysstatistics</code> 中的统计信息，并且不需要使用 <code>sp_flushstats</code>。 内存中的数据更改计数器永久存储在 <code>sysstatistics</code> 中。执行 <code>sp_flushstats</code> 时，这些数据更改将刷新到磁盘。 |
| 权限 | 只有系统管理员才能执行 <code>sp_flushstats</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

sp_forceonline_db

| | |
|----|--|
| 说明 | 提供对数据库中所有页的访问，这些页以前被恢复过程标记为可疑。 |
| 语法 | <pre>sp_forceonline_db dbname, {"sa_on" "sa_off" "all_users"}</pre> |
| 参数 | <p>dbname 是要使其联机的数据库的名称。</p> <p>sa_on 只允许具有 sa_role 的用户访问指定的页。</p> <p>sa_off 撤消以前通过 sa_on 调用 sp_forceonline_page 创建的访问特权。</p> <p>all users 允许所有用户访问指定的页。</p> |
| 示例 | <p>示例 1 允许系统管理员访问 pubs2 数据库中的所有可疑页：</p> <pre>sp_forceonline_db pubs2, "sa_on"</pre> <p>示例 2 撤消系统管理员访问 pubs2 数据库中所有可疑页的权限。现在，没有人能访问 pubs2 中的可疑页：</p> <pre>sp_forceonline_db pubs2, "sa_off"</pre> <p>示例 3 允许所有用户访问 pubs2 数据库中的所有页：</p> <pre>sp_forceonline_db pubs2, "all_users"</pre> |
| 用法 | <ul style="list-style-type: none">• 不一定要修复被强制联机的页。损坏的页也可以被强制联机。Adaptive Server 不对强制联机的页执行一致性检查。• 不能撤消带 all users 的 sp_forceonline_page。如果已将页联机使之供所有用户访问，则不能再使这些页脱机。• sp_forceonline_db 不能在事务中使用。• 若要仅使特定的脱机页联机，请使用 sp_forceonline_page。 |
| 权限 | 只有系统管理员才能执行 sp_forceonline_db 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_forceonline_page](#)、[sp_listsuspect_db](#)、[sp_listsuspect_page](#)、[sp_setsuspect_granularity](#)、[sp_setsuspect_threshold](#)

sp_forceonline_object

| | |
|----|--|
| 说明 | 提供对以前被恢复过程标记为可疑的索引的访问。 |
| 语法 | <code>sp_forceonline_object dbname, objname, indid, {sa_on sa_off all_users} [, no_print]</code> |
| 参数 | <p>dbname 是包含要使其联机的索引的数据库的名称。</p> <p>objname 是表的名称。</p> <p>indid 是要联机的可疑索引的索引 ID。</p> <p>sa_on 只允许具有 sa_role 的用户访问指定的索引。</p> <p>sa_off 撤消以前通过 sa_on 调用 <code>sp_forceonline_object</code> 创建的访问特权。</p> <p>all_users 允许所有用户访问指定的索引。</p> <p>no_print 使指定的对象联机后，不输出其它可疑对象的列表。</p> |
| 示例 | <p>示例 1 允许系统管理员访问 pubs2 数据库中 titles 表上具有 indid 3 的索引：</p> <pre>sp_forceonline_object pubs2, titles, 3 , sa_on</pre> <p>示例 2 撤消系统管理员对索引的访问权限。现在，所有人都不能访问该索引：</p> <pre>sp_forceonline_object pubs2, titles, 3, sa_off</pre> <p>示例 3 允许所有用户访问 pubs2 数据库中 titles 表上的索引：</p> <pre>sp_forceonline_object pubs2, titles, 3, all_users</pre> |
| 用法 | <ul style="list-style-type: none"> 如果仅数据锁定表上的索引具有可疑页，则整个索引将在恢复期间脱机。查询优化程序将不考虑脱机索引。恢复期间不会使所有页锁定表上的索引完全脱机；只有这些索引的个别页会脱机。可使用 <code>sp_forceonline_page</code> 使这些页联机。 使用 <code>sp_listsuspect_object</code> 可查看脱机的数据库列表。 |

- 若要修复可疑索引，请使用具有 `sa_on` 访问权限的 `sp_forceonline_object`。然后，删除并重新创建索引。

注释 如果索引位于 `systabstats` 或 `sysstatistics`（仅有的仅数据锁系统表）上，请致电 Sybase 技术支持部门寻求帮助。

- 不能撤消带有 `all_users` 的 `sp_forceonline_object`。如果已将索引联机使之供所有用户访问，则不能使其脱机。
- 不一定要修复被强制联机的索引。损坏的索引也可以被强制联机。Adaptive Server 不对强制联机的索引执行一致性检查。
- `sp_forceonline_object` 不能在事务中使用。
- `sp_forceonline_object` 仅用于其恢复故障隔离模式是“page”的数据库。使用 `sp_setsuspect_granularity` 可显示数据库的恢复故障隔离模式。
- 若要通过单个命令使数据库的所有脱机页和索引联机，请使用 `sp_forceonline_db`。

权限

只有系统管理员才能执行 `sp_forceonline_object`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

文档 有关恢复故障隔离的详细信息，请参见《系统管理指南》。

系统过程 [sp_listsuspect_object](#)、[sp_setsuspect_granularity](#)

sp_forceonline_page

| | |
|----|---|
| 说明 | 提供对以前被恢复过程标记为可疑的页的访问。 |
| 语法 | <pre>sp_forceonline_page dbname, pgid, {"sa_on" "sa_off" "all_users"}</pre> |
| 参数 | <p>dbname 是包含要联机的页的数据库的名称。</p> <p>pgid 是要联机的页的页标识符。</p> <p>sa_on 只允许具有 sa_role 的用户访问指定的页。</p> <p>sa_off 撤消以前通过 sa_on 调用 sp_forceonline_page 创建的访问特权。</p> <p>all_users 允许所有用户访问指定的页。</p> |
| 示例 | <p>示例 1 允许系统管理员访问 pubs2 数据库的第 312 页：</p> <pre>sp_forceonline_page pubs2, 312, "sa_on"</pre> <p>示例 2 撤消系统管理员访问 pubs2 数据库中第 312 页的权限。现在，所有人都不能访问该页：</p> <pre>sp_forceonline_page pubs2, 312, "sa_off"</pre> <p>示例 3 允许所有用户访问 pubs2 数据库的第 312 页：</p> <pre>sp_forceonline_page pubs2, 312, "all_users"</pre> |
| 用法 | <ul style="list-style-type: none">不能撤消带 all_users 的 sp_forceonline_page。如果已将页联机使之供所有用户访问，则不能再使这些页脱机。不一定要修复被强制联机的页。损坏的页也可以被强制联机。Adaptive Server 不对强制联机的页执行一致性检查。sp_forceonline_page 不能在事务中使用。sp_forceonline_page 仅用于其恢复故障隔离模式是“page”的数据库。使用 sp_setsuspect_granularity 可显示数据库的恢复故障隔离模式。若要通过单个命令使数据库的所有脱机页联机，请使用 sp_forceonline_db。 |
| 权限 | 只有系统管理员才能使用 sp_forceonline_page 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_forceonline_db](#)、[sp_listsuspect_db](#)、[sp_listsuspect_page](#)、[sp_setsuspect_granularity](#)、[sp_setsuspect_threshold](#)

sp_foreignkey

| | |
|----|---|
| 说明 | 在当前数据库中定义表或视图的外键。 |
| 语法 | <pre>sp_foreignkey <i>tablename</i>, <i>pktabname</i>, <i>col1</i> [, <i>col2</i>] ... [, <i>col8</i>]</pre> |
| 参数 | <p><i>tablename</i> 是包含要定义的外键的表或视图的名称。</p> <p><i>pktabname</i> 是具有外键所适用的主键的表或视图的名称。主键必须已经定义。</p> <p><i>col1</i> 是构成外键的第一个列的名称。外键必须至少有一个列，最多可以有八列。</p> |
| 示例 | <p>示例 1 publishers 表的主键是 pub_id 列。titles 表也包含 pub_id 列，该列是 publishers 的外键：</p> <pre>sp_foreignkey titles, publishers, pub_id</pre> <p>示例 2 已通过 sp_primarykey 将 parts 表的主键定义为 partnumber 和 subpartnumber 列。orders 表包含 part 和 subpart 列，这两列构成 parts 的外键：</p> <pre>sp_foreignkey orders, parts, part, subpart</pre> |
| 用法 | <ul style="list-style-type: none">• <code>sp_foreignkey</code> 将键添加到 <code>syskeys</code> 表。键使数据库设计中的隐式逻辑关系变为显式的。• <code>sp_foreignkey</code> 不强制参照完整性约束；使用 <code>create table</code> 或 <code>alter table</code> 命令的 <code>foreign key</code> 子句可以强制外键关系。• 构成外键的列的编号和顺序必须与构成主键的列的编号和顺序相同。主键和外键的数据类型（和长度）必须一致，但空值类型不必一致。• 安装进程对系统表上的相应列运行 <code>sp_foreignkey</code>。• 若要显示有关已定义键的报告，请执行 sp_helpkey。• 不能将 Java 数据类型用于 <code>sp_foreignkey</code>。 |
| 权限 | 只有表或视图的所有者才能执行 <code>sp_foreignkey</code> 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [alter table](#)、[create table](#)、[create trigger](#)

系统过程 [sp_commonkey](#)、[sp_dropkey](#)、[sp_helpjoins](#)、[sp_helpkey](#)、[sp_primarykey](#)

sp_freeldll

| | |
|----|--|
| 说明 | 卸载以前装入 XP Server 内存的动态链接库 (DLL)，以支持扩展存储过程 (ESP) 的执行。 |
| 语法 | <code>sp_freeldll dll_name</code> |
| 参数 | <i>dll_name</i> 是要从 XP Server 内存卸载的 DLL 的文件名。 |
| 示例 | 卸载 <i>sqlsrvdll.dll</i> DLL: <pre>sp_freeldll "sqlsrvdll.dll"</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_freeldll</code> 无法从事务内执行。 • <code>sp_freeldll</code> 无法释放系统 ESP 的 DLL。 • 对于使用 <code>sp_freeldll</code> 显式卸载 DLL 而言有一种替代方法，就是指定 DLL 在调用它们的 ESP 请求终止后总是被卸载。为此，将 <code>esp unload dll</code> 配置参数设置为 1 或用 <code>-u</code> 选项启动 <code>xpserver</code>。 • <code>sp_freeldll</code> 用来在不关闭 XP Server 或 Adaptive Server 的情况下更新 DLL 中的 ESP 函数。 • 如果您使用 <code>sp_freeldll</code> 卸载正在使用中的 DLL，则 <code>sp_freeldll</code> 将成功执行，但导致当前正使用该 DLL 的 ESP 失败。 |
| 权限 | 只有系统管理员才能执行 <code>sp_freeldll</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_addextendedproc](#)、[sp_dropextendedproc](#)、[sp_helpextendedproc](#)

sp_getmessage

说明 从 `print` 和 `raiserror` 语句的 `sysmessages` 和 `sysusermessages` 中检索存储的消息字符串。

语法 `sp_getmessage message_num, result output [, language]`

参数 `message_num`
是要被检索的消息的编号。

result output
是接收返回的消息文本的变量，后跟一个空格和关键字 `output`。该变量的数据类型必须为 `char`、`unichar`、`nchar`、`varchar`、`univarchar` 或 `nvarchar`。

language
是要被检索的消息的语言。`language` 必须是 `syslanguages` 表中的有效语言名称。如果包括 `language`，则检索具有指定的 `message_num` 和 `language` 的消息。如果不包括 `language`，则检索由变量 `@@langid` 指定的缺省会话语言的消息。

示例 **示例 1** 从 `sysusermessages` 检索编号为 20001 的消息：

```
declare @myvar varchar(200)
exec sp_getmessage 20001, @myvar output
```

示例 2 从 `sysusermessages` 检索编号为 20010 的消息的法语版本：

```
declare @myvar varchar(200)
exec sp_getmessage 20010, @myvar output, french
```

用法 任何应用程序都可以使用 `sp_getmessage`，并且任何用户都可以读取 `sysmessages` 和 `sysusermessages` 中存储的消息。

权限 任何用户都可以执行 `sp_getmessage`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** `print`、`raiserror`

系统过程 `sp_addmessage`、`sp_dropmessage`

sp_grantlogin

| | |
|----|---|
| 说明 | (仅限 Windows) 当集成安全模式或混合模式 (具有命名管道) 处于活动状态时, 将 Adaptive Server 角色或 default 权限指派给 Windows 用户和组。 |
| 语法 | <code>sp_grantlogin {login_name group_name} ["role_list" default]</code> |
| 参数 | <p><i>login_name</i> 是 Windows NT 用户的网络登录名。</p> <p><i>group_name</i> 是 Windows NT 组名。</p> <p><i>role_list</i> 是授予的 Adaptive Server 角色的列表。该角色列表可以包括一个或多个以下角色名: <code>sa_role</code>、<code>sso_role</code>、<code>oper_role</code>。如果指定多个角色, 则用空格分隔这些角色名, 而不要用逗号分隔。</p> <p><code>default</code> 指定 <i>login_name</i> 或 <i>group_name</i> 接收由 <code>grant</code> 语句或 <code>sp_role</code> 过程指派的缺省权限。</p> |
| 示例 | <p>示例 1 向 Windows NT 用户 “jeanluc” 指派 Adaptive Server <code>oper_role</code>:</p> <pre>sp_grantlogin jeanluc, oper_role</pre> <p>示例 2 将 <code>default</code> 值指派给 Windows NT 用户 “valle”。用户 “valle” 接收已通过 <code>grant</code> 命令或 <code>sp_role</code> 过程指派给她的任何权限:</p> <pre>sp_grantlogin valle</pre> <p>示例 3 将 Adaptive Server <code>sa_role</code> 和 <code>sso_role</code> 指派给 Windows NT 管理员组的所有成员:</p> <pre>sp_grantlogin Administrators, "sa_role sso_role"</pre> |
| 用法 | <ul style="list-style-type: none"> • 您必须首先创建 Windows NT 登录名或组, 然后才能使用 <code>sp_grantlogin</code> 指派角色。有关详细信息, 请参见相关 Windows NT 文档。 • 只有在 Adaptive Server 运行在集成安全模式或混合模式下并且连接是命名管道的情况下, <code>sp_grantlogin</code> 才处于活动状态。如果 Adaptive Server 运行在连接并非命名管道的标准模式或混合模式下, 则改用 <code>grant</code> 和 <code>sp_role</code>。 • 如果未指定 <i>role_list</i> 或 <code>default</code>, 则该过程自动指派 <code>default</code> 值。 • <code>default</code> 值不指示某一 Adaptive Server 角色。它指定用户或组应接收已通过 <code>grant</code> 命令或 <code>sp_role</code> 过程指派给它的任何权限。 |

- 将 `sp_grantlogin` 用于现有 `login_name` 或 `group_name` 将覆盖该用户或组的现有角色。

权限

只有系统管理员才能执行 `sp_grantlogin`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `grant`、`setuser`

系统过程 `sp_addlogin`、`sp_displaylogin`、`sp_droplogin`、`sp_locklogin`、`sp_logininfo`、`sp_modifylogin`、`sp_revokelgin`、`sp_role`

sp_ha_admin

| | |
|----|---|
| 说明 | 对高可用性系统中的配置了 Sybase 故障切换功能的 Adaptive Server 执行管理任务。sp_ha_admin 是使用 <i>installhavss</i> 脚本（UNIX 平台）或 <i>insthasv</i> 脚本（Windows NT）安装的。 |
| 语法 | sp_ha_admin [cleansessions help] |
| 参数 | <p>cleansessions</p> <p>从 <i>sysessions</i> 中删除旧条目。旧的 <i>sysessions</i> 条目通常是由于 Adaptive Server 未能在重新启动期间清除 <i>sysessions</i> 或由于客户端未能连接到 Adaptive Server 而被遗留下来的。</p> <p>help</p> <p>显示 sp_ha_admin 的语法。</p> |
| 示例 | <p>示例 1 从 <i>sysessions</i> 中删除由未正确退出的客户端连接遗留下来的旧条目：</p> <pre>sp_ha_admin cleansessions (return status = 0)</pre> <p>示例 2 显示 sp_ha_admin 的语法：</p> <pre>sp_ha_admin "help" sp_ha_admin Usage: sp_ha_admin command [, option1 [, option2]] sp_ha_admin commands: sp_ha_admin 'cleansessions' sp_ha_admin 'help' (return status = 0)</pre> |
| 用法 | <ul style="list-style-type: none"> • sp_ha_admin 对高可用性系统中的配置了 Sybase 故障切换功能的 Adaptive Server 执行管理任务。sp_ha_admin 不能使用 <i>installmaster</i> 脚本来安装，而应使用用于安装和配置 Sybase 的故障切换功能的 <i>installhavss</i> 脚本（在 Windows NT 上则为 <i>insthasv</i>）来安装。 • 如果成功清除了 <i>sysessions</i>，sp_ha_admin 将返回 0；如果遇到错误，则返回 1。 • 如果无法从 <i>sysessions</i> 删除任何条目（例如，无法获取 <i>sysessions</i> 上的锁），则 sp_ha_admin 将在错误日志中写入一条消息。 • 若要查看 <i>sysessions</i> 中的当前所有条目，请输入： <pre>select * from sysessions</pre> |
| 权限 | 只有具有 <i>ha_role</i> 角色的系统管理员才能执行 sp_ha_admin。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

sp_help

说明 报告与数据库对象（在 `sysobjects` 中列出的任何对象）、系统数据类型或用户定义的数据类型、用户定义的函数以及计算列和基于函数的索引有关的信息。其中的列显示 `optimistic_index_lock`。

语法 `sp_help [objname]`

参数 `objname`

是 `sysobjects` 中任何对象或 `systypes` 中任意用户定义的数据类型或系统数据类型的名称。您不能指定数据库名称。`objname` 可以包括表、视图、存储过程、日志、规则、缺省值、触发器、参照约束、加密密钥和检查约束，但在启用了 `optimistic_index_lock` 的情况下则表示表。如果对象所有者既不是运行该命令的用户也不是数据库所有者，则使用所有者名。

示例 1 显示 `sysobjects` 中对象的列表并显示每一对象的名称、所有者和对象类型。还显示 `systypes` 中由每位用户定义的数据类型的列表，并且指出数据类型名称、存储类型、长度、空值类型、缺省名和规则名称。空值类型是 0（不允许为空值）或 1（允许为空值）：

```
sp_help
```

示例 2 显示有关已分区的 `publishers` 表的信息。`sp_help` 还列出指派给指定表及其索引的任何属性，并且给出该属性的类、名称、整数值、字符值和注释。

```
sp_help publishers
```

| Name | Owner | Object_Type | Create_date |
|------------|-------|-------------|--------------------|
| publishers | dbo | user table | Oct 7 2005 11:14AM |

| Column_name | Type | Length | Prec | Scale | Nulls | Default_name | Rule_name |
|------------------|----------|---------------|----------|-------|-------|--------------|------------|
| Access_Rule_name | Computed | Column_object | Identity | | | | |
| pub_id | char | 4 | NULL | NULL | 0 | NULL | pub_idrule |
| | NULL | | | NULL | 0 | | |
| pub_name | varchar | 40 | NULL | NULL | 1 | 0 | NULL |
| | NULL | | | NULL | | | |
| city | varchar | 20 | NULL | NULL | 1 | 0 | NULL |
| | NULL | | | NULL | | | |
| state | char | 2 | NULL | NULL | 1 | 0 | NULL |
| | NULL | | | NULL | 0 | | |

Object does not have any indexes.

| keytype | object | related_objs | object_keys | related_keys |
|---------|--------|--------------|-------------|--------------|
| | | | | |


```

primary publishers -- none -- pub_id,*,*,*,*,*,* *,*,*,*,*,*,*
name          type          partition_type  partitions  partition_keys
-----
publishers  base table  roundrobin          3  NULL

partition_name          partition_id  pages  segment  create_date
-----
publishers_608002166    608002166    1  default  Oct 13 2005 11:18AM
publishers_1116527980  1116527980    1  default  Oct 13 2005 11:18AM
publishers_1132528037  1132528037    1  default  Oct 13 2005 11:19AM

Partition_Conditions
-----
NULL

Avg_pages  Max_pages  Min_pages  Ratio(Max/Avg)  Ratio(Min/Avg)
-----
1          1          1  1.0000000      1.0000000

Lock scheme Allpages
The attribute 'exp_row_size" is not applicable to tables with allpages lock scheme.

exp_row  reservepagegap  fillfactor  max_rows_per_page  identity_gap
-----
0          0          0          0          0

concurrency_opt_threshold  optimistic_index_lock  dealloc_first_txtpg
-----
0          0          0

```

示例 3 显示有关已分区的 titles 表的信息:

```

sp_help titles

Name          Owner          Object_Type  Create_date
-----
titles        db             user table   Oct 7 2005 11:14AM
(1 row affected)

Column_name  Type  Length  Prec  Scale  Nulls  Default_name  Rule_name  Access_Rule_name
-----
Identity

-----
title_id     tid      6  NULL  NULL    0  NULL          title_idrule  NULL
0
title        varchar  80  NULL  NULL    0  NULL          NULL          NULL
0
type         char    12  NULL  NULL    0  typedflt      NULL          NULL
0
pub_id       char     4  NULL  NULL    1  NULL          NULL          NULL

```

```

0
price      money      8 NULL NULL      1 NULL      NULL      NULL
0
advance    money      8 NULL NULL      1 NULL      NULL      NULL
0
total_sales int        4 NULL NULL      1 NULL      NULL      NULL
0
notes      varchar    200 NULL NULL      1 NULL      NULL      NULL
0
pubdate    datetime   8 NULL NULL      0 datedflt  NULL      NULL
0
contract   bit         1 NULL NULL      0 NULL      NULL      NULL
0
index_name      index_description      index_keys

```

Object has the following indexes

```

index_name      index_keys      index_description      index_max_rows_per_page
index_fillfactor      index_reservepagegap      index_created      index_local
-----
title_idx      total_sales      clustered      0
0      0 Oct 13 2005 5:20PM      Local Index

```

```

index_ptn_name      index_ptn_seg
-----
p1      default
p2      default
p3      default
title_idx_98505151      default

```

```

keytype      object      related_object      object_keys
related_keys
-----
foreign      roysched      titles      title_id, *, *, *, *, *, *, * title_id, *, *,
*, *, *, *, *
foreign      salesdetail      titles      title_id, *, *, *, *, *, *, * title_id, *, *,
*, *, *, *
foreign      titleauthor      titles      title_id, *, *, *, *, *, *, * title_id, *, *,
*, *, *, *
foreign      titles      publishers      pub_id, *, *, *, *, *, *, *
pub_id, *, *, *, *, *, *, *
primary      titles      -- none --      title_id, *, *, *, *, *, *, *
*, *, *, *, *, *, *, *, *

```

```

name      type      partition_type      partitions      partition_keys
-----
titles      base table      range      4      pubdate

```

```

partition_name      partition_id      pages      segment      create_date

```

```

-----
q1          937051343          1 default Oct 13 2005 5:20PM
q2          953051400          1 default Oct 13 2005 5:20PM
q3          969051457          1 default Oct 13 2005 5:20PM
q4          985051514          1 default Oct 13 2005 5:20PM

```

Partition_Conditions

```

-----
VALUES <= ("3/31/2006")
VALUES <= ("6/30/2006")
VALUES <= ("9/30/2006")
VALUES <= ("12/31/2006")
VALUES <= ("3'31'2006")

```

| Avg_pages | Max_pages | Min_pages | Ratio(Max/Avg) | Ratio(Min/Avg) |
|-----------|-----------|-----------|----------------|----------------|
| 1 | 1 | 1 | 1.000000 | 1.000000 |

Lock scheme Allpages

The attribute 'exp_row_size" is not applicable to tables with allpages lock scheme.

| exp_row | reservepagegap | fillfactor | max_rows_per_page | identity_gap |
|---------|----------------|------------|-------------------|--------------|
| 0 | 0 | 0 | 0 | 0 |

| concurrency_opt_threshold | optimistic_index_lock | dealloc_first_txdpg |
|---------------------------|-----------------------|---------------------|
| 0 | 0 | 0 |

示例 4 显示有关用户 “mary” 所拥有的触发器 marytrig 的信息。引号不可省略，因为句点是特殊字符：

```

sp_help "mary.marytrig"

Name          Owner          Object_type
-----
marytrig     mary           trigger

Data_located_on_segment  When_created
-----
not applicable           Mar 20 2002  2:03PM

```

示例 5 显示有关系统数据类型 money 的信息：

```

sp_help money

Type_name      Storage_type  Length  Prec  Scale  Nulls  Default_name
-----
Rule_name     Access_Rule_name  Identity
-----

```

```
money          money          8  NULL  NULL  1  NULL
NULL          NULL          0
```

示例 6 显示有关用户定义的数据类型 `idtype` 的信息。该报告指示了从其创建数据类型的基本类型，不管它是否允许空值、允许绑定到该数据类型的任何规则和缺省值的名称，也不管它是否具有 `IDENTITY` 属性：

```
sp_help idtype
```

```
Type_name  Storage_type  Length  Prec  Scale  Nulls  Defaul_name
-----
Rule_name  Access_Rule_name  Identity
-----
idtype     numeric        4  NULL  NULL  1  NULL
NULL      NULL          1
```

显示一个新列，指示是否启用优化索引锁定。1 指示启用该选项；0 指示不启用该选项。

```
sp_help "mytable"
```

```
-----
exp_row_size  reserve  pagegap  fillfactor  max_rows_per_page
-----
1            0        0        0            0
concurrency_opt_threshold  optimistic_index_lock
-----
0            1
```

示例 7 显示一个虚拟计算列：

```
alter table authors add fullname as au_fname + ' ' + au_lname
```

```
sp_help authors
```

```
Object has the following computed columns
```

```
Column_Name  Property
-----
fullname     virtual
```

```
Text
```

```
-----
AS au_fname + ' ' + au_lname
```

示例 8 将一个虚拟计算列显示为实现计算列：

```
alter table authors modify fullname materialized
sp_help authors

Object has the following computed columns

Column_Name Property
-----
fullname      materialized

Text
-----
AS au_fname + ' ' + au_lname
MATERIALIZED
```

示例 9 `sp_help table_name` 的结果集包含 `Decrypt_Default_name` 列，指示该列的解密缺省名称。例如，如果运行以下命令：

```
create table encr_table(coll int encrypt decrypt_default 1)
```

当对 `encr_table` 运行 `sp_help` 时，它会显示以下结果：

```
Column_name Type Length Prec Scale Nulls Default_name Rule_name Access_Rule_name
Computed_Column_object Identity Encrypted Decrypt_Default_name
-----
c1          int      4 NULL NULL 0 NULL NULL NULL
NULL                                0          1 encr_table_coll_1036527695
```

示例 10 显示谓词对象的 `Name`、`Owner`、`Object_type`、`Object_status` 和 `Create_date`：

```
grant select on tabl where coll = 5 as pred1 to robert
sp_help pred1
```

```
Name Owner Object_type Object_status Create_date
-----
pred1 dbo predicate -- none -- Feb 9 2010 12:49PM
```

用法

- 对于虚拟散列表：`sp_help` 报告：
 - 使用下面的消息报告表是虚拟散列表：


```
Object is Virtually Hashed
```
 - 使用以下语法以消息形式报告表的 `hash_key_factors`：


```
column_1:hash_factor_1,
column_2:hash_factor_2...,
max_hash_key=max_hash_value
```

例如：

```

attribute_class      attribute      int_value
char_value           -----
-----
hash clustered tables  hash key factors  NULL
id:10.0, id2:1.0, max_hash_key=1000.0  NULL

```

- `sp_help` 只查找当前数据库中的对象。
- `sp_help` 遵循用于查找对象的 Adaptive Server 规则：
 - 如果您没有指定所有者名，并且拥有具有指定名称的对象，则 `sp_help` 报告该对象的有关信息。
 - 如果您没有指定所有者名，并且不拥有具有该名称的对象，但数据库所有者拥有具有该名称的对象，则 `sp_help` 报告该数据库所有者的对象的有关信息。
 - 如果您或数据库所有者均不拥有具有指定名称的对象，则 `sp_help` 会报告一个错误情况，即使在不同所有者的数据库中存在具有该名称的对象时也是如此。用所有者的名称限定其他数据库用户（既不是您自己也不是数据库所有者）所拥有的对象，如示例 4 中所示。
 - 如果您和数据库所有者都拥有具有指定名称的对象，并且您要访问该数据库所有者的对象，则以 `dbo.objectname` 格式指定该名称。
- 如果您从 `tempdb` 发布 `sp_help`，则该命令对临时表起作用。
- 具有 IDENTITY 属性的列的“Identity”值为 1；其它列的“Identity”值为 0。在示例 2 中，没有 IDENTITY 列。
- `sp_help` 列出表上的所有索引，包括通过在 `create table` 或 `alter table` 语句中定义唯一键或主键约束创建的索引。该命令还列出与这些索引关联的任何属性。不过，`sp_help` 不提供关于为表定义的完整性约束的任何信息。请使用 `sp_helpconstraint` 来获得有关任何完整性约束的信息。
- `sp_help` 显示以下新设置：
 - 锁定方案，可用 `create table` 设置并可用 `alter table` 更改
 - 预期行大小，可用 `create table` 设置并可用 `sp_chgattribute` 更改
 - 保留页间距，可用 `create table` 设置并可用 `sp_chgattribute` 更改
 - 行锁升级设置，可用 `sp_setpglockpromote` 设置或更改并可用 `sp_drowlockpromote` 删除

- `sp_help` 包括来自以下来源的报告：
 - `sp_helpindex` — 显示用于创建索引的键的顺序以及空间管理属性
 - `sp_helppartition` — 显示表的分区信息
 - `sp_helpcomputedcolumn` — 显示表的计算列信息
- 在启用组件集成服务后，`sp_help` 显示有关远程对象的存储位置的信息。
- `sp_help` 可以显示加密密钥的有关信息。当密钥名称被指定为 `sp_help` 的参数时，该命令将列出该密钥的名称、所有者、对象类型及创建日期。
- `sp_help tablename` 指示列是否已加密，如果存在列的解密缺省名称，则还包括该名称。

权限 任何用户都可以执行 `sp_help`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关 SQLJ 例程的详细信息，请参见《Adaptive Server Enterprise 中的 Java》。

命令 `alter table`、`create table`

系统过程 `sp_chgattribute`、`sp_droprolockpromote`、`sp_helppartition`、`sp_helpcomputedcolumn`、`sp_helpconstraint`、`sp_helpindex`、`sp_setpglockpromote`

sp_help_resource_limit

说明 报告资源限制的有关信息。

语法 sp_help_resource_limit [*name* [, *appname* [, *limittime* [, *limitday* [, *scope* [, *action* [, *verbose*]]]]]]]

参数 *name*

是对其应用限制的 Adaptive Server 登录名。有关控制特定登录名的限制的信息，请指定登录名 *name*。有关与登录名无关的限制的信息，请指定 *null*。

注释 如果您不是系统管理员，请指定您自己的登录名，或指定登录名 NULL，以显示应用于您的资源限制的有关信息。

appname

是对其应用限制的应用程序名称。有关控制特定应用程序的限制的信息，请指定客户端程序在登录包中传递给 Adaptive Server 的应用程序的名称。有关与应用程序无关的限制的信息，请指定 *null*。

limittime

是强制执行限制的时间。有关在指定时间限制生效的信息，请使用以下格式用范围介于“00:00”和“23:59”之间的值指定时间：

"*HH:MM*"

有关与时间无关的限制的信息，请指定 *null*。

limitday

是强制执行限制的任何日期。有关在一周的给定星期几生效的资源限制的信息，请指定缺省服务器语言的完整星期名称，缺省服务器语言存储于 *master* 数据库的 *syslanguages* 系统表中。有关与限制强制执行的日期无关的限制的信息，请指定 *null*。

scope

是限制的作用域。请指定以下各项之一：

| scope 代码 | 有关控制以下各项的所有限制的帮助 |
|----------|--|
| 1 | 查询 |
| 2 | 查询批处理（由客户端发送给服务器的一个或多个 SQL 语句） |
| 4 | 事务 |
| 6 | 查询批处理和事务 |
| NULL | 指定的 <i>name</i> 、 <i>appname</i> 、 <i>limittime</i> 、 <i>limitday</i> 和 <i>action</i> ，不考虑其 <i>scope</i> 。 |

action

是在超出限制时采取的操作。请指定以下各项之一：

| action 代码 | 有关以下操作的所有限制的帮助 |
|-----------|---|
| 1 | 发出一个警告 |
| 2 | 中止查询批处理 |
| 3 | 中止事务 |
| 4 | 注销会话 |
| NULL | 控制指定的 <i>name</i> 、 <i>appname</i> 、 <i>limittime</i> 、 <i>limitday</i> 和 <i>scope</i> ，不考虑它们所采取的 <i>action</i> 。 |

verbose

使用此参数时，输出将以详细模式显示，值为 1 或 0（零）。

示例

示例 1 列出 `sysresourcelimits` 系统表中存储的所有资源限制：

```
sp_help_resource_limit
```

示例 2 列出对用户 “joe_user” 应用的所有限制：

```
sp_help_resource_limit joe_user
```

示例 3 列出对应用程序 `my_app` 应用的所有限制：

```
sp_help_resource_limit NULL, my_app
```

示例 4 列出在上午 9:00 强制执行的所有限制：

```
sp_help_resource_limit NULL, NULL, "09:00"
```

示例 5 对于列举上午 9:00 强制执行的限制，这是一种替代方法：

```
sp_help_resource_limit @limittype = "09:00"
```

示例 6 列出在星期一强制执行的所有限制：

```
sp_help_resource_limit NULL, NULL, NULL, Monday
```

示例 7 列出每星期一上午 9:00 对 “joe_user” 生效的任何限制：

```
sp_help_resource_limit joe_user, NULL, "09:00", Monday
```

示例 8 以详细模式列出所有限制：

```
sp_help_resource_limit null,null,null,null,null,null,1
```

示例 9 以详细模式列出所有资源限制：

```
sp_help_resource_limit @verbose=1
```

用法

- `sp_help_resource_limit` 报告所有资源限制、给定登录名或应用程序的限制、在给定时间或一周中的星期几生效的限制或具有给定作用域或操作的限制。

权限 任何用户都可以执行 `sp_help_resource_limit` 来列出对自己应用的资源限制。只有系统管理员才能执行 `sp_help_resource_limit` 来列出应用于其他用户的限制。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关资源限制的详细信息，请参见《系统管理指南》。

系统过程 [sp_add_resource_limit](#)、[sp_drop_resource_limit](#)、[sp_modify_resource_limit](#)

sp_help_qpgroup

说明 报告有关抽象计划组的信息。

语法 sp_help_qpgroup [*group* [, *mode*]]

参数 *group*
是抽象计划组的名称。

mode
是要输出的报告的类型，可以是下列各项之一：

| 模式 | 返回的信息 |
|---------|---|
| full | 组中的行数 and 计划数、使用两行或多行的计划数、最长计划的行数和计划 ID、散列键数以及散列键冲突信息。这是缺省报告模式。 |
| stats | “full” 报告中除散列键信息以外的所有信息。 |
| hash | 组中的行数和抽象计划数、散列键数以及散列键冲突信息。 |
| list | 组中的行数和抽象计划数，以及每个查询 / 计划对的以下信息：散列键、计划 ID、查询的前几个字符和计划的前几个字符。 |
| queries | 组中的行数和抽象计划数，以及每个查询的以下信息：散列键、计划 ID 和查询的前几个字符。 |
| plans | 组中的行数和抽象计划数，以及每个计划的以下信息：散列键、计划 ID 和计划的前几个字符。 |
| counts | 组中的行数和抽象计划数，以及每个计划的以下信息：行数、字符数、散列键、计划 ID 和查询的前几个字符。 |

示例 **示例 1** 报告有关数据库中所有抽象计划组的汇总信息：

```
sp_help_qpgroup

Group                                GID          Plans
-----
ap_stdin                             1             0
ap_stdout                             2             0
dev_test                              3            209
```

示例 2 报告 test_plans 组的有关信息：

```
sp_help_qpgroup test_plans
Query plans group 'test_plans', GID 8

Total Rows  Total QueryPlans
-----
          6             3
sysqueryplans rows consumption, number of query plans per row count

Rows          Plans
-----
```

2 3

Hashkeys

3

There is no hash key collision in this group.

- 用法
- 在用于某一抽象计划组名并且没有 mode 参数时， sp_help_qpgroup 的缺省模式为 full。
 - 散列键冲突指示特定用户的多个计划具有相同的散列键值。在有散列键冲突时，必须将具有匹配散列键的每个查询的查询文本与用户的查询文本进行比较，以便识别匹配的查询，因此性能可能会稍有降低。

权限 任何用户都可以执行 sp_help_qpgroup。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程 sp_help_qplan](#)

sp_help_qplan

说明 报告有关抽象计划的信息。

语法 `sp_help_qplan id [, mode]`

参数 *id*
是抽象计划的 ID。

mode
是要输出的报告的类型，可以是下列各项之一：

| 模式 | 返回的信息 |
|-------|---|
| full | 计划 ID、组 ID 和散列键，以及完整查询和计划文本。 |
| brief | 与 full 大体相同，但只输出查询和计划的约 80 个字符，而不是输出整个查询和计划。这是缺省模式。 |
| list | 散列键、ID 以及查询和计划的前 20 个字符。 |

示例 **示例 1** 输出简短抽象计划报告：

```
sp_help_qplan 800005881
gid          hashkey          id
-----
          5    2054169974    937054374

query
-----
select type, avg(price) from titles group by type

query_plan
-----
( plan
  ( store Worktab1
    ( i_scan type_price titles )
  )
  ( t_scan ( ...
```

示例 2 输出完整抽象计划报告：

```
sp_help_qplan 784005824, full
```

用法 • 如果没有提供 `mode` 参数的值，则缺省模式为 `brief`。

权限 任何用户都可执行 `sp_help_qplan` 以查看其拥有的查询的抽象计划。只有系统管理员和数据库所有者才能显示其他用户所拥有的抽象计划。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 系统过程 [sp_find_qplan](#)、[sp_help_qpgroup](#)

sp_helpapptrace

说明 确定 Adaptive Server 正在跟踪的会话。sp_helpapptrace 会返回 Adaptive Server 正在跟踪的所有会话的服务器进程 ID (spid)、对其进行跟踪的会话的 spid 以及跟踪文件的名称。

语法 sp_helpapptrace

示例

```
sp_helpapptrace

   traced_spid  tracer_spid  trace_file
-----
11              exited    /tmp/myfile1
13              14         /tpcc/sybase.15_0/myfile2
```

用法 sp_helpapptrace 会返回以下列：

- **traced_spid** — 正在跟踪的会话的 spid。
- **tracer_spid** — traced_spid 正在跟踪的会话的 spid。如果 tracer_spid 会话已退出，则输出 “exited”。
- **trace_file** — 跟踪文件的完整路径。

重新绑定跟踪

如果一个会话正在跟踪另一个会话，但该会话在未禁用跟踪的情况下退出，Adaptive Server 将允许新会话与早期的跟踪重新绑定。这意味着，不需要 sa 或 sso 就能完成它们启动的每个跟踪，但可以启动跟踪会话、退出，然后再重新绑定到此跟踪会话

sp_helppartition

- 说明** 列出表或索引的有关分区的信息。
- 语法** sp_helppartition [*tablename* [, { null | *indexname* | 'all' }], *partitionname*]]]
- 参数**
- tablename***
是当前数据库中表的名称。
 - null**
指定有关要列出的基表分区的信息。
 - indexname***
是当前表中索引的名称。将显示有关此索引的信息。
 - 'all'**
指定要列出所有的索引分区信息。
 - partitionname***
是基表或索引中的分区名称。

示例 **示例 1** 返回有关 **titles** 表中的数据分区的概要和详细信息。

```
sp_helppartition titles
go
```

| name | type | partition_type | partitions | partition_keys |
|--------|------------|----------------|------------|----------------|
| titles | base table | range | 5 | total_sales |

(1 row affected)

| partition_name | partition_id | pages | row_count | segment | create_date |
|----------------|--------------|-------|-----------|-----------|--------------------|
| smallsales | 1440005130 | 1 | 5 | titleseg1 | Sep 26 2005 5:44PM |
| smallsales2 | 1456005187 | 1 | 0 | titleseg2 | Sep 26 2005 5:44PM |
| smallsales3 | 1472005244 | 1 | 2 | titleseg3 | Sep 26 2005 5:44PM |
| mediumsales4 | 1488005301 | 1 | 8 | titleseg4 | Sep 26 2005 5:44PM |
| bigsales5 | 1504005358 | 1 | 3 | titleseg5 | Sep 26 2005 5:44PM |

Partition_Conditions

```
-----
VALUES <= (1000)
VALUES <= (2000)
VALUES <= (3000)
VALUES <= (10000)
VALUES <= (25000)
```

| Avg_pages | Max_pages | Min_pages | Ratio (Max/Avg) | Ratio (Min/Avg) |
|-----------|-----------|-----------|-----------------|-----------------|
| | | | | |


```
-----
          1          1          1          1.000000          1.000000
(return status = 0)
```

示例 2 返回有关 `titles` 表的概要分区信息和有关 `smallsales` 数据分区的详细信息。

```
sp_helppartition titles, null, smallsales
```

```
name          type          partition_type partitions  partition_keys
-----
titles        base table range          5 total_sales
```

```
(1 row affected)
```

```
partition_name partition_id pages  row_count segment  create_date
-----
smallsales      1440005130      1      5  titleseg1 Sep 26 2005 5:44PM
```

```
Partition_Conditions
```

```
-----
VALUES <= (1000)
(return status = 0)
```

示例 3 首先在 `my_titles` 表上创建非聚簇索引 `ncidx_local`，然后返回有关 `my_titles` 的概要分区信息和有关 `ncidx_local` 上的分区 `ncip4` 的详细信息。

```
create nonclustered index ncidx_local on my_titles(title_id) local index
(ncip1, ncip2, ncip3, ncip4, ncip5)
```

```
go
sp_helppartition my_titles, ncidx_local, ncip4
go
```

```
name          type          partition_type partitions  partition_keys
-----
ncidx_local   local index range          5 total_sales
```

```
(1 row affected)
```

```
partition_name partition_id pages  row_count segment  create_date
-----
ncip4          1584005643      1      8  default  Sep 26 2005 6:06PM
```

```
Partition_Conditions
```

```
-----
VALUES <= (10000)
(return status = 0)
```

用法

- **sp_helppartition** 列出表、索引和分区级别的与分区有关的信息。表或索引级别的分区信息包括索引类型（是本地还是全局索引）、分区类型、分区数和分区键（如果适用）。对于每个分区，该信息都包括分区名称、ID、页数、段名称、创建日期和分区条件（如果适用）。概要信息显示每个分区的页数、最小和最大页数以及平均页数和最大或最小页数之间的比率。
- 如果未提供表名，**sp_helppartition** 将列出当前数据库中所有用户表的所有者、表名、分区数和分区类型。
- 如果指定了 'all' 而非索引名称或 null，**sp_helppartition** 将列出指定表和基表的每个索引的表级别和索引级别分区信息。
- 如果指定特定索引，**sp_helppartition** 将列出该索引的索引级别信息。
 - 如果未指定分区名称，**sp_helppartition** 将显示索引中所有分区的分区级别信息，并显示分区的概要信息。
 - 如果指定了分区名称，**sp_helppartition** 将只显示该分区的分区级别信息。
- 如果只指定了表名称，**sp_helppartition** 将显示基表的表级别索引信息以及基表中所有分区的分区级别信息。
- 如果指定了 null 而未指定索引名称，同时还指定了分区名称，**sp_helppartition** 将显示基表的表级别分区信息以及指定分区的分区级别信息（但没有摘要信息）。
- 分区是使用 **create table**、**alter table** 和 **select into** 创建的。有关分区的详细信息，请参见这些命令。
- 使用 **sp_helpsegment** 显示存储分区的段上的已用页数和可用页数。

结果的精确性

- 在“页数”列中报告的值可能与实际值不同。若要确定该计数是否准确，请运行 **sp_statistics** 和 **sp_helppartition** 以比较数据页计数。**sp_statistics** 提供的计数始终是精确的。

如果 **sp_statistics** 报告的页计数与 **sp_helppartition** 报告的分区页的总和相差超过 5%，则运行以下命令之一以更新分区统计信息：

- dbcc checkdb
- dbcc checktable
- update all statistics
- update table statistics

然后，重新运行 **sp_helppartition** 以获得精确的报告。

权限 任何用户都可以执行 `sp_helppartition`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 目录系统过程 [sp_statistics](#)

命令 [alter table](#)、[create table](#)、[select into](#)

系统过程 [sp_helpsegment](#)

sp_helpcache

| | |
|----|--|
| 说明 | 显示与绑定到数据高速缓存的对象或指定高速缓存大小所需的开销量有关的信息。 |
| 语法 | <code>sp_helpcache {cache_name "cache_size[P K M G]", 'instance instance_name'}</code> |
| 参数 | <p>cache_name 是现有数据高速缓存的名称。</p> <p>cache_size 指定高速缓存的大小，指定 P 表示页，指定 K 表示千字节，指定 M 表示兆字节，指定 G 表示千兆字节。缺省值是 K。</p> <p>instance_name 要调查其高速缓存的实例的名称。</p> |
| 示例 | <p>示例 1 显示与绑定到 <code>pub_cache</code> 的项有关的信息：</p> <pre>sp_helpcache pub_cache</pre> <p>示例 2 显示创建 80 MB 数据高速缓存所需的开销量：</p> <pre>sp_helpcache "80M"</pre> <p>示例 3 显示与所有高速缓存和绑定到它们的所有项有关的信息：</p> <pre>sp_helpcache</pre> <p>示例 4 在集群环境中— 显示实例“blade1”上大小为 10 M 的高速缓存 C2 的开销：</p> <pre>sp_helpcache 'C2', '10M', 'instance blade1'</pre> |
| 用法 | <ul style="list-style-type: none"> • 若要查看服务器上所有数据高速缓存的大小、状态和 I/O 大小，请使用 sp_cacheconfig。 • 在使用 sp_cacheconfig 配置数据高速缓存时，您指定的所有内存都可用于该数据高速缓存。管理高速缓存的开销取自缺省数据高速缓存。<code>sp_helpcache</code> 显示指定大小的高速缓存所需的内存量。 • 在集群环境中— 如果不指定 <code>instance_name</code>，<code>sp_helpcache</code> 将显示所有高速缓存的信息。 • 若要将对象绑定到某一高速缓存，请使用 sp_bindcache。若要从某一高速缓存解除对特定对象的绑定，请使用 sp_unbindcache。若要解除对绑定到特定高速缓存的所有对象的绑定，请使用 sp_unbindcache_all。 • sp_cacheconfig 过程配置数据高速缓存。sp_poolconfig 过程配置数据高速缓存内的内存池。 |

- `sp_helpcache` 精确计算最高为 74GB 的开销。
- 虽然仍可以在系统 `tempdb` 上使用 `sp_bindcache`，但系统 `tempdb` 的绑定现在是非动态的。在重新启动服务器前，所做的更改将不生效，并且 `sp_helpcache` 报告状态 “P” 来表示挂起，除非您已显式将系统 `tempdb` 绑定到缺省数据高速缓存，在这种情况下报告状态 “V” 表示有效，因为缺省情况下系统 `tempdb` 已绑定到缺省数据高速缓存。

权限

任何用户都可以执行 `sp_helpcache`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_bindcache](#)、[sp_cacheconfig](#)、[sp_poolconfig](#)、[sp_unbindcache](#)、[sp_unbindcache_all](#)

sp_helpcomputedcolumn

说明 报告有关指定表中的计算列的信息。

语法 sp_helpcomputedcolumn {*tablename*}

参数 *tablename*
指定包含计算列的表的名称。

示例 此示例报告 *mytitles* 表中的计算列：

```
sp_helpcomputedcolumn mytitles
Column_Name Property
-----
sum_sales      materialized

Text
-----
AS price * total_sales materialized

(return status = 0)
```

用法

权限 任何用户都可以使用 sp_helpcomputedcolumn。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

sp_helpconfig

说明 报告有关配置参数的帮助信息。

语法 sp_helpconfig "*configname*"[, "*size*"]

参数

configname

是要查询的配置参数，或者是非唯一参数片段。

size

是内存的大小，指定 **B** 表示字节，指定 **K** 表示千字节，指定 **M** 表示兆字节，指定 **G** 表示千兆字节，或者指定 **P** 表示页。在使用时如果未指定大小的类型，*size* 将指定要使用此参数配置的实体的数目；例如锁或打开的索引的数目等。如果 *configname* 不是唯一参数名，则忽略 *size*。

示例 **示例 1** 返回有关以 “allow” 开头的所有配置选项的报告：

```
sp_helpconfig "allow"
```

```
Configuration option is not unique.
```

| option_name | config_value | run_value |
|--------------------------------|--------------|-----------|
| allow backward scans | 1 | 1 |
| allow nested triggers | 1 | 1 |
| allow procedure grouping | 1 | 1 |
| allow remote access | 1 | 1 |
| allow resource limits | 0 | 0 |
| allow sendmsg | 0 | 0 |
| allow sql server async i/o | 1 | 1 |
| allow updates to system tables | 0 | 0 |

示例 2 返回有关为 421 个对象描述符创建元数据高速缓存所需的内存量的报告：

```
sp_helpconfig "open objects", "421"
```

```
number of open objects sets the maximum number of database objects that are open at one time on SQL Server. The default run value is 500.
```

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 100 | 2147483647 | 500 | 500 | 243 |

```
Configuration parameter, 'number of open objects', will consume 207K of memory if configured at 421.
```

示例 3 返回与多少个数据库描述符将会填满 1 MB 数据库高速缓存有关的报告：

```
sp_helpconfig "open databases", "1M"
```

number of open databases sets the maximum number of databases that can be open at one time on SQL Server. The default run value is 12.

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 5 | 2147483647 | 12 | 12 | 433 |

Configuration parameter, 'number of open databases', can be configured to 28 to fit in 1M of memory.

示例 4 返回与多少个锁将占用 512 K 内存有关的报告：

```
sp_helpconfig "number of locks", "512K"
```

number of locks sets the number of available locks. The default run value is 5000.

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 1000 | 2147483647 | 5000 | 5000 | 528 |

Configuration parameter 'number of locks', can be configured to 4848 to fit in 512K of memory.

示例 5 返回与 allow updates to system tables 配置参数的状态有关的报告：

```
sp_helpconfig "allow updates to system tables"
```

allow updates to system tables allows system tables to be updated directly. The default is 0 (off).

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 0 | 1 | 0 | 0 | 0 |

用法

- sp_helpconfig 报告与配置参数有关的帮助信息，例如在该参数已设置为某个值的情况下需要多少内存。sp_helpconfig 还显示当前设置、用于该设置的内存量、缺省值以及最小和最大设置。

注释 “最大值”设置指的是参数的数据类型可以接受的最大数字，而不是指实际的可配置值。

在许多情况下，配置参数能允许的最大值是极高的。服务器的最大值通常受限于可用内存和其它资源，而与配置参数限制无关。

- `cluster options` 显示严格的全集群范围内的所有配置选项。
- 如果 `system_view` 设置为 `cluster`，`sp_helpconfig` 将显示集群中所有实例的配置信息。
- 如果 `system_view` 设置为 `instance`，`sp_helpconfig` 将显示当前实例的配置信息。
- 如果您将非唯一参数片断用于 `configname`，则 `sp_helpconfig` 返回具有其配置值和当前值的匹配参数的列表。请参见示例 1。
- `sp_helpconfig` 接受静态、动态和只读的选项。
- `sp_helpconfig 'restricted decrypt permission'` 返回以下显示内容：

```
sp_helpconfig 'restricted decrypt permission'
0 - restricted decrypt permission disabled (default).
1 - restricted decrypt permission enabled

Minimum Value Maximum Value Default Value Current Value
Memory Used Unit Type
-----
0 0 switch dynamic
```

计划元数据高速缓存配置

- 在计划服务器的元数据高速缓存配置时使用 `sp_helpconfig`。

例如，假定您计划将包含 2000 个用户索引的数据库移到不同服务器。若要确定需要为该服务器配置多少内存，才能使该服务器容纳该数据库的用户索引，请输入以下命令：

```
sp_helpconfig "open indexes", "2000"

number of open indexes sets the maximum number of indexes that can be
open at one time on SQL Server. The default run value is 500.

Minimum Value Maximum Value Default Value Current Value Memory Used
-----
100 2147483647 500 500 208

Configuration parameter, 'number of open indexes', will consume 829k of
memory if configured at 2000.
```

或者，假定您有 1MB 的内存可用于索引高速缓存，并且需要知道它将支持多少个索引描述符。请运行以下命令：

```
sp_helpconfig "open indexes", "1M"
```

```
number of open indexes sets the maximum number of indexes that can be
open at one time on SQL Server. The default run value is 500.
```

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 100 | 2147483647 | 500 | 500 | 208 |

```
Configuration parameter 'number of open indexes', can be configured to
2461 to fit in 1M of memory.
```

基于此输出，如果您具有 1MB 的内存，则可以创建可包含最多 2461 个索引描述符的索引描述符高速缓存。若要创建此高速缓存，请按如下所示设置 **number of open indexes** 配置参数：

```
sp_configure "number of open indexes", 2461
```

将 *sp_helpconfig* 与 *sybdiagdb* 一起使用（仅由 Sybase 技术支持部门使用）

注释 Sybase 技术支持部门出于调试目的，会在系统中创建 *sybdiagdb* 数据库。此数据库存有诊断配置数据，仅供 Sybase 技术支持部门使用。

下面的 *configname* 选项已被添加到 *sp_helpconfig* 中，供 Sybase 技术支持部门用于 *sybdiagdb* 数据库：

- *number of ccbs* — 是可用于协助调试的可配置动作点控制块的数目。
- *caps per ccb* — 是在一个可配置动作点中可随时配置的可配置动作点的最大数目。
- *average cap size* — 是存储与典型可配置动作点相关信息所需的估计内存量（以字节为单位）。

例如：

```
sp_helpconfig "number of ccbs"
```

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 0 | 100 | 0 | 0 | 0 |

```
sp_helpconfig "caps per ccb"
```

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| ----- | ----- | ----- | ----- | ----- |

5 500 50 50 0

sp_helpconfig "average cap size"

| Minimum Value | Maximum Value | Default Value | Current Value | Memory Used |
|---------------|---------------|---------------|---------------|-------------|
| 100 | 10000 | 200 | 200 | 0 |

权限

第 352 页的“将 [sp_helpconfig](#) 与 [sybdiagdb](#) 一起使用（仅由 Sybase 技术支持部门使用）”中指定的选项只能由 Sybase 技术支持部门使用。任何用户都可以使用其它 `configname` 选项执行 `sp_helpconfig`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_configure](#)、[sp_countmetadata](#)、[sp_monitorconfig](#)

sp_helpconstraint

说明 报告有关在指定表中使用的完整性约束的信息。

语法 sp_helpconstraint [*objname*][, *detail*]

参数 *objname*

是具有 `create table` 或 `alter table` 语句定义的一个或多个完整性约束的表的名称。

detail

返回有关约束的用户的信息或错误消息。

示例 **示例 1** 显示 pubs3 数据库中 store_employees 表的约束信息。store_employees 表具有 stores 表的外键 (stor_id) 以及自身参照 (mgr_id 引用 emp_id) :

```
sp_helpconstraint store_employees

name                                defn
-----
store_empl_stor_i_272004000        store_employees FOREIGN KEY
                                   (stor_id) REFERENCES stores(stor_id)
store_empl_mgr_id_288004057        store_employees FOREIGN KEY
                                   (mgr_id) SELF REFERENCES
                                   store_employees(emp_id)
store_empl_2560039432              UNIQUE INDEX( emp_id) :
                                   NONCLUSTERED, FOREIGN REFERENCE
```

(3 rows affected)

Total Number of Referential Constraints: 2

Details:

-- Number of references made by this table: 2

-- Number of references to this table: 1

-- Number of self references to this table: 1

Formula for Calculation:

```
Total Number of Referential Constraints
= Number of references made by this table
+ Number of references made to this table
- Number of self references within this table
```

示例 2 显示有关 pubs3..salesdetail 约束的更详细信息，包括约束类型和所有约束错误消息：

```

sp_helpconstraint titles, detail
name                                type
      defn
      msg
-----
-----
datedflt                            default value
      create default datedflt as getdate()

typedflt                            default value
      create default typedflt as "UNDECIDED"

titles_pub_id_96003373              referential constraint
      titles FOREIGN KEY (pub_id) REFERENCES publishers(pub_id)
      standard system error message number : 547

roysched_title__144003544          referential constraint
      roysched FOREIGN KEY (title_id) REFERENCES titles(title_id)
      standard system error message number : 547

salesdetai_title__368004342        referential constraint
      salesdetail FOREIGN KEY (title_id) REFERENCES titles(title_id)
      standard system error message number : 547

titleautho_title__432004570        referential constraint
      titleauthor FOREIGN KEY (title_id) REFERENCES titles(title_id)
      standard system error message number : 547

titles_800033162                   unique constraint
      UNIQUE INDEX ( title_id) : NONCLUSTERED, FOREIGN REFERENCE
      standard system error message number : 2601

(7 rows affected)

Total Number of Referential Constraints: 4

Details:
-- Number of references made by this table: 1
-- Number of references to this table: 3
-- Number of self references to this table: 0

Formula for Calculation:

```

Total Number of Referential Constraints
 = Number of references made by this table
 + Number of references made to this table
 - Number of self references within this table.

示例 3 显示 pubs3 数据库中所有表的列表:

```
sp_helpconstraint
```

| id | name | Num_referential_constraints |
|-----------|-----------------|-----------------------------|
| 80003316 | titles | 4 |
| 16003088 | authors | 3 |
| 176003658 | stores | 3 |
| 256003943 | salesdetail | 3 |
| 208003772 | sales | 2 |
| 336004228 | titleauthor | 2 |
| 896006223 | store_employees | 2 |
| 48003202 | publishers | 1 |
| 128003487 | roysched | 1 |
| 400004456 | discounts | 1 |
| 448004627 | au_pix | 1 |
| 496004798 | blurbs | 1 |

(11 rows affected)

用法

- sp_helpconstraint 可将外键和参考键截短为 30 个字符。
- sp_helpconstraint 输出完整性约束的名称和定义，以及该表使用的引用的数目。detail 选项返回有关约束的用户的信息或错误消息。
- sp_helpconstraint 显示可共享行内缺省值，与其显示常规行内缺省值的方式类似。
- 如果运行 sp_helpconstraint 时不使用任何参数，则将列出当前数据库中包含引用的所有表，并将显示每个表中索引的总数。sp_helpconstraint 基于各表中引用的数目，以降序列出这些表。
- sp_helpconstraint 只报告有关表的完整性约束信息（由 create table 或 alter table 语句定义）。它不报告使用 create index 语句创建的规则、触发器或索引的有关信息。使用 sp_help 可以查看与表的规则、触发器和索引有关的信息。
- 对于不具有用户定义的消息的约束，Adaptive Server 将报告与约束关联的系统错误消息。查询 sysmessages 可以获取该错误消息的实际文本。
- 您只能将 sp_helpconstraint 用于当前数据库中的表。

- 如果查询超过辅助扫描描述符的配置数目，Adaptive Server 将返回错误消息。您可以使用 `sp_helpconstraint` 确定所需扫描描述符的数目。有关 `number of aux scan descriptors` 配置参数的详细信息，请参见《系统管理指南》。
- 系统安全员可以防止执行 `sp_helpconstraint` 的大多数用户看到约束定义的源文本。若要将 `syscomments` 表的 `text` 列的 `select` 权限限制为对象所有者或系统管理员，应使用 `sp_configure` 将 `select on syscomments.text column` 参数设置为 0。这种权限限制要求在已评估的配置中运行 Adaptive Server。有关已评估配置的详细信息，请参见《系统管理指南》。

权限 任何用户都可以执行 `sp_helpconstraint`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 命令 `alter table`、`create table`

系统过程 `sp_configure`、`sp_help`、`sp_helpdb`、`sp_monitorconfig`

sp_helpdb

说明 报告有关特定数据库或所有数据库的信息。

语法 sp_helpdb [*dbname* [, *order*]]

参数 *dbname*

是要报告其信息的数据库的名称。如果没有此可选参数，sp_helpdb 将报告所有数据库的有关信息。*dbname* 可以包括通配符以返回匹配指定模式的所有数据库。

order

输出的缺省顺序是按 *lstart* 顺序，即创建或改变数据库的顺序。使用 *device_name* 和 *dbname* 将按 *device_name* 顺序显示 sp_helpdb。

示例 **示例 1** 显示 Adaptive Server 中所有数据库的有关信息。

```
sp_helpdb
```

| name | db_size | owner | bid | created | status |
|-----------------|----------|-------|-------|--------------|---|
| master | 24.0 MB | sa | 1 | Jan 07, 2004 | mixed log and data |
| model | 8.0 MB | sa | 3 | Jan 07, 2004 | mixed log and data |
| pubs2 | 8.0 MB | sa | 4 | Jan 21, 2004 | trunc log on chkpt, mixed log and data |
| sybssystemdb | 8.0 MB | sa | 31513 | Jan 07, 2004 | mixed log and data |
| sybssystemprocs | 112.0 MB | sa | 31514 | Jan 07, 2004 | trunc log on chkpt, mixed log and data |
| tempdb | 8.0 MB | sa | 2 | Feb 24, 2004 | select into/bulkcopy/ pllsort, trunc log on chkpt, mixed log and data |

```
(1 row affected)
```

```
(return status = 0)
```

示例 2 从 pubs2 内发出，显示有关 pubs2 数据库的信息，并且包括段信息：

```
1> use pubs2
2> go
1> sp_helpdb pubs2
2> go
```

| name | db_size | owner | dbid | created | status |
|-------|---------|-------|------|--------------|---|
| pubs2 | 20.0 MB | sa | 4 | Apr 13, 2005 | trunc log on chkpt, mixed log and data |

```
(1 row affected)
```

```
pubs2
```



```

device_fragments  size  usage          created          free kbytes
-----
master            10.0 MB  data and log   Apr 13 2005 10:29AM  2304
pubs_2_dev        10.0 MB  data and log   Apr 13 2005 10:33AM  9888

device            segment
-----
master            default
master            logsegment
master            system
pubs_2_dev        default
pubs_2_dev        logsegment
pubs_2_dev        system
pubs_2_dev        titleseg1
pubs_2_dev        titleseg2
pubs_2_dev        titleseg3
pubs_2_dev        titleseg4
pubs_2_dev        titleseg5
return status = 0)

```

示例 3 不是从 pubs2 内发出，显示有关 pubs2 数据库的信息：

```

sp_helpdb pubs2

name  db_size  owner  dbid  created          status
-----
pubs2  20.0 MB  sa     4     Jan 21, 2004   trunc log on chkpt, single
user,
                                     mixed log and data
(1 row affected)

device_fragments  size  usage          created          free kbytes
-----
master            10.0 MB  data and log   Apr 13 2005 10:29AM  2304
pubs_2_dev        10.0 MB  data and log   Apr 13 2005 10:33AM  9888
(return status = 0)

```

示例 4 将 *order* 参数指定为 *device_name*，以字母顺序显示 mydb 的设备片段，替换 *sp_helpdb* 的缺省排序顺序。

```

sp_helpdb mydb, device_name

name  db_size  owner  dbid  created          status
-----
mydb  4.5 MB   sa     5     Feb 27, 2003   no options set

(1 row affected)

device_fragments  size  usage          created          free kbytes
-----

```

```

-----
A          1.5 MB   data only   Feb 27 2003   7:50AM   1530
B          1.0 MB   log only    Feb 27 2003   7:50AM   not applicable
C          2.0 MB   data only   Feb 27 2003   7:50AM   846

```

示例 5 显示 `pubtune` 数据库的行锁升级属性集:

```
sp_helpdb pubtune
```

```

name      attribute_class  attribute          int_value char_value  comments
-----
pubtune   lock strategy         row lock promotion  NULL      PCT = 95, LWM = 300,
                                                HWM = 300

```

示例 6 显示一个数据库是否是 `status` 列下的用户创建的临时数据库:

```
sp_helpdb "mytempdb3"
```

```

name      db_size owner dbid created      status
-----
mytempdb 32.0 MB sa    7    Dec 2, 2001 select into/bulkcopy/pllsort, trunc
                                                log on chkpt, user created temp db

```

用法

- `sp_helpdb` 在给出 `dbname` 的情况下报告指定数据库的有关信息。如果没有为 `dbname` 提供任何值，则 `sp_helpdb` 报告 `master.dbo.sysdatabases` 中列出的所有数据库的有关信息。
- `sp_helpdb` 报告所有特定于数据库的属性和设置，如数据库是否脱机、压缩类型、大对象压缩级别、行内大对象长度、行锁升级阈值（如果数据库有相应的定义），等等。
- 如果在数据库中启用异步日志服务，`sp_helpdb` 输出中的 `attribute` 列将显示“`async log srv`”。
有关异步日志服务的详细信息，请参见 [sp_dboption](#) 以及《性能和调优：优化程序》中的第 3 章“高级优化工具”。
- 对于专用日志数据库中的日志段磁盘区段，`sp_helpdb` 在其每磁盘区段报告中为可用空间字段发出“`not applicable`”。`sp_helpdb` 还包括标题为 `free pages` 的列，它是日志段所具有的可用页的数目值。
- （仅限集群环境）如果指定的数据库是由远程实例拥有的本地临时数据库，`sp_helpdb` 将不会显示与设备相关的信息。
- `dbname` 可以包括通配符，以返回匹配指定模式的所有数据库。有关使用通配符的详细信息，请参见《参考手册：构件块》中的第 4 章“表达式、标识符和通配符”。

- 从 *dbname* 执行 `sp_helpdb dbname` 将会在报告中包括可用空间和段信息。
- `sp_helpdb` 显示有关数据库的属性的信息，给出属性的类、名称、整数值、字符值和注释（如果定义了任何属性）。示例 3 显示 `pubs2` 数据库的高速缓存绑定属性。
- 使用 `for load` 选项创建的数据库在 `sp_helpdb` 的输出中有状态“don't recover”。
- 在启用组件集成服务后，`sp_helpdb` 为指定数据库或所有数据库列出缺省存储位置。如果没有缺省存储位置，则显示指示“NULL”。

权限

任何用户都可以执行 `sp_helpdb`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

文档 《性能和调优：优化程序》中的第 3 章“高级优化工具”，以及《参考手册：构件块》中的第 4 章“表达式、标识符和通配符”。

命令 `alter database`、`create database`

系统过程 `sp_configure`、`sp_dboption`、`sp_rename`

sp_helpdevice

说明 报告有关特定设备或有关所有 Adaptive Server 数据库设备和转储设备的信息。

语法 sp_helpdevice [devname]

参数 devname

是要报告其信息的设备的名称。如果省略此参数，则 sp_helpdevice 报告所有设备的有关信息。

示例 **示例 1** 显示有关 Adaptive Server 中所有设备的信息：

```
1> sp_helpdevice
2> go
```

| device_name | physical_name | description | status | cntrltype | vdevno | vpn_low | vpn_high |
|-------------|-----------------------|---|--------|-----------|--------|---------|----------|
| dev1 | d:\sybdata\RV150.dev1 | special, dsync off, directio on, physical disk, 150.00 MB, Free: 0.00 MB | 2 | 0 | 2 | 0 | 76799 |
| dev2 | d:\sybdata\RV150.dev2 | special, dsync on, directio off, physical disk, 150.00 MB, Free: 130.00 MB | 16386 | 0 | 3 | 0 | 76799 |
| master | d:\sybdata\RV150.mas | special, dsync on, directio off, default disk, physical disk, 30.00 MB, Free: 0.50 MB | 3 | 0 | 0 | 0 | 15359 |
| sysprocsdev | d:\sybdata\RV150.ssp | special, dsync on, directio off, physical disk, 120.00 MB, Free: 0.00 MB | 16386 | 0 | 1 | 0 | 61439 |
| tapedump1 | \\.\TAPE0 | disk, dump device | 16 | 2 | 0 | 0 | 20000 |
| tapedump2 | \\.\TAPE1 | tape, 625 MB, dump device | 16 | 3 | 0 | 0 | 20000 |

(6 rows affected, return status = 0)

示例 2 报告有关名为 diskdump 的转储设备的信息：

```
sp_helpdevice diskdump
```

用法

- `sp_helpdevice` 显示每个设备未分配的空间量，在输出中以 `description` 列中的占位符 `Free` 指示

注释 每个设备上都可以保留少量未使用的空间，对于页大小较大的服务器更是如此。例如，在 16 K 服务器中，250 MB 设备的最后 2 MB 无法分配，`sp_helpdevice` 则会将此空间报告为可用空间。这是因为，16 K 服务器中分配单元的大小是 4 MB，而只有分配单元的倍数可以得到分配。

- `sp_helpdevice` 在给出 `devname` 的情况下显示有关指定设备的信息，或在没有给出任何参数的情况下显示有关 `master.dbo.sysdevices` 中所有设备的信息。
- `sysdevices` 表包含转储设备和数据库设备。

数据库设备可被指定为缺省设备，这意味着它们可用于数据库存储。在用户发出 `create database` 或 `alter database` 并且未指定数据库设备名或给出了关键字 `default` 时，可能发生上述情况。若要使数据库设备成为缺省数据库设备，请执行系统过程 `sp_diskdefault`。

- 使用 `disk init` 向系统添加数据库设备。使用 `sp_addumpdevice` 添加转储设备。
- 如果针对单个设备发出 `sp_helpdevice`，它将显示在该设备中已分配段的列表。
- `description` 列显示有关设备类型的信息。三种合并类型为：
 - `block device`
 - `file system device`
 - `raw device`

`status` 列中的数字与“`description`”列中的状态说明相对应。

`cntrtype` 列指定设备的控制器编号。对于磁盘或文件转储设备，`cntrtype` 为 2，对于磁带转储设备则为 3 - 8。对于数据库设备，`cntrtype` 通常为 0（除非执行的安装中具有特殊磁盘控制器类型）。

`vdevno` 列对于转储设备为 0，对于 `master` 数据库设备为 0，对于其它数据库设备为 1 或更大值。

`vpn_low` 和 `vpn_high` 列表示虚拟页号，每个虚拟页号在 Adaptive Server 的所有设备中是唯一的。

权限

任何用户都可以执行 `sp_helpdevice`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [disk init](#)、[dump database](#)、[dump transaction](#)、[load database](#)、[load transaction](#)

系统过程 [sp_addumpdevice](#)、[sp_deviceattr](#)、[sp_diskdefault](#)、[sp_dropdevice](#)、[sp_logdevice](#)

sp_helpextendedproc

说明 显示当前数据库中的扩展存储过程 (ESP) 以及与其相关联的 DLL 文件。

语法 sp_helpextendedproc [esp_name]

参数 esp_name

是扩展存储过程的名称。扩展存储过程必须是当前数据库中的过程。

示例 **示例 1** 列出 xp_cmdshell ESP 以及在其中存储其函数的 DLL 文件的名称:

```
use sybssystemprocs
go
sp_helpextendedproc xp_cmdshell

ESP Name      DLL Name
-----
xp_cmdshell  sybsyesp
```

示例 2 列出当前数据库中的所有 ESP 以及在其中存储其函数的 DLL 文件的名称:

```
sp_helpextendedproc

ESP Name      DLL Name
-----
xp_freedl    sybsyesp
xp_cmdshell  sybsyesp
```

用法

- 如果省略 *esp_name*, 则 sp_helpextendedproc 将列出数据库中的所有扩展存储过程。

- esp_name* 区分大小写。它必须匹配用于创建 ESP 的 *esp_name*。

权限 只有系统管理员才能执行 sp_helpextendedproc 来查看数据库中的所有 ESP。所有用户都可以执行 sp_helpextendedproc 来查看他们自己或数据库所有者所拥有的 ESP。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create procedure](#)、[drop procedure](#)

扩展系统过程 [xp_cmdshell](#)

系统过程 [sp_addextendedproc](#)、[sp_dropextendedproc](#)

sp_helpexternlogin

说明 (仅限组件集成服务) 报告有关外部登录名的信息。

语法 sp_helpexternlogin [server[, loginame[, rolename]]]

参数

server

是已使用 `sp_addserver` 添加到本地服务器的远程服务器的名称。

loginame

是本地服务器上的登录帐户。

rolename

是 Adaptive Server 用户指派的角色。

示例

示例 1 显示所有远程服务器、本地登录名、角色名和外部登录名：

```
sp_helpexternlogin
```

示例 2 显示名为 SSB 的服务器的本地登录名、角色名和外部登录名：

```
sp_helpexternlogin SSB
```

示例 3 显示名为 “milo” 的用户的远程服务器、本地登录名和外部登录名：

```
sp_helpexternlogin NULL, milo
```

示例 4 显示远程服务器 SSB (其中本地用户名为 “trixi”) 的外部登录名：

```
sp_helpexternlogin SSB, trixi
```

示例 5 为具有 sa_role 角色的本地用户显示远程服务器 SSB 的外部登录名：

```
sp_helpexternlogin SSB, NULL, sa_role
```

用法

- `sp_helpexternlogin` 显示所有远程服务器、用户的本地登录名、角色名和用户的外部登录名。
- 使用 `sp_addserver` 添加远程服务器。使用 `sp_addlogin` 添加本地登录名。

权限

任何用户都可以执行 `sp_helpexternlogin`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addexternlogin](#)、[sp_addlogin](#)、[sp_addserver](#)、[sp_dropexternlogin](#)、[sp_helpserver](#)

sp_helpgroup

说明 报告有关当前数据库中的特定组或所有组的信息。

语法 sp_helpgroup [grpname]

参数 *grpname*
是数据库中使用 `sp_addgroup` 创建的组的名称。

示例 1 显示有关当前数据库中所有组的信息：

```
sp_helpgroup

Group_name          Group_id
-----
hackers             16384
public              0
```

示例 2 显示有关组 “hackers” 的信息：

```
sp_helpgroup hackers

Group_name          Group_id          Users_in_group    Userid
-----
hackers             16384            ann                4
hackers             16384            judy                3
```

用法

- 若要获取有关缺省组 “public” 的报告，请用单引号或双引号将名称 “public” 引起来（“public” 是保留字）。
- 如果在指定组中没有任何成员，则 `sp_helpgroup` 显示标头，但不列出任何用户，如下所示：

```
Group_name          Group_id          Users_in_group    Userid
-----
```

权限 任何用户都可以执行 `sp_helpgroup`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[revoke](#)

系统过程 [sp_addgroup](#)、[sp_changegroup](#)、[sp_dropgroup](#)、[sp_helprotect](#)、[sp_helpuser](#)

sp_helpindex

说明 报告有关表中创建的索引的信息。报告有关计算列索引和基于函数的索引的信息。

语法 `sp_helpindex objname`

参数 *objname*
是当前数据库中表的名称。

示例 1 显示 `sysobjects` 表上索引的类型：

```
sp_helpindex sysobjects

index_name index_keys index_description index_max_rows_per_page
          index_fillfactor index_reservepagegap index_created
          index_local

sysobjects      id          clustered, unique          0
              0              0          Apr 12 2005  2:38PM
          Global Index
ncsysobjects  name, uid nonclustered, unique
              0              0          Apr 12 2005  2:38PM
          Global Index

(2 rows affected)
index_pt_name          index_ptn_seg
-----
sysobjects_1          system
ncsysobjects_1          system
```

示例 2 显示有关 `pubs2` 数据库中 `titles` 表中的索引的信息。`titles` 表已分区，但索引 `titleind` 未分区。`titleind` 是一个非聚簇（单分区）全局索引。

```
sp_helpindex titles

index_name index_keys index_description index_max_rows_per_page
          index_fillfactor index_reservepagegap index_created
          index_local

titleind      title          nonclustered          0
          Global Index

(1 row affected)
index_pt_name          index_ptn_seg
-----
titleind_1232004389          default
```

示例 3 显示有关 mysalesdetail 表的索引信息。mysalesdetail 按 ord_num 列上的散列分区。已在 ord_num 中创建了具有三个分区的聚簇本地索引。

```
sp_helpindex mysalesdetail

index_name index_keys index_description index_max_rows_per_page
          index_fillfactor index_reservepagegap index_created      index_local
-----
-----
-----
clust_idx  ord_num      clustered                                0
          0                                0 Apr 12 2005 2:38PM Local Index

(1 row affected)
index_pt_name      index_ptn_seg
-----
clust_idx_1344004788 default
clust_idx_1360004845 default
clust_idx_1376004902 default
```

示例 4 显示基于函数的索引：

```
create index sum_sales on mytitles (price * total_sales)
sp_helpindex mytitles

Object has the following indexes

index_name index_keys index_description index_max_rows_per_page
          index_fillfactor index_reservepagegap index_created      index_local
-----
-----
-----
sum_sales  sybf12_1    nonclustered                                0
          0                                0 Oct 12 2005 3:34PM Global Index

(1 row affected)
index_ptn_name      index_ptn_seg
-----
sum_sales_1724867646 default

(1 row affected)

Object has the following functional index keys

Internal_Index_Key_Name
-----
sybf12_1

(1 row affected)

Expression
-----
price * total_sales

(return status = 0)
```

- 用法
- `sp_helpindex` 列出表上的所有索引，包括通过定义由 `create table` 或 `alter table` 语句定义的唯一键或主键约束创建的索引。
 - `sp_helpindex` 显示指派给表上索引的任何属性（例如，高速缓存绑定）。
 - `sp_helpindex` 显示：
 - 每个索引的分区信息。
 - 索引是本地索引还是全局索引，聚簇索引还是非聚簇索引。
 - 索引的 `max_rows_per_page` 设置。
 - 有关仅数据锁定表上的聚簇索引的信息。
仅数据锁定表中聚簇索引的索引 ID (`indid`) 不等于 1。
 - 键的列顺序，指示键是按升序还是降序排列。
 - 空间管理属性值。
 - 按照该顺序的键列的名称。只显示降序顺序。例如，如果在列 `a ASC, b DESC, c ASC` 上有一个索引，则 “`index_keys`” 显示 “`a, b DESC, c`”。

权限 任何用户都可以执行 `sp_helpindex`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `create index`、`drop index`、`update statistics`
系统过程 `sp_help`、`sp_helpkey`、`sp_helppartition`

sp_helpjava

| | |
|----|--|
| 说明 | 显示有关在数据库中安装的 Java 类和关联 JAR 的信息。 |
| 语法 | sp_helpjava ["class"[, <i>java_class_name</i> [, "detail" "depends"]] "jar", <i>jar_name</i> [, "depends"]]] |
| 参数 | <p>"class" "jar" 指定是否显示有关类或 JAR 的信息。“class”和“jar”均为关键字，因此引号是必需的。</p> <p><i>java_class_name</i> 需要有关其信息的类的名称。类必须是安装在数据库中的系统类或用户定义的类。</p> <p>detail 指定您要查看有关类的详细信息。</p> <p>depends 列出依赖 JAR 中指定类的所有数据库对象，包括 SQLJ 函数、SQLJ 存储过程、视图、Transact-SQL 存储过程和表。</p> <p><i>jar_name</i> 要查看其有关信息的 JAR 的名称。JAR 必须使用 installjava 安装在数据库中。</p> |

示例 1 显示数据库中安装的所有类以及关联 JAR 文件的名称：

```
sp_helpjava
```

示例 2 显示所有类的名称：

```
sp_helpjava "class"
```

示例 3 显示有关 Address 类的详细信息：

```
sp_helpjava "class", Address, detail
Class
-----
Address

(1 row affected)
Class Modifiers
-----
public synchronized

Implemented Interfaces
-----
java.io.Serializable
```



```

Extended Superclass
-----
java.lang.Object

Constructors
-----
public Address()
public Address(java.lang.String,java.lang.String)

Methods
-----
public final native java.lang.Class java.lang.Object.getClass()
public native int java.lang.Object.hashCode()
public boolean java.lang.Object.equals(java.lang.Object)
public java.lang.String java.lang.Object.toString()
public final native void java.lang.Object.notify()
public final native void java.lang.Object.notifyAll()
public final native void java.lang.Object.wait(long) throws
java.lang.InterruptedException
public final void java.lang.Object.wait(long,int) throws
java.lang.InterruptedException
public final void java.lang.Object.wait() throws
java.lang.InterruptedException
public java.lang.String Address.display()
public void Address.removeLeadingBlanks()

Fields
-----
public java.lang.String Address.street
public java.lang.String Address.zip

```

用法

- `depends` 参数列出某个或某些类（如果类在 SQLJ 例程的 `create` 语句的 `external name` 子句中列出或用作数据库中列的数据类型）的依赖性。

权限 任何用户都可以执行 `sp_helpjava`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

文档 有关数据库中 Java 的详细信息，请参见《Adaptive Server Enterprise 中的 Java》。

命令 remove java

实用程序 extractjava、installjava

sp_helpjoins

说明 列出两个表或视图中适于连接的列。

语法 `sp_helpjoins lefttab, righttab`

参数 *lefttab*
是第一个表或视图。

righttab
是第二个表或视图。参数的顺序无关紧要。

示例 1 显示 `sales` 和 `salesdetail` 表中适于连接的列的列表：

```
sp_helpjoins sales, salesdetail
a1      a2      b1      b2      c1      c2
      d1      d2      e1      e2      f1      f2
      g1      g2      h1      h2
-----
-----
-----
stor_id stor_id ord_num ord_num NULL  NULL
      NULL  NULL  NULL  NULL  NULL  NULL
      NULL  NULL  NULL  NULL
```

示例 2 显示 `sysobjects` 和 `syscolumns` 表中适于连接的列的列表：

```
sp_helpjoins sysobjects, syscolumns
a1  a2  b1  b2  c1  c2  d1  d2  e1  e2
      f1  f2  g1  g2  h1  h2
-----
-----
id  id  NULL NULL NULL NULL NULL NULL NULL NULL
      NULL NULL NULL NULL NULL NULL
```

用法

- `sp_helpjoins` 显示的列对来自两个来源中的一个。`sp_helpjoins` 检查当前数据库中的 `syskeys` 表，以确定是否用 `sp_foreignkey` 对这两个表定义了任何外键，然后查看是否用 `sp_commonkey` 对这两个表定义了任何公用键。如果 `sp_helpjoins` 没有在那里找到任何外键或公用键，则它用相同用户定义的数据类型检查键。如果该检查失败，则它检查具有相同名称和数据类型的列。

- `sp_helpjoins` 不创建任何连接。

权限 任何用户都可以执行 `sp_helpjoins`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_commonkey](#)、[sp_foreignkey](#)、[sp_helpkey](#)、[sp_primarykey](#)

sp_helpkey

说明 报告有关特定表或视图的主键、外键或公用键的信息，或有关当前数据库中的所有键的信息。

语法 `sp_helpkey [tablename]`

参数 *tablename*
是当前数据库中表或视图的名称。如果没有指定名称，则过程报告当前数据库中定义的所有键的有关信息。

示例 显示有关当前数据库中定义的键的信息。“object_keys”和“related_keys”列引用构成键的列名：

```
sp_helpkey

keytype object          related_object object_keys          related_keys
-----
primary authors        -- none --          au_id,*,*,*,*,*,*,*  *,*,*,*,*,*,*,*
foreign titleauthor authors au_id,*,*,*,*,*,*,*  au_id,*,*,*,*,*,*
                                     *,*
```

- 用法**
- `sp_helpkey` 列出有关引用表 *tablename* 的所有主键、外键和公用键定义的信息；或者，如果省略 *tablename*，则列出有关数据库中所有键的信息。使用 `sp_primarykey`、`sp_foreignkey` 和 `sp_commonkey` 系统过程定义这些键。
 - `sp_helpkey` 不提供有关 `create table` 语句定义的 `unique` 或 `primary key` 整体性约束的信息。使用 `sp_helpconstraint` 确定为表定义了哪些约束。
 - 创建键以使在数据库设计中为隐式的逻辑关系成为显式的逻辑关系，以便应用程序可以使用这些信息。
 - 如果您指定一个对象名，`sp_helpkey` 将按照 Adaptive Server 规则来查找对象：
 - 如果您没有指定所有者名，并且拥有具有指定名称的对象，则 `sp_helpkey` 将报告有关该对象的信息。
 - 如果您没有指定所有者名，并且不拥有具有该名称的对象，但数据库所有者拥有具有该名称的对象，则 `sp_helpkey` 将报告该数据库所有者的对象的有关信息。
 - 如果您或数据库所有者均不拥有具有指定名称的对象，则 `sp_helpkey` 会报告一个错误情况，即使在不同所有者的数据库中存在具有该名称的对象时也是如此。
 - 如果您和数据库所有者都拥有具有指定名称的对象，并且您要访问数据库所有者的对象，则以 `dbo.objectname` 格式指定该名称。

- 用所有者的名称限定其他数据库用户（既不是您自己也不是数据库所有者）所拥有的对象，如“mary.myproc”中所示。

权限 任何用户都可以执行 sp_helpkey。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** [create trigger](#)

系统过程 [sp_commonkey](#)、[sp_foreignkey](#)、[sp_primarykey](#)

sp_helplanguage

说明 报告有关特定替代语言或所有语言的信息。

语法 sp_helplanguage [*language*]

参数 *language*

是用于显示有关其信息的替代语言的名称。

示例 1 显示有关替代语言“french”的信息：

```
sp_helplanguage french

langid dateformat datefirst upgrade      name
      alias
      months
      shortmonths
      days
-----
1      dmy          1          0          french
french
janvier, février, mars, avril, mai, juin, juillet, août, septembre,
octobre, novembre, décembre
jan, fév, mar, avr, mai, jui, juil, août, sep, oct, nov, déc
lundi, mardi, mercredi, jeudi, vendredi, samedi, dimanche
```

示例 2 显示有关安装的所有替代语言的信息：

```
sp_helplanguage
```

用法

- sp_helplanguage 在给出指定语言的情况下报告有关该语言的信息，或者在没有提供任何语言的情况下报告有关 master.dbo.syslanguages 中所有语言的信息。

权限 任何用户都可以执行 sp_helplanguage。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addlanguage](#)、[sp_droplanguage](#)、[sp_setlangalias](#)

sp_helplog

说明 报告包含事务日志第一页的设备的名称。

语法 sp_helplog

参数 无。

示例 将 “master” 报告为设备名：

```
sp_helplog
```

```
In database 'master', the log starts on device 'master'.
```

用法 • sp_helplog 显示包含当前数据库中事务日志的第一页的设备的名称。

权限 任何用户都可以执行 sp_helplog。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [alter database](#)、[create database](#)

系统过程 [sp_helpdevice](#)、[sp_logdevice](#)

sp_helpobjectdef

| | |
|----|--|
| 说明 | (仅限组件集成服务) 报告远程对象定义的所有者、对象和类型信息。 |
| 语法 | sp_helpobjectdef [<i>objname</i>] |
| 参数 | <p><i>objname</i></p> <p>是 <code>sysattributes</code> 表中定义的对象名称。<i>objname</i> 可以是以下任意一种形式:</p> <ul style="list-style-type: none"> • <code>dbname.owner.object</code> • <code>dbname..object</code> • <code>owner.object</code> • <code>object</code> <p><i>dbname</i> 和 <i>owner</i> 是可选的。<i>object</i> 是必需的。如果未提供 <i>owner</i>, 则 <i>owner</i> 缺省为当前用户名。如果提供 <i>dbname</i>, 则它必须是当前数据库, 并且 <i>owner</i> 必须提供或用占位符 <code>dbname..object</code> 标记。应该用引号将由多部分组成的 <i>objname</i> 引起来。</p> |
| 示例 | <p>示例 1 显示当前数据库中的所有远程对象定义:</p> <pre>sp_helpobjectdef</pre> <p>示例 2 显示数据库所有者所拥有的 <code>tb1</code> 表的远程对象定义:</p> <pre>sp_helpobjectdef "dbo.tb1"</pre> |
| 用法 | <ul style="list-style-type: none"> • 如果未提供 <i>objname</i>, <code>sp_helpobjectdef</code> 将显示所有远程对象定义。 • 在 <i>objname</i> 参数中不允许采用服务器名。 |
| 权限 | 任何用户都可以执行 <code>sp_helpobjectdef</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示: |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `create table`、`create existing table`、`drop table`

系统过程 `sp_addobjectdef`、`sp_dropobjectdef`、`sp_helpserver`

sp_helpremotelogin

说明 报告有关特定远程服务器或所有远程服务器的登录的信息。

语法 sp_helpremotelogin [*remoteserver*[, *remotename*]]

参数 *remoteserver*
是报告有关其远程登录信息的服务器的名称。

remotename
是远程服务器上特定远程用户的名称。

示例 **示例 1** 显示有关远程服务器 GATEWAY 的所有远程用户的信息：

```
sp_helpremotelogin GATEWAY
```

示例 2 显示有关本地服务器已知的所有远程服务器的所有远程用户的信息：

```
sp_helpremotelogin
```

用法

- **sp_helpremotelogin** 在给出 *remoteserver* 的情况下报告指定服务器的远程登录的有关信息，或者在没有提供任何参数的情况下报告所有服务器的远程登录的有关信息。

权限 任何用户都可以执行 sp_helpremotelogin。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addremotelogin](#)、[sp_droptremotelogin](#)、[sp_helpserver](#)

sp_helprotect

| 说明 | 报告数据库对象、用户、组或角色的权限。 | | | | | | | |
|---------|---|---------|---------|--------|--------|-----------|--------|-----------|
| 语法 | <code>sp_helprotect [name[, username[, "grant" [, "none" "granted" "enabled" role_name[, permission_name]]]]]</code> | | | | | | | |
| 参数 | <p><i>name</i> 是当前数据库中的表、视图、存储过程、SQLJ 存储过程、SQLJ 函数、用户定义的函数、用户、用户定义的角色或组的名称。如果没有提供名称，则 <code>sp_helprotect</code> 报告数据库中的所有权限。</p> <p><i>username</i> 是当前数据库中的用户、组或角色的名称。如果不使用名称，Adaptive Server 会生成空报告。</p> <p><i>grant</i> 显示使用 <code>grant</code> 选项授予 <i>name</i> 和 <i>username</i> 的特权。如果 <i>username</i> 为空，<code>sp_helprotect</code> 会列出用 <code>grant</code> 选项对 <i>name</i> 授予的特权。</p> <p><i>none</i> 在确定授予的权限时忽略授予用户的角色。</p> <p><i>granted</i> 在确定授予的权限时包括授予该用户的所有角色的有关信息。</p> <p><i>enabled</i> 在确定授予的权限时包括由该用户激活的所有角色的有关信息。</p> <p><i>role_name</i> 列出通过 <i>role_name</i> 授予的特权。</p> <p><i>permission_name</i> 允许 <code>sp_helprotect</code> 提供有关给定数据库中授予的任何特定权限的信息（授权者名称、被授者名称、表 / 列名称和可授权性）。 此参数的值可以是 <code>sysprotects.action</code> 列中的任意值。</p> | | | | | | | |
| 示例 | <p>示例 1 此 <code>grant</code> 和 <code>revoke</code> 语句系列（执行 <code>sp_helprotect titles</code>）将导致显示以下内容：</p> <pre>grant select on titles to judy grant update on titles to judy revoke update on titles(price) from judy grant select on publishers to judy with grant option go sp_helprotect titles</pre> <table> <thead> <tr> <th>grantor</th> <th>grantee</th> <th>type</th> <th>action</th> <th>object</th> <th>column</th> <th>grantable</th> </tr> </thead> </table> | grantor | grantee | type | action | object | column | grantable |
| grantor | grantee | type | action | object | column | grantable | | |

| | | | | | | |
|-----|------|-------|--------|------------|-------------|-------|
| dbo | judy | Grant | Select | titles | All | FALSE |
| dbo | judy | Grant | Update | titles | advance | FALSE |
| dbo | judy | Grant | Update | titles | notes | FALSE |
| dbo | judy | Grant | Update | titles | pub_id | FALSE |
| dbo | judy | Grant | Update | titles | pubdate | FALSE |
| dbo | judy | Grant | Update | titles | title | FALSE |
| dbo | judy | Grant | Update | titles | title_id | FALSE |
| dbo | judy | Grant | Update | titles | total_sales | FALSE |
| dbo | judy | Grant | Update | titles | type | FALSE |
| dbo | judy | Grant | Select | publishers | all | TRUE |

示例 2 发出以下 grant 语句将导致 sp_helpprotect 显示以下内容:

```
grant select, update on titles(price, advance)
to mary
with grant option
go
sp_helpprotect titles
```

| grantor | grantee | type | action | object | column | grantable |
|---------|---------|-------|--------|--------|---------|-----------|
| dbo | mary | Grant | Select | titles | advance | TRUE |
| dbo | mary | Grant | Select | titles | price | TRUE |
| dbo | mary | Grant | Update | titles | advance | TRUE |
| dbo | mary | Grant | Update | titles | price | TRUE |

示例 3 显示 “judy” 在数据库中具有的所有权限:

```
sp_helpprotect judy
```

示例 4 显示 “csmith” 对 sysusers 表具有的任何权限, 以及 “csmith” 是否具有允许 “csmith” 向其他用户授予权限的 with grant option:

```
sp_helpprotect sysusers, csmith, "grant"
```

| grantor | grantee | type | action | object | column | grantable |
|---------|---------|-------|------------|---------------|--------|-----------|
| dbo | doctor | Grant | Delete | sysusers | All | FALSE |
| dbo | doctor | Grant | Insert | sysusers | All | FALSE |
| dbo | doctor | Grant | References | sysusers | All | FALSE |
| dbo | doctor | Grant | Select | sysattributes | All | FALSE |

```
(1 row affected)
(return status = 0)
```

示例 5 显示 doctor 角色在数据库中具有的权限的有关信息：

```
sp_helprotect doctor
```

| grantor | grantee | type | action | object | column | grantable |
|---------|---------|-------|------------|---------------|--------|-----------|
| dbo | doctor | Grant | Delete | sysusers | All | FALSE |
| dbo | doctor | Grant | Insert | sysusers | All | FALSE |
| dbo | doctor | Grant | References | sysusers | All | FALSE |
| dbo | doctor | Grant | Select | sysattributes | All | FALSE |

```
(1 row affected)
(return status = 0)
```

示例 6 显示授予 “csmith” 的所有角色的有关信息：

```
sp_helprotect csmith, null, null, "granted"
```

| grantor | grantee | type | action | object | column | grantable |
|---------|---------|-------|------------|----------|--------|-----------|
| dbo | csmith | Grant | Update | sysusers | All | FALSE |
| dbo | doctor | Grant | Delete | sysusers | All | FALSE |
| dbo | doctor | Grant | Insert | sysusers | All | FALSE |
| dbo | doctor | Grant | References | sysusers | All | FALSE |

```
(1 row affected)
(return status = 0)
```

示例 7 显示授予 “rpillai” 的所有活动角色的有关信息：

```
sp_helprotect rpillai, null, null, "enabled"
```

| grantor | grantee | type | action | object | column | grantable |
|---------|---------|-------|--------|---------------|--------|-----------|
| dbo | public | Grant | Select | sysattributes | All | FALSE |

```
(1 row affected)
(return status = 0)
```

示例 8 建议 SQLJ 函数访问是公共的：

```
sp_helprotect function_sqlj
Implicit grant to public for SQLJ functions.
```

示例 9 使用 sysprotects.action 中的 “Decrypt” 操作：

```
sp_helpprotect @permission_name = "Decrypt"
```

| grantor | grantee | type | action | object | column | grantable |
|---------|----------|-------|---------|----------|--------|-----------|
| sal | hr_login | Grant | Decrypt | employee | ssn | TRUE |
| sal | hr_role | Grant | Decrypt | employee | ssn | FALSE |

用法

- **sp_helpprotect** 报告对于某一数据库对象的权限。如果提供了 *username* 参数，则只报告此用户对该数据库对象的权限。如果 *name* 不是一个对象，则 **sp_helpprotect** 将检查它是否为用户、组、角色或权限名称。如果它是用户、组或角色，则 **sp_helpprotect** 列出该用户、组或角色的权限。
- **sp_helpprotect** 只查找当前数据库中的对象和用户。
- 如果没有指定诸如 *granted*、*enabled*、*none* 或 *role_name* 之类的可选值，Adaptive Server 将会返回由当前指定用户激活的所有角色的有关信息。
- 如果指定用户不是当前用户，则 Adaptive Server 会返回授予该指定用户的所有角色的有关信息。
- 显示的信息始终包括授予该指定用户所在组的权限。
- 授予权限时，系统管理员被视作对象所有者。如果系统管理员授予对另一用户的对象的权限，则相应的所有者名称将作为授权者出现在 **sp_helpprotect** 输出中。

sp_helpprotect 和加密列

sp_helpprotect 可以报告加密列、加密密钥及用户的新信息，如下所示：

- 表和列 — 报告已被授予 **decrypt** 权限的表和列。
- 加密密钥 — 报告已被授予 **select** 权限的加密密钥。
- 用户 — 报告已被授予 **create encryption key** 权限的用户。

权限

任何用户都可以执行 **sp_helpprotect** 来查看自己的权限。只有 SSO 可以查看所有用户的权限。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[revoke](#)

系统过程 [sp_activeroles](#)、[sp_displayroles](#)

sp_helpsegment

说明 报告有关特定段或有关当前数据库中的所有段的信息。

语法 sp_helpsegment [segname]

参数 *segname*

是需要有关其信息的段的名称。如果省略此参数，将显示有关当前数据库中所有段的信息。

示例 1 报告有关当前数据库中所有段的信息：

```
sp_helpsegment
segment name                                status
-----
0 system                                    0
1 default                                  1
2 logsegment                               0
3 seg1                                      0
4 seg2                                      0
5 seg3                                      0
6 seg4                                      0
```

示例 2 报告名为 `order_seg` 的段的有关信息。其中包括绑定到此段的数据库表和索引（当前在表 / 索引级别指定此段的表 / 索引）以及当前在此段中的对象（实际位于此段中的分区）。此外，此示例还报告了此段中的总页数、可用页数、已用页数和保留页数：

```
sp_helpsegment seg1
segment name                                status
-----
3 seg1                                      0

device          size          free_pages
-----
pubs_dev1      2.0MB          240

Objects on segment 'seg1':

table_name      index_name      indid  partition_name
-----
fictionsales    fictionsales    0      q1
pb_fictionsales pb_fictionsales 0      lov

Objects currently bound to segment 'seg1':
```



```

table_name  index_name  indid
-----  -
new_titles  new_titles      0

total_size  total_pages  free-pages  used_pages  reserved pages
-----  -
2.0MB      256          240         16          0

```

示例 3 报告有关 `default` 段的信息。关键字 `default` 必须用引号引起来。因为长度原因，已对输出进行了简化。

```

          sp_helpsegment "default"

segment  name      status
-----  -
         1  default      1

device   size      free_pages
-----  -
master   14.0MB    303
pubs_dev1 2.0MB    240
pubs_dev2 2.0MB    232
pubs_dev3 2.0MB    232
pubs_dev4 2.0MB    240

Objects on segment 'default':

table_name  index_name  indid  partition_name
-----  -
au_pix      au_pix      0      au_pix_864003078
au_pix      tau_pix     0      tau_pix_864003078
...
titles      title_idx   0      p1
titles      title_idx   0      p2
titles      title_idx   0      p3
titles      title_idx   0      title_idx_985051514

```

Objects currently bound to segment 'default':

```

table_name  index_name  indid
-----  -
au_pix      au_pix      0
...
titleauthor  titleidind  3
titles      title_idx   1

```

| total_size | total_pages | free_pages | used_pages | reserved_pages |
|------------|-------------|------------|------------|----------------|
| ----- | ----- | ----- | ----- | ----- |
| 22.0MB | 2816 | 1247 | 1569 | 0 |

示例 4 报告有关存储事务日志的段的信息:

```
1> sp_helpsegment "logsegment"  
2> go
```

| segment name | status |
|--------------|--------|
| ----- | ----- |
| 2 logsegment | 0 |

| device | device size |
|-----------|-------------|
| ----- | ----- |
| master | 14.0MB |
| pubs_dev1 | 2.0MB |
| pubs_dev2 | 2.0MB |
| pubs_dev3 | 2.0MB |
| pubs_dev4 | 2.0MB |

| free_pages |
|------------|
| ----- |
| 1239 |

Objects on segment 'logsegment':

| table_name | index_name | indid | partition_name |
|------------|------------|-------|----------------|
| ----- | ----- | ----- | ----- |
| syslogs | syslogs | 0 | syslogs_8 |

Objects currently bound to segment 'logsegment':

| table_name | index_name | indid |
|------------|------------|-------|
| ----- | ----- | ----- |
| syslogs | syslogs | 0 |

| total_size | total_pages | free_pages | used_pages | reserved_pages |
|------------|-------------|------------|------------|----------------|
| ----- | ----- | ----- | ----- | ----- |
| 22.0MB | 2816 | 1239 | 13 | 0 |

(return status = 0)

用法

- `sp_helpsegment` 在给出 `segname` 的情况下显示有关指定段的信息，或在没有给出任何参数的情况下显示有关当前数据库中所有段的信息。
- 在首次创建数据库时，Adaptive Server 自动创建 `system`、`default` 和 `logsegment` 段。使用 `sp_addsegment` 向当前数据库中添加段。
- 如果您指定来自专用日志数据库的某一日志段来用于 `segname` 参数，则 `sp_helpsegment` 报告该日志段中的可用页数。
- `system`、`default` 和 `logsegment` 段分别被编号为 0、1 和 2。
- “`status`”列指示哪个段是空间的缺省池。使用 `sp_placeobject` 或者 `create table` 或 `create index` 命令的 `on segment_name` 子句将对象放置于特定段上。
- 如果该表不具有聚簇索引，则 “`indid`”列为 0；如果该表具有聚簇索引，则 “`indid`”列为 1。

权限

任何用户都可以执行 `sp_helpsegment`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `create index`、`create table`

系统过程 `sp_addsegment`、`sp_dropsegment`、`sp_extendsegment`、`sp_helpdb`、`sp_helpdevice`、`sp_placeobject`

sp_helpserver

说明 报告有关特定远程服务器或所有远程服务器的信息。

语法 sp_helpserver [server]

参数 server
是需要有关其信息的远程服务器的名称。

示例 1 显示有关远程服务器 GATEWAY 的信息：

```
sp_helpserver GATEWAY
```

示例 2 显示有关本地 Backup Server 的信息：

```
sp_helpserver SYB_BACKUP
name          network_name  security_mechanism  server_principal
              class
      status
id cost
-----
-----
-----
-----
-----
SYB_BACKUP    SYB_BACKUP          NULL                NULL
              NULL
timeouts, no net password encryption, writable, enable login redirection 1
              NULL
```

示例 3 显示有关本地服务器已知的所有远程服务器的信息：

```
sp_helpserver
```

用法

- sp_helpserver 报告有关 master.dbo.sys.servers 中所有服务器的信息，或在指定 server 的情况下报告有关特定远程服务器的信息。
- 如果安装了组件集成服务 (CIS)，sp_helpserver 会列出安全性机制、服务器主体名称以及每个服务器的服务器类。

权限 任何用户都可以执行 sp_helpserver。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_addserver](#)、[sp_dropserver](#)、[sp_helpremotelogin](#)、[sp_serveroption](#)

sp_helpsort

| | |
|----|---------------------------------|
| 说明 | 显示 Adaptive Server 的缺省排序顺序和字符集。 |
| 语法 | sp_helpsort |
| 参数 | 无。 |

示例 1 对于类 1（单字节）字符集，sp_helpsort 显示服务器的缺省排序顺序的名称、其字符集以及其主排序值的表。在 7 位终端上，显示内容如下：

```
sp_helpsort
Sort Order Description
-----
Character Set = 1, iso_1
      ISO 8859-1 (Latin-1) - Western European 8-bit character set.
Sort Order = 50, bin_iso_1
      Binary sort order for the ISO 8859/1 character set (iso_1).
Characters, in Order
-----
! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
@ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _
` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~
! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
@ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _
` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~
```

示例 2 在 8 位终端上，显示内容如下：

```
Sort Order Description
-----
Character Set = 1, iso_1
      ISO 8859-1 (Latin-1) - Western European 8-bit character set.
Sort Order = 50, bin_iso_1
      Binary sort order for the ISO 8859/1 character set (iso_1).
Characters, in Order
-----
! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
@ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _
` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~
¡ ¢ £ ¤ ¥ ¦ § ¨ © ª « ¬ ® ¯ ° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾ ¿ À
Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ à
á â ã ä å æ ç è é ê ë ì í î ï ñ ò ó ô õ ö ÷ ø ù ú û ü ý þ ÿ
```

示例 3 对于类 2（多字节）字符集，将不列出字符，但包括该字符集的说明。例如：

```
Sort Order Description
-----
Character Set = 140, euc_jis
    Japanese. Extended Unix Code mapping for JIS-X0201
    (hankaku katakana) and JIS-X0208 (double byte) roman,
    kana, and kanji.
Class 2 character set
Sort Order = 50, bin_eucjis
    Binary sort order for Japanese using the EUC JIS
    character set as a basis.
```

示例 4 对于不区分大小写的字符集，将列出不区分大小写的可用排序顺序的名称和 ID：

```

Name                                ID
-----
nocase_eucgb                        52
nocase_cp936                        52
nocase_gb18030                      52
nocase_eucjis                       52
nocase_sjis                         52
nocase_deckanji                    52
```

用法

- 二进制排序顺序是缺省设置。

权限

任何用户都可以执行 `sp_helpsort`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

sp_helptext

说明 显示编译对象的源文本，以及用户定义的函数、计算列或基于函数的索引定义的源文本。

语法 sp_helptext *objname*[*grouping_num*][, *numlines*[, *printopts*]]

参数

objname

是要显示其源文本的编译对象的名称。编译对象必须在当前数据库中。

grouping_num

是在 *objname* 表示一组过程时标识单个过程的整数。该参数告知 sp_helptext 显示组中指定过程的源文本。

在使用 *printopts* 参数时，该参数还指定生成 SQL 文本的起始行。

注释 视图、缺省值和其它非过程对象是永远不被分组的；只将 *number* 用于过程组。

numlines

指定要为其生成 SQL 文本的行数。如果还将参数 *printopts* 与 *showsql* 一起使用，*numlines* 将指定要显示的 SQL 文本的行数；如果将 *printopts* 与 *context* 一起使用，*numlines* 将被视为起始行号周围的上下文块宽度。

printopts

支持逗号分隔的输出格式属性，由用户选择。可按逗号分隔字符串的形式以任意顺序指定一个或多个这种输出选项：

- *showsql* — 生成编译对象的格式化 SQL 输出。如果 *showsql* 未显示在 *printopts* 列表中，将不调用此属性。
- *linenumbers* — 为每行 SQL 输出生成行号。
- *comments* — 将行号作为注释字段生成 (*/*<nnn>*/*)，因此如果需要，生成的 SQL 仍可用于重新创建编译对象，而无需进行进一步的编辑。
- *context* — 在指定的起始行号周围生成输出的上下文块。如果没有提供 *numlines* 参数或该参数为空，则会提供包括五行的缺省上下文块（在相关行号前后生成）。
- *noparams* — 禁止显示自动生成的参数信息。使用此输出选项可以只生成编译对象的 SQL 输出的相关部分。

- `ddlgen` — 将 SQL 文本生成为 DDL 脚本，使用 `use database` 命令和 `drop object` 命令开始输出。删除并重新创建所有编译对象，如存储过程、触发器、视图和缺省值等。

输出选项 `ddlgen` 和 `context` 是互斥的指示符。如果一起使用，将引发错误。若要在显示 SQL 文本的上下文块时获取行号，请使用 `context` 和 `linenumbers` 指示符。

示例

示例 1 显示 `pub_idrule` 的源文本。因为此规则在 `pubs2` 数据库中，所以从 `pubs2` 执行此命令：

```
sp_helptext pub_idrule

# Lines of Text
-----
1

text
-----
create rule pub_idrule
as @pub_id in ("1389", "0736", "0877",
              "1622", "1756")
   or @pub_id like "99[0-9][0-9]"
```

示例 2 显示 `sp_helptext` 的源文本。因为系统过程存储于 `sybssystemprocs` 中，所以从 `sybssystemprocs` 执行此命令：

```
sp_helptext sp_helptext
```

示例 3 显示 `myproc` 组行为（其中未指定 `number` 参数）的源文本。过程编号显示在文本旁：

```
sp_helptext myproc

# Lines of Text
-----
2
number
text
-----
1
create procedure myproc; as select 1
2
create procedure myproc;2 as select 2
(2 rows affected)
```

示例 4 显示 myproc 的源文本，并且指定 *myproc* 组中的过程，但不显示分组数。

```
sp_helptext myproc, 2
# Lines of Text
-----
1
text
-----
create procedure myproc;2 as select 2
```

示例 5 为 sp_help 生成文本：

```
sp_helptext sp_help,NULL,NULLM 'showsqli'
```

示例 6 若要为 sp_help 生成文本并生成行号：

```
sp_helptext sp_help, NULL,NULL, 'showsqli,linenumbers'
```

示例 7 若要在始于行 25 的 7 行上下文块中为 sp_help 生成文本，并在注释块中生成输入：

```
sp_helptext sp_help,25,7, 'showsqli,comments,context'
```

示例 8 为 sp_droptabledef 生成文本，并以可用于重现此过程的独立 DDL 脚本形式生成输出：

```
sp_helptext sp_droptabledef,NULL,NULL, 'showsqli,ddlgen'
-----
use sybssystemprocs

-----
IF EXISTS (SELECT 1 FROM sysobjects
WHERE name = 'sp_droptabledef'
AND type = 'P'
DROP PROCEDURE sp_droptabledef
-----
/*Sccsud="%Z%generic/sproc/src/%M%%I%%G"*/
/*
**Omni only
*/
create procedure sp_droptabledef
    @tablename varchar(92) /*tablename*/
as begin
    declare @status int
    exec @status = sp_dropobjectdef @tablename
    return(@status)
end
-----
```

```
(return status = 0)
```

示例 9 对使用分隔标识符创建的视图使用 `sp_helptext`。不需要 `set quoted_identifier on` 即可提取定义视图的 SQL。无需将其设置为 `ON` 即可使用分隔标识符创建对象。

```
set quoted_identifier ON
-----
create table "t one"
    (c1 int,
     "c two" varchar(10),
     "c three" int)
-----
create table "t two"
    ("t2 one" int,
     "t2 two" varchar(10),
     t2_three int)
-----
create view "v one" as
    select * from "t one"
    UNION
    select "t2 one","t2 two",t2_three
    from "t two"
-----
```

示例 10

对其子过程之一 `sp_showtext_output` 使用 `sp_helptext`，以标识行 813 周围的 SQL 源代码的上下文：

```
sp_helptext sp_showtext_output,
813,NULL,'context,linenumbers,showsql'
```

用法

- 显示 `syscomments` 的源文本时，`sp_helptext` 截断尾随空格
- `sp_helptext` 输出 `syscomments` 中由编译对象占用的行号（每行长度为 255 个字符），后跟编译对象的源文本。
- 源文本使用 `char(255)` 来显示，因此显示文本中会出现尾随空格。`syscomments` 中存储的文本可能不包括这些尾随空格。`syscomments` 按原样存储文本，所以其它应用程序或工具可能未包括这些尾随空格。因此，不应使用 `sp_helptext` 来获取存储文本的副本。应改用 `defncopy` 等其它工具。
- `sp_helptext` 在当前数据库的 `syscomments` 表中查找源文本。
- 可以使用 `sp_hidetext` 对源文本进行加密。
- 在 `sp_helptext` 对一组过程进行操作时，它输出来自 `syscomments` 的编号列以及源文本。

- 系统安全员可以防止执行 `sp_helptext` 的大多数用户看到编译对象的源文本。若要将 `syscomments` 表的 `text` 列的 `select` 权限限制为对象所有者或系统管理员，应使用 `sp_configure` 将 `select on syscomments.text column` 参数设置为 0。这种权限限制要求在已评估的配置中运行 Adaptive Server。有关已评估配置的详细信息，请参见《系统管理指南》。
- 即使在 `ddlgen` 模式中使用 `sp_helptext` 时，也需要 `showsql` 输出选项。
- 确保数据库的数据类型能隐式转换成要检索的对象类型。
- 如果文本已隐藏或不在 `syscomments` 中，将引发一条错误消息。但是，如果请求上下文块输出，并且文本丢失或处于隐藏状态，将输出一条报告丢失文本的消息，而不会引发错误。
- 如果使用 `ddlgen` 输出选项生成的文本包含对其它对象的引用，而这些对象在执行生成的脚本时并不存在（例如临时表），则这些文本可能仍无法正确创建编译对象。
- 如果编译对象包含 `select *` 语句，则它通常会反映此语句引用的表的整个列列表。
- 您可以为使用带引号的标识符创建的编译对象生成 SQL 文本，但如果该编译对象包含 `select *` 语句，则在 Adaptive Server 向 `syscomments` 中写入文本后，显示的扩展列列表将具有带中括号的标识符。

例如：

```
[this column], [column name with space]
```

对于自身为分隔标识符或使用分隔标识符的编译对象，在为其生成文本时无需 `set quoted_identifier ON`。

权限

任何用户都可以执行 `sp_helptext`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 `sp_checksourc`、`sp_configure`、`sp_hidetext`

sp_helpthread

说明 显示当前线程池配置。

语法 sp_helpthread [*pool_name*]

参数 *pool_name*

要显示的池的名称。如果 *pool_name* 为空，sp_helpthread 会显示所有池的配置信息。

示例 **示例 1** 显示有关所有池的信息：

```

sp_helpthread
name          type      size  idle_timeout
-----
-----
pubs_pool     Engine    2      100
                                     NULL
syb_blocking_pool  RTC      4      0
    A pool dedicated to executing blocking calls
syb_default_pool  Engine    1      100
    The default pool to run query sessions
syb_system_pool  RTC      4      0
    The I/O and system task pool

```

示例 2 显示有关 pubs_pool 的信息：

```

sp_helpthread pubs_pool
name          type      size  idle_timeout  description
-----
-----
pubs_pool     Engine    2      100           NULL

thread_id    osthread_id  state  affinity  instance_id
-----
-----
12           1248065856  IDLE   NULL      0
13           1237576000  IDLE   NULL      0

```

用法 • sp_helpthread 从 monThread 监控表中收集有关其报告的信息。

• sp_helpthread 仅在线程化模式中生成输出。

权限 任何用户都可以发出 sp_helpthread。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

sp_helpthreshold

说明 报告与当前数据库中的所有阈值或特定段的所有阈值相关联的段、可用空间值、状态和存储过程。

语法 `sp_helpthreshold [segname]`

参数 *segname*
是当前数据库中段的名称。

示例 1 显示日志段上的所有阈值：

```
sp_helpthreshold logsegment
```

示例 2 显示当前数据库中所有段上的所有阈值：

```
sp_helpthreshold
```

示例 3 显示缺省段上的所有阈值。请注意，应使用引号将保留字“default”引起来：

```
sp_helpthreshold "default"
```

用法

- `sp_helpthreshold` 显示当前数据库中所有段的阈值信息。如果提供段的名称，`sp_helpthreshold` 将列出该段中的所有阈值。
- 对于最后机会阈值而言 `status` 列是 1，而对于所有其它阈值而言该列是 0。如果数据库没有在单独的段上存储其事务日志，则数据库不具有最后机会阈值。

权限 任何用户都可以执行 `sp_helpthreshold`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** `sp_addthreshold`、`sp_droptreshold`、`sp_helpsegment`、`sp_modifythreshold`、`sp_thresholdaction`

sp_helpuser

说明 报告有关特定用户、组或别名的信息或者有关当前数据库中所有用户的信息。另外还标识用户拥有的对象和用户定义数据类型。

语法 sp_helpuser [*name_in_db* [, *display_object*]]

参数

name_in_db

是 `NULL` 或当前数据库中的有效用户的名称。

display_object

列出当前表中的所有对象以及由 *name_in_db* 拥有的用户定义的数据类型。如果 *name_in_db* 是 `NULL`，则会列出调用方的对象和用户定义数据类型。对象的输出包括 `object_name`、`object_type` 和 `create_date`，按 `object_type` 和 `object_name` 排序。用户定义的数据类型的输出包括用户类型名称。

示例

示例 1 显示有关当前数据库中所有用户的信息：

```
sp_helpuser

Users_name  ID_in_db    Group_name  Login_name
-----
ann         4           hackers    ann
dbo         1           public     sa
guest       2           public     NULL
judy        3           hackers    judy
```

示例 2 显示有关数据库所有者（用户名“dbo”）的信息：

```
sp_helpuser dbo

Users_name      ID_in_db    Group_name  Login_name
-----
dbo             1           public     sa
Users aliased to user.
Login_name
-----
andy
christa
howard
linda
```


示例 3 显示用户 bill 拥有的对象

```

sp_helpuser bill, display_object
Object_name          Object_type          Create_date
-----
proc_update_titles  stored procedures    Apr 28 2007 04:47PM
author              user table           Apr 27 2007 04:47PM
publisher           user table           Apr 27 2007 05:47PM
titles              user table           Apr 27 2007 06:47PM
vw_author_in_ca     view                 Apr 27 2007 05:47PM

```

示例 4 显示数据库所有者 (DBO) 拥有的对象:

```

sp_helpuser 'dbo', display_object
Object_name          Object_type          Create_date
-----
enter_key            encryption key       Sep 7 2007 03:37PM
sysalternatives     system table         Jul 17 2007 09:25AM
sysattributes       system table         Jul 17 2007 09:25AM
syscolumns          system table         Jul 17 2007 09:25AM
.....
sysquerymetrics     view                 Jul 17 2007 09:25AM

```

用法

- `sp_helpuser` 报告有关当前数据库的所有用户的信息。如果指定 `name_in_db`，则 `sp_helpuser` 只报告指定用户的有关信息。
- 如果在当前数据库的 `sysusers` 表中未列出指定用户，则 `sp_helpuser` 检查该用户是否是其他用户的别名或是组名。

权限

任何用户都可以执行 `sp_helpuser`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_adduser](#)、[sp_dropuser](#)、[sp_helpgroup](#)

sp_hidetext

说明 隐藏指定编译对象的源文本，以及计算列和基于函数的索引键的文本。
`sp_hidetext` 还会解密用户定义的函数的文本。

语法 `sp_hidetext [objname[, tabname[, username]]]`

参数 *objname*
指定要隐藏源文本的编译对象。

tabname
指定要隐藏源文本的表或视图的名称。

username
指定拥有要隐藏其源文本的编译对象的用户名。

示例 **示例 1** 隐藏当前数据库中所有编译对象的源文本：

```
sp_hidetext
```

示例 2 隐藏 Mary 所拥有的用户定义的存储过程 `sp_sort_table` 的源文本：

```
sp_hidetext @objname = "sp_sort_table",  
            @username = "Mary"
```

示例 3 隐藏存储过程 `pr_phone_list` 的源文本：

```
sp_hidetext "pr_phone_list"
```

示例 4 隐藏表 `my_tab` 上定义的所有检查约束、缺省值和触发器的源文本：

```
sp_hidetext @tablename = "my_tab"
```

示例 5 隐藏视图 `my_vu` 的源文本以及隐藏表 `my_tab` 上定义的所有检查约束、缺省值和触发器的源文本：

```
sp_hidetext "my_vu", "my_tab"
```

示例 6 隐藏 Tom 所拥有的所有编译对象的源文本：

```
sp_hidetext @username = "Tom"
```

用法

- `sp_hidetext` 隐藏指定编译对象的源文本。

警告！ 在执行 `sp_hidetext` 前，应确保具有源文本的备份。执行 `sp_hidetext` 的结果是不可逆的。

- 如果未提供任何参数，则 `sp_hidetext` 隐藏当前数据库中所有编译对象的源文本。

- Adaptive Server 允许谓词所有者或 SSO 隐藏谓词的文本。隐藏的 `syscomments.text` 不可用于 `sp_helprotect`。必须对用户发出警告，使其知道如果隐藏了文本，`sp_helprotect` 的 `expand_predicate` 选项将输出空谓词。
- 如果在使用 `sp_hidetext` 之后执行跨平台的 `dump` 和 `load`，则必须手动删除并重新创建所有隐藏对象。

权限

任何用户都可以使用 `sp_hidetext` 隐藏自己的编译对象的源文本。只有数据库所有者或系统管理员才能隐藏其他用户所拥有的编译对象的源文本或者使用 `sp_hidetext` 时不含任何参数。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 [dump database](#)、[dump transaction](#)、[load database](#)、[load transaction](#)
文档 有关隐藏源文本的详细信息，请参见《Transact-SQL 用户指南》。
系统过程 [sp_checksourc](#)

sp_import_qpgroup

| | |
|----|--|
| 说明 | 将抽象计划从用户表导入到抽象计划组中。 |
| 语法 | <code>sp_import_qpgroup <i>tab</i>, <i>usr</i>, <i>group</i></code> |
| 参数 | <p><i>tab</i></p> <p>是从其复制计划的表的名称。可以用 <code>dbname..tablename</code> 格式指定数据库名，但不能指定所有者名。总长度最多包含 255 个字符。</p> <p><i>usr</i></p> <p>是在导入抽象计划时其 ID 应被指派给这些抽象计划的用户的名称。</p> <p><i>group</i></p> <p>是包含要被导入的计划的抽象计划组的名称。</p> |
| 示例 | <p>将计划从表 <code>moveplans</code> 复制到 <code>new_plans</code> 组中，并且向它们提供数据库所有者的用户 ID：</p> <pre>sp_import_qpgroup moveplans, dbo, new_plans</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_import_qpgroup</code> 将计划从用户表复制到 <code>sysqueryplans</code> 中的抽象计划组。对于 <code>sp_export_qpgroup</code>，它可以用于在服务器和数据库之间复制抽象计划组，或者用于复制属于一个用户的计划并向它们指派另一用户的 ID。 • <code>sp_import_qpgroup</code> 创建抽象计划组（如果在执行过程时该组不存在）。 • 如果在执行 <code>sp_import_qpgroup</code> 时存在某一抽象计划组，则它不能包含用于指定用户的任何计划。<code>sp_import_qpgroup</code> 不通过检查查询文本来确定查询是否已存在于该组中。如果需要将一个用户的计划导入某一组（该组中为该用户已存在某些计划），请执行以下操作： <ul style="list-style-type: none"> • 使用 <code>sp_import_qpgroup</code> 将这些计划导入一个新计划组。 • 使用 <code>sp_copy_all_qplans</code> 将这些计划从新创建的组复制到目标组中。<code>sp_copy_all_qplans</code> 会检查查询以确保不创建任何重复计划。 • 如果您不再需要为导入创建的组，则用 <code>sp_copy_all_qplans</code> 删除该组中的计划，然后用 <code>sp_drop_qpgroup</code> 删除该组。 • 若要创建一个空表以批量复制抽象计划，请使用： <pre>select * into load_table from sysqueryplans where 1 = 2</pre> |
| 权限 | 只有系统管理员或数据库所有者才能执行 <code>sp_import_qpgroup</code> 。 |

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 [create plan](#)

系统过程 [sp_copy_all_qplans](#)、[sp_copy_qplan](#)、[sp_drop_all_qplans](#)、[sp_drop_qpgroup](#)、[sp_export_qpgroup](#)、[sp_help_qpgroup](#)

sp_indsuspect

说明 检查用户表中排序顺序更改后的恢复期间标记为可疑的索引。

语法 sp_indsuspect [*tab_name*]

参数 *tab_name*
是将被检查的用户表的名称。

示例 检查 newacct 表中是否存在标记为可疑的索引：

```
sp_indsuspect newacct
```

用法

- 使用 sp_indsuspect 时如果不含任何参数，将创建当前数据库中具有因排序顺序更改需被重建的索引的所有表的列表。如果使用 *tab_name* 参数，sp_indsuspect 将检查指定表中在排序顺序更改后的恢复期间标记为可疑的索引。
- 使用 sp_indsuspect 列出所有可疑索引。表所有者或系统管理员可以使用 dbcc reindex 检查列出的索引的完整性并根据需要重建它们。

权限 任何用户都可以执行 sp_indsuspect。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令 dbcc**

sp_jreconfig

说明

管理 Java PCA/JVM。启用或禁用参数和指令、更改配置值和报告配置值。

注释 您可以安全地更改 `pca_jvm_module_path`、`pca_jvm_work_dir`、`pca_jvm_dbg_agent_port`、`pca_jvm_java_dbg_agent_suspend`、`pca_jvm_java_options` 和 `pca_jvm_netio` 参数。请勿使用 `sp_jreconfig` 更改其它参数或指令，除非 Sybase 技术支持部门指示这样做。

语法

```
sp_jreconfig {
    add array_arg, new_string |
    array_clear array_arg |
    array_enable array_arg |
    array_disable array_arg |
    delete array_arg, string_value |
    disable { directive | argument | array_arg, string_value } |
    enable { directive | argument | array_arg, string_value } |
    list { list_type [, formatted ] | units | units, units_type[, formatted ] } |
    reload_config |
    report { directive[, formatted ] | directive, args[, formatted ]
            |argument[, formatted ] } |
    update { argument, old_value, new_value } }
```

参数

add

在参数数组中添加一个新参数。add 仅用于 `units_type` 为数组的参数。

array_arg

是 `units_type` 为数组的参数的名称。

new_string

是新数组元素的字符串值。

array_clear

删除参数数组中的所有元素。

array_enable

启用参数数组中的所有元素。将每个数组元素设置为启用。

array_disable

禁用但不删除参数数组中的所有元素。将每个元素设置为禁用。

delete

从参数数组中删除现有元素。delete 仅用于 `units_type` 为数组的参数。

disable

禁用指定的指令或参数。

string_value

标识要删除、启用或禁用的指定参数数组中的数组元素。

directive

是活动数据库的名称。

argument

是有效参数的名称。

enable

启用指令或参数。

list

以 `sp_jreconfig list, directives` 或 `sp_jreconfig list, enabled` 之类的形式列出相关参数的组。此外，还会以诸如 `sp_jreconfig list, units, string` 之类的形式列出特定类型的所有参数。若要查看当前的所有 *units_types* 值，请使用 `sp_jreconfig list, units`。

formatted

设置显示列表的格式以改善可读性；较长的值可能会被截断。

注释 在设置了格式的报道中，改善可读性的过程可能造成宽列被截断。此外，列标题可能会被覆盖，可能会与实际的表名称不匹配。如果将对输出进行分析或潜在的数据截断不可接受，请勿为报告设置格式。

list_type

指定列表的类型。其值包括：

- `directives` — 指令列表
- `enabled` — 已启用的参数的列表
- `disabled` — 已禁用的参数的列表
- `argnames` — 参数名、各参数的 *units_type* 以及各自所属指令的列表

units

与 *list* 搭配使用时，生成当前使用的 *units_type* 的列表。

units_type

是参数的类型。每个参数都有一个标识其类型的 *units_type*。其值包括：

- `switch`
- `string`
- `number`
- `array`

reload_config

将配置从 `sybpcidb` 表重新装载到内存中。请参见《Adaptive Server Enterprise 中的 Java》的第 2 章“管理 Java 环境”中的“恢复 `sybpcidb` 的缺省配置值”。

report

根据提供的参数创建报告。通常用于生成某个参数的报告，以查看其当前值以及其是否启用。也可用于生成指令或其参数的报告。

directive

是任何有效的指令。

args

是与报告一起用于为指定指令生成参数名列表的关键字。例如：

```
sp_jreconfig report, "PCA_JVM", "args"
```

update

修改 `units_type` 为 `string`、`number` 或 `array` 的参数的字符串或数值。当 `units_type` 为 `switch` 时，不能修改参数。

old_value

是标识要更新的现有参数或数组元素的字符串或数值。

new_value

是定义新参数或数组元素的字符串或数值。

示例

示例 1 为 `PCA_JVM_OPT` 指令生成已设置格式的报告：

```
sp_jreconfig "report", "PCA_JVM_OPT", "formatted"
```

示例 2 为 `PCA_JVM_OPT` 指令生成参数报告：

```
sp_jreconfig "report", "PCA_JVM_OPT", "args"
```

示例 3 为参数 `pca_jvm_netio` 生成报告。

```
sp_jreconfig "report", "pca_jvm_netio"
```

示例 4 为与“`pca_jvm`”匹配的所有参数生成报告。部分参数名为所有匹配的参数生成报告。

```
sp_jreconfig "report", "pca_jvm_"
```

示例 5 生成列表—显示所有指令及其状态的列表：启用或禁用。

```
sp_jreconfig "list", "directives"
```

示例 6 生成列表—显示所有参数及其单元类型和指令的列表。

```
sp_jreconfig "list", "argnames", "formatted"
```

示例 7 生成列表— 显示当前启用的所有参数的列表。

```
sp_jreconfig "list", "enabled"
```

示例 8 生成列表— 显示所有数组参数的格式化列表。

```
sp_jreconfig "list", "units", "array", "formatted"
```

示例 9 生成列表— 显示参数单元类型的列表。缺省情况下，此命令的报告已设置格式。使用 “-formatted” 选项会生成错误。

```
sp_jreconfig "list", "units"
```

示例 10 启用指令和参数— 启用 PCA_JVM_WORK_DIR 指令。您可以使用部分指令名称，前提是它包含的信息足以唯一标识相应的指令。

```
sp_jreconfig "enable", "PCA_JVM_WORK_DIR"  
sp_jreconfig "enable", "WORK_DIR"
```

示例 11 启用指令和参数— 启用 pca_jvm_netio 参数。

```
sp_jreconfig "enable", "pca_jvm_netio"
```

示例 12 禁用指令和参数— 禁用 WORK_DIR 指令。此示例使用部分指令名称，该名称包含的信息必须足以唯一标识相应的指令。

```
sp_jreconfig "disable", "WORK_DIR"
```

注释 禁用指令会导致其参数表现出禁用时的行为，但不会更改为基本状态。

示例 13 禁用指令和参数— 禁用 pca_jvm_netio 参数。

```
sp_jreconfig "disable", "pca_jvm_netio"
```

示例 14 禁用指令和参数— 禁用 PCA_JVM_WORK_DIR 中的数组元素。需要路径，但不需要权限屏蔽。请参见《Adaptive Server Enterprise 中的 Java》中的第 8 章 “使用 Java 访问文件和网络”。

```
sp_jreconfig "disable", "pca_jvm_work_dir",  
"/some/path"
```

示例 15 更新字符串、数字和数组参数— 更新字符串参数。此示例更新 pca_jvm_log_filename 参数的文件位置。

```
sp_jreconfig "update", "pca_jvm_log_filename", "/old/path/filename.log",  
"/new/path/filename.log"
```

注释 update 选项不能与指令或开关参数一起使用，因为这些项目无法修改。

示例 16 更新字符串、数字和数组参数—更新数字参数。对于存储过程，数字值必须用引号括起来（成为字符串形式）。Adaptive Server 将它们存储为数值。

```
sp_jreconfig "update", "pca_jvm_min_port", "1026",
"2056"
```

示例 17 更新字符串、数字和数组参数—对于 PCA_JVM_WORK_DIR 指令，work_dir 值由路径和可选的权限屏蔽组成。虽然权限屏蔽是可选的，但必须包括原始字符串路径才能标识 work_dir。权限屏蔽是可选的。如果不提供权限屏蔽，系统将使用缺省屏蔽（0666 的八进制等效项）。示例 a 没有设置权限屏蔽；它使用缺省屏蔽。示例 b 和 c 均将权限屏蔽设置为 0644。

```
[a] sp_jreconfig "update", "pca_jvm_work_dir",
"/old/path", "/new/working/directory"
```

```
[b] sp_jreconfig "update", "pca_jvm_work_dir",
"/old/path", "/new/working/directory(u=rw,go=r) "
```

```
[c] sp_jreconfig "update", "pca_jvm_work_dir",
"/old/path", "/new/working/directory(u+w,ugo+r) "
```

示例 18 添加数组元素—向 PCA_JVM_WORK_DIR 指令中的 pca_jvm_work_dir 参数数组中添加新元素。示例使用缺省屏蔽。示例 b 和 c 均将权限屏蔽设置为 0644。（屏蔽按从左到右的方向求值。）

```
[a] sp_jreconfig "add", "pca_jvm_work_dir",
"/new/working/directory"
```

```
[b] sp_jreconfig "add", "pca_jvm_work_dir",
"/new/working/directory(u=rw,go=r) "
```

```
[c] sp_jreconfig "add", "pca_jvm_work_dir",
"/new/working/directory(u+w,ugo+r) "
```

示例 19 删除数组元素—删除 pca_jvm_work_dir 中的数组元素。

```
sp_jreconfig "delete", "pca_jvm_work_dir",
"/new/working/directory"
```

注释 若要在 PCA_JVM_WORK_DIR 指令中删除 pca_jvm_work_dir 中的元素，可以指定部分字符串，但提供的字符串必须能标识唯一的记录。不需要权限屏蔽；只需提供路径即可，即使 work_dir 元素最初是使用特定权限屏蔽定义的也不例外。

示例 20 启用或禁用数组中的所有元素— 启用 `pca_jvm_work_dir` 数组中的所有元素。

```
sp_jreconfig "array_enable", "pca_jvm_work_dir"
```

示例 21 启用或禁用数组中的所有元素— 禁用 `pca_jvm_work_dir` 数组中的所有元素。

```
sp_jreconfig "array_disable", "pca_jvm_work_dir"
```

示例 22 清除数组中的所有记录— 删除 `pca_jvm_work_dir` 数组中的所有记录并创建一个空数组。

```
sp_jreconfig "array_clear", "pca_jvm_work_dir"
```

示例 23 重新状态缺省配置值— 将存储在 `sybpcidb` 中的配置值装载到内存中。

```
sp_jreconfig "reload_config"
```

用法

启用和禁用指令与参数

启用和禁用像开关一样工作的指令：

- *指令启用时*— Adaptive Server 使用为每个参数配置的值（启用或禁用）。这是存储在 `sybpcidb` 中的值。
- *指令禁用时*— 尽管每个参数的基值保留在 `sybpcidb` 中，但 Adaptive Server 不考虑为每个参数配置的值（启用或禁用），而是将该指令的所有参数都视作已禁用。

参数可以分别启用或禁用。参数类型包括：

- *switch*— 此类参数启用或禁用某项功能。例如，如果用于日志记录的参数已启用，则会生成一个日志文件；如果用于日志记录的参数已禁用，则不会生成日志文件。
- *字符串*— 此类参数用于字符串和数值。启用字符串或数字参数可确保 Adaptive Server 使用配置的值。禁用字符串或数字参数意味着，Adaptive Server 会忽略配置的值，而使用缺省值。配置的值和缺省值可以相同，也可以不同。
- *数组*— 数组参数是相关字符串参数的集合，其中每个参数均可单独启用或禁用。禁用个别字符串参数（又称元素）后，其值将被忽略，表现的行为就像已删除该元素一样。启用后，参数值包括在集合中，并处于活动状态。

可按需启用或禁用数组参数；无需先删除值随后再重新输入。

表 1-13: *sp_jreconfig* 的配置指令

| 指令 | 说明 |
|-----------------------------|---|
| PCA_JVM_MODULE_PATH | JVM 共享对象库的路径。如果使用的 JRE 不是 Sybase 提供的，则必须将此参数配置为指向可由 PCA/JVM 访问的位置。这可以是绝对路径，也可以是扩展 \$\$SYBASE 的相对路径。如果是绝对路径，则路径应以 “/”（在 UNIX 上）或 “\”（在 Windows 上）开头。否则，Adaptive Server 将采用相对路径并在 \$\$SYBASE 下进行查找。 |
| PCA_JVM_OPT | 除非有 Sybase 技术支持部门的指导，否则不要更改缺省值。 |
| PCA_JVM_DIR_OPTIONS | JVM 对 ROOT 和 TEMP 目录使用的目录定义。除非您是资深用户或有 Sybase 技术支持部门的指导，否则不要更改这些值。 警告！ 使用此指令时应十分谨慎。PCA_JVM_DIR_OPTIONS 指令中的 <code>pca_jvm_tmp_dir</code> 应始终指向系统临时目录。更改此位置可能会带来重大安全风险。JVM 允许打开文件以执行读写操作，并允许在该目录中创建文件。 |
| PCA_JVM_WORK_DIR | 配置 JVM 信任的目录。此参数由文件系统中的特定位置集合组成，在这些位置中，Java 程序类可执行某些文件 I/O 操作。每个目录均可具有可选的权限屏蔽来定义各目录中允许的文件 I/O 操作。 |
| PCA_JVM_MIN_JNI_VERSION | 最低的向后兼容 JNI 版本。 |
| PCA_JVM_LOGGING | JRE/VM 记录选项。 |
| PCA_JVM_EXT_CLASS_LOADER | 全局和数据库扩展类装载程序。 |
| PCA_JVM_JAVA_OPTIONS | 正常和扩展的 Java 启动选项。 |
| PCA_JVM_JAVA_DBG_AGENT_PORT | Java VM 调试代理端口号（用于通过 Java 调试程序调试 Java 应用程序）。有关详细信息，请参见《Adaptive Server Enterprise 中的 Java》。 |
| PCA_JVM_SYS_DEVICE_PATH | 特定于平台的系统设备目录（在 Solaris 上需要）。 |

表 1-14: *PCA_JVM_MODULE_PATH* 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|----------------------------------|--------|-------|------|---|
| <code>pca_jvm_module_path</code> | string | 因平台而异 | 已启用 | JVM 共享库的位置，使用 \$\$SYBASE 下的相对路径或完全限定的文件名。 |

表 1-15: *PCA_JVM_OPT* 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|--|--------|-----|------|-----------------------------|
| <code>pca_jvm_abort</code> | switch | On | 已启用 | 一旦失败即中止全部 abort(2)。 |
| <code>pca_jvm_allow_unchecked_sockops</code> | switch | 不适用 | 已禁用 | 允许执行未选中的套接字操作。 |
| <code>pca_jvm_debug</code> | switch | 不适用 | 已禁用 | 报告 PCA_DEBUG 请求。 |
| <code>pca_jvm_except</code> | switch | 不适用 | 已启用 | 报告除 PCA/VM JNI/JVM 调用之外的信息。 |
| <code>pca_jvm_heap_ratio</code> | string | 0.3 | 已启用 | VM 堆 /PCI 内存比。 |

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|----------------------------------|--------|------|------|---|
| pca_jvm_jvmti | switch | 不适用 | 已禁用 | Java VM 工具接口。 |
| pca_jvm_min_port | number | 1026 | 已启用 | 允许 VM 网络支持。 |
| pca_jvm_netio | switch | 不适用 | 已禁用 | 允许 VM 网络支持。 |
| pca_jvm_report | switch | 不适用 | 已禁用 | 报告 PCA/VM JNI/JVM 调用。 |
| pca_jvm_security_manager_enabled | switch | 不适用 | 已禁用 | 在 PCA/JVM 中启用 SecurityManager。 |
| pca_jvm_sigcache_density | number | 100 | 已启用 | PCA/VM 签名高速缓存目标密度。 |
| pca_jvm_sigcache_enabled | switch | 不适用 | 已启用 | 启用 PCA/VM 签名高速缓存。 |
| pca_jvm_sigcache_fixed_ratio | number | 50 | 已启用 | PCA/VM 签名高速缓存的大小（固定百分比）。 |
| pca_jvm_sigcache_freeboard | number | 30 | 已启用 | 高速缓存扫描时的 PCA/VM 签名高速缓存空间恢复百分比。 |
| pca_jvm_sigcache_size | number | 512 | 已启用 | PCA/VM 签名高速缓存的大小，单位为 KB。 |
| pca_jvm_sigcache_size_type | number | 1 | 已启用 | PCA/VM 签名高速缓存 size_type 0:AS_PCT 1:Kbyte 2:Mbyte。 |
| pca_jvm_sigcache_washcycle | number | 1000 | 已启用 | PCA/VM 签名高速缓存清晰守护程序周期时间 (ms)。 |
| pca_jvm_sigcache_washdaemon | switch | 不适用 | 已禁用 | 启用 PCA/VM 签名高速缓存清洗守护程序。 |
| pca_jvm_strace | switch | 不适用 | 已启用 | 对非模拟 VM 处理生成堆栈跟踪。 |

表 1-16: PCA_JVM_DIR_OPTIONS arguments

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|------------------|--------|-------|------|---------------------------|
| pca_jvm_root_dir | string | 因平台而异 | 已启用 | 系统根目录的绝对路径。是文件 I/O 所必需的。 |
| pca_jvm_tmp_dir | string | 因平台而异 | 已启用 | 系统临时目录的绝对路径。是文件 I/O 所必需的。 |

表 1-17: PCA_JVM_WORK_DIR 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|------------------|-------|-------|------|---|
| pca_jvm_work_dir | array | 因平台而异 | 已禁用 | 允许 JVM 在其中执行文件 I/O 的绝对路径（和可选的权限屏蔽）。请参见《Adaptive Server Enterprise 中的 Java》中的第 8 章“使用 Java 访问文件和网络”。 |

表 1-18: PCA_JVM_MIN_JNI_VERSION 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|-------------------------|--------|-------------------|------|-----------------|
| pca_jvm_min_jni_version | string | 'JNI_VERSION_1_2' | 已启用 | 最低的向后兼容 JNI 版本。 |

表 1-19: PCA_JVM_LOGGING 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|----------------------|--------|---------------------|------|------------------------|
| pca_jvm_ase_logging | switch | 不适用 | 已启用 | 配置 Adaptive Server 记录。 |
| pca_jvm_log_filename | string | '/tmp/Java_ym.log1' | 已禁用 | VM 用于记录的完全限定文件名。 |

表 1-20: PCA_JVM_EXT_CLASS_LOADER 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|---------------------------------|-------|------|------|-------------|
| pca_jvm_ext_class_loader_global | array | none | 已禁用 | 全局扩展类装载程序。 |
| pca_jvm_ext_class_loader_dbase | array | none | 已禁用 | 数据库扩展类装载程序。 |

表 1-21: PCA_JVM_JAVA_OPTIONS 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|----------------------|-------|--------------------------------------|------|------------------------|
| pca_jvm_java_options | array | "-Djava.awt.headless=true" | 已启用 | 以随机模式运行 Java。 |
| pca_jvm_java_options | array | "-Djava.compiler=JIT" | 已启用 | 强制 JIT 编译和优化。 |
| pca_jvm_java_options | array | "-XX:+CITune:" | 已禁用 | JIT 编译器（仅限 1.4）所花的时间。 |
| pca_jvm_java_options | array | "-XX:+Use AltSigs" | 已禁用 | 此选项似乎会使 J2SE 崩溃。 |
| pca_jvm_java_options | array | "-XX:CodeCacheExpansion Size=512000" | 已启用 | 代码高速缓存扩展大小。 |
| pca_jvm_java_options | array | "-Xbatch" | 已禁用 | 禁用的后台编译。 |
| pca_jvm_java_options | array | "-Xcheck:jni" | 已启用 | 对 JNI 函数执行附加检查。 |
| pca_jvm_java_options | array | "-Xfuture" | 已禁用 | 执行严格检查，预测未来缺省值。 |
| pca_jvm_java_options | array | "-Xincgc" | 已禁用 | 启用增量碎片收集。 |
| pca_jvm_java_options | array | "-Xint" | 已禁用 | 仅以解释模式执行。 |
| pca_jvm_java_options | array | "-Xloggc:./myGClog" | 已禁用 | 将 GC 状态和时间戳记录到文件中。 |
| pca_jvm_java_options | array | "-Xmixed" | 已禁用 | 混合模式执行（缺省设置）。 |
| pca_jvm_java_options | array | "-Xms64m" | 已禁用 | 设置初始 Java 堆大小。 |
| pca_jvm_java_options | array | "-Xmx64m" | 已禁用 | 设置最大 Java 堆大小。 |
| pca_jvm_java_options | array | "-XnoClassgc" | 已禁用 | 禁用类碎片收集。 |
| pca_jvm_java_options | array | "-Xprof" | 已禁用 | 输出 CPU 分析数据。 |
| pca_jvm_java_options | array | "-Xrs" | 已禁用 | 减少 Java/VM 对 OS 信号的使用。 |

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|----------------------|-------|----------------------------|------|---|
| pca_jvm_java_options | array | “-Xshare:auto” | 已禁用 | 配置共享类数据（设置为 auto、off 或 on）。 |
| pca_jvm_java_options | array | “-Xss64m” | 已禁用 | 设置 Java 线程堆栈大小。 |
| pca_jvm_java_options | array | -XX:MaxPermSize | 已禁用 | 设置永久堆的最大大小 |
| pca_jvm_java_options | array | “-enablessystemassertions” | 已启用 | 启用 Java/VM 系统断言 — 仅适用于使用 Sun HotSpot (TM) JavaVM 的平台。 |
| pca_jvm_java_options | array | “-esa” | 已启用 | 启用所有系统断言 — 仅适用于使用 Sun HotSpot (TM) JavaVM 的平台。 |
| pca_jvm_java_options | array | “-verbose:class” | 已禁用 | 在 JRE/VM 内部装载类。 |
| pca_jvm_java_options | array | “-verbose:gc” | 已禁用 | 碎片收集统计信息。 |
| pca_jvm_java_options | array | “-verbose:jni” | 已禁用 | Java 本地接口 (JNI) 调用。 |

表 1-22: PCA_JVM_JAVA_DBG_AGENT_PORT 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|--------------------------------|--------|------|------|-----------------------|
| pca_jvm_java_dbg_agent_port | number | 8000 | 已禁用 | 配置端口号和 Java VM 调试代理。 |
| pca_jvm_java_dbg_agent_suspend | switch | 不适用 | 已禁用 | 启用时开始挂起 Java VM 调试代理。 |

表 1-23: PCA_JVM_SYS_DEVICE_PATH 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|-------------------------|-------|-------|-------|----------------------|
| pca_jvm_sys_device_path | array | 因平台而异 | 因平台而异 | Sun OS 的内部系统选项。请勿更改。 |

权限 只有系统管理员才能执行 sp_jreconfig 来更改 PCA/JVM 的设置。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **存储过程** [sp_pciconfig](#)

sp_ldapadmin

说明 创建或列出 LDAP URL 搜索字符串、检验 LDAP URL 搜索字符串或登录名，或者指定访问帐户和可调优的 LDAPUA 相关参数。

语法 `sp_ldapadmin command [, option1 [, option2]]`
Valid *command* [, *option1* [, *option2*]] options are:

```
'set_primary_url', 'url'
'set_secondary_url', 'url'
'set_dn_lookup_url', 'url'
'set_secondary_dn_lookup_url', 'url'
'set_access_acct', 'distinguished_name', 'password'
'set_secondary_access_acct', 'distinguished_name', 'password'
'set_failback_interval', time_in_minutes
'suspend', {'primary' | 'secondary'}
'activate', {'primary' | 'secondary'}
'list'
'list_urls'
'list_access_acct'
'check_url', 'url'
'reinit_descriptors'
'check_login', 'name'
'set_timeout', timeout_in_milli_seconds
'set_log_interval', log_interval_in_minutes
'set_num_retries', num_retries
'set_max_ldapua_native_threads', max_ldapua_native_threads
'set_max_ldapua_desc', max_ldapua_desc
'set_abandon_ldapua_when_full', {true|false}
'starttls_on_primary', {true|false}
'starttls_on_secondary', {true|false}
'help'
```

参数 `set_primary_url, 'ldapurl'`

创建指定的搜索字符串 *ldapurl*。只能创建一个主搜索字符串。

ldapurl 的语法为：

```
ldapurl::=ldap://host:port/node?attributes?base | one | sub?filter
```

其中：

- *host* — 是 LDAP 服务器的主机名。
- *port* — 是 LDAP 服务器的端口号。
- *node* — 指定对象层次中搜索开始处的节点。
- *attributes* — 是要返回到结果集中的属性列表。每个 LDAP 服务器都可能支持不同的属性列表。
- *base* — 通过指定基节点的搜索限定搜索条件。

- **one** — 限定搜索条件。 **base** 指定对基节点进行搜索； **one** 指定对某一节点以及节点下的一个次级别进行搜索； **sub** 指定对某一节点以及节点的所有次级别进行搜索。
- **sub** — 指定对某一节点以及节点的所有次级别进行搜索。
- **filter** — 指定要鉴定的一个或多个属性。过滤器可以是简单形式，如 “uid=*”；也可以是复合形式，如 “(uid=*)(ou=group)”。其语法依赖于 LDAP 服务器，并且使用通配符 (*) 来描述登录名。

set_secondary_url, { 'ldapurl' | null }

创建指定的辅助搜索字符串 *ldapurl*，或者不创建辅助搜索字符串。只能创建一个辅助搜索字符串。

set_dn_lookup_url, distinguished_name_url

将 **set_dn_lookup_url** 设置为非 NULL 值时，将使用搜索的区分名算法来向 LDAP 目录服务器鉴定登录名。

distinguished_name_url 的最大长度为 255 个字符，用于搜索与登录名相关联的区分名。

set_secondary_dn_lookup_url, distinguished_name_url

将 **set_secondary_dn_lookup_url** 设置为非 NULL 值时，将创建指定的辅助区分名算法来向 LDAP 目录服务器鉴定登录名。

distinguished_name_url 的最大长度为 255 个字符，用于搜索与登录名相关联的区分名。

set_access_acct, account_distinguished_name, account_password

指定 Adaptive Server 执行搜索和其它只读管理操作所用的标识和口令。标识采用区分名格式。对于 LDAP 服务器，使用 *account_distinguished_name* 鉴定此用户。*account_distinguished_name* 和 *account_password* 都分别限制为 255 个字符。

set_secondary_access_acct, account_distinguished_name, account_password

创建 Adaptive Server 执行搜索和其它只读管理操作所用的辅助标识和口令。标识采用区分名格式。对于 LDAP 服务器，使用 *account_distinguished_name* 鉴定此用户。*account_distinguished_name* 和 *account_password* 都分别限制为 255 个字符。

set_failback_interval

设置 Adaptive Server 管家实用程序检查失败的 LDAP 服务器的间隔。

suspend, {'primary' | 'secondary'}

挂起主 URL 鉴定或辅助 URL 鉴定。

activate, {'primary' | 'secondary'}

启用主 URL 鉴定或辅助 URL 鉴定的设置。

list

显示 LDAP 搜索字符串。

list_urls

显示 LDAP URL 搜索字符串。

list_urls

显示 LDAP URL 搜索字符串。

list_access_acct

显示 LDAP 访问帐户区分名设置。

check_url, 'ldapurl'

检验 LDAP URL 搜索字符串。还可以检验某个用户帐户是否存在，但是它不鉴定用户。

check_login, login_name

检验现有的 LDAP URL 搜索字符串的某个用户帐户。它不鉴定用户。

'set_timeout' timeout_in_milli_seconds

以毫秒为单位设置 Adaptive Server 等待来自 LDAP 服务器的响应直到抛弃鉴定请求前的等待时间。

set_timeout 的缺省值为 10,000 毫秒（10 秒）。有效值介于 1 与 3,600,000（1 小时）之间。

'set_log_interval', log_interval

设置记录间隔，以分钟指定，范围为 0 到 480 分钟。缺省值为 3 分钟。0 指示输出所有消息。

set_num_retries, num_retries

设置出现瞬时错误后的重试次数。**set_num_retries** 的有效范围是 1 - 60，缺省值为 3。

'set_max_ldapua_naptive_threads, max_ldapua_native_threads

设置用于处理 LDAP 鉴定请求的引擎中能够并发运行的最大本机线程数。

set_max_ldapua_native_threads 的最小值为 1。最大值为 **max native threads** 减去使用 **sp_configure** 指定的 **number of dump threads**。缺省值与最大值相同。

sp_configure 确保 **max native threads** 满足 **set_max_ldapua_native_threads** 以及配置参数 **number of dump threads** 的值。

set_max_ldapua_desc, max_ldapua_desc

设置每个引擎的最大 LDAP 描述符数。**set_max_ldapua_desc** 的有效范围是 1 - 20，缺省值为 20。

set_abandon_ldapua_when_full, {true | false}

允许您在超过每个引擎容量的本机线程数时寻求 LDAP 用户鉴定的替代方法。

当没有更多的可用线程时，如果 **set_abandon_ldapua_when_full** 设置为 true，则会抛弃请求。如果 **enable ldap user auth** 设置为 1，则使用 Adaptive Server syslogins 对客户端进行鉴定。如果 **enable ldap user auth** 设置为 2，则客户端登录失败。

如果 **set_abandon_ldapua_when_full** 设置为 false，则会阻止鉴定请求，直到 LDAP 描述符可以接受新的鉴定请求。

help

显示 sp_ldapadmin 的使用信息。

reinit_descriptors

取消绑定所有已建立的 LDAP 服务器描述符，并重新初始化 LDAP 用户鉴定子系统。语法为：

```
sp_ldapadmin 'reinit_descriptors'
```

每当修改认证机构信任的根文件时，系统安全员必须使用 **reinit descriptors** 重新初始化 LDAP 用户鉴定。有关完整的文档，请参见《参考手册：过程》中的 sp_ldapadmin。

set_log_interval, *log_interval*

设置错误消息记录间隔的时间，以分钟为单位。**set_log_interval** 的有效范围是 0 - 480，缺省值为 3。

示例

示例 1 为 LDAP SunONE Directory Server 创建一个 LDAP URL 搜索字符串。

```
sp_ldapadmin set_primary_url,'ldap://voyager:389/  
ou=People,dc=MyCompany,dc=com??sub?uid=*'
```

搜索字符串标识出正在监听主机名 “voyager”，端口号 389（缺省的 LDAP 协议端口）的目录服务器，开始搜索的基准节点位于组织单位 (ou) “People” 中，域为 “MyCompany.com”。它返回与过滤器 uid=* 匹配的所有属性。Adaptive Server 用要进行鉴定的 Adaptive Server 登录名替换通配符。

示例 2 使用示例 1 中描述的条件创建一个在 OpenLDAP 2.0.25 中定义的 LDAP URL 搜索字符串。

```
sp_ldapadmin set_primary_url,'ldap://voyager:389/  
dc=MyCompany,dc=com??sub?cn=*'
```

示例 3 将辅助 LDAP URL 搜索字符串设置为 null，表示没有故障切换和辅助 LDAP 服务器。

```
sp_ldapadmin set_secondary_url, null
```

示例 4 用组合过滤器创建 LDAP URL 搜索字符串。

```
sp_ldapadmin set_primary_url, 'ldap://voyager:389/  
ou=people,dc=siroe,dc=com??sub?(&(uid=*) (ou=accounting))
```

示例 5 使用在 Windows 2000 控制器中找到的缺省 Microsoft Active Directory 方案：

```
1> sp_ldapadmin set_access_acct, 'cn=aseadmin, cn=Users, dc=mycompany,  
dc=com', 'aseadmin secret password'  
2> go  
  
1> sp_ldapadmin set_dn_lookup_url,  
'ldap://mydomainhostname:389/cn=Users,dc=mycompany,dc=com?  
distinguishedName?sub?samaccountname=*'  
2> go  
  
1> sp_ldapadmin set_primary_url, 'ldap://mydomainhostname:389/'  
2> go
```

“aseadmin”用户名已被添加到 Active Directory 服务器，并且已经授予对找到用户的树和对象的读取权限。获得由 distinguishedName 指定的 LDAP 属性，并用于鉴定用户。过滤器指定对属性 samaccountname=* 的搜索；用 Adaptive Server 日志记录中的名字替换 * 通配符。

例如，“samaccountname=jqpublic”向 Adaptive Server 返回值为“cn=John Q. Public, cn=Users, dc=mycompany, dc=com”的 DN 属性“distinguishedName”。Adaptive Server 使用此字符串绑定到 ldap://mydomainhostname:389。如果绑定成功，则鉴定成功。

示例 6 将最大本地线程数设置为 12：

```
sp_ldapadmin 'set_max_ldapua_native_threads', '12'
```

示例 7 将 Adaptive Server 等待来自 LDAP 服务器的响应直到抛弃鉴定请求前的等待时间设置为 25,000 毫秒：

```
sp_ldapadmin, 'set_timeout', '25000'
```

示例 8 禁用鉴定请求，直到 LDAP 描述符可以接受新的鉴定请求：

```
sp_ldapadmin 'set_abandon_ldapua_when_full', 'false'
```

示例 9 显示当前的 LDAP 值：

```
sp_ldapadmin

Primary:
URL:                'ldap://linuxpuneeng1:50917/'
DN Lookup URL:     'ldap://linuxpuneeng1:50917/dc=sybase,dc=com??sub?uid=*'
Access Account:    'cn=Directory Manager'
Active:            'TRUE'
Status:            'READY'
Secondary:
URL:                ''
DN Lookup URL:     ''
Access Account:    ''
Active:            'FALSE'
Status:            'NOT SET'
Timeout value:     '5000' milliseconds
Log interval:      '1' minutes
Number of retries: '3'
Maximum LDAPUA native threads per Engine: '400'
Maximum LDAPUA descriptors per Engine: '3'
Abandon LDAP user authentication when full: 'false'

(return status = 0)
```

用法

- LDAP 供应商确定搜索字符串的语法。在任何情况下，搜索字符串都和“cn=*”中一样以“*attribute=wildcard*”形式指定唯一标识用户的属性名。
- 组合过滤器中的第一个属性必须定义相对区分名 (RDN)。例如，“...sub?(uid=*)(ou=group)”。否则，鉴定失败。
- 添加一个搜索字符串时，Adaptive Server 检验它是否使用有效的 LDAP URL 语法以及它是否引用了现有节点。为确保有效的字符串返回期望的值，在配置 Adaptive Server 时，请仔细选择和检验搜索字符串。
- 辅助 URL 搜索字符串使故障切换到其它 LDAP 服务器成为可能。除非 LDAP Server 处于非活动状态或搜索字符串无效，否则 Adaptive Server 将使用主 URL 搜索字符串。在此事件中，Adaptive Server 使用辅助 URL 搜索字符串进行鉴定。
- 搜索型 DN 算法的登录序列要求 Adaptive Server 使用访问帐号绑定到 LDAP 服务器，然后才能执行搜索。绑定完成后，Adaptive Server 获得 LDAP 描述符（句柄）。此描述符用于在 LDAP 服务器上搜索登录的 DN。

权限

只有系统安全员才可以执行 sp_ldapadmin 命令。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

sp_listener

| | |
|-------------|--|
| 说明 | 按服务器动态启动和停止任何给定端口上的 Adaptive Server 监听器。 |
| 有关进程模式的考虑事项 | sp_listener 按引擎动态启动和停止任何给定端口上的 Adaptive Server 监听器。 |
| 语法 | 对于线程化模式，语法是： <pre>sp_listener "command", "server_name network", remaining</pre> 或者： <pre>sp_listener "command", "[protocol:]machine:port:" CN=common_name"</pre> 对于进程模式，语法是： <pre>sp_listener "command", "server_name network", engine remaining</pre> 或者： <pre>sp_listener "command", "[protocol:]machine:port:"CN=common_name", engine</pre> |
| 参数 | <p>command</p> <p>可以是任意以下值：</p> <ul style="list-style-type: none">• start — 启动各指定服务器上的指定端口上的监听器。• stop — 终止指定的监听器。• suspend — 使监听器不再接受连接。• resume — 指示挂起的监听器恢复监听。• status — 报告参数指定的监听器的状态。该状态为以下状态之一：活动、已停止或已挂起。• help — 显示 sp_listener 语法。 <p>server_name network</p> <p>是 interfaces 文件中指定的 Adaptive Server 名称，或网络的名称。</p> <p>engine</p> <p>（仅在进程模式中使用）指定此命令影响的引擎的编号。engine 可以是引号引起来的单个引擎号（“2”）、列表（“3,5,6”）、范围（“2 - 5”）或所有上述各项的组合（“2,3 - 5,7”）。</p> <hr/> <p>注释 Windows NT 会忽略 engine 参数。</p> <hr/> |

remaining

指定命令将在其可被有意义应用（即，在其中监听器处于命令可以生效的状态）的所有引擎上生效。

protocol

协议的类型；可以是以下类型之一：`tcp`、`tli`、`ssltcp`、`ssltli`、`winsock`、`sslnlwnsck`、`sslwinsock`。

machine:port

监听器连接到的计算机名和端口号（在 `interfaces` 文件中指定）。

CN=common_name

指定 SSL 认证的公用名。

只有当将 `ssltcp` 指定为协议时，才使用 `CN=common_name`。Adaptive Server 将依据 SSL 认证中的 `common_name` 对您指定的 `common_name` 进行验证。如果未包括 `CN=common_name`，Adaptive Server 将使用 `server_name`，依据 SSL 认证中的公用名进行验证。如果在认证中包括完全限定域名，该名称必须与 `CN=common_name` 匹配。

示例

示例 1 为 `interfaces` 文件中与服务器 `orion` 相对应的每一 MASTER 条目启动监听器：

```
sp_listener "start", "orion"
```

示例 2 为端口号 4226 创建 TCP 监听器：

```
sp_listener "start", "goldie:4226"
```

示例 3 为 `interfaces` 文件中服务器 `orion` 的所有 MASTER 条目创建监听器：

```
sp_listener "start", "orion", "remaining"
```

示例 4 为尚未监听此端口的所有引擎启动计算机 `goldie` 的端口 4226 上的 TCP 监听器：

```
sp_listener "start", "goldie:4226", "remaining"
```

示例 5 停止端口号 4226 上的监听器：

```
sp_listener "stop", "tcp:goldie:4226"
```

示例 6 停止端口号 4226 上的所有监听器。因为此命令包括 `remaining` 参数，所以，如果某些引擎不在监听该端口，它将不会失败：

```
sp_listener "stop", "tcp:goldie:4226", "remaining"
```

示例 7 挂起端口 4226 上的 NT Winsock 监听器：

```
sp_listener "suspend", "winsock:clouds:4226"
```

示例 8 恢复端口号 4226 上的所有活动监听器：

```
sp_listener "resume", "tcp:goldie:4226", "remaining"
```

示例 9 指定公用名 asel.big server 1.com：

```
sp_listener 'start','ssl tcp:blad1:17251:
"CN=asel.big server 1.com"', '0'
```

用法

- **sp_listener** 使用如上面语法部分中所述的两个语法中的一个。第一个语法影响在 **interfaces** 文件中列出的所有 **Adaptive Server** 主端口。第二个语法用于管理在 **interfaces** 文件中未列出的监听器。
- 属性名 “CN” 不区分大小写（可为 “CN”、“cn” 或 “Cn”），但公用名的属性值区分大小写。
- 如果您在线程化模式中运行时包括 *engine* 参数，**sp_listener** 将会忽略它。
- **sp_listener** 的语义是最基本语义：如果某一命令无法成功完成，它将被中止。

- 当 **sp_listener** 命令的主机部分为 IPv6 地址时，应使用中括号括起来。例如，

```
tcp:[2001:ec8:4008:1::123]:80
```

- 您可以单独发出 **status** 参数。**status** 参数显示 **interfaces** 文件中所有监听器的状态。
- 监听器可以是以下状态之一：已停止、已挂起或活动。**sp_listener** 允许您在这些状态之间移动监听器。移到非许可状态的请求（例如，请求停止不存在的监听器）将导致失败。使用 **sp_listener** “status” 可以确定监听器的状态。
- **remaining** 参数指定，对于您要运行的命令（**start**、**stop**、**resume** 等），该命令将为处于允许更改（例如，将状态从启动移到停止）的状态中的所有监听器成功运行。例如，如果您尝试启动引擎 1 到 6 上的监听器，但引擎 1、4 和 5 不可用，则 **sp_listener**... “remaining” 将启动引擎 2、3 和 6 上的监听器，而不考虑脱机引擎。如果包括 **remaining** 参数，则无法指定引擎号。
- 监听器的最大数目是 32。如果您在 **interfaces** 文件中创建具有两个主端口的 **Adaptive Server**，则在其它端口上最多可以再启动 30 个监听器。除第一个监听器外，每个补充监听器都占用类似于用户连接的资源，因此在配置了 25 个用户连接的设置中，在启动时启动三个监听器（对应于三个 **MASTER** 条目）将留出供 30 个用户连接使用的空间。

权限

用户必须具有 **sa_role**。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 有关 sp_listener 中与 IPV6 相关的限制，请参见《安全性管理指南》中“数据保密性”一章中的“指定公用名”一节。

sp_listsuspect_db

说明 列出由于恢复时检测到损坏而当前具有脱机页的所有数据库。

语法 sp_listsuspect_db

参数 无。

示例 列出有可疑页的数据库：

```
sp_listsuspect_db
```

用法

- sp_listsuspect_db 列出数据库名、可疑页数以及包含可疑页的对象的数目。
- 使用 sp_listsuspect_page 来标识可疑页。

权限 任何用户都可以执行 sp_listsuspect_db。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [系统过程 sp_listsuspect_page](#)、[sp_setsuspect_granularity](#)、[sp_setsuspect_threshold](#)

sp_listsuspect_object

说明 列出数据库中由于在恢复期间检测到损坏情况而导致当前脱机的所有索引。

语法 sp_listsuspect_object [dbname]

参数 *dbname*
是数据库的名称。

示例 1 列出当前数据库中的可疑索引：

```
sp_listsuspect_object
```

示例 2 列出 pubs2 数据库中的可疑索引：

```
sp_listsuspect_object pubs2
```

用法

- 如果仅数据锁定表上的索引具有可疑页，则整个索引将在恢复期间脱机。查询优化程序将不考虑脱机索引。
- 使用系统过程 [sp_forceonline_object](#) 使脱机索引处于联机状态以用于修复。
- 恢复期间不会使所有页锁定表上的索引完全脱机；只有这些索引的个别页面会脱机。可以用 [sp_forceonline_object](#) 令这些页进入联机状态。
- `sp_listsuspect_object` 为指定数据库中的每一可疑索引列出数据库名、对象 ID、对象名、索引 ID 和访问状态；或者，如果省略 *dbname*，则为当前用户数据库中的每一可疑索引列出上述项。
- `access` 列中的 SA_ONLY 值意味着该索引已被强制联机以供系统管理员使用。BLOCK_ALL 值意味着该索引对于每个人都处于脱机状态。

权限 任何用户都可以执行 `sp_listsuspect_object`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **文档** 有关恢复故障隔离的详细信息，请参见《系统管理指南》。

系统过程 [sp_forceonline_object](#)

sp_listsuspect_page

说明 列出数据库中由于恢复时检测到损坏而当前处于脱机状态的所有页。

语法 sp_listsuspect_page [dbname]

参数 *dbname*
是数据库的名称。

示例 **示例 1** 列出当前数据库中的可疑页：

```
sp_listsuspect_page
```

示例 2 列出 pubs2 数据库中的可疑页：

```
sp_listsuspect_page pubs2
```

用法

- sp_listsuspect_page 为指定数据库中的每一可疑页列出数据库名、页 ID、对象、索引 ID 和访问状态；或者，如果省略 *dbname*，则为当前用户数据库中的每一可疑页列出上述项。
- “access” 列中的 SA_ONLY 值表明该页已被强制联机以仅供系统管理员使用。BLOCK_ALL 值表明该页对于每个人都处于脱机状态。

权限 任何用户都可以执行 sp_listsuspect_page。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_listsuspect_db](#)、[sp_setsuspect_granularity](#)、[sp_setsuspect_threshold](#)

sp_lmconfig

说明 在 Adaptive Server 上配置与许可证管理相关的信息。

语法

```
sp_lmconfig
    [ 'edition' [, edition_type ] ] |
    [ 'license type' [, license_type_name ] ] |
    [ 'smtp host' [, smtp_host_name ] ] |
    [ 'smtp port' [, smtp_port_number ] ] |
    [ 'email sender' [, sender_email_address ] ] |
    [ 'email recipients' [, email_recipients ] ] |
    [ 'email severity' [, email_severity ] ]
    ]
```

参数

sp_lmconfig

不使用任何参数时，可显示以下许可证状态信息：

- 服务器名
- 许可证名称
- 版本
- 数量状态
- 有效日期

edition

是用来指定许可证版本的静态配置参数。

edition_type

指定版本类型，并具有以下可能值：

- null — 缺省值。指定 null 值时，不配置任何产品版本，Adaptive Server 将使用任意版本的许可证启动。
- EE — 指示 Enterprise Edition。
- SE — 指示 Small Business Edition。
- DE — 指示 Developer's Edition。
- XE — 指示 Express Edition。

license type

是用来为 Adaptive Server 的安装指定许可证类型的静态配置参数，该参数仅在指定非 null 版本时才有效。

license_type_name

指定 Adaptive Server 的特定安装的许可证类型。如果您使用的是 Developer's Edition (DE) 或 Express Edition (XE)，则不需要指定 *license type*。最常见的有效值为：

- SRST — 具有网络席位的服务器许可证
- SVST — 具有网络席位的备用服务器许可证
- SRCU — 具有并发用户席位的服务器许可证
- SVCU — 具有并发用户席位的备用服务器许可证
- SRIA — 具有 Internet 访问许可证的服务器许可证
- SVIA — 具有 Internet 访问许可证的备用服务器许可证
- CP — CPU 许可证
- SF — 备用 CPU 许可证
- null — 缺省值

注释 除了此列表外，*sp_lmconfig* 还接受表示专门和遗留许可证类型的两字母缩写形式。如果不接受该许可证类型，请将该类型设置为 *null* 并使用网络许可证服务器选项文件来控制此 Adaptive Server 所使用的许可证。

smtp host, smtp host name

指定用于发送许可证事件通知的电子邮件的 SMTP 主机。

smtp port, smtp port number

指定用于发送许可证事件通知的电子邮件的 SMTP 端口。

email sender, sender email address

指定在许可证事件电子邮件通知中用作发件人地址的电子邮件地址。

email recipients, email recipients

是以逗号分隔的接收许可证事件电子邮件通知的收件人列表。

email severity, email severity

是导致发送电子邮件通知的错误的最低严重性。缺省为错误，其它可能性包括警告和信息性消息。

示例 显示系统的基本许可证配置信息：

```

1> sp_lmconfig
2> go

Parameter Name      Config Value
-----
edition             EE
license type        CP
smtp host            null
email recipients    null
email severity      null
smtp port            null
email sender        null

License Name        Version    Quantity Status      Expiry Date          Server Name
-----
ASE_HA               2010.03314 2          expirable Apr 1 2010 12:00AM  cuprum
ASE_ASM              null        0          not used  null                 null
ASE_EJB              null        0          not used  null                 null
ASE_EFTS             null        0          not used  null                 null
ASE_DIRS             null        0          not used  null                 null
ASE_XRAY             null        0          not used  null                 null
ASE_ENCRYPTION       null        0          not used  null                 null
ASE_CORE             2010.03314 2          expirable Apr 1 2010 12:00AM  cuprum
ASE_PARTITIONS       null        0          not used  null                 null
ASE_RLAC             null        0          not used  null                 null
ASE_MESSAGING_TIBJMS null        0          not used  null                 null
ASE_MESSAGING_IBMQ   null        0          not used  null                 null
ASE_MESSAGING_EASJMS null        0          not used  null                 null

Property Name      Property Value
-----
PE                 EE
LT                 CP
ME                 null
MC                 null
MS                 null
MM                 null
CP                 0
AS                 A

(return status = 0)

```

用法

- 如果不指定版本或使用“null”，Adaptive Server 将查找并使用它在启动时找到的任意许可证版本。

- `sp_lmconfig` 设置的配置选项存储在 `sylapi` 属性文件中。

权限 只有系统管理员才能执行 `sp_lmconfig`。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 适用于您所用平台的 Adaptive Server 安装指南。

sp_lock

说明

报告有关当前持有锁的进程的对象名和 ID。

语法

```
sp_lock [spid1[, spid2]] | [ @verbose = int ]
```

参数

spid1

是来自 `master.dbo.sysprocesses` 表的 Adaptive Server 进程 ID 号。运行 `sp_who` 以获取锁定进程的 *spid*。

spid2

是要用于检查锁的另一个 Adaptive Server 进程 ID 号。

`@verbose = int`

显示表名而非 *spid* 的并置名称，例如 `test.testa`，后跟 *spid*。

注释 *int* 可以是任意数据，因为 `sp_lock` 只检查 `@verbose` 是否为空值。

示例

示例 1 显示串行进程的锁状态，该串行进程的 *spid* 为 7、18 和 23 并具有两个进程系列。*fid* 为 1 的系列具有 *spid* 为 1 的协调进程以及 *spid* 为 8、9 和 10 的工作进程。*fid* 为 11 的系列具有 *spid* 为 11 的协调进程以及 *spid* 为 12、13 和 14 的工作进程：

```
sp_lock
```

```
The class column will display the cursor name for locks associated with a
cursor for the current user and the cursor id for other users.
```

| fid | spid | locktype | table_id | page | dbname | class | context |
|-----|------|------------------|-----------|------|---------|-----------------|---------|
| 0 | 7 | Sh_intent | 480004741 | 0 | master | Non Cursor Lock | NULL |
| 0 | 18 | Ex_intent | 16003088 | 0 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 587 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 590 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 1114 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 1140 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 1283 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 1362 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page | 16003088 | 1398 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Ex_page-blk | 16003088 | 634 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Update_page | 16003088 | 1114 | pubtune | Non Cursor Lock | NULL |
| 0 | 18 | Update_page-blk | 16003088 | 634 | pubtune | Non Cursor Lock | NULL |
| 0 | 23 | Sh_intent | 16003088 | 0 | pubtune | Non Cursor Lock | NULL |
| 0 | 23 | Sh_intent | 176003658 | 0 | pubtune | Non Cursor Lock | NULL |
| 0 | 23 | Ex_intent | 208003772 | 0 | pubtune | Non Cursor Lock | NULL |
| 1 | 1 | Sh_intent | 176003658 | 0 | tpcd | Non Cursor Lock | Sync-pt |
| | | duration request | | | | | |

```

1  1  Sh_intent-blk 208003772    0  tpcd  Non Cursor Lock Sync-pt
duration request
1  8  Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL
1  9  Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL
1 10  Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL
11 11 Sh_intent    176003658    0  tpcd  Non Cursor Lock Sync-pt
duration request
11 12 Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL
11 13 Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL
11 14 Sh_page      176003658 41571 tpcd  Non Cursor Lock NULL

```

示例 2 显示有关 spid 7 当前持有的锁的信息。

```
sp_lock 7
```

The class column will display the cursor name for locks associated with a cursor for the current user and the cursor id for other users.

```

fid spid locktype  table_id  page dbname  class  context
----
0   7   Sh_intent  480004741    0 master  Non Cursor Lock  NULL

```

示例 3 首先在 pubs2 数据库中查询其当前持有锁的正在运行的进程的 ID (1056003762)，然后使用 @verbose 选项查询 pubs2 数据库，这会返回对象名 (master..spt_values) 和进程 ID:

```

1> use pubs2
2> go
1> sp_lock
2> go

```

The class column will display the cursor name for locks associated with a cursor for the current user and the cursor id for other users.

```

fid spid loid locktype  table_id  page row dbname class context
----
0   15  30 Sh_intent 1056003762    0  0 master Non Cursor Lock

```

```

(1 row affected)
(return status = 0)

```

```

1> sp_lock @verbose=0
2> go

```

The class column will display the cursor name for locks associated with a cursor for the current user and the cursor id for other users.

```

fid spid loid locktype page row objectName      id      class context
----
0   15  30 Sh_intent  0  0  master..spt_values 1056003762 Non Cursor
Lock

```

```
(1 row affected)
(return status = 0)
```

用法

- 不含任何参数的 `sp_lock` 报告有关当前持有锁的所有进程的信息。
- 对锁定的唯一用户控制通过在 `select` 语句中使用 `holdlock` 关键字进行。
- 使用 `object_name` 系统函数可以通过表的 ID 号得出表名。
- 在 15.0.3 版本之前的 Cluster Edition 中，`sp_lock` 仅显示与在其上发出该存储过程的实例关联的锁的相关信息。在 Cluster Edition 15.0.3 版和更高版本中，`sp_lock` 显示有关集群中所有锁的信息。
- `sp_lock` 输出首先按 `fid` 排序，然后按 `spid` 排序。
- `loid` 列标识阻塞事务的唯一锁所有者 ID。偶数 `loid` 值表示本地事务拥有该锁。如果 `loid` 值为奇数，则表示外部事务拥有该锁。
- `locktype` 列指示该锁是共享锁（“Sh”前缀）、排它锁（“Ex”前缀）还是更新锁，并且指定该锁是在表上持有（“table”或“intent”）还是在页上持有（“page”）。

“locktype”列中的“blk”后缀表明此进程正在阻塞需要获取锁的另一个进程。在此进程结束后，其它进程将继续执行。“locktype”列中的“demand”后缀表明该进程正尝试获取排它锁。有关锁类型的详细信息，请参见《性能和调优指南》。
- `class` 列表示锁是否与游标关联。它显示下列内容之一：
 - “Non Cursor Lock”表示锁没有与游标关联。
 - “Cursor Id number”指示锁与该 Adaptive Server 进程 ID 的游标 ID 号相关联。
 - 游标名表明该锁与执行 `sp_lock` 的当前用户所拥有的游标 `cursor_name` 相关联。
- `fid` 列标识锁所属的系列（包括协调进程及其工作进程）。`fid` 的值为：
 - 值零表明 `spid` 表示的任务被串行执行。它不参与并行执行。
 - 非零值表示持有锁的任务 (`spid`) 是并行执行语句的进程系列的成员（用 `fid` 标识）。如果值等于 `spid`，则表明该任务是并行执行查询的进程系列中的协调进程。

- context 列标识锁的环境。同一系列中的工作进程具有相同的环境值。“context”的合法值如下所示：

- “NULL”意味着持有该锁的任务或者是串行执行的查询，或者是事务隔离级别 1 中并行执行的查询。
- “Sync-pt duration request”意味着持有该锁的任务在完成查询前将一直持有该锁。

在以下情况下锁的上下文可以是“Sync-pt duration request”：锁是作为并行查询的一部分持有的表锁，或者锁被事务隔离级别 3 上的工作进程持有，或者锁被并行查询中的工作进程持有且在事务的持续时间内必须被持有。

- “Ind pg”指示索引页上的锁（仅适用于所有页锁定表）。
- “Inf key”指示无限键锁（对于仅数据锁定表上事务隔离级别 3 的某些范围查询）
- “Range”指示范围锁（对于仅数据锁定表上事务隔离级别 3 的范围查询）

这些新值可以与“Fam dur”（代替“Sync pt duration”）一起出现以及彼此一起出现（如果适用）。

- row 列显示行级锁的行号。
- sp_lock 输出还显示以下锁类型：
 - “Sh_row”指示共享行锁
 - “Update_row”指示更新行锁
 - “Ex_row”指示排它行锁

权限 任何用户都可以执行 sp_lock。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 命令 [kill](#)、[select](#)

系统过程 [sp_familylock](#)、[sp_who](#)

sp_locklogin

| | |
|----|---|
| 说明 | 锁定 Adaptive Server 帐户，防止用户登录；或显示所有锁定帐户的列表。 |
| 语法 | <pre>sp_locklogin login NULL wildcard_string , "lock" "unlock", [except_login_name except_role_name] [, number_of_inactive_days]</pre> <p>或：</p> <pre>sp_locklogin</pre> |
| 参数 | <p>sp_locklogin 不使用任何参数时，将显示所有锁定的登录名。</p> <p>loginame 是要被锁定或解锁的帐户的名称。</p> <p>wildcard_string 是带有通配符的用于标识一组登录名的任意字符串。</p> <p>NULL 所有登录名（包括 sa_role）都被锁定。</p> <p>lock unlock 指定是锁定还是解锁该帐户。</p> <p>except_login_name 是从锁定登录名中被排除的登录名。</p> <p>except_role_name 是从锁定角色中被排除的角色名。例如，某一角色中被排除的所有登录名。</p> <p>number_of_inactive_days 是帐户处于非活动状态的天数，范围是 1 到 32,767。</p> |
| 示例 | <p>示例 1 锁定用户 “charles” 的登录帐户：</p> <pre>sp_locklogin charles, "lock"</pre> <p>示例 2 锁定除 sa_role 的登录名之外的所有登录名：</p> <pre>sp_locklogin NULL, "lock", sa_role</pre> <p>示例 3 显示所有锁定帐户的列表：</p> <pre>sp_locklogin</pre> |

示例 4 锁定在过去 60 天内未进行鉴定的所有登录帐户：

```
sp_locklogin NULL, 'lock', NULL, 60
```

注释 如果 sp_passwordpolicy 选项 “enable last login updates” 设置为 “0”，则此命令无效。

用法

- 不使用任何参数时， sp_locklogin 将显示所有锁定的登录名。
- 锁定 / 解锁登录名时会更新 syslogins 列 lockdate、locksuid 和 lockreason。
- 使用 sp_locklogin 的条件包括：
 - 不允许任何通配符例外。
 - 不干扰现有功能。
 - 先针对登录名匹配指定的例外。如果不存在此类登录名，则针对角色检查例外。
 - 如果登录名的值为 NULL，则表示 “所有” 登录名。
 - 如果您指定的登录名或例外不存在，将显示一个错误。
 - 如果指定的要锁定的 “有效集合” 为空，则不会发生任何情况。
 - 如果例外为 NULL，（通过 login 参数）指定的登录名集合将被锁定。
 - 仅限 Sybase 高可用性故障切换 — 在 Adaptive Server 15.0 之前的版本中， sp_locklogin 将通过检验远程高可用性服务器中是否存在登录名的 suid（服务器用户 ID）来确定该服务器中是否存在要锁定或解除锁定的登录名。
在 Adaptive Server 15.0 版中， sp_locklogin 既检查 suid 又检查登录名。
- 如果指定除 lock 或 unlock 之外的任何词，则会显示错误。

权限

只有系统安全员才能执行 sp_locklogin。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 create login、alter login

sp_logdevice

说明 将日志和数据在同一设备上的数据库的事务日志移到单独的数据库设备。

语法 `sp_logdevice dbname, devname`

参数 *dbname*

是其 `syslogs` 表（包含事务日志）要放置于特定逻辑设备上的数据库的名称。

devname

是将 `syslogs` 表放置于其上的设备的逻辑名。该设备必须是与某一数据库关联的数据库设备（在 `create database` 或 `alter database` 中指定）。运行 `sp_helpdb` 可以获得有关数据库的设备的报告。

示例 **示例 1** 创建数据库 `products` 并将表 `products.syslogs` 放置于数据库设备 `logs` 上:

```
create database products on default = "10M", logs = "2M"
go
sp_logdevice products, logs
go
```

示例 2 对于日志和数据位于同一设备上的数据库 `test`，将 `test` 的日志放置于日志设备 `logdev` 上:

```
alter database test log on logdev
go
sp_logdevice test, logdev
go
```

用法

- 只有在单用户模式下才能执行 `sp_logdevice`。
- `sp_logdevice` 过程只影响 `syslogs` 的将来的空间分配。这造成一个脆弱时间范围，在此时间范围中您的日志的第一页与您的数据保留在同一设备上。因此，把事务日志放置在单独设备上的首选方法是使用 `create database` 的 `log on` 选项，它会立即将整个事务日志放置在单独设备上。
- 无论是出于恢复原因还是性能原因，都应事务日志放置于单独的数据库设备上。

非常小的不重要数据库可以将其日志与数据库的其余部分一起保存。此类数据库使用 `dump database` 备份数据库和日志并使用 `dump transaction with truncate_only` 截断日志。

- `dbcc checkalloc` 和 `sp_helplog` 显示直到下一个 `dump transaction` 为止仍分配于数据库设备上的 `syslogs` 的某些页。之后，事务日志被完全移交给在执行 `sp_logdevice` 时指定的设备。

- 事务日志所需的设备大小因更新活动量和事务日志转储频率不同而有所不同。通常，为日志设备分配的空间是为数据库本身分配的空间的 10% 到 25%。
- 只将 `sp_logdevice` 用于日志和数据位于同一设备上的数据库。不要将 `sp_logdevice` 用于日志和数据位于不同设备上的数据库。
- 若要增加分配给事务日志的存储量，请使用 `alter database`。如果您使用了 `create database` 的 `log on` 选项将事务日志放置于单独设备上，则使用此方法增加日志段的大小。如果您没有使用 `log on`，则执行 `sp_logdevice`：

```
sp_extendsegment segname, devname
```

您将 `syslogs` 放置于其上的设备或段仅用于 `syslogs`。若要增加为其余数据库分配的存储空间量，请在发出 `alter database` 时指定非日志设备的其它任何设备。

- 使用 `disk init` 命令为数据库或事务日志设置新数据库设备的格式。

权限

只有数据库所有者或系统管理员才能执行 `sp_logdevice`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

文档 《系统管理指南》

命令 `alter database`、`create database`、`dbcc`、`disk init`、`dump database`、`dump transaction`、`select`

系统过程 `sp_extendsegment`、`sp_helpdevice`、`sp_helplog`

sp_loginconfig

说明 (仅限 Windows) 显示一个或所有集成安全性参数的值。

语法 sp_loginconfig ["parameter_name"]

参数 parameter_name

是您要检查的集成安全性参数的名称。其值包括：

- login mode
- default account
- default domain
- set host
- key _
- key \$
- key @
- key #

示例 1 显示所有集成安全性参数的值：

```
sp_loginconfig

name                config_item
-----
login mode          standard
default account     NULL
default domain      NULL
set host            false
key _               domain separator
key $               space
key @               space
key #               -
```

示例 2 显示 login mode 安全性参数的值：

```
sp_loginconfig "login mode"

name                config_item
-----
login mode          standard
```

- 用法
- 集成安全性参数的值存储于 Windows NT 注册表中。有关更改这些参数的说明，请参见《Windows NT 配置指南》中有关登录安全性的章节。
 - `sp_loginconfig` 显示在您启动 Adaptive Server 后生效的 `config_item` 值。如果您在启动 Adaptive Server 后更改了这些注册表值，则这些值并不在 `sp_loginconfig` 输出中反映出来。

权限 只有系统管理员才能执行 `sp_loginconfig`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 sp_revokelogin](#)

sp_logininfo

说明 (仅限 Windows) 显示通过 [sp_grantlogin](#) 授予 Windows NT 用户和组的所有角色。

语法 sp_logininfo ["login_name" | "group_name"]

参数 *login_name*
是 Windows NT 用户的网络登录名。

group_name
是 Windows NT 组名。

示例 **示例 1** 显示授予 Windows NT 用户 “regularjoe” 的权限：

```
sp_logininfo regularjoe

account name      mapped login name  type                privilege
-----
HAZE\regularjoe  HAZE_regularjoe   user                'oper_role'
```

示例 2 显示通过 [sp_grantlogin](#) 授予 Windows NT 用户和组的所有权限：

```
sp_logininfo

account name      mapped login name  type                privilege
-----
BUILTIN\Administrators  BUILTIN\Administrators  group
      'sa_role sso_role oper_role sybase_ts_role navigator_role
      replication_role'
HAZE\regularjoe    HAZE_regularjoe     user                'oper_role'
PCSRE\randy        PCSRE_alexander     user                'default'
```

用法

- **sp_logininfo** 显示通过 [sp_grantlogin](#) 授予 Windows NT 用户和组的所有角色。

- 您在指定 Windows NT 用户名或组名时可以省略域名和域分隔符 (\)。

权限 只有系统管理员才能执行 **sp_logininfo**。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[setuser](#)

系统过程 [sp_displaylogin](#)、[sp_grantlogin](#)、[sp_revokelogin](#)、[sp_role](#)、[sp_who](#)

sp_logiosize

说明 为当前数据库的事务日志执行 I/O 时，将 Adaptive Server 使用的日志 I/O 大小更改为不同的内存池。

语法 sp_logiosize ["default" | "size" | "all"]

参数

default

如果大小为两个逻辑页的内存池可用于高速缓存中，则将当前数据库的日志 I/O 大小设置为 Adaptive Server 的缺省值（两个逻辑页）。否则，Adaptive Server 将该日志 I/O 大小设置为一个逻辑页。因为 **default** 是关键字，所以在指定此参数需使用引号。

size

是用于设置当前数据库的日志 I/O 的大小。其值是逻辑页大小的倍数，最大为该大小的 4 倍。必须用引号将该值引起来。

all

显示为按高速缓存名称分组的所有数据库配置的日志 I/O 大小。

示例

示例 1 显示为当前数据库配置的日志 I/O 大小：

```
sp_logiosize
```

```
The transaction log for database 'master' will use I/O size of 2 Kbytes.
```

示例 2 更改当前数据库的日志 I/O 大小以使用 8K 内存池。如果数据库的事务日志被绑定到不具有 8K 内存池的高速缓存，则 Adaptive Server 返回一个错误消息，指示此类内存池不存在，并且当前日志 I/O 大小不更改：

```
sp_logiosize "8"
```

示例 3 将当前数据库的日志 I/O 大小更改为 Adaptive Server 的缺省值（一个逻辑页大小）。如果对于内存池而言其逻辑页大小在事务日志使用的高速缓存中不存在，则 Adaptive Server 使用 2K 内存池：

```
sp_logiosize "default"
```

示例 4 显示为所有数据库配置的日志 I/O 大小：

```
sp_logiosize "all"
```

```
Cache name: default data cache
Data base                               Log I/O Size
-----
master                                  2 Kb
tempdb                                  2 Kb
model                                    2 Kb
sybsystemprocs                          2 Kb
pubs3                                    2 Kb
```


| | |
|-----------|------|
| pubtune | 2 Kb |
| dbccdb | 2 Kb |
| sybsyntax | 2 Kb |

用法

- **sp_logiosize** 显示或更改当前数据库的日志 I/O 大小。任何用户都可以执行 **sp_logiosize** 以显示配置的日志 I/O 大小。只有系统管理员才可以更改该日志 I/O 大小。
- 如果您指定不含参数的 **sp_logiosize**，则 Adaptive Server 显示当前数据库的日志 I/O 大小。
- 在更改日志 I/O 大小后，该大小将立即生效。Adaptive Server 将在 **sysattributes** 表中记录数据库的新 I/O 大小。
- 为 **sp_logiosize** 指定的任何值必须和一个现有内存池相对应，此内存池为数据库的事务日志使用的高速缓存而配置。使用 **sp_poolconfig** 系统过程指定这些内存池。

如果大小为两个逻辑页的内存池可用于高速缓存中，则 Adaptive Server 将数据库的缺省日志 I/O 大小定义为两个逻辑页。否则，Adaptive Server 将日志 I/O 大小设置为一个逻辑页（一个逻辑页的内存池在任何高速缓存中都始终存在）。对于大多数工作负荷，两个逻辑页的日志 I/O 大小的执行效果要比一个逻辑页的日志 I/O 大小的执行效果好得多，因此事务日志使用的每一高速缓存都应具有该逻辑页大小的内存池。有关配置高速缓存和内存池的详细信息，请参见《系统管理指南》和《性能和调优指南》。

- 如果一个或多个数据库的事务日志被绑定到 **logonly** 类型的高速缓存，则该高速缓存中其 I/O 大小大于为这些数据库定义的日志 I/O 大小的任何内存池都将不被使用。

例如，在一个 2K 服务器上，假定只有两个数据库将其事务日志绑定到包含 2K、4K 和 8K 内存池的“仅日志”高速缓存上。缺省情况下，**sp_logiosize** 按 4K 设置这些参数的日志 I/O 大小，并且不使用 8K 内存池。因此，为避免浪费高速缓存空间，配置日志 I/O 大小时应十分小心。

- 在恢复期间，只有缺省高速缓存的内存池的逻辑页大小是活动的，而与为数据库配置的日志 I/O 大小无关。事务日志被读入缺省高速缓存的此内存池，并且必须被回退或前滚的所有事务都将数据页读入缺省数据高速缓存。

权限

只有系统管理员才能执行 **sp_logiosize** 以更改当前数据库的日志 I/O 大小。任何用户都可以执行 **sp_logiosize** 以显示日志 I/O 大小值。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 系统过程 [sp_cacheconfig](#)、[sp_poolconfig](#)

sp_logintrigger

- 说明** 设置并显示全局登录触发器。全局登录触发器与个人登录脚本具有相同的特点。对于每个试图登录的用户，包括系统管理员和系统安全员，全局登录触发器都会在任何个人登录脚本之前执行。
- 语法** `sp_logintrigger 'global login trigger name'`
- 参数** *global login trigger name*
是全局登录触发器的名称。
- 如果不包含任何参数，`sp_logintrigger` 将显示当前登录触发器的状态和名称（如果有）；如果没有定义任何全局登录触发器，该命令不会生成任何行。
- 示例**
- 示例 1** 使用 `sp_logintrigger` 设置全局登录触发器：
- ```
sp_logintrigger 'master.dbo.myproc'
```
- 示例 2** 返回更新的全局登录触发器：
- ```
1> sp_logintrigger
2> go
Global login trigger          Status
-----
sybssystemprocs.dbo.myproc    Enabled
(1 row affected)
(return status = 0)
```
- 示例 3** 当全局登录触发器不存在时：
- ```
1> sp_logintrigger
2> go
Global login trigger Status

(0 rows affected)
```
- 示例 4** 删除先前使用 `sp_logintrigger` 指定的全局登录触发器：
- ```
sp_logintrigger 'drop'
```
- 用法**
- 可以使用全局变量 `@@logintrigger` 来确定是否定义并启用了全局登录触发器。
 - 全局登录触发器与个人登录脚本有一个不同之处。全局登录触发器按名称存储在 `sysattributes` 中，而个人登录脚本只是按对象 ID 存储。
- 权限** 任何用户都可以执行 `sp_logintrigger` 来显示当前全局登录触发器。若要设置新的登录触发器，需要 `sso_role`。

sp_maplogin

| | |
|----|--|
| 说明 | 将外部用户映射到 Adaptive Server 登录名。 |
| 语法 | <code>sp_maplogin (authentication_mech null), (client_username null), (action login_name null)</code> |
| 参数 | <p>authentication_mech 是为使用 <code>sp_modifylogin</code> 中的选项进行鉴定而指定的有效值之一。</p> <p>client_username 是外部用户名。此用户名可以是操作系统名、LDAP 服务器的用户名，也可以是任何其它 PAM 库能够识别的名称。空值表示任何登录名都有效。</p> <p>action 表示 <code>create login</code> 或 <code>drop</code>。使用 <code>create login</code> 时，将在鉴定登录名后立即创建登录。<code>drop</code> 用于删除登录名。</p> <p>login_name 是 <code>syslogins</code> 中已经存在的 Adaptive Server 登录名</p> |
| 示例 | <p>示例 1 将外部用户 “jsmith” 映射到 Adaptive Server 用户 “guest”。一旦通过鉴定，“jsmith” 就会获得 “guest” 的权限。审计登录记录同时显示 <code>client_username</code> 和 Adaptive Server 用户名：</p> <pre>sp_maplogin NULL, "jsmith", "guest"</pre> <p>示例 2 指示 Adaptive Server 在登录不存在的情况下为所有使用 PAM 进行鉴定的外部用户新建登录：</p> <pre>sp_maplogin PAM, NULL, "create login"</pre> |
| 用法 | |
| 权限 | 只有系统管理员才能执行 <code>sp_maplogin</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

sp_merge_dup_inline_default

- 说明** 删除现有重复行内缺省对象，将唯一行内缺省对象转换为可共享行内缺省对象。
- 语法** `sp_merge_dup_inline_default [report_only = {yes | no} [, show_progress = {yes | no}]]`
- 参数**
- report_only**
如果您指定 **yes**，则报告当前数据库中的唯一行内缺省值数量，但不执行任何更改。如果指定 **no**，则：
- `sp_merge_dup_inline_default` 删除重复的行内缺省值，所有唯一行内缺省值都更改为可共享行内缺省值
 - 现有的引用重复行内缺省值的列定义会更新以引用可共享行内缺省值
- `report_only` 的缺省值为 **yes**。
- show_progress**
如果设置为 **yes**，则当 `report_only` 设置为 **no** 时，`sp_merge_dup_inline_default` 会显示井号以显示进度。
- `show_progress` 的缺省值为 **no**。
- 示例** **示例 1** 在不带任何选项的情况下针对 `pubs2` 数据库运行 `sp_merge_dup_inline_default`。`sp_merge_dup_inline_default` 不进行任何更改，但会显示信息性消息以指示适当的唯一行内缺省值数：

```
sp_merge_dup_inline_default
=====
sp_merge_dup_inline_default is used to identify duplicate inline default objects,
subsequently to convert one of them into sharable inline default object and remove the
rest. As the result, it will remove entries from sysobjects, syscomments and
sysprocedures. It will also update entries in syscolumns, syscomments and
sysprocedures.

Following is the current state of your inline default objects found out by
sp_merge_dup_inline_default and what it could potentially do to them. By default,
sp_merge_dup_inline_default only reports the current state and this warning message.
If
you really intend to carry out the changes, please rerun this stored procedure using

sp_merge_dup_inline_default @report_only = "NO"

Database pubs2 has about 0 unique inline defaults If you convert them into sharable
inline
defaults, the rest of total 0 duplicate defaults can be removed from the system
catalogs.
=====
```

示例 2 将唯一行内缺省值转换为可共享行内缺省值:

```
sp_merge_dup_inline_default @report_only = 'NO'
```

Total 2 duplicate defaults are removed and 7 defaults are converted to sharable inline defaults. Database is modified and in single-user mode. System Administrator (SA) must reset it to multi-user mode with sp_dboption.

示例 3 如果没有重复的行内缺省值，则生成以下输出:

```
sp_merge_dup_inline_default @report_only = 'NO'
```

Database is not modified. Please try it later if duplicate inline defaults do exist and the current resource limitation is preventing this conversion process.

示例 4 包括 show_progress 参数以指示进度:

```
sp_merge_dup_inline_default @report_only = 'NO', @show_progress = "YES"
```

```
Calculating...
```

```
Converting...
```

```
[# ]
[##### ]
[##### ]
[##### ]
[##### ]
[##### ]
[##### ]
[##### ]
[##### ]
[##### ]
```

Total 2 duplicate defaults are removed and 7 defaults are converted to sharable inline defaults. Database is modified and in single-user mode. System Administrator (SA) must reset it to multi-user mode with sp_dboption

用法

- 不能在系统数据库上运行 sp_merge_dup_inline_default。
- 用户数据库必须处于单用户模式，然后才能运行 sp_merge_dup_inline_default。
- 如果系统过程中止，您可以重新运行 sp_merge_dup_inline_default。
- 如果 sp_merge_dup_inline_default 发出错误消息以指明 Adaptive Server 耗尽锁，则：
 - 增大 number of locks 的值，或者
 - 通过 sp_setpglockpromote 或 sp_setrowlockpromote 降低锁升级阈值。

重新运行 `sp_merge_dup_inline_default`，并在 `sp_merge_dup_inline_default` 结束后重置值。

- `sp_merge_dup_inline_default` 仅更改其缺省值是文字字符串常量或简单数字的行内缺省值对象（文字字符串常量不能包括转义字符串分隔符）。
- 对于任何其在 `syscomments` 中的源文本是“`encrypted`”的重复行内缺省值对象，`sp_merge_dup_inline_default` 都不将其删除。

sp_metrics

| | |
|----|---|
| 说明 | 备份、删除和刷新 QP 指标（总是在作为各自数据库中的第 1 组的缺省运行组中捕获）及其查询的统计信息。 |
| 语法 | <pre>sp_metrics ['backup' backup_group_ID 'drop', 'gid' [, 'id'] 'flush' 'help', 'command']</pre> |
| 参数 | <p>backup 将保存的 QP 指标从缺省运行组移动到备份组，将 QP 指标从旧服务器备份到备份组，并将保存的 QP 指标从缺省运行组移动到备份组。</p> <p>backup_group_ID 是将 QP 指标从旧服务器移动到的备份组的组 ID。将保存的 QP 指标从缺省运行组移动到备份组。</p> <p>drop 从系统目录中删除 QP 指标。如果不提供 'id'，sp_metrics 将删除您用 'gid' 指定的整个组。</p> <p>gid 是 QP 指标在系统目录中的组 ID。</p> <p>id 是 QP 指标在系统目录中的 ID。</p> <p>flush 向系统目录刷新内存中的所有集合指标。内存中所有语句的集合指标都将被清除。</p> <p>'help', 'command' 提供有关 sp_metrics 命令的使用信息。</p> |
| 示例 | <p>示例 1 将 QP 指标从缺省组移动到备份组。</p> <pre>sp_metrics 'backup', '3'</pre> <p>示例 2 提供有关 sp_metrics flush 的信息：</p> <pre>sp_metrics 'help', 'flush'</pre> |
| 用法 | <p>通过对 sysquerymetrics 视图使用 select 语句和 order by 来访问指标信息。</p> <p>用来将 QP 指标从旧服务器备份到备份组。将保存的 QP 指标从缺省运行组移动到备份组，从系统目录中删除 QP 指标。向系统目录刷新内存中的所有集合指标。</p> |
| 权限 | 系统管理员 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [select](#)、[set](#)

过程 [sp_configure](#)

sp_modify_resource_limit

说明 通过指定新的限制值和 / 或在超出该限制时要执行的操作，来更改资源限制。

语法 sp_modify_resource_limit {*name*, *appname*}
rangename, *limittype*, *limitvalue*, *enforced*, *action*, *scope*

参数

name

是对其应用限制的 Adaptive Server 登录名。必须指定 *name* 或 *appname*，或指定两者。若要修改应用于特定应用程序的所有用户的限制，可指定一个值为 null 的 *name*。

appname

是对其应用限制的应用程序名称。必须指定 *name* 或 *appname*，或指定两者。如果该限制应用于 *name* 所使用的所有应用程序，则指定 *appname* 的值为 null。如果该限制控制特定应用程序，则指定客户端程序在登录包中传递给 Adaptive Server 的应用程序的名称。

rangename

是强制执行限制的时间范围。您不能修改此值，但必须指定一个非空值来唯一地标识资源限制。

limittype

是对其应用限制的资源的类型。您不能修改此值，但必须指定一个非空值来唯一地标识资源限制。该值必须是下列值之一：

| 限制类型 | 说明 |
|--------------|----------------------------|
| row_count | 限制一个查询可以返回的行数 |
| elapsed_time | 限制查询批处理或事务可以运行的秒数（按挂钟时间） |
| io_cost | 限制处理查询的实际开销或优化程序的开销估计 |
| tempdb_space | 限制单个会话可以具有的 tempdb 数据库中的页数 |

limit_value

是在 Adaptive Server 强制执行该限制前登录名或应用程序可以使用的服务器资源的最大量。该最大量必须是小于或等于 2^{31} 的正整数，或者为 null 以保持现有值。下表列出了应为每种限制类型指定的值：

| 限制类型 | 限制值 |
|--------------|---------------------------------|
| row_count | 在强制执行限制前一个查询可以返回的最大行数 |
| elapsed_time | 强制执行限制前查询批处理或事务可以运行的最大秒数（按挂钟时间） |
| io_cost | 从优化程序的开销计算公式导出的无单位度量 |
| tempdb_space | 限制单个会话可以具有的临时数据库中的页数。 |

enforced

确定是在执行查询前还是执行查询时强制应用限制。不能修改此值。将 `null` 用作占位符。

action

是在超出限制时采取的操作。以下代码应用于所有限制类型：

| 操作代码 | 说明 |
|-------------------|---------|
| 1 | 发出警告 |
| 2 | 中止查询批处理 |
| 3 | 中止事务 |
| 4 | 注销会话 |
| <code>null</code> | 保持现有值 |

scope

是限制的作用域。不能修改此值。可以使用 `null` 作为占位符。

示例

示例 1 修改 `weekends` 时间范围内应用于由 “`robin`” 使用的所有应用程序的资源限制。该限制在预计查询返回超过 3000 行时发出一个警告：

```
sp_modify_resource_limit robin, NULL, weekends, row_count, 3000, NULL,
1, NULL
```

示例 2 修改每周 7 天、每天 24 小时应用于 `acctg` 应用程序的资源限制。该限制在估计查询处理时间超过 45 秒时中止查询批处理：

```
sp_modify_resource_limit NULL, acctg, "at all times", elapsed_time,
45, 2, 2, 6
```

示例 3 此例更改资源限制的值，该资源限制在 `tu_wed_7_10` 时间范围内限制 `payroll` 应用程序的所有用户的经历时间。经历时间的限制值从 120 秒减少到 90 秒。执行时间、采取的措施和作用域的值都保持不变：

```
sp_modify_resource_limit NULL, payroll, tu_wed_7_10,
elapsed_time, 90, null, null, 2
```

示例 4 此例更改资源限制采取的措施，该资源限制在 `saturday_night` 时间范围内限制 “`joe_user`” 运行的所有即席查询和应用程序的行计数。以前代表采取措施的值是 3，即查询超过指定的行计数时将中止事务。新值为 2，即中止查询批处理。限制类型、执行时间和作用域的值都保持不变。

```
sp_modify_resource_limit joe_user, NULL,
saturday_night, row_count, NULL, NULL, 2, NULL
```

- 用法
- 不能更改对其应用了限制的登录名或应用程序，也不能指定新的时间范围、限制类型、强制时间或作用域。
 - 资源限制的修改将导致在该会话的下一查询批处理开始时，该登录名和 / 或应用程序的每一会话的限制被重新绑定。
 - Adaptive Server Enterprise 提供资源限制功能，帮助系统管理员防止查询和事务独占服务器资源。但是，只有将资源限制与时间范围绑定，才能完全指定资源限制条件。

权限 只有系统管理员才能执行 sp_modify_resource_limit。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **文档** 有关详细信息，请参见《系统管理指南》。

系统过程 [sp_add_resource_limit](#)、[sp_drop_resource_limit](#)、[sp_help_resource_limit](#)

sp_modify_time_range

| | |
|----|---|
| 说明 | 更改与指定时间范围关联的开始日期、开始时间、结束日期和 / 或结束时间。 |
| 语法 | <code>sp_modify_time_range name, startday, endday, starttime, endtime</code> |
| 参数 | <p>name 是时间范围的名称。该名称必须是 <code>master</code> 数据库的 <code>sysmimeranges</code> 系统表中存储的时间范围的名称。</p> <p>startday 指定时间范围从星期几开始。它必须是缺省服务器语言的完整星期名称（缺省服务器语言存储在 <code>master</code> 数据库的 <code>syslanguages</code> 系统表中），或者为 <code>NULL</code> 以保持现有的 <code>startday</code>。</p> <p>endday 指定时间范围到星期几结束。它必须是缺省服务器语言的完整星期名称（缺省服务器语言存储在 <code>master</code> 数据库的 <code>syslanguages</code> 系统表中），或者为 <code>NULL</code> 以保持现有的结束日期。<code>endday</code> 在星期中可以早于或晚于 <code>startday</code>，或者它可以和 <code>startday</code> 是同一天。</p> <p>starttime 是时间范围开始的时间。以 24 小时制指定 <code>starttime</code>，值介于 00:00 和 23:59 之间。请使用以下格式，或使用 <code>NULL</code> 保留现有的 <code>starttime</code>： <code>"HH:MM"</code></p> <p>endtime 是时间范围结束的时间。以 24 小时制指定 <code>endtime</code>，值介于 00:00（午夜）和 23:59 之间。请使用以下格式，或使用 <code>NULL</code> 保留现有的 <code>endtime</code>： <code>"HH:MM"</code></p> <p><code>endtime</code> 在一天中的发生时间必须晚于 <code>starttime</code>，除非 <code>endtime</code> 是 00:00。</p> |

注释 对于跨整天的时间范围，将开始时间指定为“00:00”，将结束时间指定为“23:59”。

示例 **示例 1** 将“`business_hours`”时间范围的结束日期从星期五更改为星期六。并且保持现有开始日期、开始时间和结束时间不变：

```
sp_modify_time_range business_hours, NULL, Saturday, NULL, NULL
```

示例 2 为“`before_hours`”时间范围指定新的结束日期和结束时间：

```
sp_modify_time_range before_hours, Monday, Saturday, NULL, "08:00"
```

- 用法
- 不能修改“所有时间”时间范围。
 - 可以修改某一时间范围，以使该时间范围与一个或多个其它时间范围重叠。
 - 通过系统存储过程修改时间范围并不影响当前正在进行中的会话的活动时间范围。
 - 对作用域为事务的资源限制进行更改不会影响任何当前正在进行的事务。

权限 只有系统管理员才能执行 `sp_modify_time_range`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关详细信息，请参见《系统管理指南》。

系统过程 [sp_add_resource_limit](#)、[sp_add_time_range](#)、[sp_drop_time_range](#)

sp_modifylogin

| | |
|----|---|
| 说明 | Adaptive Server 15.7 和更高版本不支持此系统过程。若要修改 Adapter Server 的登录帐户，请使用 <code>alter login</code> 命令。 |
| 语法 | 无 |
| 参数 | 无 |
| 用法 | 无 |

sp_modifystats

说明 允许系统管理员或拥有权限的任何用户执行该过程和更新目标表的统计信息，以修改 `sysstatistics` 中的列的密度值。

语法

```
sp_modifystats [database].[owner].table_name,
               {"column_group" | "all"},
               MODIFY_DENSITY,
               {range | total},
               {absolute | factor},
               "value"
```

或者：

```
sp_modifystats [database].[owner].table_name, column_name,
               REMOVE_SKEW_FROM_DENSITY
```

参数

table_name

要更改的表的名称。如果该表位于另一数据库中，请指定数据库名；如果数据库中有多个具有该名称的表，请指定所有者的名称。 *owner* 的缺省值是当前用户，而 *database* 的缺省值是当前数据库。

column_group

列名称的有序列表。若要更改多个列的统计信息（例如密度值），请按用于创建统计信息的顺序列出这些列。各列名用逗号分隔。例如，如果您的表具有关于列 *a1*、*a2*、*a3*、*a4* 的密度统计信息：

- “*a1*” 修改列 *a1*。
- “*a1,a2,a3*” 修改列组 *a1,a2,a3*。
- 也可以将通配符 % 用于 *column_group* 参数，以代表一定范围的字符。例如，“*a1,%a3*” 修改组 *a1,a2,a3* 和 *a1,a4,a3*，依此类推；“*a1,%*” 修改组 *a1,a2* 和 *a1,a2,a3*，依此类推，但不修改 *a1*；“*a1%*” 修改组 *a1,a2* 和 *a1,a2,a3*，依此类推，并且修改 *a1*。

all

修改此表的所有列组。因为 “*all*” 是一个关键字，所以需要给它加引号。

MODIFY_DENSITY

允许将单列或列组的范围或总密度修改为 *value* 参数中指定的粒度。范围单元密度表示直方图中范围单元代表的所有值的平均重复数。有关详细信息，请参见 *性能和调优指南*。

range

修改范围单元密度。

total

修改总单元密度。

absolute

忽略当前值，使用 *value* 参数指定的数值。

factor

用 *value* 参数乘以当前统计值。

value

既可以是指定的密度值，也可以是当前密度的倍数。如果指定了 **absolute**，则该值必须在 0 到 1 之间（包括 0 和 1）。

column_name

是表中列的名称。

REMOVE_SKEW_FROM_DENSITY

允许系统管理员更改单列的总密度，使其等于范围密度，这在出现数据不对称时很有用。总密度表示频率单元和范围单元中所有值的重复平均数。总密度用于估计连接和搜索参数（其值在查询优化时为未知）的匹配行数。有关详细信息，请参见 *性能和调优指南*。

REMOVE_SKEW_FROM_DENSITY 也更新任何组合列统计信息的总密度，该列是组合列的最主要属性。通常，该列是其最主要属性的组合索引会产生这些组合列的统计信息，但当发出一个组合 **update statistics** 命令时，也可以产生统计信息。

示例

示例 1 将表 **tab_1** 中列组 **c00**、**c01** 的范围密度更改为 0.50000000:

```
sp_modifystats "tab_1", "c00, c01", MODIFY_DENSITY, range, absolute, "0.5"
```

示例 2 将 **tab_1** 中列组 **c00**、**c01** 的总密度乘以 0.5，也就是将其除以 2:

```
sp_modifystats "tab_1", "c00,c01", MODIFY_DENSITY, total, factor, "0.5"
```

示例 3 将表 **tab_1** 中所有列的总密度乘以 0.5。

```
sp_modifystats "tab_1", "all", MODIFY_DENSITY, total, factor, "0.5"
```

示例 4 将始于 **c12** 的所有列组的总密度更改为和范围密度相等。

```
sp_modifystats "tab_1", "c12" REMOVE_SKEW_FROM_DENSITY
```

用法

- 允许系统管理员修改 **sysstatistics** 中单列或多列的密度值。
- 使用 **optdiag** 查看表的统计信息。有关表密度和使用 **optdiag** 的详细信息，请参见《性能和调优指南》。
- 用 **sp_modifystats** 对统计信息所做的任何修改在运行 **update statistics** 后会被覆盖。为了确保使用最新的有关统计信息的修改，应该在运行 **update statistics** 后运行 **sp_modifystats**。
- 因为 **sp_modifystats** 修改存储在 **sysstatistics** 表中的信息，所以在生产系统中执行 **sp_modifystats** 前，应该备份统计信息。

权限 没有任何人可以缺省使用 `sp_modifystats`。具有 `sso_role` 角色的人士必须指定对 `sp_modifystats` 的权限。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

使用的表 `sysstatistics`

另请参见 **命令** `update statistics`

sp_modifythreshold

| | |
|----|---|
| 说明 | 通过使阈值与其它阈值过程、可用空间级别或段名关联，对该阈值进行修改。 |
| 语法 | <code>sp_modifythreshold dbname, segname, free_space [, new_proc_name][, new_free_space][, new_segname]</code> |
| 参数 | <p><i>dbname</i> 是要更改其阈值的数据库。它必须是当前数据库的名称。</p> <p><i>segname</i> 是监控其可用空间的段。指定“default”段时要使用引号。</p> <p><i>free_space</i> 是达到阈值时的可用页数。当段中的可用空间低于此水平时，Adaptive Server 执行关联的存储过程。</p> <p><i>new_proc_name</i> 是超出阈值时要执行的新存储过程。该过程可以位于当前 Adaptive Server 或某个 Open Server 上的任何数据库中。阈值不能在远程 Adaptive Server 上执行过程。</p> <p><i>new_free_space</i> 是要与阈值相关联的新可用页数。当段中的可用空间低于此水平时，Adaptive Server 执行关联的存储过程。</p> <p><i>new_segname</i> 是为其监控可用空间的新段。指定“default”段时要使用引号。</p> |
| 示例 | <p>示例 1 将 mydb 数据库的“default”段的阈值修改为在该段上的可用空间低于 175 页时执行，而不是低于 200 页时执行。NULL 是一个占位符，指示不更改过程名：</p> <pre>sp_modifythreshold mydb, "default", 200, NULL, 175</pre> <p>示例 2 修改 mydb 的 data_seg 段的阈值，以使它执行 new_proc 过程：</p> <pre>sp_modifythreshold mydb, data_seg, 250, new_proc</pre> |
| 用法 | <p>不能使用 sp_modifythreshold 更改最后机会阈值的可用空间量或段名。</p> <p>达到阈值</p> <ul style="list-style-type: none"> • 达到阈值时，Adaptive Server 执行关联的存储过程。Adaptive Server 对阈值过程使用以下搜索路径： <ul style="list-style-type: none"> • 如果过程名称不指定数据库，Adaptive Server 将在达到阈值的数据库中查找。 |

- 如果在该数据库中未找到该过程，并且过程名称以“sp_”开头， Adaptive Server 将在 sybserverprocs 数据库中查找。

如果在两个数据库中都没有找到该过程，则 Adaptive Server 向错误日志发送一条错误消息。

- Adaptive Server 使用一个**停滞值**（即全局变量 @@thresh_hysteresis）来确定阈值对可用空间变化的敏感程度。一旦阈值执行其过程，阈值将失效。在段中的可用空间量超过阈值 @@thresh_hysteresis 页之前，阈值一直处于不活动状态。这样可防止因响应可用空间的微小波动而使阈值重复执行其过程。

最后机会阈值

- 缺省情况下， Adaptive Server 监控日志所在的段上的可用空间并在可用空间量少于允许成功转储事务日志所必需的空间时执行 [sp_thresholdaction](#)。此可用空间量（**最后机会阈值**）由 Adaptive Server 计算并且用户无法更改。
- 如果在记录某个事务前达到了最后机会阈值， Adaptive Server 将挂起该事务直到日志空间被释放。使用 [sp_dboption](#) 为特定数据库更改此行为。通过将 abort tran on log full 选项设置为 true，可使 Adaptive Server 回退在达到最后机会阈值时尚未记录的所有事务。
- 无法使用 [sp_modifythreshold](#) 更改与最后机会阈值关联的可用空间值或段名。
- 只有将其日志存储在单独的段上的数据库才可具有最后机会阈值。使用 [sp_logdevice](#) 可以将事务日志移动到单独的设备上。

其它阈值

- 每个数据库最多可以有 256 个阈值，其中包括最后机会阈值。
- 每个阈值距下一个最接近的阈值必须至少为 2 乘以 @@thresh_hysteresis 页数。
- 使用 [sp_helpthreshold](#) 可以获得有关现有阈值的信息。
- 使用 [sp_droptreshold](#) 可以从段中删除阈值。

创建阈值过程

- 具有 create procedure 权限的任何用户都可以在数据库中创建阈值过程。通常，系统管理员在 master 数据库中创建 [sp_thresholdaction](#)，数据库所有者在用户数据库中创建阈值过程。
- [sp_modifythreshold](#) 不验证指定的过程是否存在。可以将某一阈值与尚不存在的过程相关联。

- `sp_modifythreshold` 进行检查以确保修改阈值过程的用户已被直接授予“sa_role”。在修改阈值过程时，所有活动的系统角色都在 `systhresholds` 中作为写入该过程的用户的有效角色输入。但是，触发阈值时，将只激活直接授予的系统角色。不激活间接授予的系统角色和用户定义的角色。
- Adaptive Server 向阈值过程传递四个参数：
 - `@dbname, varchar(30)`，它标识数据库
 - `@segment_name, varchar(30)`，它标识段
 - `@space_left, int`，它指示与阈值关联的可用页的数目
 - `@status, int`，对于最后机会阈值，此参数的值为 1；对于其它阈值，此参数的值为 0

这些参数按位置传递，而不是按名称传递；您的阈值过程可以为它们使用其它名称，但该过程必须按所示顺序以及用正确的数据类型声明它们。

- 没有必要为每个阈值创建一个不同的过程。为将维护开销降至最低，应在可由所有阈值执行的 `sybssystemprocs` 数据库中创建单个阈值过程。
- 在阈值过程中包括 `print` 和 `raiserror` 语句可将输出发送到错误日志。

执行阈值过程

- 在达到阈值时启动的任务作为后台任务执行。这些任务没有关联的终端会话或用户会话。如果您在这些任务正运行时执行 `sp_who`，则 `status` 列显示“background”。
- Adaptive Server 用修改了阈值的用户的以下权限执行阈值过程：该用户执行 `sp_modifythreshold` 时的权限减去执行该命令后已撤消的所有权限。
- 每个阈值过程都使用一个用户连接，这样它会一直占用该连接以执行过程。

禁用可用空间计数

警告！ 如果禁用可用空间计数，系统过程将无法提供有关空间分配的准确信息。

- 使用 `sp_dboption` 的 `no free space acctg` 选项可以禁用对非日志段的可用空间计数。
- 不能禁用对日志段的可用空间计数。

权限 只有数据库所有者或系统管理员才能执行 `sp_modifythreshold`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **文档** 有关详细信息，请参见《系统管理指南》。

命令 [create procedure](#)、[dump transaction](#)

系统过程 [sp_addthreshold](#)、[sp_dboption](#)、[sp_droptreshold](#)、[sp_helpthreshold](#)、[sp_thresholdaction](#)

sp_monitor

说明

显示有关 Adaptive Server 的统计信息。

语法

`sp_monitor` 语法按命令区分，因为许多类型都有其自己的参数。以下代码段先显示整个存储过程的语法，再显示每个命令类型接口的语法。

```
sp_monitor [[connection | statement], [cpu | diskio | elapsed time]]
           [event, [spid ]]
           [procedure, [dbname, [procname[, summary | detail]]]]
           [enable] [disable]
           [help],
           [deadlock][[procstack]
```

参数

connection

显示有关每个连接的信息。 `connection` 使用以下监控表：

- `monProcessSQLText`
- `monProcessActivity`

statement

显示有关每个语句的信息。 `statement` 使用以下监控表：

- `monProcessSQLText`
- `monProcessStatement`

cpu | diskio | elapsed time

这些参数用于对 `sp_monitor connection` 或 `sp_monitor statement` 的输出排序。

- `cpu` — 指示各个连接或语句消耗的 CPU 时间。
- `diskio` — 指示每个连接或语句执行的物理读取数。
- `elapsed time` — 指示每个连接或语句的 CPU 时间和等待时间的总和。

event

有三种可能的显示内容。如果指定：

- 零个选项 — 仅显示用户任务。
- `sp_monitor, event, "-1"` — 显示有关所有任务（包括用户任务和系统任务）的等待信息。
- `sp_monitor, event, "spid"` — 仅显示有关指定的服务器进程 ID 的等待信息。

spid

允许获取有关特定任务的 event 信息，方法是输入它的 *spid*。必须在引号内指定 *spid* 的数值。

procedure

显示有关存储过程的统计信息：

- ProcName — 正在被监控的存储过程。
- DBNAME — 存储过程所在的数据库。
- NumExecs — 此特定存储过程大约执行的次数。
- AvgCPUTime — 执行存储过程所需的平均 CPU 时间。
- AvgPhysicalReads — 存储过程执行的平均磁盘读取数。
- AvgLogicalReads — 存储过程执行的平均逻辑读取数。
- AvgMemUsed_KB — 存储过程使用的平均内存量（以 KB 为单位）。

procedure 使用 monSysStatement 监控表。

dbname

显示有关指定数据库的过程的信息。

procname

显示有关指定过程的信息。

summary | detail

显示摘要信息（它提供过程的所有实例的平均值），或者显示详细信息（它提供有关存储过程的每个实例的信息）。

enable

启用 sp_monitor 的新选项。它打开开始监控所需的配置参数。

disable

禁用监控。

help

显示 sp_monitor 的语法和示例，同时报告将此过程用于死锁分析的详细信息：

```
sp_monitor 'help', 'deadlock'
```

help 选项还可提供特定于命令的示例。

deadlock

指示 sp_monitor 处理 monDeadlock 表中的历史数据，并输出每个死锁实例的输出块。

procstack

检查任务的执行上下文，包括深层嵌套存储过程的执行上下文。所执行的过程堆栈从 `monProcessProcedures` 监控表中提取。

示例**示例 1** 报告有关 Adaptive Server 繁忙情况的信息：

```
sp_monitor

last_run                current_run            seconds
-----                -
Jan 29 1987 10:11AM    Jan 29 1987 10:17AM  314

cpu_busy                io_busy              idle
-----                -
4250 (215) -68%        67 (1) -0%          109 (100) -31%

packets_received        packets_sent          packet_errors
-----                -
781 (15)               10110 (9596)        0 (0)

total_read              total_write total_errors    connections
-----                -
394 (67)               5392 (53)           0 (0)            15 (1)
```

示例 2 说明如何显示有关连接的信息：

```
1> sp_monitor "connection"
2> go

spid      LoginName      ElapsedTime      LocksHeld      SQLText
-----
12        sa              90300            2              exec get_employee_salaries
27        sa              17700            1              exec get_employee_perks
```

缺省情况下，缺省输出按 `ElapsedTime` 的降序顺序排序。

示例 3 标识出执行物理读取最多的连接：

```
1> sp_monitor "connection","diskio"
2> go

spid      LoginName      Physical_Reads      LocksHeld      SQLText
-----
12        sa              117                 2              exec get_employee_salaries
27        sa              1                   0              exec get_employee_perks
```

示例 4 显示有关每条语句的信息：

```

1> sp_monitor "statement"
2> go
spid    LoginName    ElapsedTime    SQLText
-----
12      sa           100            exec get_employee_salaries

```

示例 5 显示每个任务花费时间等待的事件以及等待的持续时间，按等待时间的降序顺序报告：

```

1> sp_monitor "event"
2> go
SPID    WaitTime    Description
-----
6       108200     hk: pause for some time
29      108200     waiting for incoming network data
10      107800     waiting while allocating new client socket
15      17100      waiting for network send to complete
14      5900       waiting for CTLIB event to complete
14      400        waiting for disk write to complete
7       200        hk: pause for some time
7       100        waiting on run queue after yield
12      100        waiting for network send to complete

```

示例 6 显示 spid 14 的事件数据：

```

1> sp_monitor "event", "14"
2> go
WaitTime    Description
-----
9000 waiting for CTLIB event to complete
600 waiting for disk write to complete
200 waiting for disk write to complete
100 waiting on run queue after yield
100 wait for buffer write to complete

```

示例 7 提供最近运行的过程的摘要，按平均经历时间的降序顺序排序。本示例提供历史监控信息，而不是当前状态。

```

1> sp_monitor "procedure"
2> go

Average Procedure Statistics
=====

ProcName          DBName          AvgElapsedTime    AvgCPUTime    AvgWaitTime

```

| | AvgPhysicalReads | AvgLogicalReads | AvgPacketsSent | NumExecs | | | | |
|-----------------|------------------|-----------------|----------------|----------|----|-----|---|----|
| neworder_remote | tpcc | 1833 | 16 | 1083 | 26 | 96 | 0 | 6 |
| neworder_local | tpcc | 1394 | 13 | 1181 | 31 | 122 | 0 | 38 |
| tc_startup | tpcc | 1220 | 3 | 1157 | 0 | 3 | 0 | 59 |
| delivery | tpcc | 1000 | 0 | 800 | 23 | 49 | 0 | 2 |

用法

注释 在使用与 `sp_monitor` 关联的新参数之前，您必须设置监控表和启用所需的相关存储过程。请参见《性能和调优：监控和分析》中的“安装监控表”。

- `Adaptive Server` 跟踪它对一系列全局变量产生的影响。`sp_monitor` 显示这些全局变量的当前值以及自最后执行过程后的更改量。
- 对于每一列，这些统计信息以 `number(number)-number%` 或 `number(number)` 形式出现。
 - 第一个数值表示秒数（对于 `cpu_busy`、`io_busy` 和 `idle`）或自 `Adaptive Server` 重新启动后的总数（对于其它列）。
 - 小括号中的数值表示自最后一次运行 `sp_monitor` 后的秒数或总数。百分比符号指示自最后一次运行 `sp_monitor` 后的时间百分比。

例如，如果报告将 `cpu_busy` 显示为“4250(215)-68%”，则它意味着 CPU 自 `Adaptive Server` 最后一次启动以来处于忙状态的时间为 4250 秒，自 `sp_monitor` 最后一次运行以来处于忙状态的时间 215 秒，并且占 `sp_monitor` 自最后一次运行以来总时间的 68%。

对于 `total_read` 列，值 394(67) 意味着自 `Adaptive Server` 最后一次启动以来已有 394 个磁盘读取，其中 67 个读取是自最后一次运行 `sp_monitor` 后进行的。

- 此表显示每种选项类型访问的监控表。

表 1-24: 按监控类型访问的监控表

| 监控类型 | 访问的表 | 配置选项 | 配置选项类型 |
|------------|----------------------|------------------------------|--------|
| connection | monProcessSQLExt | max SQL text monitored | 值 |
| | | SQL batch capture | 布尔型 |
| | monProcessActivity | wait event timing | 布尔型 |
| | | per object statistics active | 布尔型 |
| procstack | monProcessProcedures | 无 | 不适用 |
| statement | monProcessSQLText | max SQL text monitored | 值 |
| | | SQL batch capture | 布尔型 |
| | monProcessStatement | statement statistics active | 布尔型 |
| | | per object statistics active | 布尔型 |
| event | monProcessWaits | wait event timing | 布尔型 |
| | | process event waits | 布尔型 |
| procedure | monSysStatement | statement statistics active | 布尔型 |
| | | per object statistics active | 布尔型 |
| | | statement pipe max messages | 值 |
| | | statement pipe active | 布尔型 |
| deadlock | monDeadlock | deadlock pipe max messages | 值 |
| | | deadlock pipe active | 布尔型 |

- `sp_monitor connection` 仅监控主动执行的 T-SQL 的连接，而不报告所有连接。
- 必须从 `master` 数据库中运行 `sp_monitor`。但是，如果要分析其它数据库中存档的死锁数据，可从该数据库中运行 `sp_monitor deadlock`。
- 在不使用任何选项调用 `sp_monitor event` 时，将不再显示所有任务（包括系统任务）。在 Adaptive Server version 15.0.2 及更高版本中，`event` 选项提供了三种可能性。如果：
 - 不提供任何选项 — 仅显示用户任务。
 - 指定 `sp_monitor, event, "-1"` — 显示有关所有任务（包括用户任务和系统任务）的等待信息。
 - 指定 `sp_monitor, event, "spid"` — 仅显示有关指定的服务器进程 ID 的等待信息。

- 表 1-25 描述 `sp_monitor` 报告中的列、等效的全局变量（如果有）及其含义。除 `last_run`、`current_run` 和 `seconds` 以外，这些列标题也是全局变量的名称，但以 `@@` 开头的的所有全局变量除外。由全局变量报告的数字单位上也有差别——全局变量报告的数字不是以毫秒计的 CPU 时间，而是计算机时钟周期。

表 1-25: `sp_monitor` 报告中的列

| 列标题 | 等效变量 | 含义 |
|-------------------------------|------------------------------|---|
| <code>last_run</code> | | <code>sp_monitor</code> 过程最后一次运行的时钟时间。 |
| <code>current_run</code> | | 当前时钟时间。 |
| <code>seconds</code> | | 自 <code>sp_monitor</code> 最后一次运行以来的秒数。 |
| <code>cpu_busy</code> | <code>@@cpu_busy</code> | Adaptive Server 的 CPU 执行 Adaptive Server 工作的秒数（CPU 时间）。 |
| <code>io_busy</code> | <code>@@io_busy</code> | Adaptive Server 执行输入和输出操作所用的秒数（CPU 时间）。 |
| <code>idle</code> | <code>@@idle</code> | Adaptive Server 处于空闲状态的秒数（CPU 时间）。 |
| <code>packets_received</code> | <code>@@pack_received</code> | Adaptive Server 读取的输入包的数目。 |
| <code>packets_sent</code> | <code>@@pack_sent</code> | Adaptive Server 写入的输出包的数目。 |
| <code>packet_errors</code> | <code>@@packet_errors</code> | Adaptive Server 读写包时检测到的错误数。 |
| <code>total_read</code> | <code>@@total_read</code> | Adaptive Server 的磁盘读取操作的数目。 |
| <code>total_write</code> | <code>@@total_write</code> | Adaptive Server 的磁盘写入操作的数目。 |
| <code>total_errors</code> | <code>@@total_errors</code> | 读写数据时 Adaptive Server 检测到的错误数。 |
| <code>connections</code> | <code>@@connections</code> | Adaptive Server 的登录或登录尝试次数。 |

- 在 Adaptive Server 启动后首次 `sp_monitor` 运行时，小括号中的数值没有意义。
- Adaptive Server 的管家任务使用服务器的空闲周期将从高速缓存更改的页写入磁盘。此进程影响 `sp_monitor` 报告的 `cpu_busy`、`io_busy` 和 `idle` 列的值。若要禁用该管家任务并消除这些影响，请将 `housekeeper free write percent` 配置参数设置为 0：

```
sp_configure "housekeeper free write percent", 0
```

- 当具有代表性的工作量在系统上运行时，必须运行 `sp_monitor`。
- 通常，将按以下顺序运行这些过程：
 - 运行 `sp_monitor enable`
 - 调用 `sp_monitor` 选项
 - 完成监控后，运行 `sp_monitor disable`

- 使用 `sp_monitor procedure` 时，返回的行数可能非常大；可能要使用 `summary` 选项，而不是 `detail` 选项。在活动的系统上，此命令可能要花一段时间才能完成。

权限 必须具有 `mon_role` 权限才能执行 `sp_monitor`。有关详细信息，请参见《性能和调优：监控和分析》中的“监控表”。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • <i>角色</i> — 当前活动角色 • <i>关键字或选项</i> — NULL • <i>以前值</i> — NULL • <i>当前值</i> — NULL • <i>其它信息</i> — 所有输入参数 • <i>代理信息</i> — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_who](#)

sp_monitorconfig

说明 显示与索引、对象、数据库和内核资源内存池的元数据描述符有关的高速缓存使用情况统计信息。sp_monitorconfig 还报告用于参照完整性查询的辅助扫描描述符的有关统计信息以及事务描述符和 DTX 参与者的使用情况统计信息。

语法 sp_monitorconfig "configname"[, "result_tbl_name"][, "full"]

参数 configname

或者是 all，或者是要查询其监控信息的配置参数名的一部分。在“用法”部分中列出了有效的配置参数。指定 all 将显示服务器中所有索引、对象、数据库和辅助扫描描述符的描述符帮助信息。

“result_tbl_name”

是为保存存储过程结果而创建的表的名称。这是一个可选参数。如果为 result_tbl_name 传递了尚不存在的表名，sp_monitorconfig 将创建一个表来保存结果集。

“full”

为您指定的 configname 返回一组值。其值为：

- config_val — 报告配置值
- system_val — 报告未配置任何值时的系统缺省值
- total_val — 报告实际使用的值

示例 示例 1 显示已打开的所有项：

```
sp_monitorconfig "open"
Configuration option is not unique.
option_name                config_value run_value
-----
number of open databases           12           12
number of open objects             500          500
currread change w/ open cursors    1            1
open index hash spinlock ratio     100          100
number of open indexes             500          500
open index spinlock ratio          100          100
open object spinlock ratio         100          100
number of open partitions          500          500
```

示例 2 显示所有配置的状态:

```
sp_monitorconfig "all"
```

```
-----
```

```
Usage information at date and time: May 6 2010 4:32PM.
```

| Name | Num_free | Num_active | Pct_act | Max_Used | Reuse_cnt | Instance_Name |
|---------------------------|----------|------------|---------|----------|-----------|---------------|
| additional network memory | 1358436 | 809440 | 37.34 | 825056 | 0 | NULL |
| audit queue size | 100 | 0 | 0.00 | 0 | 0 | NULL |
| disk i/o structures | 256 | 0 | 0.00 | 29 | 0 | NULL |
| heap memory per user | 4096 | 0 | 0.00 | 0 | 0 | NULL |
| . . . | | | | | | |
| size of process object he | 3000 | 0 | 0.00 | 0 | 0 | NULL |
| size of shared class heap | 6144 | 0 | 0.00 | 0 | 0 | NULL |
| size of unilib cache | 306216 | 816 | 0.27 | 816 | 0 | NULL |
| txn to pss ratio | 400 | 0 | 0.00 | 0 | 0 | NULL |

示例 3 显示 61 个活动对象元数据描述符，其中 439 个是可用的。
自 Adaptive Server 最后一次启动后在高峰期使用的最大值是 61:

```
sp_monitorconfig "open objects"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
```

| Name | Num_free | Num_active | Pct_act | Max_Used | Reuse_cnt | Instance_Name |
|------------------------|----------|------------|---------|----------|-----------|---------------|
| number of open objects | 439 | 61 | 12.20 | 61 | 0 | NULL |

例如，您然后将该大小重新设置为 550，以容纳使用的最大元数据描述符的数目 439，外加 10% 的额外空间:

```
sp_configure "number of open objects", 330
```

示例 4 显示索引元数据描述符的最大数目，该数目为 44:

```
sp_monitorconfig "open indexes"
```

```
Usage information at date and time: Apr 22 2002 2:49PM.
```

| Name | Num_free | Num_active | Pct_act | Max_Used | Reuse_cnt | Instance_Name |
|------------------------|----------|------------|---------|----------|-----------|---------------|
| number of open indexes | 556 | 44 | 7.33 | 44 | 0 | NULL |

可以将该大小重新设置为 100，即最小可接受值:

```
sp_configure "number of open indexes", 100
```


示例 5 将活动扫描描述符的数目显示为 30，尽管 Adaptive Server 被配置为使用 200 个这样的描述符。使用 `number of aux scan descriptors` 配置参数将该值重新设置为至少 32。较为保险的设置是 36，可以容纳 32 个扫描描述符，外加 10% 的额外空间：

```
sp_monitorconfig "aux scan descriptors"

Usage information at date and time: Apr 22 2002  2:49PM.

Name                Num_free Num_active Pct_act Max_Used Reuse_cnt Instance_Name
-----
number of aux scan descri  170      30    15.00          32         0          NULL
```

示例 6 Adaptive Server 被配置用于 5 个打开的数据库，所有这些数据库都已用于当前会话。

```
sp_monitorconfig "number of open databases"

Name                Num_free Num_active Pct_act Max_Used Reuse_cnt Instance_Name
-----
number of open databasess  0         5    100.00          5  Yes          NULL
```

但是，如 `Reuse_cnt` 列所示，需要打开其它数据库。如果所有 5 个数据库都在使用中，将会导致错误，除非可以重新使用未在使用中的数据库描述符。为防止错误发生，应将 `number of open databases` 重新设置为更高的值。

示例 7 只有 10.2% 的事务描述符当前正在使用中。但是，自 Adaptive Server 最后一次启动后在高峰期使用的事务描述符的最大数目是 523：

```
sp_monitorconfig "txn to pss ratio"

Usage information at date and time: Apr 22 2002  2:49PM.

Name                Num_free Num_active Pct_act Max_Used Reuse_cnt Instance_Name
-----
txn to pss ratio      784       80    10.20          523         0          NULL
```

示例 8 使用可选参数 `result_tbl_name` 可以创建用户表，将 `sp_monitorconfig` 结果保存到此表中：

```
create table sample_table
(Name varchar(35),Config_val int, System_val int, Total_val int,
Num_free int, Num_active int, Pct_act char(6), Max_used int,
Num_Reuse int, Date varchar(30))

create table sample_table
(Name varchar(35),
Config_val int,
System_val int,
```

```
Total_val int,
Num_free int,
Num_active int,
Pct_act char(6),
Max_Used int,
Reuse_cnt int,
Date varchar(30),
Instance_Name varchar(35))
```

创建的表名称成为 `sp_monitorconfig` 的第二个参数。捕获 `sample_table` 中的 `number of locks` 和 `number of alarms` 值:

```
sp_monitorconfig "locks", sample_table
sp_monitorconfig "number of alarms", sample_table
```

显示在 `sample_table` 中捕获的值:

```
select * from sample_table
  Name          Config_val System_val Total_val Num_free Num_active
Pct_act Max_used Reuse_cnt   Date          Instance_Name
-----
number of locks      5000      684      5000      4915      85
1.70      117      0      Aug 23 2006  6:53AM
number of alarms     40      0      40      28      12
30.00      13      0      Aug 23 2006  6:53AM
```

保存到该表的结果集被累积，直到您删除或截断该表。

注释 如果 `sample_table` 在其它数据库中，则必须提供它的用引号引起来的全限定名。

示例 9 显示所有配置的 `configure_value`、`system_value` 和 `run_value` 列:

```
sp_monitorconfig "all", null, "full"
go

Usage information at date and time: Mar 23 2004  5:15PM
Name          Configure Value System Value Run Value
Num_free  Num_active Pct_act  Max_Used Reuse_cnt Instance_Name
-----
additional network memory      0      2167876      2167876
 1358436      809440      37.34      825056      0      NULL
audit queue size      100      0      100
 100      0      0.00      0      0      NULL
disk i/o structures      256      0      256
 256      0      0.00      29      0      NULL
```

| | | | | | | | |
|---------------------------|-----|-------|---------|--|--------|--------|------|
| heap memory per user | | | 4096 | | 563 | 4096 | |
| 4096 | 0 | 0.00 | 0 | | 0 | | NULL |
| kernel resource memory | | | 4096 | | 0 | 4096 | |
| 3567 | 529 | 12.92 | 529 | | 0 | | NULL |
| max cis remote connection | | | 0 | | 100 | 100 | |
| 100 | 0 | 0.00 | 0 | | 0 | | NULL |
| . . . | | | | | | | |
| size of shared class heap | | | 6144 | | 0 | 6144 | |
| 6144 | 0 | 0.00 | 0 | | 0 | | NULL |
| size of unilib cache | | | 0307032 | | 307032 | 306216 | |
| 816 | 0 | .27 | 816 | | 0 | | NULL |
| txn to pss ratio | | | 16 | | 0 | 16 | |
| 400 | 0 | 0.00 | 0 | | 0 | | NULL |

用法

- 如果 `max cis remote connections` 配置参数具有 `config_value`，则 `system_val` 报告零 (0) 值。
- 如果使用比原始给定值小的值来重新配置资源，资源不会缩减，并且 `Num_active` 配置参数可以报告比 `Total_val` 大的数字。当 `Adaptive Server` 重新启动时，资源将缩减，并且数字正确报告。
- `sp_monitorconfig` 显示与索引、对象和数据库的元数据描述符有关的高速缓存使用情况统计信息，例如当前由服务器使用的元数据描述符的数目。
- `sp_monitorconfig` 还报告使用中的辅助扫描描述符的数目。在对表执行查询时扫描描述符管理表的单次扫描。
- `sp_monitorconfig` 监控以下资源：

| | |
|--------------------------------|------------------------------|
| additional network memory | number of open indexes |
| audit queue size | number of open objects |
| heap memory per user | number of open partitions |
| max cis remote connection | number of remote connections |
| max memory | number of remote logins |
| max number network listeners | number of remote sites |
| memory per worker process | number of sort buffers |
| max online engines | number of user connections |
| number of alarms | number of worker processes |
| number of aux scan descriptors | partition groups |
| number of devices | permission cache entries |
| number of dtx participants | procedure cache size |
| number of java sockets | size of global fixed heap |
| number of large i/o buffers | size of process object heap |
| number of locks | size of shared class heap |
| number of mailboxes | size of unilib cache |
| number of messages | txn to pss ratio |
| number of open databases | |

- `sp_monitorconfig` 输出中的列提供以下信息：
 - `num_free` — 指定当前未使用的可用元数据或辅助扫描描述符的数目。
 - `num_active` — 指定在高速缓存中安装的（即活动的）元数据或辅助扫描描述符的数目。
 - `pct_active` — 指定存入高速缓存的或活动的元数据或辅助扫描描述符的百分比。
 - `Max_Used` — 指定在启动服务器后已在使用中的元数据或辅助扫描描述符的最大数目。
 - `Reused` — 指定是否重新使用元数据描述符以容纳服务器中增加的索引、对象或数据库。返回值为 `Yes`、`No` 或 `NA`（用于不支持重用机制的配置参数，例如 `aux scan descriptors` 的数目）。
- 以 `Max_Used` 列中的值为基础，确定描述符的适当数目；确保再增加约 10% 来作为最终设置。例如，如果使用的索引元数据描述符的最大数目是 142，则可以将 `number of open indexes` 配置参数设置为 157。
- 如果 `Reused` 列指定为 `Yes`，则将该配置参数重新设置为更高的值。在描述符需被重新使用的情况下，可能会有性能问题，在有打开的数据库时尤需注意。打开的数据库中包含大量的元数据信息，这意味着若要填满一个打开的数据库，`Adaptive Server` 需要多次访问磁盘上的元数据；服务器可能还具有螺旋锁争用问题。若要检查螺旋锁争用情况，请使用系统过程 `sp_sysmon`。请参见《性能和调优系列：使用 `sp_sysmon` 监控 `Adaptive Server`》。若要确定索引、对象或数据库的当前数目，请使用 `sp_countmetadata`。
- 若要获取精确的数据，请在正常的 `Adaptive Server` 高峰期运行 `sp_monitorconfig`。您可以在高峰期运行 `sp_monitorconfig` 几次，以确保实际查找的就是使用的描述符的最大数目。
- `result_tbl_name` 使用以下语法创建表。所有结果信息都保存在该表中，不返回任何标准输出。


```
create table table_name(
    Name varchar(35), Num_free int,
    Num_active int, Pct_act char(6),
    Max_Used int, Reuse_cnt int,
    Date varchar(30))
```
- 某些配置参数（例如 `number of sort buffers` 和 `txn to pss ratio`）依赖于配置的用户连接的数目，而其它配置参数（如 `max number of network listeners`）则依赖于引擎数。

- `sp_monitorconfig` 的输出使用用户连接和联机引擎的数目来计算列 `num_free`、`num_active`、`pct_act` 和 `max_used` 的值。
- 出于性能原因，对内部监控计数器的更新不使用同步方法进行。因此，负荷较重的多引擎 Adaptive Server 在 `sp_monitorconfig` 输出中报告的数目可能不完全精确。
- 您可能会看到，空闲系统中的活动锁数大于 0。这些“活动”锁是保留的，供内部使用。

权限 只有系统管理员才能执行 `sp_monitorconfig`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 [系统过程 `sp_configure`](#)、[系统过程 `sp_countmetadata`](#)、[系统过程 `sp_helpconfig`](#)、[系统过程 `sp_helpconstraint`](#)、[系统过程 `sp_sysmon`](#)

sp_monitor_server

说明 提供服务器范围的监控信息。

语法 sp_monitor_server [server_name]

参数 无

示例 显示当前服务器监控信息：

```
sp_monitor_server
last_run          current_run          seconds
-----
May 10 2010  4:23PM          May 10 2010  4:23PM          1

(1 row affected)
cpu_busy          io_busy          idle
-----
0(0)-0%          0(0)-0%          21(0)-0%
packets_received  packets_sent      packet_errors
-----
0(0)              0(0)              0(0)
total_read        total_write        total_errors        connections
-----
1743(0)          146(0)          0(0)          1(0)
```

用法 无

权限 无

审计 无

sp_object_stats

- 说明** 显示表和索引的锁争用、锁等待时间和死锁统计信息。
- 语法** `sp_object_stats interval[, top_n[, dbname, objname[, rpt_option]]]`
- 参数**
- interval*
指定采样的时间范围。必须采用 HH:MM:SS 格式，例如 “00:20:00”。
- top_n*
是要报告的对象的数量，按争用顺序。缺省值为 10。
- dbname*
是要报告的数据库的名称。如果没有给出数据库名，则报告所有数据库中对象的争用情况。
- objname*
是要报告的表的名称。如果指定了表名，则也必须指定数据库名。
- rpt_option*
必须或者是 `rpt_locks`，或者是 `rpt_objlist`。

示例 **示例 1** 报告有关服务器范围内前 10 个对象的锁统计信息：

```
sp_object_stats "00:20:00"
```

示例 2 只报告 `pubtune` 数据库中的表，并且列出争用程度最高的 5 个表：

```
sp_object_stats "00:20:00", 5, pubtune
```

示例 3 只显示具有最高锁定活动的表的名称，即使争用和死锁情况并未发生：

```
sp_object_stats "00:15:00", @rpt_option = "rpt_objlist"
```

用法

- `sp_object_stats` 报告指定采样期间在表上获取的共享锁、更新锁和排它锁的有关信息。以下报告显示 `titles` 表：

```
Object Name: pubtune..titles (dbid=7,  
objid=208003772,lockscheme=Datapages)
```

| Page Locks | SH_PAGE | UP_PAGE | EX_PAGE\$ |
|-------------|-------------|-------------|------------|
| ----- | ----- | ----- | ----- |
| Grants: | 94488 | 4052 | 4828 |
| Waits: | 532 | 500 | 776 |
| Deadlocks: | 4 | 0 | 24 |
| Wait-time: | 20603764 ms | 14265708 ms | 2831556 ms |
| Contention: | 0.56% | 10.98% | 13.79% |

*** Consider altering pubtune..titles to Datarows locking.

- 这些值的含义为：

| 输出行 | 值 |
|------------|-----------------------|
| Grants | 立即授予锁的次数。 |
| Waits | 需要锁的任务要等待的次数。 |
| Deadlocks | 发生死锁的数目。 |
| Wait-times | 所有任务在等待锁时所用的总毫秒数。 |
| Contention | 任务必须等待或遇到死锁的次数所占的百分比。 |

- sp_object_stats 在表中的总争用大于 15% 时建议更改锁定方案，具体如下：
 - 如果表采用所有页锁定，建议改为数据页锁定。
 - 如果表使用数据页锁定，建议改为数据行锁定。
- rpt_option 指定报告类型：
 - rpt_locks 报告争用程度最高的表的授予数、等待数、死锁数和等待时间。rpt_locks 为缺省设置。
 - rpt_objlist 只报告具有最高级别锁活动的对象名。
- sp_object_stats 创建名为 tempdb..syslkstats 的表。该表在存储过程完成后不被删除，因此系统管理员可以使用 Transact-SQL 对它进行查询。
- 一次只应有一个用户执行 sp_object_stats。如果多个用户尝试同时运行 sp_object_stats，则第二个命令可能会被阻塞，否则结果可能无效。
- 每次执行 sp_object_stats 后，tempdb..syslkstats 表都会被删除并重新创建。
- tempdb..syslkstats 的结构为：

| 列名 | 数据类型 | 说明 |
|------------|----------|--------------------------------------|
| dbid | smallint | 数据库 ID |
| objid | int | 对象 ID |
| lockscheme | smallint | 整数值 1 - 3: 所有页 = 1, 数据页 = 2, 数据行 = 3 |
| page_type | smallint | 数据页 = 0, 或者索引页 = 1 |
| stat_name | char(30) | 由此行表示的统计信息 |
| stat_value | float | 授予、等待或死锁的数目，或者总等待时间 |

stat_name 列中的值由以下三部分组成：

- 第一部分是“ex”、“sh”或“up”，分别表示排它锁、共享锁或更新锁。
- 第二部分是“pg”或“row”，分别表示页锁或行锁。
- 第三部分是“grants”、“waits”、“deadlocks”或“waittime”，分别表示立即授予的锁、需要等待释放其它锁的锁、死锁或为获取锁而等待的时间。
- 如果您指定一个表名，则 sp_object_stats 显示该名称的所有表。如果多个用户拥有具有指定名称的表，则这些表的输出将显示对象 ID，而不是所有者名。

权限 只有系统管理员才能执行 sp_object_stats。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色—当前活动角色 • 关键字或选项—NULL • 以前值—NULL • 当前值—NULL • 其它信息—所有输入参数 • 代理信息—set proxy 有效时的初始登录名 |

另请参见 **命令** [alter table](#)

sp_opt_querystats

| | |
|----|--|
| 说明 | 返回选定查询的性能分析。 |
| 语法 | <code>sp_opt_querystats "query_text" help [, "diagnostic_options" null [, database_name] [, user_name]]</code> |
| 参数 | <p>"query_text" 是要分析的查询的文本，用引号引起来。</p> <p>help 显示 <code>sp_opt_querystats</code> 的语法和用法信息。</p> <p>diagnostic_options (可选) 基于 <code>set</code> 选项的诊断参数。请参见“用法”。</p> <p>null <code>sp_opt_querystats</code> 需要用三个参数指定数据库的名称。如果不需要诊断选项，请为该参数输入值 <code>null</code> 以指定 <code>database_name</code> 参数的值。</p> <p>database_name (可选) 要在其中执行查询的数据库的名称。如果您要分析的查询没有全限定表，则使用此参数。</p> <p>user_name (可选) 在 <code>database_name</code> 参数指定的数据库中执行查询的用户的名称。该用户必须已经存在于数据库中，而且执行 <code>sp_opt_querystats</code> 的登录名必须具有在该数据库中执行 <code>setuser</code> 命令的权限。</p> |
| 示例 | <p>示例 1 在 <code>pubs2</code> 数据库上分析 <code>select</code> 命令：</p> <pre>sp_opt_querystats 'select * from pubs2.dbo.authors'</pre> <p>示例 2 在 <code>pubs2</code> 数据库上分析 <code>select</code> 命令，并根据是否启用以下 <code>set</code> 命令来包括信息：<code>set showplan</code>、<code>set statistics io</code>、<code>set option show</code>、<code>set statistics plancost on</code>：</p> <pre>sp_opt_querystats 'select * from pubs2.dbo.authors', 'showplan,statio,option_show, plancost'</pre> |
| 用法 | <ul style="list-style-type: none"> • 必须包括 <code>exec</code> 命令，<code>sp_opt_querystats</code> 才能执行查询。 • 若要以其它用户的身份运行 <code>sp_opt_querystats</code>，请在 <code>exec immediate</code> 命令中或输出查询上下文中包括 <code>setuser</code> 命令。 • 必须包括 <code>showdata</code> 命令，<code>sp_query_stats</code> 才能返回结果集。 • 发出 <code>set quoted_identifier on</code> 后，可以将 <code>sp_opt_querystats</code> 选项用引号引起来。例如： <pre>sp_opt_querystats 'select "col" from "MYTABLE"', 'all','DB'</pre> |

- *diagnostic_option* 为以下值之一：

| diagnostic_option | set 选项 | 注释 |
|--------------------------|------------------------------------|--|
| statio | set statistics io on | |
| stattime | set statistics time on | |
| showplan | set showplan on | |
| missingstats | set option show_missing_stats long | |
| resource | set statistics resource on | |
| plancost | set statistics plancost on | |
| switches | show switches | |
| option_show_long | set option show long | option_show_long 和 option_show 是互斥的。 |
| option_show | set option show on | |
| showdata | set nodata on | 如果您包括 showdata, 则不会执行 set nodata on。 |
| exec | set noexec on | 如果您包括 exec, 则不会执行 set noexec on。 |
| allrows_mix | set plan optgoal allrows_mix | allrows_mix、allrows_oltp 和 allrows_dss 是互斥的。 |
| allrows_oltp | set plan optgoal allrows_oltp | |
| allrows_dss | set plan optgoal allrows_dss | |
| diagmode | 返回增强的进度信息。 | |
| all | 启用前八个选项 | all 和 allexec 无法与其它参数组合, 而且是互斥的。 allexec 选项包括 all 选项。 |
| allexec | 启用前七个选项 | |

- 如果您包括多个选项, 或者如果您指定关键字 **all**, 则必须用引号将选项列表引起来。
- 在没有任何选项的情况下运行 `sp_opt_querystats` 与在带有 **all** 选项的情况下运行是相同的。

sp_options

| | |
|----|--|
| 说明 | 显示选项值。 |
| 语法 | <pre>sp_options [[show help [, option_name category_name null [, dflt non_dflt null [, spid]]]]]</pre> |
| 参数 | <p>show 列出所有选项的当前值和缺省值，并按这些值的类别进行分组。发出带指定选项名的 <code>sp_options</code> 显示，将为您提供有关单个选项的当前值和缺省值。您还可以指定会话 ID，并指定是要查看具有缺省设置的选项，还是要查看具有非缺省设置的选项。</p> <p>help 指示您希望显示用法信息。发出不带参数的 <code>sp_options</code> 时会得到相同的结果。</p> <p>option_name 是选项的名称。</p> <p>category_name 是选项的类别。</p> <p>null 指示要查看其设置的选项。</p> <p>dflt non_dflt null 指示是显示具有缺省设置的选项，还是显示具有非缺省设置的选项。</p> <p>spid 指定会话 ID。使用会话 ID 查看其它会话设置。</p> |

示例 **示例 1** 查看 `sp_options` 的用法：

```
1> sp_options
2> go

Usage:
sp_options [ [show | help
             [, <option_name>|<category_name>|null
             [, dflt | non_dflt | null
             [, <spid>] ] ] ] ]
```

示例 2 查看所有当前选项和缺省选项的列表：

```
1> sp_options show
2> go
Category: Query Tuning
name                currentsetting      defaultsetting      scope
-----
```

| | | | | |
|-----------------------------|-------|-------------|-------------|---|
| optgoal | | allrows_mix | allrows_mix | 0 |
| opttimeoutlimit | 40 | | 10 | 0 |
| merge_join | 1 | | 1 | 4 |
| hash_join | 0 | | 0 | 4 |
| nl_join | 1 | | 1 | 4 |
| distinct_sorted | 1 | | 1 | 4 |
| distinct_sorting | 1 | | 1 | 4 |
| distinct_hashing | 1 | | 1 | 4 |
| group_sorted | 1 | | 1 | 4 |
| group_hashing | 1 | | 1 | 4 |
| group_inserting | 0 | | 0 | 4 |
| order_sorting | 1 | | 1 | 4 |
| append_union_all | 1 | | 1 | 4 |
| merge_union_all | 1 | | 1 | 4 |
| merge_union_distinct | 1 | | 1 | 4 |
| hash_union_distinct | 1 | | 1 | 4 |
| store_index | 1 | | 1 | 4 |
| bushy_space_search | 0 | | 0 | 4 |
| parallel_query | 1 | | 1 | 4 |
| replicated_partition | 0 | | 0 | 4 |
| asel25_primed | 0 | | 0 | 4 |
| index_intersection | 0 | | 0 | 4 |
| index_union | 1 | | 1 | 4 |
| multi_table_store_ind | 0 | | 0 | 4 |
| advanced_aggregation | 0 | | 0 | 4 |
| opportunistic_distinct_view | 1 | | 1 | 4 |
| repartition_degree | 3 | | 1 | 2 |
| scan_parallel_degree | 0 | | 1 | 2 |
| resource_granularity | 10 | | 10 | 2 |
| parallel_degree | 0 | | 1 | 2 |
| statistics_simulate | 0 | | 0 | 4 |
| forceplan | 0 | | 0 | 7 |
| prefetch | 1 | | 1 | 6 |
| metrics_capture | 0 | | 0 | 6 |
| process_limit_action | quiet | | quiet | 2 |
| plan_replace | 0 | | 0 | 4 |
| plan_exists_check | 0 | | 0 | 4 |
| plan_dump | 0 | | 0 | 4 |
| plan_load | 0 | | 0 | 4 |

(39 rows affected)

(return status = 0)

示例 3 查看单个选项的当前设置和缺省设置:

```
1> sp_options show, "index_intersection"
2> go
```

| name | category | currentsetting | defaultsetting | scope |
|--------------------|--------------|----------------|----------------|-------|
| index_intersection | Query Tuning | 0 | 0 | 4 |

```
(1 row affected)
(return status = 0)
```

示例 4 仅显示单个选项的缺省设置:

```
1> sp_options show, "index_intersection", dflt
2> go
```

| name | defaultsetting |
|--------------------|----------------|
| index_intersection | 0 |

```
(1 row affected)
(return status = 0)
```

示例 5 显示某个类别的当前设置和缺省设置:

```
1> sp_options show, "Query Tuning"
2> go
```

Category: Query Tuning

| name | currentsetting | defaultsetting | scope |
|------------------|----------------|----------------|-------|
| optgoal | allrows_mix | allrows_mix | 0 |
| opttimeoutlimit | 10 | 10 | 0 |
| merge_join | 1 | 1 | 4 |
| hash_join | 0 | 0 | 4 |
| nl_join | 1 | 1 | 4 |
| distinct_sorted | 1 | 1 | 4 |
| distinct_sorting | 1 | 1 | 4 |
| distinct_hashing | 1 | 1 | 4 |
| group_sorted | 1 | 1 | 4 |
| group_hashing | 1 | 1 | 4 |
| group_inserting | 0 | 0 | 4 |
| order_sorting | 1 | 1 | 4 |
| append_union_all | 1 | 1 | 4 |

| | | | |
|-----------------------------|-------|-------|---|
| merge_union_all | 1 | 1 | 4 |
| merge_union_distinct | 1 | 1 | 4 |
| hash_union_distinct | 1 | 1 | 4 |
| store_index | 1 | 1 | 4 |
| bushy_space_search | 0 | 0 | 4 |
| parallel_query | 1 | 1 | 4 |
| replicated_partition | 0 | 0 | 4 |
| ase125_primed | 0 | 0 | 4 |
| index_intersection | 0 | 0 | 4 |
| index_union | 1 | 1 | 4 |
| multi_table_store_ind | 0 | 0 | 4 |
| advanced_aggregation | 0 | 0 | 4 |
| opportunistic_distinct_view | 1 | 1 | 4 |
| repartition_degree | 3 | 1 | 2 |
| scan_parallel_degree | 0 | 1 | 2 |
| resource_granularity | 10 | 10 | 2 |
| parallel_degree | 0 | 1 | 2 |
| statistics simulate | 0 | 0 | 4 |
| forceplan | 0 | 0 | 7 |
| prefetch | 1 | 1 | 6 |
| metrics_capture | 0 | 0 | 6 |
| process_limit_action | quiet | quiet | 2 |
| plan replace | 0 | 0 | 4 |
| plan exists check | 0 | 0 | 4 |
| plan dump | 0 | 0 | 4 |
| plan load | 0 | 0 | 4 |

(39 rows affected)

(return status = 0)

示例 6 显示查询调优类别的缺省设置:

```
1> sp_options show, "Query Tuning", dflt
2> go
```

Category: Query Tuning

| name | defaultsetting |
|------------------|----------------|
| optgoal | allows_mix |
| opttimeoutlimit | 10 |
| merge_join | 1 |
| hash_join | 0 |
| nl_join | 1 |
| distinct_sorted | 1 |
| distinct_sorting | 1 |
| distinct_hashing | 1 |

```

group_sorted 1
group_hashing 1
group_inserting 0
order_sorting 1
append_union_all 1
merge_union_all 1
merge_union_distinct 1
hash_union_distinct 1
store_index 1
bushy_space_search 0
parallel_query 1
replicated_partition 0
ase125_primed 0
index_intersection 0
index_union 1
multi_table_store_ind 0
advanced_aggregation 0
opportunistic_distinct_view 1
repartition_degree 1
scan_parallel_degree 1
resource_granularity 10
parallel_degree 1
statistics simulate 0
forceplan 0
prefetch 1
metrics_capture 0
process_limit_action quiet
plan replace 0
plan exists check 0
plan dump 0
plan load 0

(39 rows affected)
(return status = 0)

```

示例 7 显示查询调优类别中设置为非缺省设置的选项:

```

1> sp_options show, "Query Tuning", non_dflt
2> go

```

Category: Query Tuning

| name | currentsetting | defaultsetting |
|----------------------|----------------|----------------|
| repartition_degree | 3 | 1 |
| scan_parallel_degree | 0 | 1 |


```
parallel_degree          0          1
```

```
(3 rows affected)
(return status = 0)
```

示例 8 显示查询调优类别中的选项:

```
1> sp_options, show, null
2> go
```

Category: Query Tuning

| name | currentsetting | defaultsetting | scope |
|-----------------------------|----------------|----------------|-------|
| optgoal | allows_mix | allows_mix | 0 |
| opttimeoutlimit | 10 | 10 | 0 |
| merge_join | 1 | 1 | 4 |
| hash_join | 0 | 0 | 4 |
| nl_join | 1 | 1 | 4 |
| distinct_sorted | 1 | 1 | 4 |
| distinct_sorting | 1 | 1 | 4 |
| distinct_hashing | 1 | 1 | 4 |
| group_sorted | 1 | 1 | 4 |
| group_hashing | 1 | 1 | 4 |
| group_inserting | 0 | 0 | 4 |
| order_sorting | 1 | 1 | 4 |
| append_union_all | 1 | 1 | 4 |
| merge_union_all | 1 | 1 | 4 |
| merge_union_distinct | 1 | 1 | 4 |
| hash_union_distinct | 1 | 1 | 4 |
| store_index | 1 | 1 | 4 |
| bushy_space_search | 0 | 0 | 4 |
| parallel_query | 1 | 1 | 4 |
| replicated_partition | 0 | 0 | 4 |
| asel25_primed | 0 | 0 | 4 |
| index_intersection | 0 | 0 | 4 |
| index_union | 1 | 1 | 4 |
| multi_table_store_ind | 0 | 0 | 4 |
| advanced_aggregation | 0 | 0 | 4 |
| opportunistic_distinct_view | 1 | 1 | 4 |
| repartition_degree | 3 | 1 | 2 |
| scan_parallel_degree | 0 | 1 | 2 |
| resource_granularity | 10 | 10 | 2 |
| parallel_degree | 0 | 1 | 2 |
| statistics_simulate | 0 | 0 | 4 |
| forceplan | 0 | 0 | 7 |

```

prefetch                1                1                6
metrics_capture         0                0                6
process_limit_action    quiet          quiet           2
plan replace            0                0                4
plan exists check      0                0                4
plan dump               0                0                4
plan load               0                0                4
(39 rows affected)
(return status = 0)

```

示例 9 显示查询调优类别的缺省设置列表:

```

1> sp_options show, null, dflt
2> go
Category: Query Tuning

```

| name | defaultsetting |
|-----------------------------|----------------|
| optgoal | allows_mix |
| opttimeoutlimit | 10 |
| merge_join | 1 |
| hash_join | 0 |
| nl_join | 1 |
| distinct_sorted | 1 |
| distinct_sorting | 1 |
| distinct_hashing | 1 |
| group_sorted | 1 |
| group_hashing | 1 |
| group_inserting | 0 |
| order_sorting | 1 |
| append_union_all | 1 |
| merge_union_all | 1 |
| merge_union_distinct | 1 |
| hash_union_distinct | 1 |
| store_index | 1 |
| bushy_space_search | 0 |
| parallel_query | 1 |
| replicated_partition | 0 |
| ase125_primed | 0 |
| index_intersection | 0 |
| index_union | 1 |
| multi_table_store_ind | 0 |
| advanced_aggregation | 0 |
| opportunistic_distinct_view | 1 |
| repartition_degree | 1 |
| scan_parallel_degree | 1 |

```

resource_granularity      10
parallel_degree           1
statistics simulate      0
forceplan                 0
prefetch                  1
metrics_capture           0
process_limit_action      quiet
plan replace              0
plan exists check         0
plan dump                 0
plan load                 0

```

```

(39 rows affected)
(return status = 0)

```

示例 9 显示查询调优类别中设置为非缺省设置的选项:

```

1> sp_options show, null, non_dflt
2> go

```

Category: Query Tuning

| name | currentsetting | defaultsetting |
|----------------------|----------------|----------------|
| repartition_degree | 3 | 1 |
| scan_parallel_degree | 0 | 1 |
| parallel_degree | 0 | 1 |

```

(3 rows affected)
(return status = 0)

```

示例 10 如果您输入 `sp_options` 无法识别的参数, 则会收到以下消息:

```

1> sp_options show, "incorrect option"
2> go

```

```

Msg 19615, Level 16, State 1:
Procedure 'sp_options', Line 436:
No option or category matching 'incorrect option' is
found. Valid categories are:
category

```

```

-----
Query Tuning
(1 row affected)
(return status = 1)

```

示例 11 查看正确用法:

```
1> sp_options help
2> go

Usage:
sp_options [ [show | help
             [, <option_name>|<category_name>|null
             [, dflt | non_dflt | null
             [, <spid>] ] ] ] ]
```

用法

使用 `sp_options` 来查看下列选项的设置:

- set plan dump / load
- set plan exists check
- set forceplan
- set plan optgoal
- set [optCriteria]
- set plan opttimeoutlimit
- set plan replace
- set statistics simulate
- set metrics_capture
- set prefetch
- set parallel_degree number
- set process_limit_action
- set resource_granularity number
- set scan_parallel_degree number
- set repartition_degree number

sp_passthru

说明 (仅限组件集成服务) 使用户可以将 SQL 命令缓冲区传递到远程服务器。

语法 `sp_passthru server, command, errcode, errmsg, rowcount
[, arg1, arg2, ... argn]`

参数 *server*

是 SQL 命令缓冲区将被传递到的远程服务器的名称。此服务器的类必须是支持的非本地服务器类。

command

是 SQL 命令缓冲区。最多可以容纳 255 个字符。

errcode

是远程服务器返回的错误代码 (如果有)。如果在远程服务器上没有任何发生任何错误, 则返回的值为 0。

errmsg

是远程服务器返回的错误消息。最多可以容纳 1024 个字符。只有在 *errcode* 为非零数值时才设置此参数, 否则返回 NULL。

rowcount

是命令缓冲区中最后一个命令影响的行数。如果该命令是 `insert`、`delete` 或 `update`, 则该值表示影响的行数, 即使没有返回任何内容。如果最后一个命令是查询, 则该值表示从外部服务器返回的行数。

arg1 argn

接收命令缓冲区中最后一个命令返回的最后一行的结果。您可以指定最多 250 个 *arg* 参数。所有参数都必须被声明为输出参数。

示例 返回输出参数 *@oradate* 中来自 Oracle 服务器的日期。如果发生 Oracle 错误, 则错误代码放置在 *@errcode* 中, 而相应消息放置在 *@errmsg* 中, 并且 *@rowcount* 设置为 1:

```
sp_passthru ORACLE, "select date from dual", @errcode output,  
@errmsg output, @rowcount output, @oradate output
```

用法

- 用户可通过 `sp_passthru` 将 SQL 命令缓冲区传递到远程服务器。假定所传递的一个或多个 SQL 语句的语法是接收缓冲区的服务器类的本机语法。不执行任何转换或解释。远程服务器的结果被放置在输出参数中 (可选)。

只有在安装和配置了组件集成服务的情况下才使用 `sp_passthru`。

- 您可以在命令缓冲区中包括多个命令。对于某些服务器类, 这些命令必须用分号分隔。有关直通模式中查询缓冲区处理的更完整论述, 请参见《组件集成服务用户指南》。

返回参数

- 输出参数 *arg1 ... argn* 将被设置为由命令缓冲区中最后一个命令返回的最后一行的相应列的值。该参数的位置将确定该参数将包含哪个列的值。*arg1* 从列 1 接收值, *arg2* 从列 2 接收值, 依此类推。
- 如果所具有的可选参数的数目少于返回的列数, 则忽略超出的列。如果参数数目超过列数, 则剩余参数将被设置为 NULL。
- 将进行尝试以将每一列转换为输出参数的数据类型。如果这些数据类型类似, 足以允许隐式转换, 则尝试将成功。有关隐式转换的信息, 请参见《参考手册: 构件块》的第 2 章“Transact-SQL 函数”。有关在处于直通模式时表示来自各服务器类的数据类型的信息, 请参见《组件集成服务用户指南》。

权限

任何用户都可以执行 sp_passthru。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_autoconnect](#)、[sp_remotesql](#)

sp_password

| | |
|----|---|
| 说明 | Adaptive Server 15.7 和更高版本不支持此系统过程。若要添加或更改 Adapter Server 的登录帐户的口令，请使用 <code>create login</code> 和 <code>alter login</code> 命令。 |
| 语法 | 无 |
| 参数 | 无 |
| 用法 | 无 |

sp_passwordpolicy

说明 一个接口，具有 `sso_role` 的用户可以使用该接口配置登录名和口令策略选项。

语法 指定、删除并列出新口令复杂程度选项：

```
sp_passwordpolicy {"set" | "clear" | "list"}, policy_option, option_value
```

检验口令复杂程度选项：

```
sp_passwordpolicy 'validate password options'
```

为网络登录口令加密生成非对称密钥对：

```
sp_passwordpolicy "regenerate keypair"
```

使口令到期：

```
sp_passwordpolicy "expire role passwords", "[rolename | wildcard]"
```

```
sp_passwordpolicy "expire login passwords", "[login_name | wildcard]"
```

```
sp_passwordpolicy "expire stale role passwords", "datetime"
```

```
sp_passwordpolicy "expire stale login passwords", "datetime"
```

显示所有命令、选项及其值的简要说明：

```
sp_passwordpolicy "help"
```

参数

set

为选项设定一个值。使用 `set` 时，必须指定 *policy_option*。

clear

删除 `master.dbo.sysattributes` 表中指定选项所对应的行。如果未指定任何策略选项，`clear` 将删除 `sysattributes` 表中的所有选项行。使用 `clear` 时，必须指定 *policy_option*。

list

列出指定选项的值。使用 `list` 时，必须指定 *policy_option*。

policy_option, option_value

字符串或 (varchar)。是 set、clear 和 list 的选项参数，*option_value* 为其值：

- **allow password downgrade** — 结束口令降级周期。在口令降级期间，口令以新旧两种编码存储在 **syslogins** 中，以允许在服务器降级（例如，降级至 Adaptive Server 15.0.2）时保留用户口令。
- **disallow simple passwords** — 值为 1 将打开该选项，值为 0 将关闭该选项。
- **enable last login updates** — 在 Adaptive Server 鉴定中启用或禁用记录每次登录发生时的时间戳的代码。第一个参数 “set” 设置此属性的值。“list” 显示此属性的当前值，参数 “clear” 从 **sysattributes** 中删除该行。在升级后或新安装时，**sysattributes** 中不存在此属性。此属性行不存在或者值为 1 时出现登录时间戳。如果此属性值为 0，则不保留登录时间戳。
- **expire login** — 指定在创建或重置登录名时将登录名状态更改为到期。您在首次登录时需要更改口令。
- **keypair regeneration period** — 指示 RSA 密钥对的重新生成期间。其选项值为 { ([*keypair regeneration frequency*], [*datetime of first generation*]) | (*keypair regeneration frequency*, [*datetime of first generation*]) }

keypair regeneration frequency — 是 RSA 密钥对的重新生成频率。有效值（以小时为单位）的范围为 1 到 8,760。缺省值为 NULL，这种情况下每隔 24 小时重新生成一次密钥对。它使用以下各项来指定持续时间的格式指示符：

- 'T*M' — 指示持续时间（以分钟为单位），将星号 (*) 替换为数字值，如 “T2M” 表示两分钟。
- 'H' — 指示持续时间（以小时为单位）。
- 'D' — 指示持续时间（以天为单位）。如果不指定其它格式，则这是缺省值。
- 'W' — 指示持续时间（以周为单位）。
- 'M' — 指示持续时间（以月为单位）。
- 'Y' — 指示持续时间（以年为单位）。

datetime of first generation — 是首次生成密钥对的日期和时间。如果仅为 *datetime of first generation* 的值指定时间，RSA 密钥对重新生成会被排定在下一个 24 小时周期内的时间进行。如果：

- 指定 *datetime of first generation* — 只要该时间过去， Adaptive Server 会立即重新生成新的 RSA 密钥对；否则 Adaptive Server 会等到指定的时间为止。
- 不指定 *datetime of first generation* — Adaptive Server 会在通过最新 RSA 密钥对生成时间的基础上加上 *keypair regeneration period* 而获得的时间重新生成新的 RSA 密钥对（如果此计算时间未过）；否则 Adaptive Server 会立即重新生成新的 RSA 密钥对。

后续的密钥对生成是根据最新密钥对生成时间以及 *keypair regeneration period* 值而发生的。

注释 不能同时将 *keypair regeneration frequency* 和 *datetime of first generation* 的值设置为 NULL。

- *keypair error retry [wait | count]* — 指定可以针对尝试失败后重新生成密钥对而设置的多种配置：
 - *wait* — 指定发生故障后在重新生成密钥对之前等待的时间。
 - *count* — 指定您希望 Adaptive Server 在失败后多少次尝试重新生成密钥对。
- *maximum failed logins* — 指示会话中允许的失败登录的最大次数，在此次数之后帐号将被锁定。
- *min alpha in password* — 指示口令中字母字符的最小数目。
- *min digits in password* — 指示口令中允许包含的数字最小数目。
- *min lower char in password* — 指示口令中允许包含的小写字母最小数目。
- *min special char in password* — 指示口令中允许包含的特殊字符最小数目。
- *min upper char in password* — 指示口令中允许包含的大写字母最小数目。
- *minimum password length* — 指示口令的最小长度。
- *password exp warn interval* — 指示口令有效期警告间隔（天数）。
- *systemwide password expiration* — 指示系统范围的口令有效期（以天为单位）。

"expire login passwords", "[login_name | wildcard]"

使登录口令、所有登录名或与通配符模式匹配的登录名到期。master 数据库目录 `syslogins` 中的列状态更新为状态位 `LOGIN_EXPIRED (0x4)`，表示口令已到期。

"expire role passwords", "[rolename | wildcard]"

使一个角色、所有角色或者与通配符模式匹配的角色的口令到期。master 数据库目录 `sysrvroles` 中的列状态更新为状态位 `ROLE_EXPIRED (0x4)`，表示口令已到期：

"expire stale login passwords", "datetime"

在指定日期时间后尚未更改的登录口令到期。master 数据库目录 `syslogins` 中的列状态更新为状态位 `LOGIN_EXPIRED (0x0004)`，表示口令已到期。有关如何输入 `datetime` 值的说明，请参见 *Adaptive Server 15.0 《参考手册：构件块》* 的第一章“系统和用户定义的数据类型”中的“输入日期和时间数据”。

"expire stale role passwords", "datetime"

在指定日期时间后尚未更改的角色口令到期。master 数据库目录 `sysrvroles` 中的列状态更新为状态位 `ROLE_EXPIRED (0x4)`，表示口令已到期。

"regenerate keypair"

生成用于网络登录口令加密的非对称密钥对。此选项没有目录更新；这些操作仅发生在内存字段中。

'validate password options'

报告设置的口令复杂程度选项值中的错误或不一致，包括长度和有效期方面的错误和不一致。结果以表格格式报告，其中每一行表示一个验证步骤、该步骤的结果以及执行的验证测试。结果为通过、未通过或不适用 (NA) 这三者之一。如果有任何验证测试未通过，则返回状态设置为 1。

示例

示例 1 将口令有效期警告间隔设置为口令到期前七天：

```
sp_passwordpolicy 'set',
    'password exp warn interval', '7'
```

示例 2 列出特殊字符的最小数目的选项：

```
sp_passwordpolicy 'list',
    'min special char in password'
```

示例 3 将 `disallow simple passwords` 重置为缺省值：

```
sp_passwordpolicy 'clear', 'disallow simple passwords'
```

示例 4 以下示例演示了如何使用 `validate password options`。为清楚起见，对其中的输出重新设置了格式，这些输出与您在执行此过程时屏幕上显示的输出并不相似

- 以下口令复杂程度选项及其值存储在服务器中：

```

minimum password length:      8
min alpha in password:       2
min digits in password:      2
min upper char in password:  2
min lower char in password:  2

```

若要验证这些选项，请输入：

```
sp_passwordpolicy 'validate password options'
```

| Validation Step | Pass/Fail/NA | Validation Test |
|-----------------------------|--------------|---|
| ----- | ----- | ----- |
| min alpha in password | Fail | 'min alpha in password' >= 'min upper char in password' + 'min lower char in password' |
| minimum password length - 1 | Pass | 'minimum password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| minimum password length - 2 | Pass | 'minimum password length' >= 'min digits in password' + min special char in password' + 'min upper char in password' + 'min lower char in password' |
| maximum password length - 1 | Pass | 'max password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| maximum password length - 2 | Pass | 'max password length' >= 'min digits in password' + 'min special char in password' + 'min upper char in password' + 'min lower char in password' |
| password exp warn interval | NA | 'password exp warn interval' <= 'systemwide password expiration' |

```

(6 rows affected)
(return status = 1)

```

存在一处故障：min upper char in password + min lower char in password 之和大于 min alpha in password 的值，因此验证步骤 min alpha in password 失败。

示例 5 将 HouseKeeper 任务设置为每隔两个小时自动重新生成密钥对，开始时间为 2007 年 8 月 15 日中午 12:01:

```
sp_passwordpolicy "set", "keypair regeneration period",
    "2H", "Aug 15 2007 12:01 AM"
```

示例 6 设置 Adaptive Server 在失败后尝试重新生成密钥对之前应等待的时间:

```
sp_passwordpolicy 'set', 'keypair error retry wait', '10'
```

示例 7 将 Adaptive Server 在失败后尝试重新生成密钥对的次数设置为 5:

```
sp_passwordpolicy 'set', 'keypair error retry count', '5'
```

示例 8 显示所有命令、选项及其值的简要说明:

```
sp_passwordpolicy "help"
go
sp_passwordpolicy Usage: sp_passwordpolicy 'help'
sp_passwordpolicy Usage: sp_passwordpolicy command [, option1 [, option2 [,
option3]]]
sp_passwordpolicy commands:
sp_passwordpolicy 'set',
    {'enable last login updates' | 'disallow simple passwords' |
    'min digits in password' | 'min alpha in password' |
    'min special char in password' | 'min upper char in password' |
    'min lower char in password' | 'password exp warn interval' |
    'systemwide password expiration' | 'minimum password length' |
    'maximum failed logins' | 'expire login' |
    'allow password downgrade' | 'keypair error retry wait' |
    'keypair error retry count'},
    'value'
sp_passwordpolicy 'set', 'keypair regeneration period',
    {'regeneration_period' |
    null, 'datetime' |
    'regeneration_period', 'datetime'}
sp_passwordpolicy 'list',
    ['enable last login updates' | 'disallow simple passwords' |
    'min digits in password' | 'min alpha in password' |
    'min special char in password' | 'min upper char in password' |
    'min lower char in password' | 'password exp warn interval' |
    'systemwide password expiration' | 'minimum password length' |
    'maximum failed logins' | 'expire login' |
    'allow password downgrade' |
    'keypair error retry wait' | 'keypair error retry count' |
    'keypair regeneration period']
sp_passwordpolicy 'clear',
```

```

        {'enable last login updates' | 'disallow simple passwords' |
        'min digits in password' | 'min alpha in password' |
        'min special char in password' | 'min upper char in password' |
        'min lower char in password' | 'password exp warn interval' |
        'systemwide password expiration' | 'minimum password length' |
        'maximum failed logins' | 'expire login' |
        'keypair error retry wait' |
        'keypair error retry count' | 'keypair regeneration period'}
sp_passwordpolicy 'expire login passwords'[, '{loginame | wildcard}']
sp_passwordpolicy 'expire role passwords'[, '{rolename | wildcard}']
sp_passwordpolicy 'expire stale login passwords', 'datetime'
sp_passwordpolicy 'expire stale role passwords', 'datetime'
sp_passwordpolicy 'regenerate keypair'[, 'datetime']
sp_passwordpolicy 'validate password options'
(return status = 0)

```

示例 9 验证存储在 Adaptive Server 中的以下选项:

```

minimum password length:      8
min digits in password:       2
min special char in password:  2
min alpha in password:        6
min upper char in password:    3
min lower char in password:    3

```

```
sp_passwordpolicy 'validate password options'
```

| Validation Step | Pass/Fail/NA | Validation Test |
|-----------------------------|--------------|--|
| min alpha in password | Pass | 'min alpha in password' >= 'min upper char in password' + 'min lower char in password' |
| minimum password length-1 | Fail | 'minimum password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| minimum password length-2 | Fail | 'minimum password length' >= 'min digits in password' + 'min special char in password' + 'min upper char in password' + 'min lower char in password' |
| maximum password length-1 | Pass | 'max password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| maximum password length - 2 | Pass | 'max password length' >= 'min digits in password' + 'min |

```

special char in password' + 'min
upper char in password' + 'min
lower char in password'

password exp warn interval    NA    'password exp warn interval' < =
                                'systemwide password expiration'

(6 rows affected)
(return status = 1)

```

步骤 2 和步骤 3 中存在两处故障。min digits in password、min special char in password 和 min alpha in password 之和大于 minimum password length 的值，因此验证步骤 minimum password length -1 失败。min digits in password、min special char in password、min upper char in password 和 min lower char in password 之和大于 minimum password length 的值，因此验证步骤 minimum password length -2 失败。

示例 10 以下示例说明了 'validate password options' 选项。为清楚起见，对其中的输出重新设置了格式，这些输出与您在执行此过程时屏幕上显示的输出并不相似。

以下口令复杂程度选项及其值存储在服务器中：

```

minimum password length:      8
min alpha in password:        2
min digits in password:       2
min upper char in password:   2
min lower char in password:   2

```

```

sp_passwordpolicy 'validate password options'
Validation Step      Pass/Fail/NA    Validation Test
-----
min alpha in password    Fail            'min alpha in password' > = 'min
upper char in password + 'min
lower char in password'

minimum password length - 1    Pass            'minimum password length' > = 'min
digits in password' + 'min special
char in password' + 'min alpha in
password'

minimum password length - 2    Pass            'minimum password length' > = 'min
digits in password' + min special
char in password' + 'min upper
char in password' + 'min lower
char in password'

maximum password length - 1    Pass            'max password length' > = 'min
digits in password' + 'min

```

```

special char in password' + 'min
alpha in password'

maximum password length - 2 Pass      'max password length' >= 'min
digits in password' + 'min special
char in password' + 'min upper
char in password' + 'min lower
char in password'

password exp warn interval  NA        'password exp warn interval' <=
'systemwide password expiration'

(6 rows affected)
(return status = 1)

```

若要验证这些选项，请输入：

存在一处故障：min upper char in password + min lower char in password 之和大于 min alpha in password 的值，因此验证步骤 min alpha in password 失败。

验证存储在 Adaptive Server 中的以下选项：

```

minimum password length:      8
min digits in password:      2
min special char in password: 2
min alpha in password:       6
min upper char in password:   3
min lower char in password:  3

```

```

sp_passwordpolicy 'validate password options'
Validation Step      Pass/Fail/NA      Validation Test
-----
min alpha in password      Pass              'min alpha in password' >= 'min upper
char in password' + 'min lower
char in password'

minimum password length-1  Fail              'minimum password length' >= 'min
digits in password' + 'min special
char in password' + 'min alpha in
password'

minimum password length-2  Fail              'minimum password length' >= 'min
digits in password' + 'min special
char in password' + 'min upper
char in password' + 'min lower
char in password'

maximum password length-1  Pass              'max password length' >= 'min
digits in password' + 'min special
char in password' + 'min alpha in
password'

```



```

maximum password length - 2    Pass    'max password length' > = 'min
                                digits in password' + 'min
                                special char in password' + 'min
                                upper char in password' + 'min
                                lower char in password'

password exp warn interval    NA      'password exp warn interval' < =
                                'systemwide password expiration'

(6 rows affected)
(return status = 1)

```

步骤 2 和步骤 3 中存在两处故障。

min digits in password、min special char in password 和 min alpha in password 之和大于 minimum password length 的值，因此验证步骤 minimum password length -1 失败。min digits in password、min special char in password、min upper char in password 和 min lower char in password 之和大于 minimum password length 的值，因此验证步骤 minimum password length -2 失败。

验证存储在 Adaptive Server 中的以下选项：

```

minimum password length:      8
min digits in password:      11
min special char in password: 11
min alpha in password:       11
min upper char in password:   1
min lower char in password:   1

```

```
sp_passwordpolicy 'validate password options'
```

| Validation Step | Pass/Fail/NA | Validation Test |
|---------------------------|--------------|---|
| min alpha in password | Pass | 'min alpha in password' > = 'min upper char in password' + 'min lower char in password' |
| minimum password length-1 | Fail | 'minimum password length' > = 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| minimum password length-2 | Fail | 'minimum password length' > = 'min digits in password' + 'min special char in password' + 'min upper char in password' + 'min lower char in password' |
| maximum password length-1 | Fail | 'max password length' > = 'min digits in password' + 'min special |

```

char in password' + 'min alpha in
password'

maximum password length - 2    Pass    'max password length' >= 'min
digits in password' + 'min special
char in password' + 'min upper
char in password' + 'min lower
char in password'

password exp warn interval    NA    'password exp warn interval' <=
'systemwide password expiration'

(6 rows affected)
(return status = 1)

```

有三处故障，其中包括一处严重故障，即测试最大口令长度时出现的一处故障：各个必需的口令组成部分之和大于允许的最大口令。

验证存储在 Adaptive Server 中的以下选项：

```

minimum password length:      8
min digits in password:       2
min special char in password: 1
min alpha in password:        4
min upper char in password:   0
min lower char in password:   0

```

```
sp_passwordpolicy 'validate password options'
```

| Validation Step | Pass/Fail/NA | Validation Test |
|---------------------------|--------------|--|
| ----- | ----- | ----- |
| min alpha in password | Pass | 'min alpha in password' >= 'min upper char in password' + 'min lower char in password' |
| minimum password length-1 | Pass | 'minimum password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |
| minimum password length-2 | Pass | 'minimum password length' >= 'min digits in password' + 'min special char in password' + 'min upper char in password' + 'min lower char in password' |
| maximum password length-1 | Pass | 'max password length' >= 'min digits in password' + 'min special char in password' + 'min alpha in password' |

```

maximum password length - 2   Pass          'max password length' >= 'min
                                digits in password' + 'min
                                special char in password' + 'min
                                upper char in password' + 'min
                                lower char in password'

password exp warn interval    NA           'password exp warn interval' <=
                                'systemwide password expiration'

(6 rows affected)
(return status = 0)

```

这些设置不存在故障。此示例报告所有 5 行都已返回，且返回状态为 0。

用法

`sp_passwordpolicy` 信息存储在 `master.dbo.sysattributes` 表中。

重新生成密钥对

一旦 Adaptive Server 重新生成新的 RSA 密钥对，后续的生成将会使用上次生成密钥对时的公式，并结合使用您为 *keypair regeneration frequency* 指定的值。

keypair regeneration period 的值存储在 `master..sysattributes` 中的新口令策略类下面。

该选项的缺省值 NULL 表示该行不存在于 `sysattributes` 中，而且密钥对是在每隔 24 小时重新启动 Adaptive Server 时生成的。

以下两个存储过程具有同样的效果：

```

sp_passwordpolicy 'set', 'keypair regeneration period', NULL [,
    datetime of first generation]
sp_passwordpolicy 'regenerate keypair' [, datetime of first generation]

```

以下两个全局变量使用 *keypair regeneration period* 中的信息：

- `@@lastkpgendate` — 反映上次生成密钥对的日期和时间。
- `@@nextkpgendate` — 反映下次生成的时间。

登录口令复杂程度检查

以下登录口令复杂程度检查已扩展到角色口令：

- disallow simple passwords
- min digits in password
- min alpha in password
- min special char in password:
- min upper char in password

- min lower char in password
- systemwide password expiration
- password exp warn interval
- minimum password length
- maximum failed logins
- expire login

高可用性和口令策略选项

Adaptive Server 高可用性功能在主服务器和辅助服务器之间同步以下口令策略选项:

- disallow simple passwords
- min digits in password
- min alpha in password
- min special char in password:
- min upper char in password
- min lower char in password
- systemwide password expiration
- password exp warn interval
- minimum password length
- maximum failed login
- expire login
- keypair regeneration period
- keypair error retry wait
- keypair error retry count

Adaptive Server 使用 “password policy” 定额属性在主服务器和辅助服务器上检查这些值是否存在不一致 (keypair regeneration period、keypair error retry wait 和 keypair error retry count 除外)。如果这些值在两个服务器上相同, 则高可用性建议检查会成功, 否则会失败。例如:

```
sp_companion "MONEY1", do_advisory, 'all'  
go
```

| Attribute Name | Attrib Type | Local Value | Remote Value | Advisory |
|----------------|-------------|-------------|--------------|----------|
| ----- | ----- | ----- | ----- | ----- |

| | | | | |
|-----------------|-------------|----|----|---|
| expire login | password po | 1 | 0 | 2 |
| maximum failed | password po | 3 | 5 | 2 |
| min alpha in pa | assword po | 10 | 12 | 2 |

输出的 `advisory` 列的值设置为 2，表示除非两个协同服务器上的值相同，否则用户不能进行聚簇操作。

`sp_companion do_advisory` 的输出也指示两个服务器上特定口令策略检查的不一致。

审计

`sp_passwordpolicy` 中的 `set` 和 `clear` 命令通过审计事件 115 “Password Administration” 进行审计。

审计选项 “password” 审计以下操作：

- `sp_passwordpolicy 'set', 'option_name', 'option_value'`
- `sp_passwordpolicy 'clear', 'option_name'`
- `sp_passwordpolicy 'expire login passwords'`
- `sp_passwordpolicy 'expire stale login passwords'`
- `sp_passwordpolicy 'regenerate keypair'`
- `sp_passwordpolicy 'expire role passwords'`
- `sp_passwordpolicy 'expire stale role passwords'`

“password” 审计选项还审计 RSA 密钥对重新生成期间的管理（会生成 `AUD_EVT_PASSWORD_ADMIN(115)` 审计事件）。

sp_pciconfig

说明

管理 Java PCI Bridge。启用或禁用参数和指令、更改配置值和报告配置值。

注释 请勿使用 `sp_pciconfig` 更改参数或指令，除非 Sybase 技术支持部门指示这样做。

语法

```
sp_pciconfig {
    disable { directive | argument } |
    enable { directive | argument } |
    list { list_type [, formatted ] | units | units, units_type [, formatted ] } |
    report { directive [, formatted ] |
        directive, args [, formatted ] |
        argument [, formatted ] } |
    update { number_arg, old_value new_value }
```

参数

disable

禁用指定的指令或参数。

directive

是任意有效指令的名称。

argument

是任意有效参数的名称。

enable

启用指定的指令或参数。

list

以诸如 `sp_pciconfig "list", "directive"` 或 `sp_pceiconfig "list", "enabled"` 之类的形式列出相关参数的组。此外，还会以诸如 `sp_pciconfig "list", "units", "switch"` 之类的形式列出特定类型的所有参数。

list_type

指定列表的类型。其值包括：

- `directives` — 指令列表
- `enabled` — 已启用的参数的列表
- `disabled` — 已禁用的参数的列表
- `argnames` — 参数名列表

formatted

指定要为所显示的列表设置格式以提高可读性。

注释 在设置了格式的报表中，改善可读性的过程可能造成宽列被截断。此外，列标题可能会被覆盖，可能会与实际的表列名称不匹配。如果将对输出进行分析或潜在的数据截断不可接受，请勿为报告设置格式。

units

与 `list` 搭配使用时，生成当前使用的 `units_type` 的列表。

report

根据提供的参数创建报告。通常用于生成某个参数的报告，以查看其当前值以及其是否启用。也可用于生成指令或其参数的报告。

directive

指定某一指定指令内的所有参数。

update

修改单位为数字的参数的数值。不能与单位为开关的参数搭配使用。

number_arg

是 `units` 为数字的一个参数。

old_value

是 `number_arg_name` 的当前值。

new_value

是 `number_arg_name` 的新值。

用法

启用和禁用像开关一样工作的指令。如果指令：

- 已启用 — `Adaptive Server` 使用为每个参数配置的值（启用或禁用）。这是存储在 `sybpcidb` 中的值。
- 已禁用 — 尽管每个参数的基值保留在 `sybpcidb` 中，但 `Adaptive Server` 不考虑为每个参数配置的值（启用或禁用），而是将该指令的所有参数都视作已禁用。

参数可以分别启用或禁用。 `sp_pciconfig` 指令的参数属于以下类型：

- `switch` — 此类参数启用或禁用某项功能。例如，如果用于日志记录的参数已启用，则会生成一个日志文件；如果用于日志记录的参数已禁用，则不会生成日志文件。
- `string` — 此类参数用于字符串和数字，其处理方式与字符串类似。启用字符串参数可确保 `Adaptive Server` 使用配置的值。禁用字符串参数意味着， `Adaptive Server` 会忽略配置的值，而使用缺省值。配置的值和缺省值可以相同，也可以不同。

表 1-26: sp_pciconfig 的配置指令

| 指令 | 说明 |
|-------------------|----------------------------------|
| PCI_BRIDGE_X_OPT | PCI Bridge 配置参数 |
| PCI_BRIDGE_LOGOPT | 插件 <code>diagserver</code> 的报告工具 |
| PCI_BRIDGE_INSTR | PCI Bridge 的检测设置 |

表 1-27: PCI_BRIDGE_X_OPT 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|-------------------------------|--------|------|------|--|
| pci_xopt_maxthreads | number | 1056 | 已启用 | PCI Bridge PLB 控制的可用线程数上限。 |
| pci_xopt_event_scheduling | number | 0 | 已启用 | 缺省的 PCI Bridge 调度。 |
| pci_xopt_failover_engine | number | -1 | 已启用 | 槽应故障切换到的缺省引擎。 |
| pci_xopt_runtime_alloc_escape | number | 1 | 已启用 | 允许在内存分配请求超出 PC Bridge 的最大内存分配单元时进行运行期溢出。 |
| pci_xopt_slotring_cycle | number | -1 | 已启用 | 禁用 PCI Bridge 槽环清洗。 |
| pci_xopt_slotring_wash_th | number | 76 | 已启用 | 缺省的 PCI Bridge 槽环清洗阈值百分比。 |

表 1-28: PCI_BRIDGE_LOGOPT 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|--------------------|--------|-----|------|---------------------------------------|
| pci_logopt_asehi | switch | 无 | 已禁用 | PCI Bridge ASE 主机接口调度日志记录。 |
| pci_logopt_jst | switch | 无 | 已禁用 | PCI Bridge Job Scheduler 任务调度日志记录。 |
| pci_logopt_jvm | switch | 无 | 已禁用 | PCI Bridge JVM 调度日志记录。 |
| pci_logopt_omni | switch | 无 | 已禁用 | PCI Bridge OMNI 调度日志记录。 |
| pci_logopt_pci | switch | 无 | 已禁用 | 常规 PCI Bridge 日志记录 (probe [pci/pca])。 |
| pci_logopt_runtime | switch | 无 | 已禁用 | PCI Bridge 运行期调度日志记录。 |
| pci_logopt_xml | switch | 无 | 已禁用 | PCI Bridge XML 调度日志记录。 |

表 1-29: PCI_BRIDGE_INSTR 参数

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|---------|--------|-----|------|-----------------------------|
| BRIDGE | number | 1 | 已禁用 | 强制进行完全检测 (有杂音)。 |
| CELL | number | 1 | 已禁用 | 强制所有 CELL 同步进行报告。 |
| JAVA | number | 1 | 已禁用 | 强制与 Java 相关的所有条目进行报告。 |
| JCS | number | 1 | 已禁用 | 强制所有 JCS 条目进行报告。 |
| JDBC | number | 1 | 已禁用 | 强制所有 JDBC 条目进行报告。 |
| JVMHOST | number | 1 | 已禁用 | 强制所有 ASE JVM 主机 API 条目进行报告。 |
| JVMJNI | number | 1 | 已禁用 | 强制所有 JVM JNI 外部条目进行报告。 |
| PCIS | number | 1 | 已禁用 | 强制所有 PCI 服务代码进行报告。 |
| PLB | number | 1 | 已禁用 | 强制所有 PLB 代码进行报告。 |

| 参数 | 单元类型 | 缺省值 | 缺省状态 | 说明 |
|-----------------|--------|-----|------|------------------------------|
| SLOTRING | number | 1 | 已禁用 | 强制所有“槽环”代码进行报告。 |
| SYNC | number | 1 | 已禁用 | 强制所有同步代码进行报告。 |
| TPM | number | 1 | 已禁用 | 强制所有 TPM 代码进行报告。 |
| fetch_classdata | number | 1 | 已启用 | 强制所有 fetch_classdata 命中进行报告。 |
| pcis_service | number | 2 | 已禁用 | 强制所有 pcis_service 命中冻结。 |

权限 只有系统管理员才能执行 `sp_pciconfig` 来更改 PCI 子系统的设置。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **存储过程** [sp_jreconfig](#)

sp_placeobject

| | |
|----|---|
| 说明 | 指定表或索引的未来空间分配将在特定段上进行。 |
| 语法 | <code>sp_placeobject segname, objname</code> |
| 参数 | <p>segname 是将表或索引放置到的段的名称。</p> <p>objname 是为其将后续的空间分配放置于段 segname 上的表或索引的名称。 请以 “tablename.indexname” 形式指定索引名称。</p> |
| 示例 | <p>示例 1 指定表 authors 的所有后续空间分配在名为 “segment3” 的段上进行：</p> <pre>sp_placeobject segment3, authors</pre> <p>示例 2 指定 employee 表的名为 employee_nc 的索引的所有后续空间分配在名为 indexes 的段上进行：</p> <pre>sp_placeobject indexes, 'employee.employee_nc'</pre> |
| 用法 | <ul style="list-style-type: none">• 您不能更改系统表的将来空间分配的位置。• 将表或索引放置于特定段上并不影响任何现有表或索引数据的位置。它只影响将来的空间分配。其中包括表 / 索引中的所有现有分区以及以后添加的任何新分区（如果没有为新分区指定任何段）。更改表或索引所使用的段可以将数据分布在多个段上。• 如果您将 sp_placeobject 用于聚簇索引，则该表将与索引一起移动。• 您可以在用 create table 或 create index 创建表或索引时指定段。也可以将分区级别的一个段指定为分区定义的一部分。没有段规范的分区将使用在表 / 索引级别指定的段。如果没有为表 / 索引级别指定任何段，数据将进入缺省段。• 在 sp_placeobject 跨多个磁盘片段拆分一个表或索引，诊断命令 dbcc 将显示在执行 sp_placeobject 前已用于存储的片段上驻留的数据的有关信息。忽略这些消息。 |
| 权限 | 只有表所有者、数据库所有者或系统管理员才能执行 sp_placeobject 。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [alter table](#)、[dbcc](#)

系统过程 [sp_addsegment](#)、[sp_dropsegment](#)、[sp_extendsegment](#)、[sp_helpindex](#)、[sp_helpsegment](#)

sp_plan_dbccdb

说明 为新的 dbccdb 和 dbccalt 数据库推荐合适的大小、为 dbccdb 和 dbccalt 列出合适的设备，并为目标数据库建议高速缓存大小和合适的工作进程数。

语法 sp_plan_dbccdb [dbname]

参数 dbname

指定目标数据库的名称。如果未指定 *dbname*，sp_plan_dbccdb 会为 master..sysdatabases 中的所有数据库提出建议。

示例 1 就创建适合于检查 master 数据库的 dbccdb 数据库返回配置建议。dbccdb 数据库在此命令运行时已存在，因此提供现有数据库的大小以供比较：

```
sp_plan_dbccdb master

Recommended size for dbccdb database is 50MB (data = 48MB, log = 2MB).
dbccdb database already exists with size 280MB.
Recommended values for workspace size, cache size and process count are:
dbname          scan ws  text ws  cache  comp mem  process count
master          128K   48K     640K   0K      1
```

示例 2 就创建适合于检查服务器中所有数据库的 dbccdb 数据库返回配置建议。输出包含压缩内存需求，仅对于使用任何压缩设备的存档数据库，压缩内存需求才具有非零值。在运行此命令时不存在任何 dbccdb 数据库：

```
sp_plan_dbccdb

Recommended size for dbccdb database is 50MB (data = 48MB, log = 2MB).
dbccdb database already exists with size 280MB.
Recommended values for workspace size, cache size and process count are:
dbname          scan ws  text ws  cache  comp mem  process count
master          128K   48K     640K   0K      1
tempdb         656K  176K    1280K   0K      2
model           64K   48K     640K   0K      1
sybssystemdb    64K   48K     640K   0K      1
sybssystemprocs 1488K  384K    640K   0K      1
sybsecurity    272K   80K    1280K   0K      2
adb            80K   64K    1920K  12M     3
```

示例 3 就创建适合于检查 pubs2 的 dbccdb 数据库返回配置建议：

```
sp_plan_dbccdb pubs2

Recommended size for dbccdb is 4MB.
Recommended devices for dbccdb are:
Logical Device Name      Device Size Physical Device Name
-----
sprocdev                28672      /remote/sybase/devices/srv_sprocs_dat
tun_dat                 8192      /remote/sybase/devices/srv_tun_dat
tun_log                 4096      /remote/sybase/devices/srv_tun_log
Recommended values for workspace size, cache size and process count are:
dbname   scan ws   text ws   cache   process count
pubs2    64K        64K      640K    1
```

用法

- `sp_plan_dbccdb` 建议用于创建新的 `dbccdb` 和 `dbccalt` 数据库的合适大小，列出用于新数据库的合适设备，并且建议用于目标数据库的高速缓存大小和工作进程的合适数目。
- 如果您指定 `dbccdb`，则 `sp_plan_dbccdb` 将建议用于 `dbccalt`（替代数据库）的值。如果您指定 `dbccalt`，则 `sp_plan_dbccdb` 将建议用于 `dbccdb` 的值。
- `sp_plan_dbccdb` 不报告现有 `dbccdb` 和 `dbccalt` 数据库的值。若要收集现有 `dbccdb` 或 `dbccalt` 数据库的配置参数，请使用 `sp_dbcc_evaluatedb`。
- 有关用于维护 `dbccdb` 以及用于从 `dbccdb` 生成报告的 `dbcc` 存储过程的信息，请参见第 4 章“`dbcc` 存储过程”。

权限

只有系统管理员或数据库所有者才能执行 `sp_plan_dbccdb`。只有系统管理员才能在不指定数据库名的情况下执行 `sp_plan_dbccdb`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `dbcc`

系统过程 `sp_dbcc_evaluatedb`

sp_poolconfig

说明 在数据高速缓存中创建内存池，删除其中的内存池，调整这些内存池的大小，以及提供有关这些内存池的信息。

语法 在现有高速缓存中创建内存池或者更改内存池的大小：

```
sp_poolconfig cache_name[, "mem_size [P | K | M | G]", "config_poolK"  
[, "affected_pool K"], instance instance_name]
```

更改池的清洗大小：

```
sp_poolconfig cache_name, "affected_poolK", "wash=size[P|K|M|G]"
```

更改池的异步预取百分比：

```
sp_poolconfig cache_name, "affected_poolK",  
"local async prefetch limit=percent "
```

参数

cache_name

是现有数据高速缓存的名称。

mem_size

是要创建的内存池的大小或具有指定 I/O 大小的现有池的新总大小。池的最小大小为 256 个逻辑服务器页。对于逻辑页大小为 2 K 的服务器，最小大小为 256 K。指定大小的单位，用 **P** 表示页，用 **K** 表示千字节，用 **M** 表示兆字节，用 **G** 表示千兆字节。缺省单位为千字节。

config_pool

是内存池中（在其中内存将被分配或删除）执行的 I/O 大小。

有效 I/O 大小是逻辑页大小的倍数，最大是该大小的 4 倍。

affected_pool

是要释放内存的或要修改池属性（如 'wash size' 和 'prefetch limit'）的内存池中执行的 I/O 大小。如果未指定 **affected_pool**，则内存取自最低逻辑页大小的内存池。

instance_name

在集群环境中— 是要调整缓冲池的实例的名称。

wash=size

更改内存池的清洗大小（高速缓存中 Adaptive Server 将脏页写入磁盘的位置）。

local async prefetch limit=percent

设置可用于保存已由异步预取读入高速缓存、但尚未使用的缓冲区的池中缓冲区所占的百分比。有效值为 0 到 100。如将预取限制设置为 0，将禁用池中的异步预取操作。

示例

示例 1 在 `pub_cache` 数据高速缓存中创建空间为 10MB 的 16K 池。所有空间都取自缺省的 2K 内存池：

```
sp_poolconfig pub_cache, "10M", "16K"
```

示例 2 从 `pub_cache` 的 64K 池中取出 16MB 空间创建到 32K 池：

```
sp_poolconfig pub_cache, "16M", "32K", "64K"
```

示例 3 报告 `pub_cache` 的当前配置：

```
sp_poolconfig "pub_cache"
```

示例 4 从 `pub_cache` 删除 16K 内存池，并将分配给它的所有内存放在 2K 池中：

```
sp_poolconfig pub_cache, "0K", "16K"
```

示例 5 将 `pubs_cache` 中 2K 池的清洗大小更改为 508K：

```
sp_poolconfig pub_cache, "2K", "wash=508K"
```

示例 6 将 2 K 池的异步预取限制更改为 15%：

```
sp_poolconfig pub_cache, "2K", "local async prefetch limit=15"
```

示例 7 在集群环境中— 在实例 `blade1` 上的缺省数据高速缓存内创建大小为 25 MB 的 16 KB 缓冲池：

```
sp_poolconfig 'default data cache', '25M', '16K', 'instance blade1'
```

示例 8 在集群环境中— 显示实例 `blade1` 上的缺省数据高速缓存中的缓冲池配置：

```
sp_poolconfig 'default data cache', 'instance blade1'
```

示例 9 在集群环境中— 显示集群中所有实例上命名高速缓存 `c_log` 的缓冲池配置：

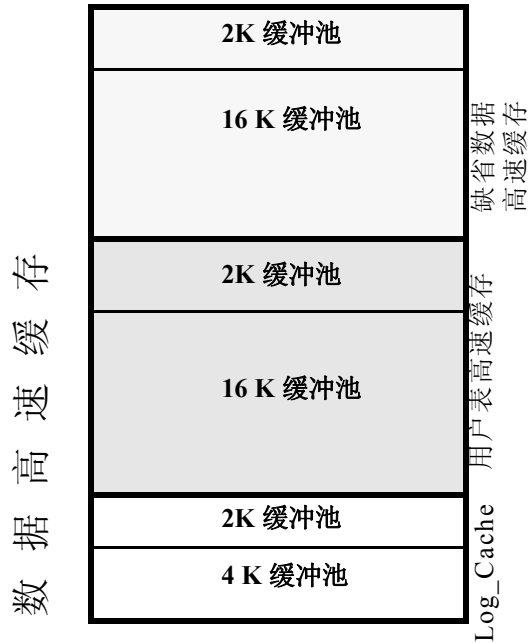
```
sp_poolconfig c_log
```

用法

- 在您使用 `sp_cacheconfig` 创建数据高速缓存时，所有空间都被分配给逻辑页大小的内存池。`sp_poolconfig` 将数据高速缓存划分为若干具有更大 I/O 大小的附加池。
- 如果高速缓存中没有大 I/O 内存池，则 `Adaptive Server` 将以逻辑页大小单位（数据页大小）对绑定到该高速缓存的所有对象执行 I/O。您常常可以通过配置执行大 I/O 的池来提高性能。16 K 内存池将对逻辑页大小为 2 K 的服务器在单个 I/O 中读写八个数据页。
- 高速缓存名称和 I/O 大小的组合必须是唯一的。换言之，在特定数据高速缓存中只能用 `sp_poolconfig` 命令指定一个具有给定 I/O 大小的池。

- 一次对于单个高速缓存只能有一个 sp_poolconfig 命令处于活动状态。如果第一个命令未完成就发出第二个 sp_poolconfig 命令，它将在第一个命令完成前处于休眠状态。
- **图 1-3** 显示服务器上的一个数据高速缓存，它将 2K 逻辑页用于以下项：
 - 缺省数据高速缓存包含一个 2 K 的缓冲池和一个 16 K 的缓冲池
 - 用户高速缓存包含一个 2 K 的缓冲池和一个 16 K 的缓冲池
 - 日志高速缓存包含一个 2 K 的缓冲池和一个 4 K 的缓冲池

图 1-3：带有缺省高速缓存和用户定义的高速缓存的数据高速缓存



- 在 2 K 页大小服务器的缺省数据高速缓存中，可以创建 I/O 大小最大为 16 K 的缓冲池。
- 内存池的最小大小是 256 个逻辑页（例如逻辑页大小为 2K 的服务器的最小大小是 512K）。您不能通过将内存转移到另一个池，来将任何高速缓存中的任何内存池的大小减少到 256 页以下。

- 两种情况可能导致池小于 512K：
 - 如果您尝试通过将池的大小设置为零来删除该池，并且其中某些页正在使用中，则 `sp_poolconfig` 将尽可能多的减少池大小，并显示警告消息。池的状态将被设置为 “Unavailable/deleted”。
 - 如果您尝试移动缓冲区以创建新池，并且无法将足够的缓冲区移入这个新池中，则 `sp_poolconfig` 将移动尽可能多的缓冲区，并且高速缓存状态被设置为 “Unavailable/too small”。

在上述两种情况中，都可以在以后重新尝试执行命令。该池在服务器重新启动后还将被删除或更改为期望的大小。

- 您可以在 Adaptive Server 处于活动状态时创建内存池，不必进行重新启动这些内存池即可生效。但是，Adaptive Server 只能移动 “可用” 缓冲区（未在使用中的缓冲区或不包含尚未写入磁盘的更改的缓冲区）。在配置池或更改其大小时，Adaptive Server 将尽可能多的内存移入该池中并输出信息性消息，显示请求的大小和该池的实际大小。在 Adaptive Server 重新启动后，所有池都以配置的大小创建。
- 某些 `dbcc` 命令和 `drop table` 只执行逻辑页大小 I/O。 `dbcc checkstorage` 可执行大 I/O， `dbcc checkdb` 可对表执行大 I/O，对索引执行逻辑页大小 I/O。
- 在为事务日志配置的 I/O 是逻辑页大小的两倍的情况下，大多数 Adaptive Server 的执行效果最佳。如果缺省高速缓存或具有绑定到它的事务日志的高速缓存是用两倍逻辑页大小的内存池配置的，则 Adaptive Server 使用两倍逻辑页大小的缺省 I/O 大小。否则，它将使用逻辑页大小的内存池。
- 您可以使用 `sp_logiosize` 系统过程增加数据库的缺省日志 I/O 大小。但是，您指定的 I/O 大小在绑定到事务日志的高速缓存中必须具有相同大小的内存池。如果不是这样，Adaptive Server 将使用逻辑页大小的内存池。

清洗百分比

- 清洗大小的缺省值的计算方式如下：
 - 如果池小于 300MB，则缺省清洗大小设置为该池中缓冲区的 20%
 - 如果池大于 300MB，则缺省清洗大小为 300MB 中缓冲区数的 20%
- 清洗大小的最小设置为 10 个缓冲区，最大设置为池大小的 80%。

- 每个内存池都在该池中缓冲区链的最近使用最少的 (LRU) 端包含一个清洗区。一旦脏页（在处于高速缓存中时已被更改的页）移入该清洗区后，Adaptive Server 就将启动对这些页的异步写入。清洗区必须足够大，以便页可在到达池的 LRU 端之前写入磁盘。在 Adaptive Server 需要等待干净缓冲区时，性能会受到影响。

缺省百分比（在清洗区中放置 20% 的缓冲区）对于大多数应用程序而言是足够的。如果您使用非常大的内存池，并且应用程序具有非常高的数据修改率，则最好将该大小增加到池的 1% 或 2%。运行 sp_sysmon 来查看建议，或与 Sybase 技术支持部门联系以获取有关选择有效清洗大小的详细信息。

本地异步预取百分比

- 池的异步预取百分比的缺省值是由配置参数 global async prefetch limit 设置的。池限制始终高于全局限制。
- 若要禁用池中的预取操作（如果全局限制是非零数值），则将池的限制设置为 0。
- 有关异步预取限制的更改对性能的影响的信息，请参见《性能和调优指南》。

权限

只有系统管理员才能执行 sp_poolconfig 以重新配置数据高速缓存内的内存池。任何用户都可以使用 sp_poolconfig 获取有关内存池的信息。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_cacheconfig](#)、[sp_helpcache](#)、[sp_logiosize](#)、[sp_unbindcache](#)、[sp_unbindcache_all](#)

sp_post_xpload

| | |
|----|--|
| 说明 | 在规模类型不同的跨平台 <code>load database</code> 后，检查并重建索引。 |
| 语法 | <code>sp_post_xpload</code> |
| 示例 | 从另一个平台装载数据库后，执行以下命令可以重建其索引： <pre>sp_post_xpload</pre> |
| 用法 | <ul style="list-style-type: none">• 以下索引在数据库中的所有用户表上重建：<ul style="list-style-type: none">• APL 表上的非聚簇索引• DOL 表上的聚簇索引• DOL 表上的非聚簇索引• 系统表上的索引并非仅使用 <code>sp_post_xpload</code> 加以处理。系统表索引在执行 <code>online database</code> 时重建。• 也可以使用 <code>drop index</code> 和 <code>create index</code> 来重建索引。• 只有在跨字节顺序类型不同的平台装载数据库时才应运行 <code>sp_post_xpload</code>。• 在索引状态可疑的情况下，可通过执行 <code>sp_xpload</code>、<code>drop index</code> 或 <code>create index</code> 来重置索引。• 执行 <code>load database</code> 之后，存储过程在初次执行时需要从 <code>syscomments</code> 中的 SQL 文本重新编译。如果没有用文本重新编译的权限，可使用 <code>dbcc upgrade_object</code> 来升级这些对象。 <p><i>在跨平台转储和装载操作中处理可疑分区</i></p> <ul style="list-style-type: none">• 在跨两个字节顺序类型不同的平台执行 <code>load database</code> 后，首次执行 <code>online database</code> 命令的过程中，散列分区被标记为可疑分区。• 对于具有使用 <code>unichar</code> 或 <code>univarchar</code> 分区键在内部生成的分区条件的循环分区，上面的所有全局聚簇索引均被标记为可疑索引。• 数据库联机后，使用 <code>sp_post_xpload</code> 可修复可疑分区和索引。 |
| 权限 | 只能由系统管理员执行。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见 dump database、load database

sp_primarykey

说明 定义表或视图的主键。

语法 `sp_primarykey tablename, col1 [, col2, col3, ..., col8]`

参数

tablename
是要定义主键的表或视图的名称。

col1
是构成主键的第一列的名称。主键可由 1 到 8 个列组成。

示例 **示例 1** 将 `au_id` 字段定义为表 `authors` 的主键：

```
sp_primarykey authors, au_id
```

示例 2 将字段 `lastname` 和 `firstname` 的组合定义为表 `employees` 的主键：

```
sp_primarykey employees, lastname, firstname
```

用法

- 执行 `sp_primarykey` 会向 `syskeys` 表添加键。只有表或视图的所有者才能定义其主键。`sp_primarykey` 不强制执行参照完整性约束；使用 `create table` 或 `alter table` 命令的 `primary key` 子句来强制执行主键关系。
- 用 `sp_primarykey`、`sp_commonkey` 和 `sp_foreignkey` 定义键，以使在数据库设计中为隐式的逻辑关系成为显式的逻辑关系。应用程序可以使用这些信息。
- 一个表或视图只能有一个主键。若要显示有关已定义键的报告，请执行 `sp_helpkey`。
- 安装进程对系统表的相应列运行 `sp_primarykey`。

权限 只有指定表或视图的所有者才能执行 `sp_primarykey`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** `alter table`、`create table`、`create trigger`

系统过程 `sp_commonkey`、`sp_dropkey`、`sp_foreignkey`、`sp_helpjoins`、`sp_helpkey`

sp_processmail

| | |
|----|--|
| 说明 | (仅限 Windows) 使用 <code>xp_findnextmsg</code> 、 <code>xp_readmail</code> 、 <code>xp_sendmail</code> 和 <code>xp_deletemail</code> 系统扩展存储过程 (ESP) 查阅、处理、发送和删除 Adaptive Server 收件箱中的邮件。 |
| 语法 | <code>sp_processmail [subject] [, originator [, dbuser [, dbname [, filetype [, separator]]]]]</code> |
| 参数 | <p>subject 是邮件主题。如果您指定了 <i>subject</i>，但未指定 <i>originator</i>，则 <code>sp_processmail</code> 将处理收件箱中具有指定主题的所有未查阅的邮件。如果您同时指定了 <i>subject</i> 和 <i>originator</i>，<code>sp_processmail</code> 将处理具有指定主题的由指定发件人发送的所有未查阅的邮件。如果您既没有指定 <i>subject</i>，也没有指定 <i>originator</i>，则 <code>sp_processmail</code> 将处理 Adaptive Server 收件箱中所有未查阅的邮件。</p> <p>originator 是传入邮件的发送者。如果您指定了 <i>originator</i>，但未指定 <i>subject</i>，则 <code>sp_processmail</code> 将处理收件箱中由指定发件人发送的所有未查阅的邮件。</p> <p>dbuser 指定用于为执行邮件中的查询提供用户环境的 Adaptive Server 登录名。缺省值为 “guest”。</p> <p>dbname 指定用于为执行邮件中的查询提供数据库环境的数据库名。缺省值为 “master”。</p> <p>filetype 指定包含查询结果的附加文件的文件扩展名。缺省扩展名为 “.txt”。</p> <p>separator 指定在查询结果中用作列分隔符的字符。它与 <code>isql</code> 的 <code>/s</code> 选项相同。缺省值为制表符。</p> |
| 示例 | <p>示例 1 处理 Adaptive Server 收件箱中由邮件用户 “janet” 递交的主题为 “SQL Report” 的所有未读邮件，以用户 “sa” 身份处理 <code>salesdb</code> 数据库中收到的查询，并在附加到邮件的 <code>.res</code> 文件中将查询结果返回给 “janet”。返回结果中的列由分号分隔：</p> <pre>sp_processmail @subject="SQL REPORT", @originator="janet", @dbuser="sa", @dbname="salesdb", @filetype="res", @separator=";"</pre> |

示例 2 以 master 数据库中的用户“sa”身份处理 Adaptive Server 收件箱中的所有未读邮件，并在附加到邮件的 .txt 文件中返回查询结果。返回结果中的列由制表符分隔：

```
sp_processmail @dbuser="sa"
```

用法

- sp_processmail 使用 [xp_findnextmsg](#)、[xp_readmail](#)、[xp_sendmail](#) 和 [xp_deletemail](#) 系统 ESP 查阅、处理、发送和删除 Adaptive Server 收件箱中的邮件。
- sp_processmail 将传出邮件发送到要处理传入邮件的发件人。
- sp_processmail 在调用 ESP 时使用缺省参数，但 [xp_sendmail](#) 的 *dbuser*、*dbname*、*attachname* 和 *separator* 参数除外，可以用 sp_processmail 的参数替换这些参数。
- sp_processmail 将所有邮件作为 Adaptive Server 查询处理。它将读取来自 Adaptive Server 收件箱中的邮件并将查询结果返回到邮件的发件人以及 Adaptive Server 邮件的附件中的所有抄送和密件抄送的收件人处。sp_processmail 为附加文件生成一个名称。该名称由以下几部分组成：开头为“syb”，后跟五位随机数字，再后面是 *filetype* 参数指定的扩展名；例如“syb84840.txt”。
- sp_processmail 在处理邮件后从收件箱中删除这些邮件。
- *subject* 和 *originator* 参数指定应处理哪些邮件。如果上述两个参数均未提供，sp_processmail 将处理 Adaptive Server 收件箱中所有未查阅的邮件。
- sp_processmail 不处理传入邮件的附件。查询必须位于传入邮件的正文中。

权限

只有系统管理员才能执行 sp_processmail。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

扩展存储过程 [xp_deletemail](#)、[xp_findnextmsg](#)、[xp_readmail](#)、[xp_sendmail](#)、[xp_startmail](#)

实用程序 isql

sp_procxmode

- 说明** 显示或更改与存储过程关联的执行模式。
- 语法** `sp_procxmode [procname [, tranmode]]`
- 参数**
- procname*
是存储过程的名称，您要检查或更改的就是该存储过程的事务模式。
- tranmode*
是存储过程的新执行模式。对于事务模式，值包括 “chained”、
“unchained” 和 “anymode”，此外还有 '[No] Dynamic Ownership Chain'。
- 示例** **示例 1** 显示当前数据库中所有存储过程的事务模式：

```
sp_procxmode

procedure name      user name      transaction mode
-----
byroyalty           dbo            Unchained
discount_proc      dbo            Unchained
history_proc       dbo            Unchained
insert_sales_proc  dbo            Unchained
insert_detail_proc dbo            Unchained
storeid_proc       dbo            Unchained
storename_proc     dbo            Unchained
title_proc         dbo            Unchained
titleid_proc       dbo            Unchained
```

示例 2 显示存储过程 `byroyalty` 的事务模式：

```
sp_procxmode byroyalty

procedure name      transaction mode
-----
byroyalty           Unchained
```

示例 3 将 `pubs2` 数据库中存储过程 `byroyalty` 的事务模式从 “unchained” 更改为 “chained”：

```
sp_procxmode byroyalty, "chained"
```

- 用法**
- 若要更改某一存储过程的事务模式，您必须是该存储过程的所有者、包含该存储过程的数据库的所有者或系统管理员。数据库所有者或系统管理员可以通过用数据库和用户名限定其他用户的存储过程的模式，对该模式进行更改。例如：

```
sp_procxmode "otherdb.otheruser.newproc", "chained"
```


- 要使用 `sp_procxmode`，请使用 `set` 命令的 `chained` 选项禁用链式事务模式。缺省情况下，该选项被禁用。
- 在使用不含参数的 `sp_procxmode` 时，它报告当前数据库中每一存储过程的事务模式。
- 若要检查存储过程的事务模式（并不更改它），请输入：

```
sp_procxmode procname
```

- 若要更改存储过程的事务模式，请输入：

```
sp_procxmode procname, tranmode
```

- 在创建存储过程时，Adaptive Server 用当前会话的事务模式标记它。这意味着：
 - 您只能在使用链式事务模式的会话中执行“链式”存储过程。
 - 您只能在使用非链式事务模式的会话中执行“非链式”存储过程。

若要在链式或非链式会话中执行特定存储过程，请将其事务模式设置为“`anymode`”。

- 如果您尝试在错误的事务模式下运行某一存储过程，则 Adaptive Server 将返回一个警告消息，但当前事务（如果有）不会受到影响。
- 执行 `sp_procxmode procname, 'Dynamic Ownership Chain'` 可确保存储过程中的任何 Dynamic SQL (`execute immediate`) 语句的权限都与过程创建者进行了核对。
- 执行 `sp_procxmode procname, 'No Dynamic Ownership Chain'`（省略时的缺省行为）可确保存储过程中的任何 Dynamic SQL (`execute immediate`) 语句的权限都与过程执行者进行了核对。

权限

只有系统管理员、数据库所有者或过程的所有者才能执行 `sp_procxmode` 来更改事务模式。任何用户都可以执行 `sp_procxmode` 来显示事务模式。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 `begin transaction`、`commit`、`save transaction`、`set`

sp_querysmobj

说明 (仅限 Tivoli Storage Manager) 在 Tivoli Storage Manager (TSM) 中查询一系列 Adaptive Server 备份对象。

注释 仅在您的站点已获得 TSM 许可时才支持 sp_querysmobj。

语法

```
sp_querysmobj "syb_tsm", "output_file", "server_name"  
              {"database_name", "object_name", "dump_type",  
               "until_time", "bs_name"}
```

参数

syb_tsm

是调用 libsyb_tsm.so 模块的关键字, 该模块用于实现与 TSM 的通信。

output_file

是 Backup Server 将 TSM 备份对象的列表写入到的文件。

server_name

是与 TSM 备份对象关联的 Adaptive Server 的名称。

database_name

是与 TSM 备份对象关联的数据库的名称。星号 (*) 表示所有数据库。

object_name

是 dump database 或 dump transaction 命令中提供的 TSM 备份对象的名称。如果省略此参数, 则会查询所有备份对象。星号 (*) 表示所有备份对象。

dump_type

是要查询的备份对象类型。有效值包括:

- DB — 通过 dump database 命令创建的数据库备份对象。
- XACT — 通过 dump transaction 命令创建的数据库备份对象。
- * — 所有数据库备份对象。这是缺省值。

until_time

是日期时间戳。将查询在指定时间前与 sp_querysmobj 中输入的条件匹配的所有备份对象。如果省略此参数, 则会查询与指定条件匹配的所有备份对象。

bs_name

是远程 Backup Server 的名称。如果省略 bs_name, 则会使用缺省的 SYB_BACKUP。

示例 1 查询 Adaptive Server “demo_srv1” 的所有 TSM 备份对象，并将列表写入到 `/tmp/qtsm/5_1.out` 中。

```
sp_querysmobj "syb_tsm", "/tmp/qtsm/5_1.out", "demo_srv1"
```

示例 2 查询 Adaptive Server “demo_srv1” 和数据库 pubs2 的所有 TSM 备份对象，并将列表写入到 `/tmp/qtsm/5_2.out` 中。

```
sp_querysmobj "syb_tsm", "/tmp/qtsm/5_2.out", "demo_srv1", "pubs2"
```

示例 3 查询 Adaptive Server “demo_srv1” 和数据库 pubs2 的所有 TSM 数据库备份对象，并将列表写入到 `/tmp/qtsm/5_3.out` 中。

```
sp_querysmobj "syb_tsm", "/tmp/qtsm/5_3.out", "demo_srv1", "pubs2", "*", "DB"
```

用法 有关 Sybase 对 TSM 的支持的详细信息，请参见《将 Backup Server 与 IBM Tivoli Storage Manager 配合使用》。

权限 只有系统管理员和具有操作员角色的用户才能执行 `sp_querysmobj`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [sp_deletesmobj](#)

sp_recompile

| | |
|----|---|
| 说明 | 导致使用指定表的各个存储过程和触发器在下次运行时重新编译。 |
| 语法 | <code>sp_recompile objname</code> |
| 参数 | <i>objname</i> 是当前数据库中表的名称。 |
| 示例 | 下次运行使用表 <code>titles</code> 的每个触发器或存储过程时，重新编译该触发器或存储过程： <pre>sp_recompile titles</pre> |
| 用法 | <ul style="list-style-type: none">对于存储过程和触发器使用的查询，仅在对存储过程和触发器进行编译时优化一次。当 <code>systabstats</code> 统计信息（例如行计数或集群比）大幅变化时，已编译的存储过程和触发器可能不再高效，通过 <code>sp_recompile</code> 重新编译，其效率可能会有所改观。通过重新编译对表进行操作的存储过程和触发器可以优化查询以获得最高效率。 <hr/> <p>注释 <code>create index</code> 和 <code>update statistics</code> 会造成较小的模式更改，这会 自动重新编译存储过程和触发器。在此类情况下使用 <code>sp_recompile</code> 会导致多余的重新编译。</p> <hr/> <ul style="list-style-type: none"><code>sp_recompile</code> 只在当前数据库中查找 <i>objname</i>，而且只对当前数据库中的触发器和存储过程进行重新编译。<code>sp_recompile</code> 不会影响其它数据库中依赖于该表的对象。不能对系统表使用 <code>sp_recompile</code>。在 Adaptive Server 版本 12.5 和之前的版本中，<code>sp_recompile</code> 会影响您执行的即席查询。Adaptive Server 会返回一个模式更改错误（错误号 540）并中止该即席查询。<code>sp_recompile</code> 现在不再影响这样的即席查询，您也不会再看到错误 540。 <hr/> <p>注释 <code>sp_recompile</code> 仍会影响在运行 <code>sp_recompile</code> 之前就开始执行的即席查询（并发执行）。</p> <hr/> |
| 权限 | 只有数据库所有者和系统管理员才能使用 <code>setuser</code> 命令以另一个数据库用户的身份重命名归其他用户所有的对象。任何用户都可以执行 <code>sp_recompile</code> 来重新编译他们自己的对象。 |

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create index](#)、[update statistics](#)

sp_refit_admin

| | |
|----|---|
| 说明 | (仅限集群环境) 提供接口以执行各种与 disk refit 相关的操作, 例如显示 disk refit 进程的当前状态、重置 disk refit 进程的状态、跳过实例的 disk refit 进程等: |
| 语法 | <pre>sp_refit_admin ['help'] 'status' ['reset' 'skiprefit' [, <i>instance_name</i>]] ['removedevice', <i>device_name</i>]</pre> |
| 参数 | <p>help</p> <p>显示有关 sp_refit_admin 语法和用法的信息。</p> <p>status</p> <p>显示 disk refit 进程的当前状态。它会列出所有实例及其 disk refit 仍处于挂起状态的私有设备。如果不存在此类设备, 它会输出一条消息报告此情况。</p> <p>reset</p> <p>重置 disk refit 进程的状态。它带有一个可选参数 <i>instance_name</i>。</p> <p>如果不提供 <i>instance_name</i>, 此参数会将 disk refit 进程重置回第一阶段开始时的状态, 以便后续的 disk refit 命令从第一阶段开始 disk refit 进程, 并重新调整所有的常规可共享设备, 以及相应实例的私有设备。</p> <p>如果不提供 <i>instance_name</i>, 此参数会将 disk refit 进程重置回该实例第二阶段开始时的状态, 以便该实例上后续的 disk refit 命令从该实例的第二阶段开始 disk refit 进程, 并且仅重新调整该实例的私有设备。</p> <p>skiprefit</p> <p>针对集群中的一个或全部实例跳过 disk refit 进程第二阶段的运行, 而不删除该设备。此参数仅在完成 disk refit 进程第一阶段后才有意义。它带有一个可选参数 <i>instance_name</i>。</p> <p>removedevice</p> <p>从 disk refit 进程中删除设备。此参数需要待删除设备的名称, 将其作为输入参数 <i>device_name</i> 或 <i>instance_name</i>。</p> |
| 示例 | <p>示例 1 将 disk refit 进程的状态重置为第一阶段开始时的状态:</p> <pre>sp_refit_admin 'reset'</pre> <p>执行 reset 后, 用户必须运行 disk refit 进程的第一阶段和第二阶段。</p> <p>示例 2 将名为 'cluster1_instance1' 的实例上 disk refit 进程的状态重置为该实例第二阶段开始时的状态:</p> <pre>sp_refit_admin 'reset', 'cluster1_instance1'</pre> |

此接口删除在 'cluster1_instance1' 拥有的私有设备上创建的所有数据库的 `sysdatabases` 条目，以及与 'cluster1_instance1' 拥有的私有设备对应的 `sysusages` 条目。执行后，必须在 'cluster1_instance1' 上运行 `disk refit` 的第二阶段。

示例 3 跳过实例 'cluster1_instance1' 的所有待重新调整私有设备的 `disk refit` 进程：

```
sp_refit_admin 'skiprefit', 'cluster1_instance1'
```

此示例删除使用由 'cluster1_instance1' 拥有的任何待重新调整私有设备的所有数据库对应的 `sysdatabases` 条目，并删除 `sysusages` 中与所有已删除的数据库对应的条目。

要跳过集群中所有实例的所有待重新调整私有设备上的 `disk refit` 进程，请输入：

```
sp_refit_admin 'skiprefit'
```

示例 4 从 `disk refit` 进程中删除设备 'device1'：

```
sp_refit_admin 'removedevice', 'device1'
```

此操作会删除在 'device1' 上创建的所有数据库的 `sysdatabases` 条目，以及与 'device1' 对应的所有 `sysusages` 条目。它还会从 `sysdevices` 中删除 'device1'。

用法

- 执行 `skiprefit` 后您必须按照《集群用户指南》中第 12 章“故障排除”中的说明操作，以确保在恢复正常工作前系统表保持一致。
- 仅在 `disk refit` 进程执行期间使用 `removedevice`，以从重新调整进程中删除设备。请勿使用它来代替 `sp_dropdevice`
- 即使实例启动时使用的是 `-m` 选项且跟踪标志 3608 设置为 ON，也可以使用 `sp_refit_admin`。

权限

只有具有系统管理员权限的用户才能执行 `sp_refit_admin`。

另请参见

有关使用 `disk refit` 时遇到的问题的信息，请参见《故障排除和错误指南》。

sp_remap

说明 重新映射来自版本 4.8 之后、版本 10.0 之前的数据库的存储过程、触发器、规则、缺省值或视图，使其与版本 10.0 和更高版本兼容。对于升级过程未能重新映射的原有对象，请使用 `sp_remap`。

语法 `sp_remap objname`

参数 `objname`
是当前数据库中某个存储过程、触发器、规则、缺省值或视图的名称。

示例 1 重新映射名为 `myproc` 的存储过程：

```
sp_remap myproc
```

示例 2 重新映射名为 `default_date` 的规则。请在运行该过程之前执行 `use my_db` 语句打开 `my_db` 数据库：

```
sp_remap "my_db..default_date"
```

用法

- 如果 `sp_remap` 未能重新映射某一对象，请从数据库中删除该对象，然后再重新创建它。在对某个对象运行 `sp_remap` 之前，最好使用 `defncopy` 实用程序将其定义复制到一个操作系统文件中。有关 `defncopy` 的详细信息，请参见《实用程序指南》。
- `sp_remap` 可能会导致事务日志很快填满。运行 `sp_remap` 之前，请使用 `dump transaction` 命令转储事务日志（如果需要）。
- 只能对当前数据库中的对象使用 `sp_remap`。
- `sp_remap` 不会对已成功升级到当前版本的对象进行任何更改。

权限 只有系统管理员或对象所有者才能执行 `sp_remap`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** [dump transaction](#)

系统过程 [sp_helptext](#)

实用程序 [defncopy](#)

sp_remotoption

说明 显示或更改远程登录选项。

语法 sp_remotoption [*remoteserver*[, *loginame*
[, *remotename*[, *optname*[, *optvalue*]]]]]

参数 *remoteserver*

是将在该服务器上执行 RPC 的服务器的名称。

注释 本手册页使用术语“本地服务器”来表示正在执行从“远程服务器”运行的远程过程的服务器。

loginame

是标识 *remoteserver*、*loginame* 和 *remotename* 组合的本地登录身份的登录名。

remotename

是标识 *remoteserver*、*loginame* 和 *remotename* 组合的远程登录身份的远程用户名。

optname

是要更改的选项的名称。目前只有一个选项，即 **trusted**。该选项表示本地服务器将直接接受来自其它服务器的远程登录，而不会针对特定的远程登录名进行用户访问权验证。缺省情况下使用口令验证。

Adaptive Server 可以识别属于选项名一部分的任何唯一字符串。如果选项名内嵌有空格，则要将该选项名用引号引起来。

optvalue

要么为 **true**，要么为 **false**。为 **true** 时将启用该选项，为 **false** 时将禁用该选项。

示例

示例 1 显示远程登录选项的列表：

```
sp_remotoption
Settable remote login options.
remotelogin_option
-----
受托
```

示例 2 将来自远程服务器 GATEWAY 的远程登录名定义为 **trusted**；也就是说，不检查口令：

```
sp_remotoption GATEWAY, churchy, pogo, trusted, true
```

示例 3 将来自远程服务器 GATEWAY 的远程登录名 “pogo” 定义为非可信登录名；也就是说，将检查口令：

```
sp_remoteoption GATEWAY, churchy, pogo, trusted, false
```

示例 4 将来自 GATEWAY 并映射到本地服务器上的登录名 “albert” 的所有登录名定义为可信登录名：

```
sp_remoteoption GATEWAY, albert, NULL, trusted, true
```

用法

- 若要显示远程登录选项的列表，请执行不带任何参数的 `sp_remoteoption`。
- 如果已使用 `sp_addremotelogin` 将来自远程服务器的所有用户都映射到了同一个本地名称，请为那些用户指定 `trusted`。例如，如果来自远程服务器 GOODSRV 的所有映射到 “albert” 的用户都是可信用户，请指定：

```
sp_remoteoption GOODSRV, albert, NULL, trusted, true
```

如果未将登录名指定为 `trusted`，他们将不能在本地服务器上执行 RPC，除非他们在登录到远程服务器时指定了本地服务器口令。在 Open Client™ Client-Library™ 中，用户可以使用 `ct_remote_pwd` 例程为服务器到服务器的连接指定口令。isql 和 bcp 不允许用户为 RPC 连接指定口令。

如果用户使用 “统一登录” 登录到远程服务器，那么相应登录名在本地服务器上也必须可信的，否则用户在登录到远程服务器时必须指定服务器口令。

有关为远程过程调用和为使用 “统一登录” 而设置服务器的详细信息，请参见《系统管理指南》。

权限

只有系统安全员才能执行 `sp_remoteoption`。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 `sp_addremotelogin`、`sp_droptremotelogin`、`sp_helpremotelogin`

实用程序 isql

sp_remotesql

说明 (仅限组件集成服务) 建立与远程服务器的连接, 将查询缓冲区从客户端传递到远程服务器, 然后再将结果传递回客户端。

语法 `sp_remotesql server, query [, query2, ... , query254]`

参数

server

是使用 `sp_addserver` 定义的远程服务器的名称。

query

是查询缓冲区, 其最大长度为 255 个字符。

query2 query254

是查询缓存区, 其最大长度为 255 个字符。如果提供了这些参数, 将把它们与 `query1` 的内容并置为一个查询缓冲区。

示例

示例 1 将查询缓冲区传递到 `FREDS_SERVER`, 该服务器将解释 `select @@version` 并将结果返回到客户端。Adaptive Server 不解释结果:

```
sp_remotesql FREDS_SERVER, "select @@version"
```

示例 2 说明如何在存储过程中使用 `sp_remotesql`。此示例和示例 1 向客户端返回相同的信息:

```
create procedure freds_version
as
exec sp_remotesql FREDS_SERVER, "select @@version"
go
exec freds_version
go
```

示例 3 服务器将两个查询缓冲区并置为一个缓冲区, 并将完整的 `insert` 语句传递到 `DCO_SERVER` 以进行处理。`insert` 语句的语法是 `DCO_SERVER` 能够理解的格式。此服务器不解释返回的信息。此示例还检查 `@@error` 中返回的值。

```
sp_remotesql DCO_SERVER,
"insert into remote_table
(numbercol,intcol, floatcol,datecol )",
"values (109.26,75, 100E5,'10-AUG-85')"
```

select @@error

示例 4 说明如何将本地变量用作 `sp_remotesql` 的参数:

```
declare @servname varchar(30)
declare @querybuf varchar(200)
select @servname = "DCO_SERV"
select @querybuf = "select table_name
from all_tables"
```

```

        where owner = 'SYS'"
exec sp_remotesql @servname, @querybuf
    
```

用法

- **sp_remotesql** 建立与远程服务器的连接，将查询缓冲区从客户端传递到远程服务器，然后再将结果传递回客户端。本地服务器不解释结果。
- 可在另一个存储过程内使用 **sp_remotesql**。
- 查询缓冲区参数必须是字符表达式，其最大长度为 255 个字符。如果使用类型不是 **char** 或 **varchar** 的查询缓冲区，将出现数据类型转换错误。
- 如果从远程服务器返回的最后一条错误消息的严重级别大于 10，**sp_remotesql** 会将全局变量 **@@error** 设置为该消息的值。
- 如果在事务内发出 **sp_remotesql**，Adaptive Server 会首先检验远程服务器上是否已启动了事务，然后再传递要执行的查询缓冲区。当该事务终止时，系统将指示远程服务器提交该事务。查询缓冲区的内容执行的工作是由该事务定义的工作单元的一部分。

如果查询缓冲区中包含事务控制语句，将由客户端负责确保事务按预期执行 **commit** 和 **rollback**。在查询缓冲区中混用 **Transact-SQL** 和事务控制命令可能会导致不可预料的结果。

- 与远程服务器的连接由本地服务器管理。将 **connect to** 或 **disconnect** 命令嵌入到查询缓冲区中会导致需要由远程服务器解释的结果。这没有必要，我们也不建议这样做。通常，结果是语法错误。

权限

任何用户都可以执行 **sp_remotesql**。

审计

sysaudits 表的 **event** 和 **extrainfo** 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [connect to...disconnect](#)

系统过程 [sp_addserver](#)、[sp_autoconnect](#)、[sp_passthru](#)

sp_rename

| | |
|----|---|
| 说明 | 更改当前数据库中用户创建的对象或用户定义的数据类型的名称。 |
| 语法 | <code>sp_rename objname, newname [,"index" "column"]</code> |
| 参数 | <p>objname 是用户创建的对象（表、视图、列、存储过程、索引、触发器、缺省值、规则、检查约束、参照约束或用户定义的数据类型）的原始名称。如果要重命名的对象是表中的某一列，objname 的格式必须为“<i>table.column</i>”。如果该对象是索引，objname 的格式必须为“<i>table.indexname</i>”。</p> <p>newname 是该对象或数据类型的新名称。该名称必须符合标识符的规则，而且在当前数据库中必须唯一。</p> <p>index 指定要重命名的对象是索引，而不是列。使用此参数可以重命名与列具有相同名称的索引，而无需删除并重新创建该索引。</p> <p>column 指定要重命名的对象是列，而不是索引。此参数与 <i>index</i> 参数是同一选项中的参数。</p> |
| 示例 | <p>示例 1 将 <code>titles</code> 表重命名为 <code>books</code>：</p> <pre>sp_rename titles, books</pre> <p>示例 2 将 <code>books</code> 表中的 <code>title</code> 列重命名为 <code>bookname</code>：</p> <pre>sp_rename "books.title", bookname</pre> <p>示例 3 将 <code>books</code> 表中的 <code>titleind</code> 索引重命名为 <code>titleindex</code>：</p> <pre>sp_rename "books.titleind", titleindex</pre> <p>示例 4 将用户定义的数据类型 <code>tid</code> 重命名为 <code>bookid</code>：</p> <pre>sp_rename tid, bookid</pre> <p>示例 5 将 <code>titles</code> 表中的 <code>title_id</code> 索引重命名为 <code>isbn</code>：</p> <pre>sp_rename "titles.title_id", isbn, "index"</pre> |
| 用法 | <ul style="list-style-type: none"><code>sp_rename</code> 用于更改用户创建的对象或数据类型的名称。在哪个数据库中发出 <code>sp_rename</code>，就只能更改哪个数据库中的对象或数据类型的名称。重命名列或索引时，不要在 <i>newname</i> 中指定表名。请参见示例 2、3 和 5。 |

- 如果列与索引具有相同的名称，请使用 [“index” | “column”] 参数，该参数指定是否重命名索引或列。在下面的示例中，假定名称为 idx 的索引和列都存在：

```

sp_rename "t.idx", new_idx, "column"
-----
Column name has been changed. (Return status = 0)
sp_rename "t.idx", new_idx, "index"
-----
Index name has been changed. (Return status = 0)

```

- 如果更改视图引用的对象名或列名，将显示一条警告消息，例如：
更改对象或列名称可能破坏
现有存储过程、高速缓存的语句或
其它已编译的对象。
- `sp_engine` 可以在没有打开的事务时在使用链式事务模式的会话中运行：
- 不能更改系统对象和系统数据类型的名称。

警告！ 依赖于名称发生了更改的对象的过程、触发器和视图将不再有效。请在执行 `sp_rename` 前更改所有依赖对象的定义。使用 [sp_depends](#) 可以找到相关对象。

权限 只有数据库所有者和系统管理员才能使用 `setuser` 命令以另一个数据库用户的身份重命名归其他用户所有的对象。任何用户都可以执行 `sp_rename` 来重命名他们自己的对象。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_depends](#)、[sp_rename](#)

sp_rename_qpgroup

说明 重命名抽象计划组。

语法 `sp_rename_qpgroup old_name, new_name`

参数 *old_name*
是抽象计划组的当前名称。

new_name
是该组的新名称。指定的 *new_name* 不能是数据库中现有抽象计划组的名称。

示例 `sp_rename_qpgroup dev_plans, prod_plans`

将该组的名称从 `dev_plans` 更改为 `prod_plans`。

用法

- 使用 `sp_rename_qpgroup` 可以重命名抽象计划组。不能使用现有计划组的名称作为新名称。
- `sp_rename_qpgroup` 不会影响被重命名的组的内容。现有抽象计划的 ID 不会更改。
- 不能重命名缺省抽象计划组 `ap_stdin` 和 `ap_stdout`。
- 不能在事务中运行 `sp_rename_qpgroup`。

权限 只有系统管理员或数据库所有者才能执行 `sp_rename_qpgroup`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_help_qpgroup](#)

sp_renamedb

| | |
|----|---|
| 说明 | 更改用户数据库的名称。 |
| 语法 | <code>sp_renamedb dbname, newname</code> |
| 参数 | dbname 是数据库的原始名称。 newname 是数据库的新名称。数据库名称必须符合标识符的规则，而且必须唯一。 |

示例 1 将 `accounting` 数据库重命名为 `financial`:

```
sp_renamedb accounting, financial
```

示例 2 将名为 `work`（Transact-SQL 保留字）的数据库重命名为 `workdb`。下面的示例显示如何在重命名 `work` 数据库之前使用 `sp_dboption` 将该数据库置于单用户模式，并在重命名之后将其恢复到多用户模式:

```
sp_dboption work, single, true
go
use work
go
checkpoint
go
sp_renamedb work, workdb
go
use master
go
sp_dboption workdb, single, false
go
use workdb
go
checkpoint
go
```

- 用法**
- `sp_renamedb` 更改数据库的名称。不能重命名系统数据库或具有外部参照完整性约束的数据库。
 - 系统管理员必须在重命名数据库之前使用 `sp_dboption` 将其切换到单用户模式，并在重命名之后将其恢复到多用户模式。
 - 如果数据库中有任何表引用另一个数据库中的表（或被引用），`sp_renamedb` 将失败。使用下面的查询可确定哪些表和外部数据库对当前数据库中的主键表具有外键约束：

```
select object_name(tableid), db_name(frgndbid)
from sysreferences
```



```
where frgnidbid is not null
```

使用下面的查询可确定哪些表和外部数据库对当前数据库中的外键表具有主键约束：

```
select object_name(reftabid), db_name(pmrydbid)
from sysreferences
where pmrydbid is not null
```

使用 `alter table` 删除这些表中的跨数据库约束。然后，重新运行 `sp_renamedb`。

- 更改了数据库名称后：
 - 删除所有包含该数据库名称的存储过程、触发器和视图
 - 更改被删除对象的源文本以反映新数据库名称
 - 重新创建删除的对象
 - 更改所有参照该数据库的应用程序和 SQL 源脚本：或者使用 `use database_name` 命令，或者作为全限定标识符的一部分（格式为 `dbname.[owner].objectname`）
- 如果您使用脚本对数据库运行 `dbcc` 命令或 `dump database` 和 `dump transaction` 命令，一定要更新这些脚本。

警告！ 依赖于名称已更改的数据库的过程、触发器和视图可以工作到重新创建它们为止。请在执行 `sp_renamedb` 后更改所有相关对象的定义。使用 `sp_depends` 可以找到相关对象。

权限 只有系统管理员才能执行 `sp_renamedb`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** `create database`

系统过程 `sp_changedbowner`、`sp_dboption`、`sp_depends`、`sp_helpdb`、`sp_rename`

sp_reportstats

说明 报告有关系统使用情况的统计信息。

语法 sp_reportstats [*loginame*]

参数 *loginame*
是要显示其会计汇总的用户的登录名。

示例 **示例 1** 显示所有 Adaptive Server 用户的当前会计汇总的报告：

```
sp_reportstats
```

| Name | Since | CPU | Percent CPU | I/O | Percent I/O |
|-----------|-------------|-----------|-------------|------|-------------|
| julie | jun 19 1993 | 10000 | 24.9962% | 5000 | 24.325% |
| jason | jun 19 1993 | 10002 | 25.0013% | 5321 | 25.8866% |
| ken | jun 19 1993 | 10001 | 24.9987% | 5123 | 24.9234% |
| kathy | jun 19 1993 | 10003 | 25.0038% | 5111 | 24.865% |
| Total CPU | | Total I/O | | | |
| 40006 | | 20555 | | | |

示例 2 显示用户 “kathy” 的当前会计汇总的报告：

```
sp_reportstats kathy
```

| Name | Since | CPU | Percent CPU | I/O | Percent I/O |
|-----------|-------------|-----------|-------------|-------|-------------|
| kathy | Jul 24 1993 | 498 | 49.8998% | 48392 | 9.1829% |
| Total CPU | | Total I/O | | | |
| 998 | | 98392 | | | |

用法

- sp_reportstats 输出所有登录名的当前会计汇总，以及各个登录名的单独统计信息和占总统计信息的百分比。sp_reportstats 接受一个参数：要报告其统计信息的帐户的登录名。如果不指定任何参数，sp_reportstats 将报告所有帐号的统计信息。
- 报告中 “CPU” 的单位为 Adaptive Server 时钟周期。
- “probe” 用户用于两阶段提交探查进程，这种进程使用 “询问和响应” 机制来访问 Adaptive Server。

权限 只有系统管理员才能执行 sp_reportstats。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_clearstats](#)、[sp_configure](#)

sp_revokelogin

说明 (仅限 Windows NT) 当集成安全模式或混合模式 (具有命名管道) 处于活动状态时, 撤消 Windows NT 用户和组的 Adaptive Server 角色和缺省权限。

语法 sp_revokelogin {login_name | group_name}

参数 *login_name*
是 Windows NT 用户的网络登录名。

group_name
是 Windows NT 组名。

示例 **示例 1** 撤消名为 “jeanluc” 的 Windows NT 用户的所有权限:

```
sp_revokelogin jeanluc
```

示例 2 撤消 Windows NT “Administrators” 组的所有角色:

```
sp_revokelogin Administrators
```

用法

- 只有当 Adaptive Server 在 “集成安全” 模式或 “混合” 模式 (连接为 “命名管道”) 下运行时, 才应使用 sp_revokelogin。如果 Adaptive Server 在 “标准” 模式下运行, 或在使用除 “命名管道” 之外的连接的 “混合” 模式下运行, 应使用 revoke 命令。
- 如果您使用 sp_revokelogin 撤消了某个用户的角色和缺省特权, 该用户将无法再通过受托连接登录到 Adaptive Server。

权限 只有系统管理员才能执行 sp_revokelogin。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **命令** grant、revoke、setuser

系统过程 sp_droplogin、sp_dropuser、sp_logininfo

sp_role

| | |
|----|--|
| 说明 | 授予 Adaptive Server 登录帐户角色，或撤消其角色。 |
| 语法 | <code>sp_role {"grant" "revoke"}, rolename, loginame</code> |
| 参数 | <p><code>grant</code> <code>revoke</code></p> <p>指定是要授予 <i>loginame</i> 角色，还是要撤消其角色。</p> <p><i>rolename</i></p> <p>是要授予或撤消的角色。</p> <p><i>loginame</i></p> <p>是作为角色授予或撤消对象的登录帐户。</p> |
| 示例 | <p>将系统管理员角色授予名为 “alexander” 的登录帐户：</p> <pre>sp_role "grant", sa_role, alexander</pre> |
| 用法 | <ul style="list-style-type: none"> • <code>sp_role</code> 将角色授予 Adaptive Server 登录帐号或撤消其角色。 • 将角色授予某个用户后，它将在该用户下次登录到 Adaptive Server 时生效。或者，该用户可以通过使用 <code>set role</code> 命令使该角色立即生效。例如，以下命令为用户启用系统管理员角色： <pre>set role sa_role on</pre> <p>但是，必须运行 <code>sp_modifylogin</code> 才能启用登录。有关详细信息，请参见 sp_modifylogin。</p> • 不能在用户已登录的情况下撤消此用户的角色。 • 用户登录时，授予用户的所有系统定义角色都处于活动状态 (on)。若要关闭某个角色，请使用 <code>set</code> 命令。例如，若要停用系统管理员角色，请使用以下命令： <pre>set role "sa_role" off</pre> |
| 权限 | 只有系统管理员才能执行 <code>sp_role</code> 将系统管理员角色授予其他用户。只有系统安全员才能通过执行 <code>sp_role</code> 将除 “sa” 之外的任何角色授予其他用户。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [grant](#)、[revoke](#)、[set](#)

函数 [proc_role](#)、[show_role](#)

系统过程 [sp_activeroles](#)、[sp_displayroles](#)、[sp_displayroles](#)

sp_securityprofile

| | |
|----|---|
| 说明 | 列出与登录配置文件关联的属性或绑定。 |
| 语法 | <pre>sp_securityprofile 'attributes','login profile', {wildcard login_profile_name 'default'} sp_securityprofile 'bindings', 'login profile' [, {wildcard login_profile_name 'default'} [, 'login' ,{wildcard login_name}]] sp_securityprofile 'help'</pre> |
| 参数 | <p>attributes 指定它可列出与登录配置文件关联的属性。</p> <p>login profile 指定它可获取有关登录配置文件的信息。</p> <p>bindings 如果指定了 login，则可列出登录帐户的绑定。如果指定了 login profile，则可列出登录配置文件的绑定。</p> <p>login 指定它可获取有关登录帐户的信息。</p> <p>wildcard login_profile_name default 指定要从中获取信息的登录配置文件。选项包括登录配置文件的特定名称、缺省登录配置文件，也可以使用通配符来标识登录配置文件。</p> <p>wildcard login_name 指定它可使用特定登录帐户名或允许使用通配符标识登录帐户。</p> <p>help 显示用法。</p> |

示例 1 列出缺省登录配置文件的所有属性。

```
sp_securityprofile 'attributes', 'login profile',
    'default'
```

| Name | Value |
|--------------------------|-------------------|
| login profile | def_login_profile |
| default | yes |
| default database | master |
| default language | NULL |
| login script | NULL |
| auto activated roles | emp_role |
| auto activated roles | def_role |
| manually activated roles | special_role |

```

authenticate with      ANY
track lastlogin       TRUE
stale period          180D

```

示例 2 显示与所有登录配置文件关联的所有属性。

```
sp_securityprofile 'attributes', 'login profile', '%'
```

```

Name                    Value
-----
login profile           def_login_profile
default                 yes
default database       master
default language       NULL
login script            NULL
auto activated roles   emp_role
auto activated roles   def_role
authenticate with      ANY
track lastlogin       TRUE
stale period          180D

```

```

Name                    Value
-----
login profile           eng_login_profile
default                 work
default database       engr_script
login script            emp_role
auto activated roles   def_role
auto activated roles   engr_role
authenticate with      LDAP

```

```

Name                    Value
-----
login profile           mgr_login_profile
default                 work
default database       mgr_script
login script            emp_role
auto activated roles   def_role
auto activated roles   mgr_role
manually activated roles activate_emp_role
authenticate with      LDAP

```

```

Name                    Value
-----
login profile           sa_login_profile

```



```
manually activated roles      admin_role
default
```

示例 3 显示与特定登录配置文件关联的所有登录帐户。

```
sp_securityprofile 'bindings', 'login profile',
'engr_login_profile'
```

```
Login name          Login profile name
-----
anderson            eng_login_profile
gupta               eng_login_profile
lchang              eng_login_profile
tsato               eng_login_profile
```

示例 4 显示名为 sa 的登录帐户的登录配置文件：

```
sp_securityprofile 'bindings', 'login profile', null,
'login', 'sa'
```

```
Login name          Login profile name
-----
sa                  sa_login_profile
```

用法
权限

- 对于配置文件中未设置的属性，遵循优先规则。有关详细信息，必须具有 `sso_role` 才能查看所有登录配置文件的属性和绑定。

非特权登录帐户：

- 只能查看与登录名关联的登录配置文件（直接或缺省登录配置文件）的属性。
- 不能查看登录配置文件与登录帐户的绑定。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

命令 creat login profile、alter login profile

文档 《安全性管理指南》中的“Applying login profile and password policy attributes”（应用登录配置文件和口令策略属性）

系统过程 sp_displaylogin

sp_sendmsg

说明 将消息发送到用户数据报协议 (UDP) 端口。

语法 `sp_sendmsg ip_address, port_number, message`

参数

ip_address
是运行着 UDP 应用程序的计算机的 IP 地址。

port_number
是 UDP 端口的端口号。

message
是要发送的消息，长度不超过 4096 个字符。

示例 `sp_sendmsg "120.10.20.5", 3456, "Hello World"`

用法

- Windows NT 不支持 `sp_sendmsg`。
- 系统安全员必须将配置参数 `allow sendmsg` 设置为 1，才能启用 UDP 消息传送功能。
- 对 `sp_sendmsg` 不执行任何安全检查。Sybase 强烈建议当使用 `sp_sendmsg` 在网络上发送敏感信息时一定要谨慎。用户启用此功能，即表示其已接受因使用此功能而导致的任何安全问题。
- 这个示例 C 程序监听指定的端口，并输出它接收到的消息。例如，若要接收示例 1 中调用 `sp_sendmsg` 发送的消息，请使用：

```

updmon 3456
#include <stdlib.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <fcntl.h>

main(argc, argv)
int argc; char *argv[];
{
    struct sockaddr_in saddr;
    int portnum, sck, dummy, msglen;
    char msg[256];

    if (argc < 2) {
        printf( "Usage: updmon <udp portnum>\n" );
        exit(1);
    }

```

```

    }

    if ((portnum=atoi(argv[1])) < 1) {
        printf("Invalid udp portnum\n");
        exit(1);
    }

    if ((sck=socket(AF_INET,SOCK_DGRAM,IPPROTO_UDP)) < 0) {
        printf("Couldn't create socket\n");
        exit(1);
    }

    sadr.sin_family = AF_INET;
    sadr.sin_addr.s_addr = inet_addr("0.0.0.0");
    sadr.sin_port = portnum;

    if (bind(sck,&sadr,sizeof(sadr)) < 0) {
        printf("Couldn't bind requested udp port\n");
        exit(1);
    }

    for (;;)
    {
        if ((msglen=recvfrom(sck,msg,sizeof(msg),0,NULL,&dummy))
        < 0)
            printf("Couldn't recvfrom() from udp port\n");
        printf("%.s\n", msglen, msg);
    }
}

```

权限 任何用户都可以执行 sp_sendmsg。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **快捷键** [syb_sendmsg](#)

sp_serveroption

说明 显示或更改远程服务器选项。

语法 sp_serveroption [server, optname, optvalue]

参数 server

是要设置选项的远程服务器的名称。

optname

是要设置或取消设置的选项的名称。表 1-30 列出了选项名称。

表 1-30: sp_serveroption 选项

| 选项 | 含义 |
|------------------------------|--|
| mutual authentication | 使用 Kerberos 鉴定为所有与远程服务器的连接设置相互鉴定。 |
| external engine auto start | 指定 EJB Server 在每次 Adaptive Server 启动时启动。缺省值为 true；启动 Adaptive Server 时也同时启动 EJB Server。 |
| net password encryption | 指定是使用客户端口令加密握手，还是使用常规（未加密口令）握手序列启动与远程服务器的连接。缺省值为 false，无网络加密。 |
| net password encryption reqd | <p>Adaptive Server 允许使用非对称加密，以采用 RSA 公用密钥加密算法将口令从客户端安全地传输到服务器。Adaptive Server 生成非对称密钥对，并将公用密钥发送给使用新的登录协议的客户端。客户端在将用户登录口令发给服务器之前，会使用公用密钥对其进行加密。服务器则使用私有密钥将口令解密，以开始对进行连接的客户端进行鉴定。</p> <p>可将 Adaptive Server 配置为要求客户端使用此协议。将 Adaptive Server 配置参数 net password encryption reqd 设置为要求所有基于用户名和口令的鉴定请求使用 RSA 非对称加密。net password encryption reqd 的有效值包括：</p> <ul style="list-style-type: none"> • 0 — 允许客户端选择在网络中用于登录口令的加密算法，包括无口令加密。这是此配置参数的缺省值，提供的功能与早期版本基本类似。此值允许选择要由客户端应用程序建立的网络口令加密。 • 1 — 将客户端限制为使用 RSA 或 Sybase 专有加密算法在网络中对登录口令进行加密。此值提供限制性更强的设置，允许旧的客户端使用 Sybase 专有算法连接，新的客户端使用更强大的 RSA 算法连接。客户端尝试不使用口令加密进行连接时会失败。 • 2 — 将客户端限制为仅使用 RSA 加密算法在网络中对登录口令进行加密。此值提供强大的 RSA 口令加密，要求使用更新的客户端。客户端尝试不使用 RSA 加密进行连接时会失败。 |
| allow password downgrade | |
| readonly | （仅适用于组件集成服务）指定对指定服务器的访问为只读访问。 |
| security mechanism | 此选项指定远程服务器的安全机制。如果您的登录是使用 Kerberos 机制进行鉴定的，请对与远程服务器的连接启用 Kerberos 鉴定。 |
| server cost | （仅适用于组件集成服务）指定用户控制下一个交换的成本（针对每个服务器）。有关详细信息，请参见《了解 CIS》中的第 2 章“了解组件集成服务”。 |

| 选项 | 含义 |
|-----------------------------|--|
| server logins | <p>(仅适用于组件集成服务) 为完全支持远程登录, ClientLibrary 提供了一些连接属性, 使 CIS 能够请求服务器连接。此连接在接收服务器上被识别为服务器连接 (而不是普通客户端连接), 从而使远程服务器能够通过使用 <code>sysremotelogins</code> 来检验连接, 就像该连接是由站点处理器建立的一样。</p> <p>启用时, Omni 将使用 <code>CS_LOGIN_TYPE</code> 连接属性连接到指定的服务器, 并将类型设置为 <code>LREMUSER</code>。而且, 如果远程服务器是 Adaptive Server, <code>CS_LOGIN_REMOTE_SERVER</code> 属性将设置为本地服务器名称的值, 并使用 <code>ct_remote_pwd()</code> 设置远程口令。</p> |
| server principal | 设置远程服务器的服务器主体名。 |
| negotiated logins | <p>(仅用于组件集成服务) 如果需要到 XP Server 或 Backup Server 的 CIS 连接, 则必须使用此选项。</p> <p>启用时, Omni 将使用 <code>CS_SEC_CHALLENGE</code> 属性连接到指定的服务器, 并建立可对来自 XP Server 和 Backup Server 的登录询问做出响应的回调处理程序。</p> |
| timeouts | 取消设置时 (<code>false</code>), 将禁用本地服务器使用的常规超时代码, 这样站点连接处理器就不会在一分钟没有逻辑连接后自动删除该物理连接了。缺省值为 <code>false</code> 。 |
| use message confidentiality | 为所有使用 Kerberos 鉴定与远程服务器建立的连接设置消息保密性。 |
| use message integrity | 为所有使用 Kerberos 鉴定与远程服务器建立的连接设置消息完整性。 |
| cis hafailover | (仅用于组件集成服务) 如果启用, 则指示 Open Client 在连接失败时使用自动故障切换。在此情况下, CIS 连接失败会自动故障切换到在目录服务中被指定为故障切换服务器的服务器 (例如 <code>interface</code> 文件和 <code>ldap</code> 服务器)。 |

Adaptive Server 接受作为选项名称一部分的任何唯一字符串。如果选项名内嵌有空格, 则要将该选项名用引号引起来。

optvalue

除 `security mechanism` 选项外, 对所有选项均为 `true` (开启) 或 `false` (关闭)。

对于 `security mechanism` 选项, 请指定安全机制的名称。若要查看服务器上可用安全机制的名称, 请执行以下查询:

```
select * from syssecmechs
```

示例

示例 1 显示服务器选项的列表:

```
sp_serveroption
Settable server options.
-----
cis hafailover
enable login redirection
external engine auto start
incompatible sort order
mutual authentication
```

```

negotiated logins
net password encryption
readonly
relocated joins
security mechanism
server cost
server logins
server principal
timeouts
use message confidentiality
use message integrity

```

示例 2 告知服务器不要使与远程服务器 GATEWAY 的不活动物理连接超时:

```
sp_serveroption GATEWAY, "timeouts", false
```

示例 3 指定在连接到远程服务器 GATEWAY 时, GATEWAY 将发回一个加密密钥用于加密要发送到该服务器的口令:

```
sp_serveroption GATEWAY, "net password encryption", true
```

示例 4 指定 EJB Server SYB_EJB 在每次 Adaptive Server 启动时启动:

```
sp_serveroption SYB_EJB, "external engine auto start", true
```

示例 5 为与远程服务器 S2 的连接指定 Kerberos 鉴定。

```
sp_serveroption S2, "security mechanism", csfkrb5
```

示例 6 使用 Kerberos 鉴定为所有与远程服务器的连接指定相互鉴定。

```
sp_serveroption TEST3, "mutual authentication", true
```

示例 7 禁用自动启动, 其中, SYB_EJB 是 EJB Server 的逻辑名称:

```
sp_serveroption 'SYB_EJB', 'external engine auto start', 'false'
```

若要启用自动启动, 请输入:

```
sp_serveroption 'SYB_EJB', 'external engine auto start', 'true'
```

有关使用 `external engine auto start` 的详细信息, 请参见《EJB Server 用户指南》的第 2 章“入门”中的“自动启动 EJB Server”。

用法

- 若要显示可由用户设置的服务器选项的列表, 请使用不带任何参数的 `sp_serveroption`。
- 将 `timeouts` 设置为 `false` 后, 站点处理器将继续运行, 直到两台服务器中的一台关闭。

- **net password encryption** 选项允许客户端指定在启动远程过程调用时以明文形式还是以加密形式在网络上发送口令。如果 **net password encryption** 为 **true**，发送的初始登录包中没有口令，并且客户端向远程服务器指示需要加密。远程服务器发回一个加密密钥，客户端将用它来加密其口令。然后，客户端加密其口令，当加密口令到达远程服务器后，远程服务器使用该密钥对口令进行鉴定。
- 若要为特定的 **isql** 会话设置网络口令加密，可以使用 **isql** 的命令行选项。有关详细信息，请参见相应于所用平台的实用程序手册。
- 连接到版本 10.0 之前的 SQL Server 时，不能使用 **net password encryption** 选项。
- **security mechanism**、**mutual authentication**、**use message confidentiality** 和 **use message integrity** 选项仅适用于 Kerberos 登录。

权限

只有系统管理员才能执行 **sp_serveroption** 来设置 **timeouts** 选项。任何用户都可以执行不带参数的 **sp_serveroption** 以显示选项列表。

只有系统安全员才能设置 **net password encryption**、**security mechanism**、**mutual authentication**、**use message confidentiality** 和 **use message integrity** 选项。

审计

sysaudits 表的 **event** 和 **extrainfo** 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 有关服务器选项的详细信息，请参见《系统管理指南》。

系统过程 [sp_helpserver](#)、[sp_password](#)

实用程序 [isql](#)

sp_set_qplan

说明 更改现有计划的抽象计划文本，而不更改关联的查询。

语法 `sp_set_qplan id, plan`

参数 `id`
是抽象计划的 ID。

`计划`
是新的抽象计划。

示例

```
sp_set_qplan 563789159,
    "( g_join (scan t1) (scan t2))"
```

用法

- 使用 `sp_set_qplan` 可以更改现有计划的抽象计划。可以指定长度最大为 255 个字符的计划。如果抽象计划的长度超过 255 个字符，可以先使用 `sp_drop_qplan` 删除旧计划，然后再使用 `create plan` 为查询创建新计划。
- 如果用 `sp_set_qplan` 更改计划，则不检查计划的抽象计划语法是否有效，也不检查计划与 SQL 文本的兼容性。对于所有使用 `sp_set_qplan` 修改的计划，应立即通过针对指定的 ID 运行查询来检查其是否正确。
- 若要查找计划的 ID，请使用 `sp_help_qpgroup`、`sp_help_qplan` 或 `sp_find_qplan`。`create plan` 还返回计划 ID，这些计划 ID 包括在 `showplan` 输出中。

权限 任何用户都可以执行 `sp_set_qplan` 来更改属于该用户的计划的文本。只有系统管理员或数据库所有者才能更改属于其他用户的计划的文本。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** `create plan`

系统过程 `sp_drop_qpgroup`、`sp_drop_qplan`、`sp_find_qplan`、`sp_help_qplan`

sp_setlangalias

说明 指定或更改替代语言的别名。

语法 sp_setlangalias *language*, *alias*

参数

language
是替代语言的正式语言名称。

alias
是替代语言的新本地别名。

示例 sp_setlangalias french, français

此命令为正式语言名称 “french” 指派别名 “français”。

用法

- *alias* 将替换该正式名称的 syslanguages.alias 的当前值。
- set language 命令可以使用新的 *alias* 代替该正式语言名称。

权限 只有系统管理员才能执行 sp_setlangalias。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 命令 [set](#)

系统过程 [sp_addlanguage](#)、[sp_droplanguage](#)、[sp_helplanguage](#)

sp_setpglockpromote

| | |
|----|--|
| 说明 | 设置或更改数据库、表或 Adaptive Server 的锁升级阈值。 |
| 语法 | <pre>sp_setpglockpromote {"database" "table"}, objname, new_lwm, new_hwm, new_pct sp_setpglockpromote server, NULL, new_lwm, new_hwm, new_pct</pre> |
| 参数 | <p>server 为锁升级阈值设置全服务器范围的值。</p> <p>"database" "table" 指定是为数据库还是为表设置锁升级阈值。“database”和“table”是 Transact-SQL 关键字，因此需要使用引号。</p> <p>objname 是要设置锁升级阈值的表或数据库的名称或 null（如果要设置全服务器范围的值）。</p> <p>new_lwm 指定要为低水印 (LWM) 阈值设置的值。LWM 必须小于或等于高水印 (HWM)。LWM 的最小值是 2。该参数可以是 null。</p> <p>new_hwm 指定要为锁升级 HWM 阈值设置的值。HWM 必须大于或等于 LWM。HWM 的最大值是 2,147,483,647。该参数可以是 null。</p> <p>new_pct 指定要为锁升级百分比 (PCT) 阈值设置的值。PCT 必须介于 1 到 100 之间。该参数可以是 null。</p> |
| 示例 | <p>示例 1 将全服务器范围的锁升级 LWM 设置为 200，将 HWM 设置为 300，将 PCT 设置为 50：</p> <pre>sp_setpglockpromote "server", NULL, 200, 300, 50</pre> <p>示例 2 为 master 数据库设置锁升级阈值：</p> <pre>sp_setpglockpromote "database", master, 1000, 1100, 45</pre> <p>示例 3 为 pubs2 数据库中的 titles 表设置锁升级阈值。必须从 pubs2 数据库中发出此命令：</p> <pre>sp_setpglockpromote "table", "pubs2..titles", 500, 700, 10</pre> <p>示例 4 将 master 数据库的 HWM 阈值更改为 1600。这些阈值以前是用 sp_setpglockpromote 设置的。必须从 master 数据库中发出此命令：</p> <pre>sp_setpglockpromote "database", master, @new_hwm=1600</pre> |

用法

- 可以使用 `sp_helpdb dbname` 来显示数据库级别的锁升级，使用 `sp_helpdb tablename` 来显示表级别的锁。
- `sp_setpglockpromote` 为表、数据库或 Adaptive Server 配置锁升级值。Adaptive Server 获取表上的页锁，直到锁数超过锁升级阈值为止。`sp_setpglockpromote` 可以更改对象、数据库或服务器的锁升级阈值。如果 Adaptive Server 成功获取了表锁，这些页锁将被释放。
当表上的页锁数超过 HWM 阈值时，Adaptive Server 将尝试升级到表锁。当表上的页锁数小于 LWM 时，Adaptive Server 不会尝试升级到表锁。当表上的页锁数在 HWM 和 LWM 之间，并且锁数超过 PCT 阈值时，Adaptive Server 将尝试升级到表锁。
- 表的锁升级阈值将替换数据库或全服务器范围的设置。数据库的锁升级阈值将替换全服务器范围的设置。
- Adaptive Server 的锁升级阈值不需要初始化，但必须通过使用 `sp_setpglockpromote` 指定 LWM、HWM 和 PCT 来初始化数据库和表的锁升级阈值。第一次对数据库或表运行此存储过程时，它将在 `sysattributes` 中为该对象创建一行。阈值初始化之后，即可单独对各个值进行修改，如示例 4 中所示。
- 对于表或数据库，`sp_setpglockpromote` 在一个事务中设置 LWM、HWM 和 PCT。如果 `sp_setpglockpromote` 在更新其中任何值时遇到错误，所有更改都将中止，并且事务将回退。对于全服务器范围的更改，可能会有一个或多个阈值更新失败，而其它阈值则更新成功。如果有任何值更新失败，Adaptive Server 将返回一条错误消息。
- 若要查看全服务器范围的锁升级阈值设置，请使用 `sp_configure "lock promotion"` 来查看全部三个阈值。若要查看数据库的锁升级设置，请使用 `sp_helpdb`。若要查看表的锁升级设置，请使用 `sp_help`。

权限

只有系统管理员才能执行 `sp_setpglockpromote`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 `sp_configure`、`sp_droplockpromote`、`sp_help`、`sp_helpdb`

sp_setpsex

| | |
|----|---|
| 说明 | 在会话处于活动状态时，为会话设置自定义执行属性。 |
| 语法 | <code>sp_setpsex <i>spid</i>, <i>exeattr</i>, <i>value</i></code> |
| 参数 | <p><i>spid</i> 是要设置执行变量的会话的 ID。使用 <code>sp_who</code> 可以查看 <code>spid</code>。</p> <p><i>exeattr</i> 标识要设置的执行属性。值为 <code>priority</code> 和 <code>enginegroup</code>。</p> <p><i>value</i> 是 <i>exeattr</i> 的新值。各个属性的值如下所示：</p> <ul style="list-style-type: none"> 如果 <i>exeattr</i> 为 <code>priority</code>，则 <i>value</i> 为 <code>HIGH</code>、<code>MEDIUM</code> 或 <code>LOW</code>。 如果 <i>exeattr</i> 为 <code>enginegroup</code>，则 <i>value</i> 为现有引擎组的名称。 |
| 示例 | <p>此示例将 ID 为 1 的进程的优先级设置为 <code>HIGH</code>：</p> <pre>sp_setpsex 1, "priority", "HIGH"</pre> |
| 用法 | <ul style="list-style-type: none"> 使用 <code>sp_setpsex</code> 指定的执行属性值仅对当前会话有效，在该会话终止后不再适用。 使用 <code>sp_setpsex</code> 时应谨慎，否则可能会导致性能下降。如果进程获取不到 CPU 时间，使用 <code>sp_setpsex</code> 即时更改属性会有所帮助；但是，如果性能问题是由其它原因（例如锁）造成的，更改执行属性可能反而会导致问题更加严重。 由于只能为会话设置执行属性，因此不能对工作进程 <code>spid</code> 执行 <code>sp_setpsex</code>。 除管家 <code>spid</code> 外，不能为系统 <code>spids</code> 设置执行属性。 如果相关引擎组中没有联机引擎，<code>sp_setpsex</code> 不起作用。 |
| 权限 | 只有系统管理员才能不受限制地执行 <code>sp_setpsex</code> 。任何用户都可以执行 <code>sp_setpsex</code> 来降低属于该用户的进程的优先级。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|---|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — <code>NULL</code> 以前值 — <code>NULL</code> 当前值 — <code>NULL</code> 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_addexclass](#)、[sp_bindexclass](#)、[sp_dropexclass](#)、[sp_showexclass](#)

sp_setrowlockpromote

| | |
|----|--|
| 说明 | 为某个数据行锁定表、数据库中的所有数据行锁定表或服务器上的所有数据行锁定表设置或更改行锁升级阈值。 |
| 语法 | <pre>sp_setrowlockpromote "server", NULL, new_lwm, new_hwm, new_pct sp_setrowlockpromote {"database" "table"}, objname, new_lwm, new_hwm, new_pct</pre> |
| 参数 | <p>server 为行锁升级阈值设置全服务器范围的值。</p> <p>"database" "table" 指定为数据库还是为表设置行锁升级阈值。</p> <p>objname 是要设置行锁升级阈值的表或数据库的名称或 null（如果要设置全服务器范围的值）。</p> <p>new_lwm 指定要为低水印 (LWM) 阈值设置的值。LWM 必须小于或等于高水印 (HWM)。LWM 的最小值是 2。该参数可以是 null。</p> <p>new_hwm 指定要为高水印 (HWM) 阈值设置的值。HWM 必须大于或等于 LWM。HWM 的最大值是 2,147,483,647。该参数可以是 null。</p> <p>new_pct 指定要为锁升级百分比 (PCT) 阈值设置的值。PCT 必须介于 1 到 100 之间。该参数可以是 null。</p> |
| 示例 | <p>示例 1 为 engdb 数据库中的所有数据行锁定表设置行锁升级值：</p> <pre>sp_setrowlockpromote "database", engdb, 400, 400, 95</pre> <p>示例 2 为 sales 表设置行锁升级值：</p> <pre>sp_setrowlockpromote "table", sales, 250, 250, 100</pre> |
| 用法 | <ul style="list-style-type: none"> • 可以使用 <code>sp_helpdb dbname</code> 来显示数据库级别的锁升级，使用 <code>sp_helpdb tablename</code> 来显示表级别的锁。 • <code>sp_setrowlockpromote</code> 可设置或更改表、数据库或 Adaptive Server 的行锁升级阈值。 <p>Adaptive Server 获取数据行锁定表上的行锁，直到锁数超过锁升级阈值为止。如果 Adaptive Server 成果获取了表锁，这些行锁将被释放。</p> |

当表上的行锁数超过了 HWM 时，Adaptive Server 将尝试升级到表锁。当表上的行锁数小于 LWM 时，Adaptive Server 不会尝试升级到表锁。当表上的行锁数在 HWM 和 LWM 之间，并且行锁数超过了 PCT 阈值（表中行数的百分比）时，Adaptive Server 将尝试升级到表锁。

- 锁升级始终从一层升级到另一层，也就是说，行锁将升级到表锁。Adaptive Server 不会从行锁升级到页锁。
- 表的锁升级阈值将替换数据库或全服务器范围的设置。数据库的锁升级阈值将替换全服务器范围的设置。
- 若要更改数据库的锁升级阈值，必须使用 master 数据库。若要更改数据库中表的锁升级阈值，必须使用表所在的数据库。
- 全服务器范围的行锁升级阈值也可以用 sp_configure 设置。使用 sp_setrowlockpromote 更改全服务器范围的值时，它将更改配置文件并保存配置文件。第一次安装 Adaptive Server 时，由配置参数设置的全服务器范围行锁升级阈值为：

| | |
|------------------------|-----|
| row lock promotion HWM | 200 |
| row lock promotion LWM | 200 |
| row lock promotion PCT | 100 |

有关详细信息，请参见《系统管理指南》。

- 系统过程 sp_sysmon 报告有关行锁升级的信息。
- 数据库级别的行锁升级阈值存储在 master..sysattributes 表中。如果转储数据库，然后将其装载到另一台服务器上，则必须在新服务器上设置行锁升级阈值。对象级别的行锁升级阈值存储在用户数据库的 sysattributes 表中，转储中包括这些阈值。

权限

只有系统管理员才能执行 sp_setrowlockpromote。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_configure](#)、[sp_droprolockpromote](#)、[sp_helpdb](#)、[sp_sysmon](#)

sp_setsuspect_granularity

说明 显示或设置用户数据库的恢复故障隔离模式，该模式控制在检测到数据损坏时将如何进行恢复。

语法 `sp_setsuspect_granularity [dbname
[, "database" | "page" [, "read_only"]]]`

参数 *dbname*

是数据库的名称，就是要显示或设置该数据库的恢复故障隔离模式。对于显示，缺省为当前数据库。对于设置，您必须在 **master** 数据库中并指定目标 *dbname*。

database

如果恢复进程检测到其任何数据的状态为可疑，则将整个数据库标记为可疑，使其不可访问。

page

如果恢复进程检测到数据库中有损坏的数据，则只将损坏的页标记为可疑，使那些页不可访问。但仍可访问其余的数据。

read_only

如果指定，在恢复进程将任何页标记为可疑时，将整个数据库标记为 *read only*。

示例 **示例 1** 显示当前数据库的恢复故障隔离模式：

```
sp_setsuspect_granularity
DB Name  Cur. Suspect Gran.  Cfg. Suspect Gran.  Online mode
-----  -
pubs2    database            database            read/write
```

示例 2 显示 **pubs2** 数据库的当前恢复故障隔离模式和配置的恢复故障隔离模式：

```
sp_setsuspect_granularity pubs2
```

示例 3 下次在 **pubs2** 数据库中运行恢复时，如果检测到任何损坏页，只会使可疑页脱机，而仍会使该数据库中其余的部分联机：

```
sp_setsuspect_granularity pubs2, "page"
DB Name          Cur. Suspect Gran.  Cfg. Suspect Gran.
-----  -
pubs2            database            database
sp_setsuspect_granularity: The new values will become effective
during the next recovery of the database 'pubs2'.
```

示例 4 下次在 pubs2 数据库中运行恢复时，如果检测到任何损坏页，只会使可疑页脱机，而仍会以只读模式使该数据库中其余的部分联机：

```
sp_setsuspect_granularity pubs2, "page", "read_only"
```

示例 5 下次在 pubs2 数据库中运行恢复时，如果检测到任何损坏数据，将把整个数据库标记为可疑并使其脱机：

```
sp_setsuspect_granularity pubs2, "database"
```

用法

- **sp_setsuspect_granularity** 显示和设置恢复故障隔离模式。该模式控制恢复进程在检测到它所需要的任何数据已损坏时，将整个数据库标记为可疑，还是只将损坏页标记为可疑。有关详细信息，请参见《系统管理指南》。
- 用户数据库的缺省恢复故障隔离模式为“database”。只能设置用户数据库的恢复故障隔离模式，而不能设置系统数据库的恢复故障隔离模式。
- 您必须在 master 数据库中才能设置故障恢复隔离模式。
- 因损坏而标记为可疑的数据在 Adaptive Server 启动过程中将保持其状态。如果某些页已被标记为可疑，在重新启动服务器后它们仍保持脱机状态。
- 如果数据库的一部分或整个数据库被标记为可疑，用户将无法访问可疑数据，除非系统管理员使用 **sp_forceonline_db** 和 **sp_forceonline_page** 过程使可疑数据可访问。
- 即使恢复故障隔离模式为“page”，一般数据库损坏（例如，数据库日志损坏或不特定于某一页的其它资源不可用）也会导致整个数据库被标记为可疑。
- 如果不指定 page 或 database，Adaptive Server 将显示当前设置和配置的设置。当前设置是上次在该数据库中执行恢复时生效的那个设置。配置的设置是下次在该数据库中执行恢复时将生效的那个设置。
- 如果数据库以 read_only 模式联机，任何用户都不能修改数据库中的任何数据，包括不受可疑页影响并因此而联机的数据。但是，系统管理员可以使用 **sp_dboption** 系统过程将 read only 设置为 false，使数据库可写。这种情况下，用户可以修改联机数据，但可疑数据仍不可访问。

权限

只有系统管理员才能执行 **sp_setsuspect_granularity** 来设置恢复故障隔离模式。任何用户都可以执行 **sp_setsuspect_granularity** 来显示设置。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [dump database](#)、[dump transaction](#)、[load database](#)

系统过程 [sp_dboption](#)、[sp_forceonline_db](#)、[sp_forceonline_page](#)、[sp_listsuspect_db](#)、[sp_listsuspect_page](#)、[sp_setsuspect_threshold](#)

sp_setsuspect_threshold

说明 显示或设置 Adaptive Server 在数据库中的可疑页数达到多少（允许的最大可疑页数）之后就会将整个数据库标记为可疑。

语法 `sp_setsuspect_threshold [dbname [, threshold]]`

参数 *dbname*

是数据库的名称，就是要显示或设置该数据库的可疑性增加阈值。缺省为当前数据库。

threshold

指示恢复进程在可疑数据页数达到多少（允许的最大可疑数据页数）之后就会将整个数据库标记为可疑。缺省值为 20 页。最小值为 0。

示例 **示例 1** 将最大可疑页数设置为 5。如果可疑页数超过 5，恢复进程将把整个数据库标记为可疑：

```
sp_setsuspect_threshold pubs2, 5
```

示例 2 显示 pubs2 数据库的可疑性增加阈值的当前设置和配置的设置：

```
sp_setsuspect_threshold pubs2
```

示例 3 显示当前用户数据库的恢复故障隔离阈值的当前设置和配置的设置：

```
sp_setsuspect_threshold
```

用法

- 您必须在 master 数据库中才能使用 sp_setsuspect_threshold 设置可疑性增加阈值。
- 如果不指定页数，Adaptive Server 将显示当前设置和配置的设置。当前设置是上次在该数据库中执行恢复时生效的那个设置。配置的设置是下次在该数据库中执行恢复时将生效的那个设置。

权限 只有系统管理员才能执行 sp_setsuspect_threshold 来设置增加阈值。任何用户都可以执行 sp_setsuspect_threshold 来显示当前设置。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_forceonline_db](#)、[sp_forceonline_page](#)、[sp_listsuspect_db](#)、[sp_listsuspect_page](#)、[sp_setsuspect_granularity](#)

sp_setup_table_transfer

| | |
|----|--|
| 说明 | 在包含标记为增量传输的表的每个数据库中各运行一次，以在相应数据库中创建 <code>spt_TableTransfer</code> 表。 |
| 语法 | <code>sp_setup_table_transfer</code> |
| 用法 | 尽管 <code>sp_setup_table_transfer</code> 是可选的，但 Sybase 仍然建议在传输表前运行它。如果不运行 <code>sp_setup_table_transfer</code> ，Adaptive Server 将在表被标记为进行增量传输或您首次执行传输时自动创建 <code>spt_TableTransfer</code> 。 |
| 权限 | 必须具有 <code>sa_role</code> 或者是数据库所有者才能运行 <code>sp_setup_table_transfer</code> 。 |

sp_show_options

| | |
|----|---|
| 说明 | 输出在当前会话中设置的所有服务器选项。 |
| 语法 | <code>sp_show_options</code> |
| 用法 | <code>@@options</code> — 由与服务器选项对应的位组成的数组。对于每个选项，“low”为 <code>@@options</code> 中的字节编号，“high”为该字节内与此选项对应的位。如果该位已设置，则输出对应选项的名称。 |

sp_showcontrolinfo

说明 显示有关线程池指派、绑定的客户端应用程序、登录名和存储过程的信息。

有关进程模式的考虑事项 如果为 Adaptive Server 配置了进程模式，sp_showcontrolinfo 将显示有关引擎组指派、绑定的客户端应用程序、登录名和存储过程的信息。

语法 sp_showcontrolinfo [object_type, object_name, spid]

参数

object_type

可以是以下值之一：

- AP（表示应用程序）
- LG（表示登录名）
- PR（表示存储过程）
- EG（表示线程池（线程化模式）或引擎组（进程模式））
- SV（表示服务任务）
- PS（表示进程）
- DF（表示用户定义的缺省执行类）

如果不指定 *object_type* 或将 *object_type* 指定为 null，则 sp_showcontrolinfo 将显示有关所有类型的信息。

object_name

是应用程序、登录名、存储过程或引擎组的名称。如果指定 PS 或 DF 作为 *object_type*，请勿指定 *object_name*。如果不指定 *object_name*（或将 *object_name* 指定为 null），sp_showcontrolinfo 将显示有关所有对象名称的信息。

spid

是 Adaptive Server 进程 ID。仅当将 PS 指定为 *object_type* 时，才需要指定 spid。如果不指定 spid（或将 spid 指定为 null），sp_showcontrolinfo 将显示有关所有 spid 的信息。使用 [sp_who](#) 可以查看 spid。

示例

示例 1 显示所有由用户指定的、执行类与对象的绑定：

```
sp_showcontrolinfo
```

示例 2 显示 isql 应用程序的执行类：

```
sp_showcontrolinfo 'AP', 'isql'
```

示例 3 显示分配给线程池的所有进程的执行类：

```
sp_showcontrolinfo 'PS'
```


示例 4 显示 spid 7 的执行类：

```
sp_showcontrolinfo 'PS', null, 7
```

用法

- 如果不指定任何参数，`sp_showcontrolinfo` 将显示有关所有用户指定的线程池指派、绑定的客户端应用程序、登录名和存储过程的信息。如果指定了 `object_type` 参数，`sp_showcontrolinfo` 将分别显示有关应用程序、登录名或存储过程与执行类的绑定、线程池构成和会话级别属性绑定的信息。请参见《性能和调优系列：基础知识》中的第 4 章“分配引擎资源”。
- 如果是在进程模式中运行，`sp_showcontrolinfo` 会将 `thread_pool` 替换为 `engine_group` 和 `engine` 列。
- 如果 `object_type` 不是 PR，应从 master 数据库执行 `sp_showcontrolinfo`。如果 `object_type` 是 PR，应从该过程所在的数据库中执行 `sp_showcontrolinfo`。
- 如果 `object_type` 是：
 - `null` — `sp_showcontrolinfo` 将显示与其它参数匹配的对象执行类信息。
 - `DF` — `object_name` 和 `spid` 应为空，而且 `sp_showcontrolinfo` 显示有关用户定义的缺省执行类的信息。
- 如果 `object_name` 为 `null`，`sp_showcontrolinfo` 将显示所有应用程序、登录和存储过程的绑定信息。
- 如果 `spid` 为 `null`，`sp_showcontrolinfo` 将显示与其它参数匹配的对象执行类信息。

权限

任何用户都可以执行 `sp_showcontrolinfo`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_addexclass](#)、[sp_bindexclass](#)、[sp_clearpsex](#)、[sp_dropengine](#)、[sp_dropexclass](#)、[sp_showexclass](#)、[sp_showpsex](#)、[sp_unbindexclass](#)、[sp_who](#)

实用程序 `isql`

sp_showexeclass

说明 显示与指定的执行类关联的执行类属性，以及与其关联的线程池名称。

有关进程模式的考虑事项 在进程模式中，`sp_showexeclass` 显示执行类属性以及与指定的执行类相关联的所有引擎组中的引擎。

语法 `sp_showexeclass [execlassname]`

参数 `execlassname`
是执行类的名称。

示例 1 显示所有执行类的优先级和线程池：

```
sp_showexeclass
classname          priority          threadpool
-----
EC1                  HIGH            syb_default_pool
EC2                  MEDIUM        syb_default_pool
EC3                  LOW            syb_default_pool
```

示例 2 显示执行类 EC1 的属性值：

```
sp_showexeclass 'EC1'
classname          priority          threadpool
-----
EC1                  HIGH            syb_default_pool
```

用法

- 如果 `execlassname` 为 NULL 或未指定，`sp_showexeclass` 将显示所有执行类的优先级和线程池属性值，包括系统定义的类 EC1、EC2 和 EC3 的属性值。

权限 任何用户都可以执行 `sp_showexeclass`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_addexeclass](#)、[sp_bindexeclass](#)、[sp_dropexeclass](#)、[sp_showcontrolinfo](#)、[sp_unbindexeclass](#)

sp_showoptstats

| | |
|----|---|
| 说明 | <p><code>sp_showoptstats</code> 在功能上与 XML 文档中的 <code>optdiag</code> 独立实用程序类似，但采用系统过程格式，它提取和显示系统表（如 <code>systabstats</code> 和 <code>sysstatistics</code>）中的各种数据对象的统计信息和直方图。</p> |
| 语法 | <code>sp_showoptstats</code> <i>[[database_name].[owner].]table_name</i> , [<i>column_name</i>], [h] |
| 参数 | <p>database_name</p> <p>是 <code>sp_showoptstats</code> 要为其显示统计信息和直方图的数据库的名称。<i>dbname</i> 具有以下限制：</p> <ul style="list-style-type: none">• 不支持跨数据库执行• 必须当前位于指定的数据库中才能执行 <code>sp_showoptstats</code>。• 如果不指定数据库，<code>sp_showoptstats</code> 会显示有关当前数据库的统计信息和直方图。 <p>owner</p> <p>是表所有者的名称。如果不指定所有者名称，则会使用当前用户或 <code>dbo</code>。</p> <p>table_name</p> <p>是 <code>sp_showoptstats</code> 要为其显示统计信息和直方图的表的名称。<i>table_name</i> 具有以下限制：</p> <ul style="list-style-type: none">• 如果不指定表，<code>sp_showoptstats</code> 会显示有关当前数据库中所有表的统计信息和直方图。但是，若要缩短输出的长度，<code>sp_showoptstats</code> 就不会在数据库级别显示列统计信息和直方图。• <i>table_name</i> 必须存在于当前数据库中。 <p>column_name</p> <p>是 Adaptive Server 要为其显示统计信息和直方图的列的名称。如果不指定列，Adaptive Server 会显示您指定的表上的所有列和索引的统计信息。如果指定 <i>column_name</i>，<code>sp_showoptstats</code> 会仅显示该列的统计信息和直方图。</p> <p>h</p> <p>显示关于该过程的帮助信息。</p> |

示例 **1** 显示 pubs2 数据库中所有用户表的统计信息：

```
1> use pubs2
2> go
1> sp_showoptstats 'pubs2..publishers'
2> go
```

示例 2 显示 pubs2 数据库中的 publishers 表的统计信息和直方图（XML 格式）：

```

1> use pubs2
2> go
1> sp_showoptstats publishers
2> go
The latest output is:
-----
<?xml version="1.0" encoding="UTF-8"?>
  <optStats>
    <procVersion>sp_showoptstats/1.0/AnyPlatform/AnyOS/Fri Feb 18 18:18:18
2011</procVersion>
    <serverVersion>Adaptive Server Enterprise/15.7/EBF XXXXX SMP Drop#2/B/NT
(IX86)/Windows 2008 R2/aseasap/2617/32-bit/DEBUG/Wed Nov 17 02:14:21
2010</serverVersion>
    <serverName></serverName>
    <specifiedDatabase>pubs2</specifiedDatabase>
    <specifiedTableOwner></specifiedTableOwner>
    <specifiedTable>publishers</specifiedTable>
    <specifiedCol></specifiedCol>
    <tables>
      <tableOwner>dbo</tableOwner>
      <tableName>publishers</tableName>
      <clusteredIndStats>
        <indName>pubind</indName>
        <colList>"pub_id"</colList>
        <stats>
          <pgCnt>1</pgCnt>
          <emptyPgCnt>0</emptyPgCnt>
          <rowCnt>3.000000000000000</rowCnt>
          <fwdRowCnt>0.000000000000000</fwdRowCnt>
          <delRowCnt>0.000000000000000</delRowCnt>
          <CRCnt>1.000000000000000</CRCnt>
          <oamAllocPgCnt>2</oamAllocPgCnt>
          <firstExtLeafPgs>0</firstExtLeafPgs>
          <dataRowSz>39.333333333333360</dataRowSz>
          <indHeight>1</indHeight>
          <joinDegree>0.000000000000000</joinDegree>
          <unusedPgCnt>14</unusedPgCnt>
          <oamPgCnt>1</oamPgCnt>
          <derivedStats>
            <clusterRatio>0.000000000000000</clusterRatio>
            <spaceUtil>0.0072162426614481</spaceUtil>
            <IOEfficiency>0.500000000000000</IOEfficiency>
          </derivedStats>
        </stats>
      </clusteredIndStats>
    <colStats>
      <colName>pub_id</colName>
      <lastUpdate>Dec 10 2010 3:58:14:266PM</lastUpdate>

```

```

<cellDensity>0.3333333333333333</cellDensity>
<totalDensity>0.3333333333333333</totalDensity>
<select>default used (0.33)</select>
<inBetSel>default used (0.25)</inBetSel>
<rangeVal>0.3333333333333333</rangeVal>
<totalVal>0.3333333333333333</totalVal>
<avgColWidth>default used (4.00)</avgColWidth>
<histogram>
  <colName>pub_id</colName>
  <dataType>char (4)</dataType>
  <requestedStepCnt>20</requestedStepCnt>
  <actualStepCnt>6</actualStepCnt>
  <samplingPct>0</samplingPct>
  <steps>
    <step>1</step>
    <weight>0.00000000</weight>
    <equation>&lt;</equation>
    <value>"0736"</value>
  </steps>
  <steps>
    <step>2</step>
    <weight>0.33333334</weight>
    <equation>=</equation>
    <value>"0736"</value>
  </steps>
  <steps>
    <step>3</step>
    <weight>0.00000000</weight>
    <equation>&lt;</equation>
    <value>"0877"</value>
  </steps>
  <steps>
    <step>4</step>
    <weight>0.33333334</weight>
    <equation>=</equation>
    <value>"0877"</value>
  </steps>
  <steps>
    <step>5</step>
    <weight>0.00000000</weight>
    <equation>&lt;</equation>
    <value>"1389"</value>
  </steps>
  <steps>
    <step>6</step>
    <weight>0.33333334</weight>
    <equation>=</equation>
    <value>"1389"</value>
  </steps>
</histogram>
</colStats>

```

```
<noStatsCol>city, pub_name, state
</noStatsCol>
</tables>
</optStats>
```

示例 3 显示过程的语法:

```
1> sp_showoptstats a,b,h
2> go
Usage: sp_showoptstats [[database.[owner].]table], [column], [option]
(return status = 0)
```

用法

- sp_showoptstats 不包括系统表，除非您显式指定了它们。
- 不可打印字符和 univarchar 字符以十六进制格式显示。
- sp_showoptstats 显示全局和分区级统计信息。
- 当输出大于您为 @@textsize 设置的值时，Adaptive Server 会返回消息以增大 @@textsize 设置，从而使其能显示大的输出。
- 包含句点 (.) 的参数值需要双引号。
- 可以针对系统表发出 sp_showoptstats。
- 如果您仅指定数据库和所有者，则 sp_showoptstats 不返回统计信息。

另请参见

实用程序 optdiag

文档 《性能和调优系统：利用统计分析改进性能》中的第 2 章 “Statistics Tables and Displaying Statistics with optdiag”（统计信息表和使用 optdiag 显示统计信息）；《实用程序指南》中的 optdiag 参考页。

sp_showplan

- 说明** 针对任何用户连接，显示当前 SQL 语句或同一批处理中前一语句的 showplan 输出。
- 语法**
- ```
sp_showplan spid, batch_id output,
 context_id output,
 stmt_num output
```
- 若要显示当前 SQL 语句的 showplan 输出而不指定 batch\_id、context\_id 或 stmt\_num，请使用以下语法：
- ```
sp_showplan spid, null, null, null
```
- 参数**
- spid*
是任何用户连接的进程 ID。使用 sp_who 可以查看 spid。
- batch_id*
是唯一的非负批处理编号
- context_id*
是批处理中执行的每个过程（或触发器）的唯一编号。
- stmt_num*
是批处理中当前语句的编号。stmt_num 必须为正数。
- 示例**
- 示例 1** 显示 spid 值为 99 的用户会话中当前运行的语句的查询计划，以及 batch_id、context_id 和 statement_id 参数的值。在对 spid 为 99 的用户会话执行的后续 sp_showplan 迭代中，可以使用这些值来检索查询计划：
- ```
declare @batch int
declare @context int
declare @statement int
exec sp_showplan 99, @batch output, @context output,
@statement output
```
- 示例 2** 显示 spid 值为 99 的用户会话中当前运行的语句的 showplan 输出：
- ```
sp_showplan 99, null, null, null
```
- 用法**
- sp_showplan 显示当前正在执行的 SQL 语句或同一批处理中前一语句的 showplan 输出。
 - 若要查看同一批处理中前一语句的查询计划，请使用相同的参数值再次执行 sp_showplan，但将语句编号减 1。使用这种方法，可以向后查看语句批处理中的所有语句，直到查询编号 1 为止。
 - sp_showplan 可以独立于 Adaptive Server Monitor™ Server 运行。
 - sp_showplan 可以在您使用 sp_procxmode 将事务模式更改为 anymode 后在使用链式事务的会话中运行。

- 如果某个 SQL 批处理的 *context_id* 大于 0，则当前语句嵌入在从原 SQL 批处理调用的存储过程（或触发器）中。选择具有相同 *spid* 值的 *sysprocesses* 行可以显示过程 ID 和语句 ID。

权限

只有系统管理员才能执行 `sp_showplan`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见

系统过程 [sp_who](#)

sp_showpsex

说明 显示 Adaptive Server 上运行的所有客户端会话的执行类、当前优先级和线程池关联。

有关进程模式的考虑事项 sp_showpsex 显示引擎信息，而不是任务关联。

语法 sp_showpsex [spid]

参数 spid

是 Adaptive Server 会话 ID，就是要显示该会话 ID 的报告。该 spid 必须属于执行 sp_showpsex 的应用程序或登录帐户。使用 sp_who 可以列出 spid。

示例 1 显示所有当前客户端会话的执行类、当前优先级和关联：

```
sp_showpsex
```

| spid | appl_name | login_name | exec_class | current_priority | task_affinity |
|------|-----------|------------|------------|------------------|------------------|
| 5 | NULL | NULL | NULL | LOW | syb_default_pool |
| 6 | NULL | NULL | NULL | MEDIUM | syb_default_pool |
| 7 | NULL | NULL | NULL | LOW | syb_default_pool |
| 26 | isql | sa | EC2 | MEDIUM | syb_default_pool |

示例 2 显示 spid 为 5 的进程的应用程序名、登录名、当前优先级和引擎密切连接：

```
sp_showpsex 5
```

| spid | appl_name | login_name | exec_class | current_priority | task_affinity |
|------|-----------|------------|------------|------------------|------------------|
| 5 | NULL | NULL | NULL | LOW | syb_default_pool |

用法

- sp_showpsex 显示所有会话（具有 spid 的对象）的执行类、当前优先级和密切连接。请参见《性能和调优系列：基础知识》中的第 4 章“分配引擎资源”。
- 如果 spid 为 NULL 或未指定，sp_showpsex 将报告有关当前在 Adaptive Server 上运行的所有会话的信息。
- sp_showpsex 不会报告以下系统进程的信息：死锁、检查点、网络、审计和镜像处理器。它显示管家 spid 的信息。

权限 任何用户都可以执行 sp_showpsex。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_addengine](#)、[sp_addexeclass](#)、[sp_bindexclass](#)、[sp_clearpsex](#)、[sp_dropengine](#)、[sp_dropexeclass](#)、[sp_showcontrolinfo](#)、[sp_showexeclass](#)、[sp_unbindexclass](#)

sp_spaceusage

说明

报告表、索引或事务日志的空间使用情况并估计数据库中表和索引的碎片数。估计值是使用数据和索引行的平均行长度以及表中的行数计算的。可将空间使用情况和碎片数据存档，以用于以后的报告和趋势分析。
sp_spaceusage 支持多种操作，包括 **help**、**display**、**archive** 和 **report**，以指示当前的 Adaptive Server 空间使用情况。

语法

“help” 操作语法：

```
sp_spaceusage 'help'[, 'all']
sp_spaceusage 'help' [, {'display' | 'display summary'
                          | 'report' | 'report summary' | 'archive'}
                      [, {'table' | 'index' | 'tranlog'}]]
```

“display” 操作语法：

```
sp_spaceusage 'display summary [using unit= {KB | MB | GB | PAGES}
]',
    {'table' | 'index'}, name
    [,where_clause [,order_by[,command] ] ]
sp_spaceusage 'display [using unit= {KB | MB | GB | PAGES} ]',
    {'table' | 'index'}, name
    [,select_list
    [,where_clause [,order_by[,command] ] ] ]
sp_spaceusage 'display [using unit={KB | MB | GB | PAGES} ]',
    'tranlog' [, name[,select_list[,where_clause [,order_by]]]]
```

“archive” 操作语法：

```
sp_spaceusage 'archive [ using_clause ]',
    {'table' | 'index'}, name[,where_clause[,command] ]
sp_spaceusage 'archive [ using_clause ]',
    'tranlog' [,name[,where_clause] ]
```

“report” 操作语法：

```
sp_spaceusage 'report summary [ using_clause ]',
    {'table' | 'index'}, name
    [,where_clause [,order_by[,from_date [,to_date]]]]
sp_spaceusage 'report [ using_clause ]',
    {'table' | 'index'}, name
    [,select_list[,where_clause [,order_by[,from_date [,to_date]]]]]
sp_spaceusage 'report [ using_clause ]',
    'tranlog' [, name
    [,select_list[,where_clause [,order_by
    [,from_date [,to_date]]]]]]
using_clause = USING using_item [, using_item ...]
```

```
using_item = { unit={ KB | MB | GB | PAGES }  
              | dbname=database_name | prefix=string }
```

参数

help

显示完整的 sp_spaceusage 语法。help *action* 显示受支持的各项操作的语法。

display

显示指定对象的当前空间使用信息。

display summary

显示指定对象的当前空间使用信息摘要。

archive

将空间使用报告存档到表中。如果存档表尚不存在，sp_spaceusage 将创建一个。新数据将附加到现有数据中。使用 *using* 子句可为存档表名和存档表所在的数据库指定前缀。

report

根据以前存档的数据，报告指定对象的空间使用信息。该输出与 display 操作相同。包括可选的 *using* 子句，用以指定存档表。

report summary

根据以前存档的数据，报告指定对象的空间使用信息摘要。该输出与 display summary 操作相同。包括可选的 *using* 子句，用以指定存档表。

using_item

为存档表指定单位、存档数据库名称和前缀字符串。可以使用千字节 (KB)、兆字节 (MB)、千兆字节 (GB) 和页表示 *unit* 大小。缺省情况下，*unit* 大小为 KB，当前数据库是存档数据库，并且不使用前缀字符串。

name

是实体的名称。根据实体类型，可以包括由多个部分组成的名称，如 *owner_name.table_name* 或 *owner_name.table_name.index_name*。对于实体类型 tranlog，名称必须为 syslogs 或 NULL。在由多个部分组成的名称中，每个部分都允许使用模式指示符，以支持在一次传递中报告多个对象。

select_list

是以逗号分隔的一组列，要在输出列中选择这些列供 display 和 report 操作使用。使用 * 将包括输出中的所有列。可使用 *alias=name* 表示法对列进行重命名。

where_clause

是应用于结果集的过滤器。与 display、report 或 archive 操作一起使用可选择性地过滤掉不必要的数据。

order_by

按排序顺序返回指定列中的查询结果。

command

在收集空间使用信息之前为限定对象对所选实体（表、列等）运行的命令。支持以下命令：**update statistics**、**update table statistics** 和 **update index statistics**。

from_date

指定您感兴趣的时间范围的开始时间。

to_date

指定您感兴趣的时间范围的结束时间。

示例

示例 1 显示 **display** 操作的简要说明、语法和用法信息：

```
sp_spaceusage 'help', 'display'

Display the space usage information for an entity in the current database.

Usage:
sp_spaceusage 'display', {'table'|'index'}, <name>
    [,<select_list> [,<where_clause> [,<order_by> [,<command>]]]]

sp_spaceusage 'display summary', {'table'|'index'}, <name>
    [,<where_clause> [,<order_by> [,<command>]]]]

sp_spaceusage 'display', 'tranlog' [,{ 'syslogs'|NULL}
    [,<select_list> [,<where_clause> [,<order_by>]]]]

For more information, use:
sp_spaceusage 'help', 'display', 'table'
sp_spaceusage 'help', 'display', 'index'
sp_spaceusage 'help', 'display', 'tranlog'
```

示例 2 显示 **titles** 表的空间使用摘要：

```
sp_spaceusage 'display summary', 'table', 'titles'

All the page counts in the result set are in the unit 'KB'.
OwnerName  TableName  Type    UsedPages  RsvdPages  ExpRsvdPages  PctBloatRsvdPages
-----
dbo         titles     DATA   6.0        30.0       16.0          87.50
dbo         titles     INDEX   8.0        64.0       32.0          50.00
```

示例 3 显示 titles 表的空间使用信息:

```
sp_spaceusage 'display', 'table', 'titles'
```

All the page counts in the result set are in the unit 'KB'.

| OwnerName | TableName | IndId | NumRows | UsedPages | RsvdPages |
|------------|--------------|------------------|-------------------|-----------|-----------|
| ExtentUtil | ExpRsvdPages | PctBloatUsePages | PctBloatRsvdPages | | |
| dbo | titles | 0 | 18.0 | 6.0 | 30.0 |
| 20.00 | 16.0 | 0.0 | | 87.50 | |
| dbo | titles | 1 | NULL | 4.0 | 32.0 |
| 12.50 | 16.0 | 0.00 | | 100.00 | |
| dbo | titles | 2 | NULL | 4.0 | 32.0 |
| 12.50 | 16.0 | 0.00 | | 100.00 | |

示例 4 显示 titles 表上名称以 title 开头的所有索引的空间使用信息（以兆字节为单位）:

```
sp_spaceusage 'display using unit-MB', 'index', 'titles.title%'
```

All the page counts in the result set are in the unit 'MB'.

| OwnerName | TableName | IndId | IndexName | UsedPages | RsvdPages |
|------------|--------------|-------------------|-------------------|------------|------------|
| ExtentUtil | ExpRsvdPages | PctBloatUsedPages | PctBloatRsvdPages | | |
| dbo | titles | 0 | titles | .005859375 | .029296875 |
| 20.00 | .015625 | 0.00 | | 87.50 | |
| dbo | titles | 1 | titleidind | .00390625 | .03125 |
| 12.50 | .015625 | 0.00 | | 100.00 | |
| dbo | titles | 2 | titleind | .00390625 | .03125 |
| 12.50 | .015625 | 0.00 | | 100.00 | |

(1 row affected)

(return status = 0)

示例 5 显示 titles 表中名称以 title 开头的所有索引的空间使用摘要:

```
sp_spaceusage 'display summary', 'index', 'titles.title%'
```

All the page counts in the result set are in the unit of 'KB'.

| OwnerName | TableName | IndexName | IndId | UsedPages | RsvdPages | ExpRsvdPages |
|-------------------|-----------|------------|-------|-----------|-----------|--------------|
| PctBloatRsvdPages | | | | | | |
| dbo | titles | titles | 0 | 6.0 | 30.0 | 16.0 |
| 46.67 | | | | | | |
| dbo | titles | titleidind | 1 | 4.0 | 32.0 | 16.0 |
| 50.00 | | | | | | |

```

dbo          titles      titleind    2      4.0          32.0        16.0
50.00

```

示例 6 显示 `titles` 表中以 `title` 开头并且 `PctBloatRsvdPages` 值小于 50 的所有索引的空间使用摘要：

```

sp_spaceusage 'display summary', 'index', 'titles.title%',
'where PctBloatRsvdPages < 50'

```

All the page counts in the result set are in the unit 'KB'.

```

OwnerName  TableName  IndexName  IndId  UsedPages  RsvdPages  ExprsvdPages
PctBloatRsvdPages
-----
-----
dbo         titles    titles     0      6.0        30.0       16.0
46.67

```

示例 7 按 `PctBloatRsvdPages` 的降序显示 `titles` 表中 `PctBloatRsvdPages` 值大于 30 的所有索引的空间使用摘要：

```

1> sp_spaceusage 'display summary', 'index', 'titles.title%', 'where
PctBloatRsvdPages > 30', 'order by PctBloatRsvdPages desc'

```

All the page counts in the result set are in the unit 'KB'.

```

OwnerName  TableName  IndexName  IndId  UsedPages  RsvdPages  ExprsvdPages
PctBloatRsvdPages
-----
-----
dbo         titles    titleidind 1      4.0        32.0       16.0
50.00
dbo         titles    titleind   2      4.0        32.0       16.0
50.00
dbo         titles    titles     0      6.0        30.0       16.0
46.67

```

示例 8 对 `authors` 表运行 `update table statistics` 并以 `pages` 为单位汇总其空间使用信息：

```

sp_spaceusage 'display summary using unit=pages', 'table', 'authors', null,
null, null, 'update table statistics'

```

All the page counts in the result set are in the unit 'pages'.

```

OwnerName  TableName  Type  UsedPages  RsvdPages  ExprsvdPages  PctBloatRsvdPages
-----
-----
dbo         authors   DATA  2.0        16.0       8.0           100.00
dbo         authors   INDEX  4.0        32.0       16.0          50.00

```

示例 9 显示当前数据库 (pubs2) 的事务日志的空间使用信息:

```
sp_spaceusage 'display', 'tranlog'
```

| TableName | TotalPages | UsedPages | CLRPages | FreePages | PctUsedPages | PctFreePages |
|-----------|------------|-----------|----------|-----------|--------------|--------------|
| syslogs | 4096.0 | 18.0 | 0.0 | 1482.0 | 0.43 | 36.18 |

示例 10 将当前数据库中 authors 表的空间使用信息存档到缺省表中 (对于表和索引, 缺省表为 spaceusage_object):

```
sp_spaceusage 'archive', 'table', 'authors'
```

```
Data was successfully archived into table 'pubs2.dbo.spaceusage_object'.
```

示例 11 将 authors 表的空间使用信息存档到 pubs3 数据库中的缺省表中 (对于表和索引, 缺省表为 spaceusage_object):

```
sp_spaceusage 'archive using dbname = pubs3', 'table', 'authors'
```

```
Data was successfully archived into table 'pubs3.dbo.spaceusage_object'.
```

示例 12 对 authors 表运行 update table statistics 并将其空间使用信息存档到当前数据库中前缀为 monday_ 的表 (此示例中为 monday_spaceusage_object) 中

```
1> sp_spaceusage 'archive using dbname = pubs2, prefix=monday_',
'table','authors', null, 'update table statistics'
```

示例 13 将当前数据库的事务日志的空间使用信息存档到 pubs3 数据库的缺省表 (对于事务日志, 缺省表为 spaceusage_tranlog) 中:

```
sp_spaceusage 'archive using dbname=pubs3', 'tranlog'
```

```
Data was successfully archived into table 'pubs3.dbo.spaceusage_tranlog'.
```

示例 14 根据当前数据库中的缺省表 (对于表或索引, 缺省表为 spaceusage_object), 详细报告上次存档的 authors 表的空间使用信息:

```
sp_spaceusage 'report', 'table', 'authors'
```

```
All the page counts in the result set are in the unit 'KB'.
```

```
All the data in the result set are dated 'Jun 15 2007 11:50PM'.
```

| OwnerName | TableName | IndId | NumRows | UsedPages | RsvdPages | ExtentUtil |
|--------------|-----------|-----------|----------|-----------|-----------|------------|
| ExprsvdPages | PctBloat | UsedPages | PctBloat | RsvdPages | | |
| dbo | authors | 0 | 23.0 | 4.0 | 32.0 | 12.50 |
| 16.0 | | 0.00 | | 100.00 | | |
| dbo | authors | 1 | NULL | 4.0 | 32.0 | 12.50 |
| 16.0 | | 0.00 | | 100.00 | | |
| dbo | authors | 2 | NULL | 4.0 | 32.0 | 12.50 |
| 16.0 | | 0.00 | | 100.00 | | |


```
(1 row affected)
(return status = 0)
```

示例 15 根据 pubs3 数据库的缺省表，报告上次存档的 authors 表的空间使用信息摘要：

```
sp_spaceusage 'report summary using dbname=pubs3', 'table', 'authors'

All the page counts in the result set are in the unit 'KB'.
All the data in the result set are dated 'Jan 17 2007 11:29AM'.
OwnerName  TableName  Type  UsedPages  RsvdPages  ExpRsvdPages  PctBloatRsvdPages
-----
dbo         authors   DATA  4.0        32.0       16.0          100.00
dbo         authors   INDEX  8.0        64.0       32.0          50.00
```

示例 16 根据当前数据库中的 monday_spaceusage_object 表，报告上次存档的 authors 表的空间使用信息摘要（以兆字节为单位）：

```
sp_spaceusage 'report summary using prefix=monday_, unit=MB', 'table',
'authors'

All the page counts in the result set are in the unit 'MB'.
All the data in the result set are dated 'Jan 17 2007 11:29AM'.
OwnerName  TableName  Type  UsedPages  RsvdPages  ExpRsvdPages  PctBloatRsvdPages
-----
dbo         authors   DATA  .00390625  .03125     .015625       100.00
dbo         authors   INDEX  .0078125   .0625      .03125        50.00
```

示例 17 根据当前数据库中的缺省表，报告在 2007 年 6 月 9 日或之后存档的、authors 表上所有索引的空间使用信息：

```
sp_spaceusage 'report', 'index', 'authors.%', null, null, null, 'Jun 9 2007'

All the page counts in the result set are in the unit 'KB'.
ArchiveDateTime  OwnerName  TableName  IndId  IndexName  UsedPages
RsvdPages  ExtentUtil  ExpRsvdPages  PctBloatUsedPages  PctBloatRsvdPag
es
-----
Jun  9 2007 12:06AM  dbo         authors   0        authors   4.0
32.0        12.50      16.0      0.00     100.00
Jun 10 2007 12:05AM  dbo         authors   0        authors   4.0
32.0        12.50      16.0      0.00     100.00
Jun 11 2007 11:35PM  dbo         authors   0        authors   4.0
32.0        12.50      16.0      0.00     100.00
Jun  9 2007 12:06AM  dbo         authors   1        auidind   4.0
32.0        12.50      16.0      0.00     100.00
```

```

Jun 10 2007 12:05AM  dbo          authors      1      auidind      4.0
    32.0          12.50          16.0          0.00  100.00
Jun 11 2007 11:35PM  dbo          authors      1      auidind      4.0
    32.0          12.50          16.0          0.00  100.
Jun  9 2007 12:06AM  dbo          authors      2      aunmind      4.0
    32.0          12.50          16.0          0.00  100.00
Jun 10 2007 12:05AM  dbo          authors      2      aunmind      4.0
    32.0          12.50          16.0          0.00  100.00
Jun 11 2007 11:35PM  dbo          authors      2      aunmind      4.0
    32.0          12.50          16.0          0.00  100.00

```

(1 row affected)
(return status = 0)

示例 18 根据当前数据库中的缺省表，报告在 2007 年 6 月 10 日与 2007 年 6 月 15 日之间存档的 authors 表的空间使用信息：

```

sp_spaceusage 'report', 'table', 'authors', null, null, null, 'Jun 10 2007',
'Jun 15 2007'

```

All the page counts in the result set are in the unit 'KB'.

| ArchiveDateTime | OwnerName | TableName | IndId | NumRows | UsedPages |
|---------------------|------------|--------------|----------|-----------|-------------------|
| RsvdPages | ExtentUtil | ExpRsvdPages | PctBloat | UsedPages | PctBloatRsvdPages |
| ----- | ----- | ----- | ----- | ----- | ----- |
| Jun 10 2007 12:05AM | dbo | authors | 0 | 23.0 | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 11 2007 11:35PM | dbo | authors | 0 | 23.0 | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 13 2007 11:46PM | dbo | authors | 0 | 23.0 | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 14 2007 11:46PM | dbo | authors | 0 | 23.0 | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 14 2007 11:46PM | dbo | authors | 0 | 23.0 | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 10 2007 12:05AM | dbo | authors | 1 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 11 2007 11:35PM | dbo | authors | 1 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 13 2007 11:46PM | dbo | authors | 1 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 14 2007 11:46PM | dbo | authors | 1 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 14 2007 11:46PM | dbo | authors | 1 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 10 2007 12:05AM | dbo | authors | 2 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |
| Jun 11 2007 11:35PM | dbo | authors | 2 | NULL | 4.0 |
| 32.0 | 12.50 | 16.0 | 0.00 | | 100.00 |

```

Jun 13 2007 11:46PM dbo authors 2 NULL 4.0
32.0 12.50 16.0 0.00 100.00
Jun 14 2007 11:46PM dbo authors 2 NULL 4.0
32.0 12.50 16.0 0.00 100.00
Jun 14 2007 11:46PM dbo authors 2 NULL 4.0
32.0 12.50 16.0 0.00 100.00

```

```

(1 row affected)
(return status = 0)

```

用法

- `sp_spaceusage` 提供当前数据库的表、索引和事务日志的空间使用信息。
- `sp_spaceusage` 输出中显示的列集合取决于操作和实体类型。缺省情况下，只显示一组标准列。但是，使用 `select_list` 参数可以包括其他列，在 `select` 列表中使用 * 通配符可以查看所有这些列。表 1-31 和表 1-32 分别为实体类型 `table`、`index` 和 `tranlog` 提供了所有输出列名称及其说明的集合。`select list`、`where clause`、`orderby clause` 参数中的列名必须属于这些表中列出的集合。

表 1-31: table 或 index 实体类型的输出列

| 列名 | 说明 |
|-----------------|--------------------------|
| ArchiveDateTime | 数据的时间戳 |
| ServerName | 服务器名 |
| MaxPageSize | @@maxpagesize 中的逻辑页大小 |
| DBName | 对象的数据库名称 |
| OwnerName | 对象的所有者名称 |
| TableName | 表名 |
| Id | 表的 ID |
| IndId | 索引的 ID |
| IndexName | 索引名 |
| PtnId | 分区的 ID |
| PtnName | 分区名称 |
| DataPtnID | 索引涵盖其数据的数据分区的 ID |
| RowSize | 分区中的行数 |
| RowCount_ts | systabstats 表中的此分区所具有的行数 |
| NumFwdRows | 分区中转移的行数 |
| NumDelRows | 分区中删除的行数 |
| PctFwdRows | 分区中转移的行所占的百分比 |
| NonLeafRowSize | 分区中非叶行的平均行宽 |
| FF | 分区中的填充因子 |
| MRPP | 分区中每页的最大行数 |

| 列名 | 说明 |
|-------------------|--------------------------------|
| ERS | 分区中的期望行宽 |
| RPG | 分区中的保留页间距 |
| IndexHeight | 分区中索引树的高度 |
| OAMAPageCount | OAM 和 AP 页数 (以页为单位) |
| Extent0PageCount | 扩充 0 中的页数 (以页为单位) |
| Status | sysindexes 表的 status |
| Sysstat | sysobjects 表的 sysstat |
| Sysstat2 | sysobjects 表的 sysstat2 |
| LockScheme | 表的锁定方案 |
| NumVarCols | 表包含的变量列数 |
| HasAPLCl | 指示表是否具有 APL Cl |
| SpUtil | 分区的空间利用派生的统计信息 |
| DPCR | 分区的数据页集群比派生的统计信息 |
| DRCR | 分区的数据行集群比派生的统计信息 |
| IPCR | 分区的索引页集群比派生的统计信息 |
| LGIO | 分区的大 IO 效率派生的统计信息 |
| ExtentUtil | 分区的扩充利用 |
| EmptyPages | 分区中的空页数 (以单元表示) |
| DataPages | 分区中的数据页数 (以单元表示) |
| UsedPages | 分区中的已用页数 (以单元表示) |
| RsvdPages | 分区中的保留页数 (以单元表示) |
| LeafPages | 分区中的叶页数 (以单元表示) |
| ExpDataPages | 在已压缩数据的情况下, 分区中的期望数据页数 (以单元表示) |
| ExpUsedPages | 在已压缩数据的情况下, 分区中的期望已用页数 (以单元表示) |
| ExpRsvdPages | 在已压缩数据的情况下, 分区中的期望保留页数 (以单元表示) |
| ExpLeafPages | 在已压缩数据的情况下, 分区中的索引树的期望高度 |
| PctBloatUsedPages | 分区中的已用页的增长百分比 |
| PctBloatRsvdPages | 分区中的保留页的增长百分比 |
| PctBloatLeafPages | 分区中叶页的增长百分比 |
| PctEmptyPages | 分区中空数据页所占百分比 |

表 1-32: 可用于 tranlog 实体类型的参数

| 列名 | 说明 |
|-----------------|---------------------------|
| ArchiveDateTime | 数据的时间戳 |
| ServerName | 服务器名 |
| MaxPageSize | 在 @@maxpagesize 中设置的逻辑页大小 |
| DBName | 对象的数据库名称 |

| 列名 | 说明 |
|--------------|----------------------|
| OwnerName | 对象的所有者名称 |
| TableName | 事务日志的名称, 例如 syslogs |
| Id | syslogs 表的 ID |
| IsMLD | 是否为“混合的日志和数据”事务日志? |
| IsLogFull | 事务日志是否已满? |
| LCTPages | 日志的“最后机会阈值”值 (以单元表示) |
| TotalPages | 日志总页数 (以单元表示) |
| UsedPages | 已用日志页数 (以单元表示) |
| CLRPages | 为回退保留的日志页数 (以单元表示) |
| FreePages | 尚未使用的日志页数 (以单元表示) |
| PctUsedPages | 已用日志页所占百分比 |
| PctFreePages | 空闲日志页所占百分比 |

- PctBloatUsedPages 和 PctBloatReservedPages 列分别估计表使用和保留的页数比最小值多多少。这些值表示对表运行 reorg rebuild 对您有多大好处。

ExtentUtil 是实际使用的页数与为对象保留的页数之间的比值。如果值接近 100, 则表示扩充中为对象保留的大多数页当前都已使用。

表 1-33 给出了测量的对照表。

表 1-33: 解释 PctBloatUsedPages 和 PctBloatReservedPages 值

| PctBloatUsed — Pages 值 | PctBloatReserved — Pages 值 | 解释 |
|------------------------|----------------------------|--|
| 接近于 0, 较小值 | 接近于 0, 较小值 | 表示表压缩良好, 所有已分配页和分配单元均已完全使用。ExtentUtil 应接近于 1.0 |
| 接近于 0, 较小值 | 不接近于 0, 较大值 | 表示已用页压缩良好, 但表的扩充未充分使用, 并且存在大量页间碎片, 可能是由于大量的删除操作或空页引起的。systabstats 中的 unusedpgcnt 很可能也很大。PctBloatReservedPages 的值较大表明 ExtentUtil 很可能远小于 1.0。通过运行 reorg rebuild 或许能够解决大多数问题。 |
| 不接近于 0, 较大值 | 接近于 0, 较小值 | 表示存在大量页内分段, 但页间分段不多。由于扩充利用率可能很高, ExtentUtil 值应接近于 1.0。 运行 reorg compact 可能会帮助解决这些问题。 |
| 不接近于 0, 较大值 | 不接近于 0, 较大值 | PctBloatUsedPages 的值较大表示存在大量页内分段, 已用页中的数据行未完全压缩 (已用页包含的大部分是可用空间)。因为页间和页内分段可能会导致 PctBloatReservedPages 的值较大, 所以 Extent_Util 的值可能仍小于 1.0。运行 reorg compact 和 reorg rebuild 可能会解决这些问题。 |

- 在其中存档空间使用数据的数据库必须已启用 `sp_dboption ... select into`。
- 如果在存档时不存在存档表，则会创建存档表，否则结果将附加到当前表中。因此，运行 `sp_spaceusage` 的所有用户都必须在存档数据库中拥有 `create table` 权限。
- 在存档或报告数据时，只考虑将运行 `sp_spaceusage` 的用户所拥有的表视为存档表。忽略其他用户拥有的同名表。缺省情况下，对于表或索引，结果会存档至 `spaceusage_object` 表或根据该表报告结果；对于事务日志，结果会存档至 `spaceusage_tranlog` 或根据它报告结果。
- 在根据存档数据进行报告时，只能将 `from_date` 和 `to_date` 参数用于 `report` 操作。生成报告时，Adaptive Server 只使用存档表中指定时间范围内的数据。如果不包括 `from_date` 或 `NULL`，Adaptive Server 将使用 `to_date` 之前的所有存档数据。如果不包括 `to_date` 或 `NULL`，Adaptive Server 将使用当前日期作为 `to_date` 的值。如果既不包括 `from_date`，也不包括 `to_date`，Adaptive Server 将使用存档表中的最新数据生成报告。
- `sp_spaceusage` 结果是根据统计数据估计的。这些估计的精确性取决于所提供的统计信息。可运行 `update statistics` 来提高结果的精确性。

权限

任何用户都可以运行 `sp_spaceusage`。但是，他们可能无法查看其无权查看的表的某些信息。

sp_spaceused

说明 显示指定的表或当前数据库中的所有表使用的行数、数据页数、索引大小和空间的估计值。

语法 sp_spaceused [*objname* [,1]]

参数 *objname*

是表的名称，就是要报告该表的情况。如果省略，将显示当前数据库中使用的空间的摘要。

1

分别输出有关表的索引和 **text/image** 存储的信息。

示例 1 报告为 **titles** 表分配（保留）的空间量，用于数据的空间量，用于索引的空间量和可用（未使用的）空间量：

```
sp_spaceused titles
name          rowtotal  reserved  data    index_size  unused
-----
titles        18         46 KB    6 KB   4 KB        36 KB
```

示例 2 除有关 **titles** 表的信息外，还输出该表上各个索引的信息：

```
sp_spaceused titles, 1
index_name    size      reserved  unused
-----
titleidind    2 KB     32 KB    24 KB
titleind      2 KB     16 KB    14 KB

name          rowtotal  reserved  data    index_size  unused
-----
titles        18         46 KB    6 KB   4 KB        36 KB
```

示例 3 分别显示 **text/image** 页存储使用的空间和该表使用的空间。**text/image** 存储的对象名称为“t”加上表的名称：

```
sp_spaceused blurbs,1
index_name    size      reserved  unused
-----
blurbs        0 KB     14 KB    12 KB
tblurbs       14 KB    16 KB    2 KB

name          rowtotal  reserved  data    index_size  unused
-----
blurbs        6         30 KB    2 KB   14 KB       14 KB
```

示例 4 输出当前数据库中使用的空间的摘要：

```
sp_spaceused
database_name  database_size
-----
master          5 MB
reserved      data          index_size  unused
-----
2176 KB       1374 KB       72 KB       730 KB
```

示例 5 报告为事务日志保留的空间量和可供事务日志使用的空间量：

```
sp_spaceused syslogs
name          rowtotal  reserved  data      index_size  unused
-----
syslogs      Not avail. 32 KB     32 KB     0 KB        0 KB
```

用法

- **sp_spaceused** 显示指定的表或当前数据库中的所有表使用的数据页数和空间的估计值，以及表中行数的估计值。**sp_spaceused** 使用 **rowcnt** 内置函数计算 **rowtotal** 值。该函数使用的每个数据页平均行数基于该对象的分配页中的值。该方法非常快，但结果为估计值，而且更新和插入活动会更改实际值。**update statistics** 命令、**dbcc checktable** 和 **dbcc checkdb** 会更新每页行数的估计值，因此，在执行了这些命令中的一个命令之后 **rowtotal** 最准确。如果需要精确的行数，应使用 **select count(*)**。
- **sp_spaceused** 报告受表、聚簇索引和非聚簇索引影响的空间量。
- **sp_spaceused** 报告的分配（保留）空间量是数据、索引大小和可用（未使用的）空间的总和。
- **text** 和 **image** 列（作为单独的数据库对象存储）使用的空间在 **index_size** 列中单独报告，并且包括在表的摘要行中。**index_size** 列中 **text/image** 存储的对象名称为“t”加上表的名称。
- 对 **syslogs** 使用时，**sp_spaceused** 报告的 **rowtotal** 为“Not available”。请参见示例 5。

权限

任何用户都可以执行 **sp_spaceused**。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

目录存储过程 [sp_statistics](#)

命令 [create index](#)、[create table](#)、[drop index](#)、[drop table](#)

系统过程 [sp_helpindex](#)

sp_ssladmin

| | |
|----|---|
| 说明 | 添加、删除或显示 Adaptive Server 的服务器认证列表。 |
| 语法 | <pre>sp_ssladmin {[addcert, <i>certificate_path</i> [, <i>password</i> NULL]] [dropcert, <i>certificate_path</i>] [lscert] [help]} [lsciphers] [setciphers, {"FIPS" "Strong" "Weak" "All" <i>quoted_list_of_ciphersuites</i>}]</pre> |
| 参数 | <p>addcert 在认证文件中为本地服务器添加认证。</p> <p><i>certificate_path</i> 指定本地服务器上认证文件的绝对路径。</p> <p><i>password</i> 将新服务器认证添加到认证文件中时，用于加密私有密钥的口令。</p> <p>NULL 用于通过在启动期间从命令行请求口令，要求以有人看管的方式启动 Adaptive Server。</p> <p>dropcert 从认证文件中删除认证。</p> <p>lscert 列出认证文件中的认证。</p> <p>help 显示 sp_ssladmin 的联机帮助。</p> <p>lsciphers 显示任何密码成套程序优先选项集的值。</p> |

setciphers, {"FIPS" | "Strong" | "Weak" | "All" | *quoted_list_of_ciphersuites*};
 设置特定的密码成套程序优先选项。选择以下选项之一：

- “FIPS” — 是符合 FIPS 要求的加密算法、散列算法和密钥交换算法的集合。此列表包括的算法有 AES、3DES、DES 和 SHA1。
- “Strong” — 是所用密钥长度大于 64 位的加密算法的集合。
- “Weak” — 是属于所有受支持的密码成套程序的集合但又不属于 strong 集合的加密算法的集合。
- “All” — 是缺省密码成套程序的集合。
- *quoted_list_of_ciphersuites* — 以逗号分隔的列表形式指定密码成套程序集，列表中各项按优先顺序排列。该列表首尾用引号（“ ”）引起来。此带引号的列表可以包含任何预定义集合以及各个密码成套程序名称。未知的密码成套程序名称将导致报告错误，但对优先选项无影响。有关定义的集合中包含的密码成套程序的列表，请参见《系统管理指南》中的第 19 章“数据保密性”。

示例

示例 1 在 `/sybase/ASE-12_5/certificates` 的绝对路径（在 Windows 上为 `x:\sybase\ASE-12_5\certificates`）下的认证文件中为本地服务器添加一个条目，即 `Server1.crt`。私有密钥用口令“`mypassword`”进行加密。该口令应该是创建私有密钥时所指定的那个口令：

```
sp_ssladmin addcert, "/sybase/ASE-12_5/certificates/Server1.crt",
"mypassword"
```

示例 2 从位于 `/sybase/ASE-12_5/certificates`（在 Windows 上为 `x:\sybase\ASE-12_5\certificates`）下的认证文件中删除认证 `Server1.crt`：

```
sp_ssladmin dropcert , "/sybase/ASE-12_5/certificates/Server1.crt"
```

示例 3 列出本地服务器上的所有服务器认证：

```
sp_ssladmin lscert
go

certificate_path
-----
/sybase/ASE-12_5/certificates/Server1.crt
```

示例 4 最初启动时，在设置任何密码成套程序优先选项前，`sp_ssladmin lscipher` 不显示任何优先选项。

```
1> sp_ssladmin lscipher
2> go

Cipher Suite Name Preference
-----
(0 rows affected)
```

```
(return status = 0)
```

下面的示例指定使用 FIPS 算法的密码成套程序的集合：

```
1> sp_ssladmin setcipher, 'FIPS'  
2> go
```

sp_ssladmin 输出的优先选项为 0（零）表示 Adaptive Server 未使用密码成套程序。如果输出的是其它非零值，则该数值表示 Adaptive Server 在 SSL 握手过程中使用该算法的优先顺序。SSL 握手的客户端选择这些密码成套程序中与其接受的密码成套程序的列表匹配的一个密码成套程序。

示例 5 使用带引号的密码成套程序列表来设置 Adaptive Server 中的优先选项：

```
1> sp_ssladmin setcipher, 'TLS_RSA_WITH_AES_128_CBC_SHA,  
TLS_RSA_WITH_AES_256_CBC_SHA'  
2> go
```

用法

- Adaptive Server 监听器必须为客户端提供一个认证。认证中的公用名必须与 *interfaces* 文件中客户端使用的公用名匹配。如果不匹配，服务器鉴定和登录将失败。
- 如果将口令指定为 NULL，必须用 *-y* 标志启动 *dataserver*。该标志将提示管理员在命令行上输入私有密钥口令。
- 使用 NULL 作为口令是为了在 SSL 加密会话开始之前的 SSL 初始配置过程中保护口令。

在建立了 SSL 连接并且重新启动了 Adaptive Server 后，再次使用 *sp_ssladmin*，这一次使用实际口令。这时，Adaptive Server 将加密并存储口令。以后从命令行启动 Adaptive Server 时都会使用加密的口令；启动期间不必在命令行上指定口令。

- 可在 *interfaces* 文件（在 Windows 上为 *sql.ini*）中将 *hostname* 指定为 “localhost” 以防止客户端远程连接。只能建立本地连接，而且永远不会通过网络连接传送口令。

使用 *Isciphers* 和 *setciphers* 设置密码成套程序

通过 *Isciphers* 和 *setciphers* 选项可以限制 Adaptive Server 使用的密码成套程序集，这样，系统安全员便能够控制客户端到服务器的连接或 Adaptive Server 的外发连接可使用的加密算法的种类。缺省情况下，Adaptive Server 使用内部定义的密码成套程序优先选项集。有关详细信息，请参见《系统管理指南》中的第 19 章“数据保密性”。

`sp_ssladmin setciphers` 将密码成套程序优先选项设置为给定的有序列表。这样可将可用的 SSL 密码成套程序限制为由 "FIPS"、"Strong"、"Weak"、"All" 组成的指定集合或带引号的密码成套程序列表。此设置会在下一个监听器启动时生效，并且需要您重新启动 Adaptive Server 以确保所有监听器都使用新设置。

您可以显示已经使用 `sp_ssladmin lsciphers` 设置的任何密码成套程序优先选项。如果没有设置任何优先选项，`sp_ssladmin lsciphers` 将返回 0 行，表示未设置任何优先选项，并且 Adaptive Server 将使用其缺省（内部）优先选项。

权限

必须具有系统安全员角色才能使用 `sp_ssladmin`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

文档 《系统管理指南》中的第 19 章“数据保密性”。

sp_syntax

| | |
|----|--|
| 说明 | 显示 Transact-SQL 语句、系统过程、实用程序和 Adaptive Server 的其它例程的语法，具体取决于服务器上有哪些产品和相应的 sp_syntax 脚本。 |
| 语法 | <code>sp_syntax word [, mod][, language]</code> |
| 参数 | <p>word 是命令或例程的名称或部分名称；例如，指定 “help” 将列出所有提供帮助的系统过程。若要包括空格或 Transact-SQL 保留字，请将其用引号引起。</p> <p>mod 是模块的名称或部分名称，例如，“Transact-SQL” 或 “Utility”。各个 sp_syntax 安装脚本会添加不同的模块。使用不带参数的 sp_syntax 可以查看服务器上有哪些模块。</p> <p>language 是要检索的语法说明的语言。language 必须为 syslanguages 表中的有效名称。</p> |
| 示例 | <p>示例 1 显示服务器上所有可用的 sp_syntax 模块：</p> <pre> sp_syntax sp_syntax provides syntax help for Sybase products. These modules are installed on this Server: Module ----- OpenVMS Transact-SQL UNIX Utility System Procedure Usage: sp_syntax command [, module [, language]] </pre> <p>示例 2 显示包含单词或单词片断 “disk” 的所有例程的语法和功能说明。由于 “disk” 是 Transact-SQL 保留字，因此应使用引号将其引起来：</p> <pre> sp_syntax "disk" </pre> |
| 用法 | <ul style="list-style-type: none"> sp_syntax 的文本在数据库 sybsyntax 中。请使用适用于所用平台的配置文档中所述的安装脚本将 sp_syntax 和 sybsyntax 数据库装载到 Adaptive Server 上。如果您无法访问 sp_syntax，请向系统管理员咨询如何将其安装到您的服务器上。 |

- 可以在要搜索的命令名称中使用通配符。但是，如果要查找包含字符“_”的命令或函数，可能会得到意外的结果，因为下划线通配符表示任意单个字符。

权限 任何用户都可以执行 `sp_syntax`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

使用的表 `sybsyntax..sybsyntax`

另请参见 **系统过程** [sp_helpdb](#)

sp_sysmon

说明 显示性能信息。

语法

```
sp_sysmon begin_sample
sp_sysmon { end_sample | interval }[, section[, applmon] ]
           [, 'cache wizard' [, top_N [, filter] ]]
```

参数 begin_sample

开始采样。如果指定了 begin_sample，则不能指定部分。

section

是 sp_sysmon 输出的部分中某一个部分的缩写。报告部分的值和相应名称为：

| 参数 | 报告部分 |
|-----------|---------------|
| apppgmt | 应用程序管理 |
| dcache | 数据高速缓存管理 |
| diskio | 磁盘 I/O 管理 |
| esp | ESP 管理 |
| indexmgmt | 索引管理 |
| kernel | 内核使用率 |
| locks | 锁管理 |
| memory | 内存管理 |
| mdcache | 元数据高速缓存管理 |
| monaccess | 对执行 SQL 的监控访问 |
| netio | 网络 I/O 管理 |
| parallel | 并行查询管理 |
| pcache | 过程高速缓存管理 |
| recovery | 恢复管理 |
| taskmgmt | 任务管理 |
| xactmgmt | 事务管理 |
| xactsum | 事务配置文件 |
| wpm | 工作进程管理 |

applmon

指定输出应用程序细节、应用程序和登录名细节或无应用程序细节。缺省设置为省略应用程序细节。有效值及其报告的信息为：

- **appl_only** — 按应用程序名输出的 CPU、I/O、优先级更改和资源限制违反信息。
- **appl_and_login** — 按应用程序名和登录名输出的 CPU、I/O、优先级更改和资源限制违反信息。
- **no_appl** — 跳过报告的“按应用程序”或“按登录名”部分。这是缺省值。

只有在输出完整报告并且为 *section* 指定了 **appmgmt** 时此参数才有效。

end_sample

结束采样并输出报告。

interval

指定采样的时间范围。必须采用 HH:MM:SS 格式，例如“00:20:00”。

'cache wizard'

可帮助监控和配置数据高速缓存以获得最佳性能。

top_N

是一种 **varchar** 数据类型，它根据指定间隔内的逻辑读取次数（显示在 LR/sec 列中）的排列条件限制在“对象部分”报告的对象列表。

根据指定值是正整数还是负整数，排列顺序为升序或降序。通过指定值“0”，可以获取在间隔结束时占用高速缓存的对象的整个列表。缺省值为 10。

filter

是一种 **varchar** 数据类型，用来为报告中包括的高速缓存指定模式。

例如，如果将它指定为 **default data cache**，报告将只包含有关缺省数据高速缓存的信息。如果将它指定为 **emp%**，输出将包括名称与此模式匹配的所有高速缓存的有关信息。

如果未指定任何值，输出将包含所有高速缓存，且首先显示缺省数据高速缓存，然后按字母顺序显示其它高速缓存。

示例**示例 1** 在 10 分钟后输出监控信息：

```
sp_sysmon "00:10:00"
```

示例 2 在 5 分钟后只输出 **sp_sysmon** 报告的“磁盘管理”部分：

```
sp_sysmon "00:05:00", diskio
```

示例 3 开始采样，执行过程和查询，结束采样，然后只输出报告的“数据高速缓存”部分：

```
sp_sysmon begin_sample
go
execute proc1
go
execute proc2
go
select sum(total_sales) from titles
go
sp_sysmon end_sample, dcache
go
```

示例 4 输出完整报告，并且包括各个登录名的应用程序和登录名详细信息：

```
sp_sysmon "00:05:00", @applmon = appl_and_login
```

示例 5 不清除计数器的报告用法：

```
sp_sysmon "00:01:00", kernel, noclear
```

也可以使用：

```
sp_sysmon "00:01:00", noclear
```

注释 仅当在 `sp_sysmon` 中指定了采样间隔时，才能使用 `noclear` 参数。如果指定 `begin_sample` 或 `end_sample`，则不能使用 `noclear`。

示例 6 使用高速缓存向导输出报告：

```
sp_sysmon '00:00:30', 'cache wizard'
```

```
=====
Cache Wizard
=====

-----
default data cache
-----

Run Size      :   100.00 Mb   Usage%       :           2.86
LR/sec       :    41.10   PR/sec        :    22.57   Hit%:   45.09
Cache Partitions:    4       Spinlock Contention%:    0.00
Buffer Pool Information
-----

IO Size Wash Size Run Size   APF%   LR/sec   PR/sec   Hit%   APF-Eff% Usage%
-----
```

```

4 Kb      3276 Kb    16.00 Mb  10.00    0.47    0.13  71.43    n/a      0.20
2 Kb      17200 Kb    84.00 Mb  10.00    40.63   22.43  44.79    n/a      3.37

```

(1 row affected)

Object Statistics

```

-----
Object                                     LR/sec  PR/sec  Hit%   Obj_Cached%  Cache_Occp%
-----
empdb.dbo.t1                             0.57    0.30   47.06   56.25        0.02
empdb.dbo.t2                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t3                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t4                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t5                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t6                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t8                             0.30    0.30    0.00   56.25        0.02
empdb.dbo.t7                             0.57    0.20  64.71   62.50        0.02
tempdb.dbo.tempcachedobjstats            3.63    0.00 100.00   50.00        0.01
tempdb.dbo.tempobjstats                  0.47    0.00 100.00   25.00        0.00

```

```

-----
Object                                     Obj Size  Size in Cache
-----
empdb.dbo.t1                             32 Kb    18 Kb
empdb.dbo.t2                             32 Kb    18 Kb
empdb.dbo.t3                             32 Kb    18 Kb
empdb.dbo.t4                             32 Kb    18 Kb
empdb.dbo.t5                             32 Kb    18 Kb
empdb.dbo.t6                             32 Kb    18 Kb
empdb.dbo.t8                             32 Kb    18 Kb
empdb.dbo.t7                             32 Kb    20 Kb
tempdb.dbo.tempcachedobjstats            16 Kb     8 Kb
tempdb.dbo.tempobjstats                  16 Kb     4 Kb

```

company_cache

```

Run Size      : 1.00 Mb  Usage%      : 0.39
LR/sec        : 0.07    PR/sec      : 0.07    Hit%: 0.00
Cache Partitions: 1    Spinlock Contention%: 0.00

```

Buffer Pool Information

```

-----
IO Size Wash Size  Run Size  APF%  LR/sec  PR/sec  Hit%  APF-Eff%  Usage%
-----
2 Kb      204 Kb    1.00 Mb  10.00  0.07   0.07   0.00   n/a      0.39

```

Object Statistics

```

-----
Object                                     LR/sec  PR/sec  Hit%   Obj_Cached%  Cache_Occp%
-----

```

empdb.dbo.history 0.07 0.07 0.00 25.00 0.39

| Object | Obj Size | Size in Cache |
|-------------------|----------|---------------|
| empdb.dbo.history | 16 Kb | 4 Kb |

companydb_cache

| | | | | | | |
|-------------------|---|---------|-----------------------|---|--------|-------------|
| Run Size | : | 5.00 Mb | Usage% | : | 100.00 | |
| LR/sec | : | 380.97 | PR/sec | : | 56.67 | Hit%: 85.13 |
| Cache Partitions: | | 1 | Spinlock Contention%: | | 0.00 | |

Buffer Pool Information

| IO Size | Wash Size | Run Size | APF% | LR/sec | PR/sec | Hit% | APF-Eff% | Usage% |
|---------|-----------|----------|-------|--------|--------|-------|----------|--------|
| 2 Kb | 1024 Kb | 5.00 Mb | 10.00 | 380.97 | 56.67 | 85.13 | 98.42 | 100.00 |

Object Statistics

| Object | LR/sec | PR/sec | Hit% | Obj_Cached% | Cache_Occp% |
|-----------------------------|--------|--------|-------|-------------|-------------|
| company_db.dbo.emp_projects | 41.07 | 22.80 | 44.48 | 19.64 | 9.45 |
| company_db.dbo.dept_det | 93.03 | 20.67 | 77.79 | 99.08 | 54.53 |
| company_db.dbo.emp_perf | 116.70 | 2.63 | 97.74 | 97.77 | 34.18 |
| company_db.dbo.dept_locs | 0.43 | 0.17 | 61.54 | 50.00 | 0.16 |

| Object | Obj Size | Size in Cache |
|-----------------------------|----------|---------------|
| company_db.dbo.emp_projects | 2464 Kb | 484 Kb |
| company_db.dbo.dept_det | 2818 Kb | 2792 Kb |
| company_db.dbo.emp_perf | 1790 Kb | 1750 Kb |
| company_db.dbo.dept_locs | 16 Kb | 8 Kb |

TUNING RECOMMENDATIONS

Usage% for 'default data cache' is low (< 5%)
Usage% for 4k buffer pool in cache:default data cache is low (< 5%)
Usage% for 2k buffer pool in cache:default data cache is low (< 5%)

Usage% for 'company_cache' is low (< 5%)
Usage% for 2k buffer pool in cache:company_cache is low (< 5%)
Consider adding a large I/O pool for 'companydb_cache'

- 用法
- 在 Adaptive Server 15.0.1 版和更高版本中，`sp_sysmon` 的缺省行为是不清除监控计数器。
如果需要清除监控计数器，请使用 `sp_sysmon` 和 `clear` 选项。出于兼容性考虑，Adaptive Server 将 `noclear` 选项作为有效参数接受，但它不影响 `sp_sysmon` 的行为。
然而，如果您使用 `begin_sample` 和 `end_sample` 选项来运行 `sp_sysmon` 以开始和结束采样周期，则 `sp_sysmon` 始终会清除监控计数器。若使用 `begin_sample` 或 `end_sample` 及 `noclear` 选项运行 `sp_sysmon`，则 Adaptive Server 将发出错误消息。
 - `sp_sysmon` 显示有关 Adaptive Server 性能的信息。它将内部计数器设置为 0，然后等待指定的时间间隔，与此同时，服务器上的活动将导致计数器增大。间隔结束后，`sp_sysmon` 将根据计数器中的值输出信息。有关详细信息，请参见《性能和调优指南》。
 - 若要只输出报告的某一部分，可使用 `sp_sysmon applmon` 中的有效值。
 - 如果以批处理模式使用 `sp_sysmon`（使用 `begin_sample` 和 `end_sample`），执行之间的时间间隔必须至少为一秒。可以使用 `waitfor delay "00:00:01"` 来延长批处理的执行时间。
 - 在采样间隔期间，结果以带符号的整数值存储。特别是在具有多个 CPU 且活动较频繁的系统上，这些计数器可能会溢出。如果 `sp_sysmon` 输出的结果中有负数，请缩短采样时间。

权限 只有系统管理员才能执行 `sp_sysmon`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 文档 《性能和调优指南》中的“Monitoring performance with `sp_sysmon`”（使用 `sp_sysmon` 监控性能）。

sp_tab_suspectptn

| | |
|------|--|
| 说明 | 列出具有可疑分区的表。基于字符的分区键的域分区表会在排序顺序更改后变为可疑，散列分区表会在跨平台转储装载后变为可疑。 |
| 语法 | <code>sp_tab_suspectptn [table_name]</code> |
| 参数 | <i>table_name</i> 是包含可疑分区的表的名称。 |
| 用法 | 如果： <ul style="list-style-type: none">• 提供表名 — 则 Adaptive Server 只检查由 <i>table_name</i> 指定的表。• 不提供表名 — 则 Adaptive Server 会检查当前数据库中的所有表。 |
| 权限 | 任何用户都可以使用 <code>sp_tab_suspectptn</code> 。 |
| 另请参见 | 存储过程 sp_indsuspect |

sp_tempdb

说明

sp_tempdb 允许用户：

- 创建和管理临时数据库组。
- 将用户或应用程序绑定到 **default** 或其它临时数据库组，或绑定到特定的本地临时数据库。
- 管理到本地临时数据库和临时数据库组的绑定。

这些绑定存储在 **master** 数据库中的 **sysattributes** 表中。

sp_tempdb 提供了用于维护 **sysattributes** 中与多个临时数据库相关的绑定的绑定接口。

语法

```
sp_tempdb [
    [ { "create" | "drop" }, "groupname" ] |
    [ { "add" | "remove" }, "tempdbname", "groupname" ] |
    [ { "bind", "objtype", "objname", "bindtype", "bindobj"
      [, "scope", "hardness" ] } |
      { "unbind", "objtype", "objname" [, "scope" ] "instance_name" } ] |
    [ "unbindall_db", "tempdbname" ] |
    [ show [, "all" | "gr" | "db" | "login" | "app" [, "name" ] ] |
    [ who, "dbname" ]
    [ help ] ]
```

参数

create

创建 **default** 临时数据库组。

drop

删除临时数据库组。

groupname

是临时数据库组的名称。

add

将临时数据库添加到 **default** 临时数据库组中。

remove

从 **default** 临时数据库组中删除临时数据库。

tempdbname

是要添加或删除的临时数据库的名称。对于 Cluster Edition，**tempdbname** 必须为本地用户临时数据库。

bind

将登录名和应用程序绑定到临时数据库或 **default** 临时数据库组。

unbind

解除登录名和应用程序与临时数据库或 **default** 临时数据库组的绑定。

objtype

是对象类型。有效值包括：

- login_name （或 LG）
- application_name （或 AP）

值不区分大小写。

objname

是要绑定或解除绑定的对象的名称。

bindtype

是绑定类型。有效值包括：

- group （或 GR）
- database （或 DB）

值不区分大小写。

bindobj

是要绑定的对象的名称。视 *bindtype* 而定，该对象为一个组或一个数据库。

scope

NULL。

instance_name

在集群环境中— 是拥有要解除绑定的本地临时数据库的实例的名称。此选项仅适用于 Cluster Edition。

hardness

hardness — 为 hard、soft 或 NULL。缺省值为 soft。

将 *hardness* 的值设置为 hard 时，如果未能按照绑定指派临时数据库，将导致登录失败。

将其值设置为 soft 时，此类故障会导致从缺省组中指派临时数据库或指派本地系统临时数据库。

unbindall_db

删除给定的临时数据库的所有登录名绑定和应用程序绑定。它不会删除任何“数据库到组”成员关系。使用此选项时，需要指定 *tempdbname* 变量。

此操作不会影响活动会话的现有指派。

show

显示 `sysattributes` 表中存储的有关现有组、组成员、登录帐户和应用程序绑定和指派到给定数据库的活动会话的信息。其值为：

- **all** 或不带参数 — 显示 `default` 临时数据库组，所有“数据库与组”的成员关系，以及所有登录名和应用程序绑定。
- **gr** — 显示 `default` 临时数据库组。无论是否将 `name` 选项指定为“`default`”，`sp_tempdb show` 都会显示所有绑定到 `default` 临时数据库组的临时数据库。
- **db** — 显示所有“数据库和临时数据库与组的”成员关系。如果提供了 `name`，则只会输出数据库 `name` 的“数据库到组”成员关系。
- **login** — 显示登录名不为 `NULL` 的所有登录名绑定。如果提供了 `name`，则只会输出登录名 `name` 的绑定。
- **app** — 显示所有应用程序不为 `NULL` 的绑定。如果提供了 `name`，则只会输出应用程序 `name` 的绑定。

注释 `tempdb` 始终属于 `default` 数据库组。

who

显示所有指派到给定临时数据库的活动会话。使用 `who` 参数时，必须使用以下参数：

- **dbname** — 是临时数据库的名称。如果提供的 `dbname` 不是临时数据库名称，`sp_tempdb who` 虽然可以执行，但不会报告任何绑定到该数据库的活动会话。

如果 `system_view` 设置为 `cluster`，则会检查集群的所有活动会话。如果 `system_view` 设置为 `instance`，则会检查当前实例上处于活动状态的会话

可以从集群中的任何实例上执行此命令。

help

显示用法信息。执行 `sp_tempdb` 而不指定命令与执行 `sp_tempdb "help"` 是等效的。

示例

示例 1 将 `mytempdb1` 添加到 `default` 组：

```
sp_tempdb "add", "mytempdb1", "default"
```

示例 2 从 `default` 组中删除 `mytempdb1`：

```
sp_tempdb "remove", "mytempdb1", "default"
```

示例 3 将登录名 “sa” 绑定到 default 组：

```
sp_tempdb "bind", "lg", "sa", "GR", "default"
```

在此示例中，objtype 的值为 login_name。可以将 login_name 替换为 lg 或 LG。

在此示例中，bindtype 的值为 group。可将 group 替换为 gr 或 GR。

示例 4 将上例中登录名 “sa” 的绑定从 default 组更改为 mytempdb1：

```
sp_tempdb "bind", "lg", "sa", "DB", "mytempdb1"
```

在此示例中，bindtype 的值为 database。可将 database 替换为 db 或 DB。

示例 5 将 isql 绑定到 mytempdb1：

```
sp_tempdb "bind", "ap", "isql", "DB", "mytempdb1"
```

在此示例中，objtype 的值为 application_name。可将 application_name 替换为 ap 或 AP。

示例 6 将上例中 isql 的绑定从 mytempdb1 更改为 default 组：

```
sp_tempdb "bind", "ap", "isql", "GR", "default"
```

示例 7 删除登录名 “sa” 和应用程序 “isql” 的绑定。

```
sp_tempdb "unbind", "lg", "sa"
sp_tempdb "unbind", "ap", "isql"
```

示例 8 删除 mytempdb1 数据库的所有登录名和应用程序绑定：

```
sp_tempdb "unbindall_db", "mytempdb1"
```

示例 9 演示 sp_temp show 命令。我们选择了该命令的一些不同用法，并显示了经删节的示例输出。

```
sp_tempdb show

Temporary Database Groups
-----
default

Database                                     GroupName
-----
tempdb                                       default
mytempdb                                    default
mytempdb1                                   default
mytempdb2                                   default
mytempdb3                                   default

Login   Application   Group   Database   Hardness
```

```

-----
NULL    isql          default  NULL      SOFT
sa      NULL            NULL    mytempdb3  HARD

```

示例 10 显示 default 临时数据库组:

```

sp_tempdb show, "gr"
Temporary Database Groups
-----
default

```

示例 11 显示绑定到 default 组的所有临时数据库组的名称:

```

sp_tempdb show, "gr", "default"
Member Databases
-----
tempdb
mytempdb
mytempdb1
mytempdb2
mytempdb3

```

示例 12 显示所有“数据库与组”的成员关系:

```

sp_tempdb show, "db"
Database          Group
-----
tempdb            default
mytempdb          default
mytempdb1         default
mytempdb2         default
mytempdb3         default

```

示例 13 显示 mytempdb1 数据库的所有“数据库与组”的成员关系。

```

sp_tempdb show, "db", "mytempdb1"
Database          Group
-----
mytempdb1         default

```

示例 14 显示登录名不为 NULL 的所有登录名绑定:

```
sp_tempdb show, "login"
```

| Login | Application | Group | Database | Hardness |
|-------|-------------|-------|-----------|----------|
| sa | NULL | NULL | mytempdb3 | HARD |

示例 15 显示指派到系统 tempdb 的所有活动会话:

```
sp_tempdb who, "tempdb"
```

| spid | loginame |
|------|----------|
| 2 | NULL |
| 3 | NULL |
| 4 | NULL |
| 5 | NULL |
| 6 | NULL |
| 7 | NULL |
| 8 | NULL |

示例 16 显示指派到用户创建的临时数据库 mytempdb3 的所有活动会话:

```
sp_tempdb who, "mytempdb3"
```

| spid | loginame |
|------|----------|
| 17 | sa |

示例 17 显示用法信息:

```
sp_tempdb help
```

Usage:

```
sp_tempdb 'help'
```

```
sp_tempdb 'create', <groupname>
```

```
sp_tempdb 'drop', <groupname>
```

```
sp_tempdb 'add', <tempdbname>, <groupname>
```

```
sp_tempdb 'remove', <tempdbname>, <groupname>
```

```
sp_tempdb 'bind', <objtype>, <objname>, <bindtype>, <bindobj>, <scope>, <hardness>
```

```
sp_tempdb 'unbind', <objtype>, <objname>, <scope>
```

```
sp_tempdb 'unbindall_db', <tempdbname>
```

```
sp_tempdb 'show', <command>, <name>
```

```
sp_tempdb 'who', <dbname>
```

```

<objtype> = ['LG' ('login_name') | 'AP' ('application_name')];
<bindtype> = ['GR' ('group') | 'DB' ('database')]
<hardness> = ['hard' | 'soft']
<command> = ['all' | 'gr' | 'db' | 'login' | 'app']

```

示例 18 显示所有临时数据库以及这些临时数据库所属的组的名称:

```

create temporary database mytempdb
-----
CREATE DATABASE: allocating 1536 logical pages (3.0
megabytes) on disk 'master'.|

create temporary database mytempdb1
-----
CREATE DATABASE: allocating 1536 logical pages (3.0
megabytes) on disk 'master'.

sp_tempdb 'add', mytempdb,'default'
-----
(return status = 0

sp_tempdb show, db
-----
Database Group
-----
tempdb default
mytempdb default
mytempdb1
(3 rows affected)
(return status = 0)

```

示例 19 显示已分配给指定临时数据库的所有活动会话的登录名和应用程序名称:

```

sp_addlogin anunay, anunay
-----
sp_tempdb "bind", lg, sa, DB, mytempdb3
-----
(return status = 0)

sp_tempdb "bind", lg, anunay, DB, mytempdb3
-----
(return status = 0)

starting sessions
-----
${ISQL} -J -U anunay -P anunay -I${SYBASE}/interfaces -w200

```

```
sp_tempdb who, mytempdb3
-----
spid loginame application
-----
11 sa isql
13 anunay isql

(2 rows affected)
(return status = 0)
```

用法

show 与 who

要显示所有临时数据库的用户分布情况，请同时使用 `show` 和 `who` 这两个选项：

- 要获得所有临时数据库的名称，请执行

```
sp_tempdb 'show'
```

- 将每个临时数据库名称传递给

```
sp_tempdb 'who', tempdbname
```

在 Adaptive Server 15.0 版及更高版本中，可以通过执行 `sp_who` 获得相同的输出。

create 与 drop

使用 `sp_tempdb create` 存储过程时，`groupname` 变量：

- 必须是有效的标识符
- 不能已经存在

`default` 组是系统生成的组，`tempdb` 始终属于该组。如果执行了以下操作，则 `default` 组已经存在：

- 使用包含此功能的 Adaptive Server 进行了升级，或
- 创建了新的主设备。

如果 `default` 组不存在，可以使用以下命令创建该组：

```
sp_tempdb create, "default"
```

如果 `default` 组已经存在，尝试创建该组时将出现一条错误消息。

add 与 remove

如果将某个临时数据库添加到 **default** 临时数据库组，该临时数据库和该组名必须已经存在。使用 **sp_tempdb add** 将 *tempdbname* 添加到属于 **default** 临时数据库组的数据库的集合中后，*tempdbname* 将可用于该组内的循环指派。

注释 如果 *tempdbname* 不在 Adaptive Server 中可用临时数据库的全局列表内，**sp_tempdb add** 将失败。

用户创建的临时数据库无需属于 **default** 临时数据库组。系统 **tempdb** 隐式属于 **default** 组。

如果尝试将某个已经属于 **default** 临时数据库组的临时数据库添加到该组，将出现一条错误消息，而且 **sysattributes** 中不会发生任何更改。

权限

缺省情况下，只有系统管理员或具有 SA 角色的用户才能执行 **sp_tempdb**。

审计

sysaudits 表的 **event** 和 **extrainfo** 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

sp_tempdb_markdrop

| | |
|----|---|
| 说明 | (在集群环境中) 将本地系统临时数据库置于删除状态。 |
| 语法 | <code>sp_tempdb_markdrop database_name [, {'mark' 'unmark'}]</code> |
| 参数 | <p><i>database_name</i> 是要删除的本地系统临时数据库的名称</p> <p>mark 将指定的数据库标记为待删除。</p> <p>unmark 清除数据库的此标记。</p> |
| 示例 | <p>示例 1 将名为 “old_cluster_tempdb1” 的本地系统临时数据库标记为待删除:</p> <pre>sp_tempdb_markdrop 'old_cluster_tempdb1', 'mark'</pre> <p>示例 2 删除本地系统临时数据库 “old_cluster_tempdb1” 上的此标记:</p> <pre>sp_tempdb_markdrop 'old_cluster_tempdb1', 'unmark'</pre> |
| 用法 | <p>要删除最后一个本地临时数据库, 请按以下步骤操作:</p> <ol style="list-style-type: none">1 使用 <code>sp_tempdb_markdrop</code> 将本地系统临时数据库置于删除状态。2 关闭拥有最后一个本地临时数据库的实例, 然后重新启动它。 <hr/> <p>注释 将本地系统临时数据库标记为待删除后, 如果没有其它实例处于活动状态, 所有者实例将会重新启动。该实例启动后不会使用标记的本地系统临时数据库。</p> <hr/> <ol style="list-style-type: none">3 使用 <code>drop database</code> 删除最后一个本地系统临时数据库。 |

sp_thresholdaction

说明 在日志段中的可用页数低于最后机会阈值时自动执行，除非该阈值与其它过程关联。Sybase 不提供此过程。

语法 达到阈值时，Adaptive Server 将按照位置将以下参数传递到阈值过程：

```
sp_thresholdaction @dbname,
                  @segment_name,
                  @space_left,
                  @status
```

参数

@dbname

是达到该阈值的数据库的名称。

@segment_name

是达到该阈值的段的名称。

@space_left

是阈值大小，以逻辑页为单位。

@status

对于最后机会阈值，其值为 1；对于所有其它阈值，其值为 0。

示例

为最后机会阈值创建一个阈值过程，用以将事务日志转储到磁带设备：

```
create procedure sp_thresholdaction
    @dbname varchar(30),
    @segmentname varchar(30),
    @space_left int,
    @status int
as
    dump transaction @dbname to tapedump1
```

用法

- **sp_thresholdaction** 必须由数据库所有者（在用户数据库中）创建，或由系统管理员（在 **sybssystemprocs** 数据库中）创建，或由具有 **create procedure** 权限的用户创建。
- 可以为数据库中的任何段添加阈值并创建阈值过程。
- 达到最后机会阈值时，Adaptive Server 将在发生阈值事件的数据库中搜索 **sp_thresholdaction** 过程。如果该数据库中没有，Adaptive Server 将在 **sybssystemprocs** 中搜索。如果 **sybssystemprocs** 中也没有，它将在 **master** 中搜索。如果 Adaptive Server 找不到该过程，它会将一条错误消息发送到错误日志。
- **sp_thresholdaction** 应包含一个 **dump transaction** 命令来截断事务日志。

- 按照设计，最后机会阈值留出了足够的可用空间用以记录 dump transaction 命令。但可能没有足够的空间来记录对数据库执行的其它用户事务。只能执行事务日志中未记录的命令（select、快速 bcp、readtext 和 writetext）以及释放额外日志空间可能需要的命令（dump transaction、dump database 和 alter database）。缺省情况下，会将其它命令挂起并将一条消息发送到错误日志。若要中止这些命令而不是将其挂起，请使用 sp_dboption 的 abort tran on log full 选项，然后使用 checkpoint 命令。

唤醒挂起的进程

- 在 dump transaction 命令释放了足够的日志空间后，系统会自动唤醒并完成挂起的进程。
- 如果快速 bcp、writetext 或 select into 导致未记录自上次备份以来对数据库所做的更改，则最后机会阈值过程不能执行 dump transaction 命令。出现这种情况时，请使用 dump database 复制该数据库，然后使用 dump transaction 截断事务日志。
- 如果这没有释放足够的空间以唤醒挂起的进程，则可能需要增大事务日志的大小。使用 alter database 命令的 log on 选项可以分配额外的日志空间。
- 不得已时，系统管理员可以使用 sp_who 来确定哪些进程处于挂起状态，然后使用 kill 命令将其注销。

审计

sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create procedure](#)、[dump transaction](#)

系统过程 [sp_addthreshold](#)、[sp_dboption](#)、[sp_droptreshold](#)、[sp_helpsegment](#)、[sp_helpthreshold](#)、[sp_modifythreshold](#)、[sp_who](#)

sp_tran_dumpable_status

说明 如果无法对数据库执行事务转储，`sp_tran_dumpable_status` 将显示无法转储的原因。

语法 `sp_tran_dumpable_status [database_name]`

参数 *database_name*
要调查的数据库的名称。

示例 说明当前无法对 `sybsystemprocs` 进行事务转储的原因：

```

sp_tran_dumpable_status sybsystemprocs
bit          description
-----
                2          Log is not on its own device
                8          Trunc log on ckpt is set
               32          Dump tran with truncate_only
               64          Database is new or upgraded

```

用法 此系统过程仅会调用 `tran_dumpable_status` 内置函数。

权限 任何用户都可以执行此过程。

sp_transactions

说明 报告有关活动事务的信息。

语法 sp_transactions ["xid", *xid_value*] |
 ["state", {"heuristic_commit" | "heuristic_abort"
 | "prepared" | "indoubt"} [, "xactname"]] |
 ["gtrid", *gtrid_value*]

参数 *xid_value*
 是 master.dbo.systransactions 的 xactname 列中的事务名称。

gtrid_value
 是由 Adaptive Server 协调的事务的全局事务 ID 名称。

示例 1 显示有关所有活动事务的一般信息：

```
sp_transactions

xactkey          type  coordinator starttime      state
connection dbid spid  loid failover  srvname  namelen  xactname
-----
-----
0x00000b1700040000dd6821390001 Local    None      Jun 1 1999 3:47PM Begun
Attached        1    1    2 Resident Tx      NULL      17
$user_transaction
0x00000b1700040000dd6821390001 Remote  ASTC      Jun 1 1999 3:47PM Begun
NA              0    8    0 Resident Tx      caserv2 108

00000b1700040000dd6821390001-aa01f04ebb9a-00000b1700040000dd6821390001-aa0
1f04ebb9a-caserv1-caserv1-0002
```

示例 2 显示指定事务的详细信息：

```
sp_transactions "xid",
"00000b1700040000dd6821390001-aa01f04ebb9a-00000b1700040000dd6821390001-aa
01f04ebb9a-caserv1-caserv1-0002"

xactkey          type  coordinator starttime      state
connection dbid spid  loid failover  srvname  namelen  xactname
commit_node      parent_node
gtrid
-----
-----
-----
0x00000b2500080000dd6821960001 External  ASTC      Jun 1 1999 3:47PM Begun
Attached        1    8   139 Resident Tx      NULL      108

00000b1700040000dd6821390001-aa01f04ebb9a-00000b1700040000dd6821390001-aa0
1f04ebb9a-caserv1-caserv1-0002

caserv1          caserv1
00000b1700040000dd6821390001-aa01f04ebb9a
```


sp_transactions 输出的各个列的说明

- xactkey 列显示 Adaptive Server 用于标识事务的内部事务键。
- type 列指示事务的类型：
 - “Local” 表示该事务是用 begin transaction 语句在本地 Adaptive Server 上显式启动的。
 - “Remote” 表示事务在远程 Adaptive Server 上执行。
 - “External” 表示该事务有一个与其相关联的外部事务协调器。例如，由远程 Adaptive Server、MSDTC 或 X/Open XA 事务管理器协调的事务标记为 “External”。
 - “Dtx_State” 是由 Adaptive Server 协调的分布式事务的一种特殊状态。它表示已在本地服务器上提交或中止了某个事务，但 Adaptive Server 无法在远程参与者上解析该事务的分支。如果 Adaptive Server 失去了与它正在协调的服务器的联系，则可能会出现这种情况。
- coordinator 列指示用于管理分布式事务的方法或协议：

| sp_transactions “coordinator” 值 | 含义 |
|------------------------------------|--|
| 无 | 事务不是分布式事务，因此不需要协调协议。 |
| ASTC | 事务是用 Adaptive Server 事务协调服务协调的。 |
| XA | 事务是由符合 X/Open XA 的事务管理器通过 Adaptive Server XA-Library 接口协调的。这样的事务管理器包括 Encina、CICS 和 Tuxedo，等等。 |
| DTC | 事务是由 MSDTC 协调的。 |
| SYB2PC | 事务是使用 Sybase 两阶段提交协议协调的。 |

- starttime 列显示该事务启动的时间。
- state 列指示运行 sp_transactions 时该事务的状态：

| sp_transactions “state” 值 | 含义 |
|------------------------------|------------------------|
| Begun | 事务已经启动，但尚未执行任何更新。 |
| Done Command | 事务已完成一个更新命令。 |
| Done | X/Open XA 事务已修改完数据。 |
| Prepared | 事务已成功准备就绪。 |
| In Command | 事务当前正在修改数据。 |
| In Abort Cmd | 已中止了事务中当前命令的执行。 |
| Committed | 已成功提交了事务，并且已写入了提交日志记录。 |
| In Post Commit | 已成功提交了事务，但当前正在释放事务资源。 |

| sp_transactions “state” 值 | 含义 |
|--------------------------------------|--|
| In Abort Tran | 正在中止事务。这可能是由显式执行某个命令导致的，也可能是因系统故障导致的。 |
| In Abort Savept | 正在将事务回退到一个保存点。 |
| Begun-Detached | 事务已开始，但当前没有线程附加到该事务。 |
| Done Cmd-Detached | 事务已修改完数据，而且当前没有线程附加到该事务。 |
| Done-Detached | 事务不会再修改任何数据，而且当前没有线程附加到该事务。 |
| Prepared-Detached | 事务已成功准备就绪，而且当前没有线程附加到该事务。 |
| Heur Committed | 已使用 <code>dbcc complete_xact</code> 命令尝试提交了事务。 |
| Heur Rolledback | 已使用 <code>dbcc complete_xact</code> 命令尝试回退了事务。 |

- `connection` 列指示该事务当前是否与线程相关联：
 - “Attached” 表示该事务有相关联的控制线程。
 - “Detached” 表示当前没有线程与该事务相关联。一些外部事务管理器（例如，CICS 和 TUXEDO）使用 X/Open XA “suspend” 和 “join” 语义将不同的线程与同一事务相关联。
- `dbid` 列显示启动该事务的数据库的数据库 ID。
- `spid` 列指示与该事务相关联的服务器进程 ID。如果事务状态为 “Detached”，则 “spid” 值为 0。
- `loid` 列指示 `master.dbo.systransactions` 中的唯一锁所有者 ID。
- `failover` 列指示该事务的故障切换状态：
 - “Resident Tx” 表示事务已启动，并且正在同一服务器上执行。在正常工作条件下，不使用 Adaptive Server 高可用性功能的系统上将显示 “Resident Tx”。
 - 对辅助协同服务器进行故障切换后，将显示 “Failed-over Tx”。“Failed-over Tx” 表示事务原来在主服务器上启动并且已达到就绪状态，但被自动迁移到辅助协同服务器上（例如，因为主服务器发生了系统故障）。就绪事务的迁移对于外部协调服务来说是透明的。
 - “Tx by Failover-Conn” 表示曾尝试在指定的服务器上启动该事务，但却在辅助协同服务器上启动了该事务。在原服务器发生故障切换时，可能会出现这种情况。
- `srvname` 列显示正在执行该事务的远程服务器的名称。此列仅对远程事务有意义。对于本地和外部事务，`srvname` 为空。
- `namelen` 列显示 `xactname` 值的总长度。

- *xactname* 是事务名称。对于本地事务，事务名称可能被定义为 `begin transaction` 命令的一部分。外部事务管理器以多种格式提供唯一的事务名称。例如，符合 X/Open XA 的事务管理器提供的事务 ID (*xid*) 由一个全局事务标识符和一个分支限定符组成，两者都存储在 *xactname* 中。
- 对于由 Adaptive Server 协调的事务，*gtrid* 列显示全局事务 ID。事务分支是同一分布式事务的一部分，它们共享同一个 *gtrid*。可以使用特定的 *gtrid* 和 `sp_transactions gtrid` 关键字来确定同一分布式事务中其它事务分支的状态。

`sp_transactions` 不能显示具有外部事务协调器的事务的 *gtrid*。对于由符合 X/Open XA 的事务管理器、MSDTC 或 SYB2PC 来协调的事务，*gtrid* 列将显示外部协调器提供的完整事务名称。

- 对于由 Adaptive Server 协调的事务，*commit_node* 列显示执行该分布式事务的最外层块的服务器。这个最外层块最终确定所有下级事务的提交状态。

对于不是由 Adaptive Server 协调的事务，*commit_node* 显示表 1-34 中列出的值之一。

表 1-34: *commit_node* 和 *parent_node* 的值

| 值 | 含义 |
|--------------------|---|
| <i>server_name</i> | 提交节点或父节点是具有指定 <i>server_name</i> 的 Adaptive Server。 |
| XATM | 提交节点或父节点是符合 X/Open XA 的事务管理器。 |
| MSDTCM | 提交节点或父节点是 MSDTC。 |
| SYB2PCTM | 事务是使用 SYB2PC 协议协调的。 |

- 对于由 Adaptive Server 协调的事务，*parent_node* 列显示在本地服务器上协调外部事务的服务器。

对于不是由 Adaptive Server 协调的事务，*parent_node* 显示表 1-34 中列出的值之一。

注释 *commit_node* 和 *parent_node* 的值可能会因分布式事务中的层次级别而有所不同。

权限

任何用户都可以执行 `sp_transactions`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** `sp_lock`、`sp_who`

sp_unbindcache

说明 解除数据库、表、索引、text 对象或 image 对象与数据高速缓存的绑定。

语法 sp_unbindcache *dbname* [,*owner*.]*tablename*
[, *indexname* | "text only"]]

参数 *dbname*
是要解除绑定的数据库的名称，或包含要解除绑定的对象的数据库的名称。

owner
是表所有者的名称。如果表属于数据库所有者，则所有者名称是可选的。

tablename
是要解除与高速缓存的绑定的表的名称，或其要解除其索引、text 对象或 image 对象与高速缓存的绑定的表的名称。

indexname
是要解除与高速缓存的绑定的索引的名称。

text only
解除 text 或 image 对象与高速缓存的绑定。

示例 **示例 1** 解除 titles 表与它绑定到的高速缓存的绑定：

```
sp_unbindcache pubs2, titles
```

示例 2 解除 titleidind 索引与它绑定到的高速缓存的绑定：

```
sp_unbindcache pubs2, titles, titleidind
```

示例 3 解除 au_pix 表的 text 或 image 对象与它绑定到的高速缓存的绑定：

```
sp_unbindcache pubs2, au_pix, "text only"
```

示例 4 解除事务日志 syslogs 与其高速缓存的绑定：

```
sp_unbindcache pubs2, syslogs
```

用法

- 解除了数据库或数据库对象与高速缓存的绑定后，该高速缓存的所有后续 I/O 都将在缺省数据高速缓存中执行。解除绑定的高速缓存中的所有脏页都将被写入磁盘，而所有干净页都将被从该高速缓存中清除。有关详细信息，请参见 [性能和调优指南](#)。
- 如果您尝试使用 sp_unbindcache 解除正在使用的数据库的绑定，Adaptive Server 将发出错误编号 857。
- 解除高速缓存绑定的操作会立即生效，而不需要重新启动服务器，但系统 tempdb 除外。

- 虽然仍可以对系统 `tempdb` 使用 `sp_unbindcache`，但系统 `tempdb` 的绑定现在不是动态的。在重新启动服务器之前：
 - 所做的更改不会生效。
 - 除非将系统 `tempdb` 显式绑定到缺省数据高速缓存，否则 `sp_helpcache` 报告状态 “P”（表示挂起），显式绑定到缺省数据高速缓存后，状态显示为 “V”（表示有效），这是因为缺省情况下，系统 `tempdb` 已被绑定到缺省数据高速缓存。
- 删除数据库、表或索引时，将自动删除其高速缓存绑定。
- 若要解除数据库的绑定，必须在使用 `master` 数据库。对于表、索引、`text` 对象或 `image` 对象，必须在使用存储这些对象的数据库。
- 若要解除数据库中任何系统表的绑定，您必须正在使用该数据库，而且该数据库必须处于单用户模式。请使用以下命令：

```
sp_dboption db_name, "single user", true
```

有关详细信息，请参见 `sp_dboption`。
- 以下过程提供其相应对象的绑定信息：`sp_helpdb` 适用于数据库；`sp_help` 适用于表；`sp_helpindex` 适用于索引。
- `sp_helpcache` 输出绑定到高速缓存的对象的名称。
- 解除表或其索引与高速缓存的绑定时，`sp_unbindcache` 需要获取一个排它表锁。在解除绑定期间，不能读取任何页。如果用户在某个表上持有锁，而您又对该对象发出了 `sp_unbindcache` 命令，则 `sp_unbindcache` 任务将等到锁被释放才开始执行。
- 使用 `sp_bindcache` 或 `sp_unbindcache` 更改了某个对象的高速缓存绑定后，下次执行引用该对象的存储过程时，将重新对其进行编译。更改了某个数据库的绑定后，下次执行引用该数据库中的对象的存储过程时，将重新对其进行编译。
- 若要解除所有对象与某个高速缓存的绑定，请使用系统过程 `sp_unbindcache_all`。

权限

只有系统管理员才能执行 `sp_unbindcache`。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none">• 角色 — 当前活动角色• 关键字或选项 — NULL• 以前值 — NULL• 当前值 — NULL• 其它信息 — 所有输入参数• 代理信息 — set proxy 有效时的初始登录名 |

另请参见

系统过程 [sp_bindcache](#)、[sp_dboption](#)、[sp_help](#)、[sp_helpdb](#)、[sp_helpcache](#)、[sp_helpdb](#)、[sp_helpindex](#)、[sp_unbindcache_all](#)

sp_unbindcache_all

| | |
|----|--|
| 说明 | 解除绑定到某个高速缓存的所有对象与该高速缓存的绑定。 |
| 语法 | <code>sp_unbindcache_all cache_name</code> |
| 参数 | cache_name 是要解除对象绑定的数据高速缓存的名称。 |
| 示例 | 解除绑定到 <code>pub_cache</code> 的所有数据库、表、索引、 <code>text</code> 对象和 <code>image</code> 对象与该高速缓存的绑定： <pre>sp_unbindcache_all pub_cache</pre> |
| 用法 | <ul style="list-style-type: none"> 解除了实体与高速缓存的绑定后，该高速缓存的所有后续 I/O 都将在缺省高速缓存中执行。 若要分别解除各个对象与高速缓存的绑定，请使用系统过程 sp_unbindcache。 如果系统 <code>tempdb</code> 绑定到 <code>pub_cache</code>，则不能使用 <code>sp_unbindcache_all</code>。如果使用了该过程，则会出现一条错误消息，而且 <code>sp_unbindcache_all</code> 会拒绝解除所有对象的绑定。 请先使用 <code>sp_unbindcache</code> 解除系统 <code>tempdb</code> 的绑定。 有关解除高速缓存绑定的详细信息，请参见 sp_unbindcache。 |
| 权限 | 只有系统管理员才能执行 <code>sp_unbindcache_all</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — set proxy 有效时的初始登录名 |

另请参见 **系统过程** [sp_bindcache](#)、[sp_helpcache](#)、[sp_unbindcache](#)

sp_unbindefault

| | |
|----|--|
| 说明 | 解除创建的缺省值与列或用户定义数据类型的绑定。 |
| 语法 | <code>sp_unbindefault <i>objname</i> [, futureonly]</code> |
| 参数 | <p><i>objname</i></p> <p>是要解除缺省值绑定的表和列或用户定义数据类型的名称。如果此参数的格式不是“<i>table.column</i>”，则会将 <i>objname</i> 视为用户定义的数据类型。解除缺省值与用户定义数据类型的绑定时，将同时解除所有属于该类型并且与该用户定义数据类型具有相同缺省值的列的绑定。属于该类型但缺省值已更改的列不受影响。</p> <p><i>futureonly</i></p> <p>防止属于指定用户定义数据类型的现有列丢失其缺省值。解除缺省值与列的绑定时，将忽略此参数。</p> |
| 示例 | <p>示例 1 解除缺省值与 <code>employees</code> 表的 <code>startdate</code> 列的绑定：</p> <pre>sp_unbindefault "employees.startdate"</pre> <p>示例 2 解除缺省值与名为 <code>ssn</code> 的用户定义数据类型和所有属于该类型的列的绑定：</p> <pre>sp_unbindefault ssn</pre> <p>示例 3 解除缺省值与用户定义数据类型 <code>ssn</code> 的绑定，但不影响属于该类型的现有列：</p> <pre>sp_unbindefault ssn, futureonly</pre> |
| 用法 | <ul style="list-style-type: none">使用 <code>sp_unbindefault</code> 可以删除用 <code>sp_bindefault</code> 创建的缺省值。使用 <code>alter table</code> 可以删除用 <code>create table</code> 或 <code>alter table</code> 语句声明的缺省值。属于用户定义数据类型的列将丢失其当前缺省值，除非其缺省值已更改，或可选的第二个参数的值为 <code>futureonly</code>。若要显示某个缺省值的文本，请使用该缺省值的名称作为参数执行 <code>sp_helptext</code>。 |
| 权限 | 只有对象所有者才能执行 <code>sp_unbindefault</code> 。 |

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|-----------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |
| 67 | unbind | sp_unbindefault | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — NULL • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 [create default](#)、[drop default](#)

系统过程 [sp_bindefault](#)、[sp_helptext](#)

sp_unbindexclass

| | |
|----|---|
| 说明 | 在指定作用域内，删除以前与客户端应用程序、登录名、存储过程或缺省执行类关联的执行类属性。 |
| 语法 | <code>sp_unbindexclass object_name, object_type, scope</code> |
| 参数 | <p>object_name 是要解除与执行类的关联的应用程序、登录名或存储过程的名称。如果 object_type 为 DF，object_name 就应该为空。</p> <p>object_type 将 object_name 的类型标识为 AP、LG、PR 或 DF，表示应用程序、登录名、存储过程或缺省执行类。</p> <p>scope 对于应用程序或登录名，是要执行解除绑定的应用程序名或登录名。对于存储过程，是存储过程所有者的名称（用户名）。对于对象类型 DF，它为空。</p> |
| 示例 | 解除应用程序 <code>isql</code> 作用域内的登录名“sa”与某个执行类的关联。“sa”会自动将其自身绑定到另一个执行类，具体情况取决于其它绑定规范、优先顺序和作用域界定规则。如果没有其它绑定适用，该对象将绑定到缺省执行类 EC2: |
| 用法 | <pre>sp_unbindexclass 'sa', 'lg', 'isql'</pre> <ul style="list-style-type: none"> • 参数必须与 <code>sysattributes</code> 系统表中的现有条目匹配。 • 如果指定的作用域是空值，Adaptive Server 将解除作用域为空值的对象的绑定（如果有这样的对象）。 • 作用域为空值并不表示应对所有绑定的对象执行解除绑定。 • 解除存储过程与执行类的绑定时，必须使用存储过程所有者的名称（用户名）作为 <code>scope</code> 参数。 • 如果解除存储过程与用户定义缺省执行类的绑定，则所有用用户定义缺省执行类属性运行的任务都会用系统定义缺省执行类 EC2 的属性运行。 • 在解除绑定前后都可以删除存储过程。 • 如果某个用户是数据库中某个绑定到执行类的存储过程的所有者，则不能从该数据库中删除该用户。 • 从数据库中删除类型为 PR 的对象之前，应先解除这些对象的绑定。 • 如果相关联的引擎组中没有联机引擎，而且有活动进程绑定到相关联的执行类，则解除绑定将失败。 |

- 视优先顺序和范围规则而定，要解除绑定的执行类可能已对 *object_name* 的生效，也可能尚未生效。该对象会自动将其自身绑定到另一个执行类，具体情况取决于其它绑定规范、优先顺序和作用域界定规则。如果没有其它绑定适用，该对象将绑定到缺省执行类。如果没有用户定义的缺省执行类，该对象将绑定到类 EC2。

权限 只有系统管理员才能执行 `sp_unbindexclass`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_addexclass](#)、[sp_bindexclass](#)、[sp_dropexclass](#)、[sp_showexclass](#)

实用程序 `isql`

sp_unbindmsg

| | |
|----|---|
| 说明 | 解除用户定义的消息与约束的绑定。 |
| 语法 | <code>sp_unbindmsg <i>constname</i></code> |
| 参数 | <i>constname</i> 是要解除消息绑定的约束的名称。 |
| 示例 | 解除用户定义的消息与约束 <code>positive_balance</code> 的绑定： <pre>sp_unbindmsg positive_balance</pre> |
| 用法 | <ul style="list-style-type: none"> 一个约束只能绑定一条消息。若要更改绑定到约束的消息，请使用 sp_bindmsg；新消息号将替代任何现有绑定的消息。不需要先使用 <code>sp_unbindmsg</code> 解除绑定。 若要从 <code>sysusermessages</code> 表检索消息文本，请执行 sp_getmessage。 |
| 权限 | 只有对象所有者才能执行 <code>sp_unbindmsg</code> 。 |
| 审计 | <code>sysaudits</code> 表的 <code>event</code> 和 <code>extrainfo</code> 列中的值如下所示： |

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|--------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — 所有输入参数 代理信息 — <code>set proxy</code> 有效时的初始登录名 |
| 69 | unbind | sp_unbindmsg | <ul style="list-style-type: none"> 角色 — 当前活动角色 关键字或选项 — NULL 以前值 — NULL 当前值 — NULL 其它信息 — NULL 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **系统过程** [sp_addmessage](#)、[sp_bindmsg](#)、[sp_getmessage](#)

sp_unbindrule

| | |
|----|---|
| 说明 | 解除规则与列或用户定义数据类型的绑定。 |
| 语法 | <code>sp_unbindrule <i>objname</i> [, futureonly [, "accessrule" "all"]]</code> |
| 参数 | <p>objname 是要解除规则绑定的表和列或用户定义数据类型的名称。如果此参数的格式不是“<i>table.column</i>”，则会将 <i>objname</i> 视为用户定义的数据类型。解除规则与用户定义数据类型的绑定时，将同时解除它与属于该类型的列的绑定。已经绑定到其它规则的列不受影响。</p> <p>futureonly 防止属于指定用户定义数据类型的列丢失其规则。解除规则与列的绑定时，将忽略此选项。</p> <p>accessrule 表示要解除绑定到 <i>objname</i> 的访问规则的绑定。</p> <p>all 指定要解除所有绑定到 <i>objname</i> 的规则绑定。</p> |
| 示例 | <p>示例 1 解除规则与 <code>employees</code> 表的 <code>startdate</code> 列的绑定：</p> <pre>sp_unbindrule "employees.startdate"</pre> <p>示例 2 解除规则与名为 <code>def_ssn</code> 的用户定义数据类型和所有属于该类型的列的绑定：</p> <pre>sp_unbindrule def_ssn</pre> <p>示例 3 用户定义数据类型 <code>ssn</code> 将不再具有规则，但现有 <code>ssn</code> 列不受影响：</p> <pre>sp_unbindrule ssn, futureonly</pre> <p>示例 4 使用 <code>all</code> 参数可以同时解除访问规则绑定和域规则绑定。例如，若要解除 <code>publishers</code> 表上的所有访问规则和域规则绑定，请执行以下命令：</p> <pre>sp_unbindrule publishers, null, "all"</pre> <p>若要解除访问规则与某个用户定义数据类型的绑定，并且希望更改只影响以后对该数据类型的使用，请发出以下命令：</p> <pre>sp_unbindrule def_ssn, futureonly, "accessrule"</pre> <p>若要同时解除访问规则和域规则与该数据类型的绑定，并且希望更改只影响以后对该数据类型的使用，请发出以下命令：</p> <pre>sp_unbindrule def_ssn, futureonly, "all"</pre> |

示例 5 下面的访问规则绑定到 publishers 表:

```
sp_bindrule empl_access, "publishers.pub_id"
```

若要解除该规则的绑定, 请发出以下命令:

```
sp_unbindrule "empl_access", NULL, "accessrule"
```

用法

- 执行 `sp_unbindrule` 将解除用于当前数据库中的列或用户定义数据类型的规则。如果不希望解除规则与现有 `objname` 列的绑定, 可使用 `futureonly` 作为第二个参数。
- 不能使用 `sp_unbindrule` 解除检查约束的绑定。请使用 `alter table` 删除该约束。
- 若要解除规则与表中某一列的绑定, 请以 “`table.column`” 格式指定 `objname` 参数。
- 将解除规则与属于该用户定义数据类型的所有现有列的绑定, 除非已经更改了规则或可选的第二个参数的值为 `futureonly`。
- 若要显示规则的文本, 请使用规则名称作为参数执行 `sp_helptext`。

权限

只有对象所有者才能执行 `sp_unbindrule`。

审计

`sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见

命令 `create rule`、`drop rule`

系统过程 `sp_bindrule`、`sp_helptext`

sp_version

说明 返回上次运行的安装脚本（*installmaster*、*installdbccdb* 等）的版本信息以及关于它是否成功运行的信息。

语法 `sp_version [script_file, [all]]`

参数 *script_file*
是安装脚本的名称（缺省值为 NULL）。

all

报告有关安装脚本的详细信息，例如运行日期以及运行所花费的时间。

示例 **示例 1** 返回已运行的所有安装脚本的脚本名称、版本和状态：

```
sp_version
Script          Version
Status
-----
installmaster  15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu
Sep 23 22:12:12 2004
Complete
installmaster  15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu
Sep 23 22:12:12 2004
Complete
installmodel   15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1861/32-
bit/OPT/Mon Sep 27 23:40:02 2004
Complete
```

示例 2 返回有关 *installmaster* 安装脚本的信息：

```
sp_version      installmaster
-----
installmaster  15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu
Sep 23 22:12:12 2004
Complete
```

示例 3 返回运行的所有安装脚本的脚本文件名、日期、时间、版本和状态：

```
sp_version null, 'all'
Script
Version          Status
Start/End Date
-----
installdbccdb   15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1861/32-
bit/OPT/Mon Sep 27 23:40:02 2004
Complete [Started=Sep 29 2004 4:41PM]-[Completed=Sep 29 2004 4:42PM]
```

```
installmaster
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu Sep 23 22:12:
12 2004
Complete [Started=Sep 29 2004 3:49PM]-[Completed=Sep 29 2004 3:58PM]
installmodel
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1861/32-bit/OPT/Mon Sep 27 23:
40:02 2004
Complete [Started=Sep 29 2004 4:51PM]-[Completed=Sep 29 2004 4:51PM]
```

示例 4 返回名称类似于 *install%* 的所有安装脚本的脚本文件名、版本和安装状态:

```
sp_version 'install%'
Script
Version          Status
-----
installdbccdb
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1861/32-bit/OPT/Mon Sep 27
23:40:02 2004      Complete
installmaster
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu Sep 23 22:12:
12 2004            Complete
installmodel
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1861/32-bit/OPT/Mon Sep 27 23:
40:02 2004            Complete
```

示例 5 返回与通配符 “install%” 匹配的安裝脚本的所有详细信息:

```
sp_version 'install%', 'all'
Script
Version          Status
Start/End Date
-----
installmaster
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu Sep 23 22:12:
12 2004
Complete [Started=Sep 29 2004 3:49PM]-[Completed=Sep 29 2004 3:58PM]
```

示例 6 返回有关 *installmaster* 安装脚本的所有详细信息:

```
sp_version 'installmaster', 'all'
Script
Version          Status
Start/End Date
-----
installmaster
15.0/EBF XXXXX/B/Sun_svr4/OS 5.8/asemain/1/32-bit/OPT/Thu Sep 23 22:12:
12 2004
```

Complete [Started=Sep 29 2004 3:49PM]-[Completed=Sep 29 2004 3:58PM]

用法 通过 `sp_version` 可以确定在 Adaptive Server 上安装的脚本 (`installmaster`、`installdbccdb` 等) 的当前版本, 并可以确定脚本运行是否成功以及完成所花费的时间

权限 任何用户都可以运行 `sp_version`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|-----------------------------|------------|--|
| 38 | <code>exec_procedure</code> | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

sp_volchanged

| | |
|----|--|
| 说明 | 通知 Backup Server 操作员在转储或装载期间执行了所请求的卷处理操作。 |
| 语法 | <code>sp_volchanged session_id, devname, action [, fname [, vname]]</code> |
| 参数 | <p>session_id 标识请求更换卷的 Backup Server 会话。请使用 Backup Server 的卷更换请求中指定的 <code>@session_id</code> 参数。</p> <p>devname 是装入了新卷的设备。请使用 Backup Server 的卷更换请求中指定的 <code>@devname</code> 参数。如果 Backup Server 与 Adaptive Server 不在同一台计算机上，请使用以下格式： <code>device at backup_server_name</code></p> <p>action 指示 Backup Server 应 <code>abort</code>（中止）、<code>proceed</code>（继续），还是 <code>retry</code>（重试）该转储或装载操作。</p> <p>fname 是要装载的文件。如果未在 <code>sp_volchanged</code> 中指定文件名，Backup Server 将装载 <code>load</code> 命令的 <code>file = filename</code> 参数指定的文件。如果 <code>sp_volchanged</code> 和 <code>load</code> 命令都未指定要装载哪个文件，Backup Server 将装载磁带上的第一个文件。</p> <p>vname 是 ANSI 磁带标签上显示的卷名。Backup Server 在覆盖现有转储、转储到全新的磁带或转储到内容不可识别的磁带时，会将卷名写入 ANSI 磁带标签。如果未在 <code>sp_volchanged</code> 中指定 <code>vname</code>，Backup Server 将使用 <code>dump</code> 命令中指定的 <code>dumpvolume</code> 值。如果 <code>sp_volchanged</code> 和 <code>dump</code> 命令都未指定卷名，Backup Server 会将 ANSI 磁带标签的名称字段留为空白。</p> <p>在装载过程中，Backup Server 使用 <code>vname</code> 来确认是否装入了正确的磁带。如果未在 <code>sp_volchanged</code> 中指定 <code>vname</code>，Backup Server 将使用 <code>load</code> 命令中指定的 <code>dumpvolume</code>。如果 <code>sp_volchanged</code> 和 <code>load</code> 命令都未指定卷名，Backup Server 在装载转储之前将不检查 ANSI 磁带标签的名称字段。</p> |
| 示例 | 操作员更换了磁带，然后发出以下命令： <code>sp_volchanged 8, "/dev/nrmt4", RETRY</code> |

来自 Backup Server 的以下消息指示装入的磁带尚未到期:

```
Backup Server: 4.49.1.1: OPERATOR: Volume to be overwritten on
'/dev/rmt4' has not expired: creation date on this volume is Sunday, Nov.
15, 1992, expiration date is Wednesday, Nov. 25, 1992.
Backup Server: 4.78.1.1: EXECUTE sp_volchanged
    @session_id = 8,
    @devname = '/auto/remote/pubs3/SERV/Masters/testdump',
    @action = { 'PROCEED' | 'RETRY' | 'ABORT' }
```

用法

- 如果 Backup Server 检测到当前装入的卷有问题，它将请求更换卷：
 - （在 OpenVMS 系统上）Backup Server 会将卷更换消息发送到正在运行它的计算机上的操作员终端。使用 dump 或 load 命令的 with notify = client 选项可将其它 Backup Server 消息发送到启动 dump 或 load 请求的终端会话。
 - （在 UNIX 系统上）Backup Server 会将消息发送到启动 dump 或 load 请求的客户端。使用 dump 或 load 命令的 with notify = operator_console 选项可将消息发送到启动 Backup Server 的终端。
 - 装入另一个卷后，操作员可从能够与执行转储或装载的 Backup Server 进行通信的任何 Adaptive Server 执行 sp_volchanged。操作员不必登录到发起该转储或装载的 Adaptive Server。
- （在 OpenVMS 系统上）操作系统（而不是 Backup Server）在检测到已到达卷末尾或指定的驱动器脱机时，会请求更换卷。操作员应使用 OpenVMS REPLY 命令答复这些消息。
- （在 UNIX 系统中）Backup Server 在磁带容量满后请求更改卷。操作员装入另一磁带，然后执行 sp_volchanged。表 1-35 演示了此过程。

表 1-35: 在 UNIX 系统上更换磁带卷

| 顺序 | 操作员, 使用 isql | Adaptive Server | Backup Server |
|----|---|--|---|
| 1 | <ul style="list-style-type: none"> • 发出 dump database 命令 | | |
| 2 | | <ul style="list-style-type: none"> • 向 Backup Server 发出转储请求 | |
| 3 | | | <ul style="list-style-type: none"> • 接收到来自 Adaptive Server 的转储请求消息 • 将装入磁带的消息发送给操作员 • 等待操作员的答复 |

| 顺序 | 操作员, 使用 isql | Adaptive Server | Backup Server |
|----|--|---|---|
| 4 | <ul style="list-style-type: none"> • 接收到来自 Backup Server 的卷更换请求 • 装入磁带 • 执行 sp_volchanged | | |
| 5 | | | <ul style="list-style-type: none"> • 检查磁带 • 如果磁带没有问题, 则开始转储 • 磁带已满后, 向操作员发出卷更换请求 |
| 6 | <ul style="list-style-type: none"> • 接收到来自 Backup Server 的卷更换请求 • 装入磁带 • 执行 sp_volchanged | | |
| 7 | | | <ul style="list-style-type: none"> • 继续转储 • 转储完成后, 将消息发送给操作员和 Adaptive Server |
| 8 | <ul style="list-style-type: none"> • 接收到转储已完成的消息 • 取出磁带并给磁带标号 | <ul style="list-style-type: none"> • 接收到转储已完成的消息 • 释放锁 • 完成 dump database 命令 | |

权限 任何用户都可以执行 sp_volchanged。

审计 sysaudits 表的 event 和 extrainfo 列中的值如下所示:

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|---|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — set proxy 有效时的初始登录名 |

另请参见 [命令 dump database、dump transaction、load database、load transaction](#)
[实用程序 isql](#)

sp_webservices

说明 创建和管理 Adaptive Server Web 服务引擎使用的代理表。

语法 创建代理表:

```
sp_webservices 'add', 'wsdl_uri' [, sds_name]
                [, 'method_name=proxy_table
                [,method_name=proxy_table ]* ' ]
```

显示 sp_webservices 的用法信息:

```
sp_webservices help [, 'option']
```

列出映射到 WSDL 文件的代理表:

```
sp_webservices 'list' [, 'wsdl_uri'] [, sds_name]
```

修改超时设置:

```
sp_webservices 'modify', 'wsdl_uri', 'timeout=time'
```

删除映射到 WSDL 文件的代理表:

```
sp_webservices 'remove', 'wsdl_uri' [, sds_name]
```

用户定义的 Web 服务的选项

为用户定义的 Web 服务创建数据库别名:

```
sp_webservices 'addalias' alias_name , database_name
```

配置用户定义的 Web 服务:

```
sp_webservices 'deploy', ['all' | 'service_name']
```

在用户定义的 Web 服务中删除数据库别名:

```
sp_webservices 'dropalias' alias_name
```

列出用户定义的 Web 服务中映射到 WSDL 文件的代理表:

```
sp_webservices 'listudws' [, 'service_name']
```

列出用户定义的 Web 服务的别名或数据库别名。

```
sp_webservices 'listalias'
```

取消配置用户定义的 Web 服务:

```
sp_webservices 'undeploy', ['all' | 'service_name']
```

参数

'add', 'wsdl_uri' [, *sds_name*] [, 'method_name=proxy_table[,
method_name=proxy_table]*']

用于为 WSDL 文件指定的 Web 方法创建代理表。如果成功使用了 **add** 选项，将自动调用 **list** 选项以描述新代理表的模式。

- *wsdl_uri* — 是要映射到新代理表的 WSDL 文件的位置。如果指定了此参数，则 Web 服务会确保 **syswsdl** 表中存在 URI。
- *sds_name* — 在 *interfaces* 或 *sql.ini* 文件中为 ASE Web 服务引擎指定的名称。缺省值为 **ws**。如果 **sysattributes** 表中不存在任何条目，则会产生错误。
- *method_name* — 是映射到代理表的 Web 方法的名称。所指定的 *method_name* 必须是在关联的 WSDL 文件中指定的 Web 方法的名称。
- *proxy_table* — 是在 *method_name* 中指定的 Web 方法映射到的代理表的名称。

'addalias' *alias_name* , *database_name*

用于创建在用户定义的 Web 服务中表示数据库名称的别名，其中：

- *alias_name* — 是指定数据库的别名。该参数是必需的。
- *database_name* — 是指定的别名对应的数据库的名称。该参数是必需的。

使用别名可以在指定 URL 中表示数据库名称的部分时进行更好的控制。将别名与 **create service** 命令的 **userpath** 选项搭配使用时，可以完全控制用于访问用户定义的 Web 服务的 URL。

'deploy', ['all' | '*service_name*']

用于配置用户定义的 Web 服务，以使 ASE Web 服务引擎可通过 HTTP 或 HTTPS 对其进行访问，其中：

- **all** — 指定为当前数据库配置所有用户定义的 Web 服务。
- *service_name* — 是要配置的用户定义 Web 服务的名称。

deploy 和 **undeploy** 选项用于控制用户定义的 Web 服务何时可用。需要具备系统角色 **webservices_role** 特权才能使用此选项。

如果指定了 **all** 参数，则 ASE Web 服务引擎将删除其用户定义的 Web 服务的内部高速缓存，并从 Adaptive Server Enterprise 中重新读取有关用户定义的 Web 服务的所有元数据。

不能删除或重命名当前配置的用户定义的 Web 服务。

'dropalias' alias_name

用于删除表示数据库名称的别名，其中 *alias_name* 是要删除的别名。

如果一个已配置的用户定义的 Web 服务正在引用一个别名，则无法删除该别名。要删除该别名，必须先取消配置引用该别名的用户定义的 Web 服务。

help[, 'option']

提供阐明如何使用 `sp_webservices` 存储过程的说明和示例。'option' 的有效值包括 `add`、`list`、`remove` 和 `modify`。

如果没有为 *option* 指定任何值，则 `help` 选项将输出 `add`、`addalias`、`deploy`、`dropalias`、`list`、`listalias`、`listudws`、`modify`、`remove` 和 `undeploy` 选项的简短语法描述。

'list' [, 'wsdl_uri'] [, sds_name]

列出 WSDL 文件中描述的 Web 方法，其中：

- *wsdl_uri* — 是所映射的 WSDL 文件的 URI。如果没有为 *wsdl_uri* 指定值，则 `list` 选项将显示有关已经映射到代理表的所有 Web 方法的信息。
- *sds_name* — 是在 `interfaces` 或 `sql.ini` 文件中为 ASE Web 服务引擎指定的 SDS 服务器的名称。缺省值为 `ws`。如果 `sysattributes` 表中不存在任何条目，则会产生错误。

如果 *wsdl_uri* 和 *sds_name* 这两个参数都没有指定，则会列出 `sysattributes` 表中的所有条目并按 `wsdlid` 排序。

如果 WSDL 文件中描述的 Web 方法：

- 已经映射到代理表 — 则 `list` 选项将输出有关各个代理表的信息。
- 尚未映射到代理表 — 则 `list` 选项会输出可用于创建代理表的 SQL。

'listalias'

用于列出用户定义的 Web 服务中的所有别名。

'listudws' [, 'service_name']

用于列出当前数据库的用户定义的 Web 服务，其中 *service_name* 是要列出的用户定义的 Web 服务的名称。

如果不指定 *service_name* 参数，则列出所有由用户定义的 Web 服务。

'modify', 'wsdl_uri', 'timeout=time'

用于修改 WSDL 文件的属性信息，其中：

- *wsdl_uri* — 是要更改属性信息的 WSDL 文件的 URI。
- *time* — 是一段以秒为单位的时间间隔，在此时间间隔内 Web 方法必须做出响应，否则此时间间隔结束时操作将中止。

'remove', 'wsdl_uri' [, *sds_name*]

用于删除 Web 方法的代理表映射，其中：

- *wsdl_uri* — 是要删除代理表的 WSDL 文件的 URI。
- *sds_name* — 是在 *interfaces* 或 *sql.ini* 文件中为 ASE Web 服务引擎指定的 SDS 服务器的名称。缺省值为 *ws*。

注释 如果 *sysattributes* 表中不存在任何条目，则会产生错误。

'undeploy', [all | '*service_name*']

用于使 Adaptive Server Enterprise Web 服务引擎无法通过 HTTP 或 HTTPS 访问用户定义的 Web 服务，其中：

- *all* — 指定要为当前数据库取消配置所有由用户定义的 Web 服务。
- *service_name* — 是要取消配置的用户定义 Web 服务的名称。

使用 *deploy* 和 *undeploy* 选项可控制用户定义的 Web 服务何时可用。需要具备系统角色 *webservices_role* 特权才能使用此选项。

示例

示例 1 调用一个 RPC/ 编码 Web 方法，以显示两种货币之间的汇率。

1 使用 *sp_webservices* 的 *add* 选项将 Web 方法映射到代理表：

```
1> sp_webservices 'add',
'http://www.xmethods.net/sd/2001/CurrencyExchangeService.wsdl'
2> go
```

getRate Web 方法即会映射到与其同名的代理表。

2 通过从该代理表中选择数据来调用该 Web 方法：

```
1> select * from getRate where _country1 = 'usa' and _country2 = 'india'
2> go
```

前面的 *select* 返回的结果显示指定参数的汇率：

```
Result          _country1      _country2
43.000000      usa            india
(1 row affected)
```

示例 2 调用一个 Web 方法以显示 XML 文档中的股票信息。

1 使用 `sp_webservices` 的 `add` 选项将 Web 方法映射到代理表:

```
1> sp_webservices "add" , "http://www.webserviceX.net/stockquote.asmx?WSDL"
2> go
```

`GetQuote` Web 方法即会映射到与其同名的代理表。

2 通过选择 `GetQuote` 代理表的 `outxml` 列来调用该 Web 方法:

```
1> select outxml from GetQuote where _inxml = '<?xml version="1.0"
encoding="utf-8"?>
2>     <GetQuote xmlns="http://www.webserviceX.NET/">
3>         <symbol>SY</symbol>
4>     </GetQuote>'
5> go
```

前面的 `select` 返回的结果显示 XML 文档中的报价信息:

```
outxml

<?xml version="1.0" encoding="UTF-8" ?><GetQuoteResponse
xmlns="http://www.webserviceX.NET/"><GetQuoteResult><StockQuotes><Stock>
<Symbol>SY</Symbol><Last>21.48</Last><Date>7/21/2005</Date><Time>4:01pm
</Time><Change>+1.72</Change><Open>20.00</Open><High>21.60</High>
<Low>19.91</Low><Volume>2420100</Volume><MktCap>1.927B</MktCap>
<PreviousClose>19.76</PreviousClose><PercentageChange>+8.70%
</PercentageChange><AnnRange>12.75 - 20.44</AnnRange><Earnings>0.706</Earnings>
<P-E>27.99</P-E><Name>SYBASE INC</Name></Stock></StockQuotes>
</GetQuoteResult></GetQuoteResponse>
```

(1 row affected)

示例 3 通过一个显示股票信息的视图调用在上一个示例中映射到代理表的 `GetQuote` Web 方法。

1 创建一个表来保存表示股票的股票代码, 以便使用此 Web 服务:

```
1> create table stocksymbol(symbol varchar(100))
2> go
```

2 向 `stocksymbol` 表中插入数据:

```
1> insert stocksymbol values("SY")
2> insert stocksymbol values("ORCL")
3> go
```

3 创建一个调用 `GetQuote` Web 方法的视图:

```
1> CREATE VIEW getstockvw as
2> select Symbol = xmlextract('//Stock/Symbol/text()',outxml returns varchar(5)),
3>     Name = xmlextract('//Stock/Name/text()',outxml returns varchar(20)),
4>     Time = xmlextract('//Stock/Time/text()',outxml returns varchar(10)),
```

```

5> Date = xmlextract('//Stock/Date/text()',outxml returns date),
6> High = xmlextract('//Stock/High/text()',outxml returns decimal(15,2)),
7> Low = xmlextract('//Stock/Low/text()',outxml returns decimal(15,2))
8> FROM GetQuote ,stocksymbol
9> WHERE _inxml = '<GetQuote
xmlns="http://www.webserviceX.NET/"><symbol>'+symbol+'</symbol></GetQuote>'
10> go

```

4 从 getstockvw 视图进行选择以查看 GetQuotes 方法的输出:

```

1> select * from getstockvw
2> go

```

前面的 **select** 返回的结果显示视图定义指定的参数的报价信息:

| Symbol | Name | Time | Date | High | Low |
|--------|--------------|--------|-------------|-------|-------|
| SY | SYBASE INC | 4:01pm | Jul 21 2005 | 21.60 | 19.91 |
| ORCL | ORACLE CORP | 4:00pm | Jul 21 2005 | 14.05 | 13.54 |
| MSFT | MICROSOFT CP | 4:00pm | Jul 21 2005 | 26.48 | 26.19 |

(3 rows affected)

示例 4 显示由用户 “bob” 在 pubs2 数据库中输入的以下命令的审计表条目:

```
sp_webservices 'deploy', 'all'
```

对应的审计表条目列出了 110、bob 和 pubs2，分别作为 event、loginname 和 dbname 列中的值。extrainfo 列包含:

```
webservices_role; deploy_all; ; ; ; bob/ase;
```

示例 5 显示由用户 “bob” 在 pubs2 数据库中输入的以下命令的审计表条目:

```
sp_webservices 'deploy', 'rawservice'
```

对应的审计表条目列出了 110、bob 和 pubs2，分别作为 event、loginname 和 dbname 列中的值。extrainfo 列包含:

```
webservices_role; deploy; ; ; ; bob/ase;
```

示例 6 显示由用户 “bob” 在 pubs2 数据库中输入的以下命令的审计表条目:

```
sp_webservices 'undeploy', 'all'
```

对应的审计表条目列出了 111、bob 和 pubs2，分别作为 event、loginname 和 dbname 列中的值。extrainfo 列包含:

```
webservices_role; undeploy_all; ; ; ; bob/ase;
```


示例 2 显示由用户“bob”在 pubs2 数据库中输入的以下命令的审计表条目：

```
sp_webservices 'undeploy', 'rawservice'
```

对应的审计表条目列出了 111、bob 和 pubs2，分别作为 event、loginname 和 dbname 列中的值。extrainfo 列包含：

```
webservices_role; deploy; ; ; ; bob/ase;
```

有关 sysaudits 表列的完整说明，请参见 Adaptive Server Enterprise 《系统管理指南》。

用法

sp_webservices add

如果没有为 Web 方法指定 *method_name* 和 *proxy_table* 值，则缺省情况下为 Web 方法生成的代理表是在 WSDL 文件中指定的 Web 方法的名称。如果已有一个与此 Web 方法同名的代理表，则会生成一个具有如下形式的名称的新代理表：

```
method_nameN
```

其中：

- *method_name* — 是缺省的代理表名称
- *N* — 是一个介于 1 和 9 之间的数字，表示该 Web 方法的每个后续映射。可以有最多 99 个重复代理表。

如果要为 Web 方法指定 *method_name* 和 *proxy_table* 值，则代理表的名称必须是新的。如果已存在一个具有 *proxy_table* 中指定了名称的代理表，则会产生错误，而且在 add 选项中指定的所有 Web 方法都不会映射到代理表。

add 选项的输出中列出了已成功映射到代理表的方法，以及那些尚未映射的方法。未映射的 Web 方法的代理表名称在 add 选项的输出中以 NULL 表示。

注释 对于为 RPC/ 编码 Web 方法和文档 / 文字 Web 方法生成的代理表来说，用于输入和输出的列是不同的。在表示 RPC/ 编码 Web 方法的代理表中，每个输入和输出参数都有对应的一列。表示文档 / 文字 Web 方法的代理表包含 _inxml 和 outxml 两列。

用户定义的 Web 服务的安全性

需要具备系统角色 webservices_role 才能使用 sp_webservices 的 deploy 和 undeploy 选项。要执行用户定义的 Web 服务，必须具有用于执行相应存储过程的有效登录名和权限。

要创建、删除和执行用户定义的 Web 服务，需要具有在 Adaptive Server Enterprise 中创建、删除和执行存储过程所需的特权。有关如何使用 `grant` 和 `revoke` 命令设置适当特权的详细信息，请参见 Adaptive Server Enterprise 《系统管理指南》。

审计

- 审计事件号 110 对应于 `sp_webservices` 的 `deploy` 选项。
- 审计事件号 111 对应于 `sp_webservices` 的 `undeploy` 选项。

可以将用户定义的 Web 服务作为 Adaptive Server Enterprise 中的存储过程进行建模。在处理用户定义的 Web 服务过程中，Adaptive Server Enterprise 将使用现有审计范围为存储过程生成以下事件：

- 创建用户定义的 Web 服务 — 生成名为 “Create Procedure” 的事件 11
- 删除用户定义的 Web 服务 — 生成名为 “Drop Procedure” 的事件 28
- 执行用户定义的 Web 服务 — 生成名为 “Execution of Stored Procedure” 的事件 38

有关现有审计功能的详细信息，请参见 《系统管理指南》。

除现有审计功能外，Adaptive Server Enterprise 还为 `sp_webservices` 的 `deploy` 和 `undeploy` 选项提供了两个审计事件。

审计记录存储在 `sysaudits` 系统表中。可以使用以下命令启用针对 Web 服务的审计：

```
sp_audit "security", "all", "all", "on"
```

另请参见

命令 `create service`

文档 《Web 服务用户指南》。

sp_who

说明 报告有关当前所有的 Adaptive Server 用户和进程的信息，或有关特定用户或进程的信息。包括 thread_pool 列，用以说明 Adaptive Server 用来执行任务的线程池。

有关进程模式的考虑事项 sp_who 不包括 threadpool 列。

语法 sp_who [*loginame* | "*spid*"]

参数 *loginame*

是一个用户的 Adaptive Server 登录名，您请求报告的就是该用户的情况。

spid

是要报告其相关信息的进程的编号。请用引号将进程编号引起（Adaptive Server 要求编号为 char 类型）。

示例 **示例 1** 报告 Adaptive Server 上运行的进程的情况。虽然除 sp_who 外没有任何其它用户进程正在运行，服务器仍会显示活动。在空闲循环期间，管家清洗任务将脏缓冲区移动到缓冲区清洗区域中，管家杂事任务执行其它维护任务。管家碎片收集任务以普通用户的优先级工作。该任务清除在逻辑上已被删除的数据并重置各行，使表重新拥有空间。

```
sp_who
-----
```

| fid | spid | status | loginame | origname | hostname | blk_spid | dbname |
|-----|------------|-------------------|----------|-------------|------------|------------------|--------|
| | tempdbname | cmd | | block_xloid | threadpool | | |
| 0 | 2 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | DEADLOCK TUNE | | | 0 | syb_default_pool | |
| 0 | 3 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | ASTC HANDLER | | | 0 | syb_default_pool | |
| 0 | 4 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | CHECKPOINT SLEEP | | | 0 | syb_default_pool | |
| 0 | 5 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | HK WASH | | | 0 | syb_default_pool | |
| 0 | 6 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | HK GC | | | 0 | syb_default_pool | |
| 0 | 7 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | HK CHORES | | | 0 | syb_default_pool | |
| 0 | 8 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | PORT MANAGER | | | 0 | syb_default_pool | |
| 0 | 9 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | NETWORK HANDLER | | | 0 | syb_default_pool | |
| 0 | 10 | sleeping | NULL | NULL | NULL | 0 | master |
| | tempdb | LICENSE HEARTBEAT | | | 0 | syb_default_pool | |
| 0 | 13 | sleeping | NULL | NULL | NULL | 0 | master |

```

tempdb          NETWORK HANDLER          0  syb_default_pool
0  14  sleeping      NULL      NULL      NULL      0      master
tempdb          NETWORK HANDLER          0  syb_default_pool
0  17  sleeping      NULL      NULL      NULL      0      master
tempdb          NETWORK HANDLER          0  syb_default_pool
0  20  sleeping      NULL      NULL      NULL      0      master
tempdb          NETWORK HANDLER          0  syb_default_pool
0  26  running        sa      sa  tiger.sybase.com  0      master
tempdb          INSERT          0  syb_default_pool

```

示例 2 报告 Adaptive Server 上运行的进程的情况。进程 11（对表执行 `select into`）被进程 8（对同一个表执行 `begin transaction`，随后对其执行 `insert`）阻塞。对于进程 8，当前 `loginame` 为 “robert”，但原 `loginame` 为 “sa”。登录名 “sa” 执行了一个 `set proxy` 命令来模拟用户 “robert”：

sp_who

```

fid  spid  status      loginame  origname  hostname      blk_spid  dbname
tempdbname  cmd      block_xloid  threadpool
-----
0    1  recv sleep      bird      bird      jazzy         0      master
tempdb  AWAITING COMMAND  0      syb_default_pool
0    2  sleeping      NULL      NULL      0      syb_default_pool
tempdb  NETWORK HANDLER  0
0    3  sleeping      NULL      NULL      0      syb_default_pool
tempdb  MIRROR HANDLER  0
0    4  sleeping      NULL      NULL      0      syb_default_pool
tempdb  AUDIT PROCESS  0
0    5  sleeping      NULL      NULL      0      syb_default_pool
tempdb  CHECKPOINT SLEEP  0
0    6  recv sleep      rose      rose petal   0      master
tempdb  AWAITING COMMAND  0      syb_default_pool
0    7  sleeping      NULL      NULL      actor        0      sybsystemdb
tempdb  ASTC HANDLER  0      syb_default_pool
0    8  running        robert    sa      helos         0      master
tempdb  SELECT          0      syb_default_pool
0    9  send sleep      daisy     daisy     chain         0      pubs2
tempdb  SELECT          0      syb_default_pool
0   10  alarm sleep      lily     lily     pond          0      master
tempdb  WAITFOR         0      syb_default_pool
0   11  lock sleep      viola    viola    cello         8      pubs2
tempdb  INSERT          0      syb_default_pool

```

示例 3 报告用户 “joe” 正在运行的进程的情况：

```
sp_who joe
fid spid status  loginame  origname  hostname          blk_spid  dbname
  tempdbname cmd          block_xloid  threadpool
-----
0   28  recv sleep    joe       joe  tiger.sybase.com    0         pubs2
      tempdb          SELECT    0         syb_default_pool
```

示例 4 报告编号为 17 的 Adaptive Server 进程正在执行什么操作：

```
sp_who "17"
fid spid status  loginame  origname  hostname          blk_spid  dbname
  tempdbname cmd          block_xloid  threadpool
-----
0   17  sleeping NULL     NULL     NULL              0         pubs2
      tempdb  NETWORK HANDLER 0         syb_default_pool
```

示例 5 报告有关一个系统引起的回退的信息，该回退或者属于一个事务，或者属于一个命令：

```
sp_who
fid spid status  loginame  origname  hostname          blk_spid  dbname
  tempdbname cmd          block_xloid  threadpool
-----
0   28  running  joe       joe  tiger.sybase.com    0         pubs2
      tempdb  rollback  0         syb_default_pool
```

用法

- `sp_who` 报告有关指定的用户或 Adaptive Server 进程的信息。
- 如果不指定参数，`sp_who` 将报告所有数据库中哪些用户正在运行哪些进程。
- `sp_who` 返回的列包括：

| 列 | 说明 |
|----------|---|
| fid | 标识锁属于哪个系列（包括协调进程及其工作进程）。有关详细信息，请参见 sp_familylock 。 |
| spid | 标识进程编号。系统管理员可以将此编号与 Transact-SQL <code>kill</code> 命令搭配使用来停止相应进程。 |
| status | 指示该进程是处于运行状态还是处于休眠状态。 |
| loginame | 启动该进程的用户的登录名或别名。对于所有系统进程， <code>loginame</code> 均为 NULL。 |
| origname | 如果 <code>loginame</code> 是一个别名， <code>origname</code> 会显示实际登录名。否则， <code>origname</code> 显示的信息与 <code>loginame</code> 相同。 |
| hostname | 相应数据库所在的服务器的名称。 |

| 列 | 说明 |
|-------------|--|
| blk_spid | 如果有阻塞进程，则包含该进程的进程 ID。阻塞进程（可能是受影响的进程或持有排它锁）是指持有另一进程所需要的资源的进程。 |
| dbname | 指示正在运行该进程的数据库的名称。 |
| tempdb | 分配给会话的临时数据库。 |
| cmd | 标识当前正在执行的命令或进程。对条件语句（例如，if 或 while 循环）进行计算将返回 cond。 |
| block_xloid | 标识阻塞事务的唯一锁所有者 ID。 |
| threadpool | 任务使用的线程池。 |

- 在单引擎服务器上运行 `sp_who` 将显示当前正在运行的 `sp_who` 进程以及所有其它可运行或处于其中一种休眠状态的进程。在多引擎服务器中，每个引擎都可能会有一个“正在运行”的进程。
- 如果启用了镜像磁盘或远程过程调用，镜像处理器和站点处理器也会出现在 `sp_who` 输出的报告中。

权限 任何用户都可以执行 `sp_who`。

审计 `sysaudits` 表的 `event` 和 `extrainfo` 列中的值如下所示：

| 事件 | 审计选项 | 审计的命令或访问权限 | extrainfo 中的信息 |
|----|----------------|------------|--|
| 38 | exec_procedure | 执行过程 | <ul style="list-style-type: none"> • 角色 — 当前活动角色 • 关键字或选项 — NULL • 以前值 — NULL • 当前值 — NULL • 其它信息 — 所有输入参数 • 代理信息 — <code>set proxy</code> 有效时的初始登录名 |

另请参见 **命令** `kill`

系统过程 `sp_familylock`、`sp_lock`

sp_xmlschema

| | |
|----|--|
| 说明 | 创建并维护 Adaptive Server 数据库中的 <code>spt_xmlcatalog</code> 用户表。 <code>spt_xmlcatalog</code> 存储供 <code>xmlvalidate</code> 函数用来验证 XML 文档的模式定义。 |
| 语法 | 有关 <code>sp_xmlschema</code> 的语法、示例和用法的信息，请参见《XML 服务》一书。 |
| 用法 | |

目录存储过程

本章介绍可以从表形式的系统表中检索信息的目录存储过程。

涉及的主题包括：

| 主题 | 页码 |
|---------------------------|-----|
| 概述 | 679 |
| 指定可选参数 | 680 |
| 模式匹配 | 681 |
| 系统过程表 | 681 |
| ODBC 数据类型 | 681 |

概述

表 2-1 列出了本章涉及的目录存储过程。

表 2-1：目录存储过程

| 过程 | 说明 |
|--------------------------------------|--|
| sp_column_privileges | 返回表或视图中的一列或多列的权限信息。 |
| sp_columns | 返回有关可以存储在一列或多列中的数据类型的信息。 |
| sp_databases | 返回 Adaptive Server 中的数据库的列表。 |
| sp_datatype_info | 返回有关特定数据类型或所有受支持数据类型的信息。 |
| sp_fkeys | 返回使用 <code>create table</code> 或 <code>alter table</code> 命令在当前数据库中创建的外键约束的相关信息。 |
| sp_pkeys | 返回使用 <code>create table</code> 或 <code>alter table</code> 命令为单个表创建的主键约束的相关信息。 |
| sp_server_info | 返回 Adaptive Server 属性名和当前值的列表。 |
| sp_special_columns | 返回能够唯一地标识表或视图中的行的一组最佳列；还可以返回在事务更新了行中的任意值后随之自动更新的一组列。 |
| sp_sproc_columns | 返回有关存储过程的输入参数和返回参数的信息。 |
| sp_statistics | 返回单个表的索引列表。 |
| sp_stored_procedures | 返回有关一个或多个存储过程的信息。 |
| sp_table_privileges | 返回表或视图中的所有列的特权信息。 |
| sp_tables | 返回可以出现在 <code>from</code> 子句中的对象的列表。 |

目录存储过程用于在表形式的系统表中检索信息。

目录存储过程是在安装时通过 `installmaster` 创建的，这些过程位于 `sybsystemprocs` 数据库中，归系统管理员所拥有。

许多目录存储过程都可以从任意数据库运行。如果从 `sybsystemprocs` 之外的数据库执行目录存储过程，那么，从哪个数据库执行它，它就从哪个数据库中的系统表中检索信息。

所有目录存储过程都在隔离级别 1 上执行。

所有目录存储过程都会报告返回状态。例如，下面的语句意味着过程执行成功。本书中的示例不包括返回状态：

```
return status = 0
```

指定可选参数

如果目录存储过程的参数值包含标点符号或嵌入的空格，或者该值是保留字，则必须用单引号或双引号引起来。如果参数是由数据库名或所有者名限定的对象名，则整个名称必须用单引号或双引号括起来。

注释 不要将分隔标识符用作目录存储过程参数。这样做可能导致意外结果。

在多数情况下，更为简便的方式是按照以下形式为目录存储过程提供参数：

```
@parametername = value
```

而不是提供所有参数。语法语句中的参数名称要与过程定义的参数名称匹配。

例如，`sp_columns` 的语法如下：

```
sp_columns table_name [, table_owner]
           [, table_qualifier] [, column_name]
```

要使用 `sp_columns` 查找有关特定列的信息，可使用：

```
sp_columns publishers, @column_name = "pub_id"
```

该命令提供的信息与指定了所有参数的命令所提供的信息相同：

```
sp_columns publishers, "dbo", "pubs2", "pub_id"
```

也可以用“null”作为占位符：

```
sp_columns publishers, null, null, "pub_id"
```

如果指定的参数个数比系统过程需要的参数多，Adaptive Server 会忽略多余的参数。

模式匹配

Adaptive Server 通过正则表达式提供范围广泛的模式匹配。不过，为实现最大程度的互用性，将只采用 SQL 标准模式匹配（% 和 _ 通配符）。

系统过程表

目录存储过程 `sp_columns`、`sp_datatype_info`、`sp_special_columns` 和 `sp_sproc_columns` 使用 `sybssystemprocs` 数据库中的目录存储过程表 `spt_datatype_info`、`spt_datatype_info_ext` 和 `spt_server_info` 将内部系统值（如状态位）转换为人工可读的格式。

目录存储过程 `sp_column_privileges` 和 `sp_table_privileges` 会先创建临时表，再将其删除。

ODBC 数据类型

表 2-2 和表 2-3 列出了由 `sp_columns` 和 `sp_sproc_columns` 在 `data_type` 列中返回的数据类型代码编号及与之匹配的数据类型名。对数据类型代码编号的描述源自开放式数据库互连 (ODBC) 应用程序编程接口 (API)。

表 2-2: ODBC 数据类型的代码编号

| 数据类型 | 代码编号 |
|------------------|------|
| char | 1 |
| decimal | 3 |
| double precision | 8 |
| float | 6 |
| integer | 4 |
| numeric | 2 |
| real | 7 |
| smallint | 5 |
| varchar | 12 |
| wchar | -8 |
| wvarchar | -9 |
| wlongvarchar | -10 |

表 2-3: 扩展数据类型的代码编号

| 数据类型 | 代码编号 |
|------------------------------|------|
| bigint | -5 |
| binary (bit 数据类型) | -2 |
| bit | -7 |
| date | 9 |
| java.lang.Object | 1111 |
| long univarchar | -10 |
| long varbinary | -4 |
| long varchar | -1 |
| time | 10 |
| timestamp | 11 |
| tinyint | -6 |
| unichar | -8 |
| univarchar | -9 |
| varbinary (bit-varying 数据类型) | -3 |

sp_column_privileges

| | |
|----|--|
| 说明 | 返回表或视图中的一列或多列的权限信息。 |
| 语法 | <code>sp_column_privileges table_name [, table_owner [, table_qualifier [, column_name]]]</code> |
| 参数 | <p>table_name 是表的名称。不支持在模式匹配中使用通配符。</p> <p>table_owner 是表所有者的名称。不支持在模式匹配中使用通配符。如果不指定表所有者，<code>sp_column_privileges</code> 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。</p> <p>table_qualifier 是数据库的名称。值为当前数据库的名称和 <code>null</code>。</p> <p>column_name 是要显示其权限的列的名称。使用通配符可请求多个列的信息。如果不指定列名，将返回指定表中所有列的权限信息。</p> |

示例

```
sp_column_privileges discounts, null, null, discounttype
table_qualifier      table_owner      table_name      column_name
  grantor      grantee      privilege      is_grantable
-----
pubs2          dbo          discounts      discounttype
  dbo          dbo          SELECT          YES
pubs2          dbo          discounts      discounttype
  dbo          dbo          UPDATE          YES
pubs2          dbo          discounts      discounttype
  dbo          dbo          REFERENCE      YES
pubs2          dbo          discounts      discounttype
  dbo          guest        SELECT          NO
pubs2          dbo          discounts      discounttype
  dbo          guest        UPDATE          NO
pubs2          dbo          discounts      discounttype
  dbo          guest        REFERENCE      NO
```

用法

- sp_column_privileges 的结果集如下：

| 列 | 数据类型 | 说明 |
|-----------------|-------------|--|
| table_qualifier | varchar(32) | 为 <i>table_name</i> 参数指定的表所在的数据库的名称。 |
| table_owner | varchar(32) | 表所有者。如果没有为 <i>table_owner</i> 参数指定值，则该值为当前所有者或数据库所有者。 |
| table_name | varchar(32) | 为 <i>table_name</i> 参数指定的名称。该值不能为 NULL。 |
| column_name | varchar(32) | 指定的列名。如果没有在语句中指定列名，结果中将包含指定表中的所有列。 |
| grantor | varchar(32) | 将 <i>column_name</i> 的权限授予 <i>grantee</i> 的数据库用户的用户名。该值不能为 NULL。 |
| grantee | varchar(32) | 被 <i>grantor</i> 授予 <i>column_name</i> 的权限的数据库用户的用户名。该值不能为 NULL。 |
| privilege | varchar(32) | 标识列特权。可以是下列之一： <ul style="list-style-type: none">• SELECT — 允许被授予者检索该列的数据。• UPDATE — 允许被授予者更新该列中的数据。• REFERENCE — 仅允许被授者将该列用于参照约束。 |
| is_grantable | varchar(3) | 指示是否允许被授予者将特权授予其他用户。值为 YES、NO 和 NULL。 |

权限

任何用户都可以执行 sp_column_privileges。

sp_columns

说明 返回有关可以存储在一列或多列中的数据类型的信息。

语法 `sp_columns table_name [, table_owner] [, table_qualifier] [, column_name]`

参数 *table_name*
是表名或视图名。使用通配符可请求有关多个表的信息。

table_owner
是表或视图的所有者。使用通配符可请求由多个用户拥有的表的相关信息。如果不指定表所有者，`sp_columns` 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。

table_qualifier
是数据库的名称。它可以是当前数据库，也可以是 NULL。

column_name
是列名，要获取的就是该列的信息。使用通配符可请求有关多个列的信息。

示例 **示例 1** 显示 `publishers` 表中以“p”开头的所有列的相关信息：

```
sp_columns "publishers", null, null, "p%"

table_qualifier table_owner table_name column_name data_type type_name
precision length scale radix nullable remarks ss_data_type colid
-----
-----
pubs2          dbo          publishers  pub_id      1          char
NULL          4           NULL       NULL       0          NULL       47          1
pubs2          dbo          publishers  pub_name    12         varchar
NULL          40          NULL       NULL       1          NULL       39          2
```

示例 2 显示以“s”开头的表中以“st”开头的所有列的相关信息：

```
sp_columns "s%", null, null, "st%"
```

用法 `sp_columns` 的结果集如下：

| 列 | 数据类型 | 说明 |
|------------------------------|--------------------------|---|
| <code>table_qualifier</code> | <code>varchar(32)</code> | 为 <code>table_name</code> 参数指定的表所在的数据库的名称。 |
| <code>table_owner</code> | <code>varchar(32)</code> | 表所有者。如果没有为 <code>table_owner</code> 参数指定值，则该值为当前所有者或数据库所有者。 |
| <code>table_name</code> | <code>varchar(32)</code> | NOT NULL。 |
| <code>column_name</code> | <code>varchar(32)</code> | NOT NULL。 |

| 列 | 数据类型 | 说明 |
|-------------------|--------------|--|
| data_type | smallint | ODBC 数据类型的整数代码。如果这个数据类型无法映射为 ODBC 类型，则为 NULL。 |
| type_name | varchar(30) | 表示数据类型的字符串。基础 DBMS 提供该数据类型的名称。 |
| precision | int | 有效位数。 |
| length | int | 数据类型的字节长度。 |
| scale | smallint | 小数点右边的位数。 |
| radix | smallint | 数值数据类型的基数。 |
| nullable | smallint | 值为 1 表示可以为 NULL；值为 0 则表示 NOT NULL。 |
| remarks | varchar(254) | |
| ss_data_type | smallint | 一种 Adaptive Server 数据类型。 |
| colid | tinyint | 附加到结果集内的列。 |
| column_def | varchar(255) | NULL。 |
| sql_data_type | smallint | 一种 Adaptive Server 数据类型。 |
| sql_datetime_sub | smallint | NULL。 |
| char_octet_length | int | 如果 char_octet_length 的数据类型为以下类型，char_octet_length 的值就与 precision 列的值相同： <ul style="list-style-type: none"> • binary • char • image • nchar • nvarchar • sysname • text • timestamp • varbinary • varchar 否则，char_octet_length 的值为 0。 |
| ordinal_position | int | 表中列的顺序位置。表中的第一列为 1。 |
| is_nullable | varchar(3) | 说明列或参数是否允许将 NULL 作为值。来自 syscolumns。 |

对于定义为

- double precision

的列，sp_columns 报告其 type_name 为 float，其 data_type 为 6。Adaptive Server double precision 数据类型是浮点类型的一种实现，它支持 ODBC 规范中指定的范围内的值。

权限

任何用户都可以执行 sp_columns。

sp_databases

说明 返回 Adaptive Server 中的数据库的列表。

语法 sp_databases

参数 无。

示例

```
sp_databases

database_name      database_size      remarks
-----
master             5120              NULL
model              2048              NULL
mydb               2048              NULL
pubs2              2048              NULL
sybsecurity        5120              NULL
sybssystemprocs    16384             NULL
tempdb             2048              NULL
```

用法

- sp_databases 的结果集如下：

| 列 | 数据类型 | 说明 |
|---------------|--------------|----------------------------|
| database_name | char(32) | NOT NULL 数据库名。 |
| database_size | int | 数据库大小（以 KB 计）。 |
| remarks | varchar(254) | Adaptive Server 始终返回 NULL。 |

权限

任何用户都可以执行 sp_databases。

sp_datatype_info

说明 返回有关特定 ODBC 数据类型或所有 ODBC 数据类型的信息。

语法 sp_datatype_info [*data_type*]

参数 *data_type*
是指定 ODBC 数据类型的代码编号，要返回的就是该数据类型的相关信息。数据类型代码列在[第 682 页的表 2-2](#)和[第 682 页的表 2-3](#)中。

用法

- sp_datatype_info 的结果集如下：

| 列 | 数据类型 | 说明 |
|--------------------|--------------|--|
| type_name | varchar(30) | 与 DBMS 相关的数据类型名（与 sp_columns 结果集中的 type_name 列相同）。 |
| data_type | smallint | 某一 ODBC 类型的所有列都映射到的 ODBC 类型的代码。 |
| precision | int | 数据源上的数据类型的最大精度。不适用精度的数据类型返回零。 |
| literal_prefix | varchar(32) | 文字开头使用的一个或多个字符。例如，字符类型可使用单引号 (')，二进制类型可使用 0x。 |
| literal_suffix | varchar(32) | 文字结尾使用的一个或多个字符。例如，字符类型可使用单引号 (')，而二进制类型不使用任何字符。 |
| create_params | varchar(32) | 关于此数据类型的创建参数的说明。 |
| nullable | smallint | 值为 1 表示创建的这种数据类型可支持空值；值为 0 则表示不支持。 |
| case_sensitive | smallint | 值为 1 表示此类型的所有列都区分大小写（对于归类）；值为 0 则表示不区分大小写。 |
| searchable | smallint | 值为 1 表示此类型的列可用在 where 子句中。 |
| unsigned_attribute | smallint | 值为 1 表示数据类型无符号；值为 0 则表示数据类型有符号。 |
| money | smallint | 值为 1 表示属于 money 数据类型；值为 0 则表示不属于。 |
| auto_increment | smallint | 值为 1 表示该数据类型自动递增；值为 0 则表示不自动递增。 |
| local_type_name | varchar(128) | 数据类型的数据源相关名称的本地化版本。 |

权限 任何用户都可以执行 sp_datatype_info。

sp_fkeys

- 说明** 返回使用 `create table` 或 `alter table` 命令在当前数据库中创建的外键约束的相关信息。
- 语法**
- ```
sp_fkeys pktable_name [, pktable_owner]
 [, pktable_qualifier] [, fktable_name]
 [, fktable_owner] [, fktable_qualifier]
```
- 参数**
- pktable\_name*  
是主键表的名称。不支持在模式匹配中使用通配符。必须指定 *pktable\_name* 或 *fktable\_name*，或同时指定两者。
- pktable\_owner*  
是主键表所有者的名称。不支持在模式匹配中使用通配符。如果不指定表所有者，`sp_fkeys` 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。
- pktable\_qualifier*  
是包含主键表的数据库的名称。它可以是当前数据库，也可以是 NULL。
- fktable\_name*  
是外键表的名称。不支持在模式匹配中使用通配符。必须指定 *fktable\_name* 或 *pktable\_name*，或同时指定两者。
- fktable\_owner*  
是外键表所有者的名称。不支持在模式匹配中使用通配符。如果未指定 *fktable\_owner*，`sp_fkeys` 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。
- fktable\_qualifier*  
是包含外键表的数据库的名称。它可以是当前数据库，也可以是 null。
- 用法**
- `sp_fkeys` 返回使用 `create table` 或 `alter table` 命令在当前数据库中创建的外键约束的相关信息。外键是表中在逻辑上依赖于另一个表的主键列的键列。
  - `sp_fkeys` 的结果集如下：

| 列                        | 数据类型        | 说明         |
|--------------------------|-------------|------------|
| <i>pktable_qualifier</i> | varchar(32) | 包含主键表的数据库。 |
| <i>pktable_owner</i>     | varchar(32) | 主键表的所有者。   |
| <i>pktable_name</i>      | varchar(32) | NOT NULL。  |
| <i>pkcolumn_name</i>     | varchar(32) | NOT NULL。  |
| <i>fktable_qualifier</i> | varchar(32) | 包含外键表的数据库。 |

| 列             | 数据类型        | 说明                                  |
|---------------|-------------|-------------------------------------|
| fktable_owner | varchar(32) | 外键表的所有者。                            |
| fktable_name  | varchar(32) | NOT NULL。                           |
| fkcolumn_name | varchar(32) | NOT NULL。                           |
| key_seq       | smallint    | NOT NULL。多列主键中的列的序列号。               |
| update_rule   | smallint    | 当 SQL 操作为 UPDATE 时对外键应用的操作。为该列返回零值。 |
| delete_rule   | smallint    | 当 SQL 操作为 DELETE 时对外键应用的操作。为该列返回零值。 |

- 主键和外键都必须已经在 `create table` 或 `alter table` 语句中声明。
- 如果提供了主键表名，而外键表名为 `NULL`，`sp_fkeys` 将返回包含引用给定表的外键的所有表。如果提供了外键表名，而主键表名为 `NULL`，则 `sp_fkeys` 将返回所有通过主键 / 外键关系与该外键表中的外键关联的表。
- `sp_fkeys` 不返回使用 `sp_commonkey`、`sp_foreignkey` 或 `sp_primarykey` 声明的键的相关信息。

权限

任何用户都可以执行 `sp_fkeys`。

## sp\_pkeys

**说明** 返回使用 `create table` 或 `alter table` 命令为单个表创建的主键约束的相关信息。

**语法** `sp_pkeys table_name [, table_owner] [, table_qualifier]`

**参数** `table_name`

是表的名称。不支持在模式匹配中使用通配符。

`table_owner`

是表所有者的名称。不支持在模式匹配中使用通配符。如果未指定 `table_owner`，`sp_pkeys` 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。

`table_qualifier`

是包含该表的数据库的名称。它可以是当前数据库，也可以是 NULL。

**用法**

- `sp_pkeys` 的结果集如下：

| 列                            | 数据类型                     | 说明                                                          |
|------------------------------|--------------------------|-------------------------------------------------------------|
| <code>table_qualifier</code> | <code>varchar(32)</code> | 数据库名。该字段可以为 NULL。                                           |
| <code>table_owner</code>     | <code>varchar(32)</code> | 表所有者。如果没有为 <code>table_owner</code> 参数指定值，则该值为当前所有者或数据库所有者。 |
| <code>table_name</code>      | <code>varchar(32)</code> | NOT NULL。                                                   |
| <code>column_name</code>     | <code>varchar(32)</code> | NOT NULL。                                                   |
| <code>key_seq</code>         | <code>smallint</code>    | NOT NULL。多列主键中的列的序列号。                                       |

- 必须已经使用 `create table` 或 `alter table` 语句而不是 `sp_primarykey` 声明了主键。
- **主键** 一词指表的逻辑主键。Adaptive Server 期望每个逻辑主键都定义一个唯一索引，并且该唯一索引也在 `sp_statistics` 中返回。

**权限**

任何用户都可以执行 `sp_pkeys`。

## sp\_server\_info

**说明** 返回 Adaptive Server 属性名和当前值的列表。

**语法** sp\_server\_info [attribute\_id]

**参数** attribute\_id  
是服务器属性的整型 ID。

**示例 1**

```
sp_server_info 12
attribute_id attribute_name attribute_value

12 MAX_OWNER_NAME_LENGTH 0
```

**示例 2** 返回通过必需行描述的服务器属性及其值的列表：

```
sp_server_info
```

**用法** sp\_server\_info 的结果集如下：

| 列               | 数据类型         | 说明        |
|-----------------|--------------|-----------|
| attribute_id    | int          | NOT NULL。 |
| attribute_name  | varchar(60)  | NOT NULL。 |
| attribute_value | varchar(255) |           |

由 sp\_server\_info 返回的结果集内的强制行包括：

| ID | 服务器属性名                | 说明                                           | 值          |
|----|-----------------------|----------------------------------------------|------------|
| 1  | DBMS_NAME             | DBMS 的名称。                                    | SQL SERVER |
| 2  | DBMS_VER              | DBMS 的版本。                                    | @@version  |
| 6  | DBE_NAME              | 未使用                                          |            |
| 10 | OWNER_TERM            | Adaptive Server 中表示表所有者的术语（由三部分构成的名称中的第二部分）。 | owner      |
| 11 | TABLE_TERM            | Adaptive Server 中表示表的术语（由三部分构成的名称中的第三部分）。    | table      |
| 12 | MAX_OWNER_NAME_LENGTH | 表所有者名称的最大长度（由三部分构成的名称中的第二部分）。                | 30         |
| 13 | TABLE_LENGTH          | 表名的最大字符数。                                    | 30         |
| 14 | MAX_QUAL_LENGTH       | 表限定符名称的最大长度（由三部分构成的表名中的第一部分）。                | 30         |
| 15 | COLUMN_LENGTH         | 列名的最大字符数。                                    | 30         |

| ID  | 服务器属性名                 | 说明                                                                                     | 值          |
|-----|------------------------|----------------------------------------------------------------------------------------|------------|
| 16  | IDENTIFIER_CASE        | 表示数据库中的用户定义名称（表名、列名和存储过程名）是否区分大小写（指这些对象显示在系统目录中的大小写形式）。                                | MIXED      |
| 18  | COLLATION_SEQ          | 此服务器采用的字符集顺序。                                                                          |            |
| 19  | SAVEPOINT_SUPPORT      | 基础 DBMS 是否支持指定的保存点？                                                                    | Y          |
| 20  | MULTI_RESULT_SETS      | 基础 DBMS 或网关本身是否支持多个结果集（是否可通过网关发送多个语句，并将多个结果集返回给客户端）？                                   | Y          |
| 22  | ACCESSIBLE_TABLES      | 在 <code>sp_tables</code> 中，网关是否只返回当前用户（即至少对表拥有 <code>select</code> 特权的用户）可以访问的表、视图等对象？ | Y          |
| 100 | USERID_LENGTH          | 用户名的最大字符数。                                                                             | 30         |
| 101 | QUALIFIER_TERM         | Adaptive Server 中表示表限定符的术语（由三部分构成的名称中的第一部分）。                                           | database   |
| 102 | NAMED_TRANSACTIONS     | 基础 DBMS 是否支持指定的事务？                                                                     | Y          |
| 103 | SPROC_AS_LANGUAGE      | 是否可以将存储过程作为语言事件执行？                                                                     | Y          |
| 103 | REMOTE_SPROC           | 是否可以通过 DB-Library 中的远程存储过程 API 执行存储过程？                                                 | Y          |
| 104 | ACCESSIBLE_SPROC       | 在 <code>sp_stored_procedures</code> 中，网关是否只返回可由当前用户执行的存储过程？                            | Y          |
| 105 | MAX_INDEX_COLS         | DBMS 的索引中的最大列数。                                                                        | 32         |
| 106 | RENAME_TABLE           | 是否可以重命名表？                                                                              | Y          |
| 107 | RENAME_COLUMN          | 是否可以重命名列？                                                                              | Y          |
| 108 | DROP_COLUMN            | 是否可以删除列？                                                                               | Y          |
| 109 | INCREASE_COLUMN_LENGTH | 是否可以增大列的大小？                                                                            | N          |
| 110 | DDL_IN_TRANSACTION     | 事务中是否可以出现 DDL 语句？                                                                      | Y          |
| 111 | DESCENDING_INDEXES     | 是否支持降序索引？                                                                              | Y          |
| 112 | SP_RENAME              | 是否可以重命名存储过程？                                                                           | Y          |
| 500 | SYS_SPROC_VERSION      | 当前实现的目录存储过程的版本。                                                                        | 01.01.2822 |

权限

任何用户都可以执行 `sp_server_info`。

## sp\_special\_columns

**说明** 返回唯一标识表或视图中的行的一组最佳列；还可以返回一个 `timestamp` 列的列表，其中的列值在事务更新行中的任意值后自动生成。

**语法** `sp_special_columns table_name [, table_owner] [, table_qualifier] [, col_type]`

**参数** *table\_name*

是表名或视图名。不支持在模式匹配中使用通配符。

*table\_owner*

是表或视图所有者的名称。不支持在模式匹配中使用通配符。如果不指定表所有者，`sp_special_columns` 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。

*table\_qualifier*

是数据库的名称。它可以是当前数据库，也可以是 NULL。

*col\_type*

为“R”时，返回有关一些列的信息，这些列中的值唯一标识表中的任一行；为“V”时，返回有关 `timestamp` 列的信息，这些列的值是在每次插入或更新行后由 Adaptive Server 生成的。

**示例** **示例 1** 返回 `systypes` 的一组最佳列：

```
sp_special_columns systypes
scope column_name data_type type_name precision length scale
----- -
0 name 12 varchar 30 30 NULL
```

**示例 2** 返回 `authors` 表的一组最佳列，这些列中的值唯一地标识表中的任一行：

```
sp_special_columns @table_name=authors, @col_type=R
scope column_name data_type type_name precision length scale
----- -
0 au_id 12 varchar 11 11 NULL
```



用法

- `sp_special_columns` 的结果集如下：

| 列                        | 数据类型                     | 说明                                                                                                           |
|--------------------------|--------------------------|--------------------------------------------------------------------------------------------------------------|
| <code>scope</code>       | <code>int</code>         | NOT NULL。行 ID 的实际范围。Adaptive Server 始终返回 0。                                                                  |
| <code>column_name</code> | <code>varchar(30)</code> | NOT NULL。列标识符。                                                                                               |
| <code>data_type</code>   | <code>smallint</code>    | ODBC 数据类型的整数代码。如果该数据类型无法映射为 ANSI/ISO 类型，则该值为 NULL。本机数据类型名在 <code>type_name</code> 列中返回。（请参见 ODBC 数据类型表 2-2。） |
| <code>type_name</code>   | <code>varchar(13)</code> | 数据类型的字符串表示形式。这是基础 DBMS 给出的数据类型名。                                                                             |
| <code>precision</code>   | <code>int</code>         | 有效位数。                                                                                                        |
| <code>length</code>      | <code>int</code>         | 数据类型的字节长度。                                                                                                   |
| <code>scale</code>       | <code>smallint</code>    | 小数点右边的位数。                                                                                                    |

权限

任何用户都可以执行 `sp_special_columns`。

## sp\_sproc\_columns

- 说明** 返回有关存储过程的输入参数和返回参数的信息。
- 语法** `sp_sproc_columns procedure_name [, procedure_owner] [, procedure_qualifier] [, column_name]`
- 参数**
- procedure\_name*  
是存储过程的名称。不支持在模式匹配中使用通配符。
- procedure\_owner*  
是存储过程的所有者。不支持在模式匹配中使用通配符。如果未指定所有者，`sp_sproc_columns` 将返回所有列。
- procedure\_qualifier*  
是数据库的名称。它可以是当前数据库，也可以是 NULL。
- column\_name*  
是要获取其相关信息的参数的名称。如果不提供参数名，`sp_sproc_columns` 将返回该存储过程的所有输入参数和返回参数的相关信息。

**用法** • `sp_sproc_columns` 的结果集如下：

| 列                                | 数据类型                      | 说明                                                                                      |
|----------------------------------|---------------------------|-----------------------------------------------------------------------------------------|
| <code>procedure_qualifier</code> | <code>varchar(30)</code>  | 过程限定符的名称。可以为 NULL。                                                                      |
| <code>procedure_owner</code>     | <code>varchar(30)</code>  | 过程所有者名。始终会返回值。                                                                          |
| <code>procedure_name</code>      | <code>varchar(41)</code>  | 过程名称。始终会返回值。                                                                            |
| <code>column_name</code>         | <code>varchar(30)</code>  | 返回 <i>table_name</i> 的每个列的列名。始终会返回值。                                                    |
| <code>column_type</code>         | <code>smallint</code>     |                                                                                         |
| <code>data_type</code>           | <code>smallint</code>     | ODBC 数据类型的整数代码。如果该数据类型无法映射为 ANSI/ISO 类型，则该值为 NULL。本机数据类型名在 <code>type_name</code> 列中返回。 |
| <code>type_name</code>           | <code>char(30)</code>     | 数据类型的字符串表示形式。这是基础 DBMS 给出的数据类型名。                                                        |
| <code>precision</code>           | <code>int</code>          | 有效位数。                                                                                   |
| <code>length</code>              | <code>int</code>          | 数据类型的字节长度。                                                                              |
| <code>scale</code>               | <code>smallint</code>     | 小数点右边的位数。                                                                               |
| <code>radix</code>               | <code>smallint</code>     | 数值类型的基数。                                                                                |
| <code>nullable</code>            | <code>smallint</code>     | 值为 1 表示创建的这种数据类型可支持空值；值为 0 则表示不支持。                                                      |
| <code>remarks</code>             | <code>varchar(254)</code> | 过程列的说明。NULL。                                                                            |
| <code>ss_data_type</code>        | <code>tinyint</code>      | 一种 Adaptive Server 数据类型。                                                                |
| <code>colid</code>               | <code>tinyint</code>      | <code>syscolumns</code> 的列 ID。                                                          |
| <code>column_def</code>          | <code>varchar(255)</code> | NULL。                                                                                   |
| <code>sql_data_type</code>       | <code>smallint</code>     | 一种 Adaptive Server 数据类型。                                                                |
| <code>sql_datetime_sub</code>    | <code>smallint</code>     | NULL。                                                                                   |

| 列                 | 数据类型        | 说明                                                                                                                                                                                                                                                                                                                                                 |
|-------------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| char_octet_length | int         | <p>如果 char_octet_length 的数据类型为以下类型，char_octet_length 的值就与 precision 列的值相同：</p> <ul style="list-style-type: none"> <li>• binary</li> <li>• char</li> <li>• image</li> <li>• nchar</li> <li>• nvarchar</li> <li>• sysname</li> <li>• text</li> <li>• timestamp</li> <li>• varbinary</li> <li>• varchar</li> </ul> <p>否则，char_octet_length 的值为 0。</p> |
| ordinal_position  | int         | 参数列表中参数的顺序位置。列表中的第一个参数为 1，返回的值具有序数。                                                                                                                                                                                                                                                                                                                |
| is_nullable       | varchar(3)  | 说明列或参数是否允许将 NULL 作为值。来自 syscolumns。                                                                                                                                                                                                                                                                                                                |
| mode              | varchar(20) | <p>存储在 syscolumns 中的参数模式信息，包含：</p> <ul style="list-style-type: none"> <li>• 对于 SQL 过程 — in、out 或 “返回值”。</li> <li>• 对于 SQLJ 过程 (Java) — in、out、inout 或 “返回值”。</li> </ul>                                                                                                                                                                            |

对于定义为

- double precision

的参数，sp\_sproc\_columns 报告其 type\_name 为 float，data\_type 为 6。Adaptive Server double precision 数据类型是浮点类型的一种实现，它支持 ODBC 规范中指定的范围内的值。

权限

任何用户都可以执行 sp\_sproc\_columns。

## sp\_statistics

- 说明** 返回单个表的索引列表。
- 语法** `sp_statistics table_name [, table_owner] [, table_qualifier] [, index_name] [, is_unique]`
- 参数**
- table\_name**  
是表的名称。不支持在模式匹配中使用通配符。
  - table\_owner**  
是表的所有者。不支持在模式匹配中使用通配符。如果未指定 *table\_owner*, `sp_statistics` 会先查找当前用户拥有的表, 再查找数据库所有者拥有的表。
  - table\_qualifier**  
是数据库的名称。它可以是当前数据库, 也可以是 NULL。
  - index\_name**  
是索引名。不支持在模式匹配中使用通配符。
  - is\_unique**  
为 Y 时仅返回唯一索引; 为 N 时, 唯一和不唯一的索引都返回。

**示例** `sp_statistics publishers`

```

table_qualifier table_owner
table_name non_unique
index_qualifier index_name
type seq_in_index column_name collation
cardinality pages

pubs2 dbo
publishers NULL
NULL NULL
0 NULL NULL NULL
3 1
pubs2 dbo
publishers 0
publishers pubind
1 1 pub_id A
3 1

```

用法

- `sp_statistics` 的结果集如下：

| 列                            | 数据类型                     | 说明                                        |
|------------------------------|--------------------------|-------------------------------------------|
| <code>table_qualifier</code> | <code>varchar(32)</code> | 数据库名。该字段可以为 NULL。                         |
| <code>table_owner</code>     | <code>varchar(32)</code> |                                           |
| <code>table_name</code>      | <code>varchar(32)</code> | NOT NULL。                                 |
| <code>non_unique</code>      | <code>smallint</code>    | NOT NULL。值为 0 表示唯一，值为 1 表示不唯一。            |
| <code>index_qualifier</code> | <code>varchar(32)</code> |                                           |
| <code>index_name</code>      | <code>varchar(32)</code> |                                           |
| <code>type</code>            | <code>smallint</code>    | NOT NULL。值为 0 表示聚簇，值为 2 表示散列，值为 3 表示其它情况。 |
| <code>seq_in_index</code>    | <code>smallint</code>    | NOT NULL。                                 |
| <code>column_name</code>     | <code>varchar(32)</code> | NOT NULL。                                 |
| <code>collation</code>       | <code>char(1)</code>     | 值为 A 表示升序；值为 D 表示降序；值为 NULL 表示不适用。        |
| <code>cardinality</code>     | <code>int</code>         | 表中的行数或索引中的唯一值数。                           |
| <code>pages</code>           | <code>int</code>         | 用于存储索引或表的页数。                              |

- 结果集内的索引以升序显示，按照 `non-unique`、`type`、`index_name` 和 `seq_in_index` 各列进行排序。
- 索引类型 `hashed` 接受精确匹配或范围搜索，但涉及模式匹配的搜索并不使用该索引。

权限

任何用户都可以执行 `sp_statistics`。

## sp\_stored\_procedures

- 说明** 返回有关一个或多个存储过程的信息。
- 语法** sp\_stored\_procedures [sp\_name [, sp\_owner [, sp\_qualifier]]]
- 参数**
- sp\_name**  
是存储过程的名称。使用通配符可请求有关多个存储过程的信息。
- sp\_owner**  
是存储过程的所有者。使用通配符可请求归多个用户拥有的过程的相关信息。
- sp\_qualifier**  
是数据库的名称。它可以是当前数据库，也可以是 NULL。

- 用法**
- sp\_stored\_procedures 仅返回当前数据库中的存储过程的相关信息。
  - sp\_stored\_procedures 的结果集如下：

| 列                   | 数据类型         | 说明                                       |
|---------------------|--------------|------------------------------------------|
| procedure_qualifier | varchar(30)  | 数据库的名称。                                  |
| procedure_owner     | varchar(30)  |                                          |
| procedure_name      | varchar(41)  | NOT NULL。                                |
| num_input_params    | int          | NOT NULL。始终返回 -1。                        |
| num_output_params   | int          | NOT NULL。值 $\geq 0$ 显示参数个数；-1 表示参数个数不确定。 |
| num_result_sets     | int          | NOT NULL。始终返回 -1。                        |
| remarks             | varchar(254) | NULL。                                    |

- sp\_stored\_procedures 可以返回当前用户不具有其执行权限的存储过程的名称。不过，如果 [sp\\_server\\_info](#) 的结果集内的服务器属性 `accessible_sproc` 为 “Y”，则只返回可由当前用户执行的存储过程。

- 权限** 任何用户都可以执行 sp\_stored\_procedures。

## sp\_table\_privileges

|    |                                                                                                                                                                                                                                                |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 返回表或视图中的所有列的特权信息。                                                                                                                                                                                                                              |
| 语法 | <code>sp_table_privileges table_name [, table_owner[, table_qualifier]]</code>                                                                                                                                                                 |
| 参数 | <p><i>table_name</i><br/>是表的名称。不支持在模式匹配中使用通配符。</p> <p><i>table_owner</i><br/>是表所有者的名称。不支持在模式匹配中使用通配符。如果不指定表所有者，<code>sp_table_privileges</code> 会先查找当前用户拥有的表，再查找数据库所有者拥有的表。</p> <p><i>table_qualifier</i><br/>是数据库的名称。它可以是当前数据库，也可以是 NULL。</p> |

用法

- `sp_table_privileges` 的结果集如下：

| 列                            | 数据类型                     | 说明                                                                                                                                                                                                                                                          |
|------------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>table_qualifier</code> | <code>varchar(32)</code> | 数据库的名称。该字段可以为 NULL。                                                                                                                                                                                                                                         |
| <code>table_owner</code>     | <code>varchar(32)</code> |                                                                                                                                                                                                                                                             |
| <code>table_name</code>      | <code>varchar(32)</code> | NOT NULL。                                                                                                                                                                                                                                                   |
| <code>grantor</code>         | <code>varchar(32)</code> | NOT NULL。                                                                                                                                                                                                                                                   |
| <code>grantee</code>         | <code>varchar(32)</code> | NOT NULL。                                                                                                                                                                                                                                                   |
| <code>privilege</code>       | <code>varchar(32)</code> | 标识表特权。可以是下列之一： <ul style="list-style-type: none"> <li>• SELECT — 允许被授予者检索表的一列或多列数据。</li> <li>• INSERT — 允许被授予者插入包含数据的行。</li> <li>• UPDATE — 允许被授予者更新表的一列或多列中的数据。</li> <li>• DELETE — 允许被授予者从表中删除数据行。</li> <li>• REFERENCE — 允许被授予者在约束中引用表的一列或多列。</li> </ul> |
| <code>is_grantable</code>    | <code>varchar(3)</code>  | 指示是否允许被授予者将特权授予其他用户。值为 YES、NO 和 NULL。                                                                                                                                                                                                                       |

权限 任何用户都可以执行 `sp_table_privileges`。

## sp\_tables

|    |                                                                                                                                                                                                                                                                                              |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 返回可以出现在 from 子句中的对象的列表。                                                                                                                                                                                                                                                                      |
| 语法 | sp_tables [table_name] [, table_owner][, table_qualifier][, table_type]                                                                                                                                                                                                                      |
| 参数 | <p><b>table_name</b><br/>是表的名称。使用通配符可请求有关多个表的信息。</p> <p><b>table_owner</b><br/>是表的所有者。使用通配符可请求有关多个表的信息。</p> <p><b>table_qualifier</b><br/>是数据库的名称。可接受的值包括当前数据库的名称和 NULL。</p> <p><b>table_type</b><br/>是一组值，用逗号分隔，提供属于指定表类型的所有表的相关信息，其中包括以下类型：</p> <pre>'TABLE', 'SYSTEM TABLE', 'VIEW'</pre> |

---

**注释** 应将各种表类型放在单引号之内，并将整个参数放在双引号之内。应以大写形式输入表类型。

---

示例

```
sp_tables @table_type = 'TABLE', 'VIEW'
```

此过程将返回当前数据库中类型为 TABLE 和 VIEW 的所有表的相关信息，但不包括系统表的相关信息。

- 用法
- Adaptive Server 不一定检查 *table\_name* 的读写权限。即便能够显示该表的相关信息，也不一定能访问该表。
  - 结果集包括表、视图以及 DBMS 产品网关的同义词和别名。
  - 如果 sp\_server\_info 的结果集内的服务器属性 accessible\_tables 为“Y”，则只返回当前用户可以访问的表。
  - sp\_tables 的结果集如下：

| 列               | 数据类型         | 说明                                            |
|-----------------|--------------|-----------------------------------------------|
| table_qualifier | varchar(30)  | 数据库名。该字段可以为 NULL。                             |
| table_owner     | varchar(30)  |                                               |
| table_name      | varchar(30)  | NOT NULL。表名。                                  |
| table_type      | varchar(32)  | NOT NULL。下列值之一：“TABLE”、“VIEW”、“SYSTEM TABLE”。 |
| remarks         | varchar(254) | NULL                                          |



|      |                                                                                                                                             |
|------|---------------------------------------------------------------------------------------------------------------------------------------------|
| 权限   | 任何用户都可以执行 <code>sp_tables</code> 。                                                                                                          |
| 使用的表 | <code>master.dbo.sysattributes</code> 、 <code>master.dbo.sysloginroles</code> 、 <code>master.dbo.syssrvroles</code> 、 <code>sysroles</code> |



# 系统扩展存储过程

本章介绍 Sybase 提供的各种系统扩展存储过程 (ESP)。ESP 是在安装期间由 `installmaster` 创建的。它们位于 `sybserverprocs` 数据库中，并由系统管理员拥有。可以从任意数据库运行这些过程。

涉及的主题包括：

| 主题                              | 页码  |
|---------------------------------|-----|
| <a href="#">概述</a>              | 705 |
| <a href="#">系统 ESP 的权限</a>      | 706 |
| <a href="#">与系统 ESP 关联的 DLL</a> | 706 |
| <a href="#">使用系统 ESP</a>        | 706 |

## 概述

表 3-1 列出了本章讨论的各种系统扩展存储过程。

**表 3-1：系统扩展存储过程**

| 过程                             | 说明                                    | 平台           |
|--------------------------------|---------------------------------------|--------------|
| <a href="#">xp_cmdshell</a>    | 在运行 Adaptive Server 的主机系统上执行本机操作系统命令。 | 所有支持 DLL 的平台 |
| <a href="#">xp_deletemail</a>  | 从 Adaptive Server 收件箱中删除邮件。           | 仅适用于 NT      |
| <a href="#">xp_enumgroups</a>  | 显示特定 Windows NT 域的组。                  | 仅适用于 NT      |
| <a href="#">xp_findnextmsg</a> | 在 Adaptive Server 收件箱中检索下一封邮件的邮件标识符。  | 仅适用于 NT      |
| <a href="#">xp_logevent</a>    | 用于在 Windows NT 事件日志中记录用户定义的事件。        | 仅适用于 NT      |
| <a href="#">xp_readmail</a>    | 查阅 Adaptive Server 收件箱中的邮件。           | 仅适用于 NT      |
| <a href="#">xp_sendmail</a>    | 使用 MAPI 接口将邮件发送给指定收件人。                | 仅适用于 NT      |
| <a href="#">xp_startmail</a>   | 启动 Adaptive Server 邮件会话。              | 仅适用于 NT      |
| <a href="#">xp_stopmail</a>    | 停止 Adaptive Server 邮件会话。              | 仅适用于 NT      |

## 系统 ESP 的权限

权限是在 `sybserverprocs` 数据库中设置的。

具有 `sa_role` 的用户对系统 ESP 拥有缺省执行权限。这些系统管理员可以将执行权限授予其他用户。

## 与系统 ESP 关联的 DLL

通过在 `sybserverprocs` 数据库中运行 `sp_helpextendedproc`，可以获得与系统 ESP 关联的 DLL 的名称。

## 使用系统 ESP

系统 ESP 遵循与常规系统过程相同的调用约定。系统 ESP 只有一条额外要求：必须运行 Open Server 应用程序 XP Server。Adaptive Server 将在首次调用 ESP 时启动 XP Server。XP Server 将一直运行到 Adaptive Server 关闭为止。

## xp\_cmdshell

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 在运行 Adaptive Server 的主机系统上执行本机操作系统命令。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 语法 | <code>xp_cmdshell command[, no_output] [return_status   no_wait]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 参数 | <p><b>command</b><br/>是操作系统命令字符串；最长为 8192 个字节。</p> <p><b>no_output</b><br/>如果指定，则不显示命令的任何输出。</p> <p><b>return_status</b><br/>如果指定，则返回 <b>command</b> 参数中指定的操作系统命令的完成状态。如果不使用此参数，则返回值为 0 表示成功，为 1 则表示失败。</p> <p><b>no_wait</b><br/>如果指定，则 <b>xp_cmdshell</b> 操作会立即将控制权还给调用方，指定的命令会作为后台进程执行。您不会看到输出，返回的结果仅反映作为后台进程启动命令时是成功还是失败，不反映进程本身是成功还是失败。</p>                                                                                                                                                                                                                                                               |
| 示例 | <p><b>示例 1</b>（在 Windows 上）将 C 驱动器上名为 <b>log</b> 的文件不加提示地复制到 A 驱动器上名为 <i>log.0102</i> 的文件：</p> <pre>xp_cmdshell 'copy C:\log A:\log.0102', no_output</pre> <p><b>示例 2</b>（在 UNIX 上）执行操作系统的 <b>ls</b> 命令，并将列表目录内容作为一行数据返回：</p> <pre>xp_cmdshell 'ls'</pre>                                                                                                                                                                                                                                                                                                                                                      |
| 用法 | <ul style="list-style-type: none"> <li>• <b>xp_cmdshell</b> 将包括操作系统错误在内的所有输出作为文本行返回到一列中。</li> <li>• <b>xp_cmdshell</b> 从 XP Server 的当前目录运行。</li> <li>• 返回的输出的列宽为 80 个字符。输出结果未经格式化。</li> <li>• <b>xp_cmdshell</b> 不能执行需要与用户交互的命令，如 “login”。</li> <li>• 通过 <b>xp_cmdshell</b> 执行的操作系统命令的用户环境受控于 <b>xp_cmdshell context</b> 配置参数的值。如果此参数设置为 1（缺省值），<b>xp_cmdshell</b> 会将权限限制为在操作系统级具有系统管理特权的用户。如果此参数设置为 0，<b>xp_cmdshell</b> 将使用 Adaptive Server 运行所用的操作系统帐户的安全性环境。因此，使用 <b>xp_cmdshell</b> 时如果将 <b>xp_cmdshell context</b> 配置参数设置为 0，任何用户都可以使用运行 Adaptive Server 的帐户的权限来执行操作系统命令。此帐户的限制可能少于用户本人帐户的限制。</li> </ul> |

- 无论 xp\_cmdshell context 的值如何，如果执行 xp\_cmdshell 的用户不是系统管理员（不具有 sa\_role），则系统管理员必须已经授予该用户明确的权限才能执行 xp\_cmdshell。例如，以下语句授予“joe”执行 xp\_cmdshell 的权限：

```
grant execute on xp_cmdshell to joe
```

- 若要查看 xp\_cmdshell 是否已经成功生成外部命令 XP Server，请输入以下命令，其中 *command* 是用 xp\_cmdshell 运行的命令的名称：

```
exec @ret = xp_cmdshell command
```

如果 xp\_cmdshell 执行成功，`exec @ret = xp_cmdshell command` 返回的值为 0。如果 xp\_cmdshell 失败，`exec @ret = xp_cmdshell command` 返回的值为 1。

- 若要查看使用 xp\_cmdshell 运行的命令本身是否成功，请输入以下命令，其中 *command* 是用 xp\_cmdshell 运行的命令的名称：

```
exec @ret = xp_cmdshell command, return_status
```

`exec @ret = xp_cmdshell command, return_status` 会使 xp\_cmdshell 返回该命令的实际退出状态码。如果发生故障导致 XP Server 无法运行该命令，xp\_cmdshell 将返回值 1。如果该命令运行成功，xp\_cmdshell 将返回值 0。

如果命令执行成功，`exec @ret = xp_cmdshell command` 返回的值为 0。如果命令失败，`exec @ret = xp_cmdshell command` 返回的值为 1。

---

**注释** `exec @ret = xp_cmdshell command` 和 `exec @ret = xp_cmdshell command, return_status` 都是向后兼容的。不使用 `return_status` 参数的旧存储过程将 `exec @ret = xp_cmdshell command, return_status` 视为 `exec @ret = xp_cmdshell command`。

---

此外，`no_output` 参数仍可与 `return_status` 搭配使用，顺序不限。

- 使用 xp\_cmdshell 远程过程创建代理表时，必须使用 `cmdstr` 列名：

```
create existing table xpoutput
(
 cmdstr varchar(255) null
)
external procedure at "THIS...xp_cmdshell"

select cmdstr from xpoutput where cmdstr = "date"
```

如果不使用 `cmdstr`，则会看到一条错误消息。

有关从代理表返回的结果的详细信息，请参见《组件集成服务用户指南》中的“将远程过程作为代理表”一节。

权限

缺省情况下，只有系统管理员才能执行 `xp_cmdshell`。系统管理员可以将执行权限授予其他用户。

另请参见

有关 `xp_cmdshell context` 的详细信息，请参见《系统管理指南》。

## xp\_deletemail

|    |                                                                                                                                                                                                                                                                  |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | (仅限 Windows) 从 Adaptive Server 收件箱中删除邮件。                                                                                                                                                                                                                         |
| 语法 | <code>xp_deletemail [msg_id]</code>                                                                                                                                                                                                                              |
| 参数 | <code>msg_id</code><br>是要删除的邮件的邮件标识符。                                                                                                                                                                                                                            |
| 示例 | <p><b>示例 1</b> 从 Adaptive Server 收件箱中删除带有 <code>cur_msg_id</code> 变量中指定的邮件标识符的邮件：</p> <pre>1&gt; declare @cur_msg_id binary(255) 2&gt; exec xp_deletemail @msg_id = @cur_msg_id</pre> <p><b>示例 2</b> 从 Adaptive Server 收件箱中删除第一封邮件：</p> <pre>xp_deletemail</pre> |
| 用法 | <ul style="list-style-type: none"><li>• 使用 <code>xp_findnextmsg</code> 可以获取 <code>msg_id</code>。</li><li>• 如果未使用 <code>msg_id</code> 参数，要删除的邮件缺省为收件箱中的第一封邮件。</li></ul>                                                                                           |
| 权限 | 缺省情况下，只有系统管理员才能执行 <code>xp_deletemail</code> 。系统管理员可以将此权限授予其他用户。                                                                                                                                                                                                 |



## xp\_enumgroups

|    |                                                                                                                                                                                    |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | (仅限 Windows) 显示指定的 Windows NT 域的组。                                                                                                                                                 |
| 语法 | <code>xp_enumgroups [domain_name]</code>                                                                                                                                           |
| 参数 | <i>domain_name</i><br>是要列出用户组的 Windows NT 域。                                                                                                                                       |
| 示例 | <b>示例 1</b> 列出运行 XP Server 的 Windows NT 计算机上的所有用户组:<br><pre>xp_enumgroups</pre> <b>示例 2</b> 列出 PCS 域中的所有用户组:<br><pre>xp_enumgroups 'PCS'</pre>                                     |
| 用法 | <ul style="list-style-type: none"><li>• 如果未传递任何参数, <code>xp_enumgroups</code> 将显示所有本地用户组。</li><li>• <i>域</i>是指共享同一用户帐户数据库和安全性策略的指定计算机集合。</li><li>• 返回状态为 0 表示成功; 1 表示失败。</li></ul> |
| 权限 | 缺省情况下, 只有系统管理员才能执行 <code>xp_enumgroups</code> 。系统管理员可以将此权限授予其他用户。                                                                                                                  |

## xp\_findnextmsg

|    |                                                                                                                                                                                                                                                                                                                                                         |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | (仅限 Windows) 检索 Adaptive Server 收件箱中的下一个邮件标识符。                                                                                                                                                                                                                                                                                                          |
| 语法 | <code>xp_findnextmsg @msg_id = @msg_id output[, type]<br/>[, unread_only = {true   false}]</code>                                                                                                                                                                                                                                                       |
| 参数 | <p><b>msg_id</b><br/>在输入时, 指定紧邻要检索的邮件标识符的上一个邮件标识符。将检索到的邮件标识符放入 <i>msg_id</i> 输出参数, 该参数必须是二进制类型。</p> <p><b>type</b><br/>是基于 MAPI 邮件定义的输入邮件类型。唯一受支持的邮件类型是 CMC:IPM。如果值为 NULL 或没有值, 则缺省为 CMC:IPM。</p> <p><b>unread_only</b><br/>如果此参数设置为 true, xp_findnextmsg 将只考虑未读邮件。如果此参数设置为 false, xp_findnextmsg 在检索下一个邮件标识符时将考虑所有邮件, 包括已读和未读邮件。缺省值为 true。</p>         |
| 示例 | <p><b>示例 1</b> 在 <i>@out_msg_id</i> 输出变量中, 返回在 <i>@out_msg_id</i> 指定的邮件之后的下一封未读邮件的邮件标识符:</p> <pre>xp_findnextmsg @msg_id = @out_msg_id output</pre> <p><b>示例 2</b> 在 <i>@out_msg_id</i> 输出变量中, 返回在 <i>@out_msg_id</i> 指定的邮件之后的下一封邮件的邮件标识符。可以是已读邮件, 也可以是未读邮件:</p> <pre>xp_findnextmsg @msg_id = @out_msg_id output, NULL,<br/>@unread_only = false</pre> |
| 用法 | <ul style="list-style-type: none"><li>如果 xp_findnextmsg 在收件箱中找不到更多的邮件, 它会返回状态 1。</li><li><a href="#">xp_deletemail</a> 和 <a href="#">xp_readmail</a> 使用 xp_findnextmsg 返回的邮件标识符。</li></ul>                                                                                                                                                              |
| 权限 | 缺省情况下, 只有系统管理员才能执行 xp_findnextmsg。系统管理员可以将此权限授予其他用户。                                                                                                                                                                                                                                                                                                    |

## xp\_logevent

| 说明    | (仅限 Windows) 用于在 Adaptive Server 中向 Windows NT 事件日志记录用户定义的事件。                                                                                                                                                                                                                                                                                                                                                             |      |   |    |     |     |                      |       |    |   |                     |    |    |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|---|----|-----|-----|----------------------|-------|----|---|---------------------|----|----|
| 语法    | <code>xp_logevent error_number, message[, type]</code>                                                                                                                                                                                                                                                                                                                                                                    |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 参数    | <p><b>error_number</b><br/>是用户分配的错误号。必须等于或大于 50000。</p> <p><b>message</b><br/>是在事件查看器的说明字段中显示的消息的文本。消息最长为 255 个字节。消息用引号引起来。</p> <p><b>type</b><br/>说明事件的紧急程度。值包括 <code>informational</code>、<code>warning</code> 和 <code>error</code>。缺省值为 <code>informational</code>。值用引号引起来。</p>                                                                                                                                        |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 示例    | <p><b>示例 1</b> 在 Windows NT 事件日志中记录信息性事件，编号为 55555。事件详细信息窗口中的说明文本为 “Email message deleted”：</p> <pre>xp_logevent 55555, 'Email message deleted.'</pre> <p><b>示例 2</b> 在 Windows NT 事件日志中记录错误事件，编号为 66666。事件详细信息窗口中的说明文本为 “DLL not found”：</p> <pre>xp_logevent 66666, 'DLL not found.', 'error'</pre>                                                                                                                     |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 用法    | <ul style="list-style-type: none"> <li>下表说明使用 <code>xp_logevent</code> 生成的事件的缺省事件详细信息：</li> </ul> <table border="1"> <thead> <tr> <th>详细信息</th> <th>值</th> </tr> </thead> <tbody> <tr> <td>用户</td> <td>不适用</td> </tr> <tr> <td>计算机</td> <td>运行 XP Server 的计算机的名称</td> </tr> <tr> <td>事件 ID</td> <td>12</td> </tr> <tr> <td>源</td> <td>Adaptive Server 的名称</td> </tr> <tr> <td>类别</td> <td>用户</td> </tr> </tbody> </table> | 详细信息 | 值 | 用户 | 不适用 | 计算机 | 运行 XP Server 的计算机的名称 | 事件 ID | 12 | 源 | Adaptive Server 的名称 | 类别 | 用户 |
| 详细信息  | 值                                                                                                                                                                                                                                                                                                                                                                                                                         |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 用户    | 不适用                                                                                                                                                                                                                                                                                                                                                                                                                       |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 计算机   | 运行 XP Server 的计算机的名称                                                                                                                                                                                                                                                                                                                                                                                                      |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 事件 ID | 12                                                                                                                                                                                                                                                                                                                                                                                                                        |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 源     | Adaptive Server 的名称                                                                                                                                                                                                                                                                                                                                                                                                       |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 类别    | 用户                                                                                                                                                                                                                                                                                                                                                                                                                        |      |   |    |     |     |                      |       |    |   |                     |    |    |
| 权限    | 只有系统管理员才能执行 <code>xp_logevent</code> 。                                                                                                                                                                                                                                                                                                                                                                                    |      |   |    |     |     |                      |       |    |   |                     |    |    |

## xp\_readmail

说明 (仅限 Windows) 查阅 Adaptive Server 收件箱中的邮件。

语法

```
xp_readmail [msg_id]
 [, recipients output]
 [, sender output]
 [, date_received output]
 [, subject output]
 [, cc output]
 [, message output]
 [, attachments output]
 [, suppress_attach = {true | false}]
 [, peek = {true | false}]
 [, unread = {true | false}]
 [, msg_length output]
 [, bytes_to_skip [output]]
 [, type [output]]
```

### 参数

#### *msg\_id*

指定 xp\_readmail 要查阅的邮件的邮件标识符。如果未使用 *msg\_id* 参数, 邮件缺省为邮箱中的第一封未读邮件 (如果 *unread* 为 true), 或是邮箱中的第一封邮件 (如果 *unread* 为 false)。

#### *recipients*

是用分号分隔的收件人列表。

#### *sender*

是发件人。

#### *date\_received*

是收到邮件的日期。

#### *subject*

是邮件主题。

#### *cc*

是邮件的抄送 (cc'd) 收件人列表 (用分号分隔)。

#### *message*

是邮件的正文文本。如果邮件正文的长度 (来自 *msg\_length* 输出参数) 大于 255, 则应使用 *byte\_to\_skip* 和 *msg\_length* 参数以 255 个字节为单位查阅该邮件。

#### *attachments*

是附件临时路径的列表 (用分号分隔)。如果 *suppress\_attach* 为 true, *attachments* 将被忽略。

#### *suppress\_attach*

如果设置为 true, 则不会为附件创建临时文件。缺省值为 true。

**peek**

如果设置为 **false**，在查阅邮件后将其标记为已读。如果设置为 **true**，即便已查阅邮件，仍将其标记为未读。缺省值为 **false**。

**unread\_only**

如果设置为 **true**，**xp\_readmail** 将只考虑未读邮件。如果设置为 **false**，**xp\_readmail** 将考虑所有邮件，不论标记为已读还是未读。缺省值为 **true**。

**msg\_length**

是以字节计的邮件总长度。与 **bytes\_to\_skip** 参数一同使用，允许 **xp\_readmail** 以 255 个字节为单位查阅邮件。

**bytes\_to\_skip**

在输入时，如果不为 0，则指定在将邮件的下一批 255 个字节读入 **message** 输出参数前应跳过的字节数。在输出时，包含邮件中的偏移量（**bytes\_to\_skip** 的先前值加上调用后输出的 **msg\_length**），从这个位置开始查阅下一批 255 个字节。

**type**

是基于 MAPI 邮件定义的邮件类型。唯一受支持的邮件类型是 **CMC:IPM**。如果值为 **NULL** 或没有值，则缺省为 **CMC:IPM**。

**示例**

**示例 1** 使用 **xp\_readmail** 查阅收件箱中的第一封未读邮件。它从 **@msgid** 变量中获取此邮件的邮件标识符（邮件标识符已由 **xp\_findnextmsg** ESP 存储在该变量中）。**xp\_readmail** 将发件人姓名存储在 **@originator** 变量中，将邮件正文存储在 **@mess** 变量中：

```
declare @msgid binary(255)
declare @originator varchar(20)
declare @mess varchar(255)
exec xp_findnextmsg @msg_id = @msgid output
exec xp_readmail @msg_id = @msgid,
@sender = @originator output,
@message = @mess output
```

**示例 2** 读取 **xp\_findnextmsg** 输出的邮件标识符所对应的邮件的前 255 个字节。如果邮件总长度超过 255 个字节，则读取下一批 255 个字节，依此类推，直到读取所有字节：

```
declare @msgid binary(255)
declare @mess varchar(255)
declare @msg_length char(255)
declare @len int
declare @skip int
exec xp_findnextmsg @msgid output
exec xp_readmail @msg_id = @msgid,
@message = @mess output,
```

```
@msg_length = @len output,
@bytes_to_skip = @skip output
print @mess
if (@len > 255)
begin
 while (@skip < @len)
 begin
 xp_readmail @msg_id = @msgid,
 @message = @mess output,
 @bytes_to_skip = @skip output
 print @mess
 end
end
end
```

**用法**

- **xp\_readmail** 可以查阅 Adaptive Server 收件箱中的邮件。
- 若要获取收件箱中下一封邮件的邮件标识符，请使用 [xp\\_findnextmsg](#)。

**权限**

缺省情况下，只有系统管理员才能执行 **xp\_readmail**。系统管理员可以将此权限授予其他用户。

## xp\_sendmail

**说明** (仅限 Windows) 向指定的收件人发送邮件。邮件可以是文本，也可以是 Transact-SQL 查询的结果。

**语法**

```
xp_sendmail recipient [, recipient] ...
 [, subject]
 [, cc_recipient] ...
 [, bcc_recipient] ...
 [, {query | message}]
 [, attachname]
 [, attach_result = {true | false}]
 [, echo_error = {true | false}]
 [, include_file [, include_file] ...]
 [, no_column_header = {true | false}]
 [, no_output = {true | false}]
 [, width]
 [, separator]
 [, dbuser]
 [, dbname]
 [, type]
 [, include_query = {true | false}]
```

**参数**

*recipient*  
是要接收邮件的用户的电子邮件地址。至少需要一个收件人。用分号分隔多个收件人。

*subject*  
是可选的邮件主题。如果未使用，则缺省为“Sybase SQL Server Message”。

*cc\_recipient*  
是邮件的抄送 (cc'd) 收件人列表（用分号分隔）。

*bcc\_recipient*  
是邮件的密送 (bcc'd) 收件人列表（用分号分隔）。

*query*  
是一个或多个 Transact-SQL 语句。结果将发送给收件人。如果使用 *query*，则无法使用 *message*。

*message*  
是所发送的邮件的文本。如果使用 *message*，则无法使用 *query*。有关使用 *message* 时忽略的选项的完整列表，请参见“用法”部分。

**attachname**

是使用 `query` 参数时包含查询结果的文件的名称，该文件作为附件包含在邮件中。如果使用 `attachname`，`attach_result` 必须设置为 `true`。如果 `attach_result` 为 `true` 但未指定 `attachname`，附加文件的结果文件名的前缀为 “`syb`”，随后依次是 5 位随机数字和扩展名 “.txt”，例如 `syb84840.txt`。如果使用 `message` 参数，此参数将被忽略。

**attach\_result**

如果设置为 `true`，则将查询结果作为邮件附件的形式发送。如果设置为 `false`，则将结果直接发送到邮件正文中。缺省值为 `false`。如果使用 `message` 参数，此参数将被忽略。

**echo\_error**

如果设置为 `true`，则将 Adaptive Server 邮件（包括有关受影响的行计数的邮件）与查询结果一同发送。如果设置为 `false`，则不发送 Adaptive Server 邮件。缺省值为 `true`。如果使用 `message` 参数，此参数将被忽略。

**include\_file**

是要作为附件包含到邮件中的文件列表，各文件之间用分号分隔。这些文件可以通过文件名、路径名或相对路径名来指定，并且可以是文本或二进制文件。

**no\_column\_header**

如果设置为 `true`，列标题将随查询结果一同发送。如果设置为 `false`，则不发送列标题。缺省值为 `false`。如果使用 `message` 参数，此参数将被忽略。

**no\_output**

如果设置为 `true`，则不向发送邮件的会话发送输出结果。如果设置为 `false`，发送邮件的会话将收到输出结果。缺省值为 `false`。如果使用 `message` 参数，此参数将被忽略。

**width**

指定在邮件中发送查询结果时结果集的宽度（以字符计）。`width` 与 `isql` 中的 `/w` 选项等效。在达到指定的 `width` 时，结果行将在新行字符处换行。缺省值为 80 个字符。如果使用 `message` 参数，此参数将被忽略。

**separator**

指定在邮件中发送查询结果时用作列分隔符的字符。`separator` 与 `isql` 中的 `/s` 选项等效。缺省值为制表符。如果使用 `message` 参数，此参数将被忽略。



**dbuser**

指定使用 *query* 参数时，用作执行查询的用户环境的数据库用户名。缺省值为 “guest”。如果使用 *message* 参数，此参数将被忽略。

**dname**

指定使用 *query* 参数时，用作执行查询的数据库环境的数据库名。缺省值为 “master”。如果使用 *message* 参数，此参数将忽略。

**type**

是基于 MAPI 邮件定义的输入邮件类型。唯一受支持的邮件类型是 CMC:IPM。如果值为 NULL 或没有值，则缺省为 CMC:IPM。

**include\_query**

如果设置为 true，则将 *query* 参数中使用的一个或多个查询附加到结果集中。如果设置为 false，则不附加查询。缺省值为 false。如果使用 *message* 参数，*include\_query* 将被忽略。

**示例**

**示例 1** 使用 *xp\_sendmail* 将有关 Adaptive Server 备份状态的文本消息发送给 “sally” 和 “ramon”，同时抄送给 “admin” 组：

```
xp_sendmail @recipient = "sally;ramon",
@subject = "Adaptive Server Backup Status",
@message = "Adaptive Server Backup for SERVER2 is
complete.",
@copy_recipient="admin"
```

**示例 2** 向 “peter” 发送 *authors* 表的查询结果。结果作为附件发送到邮件中，附件包含一个名为 *au\_lis.res* 的文件，该文件所在目录就是服务器的执行目录：

```
xp_sendmail "peter",
@query = "select * from authors",
@attachname = "au_list.res",
@attach_result= true
```

**用法**

- 以下参数与使用 *query* 参数时在邮件中发送的查询结果相关。如果使用 *message* 参数代替，它们将被忽略：*attachname*、*attach\_result*、*echo\_error*、*no\_column\_header*、*no\_output*、*width*、*separator*、*dbuser*、*dname*、*include\_query*。

**权限**

缺省情况下，只有系统管理员才能执行 *xp\_sendmail*。系统管理员可以将此权限授予其他用户。

## xp\_startmail

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | (仅限 Windows) 启动 Adaptive Server 邮件会话。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 语法 | <code>xp_startmail [mail_user] [, mail_password]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 参数 | <p><i>mail_user</i></p> <p>是 Adaptive Server 用于登录 Windows NT 邮件系统的邮件配置文件名。如果未使用 <i>mail_user</i>, <code>xp_startmail</code> 将使用建立 Sybmail Adaptive Server 帐户所用的邮件用户名。</p> <p><i>mail_password</i></p> <p>是 Adaptive Server 登录 Windows NT 邮件系统所用的邮件口令。如果未使用 <i>mail_password</i>, <code>xp_startmail</code> 将使用建立 Sybmail Adaptive Server 帐户所用的邮件口令。</p>                                                                                                                                                                                                                                                                                                                                       |
| 示例 | <p><b>示例 1</b> 使用 Sybmail 用户帐户的邮件用户名和口令启动 Adaptive Server 邮件会话:</p> <pre>xp_startmail</pre> <p><b>示例 2</b> 用 “mailuser” 作为配置文件名并用该配置文件名关联的口令来启动 Adaptive Server 邮件会话:</p> <pre>xp_startmail "mailuser", "tre55uu"</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 用法 | <ul style="list-style-type: none"> <li>• 如果已在运行 Adaptive Server 邮件会话, <code>xp_startmail</code> 将不会启动邮件会话。</li> <li>• 必须先启动 Adaptive Server 邮件会话 (通过显式调用 <code>xp_startmail</code> 或将 Adaptive Server 配置为在系统启动时自动启动 Adaptive Server 邮件会话), 然后才能执行与 Sybmail 相关的任何系统 ESP 或 <code>sp_processmail</code> 存储过程。有关在系统启动时自动启动 Adaptive Server 邮件会话的信息, 请参见《系统管理指南》中的 <code>start mail session</code>。</li> <li>• 如果未启动 Windows NT automail 会话, 则必须将 <i>mail_user</i> 和 <i>mail_password</i> 参数与 <code>xp_startmail</code> 一同使用。</li> <li>• 若要从 “sybmail” 用户帐户的 <i>fullname</i> 字段查看缺省的 <i>mail_user</i> 值, 请按如下方式使用 <code>sp_displaylogin</code> 系统过程:</li> </ul> <pre>sp_displaylogin sybmail</pre> |
| 权限 | 缺省情况下, 只有系统管理员才能执行 <code>xp_startmail</code> 。系统管理员可以将此权限授予其他用户。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

## xp\_stopmail

|    |                                                                                                                                        |
|----|----------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | (仅限 Windows) 停止 Adaptive Server 邮件会话。                                                                                                  |
| 语法 | xp_stopmail                                                                                                                            |
| 参数 | 无                                                                                                                                      |
| 示例 | 停止 Adaptive Server 邮件会话：<br><pre>xp_stopmail</pre>                                                                                     |
| 用法 | <ul style="list-style-type: none"><li>用 xp_stopmail 终止 Adaptive Server 邮件会话后，将无法执行与 Sybmail 相关的系统 ESP 和 sp_processmail 存储过程。</li></ul> |
| 权限 | 缺省情况下，只有系统管理员才能执行 xp_stopmail。系统管理员可以将此权限授予其他用户。                                                                                       |



本章介绍 dbcc 存储过程。

涉及的主题包括：

| 主题                       | 页码  |
|--------------------------|-----|
| <a href="#">概述</a>       | 723 |
| <a href="#">指定对象名和日期</a> | 724 |

## 概述

这些过程只能访问 dbccdb 数据库的表或替代数据库 dbccalt 中的表。有关设置 dbccdb 或 dbccalt 的详细信息，请参见《系统管理指南》。有关这些数据库中使用的表的信息，请参见《参考手册：表》中的第 2 章“dbccdb 表”。

表 4-1 列出了本章描述的各个 dbcc 存储过程。有关 dbcc 系统过程 sp\_plan\_dbccdb 的详细信息，请参见 sp\_plan\_dbccdb。有关此系统过程和 dbcc 存储过程的详细信息，请参见《系统管理指南》。

**表 4-1: dbcc 存储过程**

| 过程名称                                       | 说明                                                      |
|--------------------------------------------|---------------------------------------------------------|
| <a href="#">sp_dbcc_alterws</a>            | 将指定工作空间的大小改为指定值，并初始化该工作空间。                              |
| <a href="#">sp_dbcc_configreport</a>       | 生成一个报告，该报告描述针对指定数据库的 dbcc checkstorage 操作使用的配置信息。       |
| <a href="#">sp_dbcc_createws</a>           | 在指定段与数据库上创建指定类型与大小的工作空间。                                |
| <a href="#">sp_dbcc_deletedb</a>           | 从 dbccdb 中删除与指定目标数据库相关的所有信息。                            |
| <a href="#">sp_dbcc_deletehistory</a>      | 删除在指定日期和时间之前对目标数据库执行的 dbcc checkstorage 操作的结果？          |
| <a href="#">sp_dbcc_differentialreport</a> | 生成报告，突出说明两次 dbcc 操作之间 I/O 统计信息的变化及出现的故障。                |
| <a href="#">sp_dbcc_evaluatedb</a>         | 重新计算目标数据库的配置信息，并将其与当前配置信息进行比较。                          |
| <a href="#">sp_dbcc_exclusions</a>         | 允许用户创建和管理供 checkverify 和 sp_dbcc_faultreport 使用的持久排除列表。 |

| 过程名称                                     | 说明                                                                                                                                                                 |
|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">sp_dbcc_faultreport</a>      | 生成一个报告，该报告中包含在指定日期针对目标数据库中的指定对象执行的 <code>dbcc checkstorage</code> 操作的故障统计信息，这些统计信息按表和索引的顺序列出。                                                                      |
| <a href="#">sp_dbcc_fullreport</a>       | 运行 <a href="#">sp_dbcc_summaryreport</a> 、 <a href="#">sp_dbcc_configreport</a> 、 <a href="#">sp_dbcc_statisticsreport</a> 和 <a href="#">sp_dbcc_faultreport</a> 。 |
| <a href="#">sp_dbcc_help_fault</a>       | 提供指定故障类型的说明和建议的修复。                                                                                                                                                 |
| <a href="#">sp_dbcc_patch_finishtime</a> | 便于报告中中止的 <code>checkverify</code> 和 <code>checkstorage</code> 操作的情况。                                                                                               |
| <a href="#">sp_dbcc_recommendations</a>  | 根据 <code>checkstorage</code> 的运行情况为数据库列出建议，并按表和索引对建议分组。                                                                                                            |
| <a href="#">sp_dbcc_runcheck</a>         | 在指定数据库上运行 <code>dbcc checkstorage</code> ，然后运行 <a href="#">sp_dbcc_summaryreport</a> 或指定的报告。                                                                       |
| <a href="#">sp_dbcc_statisticsreport</a> | 针对目标数据库中的指定对象生成分配统计信息报告。                                                                                                                                           |
| <a href="#">sp_dbcc_summaryreport</a>    | 生成指定数据库的摘要报告。                                                                                                                                                      |
| <a href="#">sp_dbcc_updateconfig</a>     | 使用目标数据库的配置信息更新 <code>dbccdb</code> 中的 <code>dbcc_config</code> 表。                                                                                                  |

## 指定对象名和日期

有些 `dbcc` 存储过程使用参数来表示对象名和日期。本节提供有关指定对象名和日期的重要信息。

### 指定对象名

对象名仅指定要为其生成报告的表或索引的名称。指定对象名时，必须同时指定数据库名 (`dbname`)。不能为对象指定所有者。如果指定的对象名在目标数据库中不唯一，系统过程将为具有指定名称的所有对象生成报告。

### 指定日期

使用以下语法指定日期和时间（可选）：

```
mm/dd/yy[:hh:mm:ss]
```

采用 24 小时制。

指定日期后，系统过程会对其作出如下解释：

- 如果指定了日期和时间，则为报告选择在指定日期和时间完成的 dbcc 操作。
- 如果指定的日期为当前日期，但未指定时间，时间将自动设置为当前时间。将为报告选择在过去的 24 小时内，完成时间最接近当前时间的 dbcc 操作。
- 如果指定的日期不是当前日期，且未指定时间，时间将自动设置为“23:59:59”。这种情况下将为报告选择完成日期和时间最接近指定日期和系统提供时间的 dbcc checkstorage 操作。

例如，假定最近的 dbcc checkstorage 操作的完成时间为 1997 年 3 月 4 日 10:20:45。

如果您指定的日期为“03/04/97”，系统过程会将日期解释为 03/04/97:23:59:59。此日期及时间将与实际完成日期及时间（即 03/04/97:10:20:45）进行比较。

如果指定日期为“03/04/97:10:00:00”，则不会为报告选择在 10:20:45 完成的操作，因为只有指定时间或在此之前完成的操作才符合标准。

如果指定日期为“03/06/97”，则不会生成任何报告，因为最近一次操作的完成时间比该时间早 24 小时以上。

## sp\_dbcc\_alterws

|      |                                                                                                                                                                                                                                                                                                                                                                                      |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明   | 将指定工作空间的大小改为指定值，并初始化该工作空间。                                                                                                                                                                                                                                                                                                                                                           |
| 语法   | <code>sp_dbcc_alterws dbname, wsname, "wssize[K M]"</code>                                                                                                                                                                                                                                                                                                                           |
| 参数   | <p><b>dbname</b><br/>是工作空间驻留的数据库的名称。指定 <code>dbccdb</code> 或 <code>dbccalt</code>。</p> <p><b>wsname</b><br/>指定要变更的工作空间的名称。</p> <p><b>wssize</b><br/>是工作空间的新大小，以 K（千字节）或 M（兆字节）为单位指定。如果不指定 K 或 M，<code>wssize</code> 将指定页数。页大小取决于平台。工作空间最小为 24 页。</p>                                                                                                                                 |
| 示例   | 将 <code>dbccdb</code> 上的 <code>scan_ws_000001</code> 工作空间的大小改为 30MB：<br><pre>sp_dbcc_alterws dbccdb, scan_ws_000001, "30M"</pre> 工作空间 <code>scan_ws_000001</code> 的大小已成功地更改为 30MB                                                                                                                                                                                                    |
| 用法   | <ul style="list-style-type: none"><li>• <code>sp_dbcc_alterws</code> 将指定工作空间的大小改为指定值，并初始化该工作空间。</li><li>• 要实现最佳性能，应确保在变更工作空间之前已经配置了至少 16K 的缓冲池。</li><li>• 使用 <a href="#">sp_plan_dbccdb</a> 在变更工作空间之前确定估计大小。</li><li>• 要变更的工作空间必须已经存在。有关创建工作空间的信息，请参见 <a href="#">sp_dbcc_createws</a>。</li><li>• 若要删除工作空间，请在 <code>dbccdb</code> 中发出：<pre>drop table workspace_name</pre></li></ul> |
| 权限   | 只有系统管理员或数据库所有者才能运行 <code>sp_dbcc_alterws</code> 。                                                                                                                                                                                                                                                                                                                                    |
| 另请参见 | 有关 <code>scan</code> 和 <code>text</code> 工作空间以及 <code>dbccalt</code> 数据库的详细信息，请参见《系统管理指南》。                                                                                                                                                                                                                                                                                           |
|      | <b>命令</b> <a href="#">dbcc</a>                                                                                                                                                                                                                                                                                                                                                       |
|      | <b>dbcc 存储过程</b> <a href="#">sp_dbcc_createws</a> 、 <a href="#">sp_dbcc_evaluatedb</a>                                                                                                                                                                                                                                                                                               |
|      | <b>系统过程</b> <a href="#">sp_plan_dbccdb</a> 、 <a href="#">sp_helpdb</a>                                                                                                                                                                                                                                                                                                               |



## sp\_dbcc\_configreport

**说明** 生成一个报告，该报告描述针对指定数据库的 dbcc checkstorage 操作使用的配置信息。

**语法** sp\_dbcc\_configreport [dbname]

**参数** *dbname*  
指定数据库的名称。如果未指定 *dbname*，报告将包含 dbccdb..dbcc\_operation\_log 中所有数据库的相关信息。

**示例** 生成一个报告，以报告与针对 sybsystemprocs 数据库的 dbcc 相关的配置信息。在“Value”列中列出对象名，如果适用的话，还会列出大小：

```
sp_dbcc_configreport
Reporting configuration information of database sybsystemprocs.

Parameter Name Value Size

database name sybsystemprocs 51200K
dbcc named cache default data cache 1024K
text workspace textws_001 (id = 544004969) 128K
scan workspace scanws_001 (id = 512004855) 1024K
max worker processes 1
operation sequence number 2
```

**用法**

- sp\_dbcc\_configreport 生成报告，描述指定数据库的 dbcc 操作使用的配置信息。这些信息存储在 dbcc\_config 表中。
- 若要确定最新配置参数的值，请在运行 sp\_dbcc\_configreport 之前先运行 sp\_dbcc\_updateconfig。
- 若要更改某个工作空间的配置值，请使用 sp\_dbcc\_alterws。

**权限** 对指定的数据库名有效的任何用户都可以运行 sp\_dbcc\_configreport。  
指定的数据库名。

**另请参见** **命令** dbcc

**dbcc 存储过程** sp\_dbcc\_alterws、sp\_dbcc\_fullreport、  
sp\_dbcc\_statisticsreport、sp\_dbcc\_summaryreport、sp\_dbcc\_updateconfig

## sp\_dbcc\_createws

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 在指定段与数据库上创建指定类型与大小的工作空间。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 语法 | <code>sp_dbcc_createws dbname, segname, [wsname], wstype, "wssize[K M]"</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 参数 | <p><b>dbname</b><br/>是数据库名称，将要在此数据库中创建工作空间。值为 <code>dbccdb</code> 和 <code>dbccalt</code>。</p> <p><b>segname</b><br/>是工作空间的段名。</p> <p><b>wsname</b><br/>是工作空间的名称。如果该值为 <code>null</code>，<code>sp_dbcc_createws</code> 会为 <code>scan</code> 工作空间生成名称 <code>scan_wsnnnnnn</code>，为 <code>text</code> 工作空间生成名称 <code>text_wsnnnnnn</code>，其中 <code>nnnnnn</code> 是一个 6 位的唯一数。</p> <p><b>wstype</b><br/>指定要创建的工作空间的类型。值为 <code>scan</code> 和 <code>text</code>。</p> <p><b>wssize</b><br/>是工作空间的大小，以 <b>K</b>（千字节）或 <b>M</b>（兆字节）指定。如果不指定 <b>K</b> 或 <b>M</b>，<code>wssize</code> 将指定页数。工作空间最小为 24 页。</p> |
| 示例 | <p><b>示例 1</b> 在 <code>dbccdb</code> 中的 <code>scanseg</code> 段上，创建一个名为 <code>scan_wspubs2</code>、大小为 10MB 的 <code>scan</code> 工作空间：</p> <pre>sp_dbcc_createws dbccdb, scanseg, scan_wspubs2, scan, "10M"</pre> <p><b>示例 2</b> 在 <code>dbccdb</code> 中的 <code>textseg</code> 段上，创建一个名为 <code>text_ws000001</code>、大小为 14MB 的 <code>scan</code> 工作空间：</p> <pre>sp_dbcc_createws dbccdb, textseg, text, "14M"</pre>                                                                                                                                                                                     |
| 用法 | <ul style="list-style-type: none"> <li>• <code>sp_dbcc_createws</code> 创建具有指定名称和大小的工作空间并初始化该工作空间。</li> <li>• 创建工作空间之前，应使用 <code>sp_addsegment</code> 创建段。</li> <li>• 创建工作空间之前，应确保已经配置了至少 16K 的缓冲池，以实现最佳性能。</li> <li>• 创建工作空间时，务必在所需的设备空间中添加 5% 的开销，因为创建工作空间时会使用大型页分配方案。</li> <li>• 使用 <code>sp_plan_dbccdb</code> 确定估计大小。</li> <li>• 创建工作空间后，运行 <code>sp_dbcc_updateconfig</code> 在 <code>dbcc_config</code> 中记录这些新的配置信息。</li> </ul>                                                                                                                                                                 |

- 每个工作空间都必须具有唯一名称。
- 若要删除工作空间，请在 dbccdb 中发出：

```
drop table workspace_name
```

权限

只有系统管理员或数据库所有者才能运行 `sp_dbcc_createws`。

另请参见

有关 `scan` 和 `text` 工作空间以及 `dbccalt` 数据库的详细信息，请参见《系统管理指南》。

**命令** `dbcc`

**dbcc 存储过程** `sp_dbcc_alterws`、`sp_dbcc_evaluatedb`

**系统过程** `sp_addsegment`、`sp_plan_dbccdb`、`sp_helpsegment`

## sp\_dbcc\_deletedb

**说明** 从 dbccdb 中删除与指定目标数据库相关的所有信息。

**语法** sp\_dbcc\_deletedb [*dbname* | *dbid*]

**参数** *dbname*

指定目标数据库的名称，要删除的就是该数据库的配置信息。如果不指定 *dbname* 的值，Adaptive Server 会删除 dbccdb..dbcc\_config 中的所有数据库的数据。如果目标数据库为 dbccdb，并且 dbccalt 存在，则 Adaptive Server 将从 dbccalt 中删除数据。

*dbid*

指定目标数据库的数据库 ID 号，要删除的就是该数据库的配置信息。

**示例** 从 dbccdb 中删除名为 engdb 的数据库的所有信息：

```
sp_dbcc_deletedb "engdb"
```

```
All information for database engdb has been deleted from dbccdb.
```

**用法**

- sp\_dbcc\_deletedb 会从 dbccdb 中删除与指定目标数据库相关的所有信息，包括配置信息和以前的 dbcc checkstorage 操作的结果。
- 如果删除的数据库是 dbccdb 并且 dbccalt 数据库存在，sp\_dbcc\_deletedb 将从 dbccalt 中删除配置信息和 dbccdb 的结果。
- 若要删除特定日期之前执行的 dbcc checkstorage 操作的结果，请使用 sp\_dbcc\_deletehistory。
- 对于已删除的数据库，使用 *dbid* 选项是唯一能够删除 dbccdb 数据库内容的方法。

**权限** 只有系统管理员或数据库所有者才能运行 sp\_dbcc\_deletedb。

**另请参见** 有关 dbccalt 数据库的信息，请参见《系统管理指南》。

**命令 dbcc**

**dbcc 存储过程** [sp\\_dbcc\\_deletehistory](#)、[sp\\_dbcc\\_evaluatedb](#)

**系统过程** [sp\\_plan\\_dbccdb](#)

## sp\_dbcc\_deletehistory

**说明** 删除在指定日期和时间之前对目标数据库执行的 dbcc checkstorage 操作的结果。

---

**注释** sp\_dbcc\_deletehistory 不释放与已删除的历史数据相关的任何空间，因为工作空间是预先分配的并具有固定大小。

---

**语法** sp\_dbcc\_deletehistory [*cutoffdate* [, *dbname* | *dbid*]]

**参数**

*cutoffdate*

删除在此日期或此日期之前创建的所有条目。该参数的类型为 **datetime**。如果未指定日期，则只保留上一操作的结果。有关详细信息，请参见第 724 页的“指定日期”。

*dbname*

指定要从中删除数据的数据库的名称。如果未指定，sp\_dbcc\_deletehistory 将删除 dbccdb..dbcc\_config 中的所有数据库的历史信息。

*dbid*

指定目标数据库的数据库 ID 号，要删除的就是此数据库的历史记录信息。

**示例** 删除在 1997 年 3 月 4 日当天或之前对数据库 pubs2 执行的所有操作的结果：

```
sp_dbcc_deletehistory "03/04/1997", "pubs2"
```

**用法**

- sp\_dbcc\_deletehistory 删除在指定日期和时间之前在目标数据库上执行的 dbcc checkstorage 操作的结果。
- 如果目标数据库是 dbccdb，并且 dbccalt 数据库存在，sp\_dbcc\_deletehistory 将从 dbccalt 中删除 dbccdb 的历史数据。
- 为 *cutoffdate* 指定的值要与每个 dbcc 操作的完成时间作比较。
- 使用 *dbid* 选项，可以为已经删除的数据库删除 dbccdb 数据库的历史数据。
- 对于已删除的数据库，使用 *dbid* 选项是唯一能够删除 dbccdb 数据库的历史数据的方法。
- 若要查看 dbcc checkstorage 的运行日期，以便选择 *cutoffdate* 的值，请运行 [sp\\_dbcc\\_summaryreport](#)。

权限

- 只有系统管理员或数据库所有者才能在特定数据库上运行 `sp_dbcc_deletehistory`。
- 只有系统管理员才能在不指定数据库名的情况下运行 `sp_dbcc_deletehistory`。

另请参见

有关 `dbccalt` 数据库的信息，请参见《系统管理指南》。

**命令** `dbcc`

**dbcc 存储过程** `sp_dbcc_deletedb`、`sp_dbcc_evaluatedb`

**系统过程** `sp_plan_dbccdb`

## sp\_dbcc\_differentialreport

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 生成报告，突出说明两次 dbcc 操作之间 I/O 统计信息的变化及出现的故障。                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 语法 | <pre>sp_dbcc_differentialreport [dbname [, objectname]],                            [db_op] [, "date1" [, "date2"]]</pre>                                                                                                                                                                                                                                                                                                                                                                         |
| 参数 | <p><b>dbname</b><br/>指定数据库的名称。如果不指定 <i>dbname</i>，报告将包含 dbccdb..dbcc_operation_log 中所有数据库的相关信息。</p> <p><b>objectname</b><br/>指定要为其生成报告的表或索引的名称。如果未指定 <i>object_name</i>，则报告有关目标数据库中的所有对象的统计信息。</p> <p><b>db_op</b><br/>指定要在报告中使用的数据源。值只能是 checkstorage。将针对 <i>db_op</i> 指定的数据为目标数据库中的指定对象生成 <i>date1</i> 和 <i>date2</i> 的报告。如果未指定日期，将比较最近两个 <i>db_op</i> 类型的操作。</p> <p><b>date1</b><br/>指定要比较的 dbcc checkstorage 操作的第一个日期。</p> <p><b>date2</b><br/>指定要比较的 dbcc checkstorage 操作的最后一个日期。</p>        |
| 示例 | <p>生成一个报告，用以说明在 1997 年 5 月 1 日和 1997 年 5 月 4 日之间，sysprocedures 表中 I/O 统计信息的变化及出现的故障：</p> <pre>sp_dbcc_differentialreport master, sysprocedures,                            checkstorage, "05/01/97", "05/04/97"</pre>                                                                                                                                                                                                                                                                             |
| 用法 | <ul style="list-style-type: none"> <li>• sp_dbcc_differentialreport 生成报告，突出说明两次 dbcc 操作之间 I/O 统计信息的变化及出现的故障。它会比较两个 dbcc checkstorage 实例报告的计数器值。只有发生变化的值才会报告出来。</li> <li>• 如果只指定一个日期，指定日期所选的 dbcc checkstorage 操作的结果将与所选操作之前的那次 dbcc checkstorage 操作的结果作比较。</li> <li>• 如果未指定任何日期，将比较最近两次 dbcc checkstorage 操作的结果。</li> <li>• 如果 sp_dbcc_differentialreport 返回 <i>object_name</i> 的值，则表示完成 dbcc checkstorage 操作后删除了该对象。</li> <li>• 如果指定的各次操作之间未发生任何变化，sp_dbcc_differentialreport 不会生成报告。</li> </ul> |
| 权限 | 对指定的数据库名有效的任何用户都可以运行 sp_dbcc_differentialreport。                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

另请参见

**命令** [dbcc](#)

**dbcc 存储过程** [sp\\_dbcc\\_fullreport](#)、[sp\\_dbcc\\_statisticsreport](#)、[sp\\_dbcc\\_summaryreport](#)、[sp\\_dbcc\\_updateconfig](#)



## sp\_dbcc\_evaluatedb

**说明** 重新计算目标数据库的配置信息，并将其与当前配置信息进行比较。

**语法** sp\_dbcc\_evaluatedb [dbname]

**参数** *dbname*  
指定目标数据库的名称。如果未指定 *dbname*，sp\_dbcc\_evaluatedb 将比较 dbcc\_config 表中列出的所有数据库。

**示例** 重新计算当前数据库 sybsystemprocs 的配置信息，并为某些参数建议新值：

```
1> sp_dbcc_evaluatedb
2> go
```

Recommended values for workspace size, cache size and process count are:

```
Database name :one_G
 current suggested
scan workspace size : 750M 16M
text workspace size : 2K 48K
cache size : 10240K 1280K
process count : 3 2
compression mem size: 2048K 12M
```

```
Each of the reported quantities is reported in a scaled unit according to
G if size > 10G
M if 10M < size <=10 G
K otherwise
```

- 用法**
- 当存在具有压缩数据或日志设备的存档数据库时，输出将包含一行就压缩内存大小提出建议的内容。
  - sp\_dbcc\_evaluatedb 重新计算目标数据库的配置信息，并与当前配置信息进行比较。它使用 dbcc\_counters 表中记录的目标数据库的计数器值。
  - 高速缓存大小是高速缓存中的 16K 缓冲池的大小。对于 2K 缓冲池，此高速缓存的最小容量必须是推荐值加上 512。
  - 当目标数据库的大小和数据分配模式变化时，可运行 sp\_dbcc\_evaluatedb 优化配置信息。
  - 若是首次收集目标数据库的配置信息，请使用 sp\_plan\_dbccdb。
  - 若要确保是对最近的配置参数求值，请在运行 sp\_dbcc\_evaluatedb 之前先运行 sp\_dbcc\_updateconfig。

权限

- 只有系统管理员或数据库所有者才能运行 `sp_dbcc_evaluatedb`。
- 只有系统管理员才能在不指定数据库名的情况下运行 `sp_dbcc_evaluatedb`。

另请参见

**命令** [dbcc](#)

**dbcc 存储过程** [sp\\_dbcc\\_updateconfig](#)

**系统过程** [sp\\_plan\\_dbccdb](#)

## sp\_dbcc\_exclusions

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 允许用户创建和管理供 <code>checkverify</code> 和 <code>sp_dbcc_faultreport</code> 使用的持久排除列表。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 语法 | <code>sp_dbcc_exclusions dbname, op, type, exclusion_list</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 参数 | <p><b>dbname</b><br/>是排除项适用的数据库的名称，如果适用于所有数据库，则为 <code>null</code>。</p> <p><b>op</b><br/>是要执行的操作。有效值包括：</p> <ul style="list-style-type: none"> <li>• <code>add</code> — 注册新排除项（忽略重复项）。</li> <li>• <code>drop</code> — 如果以前已注册指定的排除项，则将其删除。</li> <li>• <code>listall</code> — 列出为所有数据库记录的排除。</li> </ul> <p><b>type</b><br/>是要排除的项的类型。可接受的值有 <code>faults</code>、<code>tables</code>、<code>combo</code> 或 <code>null</code>（当 <code>op</code> 为 <code>null</code> 或 <code>listall</code> 时）。<code>varchar</code> 类型。</p> <p><b>exclusion_list</b><br/>是由故障、表、表条目和故障条目或 <code>null</code> 组成的逗号分隔列表。<code>varchar</code> 类型。</p>                                                                                                                                                                                                          |
| 示例 | <p><b>示例 1</b> 从所有数据库上的 <code>sp_dbcc_faultreport</code> 处理中排除表 <code>syslogs</code> 和 <code>syscomments</code>：</p> <pre>sp_dbcc_exclusions null, 'add', 'tables', 'syslogs, syscomments'</pre> <p><b>示例 2</b> 从 <code>my_db</code> 数据库的处理中排除故障类型 <code>100036</code>：</p> <pre>sp_dbcc_exclusions my_db, 'add', 'faults', '100036'</pre> <p><b>示例 3</b> 将以下内容添加到与 <code>my_db</code> 对应的排除列表：与表 <code>mytable</code> 相关的故障类型 <code>100002</code>，以及与 <code>syslogs</code> 相关的故障类型 <code>100035</code>：</p> <pre>sp_dbcc_exclusions my_db, 'add', 'combo', 'mytable:100002, syslogs:100035'</pre> <p><b>示例 4</b> 从与 <code>my_db</code> 对应的排除列表中删除故障类型 <code>100036</code>：</p> <pre>sp_dbcc_exclusions my_db, 'drop', 'faults', '100036'</pre> <p><b>示例 5</b> 显示与 <code>my_db</code> 对应的排除列表：</p> <pre>sp_dbcc_exclusions my_db</pre> |

**示例 6** 显示所有数据库的已记录的排除：

```
sp_dbcc_exclusions null, 'listall'
```

”D” 如果指定了

- *listall*, *dbname* 必须为 *null*。如果 *op* 为 *null*, *sp\_dbcc\_exclusions* 会列出指定数据库的记录的排除。
- 只有系统管理员或数据库所有者可以运行 *dbname* 参数不是 *null* 的 *sp\_dbcc\_exclusions*。
- 如果 *dbname* 和 *op* 参数为 *null*, 则用户必须是系统管理员, 或者至少拥有一个已记录了排除的数据库。
- 如果 *dbname* 参数为 *null* 而 *op* 参数为 *listall*, 则用户必须是系统管理员, 或者至少拥有一个已记录了排除的数据库。如果用户不是系统管理员, 将只报告用户拥有的数据库的记录的排除。

权限

只有系统管理员才能不受限制地运行 *sp\_dbcc\_exclusions*。

## sp\_dbcc\_faultreport

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 生成报告，涉及在指定日期针对目标数据库中的指定对象执行的 dbcc checkstorage 操作的故障统计信息。报告按顺序列出表和索引。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 语法 | <pre>sp_dbcc_faultreport [report_type [, dbname [, objectname                     [, date [, hard_only [, exclusion_mode[, exclusion_faults                     [, exclusion_tables [, exclusion_combo                     [, display_recommendations [, opid [, fault_type_in]]]]]]]]]]]]]</pre>                                                                                                                                                                                                                                                                                                                            |
| 参数 | <p><b>report_type</b><br/>指定故障报告的类型。有效值为 short 和 long。缺省值为 short。</p> <p><b>dbname</b><br/>指定目标数据库的名称；例如 master..sysdatabases。如果未指定 <b>dbname</b>，报告将包含 dbccdb..dbcc_operation_log 中所有数据库的相关信息。</p> <p><b>object_name</b><br/>指定要为其生成报告的表或索引的名称。如果未指定 <b>object_name</b>，则报告有关目标数据库中的所有对象的统计信息。</p> <p><b>date</b><br/>指定完成 dbcc checkstorage 操作的确切日期和时间。可以在 dbcc_operation_log.finish 中找到这个值。也可以创建该值，方法是在 sp_dbcc_summaryreport 输出中将 start time 的日期与 end time 的小时数和分钟数进行组合。如果没有指定 <b>date</b>，则 Adaptive Server 使用最近操作的日期。</p> <p>指定 <b>date</b> 参数时，务必要让输入的时间晚于操作完成的日期。<br/>sp_dbcc_faultreport 不能报告在此参数中输入的时间之后发生的故障。</p> |

---

**注释** 若只想指定 **date** 参数，请对其它所有参数使用“null”。如果完全省略某个参数，sp\_dbcc\_faultreport 将无法生成正确的报告。

---

### *hard\_only*

指定 1 时启用硬故障的报告。有效值为 0 或 1，缺省值是 0。

### *display\_recommendations*

启用报告 sp\_dbcc\_recommendations 生成的建议，参数 **exclusion\_mode**、**exclusion\_faults**、**exclusion\_tables**、**display\_recommendations** 和 **exclusion\_combo** 引用排除支持，并且是可选的。

**exclusion\_mode**

为 `varchar` 类型，缺省情况下启用。要禁用此参数，在每次运行 `sp_dbcc_faultreport` 时都必须提供 “ignore”。使用下列命令之一：

- `ignore` — 忽略永久排除项列表，如果提供了临时排除项列表，则使用临时排除项列表（`varchar` 类型）。
- `extend` — 应用临时排除项列表和永久排除项列表（`varchar` 类型）。

**exclusion\_faults**

是要从报告中排除的故障类型的列表，用逗号分隔（`varchar` 类型）。

**exclusion\_tables**

是要从报告范围中排除的表的列表，用逗号分隔（`varchar` 类型）。

**exclusion\_combo**

是要从报告范围中排除的故障 / 表组合的列表，用逗号分隔（`varchar` 类型）。

**opid**

为特定日期的特定（而非最新）操作 ID 启用故障报告。缺省情况下未指定任何操作 ID。

**fault\_type\_in**

为特定故障类型启用故障报告。缺省值为 `NULL`。

**示例**

**示例 1** 生成一个短报告，用以报告关于在 `sybssystemprocs` 数据库的表中发现的故障。该报告包含表名、出现故障的索引号、故障的类型代码、故障的简短说明，以及出现故障的页号：

```
sp_dbcc_faultreport "short"
```

```
Database Name :sybssystemprocs
```

| Table Name    | Index | Type Code | Description        | Page Number |
|---------------|-------|-----------|--------------------|-------------|
| sysprocedures | 0     | 100031    | page not allocated | 5702        |
| sysprocedures | 1     | 100031    | page not allocated | 14151       |
| syslogs       | 0     | 100022    | chain start error  | 24315       |
| syslogs       | 0     | 100031    | page not allocated | 24315       |

**示例 2** 生成一个长报告，用以报告关于在 `sybsystemprocs` 数据库的表中发现的故障。下例显示了长报告输出的第一部分。完整报告将为 `dbcc checkstorage` 在其中发现故障的**目标数据库**中的每个对象都重复这些信息。在“page header”字段下显示的较长数字字符串之后的数据（“Header for 14151, next 14216, previous 14150 ...”）说明“page header”字符串的组成部分：

```
sp_dbcc_faultreport "long"

Generating 'Fault Report' for object sysprocedures in database
sybsystemprocs.

Type Code:100031; Soft fault, possibly spurious
Page reached by the chain is not allocated.
page id:14151
page
header:0x00003747000037880000374600000005000648B803EF0001000103FE0080000F
Header for 14151, next 14216, previous 14150, id = 5:1
 time stamp = 0x0001000648B8, next row = 1007, level = 0
 free offset = 1022, minlen = 15, status = 128(0x0080)
.
.
.
```

**示例 3** 从 `sp_dbcc_summaryreport` 的输出中，为在 End Time 指定的日期和时间完成的操作生成所有数据库中所有表发生的故障的短报告。在 `date` 参数中应使用准确的结束时间，这一点很重要。例如，如果输入：

```
7/25/2000 9:58
```

而不是

```
7/25/2000 9:58:0:190
```

报告会包含截至 9:58 而不是此后发生的故障。如果不想输入操作结束的确切时间，则可以使用 9:59:

```
sp_dbcc_faultreport "short", NULL, NULL,
"07/25/00 9:59"
```

在这种情况下，报告会包含截至 9:59 发生的故障。

**示例 4** 生成一个简短形式的报告，仅报告最近一次对名为 `mydb` 的数据库运行 `checkstorage` 时报告的硬故障：

```
sp_dbcc_faultreport short, mydb, @hard_only = 1
```

**示例 5** 向数据库 `my_db` 的故障报告添加建议的修复：

```
sp_dbcc_faultreport @dbname = my_db,
@display_recommendations = 1
```

**示例 6** 生成的故障报告不包含修复建议：

```
sp_dbcc_faultreport @dbname = my_db
```

**示例 7** 对数据库 my\_db 运行禁用了持久排除列表的 sp\_dbcc\_faultreport:

```
sp_dbcc_faultreport @dbname = 'my_db', @exclusion_mode = 'ignore'
```

**示例 8** 对数据库 my\_db 运行启用了持久排除列表的 sp\_dbcc\_faultreport, 且对其进行了扩展, 使其不处理故障类型 100036:

```
sp_dbcc_faultreport @dbname = 'my_db', @exclusion_mode = 'extend',
@exclusion_faults = '100036'
```

**示例 9** 对数据库 my\_db 运行启用了持久排除列表的 sp\_dbcc\_faultreport, 且对其进行了扩展, 使其不处理表 tab:

```
sp_dbcc_faultreport @dbname = 'my_db', @exclusion_mode = 'extend',
@exclusion_tables = 'tab'
```

**示例 10** 对数据库 my\_db 运行禁用了持久排除列表的 sp\_dbcc\_faultreport, 并启用了临时排除列表, 以便不处理表 tab 和故障类型 100036:

```
sp_dbcc_faultreport @dbname = 'my_db', @exclusion_mode = 'ignore',
@exclusion_faults = '100036', @exclusion_tables = 'tab'
```

**示例 11** 对数据库 my\_db 运行 sp\_dbcc\_faultreport, 同时禁用永久排除项列表并启用一个临时排除项列表, 以使与表 mytable 有关的故障类型“100002”和与表 tab 有关的故障类型 100035 排除在处理范围之外:

```
sp_dbcc_faultreport @dbname = 'my_db', @exclusion_mode = 'ignore',
@exclusion_combo = 'mytable:100002, tab:100035'
```

**示例 12** 生成一个长形式报告, 用以报告最近一次对 mydb 数据库运行 checkstorage 时所报告的 100029 故障 (100029 是页头错误的故障类型):

```
sp_dbcc_faultreport long, mydb, @fault_type_in = 100029
```

**示例 13** 生成一个简短形式的报告, 用以报告对 mydb 数据库运行操作 ID 为 5 的 checkstorage 时所报告的故障:

```
sp_dbcc_faultreport short, mydb, @opid = 5
```

## 用法

- sp\_dbcc\_faultreport 可以生成显示目标数据库的指定对象所发生的所有故障的报告。
- sp\_dbcc\_faultreport 在以下情况下将发出大量错误消息号 10028:
  - 使用 sp\_placeobject 令具有现有分配单元的对象将新分配单元放置在新段上。
  - 使用 sp\_dropsegment 从片段中删除某个段, 而该片段包含指派给该段的某个对象的分配单元。



错误消息号 100028 只是信息性消息，并不表示严重错误。如果不希望接收这类消息，可以自行创建报告过程，不报告此类（或其它）错误。一种方法就是将以下命令添加到 *installdbccdb* 脚本的标准 `sp_dbcc_faultreport` 存储过程的最前面：

```
print "removing 100028 errors from dbcc_faults table"
delete dbcc_faults where type_code = 100028
```

- 如果 `sp_dbcc_faultreport` 返回 *object\_name* 的值，则表示完成 dbcc checkstorage 操作后删除了该对象。

权限

对指定的数据库名有效的任何用户都可以运行 `sp_dbcc_faultreport`。

另请参见

有关故障 ID 和故障状态的信息，请参见《系统管理指南》中描述的 `type_code` 列。

**命令** `dbcc`

**dbcc 存储过程** `sp_dbcc_fullreport`、`sp_dbcc_statisticsreport`、`sp_dbcc_summaryreport`、`sp_dbcc_updateconfig`

## sp\_dbcc\_fullreport

|      |                                                                                                                                                                                                                                                                                                                                                           |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明   | 在指定的 <i>date</i> 或在此之前为 <i>database..object_name</i> 运行 <code>sp_dbcc_summaryreport</code> 、 <code>sp_dbcc_configreport</code> 、 <code>sp_dbcc_statisticsreport</code> 和 <code>sp_dbcc_faultreport short</code> 。                                                                                                                                         |
| 语法   | <code>sp_dbcc_fullreport [dbname [, objectname [, date]]]</code>                                                                                                                                                                                                                                                                                          |
| 参数   | <p><b>dbname</b><br/>指定数据库的名称。例如，<code>master..sysdatabases</code>。如果没有指定 <i>dbname</i>，报告将包含 <code>dbccdb..dbcc_operation_log</code> 中所有数据库的相关信息。</p> <p><b>object_name</b><br/>指定要为其生成报告的表或索引的名称。如果没有指定 <i>object_name</i>，则报告有关目标数据库中的所有对象的统计信息。</p> <p><b>date</b><br/>指定执行 <code>dbcc checkstorage</code> 操作的日期。如果没有指定 <i>date</i>，则使用最近一次操作的日期。</p> |
| 示例   | 为最近针对 <code>master</code> 数据库中的 <code>sysprocedures</code> 表运行的 <code>dbcc checkstorage</code> 操作运行 <code>sp_dbcc_summaryreport</code> 、 <code>sp_dbcc_configreport</code> 、 <code>sp_dbcc_statisticsreport</code> 和 <code>sp_dbcc_faultreport short</code> ： <pre>sp_dbcc_fullreport master, sysprocedures</pre>                                         |
| 用法   | <ul style="list-style-type: none"><li><code>sp_dbcc_fullreport</code> 在指定的日期或在此之前对 <i>database..object_name</i> 运行 <code>sp_dbcc_summaryreport</code>、<code>sp_dbcc_configreport</code>、<code>sp_dbcc_statisticsreport</code> 和 <code>sp_dbcc_faultreport short</code>。</li></ul>                                                                         |
| 权限   | 对指定的数据库名有效的任何用户都可以运行 <code>sp_dbcc_fullreport</code> 。                                                                                                                                                                                                                                                                                                    |
| 另请参见 | <p><b>命令</b> <code>dbcc</code></p> <p><b>dbcc 存储过程</b> <code>sp_dbcc_statisticsreport</code>、<code>sp_dbcc_summaryreport</code>、<code>sp_dbcc_updateconfig</code></p>                                                                                                                                                                                     |

## sp\_dbcc\_help\_fault

|    |                                                                                                                                                                  |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 提供指定故障类型的说明和建议的修复。                                                                                                                                               |
| 语法 | <code>sp_dbcc_help_fault [fault_type]</code>                                                                                                                     |
| 参数 | <p><i>fault_type</i></p> <p>是故障类型，应报告的就是此故障类型的说明和建议修复方法。此参数为 int 类型。如果未提供 <i>fault_type</i>，sp_dbcc_help_fault 将报告所有故障类型的情况。</p>                                 |
| 示例 | <p><b>示例 1</b> 要查看故障类型 100038 的说明及其建议的修复，请输入：</p> <pre>sp_dbcc_help_fault 100038</pre> <p><b>示例 2</b> 要查看所有故障类型的说明及其建议的修复，请输入：</p> <pre>sp_dbcc_help_fault</pre> |
| 用法 | sp_dbcc_help_fault 提供指定故障类型的说明以及建议的修复。                                                                                                                           |
| 权限 | 任何用户都可以运行 sp_dbcc_help_fault。                                                                                                                                    |

## sp\_dbcc\_patch\_finishtime

- 说明** 便于报告中止的 `checkverify` 和 `checkstorage` 操作的情况。
- 语法** `sp_dbcc_patch_finishtime dbname, opid [,optype [,seq [,finishtime]]]`
- 参数**
- dbname**  
是 `checkstorage` 或 `checkverify` 在中止时所操作的数据库的名称。此参数的类型为 `varchar`。
  - opid**  
是与中止的操作相对应的操作 ID。此参数的类型为 `smallint`。
  - optype**  
是要调查的操作的类型。可接受的值为 “`checkstorage`” 或 “`checkverify`”。此参数的类型为 `varchar`。
  - seq**  
是 `checkverify` 序列号（`checkstorage` 不使用序列号，但 `checkverify` 需要）。此参数的类型为 `smallint`。
  - finishtime**  
是一个 `datetime` 值，表示 `checkstorage` 或 `checkverify` 操作的中止时间。缺省值为当前时间。
- 示例** **示例 1** 出现以下错误时，为数据库 `my_db` 启用 `checkstorage` 和 `checkverify` 报告：

```
dbcc checkstorage (my_db)
```

```
Checking my_db:Logical pagesize is 2048 bytes
00:00000:00014:2003/01/20 11:50:05.01 server Error:9960, Severity:20,
State:1
A non-recoverable error has occurred in the CHECKSTORAGE operation.The
operation has been aborted.

Msg 9970, Level 20, State 1:
Line 2:
DBCC cannot update the finish time in dbcc_operation_log table for this
operation(opid = '1') of database 'my_db'.This can be patched by executing
sp_dbcc_patch_finishtime.
```

**示例 2** 出现以下错误时，为数据库 `my_db` 启用 `checkstorage` 和 `checkverify` 报告：

```
dbcc checkstorage (my_db)
```

```
Checking my_db:Logical pagesize is 2048 bytes
00:00000:00014:2003/01/20 11:50:05.01 server Error:9960, Severity:20,
State:1
A non-recoverable error has occurred in the CHECKSTORAGE operation.The
operation has been aborted.
```

```
Msg 9970, Level 20, State 1:
```

```
Line 2:
```

```
DBCC cannot update the finish time in dbcc_operation_log table for this
operation(opid = '1') of database 'my_db'.This can be patched by executing
sp_dbcc_patch_finishtime.
```

使用错误消息中包括的信息执行 `sp_dbcc_patch_finishtime`：

```
sp_dbcc_patch_finishtime my_db, 1
```

用法

当 `checkstorage` 或 `checkverify` 操作中止时，就会输出一条消息，其中包括该操作的 ID 和该操作中止时正在检查的数据库的名称。中止的 `checkverify` 操作还会在消息中提供一个序列号。消息会指示用户运行 `sp_dbcc_patch_finishtime`，并提供 `dbname`、`opid`，如果是 `checkverify` 操作，还会提供序列号 `seq`。在执行 `sp_dbcc_patch_finishtime` 后，可以创建有关已中止的操作的故障报告。

权限

只有系统管理员或数据库所有者才能运行 `sp_dbcc_patch_finishtime`。

## sp\_dbcc\_recommendations

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 分析由与指定的操作 ID 或日期相对应的 <code>checkstorage</code> 操作报告的故障，并为目标数据库中的指定对象生成推荐的更正操作列表。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 语法 | <code>sp_dbcc_recommendations dbname [,"date"[, <i>opid</i> [,"objectname"]]]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 参数 | <p><i>dbname</i></p> <p>是数据库的名称，就是要为此数据库生成建议。类型为 <code>varchar</code>，并且该参数是必需的。</p> <p><i>date</i></p> <p>是一个表示 <code>dbcc checkstorage</code> 操作完成的日期和时间的 <code>datetime</code> 值，将为该操作分析报告的故障。如果没有指定 <i>date</i> 或 <i>opid</i>，则 Adaptive Server 使用最近操作的日期。如果同时指定 <i>date</i> 和 <i>opid</i>，Adaptive Server 将忽略日期。<i>date</i> 是可选的。</p> <p><i>opid</i></p> <p>是 <code>checkstorage</code> 操作的操作 ID，将为该操作分析报告的故障。如果没有指定 <i>opid</i> 或 <i>date</i>，则 Adaptive Server 使用最近操作的日期。如果同时指定 <i>date</i> 和 <i>opid</i>，Adaptive Server 将忽略 <i>date</i>。此参数的类型为 <code>int</code>。</p> <p><i>objectname</i></p> <p>是 <code>sp_dbcc_recommendations</code> 为其生成建议的对象的名称。如果没有指定 <i>objectname</i>，则为数据库中所有对象生成建议。此参数的类型为 <code>varchar</code>。</p> |
| 示例 | <p><b>示例 1</b> 根据与操作 ID 2 相对应的 <code>checkstorage</code> 操作报告的故障，为数据库 <code>my_db</code> 中的对象 <code>t1</code> 生成建议的修复列表：</p> <pre>sp_dbcc_recommendations my_db, null, 2, 't1'</pre> <p><b>示例 2</b> 根据 2002 年 9 月 15 日下午 7:10:18:463 完成的 <code>checkstorage</code> 操作报告的故障，为数据库 <code>my_db</code> 中的所有对象生成建议的修复列表：</p> <pre>sp_dbcc_recommendations my_db, 'Sep 15 2002 7:10:18:463PM'</pre> <p><b>示例 3</b> 根据最近的 <code>checkstorage</code> 操作，为数据库 <code>my_db</code> 中的所有对象生成建议的修复列表：</p> <pre>sp_dbcc_recommendations my_db</pre>                                                                                                                                                                                                    |
| 用法 | <code>sp_dbcc_recomendations</code> 会分析由与指定的操作 ID 或日期相对应的 <code>checkstorage</code> 操作报告的故障，并为目标数据库中的指定对象生成推荐的更正操作列表                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 权限 | 目标数据库的任何有效用户都可以运行 <code>sp_dbcc_recommendations</code> 。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

## sp\_dbcc\_runcheck

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明   | 在指定数据库上运行 dbcc checkstorage，然后运行 <a href="#">sp_dbcc_summaryreport</a> 或您指定的报告。                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 语法   | <code>sp_dbcc_runcheck dbname [, user_proc]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 参数   | <p><b>dbname</b><br/>指定要检查的数据库的名称。</p> <p><b>user_proc</b><br/>指定要代替 <a href="#">sp_dbcc_summaryreport</a> 运行的 dbcc 存储过程或用户创建的存储过程的名称。</p>                                                                                                                                                                                                                                                                                                                                                                                                        |
| 示例   | <p><b>示例 1</b> 检查数据库 engdb 并根据查到的信息生成摘要报告：</p> <pre>sp_dbcc_runcheck "engdb"</pre> <p><b>示例 2</b> 检查数据库 pubs2 并生成完整报告：</p> <pre>sp_dbcc_runcheck "pubs2", sp_dbcc_fullreport</pre>                                                                                                                                                                                                                                                                                                                                                                |
| 用法   | <ul style="list-style-type: none"><li>• <code>sp_dbcc_runcheck</code> 在指定数据库上运行 dbcc checkstorage。</li><li>• 在 dbcc checkstorage 操作完成后，<code>sp_dbcc_runcheck</code> 将运行 <a href="#">sp_dbcc_summaryreport</a> 生成摘要报告。如果为 <code>dbcc_report</code> 指定了其它某个可生成报告的 dbcc 存储过程，<code>sp_dbcc_runcheck</code> 将运行该过程而不是 <a href="#">sp_dbcc_summaryreport</a>。有关 dbccdb 附带的所有可生成报告的存储过程的简短说明和示例，请参见《系统管理指南》。</li><li>• 可以自行编写用于生成报告的存储过程并指定其名称作为 <code>user_proc</code>。存储过程必须是独立的。<code>sp_dbcc_runcheck</code> 不能向 Adaptive Server 传递任何参数。</li></ul> |
| 权限   | 只有系统管理员或数据库所有者才能运行 <code>sp_dbcc_runcheck</code> 。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 另请参见 | <p><b>命令</b> <a href="#">dbcc</a></p> <p><b>dbcc 存储过程</b> <a href="#">sp_dbcc_summaryreport</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                               |

## sp\_dbcc\_statisticsreport

**说明** 针对目标数据库中的指定对象生成分配统计信息报告。

**语法** sp\_dbcc\_statisticsreport [dbname [, objectname [, date]]]

**参数**

*dbname*

指定**目标数据库**。如果未指定 *dbname*，报告将包含 dbccdb..dbcc\_operation\_log 中所有数据库的相关信息。

*objectname*

指定要为其生成报告的表或索引的名称。如果没有指定 *objectname*，Adaptive Server 将报告有关目标数据库中的所有对象的统计信息。

*date*

指定执行 dbcc checkstorage 操作的日期。如果没有指定 *date*，则 Adaptive Server 使用最近操作的日期。

**示例**

针对 sybsystemprocs 数据库中的 sysobjects 表生成统计信息报告：

```
sp_dbcc_statisticsreport 'sybsystemprocs',
 'sysobjects'
```

```
Statistics Report on object sysobjects in database
sybsystemprocs
```

| Parameter Name | Index Id | Value   |
|----------------|----------|---------|
| count          | 0        | 241.0   |
| max size       | 0        | 99.0    |
| max count      | 0        | 22.0    |
| bytes data     | 0        | 19180.0 |
| bytes used     | 0        | 22113.0 |
| count          | 1        | 14.0    |
| max size       | 1        | 9.0     |
| max level      | 1        | 0.0     |
| max count      | 1        | 14.0    |
| bytes data     | 1        | 56.0    |
| bytes used     | 1        | 158.0   |
| count          | 2        | 245.0   |
| max level      | 2        | 1.0     |
| max size       | 2        | 39.0    |
| max count      | 2        | 71.0    |
| bytes data     | 2        | 4377.0  |
| bytes used     | 2        | 6995.0  |

| Parameter Name | Index Id | Partition | Value | Dev_name |
|----------------|----------|-----------|-------|----------|
| -----          | -----    | -----     | ----- | -----    |
| -----          | -----    | -----     | ----- | -----    |



|                     |   |   |      |        |
|---------------------|---|---|------|--------|
| page gaps           | 0 | 1 | 13.0 | master |
| pages used          | 0 | 1 | 15.0 | master |
| extents used        | 0 | 1 | 3.0  | master |
| overflow pages      | 0 | 1 | 0.0  | master |
| pages overhead      | 0 | 1 | 1.0  | master |
| pages reserved      | 0 | 1 | 7.0  | master |
| page extent gaps    | 0 | 1 | 11.0 | master |
| ws buffer crosses   | 0 | 1 | 2.0  | master |
| page extent crosses | 0 | 1 | 11.0 | master |
| pages used          | 1 | 1 | 2.0  | master |
| extents used        | 1 | 1 | 1.0  | master |
| overflow pages      | 1 | 1 | 0.0  | master |
| pages overhead      | 1 | 1 | 1.0  | master |
| pages reserved      | 1 | 1 | 6.0  | master |
| page extent gaps    | 1 | 1 | 0.0  | master |
| ws buffer crosses   | 1 | 1 | 0.0  | master |
| page extent crosses | 1 | 1 | 0.0  | master |
| page gaps           | 2 | 1 | 4.0  | master |
| pages used          | 2 | 1 | 6.0  | master |
| extents used        | 2 | 1 | 1.0  | master |
| overflow pages      | 2 | 1 | 0.0  | master |
| pages overhead      | 2 | 1 | 1.0  | master |
| pages reserved      | 2 | 1 | 2.0  | master |
| page extent gaps    | 2 | 1 | 0.0  | master |
| ws buffer crosses   | 2 | 1 | 0.0  | master |
| page extent crosses | 2 | 1 | 0.0  | master |

## 用法

- `sp_dbcc_statisticsreport` 为目标数据库中的指定对象生成分配统计信息报告。它使用 `dbcc_counters` 表中的数据，该表存储有关目标数据库中每个对象的页利用率和错误统计信息的信息。
- 如果 `sp_dbcc_statisticsreport` 返回的 `object_name` 为数值，则表示完成 `dbcc checkstorage` 操作后删除了该对象。
- `sp_dbcc_statisticsreport` 为数据类型 5000 - 5024 报告记录在 `dbcc_counters` 表中的值。有关详细信息，请参见《参考手册：表》中的 `dbcc_counters`。

对于 `bytes data`、`bytes used` 和 `overflow pages`，`sp_dbcc_statisticsreport` 报告为所有分区和设备报告的值的总和。

对于 `count`、`max count`、`max size` 和 `max level`，`sp_dbcc_statisticsreport` 报告所有分区和设备的最大值。

`sp_dbcc_statisticsreport` 为目标数据库中的对象使用的每个设备和分区报告以下各行信息：

- extents used
- io errors
- page gaps
- page extent crosses
- page extent gaps
- page format errors
- pages reserved
- pages overhead
- pages misallocated
- pages not allocated
- pages not referenced
- pages used

`page gaps`、`page extent crosses` 和 `page extent gaps` 表示对象的数据页如何在数据库设备上分配。较大值表示使用较大缓冲区和数据预取时的效率较低。

- 如果在同一天在目标数据库运行了多个 `dbcc checkstorage` 操作，`sp_dbcc_statisticsreport` 将根据在指定时间之前完成的最后一个 `dbcc checkstorage` 操作的结果生成报告。

权限

对指定的数据库名有效的任何用户都可以运行 `sp_dbcc_statisticsreport`。

另请参见

**命令** [dbcc](#)

**dbcc 存储过程** [sp\\_dbcc\\_fullreport](#)、[sp\\_dbcc\\_summaryreport](#)、[sp\\_dbcc\\_updateconfig](#)

## sp\_dbcc\_summaryreport

- 说明** 生成指定数据库的摘要报告。
- 语法** sp\_dbcc\_summaryreport [dbname [, date [, op\_name [, display\_recommendations]]]]
- 参数**
- dbname**  
指定数据库的名称，就是要为该数据库生成报告。如果没有指定 **dbname**，sp\_dbcc\_summaryreport 将为 dbccdb..dbcc\_operation\_log 中的所有数据库生成报告，操作执行的日期应为 **date** 选项指定的日期和时间或是此前的日期和时间。
- date**  
指定执行 dbcc checkstorage 的日期。如果没有指定日期，sp\_dbcc\_summaryreport 将使用对目标数据库执行的最近一次 dbcc checkstorage 操作的日期。此参数的数据类型为 datetime。如果对 **date** 同时指定了日期和时间，则报告在指定时间或此前执行的所有操作的摘要结果。如果未指定日期，则报告所有操作。
- opname**  
指定操作。**opname** 可以是 checkstorage，这是缺省操作，或是 checkverify，也可以是这两者。如果未指定 **opname**，则对所有操作都生成报告。
- display\_recommendations**  
启用报告由 sp\_dbcc\_recommendations 生成的建议

**示例** **示例 1** 生成 sybssystemprocs 数据库的摘要报告，报告中提供了有关执行的所有 dbcc checkstorage 和 dbcc checkverify 操作的信息：

```
sp_dbcc_summaryreport
```

```
DBCC Operation :checkstorage
```

| Database Name   | Start time          | End Time     | Operation ID |
|-----------------|---------------------|--------------|--------------|
| Hard Faults     | Soft Faults         | Text         | Columns      |
|                 |                     | Abort        | Count        |
|                 |                     |              | User Name    |
| sybssystemprocs | 05/11/1999 14:53:11 | 14:53:32:163 | 1            |
| 0               | 0                   | 0            | sa           |
| sybssystemprocs | 05/11/1999 14:55:06 | 14:55:29:200 | 2            |
| 0               | 0                   | 0            | sa           |
| sybssystemprocs | 05/11/1999 14:56:10 | 14:56:27:750 | 3            |
| 0               | 0                   | 0            | sa           |

```
DBCC Operation :checkverify
```

| Database Name | Start time  | End Time  | Operation ID |
|---------------|-------------|-----------|--------------|
| Hard Faults   | Soft Faults | User Name |              |

```


sybssystemprocs 05/11/1999 14:55:29 14:55:29:310 2
 0 0 sa

```

**示例 2** 生成用户数据库 `testdb` 的摘要报告，报告中提供了有关执行的所有 `dbcc checkstorage` 操作的信息。`dbcc checkstorage` 是此数据库执行的唯一操作，所以报告中不显示 `dbcc checkverify` 的信息：

```

sp_dbcc_summaryreport "testdb"
DBCC Operation :checkstorage

```

| Database Name | Start time          | End Time     | Operation ID |
|---------------|---------------------|--------------|--------------|
| Hard Faults   | Soft Faults         | Text Columns | Abort Count  |
|               |                     |              | User Name    |
| testdb        | 05/11/1999 14:55:29 | 14:55:49:903 | 1            |
| 0             | 0                   | 0            | sa           |
| testdb        | 05/11/1999 14:55:50 | 14:56:9:546  | 2            |
| 0             | 0                   | 0            | sa           |
| testdb        | 05/11/1999 14:56:28 | 14:56:40:666 | 3            |
| 0             | 0                   | 0            | sa           |

**示例 3** 生成 `sybssystemprocs` 数据库的摘要报告，报告中提供了有关执行的所有 `dbcc checkverify` 操作的信息。由于 `dbcc checkverify` 是指定的操作，所以报告中不显示 `dbcc checkstorage` 的信息：

```

1> sp_dbcc_summaryreport null, null, "checkverify"
2> go

```

```

DBCC Operation :checkverify

```

| Database Name | Start time          | End Time    | Operation ID | Run Srl   |
|---------------|---------------------|-------------|--------------|-----------|
| Table Name    | Table Id            | Hard Faults | Soft Faults  | User Name |
| testdb        | 08/31/2004 11:06:11 | 11:6:11:370 | 3            | 1         |
| NULL          | NULL                | 0           | 0            | sa        |

```

(1 row affected)

```

**示例 4** 生成 sybssystemprocs 数据库的摘要报告，提供有关执行的所有 dbcc checkstorage 操作的信息。由于 dbcc checkstorage 是指定的操作，所以报告中不显示 dbcc checkverify 的信息：

```
sp_dbcc_summaryreport sybssystemprocs, null, "checkstorage"
DBCC Operation :checkstorage

Database Name Start time End Time Operation ID
 Hard Faults Soft Faults Text Columns Abort Count User Name

sybssystemprocs 05/11/1999 14:53:11 14:53:32:163 1
0 0 0 0 sa
sybssystemprocs 05/11/1999 14:55:06 14:55:29:200 2
0 0 0 0 sa
sybssystemprocs 05/11/1999 14:56:10 14:56:27:750 3
0 0 0 0 sa
```

**示例 5** 向数据库 my\_db 的摘要报告添加建议的修复：

```
sp_dbcc_summaryreport @dbname = my_db,
@display_recommendations = 1
```

#### 用法

- `sp_dbcc_summaryreport` 针对指定数据库生成 `checkstorage` 和 / 或 `checkverify` 操作的摘要报告。
- 报告将指明接受检查的数据库的名称、`dbcc checkstorage` 运行的开始和结束时间，以及找到的软硬故障数。
- “Operation ID” 列包含编号，用于标识在特定时间对给定数据库执行的各次 `dbcc checkstorage` 操作的结果。报告中提供的编号来自 `dbcc_operation_log` 表的 `opid` 列。有关详细信息，请参见《系统管理指南》。
- “Text Columns” 列显示 `dbcc checkstorage` 在运行期间找到的非空文本列的数目。
- “Abort Count” 列显示包含错误的表数，这些错误导致 `dbcc checkstorage` 中止对表的检查。有关错误的详细信息，请运行 `sp_dbcc_faultreport`。

#### 权限

对指定的数据库名有效的任何用户都可以运行 `sp_dbcc_summaryreport`。

#### 另请参见

**命令** `dbcc`

**dbcc 存储过程** `sp_dbcc_fullreport`、`sp_dbcc_statisticsreport`、`sp_dbcc_updateconfig`

## sp\_dbcc\_updateconfig

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 说明 | 使用目标数据库的配置信息更新 dbccdb 中的 dbcc_config 表。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 语法 | sp_dbcc_updateconfig <i>dbname</i> , <i>type</i> , " <i>str1</i> " [, " <i>str2</i> "]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 参数 | <p><i>dbname</i><br/>是要更新配置信息的目标数据库的名称。要配置缺省值，请输入空值 <i>dbname</i> 参数。</p> <p><i>type</i><br/>指定 dbcc_types 表中的类型名。第 757 页的表 4-2 显示了 <i>type</i> 的有效值。</p> <p><i>str1</i><br/>指定 <i>dbcc_config</i> 表中要更新的指定 <i>type</i> 的第一个配置值。第 757 页的表 4-2 说明了指定 <i>type</i> 的 <i>str1</i> 期望值。</p> <p><i>str2</i><br/>指定 <i>dbcc_config</i> 表中要更新的指定 <i>type</i> 的第二个配置值。第 757 页的表 4-2 说明了指定 <i>type</i> 的 <i>str2</i> 期望值。</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 示例 | <p><b>示例 1</b> 在检查 pubs2 数据库时，用要使用的 dbcc checkstorage 最大工作进程数来更新 dbcc_config。新的最大工作进程数为 4:</p> <pre>sp_dbcc_updateconfig pubs2, "max worker processes", "4"</pre> <p><b>示例 2</b> 下面的语句将 max worker processes 设置为 2:</p> <pre>sp_dbcc_updateconfig null, 'max worker processes', '2'</pre> <p><b>示例 3</b> 用 dbcc 命名高速缓存“pubs2_cache”的大小来更新 dbcc_config。新的大小为 10K:</p> <pre>sp_dbcc_updateconfig pubs2, "dbcc named cache", pubs2_cache, "10K"</pre> <p><b>示例 4</b> 用 pubs2 数据库的 scan 工作空间的新名称来更新 dbcc_config。新名称为 scan_pubs2。此更新操作要在使用 sp_dbcc_alterws 更改 scan 工作空间的名称之后执行:</p> <pre>sp_dbcc_updateconfig pubs2, "scan workspace", scan_pubs2</pre> <p><b>示例 5</b> 用 pubs2 数据库的 text 工作空间的新名称来更新 dbcc_config。新名称为 text_pubs2。此更新操作要在使用 sp_dbcc_alterws 更改 text 工作空间的名称之后执行:</p> <pre>sp_dbcc_updateconfig pubs2, "text workspace", text_pubs2</pre> <p><b>示例 6</b> 用 pubs2 数据库的 OAM 计数阈值来更新 dbcc_config。新值为 5:</p> <pre>sp_dbcc_updateconfig pubs2, "OAM count threshold", "5"</pre> |

**示例 7** 用 pubs2 数据库的 I/O 错误中止值来更新 dbcc\_config。新值为 3：  
`sp_dbcc_updateconfig pubs2, "IO error abort", "3"`

**示例 8** 用 pubs2 数据库的链接错误中止值来更新 dbcc\_config。新值为 8：  
`sp_dbcc_updateconfig pubs2, "linkage error abort", "8"`

**示例 9** 为数据库 my\_db 启用 automatic workspace expansion：  
`sp_dbcc_updateconfig my_db, "enable automatic workspace expansion", "1"`

## 用法

- `sp_dbcc_updateconfig` 更新目标数据库的 `dbcc_config` 表。
- 如果目标数据库名为 `dbccdb` 并且数据库 `dbccalt` 存在，`sp_dbcc_updateconfig` 将更新 `dbccalt` 中的 `dbcc_config` 表。
- 如果在 `dbcc_config` 中未找到目标数据库名，`sp_dbcc_updateconfig` 会添加该名称，并在更新其它配置信息之前先将操作序列号设置为 0。
- 如果指定 `type` 的期望值是一个数字，`sp_dbcc_updateconfig` 会将您提供的 `str1` 和 `str2` 的值转换为数字。
- 表 4-2 显示了用于 `type` 的有效类型名以及 `str1` 或 `str2` 的期望值。

表 4-2: 类型名和期望值

| <b>type 名</b>                 | <b>str1 或 str2 的期望值</b>                                                              |
|-------------------------------|--------------------------------------------------------------------------------------|
| dbcc named cache              | <i>str1</i> 指定的高速缓存名称，以及 <i>str2</i> 指定的新大小（以 KB 或 MB 计）或 2K 页数。                     |
| IO error abort                | <i>str1</i> 指定的新错误计数。该值必须是大于 0 的数字。 <i>str2</i> 不与此类型一起使用。                           |
| linkage error abort           | <i>str1</i> 中指定的新链接错误计数。该值必须是大于 0 的数字。 <i>str2</i> 不与此类型一起使用。                        |
| max worker processes          | <i>str1</i> 中指定的新工作进程数。该值必须是大于 0 的数字。 <i>str2</i> 不与此类型一起使用。                         |
| OAM count threshold           | <i>str1</i> 指定的新阈值计数。该值必须是大于 0 的数字。 <i>str2</i> 不与此类型一起使用。                           |
| scan workspace                | scan 工作空间的新名称，由 <i>str1</i> 指定。 <i>str2</i> 不与此类型一起使用。                               |
| text workspace                | text 工作空间的新名称，由 <i>str1</i> 指定。 <i>str2</i> 不与此类型一起使用。                               |
| automatic workspace expansion | 如果各个段有足够的空间，则允许 <code>checkstorage</code> 自动扩展工作空间。采用缺省值 1 时会启用自动工作空间扩展，采用值 0 时会禁用它。 |

- 有关 `type` 名称和值的详细信息，请参见《系统管理指南》。

## 权限

只有系统管理员或数据库所有者才能运行 `sp_dbcc_updateconfig`。

## 另请参见

**命令** `dbcc`

**dbcc 存储过程** `sp_dbcc_alterws`、`sp_dbcc_evaluatedb`

**系统过程** `sp_plan_dbccdb`





# 索引

## 符号

- @ (at 符号) 和过程参数 3
- , (逗号)
  - 用户定义的数据类型 54
- () (小括号)
  - 用户定义的数据类型 54
- “ ” (引号)
  - 单, 和 **quoted\_identifier** 118
  - 将保留字引起来 112
  - 将参数值括起来 3, 680
- .. (圆点) 在数据库对象名称中 38
- @@cpu\_busy** 全局变量和 **sp\_monitor** 483
- @@packet\_errors** 全局变量和 **sp\_monitor** 483
- @@thresh\_hysteresis** 全局变量
  - 阈值放置和 51
- 0 返回状态 1, 680
- <ix\_commands>p\_cleanpwdchecks 存储过程 127

## 数字

- 7 位 ASCII 字符, 使用 **sp\_checknames** 检查 105
- 7 位终端, **sp\_helpsort** 输出 396
- 7 位终端, **sp\_helpsort** 输出 ] 396
- 8 位终端, **sp\_helpsort** 输出 396
- 8 位终端, **sp\_helpsort** 输出 ] 396

## 英文

- abort tran on log full** 数据库选项 195
- adhoc** 审计选项 63
- all** 审计选项 63
- allow nulls by default** 数据库选项 195
- ALS (异步日志服务) 358
  - sp\_helpdb** 中的输出 360
- alter database** 命令
  - sp\_dbremap** 和 203

- alter** 审计选项 63
- ANYENGINE** 引擎组 22
- ASCII 字符
  - 使用 **sp\_checknames** 检查 105
- @@connections** 全局变量和 **sp\_monitor** 483
- auto identity** 数据库选项 195
- Backup Server
  - 另请参见 《实用程序指南》
  - 多个 48
  - 卷处理消息 662–664
  - 有关信息 394
  - 转储的量, 指定 274
- bcp**
  - 使用 **sp\_audit** 的审计选项 63
- bcp** (批量复制实用程序)
  - select into/bulkcopy/pllsort** 和 198
  - 快速版本 198
- bind** 审计选项 63
- checkpoint** 命令
  - 设置数据库选项和 191
- cmdtext** 审计选项 63
- concurrency\_opt\_threshold** 选项, **sp\_chgattribute** 122
- CPU 使用率, 监控 483
- create database** 命令
  - log on** 选项, 与 **sp\_logdevice** 相比 448
- create index** 命令和 **sp\_extendsegment** 301
- create table** 命令和 **sp\_extendsegment** 301
- create** 审计选项 63
- cursors 177
- days
  - 时间范围中 13
  - 替代语言 32
- dbaccess** 审计选项 63
- dbcc** 审计选项 63
- dbccdb** 数据库
  - 报告 I/O 统计信息 733
  - 报告分配统计信息 750

- 报告故障信息 733, 739
- 报告配置信息 727, 739, 744
- 报告所有细节 744
- 报告综合信息 744
- 创建工作空间 728
- 更改工作空间大小 726
- 删除 **dbcc checkstorage** 历史记录 731
- 删除目标数据库信息 730
- 要使用的存储过程 723
- dbcc** (数据库一致性检查程序)
  - 脚本和 **sp\_checkreswords** 114
  - 空间分配 528
- DB-Library 程序和更改标识符名称 113
- dbo use only** 数据库选项
  - 使用 **sp\_dboption** 设置 196
- ddl in tran** 数据库选项 196
- default* 段
  - 删除 268
  - 映射 45
- defncopy** 实用程序命令 113
- delete** 审计选项 63
- disk** 审计选项 63
- drop logins** 选项, **sp\_dropserver** 270
- drop** 审计选项 63
- dropmessages** 选项, **sp\_droplanguage** 261
- dsync 设置 215
- dump transaction** 命令
  - trunc log on chkpt** 和 199
- dump transaction** 命令和 **sp\_logdevice** 448
- dump** 审计选项 63
- errors** 审计选项 63
- exec\_procedure** 审计选项 63
- exec\_trigger** 审计选项 63
- exp\_row\_size** 选项
  - sp\_chgattribute** 122
  - sp\_help** 报告 332
- expand\_down** 参数
  - sp\_activeroles** 6
  - sp\_displayroles** 232
  - sp-displayroles** 232
- external engine auto start** 选项
  - sp\_serveroption** 571
- fid* (系列 ID) 编号 305
  - sp\_lock** 报告 443
- from** 关键字和子句中显示的 **sp\_tables** 对象列表 702
- func\_dbaccess** 审计选项 63
- func\_obj\_access** 审计选项 63
- futureonly** 选项
  - sp\_bindefault** 80
  - sp\_bindrule** 88, 89
  - sp\_unbindefault** 652
  - sp\_unbindrule** 657
- grant** 审计选项 64
- grant** 选项
  - sp\_helprotect** 385
  - sp\_role** 563
- guest 用户和 *sybserverprocs* 数据库 2
- Halloween 问题
  - 使用 **unique auto\_identity index** 选项避免 199
- holdlock** 关键字
  - select** 443
- I/O
  - concurrency\_opt\_threshold** 和 122
  - 配置大小 532
  - 日志大小 454
  - 使用情况统计信息 560
  - 限制 9
- ID, 时间范围 14
- ID, 用户
  - 另请参见 登录名
- identity in nonunique index** 数据库选项, 使用 **sp\_dboption** 设置 197
- IDENTITY 列
  - 非唯一索引 197
  - 数据库选项使用 197
  - 自动 195, 199
- image* 数据类型, 大小 613
- insert** 审计选项 64
- interfaces 文件
  - sp\_addserver** 和 47
  - 更改服务器名 117
- Java 项和 **sp\_helpjava** 系统过程 374
- kill** 命令和 **sp\_who** 675
- LASTONLINE* 引擎组 22
- LDAP 用户鉴定
  - 每个引擎的 LDAP 描述符数 428
  - 每个引擎的最大 *ldapua* 本机线程数 427
- LDAP, URL 搜索字符串 423

- load**
  - 重建索引 537
- load** 审计选项 64
- local** 选项, **sp\_addserver** 46
- lock | unlock** 选项, **sp\_locklogin** 445
- log on** 选项
  - create database**, 和 **sp\_logdevice** 448
- login** 审计选项 64
- login\_locked** 审计选项 64
- logout** 审计选项 64
- logsegment** 日志存储, 删除 268
- master** 数据库
  - sp\_dboption** 和 191
  - 使用 **sp\_checkreswords** 检查 111
  - 系统过程表 5
  - 阈值和 51, 474
- max\_rows\_per\_page** 选项
  - sp\_chgattribute** 122
  - 使用 **sp\_relimit** 更改 122
- memory**
  - 从 XP Server 释放 320
- model** 数据库
  - 更改数据库选项 198
- model** 数据库, 更改数据库选项 191
- MRU** 替换策略
  - 禁用 99
- mutual authentication** 选项, **sp\_serveroption** 571
- negotiated logins** 选项
  - sp\_serveroption** 572
- net password encryption** 选项
  - sp\_serveroption** 571
- no chkpt on recovery** 数据库选项, 使用
  - sp\_dboption** 设置 197
- no free space acctg** 数据库选项, 使用 **sp\_dboption** 设置 197
- ODBC**。请参见 开放式数据库互连 (ODBC) API 数据类型。
- Open Client** 应用程序
  - 连接安全性 42
- optdiag** 实用程序
  - 刷新内存中的统计信息 311
- output** 选项
  - sp\_getmessage** 321
- packet\_errors** 全局变量
  - sp\_monitor** 483
- PC DB-Library**。请参见 DB-Library 程序
- “probe” 登录帐户 560
- “public” 组
  - 另请参见组
  - sp\_addgroup** 和 31
  - sp\_adduser** 和 60
  - sp\_changegroup** 和 103
  - sp\_helpgroup** 报告 369
  - 信息报告 369
- read only** 数据库选项
  - 使用 **sp\_dboption** 设置 197
  - 使用 **sp\_setsuspect\_granularity** 设置 583
- readonly** 选项, **sp\_serveroption** 571
- reference** 审计选项 64
- reindex** 选项, **dbcc**
  - 在 **sp\_indsuspect** 后 412
- reservepagegap** 选项
  - sp\_chgattribute** 122
  - sp\_help** 报告 332
- revoke** 审计选项 64
- revoke** 选项, **sp\_role** 563
- rpc** 审计选项 64
- security mechanism** 选项, **sp\_serveroption** 571
- security** 审计选项 64
- select into/bulkcopy/pilsort** 数据库选项
  - 事务日志和 197
- select** 审计选项 64
- server cost** 选项
  - sp\_serveroption** 571
- server login** 选项
  - sp\_serveroption** 572
- set** 命令
  - sp\_setlangalias** 和 **language** 选项 576
- setuser** 审计选项 64
- single user** 数据库选项
  - 使用 **sp\_dboption** 设置 198
- size**
  - text** 存储 613
- size of auto identity column** 配置参数 195, 199
- sp\_activeroles** 系统过程 6–7
- sp\_add\_qpgroup** 系统过程 8
- sp\_add\_resource\_limit** 系统过程 9–12
- sp\_add\_time\_range** 系统过程 13–15
- sp\_addalias** 系统过程 16–17
- sp\_addauditrecord** 系统过程 18–19

- sp\_addaudittable 系统过程 20–21
- sp\_addengine 系统过程 22–23
- sp\_addexeclass 系统过程 24–25
- sp\_addextendedproc 系统过程 26–27
- sp\_addexternlogin 系统过程 28–30
- sp\_addgroup 系统过程 31
- sp\_addlanguage 系统过程 32–34
- sp\_addmessage 系统过程 36–37
- sp\_addobjectdef 系统过程 38–40
- sp\_addremotelogin 系统过程 41–43
- sp\_addsegment 系统过程 44–45
  - 在既有数据又有日志的数据库中 45
- sp\_addserver 系统过程 46–49
- sp\_addthreshold 系统过程 50–53
- sp\_addtype 系统过程 54–57
- sp\_addumpdevice 系统过程 58–59
- sp\_adduser 系统过程 60–61
- sp\_altermessage 系统过程 62
- sp\_audit 系统过程 63–68
- sp\_autoconnect 系统过程 69–70
- sp\_autoformat 系统过程 71–75
- sp\_bindcache 系统过程 76–79
- sp\_bindefault 系统过程 80–82
  - create default 和 81
- sp\_bindexclass 系统过程 83–85
- sp\_bindmsg 系统过程 86–87
- sp\_bindrule 系统过程 88–89
- sp\_cacheconfig 系统过程 90–98
- sp\_cachestrategy 系统过程 99–101
- sp\_changedbowner 系统过程 102
- sp\_changegroup 系统过程 103–104
  - 另请参见 sp\_dropgroup
  - sp\_dropgroup 和 258
- sp\_checknames 系统过程 105–107
- sp\_checkreswords 系统过程 108–119
  - 返回状态 111
- sp\_checksourc 系统过程 120–121
- sp\_chgattribute 系统过程 122–126
- sp\_clearpsex 系统过程 128
- sp\_clearstats 系统过程 129–130
- sp\_client\_addr 系统过程 131
- sp\_cluster\_logical 存储过程 135
- sp\_cmp\_all\_qplans 系统过程 156
- sp\_cmp\_qplans 系统过程 158
- sp\_column\_privileges 目录存储过程 683–684
- sp\_columns 目录存储过程 685–686
  - sp\_datatype\_info 和 688
  - 数据类型代码编号 681
- sp\_commonkey 系统过程 160–161
- sp\_companion 系统过程 162–164
- sp\_compatmode 165
- sp\_configure 系统过程 166–172
  - 设置显示级别 223
- sp\_copy\_all\_qplans 系统过程 173
- sp\_copy\_qplan 系统过程 174
- sp\_countmetadata 系统过程 175
- sp\_cursorinfo 系统过程 177–179
- sp\_databases 目录存储过程 687
- sp\_datatype\_info 目录存储过程 688
- sp\_dbcc\_alterws 存储过程 726
- sp\_dbcc\_configreport 存储过程 727
- sp\_dbcc\_createws 存储过程 728–729
- sp\_dbcc\_deletedb 存储过程 730
- sp\_dbcc\_deletehistory 存储过程 731–732
- sp\_dbcc\_differentialreport 存储过程 733–734
- sp\_dbcc\_evaluatedb 存储过程 735–736
- sp\_dbcc\_exclusions 737
- sp\_dbcc\_faultreport 存储过程 739–743
- sp\_dbcc\_fullreport 存储过程 744
- sp\_dbcc\_help\_fault 745
- sp\_dbcc\_patch\_finishtime 746
- sp\_dbcc\_recommendations 748
- sp\_dbcc\_runcheck 存储过程 749
- sp\_dbcc\_statisticsreport 存储过程 750–752
- sp\_dbcc\_summaryreport 存储过程 753–755
- sp\_dbcc\_updateconfig 存储过程 756–757
- sp\_dbextend 系统过程 180–186
- sp\_dboption 系统过程 187–200
- sp\_dbrecovery\_order 系统过程 201–202
- sp\_dbremap 系统过程 203
- sp\_defaultloc 系统过程 204–206
- sp\_deletesmobj 存储过程, 语法和用法 207
- sp\_depends 系统过程 209–214
  - 数据库对象所有者和 209
- sp\_deviceattr 系统过程 215–216
- sp\_diskdefault 系统过程 217–218
- sp\_displayaudit 系统过程 219–222
- sp\_displaylevel 系统过程 223–224
- sp\_displaylogin 系统过程 225–231
- sp\_displayroles 系统过程 232
- sp\_drop\_all\_qplans 系统过程 244

- sp\_drop\_qpgroup 系统过程 245
- sp\_drop\_qplan 系统过程 246
- sp\_drop\_resource\_limit 系统过程 247–249
- sp\_drop\_time\_range 系统过程 250
- sp\_dropalias 系统过程 242–243
- sp\_dropdevice 系统过程 251
- sp\_dropengine 系统过程 252
- sp\_dropexclass 系统过程 254
- sp\_dropextendedproc 系统过程 255
- sp\_dropexternlogin 系统过程 256
- sp\_dropglockpromote 系统过程 257
- sp\_dropgroup 系统过程 258
- sp\_dropkey 系统过程 259–260
- sp\_droplanguage 系统过程 261
- sp\_dropmessage 系统过程 263
- sp\_dropobjectdef 系统过程 264–265
- sp\_dropremotelogin 系统过程 266
- sp\_droprowlockpromote 系统过程 267
- sp\_dropsegment 系统过程 268–269
  - sp\_placeobject 和 268
- sp\_dropserver 系统过程 270
- sp\_droptreshold 系统过程 271
- sp\_droptype 系统过程 272
- sp\_dropuser 系统过程 273
- sp\_dumpoptimize 系统过程 274–278
- sp\_encryption 系统过程 279–290
- sp\_engine 系统过程 291–294
- sp\_estspace 系统过程 295–299
- sp\_export\_qpgroup 系统过程 300
- sp\_extendsegment 系统过程 301–302
  - create index 和 301
  - create table 和 301
- sp\_extengine 系统过程 303
- sp\_extrapwdchecks 304
- sp\_familylock 系统过程 305–306
- sp\_find\_qplan 系统过程 307–308
- sp\_fixindex 系统过程 309–310
- sp\_fkeys 目录存储过程 689–690
- sp\_flushstats 系统过程 311
- sp\_forceonline\_db 系统过程 312–313
- sp\_forceonline\_object 系统过程 314–315
- sp\_forceonline\_page 系统过程 316–317
- sp\_foreignkey 系统过程 318–319
- sp\_freelld 系统过程 320
- sp\_getmessage 系统过程 321
- sp\_grantlogin 系统过程 322
- sp\_ha\_admin 系统过程 324
  - 用 installhasvss 安装 324
- sp\_help 系统过程 326–333
- sp\_help\_qpgroup 系统过程 337–338
- sp\_help\_qplan 系统过程 339–340
- sp\_help\_resource\_limit 系统过程 334–336
- sp\_helppartition 系统过程 342
- sp\_helpcache 系统过程 346–347
- sp\_helpcomputedcolumn 系统过程 348
- sp\_helpconfig 系统过程 349–353
- sp\_helpconstraint 系统过程 354–357
- sp\_helpdb 361
- sp\_helpdb 系统过程 358–361
  - tempdb 和 360
- sp\_helpdevice 系统过程 362–364
- sp\_helpextendedproc 系统过程 365–366
- sp\_helpexternlogin 系统过程 367
- sp\_helpgroup 系统过程 369–370
- sp\_helpindex 系统过程 371–373
- sp\_helpjava 系统过程 374–376
- sp\_helpjoins 系统过程 377–378
- sp\_helpkey 系统过程 379–380
- sp\_helplanguage 系统过程 381
- sp\_helplog 系统过程 382
- sp\_helpobjectdef 系统过程 383
- sp\_helpremotelogin 系统过程 384
- sp\_helpprotect
  - 和加密列 388
- sp\_helpprotect 系统过程 385–389
- sp\_helpsegment 系统过程 390–393
- sp\_helpserver 系统过程 394–395
- sp\_helpsort 系统过程 396–397
- sp\_helptext 存储过程, 更改 401
- sp\_helptext 系统过程 398–402
- sp\_helpthreshold 系统过程 405
- sp\_helpuser 系统过程 406–407
- sp\_hidetext 系统过程 408
- sp\_import\_qpgroup 系统过程 410–411
- sp\_indsuspect 系统过程 412
- sp\_reconfig 413
- sp\_ldapadmin 系统过程 423–428
- sp\_listener 系统过程 430
- sp\_listsuspect\_db 系统过程 434
- sp\_listsuspect\_object 系统过程 435
- sp\_listsuspect\_page 系统过程 436
- sp\_lmconfig 系统过程 437–440

- sp\_lock 系统过程 441–444
- sp\_locklogin 系统过程 445
- sp\_logdevice 系统过程 448–449
  - 对 create database 的 log on 扩展和 448
- sp\_loginconfig 系统过程 450–451
- sp\_logininfo 系统过程 452–453
- sp\_logiosize 系统过程 454
- sp\_maplogin 系统过程 458
- sp\_metrics 系统过程 462–463
- sp\_modify\_resource\_limit 系统过程 464–466
- sp\_modify\_time\_range 系统过程 467–468
- sp\_modifystats 系统过程 470–472
- sp\_modifythreshold 系统过程 473–476
- sp\_monitor 系统过程 477–484
  - @@connections 和 483
- sp\_monitorconfig 系统过程 485–491
- sp\_object\_stats 系统过程 493–495
- sp\_passthru 系统过程 507–508
- sp\_passwordpolicy 510
- sp\_pciconfig 524
- sp\_pkeys 目录存储过程 691
- sp\_placeobject 系统过程 528–529
- sp\_plan\_dbccdb 系统过程 530–531
- sp\_poolconfig 系统过程 532–536
- sp\_post\_xoload 系统过程 537–538
  - 重建索引 537
- sp\_primarykey 系统过程 539
  - sp\_foreignkey 和 318
- sp\_processmail 系统过程 540–541
- sp\_procmode
  - 链式事务模式 543
- sp\_procmode 系统过程 542–543
- sp\_querysmobj 存储过程, 语法和用法 544
- sp\_recompile 系统过程 546–547
- sp\_remap 系统过程 550
- sp\_remoteoption 系统过程 551–552
- sp\_remoteoption 系统过程。
- sp\_remotesql 系统过程 553–554
- sp\_rename 系统过程 555–556
- sp\_rename\_qpgroup 系统过程 557
- sp\_renamedb 系统过程 115, 558–559
- sp\_reportstats 系统过程 560–561
- sp\_revokelogin 系统过程 562
- sp\_role 系统过程 563–564
- sp\_sendmsg 系统过程 569–570
- sp\_server\_info 目录存储过程 692–693
  - sp\_tables 和 702
- sp\_serveroption
  - external engine auto start 571
  - negotiated logins 572
  - server costt 571
  - server login 572
- sp\_serveroption 系统过程 571–574
- sp\_set\_qplan 系统过程 575
- sp\_setlangalias 系统过程 576
- sp\_setpglockpromote 系统过程 577–578
- sp\_setpsexex 系统过程 579
- sp\_setrowlockpromote 系统过程 581
- sp\_setsuspect\_granularity 系统过程 583–585
- sp\_setsuspect\_threshold 系统过程 586–587
- sp\_showcontrolinfo 系统过程 590
- sp\_showexeclass 系统过程 592
- sp\_showplan 系统过程 597
- sp\_showpsexex 系统过程 599
- sp\_spaceused 系统过程 613–615
- sp\_special\_columns 目录存储过程 694
- sp\_sproc\_columns 目录存储过程 696
  - 数据类型代码编号 681
- sp\_ssladmin 系统过程 616–619
- sp\_statistics 目录存储过程 698
- sp\_stored\_procedures 目录存储过程 700
  - sp\_server\_info 信息 693
- sp\_syntax 系统过程 620–621
- sp\_sysmon 系统过程 622–627
- sp\_tab\_suspectptn 628
- sp\_table\_privileges 目录存储过程 701
- sp\_tables 目录存储过程 702
  - sp\_server\_info 信息 693
- sp\_tempdb 系统过程 629–637
- sp\_thresholdaction 系统过程 639–640
  - 阈值过程 51, 474
- sp\_transactions 系统过程 642–647
- sp\_unbindcache 系统过程 648–650
- sp\_unbindcache\_all 系统过程 651
- sp\_unbinddefault 系统过程 652–653
- sp\_unbindexclass 系统过程 654
- sp\_unbindmsg 系统过程 656
- sp\_unbindrule 系统过程 657–658
- sp\_version 系统过程 659–661
- sp\_volchanged 系统过程 662–664
- sp\_webservices 系统过程 665–672

- sp\_who** 返回的 *cmd* 676
- sp\_who** 系统过程 673–676
  - 返回的列 675
- spid* 编号
  - sp\_who** 输出 675
- spt\_committab* 表 5
- spt\_datatype\_info* 表 681
- spt\_datatype\_info\_ext* 表 681
- spt\_monitor* 表 5
- spt\_server\_info* 表 681
- spt\_values* 表 5
- SQL 标准
  - SQL 模式匹配 681
  - 用户定义的数据类型和 55
- sybdiagdb* 数据库 352
- sybsyntax* 数据库 620
- sysystemprocs* 数据库
  - 权限和 2
- sysalternates* 表
  - sp\_dropalias** 和 242
  - sysusers* 表和 16
  - 别名 16
- syscomments* 表
  - 源文本 401
- sysconstraints* 表
  - sp\_bindmsg** 和 86
- sysdevices* 表 217, 363
- syskeys* 表
  - sp\_dropkey** 和 259
  - sp\_foreignkey** 和 318
  - sp\_primarykey** 和 539
- syslanguages* 表 381
  - sp\_droplanguage** 和 261
- syslkstats* 表 494
- syslogs* 表 448
  - 放置在单独设备上 448
- sysmessages* 表
  - 错误消息文本 321
- sysremotelogins* 表 41–43, 270
  - sp\_dropremotelogin** 和 266
- sysresourcelimits* 表
  - sp\_help\_resource\_limit** 和 335
  - 适用于登录会话的限制 11
- sys.servers* 表
  - sp\_addserver** 和 46
  - sp\_helpserver** 和 394
- sys.sessions*
  - 删除旧条目 324
- sys.statstats* 表
  - 将统计信息刷新到 311
- system* 段
  - 删除 268
  - 映射 45
- sys.time.ranges* 表
  - ID 号存储 14
  - 范围名称存储 9
- sys.types* 表 272
- sys.user.messages* 表
  - sp\_dropmessage** 和 263
  - 错误消息文本 321
- sys.users* 表
  - sysalternates* 表和 16
- table\_access** 审计选项 64
- tape** 选项, **sp\_addumpdevice** 58
- tempdb*
  - 限制每个会话使用的页数 9, 464
- tempdb* 数据库
  - auto identity** 数据库选项和 195
  - unique auto\_identity index** 数据库选项和 199
- tempdb\_space* 9, 464
- tempdbs*
  - sp\_helpdb** 和 360
- text* 数据类型
  - 存储的大小 613
- thresh\_hysteresis** 全局变量
  - 阈值放置和 ] 51
- timeouts** 选项, **sp\_serveroption** 572
- Transact-SQL
  - 保留字 111
- true | false** 子句
  - sp\_dboption** 187
  - sp\_remotoption** 551
- true** 选项, **sp\_changedbowner** 102
- trunc log on chkpt** 数据库选项 198
- truncate** 审计选项 64
- trusted** 选项, **sp\_remotoption** 551
- UDP 消息传送 569

**unbind** 审计选项 65  
**unique auto\_identity index** 数据库选项 199  
**update** 审计选项 65  
 us\_english 语言 33  
**use message confidentiality** 服务器选项 572  
**use message integrity** 服务器选项 572  
**view\_access** 审计选项 65  
**wash** 关键字, **sp\_poolconfig** 532  
**writetext** 命令  
     **select into/bulkcopy/pllsort** 数据库选项 197  
 XP Server 706  
     释放内存 320  
**xp\_cmdshell context** 配置参数 707  
**xp\_cmdshell** 系统扩展存储过程 707  
**xp\_deletemail** 系统扩展存储过程 710  
     **sp\_processmail** 和 541  
**xp\_enumgroups** 系统扩展存储过程 711  
**xp\_findnextmsg** 系统扩展存储过程 712  
     **sp\_processmail** 和 541  
**xp\_logevent** 系统扩展存储过程 713  
**xp\_readmail** 系统扩展存储过程 714  
     **sp\_processmail** 和 541  
**xp\_sendmail** 系统扩展存储过程 717  
     **sp\_processmail** 和 541  
**xp\_startmail** 系统扩展存储过程 720  
**xp\_stopmail** 系统扩展存储过程 721

## A

安全性  
     用户定义的 Web 服务 671

## B

帮助  
     **sp\_sysmon** 显示 622  
 帮助报告  
     编译对象的源文本 398  
     表 326  
     登录名 384  
     段 390  
     键 379  
     扩展存储过程 365

连接 377  
     另请参见信息 (服务器) 326  
 权限 385  
 数据库 358  
     数据库对象 326  
     数据库设备 362  
     数据类型 326  
 索引 371  
     系统过程 326–407  
     用户 406–407  
     语言, 替代 381  
     远程服务器 394  
     约束 354  
     转储设备 362  
     资源限制 334  
     组 369  
     阈值 405  
 系统过程  
     绑定  
         对象绑定到数据高速缓存 76–79  
         规则 88–89  
         解除绑定和 648–650, 652  
         缺省值 80–82  
         数据高速缓存 76–79  
         用户消息到约束 86–87  
     保护系统  
         组 31  
     保留字  
         作为标识符 108–119  
         目录存储过程和 680  
         系统过程 3  
     报告  
         **sp\_who** 673–676  
         计划组 337  
     报告 **dbccdb** 数据库中的  
         I/O 统计信息 733  
         分配统计信息 750  
         故障信息 733, 739  
         配置信息 727, 739, 744  
         所有细节 744  
         综合信息 744  
     本地别名, 语言 576



- 本地服务器 46
    - 另请参见 远程服务器
  - 服务器
    - 本地化, 更改语言名和文件 117
    - 比较计划 156, 158
    - 比较计划组 156
    - 编号
      - 另请参见 ID, 用户
    - ODBC 数据类型代码 681
    - 设备 363
    - 数据类型代码 681
  - 编译和 **sp\_recompile** 546
  - 标点符号
    - 用户定义的数据类型 54
    - 用引号引起来 3, 680
  - 标度, 数据类型
    - 用户定义的数据类型 54
  - 标识
    - 替代 16
  - 标识符
    - set quoted\_identifier on** 112, 117–118
    - sp\_checkreswords** 和 112
    - 保留字和 108–119
    - 重命名 113
    - 带引号 112
    - 分隔 112
  - 表
    - sp\_column\_privileges** 返回的列权限信息 683–684
    - sp\_placeobject** 空间分配 528–529
    - sp\_recompile** 546
    - sp\_table\_privileges** 信息 701
    - sp\_tables** 702
    - 绑定到数据高速缓存 76
    - 重命名 555–556
    - 对象依赖性和 209–214
    - 更改名称 114
    - 估计空间 295
    - 解除与数据高速缓存的绑定 648
    - 具有可疑索引 412
    - 连接的公用键 160–161
    - 列信息 685
    - 删除键 259
    - 删除行锁升级阈值 267
    - 设置行锁升级阈值 581
    - 使用 **sp\_checkreswords** 检查名称 111
    - 使用的空间 614
    - 锁, 类型 305, 443
    - 锁定对象 305, 443
    - 锁升级阈值 578
    - 系统过程 5, 681
    - 之间的公用键 160–161
    - 主键 539
  - 别名, 服务器 47
  - 别名, 用户
    - sysalternates** 表 16, 242
    - 分配与别名相比不同的名称 60
    - 另请参见 登录名
    - 删除 242–243, 273
    - 数据库所有权移交 102
    - 有关帮助 407
    - 指定 16–17
  - 用户
    - 别名, 语言
      - 定义 32
      - 指定 576
    - 并发优化 122
- ## C
- 参考信息
    - dbcc** 存储过程 723
    - 目录存储过程 679
    - 系统过程 1–5
    - 系统扩展存储过程 705
  - 参数, 过程
    - 提供方式 3, 680
  - 参照完整性约束
    - 重命名 555–556
    - 将用户消息绑定到 86
  - 操作
    - 为资源限制指定 10
    - 资源限制信息 335
    - 资源限制修改 465
    - 操作系统命令 707

- 操作系统命令的用户环境 (**xp\_cmdshell**) 707
- 层次
  - 另请参见 优先级
  - 角色, 使用 **sp\_activeroles** 显示 6
  - 数据高速缓存绑定 77
  - 锁升级阈值 578, 582
  - 用户定义的数据类型 56
- 查询
  - sp\_tables** 和 702
  - 编译和优化 546
- 查询处理, 使用 **sp\_add\_resource\_limit** 进行限制 9
- 查询计划, 使用 **sp\_recompile** 重新进行编译 546
- 查找
  - 保留字 108
  - 段 390
  - 对象信息 326
  - 对象依赖性 209, 213
  - 分区信息 342
  - 服务器名 394
  - 高速缓存绑定 90, 346
  - 配置参数 349
  - 权限 385
  - 设备 362
  - 数据库对象 332
  - 数据库设置 358
  - 数据库选项 187
  - 数据库中的用户 406
  - 数据类型 326
  - 语言 381
  - 阈值 405
  - 约束 354
  - 资源限制 334
  - 字符集 396
- 长度可变的字符数据, 设置格式 71–75
- 撤消源文本的加密 408
- 池, 内存
  - 配置 532
  - 缺省值 92
- 冲突, 散列键 338
- 重叠的时间范围 14
- 重建索引
  - sp\_post\_xoload** 537
- 重命名 555–556
  - 另请参见 **sp\_rename** 系统过程
  - 数据库 558–559
  - 相关警告 556, 559
- 重新编译
  - 存储过程 546
- 抽象计划
  - 用 **sp\_help\_qplan** 查看 339
  - 有关信息 339
- 抽象计划组
  - 重命名 557
  - 导出 300
  - 导入 410
  - 删除 245
  - 添加 8
- 触发器
  - sp\_recompile** 546
  - 重命名 555–556
  - 重命名的数据库和 559
  - 重新映射 550
  - 对象依赖性和 209–214
  - 更改名称 114
  - 使用 **sp\_checkreswords** 检查名称 111
  - 显示源文本 398
- 创建
  - dbccdb** 工作空间 728
  - 抽象计划组 8
  - 扩展存储过程 26–27
  - 时间范围 13
  - 数据类型 54–57
  - 限制 9
  - 用户别名 16–17
  - 用户定义的审计记录 63
  - 用户组 31
  - 执行类 24
  - 指定的时间范围 13
  - 资源限制 9
  - 阈值 50–53
- 磁带转储设备
  - 添加 58–59
- 磁盘镜像
  - sp\_who** 报告 676

## 磁盘设备

添加 58–59

从数据库取消段的映射 268–269

## 存储过程

for *dbccdb* 数据库 723

**sp\_checkreswords** 和 112

**sp\_cluster logical** 135

**sp\_dbcc\_alterws** 726

**sp\_dbcc\_configreport** 727

**sp\_dbcc\_createws** 728–729

**sp\_dbcc\_deleteadb** 730

**sp\_dbcc\_deletehistory** 731–732

**sp\_dbcc\_differentialreport** 733–734

**sp\_dbcc\_evaluatedb** 735–736

**sp\_dbcc\_faultreport** 739–743

**sp\_dbcc\_fullreport** 744

**sp\_dbcc\_runcheck** 749

**sp\_dbcc\_statisticsreport** 750–752

**sp\_dbcc\_summaryreport** 753–755

**sp\_dbcc\_updateconfig** 756–757

**sp\_recompile** 546

重命名 555–556

重命名的数据库和 559

重新映射 550

对象依赖性和 209–214

高速缓存绑定和 78, 649

另请参见 数据库对象 1

目录 679–702

用 **sp\_procxmode** 更改事务模式 542–543

## 系统过程

## 错误

数目 483

## D

## 大小

*image* 数据类型 613

日志设备 449

## 带引号的标识符

测试 112

使用 112, 117–118

## 代码

ODBC 数据类型 681

数据类型 688

单引号。请参见 引号

单用户模式 198

**sp\_renamedb** 和 558

当前使用情况统计信息 560–561

## 当前数据库

来自 **sp\_helpdb** 的信息 361

使用的空间 613–615

当前锁, **sp\_lock** 系统过程 441

导出计划组 300

导入抽象计划组 410

## 登录名

“probe” 560

*sysremotelogins* 表 41–43, 266, 270

别名 16, 242

更改当前数据库所有者 102

会计统计信息 129, 560

解锁 445

另请参见 远程登录名

删除 270

删除资源限制, 从 247

数目 483

锁定 445

修改资源限制 464

有关信息 225, 384

远程 266, 270

远程选项 551

资源限制信息 334

资源限制应用于 9

## 用户

## 第一页

分区, 用 **sp\_helppartition** 显示 342

日志设备 382

动态链接库 (DLL), 卸载 320

## 逗号 (,)

用户定义的数据类型 54

## 段

**sp\_helpthreshold** 报告 405

更改名称 115, 117

监控剩余空间 50–53, 473–476

扩展 45, 301

另请参见 数据库设备 528

删除 268–269

使用 **sp\_checkreswords** 检查名称 111

添加 44–45  
 映射 45  
 有关信息 390  
 空间分配  
 段,设备空间和 **sp\_placeobject** 528  
 对象  
   重新映射 550  
 对象。请参见 数据库对象 326  
 对象名,数据库  
   使用 **sp\_checknames** 检查 106  
   使用 **sp\_checkreswords** 检查 111  
 对象所有者。请参见 数据库对象所有者。  
 多字节字符集  
   **sp\_helpsort** 输出 397  
   排序顺序 397

## F

返回状态  
   **sp\_checkreswords** 111  
   目录存储过程 680  
   系统过程 1  
 范围,为资源限制指定 9  
 范围锁 444  
 非聚簇索引的 **max\_rows\_per\_page** 的计算公式 125  
 非空值  
   **sp\_addtype** 和 55  
   用户定义的数据 55  
 分布式事务管理 (DTM) 642  
 分隔标识符  
   测试 112  
   使用 112, 117–118  
 服务器  
   **sp\_server\_info** 信息 692  
   本地 46  
   监控活动 477  
   另请参见 进程 (服务器任务) 673  
   名称 46  
   删除 270  
   设置行锁升级阈值 581  
   升级和 **sp\_checknames** 106  
   升级和 **sp\_checkreswords** 111  
   属性名 692

添加 46–49  
 选项,使用 **sp\_serveroption** 更改 571–574  
 有关远程登录名的信息 384  
 远程 394  
 远程服务器  
   服务器别名 47  
   服务器进程 ID 号。请参见 进程 (服务器任务)  
   服务器信息选项。请参见 信息 (服务器)  
 复制  
   计划 173, 174  
   计划组 173

## G

高可用性  
   配置 Adaptive Server 162  
 高速缓存,数据  
   **logonly** 类型 97  
   恢复和 93  
   将对象绑定到 76  
   解除对象的绑定 648  
   解除所有对象的绑定 651  
   开销 96, 346  
   内存池 532–536  
   配置 90–98  
   删除 97  
   有关信息 93, 346  
   状态 95  
 格式,指定的时间范围中的时间 13  
 隔离级别  
   **identity in nonunique index** 数据库选项和 197  
   目录存储过程 680  
   系统过程 1  
 更改  
   **dbccdb** 工作空间大小 726  
   抽象计划组的名称 557  
   对象名 555–556  
   时间范围 467  
   数据高速缓存中的内存池 532  
   数据库所有者 102  
   数据库选项 187–200  
   用户组 103–104

语言别名 576  
 资源限制 464  
 阈值 473–476  
 更新行锁 444  
 工作进程系列  
   由 **sp\_lock** 报告的 *fid* 443  
   有关 *fid* 的 **sp\_familylock** 报告 305  
 工作空间  
   删除 729  
 公用键  
   *See also* 外键 160  
   报告 379–380  
   定义 160–161  
   连接候选和 377  
   删除 259  
 主键  
 共享锁 305, 443  
 共享行锁 444  
 故障, 介质和 **trunc log on chkpt** 数据库选项 198  
 故障隔离  
   索引级 314, 435  
 关键字  
   作为标识符 108  
 规则  
   绑定 88–89  
   重命名 555–556  
   重新映射 550  
   更改名称 114  
   解除绑定 657–658  
   命名用户创建的 88  
   使用 **sp\_checkreswords** 检查名称 111  
   系统表和 89  
   显示源文本 398  
   用于查找对象 213, 332  
 归类序列。请参见 排序顺序  
 过程 请参见 存储过程 1

## H

行, 表  
   使用 **sp\_spaceused** 计算数量 614  
   限制返回的数目 9  
 行锁 444

行锁升级阈值  
   **sp\_helpdb** 报告数据库设置 358  
   使用 **sp\_droprolockpromote** 删除 267  
   使用 **sp\_setrowlockpromote** 设置 581  
 恢复  
   列出可疑数据库 434  
   列出脱机页 436  
   设置模式 583  
   设置阈值 586  
   使用 **sp\_forceonline\_db** 强制可疑页联机 312  
   使用 **sp\_forceonline\_page** 强制可疑页联机 316  
   数据高速缓存和 93  
   显示模式 583  
 恢复故障隔离 314, 435  
 会计, 收费退回式  
   **sp\_clearstats** 129  
 获取当前的数据库信息 361  
 获取消息。请参见 **sp\_getmessage** 系统过程。

## J

计划  
   **sp\_showplan** 输出 597  
   比较 156, 158  
   查找 307  
   复制 173, 174  
   更改 575  
   删除 244, 246  
   搜索 307  
   修改 575  
 计划组  
   报告 337  
   比较 156  
   创建 8  
   导出 300  
   复制 173  
   复制到表 300  
   删除 245  
   删除所有的计划 244  
   添加 8  
   有关信息 337  
 计算机周期 483

## 索引

记录, 审计 18  
加密  
    编译对象源文本 408  
    撤消 408  
加密列  
    **sp\_helprotect** 388  
监控  
    剩余空间 50, 51, 474  
    锁争用 493  
    系统活动 477  
检查点进程  
    **trunc log on chkpt** 数据库选项 199  
检查口令。请参见 口令  
检查约束  
    重命名 555-556  
    将用户消息绑定到 86  
    显示源文本 398  
检索  
    错误消息文本 321  
键  
键, 表  
    syskeys 表 160, 318, 539  
    另请参见 公用键 373  
    删除 259  
    有关信息 379  
索引  
将来空间分配。请参见 空间分配 528  
角色  
    使用 **sp\_activeroles** 显示 6  
角色层次, 显示  
    使用 **sp\_activeroles** 6  
    使用 **sp\_displayroles** 232  
角色层次。请参见 角色层次  
较低级和较高级的角色。请参见 角色层次  
解除绑定  
    对象与高速缓存 648-650  
    缺省值 652-653  
    数据高速缓存 648-650  
解锁登录帐户 445  
进程 (服务器任务)  
    ID 编号 673  
    **sp\_showplan** 显示 597-598  
    **sp\_who** 报告 673-676  
    检查持有的锁 441  
    检查锁 305-306, 441-444

进程逻辑名。请参见 逻辑设备名  
精度, 数据类型  
    **sp\_help** 报告 329  
    用户定义的数据类型 54  
镜像。请参见 磁盘镜像  
聚簇索引, *indid* 不等于 1 373  
卷处理 662

## K

开放式数据库互连 (ODBC) API 数据类型 681  
开销, 数据高速缓存 96  
可疑分区, 跨平台转储和装载 537  
可疑数据库, 列出 434  
可疑索引  
    强制联机 314, 435  
可疑页  
    恢复时隔离 586-587  
    联机 312-313, 316-317  
    列表 436  
空白  
    目录存储过程参数值 680  
    在系统过程参数值中 3  
空间  
    **sp\_spaceused** 过程 613-615  
    估计表和索引大小 295-299  
    另请参见 大小  
    未使用 614  
    用 **sp\_modifythreshold** 监控剩余 473-476  
空间分配  
段  
空间分配  
    **sp\_placeobject** 过程 528-529  
    另请参见 数据库设备 528  
    日志设备 449  
    未来 528-529  
空间管理属性  
    使用 **sp\_chgattribute** 更改 122  
空值  
    **sp\_addtype** 和 54  
    用户定义的数据类型 54  
控制器, 设备  
    **sp\_helpdevice** 和数字 363

## 口令

**sp\_remoteyoption** 和 551

**sp\_serveroption** 和 574

可信登录名或检验 551

上次更改的日期 230

在网络上加密 574

## 口令安全性

在网络中保护登录口令 571

## 口令复杂程度

127

**sp\_extrapwdchecks** 304

**sp\_passwordpolicy** 510

跨平台转储和装载, 处理可疑分区 537

## 扩展

段 301

## 扩展存储过程

创建 26–27

删除 255

显示 365

## L

## 连接 160

**sp\_commonkey** 160

有关信息 377

链接用户。请参见别名, 用户

链式事务模式和 **sp\_procxmode** 543

## 两阶段提交

探查进程 560

## 列

**sp\_who** 返回的 675

公用键 160–161

规则 88–89

解除缺省值的绑定 652–653

连接和 377

权限 683

缺省值用于 80–82

使用 **sp\_unbindrule** 解除规则的绑定 657–658

数据类型 685

外键 318–319, 689

依赖性, 查找 114

主键 539

## 列表

设备 363

数据库选项 187

## 列出

**dbcc** 存储过程 723

目录存储过程 679

系统过程 1

列对。请参见连接 377

列对。请参见公用键 160

## 列名

更改 114, 555–556

使用 **sp\_checknames** 检查 106

列数据。参见“数据类型”

## 临时表

**sp\_help** 和 332

系统过程 5

临时名称。请参见别名, 用户

路径名, 转储设备 58

逻辑设备名 58, 217

**syslogs** 表 448

## M

每个引擎的描述符数 428

每个引擎的最大 **ldapua** 本机线程数 427

## 名称

DLL 文件 320

别名 16, 242, 273

分配不同的, 与别名相比 60

服务器 46

服务器属性 692

更改标识符 113

更改数据库对象 555–556

使用 **sp\_checknames** 检查 106

使用 **sp\_checkreswords** 检查 108

远程用户 266

## 命名

时间范围 13

用户定义的数据类型 56

组 31

模块, 显示语法 620

模式匹配和目录存储过程参数 681

目录存储过程 679–702  
**sp\_columns** 685–686  
**sp\_databases** 687  
**sp\_datatype\_info** 688  
**sp\_fkeys** 689–690  
**sp\_pkeys** 691  
**sp\_server\_info** 692–693  
**sp\_special\_columns** 694  
**sp\_sproc\_columns** 696  
**sp\_statistics** 698  
**sp\_stored\_procedures** 700  
**sp\_tables** 702  
 返回状态 680  
 列表 679  
 语法 680–681

## N

内存  
 映射 203  
 由配置参数使用 349  
 内存池  
**sp\_logiosize** 和 454, 458  
 配置 532  
 配置清洗百分比 535  
 配置异步预取限制 536  
 缺省值 92  
 事务日志和 535  
 最小大小 534

## P

排它锁 305, 443  
 排它行锁 444  
 排序顺序  
 更改, 和 **sp\_indsuspect** 系统过程 412  
 有关信息 396  
 配置参数  
 帮助信息 349  
 更改 166–172  
 显示级别 223  
 配置参数的“basic”显示级别 223

配置参数的“comprehensive”显示级别 223  
 配置参数的“intermediate”显示级别 223  
 批量复制。请参见 **bcp** (批量复制实用程序)

## Q

嵌套游标 177  
 强制执行资源限制 10  
 清除会计统计信息 129–130  
 清洗区  
 配置 535  
 缺省值 535  
 取消加密源文本 408  
 权限  
**sp\_column\_privileges** 683  
 撤销 385  
 授予 385  
 系统过程 2  
 显示用户的 225  
 新数据库所有者 102  
 有关信息 385  
 转储设备和 59  
 全局变量  
 另请参见各个变量名  
**sp\_monitor** 报告 481  
 全天时间范围 13  
 缺省设置  
 配置参数 169  
 缺省数据库  
 另请参见 *sysdevices* 表  
 缺省数据库设备  
**sp\_helpdevice** 和 363  
 使用 **sp\_diskdefault** 设置状态 217  
 缺省语言  
 添加 32–34  
 缺省值  
 绑定 80–82  
 重命名 114, 555–556  
 解除绑定 652–653  
 使用 **sp\_checkreswords** 检查名称 111  
 系统表和 81  
 显示源文本 398



## R

- 日期分量, 顺序 32
- 日志段 528
  - sp\_helplog** 报告 382
  - sp\_helpthreshold** 报告 405
- 日志记录
  - 消息 62
  - 用户定义的消息 36
- 日志设备信息 382

## S

- 散列键冲突 338
- 删除
  - 另请参见* 删除
  - dbccdb* 中的 **dbcc checkstorage** 历史记录 731
  - dbccdb* 中的目标数据库信息 730
  - 抽象计划组 245
  - 从数据库中删除段 268–269
  - 从数据库中删除用户 273
  - 从组中删除用户 103
  - 工作空间 729
  - 过程 255
  - 计划 244, 246
  - 具有别名的用户 242–243
  - 时间范围 250
  - 数据库设备 251
  - 锁升级阈值 257
  - 文件 251
  - 行锁升级阈值 267
  - 用户定义的数据类型 272
  - 用户定义的消息 263
  - 远程登录名 266, 270
  - 远程服务器 270
  - 资源限制 247
  - 组 258
- 删除。*请参见* 删除 242
- 设备
  - 另请参见* *sysdevices* 表
  - dsync** 设置 215
  - 更改名称 115, 117
  - 关于日志的信息 382
  - 设备片段, **sp\_helpdb** 报告 358
  - 设置**
    - 审计选项 63
  - 设置长度可变的字符数据的格式 71–75
  - 审计
    - 添加审计表 20
    - 选项, 显示 219
  - 审计选项
    - adhoc** 63
    - all** 63
    - alter** 63
    - bcp** 63
    - bind** 63
    - cmdtext** 63
    - create** 63
    - dbaccess** 63
    - dbcc** 63
    - delete** 63
    - disk** 63
    - drop** 63
    - dump** 63
    - errors** 63
    - exec\_procedure** 63
    - exec\_trigger** 63
    - func\_dbaccess** 63
    - func\_obj\_access** 63
    - grant** 64
    - insert** 64
    - load** 64
    - login** 64
    - login\_locked** 64
    - logout** 64
    - reference** 64
    - revoke** 64
    - rpc** 64
    - select** 64
    - setuser** 64
    - table\_access** 64
    - truncate** 64
    - unbind** 65
    - update** 65
    - view\_access** 65
  - 安全性** 64
  - 设置** 63
  - 审计追踪, 添加注释 18
  - 升级, 锁 577

## 时间范围

- ID 14
- 重叠 14
- 创建 13
- 更改活动时间范围 14
- 全天 13
- 删除 250
- “所有时间” 14, 250
- 添加 13
- 修改 467

## 时间间隔

- 估计索引创建 295
- 限制 9
- 自上次运行 **sp\_monitor** 481

## 实用程序命令

- 另请参见实用程序指南手册*
- 显示语法 620–621

## 使用情况统计信息 560

## 事务

- 模式 542–543

## 事务日志

- select into/bulkcopy/pllsort** 数据库选项 197
- trunc log on chkpt** 选项和 199
- 在单独的设备上 448–449
- 日志 I/O 大小和 535
- 数据高速缓存和 535
- 阈值和 271

## 视图

- 重命名 114, 555–556
- 重命名的数据库和 559
- 对象依赖性和 209–214
- 列 685
- 删除键 259
- 使用 **sp\_checkreswords** 检查名称 111
- 显示源文本 398
- 之间的公用键 160–161
- 主键 539

## 收费退回式会计

- sp\_clearstats** 过程 129–130
- sp\_reportstats** 过程 560–561

授权。*请参见* 权限。

## 受托模式

- 远程登录名和 42

## 输出, 数目

- 包 483

## 输入包, 数目 483

## 属性

- sp\_addobjectdef** 和 40
- 服务器 (**sp\_server\_info**) 692
- 执行类 24

## 数据高速缓存

- logonly** 类型 97
- status** 95
- 恢复和 93
- 将对象绑定到 76
- 解除对象的绑定 648
- 解除所有对象的绑定 651
- 开销 96, 346
- 内存池 532–536
- 配置 90–98
- 删除 97
- 有关信息 93, 346

## 数据库

- 另请参见 数据库对象*
- 绑定到数据高速缓存 76, 77
- 重命名 558–559
- 存储信息 613
- 关于使用的存储空间的信息 361, 613
- 解除与数据高速缓存的绑定 648
- 空间不足 639
- 列出可疑 434
- 列出可疑页 436
- 删除段 268–269
- 删除行锁升级阈值 267
- 删除用户 273
- 设置行锁升级阈值 581
- 使用 **sp\_checknames** 检查 106
- 使用 **sp\_databases** 列出 687
- 锁升级阈值 578
- 所有权 102
- 添加用户 60
- 添加组 31
- 用 **sp\_helpdb** 列出 358
- 有关帮助 358
- 阈值 639

- 数据库。
- 数据库对象
  - sp\_tables** 列表 702
  - 绑定到高速缓存 76
  - 查找 213, 332
  - 重命名 555–556
  - 将规则绑定到 88
  - 将缺省值绑定到 80–82
  - 列表 326
  - 权限 385
  - 使用的空间 613–615
  - 依赖性 209–214
- 数据库对象所有者和 **sp\_depends** 系统过程 209
- 数据库恢复顺序
  - sp\_dbrecovery\_order** 系统过程 201–202
  - 系统数据库和 202
- 数据库设备
  - defaulton** 或 **defaultoff** 状态 217–218
  - dsynch 设置 215
  - sp\_helpdevice** 系统过程 362
  - 列表 362
  - 删除 251
  - 删除段 268–269
  - 状态 217
- 数据库设计
  - 逻辑关系 160, 318
  - 删除键 259
- 权限
- 数据库所有者
  - dbo use only** 数据库选项 196
  - 更改 102
  - 另请参见 数据库对象所有者 102
  - 添加用户 60
  - 有关信息 406–407
  - 转交所有权 102
- 数据库文件。请参见 文件。
- 数据库选项 195–200
  - 另请参见 各选项名
  - 列表 187–200
  - 显示设置 191, 358
- 数据类型
  - ODBC 681
  - sp\_datatype\_info** 信息 688
  - sp\_help** 信息 333
  - 层次 56
  - 代码 681, 688
  - 解除缺省值的绑定 652–653
  - 缺省值和 80–82
  - 删除用户定义的 272
  - 使用 **sp\_unbindrule** 解除规则的绑定 657–658
  - 物理 54
  - 数据类型, 自定义。请参见 用户定义的数据类型
  - 数据类型优先级。请参见 优先级
  - 数据依赖性。参见 依赖性, 数据库对象
  - 数目 (数量)
    - open objects 175
  - sp\_countmetadada** 报告的数据库数 175
  - 每个用户的组数 103
  - 每个约束的消息数 86
  - 索引 175
- 数字
  - spid** (服务器进程 ID) 673
  - 全局变量单位 483
  - 消息 36, 62, 263, 321
  - 星期名称和 32
- 排序顺序
- 顺序
  - 另请参见 索引 373
  - 日期分量 32
- 损坏的数据库
  - 恢复故障隔离模式 583
  - 列表 434
- 损坏页
  - 恢复时隔离 586–587
  - 联机 316–317
  - 列表 436
- 索引
  - sp\_placeobject** 空间分配 528–529
  - sp\_statistics** 698
  - 绑定到数据高速缓存 76
  - 重命名 114, 555–556
  - 非唯一索引中的 IDENTITY 列 197
  - 估计空间和时间要求 295
  - 解除与数据高速缓存的绑定 648
  - 可疑 412
  - 使用 **sp\_checknames** 检查名称 106

使用 **sp\_checkreswords** 检查名称 111  
 使用的空间 614  
 顺序, 由 **sp\_helpindex** 报告 373  
 有关信息 371

索引页  
 锁 444

锁  
 “FAM DUR” 状态 306  
**sp\_familylock** 系统过程 305–306  
**sp\_lock** 系统过程 441–444  
 共享表 305, 443  
 共享页 305, 443  
 行 444  
 类型 305, 443  
 排它表 305  
 排它表和页 443  
 排它页 305  
 显示信息 441, 493  
 页 305, 443  
 意图表 305, 443  
 由 **sp\_lock** 报告 441

锁定  
 登录名 445  
 高速缓存绑定和 77  
 高速缓存解除绑定和 649  
 监控争用 493  
 控制 441–444

锁定方案, **sp\_help** 报告 332  
 锁升级阈值 577  
**sp\_help** 报告 332  
**sp\_helpdb** 报告数据库设置 358  
 使用 **sp\_droprowlockpromote** 删除行 267  
 使用 **sp\_setrowlockpromote** 设置行 581

所有权和转储设备 59  
 所有者。请参见 数据库对象所有者 102

## T

探查进程, 两阶段提交 560  
 替代标识。请参见 别名, 用户  
 替代语言。请参见 语言, 替代  
 替换用户定义的消息 36

## 添加

别名 16–17  
 抽象计划组 8  
 段 44–45  
 服务器 46–49  
 日期字符串 32–34  
 时间范围 13  
 外键 318–319  
 限制 9  
 向数据库添加组 31  
 消息到 *sysusermessages* 36–37  
 引擎到组 22  
 引擎组 22  
 用户定义的数据类型 54–57  
 用户添加到数据库 60–61  
 用户添加到组 60–61, 103–104  
 远程登录名 41–43  
 执行类 24  
 指定的时间范围 13  
 转储设备 58–59  
 资源限制 9  
 阈值 50–53

通配符  
 SQL 标准模式匹配 (**\$** 和 **\_**) 681

统计信息  
**sp\_clearstats** 过程 129  
**sp\_monitor** 477  
**sp\_reportstats** 560–561  
 刷新到 *sysabstats* 311  
 由全局变量返回 477

## W

### 外键

**sp\_fkeys** 信息 689  
**sp\_helpkey** 和 379  
 插入 318–319  
 删除 259

未使用的空间  
**sp\_spaceused** 报告 614

## 文本

- 使用 **defncopy** 复制 113
- 用户定义的消息 36

## 事务日志

## 文件

- 本地化 117
- 接口, 和服务器名 47
- 另请参见表 702
- 删除 251
- 执行 **sp\_dropdevice** 后无法访问 251

## 文件名

- DLL 320
- 配置文件 166

## 物理设备名 58

## 物理数据类型 54

## X

## 系统表

- 绑定到高速缓存 77
- 更新 1
- 规则和 89
- 空间分配 528
- 缺省值和 81
- 直接更新危险 115

## 系统过程

- sp\_jreconfig** 413
- sp\_pciconfig** 524
- 帮助报告 326–407
- 返回状态 1
- 更改名称 114
- 扩展存储过程 705–721
- 列表 1
- 目录存储 679–702
- 权限 2
- 使用 1
- 显示语法 620–621
- 显示源文本 398

## 系统过程表 5

- 目录存储过程和 681

## 系统过程结果。请参见 信息 (服务器)

## 系统角色

- 使用 **sp\_activeroles** 显示 6

## 系统扩展存储过程 705–721

- xp\_cmdshell** 707
- xp\_deletemail** 710
- xp\_enumgroups** 711
- xp\_findnextmsg** 712
- xp\_logevent** 713
- xp\_readmail** 714
- xp\_sendmail** 717
- xp\_startmail** 720
- xp\_stopmail** 721

## 显示

- 编译对象的源文本 398
- 模块的语法 620
- 数据库选项 187–200
- 字符集 396

## 限制的时间

- 时间范围修改 467
- 为时间范围指定 13
- 资源限制信息 334

## 限制的天数

- 时间范围修改 467
- 为时间范围指定 13
- 资源限制信息 334

## 限制类型 9

- I/O 开销 9
- tempdb\_space 9
- 返回的行数 9
- 经历的时间 9
- 修改值 464
- 指定值 9

## 消息

- sp\_getmessage** 过程 321
- sysusermessages** 表 36–37
- 编号 36, 62, 263, 321
- 日志记录 62
- 删除用户定义的 263
- 使用 **sp\_droplanguage** 删除系统 261
- 使用 **sp\_unbindmsg** 解除绑定 656
- 添加用户定义的 36–37
- 系统过程 4
- 语言设置 263, 321

消息 **output** 参数, **sp\_getmessage** 321

## 索引

- 小括号 ()
    - 用户定义的数据类型 54
  - 协同服务器
    - 配置 162–164
  - 信息 (服务器)
    - 当前的锁 441
    - 可疑索引 412
    - 索引 371
  - 星期日期值
    - 第一 32
    - 名称和编号 32
  - 性能
    - 并发优化 122
    - 有关信息 622
  - 修改
    - 配置参数 166
    - 配置参数显示级别 223
    - 指定的时间范围 467
    - 资源限制 464
    - 阈值 473
  - 修改抽象计划 575
  - 虚拟页号 363
  - 选项
    - 另请参见 配置参数
    - 远程登录名 551–552
    - 远程服务器 571–574
- ## Y
- 页, 数据
    - 使用 **sp\_spaceused** 计算数量 614
    - 锁定对象 305, 443
  - 页锁, 类型 305, 443
  - 依赖性, 数据库对象
    - sp\_depends** 系统过程 209–214
    - 重新编译和 556
    - 更改名称 113
  - 移动
    - 表 528
    - 事务日志 448
    - 索引 528
    - 用户到新组 103
  - 已编译对象
    - 检查源文本是否存在 120
    - 显示源文本 398
    - 隐藏源文本 408
  - 意图表锁 305, 443
  - 异步日志服务 (ALS) 358
    - sp\_helpdb** 中的输出 360
  - 异步预取
    - 配置限制 536
  - 引号 ( “ ” )
    - 单, 和 **quoted\_identifier** 118
    - 将保留字引起来 112
    - 将参数值括起来 3, 680
  - 引用, 对象。参见 依赖性, 数据库对象
  - 应用程序
    - 删除资源限制, 从 247
    - 修改资源限制 464
    - 资源限制信息 334
    - 资源限制应用于 9
  - 映射
    - 数据库 203
    - 远程用户 41
  - 用户
    - sp\_who** 报告 673–676
    - sysusers** 表 16
    - 从数据库中删除 273
    - 更改名称 117
    - 更改组 103–104
    - 会计统计信息 129, 560
    - 权限 385
    - 删除具有别名的 242–243
    - 删除远程 270
    - 添加 60–61
    - 系统过程权限和 2
    - 信息 225
    - 有关信息 406
    - 远程 384
  - 用户 ID
    - 显示 230
    - 用 **sp\_import\_qpgroup** 更改 410
  - 用户创建的对象。参见 数据库对象
  - 用户的标识。参见 别名

- 用户定义的 Web 服务
  - 安全性 671
- 用户定义的错误消息中的格式化学字符串 37
- 用户定义的过程
  - 使用 **sp\_addextendedproc** 创建 ESP 26
- 用户定义的审计记录 63
- 用户定义的数据类型
  - 层次 56
  - 创建 54–57
  - 更改名称 114
  - 将规则绑定到 88
  - 将缺省值绑定到 80–82
  - 解除缺省值的绑定 652–653
  - 命名 56
  - 删除 272
  - 使用 **sp\_checkreswords** 检查名称 111
  - 使用 **sp\_unbindrule** 解除规则的绑定 657–658
- 用户定义的消息 36–37
  - 使用 **sp\_unbindmsg** 解除绑定 656
- 用户定义角色
  - 使用 **sp\_activeroles** 显示 6
- 登录名
- 用户名
  - 更改 115
  - 另请参见 数据库对象所有者 384
  - 使用 **sp\_checkreswords** 检查 111
- 用户权限。请参见 数据库所有者 196
- 用户数据报协议消息传送 569
- 用于代替省略的名称元素的圆点 (..) 38
- 优化
  - 查询 (**sp\_recompile**) 546
- 优化索引锁定 122, 123, 124, 326, 330
- 优先级 373
  - sp\_setpsex** 579
  - 规则绑定 89
  - 将缺省值绑定到列和数据类型 81
  - 资源限制 11
- 邮件, 服务器
  - 查阅 714
  - 处理 540–541
  - 发送 717
  - 启动会话 720
  - 删除 710
  - 停止会话 721
- 有关信息, 报告
  - 编译对象的源文本 398
  - 当前的锁 441
  - 登录名 673
  - 段 390
  - 服务器进程 673
  - 服务器用户 225, 673
  - 高速缓存绑定 78
  - 键 379
  - 可疑索引 412
  - 空间使用 613
  - 扩展存储过程 365
  - 连接列 377
  - 权限 385
  - 日志的第一页 382
  - 事务日志设备 382
  - 数据高速缓存 93
  - 数据库 358
  - 数据库对象 326
  - 数据库设备 362
  - 数据库所有者 406
  - 数据类型 326
  - 索引 371
  - 锁 441, 493
  - 统计信息, 监控 477
  - 性能 622
  - 用户, 数据库 406
  - 语言 381
  - 远程服务器 394
  - 远程服务器登录名 384
  - 转储设备 363
  - 资源限制 334
  - 组 369, 406
  - 阈值 405
- 语法
  - 检查是否存在保留字 111
  - 目录存储过程 680–681
- 语言, 替代
  - syslanguages** 表 381
  - 安装 32
  - 别名 576
  - 不带语言模块 32

- 更改名称 115, 117
  - 日期格式 32
  - 删除 261
  - 删除消息 263
  - 使用 **sp\_checkreswords** 检查 111
  - 系统消息和 321
  - 用户定义的消息 36
  - 有关信息 381
  - 正式名称 576
  - 预取
    - 禁用 99
    - 启用 99
  - 阈值
    - 达到 50
    - 更改 473–476
    - 降低 I/O 的优化 122
    - 禁用 52, 271, 475
    - 删除 271
    - 添加 50–53
    - 停滞值 51, 474
    - 行锁升级 581
    - 有关信息 405
    - 之间的空间 51
    - 最大数目 51, 474
    - 最后机会 51, 271, 474
  - 阈值过程 51
    - 参数传递给 51, 475
    - 创建 639
    - 执行 52, 475
  - 源文本
    - 加密, 撤销 408
    - 检查是否存在 120
    - 显示 398
    - 隐藏 408
  - 用户
    - 远程登录名
      - sp\_remotoption** 551–552
      - sysremotelogins* 表 41–43
      - 另请参见 登录名
      - 删除 266
      - 受托模式或非受托模式 551
      - 有关信息 384
    - 远程服务器
      - 另请参见 服务器
      - sp\_remotoption** 和 551–552
      - 更改名称 115, 117
      - 名称 46
      - 删除登录名 266
      - 有关登录名的信息 384
      - 有关信息 394
    - 远程用户。请参见 远程登录名
    - 约束
      - 重命名 555–556
      - 将用户消息绑定到 86
      - 使用 **sp\_unbindmsg** 解除消息绑定 656
      - 显示源文本 398
      - 有关信息 342, 354
    - 月份值
      - 替代语言 32
- ## Z
- 在网络中保护登录口令 571
  - 占位符
    - 错误消息百分比符号 (%) 37
  - 帐户。请参见 登录名。
  - 争用, 锁
    - 用 **sp\_object\_stats** 监控 493–495
  - 语言, 替代
    - 正式的语言名称 33, 576
    - 另请参见 别名 33
  - 直接更新
    - 系统表 115
  - 直通模式
    - sp\_autoconnect** 系统过程 69
    - sp\_passthru** 系统过程 507
    - sp\_remotesql** 系统过程 553
  - 执行操作系统命令 707
  - 值
    - 使用 **sp\_server\_info** 显示 692
  - 指定的时间范围
    - ID 14
    - 重叠 14
    - 创建 13
    - 更改活动时间范围 14



- 全天 13
- 删除 250
- “所有时间” 14, 250
- 添加 13
- 修改 467
- 指定的时间范围从星期几开始 13
- 指定的时间范围到星期几结束 13
- 指定的时间范围的结束时间 13
- 指定的时间范围的开始时间 13
- 终端
  - 7 位, **sp\_helpsort** 输出示例 396
  - 8 位, **sp\_helpsort** 输出示例 396
- 主键
  - sp\_dropkey** 过程 259
  - sp\_foreignkey** 和 318
  - sp\_helpkey** 和 379
  - sp\_primarykey** 定义 539
- 注释
  - 增加到审计追踪中 18
- 转储, 数据库
  - 已中断 203
- 日志设备
- 转储设备
  - 列表 363
  - 另请参见 数据库设备 382
  - 权限和所有权问题 59
  - 删除 251
  - 添加 58–59
- 转储数据库 274
- 转换
  - 用户定义的消息 37
- 状态
  - 数据库设备 217
- 资源管理器 466
- 资源限制
  - 创建 9
  - 类型 9
  - 删除 247
  - 修改 464
  - 修改示例 465
  - 有关信息 334
- 资源限制的作用域
  - 对活动时间范围的更改和 14
  - 有关信息 334
  - 指定 10
- 自定义审计记录 63
- 自定义数据类型。请参见 用户定义的数据类型 字符集
  - sp\_helpsort** 显示 396
  - 多字节 397
  - 更改名称 115, 117
  - 使用 **sp\_checknames** 检查 105
  - 使用 **sp\_checkreswords** 检查 111
- 字符集的二进制排序顺序 397
- 阻塞进程
  - sp\_lock** 报告 305, 443
  - sp\_who** 报告 676
- 组
  - 另请参见 “public” 组
  - sp\_addgroup** 31
  - sp\_adduser** 过程 60
  - Windows NT 域 711
  - 更改 103–104
  - 删除 258
  - 有关信息 369
- 最后机会阈值 51, 474

