



集群用户指南

**Adaptive Server<sup>®</sup> Enterprise**

15.7

文档 ID: DC00986-01-1570-01

最后修订日期: 2012 年 2 月

版权所有 © 2012 by Sybase, Inc. 保留所有权利。

本出版物适用于 Sybase 软件及所有后续版本, 除非在新版本或技术说明中另有说明。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。

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, 本书的任何部分不得以任何形式、任何手段(电子的、机械的、手动、光学的或其它手段)进行复制、传播或翻译。

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 和文中列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

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

Java 和所有基于 Java 的标记都是 Oracle 和/或其分公司在美国和其它国家/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

<b>第 1 部分</b>	<b>配置 CLUSTER EDITION .....</b>	<b>1</b>
<b>第 1 章</b>	<b>Cluster Edition 概述 .....</b>	<b>3</b>
	Cluster Edition 的定义 .....	3
	Adaptive Server 集成集群件 .....	6
	集群协调器 .....	7
	仲裁设备 .....	8
	Cluster Edition 中的数据库设备 .....	8
	专用安装和共享安装 .....	10
	Cluster Edition 中的 Backup Server .....	11
	Cluster Edition 如何增强非集群版本 .....	11
	在集群中使用互连网络 .....	12
	监控实例之间的链路 .....	12
	建议的配置方案 .....	14
	OLTP 应用程序的 HA 故障切换 .....	15
	DSS 报告应用程序的水平可扩展性 .....	16
	OLTP 应用程序的水平可扩展性 .....	16
	Cluster Edition 中的新增客户端技术 .....	17
	复制支持 .....	17
<b>第 2 章</b>	<b>客户端应用程序和客户端/服务器交互 .....</b>	<b>19</b>
	Open Client .....	20
	在 Client-Library 应用程序中启用故障切换 .....	21
	客户端/服务器交互 .....	22
	登录重定向 .....	22
	连接迁移 .....	24
	上下文迁移 .....	26
	扩展的高可用性故障切换 .....	28
	在集群环境中使用 isql .....	29

	在集群环境中使用远程过程调用 .....	29
	远程服务器在其中是集群的 RPC .....	29
	本地服务器在其中是集群的 RPC .....	30
	本地和远程服务器在其中都是同一集群中的实例的 RPC .....	30
	sp_serveroption .....	30
	当节点断电时重新连接客户端 .....	31
<b>第 3 章</b>	<b>在集群环境中使用安全性 .....</b>	<b>35</b>
	在集群环境中使用加密列 .....	35
	在集群环境中使用 SSL .....	35
	使用 sp_listener 指定公用名 .....	37
	将 LDAP 用作目录服务 .....	37
	LDAP 目录服务与 Sybase interfaces 文件 .....	38
	libtcl*.cfg 文件 .....	41
	启用 LDAP 目录服务 .....	41
	向目录服务添加服务器 .....	43
	多目录服务 .....	44
	加密口令 .....	45
	性能 .....	45
	从 interfaces 文件迁移到 LDAP .....	46
	将 LDAP 目录服务与共享磁盘集群一同使用 .....	47
<b>第 4 章</b>	<b>在集群环境中使用监控表 .....</b>	<b>49</b>
	集群的监控表 .....	49
	配置系统视图 .....	50
	配置监控表 .....	50
	管理消息管道 .....	51
	RPC 的更改 .....	51
	特定于 Cluster Edition 的监控表 .....	52
	对所有实例返回相同信息的监控表 .....	52
	返回特定实例信息的监控表 .....	53
<b>第 5 章</b>	<b>在集群环境中使用 Backup Server .....</b>	<b>55</b>
	转储期间加入集群的节点 .....	55
	多个 Backup Server .....	56
	将 Cluster Edition 配置为使用多个 Backup Server .....	57
	启动和停止 Backup Server .....	59
	备份到介质 .....	59
	对存储过程的更改 .....	59

<b>第 6 章</b>	<b>管理负载</b>	<b>61</b>
	逻辑集群资源	62
	系统逻辑集群	62
	设置逻辑集群	63
	创建逻辑集群	64
	向逻辑集群中添加实例	66
	向逻辑集群中添加路由	66
	启动逻辑集群	67
	指派路由规则	68
	路由规则	68
	配置逻辑集群特性	69
	开放式逻辑集群	70
	向下路由模式	70
	System-view 特性	71
	启动模式	72
	故障切换模式	72
	Fail_to_any 特性	73
	负载配置文件特性	73
	登录分配模式	73
	操作释放	74
	收集模式	74
	角色	75
	配置故障切换	75
	添加故障切换资源	76
	管理逻辑集群	76
	用户任务和逻辑集群	77
	管理负载管理器线程	77
	查看有关逻辑集群的信息	78
	创建和删除逻辑集群	80
	向逻辑集群中添加资源	80
	从逻辑集群中删除资源	80
	添加、移动和删除路由	81
	迁移连接	81
	管理故障切换、故障恢复和计划的停机时间	82
	集群和实例状态	82
	状态如何更改	83
	异步命令和逻辑集群状态	85
	使用操作描述符	85
	示例：安排和重新安排故障切换	86
	使用 failover、failback、online、offline 和 deactivate	87
	分配负载	88
	负载指标	89
	创建用户指标	90
	对负载指标进行加权处理	90

	负载阈值 .....	90
	负载配置文件 .....	91
	使用样本负载配置文件 .....	92
	创建和配置您自己的负载配置文件 .....	92
	创建负载配置文件 .....	93
	建立负载配置文件 .....	93
	将负载配置文件与逻辑集群关联 .....	94
	更改负载配置文件 .....	95
	故障排除 .....	95
<b>第 7 章</b>	<b>集群高速缓存配置 .....</b>	<b>97</b>
	全局高速缓存 .....	97
	本地高速缓存 .....	98
	创建和配置命名数据高速缓存 .....	98
	获取有关命名高速缓存的信息 .....	98
	创建新高速缓存 .....	99
	配置和使用多个缓冲池 .....	109
	sp_poolconfig .....	109
	在缓冲池之间移动内存 .....	110
	更改池的清洗大小 .....	111
	更改池的本地异步预取百分比 .....	112
	删除缓冲池 .....	112
	将对象绑定到命名高速缓存 .....	113
	绑定对象的语法 .....	114
	获取有关绑定的高速缓存的信息 .....	114
	删除高速缓存绑定 .....	115
	修改配置文件 .....	115
	本地命名高速缓存的格式 .....	116
	本地高速缓存条目中的额外行 .....	117
	具有全局配置的已删除命名高速缓存 .....	117
	具有本地配置的命名高速缓存 .....	118
	具有有效配置的已删除条目 .....	118
	在存在全局配置的情况下创建本地配置 .....	119
	限制 .....	120
<b>第 8 章</b>	<b>使用临时数据库 .....</b>	<b>121</b>
	临时数据库的类型 .....	122
	本地临时数据库 .....	122
	摘要信息 .....	124
	创建临时数据库 .....	125
	创建本地系统临时数据库 .....	125
	创建本地和全局用户临时数据库 .....	126

	将用户和应用程序绑定到临时数据库 .....	126
	创建和管理临时数据库组 .....	127
	可以绑定的内容 .....	127
	如何确定会话绑定 .....	128
	创建和管理绑定 .....	129
	显示组和绑定信息 .....	130
	删除临时数据库 .....	130
	临时数据库的限制 .....	131
	本地数据库的私有设备支持 .....	132
	为临时数据使用私有设备 .....	133
	使用 disk init 创建私有设备 .....	133
	使用 disk reinit 重新初始化私有设备 .....	133
	使用 sp_dropdevice 删除私有设备 .....	134
	使用 sp_helpdevice 显示私有设备信息 .....	134
	对私有设备使用 create database 和 alter database .....	137
	使用 disk refit .....	137
<b>第 9 章</b>	<b>在群集环境中运行 Job Scheduler .....</b>	<b>139</b>
	安装和配置 Job Scheduler .....	139
	在群集环境中运行 Job Scheduler .....	139
	关闭 Job Scheduler .....	140
	重定向预定作业 .....	140
<b>第 10 章</b>	<b>实例恢复 .....</b>	<b>143</b>
	单一实例恢复 .....	143
	单一事务日志 .....	144
	多个同时发生的故障 .....	144
	启用多个同时进行的故障切换 .....	144
	恢复算法 .....	146
<b>第 11 章</b>	<b>附加主题 .....</b>	<b>147</b>
	锁 .....	148
	死锁数 .....	148
	保留锁 .....	148
	集群锁请求和任务请求状态 .....	149
	内存 .....	149
	阈值 .....	150
	dbcc thresholds 输出 .....	150
	dbcc dbtable 输出 .....	150
	带有 remap 选项的 dbcc dbrepair .....	150
	带有 newthreshold 选项的 dbcc dbrepair .....	151
	集群进程间通信 .....	152
	分布式检查点 .....	152

仲裁设备心跳 .....	152
配置仲裁设备心跳 .....	153
使用 InfiniBand .....	153
设置缓冲区空间 .....	154
在集群中配置 InfiniBand .....	154
专用安装模式 .....	155
维护服务器配置文件 .....	155
在集群环境中使用 Java .....	157
向存档数据库添加空间 .....	158
共享磁盘集群中的分布式事务 .....	158
在共享磁盘集群中使用 DTM .....	159
集群作为资源管理器 .....	159
在非所有者实例上处理请求 .....	160
处理实例故障 .....	160
使用 ASTC 进行事务协调 .....	161
连接迁移的影响 .....	161
配置和系统问题 .....	162
支持 mount 和 unmount 命令 .....	164
使用 sp_showplan .....	164

## 第 12 章

<b>将 Cluster Edition 与 Veritas Cluster Server 结合使用 .....</b>	<b>165</b>
支持的平台、要求和限制 .....	168
在 VCS 上安装和配置 Cluster Edition .....	169
安装 Cluster Edition .....	170
为 Storage Foundation 集成创建新的 Adaptive Server 集群 .....	170
将现有集群转换为使用 SF for Sybase CE .....	172
管理 VCS 控制下的集群 .....	173
启动和停止实例 .....	173
添加和删除实例 .....	174
增大用户连接数 .....	174
更改字符集或排序顺序 .....	174
成员资格模式 .....	175
确定成员资格模式 .....	175
更改成员资格模式 .....	176
了解故障情况 .....	177
VCS 故障排除 .....	178
Cluster Edition 启动失败 .....	178
Veritas 日志: “Sybase home directory does not exist” .....	179
实例日志: “failed to bind to socket” .....	180
实例日志: “Membership service failed to receive initial message from Veritas cluster membership after 301 seconds. Exiting...” .....	180

实例日志: “Failed to open quorum device 'device_path'. OS error 13, 'Permission denied'” .....	181
实例日志: “basis_dsizecheck: attempt to open device 'device_path' failed, system error is: Permission denied” .....	181
实例日志: “The configuration area in master device appears to be corrupt.” .....	182
Veritas 日志: “Path not found” .....	183
VCS 在实例启动之后关闭实例并引发资源故障 .....	183
VCS 无法关闭实例资源 .....	185
VCS 组的资源故障 .....	185
VCS 启动失败 .....	186
<b>第 13 章</b>	
<b>故障排除 .....</b>	<b>189</b>
验证集群环境 .....	190
使用来自早期版本的 dataserver 二进制文件重新启动集群 .....	191
访问磁盘设备时出错 .....	192
验证集群是否已关闭 .....	193
使用 sybcluster 创建集群失败, 并出现错误 -131 .....	193
集群创建失败, 文件留在 \$SYBASE 目录中 .....	194
Unified Agent 已启动, 但 sybcluster connect 失败 .....	194
磁盘设备被占用 .....	195
实例无法加入集群 .....	195
私有互连失败 .....	195
客户端连接故障切换失败 .....	196
客户端无法重新连接到备用高可用性服务器 .....	196
如果所有连接都使用 SSL, 则 sybcluster 无法连接 .....	197
jConnect 样本禁用 HA .....	197
PC 客户端安装 — java.lang.NoClassDefFound 错误 .....	198
集群条目 “name” 不包含任何服务器 .....	198
更改口令后, sybcluster 无法管理集群 .....	199
“找不到”代理 .....	200
Sybase Central 无法注册 AMCP 插件 .....	200
UAF 插件注册错误 .....	201
磁盘上的数据不可用: 影响数据库创建的问题 .....	202
启用 I/O 防护后, 设备访问权限被拒绝 .....	202
sybcluster 无法找到 interfaces 文件 .....	202
IBM 错误 .....	203
未启用异步 I/O .....	203
设备权限不正确 .....	204
正在使用设备的另一计算机 .....	204
运行 chdev 时出错 .....	205

<b>第 14 章</b>	<b>使用 Adaptive Server 插件管理集群 .....</b>	<b>207</b>
	管理共享磁盘集群 .....	207
	连接到集群 .....	208
	使用工具栏断开与集群的连接 .....	209
	在 Adaptive Server 插件中启用 Unified Agent 功能 .....	209
	更改服务器发现设置 .....	209
	显示集群属性 .....	211
	启动集群 .....	215
	关闭集群 .....	215
	删除集群 .....	216
	删除服务器组 .....	216
	显示集群的状态 .....	216
	管理集群实例 .....	217
	创建共享数据库设备 .....	218
	管理多个临时数据库 .....	219
	管理本地临时数据库 .....	219
	系统临时数据库 .....	220
	添加用户创建的全局临时数据库 .....	221
	添加用户创建的本地临时数据库 .....	221
	向组中添加临时数据库 .....	221
	管理负载 .....	223
	负载配置文件 .....	223
	管理逻辑集群 .....	228
	逻辑集群属性 .....	229
	查看负载状态 .....	233
	管理路由 .....	236
	路由属性 .....	236
<b>第 15 章</b>	<b>用 sybcluster 管理集群 .....</b>	<b>237</b>
	使用 sybcluster .....	238
	sybcluster 和 Unified Agent Framework .....	240
	启动 sybcluster .....	240
	创建集群 .....	240
	连接到集群 .....	241
	鉴定用户 .....	241
	设置用户名和口令 .....	241
	标识 Unified Agent .....	242
	启动集群 .....	244
	管理集群 .....	244
	创建集群 .....	244
	检验集群 .....	245
	显示有关可用 Unified Agent 的信息 .....	245
	显示集群信息 .....	246
	更改集群配置值 .....	247



<b>第 17 章</b>	<b>自定义 Cluster Edition 的本地化 .....</b>	<b>271</b>
	本地化支持概述 .....	271
	语言模块 .....	272
	服务器的缺省字符集 .....	272
	支持的字符集 .....	273
	字符集转换 .....	277
	服务器与客户端之间的转换 .....	278
	排序顺序 .....	278
	可用的排序顺序 .....	279
	语言模块 .....	282
	安装新的语言模块 .....	282
	消息语言 .....	282
	本地化 .....	283
	本地化目录 .....	283
	关于目录 .....	283
	关于 charsets 目录 .....	284
	关于 locales.dat 文件 .....	284
	更改本地化配置 .....	286
	Cluster Edition 本地化 .....	286
	Backup Server 本地化 .....	287
	排序顺序 .....	288
	字符集 .....	290
	charset 实用程序 .....	291
<b>第 18 章</b>	<b>为 Cluster Edition 增加可选功能 .....</b>	<b>293</b>
	添加审计 .....	293
	审计系统设备和数据库 .....	293
	在 Cluster Edition 中运行 auditinit .....	294
	审计设备的预安装任务 .....	294
	安装审计 .....	295
	安装 Transact-SQL 语法的联机帮助 .....	301
	联机语法帮助: sp_syntax .....	301
	sybsyntax 数据库的缺省设备 .....	302
	安装 sybsyntax .....	302
<b>第 19 章</b>	<b>记录错误消息与事件 .....</b>	<b>305</b>
	Cluster Edition 错误记录 .....	305
	启用和禁用错误记录 .....	306
	设置错误日志的路径 .....	306
	设置 Cluster Edition 错误日志路径 .....	306
	管理消息 .....	307
	记录用户定义的消息 .....	307
	记录审计事件 .....	308

<b>第 20 章</b>	<b>设置网络通信</b> .....	<b>309</b>
	Cluster Edition 如何确定要使用的目录服务条目 .....	310
	客户端如何使用目录服务 .....	311
	创建目录服务条目 .....	311
	支持的目录驱动程序 .....	311
	interfaces 文件的内容 .....	312
	异构环境和同构环境 .....	313
	了解 interfaces 文件的格式 .....	314
	interfaces 文件条目的组成部分 .....	315
	创建主 interfaces 文件 .....	316
	使用 dsedit 或 dscpp 创建主 interfaces 文件 .....	317
	使用文本编辑器创建主 interfaces 文件 .....	317
	配置多个网络的 interfaces 文件 .....	318
	配置多个网络处理程序的服务器 .....	318
	配置客户端连接 .....	319
	配置查询端口备份 .....	320
	IPv6 支持 .....	321
	了解 IPv6 .....	321
	IPv6 基础结构 .....	322
	启动可识别 IPv6 的 Cluster Edition .....	323
	故障排除 .....	324
	服务器启动失败 .....	324
	执行 ESP 时出错 .....	325
<b>索引</b> .....		<b>327</b>



第 1 部分

# 配置 Cluster Edition

本部分讲述配置 Adaptive Server<sup>®</sup> 使其在集群环境中运行的过程。

---

主题	页码
<a href="#">Cluster Edition 的定义</a>	3
<a href="#">Cluster Edition 如何增强非集群版本</a>	11
<a href="#">在集群中使用互连网络</a>	12
<a href="#">建议的配置方案</a>	14
<a href="#">Cluster Edition 中的新增客户端技术</a>	17
<a href="#">复制支持</a>	17

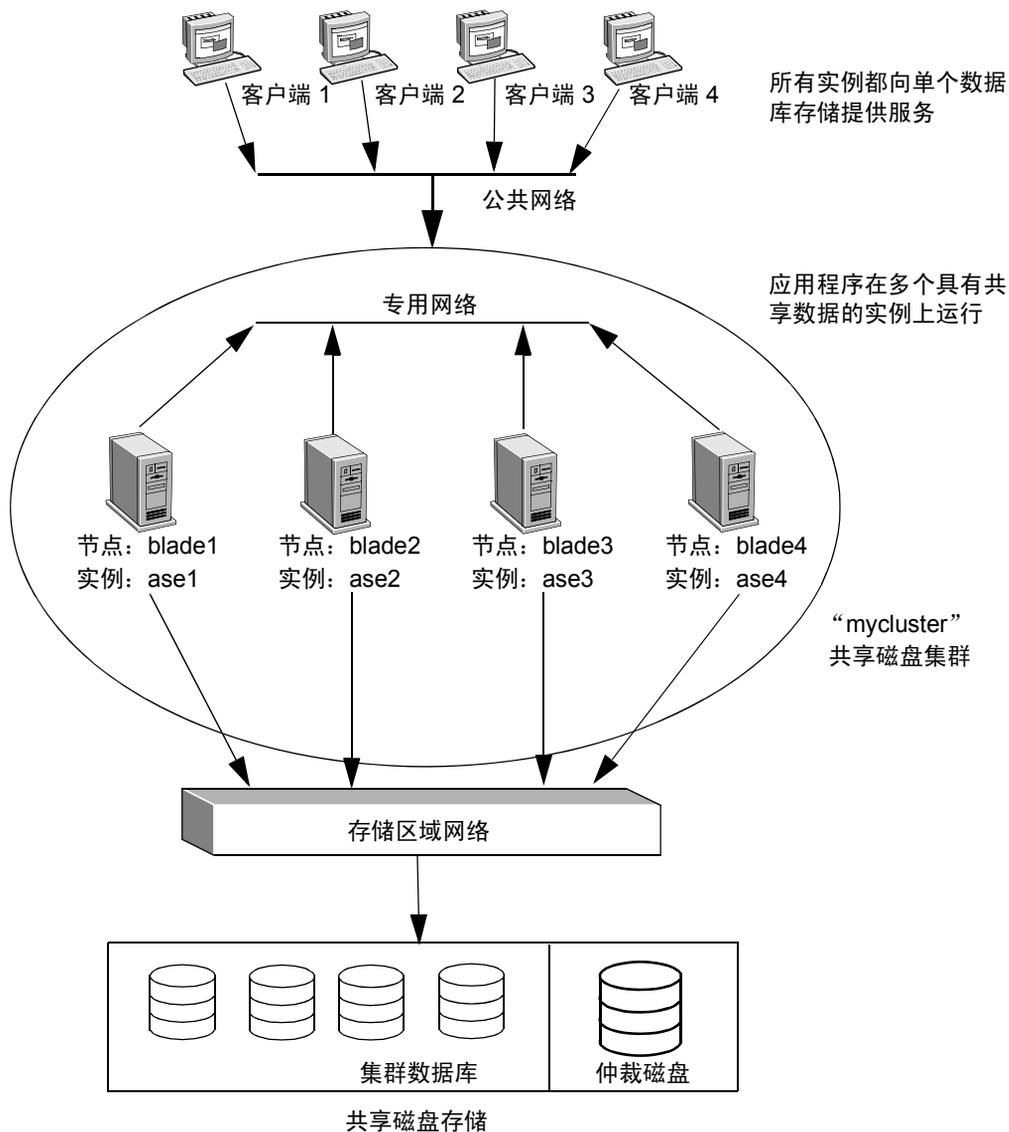
## Cluster Edition 的定义

Cluster Edition 能让您配置多个 Adaptive Server 作为一个共享磁盘集群运行。多台计算机与一组共享磁盘和高速专用互连（例如，千兆位以太网）连接，能让 Adaptive Server 使用多个物理和逻辑主机进行“扩展”。

在集群环境中，每台计算机都称为一个**节点**，每个 Adaptive Server 都称为一个**实例**。实例连接在一起会形成一个**集群**，它们彼此协作来管理共享磁盘上的一组数据库。在所有情况下，这些实例都表现为单个系统，所有数据都可从任何**实例**中访问到。Cluster Edition 分配对集群来说唯一的 SPID，因此 SPID 可在集群的所有实例当中标识单个进程。

在图 1-1 所示的集群系统中，客户端与名为“mycluster”的共享磁盘集群连接，该集群中包含分别在计算机“blade1”、“blade2”、“blade3”和“blade4”上运行的“ase1”、“ase2”、“ase3”和“ase4”实例。在此示例中，每个节点上驻留一个实例。

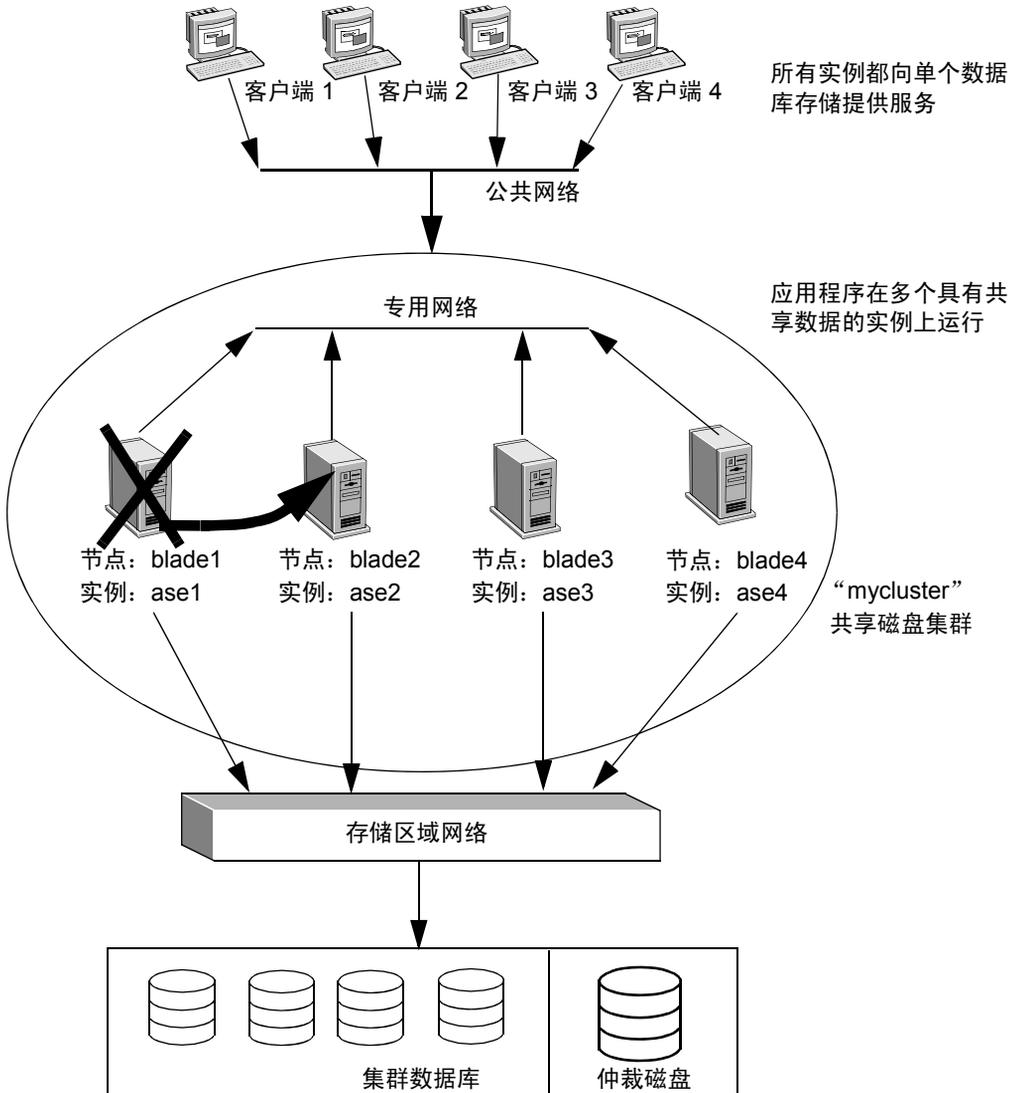
图 1-1: 主要的支持集群的组件



如果一个集群成员发生故障，其负载可以转移到未失败的集群成员上。例如，如果“ase1”发生故障，与该实例连接的客户端就可以通过故障切换功能切换到其余的任何处于活动状态的实例。请参见第 5 页的图 1-2。

**注释** Cluster Edition 可以处理多个不是同时发生的故障，而且可以在后续故障发生之前从初始故障中完全恢复过来。

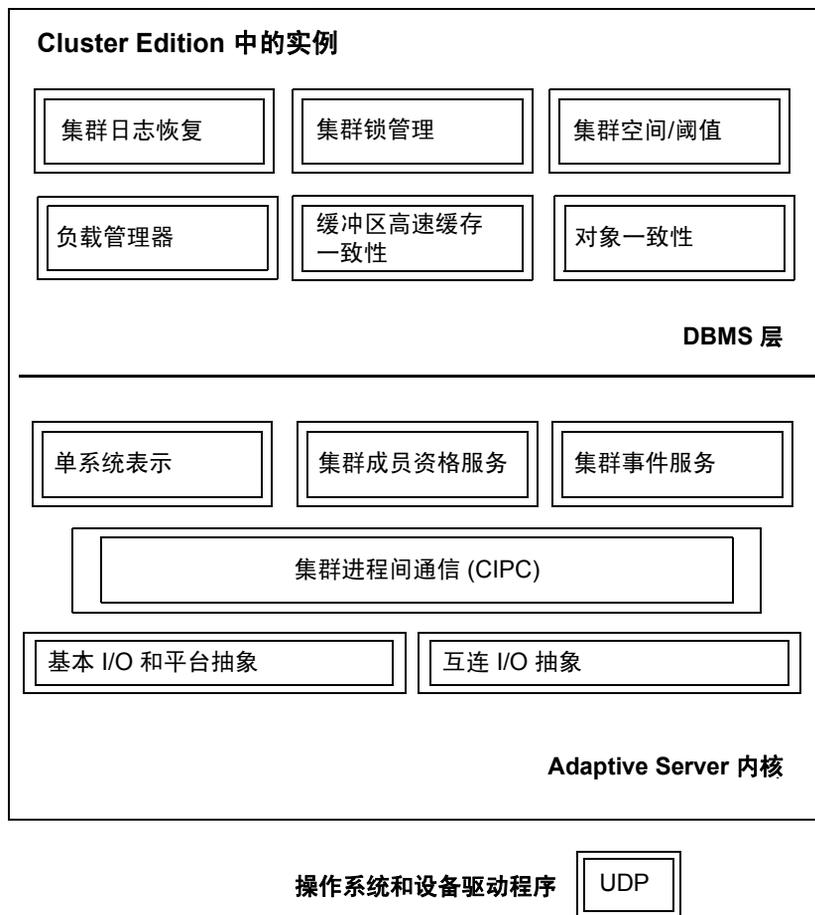
图 1-2: 集群处理故障的方式



## Adaptive Server 集成集群件

Cluster Edition 集群件直接集成到 Adaptive Server 中。不需要外接集群件即可运行 Adaptive Server。有些组件是 Cluster Edition 的新增组件；而其它组件则是现有 Adaptive Server 基础结构的支持集群扩展。图 1-3 说明了这些组件。

图 1-3: 实例的主要支持集群的组件



**Adaptive Server 内核**

这些是新增的、固有的集群基础结构组件。

- 集群成员资格服务 — 管理集群成员资格，检测并处理实例故障。
- 集群进程间通信 (CIPC) — 提供消息传送服务，以及一个互连抽象层，该抽象层能让实例通过冗余通路彼此进行通信。
- 集群事件服务 — 支持全集群范围事件的一般事件发布和订阅机制。

**DBMS 层**

已经对 Adaptive Server DBMS 层中的这些主要组件进行了扩展，可以将其用在 Cluster Edition 环境中：

- 缓冲区高速缓存一致性 — 处理与共享缓冲区高速缓存有关的一致性问题，而且支持分配页、索引页、数据页、对象分配映射 (OAM) 页和全局分配映射 (GAM) 页的高速缓存到高速缓存转移。
- 集群锁管理程序 — 支持在集群中对一致性控制进行分布式锁定。
- 集群日志和恢复 — 处理所有实例的日志，并对数据库恢复进行故障切换。
- 集群空间和阈值 — 处理分布式环境中的空间和阈值管理。
- 对象一致性 — 处理与共享及转移元数据和全局变量有关的一致性问题。对象一致性必须将共享对象的更新串行化，而且使最新的更改对集群中的所有实例可用。
- 负载管理器 — 一个 Adaptive Server 模块，提供应用程序级的资源分配、可用性和负载分配管理。

## 集群协调器

集群协调器处理与成员资格管理和恢复有关的特定任务。任何尝试加入现有集群的实例都会先与集群协调器联系。

没有任何启动参数指示某个特定实例是集群协调器，配置集群协调器和配置集群中的其它实例没有任何差别。最初，集群协调器是您启动的第一个实例。如果集群协调器退出，则会有另一个实例动态担当协调器一职。

## 仲裁设备

仲裁设备包含集群的配置信息，并由所有集群成员共享。仲裁设备必须位于原始分区上，而且必须可供所有承载集群实例的节点访问。

Adaptive Server Cluster Edition 针对以下目的使用仲裁设备：

- 提供一个位置，以便执行集群成员资格管理，包括成员加入的投票和仲裁
- 提供一个持久位置，以便存储实例和 UAF 使用的配置数据
- 通信媒介和同步点

仲裁设备包含有关以下内容的信息：

- 集群的名称、集群中的实例的名称、`interfaces` 文件所在目录的路径、日志文件、主设备，以及其它必需配置信息
- 用以指示集群中每个实例的状态（已启动或已关闭）的集群视图记录
- `Adaptive Server` 用来在检测到实例故障时确定集群成员资格是否正确的区域

在配置集群时创建仲裁设备（请参见《安装指南》）。在初次配置后，可使用 `sybcluster` 或 `qrmutil` 实用程序来备份、恢复和重新配置仲裁设备。有关详细信息，请参见《实用程序指南》。

## Cluster Edition 中的数据库设备

在 Cluster Edition 中，数据库设备（除那些由本地用户临时数据库使用的私有设备外）必须是裸设备（也称为字符设备）。不能将块设备用作数据库设备，因为它们可以在各个硬件节点上对磁盘写入进行缓冲，导致集群实例之间出现数据不一致。

可以在块设备上创建私有设备。私有设备仅由本地用户临时数据库使用。有关配置裸设备的详细信息，请参见操作系统文档。

可以在私有设备上创建本地用户临时数据库，但必须在共享设备上创建本地系统临时数据库。对于 Cluster Edition，可以使用更低廉的本地文件系统设备（块设备）来管理集群中临时数据的存储需要。这些设备是作为私有设备添加的，只能由本地用户临时数据库使用。请参见第 8 章“使用临时数据库”。

例如，在 Linux 系统上，路径 `/dev/sda` 是块设备，不应使用它。但可以将该块设备绑定到 `/dev/raw/raw1` 等裸设备上。

在 Linux 系统上，可以通过使用命令显示的文件类型来区分字符（原始）设备。块设备包含 `b` 作为文件类型，而字符（原始）设备包含 `c` 作为文件类型：

```
[joeadministrator@isles ~]$ ls -l /dev/sda
brw-rw---- 1 root disk 8, 0 Nov 29 06:15 /dev/sda
```

```
[joeadministrator@isles ~]$ ls -l /dev/raw/raw1
crw----- 1 sybase sybase 162, 1 Nov 29 12:17 /dev/raw/raw1
```

在 Solaris 系统上，路径 `/dev/dsk/c0t0d0s1` 是块设备，不应使用它。但可以使用路径 `/dev/rdisk/c0t0d0s1` 将该存储当作字符设备来访问。在字符设备上使用 `ls -l` 命令表示符号链接结尾处的 `raw`：

```
janeadministrator% ls -l /dev/dsk/c0t0d0s1
lrwxrwxrwx 1 root root 49 Apr 23 2007 /dev/dsk/c0t0d0s1 ->
../../devices/pci@780/pci@0/pci@9/scsi@0/sd@0,0:b
janeadministrator% ls -l /dev/rdisk/c0t0d0s1
lrwxrwxrwx 1 root root 53 Apr 23 2007 /dev/rdisk/c0t0d0s1 ->
../../devices/pci@780/pci@0/pci@9/scsi@0/sd@0,0:b,raw
```

在 HP Itanium 系统上，路径 `/dev/disk/disk4` 是块设备，不应使用它。但可以使用路径 `/dev/rdisk/disk4` 将该存储当作字符设备来访问。可以通过使用 `ls -l` 命令显示的文件类型来区分字符（原始）设备。块设备包含 `b` 作为文件类型，而字符（原始）设备包含 `c` 作为文件类型：

```
[jphui@hpiastr1-HP-UX]:/> ls -l /dev/disk/disk4
brw-r----- 1 bin sys 3 0x000000 Feb 23 11:40 /dev/disk/disk4
[jphui@hpiastr1-HP-UX]:/> ls -l /dev/rdisk/disk4
crw-rw-rw- 1 bin sys 23 0x000000 Feb 23 11:40 /dev/rdisk/disk4
```

在 IBM AIX 系统上，可以通过使用 `ls -l` 命令显示的文件类型来区分字符（原始）设备。块设备包含 `b` 作为文件类型，而字符（原始）设备包含 `c` 作为文件类型。路径 `/dev/hdisk1` 是块设备，不应使用它。但可以使用路径 `/dev/rhdisk1` 将该存储当作字符设备来访问：

```
janeadministrator% ls -l /dev/hdisk1
brwxrwxrwx 1 root root 49 Apr 23 2007 /dev/hdisk1

janeadministrator% ls -l /dev/rhdisk1
crwxrwxrwx 1 root root 53 Apr 23 2007 /dev/rhdisk1
```

Cluster Edition 中的数据库设备必须支持 SCSI-3 持久组保留 (SCSI PGR)。Cluster Edition 使用 SCSI PGR 来保证集群成员资格更改期间的数据一致性。Sybase® 不能在不支持 SCSI PGR 的磁盘子系统（Sybase 支持将该配置用于容忍数据损坏的测试和开发环境）上保证数据一致性。

PGR 是 SCSI-3 协议的一种功能。但许多采用更低廉的 SATA 磁盘的存储区域网络 (SAN) 仍提供该功能。请与您的存储供应商联系，以检验系统是否支持 SCSI-3 持久组保留。

有关设备和 I/O 防护的其它信息，请参见所用平台的安装指南。

---

**注释** I/O 防护在 Solaris 非全局区域上不受支持。

---

## 专用安装和共享安装

可以选择是使用共享安装还是专用安装来配置集群：

- 共享安装 — 需要使用网络文件系统 (NFS) 或集群文件系统创建的共享文件系统。使用共享安装创建的集群支持单一 `$SYBASE` 安装目录、Adaptive Server 主目录和服务器配置文件。
- 专用安装 — 支持每个实例有单独的 `$SYBASE` 安装目录、Adaptive Server 主目录和服务器配置文件。服务器配置文件之间的奇偶校验由仲裁设备上的主配置文件维护。

---

**注释** Sybase 建议在使用专用安装时将 LDAP 作为目录服务。

---

集群输入文件中包含一个指明安装模式是共享安装还是专用安装的条目：

```
installation mode = private | shared
```

可以在 `sybcluster`、Adaptive Server 联机帮助文档和所用平台的安装指南中找到有关设置专用或共享安装的说明。另外，有关如何在使用专用安装模式时维护服务器配置文件的说明，请参见第 155 页的“专用安装模式”。

## Cluster Edition 中的 Backup Server

Cluster Edition 能让您配置多个 Backup Server 来执行转储和装载操作。dump 和 load 命令会根据集群的配置情况路由到相应的 Backup Server:

- 专用 — 为集群中的每个实例指派特定的 Backup Server。
- 循环 — 不为实例指派特定的 Backup Server，但在执行转储或装载时，集群会根据可用性为作业指派 Backup Server。

可以让集群只有一个 Backup Server（名为 SYB\_BACKUP）。请参见第 5 章“在集群环境中使用 Backup Server”。

## Cluster Edition 如何增强非集群版本

通过 Cluster Edition，Sybase 扩展了对对称多重处理 (SMP)、非集群服务器的支持，并引入了可以在共享磁盘环境中工作的 Adaptive Server。

可以将多个非集群服务器组合起来，提供共享数据库的单系统视图，从而提高可靠性和易管理性。

一组松散耦合的 Adaptive Server，每个都可以充当非集群服务器，它们共同协作为用户提供单一数据库系统图像。

集群中的 Adaptive Server 共同管理驻留在共享磁盘设备上的单个 Adaptive Server 数据库安装。

Cluster Edition 体系结构的主要优点:

- 提高了可用性 — 该体系结构的共享磁盘性质意味着：只要有一个集群成员保持活动状态，应用程序便可以继续运行，即使在多个其它成员失败之后也是如此。
- 简化了管理 — 在集群成员资格发生更改时不需要数据重新分区，因为数据由所有实例共享。

---

**注释** Cluster Edition 提供分布式体系结构。节点间通信是通过网络互连发生的，而非集群 Adaptive Server 的通信是通过共享内存（高速系统内存）发生的。将节点间消息传送降至最低的应用程序在 Cluster Edition 环境中可实现最佳性能。

---

## 在集群中使用互连网络

Cluster Edition 能让您在集群中的实例之间配置一个或两个互连网络。虽然一个互连网络就已足够，但两个互连网络能让集群具有冗余度和更为稳健。互连网络在集群中的每个实例之间形成一组逻辑链路。这些链路在实例之间发送消息，并且受到 Cluster Edition 的监控。如果检测到任何故障，Cluster Edition 会通过备用网络在实例之间重新路由通信流。

互连链路可能会由于多种原因失败：网络堆栈内的物理故障（如电缆断开或损坏）、电源故障（如一个网络基础结构设备）或软件故障。Cluster Edition 通过监控实例之间的通信流来检测这些故障。每个实例都监控在多个链路上发送的消息。只要有传入的消息，就可认为链路正常。

实例可能由于节点发生故障或链路关闭而不会始终发送消息。如果集群中的特定实例处于空闲状态，Cluster Edition 会启动主动探查机制来确定是否可以使用网络链路与支持该实例的节点联系。该机制可确保非活动期间不会错误地触发链路故障和后续的切换事件。

如果一个链路标记为无法运行，系统就会定期进行尝试，以便在链路恢复时建立连接，无需人工干预便可继续正常运行。

---

**注释** 如果在 Cluster Edition 中定义了多个网络，系统会自动实施链路监控。

---

## 监控实例之间的链路

monCIPCLinks 监控表监控集群中实例之间的链路的状态。monCIPCLinks 针对每个链路有两个状态：“被动”和“主动”。

---

**注释** 逻辑集群及该集群中的每个实例可以有不同的状态。

- 逻辑集群具有总体（或全局）状态，该状态确定集群是处于脱机还是联机状态。
- 每个实例都有相应的状态，用以描述逻辑集群所识别的该特定实例的状态。

有关集群“状态”的详细说明，请参见第 82 页的“[集群和实例状态](#)”。

---

“被动”状态用于监控通过链路发送的日常消息。当链路上没有消息通信流时，Cluster Edition 会收集主动状态。链路的状态被描述为“状态”，每个状态都有一个与其关联的使用期限（以毫秒作单位）。状态有“Up”（已开启）、“Down”（已关闭）和“In doubt”（受到怀疑）。当实例之间不发送消息时，状态为“In doubt”。

当集群正常运行时，可以使用经常性节点间通信流来确定链路的状态。这叫做被动监控，用于维护链路的被动状态。如果在非活动时限内集群中发生了用于确定状态的监控，所定义的状态就可能变得失效而不可靠（即，在非活动时限内被确定是 Up 的状态有可能实际上是 Down，但非活动使 monCIPCLinks 表无法在结果集中显示这个事实。这种非活动状态在 PassiveState 列中被描述为“In doubt”。一旦链路标记为“In doubt”，就会触发主动链路状态监控，ActiveState 列所描述的值就会有效。

每个主动和被动状态都有一个与其关联的使用期限，显示上次更新状态的时间。如果正常通信流足以维护链路状态，主动状态就不会更新，与该状态关联的使用期限值就会变大。值较大意味着关联的状态可能不再准确地表示链路的真实状态。

如果实例不发送消息，PassiveState 就会显示为 In doubt，但 ActiveState 会显示实际状态：Up、In doubt 或 Down。

下例显示了包含两个节点的集群，其中两个链路都在运行，而且它们之间有通信流。由于所有链路的 PassiveStateAge 都是 0，因此可以假定输出是链路状态的真实反映：

InstanceID	LocalInterface	RemoteInterface	PassiveState
PassiveStateAge	ActiveState	ActiveStateAge	
2	ase2	ase2	Up
0	Up	10300	
2	blade2	blade1	Up
0	Up	0	
1	ase1	ase2	Up
0	In doubt	17900	
1	blade1	blade2	Up
0	Up	100	

下例显示了同样的包含两个节点的集群，描述的是主互连网络失败后的情况。网络结束点“ase1”和“ase2”之间的链路的 **PassiveState** 值是“**In doubt**”，**PassiveStateAge** 值“大”（说明 **ActiveState** 表示链路的真实状态）。**ActiveState** 值更小，将链路显示为“**Down**”：

InstanceID	LocalInterface PassiveStateAge	RemoteInterface ActiveState	ActiveStateAge	PassiveState
2	ase2 13500	Down	700	In doubt
2	blade2 0	Up	700	Up
1	ase1 13600	Down	400	In doubt
1	blade1 0	Up	400	Up

**注释** 链路发生故障和主动状态真实反映链路状态这两者之间略有延迟。

忽略任何 **ActiveStateAge** 值“大”的状态，因为这表明链路的存在时间较长，值可能不准确。当链路状态的存在时间较长，**ActiveStateAge** 值“大”时，主动监控会因没有消息而被触发，但还未确定链路状态。

**注释** 在集群输入文件中设置主辅助互连网络和辅助互连网络时，除非两个互连网络都在运行，否则不要重新启动集群。

## 建议的配置方案

通常，**Cluster Edition** 支持本节中讲述的方案，但那些包含所用平台的《发行公告》中所列功能的方案除外。

多数考虑采用集群数据库体系结构的用户都有以下目标：

- 提高可用性 — 如果某个节点发生故障，集群中的其它节点会继续运行，数据库仍然可用。
- 提高可管理性 — 可以将多个应用程序和数据库整合到一个集群中，从而降低管理的复杂程度并降低扩展成本。

- 提高可扩展性 — 支持多个节点，使集群数据库可以扩展，打破单节点环境所施加的限制。通常叫做“垂直可扩展性”，这意味着可通过增加更多 CPU、内存、主机总线适配器 (HBA)、网络接口卡 (NIC) 等，来提高单个节点的处理能力。
- 降低总拥有成本 — 软件可以配置到行业标准的、非专有的硬件上，从而降低了购买、维护和支持的成本。

## OLTP 应用程序的 HA 故障切换

如果您当前使用高可用性 (HA) 功能运行生产环境，请使用下述 Cluster Edition 方案之一来替换当前由主动 - 被动 (备用) 或主动 - 主动 (协同) 配置中的 Sybase 故障切换提供的 HA 功能，或操作系统提供的集群件 (例如，Sun Cluster 和 Veritas Cluster)。

选择并配置最能反映公司需要的故障切换方案 (给定应用和财务限制的服务级协议要求)。

- *1:1 主动 - 被动*

集群的节点和实例是成对设置的，即一个空闲 (被动) 节点和实例等待对应的处于活动状态的节点发生故障。这种方案仅在极端环境 (即，要求不容忍在任何故障切换方案中发生服务降级，包括多个故障) 中才经济、高效。

- *1:1 主动 - 主动*

集群的节点和实例是成对设置的，而且每对节点和实例都在服务于单独的应用程序和数据库 (实例) 的同时，以“协同模式”彼此进行监控，以检测对方是否发生故障。虽然这种方案模拟当前 Sybase HA 选项并充分利用资源，但故障切换处理期间的服务级别会降低，除非您在正常处理期间保持足够低的资源容量 (小于 50%) 以提供足够的容量供单个实例运行二者的负载。

- *N:1 (N 个主动节点, 1 个被动备用节点)*

这种方案提供单个被动备用节点和实例来监控任意数量的主动实例。这是最经济、高效的方案；将实例通过故障切换功能切换到单个节点的多个节点故障可能会导致服务级别降低，而这可能会使该方案变得不可接受。为被动节点配置更多容量 (例如，CPU 和内存) 可减缓大多数可能故障切换方案中的降级。

- $N:M$  ( $N$  个主动节点和实例,  $M$  个被动备用节点)

此模型提供任意数量 ( $M$ ) 的被动备用实例来监控任意数量 ( $N$ ) 的主动实例。此选项不仅可降低成本, 还可处理大多数典型的故障方案。被动备用节点数量的决定因素通常有: 多节点故障方案中的成本、系统级平均无故障时间间隔 (MTBF) 统计信息以及在降低的服务级别运行所造成的业务影响。在多数故障方案中, 都可以为被动节点配置更多容量来减缓这种降级。

## DSS 报告应用程序的水平可扩展性

如果应用程序的联机活动主要由大规模报告和决策支持系统 (DSS) 组成, 从大量用户那里查询, 请考虑创建多节点集群, 所有 Adaptive Server 实例都服务于同样的应用程序。

当组织该系统时, 请注意:

- 随着负载从一个节点扩展到  $N$  个节点, 用户、查询和响应时间的可扩展性。
- 客户端在实例之间的负载分布以及与实例容量和性能的关系。

## OLTP 应用程序的水平可扩展性

读写密集型应用程序 (如联机事务处理 (OLTP) 应用程序) 过去总是遇到可扩展性难题, 因为在资源争夺的同时还要维护数据库系统的 ACID 属性。当使用共享磁盘集群技术时, OLTP 应用程序的水平可扩展性增加了更多难题, 因为在共享磁盘集群中的服务器实例之间需要数据一致性和消息传送。计算机和网络的物理性质不允许 OLTP 应用程序在共享磁盘集群上无限扩展。随着节点的数量以及用户的负载和数量不断增加, 在节点之间维护缓冲区一致性而需要的节点间消息传送量呈指数增长态势。

在共享磁盘集群上扩展 OLTP 负载的最佳方法是: 将应用程序和数据分区, 成为相互排斥的集合 (即, 将数据分离到不同的数据库中), 以避免在服务器实例之间发生处理协作, 并且从同一实例访问某个应用程序的数据。因此, 您必须仔细考虑如何在数据库级别将 “数据” 分区, 以消除参与实例之间的日志和数据争夺。可以使用逻辑集群和负载管理, 通过单个实例简化对这种分段 “数据” 的访问。

逻辑集群能让您将独立的实例分配给不同应用程序或负载，以便从逻辑上让这些数据组在单个集群下运行。这样会减少实例间的访问，而且，通过与每个实例的专用临时数据库相结合，可帮助在 Cluster Edition 中配置支持持续数据可用性的 OLTP 应用程序。

## Cluster Edition 中的新增客户端技术

Cluster Edition 支持单系统表示。也就是说，组成集群的多个实例对客户端来说是单个系统。新增客户端技术能让客户端在以逻辑方式连接到集群的同时，以物理方式连接到各个实例。这种逻辑连接能让 Adaptive Server 将客户端重定向到集群中的多个实例，并动态为客户端提供高可用性故障切换数据。请参见第 2 章“客户端应用程序和客户端/服务器交互”和 New Features Open Server 15.0 and SDK 15.0 for Microsoft Windows, Linux, and UNIX（《针对 Microsoft Windows、Linux 和 UNIX 的 Open Server 15.0 和 SDK 15.0 新增功能》）。

新增客户端技术包括：

- 登录重定向 — 当客户端重新连接到集群中的另一个实例时。
- 连接迁移 — 当已建立的连接移至集群中的另一个实例时。
- 扩展的高可用性故障切换 — 能让客户端多次进行故障切换，直到找到第一个可用的或一个装载最少的实例。

## 复制支持

Cluster Edition 支持使用 Replication Server 和 RepAgent 线程实现的复制。集群数据库可以是 Sybase 集群系统中的源或目的地。可以从集群中的任何实例执行所有任务，例如配置 RepAgent 或将表标记为复制。复制状态在整个集群中是相关联的。

### 配置 RepAgent

在集群系统中配置主数据库时，您指定的服务器名称应该是集群名称。可以使用 `select @@servername` 显示集群名称。

`sp_config_rep_agent` 的语法不需要集群或实例名称。缺省情况下，Cluster Edition 和非集群 Adaptive Server 版本都采用 `select @@servername` 值。在 Cluster Edition 中，该语句返回当前集群名称。例如：

```
1> select @@servername
2> go

-----
MYCLUSTER
```

### 启动 RepAgent

缺省情况下，RepAgent 在协调器上启动。

但可以将 RepAgent 配置为在集群中的任何实例上启动。例如，要将主数据库 pdb 上的 RepAgent 配置为始终在实例 “ase2” 上启动，请输入：

```
sp_config_rep_agent pdb, "cluster instance name",
"ase2"
```

配置后，必须使用 `sp_start_rep_agent` 重新启动 RepAgent，新配置才能生效。

要恢复缺省行为，使 RepAgent 始终在协调器上启动，请输入：

```
sp_config_rep_agent pdb, "cluster instance name",
"coordinator"
```

一个实例在其节点上启动时，它会检查数据库是否配置为在其节点上启动。如果是且数据库标记为自动启动，RepAgent 便会启动。

当协调器启动时，它会启动所有未配置为在特定实例上启动的 RepAgent。如果协调器出现故障，或因正常关闭而停止，RepAgent 会在新的协调器上启动。

如果 RepAgent 配置为在特定实例而非协调器上启动，且此实例关闭或出现故障，则 RepAgent 会在协调器上启动。

---

**注释** Cluster Edition 不支持 Adaptive Server Enterprise Replicator，因为后者需要的 `dbcc logtransfer` 接口不被支持。

---

## 客户端应用程序和客户端/服务器交互

本章介绍客户端/服务器交互以及如何修改进行 Open Client/Client-Library 调用的应用程序以支持集群。它还说明了如何在共享磁盘集群环境中使用 isql。

主题	页码
<a href="#">Open Client</a>	20
<a href="#">在 Client-Library 应用程序中启用故障切换</a>	21
<a href="#">客户端/服务器交互</a>	22
<a href="#">在集群环境中使用 isql</a>	29
<a href="#">在集群环境中使用远程过程调用</a>	29
<a href="#">当节点断电时重新连接客户端</a>	31

---

**注释** DB-Library 可以连接到共享磁盘集群中的实例，但是不支持 Sybase 共享磁盘或高可用性功能。

---

随 Cluster Edition 提供的 Open Client™、jConnect™ for JDBC™、ODBC、OLE DB 和 ADO.NET 版本支持以下功能：

- 登录重定向 — 实例在确认登录前将传入的客户端连接重定向到其它实例的功能。登录重定向在登录序列期间进行。客户端应用程序不会收到关于已将其重定向的通知。
- 连接迁移 — 当现有客户端从集群的一个实例转移到另一个实例时发生。有关迁移可能发生的时间和连接条件的信息，请参见第 24 页的“连接迁移”。
- 扩展的高可用性故障切换 — 在扩展的故障切换配置中，Adaptive Server 会在“支持高可用性的”客户端连接时为它们提供一个故障切换地址列表。这样，如果支持高可用性的客户端或应用程序连接到的实例出现故障，则这些客户端或应用程序可以进行多次故障切换。

这些客户端的 `interfaces` 文件或目录服务中不需要有 `HAFAILOVER` 条目。不过，如果它们的 `interfaces` 文件或目录服务中确实有 `HAFAILOVER` 条目，则这些客户端会继续使用此条目，直到 `Adaptive Server` 向它们发送要连接到的故障切换地址或服务器的列表。这些客户端始终使用 `Adaptive Server` 提供的最新列表。

要实现登录重定向和连接迁移，请确保应用程序使用客户端库的当前副本：

- 如果应用程序链接到共享库 — 在库搜索路径中的旧客户端库之前包括新客户端库。
- 如果应用程序静态链接 — 重新链接应用程序。

使用 `CS_PROP_MIGRATABLE` 连接属性可启用和禁用连接迁移。缺省情况下，`CS_PROP_MIGRATABLE` 处于启用状态。请参见 `Client Library Reference Manual`（《`Client Library` 参考手册》）。

要实现扩展的故障切换，请确保应用程序使用客户端库的当前副本且您已启用高可用性。有关启用高可用性的信息，请参见《在高可用性系统中使用 `Sybase` 故障切换》。

有关将故障切换功能与 `jConnect`、`ODBC`、`OLE DB` 和 `ADO.NET` 驱动程序一起使用的信息，请参见 `New Features Open Server 15.0 and SDK 15.0 for Microsoft Windows, Linux, and UNIX`（《适用于 `Microsoft Windows`、`Linux` 和 `UNIX` 的 `Open Server 15.0` 和 `SDK 15.0` 的新增功能》）。

## Open Client

下列版本支持登录重定向、连接迁移和扩展的高可用性故障切换：

- 登录重定向 — `Open Client 15.0` 版
- 扩展的高可用性 — `OCS 15.0 ESD #3` 版
- 迁移 — `OCS 15.0 ESD #8` 版

支持这些版本的 `SDK` 组件为 `OpenClient/Client-Library`。

## 在 Client-Library 应用程序中启用故障切换

任意现有应用程序都可以连接到 Cluster Edition。但是，要使用扩展的高可用性 (HA) 功能，可能需要更改应用程序代码。

- 对于基于现有 Adaptive Server HA 功能的 HA 应用程序，不需要更改应用程序代码。
- 现有非 HA 应用程序可以从无需代码更改或需要很少更改的 Cluster Edition HA 功能的某些方面受益。但是，在这些情况下，故障切换是不透明的：当第一次检测到故障切换时应用程序收到一条错误消息。用户必须重新提交批处理或事务以启动故障切换。

对非 HA 应用程序启用故障切换：

- 对于 isql，指定当连接到 Adaptive Server 时的 -Q 选项。
- 对于与 Client-Library 链接的应用程序，设置一个启用故障切换的对应连接属性。

要使故障切换对用户透明，应用程序必须主动检查故障切换错误状态，并自动重新提交批处理或事务。

在所有情况下，您必须更新应用程序所使用的 Client-Library 版本，以使用与集群相关的 HA 功能。

在 Client-Library 应用程序中启用故障切换：

- 1 使用 `ct_config` 或 `ct_con_props` Client-Library API 调用在上下文或连接级别设置 `CS_HAFAILOVER` 属性：

```
ct_config (context, action, CS_HAFAILOVER, buf,
          buflen, &outlen);
ct_con_props(connection, action, CS_HAFAILOVER,
            buf, buflen, &outlen);
```

有关 `CS_HAFAILOVER` 属性的详细信息，请参见《Client-Library/C 参考手册》。

- 2 如果尝试连接到已关闭的实例，则行为与非集群 Adaptive Server 相同：Client-Library 尝试 `interfaces` 文件中实例名称的所有查询条目，直到其中一个匹配，或者直到没有可尝试的条目为止。对于所有实例，包括客户端 `interfaces` 文件中的查询行。应用程序可以连接到由一系列 `interfaces` 文件查询条目表示的集群。有关 `interfaces` 文件的信息，请参见 *Cluster Edition Installation Guide*（《Cluster Edition 安装指南》）。

- 3 当发生成功的故障切换时， Client-Library 发出一个名为 CS\_RET\_HAFAILOVER 的返回值，它特定于一些 Client-Library API 调用，包括：

```
ret = ct_results(cmd, result_type)
ret = ct_send(cmd)
```

在同步连接（一个需要服务器响应，并在接收到响应之前一直受到阻止的例程）期间， CS\_RET\_HAFAILOVER 从 API 调用返回。在异步连接（一个需要服务器响应立即返回 CS\_PENDING 的例程）中，这些 API 发出 CS\_PENDING， callback 函数返回 CS\_RET\_HAFAILOVER。根据返回代码，客户可进行所需的处理，设置上下文，并发送要执行的下一条命令。

- 4 使用 Open Client SDK 来重建应用程序，该 Open Client SDK 的版本至少等于随 Cluster Edition 提供的 Open Client SDK 的版本。

有关配置应用程序以实现高可用性的信息，请参见《在高可用性系统中使用 Sybase 故障切换》。

## 客户端/服务器交互

本节中的功能使用 Open Client 15.0 库，系统将自动启用这些功能。

### 登录重定向

在登录时，如果一个实例由于负载的原因通知客户端登录到另一个实例，就会发生登录重定向。

不需要为客户端重定向执行任何其它配置。

登录重定向由 Adaptive Server 的工作负荷管理器用来根据逻辑集群配置和集群的当前工作负荷将传入的连接发送到特定实例。

如果尝试连接到已关闭的实例，则行为与非集群 Adaptive Server 相同：客户端尝试给定服务器的目录服务中的所有条目，直到可以成功连接为止。因此，目录服务中的服务器条目应当包含集群中所有实例的连接信息。

下面的示例包括集群“mycluster”中运行的计算机“blade1”、“blade2”、“blade3”和“blade4”上的 Adaptive Server “ase1”、“ase2”、“ase3”和“ase4”。

```
ase1
  query tcp ether blade1 19786
ase2
  query tcp ether blade2 19786
ase3
  query tcp ether blade3 19786
ase4
  query tcp ether blade4 19786
mycluster
  query tcp ether blade1 19786
  query tcp ether blade2 19786
  query tcp ether blade3 19786
  query tcp ether blade4 19786
```

例如，如果客户端连接到集群“mycluster”，它首先尝试连接到“ase1”实例。如果“ase1”关闭，它将尝试 `interfaces` 文件中的下一个条目“ase2”，依此类推。成功连接后，负载管理器可以根据负载规则将客户端重定向到另一个实例。

虽然按 `interfaces` 文件中指定的顺序尝试实例，但是当主机或网络处于无法访问的状态或关闭时，连接尝试失败可能会花费很长时间。您可以通过向连接信息中添加登录超时来提高重试尝试的速度。

在上面的示例中，如果您指定的登录超时时间间隔比连接客户端的缺省值短，则该客户端尝试更快地连接到实例“ase2”。

请参见 Adaptive Server 《实用程序指南》中的 `isql -l` 参数说明和 Client-Library Reference Manual（《Client-Library 参考手册》）中的 `CS_LOGIN_TIMEOUT` 属性。

## 用于登录重定向的连接属性

设置用于配置登录重定向的连接属性：

- `CS_PROP_REDIRECT` — 启用和禁用登录重定向。
- `CS_DS_RAND_OFFSET` — 禁用或启用让从目录服务随机检索的第一个查询条目来查找 `ct_connect`。缺省情况下，此属性设置为 `false`。

请参见 Client Library Reference Manual（《Client Library 参考手册》）。

## 连接迁移

连接迁移在现有客户端从集群的一个实例转移到另一个实例时发生。例如，连接迁移的原因可能是当前连接到的实例关闭以进行维护，也可能是为了负载均衡。转移对客户端应用程序是透明的。连接迁移允许集群在整个逻辑集群中平衡负载。

连接迁移允许负载管理器在管理性故障切换、故障恢复或逻辑集群脱机期间在实例之间正常移动现有连接。负载管理器可以使用迁移来进行动态负载分配，在此过程中，一些现有连接在实例间进行迁移以更平均地分配负载。

当实例使用 Open Client 15.0 客户端库时，会自动启用连接迁移。不需要为连接迁移执行任何其它配置。

## 迁移与故障切换的区别

迁移是 Adaptive Server 请求的计划好的受控事件。故障切换是未计划的事件，它在 Adaptive Server 崩溃或网络断开连接后发生。

应用程序并不知道迁移，您不需要为迁移编写代码。但是，您必须专门编写应用程序代码以获得故障切换支持。

迁移在新实例上还原完整会话上下文。故障切换不会还原完整会话上下文，而是由应用程序来还原其自己的上下文。

## 何时发生迁移？

负载管理器可以在请求成功发送到客户端时启动迁移。特别是，迁移可以在以下情况下发生：

- 对于完成登录的连接：
  - 在实例从客户端接收新的批处理之后，分析和执行批处理之前。
  - 在实例完成对客户端批处理的处理之后，将最终完成结果发送到客户端之前。
  - 在实例不代表客户端执行任何批处理时。
- 根据负载管理器的算法。负载管理器迁移某些客户端的原因可能是为了负载均衡，也可能是因为当前实例关闭以进行维护。
- 当连接上下文容纳迁移时。负载管理器将连接的目标设置为迁移，但是这些连接只有在其上下文允许这样做时才进行迁移。尤其是，在事务内部不会发生迁移。

---

**注释** 执行 Java 的连接不支持连接迁移。请参见第 157 页的“在集群环境中使用 Java”。

---

## 迁移的上下文

源实例在成功完成迁移时将客户端的完整上下文传播到目标实例，然后目标实例检索上下文。

客户端的完整上下文在迁移完成后还原，从而使迁移完全对客户端透明。但是，迁移的连接需要新的 `spid`。

客户端的上下文由下列元素组成：

- 当前数据库的名称
- 命令的任何挂起批处理（如果迁移在预批处理模式下发生）
- 客户端的登录记录
- 客户端的语言和字符集
- 客户端的功能
- 任何监控计数器和统计
- 任何角色
- 任何 `set` 选项
- 跟踪标志 3604 和 3605

## 迁移条件

迁移是异步事件：发出让任务进行迁移的请求，任务只有在达到抑制状态时才进行迁移。对于 Cluster Edition，抑制状态是指以下状态：

- 未执行查询批处理
- 没有打开的事务
- 没有会话级别的临时表
- 没有声明的游标
- 自初始连接以来未更改过口令
- 未运行 `set user` 或 `set proxy`
- 未使用逻辑进程管理器绑定到引擎

- 未使用 ASE Active Messaging
- 不是与入站节点处理器关联的逻辑连接
- 没有在单一用户模式下打开数据库。

## 上下文迁移

当 Adaptive Server 将现有连接迁移到另一个实例时，它还必须从现有连接迁移一些上下文，例如当前数据库和 `set` 选项。

## 配置客户端迁移

`idle migration timer` 和 `session idle timer` 配置参数控制空闲客户端何时迁移。这两个参数之和确定迁移预计完成所需时间的秒数上限。

`idle migration timer` 的缺省设置为 60，以秒计。`session idle timer` 的缺省设置为 600，以秒计。

如果将 `idle migration timer` 设置为 0，则实例在发送迁移请求后立即关闭发出迁移请求的连接。如果将 `session idle timer` 设置为 0，则实例使任何在 `idle migration timer` 到期之前未完成的空闲迁移失效。

将 `session idle timer` 设置为较大的值可以增加空闲连接成功迁移的机会。但是，实例必须将迁移客户端的上下文保留很长时间。这还意味着同时将 `idle migration timer` 和 `session idle timer` 设置为 0 会禁用迁移。请参见《系统管理指南：第一卷》。

下面的示例描述 `idle migration timer` 和 `session idle timer` 的各种配置。

两个参数都设置为缺省值

- 如果实例针对无法处理异步通知的空闲客户端发出迁移请求，并且用户或应用程序在 `idle migration timer` 到期之前在该客户端上发出命令，则该客户端立即迁移，该命令将在目录实例上执行。

“预先读取”客户端在应用程序请求任何挂起的数据之前，从网络读取这些数据。
- 如果满足所有下列条件，则实例关闭与客户端的连接：
  - `idle migration timer` 和 `session idle timer` 都设置为其缺省值。
  - 实例针对空闲的非预先读取客户端发出迁移请求。
  - 客户端在 `idle migration timer` 期间保持空闲。

- 如果在实例关闭连接之后的前 600 秒（idle migration timer 的缺省值）在客户端上发出命令，则客户端成功迁移，然后命令在目标实例上执行。
- 如果满足下列条件，则实例关闭与客户端的连接：
  - idle migration timer 和 session idle timer 都设置为其缺省值。
  - 实例针对空闲的非预先读取客户端发出迁移请求。
  - 在 idle migration timer 到期之前没有针对客户端发出任何命令。
- 如果满足下列条件，则实例使迁移无效：
  - idle migration timer 和 session idle timer 都设置为其缺省值。
  - 实例针对空闲的非预先读取客户端发出迁移请求。
  - 在 idle migration timer 到期之前没有针对客户端发出任何命令。

由于发出了命令，客户端在检测到迁移请求时立即尝试迁移。但是，实例拒绝迁移请求，客户端尝试在初始实例上继续。由于与实例的连接关闭（idle migration timer 超时），如果客户端具有 HA 功能，它将尝试 HA 故障切换，否则它报告与应用程序断开连接。

idle migration timer 设置为 0， session idle timer 设置为 600

如果实例针对空闲的非预先读取客户端发出迁移请求，则该实例在发送迁移请求后立即关闭与客户端的连接。如果在初始 600 秒期间针对客户端发出命令，则客户端成功迁移，否则它将以上一项目符号中所述的方式失败。

idle migration timer 设置为 60， session idle timer 设置为 0

实例针对空闲的非预先读取客户端发出迁移请求。如果在 idle migration timer 到期之前针对客户端发出命令，则客户端成功迁移。但是，如果在 idle migration timer 到期之前未发出任何命令，客户端也可以进行迁移，这是因为 session idle timer 设置为 0。

idle migration timer 和 session idle timer 设置为 0

如果这两个参数都设置为 0，则没有实例会迁移客户端。

## 扩展的高可用性故障切换

当“支持 HA 的”客户端连接时，Adaptive Server 会为其提供一个故障切换地址列表。这样，当支持高可用性的客户端或应用程序连接到的实例变得不可用时，这些客户端或应用程序可以进行多次故障切换。如果该实例尚未向客户端发送故障切换列表，则客户端会使用 `interfaces` 文件中的 `HAFILOVER` 条目信息。

在下面的示例中，如果在登录期间出现网络故障且实例尚未发送扩展的高可用性列表，则允许支持 HA 的客户端进行故障切换：

```
ase1
      query tcp ether blade1 19786
ase2
      query tcp ether blade2 19786

mycluster
      query tcp ether blade1 19786
      query tcp ether blade2 19786
      hafailover mycluster
```

`HAFILOVER` 条目应使用集群别名作为服务器名称，原因是客户端应用程序会尝试每个 `query` 行，直到它与集群中的实例建立连接为止。

扩展的故障切换需要 Open Client 15.0 ESD #3 或更高版本。Cluster Edition 中的客户端库包含 ESD #8。

Open Client 使用 `CS_PROP_EXTENDEDFAILOVER` 属性进行扩展的故障切换。有关详细信息，请参见《Client-Library/C 参考手册》。

## HA 故障切换与集群中的故障切换的区别

在客户端上，启用高可用性意味着应用程序在网络故障期间收到错误代码 `CS_RET_HAFILOVER`，客户端库将自动重新连接到服务器。在高可用性中，应用程序再次尝试首先连接到初始服务器，如果连接失败，则尝试辅助服务器（根据主服务器的 `interfaces` 文件条目中的设置）。如果您使用随 Cluster Edition 提供的 Open Client 库，则实例将故障切换目标列表发送到客户端。应用程序使用此列表确定当实例出现故障时在何处进行连接。

无论您是使用高可用性还是使用 Cluster Edition，都可以随时进行故障切换。

## 在集群环境中使用 *isql*

缺省情况下，可以使用 *isql* 连接到共享磁盘集群环境中的 Adaptive Server 实例。但是，要连接到共享磁盘集群中的 Adaptive Server 实例并为客户端启用高可用性故障切换或扩展的高可用性故障切换，则必须使用 *-Q* 选项启动 *isql*。

---

**注释** 虽然 *isql -Q* 可以使用扩展的高可用性功能，但是它不是透明的：当实例出现故障时，*isql* 收到一条错误，您必须重新提交批处理或事务。

---

请参见《实用程序指南》。

## 在集群环境中使用远程过程调用

远程过程调用 (RPC) 允许客户端在远程服务器上启动存储过程。远程服务器如同本地客户端请求那样在其上下文中运行系统过程，并将结果通过网络发送回原始服务器。

除了服务器对服务器的 RPC，Cluster Edition 还包含其它三类 RPC。在第一类包含的 RPC 中，远程服务器是一个集群。在第二类包含的 RPC 中，本地或起始服务器是一个集群实例。在第三类包含的 RPC 中，本地和远程服务器都是同一集群中的实例。

在早期版本的 Adaptive Server 中，*cis rpc handling* 的缺省值为 0。在 Cluster Edition 中，缺省值为 1，这会强制 RPC 处理将组件集成服务 (CIS) 作为缺省 RPC 处理机制而不是节点处理器。

集群实例使用集群名称（而不是实例名称）将其自己标识给远程服务器。*@@servername* 全局变量返回集群的名称。

## 远程服务器在其中是集群的 RPC

由于 Cluster Edition 支持单系统映像，因此使用集群作为远程服务器对发送 RPC 请求的服务器的影响很小。

如果发送 RPC 的服务器使用 *cis rpc handling*，则集群将进站请求视为常规客户端连接。负载管理器尝试根据配置的路由规则将 RPC 路由到相应的实例和逻辑集群。只要启动服务器指明它支持登录重定向，负载管理器规则就可以决定登录重定向。Adaptive Server 15.0 及更高版本支持针对 RPC 请求的登录重定向。

如果启动服务器使用节点处理器，则绕过集群负载管理器，RPC 在接受连接的实例的系统逻辑集群中运行。

## 本地服务器在其中是集群的 RPC

Cluster Edition 不支持通过节点处理器的出站 RPC，缺省情况下使用的是 CIS RPC 处理。由于集群发出的出站 RPC 是使用集群名称标识的，因此将远程服务器配置为接受来自集群的 RPC，而不是接受来自单个实例的 RPC。

## 本地和远程服务器在其中都是同一集群中的实例的 RPC

集群中的实例有时使用 RPC 进行集群内通信。当集群启动时，Adaptive Server 自动将每个集群实例添加到 `syssservers` 表，并从不再位于集群定义中的 `syssservers` 删除所有集群实例。当在运行时动态添加和删除实例时，也会这样做。这些 `syssservers` 条目设置了 `cluster instance` 状态位。由于集群内 RPC 针对特定实例，`syssservers` 条目未设置 `enable login redirection` 状态位。

## sp\_serveroption

enable login  
redirection

`sp_serveroption` 系统过程包含 `enable login redirection` 和 `cluster instance` 选项。

`sp_serveroption enable login redirection` 确定传入的 RPC 请求是否可以发送给集群中的另一个实例。语法为：

```
sp_serveroption instance_name, 'enable login redirection', [ true | false ]
```

其中：

- *instance\_name* — 是您为其设置 `enable login redirection` 的实例的名称。此实例必须包括在 `syssservers` 中，且自动针对所有实例包括。
- `true` — 表示实例可以将传入 RPC 请求重定向到集群中的另一个实例。
- `false` — 表示实例不能重定向 RPC 请求。

缺省情况下，启用 `enable login redirection`。（必须具有 `sa_role` 才能运行 `sp_serveroption`。）

`cluster instance` `cluster instance` 标识存储实例信息的 `syssservers` 条目，其中 `instance_name` 是您添加的实例的名称。

```
sp_serveroption instance_name, 'cluster instance', [ true | false ]
```

缺省情况下，对每个远程服务器禁用 `cluster_instance`（设置为 `false`）。

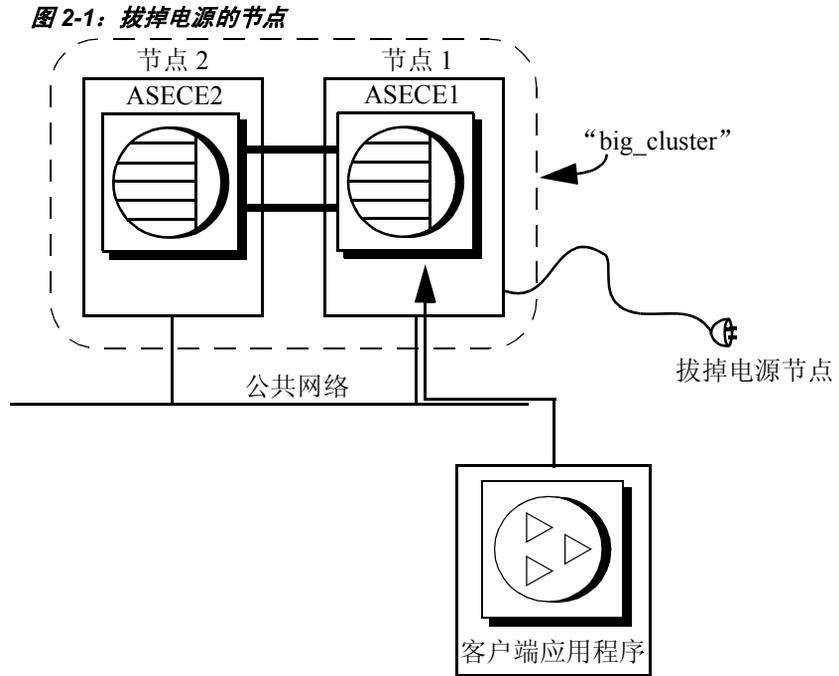
Cluster Edition 自动为本地集群中的实例管理 `syssservers` 行。您不需要手动设置或清除 `cluster instance` 标志。

## 当节点断电时重新连接客户端

如果从计算机拔掉网络电缆或者客户端连接到的节点断电，将无法访问客户端接口。客户端接口无任何结果地等待服务器的答复，或者等待集群发出发送操作。

在图 2-1 的情况中，客户端应用程序连接到“big\_cluster”，它由分别运行“ASECE1”和“ASECE2”实例的“Node1”和“Node2”组成。客户端应用程序连接到“Node1”上运行的“ASECE1”实例。

如果“Node1”断开电源，则客户端应用程序等待来自该节点联系信息。避免此情况的唯一方法是将客户端应用程序配置为假设该节点在指定的时间段后关闭。于是，它连接到集群中的另一个节点。



操作系统检测到崩溃，断开客户端的连接，将从连接的远程端对接口进行故障切换。

要减少检测集群丢失主机或公共网络与运行实例的节点断开连接所需的时间，您可以：

- 将运行客户端的主机上的 TCP *keepalive* 设置为合理的值。
- 设置客户端应用程序的 *timeout* 值。

将 TCP *keepalive* 设置为较小的值

TCP *keepalive* 参数最终将客户端接口标记为出现故障。但是，由于 TCP *keepalive* 值的缺省值是很长的时间段（在某些系统上它可能设置为两个小时），可能需要三小时或更长时间客户端接口才能进行故障切换。将 *keepalive* 设置为较小的值（几分钟）对于大型组织而言是不切实际的，但是您可以将 *keepalive* 设置为适合您的节点和 HAFAILOVER 功能的时间段。

在客户端计算机上设置 TCP *keepalive*。根据您使用的操作系统，相应的值将有所不同。有关详细信息，请参见客户端操作系统文档。

如果您要测试客户端超时，请将表 2-1 前两列中的参数值设置为几分钟，将第三列中的参数值设置为较低的值。

表 2-1: 设置 TCP keepalive

操作系统	参数在探查连接之前等待的时间量	参数探查之间的时间量	在断开连接之前参数探查连接的最大时间量或最多尝试次数
Solaris	不适用	tcp_keepalive_interval 以毫秒计	不适用
Linux	tcp_keepalive_time 以秒为计	tcp_keepalive_intvl 以秒为计	tcp_keepalive_probes 以绝对数值计
Windows XP	KeepAliveTime 以秒为计	KeepAliveInterval 以秒为计	TCPMaxDataRetransmissions 以绝对数值计
HP-UX	tcp_time_wait_interval 以毫秒计	tcp_keepalive_interval 以毫秒计	tcp_keepalive_kill 以毫秒计

## 设置客户端的超时值

可以为 Client-Library 程序连接设置两个不同的超时属性:

- CS\_LOGIN\_TIMEOUT — 确定客户端在连接到无法访问的主机时等待的时间。
- CS\_TIMEOUT — 确定客户端等待命令完成的时间。

根据您的配置超时事件的方式, 客户端可能出现故障, 也可能故障切换到另一个节点。

可以配置客户端以设置 Client-Library CS\_TIMEOUT 参数, 以确定超时之前等待的时间。

必须设置 CS\_TIMEOUT 和 CS\_LOGIN\_TIMEOUT 参数或 isql -t 和 -l 参数, 客户端才能在节点突然断电时进行故障切换。

有关 Client-Library 参数 CS\_TIMEOUT 和 CS\_LOGIN\_TIMEOUT 的详细信息, 请参见《Client-Library/C 参考手册》。有关 CS\_HAFAILOVER 的信息, 请参见第 22 页的“客户端/服务器交互”。

有关使用 isql -t 和 -l 参数的信息, 请参见 Adaptive Server Enterprise Utility Guide (《Adaptive Server Enterprise 实用程序指南》)。



本章讨论如何在集群环境中配置安全性。

主题	页码
<a href="#">在集群环境中使用加密列</a>	35
<a href="#">在集群环境中使用 SSL</a>	35
<a href="#">将 LDAP 用作目录服务</a>	37

有关审计的信息，请参见第 293 页的“添加审计”。

## 在集群环境中使用加密列

您不能：

- 在本地临时数据库中创建加密密钥
- 在本地临时数据库中设置系统加密口令

请参见《加密列用户指南》。

## 在集群环境中使用 SSL

Cluster Edition 允许目录服务条目中指定的服务器名不同于 SSL 服务器证书用于执行 SSL 握手的公用名。这样，您将能够使用完全限定域名来作为 SSL 证书公用名（例如 *server1.bigcompany.com*）并对多个服务器使用相同的证书。

若要将公用名添加到 `interfaces` 文件中，请使用以下格式：

```
ase1
  master tcp ether host_name port_number ssl="CN='common_name'"
  query tcp ether host_name port_number ssl="CN='common_name'"
ase2
  master tcp ether host_name port_number ssl="CN='common_name'"
  query tcp ether host_name port_number ssl="CN='common_name'"
ase3
  master tcp ether host_name port_number ssl="CN='common_name'"
  query tcp ether host_name port_number ssl="CN='common_name'"
mycluster
  query tcp ether host_name port_number ssl="CN='common_name'"
  query tcp ether host_name port_number ssl="CN='common_name'"
  query tcp ether host_name port_number ssl="CN='common_name'"
```

其中 `common_name` 是集群节点的完全限定域名。 `common_name` 可以包含空格。`interfaces` 文件中定义的实例可以使用也可以不使用同一公用名。

---

**注释** 只能将一个 SSL 证书添加到 `master` 数据库。由于集群中的每个实例共享同一磁盘，它们都对 SSL 服务器证书使用相同的路径。Sybase 建议所有实例都使用同一公用名。

---

例如，下面是集群 `mycluster` 的示例 `interfaces` 文件条目：

```
ase1
  master tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
  query tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
ase2
  master tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
  query tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
ase3
  master tcp ether blade3 19986 ssl="CN='ase1.big server 1.com'"
  query tcp ether blade3 19986 ssl="CN='ase1.big server 1.com'"
mycluster
  query tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
  query tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
  query tcp ether blade3 19986 ssl="CN='ase1.big server 1.com'"
```

## 使用 sp\_listener 指定公用名

sp\_listener 包括 CN=*common\_name* 参数，从而使您能够为 SSL 证书指定公用名。语法为：

```
sp_listener 'command', '[protocol:]machine_name:port_number:
"CN=common_name", 'engine_number'
```

其中，只有当将 ssltcp 指定为协议时，才使用 CN=*common\_name*。如果包括了指定的 *common\_name*，则使用它来验证 SSL 证书中的 *common\_name*。如果未包括 CN=*common\_name*，Adaptive Server 将使用 *server\_name* 来验证 SSL 证书中的公用名。CN=*common\_name* 必须匹配证书中的公用名条目。如果在证书中包括完全限定域名，该名称必须与 CN=*common\_name* 匹配。

属性名“CN”不区分大小写，但公用名的属性值区分大小写。例如，属性名可以为“CN”、“Cn”或“cn”。

例如，下面的命令指定公用名 ase1.big server 1.com：

```
sp_listener 'start', 'ssltcp:blade1:17251:"CN=ase1.big server 1.com"', '0'
```

有关 sp\_listener 的详细信息，请参见 Adaptive Server Reference Manual（《Adaptive Server 参考手册》）。

## 将 LDAP 用作目录服务

Adaptive Server 使用目录服务通过网络建立客户端与 RPC 的连接。本章介绍有关使用 LDAP 目录服务建立连接的信息。

**轻量目录访问协议 (LDAP)** 是一种用于访问目录服务的业界标准。目录服务允许组件通过区分名 (DN)，从存储并管理整个企业或网络所使用的服务器、用户和软件信息的 LDAP 服务器上查找信息。

LDAP 服务器所在的平台可能与运行 Adaptive Server 或客户机的平台不同。LDAP 用于定义通信协议以及客户端与服务器之间交换的消息内容。消息是各种操作码，如客户端读、写和查询请求以及服务器响应等，其中包括数据格式信息。

LDAP 服务器存储和检索以下信息：

- Adaptive Server，例如 IP 地址、端口号和网络协议
- 安全性机制和过滤器
- 高可用性协同服务器名

可将 LDAP 服务器配置成具有下列访问限制：

- 匿名验证 — 所有数据对任何用户均可见。
- 用户名和口令鉴定 — Adaptive Server 使用 UNIX 平台的用户名和口令：
  - 在 32 位平台上使用 `$$SYBASE/$$SYBASE_OCS/config/libtcl.cfg`
  - 在 64 位平台上使用 `$$SYBASE/$$SYBASE_OCS/config/libtcl64.cfg`

用户名和口令鉴定属性用于建立和结束与 LDAP 服务器的会话连接。

**注释** 传递给 LDAP 服务器进行用户鉴定的用户名和口令与访问 Adaptive Server 时所用的截然不同。

在 `libtcl.cfg` `libtcl64.cfg` 或 `libtcl64.cfg` 文件（统称为 `libtcl*.cfg` 文件）中指定 LDAP 服务器后，使用 `libtcl*.cfg` 文件中的目录服务排序列表搜索服务器信息。如果在那里没有找到该信息，则搜索 `interfaces` 文件。

对于 Cluster Edition，可以在仲裁文件中设置 `interfaces` 文件。当仲裁文件指定 `interfaces` 文件时，Cluster Edition 忽略 `libtcl*.cfg` 文件中指定的目录服务。

如果服务器支持多目录服务，则应在 `libtcl*.cfg` 中指定搜索它们的顺序，不能用 `dataserver` 命令行选项指定搜索顺序。请参见第 44 页的“多目录服务”。

## LDAP 目录服务与 Sybase interfaces 文件

LDAP 驱动程序能实施目录服务，使其与 LDAP 服务器协同工作。LDAP 目录是一种基础结构，可提供：

- 一种基于网络的、可替代传统 Sybase interfaces 文件的方式
- 信息的单一层次视图，包括用户、软件、资源、网络、文件等

表 3-1 重点介绍了 Sybase interfaces 文件与 LDAP 服务器的差别。

**表 3-1: interfaces 文件与 LDAP 目录服务**

interfaces 文件	目录服务
因平台而异	与平台无关
每次 Sybase 安装各不相同	集中式和层次化
包含单独的主条目和查询条目	每个服务器对应有一个条目，客户端和服务端均可访问
无法存储有关服务器的元数据	可存储有关服务器的元数据

LDAP 目录服务比 Sybase interfaces 文件支持的属性更多。这些属性可能包括服务器版本、服务器状态等。有关属性列表，请参见表 3-2。

**注释** 只能通过重入库支持 LDAP。当使用 LDAP 目录服务连接到服务器时，必须使用 `isql_r`，而不能使用 `isql`。

表 3-2 列出了 Sybase LDAP 目录条目。

**表 3-2: Sybase LDAP 目录定义**

属性名	值类型	说明
ditbase	<i>interfaces</i> 文件或 <i>libtcl.cfg</i>	对象树的 DIT 基址。如果指定了 <i>libtcl.cfg</i> 文件，则忽略 <i>interfaces</i> 文件。对于指定的连接，可以使用 <code>ct_con_prop()</code> 替换 <i>libtcl.cfg</i> 文件。
dn	字符串	区分名。必须是可标识对象的唯一名称。
sybaseVersion	整数	服务器版本号。
sybaseServername	字符串	服务器名。
sybaseService	字符串	服务类型: Sybase Adaptive Server、Sybase SQL Server 或 ASE。
sybaseStatus	整数	状态: 1 = 活动, 2 = 停止, 3 = 失败, 4 = 未知。
sybaseAddress	字符串	每个服务器地址包括: <ul style="list-style-type: none"> <li>• 协议: TCP、NAMEPIPE、SPX DECNET (条目区分大小写)。</li> <li>• 地址: 此协议类型的任何有效地址。</li> </ul> <hr/> <p><b>注释</b> <code>dscp</code> 将此属性拆分为传输类型和传输地址。</p> <hr/> <ul style="list-style-type: none"> <li>• 过滤器: “无”、<code>ssl</code> 或 <code>ssl=“CN=common_name”</code>。</li> </ul>
sybaseSecurity (可选)	字符串	安全性 OID (对象 ID)。
sybaseRetryCount	整数	此属性映射到 <code>CS_RETRY_COUNT</code> ，用于指定 <code>ct_connect</code> 重试与服务器名相关联的网络地址序列的次数。
sybaseRetryDelay	整数	此属性映射到 <code>CS_LOOP_DELAY</code> ，用于指定延迟秒数，它是 <code>ct_connect</code> 在重试整个地址序列前等待的时间。
sybaseHAservername (可选)	字符串	用于故障切换保护的辅助服务器。

针对 TCP 连接和故障切换计算机的传统 `interfaces` 文件如下:

```
looeey
  master tcp ether huey 5000
  query tcp ether huey 5000
  hafailover secondary
```

针对 TCP 和故障切换计算机的 LDAP 条目示例如下：

```
dn: sybaseServername=foobar, dc=sybase, dc=com
objectClass: sybaseServer
sybaseVersion: 1500
sybaseServername: foobar
sybaseService: ASE
sybaseStatus: 4
sybaseAddress: TCP#1#foobar 5000
sybaseRetryCount: 12
sybaseRetryDelay: 30
sybaseHAServernam: secondary
```

LDAP 目录服务中的所有条目都称为实体。每个实体有一个区分名 (DN)，存储在基于其 DN 的分层树结构中。此树称为**目录信息树 (DIT)**。客户端应用程序使用 DIT 基址指定实体的存储位置。请参见第 41 页的“libtcl\*.cfg 文件”。

在上面的示例中，该条目描述的是名为“foobar”的 Adaptive Server，它在 5000 号端口监听 TCP 连接。位于 TCP 和 5000 之间的值 1 表示该条目同时用于 QUERY 和 MASTER 条目。对于 LDAP 目录服务，此值始终应当为 1。该实体还指定重试计数 12（次）和重试延迟 30（秒）。在客户端找到服务器响应的地址后，客户端与服务器间即开始登录对话。

可以在 UNIX `$$SYBASE/$$SYBASE_OCS/config` 中找到 Sybase LDAP 目录架构的完整列表。

在同一目录中，还有一个名为 `sybase-schema.conf` 的文件，它包含相同的架构，但使用的是 Netscape 专用语法。

由于 LDAP 对每个属性支持多个条目，因此每个地址属性必须包含单个服务器的地址，其中包括协议、访问类型和地址。请参见表 3-2 中的 `sybaseAddress`。

例如，下面是一个 Windows 服务器的 LDAP 条目，该服务器使用不同的连接协议在两个地址上进行监听：

```
sybaseAddress = TCP#1#TOEJAM 4444
sybaseAddress = NAMEPIPE#1#\pipe\sybase\query
```

---

**注释** 地址字段中的每个条目用 # 字符分隔。

---

可用 `dsedit` 编辑这些条目。请参见第 43 页的“向目录服务添加服务器”。

为确保所有 Sybase 产品的跨平台兼容性，协议和地址属性字段应采取独立于平台和产品的格式。

## *libtcl\*.cfg* 文件

可使用 *libtcl\*.cfg* 文件指定 LDAP 服务器名称、端口号、DIT 基址、用户名和口令，以鉴定到 LDAP 服务器的连接。

*libtcl\*.cfg* 文件旨在为 Open Client/Open Server 和基于 Open Client/Open Server 的应用程序提供配置信息，如驱动程序、目录和安全性服务。为获得配置信息，32 位实用程序（如 *dsedit*）会查阅 *libtcl.cfg*，而 64 位应用程序则使用 *libtcl64.cfg* 文件。

应对 *libtcl.cfg* 和 *libtcl64.cfg* 文件进行编辑，以确保 32 位与 64 位应用程序间的兼容性。

缺省的 *libtcl.cfg* 文件位于 `$$SYBASE/$$SYBASE_OCS/config` 中。

如果在 *libtcl.cfg* 文件中指定了 LDAP，则不使用 *interfaces* 文件。

---

**注释** 启动时，使用 `-l` 选项的 Open Client/Open Server 应用程序将替换 *libtcl.cfg* 文件，并使用 *interfaces* 文件。

---

最简单的 *libtcl.cfg* 文件格式如下：

```
[DIRECTORY]
ldap=libsybdlldap.dll ldapurl
```

其中 *ldapurl* 定义为：

```
ldap://host:port/ditbase
```

使用这些相同属性的下列 LDAP 条目是匿名连接，且只有在 LDAP 服务器允许只读访问时才有效。

```
ldap=libsybdlldap.dll ldap://seashore/d=sybase,dc=com
```

可在 *libtcl.cfg* 文件中指定用户名和口令作为 LDAP URL 的扩展，以便在连接时启用口令鉴定。

## 启用 LDAP 目录服务

若要使用目录服务，您必须：

- 1 根据供应商提供的文档配置 LDAP 服务器。
- 2 将 LDAP 库的位置添加到所用平台的 Unix 装载库路径变量中。

3 配置 *libtcl.cfg* 文件以使用目录服务。

使用任何标准的 ASCII 文本编辑器进行如下操作：

- 在 *libtcl.cfg* 文件的 *[DIRECTORY]* 条目下，从 LDAP URL 行的开头删除分号 (;) 注释标记。
- 在 *[DIRECTORY]* 条目下添加 LDAP URL。有关支持的 LDAP URL 值，请参见表 3-3。

---

**警告！** LDAP URL 必须放在一行内。

---

```
file libtcl.cfg:
ldap=libsybdlldap.so ldap://host:port/ditbase??scope??bindname=
username?password
```

```
file libtcl64.cfg
ldap=libsybdlldap64.so
ldap://host:port/ditbase??scope??bindname=username?password
```

例如：

```
[DIRECTORY]
ldap=libsybdlldap.so ldap://huey:11389/dc=sybase,dc=com??one??
bindname=cn=Manager,dc=sybase,dc=com?secret
```

“one” 表示搜索范围为在 DIT 基址的下一级中检索条目。

表 3-3 定义了 *ldapurl* 变量的关键字。

**表 3-3: *ldapurl* 变量**

关键字	说明	缺省值
<i>host</i> (必需)	运行 LDAP 服务器的计算机的主机名或 IP 地址	无
<i>port</i>	LDAP 服务器监听的端口号	389
<i>ditbase</i> (必需)	缺省 DIT 基址	无
<i>username</i>	要鉴定的用户区分名 (DN)	NULL (匿名鉴定)
<i>password</i>	要鉴定的用户口令	NULL (匿名鉴定)

- 4 检验相应环境变量是否指向必需的第三方库。Netscape LDAP SDK 库位于 *\$\$SYBASE/\$\$SYBASE\_OCS/lib3p* 或 *lib3p64* 中。Unix 装载库路径环境变量必须指向此目录。
- 5 使用 *dscp* 或 *dsedit* 向 LDAP 服务器添加服务器条目。请参见第 43 页的“向目录服务添加服务器”。

## 向目录服务添加服务器

---

**警告！** 大多数 LDAP 服务器都有 `ldapadd` 实用程序，用于添加目录条目。Sybase 建议您改用 `dsedit`，因为它具有一般工具不提供的内置语义检查功能。

---

每一服务器条目都由一组属性构成。添加或修改服务器条目时，系统将提示有关服务器属性的信息。某些属性提供缺省值，其它属性则需要用户输入。提供缺省值时，缺省值会出现在方括号 “[ ]” 中。有关所接受的值的信息，请参见第 39 页的表 3-2。

### ❖ 使用 `dsedit` 向目录服务添加服务器条目

添加、删除或修改 LDAP 服务器条目前，必须要向 `libtcl.cfg` 文件添加 LDAP URL。请参见第 41 页的“`libtcl*.cfg` 文件”。

使用 `dsedit` 向目录服务添加服务器：

- 1 执行 `SYBASE.csh` 或 `SYBASE.sh` 来设置环境变量。
- 2 执行 `cd`，定位到 `$SYBASE/$SYBASE_OCS/bin`。
- 3 执行 `dsedit`。
- 4 从服务器列表中选择“LDAP”，并单击“确定”。
- 5 单击“添加新的服务器条目”。
- 6 请输入：
  - 服务器名称 — 这是必需的。
  - 安全性机制 — 可选。这是高可用性故障切换服务器的名称（如果有）。
- 7 单击“添加新的网络传输”，然后：
  - 选择传输类型。
  - 输入主机名。
  - 输入端口号。
  - （可选）输入 SSL 过滤字符串。

8 单击两次“确定”，退出 `dsedit`。

若要查看服务器条目，请在 Netscape `http://host:port/ditbase??one` 中输入以下 URL。

例如：

```
ldap://huey:11389/dc=sybase,dc=com??one
```

---

**注释** Microsoft Internet Explorer 不识别 LDAP URL。

---

有关 `dscp` 的详细信息，请参见位于 <http://www.sybase.com/support/manuals> 的“11.1.x 一般集合”（11.1.x Generic Collection）上的 Open Client/Server Configuration Guide（《Open Client/Server 配置指南》）。

## 多目录服务

任何类型的 LDAP 服务都称为 LDAP 服务器，无论它是实际的服务器还是其它 LDAP 服务的网关。

可为高可用性故障切换保护指定多目录服务。并非列表中的每个目录服务都必须是 LDAP 服务器。

例如：

```
[DIRECTORY]
ldap=libsybldap.so ldap://test:389/dc=sybase,dc=com
ldap=libsybldap.so ldap://huey:11389/dc=sybase,dc=com
```

在本例中，如果到 `test:389` 的连接失败，则此连接会故障切换到尝试 `huey:11389` 的 LDAP 服务器。不同的供应商使用不同的 DIT 基址格式。

---

**注释** 有关详细信息，请参见位于 <http://www.sybase.com/support/manuals> 上的《Open Client Library/C 程序员指南》和《Open Client Library/C 参考手册》。

---

## 加密口令

*libtcl.cfg* 文件中的条目为人工可读格式。Sybase 为基本口令加密提供了 *pwdcrypt* 实用程序。*pwdcrypt* 使用一种简单算法，当该算法应用于键盘输入时，它生成能代替口令的加密值。*pwdcrypt* 位于 *\$\$SYBASE/\$\$SYBASE\_OCS/bin* 中。

从 *\$\$SYBASE/\$\$SYBASE\_OCS* 目录，输入：

```
bin/pwdcrypt
```

按系统提示输入两次口令。

*pwdcrypt* 生成加密的口令。例如：

```
0x01312a775ab9d5c71f99f05f7712d2cded2i8d0ae1ce78868d0e8669313d1bc4c706
```

使用任何标准的 ASCII 文本编辑器将加密的口令复制并粘贴到 *libtcl.cfg* 文件。加密前，文件条目显示为：

```
ldap=libsybldap.so
ldap://seashore/dc=sybase,dc=com??one??bindname=uid=Manager,dc=sybase,
dc=com?password
```

使用加密字符串替代口令：

```
ldap=libsybldap.so
ldap://seashore/dc=sybase,dc=com??one??bindname=uid=Manager,dc=sybase,dc=com?
0x01312a775ab9d5c71f99f05f7712d2cded2i8d0ae1ce78868d0e8669313d1bc4c706
```

---

**警告！** 即使口令已被加密，您仍要使用文件系统安全性对其进行保护。

---

## 性能

使用 LDAP 服务器时的执行速度可能比使用 *interfaces* 文件时的执行速度慢，这是因为 LDAP 服务器需要时间进行网络连接和检索数据。由于这种连接是在启动 Adaptive Server 时建立的，因此在登录时就能看出性能的差别。在常规系统负载期间，延迟并不明显。而当系统负载很高且有很多连接时，尤其是在有持续时间短的重复连接时，使用 LDAP 服务器与使用传统 *interfaces* 文件相比，整体性能上的差异就会相当明显。

## 从 *interfaces* 文件迁移到 LDAP

没有直接的方法可将使用 *interfaces* 文件的现有服务器升级到使用轻量目录服务的服务器。若要将早期版本的 Adaptive Server 升级到 Adaptive Server 15.0 版，请参见您的平台的安装指南。

将服务器升级后，即可对服务器进行配置，以使用 LDAP 服务。

- 1 关闭服务器。有关启动和停止集群的信息，请参见《集群用户指南》。有关共享内存服务器的信息，请参见 *Configuration Guide for Adaptive Server 15.0*（《Adaptive Server 15.0 配置指南》）第 2 章“Starting and Stopping Servers”（启动和停止服务器）。
- 2 编辑 `$$SYBASE/$SYBASE_OCS/config/libtcl.cfg` 或 `libtcl64.cfg` 文件，以添加目录服务。请参见第 41 页的“启用 LDAP 目录服务”。
- 3 使用 `dsedit` 或 `dscp` 将集群服务器的服务器、集群和实例条目添加到目录服务中。请参见第 43 页的“向目录服务添加服务器”。
- 4 使用 `qrmutil` 验证集群仲裁文件中针对集群和实例定义的接口目录属性是否为空。例如，若要显示仲裁文件中的值，请输入：

```
$$SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=path_to_your_quorum --  
display=config
```

如果在任何实例上为 *interface\_dir* 属性定义了路径，或者为集群定义了路径，则必须重置该值。使用路径值指定此属性会强制实例使用 *interfaces* 文件和覆盖 `libtcl.cfg` 和 `libtcl64.cfg` 文件中的信息。

例如，使用这些 `qrmutil` 命令重新设置 *interface\_dir* 属性的值。*interface\_dir* 的值是两个单引号，意味着它是一个空字符串。

```
$$SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=path_to_your_quorum --  
interface_dir=''
```

```
$$SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=path_to_your_quorum --  
instance=name_of_instance_to_reconfig --interface_dir=''
```

- 5 重新启动服务器或集群。

## 将 LDAP 目录服务与共享磁盘集群一同使用

Cluster Edition 可以使用 LDAP 目录服务指定其集群和实例条目。您必须为集群仲裁文件中的 `interface_dir` 属性指定空字符串。不要使用 `dataserver` parameter `-i interfaces_path` 指定 `interfaces` 文件的路径。

如果您未指定 `interface_dir` 的值，则 Cluster Edition 使用 `$$SYBASE/OCS-15_0/config/libtcl64.cfg`（对于 64 位服务器和客户端）或 `$$SYBASE/OCS-15_0/config/libtcl.cfg`（对于 32 位服务器和客户端）中定义的目录服务排序列表。服务器搜索 `libtcl64.cfg` 中定义的目录服务后，搜索缺省位置中的 `interfaces` 文件。

Open Client 应用程序可以使用 LDAP 目录服务来存储集群和实例服务器条目。例如，对于具有两个实例（“ase1”和“ase2”）的名为“mycluster”的集群，`interfaces` 文件类似如下：

```
ase1
    master tcp ether blade1 10945
    query tcp ether blade1 10945
ase2
    master tcp ether blade2 10955
    query tcp ether blade2 10955
mycluster
    query tcp ether blade1 10945
    query tcp ether blade2 10955
```

必须使用 `dsedit` 或 `dscp` 将服务器名称“ase”、“ase2”和“mycluster”的等效 LDAP 目录服务条目添加到 LDAP 目录服务。请参见第 43 页的“向目录服务添加服务器”。有关 `dsedit` 和 `dscp` 的详细信息，请参见《实用程序指南》。

客户端可以使用集群名称（在此例中为“mycluster”）或特定于实例的服务器名称（“ase1”或“ase2”）连接到集群中的任何实例。

当客户端使用 SSL 连接到使用 SSL 的非集群 Adaptive Server 时，将在 `interfaces` 文件中的端口号后面放置 SSL 过滤器。目录服务包括您使用 `dsedit` 或通过手动编辑添加的公用名。通常，对整个集群使用一个具有公用名的 SSL 证书，而不是对每个实例使用一个证书。请参见第 35 页的“在集群环境中使用 SSL”。

下面的示例向集群 “mycluster” 的 interfaces 文件条目中添加 SSL 过滤器：

```
mycluster
```

```
query tcp ether blade1 10945 ssl="cn=mycluster.domain.com"  
query tcp ether blade2 10955 ssl="cn=mycluster.domain.com"
```

添加到 LDAP 目录服务中的条目必须指定具有 SSL 过滤器的公用名：  
ssl="cn=mycluster.domain.com"。

例如，下面的 dscp 会话为集群 “mycluster” 添加上述示例条目：

```
% dscp  
  
>> open ldap  
  
ok  
  
Session 1 ldap>> add mycluster  
Service: [ASE]  
Transport Type: [tcp]  
Transport Address: blade1 10945 ssl="cn=mycluster.domain.com"  
Transport Type: [tcp]  
Transport Address: blade2 10955 ssl="cn=mycluster.domain.com"  
Transport Type: [tcp]  
Transport Address:  
Security Mechanism [] :  
HA Failoverserver:  
Retry Count:  
Retry Delay:  
Added mycluster  
  
Session 1 ldap>> read mycluster  
  
DIT base for object: dc=domain,dc=com  
Distinguish name: sybaseServername=mycluster, dc=domain,dc=com  
Server Entry Version: 15001  
Server Name: mycluster  
Server Service: ASE  
Server Status: 4 (Unknown)  
Server Address:  
Transport Type: tcp  
Transport Address: yellowstar 2521 ssl="cn=mycluster.domain.com"  
Transport Type: tcp  
Transport Address: yellowstar 2525 ssl="cn=mycluster.domain.com"  
  
Session 1 ldap>> quit
```

## 在集群环境中使用监控表

本章列出集群式 Adaptive Server 的监控表，并讲述如何配置和管理它们。有关这些表的列和数据类型说明，请参见《参考手册：表》。

主题	页码
<a href="#">集群的监控表</a>	49

### 集群的监控表

在集群环境中，监控表是以实例为单位进行报告的，而不是返回全集群范围的结果。因此，您可以在整个集群内监控进程和查询的活动，更好地掌握可能在多个实例上打开的对象的统计信息以及集群中每个实例上的资源使用情况。例如，如果您在监控表中查询有关某个表的信息，而该表可能被集群中的多个实例打开或访问，因此，该表的描述符（以及关联的统计信息）可能位于实例上的内存中。统计信息不是按集群进行聚合的。而是会返回一个由所有实例的统计结果组成的联合结果集，其中的各行是分别从每个实例中收集的。结果集中的每个实例都是用所在行的 InstanceID 列标识的。

## 配置系统视图

`system_view` 是特定于会话的设置，允许您控制查询从监控表、`sysprocesses`、`sp_who` 和其它命令返回的监控数据的作用域。如果将 `system_view` 设置为 `cluster`，监控表查询会返回集群中所有活动实例中的数据。如果将 `system_view` 设置为 `instance`，监控表查询会返回客户端所连接的实例上的活动进程或对象的数据。

使用 `set` 命令可以配置会话的作用域：

```
set system_view {instance | cluster | clear}
```

其中：

- `instance` — 仅返回本地实例的统计信息。跨集群请求不发送到集群中的任何其它实例。
- `cluster` — 返回集群中所有实例的统计信息。
- `clear` — 将系统视图返回到所配置的缺省设置。

以下示例将修改会话设置，以使监控表查询仅返回客户端所连接的实例的数据：

```
set system_view instance
```

以下示例将修改会话设置，以使监控表查询返回集群的数据：

```
set system_view cluster
```

以下示例会清除系统视图的当前设置，并将系统视图返回到缺省设置。

```
set system_view clear
```

如果您在查询监控表或调用监控表 RPC 时不指定 `InstanceID`，实例就会使用当前的 `system_view` 配置。

会话的系统视图是从其主机逻辑集群继承的。选择 `@@system_view` 全局变量可以确定当前系统视图。

## 配置监控表

使用监控表的配置参数可以确定全集群范围的行为或仅实例的行为。缺省情况下，所有监控表配置值适用于全集群范围。

## 管理消息管道

以下参数确定集群和实例如何管理用于存储历史监控表数据的内存：

- `deadlock pipe max messages`
- `errorlog pipe max messages`
- `sql text pipe max messages`
- `plan text pipe max messages`
- `statement pipe max messages`

这些参数可以在集群范围内全局配置，也可以针对每个实例单独配置。这些参数为管道分配内存。实例可以动态地向管道添加内存，但不能动态地从管道删除内存，因此，如果您减少参数的大小，就必须重新启动实例，新的管道大小才能生效。

下面是一些用于确定参数大小的算法。

- 对于各个实例，每个管道配置需要的内存为：  
 $configuration\_value \times number\_of\_engines$
- 若要为每个管道配置全局设置内存：  
 $configuration\_value \times number\_of\_engines \times number\_of\_instances$
- 如果已经针对每个实例分别设置了管道配置的值，则集群需要的内存量为：  
 $(instance\_1\_value \times number\_of\_engines) + (instance\_2\_value \times number\_of\_engines) + \dots + (instance\_n\_value \times number\_of\_engines)$

## RPC 的更改

如果您调用 RPC，但不包含 `InstanceID` 参数，监控表就会使用系统视图设置来确定如何报告统计信息。如果您将系统视图设置为 `instance`，则所有收集的数据都是本地的。如果您将系统视图设置为 `cluster`，则监控表与形成集群实例的实例进行通信，而不是与逻辑集群通信。

## 特定于 Cluster Edition 的监控表

某些监控表提供有关集群式和非集群式 Adaptive Server 的信息。表 4-1 列出了专门提供集群式 Adaptive Server 的信息的监控表。

**表 4-1: 专门针对 Cluster Edition 的监控表**

monClusterCacheManager	monTempdbActivity
monPCM	monSysLoad
monDBRecovery	monLogicalCluster
monDBRecoveryLRTypes	monLogicalClusterinstance
monCMSFailover	monLogicalClusterRoute
monFailoverRecovery	monLogicalClusterAction
monCLMObjectActivity	monProcessMigration
monCIPC	monWorkloadProfile
monCIPCEndpoints	monWorkloadRaw
monCIPCLinks	monWorkloadPreview
monCIPCMesh	

## 对所有实例返回相同信息的监控表

表 4-2 给出了对所有实例返回相同信息的监控表。

**表 4-2: 对所有实例包含相同信息的监控表**

表名	说明
monMon	元数据视图在所有实例上都是相同的。
monTableColumns	元数据视图在所有实例上都是相同的。
monTableParameters	元数据视图在所有实例上都是相同的。
monTables	元数据视图在所有实例上都是相同的。
monWaitClassInfo	说明列表在所有实例上都是相同的。
monWaitEventInfo	说明列表在所有实例上都是相同的。

## 返回特定实例信息的监控表

表 4-3 列出了包含 InstanceID 列的监控表。

**表 4-3: 包含 InstanceID 列的监控表**

monCachePool	monDataCache
monCachedProcedures	monDeviceIO
monDeadLock	monErrorLog
monEngine	monIOQueue
monLicense	monLocks
monOpenDatabases	monNetworkIO
monOpenPartitionActivity	monOpenObjectActivity
monProcess	monProcedureCache
moProcessLookup	monProcessActivity
monProcessObject	monProcessNetIO
monProcessSQLText	monProcessProcedures
monProcessWaits	monProcessStatement
monResourceUsage	monProcessWorkerThread
monSysPlanText	monState
monSysStatement	monSysSQLText
monSysWorkerThread	monSysWaits
monCachedObject	



本章讨论有关将 Backup Server 用于集群的问题。

主题	页码
转储期间加入集群的节点	55
多个 Backup Server	56

## 转储期间加入集群的节点

早期版本的 Cluster Edition 不允许节点在数据库或事务日志转储期间加入或离开集群。转储期间启动的任何实例均需等到转储结束后才能完成启动和关闭。

对于 Cluster Edition 15.0.3 及更高版本，节点（以及其上运行的实例）可以在另一个实例运行 `dump database` 或 `dump transaction` 时加入或离开集群。

只有在 `dump database` 的第一阶段，节点才能加入和离开集群。在 `dump database` 命令执行期间将显示阶段：

```
Backup Server: 3.43.1.1: Dump phase number 1 completed.  
Backup Server: 3.43.1.1: Dump phase number 2 completed.  
Backup Server: 3.43.1.1: Dump phase number 3 completed.
```

如果节点尝试在 `dump database` 的第二阶段或第三阶段加入或离开，则它将等到 `dump database` 完成才能加入。

正常关闭后，节点可以在数据库转储期间正常离开集群。

一旦为集群配置了多个 Backup Server，则不需要执行任何附加配置就可以让节点在 `dump database` 或 `dump transaction` 期间加入或离开集群。

## 多个 Backup Server

Cluster Edition 15.0.3 版之前的版本对集群使用单一 Backup Server，对于 dump 和 load 命令而言，这是一个潜在的瓶颈。

Cluster Edition 15.0.3 及更高版本允许集群通过下列方法之一使用多个 Backup Server:

- 专用方法 — 为每个实例指派了特定 Backup Server。
- 循环方法 — 在执行 dump 或 load 命令时，Cluster Edition 根据可用性从组中为实例指派 Backup Server。
- 名为 SYB\_BACKUP 的单一 Backup Server。

Cluster Edition 将有关所有 Backup Server 的信息存储在 syssservers 中。当您从任何实例发出 dump 或 load 命令时，Cluster Edition 在 syssservers 中搜索 Backup Server 名称条目 SYB\_BACKUP。如果它找到指向 srvnetname 条目 \$DEDICATED 或 \$ROUNDROBIN 的 SYB\_BACKUP 条目，则使用相应的 Backup Server 来执行 dump 或 load 操作。

例如，“snack” 集群配置为使用专用方法，并包含名为 “cupcake” 的实例。由于 SYB\_BACKUP 条目指向 “\$dedicated”，该集群使用 cupcake\_BS Backup Server 执行任何转储或装载：

srvnid	srvstatus	srvname	srvnetname	srvclass	srvsecmech
srvcoss		srvstatus2			
1	8	SYB_BACKUP	\$dedicated	7	NULL
	NULL	2			
2	8	cupcake	cupcake	7	NULL
	1000	4			
3	8	cupcake_BS	cupcake_BS	7	NULL
	0	2			
4	8	cookie	cookie	7	NULL
	1000	4			
5	8	cookie_BS	cookie_BS	7	NULL
	0	2			

如果为循环方法配置了 snack 集群，则 cupcake 实例将使用第一个可用的 Backup Server 来执行转储和装载。

当您升级时，Cluster Edition 将所有 Backup Server 的条目添加到 syssservers。

当您从集群中删除实例时，Backup Servers 的条目会从 syssservers 中删除。

## 将 Cluster Edition 配置为使用多个 Backup Server

使用 `sp_addserver` 添加附加 Backup Server:

```
sp_addserver instance_name_BS, NULL, instance_name
```

例如，以下命令将 `cookie_BS` 条目添加到 `syssservers`:

```
sp_addserver cookie_BS, NULL, cookie
```

在启动 Backup Server 之前，必须为每个 Backup Server 插入 `interfaces` 文件条目。有关编辑 `interfaces` 文件的信息，请参见 Configuration Guide (《配置指南》) 中的第 5 章 “Setting Up Communications Across the Network” (设置网络通信)。

### 使用专用方法

要针对专用方法配置 Cluster Edition，请使用：

```
sp_addserver 'SYB_BACKUP', NULL, '$DEDICATED'
```

集群中为专用 Backup Server 方法配置的每个实例都必须包含 Backup Server。

当用户使用 `isql` 连接到实例 “cookie” 并发出 `dump` 或 `load` 命令时，实例 `cat` 在 `syssservers` 中查找指向 “\$DEDICATED” 关键字的 `SYB_BACKUP` 条目。如果指定的 Backup Server (“cookie\_BS”) 未运行，则 Cluster Edition 自动重新启动此 Backup Server。

### 使用循环方法

要针对循环方法配置 Cluster Edition，请使用：

```
sp_addserver 'SYB_BACKUP', NULL, '$ROUNDROBIN'
```

一旦您将 Cluster Edition 配置为使用循环方法，Cluster Edition 就会根据哪个 Backup Server 执行最少工作量，为 `dump` 或 `load` 命令选择 Backup Server。

例如，如果用户从 `isql` 连接到实例 `Inst1` 并从此会话发出 `dump` 命令，实例 `Inst1` 就会在 `syssservers` 中检查 `SYB_BACKUP` 条目，并发现它指向关键字 `$ROUNDROBIN`。如果 `Inst1_BS` 忙于执行另一个任务，则 Cluster Edition 移动到下一个 Backup Server `Inst2_BS`。如果 `Inst2_BS` 可用，且它上面没有任何作业正在运行，则此 Backup Server 执行转储，其索引递增以指示它正在忙，无法进行工作。但是，如果 `Inst2_BS` 繁忙，则 Cluster Edition 移动到下一个 Backup Server，依此类推，直到找到空闲 Backup Server。

另一方面，如果连接到 Inst2\_BS 的尝试失败，则 `enable backupserver HA` 参数控制是自动重新启动 Backup Server 还是尝试下一个可用 Backup Server。如果所有配置的 Backup Server 都不可用，则报告一个错误。

如果将 `enable backupserver HA` 设置为 `true` 并为循环策略配置了 Backup Server，则必须让所有节点能够在集群中启动任何 Backup Server。如果希望在您发出命令的初始节点出现故障时允许另一个节点执行转储或装载任务，则必须进行此配置。要使所有节点能够启动任何 Backup Server，请在 `interfaces` 文件中包含所有 Backup Server 的条目。

例如，`Big_Cluster` 使用循环策略，并包含实例 `Node_1`、`Node_2` 和 `Node_3`。实例 `Node_1` 的 `interfaces` 文件条目必须为在节点 `Node_2` 和 `Node_3` 上运行的 Backup Server 包含条目，以便循环策略在 `Node_1` 出现故障时选择这些节点之一作为替代节点：

```
Node_1_BS
    master tcp ether Node_1 5004
    query tcp ether Node_1 5004
    master tcp ether Node_2 5006
    query tcp ether Node_2 5006
    master tcp ether Node_3 5008
    query tcp ether Node_3 5008
```

```
Node_2_BS
    master tcp ether Node_1 5004
    query tcp ether Node_1 5004
    master tcp ether Node_2 5006
    query tcp ether Node_2 5006
    master tcp ether Node_3 5008
    query tcp ether Node_3 5008
```

```
Node_3_BS
    master tcp ether Node_1 5004
    query tcp ether Node_1 5004
    master tcp ether Node_2 5006
    query tcp ether Node_2 5006
    master tcp ether Node_3 5008
    query tcp ether Node_3 5008
```

另外，可以在您发出 `dump` 或 `load` 命令时所在的节点上手动启动 Backup Server。

## 启动和停止 Backup Server

当您使用 `sybcluster` 实用程序和 Adaptive Server 插件创建 Backup Server 时，它们将自动启动 Backup Server。有关使用 `sybcluster` 启动和停止 Backup Server 的信息，请参见《实用程序指南》。有关使用 Adaptive Server 插件启动 Backup Server 的信息，请参见联机帮助。

使用 `startserver` 命令手动启动 Backup Server。`startserver` 命令要求 Backup Server 具有 `runserver` 文件。这将启动 `MOUSE_BS` Backup Server:

```
startserver -f RUN_MOUSE_BS
```

有关 `startserver` 命令的详细信息，请参见《实用程序指南》。有关 `runserver` 文件的信息，请参见 *Configuration Guide*（《配置指南》）中的第 2 章“Starting and Stopping Servers”（启动和停止服务器）。

使用 `shutdown server_name` 命令关闭任何 Backup Server。例如:

```
shutdown MOUSE_BS
```

通过指定 `SYB_BACKUP` Backup Server 关闭所有活动的 Backup Server:

```
shutdown SYB_BACKUP
```

## 备份到介质

备份 Cluster Edition 对象的方法与在非集群 Adaptive Server 中备份对象的方法相同。请参见《系统管理指南：卷 2》。

## 对存储过程的更改

### `sp_addserver`

`sp_addserver` 将关键字 `$DEDICATED` 和 `$ROUNDROBIN` 添加为用于设置 Backup Server 方法的参数。请参见第 57 页的“[将 Cluster Edition 配置为使用多个 Backup Server](#)”。

## sp\_dumpoptimize and sp\_helpserver

当 `syssservers` 中的 `SYB_BACKUP` 条目指向关键字 `$DEDICATED` 或 `$ROUNDROBIN` 时，Cluster Edition 针对集群中的所有活动 Backup Server 运行这些系统过程：

- `sp_dumpoptimize`
- `sp_helpserver`

## sp\_volchanged

`sp_volchanged` 允许您包括特定 Backup Server 的名称。语法为：

```
sp_volchanged session_id, devname, action[, fname [, vname] [,  
backup_server_name]]
```

例如，下面的命令指定在 Backup Server `Inst3_BS` 上发出卷更改请求：

```
sp_volchanged 5, "dbl.dmp", "proceed", "Inst3_BS"
```

如果您不将 Backup Server 名称与 `$DEDICATED` 或 `$ROUNDROBIN` 方法包括在一起，则 `sp_volchanged` 会引发一条错误。

## 管理负载

本章介绍如何管理负载并为访问 Cluster Edition 的应用程序提供故障切换。

主题	页码
<a href="#">逻辑集群资源</a>	62
<a href="#">系统逻辑集群</a>	62
<a href="#">设置逻辑集群</a>	63
<a href="#">指派路由规则</a>	68
<a href="#">配置逻辑集群特性</a>	69
<a href="#">配置故障切换</a>	75
<a href="#">管理逻辑集群</a>	76
<a href="#">管理故障切换、故障恢复和计划的停机时间</a>	82
<a href="#">分配负载</a>	88
<a href="#">使用样本负载配置文件</a>	92
<a href="#">创建和配置您自己的负载配置文件</a>	92
<a href="#">故障排除</a>	95

Cluster Edition 负载管理器可针对每个业务应用程序自定义负载管理和故障切换，使每个应用程序能够以最有效的方式执行。逻辑集群是允许负载管理器提供个性化工作环境的容器。

逻辑集群是物理共享磁盘集群中一个或多个实例的抽象表示形式。每个逻辑集群都有一组作为其运行基础的实例，并可能有一组作为其故障切换目标的实例。路由规则基于应用程序、用户登录名或客户端提供的服务器别名将传入连接定向到特定逻辑集群。其它规则可能将逻辑集群限制到绑定的连接，或允许任何经过鉴定的连接访问逻辑集群。

通过基于物理集群创建逻辑集群，您可以使用同一系统上的不同负载对应用程序进行分区。负载是在应用程序级管理的，这意味着您可以依据每个应用程序使用系统的方式来管理传入连接、故障切换策略、负载分配策略和计划的停机时间。

系统管理员使用 Sybase Central 的 Adaptive Server 插件或本章中描述的命令行选项来管理负载。系统管理员：

- 配置和配置逻辑集群。此操作包括创建和删除逻辑集群、在集群中添加或删除实例、修改故障切换规则、启动和停止集群或集群中的实例、配置路由规则等。
- 选择或配置系统用于确定当前相关负载的负载配置文件。
- 监控集群中的实例和每个实例上的负载。

## 逻辑集群资源

将从物理集群中为逻辑集群分配以下资源：

- 实例 — 是逻辑集群的**基本实例**，这意味着它们将在逻辑集群启动时启动，并且故障切换将恢复它们。
- 故障切换资源 — 是实例的已排序列表，在一个或多个基本实例失败的情况下，逻辑集群将依据这些实例运行。物理集群中的任何实例都可以是故障切换资源。利用负载管理功能，您可以对资源进行分组和配置，以指定故障切换顺序和优先级。

## 系统逻辑集群

当您创建共享磁盘集群时，Adaptive Server 将自动为其创建一个系统逻辑集群。系统逻辑集群提供守护程序的逻辑集群表示形式，并能够管理某些任务。它表示物理集群，包含物理集群的所有实例，并具有与物理集群相同的名称。所有后台任务（例如检查点和管家）都在系统逻辑集群上运行。特殊规则将应用于系统逻辑集群。将在创建这个新的系统逻辑集群时为其授予 open 属性，这意味着会将所有未绑定的连接路由到该集群。

系统管理员通常不与系统逻辑集群交互。但是，以下各项将应用于系统逻辑集群：

- 路由规则。例如，您可以将系统管理员使用的登录名路由到系统逻辑集群。
- open 属性。
- 系统视图设置。
- 负载配置文件。
- 登录分配模式。

以下各项不适用于系统逻辑集群：

- 故障切换资源和规则。
- 执行以下操作的命令：
  - 创建或删除逻辑集群的资源
  - 对实例进行迁移、故障切换和故障恢复
  - 更改逻辑集群或实例状态
  - 更改故障切换模式设置
  - 更改启动模式设置
  - 设置向下路由模式

## 设置逻辑集群

可能有多个用于配置逻辑集群的选项；用于设置工作逻辑集群的基本步骤为：

- 1 创建逻辑集群。
- 2 添加实例。
- 3 指派路由规则。
- 4 启动逻辑集群。

在此示例中，我们将为“mycluster”物理集群创建三个逻辑集群：“SalesLC”、“HRLC”和“CatchallLC”。

## 创建逻辑集群

---

**注释** 也可以使用 Adaptive Server 插件来创建逻辑集群。请参见第 229 页的“添加逻辑集群”。

---

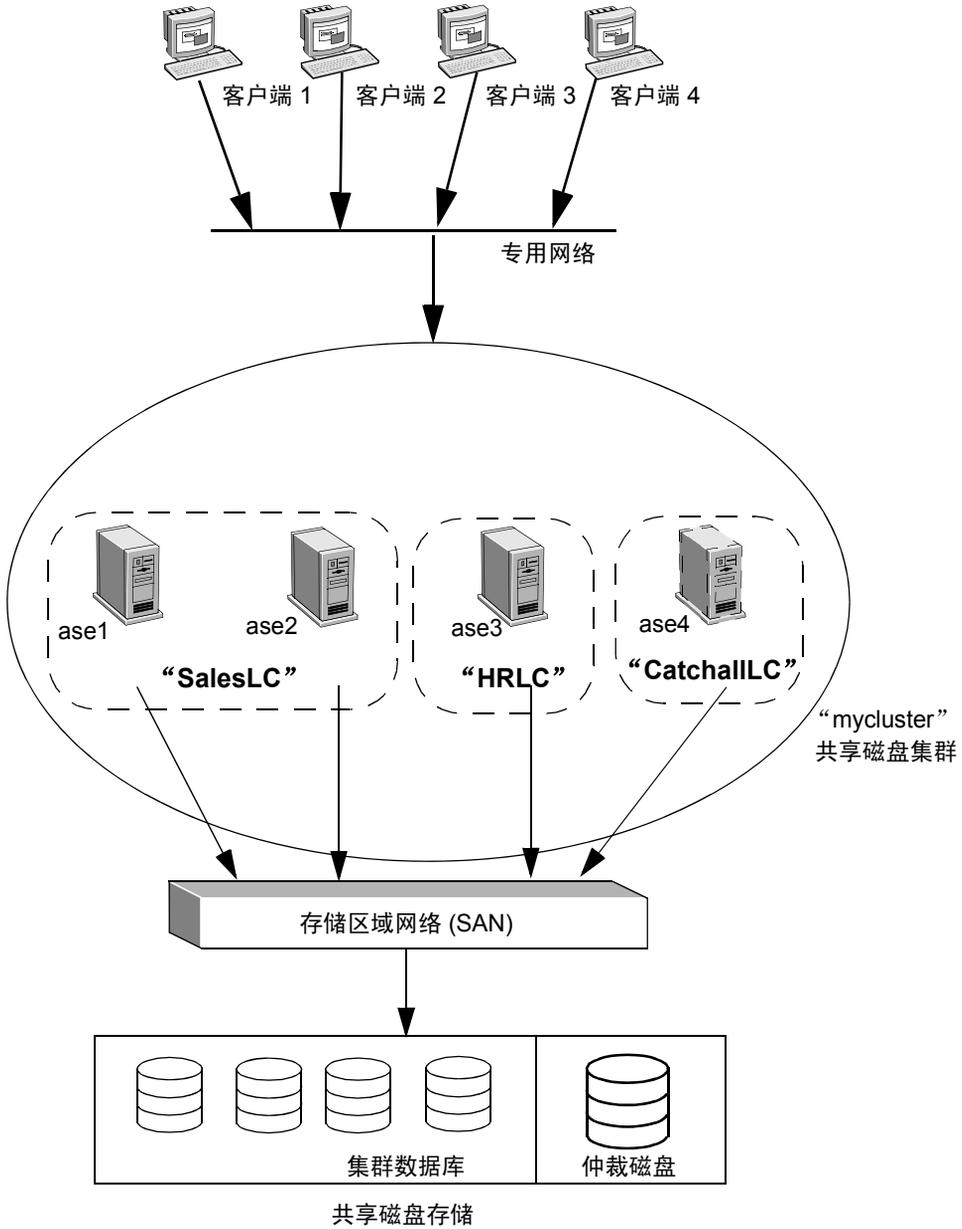
使用 `sp_cluster logical, "create"` 创建逻辑集群。

例如，假设有一个名为“mycluster”的物理集群，其中包含四个实例：“ase1”、“ase2”、“ase3”和“ase4”。我们创建三个逻辑集群：

- “SalesLC” — 用于处理销售部门中的应用程序和登录。
- “HRLC” — 用于处理人力资源部门中的应用程序和登录。
- “CatchallLC” — 稍后用于开放式逻辑集群。

若要创建“SalesLC”、“HRLC”和“CatchallLC”，请输入：

```
sp_cluster logical, "create", SalesLC
sp_cluster logical, "create", HRLC
sp_cluster logical, "create", CatchallLC
```



## 向逻辑集群中添加实例

使用 `sp_cluster logical, "add"` 向集群中添加实例。

向任务关键性的 “SalesLC” 中添加两个实例：

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

向 “HRLC” 中添加单个实例：

```
sp_cluster logical, "add", HRLC, instance, ase3
```

向 “CatchallLC” 中添加单个实例：

```
sp_cluster logical, "add", CatchallLC, instance, ase4
```

## 向逻辑集群中添加路由

使用 `sp_cluster logical, "add"` 将客户端路由到目标逻辑集群。请参见第 68 页的 “指派路由规则”。

例如，若要将应用程序 “field\_sales” 和 “sales\_reports” 路由到 “SalesLC”，请输入：

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

若要将若干销售应用程序通过 Internet 使用的登录名 “sales\_web\_user” 路由到 “SalesLC”，请输入：

```
sp_cluster logical, "add", SalesLC, route, login,
sales_web_user
```

若要将使用人力资源应用程序的所有客户端路由到 “HRLC”，请输入别名路由：

```
sp_cluster logical, "add", HRLC, route, alias,
HR_SERVER
```

---

**注释** 请确保在客户端的目录服务中包括各个服务器别名，并且那些查询条目指定物理集群所监听的地址。

---

例如，若要为客户端的目录服务设置别名：

```
ase1
    query tcp ether blade1 19786

ase2
    query tcp ether blade2 19786

ase3
    query tcp ether blade3 19786

ase4
    query tcp ether blade4 19786

mycluster
    query tcp ether blade1 19786
    query tcp ether blade2 19786
    query tcp ether blade3 19786
    query tcp ether blade4 19786

HR_SERVER
    query tcp ether blade1 19786
    query tcp ether blade2 19786
    query tcp ether blade3 19786
    query tcp ether blade4 19786
```

有关完整的语法和用法信息，请参见《参考手册：过程》。

## 启动逻辑集群

若要启动逻辑集群（即，将该集群置于联机状态），请使用 `sp_cluster logical, "online"`。

例如，若要启动“SalesLC”和“HRLC”，请输入：

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

## 指派路由规则

共享磁盘集群的每个客户端连接都与一个逻辑集群关联。该关联可基于路由规则；它也可基于路由规则的空缺，这意味着连接将路由到开放式逻辑集群。

如果由于目标逻辑集群处于脱机状态或客户端不支持登录重定向而无法将连接定向到逻辑集群，则可依据目标逻辑集群的向下路由模式来处理连接。有关指定向下路由规则的信息，请参见第 70 页的“向下路由模式”。

## 路由规则

路由规则将传入的客户端连接定向到相应的逻辑集群。将连接路由到逻辑集群之后，即可通过逻辑集群管理功能来管理为该路由输入的 Adaptive Server 任务。

共有三种类型的路由规则（或绑定）。每种绑定都使用 TDS 登录记录中的名称。

绑定路由按从最高到最低的优先级顺序列出。因此，登录路由优先于应用程序路由，后者又优先于别名路由。

- 登录路由 — 在 Adaptive Server 登录和逻辑集群之间建立关系。
- 应用程序路由 — 在应用程序名和逻辑集群之间建立关系。
- 别名路由 — 将服务器名与逻辑集群相关联。应用程序可以从置于 `interfaces` 文件中的服务器别名中选择逻辑集群。这些别名使用唯一的服务器名。

别名条目可指向物理集群中的任意位置。负载管理器通过登录重定向将其发送到正确的实例。

## 配置逻辑集群特性

每个逻辑集群都有一组特性（或属性），这些特性控制逻辑集群行为的不同方面。每个特性都有缺省值。您可以接受缺省值，也可以对其进行更改以最适合于您的应用程序环境。

若要查看逻辑集群特性的当前设置，请使用 `sp_cluster logical, "show"`。请参见第 78 页的“查看有关逻辑集群的信息”。

使用 `sp_cluster logical` 来管理以下特性：

- 开放式 — 指定要将没有具体路由计划的客户端定向到的逻辑集群。
- 系统视图 — 指定监控和信息性工具（例如 `sp_who`、`sp_lock`）以及监控表是描述集群中的实例，还是描述整个集群。
- 启动模式 — 指定是必须自动启动逻辑集群，还是必须手动启动逻辑集群。
- 向下路由模式 — 指定在路由规则指定的逻辑集群不可用的情况下如何路由由客户端连接。
- 故障切换模式 — 如何以及何时使故障切换实例联机。
- 任何实例都提供故障切换支持 — 指定是任何实例都可以作为故障切换资源，还是只有指定的实例可以作为故障切换资源。
- 负载配置文件 — 提供一系列加权指标，用于确定逻辑集群中实例的相对负载。
- 登录分配模式 — 指定 Adaptive Server 在逻辑集群跨多个实例时如何分配连接。
- 操作释放 — 在这些逻辑集群操作（`online`、`offline`、`failover` 和 `failback`）完成或取消之后手动或自动释放和清除它们。
- 收集模式 — 在以下预定义操作之一发生时手动或自动收集与另外的逻辑集群之间的连接组：`online`、`add route`、`alter route` 或 `drop route`。
- 角色 — 指示逻辑集群是“系统”、“开放式”还是“无”（如果逻辑集群不是二者之一）。

## 开放式逻辑集群

所有未通过显式路由规则路由到某个逻辑集群的连接都将路由到当前的开放式逻辑集群。当您创建新的集群时，将自动为系统逻辑集群指定开放式逻辑集群。可以为另一个逻辑集群重置 `open` 特性。但是，每个物理集群只能存在一个开放式逻辑集群。

若要指定新的开放式逻辑集群，请使用 `sp_cluster logical, "set"`。例如，若要指定 “CatchallLC” 作为开放式逻辑集群，请输入：

```
sp_cluster logical, "set", CatchallLC, "open"
```

当您为新的逻辑集群重置 `open` 特性时，Adaptive Server 将自动禁用旧的开放式逻辑集群的 `open` 特性。

您可以将 `open` 属性与向下路由模式一起使用，以便为特定逻辑集群的独占使用保留一个或多个实例。

## 向下路由模式

路由规则将传入的客户端连接定向到相应的逻辑集群。请参见第 68 页的“指派路由规则”。但是，路由规则未指定在目标逻辑集群处于脱机状态的情况下应如何处理连接，或在需要重定向但连接不支持重定向的情况下应如何处理连接。

---

**注释** `Client-Library` 属性 `CS_PROP_REDIRECT` 确定客户端连接是否支持登录重定向。缺省情况下，`CS_PROP_REDIRECT` 的值为 `true`，并且客户端连接支持登录重定向。请参见《Client-Library/C 参考手册》。

---

您可以指定向下路由模式，以便在无法应用路由规则时将连接定向。

也可以使用此特性为特定连接保留某些实例。请参见第 71 页的“资源保留”。

使用 `sp_cluster logical, "set"` 来配置向下路由模式。值包括：

- `system` — 将不可路由的连接发送到系统逻辑集群。由于系统逻辑集群对于每个实例均始终处于联机状态，因此 `system` 可确保最高的可用性。这是缺省设置。
- `open` — 将不可路由的连接发送到开放式逻辑集群。如果无法将连接发送到开放式逻辑集群，或者连接不支持重定向，则 Adaptive Server 将应用开放式逻辑集群的向下路由模式。

- **disconnect** — 断开不可路由的连接。通过断开目标逻辑集群中的实例无法提供服务的客户端的连接，此设置可强制实现资源保留。请参见第 71 页的“资源保留”。

例如，若要为“SalesLC”将向下路由模式设置为 **open**，请输入：

```
sp_cluster logical, "set", SalesLC, down_routing,
"open"
```

此示例可确保，在“SalesLC”不可用的情况下，会将绑定到该逻辑集群的客户端路由到开放式逻辑集群。

#### 资源保留

您可以将 **open** 属性与 **disconnect** 向下路由模式结合使用，以便为特定逻辑集群的独占使用保留一个或多个实例。举例来说，假设您希望为“HRLC”逻辑集群的独占使用保留实例“ase3”：

- 1 将 **open** 属性设置为不包括“ase3”的逻辑集群。
- 2 将开放式逻辑集群的向下路由模式设置为 **disconnect**，这样，即使不支持重定向的客户端也能访问该集群。

只有其路由规则指定“HRLC”的连接才能连接到“ase3”。

## System-view 特性

**system-view** 特性控制在使用系统存储过程（例如 **sp\_who** 和 **sp\_lock**）或查看监控表、虚设表等时您将如何查看集群。您可以设置 **system\_view** 特性，以便 Adaptive Server 显示有关当前实例或物理集群的信息。

例如，若要设置实例的系统视图：

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

设置集群的系统视图：

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

通过在逻辑集群级设置系统视图，将为逻辑集群中的连接提供缺省值。请参见《参考手册：过程》。

---

**注释** **system-view** 特性是一种管理工具。它的当前值不影响应用程序识别逻辑集群、其数据库或数据库对象的方式。

---

## 启动模式

启动模式特性指定逻辑集群是否在集群启动时自动启动，以及管理员是否手动启动逻辑集群。

- 在自动模式中，逻辑集群将在其任何基本实例在某个集群重新启动后进入联机状态时启动。如果其它故障切换实例当前处于联机状态，则故障切换实例将自动进入联机状态。该值为缺省值。
- 在手动模式中，只有在管理员执行 `online` 命令后，逻辑集群才会进入联机状态。

例如，若要将“SalesLC”的启动模式更改为手动，请输入：

```
sp_cluster logical, "set", SalesLC, startup, manual
```

## 故障切换模式

故障切换模式指定逻辑集群在何时以及如何依据故障切换实例运行。

使用 `sp_cluster logical, "set"` 来指定故障切换模式。值包括：

- **instance** — 指定每当某个联机实例（不管是基本实例还是故障切换实例）失败时，都按 1:1 的比例将其替换为一个故障切换实例。例如，假设逻辑集群“SalesLC”依据实例“ase1”和“ase2”运行，“ase3”和“ase4”为故障切换实例，并且故障切换模式为“instance”。如果“ase1”失败，则“ase3”将进入联机状态，并且该集群将依据“ase2”和“ase3”（或“ase4”，具体情况取决于这两个故障切换实例的相对负载）运行。**instance** 为缺省值。
- **group** — 指定只有当所有基本实例均失败时才替换基本实例，并且所有故障切换实例随后将进入联机状态。假设“SalesLC”的故障切换模式为“group”。如果“ase1”失败，则集群将继续依据“ase2”运行。没有故障切换实例将进入联机状态。但是，如果“ase1”和“ase2”均失败，则集群将依据“ase3”和“ase4”运行。

您也可以指定多个故障切换组，这样，即使第一个故障切换集内的实例失败，另一个故障切换实例集也将可进入联机状态。

例如，若要为“SalesLC”将故障切换模式设置为“group”，请输入：

```
sp_cluster logical, "set", SalesLC, failover, "group"
```

## Fail\_to\_any 特性

`fail_to_any` 特性确定逻辑集群是否能够将故障切换到集群中的任何实例，或仅将故障切换到指定的故障切换实例。只有当指定的故障切换实例无法进入联机状态时，此特性才发挥重要作用。

使用 `sp_cluster logical, "set"` 来设置 `fail_to_any` 特性。值包括：

- `true` — 指定系统始终选择其它实例来充当故障切换实例，只要集群中的任何其它实例均处于联机状态并可用即可。该值为缺省值。
- `false` — 指定只能使用指定的故障切换实例。

例如，若要为 “SalesLC” 禁用 `fail_to_any` 特性，请输入：

```
sp_cluster logical, "set", SalesLC, fail_to_any, false
```

## 负载配置文件特性

Adaptive Server 使用负载配置文件为逻辑集群中的每个实例提供负载得分。负载得分确定负载管理器何时将连接定向到其它实例来帮助平衡负载。您可以使用经由 Sybase 测试并提供的样本负载配置文件，也可以配置您自己的负载配置文件。有关详细信息，请参见第 91 页的“负载配置文件”。

## 登录分配模式

利用登录分配模式，您可以指定如何在具有多个实例的逻辑集群中分配连接。登录分配模式不影响单实例逻辑集群。

值包括：

- `affinity` — 指定接受连接的实例将保留连接，只要目标逻辑集群依据该实例运行即可。
  - 如果负载配置文件指定了负载阈值，并且实例上的负载太高，则负载管理器会将连接重定向到逻辑集群中负载最小的实例。
  - 如果目标逻辑集群未依据该实例运行，则负载管理器会将连接重定向到逻辑集群中负载最小的实例。
- `round-robin` — 指定以循环方式在承载逻辑集群的实例之间分配传入连接。例如，如果 “SalesLC” 依据 “ase1” 和 “ase2” 运行，则负载管理器会将第一个连接发送到 “ase1”，将下一个连接发送到 “ase2”，依此类推。算法中未包括负载得分。

**注释** Cluster Edition 不会为具有 `sa_role` 的连接执行基于负载的重定向（关联或循环）。但是，如果路由将 `sa_role` 连接定向到依据另一个实例运行的逻辑集群，则会重定向这些连接。

---

### Sybase 建议

Sybase 建议为具有短期、频繁连接的事务型应用程序使用关联模式，而对于应用程序服务器会建立与 Adaptive Server 的持久连接池的部分读应用程序而言，则建议使用循环模式。

## 操作释放

操作释放在逻辑集群操作（`online`、`offline`、`failover` 和 `failback`）完成或取消之后手动或自动清除所有操作。若要启用或禁用逻辑集群操作的自动清除，语法为：

```
sp_cluster logical, "set", "lc_name", "action_release",  
{"automatic" | "manual"}
```

若要手动清除所有逻辑集群操作，语法为：

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

## 收集模式

收集模式能让您使用定义的路由规则收集系统上的所有合格连接或与开放式逻辑集群之间的所有合格连接，并将它们移动至另一个逻辑集群。收集模式可以在发生预定义的操作（如 `online`、`add route`、`alter route` 或 `drop route`）时自动移动连接，或者您可以手动收集连接。然后，Adaptive Server 将查找与受影响的逻辑集群的路由规则匹配的所有连接，并将这些连接迁移到指定的逻辑集群。

若要手动收集连接并将其移动至指定的逻辑集群，语法为：

```
sp_cluster logical, "gather", "lc_name"
```

若要启用或禁用连接的自动收集，语法为：

```
sp_cluster logical, "set", "lc_name", "gather", {"automatic" | "manual" }
```

另请参见第 81 页的“迁移连接”。

## 角色

逻辑集群角色值为“系统”、“开放式”和“无”：

- 系统 — 指示指定的逻辑集群是系统逻辑集群。请参见第 62 页的“系统逻辑集群”。
- 开放式 — 指示指定的逻辑集群是客户端定向到的逻辑集群（如果客户端没有特定的路由计划）。若要使逻辑集群成为开放式逻辑集群，请参见第 70 页的“开放式逻辑集群”。
- 无 — 指示逻辑集群是系统逻辑集群还是开放式逻辑集群。

若要查看逻辑集群的当前角色，语法为：

```
sp_logical_cluster, "show", "lc_name"
```

## 配置故障切换

物理集群中的任何实例都可以是故障切换资源。逻辑集群故障切换规则不会影响基础结构、锁定重控制或恢复。可通过指定以下各项来配置逻辑集群故障切换：

- 故障切换资源 — 将故障切换到其中的特定实例或实例组。
- 故障切换模式 — 确定是对逻辑集群的个别成员进行故障切换，还是仅对集群整体进行故障切换。请参见第 72 页的“故障切换模式”。
- fail\_to\_any 特性 — 确定在指定的故障切换资源不可用的情况下，是仅将故障切换到指定的故障切换资源，还是将故障切换到任何其它实例。请参见第 73 页的“Fail\_to\_any 特性”。

当您创建逻辑集群时，fail\_to\_any 特性的缺省设置可确保在基本实例失败的情况下，会立即将该实例替换为另一个实例。这是最简单的故障切换策略，对于许多节点而言已经足够。

如果您的节点需要更精细地控制故障切换资源，则可以更改缺省设置，并将故障切换定向到特定实例或实例组。

可以为每个逻辑集群创建多达 31 个故障切换组。通过将故障切换实例分组，您可以为特定故障切换组指定优先顺序。例如，您可以确保在考虑组 2 中的实例之前先考虑组 1 中的实例，诸如此类。负载管理器依据负载在组内选择故障切换实例：先选择负载较轻的实例。

实例只能是每个逻辑集群的一个故障切换组的成员。因此，如果实例“ase4”位于“SalesLC”的故障切换组 1 中，则该实例不能同时位于“SalesLC”的故障切换组 2 中。但是，“ase4”可同时位于“SalesLC”的故障切换组 1、“HRLC”的故障切换组 2 中并且是“CatchallLC”的基本实例。

当负载管理器需要激活故障切换实例时，它将首先在组 1 中查看，然后在组 2 中查看，依此类推，直至故障切换条件得到满足为止。如果负载管理器无法激活配置的故障切换资源，它将检查 `fail_to_any` 参数配置。如果 `fail_to_any` 为 `true`，则负载管理器将尝试使用任何可用的实例来满足故障切换。如果 `fail_to_any` 为 `false`，则不进行故障切换。

## 添加故障切换资源

使用 `sp_cluster logical, "add"` 将故障切换资源添加到逻辑集群。不能向系统逻辑集群中添加故障切换资源。

每次当您使用 `sp_cluster logical, "add"` 添加故障切换资源时，Adaptive Server 都将创建一个或多个故障切换组。

如果您添加一个或多个故障切换实例（用分号分隔多个实例），则 Adaptive Server 会将所有实例置于单个组中。

例如，若要将“ase3”作为故障切换组添加到“SalesLC”，请输入：

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

也可以将故障切换实例添加到现有故障切换组。例如，假设“ase3”是故障切换组 1 的成员。若要将“ase4”添加到故障切换组 1，请输入：

```
sp_cluster logical, "add", SalesLC, failover, ase4, "1"
```

若要查看故障切换资源信息（包括故障切换组 ID），请使用 `sp_cluster logical, "show"`。

## 管理逻辑集群

本节介绍如何管理逻辑集群。

## 用户任务和逻辑集群

每个 Adaptive Server 任务 (SPID) 都在一个逻辑集群内运行。sysprocesses 中的 lcid 列是承载给定任务的逻辑集群 ID。可将此 ID 传递到 lc\_name() 内置函数来确定对应逻辑集群的名称。

单个任务可运行 lc\_name() 内置函数来确定当前逻辑集群。

## 管理负载管理器线程

负载管理器线程是一种依据每个实例运行的系统服务线程。当实例启动时，它将自动生成负载管理器线程。此线程大部分时间处于休眠状态，但会定期苏醒来处理逻辑集群操作、收集负载指标、计算每个实例上的负载、将负载信息发送到所有实例，以及履行其它管理职责。

通过使用 sp\_who 以及用于监控进程的其它 Adaptive Server 功能来查询 sysprocesses 和 monProcesses，您可以查看有关负载管理器的信息。

您可能需要更改负载管理器最大内存使用量的缺省值（请参见第 77 页的“设置负载管理器的内存要求”）。否则，负载管理器不需要维护。

## 设置负载管理器的内存要求

设置“workload manager cache size”配置参数，以指定负载管理器可使用的最大内存量。负载管理器使用的所有内存均来自内存池（大小由 workload manager cache size 设置）。

连接迁移消耗此池中的内存；每个配置的逻辑集群、路由和负载配置文件也消耗此池中的内存。诸如 failover 和 failback 等命令所产生的操作消耗此池中的内存，并且，在释放操作之前，将持续消耗此池中的内存。

基于以下准则估计内存使用量：

- 每个并发迁移连接使用四个内存页
- 三个逻辑集群使用一页
- 两个负载配置文件使用一页
- 30 个路由使用一页
- 12 个操作使用一页

使用 `sp_configure` 以 2K 页为增量设置内存池的最大值。例如，若要将 `workload manager cache size` 设置为 100 个 2K 页，请输入：

```
sp_configure "workload manager cache size", 100
```

`workload_manager_size` 是动态的；您无需重新启动服务器。缺省值为 80（即 160KB）。

该缺省值对于大多数安装已经足够。如果预计要迁移具有多个并发连接的逻辑集群，您可能需要增加内存池的大小。

## 查看有关逻辑集群的信息

若要查看有关逻辑集群的信息，您可以：

- 使用内置函数 `lc_name()`、`lc_id()`、`instance_name()` 和 `instance_id()`。
- 使用全局变量 `@@clustername`、`@@instancename` 和 `@@instanceid`。
- 查询监控表。
- 使用 `sp_cluster logical, show`。

## 查询监控表

以下监控表提供有关逻辑集群、负载和负载配置文件的信息：

- `monLogicalCluster` — 提供有关系统的逻辑集群的摘要信息。
- `monLogicalClusterInstance` — 提供有关系统的逻辑集群中每个实例的信息。
- `monLogicalClusterRoute` — 提供有关配置的路由的信息。
- `monLogicalClusterAction` — 提供有关系统的逻辑集群中操作的信息。
- `monWorkload` — 提供每个负载配置文件的每个实例的负载得分。
- `monWorkloadProfile` — 提供有关每个负载配置文件的信息。

可以使用 `Transact-SQL` 命令查询这些表来了解信息。有关每个监控表的完整说明，请参见《参考手册：表》。

## 使用 `sp_cluster logical, "show"`

您可以使用 `sp_cluster logical, "show"` 执行以下操作：

- 查看有关特定逻辑集群或所有逻辑集群的摘要信息。例如，若要查看有关 “SalesLC” 的信息，请输入：

```
sp_cluster logical, "show", SalesLC
```

若要查看有关所有逻辑集群的信息，请输入：

```
sp_cluster logical, "show"
```

- 查看有关操作的信息。例如，若要查看有关所有已完成操作的信息，请输入：

```
sp_cluster logical, "show", NULL, action, complete
```

若要检索有关 “SalesLC” 的已取消操作的信息，请输入：

```
sp_cluster logical, "show", SalesLC, action,  
cancelled
```

若要检索有关 SalesLC 的活动操作的信息，请输入：

```
sp_cluster logical, "show", SalesLC, action, active
```

- 查看有关已配置路由的信息。您可以查询特定应用程序、登录、别名或这些项目的组合。

若要查看有关 “sales\_web\_user” 登录路由到 “SalesLC” 的信息，请输入：

```
sp_cluster logical, "show", SalesLC, route, login,  
sales_web_user
```

有关 `sp_cluster logical, "show"` 的完整语法和用法信息，请参见《参考手册：过程》。

## 创建和删除逻辑集群

- 若要创建逻辑集群，请使用 `sp_cluster logical, "create"`。例如，若要创建 “FinanceLC”，请输入：

```
sp_cluster logical, "create", FinanceLC
```

- 若要删除逻辑集群，请使用 `sp_cluster logical, "drop"`。逻辑集群必须处于脱机或不活动状态，然后才能将其删除。`sp_cluster logical, "drop"` 将删除集群和所有路由、资源以及与该集群关联的属性。

例如，若要删除 “FinanceLC”，请输入：

```
sp_cluster logical, "drop", FinanceLC, cluster
```

## 向逻辑集群中添加资源

使用 `sp_cluster logical, "add"` 向逻辑集群中添加资源。您可以添加：

- 基本实例 — 请参见第 66 页的“向逻辑集群中添加实例”中的示例。
- 故障切换实例 — 请参见第 76 页的“添加故障切换资源”中的示例。

## 从逻辑集群中删除资源

使用 `sp_cluster logical, "drop"` 从逻辑集群中删除一个或多个资源。基本实例或故障切换实例必须处于脱机状态，然后才能将其删除。

您可以删除：

- 基本实例 — 例如，若要从 “SalesLC” 中删除实例 “ase2”，请输入：

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

- 故障切换实例 — 例如，若要从 “SalesLC” 中删除故障切换实例 “ase3” 和 “ase4”，请输入：

```
sp_cluster logical, "drop", SalesLC, failover,  
"ase3;ase4"
```

## 添加、移动和删除路由

- 若要添加路由，请使用 `sp_cluster logical, "add"`。例如，若要将登录“accounting”和“projects”的路由添加到“SalesLC”，请输入：

```
sp_cluster logical, "add", SalesLC, route, login,
"accounting;projects"
```

- 若要路由从一个逻辑集群移到另一个逻辑集群，请使用 `sp_cluster logical, "alter"`。例如，若要使用别名“SalesLC”创建到“My\_LC”的路由，然后将该路由从“My\_LC”移到>Your\_LC”，请输入：

```
sp_cluster logical, "add", My_LC, route, alias,
SalesLC
sp_cluster logical, "alter", Your_LC, route, alias,
SalesLC
```

- 若要删除路由，请使用 `sp_cluster logical, "drop"`。例如，若要从“SalesLC”中删除登录“projects”，请输入：

```
sp_cluster logical, "drop", SalesLC, route, login,
projects
```

## 迁移连接

使用 `sp_cluster connection "migrate"` 执行以下操作：

- 迁移连接（或另一个任务）运行所依据的逻辑集群或实例。
- 将应用程序或登录迁移到未针对其路由应用程序或登录的逻辑集群或实例。

例如，此示例将 `spid` 为 73 的连接迁移到 SalesLC 逻辑集群。

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

请参见《参考手册：过程》。

使用 `sp_cluster logical` 设置到另一个逻辑集群的手动或自动迁移，或在发生预定义事件时将几组连接收集到另一个逻辑集群。可以使用定义的路由规则将系统或逻辑集群上的所有合格连接“收集”到指定逻辑集群。Cluster Edition 将查找与此逻辑集群的路由规则匹配的所有连接，并将这些连接迁移到指定的逻辑集群。

语法为：

```
sp_cluster logical, 'gather', lc_name  
sp_cluster logical 'set', lc_name, 'gather', 'automatic | manual'
```

此示例将所有定义的连接收集到 SalesLC 逻辑集群：

```
sp_cluster logical, 'gather', SalesLC
```

此示例为 SalesLC 逻辑集群将收集设置为“手动”：

```
sp_cluster logical 'set', SalesLC, 'gather' 'manual'
```

请参见《参考手册：过程》。

## 管理故障切换、故障恢复和计划的停机时间

当操作为以下各项之一时，您可以使用 `sp_cluster logical, "action"` 手动更改逻辑集群及其实例的状态：

- failover
- failback
- online
- offline
- deactivate

## 集群和实例状态

逻辑集群及该集群中的每个实例可以有不同的状态。

- 逻辑集群具有总体（或全局）状态，该状态确定集群是处于脱机还是联机状态。
- 逻辑集群还具有**实例状态**，该状态描述逻辑集群所识别的特定实例的状态。例如，某个联机逻辑集群在其基本实例上可能处于联机状态，而在其故障切换实例上则处于脱机状态。此状态可能与实际 Adaptive Server 状态无关，因为在实际已启动并正在运行的实例上，逻辑集群可能处于脱机状态。

用户可看到五种状态。这些状态适用于逻辑集群状态以及实例状态。  
表 6-1 描述了全局和实例级的每种状态。

**表 6-1: 逻辑集群状态**

状态	在全局级	在实例级
联机	逻辑集群处于联机状态，并依据一个或多个实例运行。	联机逻辑集群在当前实例上接受和管理连接。
脱机	逻辑集群未依据任何实例运行。	逻辑集群未依据当前实例运行，该实例无法接受连接或消耗资源。
不活动	类似于脱机状态，逻辑集群未依据任何实例运行。不活动的逻辑集群不会自动启动，并且不参与故障切换。集群只能通过 <code>deactivate</code> 命令达到不活动状态。处于不活动状态之后，集群只能通过 <code>online</code> 命令进入联机状态。	逻辑集群未依据当前实例运行，该实例无法接受连接或消耗资源。此外，不活动的实例无法进行故障切换，并且在自动启动后不会进入联机状态。只能通过 <code>deactivate</code> 命令达到此状态。
失败	类似于脱机状态，逻辑集群未依据任何实例运行。当对逻辑集群的活动实例执行 <code>shutdown with nowait</code> 或逻辑集群在故障资源不可用的情况下遇到系统故障时，该逻辑集群将变为失败状态。	逻辑集群未依据当前实例运行，该实例无法接受连接或消耗资源。失败状态是通过 <code>shutdown with nowait</code> 或系统故障达到的。
时间等待	联机与脱机或不活动之间的过渡状态。联机逻辑集群在变为脱机或不活动状态之前必须进入时间等待状态。在此期间，将依据向下路由模式来路由任何新连接，而现有连接将迁移或断开连接。	联机与脱机或不活动之间的过渡状态。当逻辑集群对于某个实例处于联机状态时，它变为脱机或不活动状态之前必须进入时间等待状态。在时间等待状态中，无法将任何新连接路由到逻辑集群或实例，但现有连接将继续运行，直至它们迁移或断开连接为止。

## 状态如何更改

集群和实例状态可能会通过以下方式更改：

- 在您使用 `online`、`offline`、`failover` 和 `failback` 命令（有时使用 `action` 命令）执行状态更改时手动更改。
- 由于系统发生更改而自动更改

集群或实例的初始状态可能指示状态更改是否有效，甚至指示最终状态。表 6-2 显示手动执行的不同操作与状态如何交互。状态在行中描述；操作在列中描述。每个单元格都表示将某项操作应用于处于初始状态的逻辑集群或实例时的新状态。

**表 6-2：操作和状态的交互**

	脱机	联机	时间等待	失败	不活动
联机	联机			联机	联机
脱机		脱机/时间等待		脱机	脱机
故障恢复实例	联机			联机	联机
故障恢复集群		联机/时间等待		脱机	脱机
故障切换实例	联机			联机	联机
故障切换集群		联机/时间等待		脱机	脱机
取消操作			联机		
修改等待			时间等待		
停用	不活动	不活动/时间等待		不活动	

状态也可能因为系统发生更改而更改。表 6-3 显示了不同系统更改对集群或实例的状态产生的影响。

**表 6-3：操作和状态的交互**

	脱机	联机	时间等待	失败	不活动
实例加入集群	如果配置了自动启动，则为联机			如果启用了自动启动，则为联机	
正常关闭		时间等待			
系统故障		失败	失败		
关闭而不等待		失败	失败		
故障切换选择	联机				

**注释** 在整个集群重新启动后，将不会保留逻辑集群状态。例如，假设您为处于自动启动模式下的逻辑集群执行 `offline` 命令。在您重新启动集群后，该集群将处于联机状态。

## 异步命令和逻辑集群状态

`sp_cluster logical` 命令 `deactivate`、`failback`、`failover` 和 `offline` 是异步命令。它们会停止可能有现有事务的联机实例。必须先处理这些事务，然后才能实际使实例脱机或使其处于非活动状态。因此，可允许这些命令稍后在上下文中完成。

当您执行任何这些命令时，目标实例将处于时间等待状态，并且不再接受新连接。

每个异步命令都提供了三个用于处理现有事务的“wait”选项。值包括：

- **wait** — 允许现有连接保留指定的一段时间（例如五分钟）。支持迁移的连接将在处于抵制状态后立即迁移。不支持迁移的连接将在变为抵制状态后断开连接。可识别 HA 的客户端将进行故障切换；不可识别 HA 的客户端将断开连接。在指定时间后保留的连接将终止。
- **until** — 在指定时间（例如，下午 12:30）之前允许现有连接保留。否则，**until** 和 **wait** 采用相同的方式处理连接。
- **nowait** — 立即终止现有连接。支持迁移的连接必须立即迁移，否则将被终止。

---

**注释** 如果您在执行 `sp_cluster logical` 异步命令时未指定 `wait` 选项，则 Adaptive Server 将假定为无限期等待。

---

当使用实例的最后一个连接断开连接时，实例状态将从时间等待变为脱机或不活动。

## 使用操作描述符

利用操作描述符可以跟踪或更改操作。

当异步命令试图停止一个或多个实例时，它将生成操作描述符。操作描述符跟踪操作、`wait` 选项和处于时间等待状态的目标实例。可通过查询 `monLogicalClusterAction` 表或执行 `sp_cluster logical, "show", NULL, action` 来查看有关操作描述符的信息。

操作可处于“活动”或“完成”状态。当至少一个目标实例保持为时间等待状态时，操作将处于活动状态。当所有目标实例均不再处于时间等待状态时，则操作处于完成状态。

使用 `sp_cluster logical, action`，您可以通过以下选项管理操作描述符：

- **cancel** — 终止活动操作。由于该操作而处于时间等待状态的实例将恢复为联机状态。现有连接将保留，即使它们已标记为迁移或终止也是如此。

如果导致创建操作的命令使实例进入联机状态，则实例将保持联机状态。例如，如果操作导致取消从 `s1` 故障切换到 `f1`，则 `f1` 将保持联机状态。

- **modify\_wait** — 更改 `wait` 选项（请参见第 85 页的“异步命令和逻辑集群状态”）和与活动操作关联的时间。例如，如果某个操作在创建时设置为等待 10 分钟，请使用 `modify_wait` 更改以下各项：
  - 将延时更改为 20 分钟
  - 将延时更改为实际时钟时间（下午 4:30）
  - 将 `wait` 选项更改为 `nowait`
- **release** — 从 `monLogicalClusterAction` 表中删除完成的操作。

完成的操作保留在 `monLogicalClusterAction` 表中，以便您能够跟踪这些操作的状态。但是，完成的操作会消耗负载管理器高速缓存中的内存。请在操作完成后执行 `release` 命令以释放此内存。

---

**注释** 操作信息只存储在内存中。如果重新启动整个集群，将会从 `monLogicalClusterAction` 表中清除所有操作。

---

## 示例：安排和重新安排故障切换

可以为集群或实例执行管理故障切换。集群或实例将故障切换到以前配置故障切换资源。

在此示例中，我们将对“SalesLC”集群进行故障切换，同时将故障切换安排在凌晨 2 点进行以便稍后能够跟踪或更改操作，我们还包括了将输出操作句柄的语法：

```
declare @out_handle varchar(15)

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

假设命令输出操作句柄 “1234”，并且 SalesLC 进入时间等待状态。所有新连接都将迁移到故障切换资源。在凌晨 2 点之后保留的任何现有连接将终止，并且 “SalesLC” 将进入脱机状态。

假设我们发现必须要立即迁移所有连接。我们可以使用操作句柄来重新安排即期故障切换。请输入：

```
sp_cluster logical, "failover", SalesLC, modify_time,
"1234", nowait
```

## 使用 failover、failback、online、offline 和 deactivate

### failover

**failover** 发起从逻辑集群的基本资源到其故障切换资源的手动故障切换。之前必须使用 `sp_cluster logical, "add"` 设置故障切换资源。在发起部分集群故障切换时，请指定要进行故障切换的基本资源的列表，以及将作为基本实例的故障切换目标的故障切换资源的列表。

例如，您可以将逻辑集群的一部分故障切换到一组之前配置的故障切换资源。此处，“SalesLC” 依据实例 “ase1” 和 “ase2” 运行。若要保持使 “SalesLC” 依据 “ase2” 运行，但将 “ase1” 故障切换到以前定义的故障切换资源 “ase3”，请输入：

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

在此示例中，指定了 `no wait` 选项，该选项缺省情况下指定无限期等待。

### failback

**failback** 与 **failover** 相反。它发起从逻辑集群的故障切换资源到其基本资源的手动故障恢复。在发起部分集群故障切换时，请指定要进行故障恢复的故障切换资源的列表，以及将作为故障切换实例的故障恢复目标的基本资源的列表。

在此示例中，我们将逐步对依据 “ase3” 运行的 “SalesLC” 进行故障恢复，以便 “SalesLC” 依据 “ase1” 运行。我们指定 2 分钟的等待。

```
declare @out_handle varchar(15)

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

## online

**online** 启动逻辑集群或逻辑集群中的实例，同时将它们置于联机状态。

例如，若要在“ase1”上启动 SalesLC，请输入：

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

有关更多示例，请参见第 67 页的“启动逻辑集群”。

## offline

**offline** 停止处于联机或活动状态的逻辑集群或实例。

例如，若要使“SalesLC”脱机，等待 5 分钟，并将操作存储在局部变量内的操作句柄中，请输入：

```
declare @out_handle varchar(15)

execute
sp_cluster logical, "offline", SalesLC, cluster, wait,
00:05:00, @handle=@out_handle output
```

## deactivate

**deactivate** 与 **offline** 相同，只是它将集群或实例置于不活动状态。请参见第 88 页的“offline”。

# 分配负载

每个实例都有一个负载管理器线程，该线程负责计算当前实例上的负载，并将该信息发送到集群中的其它实例。负载管理器是一种系统服务线程；它在实例启动时生成。

Adaptive Server 使用负载测量算法来计算每个实例的负载得分。此负载得分是一个无单位的数字，可与其它负载得分进行比较来确定相对负载。因此，您可以比较集群中的负载得分，或比较特定集群不同时间的负载得分。负载得分只有在与其它负载得分进行比较时才有意义。

## 负载指标

负载管理器每 15 秒重新计算一次负载得分，同时使用负载得分（和生成的统计信息）来比较集群中所有实例的相对负载。负载管理器使用这些统计信息来填充监控表。统计信息按实例列出：负载管理器不会跟踪按 SPID 列出的统计信息

使用 `workload_metric` 来更新实例的用户指标的值，该值用于计算实例的负载总分。可以将 `workload_metric` 包括在用户定义的存储过程、触发器或外部触发的脚本中。实例每 15 秒重新计算一次负载得分。

在计算负载得分时，Adaptive Server 将考虑五个系统定义的指标，并根据需要考虑一个用户提供的指标。

- 用户连接 — 实例基于资源可用性接受新连接的能力。
- CPU 利用率 — 实例接受额外工作的能力。
- 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。
- I/O 负载 — 未完成的异步 I/O。
- 引擎差值 — 集群中各实例之间联机引擎数的差异。

---

**注释** 仅当集群中的实例具有数量不等的引擎时，才会测量引擎差值。在此情况下，引擎差值会在负载得分中添加一个衡量最大相对能力的指标。

---

- 用户指标 — 特定于用户环境的可选指标。使用 `workload_metric` 来提供它的值。请参见“[创建用户指标](#)”。

负载得分是使用以下公式计算得出的：

$$\begin{aligned}
 & \text{ConnectionsWeight} \times (\text{ConnectionsLoad})/100 \\
 & + \text{CPUWeight} \times (\text{CPULoad})/100 \\
 & + \text{RunQueueWeight} \times (\text{RunQueueLoad})/100 \\
 & + \text{UserDefinedWeight} \times (\text{UserDefinedLoad})/100 \\
 & + \text{EngineWeight} \times (\text{EngineLoad})/100 \\
 & + \text{IOWeight} \times (\text{IOLoad})/100 \\
 & \text{-----} \\
 & = \text{Load score}
 \end{aligned}$$

## 创建用户指标

可以使用 `workload_metric` 内置函数将特定于节点的指标添加到负载测量算法。在典型的示例中，您可能会使用外部监控器来监控响应时间，然后将响应时间值插入算法。

在将系统提供的指标包括在负载算法中之前，`Adaptive Server` 将对这些指标进行规范化。为了兼容，您必须同样对用户提供的指标进行规范化。例如，如果可接受的最长响应时间为 5 秒，而测量的响应时间为 2 秒，则规范化值为 40（2 是 5 的 40%），可使用 `workload_metric` 将该值输入负载算法。

## 对负载指标进行加权处理

将依据重要性对负载得分的每个组成部分进行加权处理。将对指标值进行规范化，并将结果加总，为每个实例得出一个总体负载得分。`Sybase` 提供了对于大多数节点已经足够的缺省值，但如果您要包括特定于节点的指标，则可能需要使用 `sp_cluster profile` 来调整加权。

使用 `Sybase Central` 或 `sp_cluster` 来输入和调整加权因子，系统管理员可以看到这些因子来自定义负载配置文件

## 负载阈值

`Adaptive Server` 使用每个实例的负载得分来确定：

- 如何以最佳方式分配传入的连接 — 登录重定向。
- 是否迁移现有连接 — 动态负载分配。

只会为依据多个实例运行的逻辑集群执行负载分配。`Sybase` 负载分配策略是：在一个实例过载并且其它实例可用时重定向工作。它不会试图维持完全平衡的负载得分。因此，`Adaptive Server` 将为登录重定向和动态负载分配保留负载阈值。负载阈值是当前实例上的负载与当前参与逻辑集群的最小负载实例上的负载之间的百分比差异。在 `Adaptive Server` 重定向登录或迁移现有连接之前，该值必须得到满足。

`Adaptive Server` 将为登录重定向和连接迁移保留单独的负载阈值。通常，登录重定向阈值低于连接迁移的阈值。可以使用 `sp_cluster profile` 在创建负载配置文件时配置负载阈值。

## 停滞值

负载阈值指示 Adaptive Server 何时将连接从过载的实例重定向至最小负载实例。在阈值得以满足但实际负载得分太低而不需要迁移时，停滞值可防止迁移。

例如，假设当前实例上的负载得分为 2，最小负载实例上的负载得分为 1。百分比差异为 100%，该值满足负载阈值，但实际负载得分太低，因此不适合进行迁移。

## 负载配置文件

负载配置文件将负载得分系统的所有可配置因素合并为一个指定的实体。

您可以将不同的负载配置文件分配给同一物理集群中的不同逻辑集群，从而在同一物理集群中启用具有不同负载的多个应用程序。也可以在多个逻辑集群之间共享相同的负载配置文件。

例如，共享磁盘集群可同时承载主要为只读的基于 DSS 的逻辑集群，以及用于处理频繁写入的 OLTP 逻辑集群。对于这两种集群而言，登录重定向和连接迁移的最佳阈值可能差别很大。由于能够为每个逻辑集群分配特定负载配置文件，因此可使每个集群更有效地发挥作用。

---

**注释** Adaptive Server 将收集每个实例的负载统计信息，而不考虑逻辑集群。因此，如果两个逻辑集群依据同一实例运行，则它们对于该实例将具有相同的原始数据。但是，每个逻辑集群将依据它自己的负载配置文件来解释和使用该数据。

---

Sybase 提供了针对 OLTP 环境创建的预先配置好的配置文件。您也可以使用 `sp_cluster profile` 创建自己的负载配置文件。

## 使用样本负载配置文件

Sybase 提供了两个样本负载配置文件：

- `sybase_profile_oltp` — 针对 OLTP 环境配置。它将通过禁用基于负载的登录分配和动态负载分配，尝试将所有连接保留在同一实例上。重点将放在重新排队深度上，通过重新排队深度可以很好地预测响应时间。
- `sybase_profile_dss` — 针对主要为只读的 DSS 环境配置。它使用基于负载的登录分配和动态负载分配在多个实例之间分配负载，但将重点放在平衡 CPU 使用率和用户连接数上。

表 6-4 列出了 `sybase_profile_oltp` 和 `sybase_profile_dss` 的指标。

**表 6-4: 样本负载配置文件的指标**

负载指标	“sybase_profile_oltp”	“sybase_profile_dss”
配置文件 ID	1	2
用户连接数	0	40
CPU 利用率	10	40
运行队列长度	70	0
I/O 负载	20	10
引擎差值	0	10
用户权值（未使用）	0	0
登录阈值	0	20
动态阈值	0	50
停滞	20	20

## 创建和配置您自己的负载配置文件

创建和配置您自己的负载配置文件：

- 1 创建空白负载配置文件。
- 2 通过指定各个指标加权和阈值，建立负载配置文件。
- 3 将负载配置文件与逻辑集群关联。

## 创建负载配置文件

使用 `sp_cluster profile, "create"` 建立空白负载配置文件。例如，若要创建配置文件 “my\_profile”，请输入：

```
sp_cluster profile, "create", my_profile
```

## 建立负载配置文件

通过指定以下各项来建立负载配置文件：

- 组成负载配置文件的每个指标的加权
- 负载分配阈值

## 指定负载配置文件指标的加权

使用 `sp_cluster profile, "set"` 来配置负载配置文件中涵盖的每个指标的加权：

- 用户连接数
- CPU 忙
- 运行队列长度
- I/O 负载
- 引擎差值
- 用户指标（可选，请参见第 90 页的“创建用户指标”）

有关负载指标的说明，请参见第 89 页的“负载指标”。

使用介于 0 和 255 之间的值单独设置每个指标。例如，若要设置 “my\_profile” 的加权，请输入：

```
sp_cluster profile, "set", my_profile, weight, "user  
connections", "0"
```

```
sp_cluster profile, "set", my_profile, weight,  
"cpu busy", "20"
```

```
sp_cluster profile, "set", my_profile, weight, "run  
queue", "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"
```

## 指定负载分配阈值

使用 `sp_cluster profile, "set"` 将负载分配阈值设置为介于 0 和 100 之间。如果值为零 (0)，则禁用负载分配的该因素。可以为以下各项设置单独的负载阈值：

- 登录重定向
- 动态负载分配
- 停滞值

例如，若要在 “my\_profile” 中禁用动态负载分配，请输入：

```
sp_cluster profile, "set", my_profile, threshold,
"dynamic", "0"
```

若要为 “my\_profile” 将登录重定向阈值设置为 30，请输入：

```
sp_cluster profile, "set", my_profile, threshold,
"login", "30"
```

若要为 “my\_profile” 将停滞值设置为 20，请输入：

```
sp_cluster profile, "set", my_profile, threshold,
"hysteresis", "20"
```

## 将负载配置文件与逻辑集群关联

若要将负载配置文件与逻辑集群关联，请使用 `sp_cluster logical, "set"`。例如，若要将配置文件 “my\_profile” 与 “SalesLC” 关联，请输入：

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

## 更改负载配置文件

每个逻辑集群都与一个负载配置文件（系统负载配置文件或用户创建的负载配置文件）关联。若要更改负载配置文件，请将新的负载配置文件与逻辑集群关联。

例如，若要将“SalesLC”的负载配置文件从“my\_profile”更改为“sybase\_profile\_oltp”，请输入：

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

您可以随后删除旧的负载配置文件。例如，若要删除“my\_profile”，请输入：

```
sp_cluster profile, "drop", my_profile
```

## 故障排除

Sybase 提供了若干跟踪标志，您可以使用这些跟踪标志来排除负载管理器的故障。

**表 6-5: 负载管理器的跟踪标志**

跟踪标志	说明
16403	跟踪连接到逻辑集群和从逻辑集群断开连接过程中的 SPID。
16404	跟踪路由和路由解析。
16406	跟踪客户端重定向和迁移。
16414	跟踪逻辑集群状态机中的更改，包括从联机到脱机等状态的转换。



## 集群高速缓存配置

本章介绍在 Cluster Edition 环境中配置和使用命名数据高速缓存的机制。

主题	页码
<a href="#">全局高速缓存</a>	97
<a href="#">本地高速缓存</a>	98
<a href="#">创建和配置命名数据高速缓存</a>	98
<a href="#">配置和使用多个缓冲池</a>	109
<a href="#">将对象绑定到命名高速缓存</a>	113
<a href="#">修改配置文件</a>	115
<a href="#">限制</a>	120

集群高速缓存配置会定义多个命名高速缓存，以便根据应用程序的需要具有本地或全局高速缓存。该功能能让集群实例具有本地高速缓存。对象可以绑定到本地或全局高速缓存。支持多个缓冲池，因此可通过允许使用大 I/O，为命名高速缓存支持提供更好的访问性能。

用户还可以对应用程序进行分区，以便将应用程序数据对某个服务于该应用程序的特定实例的访问本地化。

### 全局高速缓存

全局高速缓存是针对集群中的每个实例定义的。对于全局高速缓存，高速缓存大小、缓冲池设置等特性位于 `sysconfigures` 表中，集群中的所有实例都从该条目中读取信息，以便在其各自的实例上创建高速缓存。

可以将全局高速缓存的高速缓存大小、缓冲池设置等特性更改为取决于特定实例。如果特定实例具有本地设置，则会使用本地设置来创建高速缓存。如果实例没有本地设置，则会使用全局定义来创建高速缓存。也就是说，具有本地定义的实例会覆盖全局定义和设置。

---

**注释** 可以动态增加本地和全局高速缓存的大小，但不能动态降低。

---

## 本地高速缓存

您的应用程序可以为集群中的每个实例定义本地高速缓存，以满足实例的特定需要并将高速缓存绑定到对象上。集群中每个特定于实例的高速缓存不需要全局定义。

本地高速缓存是特定于实例的。可以根据实例或实例所属的逻辑集群的需要来定制配置。如果特定应用程序通常在特定实例上运行，Sybase 建议您将应用程序分区到集群中的各个实例上。这样会最大限度地发挥本地高速缓存的优势，因为可以针对特定应用程序的访问模式来配置本地高速缓存。

一个对象只能绑定到一个高速缓存上，即任何特定实例的本地或全局高速缓存上。如果不将对象绑定到任何高速缓存上，或者如果在发生故障切换时，未配置特定于实例的高速缓存，则会使用缺省数据高速缓存。为帮助实现高效访问，Adaptive Server 维护每个实例上的绑定信息。

---

**注释** 任何实例上的本地高速缓存定义都会覆盖该实例的全局定义。

---

## 创建和配置命名数据高速缓存

sp\_cacheconfig 创建和配置全局和本地数据高速缓存。当安装 Adaptive Server 时，它仅包含一个全局高速缓存，叫做缺省数据高速缓存。

### 获取有关命名高速缓存的信息

可以通过输入以下命令来显示有关高速缓存的信息：

```
sp_cacheconfig  
go
```

```
Cache Name          Status          Type  Config Value    Run Value  
-----  
default data cache Active Global,Default  0.00 Mb        8.00 Mb  
-----  
-----  
Total              0.00 Mb        8.00 Mb  
=====
```

Cache: default data cache, Status: Active, Type: Global,Default  
Config Size: 0.00 Mb, Run Size: 8.00 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	1638 Kb	0.00 Mb	8.00 Mb	10

(return status = 0)

## 创建新高速缓存

最大的数据高速缓存大小仅受系统中可用内存量的限制。创建新高速缓存所需的内存从 Adaptive Server 的全局内存中分配。

创建高速缓存后：

- 它具有缺省清洗大小。
- 异步预取大小被设置为全局异步预取限制的值。
- 它只有缺省缓冲池。

## sp\_cacheconfig

定义

创建新的命名高速缓存。此语法是根据非集群 Adaptive Server 的语法扩展而来的。它提供一个额外的选项，用以在语法的末尾指定本地配置的实例名称。如果不指定实例名称，则配置是全局配置。

语法

```
sp_cacheconfig "[cachename [,cache_size [P|K|M|G]]
[.logonly | mixed ] [,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 — 指定高速缓存的类型。

strict | relaxed — 指定高速缓存替换策略。

cache\_partition — 指定要在高速缓存中创建的分区数。

**示例假设** 下面的示例假定名为 MYCLUSTER 的共享磁盘集群中包含两个实例：

- SALES\_INSTANCE
- HR\_INSTANCE

**创建命名高速缓存** 可以创建一个命名高速缓存 log\_sales，大小为 100M，特定于实例 SALES\_INSTANCE。在实例 SALES\_INSTANCE 上执行 sp\_cacheconfig 将会显示如下输出：

```
sp_cacheconfig 'log_sales','100M','instance SALES_INSTANCE'
go
```

```

Cache Name                Status Type                Config Value Run Value
-----
default data cache        Active Global,Default      0.00 Mb     8.00 Mb
SALES_INSTANCE:log_sales  Active Mixed                100.00 Mb   100.00 Mb
-----
Total 100.00 Mb 108.00 Mb
=====
Cache: default data cache, Status: Active, Type: Global,Default
  Config Size: 0.00 Mb, Run Size: 8.00 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     1638 Kb     0.00 Mb     .00 Mb     10
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
  Config Size: 100.00 Mb, Run Size: 100.00 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     20480 Kb    0.00 Mb     100.00 Mb  10
(return status = 0)
```

缺省情况下，`isql` 连接具有集群视图。在任何实例上都会显示所有特定于实例的高速缓存。例如，实例 `HR_INSTANCE` 显示有关高速缓存 `log_sales` 的信息，而该高速缓存是特定于实例 `SALES_INSTANCE` 的高速缓存。如果您希望 `HR_INSTANCE` 仅显示特定于该实例的本地高速缓存以及全局高速缓存的列表，请将系统视图设置为 `instance`。

在实例 `HR_INSTANCE` 上执行 `sp_cacheconfig` 将会显示：

```
set system_view instance
go

Cache Name          Status Type          Config Value      Run Value
-----
default data cache Active Global,Default  0.00 Mb          8.00 Mb
-----
Total 0.00 Mb 8.00 Mb
=====
Cache: default data cache, Status: Active, Type: Global,Default
      Config Size: 0.00 Mb, Run Size: 8.00 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      1638 Kb      0.00 Mb     8.00 Mb   10
(return status = 0)
```

**将高速缓存信息的输出限制为实例级别** 要显示实例 `SALES_INSTANCE` 的高速缓存，请执行：

```
sp_cacheconfig 'instance SALES_INSTANCE'
go

Cache Name          Status Type          Config Value      Run Value
-----
default data cache Active Global,Default  0.00 Mb          8.00 Mb
SALES_INSTANCE:log_sales Active Mixed          100.00 Mb        100.00 Mb
-----
Total 100.00 Mb 108.00 Mb
=====
Cache: default data cache, Status: Active, Type: Global,Default
      Config Size: 0.00 Mb, Run Size: 8.00 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      1638 Kb      0.00 Mb     8.00 Mb   10
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
```

```

Config Size: 100.00 Mb, Run Size: 100.00 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         20480 Kb    0.00 Mb     100.00 Mb   10
(return status = 0)
    
```

此输出显示实例 SALES\_INSTANCE 的本地和全局配置。

**查询命名高速缓存是否存在** 查明指定的高速缓存是否存在。在实例 SALES\_INSTANCE 上执行 `sp_cacheconfig` 将会显示如下输出：

```

Cache Name          Status  Type  Config Value  Run Value
-----
SALES_INSTANCE:log_sales Active Mixed   100.00 Mb   100.00 Mb
-----
                        Total 100.00 Mb 100.00 Mb
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
Config Size: 100.00 Mb, Run Size: 100.00 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         20480 Kb    0.00 Mb     100.00 Mb   10
(return status = 0)
    
```

**使用 Adaptive Server 语法创建全局高速缓存** 此高速缓存是在所有实例上创建的，内存是在所有实例上针对全局高速缓存分配的。要创建全局高速缓存 `tempdb_cache`，请在实例 SALES\_INSTANCE 上运行 `sp_cacheconfig`：

```

sp_cacheconfig 'tempdb_cache', '100M'
go
    
```

```

Cache Name          Status Type          Config Value  Run Value
-----
default data cache Active Global,Default   0.00 Mb     8.00 Mb
tempdb_cache Active Global,Mixed   100.00 Mb   100.00 Mb
SALES_INSTANCE:log_sales Active Mixed   100.00 Mb   100.00
Mb
-----Total
200.00 Mb 208.00 Mb
=====
Cache: default data cache, Status: Active, Type: Global,Default
Config Size: 0.00 Mb, Run Size: 8.00 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          1638 Kb          0.00 Mb          8.00 Mb          10
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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          20480 Kb        0.00 Mb        100.00 Mb     10
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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          20480 Kb        0.00 Mb        100.00 Mb     10
(return status = 0)

```

**创建具有一个全局配置和多个本地配置的命名高速缓存** 所有高速缓存操作都可以从任何实例中执行。例如，要在 SALES\_INSTANCE 上创建一个更大的命名高速缓存 tempdb\_cache，可以连接到实例 HR\_INSTANCE，然后执行：

```
sp_cacheconfig 'tempdb_cache','150M', 'instance
SALES_INSTANCE'
```

在实例 SALES\_INSTANCE 上执行 sp\_cacheconfig 将会显示：

```

Cache Name      Status Type          Config Value   Run Value
-----
default data cache Active Global,Default 0.00 Mb       8.00 Mb
tempdb_cache Active Global,Mixed 100.00 Mb     100.00 Mb
SALES_INSTANCE:log_hr Active Mixed        150.00 Mb     150.00 Mb
SALES_INSTANCE:tempdb_cache Active Mixed 150.00 Mb     150.00 Mb
-----
Total 350.00 Mb 408.00 Mb
=====
Cache: default data cache, Status: Active, Type: Global,Default
      Config Size: 0.00 Mb, Run Size: 8.00 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          1638 Kb        0.00 Mb        8.00 Mb   10
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 150.00 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      30720          Kb 0.00Mb    150.00 Mb  10
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
Config Size: 100.00 Mb, Run Size: 100.00 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      20480 Kb     0.00 Mb      100.00 Mb  10
=====
Cache: SALES_INSTANCE:tempdb_cache, Status: Active, Type: Mixed
Config Size: 150.00 Mb, Run Size: 150.00 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      30720 Kb     0.00 Mb      150.00 Mb  10

(return status = 0)
```

---

**注释** 命名高速缓存 tempdb\_cache 的本地配置会覆盖全局配置。

例如，如果将系统视图设置为 cluster，Adaptive Server 可能会显示一个命名高速缓存的所有配置，您应忽略所有在该实例上无效的配置的运行值。例如，SALES\_INSTANCE 有一个有效的本地配置，即高速缓存 tempdb\_cache。因此，您应该忽略全局配置的运行值。

同样，HR\_INSTANCE 有一个有效的全局配置。因此，您应该在实例 HR\_INSTANCE 上忽略 tempdb\_cache 的与 SALES\_INSTANCE 有关的本地配置运行值。

---

**向现有高速缓存添加内存** 要添加内存，请使用《参考手册：过程》中记录的语法。分配的附加内存将被添加到 Adaptive Server 页大小缓冲池中。例如，在逻辑页大小为 2K 的服务器上，缓冲池最小为 2K。如果高速缓存有分区，则会将附加内存平均分配给各个分区。

**注释** 如果在某个实例上向现有全局高速缓存中添加内存失败，但在至少一个其它实例上成功，则服务器会认为该操作在全集群范围级别成功。因此，全局高速缓存可以具有不同运行值，但高速缓存只有一个配置值。`sp_cacheconfig` 显示当前实例的 `syscurconfigs` 条目中的全局高速缓存的运行值。

要将 `tempdb_cache` 大小增加到 200MB，请在实例 `HR_INSTANCE` 中执行以下命令。在实例 `HR_INSTANCE` 上执行 `sp_cacheconfig 'tempdb_cache'` 将会显示如下输出：

```

        sp_cacheconfig 'tempdb_cache','200M'
Cache Name      Status Type          Config Value    Run Value
-----
tempdb_cache Active Global,Mixed    200.00 Mb      200.00 Mb
-----
                        Total 200.00 Mb 200.00 Mb
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 200.00 Mb, Run Size: 200.00 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         40960 Kb     0.00 Mb       200.00 Mb     10
(return status = 0)

```

还可以使用第 99 页的“`sp_cacheconfig`”中讲述的实例选项来增加本地高速缓存的高速缓存大小。

**为新的快速缓存分配空间** 如果 Adaptive Server 无法分配您请求的内存量，它会分配所有可用的内存并发出错误消息，告知您已经动态分配了多少千字节。

但是，只有在重新启动 Adaptive Server 后，才会分配这些内存。如果由于内存不可用或存在资源约束（系统管理员应确保这些资源约束是暂时的），而造成空间不足，Adaptive Server 会通知您。如果这个问题不解决，之后的重新启动可能会失败。

例如，如果最大内存为 700MB，tempdb\_cache 为 100MB，使服务器的总逻辑内存达到 600MB，而您尝试让 tempdb\_cache 增加 100MB，则附加内存还在最大内存之内。然而，如果服务器只能分配 90MB，那么它会动态分配这些内存，但配置文件中高速缓存的大小字段会被更新为 100MB。之后重新启动时，由于 Adaptive Server 同时为所有数据高速缓存获取内存，因此 pub\_cache 的大小为 100MB。

**减少高速缓存** 当减小高速缓存的大小时，请重新启动 Adaptive Server。例如，要将 tempdb\_cache 的大小减小为 100M，请使用以下命令。在实例 HR\_INSTANCE 上执行 sp\_cacheconfig 'tempdb\_cache' 将会显示：

```
sp_cacheconfig 'tempdb_cache', '100M'
go

Cache Name      Status Type          Config Value      Run Value
-----
tempdb_cache    Active Global,Mixed    100.00 Mb        200.00 Mb
-----
Total 100.00 Mb 200.00 Mb
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 200.00 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         40960 Kb       0.00 Mb          200.00 Mb     10
(return status = 0)
```

重新启动 Adaptive Server 并在 HR\_INSTANCE 上执行命令后：

```
sp_cacheconfig 'tempdb_cache'
go

Cache Name      Status Type          Config Value      Run Value
-----
tempdb_cache    Active Global,Mixed    100.00 Mb        100.00 Mb
-----
Total 100.00 Mb 100.00 Mb
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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         20480 Kb       0.00 Mb          100.00 Mb     10
(return status = 0)
```

**删除命名高速缓存** 要完全删除命名高速缓存，请将其大小重新设置为 0：

```
sp_cacheconfig 'tempdb_cache','0'
```

如果有对象绑定到命名高速缓存上，就无法删除该高速缓存，Adaptive Server 会发出错误消息。

如果命名高速缓存有多个配置，则配置文件中与该高速缓存对应的条目会被删除，`sysconfigures` 中与该高速缓存对应的条目也会被删除。下次重新启动实例后，该高速缓存会被删除。如果该高速缓存只有一个配置（全局或本地），则高速缓存条目既不会从配置文件中被删除，也不会从 `sysconfigures` 中被删除。该条目会在重新启动集群后或为命名高速缓存创建新的配置后被删除。

当删除特定于实例的配置时，命名高速缓存会恢复为其全局配置（如果存在这种配置的话）。在实例 `SALES_INSTANCE` 上执行 `sp_cacheconfig` 将会显示如下输出：

```
sp_cacheconfig 'tempdb_cache', '0', 'instance SALES_INSTANCE'
go

Cache Name      Status  Type          Config Value  Run Value
-----
tempdb_cache Active Global,Mixed  100.00 Mb    100.00 Mb
-----
                        Total 100.00 Mb 100.00 Mb
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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     20480 Kb   0.00 Mb     100.00 Mb 10
(return status = 0)
```

**更改高速缓存类型** 要保留高速缓存专供事务日志使用，请将高速缓存的类型改为 `logonly`。这个更改是动态的。要在 `HR_INSTANCE` 上创建一个 `logonly` 高速缓存，请输入以下命令。在实例 `HR_INSTANCE` 上执行 `sp_cacheconfig 'log_hr'` 将会显示如下输出：

```
sp_cacheconfig 'log_hr','logonly','instance HR_INSTANCE'

Cache Name      Status Type  Config Value  Run Value
-----
HR_INSTANCE:log_hr Active Log Only 150.00 Mb    150.00 Mb
-----
                        Total 150.00 Mb 150.00 Mb
```

```
=====  
Cache: HR_INSTANCE:log_hr, Status: Active, Type: Log Only  
      Config Size: 150.00 Mb, Run Size: 150.00 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         30720 Kb      0.00 Mb    150.00 Mb    10  
(return status = 0)
```

**配置高速缓存替换策略** 如果高速缓存专用于表或索引，并且在系统到达稳定状态后高速缓存只有很少或几乎没有缓冲区替换，则可以设置宽松的 LRU（最近使用最少的）替换策略。宽松的 LRU 替换策略可提高只有很少或没有缓冲区替换发生的高速缓存的性能，也能提高大多数日志高速缓存的性能。要设置宽松的替换策略，请输入：

```
sp_cacheconfig 'log_sales','relaxed','instance SALES_INSTANCE'  
go
```

```
Cache Name          Status Type      Config Value Run Value  
-----  
SALES_INSTANCE:log_sales Active Mixed 100.00 Mb    100.00 Mb  
-----  
Total 100.00 Mb 100.00 Mb
```

```
=====  
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed  
      Config Size: 100.00 Mb, Run Size: 100.00 Mb  
      Config Replacement: relaxed LRU, Run Replacement: strict LRU  
      Config Partition: 1, Run Partition: 1  
IO Size      Wash Size Config Size      Run Size      APF Percent  
-----  
2 Kb         20480 Kb 0.00 Mb    100.00 Mb    10  
(return status = 0)
```

---

**注释** 设置高速缓存替换策略不是动态的，需要重新启动 Adaptive Server 才能生效。

---

## 配置和使用多个缓冲池

使用 `sp_poolconfig` 可创建多个缓冲池。

### sp\_poolconfig

- 说明** 创建多个缓冲池。
- 语法** `sp_poolconfig cache_name [, "mem_size [P|K|M|G]", "config_poolK" [, "affected_poolK"]] [, 'instance instance_name']`
- 参数**
- `cache_name` — 现有数据高速缓存的名称。
  - `mem_size` — 要创建的内存池的大小，或具有指定 I/O 大小的现有池的新的总大小。池的最小大小为 256 个逻辑服务器页。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` — 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，最大是逻辑页大小的 8 倍。
  - `affected_pool` — 内存池中执行的 I/O 大小（在其中释放内存或修改池的特性，如清洗大小和预取限制）。如果未指定 `affected_pool`，则内存取自最低逻辑页大小的内存池。
- 示例** **为命名高速缓存创建一个 4K 池** 在实例 SALES\_INSTANCE 上执行 `sp_poolconfig 'tempdb_cache'` 将会显示：

```
sp_poolconfig 'tempdb_cache','25M','4K'
go
```

```
Cache Name      Status Type          Config Value  Run Value
-----
tempdb_cache Active Global,Mixed    100.00 Mb    100.00 Mb
(1 row affected)
```

```
-----
Total 100.00 Mb 100.00 Mb
```

```
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
Config Size: 100.00 Mb, Run Size: 100.00 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         15360 Kb       0.00 Mb         75.00 Mb     10
4 Kb         5120 Kb        25.00 Mb        25.00 Mb     10
```

(return status = 0)

**为本地高速缓存创建池配置** 可以为命名高速缓存 “log\_hr” 创建一个 8K 池。在实例 HR\_INSTANCE 上执行 sp\_poolconfig 'tempdb\_cache' 将会显示:

```
sp_poolconfig 'log_hr', '50M', '8K', 'instance HR_INSTANCE'  
go
```

Cache Name	Status	Type	Config Value	Run Value
HR_INSTANCE:log_hr	Active	Log Only	150.00 Mb	150.00 Mb

(1 row affected)

-----  
Total 150.00 Mb 150.00 Mb

=====  
Cache: HR\_INSTANCE:log\_hr, Status: Active, Type: Log Only  
Config Size: 150.00 Mb, Run Size: 150.00 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	20480 Kb	0.00 Mb	100.00 Mb	10
8 Kb	10240 Kb	50.00 Mb	50.00 Mb	10

(return status = 0)

---

**注释** 全局高速缓存没有特定于实例的缓冲池配置。实例选项用于检测必须有池配置的本地高速缓存配置。

---

## 在缓冲池之间移动内存

要创建新的 8K 池并从一个 4K 池（而不是缺省池）获得内存，请输入:

```
sp_poolconfig 'tempdb_cache', '8M', '8K', '4K'  
go  
sp_poolconfig 'tempdb_cache'  
go
```

Cache Name	Status	Type	Config Value	Run Value
tempdb_cache	Active	Global,Mixed	100.00 Mb	100.00 Mb

(1 row affected)

-----  
Total 100.00 Mb 100.00 Mb

```
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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
-----
8 Kb         1632 Kb          8.00 Mb         8.00 Mb       10
2Kb         15360 Kb         0.00 Mb         75.00 Mb      10
4 Kb         3480 Kb          17.00 Mb        17.00 Mb      10
(return status = 0)
```

## 更改池的清洗大小

内存池的清洗大小是高速缓存中供 Adaptive Server 将脏页写入磁盘的那一点。

```
sp_poolconfig cache_name, 'affected_poolK', 'wash=size[P|K|M|G]'
[, instance 'instancename']
```

要将命名高速缓存 “log\_hr” 的 8K 池的清洗大小更改为 12480K，请在实例 HR\_INSTANCE 上执行 sp\_poolconfig 'log\_hr' 以显示：

```
sp_poolconfig 'log_hr', '8K', 'wash=12480K', 'instance HR_INSTANCE'
go
```

```
Cache Name      Status Type      Config Value      Run Value
-----
HR_INSTANCE:log_hr Active Log Only      150.00 Mb        150.00 Mb
(1 row affected)
```

```
-----
Total 150.00 Mb 150.00 Mb
```

```
=====
Cache: HR_INSTANCE:log_hr, Status: Active, Type: Log Only
      Config Size: 150.00 Mb, Run Size: 150.00 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         20480 Kb       0.00 Mb         100.00 Mb     10
8 Kb         12480 Kb       50.00 Mb        50.00 Mb      10
(return status = 0)
```

## 更改池的本地异步预取百分比

本地异步预取是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的百分比。更改池的异步预取百分比：

```
sp_poolconfig cache_name, "affected_poolK ",
              "local async prefetch limit=percent "
```

要更改命名高速缓存 `log_sales` 的本地同步预取，请在实例 `SALES_INSTANCE` 上执行 `sp_poolconfig 'log sales'` 以显示：

```
sp_poolconfig 'log_sales','2K','local async prefetch limit=20','instance
              SALES_INSTANCE'
go
```

Cache Name	Status	Type	Config Value	Run Value
SALES_INSTANCE:log_sales	Active	Mixed	100.00 Mb	100.00 Mb

(1 row affected)

```
-----
Total 100.00 Mb 100.00 Mb
```

```
=====
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
       Config Size: 100.00 Mb, Run Size: 100.00 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	20480 Kb	0.00 Mb	100.00 Mb	20

(return status = 0)

## 在所有页的页面拆分期间使用异步写入

如果在页面拆分期间 `dump database` 命令处于活动状态，Cluster Edition 就会在拆分所生成的新页上发出同步写入命令。

## 删除缓冲池

可以通过将缓冲池的大小设置为 0 来删除缓冲池。该池中的内存会添加到缺省池中。要在实例 `SALES_INSTANCE` 上从命名高速缓存 `tempdb_cache` 中删除 4k 池，请输入：

```
sp_poolconfig 'tempdb_cache','0','4K'
go
sp_poolconfig 'tempdb_cache'
go
```

```

Cache Name                Status Type Config Value Run Value
-----
tempdb_cache              Active Global,Mixed 100.00 Mb 100.00 Mb
(1 row affected)

-----
Total 100.00 Mb 100.00 Mb
=====
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
      Config Size: 100.00 Mb, Run Size: 100.00 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
-----
8Kb          1632 Kb             8.00 Mb         8.00 Mb         10
2Kb          18840 Kb            0.00 Mb         92.00 Mb         10
(return status = 0)

```

## 将对象绑定到命名高速缓存

`sp_bindcache` 为高速缓存指派数据库、表、索引、文本对象或图像对象。在将实体绑定到高速缓存之前，必须满足以下条件：

- 命名高速缓存必须存在，并且其状态必须为 “Active”。
- 数据库或数据库对象必须存在。
- 只能从表、索引或对象所在的数据库中绑定表、索引或对象。
- 要绑定系统表（包括事务日志表 `syslogs`），数据库必须处于单用户模式。
- 必须从 `master` 数据库中绑定数据库。
- 必须将数据库、用户表、索引、文本对象或图像对象绑定到类型为 “Mixed” 的高速缓存。只有 `syslogs` 表可以绑定到类型为 “Log Only” 的高速缓存。
- 要将对象绑定到高速缓存，您必须拥有该对象或数据库，或者具有系统管理员状态。
- 将对象绑定到高速缓存是动态的。

---

**注释** 本地系统临时数据库的高速缓存绑定或解除绑定不是动态的，必须重新启动所有者实例，绑定才能生效。其它临时数据库（包括全局系统临时数据库）的高速缓存绑定或解除绑定是动态的。

---

## 绑定对象的语法

语法

```
sp_bindcache cache_name, dbname  
            [, [owner. ] tablename [, indexname ["text only"] ]
```

参数

*owner* — 可选（如果表由“dbo”所有）。

示例

要将数据库 SALES 绑定到命名高速缓存 sales\_cache，请输入：

```
sp_bindcache 'sales_cache', 'SALES'
```

用法

高速缓存绑定对整个集群都有效。没有特定于实例的高速缓存绑定。如果将对象绑定到本地高速缓存，则配置了该高速缓存的实例会使用该高速缓存，所有其它实例使用 default data cache。

---

**注释** 有关 sp\_bindcache 的完整描述，请参见《参考手册：过程》。

---

## 获取有关绑定的高速缓存的信息

如果提供高速缓存名，sp\_helpcache 会提供有关该高速缓存和绑定到该高速缓存的实体的信息。

例如：

```
sp_helpcache 'sales_cache'  
-----  
Cache Name           Config Size Run Size   Overhead  
-----  
sales_cache:SALES_INSTANCE 50.00 Mb 50.00 Mb 6.39 Mb  
(1 row affected)  
-----Cache Binding Information: -----  
Cache Name Entity Name Type Index Name  
Status  
-----  
sales_cache SALES      database  
V  
(return status = 0)
```

有关 `sp_helpcache` 语法，请参见《参考手册：过程》。

## 删除高速缓存绑定

有两个命令可用于删除高速缓存绑定：

- `sp_unbindcache` 解除高速缓存上单个实体的绑定。
- `sp_unbindcache_all` 解除高速缓存上所有对象的绑定。

语法

```
sp_unbindcache dbname [, [owner. ] tablename
[, indexname | "text only"] ]
```

示例

要删除数据库 “sales” 的高速缓存绑定，请输入：

```
sp_unbindcache 'SALES'
```

用法

- 删除对象的高速缓存绑定时，当前在内存中的所有页都会从高速缓存中清除。
- 当系统或远程本地临时数据库绑定到高速缓存时，您无法在命名高速缓存上运行 `sp_unbindcache_all`。这时，应该先使用 `sp_unbindcache` 解除每个数据库的绑定，然后运行 `sp_unbindcache_all`。

## 修改配置文件

配置文件中的命名高速缓存部分包含实例信息。全局高速缓存的高速缓存部分与非集群 Adaptive Server 输出类似。

下面的定义取自非集群 Adaptive Server 环境中的配置文件：

```
[Named Cache:tempdb_cache]
  cache size = 100M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

## 本地命名高速缓存的格式

下面显示了本地命名高速缓存的格式：

```
[Named Cache:log_sales]
  [Instance: SALES_INSTANCE]
  cache size = 100M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  [Instance: SALES_INSTANCE]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

在配置文件中应首先声明全局定义，然后声明本地定义。否则，服务器将无法启动。例如，如果没有特定于实例的高速缓存配置相对应，命名高速缓存上就不允许有特定于实例的池配置。下面的示例是非法的：

```
[Named Cache:tempdb_cache]
cache size = 100M
cache status = mixed cache
cache replacement policy = DEFAULT
local cache partition number = DEFAULT
[2K I/O Buffer Pool]
pool size = DEFAULT
wash size = DEFAULT
local async prefetch limit = DEFAULT
[instance: SALES_INSTANCE]
pool size = DEFAULT
wash size = 40960K
local async prefetch limit = DEFAULT
```

## 本地高速缓存条目中的额外行

本地高速缓存和缓冲池定义有一个额外的行 ([Instance: SALES\_INSTANCE]), 它告诉您该配置属于实例 SALES\_INSTANCE。如果命名高速缓存既有全局配置, 又有本地配置, 配置文件的高速缓存部分会显示:

```
[Named Cache:tempdb_cache]
  cache size = 100M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT
[Instance: SALES_INSTANCE]
  cache size = 150M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
[Instance: SALES_INSTANCE]
  pool size = DEFAULT
  wash size = 40960k
  local async prefetch limit = DEFAULT
```

## 具有全局配置的已删除命名高速缓存

删除命名高速缓存后, 高速缓存配置条目类似如下:

```
[Named Cache: tempdb_cache]
  cache size = 100M
  cache status = deleted
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

## 具有本地配置的命名高速缓存

如果命名高速缓存是在本地配置的，高速缓存部分的条目就会类似如下：

```
[Named Cache: tempdb_cache]
[Instance: SALES_INSTANCE]
  cache size = 100M
  cache status = deleted
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
[Instance: SALES_INSTANCE]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

## 具有有效配置的已删除条目

配置文件中不应包含仍具有至少一个有效配置的已删除条目，如下列高速缓存部分的条目：

```
[Named Cache: tempdb_cache]
  cache size = 100M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT
[Instance: SALES_INSTANCE]
  cache size = 150M
  cache status = deleted
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
[Instance: SALES_INSTANCE]
  pool size = DEFAULT
  wash size = 40960K
  local async prefetch limit = DEFAULT
```

## 在存在全局配置的情况下创建本地配置

在存在全局配置的情况下创建命名高速缓存的本地配置时，所有池条目都会针对本地配置而被复制。例如，您可以具有以下全局配置：

```
[Named Cache: tempdb_cache]
  cache size = 100M
  cache status = mixed cahe
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT

[4K I/O Buffer Pool]
  pool size = 25.0000m
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

如果在该全局配置上创建一个大小为 120M 的本地配置，则配置文件的高速缓存部分类似如下：

```
[Named Cache:tempdb_cache]
  cache size = 100M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT
[Instance:SALES_INSTANCE]
  cache size = 120M
  cache status = mixed cache
  cache replacement policy = DEFAULT
  local cache partition number = DEFAULT

[2K I/O Buffer Pool]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
[Instance:SALES_INSTANCE]
  pool size = DEFAULT
  wash size = DEFAULT
  local async prefetch limit = DEFAULT

[4K I/O Buffer Pool]
  pool size = 25.0000m
  wash size = DEFAULT
  local async prefetch limit = DEFAULT
```

```
[Instance:SALES_INSTANCE]
pool size = 25.0000m
wash size = DEFAULT
local async prefetch limit = DEFAULT
```

## 限制

- 如果命名高速缓存只有本地配置和对象绑定到其上，则无法创建全局高速缓存配置。您将会看到错误 19817。例如：

```
sp_cacheconfig 'SALES', '200M', 'instance SALES_INSTANCE'
go
sp_bindcache 'SALES','SALES_DB'
go
sp_cacheconfig 'SALES', '300m'
go
-----
Error 19817, "New configuration on cache 'SALES' is not permitted."
```

要避免上例中的这种情况，请解除所有对象与高速缓存“SALES”的绑定。为高速缓存“SALES”创建新的配置，并再次绑定对象。

- 为避免物理锁死锁，Cluster Edition 中的索引页不使用大池支持。

## 使用临时数据库

本章介绍了本地临时数据库和全局临时数据库、如何创建和管理这些数据库，以及如何将登录名或应用程序绑定到某个临时数据库或临时数据库组。本章还介绍了私有设备、如何创建和管理这些设备，以及如何针对本地用户临时数据库使用这些设备。

主题	页码
<a href="#">临时数据库的类型</a>	122
<a href="#">创建临时数据库</a>	125
<a href="#">将用户和应用程序绑定到临时数据库</a>	126
<a href="#">临时数据库的限制</a>	131
<a href="#">本地数据库的私有设备支持</a>	132

临时数据库可用于存储临时表，还可用于满足其它临时工作存储需求。**Cluster Edition** 既支持本地临时数据库，也支持全局临时数据库。本地临时数据库只能由拥有它的实例访问，并且主要用于存储特定于会话的私有临时对象：#表、工作表和虚拟表。全局临时数据库可由集群中的所有实例访问，并可用于存储在当前会话结束后仍可存在的临时对象。

**Cluster Edition** 支持：

- 本地系统临时数据库
- 本地用户临时数据库
- 全局用户临时数据库
- 全局系统临时数据库（dbid 为 2）

临时数据库是从设置了下列数据库选项的 **model** 数据库继承的：

- `select into/bulkcopy`
- `trunc log on chkpt`

**Guest** 用户自动添加到临时数据库中，并且所有用户将被授予 `create table` 权限。

## 临时数据库的类型

### 本地临时数据库

本地临时数据库有两种类型：

- 本地用户临时数据库
- 本地系统临时数据库

使用 **Cluster Edition**，可以为集群中的每个实例创建临时数据库。特定于实例的临时数据库称为本地临时数据库；拥有本地临时数据库的实例称为该数据库的所有者实例。

每个本地临时数据库都绑定到特定实例，并且可从该实例对该数据库进行访问。您必须为集群中的每个实例创建一个本地系统临时数据库，但可以选择是否创建本地用户临时数据库。

### 本地用户临时数据库

您可以为每个实例创建多个本地用户临时数据库，然后将应用程序或登录名绑定到单个本地用户临时数据库或临时数据库组。

### 本地系统临时数据库

本地系统临时数据库是每个实例必需的缺省临时数据库。当配置集群时，或者当向集群中添加新实例时，应配置本地系统临时数据库。除非为实例创建并使用本地用户临时数据库，否则实例会将所有特定于会话的临时对象（例如 # 表和工作表）都存储在本地系统临时数据库中。必须在共享存储上创建本地系统临时数据库。另请参见第 133 页的“[为临时数据库使用私有设备](#)”。

在非集群 Adaptive Server 环境中，会将系统临时数据库 `tempdb`（`dbid` 为 2）添加到 `default` 临时数据库组中。在 **Cluster Edition** 中，本地系统临时数据库不属于实例的 `default` 组。仅在当前实例的 `default` 组为空并且未指定其它绑定时，才会将本地系统临时数据库分配给会话。

---

**注释** 对于 Cluster Edition，分配给会话的缺省临时数据库是实例的本地临时数据库，而不是系统 `tempdb`（`dbid` 为 2）。您可能需要对非集群环境中那些之前在自己的操作中将缺省分配的临时数据库用作系统 `tempdb`（`dbid` 为 2）的应用程序进行修改，以确保这些操作现在会应用到所分配的本地临时数据库。

例如，在非集群 `Adaptive Server` 中，如果某个应用程序使用下面的命令截断缺省临时数据库的日志：

```
dump tran tempdb with truncate_only
```

在 Cluster Edition 中，必须将该应用程序修改为截断缺省分配的临时数据库的日志：

```
declare @tempdbname varchar(30)
select @tempdbname = db_name(@@tempdbid)
dump tran @tempdbname with truncate_only
```

---

### 会话临时数据库的单独用户日志高速缓存

Cluster Edition 支持多个临时数据库，例如用户本地临时数据库、系统本地临时数据库和全局临时数据库（即 `tempdb`）。

Cluster Edition 仅对于会话临时数据库而非任何其它临时数据库支持单独用户日志高速缓存。

会话临时数据库可以是：

- 用户本地临时数据库
- 系统本地临时数据库
- 集群处于 `tempdb` 配置模式下时的全局临时数据库

在 `tempdb` 配置期间，如果不存在本地系统或本地用户临时数据库，则会话缺省临时数据库为系统 `tempdb`（数据库 ID 为 2）。在系统 `tempdb` 配置期间，无法为其使用单独用户日志高速缓存。

## 全局临时数据库

创建主设备时，系统会自动创建全局系统临时数据库（dbid 为 2）。您可以创建其它全局临时数据库。

您可以从集群中的任何实例访问全局临时数据库，这样，便可以创建可在相同实例或不同实例上的会话间共享的临时对象。不能在实例上创建 # 表和工作表之类的临时对象，因为全局临时数据库不能分配给特定的会话。

全局临时数据库与常规用户数据库只有一点不同：每次启动集群时，都将重新创建全局临时数据库。全局临时数据库提供标准的日志记录、日志刷新、I/O 行为以及运行时回退功能。之所以支持这些功能，主要是为了与早期版本的 Adaptive Server 向后兼容。

当集群关闭时，全局临时数据库中的所有对象都将丢失。但是，如果实例失败，则实例故障切换恢复步骤也会恢复全局临时数据库中的对象，这样可确保在失败的实例上的某个会话期间创建的对象可继续在未失败的实例中使用。全局临时数据库可确保：只要集群中至少存在一个实例处于活动状态，临时可共享工作空间便会继续存在。

---

**注释** 自 Adaptive Server 12.5.0.3 以来创建的应用程序都可在用户创建的临时数据库中创建可共享表。为了确保现有应用程序可继续与 Cluster Edition 配合使用，需将包含这些可共享表的用户创建的临时数据库删除，并将其重新创建为同名的全局临时数据库。

---

## 摘要信息

表 8-1 总结了这些数据库的基本特性。

**表 8-1: 临时数据库的特性**

功能	本地系统临时数据库	本地用户临时数据库	全局用户临时数据库	全局系统临时数据库
支持的临时数据库对象	特定于会话的临时对象和常规数据库对象。	特定于会话的临时对象和常规数据库对象。	仅支持常规数据库对象，不支持特定于会话的临时对象。	仅支持常规数据库对象，不支持特定于会话的临时对象*
恢复	在拥有它的实例重新启动时重新创建。	在拥有它的实例重新启动时重新创建。	在集群重新启动时重新创建，并在实例失败时以事务性方式恢复。	在集群重新启动时重新创建，并在实例失败时以事务性方式恢复。

功能	本地系统临时数据库	本地用户临时数据库	全局用户临时数据库	全局系统临时数据库
易用性	仅可从拥有它的实例访问，但必须在共享设备上创建，才可从其它实例中创建和删除它。	仅可从拥有它的实例访问。	可从任何实例访问。	可从任何实例访问。
创建	用户创建，每个实例 1 个（必需）。	用户创建，每个实例 0 个或多个。	用户创建，相应集群 0 个或多个。	系统生成 (dbid = 2)，相应集群 1 个。
是否允许绑定？	否	是	否	否
存储（共享设备或私有设备）	仅限共享存储	共享存储和本地存储	仅限共享存储	仅限共享存储

\* 在 tempdb 配置模式下，全局系统临时数据库针对启动协调器的行为与本地系统临时数据库一样，因此，它既支持特定于会话的临时对象，也支持常规数据库对象。

## 创建临时数据库

在 Cluster Edition 中，每个集群具有一个全局系统临时数据库，并且集群中的每个实例都具有一个本地系统临时数据库。其它临时数据库是可选的。全局系统临时数据库是在安装 Adaptive Server 时创建的。本节介绍了如何创建其它临时数据库。

Sybase 建议您创建大小类似的临时数据库时，这样，应用程序在访问每个实例时便不必考虑需要多少临时空间。

## 创建本地系统临时数据库

在最初启动集群和在稍后每当为该集群添加实例时，都必须为每个实例创建本地系统临时数据库。请在共享设备上创建本地系统临时数据库。您可以从任何实例创建或删除本地系统临时数据库，但是，只能从拥有相应本地系统临时数据库的实例访问该数据库。

可以使用 Adaptive Server 插件或 sybcluster 创建本地系统临时数据库。请参见所用平台的安装指南。

## 创建本地和全局用户临时数据库

您可以随时创建本地用户临时数据库和全局用户临时数据库。

创建本地用户临时数据库时，必须从拥有它的实例中创建。例如，若要在 `ase1` 上创建本地临时数据库 `local_temp1`，请输入：

```
create temporary database local_temp1
for instance ase1 on default = 50
```

您可以从集群中的任意实例创建全局用户临时数据库。例如，若要在 `ase1` 上创建全局临时数据库 `global_tempdb1`，请登录到 `ase1` 并输入：

```
create global temporary database global_tempdb1
on default = 50
```

---

**注释** 实例 `instance_name` 的 `create database` 短语是可选的。如果不包括此短语，则 Adaptive Server 会为当前实例创建该临时数据库。有关其完整语法的说明，请参见《参考手册：命令》。

---

## 将用户和应用程序绑定到临时数据库

您可以使用 `sp_tempdb` 将登录名和应用程序绑定到临时数据库或临时数据库组，并且可以使用它在非集群 Adaptive Server 和 Cluster Edition 中创建临时数据库组。在支持本地临时数据库和全局临时数据库的 Cluster Edition 中，您可以将登录名或应用程序绑定到本地用户临时数据库或临时数据库组，但不能绑定到全局临时数据库。

`default` 临时数据库组是系统生成的，始终存在。除非使用 `sp_tempdb` 向该数据库添加本地临时数据库，否则它将为空。

---

**注释** 在 Cluster Edition 中，系统临时数据库 `tempdb`（`dbid` 为 2）是全局系统临时数据库，不属于 `default` 组。此行为与非集群 Adaptive Server 中系统临时数据库 `tempdb`（`dbid` 为 2）的行为不同。在非集群 Adaptive Server 中，缺省情况下，系统临时数据库 `tempdb` 属于 `default` 组。

---

`sp_tempdb` 用户界面包括选项 `unbindall_gr`，该选项可在非集群 Adaptive Server 或 Cluster Edition 中删除指定组的所有绑定。此外，`unbind` 选项包括特定于 Cluster Edition 的参数 `instance_name`。

## 创建和管理临时数据库组

对于 Adaptive Server 的所有版本，临时数据库组的创建和管理都是相同的。

**注释** 在 Cluster Edition 中，本地系统临时数据库并不是 default 组的成员。

- 若要创建临时数据库组，请使用 `sp_tempdb create`。例如，若要创建名为 `tempdbgroup1` 的临时数据库组：

```
sp_tempdb "create", "tempdbgroup1"
```

- 若要删除临时数据库组，请使用 `sp_tempdb drop`。例如，若要删除名为 `tempdbgroup1` 的临时数据库组：

```
sp_tempdb "drop", "tempdbgroup1"
```

- 若要向临时数据库组中添加数据库，请使用 `sp_tempdb add`。例如，若要将数据库 `local_tempdb1` 添加到名为 `tempdbgroup1` 的临时数据库组中：

```
sp_tempdb "add", "local_tempdb1", "tempdbgroup1"
```

- 若要从临时数据库组中删除数据库，请使用 `sp_tempdb remove`。例如，若要将数据库 `local_tempdb1` 从名为 `tempdbgroup1` 的临时数据库组中删除：

```
sp_tempdb "remove", "local_tempdb1", "tempdbgroup1"
```

有关 `sp_tempdb` 的完整语法和用法信息，请参见《参考手册：过程》。

## 可以绑定的内容

您可以进行以下绑定：

- 将一个用户登录名或应用程序绑定到多个本地用户临时数据库，但对于每个实例，仅绑定到一个数据库。

例如，您可以将“sa”绑定到“ase1”上的 `local_tempdb1` 和“ase2”上的 `local_tempdb2`，但无法将“sa”绑定到“ase1”上的 `local_tempdb3`。

- 将一个用户登录名或应用程序仅绑定到一个数据库组。临时数据库组将出现在所有实例中。

例如，如果将 `isql` 绑定到 `default` 组，则无法再将 `isql` 绑定到其它临时数据库组。

- 将一个用户登录名或应用程序绑定到单个临时数据库或一个临时数据库组，但不能同时绑定到这两者。

例如，如果“sa”绑定到了“ase1”上的 local\_tempdb1，则无法再将其绑定到 default 组。

## 如何确定会话绑定

当启动会话时，Adaptive Server 会检查是否有适用的绑定。如果存在多个适用的绑定，则 Cluster Edition 会根据以下算法确定使用哪个绑定。一个会话仅支持一个绑定。

- 1 如果当前登录名存在绑定，则使用该绑定。否则，Cluster Edition 将搜索当前应用程序的绑定。如果这两个绑定都已找到，则使用登录名的绑定：登录名绑定优先于应用程序绑定。

例如，如果用户“sa”绑定到实例“ase1”上的 tempdb1，而 isql 绑定到“ase1”上的 tempdb2，则当用户“sa”使用 isql 在“ase1”上启动会话时，只会使用“sa”用户绑定。

---

**注释** 如果第一个绑定失败，Cluster Edition 会自动分配 default 组的成员。Cluster Edition 不会搜索其它绑定。

---

- 2 如果绑定的目标是数据库，则分配数据库。如果绑定的目标是数据库组，则分配从当前实例以循环方式选择的成员数据库。作为组成员的本地临时数据库只能分配给在所有者实例上建立的会话。无法使用组绑定将它们分配给其它实例上的会话。
- 3 如果找不到绑定或者无法进行分配，则会尝试分配以循环方式选择的 default 组成员。
- 4 如果无法对 default 组的成员执行分配，则对本地系统临时数据库执行分配。

## 创建和管理绑定

`sp_tempdb bind` 可将登录名或应用程序绑定到本地临时数据库或数据库组。例如，若要将 “sa” 登录名绑定到 `default` 组，请输入：

```
sp_tempdb "bind", "LG", "sa", "GR", "default"
```

若要将 `isql` 绑定到 `ase1` 上的 `local_tempdb1`，请输入：

```
sp_tempdb "bind", "AP", "isql", "DB", "local_tempdb1"
```

`sp_tempdb unbind` 可删除一个或多个绑定。对于 Cluster Edition，此命令包括 `instance_name` 参数，使用该参数，可以删除特定本地临时数据库的绑定。

例如，若要取消 `isql` 与 “ase2” 上的 `local_tempdb2` 的绑定，请输入：

```
sp_tempdb "unbind", "AP", "isql", NULL, "ase2"
```

---

**注释** 如果 `isql` 绑定到了数据库组，则上述命令会删除该组的此绑定。如果 `isql` 存在多个数据库绑定，则上述命令只删除 “ase2” 的绑定；其它实例上的临时数据库的绑定不会受影响。

---

若要取消用户登录名或应用程序与数据库组或任何临时数据库的绑定，请将 `unbind` 参数仅与相应登录名或应用程序名称一起使用。例如，若要删除 “sa” 登录名的绑定：

```
sp_tempdb "unbind", "lg", "sa"
```

若要取消所有登录名和应用程序与特定组的绑定，请使用 `sp_tempdb unbindall_gr`。例如，若要删除 `tempdbgroup1` 的所有绑定，请输入：

```
sp_tempdb "unbindall_gr", "tempdbgroup1"
```

若要取消所有登录名和应用程序与特定数据库的绑定，请使用 `sp_tempdb unbindall_db`。例如，若要取消 `localtempdb1` 的所有绑定，请输入：

```
sp_tempdb "unbindall_db", "localtempdb1"
```

## 显示组和绑定信息

若要显示现有组、组成员以及登录名和应用程序绑定的列表，请使用 `sp_tempdb show`。例如，若要查看临时数据库组 “tempdbgroup1” 的成员及其所有者实例，请输入：

```
sp_tempdb "show", "gr" "tempdbgroup1"
```

若要显示分配了临时数据库的活动会话的列表，请使用 `sp_tempdb who`。例如，若要显示分配了临时数据库 “localtempdb1” 的活动会话，请输入：

```
sp_tempdb "who", "localtempdb1"
```

## 删除临时数据库

删除临时数据库时有一些限制：

### 删除最后一个本地系统临时数据库以外的所有本地系统临时数据库

由于只要本地系统临时数据库的所有者实例在运行，相应本地系统临时数据库就会处于使用状态，因此当相应本地系统临时数据库的所有者实例在运行时，无法删除该本地系统临时数据库。

若要删除一个本地系统临时数据库，请关闭相应的实例，然后从另一实例中删除该本地系统临时数据库。

### 删除最后一个本地系统临时数据库

删除最后一个实例的本地系统临时数据库：

- 1 使用 `sp_tempdb_markdrop` 标记要删除的数据库。例如，如果 `ase3_tdb1` 是最后一个本地系统临时数据库，请使用以下命令标记它：

```
sp_tempdb_markdrop ase3_tdb1
```

- 2 关闭并重新启动最后一个实例。将不会使用标记为 “删除” 的本地系统临时数据库。
- 3 从此实例中删除该临时数据库。

## 临时数据库的限制

在 Cluster Edition 中：

- 尝试访问另一实例拥有的本地临时数据库会导致发生以下错误：

```
Error Number : 969
```

```
You can access database '<local-tempdbname>' only
from its owner instance '<owner-instance-name>'. You
cannot access local temporary databases from non-
owner instances except to use CREATE DATABASE and
DROP DATABASE with local system temporary databases.
```

一般而言，您必须执行用于从所有者实例访问本地临时数据库的存储过程和命令。或者，您可以使用下列方法对远程实例的本地临时数据库执行操作：

- 在访问任何本地临时数据库之前，使用 `connect to instance_name` 连接到相应的所有者实例。
- 使用 `sp_remotesql` 将 Transact-SQL 语句发送到该远程实例。例如，若要对实例 `ase1` 拥有的本地临时数据库 `local_tempdb_ase1` 执行查询：

```
sp_remotesql "ase1", "select * from local_tempdb_ase1..sysobjects"
```

- 对于存储过程，可以执行用于指定实例名称的过程。例如，若要对实例“ase1”的 `local_tempdb_ase1` 执行 `sp_spaceused`：

```
ase1.local_tempdb_ase1.dbo.sp_spaceused
```

---

**注释** 使用 `db_instanceid` 可确定本地临时数据库的所有者实例 ID。

---

- 一些用于对非集群 Adaptive Server 中的所有数据库执行操作的存储过程和命令可能会跳过针对 Cluster Edition 中的远程本地临时数据库的这些操作。若要确保在所有数据库中都执行这些操作，请再次从所有者实例对每个跳过的数据库执行操作。

例如，如果在未指定数据库名称的情况下执行 `sp_dbcc_faultreport`，它将跳过故障报告中的远程本地临时数据库。

- 无法向本地临时数据库的用户授予或撤消用户定义的角色。
- 无法授予或撤消对本地临时数据库中对象的权限。
- 无法使用本地临时数据库作为任何登录名的缺省数据库；如果尝试这样做，会导致 `sp_addlogin` 和 `sp_modifylogin` 失败。

- 不能对本地临时数据库纳入引用某个列的参照完整性约束，除非所引用的列来自同一本地临时数据库中的表。如果 `create table`、`alter table` 和 `create schema` 尝试为某一本地临时数据库创建对不同数据库中的列的引用，则它们会失败。
- 不能使用位于本地临时数据库中的加密密钥为列加密，除非表位于同一本地临时数据库上。如果 `create table` 和 `alter table` 尝试使用位于本地临时数据库中的加密密钥为列加密，而表位于另一数据库中，则它们会失败。
- 您无法指定任何本地系统临时数据库的数据库恢复顺序。本地系统临时数据库是用户创建的，但始终可以使用系统数据库恢复它们。
- 您无法从本地临时数据库上下文内使用 `sp_configure` 来更改配置选项。

## 本地数据库的私有设备支持

通过 Cluster Edition，可以使用成本低廉的存储（例如本地磁盘和文件系统设备）来满足特定于实例的临时空间需求。您可以使用这种存储（创建为私有设备）在私有设备上创建特定于实例的本地用户临时数据库。但是，不要为本地系统临时数据库使用私有设备。必须在共享存储上创建本地系统临时数据库。

在 Cluster Edition 中，私有设备仅由一个集群实例拥有和访问。说设备是私有的是说它的逻辑属性；可能可以也可能不可以从非所有者实例访问该设备的物理路径。

在 `sysdevices` 表中，私有设备由位图列 `status2` 标识，拥有该设备的实例由属于该设备的行的 `instanceid` 列标识。

尽管一个实例可以拥有许多私有设备，但一个私有设备只能由一个实例拥有。

私有设备只能用于拥有该私有设备的实例上的本地用户临时数据库。它不能用作缺省存储；如果针对某私有设备调用 `sp_diskdefault`，则存储过程会失败，并且会显示一条错误消息。

无法对私有设备执行镜像、取消镜像或重新镜像操作。

## 为临时数据使用私有设备

为了在 Cluster Edition 中使临时数据实现最佳性能，Sybase 建议您使用快速本地磁盘存储本地用户临时数据库。必须在共享存储上创建本地系统临时数据库。若要限制临时数据对共享存储的使用：

- 1 在本地磁盘上创建一个或多个私有设备。
- 2 在私有设备上创建一个或多个本地用户临时数据库。
- 3 将这些本地用户临时数据库添加到 **default** 临时数据库组中。

对于不具有显式临时数据库绑定的用户连接，将为其分配 **default** 组中的成员数据库。这会将在共享存储上创建的本地系统临时数据库限制为只能用于执行内部系统任务，而所有用户连接都使用在快速本地磁盘上创建的本地用户临时数据库。

## 使用 *disk init* 创建私有设备

使用 *disk init* 可以创建私有设备。*disk init* 包括一个可选的 *instance* 参数，该参数将设备标记为归某个实例私有。

此示例创建一个名为 *private\_dev1* 的归 *cluster1\_1* 实例私有的设备：

```
disk init
name = "private_dev1",
physname = "/usr/u/sybase/data/private_dev1.dat",
vdevno = 2, size = 5102, instance = "cluster1_1"
```

您可以从集群中的任何实例执行 *disk init* 命令。该命令在内部传递到将拥有该设备的实例。但是，该拥有设备的实例必须启动并运行才能处理该命令。如果不这样，该命令将会失败，并且不会创建私有设备。

## 使用 *disk reinit* 重新初始化私有设备

在恢复 *master* 数据库的过程中，使用 *disk reinit* 可恢复 *master* 数据库的 *sysdevices* 表中的私有设备条目。*disk reinit* 包括可选的参数 *instance*，该参数可将设备标记为归某实例私有。

此示例将 *private\_dev1* 设备恢复为 *master* 数据库的 *sysdevices* 表中实例 *cluster1\_1* 的私有设备：

```
disk reinit
name = "private_dev1"
physname = "/usr/u/sybase/data/private_dev1.dat"
```

```
vdevno = 2, size = 5120, instance = "cluster1_1"
```

可以从集群内的任意实例执行此命令。如果拥有相应设备的实例已启动并正在运行，则会将该命令传递给它。

如果拥有相应设备的实例并未启动和运行，则 `disk reinit` 会在 `sysdevices` 中插入一个与该设备相对应的行，但在该实例运行前，它不会激活该行。如果必须在发出 `disk refit` 前恢复设备，则这一点非常有用。

---

**注释** 为未运行的实例恢复私有设备时，要额外注意所传递的参数。在这些情况下，Adaptive Server 无法验证私有设备的物理路径；它假定用户提供的物理路径有效。

---

## 使用 `sp_dropdevice` 删除私有设备

使用 `sp_dropdevice` 删除私有设备。可以从集群内的任意实例执行此过程。但是，如果拥有相应私有设备的实例未运行，则此过程无法删除该私有设备。有关使用 `sp_dropdevice` 的详细信息，请参见《参考手册：过程》。

## 使用 `sp_helpdevice` 显示私有设备信息

`sp_helpdevice` 可报告设备是否是私有的。如果设备是私有的，则 `sp_helpdevice` 会输出集群中拥有该私有设备的实例的名称。

在下例中，`'regular_dev'` 是常规可共享设备，而 `'private_dev1'` 是实例 `'cluster1_1'` 拥有的私有设备。

```
1> sp_helpdevice
2> go
device_name
  physical_name
  description
  status cntrltype vdevno      vpn_low      vpn_high
```

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

```

-----
master
  /remote/percy_dev/user1/sdc-DBs/master.dat
  file system device, special, dsync on, directio off, default disk,
  physical disk, 512.00 MB, Free: 351.00 MB
      3          0          0          0          262143
private_dev1
  /percy_dev/user1/sdc-DBs/private_dev1.dat
  file system device, special, private 'cluster1_1', MIRROR DISABLED, non
  serial writes, dsync off, directio on, physical disk, 2.00 MB
      2050          0          1          0          1023
regular_dev1
  /percy_dev/user1/sdc-DBs/dev1.dat
  file system device, special, dsync off, directio on, physical disk, 2.0
  0 MB
      2          0          2          0          1023
tapedump1
  /dev/nst0
  unknown device type, disk, dump device
      16          2          0          0          20000
tapedump2
  /dev/nst1
  unknown device type, tape,          625 MB, dump device
      16          3          0          0          20000

(5 rows affected)
(return status = 0)

```

如果 `disk refit` 在某一设备上挂起，则 `sp_helpdevice` 会在其输出中反映这一点。

在此示例中，`'regular_dev'` 是常规可共享设备，而 `'private_dev1'` 和 `'private_dev2'` 是实例 `'cluster1_1'` 拥有的私有设备。进一步的 `disk refit` 在 `'private_dev1'` 上挂起。

```

1> sp_helpdevice
2> go
  device_name
  physical_name
  description
  status cntrltype vdevno          vpn_low          vpn_high

```

```

-----
-----
-----
-----

```

```

-----
-----
-----
master
  /remote/percy_dev/user1/sdc-DBs/master.dat
  file system device, special, dsync on, directio off,
  default disk, physical disk, 512.00 MB, Free: 351.00 MB
      3          0          0          0          262143
private_dev1
  /percy_dev/user1/sdc-DBs/private_dev1.dat
  file system device, special, private 'cluster1_1',
  REFIT PENDING, MIRROR DISABLED, nonserial writes, dsync off, directio
  on, physical disk, 2.00
MB
  2050          0          1          0          1023
private_dev2
  /percy_dev/user1/sdc-DBs/private_dev2.dat
  file system device, special, private 'cluster1_1', MIRROR DISABLED, non
  serial writes, dsync off, directio on, physical disk, 2.00
MB
  2050          0          3          0          1023
regular_dev1
  /percy_dev/user1/sdc-DBs/dev1.dat
  file system device, special, dsync off, directio on, physical disk, 2.00
MB
  2          0          2          0          1023
tapedump1
  /dev/nst0
  unknown device type, disk, dump device
      16          2          0          0          20000
tapedump2
  /dev/nst1
  unknown device type, tape,          625 MB, dump device
      16          3          0          0          20000
(6 rows affected)
(return status = 0)

```

## 对私有设备使用 `create database` 和 `alter database`

由于私有设备只属于集群中的一个实例，因此只能用于拥有它的实例上的本地用户临时数据库。当使用私有设备创建或扩展数据库或其日志时，除非该设备用于拥有它的实例的本地用户临时数据库，否则 `create database` 或 `alter database` 会失败，并会显示一条错误消息。

## 使用 `disk refit`

如果集群中没有私有设备，则 `disk refit` 会像平常一样运行。

如果集群中有一个或多个私有设备，则 `disk refit` 会变为一个分为两阶段的操作：

- 阶段一只对启动 `disk refit` 的实例可以访问的那些设备（常规可共享设备以及该实例拥有的私有设备）执行该命令。
- 阶段二只对集群中实例的私有设备执行 `disk refit`，一次针对一个实例执行。只有当这两个阶段均成功完成后，`disk refit` 才会完成。

### ❖ 当集群中存在私有设备时执行 `disk refit`

您必须在执行 `disk refit` 之前正确恢复 `sysdevices` 表。针对 `sysdevices` 表中任何缺少的设备条目，使用 `disk reinit` 命令可添加常规或私有设备。

有关恢复 `sysdevices` 的详细信息，请参见《系统管理指南：第二卷》。

- 1 在单用户模式下重新启动集群。通过使用 `-m` 选项启动集群中的实例可以完成此操作。单用户模式只要求集群中有一个实例在运行，并且没有其它活动连接指向集群。您必须设置跟踪标志 3608。
- 2 对相应的实例发出 `disk refit` 命令。这会删除 `sysdatabases` 和 `sysusages` 中的所有行，并从 `sysdevices` 中的设备（可由集群中的所有实例访问的常规可共享设备，或归该实例私有的设备）重建它们中的一部分。在阶段一结束后，集群会关闭。

Adaptive Server 会在阶段一结束时输出一些消息，指出 `disk refit` 是否完成。如果未完成，Adaptive Server 还会输出消息来通知用户集群中有哪些其它实例拥有私有设备，必须执行阶段二。

或者，您可以使用 `sp_refit_admin 'status'` 查找尚未针对其运行 `disk refit` 的私有设备。



## 在群集环境中运行 Job Scheduler

本章介绍如何在共享磁盘集群环境中运行 Job Scheduler。有关安装、启动和配置 Job Scheduler 的信息，请参见《Job Scheduler 用户指南》。

主题	页码
<a href="#">安装和配置 Job Scheduler</a>	139
<a href="#">在群集环境中运行 Job Scheduler</a>	139
<a href="#">关闭 Job Scheduler</a>	140
<a href="#">重定向预定作业</a>	140

Cluster Edition 允许您在共享磁盘集群环境中运行 Job Scheduler。单个 Job Scheduler 便可处理集群中的所有实例发出的所有作业调度请求。Job Scheduler 在启动协调器上运行。

如果在启动集群后手动启动 Job Scheduler，集群就会选择由协调器来承载 Job Scheduler。

### 安装和配置 Job Scheduler

运行 Job Scheduler 的协调器也叫做“Job Scheduler 实例”。要设置和启动 Job Scheduler，请按照《Job Scheduler 用户指南》的第 2 章“配置和运行 Job Scheduler”中的指导来进行操作。有关在 Cluster Edition 中安装和配置 Job Scheduler 的说明，请参见所用平台的安装指南。

### 在群集环境中运行 Job Scheduler

如果用户在 Job Scheduler 实例以外的实例上执行 Job Scheduler 操作，该实例就会向 Job Scheduler 实例发出请求。Job Scheduler 会处理该请求并向发出请求的实例发送答复。

## 关闭 Job Scheduler

在共享磁盘集群中关闭 Job Scheduler 和在非集群 Adaptive Server 配置中关闭它是一样的。

从 Adaptive Server Sybase Central 插件中：

- 1 从任意 Adaptive Server 实例中，右键单击 Scheduled Jobs 文件夹（Scheduled Jobs 文件夹显示在每个实例下面，每个实例都可以从该插件中启动所有 Job Scheduler 命令）。
- 2 选择“管理”以打开“Job Scheduler 管理”对话框。
- 3 从“任务配置”选项卡上，选择“中止”。

从命令行中运行：

```
use sybmgmtadb
go
sp_js_wakeup "stop_js", 1
go
```

## 重定向预定作业

当 Job Scheduler 代理尝试连接到一个实例以执行预定作业，但该实例太忙时，负载管理器可以将连接重定向到另一个实例，因此，集群中的任何实例都可以执行该预定作业。这是自动发生的，不需要您重新配置 Job Scheduler。

通常，会由负载管理器负责执行登录重定向。您可以通过在逻辑集群级别设置规则来影响连接的重定向方式。但如果不希望重定向预定作业，请设置新的用于允许重定向的作业属性：

- 从 Adaptive Server 插件中 — 在“作业属性”对话框中取消选择“允许重定向”。

缺省行为是允许负载管理器根据在建立连接时存在的负载规则来重定向预定作业连接。

- 从命令行中 — 当创建或修改作业时，设置重定向属性 `no_conn_redirection`。例如，要使用 `sp_sjobcreate` 为名为 `find_old_logins` 的作业设置该属性，请输入：

```
sp_sjobcreate @name='jname=find_old_logins',  
@option='jcmd=exec  
sp_find_old_logins,jproperties=no_conn_redirection=  
true'
```



## 实例恢复

本章介绍对集群实例故障（单一故障和多个故障）的支持。15.5 版之前的 Cluster Editions 仅支持单一实例故障；Sybase 15.5 版及更高版本支持多个同时发生的故障。

主题	页码
<a href="#">单一实例恢复</a>	143
<a href="#">多个同时发生的故障</a>	144
<a href="#">恢复算法</a>	146

### 单一实例恢复

Cluster Edition 提供了故障切换恢复，在恢复期间，数据库保持联机状态并可供使用。但是，如果所请求的数据已由出现故障的实例修改，且必须进行恢复，则用户进程将被阻止，直至故障切换恢复进程将数据恢复为一致状态为止。

Cluster Edition 支持：

- 在集群启动时进行的恢复，该恢复与您执行服务器冷重新启动时发生的非集群 Adaptive Server 恢复相同。
- 故障切换恢复，该恢复可对出现故障的实例在其它实例使用共享磁盘集群中的相同数据库时所修改的数据进行恢复。在本章中，“故障切换”是指实例故障切换，而非客户端连接故障切换。

恢复在集群成员资格服务检测到故障时开始。协调器实例上的恢复事件处理程序恢复系统数据库。然后，未失败的实例（包括协调器实例）以并行方式恢复用户数据库。每个实例上的其中一个 PCM 线程将执行恢复任务。协调器实例上的其它线程以及其它实例将进行其它活动。

如果跟踪标志 3483 处于启用状态，则会禁用分布式故障切换恢复，并且协调器实例上的恢复事件处理程序将以串行顺序恢复用户数据库。

在恢复完成之前，将一直阻止尝试访问由出现故障的实例所修改的数据的进程。

Cluster Edition 按照用户数据库的数据库 ID (dbid) 指定的顺序恢复用户数据库，且在实例故障切换恢复期间不使用任何用户定义的恢复顺序。

## 单一事务日志

集群的每个数据库只有一个日志，并且该日志进行了逻辑分区。Cluster Edition 的日志记录标记包含实例 ID，该实例 ID 在内部用于扫描特定实例记录的日志记录，尤其是在故障切换恢复期间记录的日志记录。检查点日志记录始终具有值为 0 的实例 ID，因此它始终包含在日志扫描范围内，而与日志扫描过程中指定的实例 ID 无关。

## 多个同时发生的故障

在 15.5 版及更高版本中，Cluster Edition 支持多个同时发生的实例故障，当单一集群视图内有多个实例出现故障时，集群将保持联机状态，并提供与单一实例出现故障时同样的故障切换恢复功能。

Cluster Edition 提供了配置参数 `cluster redundancy level`，该配置参数允许数据库管理员为集群设置可恢复的同时发生的实例故障的最大数量。

## 启用多个同时进行的故障切换

可恢复的出现故障的实例数是集群冗余级别 (CRL)，它也是确定此值的配置参数的名称：`cluster redundancy level`。CRL 是可同时出现故障（同时允许恢复以并发方式继续执行其它活动）的实例的最大数量。如果出现故障的实例数超出相应配置参数指定的最大数量，则集群将关闭。

如果将 CRL 配置为大于 1 的数量，则也会增加集群中的锁复制。CRL 位于物理集群级，并始终优先于故障切换逻辑集群配置；换句话说，您的 CRL 不能超出集群中物理实例的数量。

`cluster redundancy level` 是静态的集群参数，其缺省值为 1。该参数的最小允许值为 1，其最大允许值比 `cluster.cfg` 或仲裁文件中指定的 'maximum number of instances' 小 1。如果一个或多个实例离开集群，`cluster redundancy level` 保持不变。

`cluster redundancy level` 是 `sp_configure` 参数：

```
sp_configure 'cluster redundancy level', config_value
```

故障切换恢复可恢复出现故障的实例在其它实例正在使用同一数据库时所修改的数据。在 CRL 为 1（缺省值）的情况下，Adaptive Server 至少会保留集群中每个锁的两个副本。因此，如果一个实例出现故障，锁的一个副本将保留，以便 Adaptive Server 能够标记由出现故障的实例修改的数据。同样，在 CRL 为“n”的情况下，Adaptive Server 至少具有集群中每个锁的“n+1”个副本，并且能够为“n”个同时进行的故障切换执行故障切换恢复。

若要启动集群，`cluster redundancy level` 的值至少要比 `cluster.cfg` 或仲裁文件中指定的 `maximum number of instances` 的值小 1。因此，如果进行以下任一设置，则集群无法启动：

- `maximum number of instances` 值等于或小于 `cluster redundancy level` 的值，或者
- `ofcluster redundancy level` 的值等于或大于 `maximum number of instances` 的值

在启动集群之前设置 `cluster redundancy level`。如果在运行时更改该参数的值，您必须重新启动集群。

## 性能限制和集群冗余级别

当 CRL 值增加时，集群可支持的同时发生的实例故障数也会增加。但是，由于集群中有锁的多个副本，因此 CRL 值的增加也会导致消息传送通信量的增加。必须增加开销才能维持这一冗余级别。

例如，其它配置参数（例如 `number of locks` 和 `cache size`）对于大于 1 的 CRL 值也需要更多资源，这意味着您必须为相同的 `number of locks` 值增加 `max memory`。CRL 相当耗费资源，即使 Adaptive Server 以透明方式执行开销值计算也是如此。

除非多实例故障很常见，否则 Sybase 建议您将 `cluster redundancy level` 设置为 1。

# 恢复算法

非集群 Adaptive Server 中的内部数据库恢复进程类似于 Cluster Edition 的恢复进程。必须为单一实例故障恢复和多实例故障恢复执行相同的阶段。

**表 10-1: 集群服务器和非集群服务器的恢复步骤**

非集群恢复阶段	Cluster Edition 恢复阶段
1) 估算可恢复的日志边界 — 可恢复的日志范围为从恢复检查点中记录的最早活动事务到日志末尾。	1) 估算可恢复的日志边界（与非集群 Adaptive Server 中的相同）。
2) 分析 — 从最早的活动事务到日志末尾向前扫描可恢复的日志，生成作为不完整事务的恢复信息等。此信息在重做、撤消和撤消后传递过程中使用。	2) 分析正在恢复的数据库日志（与非集群 Adaptive Server 中相同）。
3) 重做 — 向前扫描可恢复的日志，并根据需要重新执行日志记录指定的操作。	3) 重做 — 需要未完成事务的锁。
	4) 保留补偿日志记录空间以撤消出现故障的实例上未完成的事务。
	5) 释放出现故障的实例在出现故障前获取的所有锁，不包括在第 3 步获得的用于未完成事务的锁。
4) 撤消 — 从日志末尾向回撤消至最早的未完成事务的开头，并撤消未完成事务的日志记录指定的操作。Adaptive Server 将为每个事务记录一个补偿日志记录。	6) 撤消 — 另外，未完成事务的逻辑锁按与已完成事务的逻辑锁相同的方式释放。
	7) 填充可用空间信息（阈值管理器恢复）。
5) 撤消后 — 在数据库中记录一条检查点记录，填充可用空间计数恢复信息（阈值恢复），并清除高速缓存。	8) 撤消后 — 刷新恢复实例上的所有脏缓冲区。Cluster Edition 在实例故障切换恢复之后不执行检查点。

**注释** 请参见《系统管理指南：第二卷》中的第 11 章“开发备份和恢复计划”。

## 附加主题

本章详细介绍集群体系结构的一些元素。许多主题介绍了集群和非集群 Adaptive Server 中提供的特定于集群的功能特点。

主题	页码
<a href="#">锁</a>	148
<a href="#">内存</a>	149
<a href="#">阈值</a>	150
<a href="#">集群进程间通信</a>	152
<a href="#">分布式检查点</a>	152
<a href="#">仲裁设备心跳</a>	152
<a href="#">使用 InfiniBand</a>	153
<a href="#">专用安装模式</a>	155
<a href="#">在集群环境中使用 Java</a>	157
<a href="#">向存档数据库添加空间</a>	158
<a href="#">共享磁盘集群中的分布式事务</a>	158
<a href="#">支持 mount 和 unmount 命令</a>	164
<a href="#">使用 sp_showplan</a>	164

非集群 Adaptive Server 的许多子系统都在共享磁盘集群环境下工作；已经专门为 Cluster Edition 开发了其它子系统。这些子系统是：

- 锁管理器
- 缓冲区管理器
- 集群进程间通信 (CIPC)
- 恢复（开始时间和故障切换处理）

## 锁

可以通过集群中的任何实例访问和高速缓存所有数据库数据和元数据对象。因此，这些对象像运行时全局数据结构和变量一样，遵循分布式锁定和高速缓存一致性强制要求。

集群锁管理程序 (CLM) 为全局共享可访问和可高速缓存的对象提供了分布式锁定服务。CLM 代表锁请求来创建锁和将锁排队。它仲裁共享磁盘集群环境中从任一实例向任一全局对象发出的访问权限锁请求。

锁可以具有任务、事务或实例所有权。本地锁管理程序所管理的锁具有任务或事务所有权。CLM 所管理的锁具有实例所有权；它们在该实例上的所有进程或事务之间共享，并保留到另一个实例请求冲突锁模式为止。

## 死锁数

在 Adaptive Server 的非集群版本中，为了获得所有页锁定，如果服务器无法获取页锁，它会重试尝试而不是等待锁（这会导致死锁）。`deadlock retries` 配置参数确定 Adaptive Server 尝试的重试次数。

Cluster Edition 包含一个物理锁，这可以确保节点间的页一致性。但是死锁检测机制中不包含物理锁，因此服务器无法检测到死锁。

Cluster Edition 重试失败的尝试以获取锁。但是，当服务器超出 `deadlock retries` 值时，它或者失败，或者回滚查询，而不是等待锁和去冒遇到无法检测到的死锁的风险。

## 保留锁

Cluster Edition 还使用保留锁，这些锁是全集群范围锁，它们授予给集群中具有在实例上所有进程之间共享的所有权的实例。保留锁的所有权保留到另一个实例声明冲突锁模式为止，或保留到另一个实例声明与锁关联的资源为止。保留锁的使用减少了获取和释放锁的需要，因此减少了实例内部的消息传递。

Cluster Edition 使用一些不同的保留锁。

所有锁都是保留锁。

## 集群锁请求和任务请求状态

在 Cluster Edition 中，任务锁请求触发集群锁请求，但是一旦发出集群锁请求，集群锁请求将与任务请求状态无关。

例如，假设在集群级别上授予锁之前，请求锁的任务已经停止。在锁超时或任务注销时会发生这种情况。在这种情况下，对于拥有集群锁的实例而言，`sp_lock` 的输出会一直显示 “-blk”（这表示阻塞进程，即使没有正在等待锁的任务），直到拥有集群锁的实例将其释放并降级为止。

## 内存

Cluster Edition 需要的内存量多于独立 SMP Adaptive Server。附加内存支持节点间消息传递和集群锁。

- 您可以使用 `CIPC regular message pool size` 配置参数为节点间消息传递配置内存。缺省值为 8MB。
- 为了支持分布式高速缓存一致性，当实例启动时，Cluster Edition 自动为缓冲区高速缓存和系统描述符配置锁。

数据高速缓存中的每个缓冲区都自动配置有物理锁。对于具有 2KB 个页的数据库，每个物理锁的开销大约为 24.6%；对于具有 4KB 个页的数据库，每个物理锁的开销大约为 12.3%。因此，对于 100M 数据高速缓存，物理锁的附加内存开销分别为 24MB（对于 2KB 个页）和 12MB（对于 4KB 个页）。

系统描述符的平均开销大约为每个打开对象 1.5KB。为了获得最佳性能，`lock hashtable size` 参数自动调优为每个桶 8 个锁。对于大的高速缓存配置，`lock hashtable size` 可以自动调优为几兆字节。

## 阈值

当段上的可用空间低于 Adaptive Server 维护的阈值时，将执行用户创建的存储过程。

所有数据库在日志段上都有一个**最后机会阈值**，这是备份和截断事务日志所需的可用日志页的估计数量。最后机会阈值 (LCT) 是定义的可用空间量；日志段上可用页的指定数量。

LCT 还定义当空间量低于指定的阈值时要采取的操作。此阈值监控数据库中的可用空间，防止事务日志用尽空间。

在将阈值添加到段时，必须指定用于监控此阈值的存储过程。缺省情况下，Adaptive Server 将 `sp_thresholdaction` 用于最后机会阈值。

对于用户而言，集群处理中的阈值维护与非集群 Adaptive Server 中的阈值维护没有任何区别。但是，`dbcc` 命令存在一些变化，如下所述。

### ***dbcc thresholds*** 输出

`dbcc thresholds` 输出段结构。由于段结构自身发生更改，Clustered Edition 输出有所不同。例如，段的 `sg_below` 和 `sg_above` 阈值指针是用于指定 `dbt_thresholds` 数组中两个阈值的位置的数字。

### ***dbcc dbtable*** 输出

`dbcc dbtable` 输出 `dbt_thrmgr_info`（其中包含集群阈值管理数据），并输出表中的段结构。

### **带有 *remap* 选项的 *dbcc dbrepair***

带有 `remap` 的 `dbcc dbrepair`：

- 在 `dbtable` 中重建磁盘映射
- 重新计算段可用页计数
- 根据段的阈值级别和可用页总计数设置每个段中的阈值

可以使用带有 `remap` 的 `dbcc dbrepair` 来指定段号或段名称，但是您一次只能指定一个要重新映射的段。

如果您未指定要重新映射的段，Adaptive Server 将重新映射数据库中的所有段。可以在 `dbcc dbrepair` 结尾使用可选参数来指定要重新映射的段。

`dbcc dbrepair` 包含 `fixalloc` 选项；要同时使用 `remap` 和 `fixalloc`，请在 `fixalloc` 之前添加 `remap`。Adaptive Server 15.1.1 版允许您在此命令结尾指定段号或段名称。下面的示例为 `pubs2` 数据库指定日志段：

```
dbcc dbrepair(pubs2,"remap","fixalloc",-1,"logsegment")
```

要在不使用 `fixalloc` 的情况下重新映射 `pubs2` 数据库中的日志段，请输入：

```
dbcc dbrepair(pubs2,"remap",NULL,-1,"logsegment")
```

要重新映射 `pubs2` 数据库中的所有段，请使用：

```
dbcc dbrepair(pubs2,"remap")
```

带有 `remap` 选项的 `dbcc dbrepair` 由 `sp_addsegment`、`sp_dropsegment`、`sp_modifysegment`、`sp_extendsegment`、`sp_logdevice` 和 `sp_placeobject` 使用。有关这些存储过程和 `dbcc dbrepair` 的详细信息，请参见最新版本的《Adaptive Server Enterprise 参考手册：命令》。

## 带有 *newthreshold* 选项的 *dbcc dbrepair*

带有 `newthreshold` 的 `dbcc dbrepair`：

- 将数据库中的阈值从 `systhresholds` 装载到 `dbtable->dbt_thresholds`。
- 根据段的阈值级别和段可用页计数，设置每个段中的阈值指针。
- 指定段名称或段号，将 `systhresholds` 中的所有阈值读入到 `dbt_thresholds` 中。只有 `dbt_thresholds` 段的阈值是根据属于 `dbt_thresholds` 段和段可用页计数而设置的。

带有 `newthreshold` 的 `dbcc dbrepair` 由 `sp_addthreshold`、`sp_droptreshold` 和 `sp_modifythreshold` 使用。有关这些存储过程的详细信息和 `dbcc dbrepair` 的完整文档，请参见最新版本的《Adaptive Server Enterprise 参考手册：命令》。

## 集群进程间通信

集群进程间通信 (CIPC) 是集群体系结构中的重要子系统。大多数集群子系统使用 CIPC 与集群中的其它实例进行通信。

---

**注释** 若要获得可接受的 Cluster Edition 性能，必须使用高速 CIPC 网络。

---

在集群中的实例上运行的任务使用 CIPC 与集群中其它实例上的其它任务进行通信。集群输入文件中指定的 CIPC 信息和仲裁设备上存储的 CIPC 信息用于连接集群中的所有实例。在输入文件中，CIPC 支持主互连和可选辅助互连。辅助互连（如果提供）用于缓冲区 — 高速缓存传输，从而减少了主互连的负载。

## 分布式检查点

检查点同时控制非集群 Adaptive Server 和 Cluster Edition 系统中的恢复，但是它们为其提供事务信息的进程是不同的。在 Cluster Edition 中，找到的最早活动事务是整个集群中最早的活动事务，脏缓冲区在所有集群实例上进行刷新。

在这两种类型的处理中，Adaptive Server 在记录检查点记录时找到最早的活动事务信息，将其用作截止点，并将检查点记录写入日志。然后，脏缓冲区将刷新到当前刷新序列号。只有在刷新所有脏缓冲区后，此检查点记录才注册为恢复检查点。

## 仲裁设备心跳

集群中的每个实例定期“检入”仲裁设备，创建仲裁设备心跳，以使实例能够监控集群的状态。实例使用设备心跳执行下列操作：

- 确定仲裁设备是否处于可访问状态。如果实例无法将心跳写入仲裁设备，则实例可能已丢失其存储区域网络 (SAN) 链接，或者它被阻止在集群设备之外。
- 在仲裁磁盘启动时从仲裁磁盘读取心跳值。如果实例在配置的时间段后未检测到心跳中有任何更改，则实例确定集群未运行。

实例不使用仲裁心跳来检测实例故障。

## 配置仲裁设备心跳

您可以分别使用 `quorum heartbeat interval` 和 `quorum heartbeat retries` 配置参数，配置心跳的发生频率和实例在假设仲裁设备未运行之前尝试检测仲裁心跳的次数。

`quorum heartbeat interval`

`quorum heartbeat interval` 配置参数指定仲裁心跳之间的秒数。缺省值为 5，所以每个实例每 5 秒写入一次仲裁心跳。最小值为 1 秒，最大值为 60 秒。

大多数节点不需要调优此配置参数。将 `quorum heartbeat interval` 设置为较低的数会增加心跳开销，但是可加快检测丢失的磁盘链接的速度，因此当实例被阻止或丢失其 SAN 链接时实例可以更快地终止。将 `quorum heartbeat interval` 设置为较高的值可减少心跳开销，但会延迟对丢失的磁盘链接的检测。心跳导致的开销量取决于磁盘子系统的性能。

`quorum heartbeat retries`

`quorum heartbeat retries` 配置参数指定实例在确定仲裁设备不再运行并退出之前尝试检测仲裁心跳的次数。缺省值为 2（在第三个连续仲裁心跳由于前两个仲裁心跳失败而失败后，实例终止）。最小值为 0，它表示实例应当在第一个仲裁心跳失败时终止；最大值为 32,768。

如果将此值调优为较小的数，将使实例在丢失对仲裁设备的访问时更快地进行故障切换，从而改善了应用程序恢复时间。将此值调优为较大的数会使应用程序恢复降级，从而减少了瞬时磁盘访问问题导致实例故障的机会。

## 使用 InfiniBand

Adaptive Server 15.0.1 及更高版本支持 InfiniBand (iPvIB) 在集群中的节点间进行内部通信。

---

**注释** HPIA 系统上的 Cluster Edition 不支持 InfiniBand。

---

使用 InfiniBand:

- 配置主机通道适配器 (HCA)
- 使用 InfiniBand 软件堆栈
- 对于 Linux，Cluster Edition 需要 OpenFabrics 提供的 OFED 1.2
- Solaris 10 附带了 InfiniBand 软件

**注释** Solaris 10 包含 InfiniBand。有关安装和配置 InfiniBand 的信息，请参见操作系统文档。

---

## 设置缓冲区空间

Cluster Edition 必须具有足够的缓冲区空间，以确保足够的性能，尤其是更快的互连速度。Linux 和 Solaris 系统上缓冲区空间的缺省值太小，无法用于以太网和 InfiniBand。您必须修改它们以确保足够的网络性能。

### 配置缓冲区空间

Linux

使用类似如下的命令将系统上的缓冲区空间设置为适当的大小：

```
/sbin/sysctl -w net.core.rmem_max=value  
/sbin/sysctl -w net.core.wmem_max=value
```

在大多数平台上，`rmem_max` 的缺省值设置为大约 128KB，该值太小，无法用于 Cluster Edition。将 `rmem_max` 增大到平台允许的最大值（1 兆字节是好的起始值）：

```
/sbin/sysctl -w net.core.rmem_max=1048576
```

Solaris

使用类似如下的命令将系统上的缓冲区空间设置为适当的大小：

```
ndd -set /dev/udp udp_max_buf value
```

## 在集群中配置 InfiniBand

配置主机通道适配器 (HCA) 时，`/etc/hosts` 文件包含相应的主机名和 IP 地址。当 Adaptive Server 插件或 `sybcluster` 在集群配置过程中要求输入主机名或 IP 地址时，请使用此主机名或 IP 地址。如果您手动集群时，需要添加 IP 地址。

## 专用安装模式

当您安装和创建集群时，需要选择是在共享安装模式下还是在专用安装模式下配置集群：

- 共享安装 — 支持使用网络文件系统 (NFS) 或集群文件系统创建的共享文件系统。使用共享安装创建的集群支持单一 `$SYBASE` 安装目录、Adaptive Server 主目录和服务器配置文件。
- 专用安装 — 每个实例需要单独的 `$SYBASE` 安装目录、Adaptive Server 主目录和服务器配置文件。服务器配置文件之间的奇偶校验由仲裁设备上的主配置文件维护。

---

**注释** Sybase 建议在使用专用安装时将 LDAP 作为目录服务。

---

如果安装模式是专用安装，则集群输入文件自动包含以下条目：

```
installation mode = private
```

如果安装模式是共享安装，则集群输入文件自动包含以下条目：

```
installation mode = shared
```

有关在专用或共享安装模式下安装和配置集群的说明，请参见所用平台的安装指南。

## 维护服务器配置文件

配置文件版本号跟踪服务器配置文件的最新副本。此信息存储在仲裁设备上 and 每个服务器配置文件中。每次将新配置文件存储到仲裁设备上时，都会递增版本号，Adaptive Server 将新服务器配置文件和版本号转储到每个实例。

使用 `qrmutil` 可查看任意配置文件的当前版本号：

```
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-version
```

有关完整语法，请参见《实用程序指南》。

## 更改运行时配置选项

当实例运行时，您可以使用 `sp_configure` 更新服务器配置选项。如果选项更改，Adaptive Server 将自动更新主配置文件，递增版本号，更新 `sysconfigures` 表，并指示所有实例转储到配置文件的新副本。

将 `sp_configure` 和 Cluster Edition 一起使用时：

- 配置一个仅应用于您当前连接到的实例的、特定于实例的值。
- “完全集群范围”的配置选项（即，在所有实例上都相同的一小组配置选项）无法特定于实例。
- 重新配置集群范围的值不会覆盖特定于实例的配置。

如果在配置过程中实例未运行，将不更新其配置文件和版本号。您可以让 Adaptive Server 在实例启动时检测这一点（请参见第 157 页的“如果在启动时版本号不匹配”），也可以检查版本号并从仲裁设备转储新副本：

```
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-version
```

```
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-extract=mycluster.cfg
```

## 手动更改配置文件

在配置实例未运行时，您可以手动更改配置文件：

- 1 提取配置文件。例如：

```
qrmutil --quorum/dev/raw/raw101 --ase-config-extract=mycluster.cfg
```

- 2 编辑配置文件：

```
vi mycluster.cfg
```

- 3 重新启动实例。

当您重新启动实例时，Adaptive Server 自动递增版本号，使用新版本号更新 `sysconfigures` 表，将新配置文件写入本地目录和仲裁设备，指示所有实例从仲裁设备转储配置文件的新副本。

---

**注释** 当配置实例未运行但是协调器运行时，您可以更改特定于实例的参数，但不能更改全集群范围的参数。

---

## 如果在启动时版本号不匹配

Adaptive Server 在每个实例启动时检查文件版本号。服务器配置文件中的版本号必须与仲裁设备上存储的版本号匹配。如果文件版本号不匹配，Adaptive Server 将显示一条错误消息，将配置文件的最新副本从仲裁磁盘提取到本地文件系统，并中止实例启动。

此时，您可以决定是使用最新配置文件重新启动实例，还是手动编辑该文件然后重新启动实例。

- 如果版本号现在匹配，则配置文件中没有任何更改，您可以成功重新启动实例。
- 如果您编辑了配置文件，版本号匹配，但是配置文件的内容现在已更改。重新启动实例：Adaptive Server 自动递增版本号，使用新版本号更新 `sysconfigures` 表，将新配置文件写入本地目录和仲裁设备，指示所有实例从仲裁设备转储配置文件的新副本。

## 建议配置

Sybase 建议：

- 不要手动更改服务器配置文件中的“`config file version`”参数。这应当是服务器生成的编号。
- 仅在启动实例上更新配置文件参数。如果在另一个实例上更新配置文件参数，则当该实例启动时，它将生成新的配置文件，该文件将删除其它实例中的所有用户更改。

## 在集群环境中使用 Java

Adaptive Server 在集群环境中支持第三方 Java 虚拟机 (JVM)，例如 Sun Java 2 Platform Standard Edition (J2SE)。有关如何在 Adaptive Server CE 上安装 Sybase Java 组件的说明，请参见所用平台的安装指南。有关如何在 Adaptive Server 上使用 Java 的完整说明，请参见《Adaptive Server Enterprise 中的 Java》。

通常，在集群环境和非集群环境中使用 Java 的方式是相同的。存在以下区别：

- 在 Cluster Edition 中，执行 Java 的连接不支持连接迁移。

如果连接失败，则执行 Java 的连接根据配置将故障切换到另一个实例。在故障切换后，将再次执行现有 SQL 批处理和 Java。

- 根据 JVM 是共享的还是专用的，Adaptive Server 处理静态数据的方式有所不同。
  - 在 Cluster Edition 中，每个节点有其自己的 JVM — 静态数据在节点间不可共享。如果节点关闭，JVM 也关闭，JVM 上的所有工作将丢失。您必须重新应用现存的 SQL 批处理。
  - Adaptive Server 15.0.3 的非集群版本在所有引擎和连接中共享单一 JVM — 对于从通用类装载程序装载的类，可共享静态数据。
  - 在 Adaptive Server 15.0.2 及更早版本中，每个客户端连接有其自己的专用 Sybase JVM — 客户端间不支持静态数据。

## 向存档数据库添加空间

通常，在集群和非集群 Adaptive Server 中，处理存档数据库的访问权的方式是相同的。在任一情况下，当存档数据库用尽空间时，您可以使用 `alter database` 目录向存档数据库添加空间。

在集群 Adaptive Server 中，从更新存档数据库的同一节点运行 `alter database`。如果从另一个节点运行 `alter database`，Adaptive Server 将显示一条错误消息，其中包含实际更新存档数据库的节点的编号。

此功能对 Adaptive Server 15.5 版和更高版本可用。

## 共享磁盘集群中的分布式事务

在 15.5 版及更高版本中，Adaptive Server 在其集群体系结构中支持分布式事务管理 (DTM)。Cluster Edition 上的 DTM:

- 在充当资源管理器 (RM) 时完全符合 X/Open XA 协议，不需要附加服务，例如 XA-Server。
- 确保通过远程过程调用 (RPC) 和组件集成服务 (CIS) 更新 Adaptive Server 数据的事务的提交或回退保持一致。
- 可以是其它 Adaptive Server 安装使用 Adaptive Server 事务协调 (ASTC) 机制协调的分布式事务的一部分。

- 可以使用 ASTC 机制在多个 Adaptive Server 安装之间协调分布式事务。

---

**注释** Cluster Edition 不支持 Microsoft 分布式事务协调器 (MSDTC) 专有协议。

---

## 在共享磁盘集群中使用 DTM

通常，分布式事务的用户界面在 Adaptive Server 集群环境和非集群环境中是相同的，但是，必须注意本节中所述的一些特定于集群的问题。在非集群 Adaptive Server 上使用 DTM 的应用程序可以在集群 Adaptive Server 上使用相同的应用程序。请参见 Using Adaptive Server Distributed Transaction Management Features（《使用 Adaptive Server 分布式事务管理功能》）。

## 集群作为资源管理器

虽然集群包含多个 Adaptive Server，但是它必须为用户提供单一系统视图。因此，每个应用程序必须将整个集群视为单一 RM；不应当将单个实例视为单独的 RM。

例如，当在集群中进行远程过程调用时，使用集群名称作为 RM：

```
exec cluster_name.dbname.owner.procedure_name
```

确保在集群中执行 XA 事务的事务管理器 (TM) 使用集群名称作为 XA 配置文件中的逻辑资源管理器 (LRM) 名称。请不要使用实例名称。有关 XA 配置的详细信息，请参见 XA Interface Integration Guide for CICC, Encina, and TUXEDO（《适用于 CICC、Encina 和 TUXEDO 的 XA 接口集成指南》）。

同样，当创建 XADatasource 对象时，使用 jConnect 运行 XA 事务的应用程序需要使用集群名称。例如：

```
XADatasource xads = (XADatasource)  
    ctx.lookup("server_name=cluster_name");
```

虽然集群命名为 RM，但是每个事务在集群中的单一实例（所有者实例）上运行，且仅在该实例上维护事务的状态。正在运行的事务不能在集群中的实例之间迁移。所有 Transact-SQL 语句和事务提交处理仅在拥有该事务的实例上执行。

## 在非所有者实例上处理请求

某些分布式事务机制（例如 X/Open-XA 协议）不面向连接，它们允许应用程序在不同的连接上发出命令。因此，可以在非所有者实例上发出在挂起的事务分支上继续工作的请求。同样，可以在与非所有者实例的连接上发出提交处理的请求。在集群 Adaptive Server 中，在非所有者实例上所发出的这种请求需要特殊处理。

当在非所有者实例上发出请求时，集群 Adaptive Server 确定事务的所有者实例，然后在连接库的帮助下将连接迁移到所有者实例。连接迁移对应用程序是透明的。在成功连接迁移之后，在所有者实例上处理请求。

有关连接迁移的详细信息，请参见第 2 章“客户端应用程序和客户端/服务器交互”。

ASTC 事务不需要连接迁移。ASTC 机制需要在同一连接上发出所有 Transact-SQL 语句和事务提交处理。

## 处理实例故障

X/Open-XA 和 ASTC 协议不允许 RM 在未接到 TM 的指令的情况下单方面提交或中止就绪事务。RM 必须将就绪事务保留到 TM 将事务的结束状态通知给它为止。但是，当事务分支在集群中运行时，集群中的实例可能发生故障。集群协调器在故障切换恢复过程中重新实例化出现故障的实例的就绪事务。一旦事务在集群协调器上重新实例化，TM 就可以提交或回滚事务。

---

**注释** 用于提交或回滚重新实例化的事务的命令可以作用于除当前所有者实例之外的实例。在此情况下：

- 对于 XA 事务 — 服务器将连接迁移到所有者实例，由所有者实例来处理请求。
- 对于 ASTC 事务 — 通过将请求转发到所有者实例以进行处理，然后将答复发送到 TM，将接收 commit 或 rollback 命令的实例用作代理。

---

在故障切换恢复过程中，仅重新实例化出现故障的实例上的就绪事务分支。如果出现故障的实例上运行的事务分支尚未就绪，则集群 Adaptive Server 在故障切换恢复过程中回滚其工作。

## 使用 ASTC 进行事务协调

集群 Adaptive Server 像非集群 Adaptive Server 一样，使用 ASTC 机制来协调在多个 Adaptive Server 安装中存在的分布式事务。事务协调服务对于执行分布式事务的客户端是透明的。当本地客户端事务执行 RPC 或通过 CIS 更新数据时，协调服务将为远程工作创建新的事务名称，并将该事物传播给下级远程服务器。当本地客户端提交或回退本地事务时，Adaptive Server 会将该请求与每个下级服务器进行协调，以确保同时提交或回退远程事务。

集群 Adaptive Server 执行特殊 ASTC\_HANDLER 任务以处理故障，例如连接终止、远程服务器故障或本地服务器故障。在从故障条件恢复后，ASTC\_HANDLER 将事务的结束状态传播到远程参与者。ASTC\_HANDLER 任务在集群中的每个实例上执行。在出现故障时，实例上的 ASTC\_HANDLER 任务仅将该实例上启动的分布式事务的结束状态传播到远程参与者。集群协调器上的 ASTC\_HANDLER 任务具有附加工作，即处理上次启动集群以来事务的远程参与者，或处理当某实例离开集群时该实例上的远程参与者。

## 连接迁移的影响

对于 XA 事务，Adaptive Server 集群使用连接迁移来处理非所有者实例上的请求。

## 连接迁移和性能

连接迁移可能对性能产生负面影响，因为它需要：

- 终止非所有者实例上的现有连接
- 创建与运行事务的实例的新连接
- 保存有关现有连接的上下文信息，然后在连接迁移之后将此信息恢复到所有者实例上

减少连接迁移对性能的负面影响：

- 使用逻辑集群和负载管理器功能将 XA 事务绑定到集群中的特定实例，以便不需要连接迁移。请参见第 6 章“管理负载”。
- 修改应用程序以避免不必要的断开连接和重新连接。

## 当连接无法迁移时

有时，应用程序会在处于不允许迁移的状态下的连接上发出请求。请参见第 25 页的“迁移条件”。

例如，如果应用程序连接到实例，则连接无法迁移，因此发出请求以执行有关事务的操作之前，请使用连接创建一个临时表。如果此类连接是连接到非所有者实例，则 Adaptive Server 无法处理请求，应用程序将收到特定于协议的错误代码。如果是 X/Open-XA 协议，TM 会将 XAER\_RMERR 接收为返回状态，这表明集群尚未处理 XA 命令。事务尚未回滚；它仍在所有者实例上运行。

通过在另一个连接上重新发出命令可解决此错误。XAER\_RMERR 属于一般错误，可在许多错误情况下返回。您可以使用 `xact_connmigrate_check` 和 `xact_owner_instance` 函数确定 XAER\_RMERR 错误是否是无法迁移的连接所导致。请参见第 162 页的“配置和系统问题”。

避免由于不可迁移的连接而导致的 XAER\_RMERR 错误：

- 不要将用于 XA 应用程序的连接用于一般用途，因为连接会置于不允许连接迁移的状态。
- 使用跟踪标志 3960，以避免在与非所有者实例的不可迁移连接上发出 `commit` 和 `rollback` 命令时，出现 XAER\_RMERR 错误。当您使用跟踪标志 3960 时，通过将请求从 TM 转发到所有者实例，然后将答复发送到 TM，非所有者实例充当 TM 与所有者实例之间的代理。

## 配置和系统问题

有关配置和管理 DTM 的一般信息，请参见 Using Adaptive Server Distributed Transaction Management Features（《使用 Adaptive Server 分布式事务管理功能》）。本节描述集群 Adaptive Server 上特定于 DTM 的配置和管理问题。

- 当配置 DTM 的集群时，将集群名称指定为 RM。确保应用程序在 ASTC RPC 和 TM 配置中使用集群名称。
- DTM 相关配置参数（`enable DTM`、`enable xact coordination` 和 `strict dtm enforcement`）是集群 Adaptive Server 中的集群静态参数。如果您更改了这些参数的值，则必须重新启动集群才能使它们生效。
- `syscoordinations` 和 `systransactions` 系统表中的 `instanceid` 列标识了事务的所有者实例。值为零 (0) 的 `instanceid` 表示所有者实例已关闭和出现故障，集群协调器现在是事务的所有者实例。

- 下列两个函数在集群 Adaptive Server 上支持 DTM:
  - `xact_owner_instance(XID)` — 返回运行外部事务的实例。
  - `xact_connmigrate_check(XID)` — 确定当前连接是否可以处理外部事务。`XID` 参数是可选的。

如果指定了外部事务名称 (XID):

- 在以下条件下返回 1:
  - 连接是连接到所有者实例, 或者
  - 连接是连接到非所有者实例, 连接可以迁移。
- 否则, 返回 0。

如果未指定 XID:

- 如果连接可以迁移, 则返回 1。
- 否则, 返回 0。

- 如果集群实例正在运行分布式事务分支, 不要使用正常 `shutdown instance_name` 命令关闭实例。请转而使用:

```
shutdown instance_name with nowait
```

正常 `shutdown instance_name` 命令可以回滚在实例上运行的常规用户事务, 但是它不能回滚就绪状态下的分布式事务分支。这意味着, 只有当实例上没有活动分布式事务时, 才可以使用正常实例 `shutdown`。

当使用 `nowait` 选项关闭实例时, 它触发了故障切换恢复, 该恢复在集群协调器上重新实例化实例的就绪分布式事务分支。另请参见第 160 页的“处理实例故障”。

---

**注释** 虽然不能使用正常 `shutdown instance_name` 命令, 但是您可以使用正常 `shutdown cluster_name` 命令, 即使集群中的实例上正在运行分布式事务分支也是如此。

---

## 支持 *mount* 和 *unmount* 命令

对于 Adaptive Server 15.5 及更高版本，可以在共享磁盘集群中使用 `mount database` 和 `unmount database` 命令。如果当这些命令中的一个命令正在执行时实例出现故障，则命令可能中止。当实例故障切换恢复完成时，重新发出 `mount database` 或 `unmount database`。

## 使用 `sp_showplan`

不能跨共享磁盘集群的多个节点使用 `sp_showplan`。它特定于与节点的单个连接。

## 将 Cluster Edition 与 Veritas Cluster Server 结合使用

本章介绍如何配置以及将 Cluster Edition 与 Veritas Storage Foundation for Sybase CE (SF for Sybase CE) 结合使用。

主题	页码
支持的平台、要求和限制	168
在 VCS 上安装和配置 Cluster Edition	169
管理 VCS 控制下的集群	173
成员资格模式	175
了解故障情况	177
VCS 故障排除	178

Cluster Edition 15.0.3 版及更高版本与 Veritas SF for Sybase CE 的集成使 Cluster Edition 能够利用 Veritas Storage Foundation 存储管理技术、Cluster Server (VCS) 应用程序和集群管理功能。

**注释** 通过将 Cluster Edition 15.0.3 版与 SF for Sybase CE 集成，可提供集群可用性和数据完整性。Veritas Storage Foundation 的其它版本未包含必要的集成组件，因此不应使用这些版本。

带 SF for Sybase CE 的 Cluster Edition 包括：

- Storage Foundation 集群文件系统 — 一种可用于 Cluster Edition 安装文件（位于 `$SYBASE` 中的文件和目录）、数据库设备、仲裁设备及其它应用程序文件的通用集群文件系统
- 集群卷管理器 — 创建集群节点共享的逻辑卷
- 动态多路径 — 提高存储可用性和性能
- 基于服务组的应用程序管理 — 提供监控和故障切换功能，并允许您在应用程序之间创建依赖关系，以便在应用程序所需的组件（例如磁盘卷）出现故障时能够对整个应用程序进行故障切换

- 
- 对多个集群、应用程序和数据库服务器的集成管理（使用 VCS 代理和管理主控台）
  - 支持硬件复制技术和块级复制（使用 Veritas Volume Replicator）

Cluster Edition 和 SF for Sybase CE 包括集群操作的两个关键元素：成员资格管理和 I/O 防护：

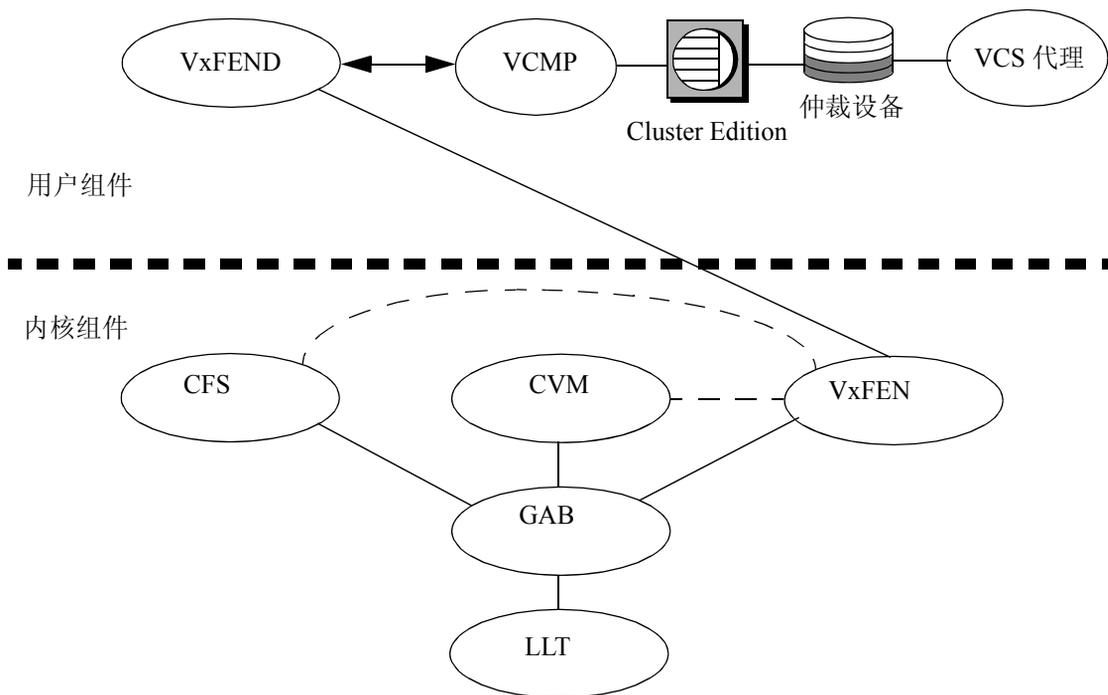
- 成员资格管理 — Cluster Edition 和 SF for Sybase CE 维护自己的成员资格管理器。成员资格管理器负责：
  - 协调登录和注销集群
  - 检测故障
  - 确定在通信丢失时集群的哪些成员保持为活动状态（称为仲裁）
  - 维护一致的集群视图

Veritas Cluster Membership 插件 (VCMP) 允许 Cluster Edition 成员资格服务与基础 Veritas 成员资格管理器同步，从而可避免两个成员资格管理器不协调、采用不同方式仲裁故障并导致集群关闭的情况。使用 VCMP 可确保 Cluster Edition 采用有利于依据 Veritas 成员资格视图内的节点运行的实例的方式进行仲裁。

- I/O 防护 — 之所以采用此名称，原因是成员资格管理器会为数据存储建立防护，并仅允许行为适当的实例或节点执行写入。使用 I/O 防护可防止不合作的集群成员造成数据损坏。Cluster Edition 和 SF for Sybase CE 协调 I/O 防护以便：
  - SF for Sybase CE 管理和执行所有防护，并且
  - Cluster Edition 能够与由于 Cluster Edition 成员资格更改而必须进行防护操作的 SF for Sybase CE 通信。

图 12-1 描述组成 Cluster Edition 和 SF for Sybase CE 的组件:

图 12-1: 集群 Veritas 系统中的组件



Sybase 提供的组件包括:

- Cluster Edition — 依据节点运行的关系数据库服务器
- 仲裁设备 — 包括集群的配置信息并由所有集群成员共享
- Veritas Cluster Membership 插件 (VCMP) — 从 VxFEND 接收成员资格更改消息, 并传递给 Cluster Edition 的成员资格服务, 后者将阻止成员资格更改, 直至 VCMP 允许它继续为止

Veritas 提供的组件包括:

- LLT (低延迟传输) — 允许 VCS 跨集群通信, 同时检测心跳和故障。
- GAB (全局原子广播) — 协调 VCS 成员资格。LLT 为 GAB 提供有关故障事件 (例如, 无心跳) 的信息。
- CVM (集群卷管理器) — 从 GAB 接收成员资格信息, 并与 VxFEN 协调。

- CFS（集群文件系统）— 一种可由集群中的多个节点同时装入和访问的文件系统。CFS 使用分布式锁管理器在节点之间维持一致性。CVM 以及存储区域网络 (SAN) 提供基础存储。
- VxFEN（内核端 I/O 防护控制）— VxFEN 从 GAB 接收成员资格信息，并在适当时执行防护。
- VxFEND（用户空间 I/O 防护守护程序）— 在用户空间中运行（而不是在内核中运行）的与 VxFEN 通信的守护程序。在成员资格更改期间，VxFEN 会向 VxFEND 发送指示成员资格发生变化的消息。VxFEND 通过 VCMP 就成员资格更改通知 Cluster Edition。
- VCS 代理 — Cluster Edition 的 VCS 监控代理。如有必要，VCS 代理可在 Cluster Edition 失败的情况下触发主机应急。VCS 代理还可供 VCS 用来启动和停止 Cluster Edition。

## 支持的平台、要求和限制

下列平台支持带 SF for Sybase CE 的 Cluster Edition:

- SPARC 上的 Solaris 10 (Sun Sparc 64)
- x86\_64 上的 Redhat Enterprise Linux 5 (Linux AMD 64)

有关所需版本级别、软件包和修补程序的信息，请参见所用平台的 Cluster Edition 发行公告和安装指南。有关 SF for Sybase CE 要求的完整列表，请参见 Veritas Storage Foundation for Sybase CE Release Bulletin（《Veritas Storage Foundation for Sybase CE 发行公告》）和 Installation and Configuration Guide（《安装和配置指南》），二者均由 Veritas 提供。

Cluster Edition 在 native 或 vcs 模式下运行。在 native 模式下，Cluster Edition 不会与集群成员资格管理器协同工作；在 vcs 模式下，集群成员资格管理器将与 Veritas 集群件协同工作。请参见第 175 页的“成员资格模式”。

若要在 vcs 成员资格模式下使用 Cluster Edition，必须执行以下操作：

- 在主机节点上安装和配置 SF for Sybase CE 5.0 版。
- 确保所有 Cluster Edition 存储（包括主设备、用户数据库设备和仲裁设备）都来自 SF for Sybase CE 托管存储。这既可以是 CVM 卷也可以是 CFS 文件。
- 将同一物理网络的 Cluster Edition 专用网络配置为 Veritas LLT 链接。请参见第 169 页的“在 VCS 上安装和配置 Cluster Edition”。

在 vcs 成员资格模式下使用 Cluster Edition 存在以下限制：

- 不能在给定的硬件节点上运行多个实例，即使这些实例属于不同的 Adaptive Server 集群也是如此。
- 一个集群中不能有四个以上的实例或节点。
- 实例故障可能会导致主机节点的应急重启。请记住，如果您计划在同一节点上运行一些不相关的关键应用程序（包括其它非集群 Adaptive Server），则该节点重新启动后这些应用程序将会被关闭。

## 在 VCS 上安装和配置 Cluster Edition

若要安装和配置 Cluster Edition 以在 Veritas 集群子系统上运行，必须执行 Veritas 文档和下列各节中介绍的步骤。

按照 Veritas SF Sybase CE Installation and Configuration Guide（《Veritas SF Sybase CE 安装和配置指南》）中的说明执行步骤 1 – 6：

- 1 安装和配置 Veritas 的 SF for Sybase CE。
- 2 在 Sybase 模式下配置 I/O 防护。
- 3 创建一个共享磁盘组、卷和装入点用于 Adaptive Server 二进制文件安装 (\$SYBASE)。
- 4 为 Adaptive Server 仲裁设备创建一个共享磁盘组、卷和装入点（如果使用 CFS）。
- 5 为 Adaptive Server 数据库设备创建一个共享磁盘组、卷和装入点（如果使用 CFS）。
- 6 确保“sybase”用户（或同等用户）具有合适的所有权以及对步骤 5 中的存储的访问权

按照本手册中第 170 页的“安装 Cluster Edition”、第 170 页的“为 Storage Foundation 集成创建新的 Adaptive Server 集群”和第 172 页的“将现有集群转换为使用 SF for Sybase CE”章节中的说明执行步骤 7 和 8：

- 7 安装 Adaptive Server Cluster Edition。
- 8 创建 Adaptive Server 集群或准备现有的集群以便使用。

按照 SF Sybase CE Installation and Configuration Guide（《SF Sybase CE 安装和配置指南》）中的说明执行步骤 9。

- 9 使用 SF for Sybase CE 安装程序为 Adaptive Server 集群创建一个 VCS 服务组。这会将集群置于 vcs 成员资格模式。

---

**注释** 一旦在 vcs 成员资格模式下配置了 Cluster Edition，请将集群作为 VCS 服务组的一部分启动。如果在 vcs 模式下启动了位于 VCS 服务组之外的 Cluster Edition，则 Cluster Edition 将挂起并关闭。

---

## 安装 Cluster Edition

执行所用平台的安装指南中的第 3 章 “Creating and Starting the Cluster”（创建和启动集群）中介绍的步骤，使用 SF for Sybase CE 创建集群。

---

**注释** 在创建 VCS 服务组之前，请安装 Cluster Edition。

---

Sybase 建议：

- 在共享安装而不是专用安装中创建 Cluster Edition
- 在 Storage Foundation 集群文件系统 (SFCFS) 的装入点上创建版本目录 (\$SYBASE)。在安装 Cluster Edition 之前先安装 SF for Sybase CE。请参见 Veritas SF Sybase CE Installation Guide (《Veritas SF Sybase CE 安装指南》)。

## 为 Storage Foundation 集成创建新的 Adaptive Server 集群

sybcluster 和 Adaptive Server 插件都会自动检测 SF for Sybase CE 是否已安装在主机节点上，并引导您选择合适的存储和网络互连。在创建集群时需满足以下条件：

- 所有存储都必须位于 SF for Sybase CE 托管设备上，使用 CVM 控制卷或 SFCFS 文件系统，同时具有以下限制：
  - 如果您直接访问 CVM 控制卷，则必须使用 `/dev/vx/rdisk`（而不是 `/dev/vx/dsk/`）以访问原始（字符）设备的方式访问它们。
  - 您可以将专门针对本地临时数据库添加的专用设备放到本地非 SF for Sybase CE 存储设备上。

- 不要使用非 Sybase CE 版的 Veritas Storage Foundation 提供的存储。除非您针对 vcs 成员资格模式配置了 Cluster Edition，否则不要使用 SF for Sybase CE 提供的存储。Adaptive Server 不能在 native 模式下执行受 VxFEN 控制的 I/O 防护，因为这可能会导致数据库损坏。
- 配置 VCMP 套接字路径。缺省情况下，VCMP 使用 `/tmp/vcmp_socket` 与 Veritas 通信。
  - a 将此路径添加到位于 `$SYBASEASE-15_0/install/RUN_instance name` 中的 `runserver` 文件：

```
export VCMP_SOCKET=new_vcmp_socket_path
```

例如：

```
export VCMP_SOCKET=/tmp/my_socket
```

- b 修改 `vxfend` 资源：

```
hares -modify vxfend Arguments "%-m sybase -k new_vcmp_socket_path"
```

例如：

```
hares -modify vxfend Arguments "%-m sybase -k /tmp/my_socket"
```

即

- Cluster Edition 主集群互连和辅助集群互连必须通过 Veritas LLT 托管网络运行。通过 `/etc/llttab` 文件确定 LLT 在使用哪些网络。在此示例中，LLT 通过 `eth0` 和 `eth1` 网络运行：

```
[admin@sdcc2 ~]$ cat /etc/llttab
set-node sdcc2
set-cluster 53
link eth0 eth-00:19:b9:b0:73:13 - ether - -
link eth1 eth-00:19:b9:b0:73:15 - ether - -
```

---

**注释** Cluster Edition 互连要求您为链接配置 IP 地址。由于 LLT 不需要 IP 地址，因此您可能需要单独配置互连。请参见操作系统文档。

---

使用 `ifconfig` 命令确定分配给给定网络的 IP 地址。可以使用此地址来配置主集群互连或辅助集群互连。在此示例中，接口 `eth0` 具有 IPv4 地址 10.22.104.141：

```
[admin@sdcc2 ~]$ /sbin/ifconfig -a eth0
eth0  Link encap:Ethernet  HWaddr 00:19:B9:B0:73:13
       inet addr:10.22.104.141  Bcast:10.22.104.255  Mask:255.255.255.0
       inet6 addr: fd77:55d:59d9:168:219:b9ff:feb0:7313/64 Scope:Global
       inet6 addr: fe80::219:b9ff:feb0:7313/64 Scope:Link
```

```
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:19710010 errors:0 dropped:0 overruns:0 frame:0
TX packets:11707065 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:7506123489 (6.9 GiB)  TX bytes:1839949672 (1.7 GiB)
Interrupt:169 Memory:f8000000-f8012100
```

创建集群之后，将该集群重置为 vcs 模式。

## 将现有集群转换为使用 SF for Sybase CE

如果要将 Cluster Edition 转换为与 SF for Sybase CE 一起使用，则 Storage Foundation 软件必须管理所有数据库存储，并且 Cluster Edition 互连必须通过 LLT 链接进行。

## 重新分配数据库存储

将现有数据库从非 Storage Foundation 存储转移为 Storage Foundation 存储的方式因节点不同而异，具体取决于所涉及数据库设备的配置、大小和数目。

## 重新配置互连

重新配置互连：

- 1 使用 `ifconfig` 命令确定 LLT 链接的合适 IP 地址。
- 2 使用 `shutdown cluster` 命令关闭集群。
- 3 使用 `qrmutil` 重新配置互连。使用相应的实例名称、主地址和辅助地址对集群中的每个实例运行下面的命令：

```
qrmutil --quorum_dev=path_to_quorum_device --instance=instance_name
--primary-address=address_of_primary_interconnect
--secondary-address=address_of_secondary_interconnect
```

此示例将 “ase1” 实例配置为使用主地址 192.168.0.1 和辅助地址 192.168.0.2：

```
qrmutil --quorum_dev=/sybase_cfs/quorum.dat --instance=ase1
--primary-address=192.168.0.1 --secondary-address=192.168.0.2
```

或者，如果不对每个实例执行步骤 3，则可以改为将当前配置提取到一个文件（如步骤 4 中所示），为每个实例编辑该文件，然后使用 `--load-config` 参数将该文件重新加载到仲裁设备上。

- 4 使用 `qrmutil --extract-config` 参数创建新的集群配置的备份。此示例将集群配置备份到 `quorum.bak` 文件：

```
qrmutil --quorum_dev=/sybase_cfs/quorum.dat  
--extract-config=/sybase_cfs/quorum.bak
```

- 5 重新启动 Cluster Edition。

## 管理 VCS 控制下的集群

必须通过将 Cluster Edition 成员资格模式更改为 `vcs`，将 Cluster Edition 置于 VCS 的控制之下，以此激活允许 Cluster Edition 和 SF for Sybase CE 一起工作的软件。

有关创建 VCS 服务组的说明，请参见 SF Sybase CE Installation and Configuration Guide（《SF Sybase CE 安装和配置指南》）中的“Configuring a Sybase ASE CE cluster under VCS control using the SF for Sybase CE installer”（使用 SF for Sybase CE 安装程序配置 VCS 控制下的 Sybase ASE CE 集群）。

## 启动和停止实例

尽管 Cluster Edition 可识别 `Transact SQL shutdown cluster`、`shutdown cluster with nowait`、`sybcluster shutdown` 命令以及 Adaptive Server 插件启动和停止选项，但在 `vcs` 成员资格模式下，Sybase 仍建议您使用 VCS 代理启动和停止实例。

请使用对应的 VCS 服务组启动和停止在 `vcs` 成员资格模式下运行的实例，该服务组使用 VCS 代理启动和停止实例。使用 VCS Cluster Manager（Java 主控台）或 `hagrp` 命令行实用程序控制 VCS 代理。

请参见 SF Sybase CE Administrator's Guide（《SF Sybase CE 管理员指南》）中的“Administering the Sybase Agent”（管理 Sybase 代理）。有关 VCS 代理和资源管理的详细信息，请参见 Veritas Cluster Server User's Guide（《Veritas Cluster Server 用户指南》）。

## 添加和删除实例

向集群中添加实例或从集群中删除实例涉及以下操作：

- 向 VCS 中添加主机节点或从 VCS 中删除主机节点（如果是删除实例，则为可选操作）
- 添加或删除实例

您可以任意顺序执行这些任务，但不能在这两个任务没有都完成之前启动实例。请参见第 14 章“使用 Adaptive Server 插件管理集群”和第 15 章“用 sybcluster 管理集群”。有关向 VCS 集群添加节点的信息，请参见 SF Sybase CE Administrator's Guide（《SF Sybase CE 管理员指南》）中的“Adding a node to SF Sybase CE clusters”（向 SF Sybase CE 集群添加节点）。

## 增大用户连接数

当增大用户连接数时，还必须增大文件描述符限制。但是，在 VCS 之上运行的 Veritas 不将文件描述符限制更改从一个用户会话传播到另一个用户会话。例如，如果在当前会话中更改文件描述符限制：

```
ulimit -n 8194
```

后续的 UNIX 会话都具有缺省的文件描述符数设置 (1024)。因此，虽然 Veritas VCS 管理集群并使实例联机，但它以用户 sybase 的身份登录到 Linux 操作系统（启动新的 UNIX 会话）并对文件描述符数使用缺省值。

若要确保所有 sybase 用户会话都有正确的文件描述符限制，请将以下行添加到 */etc/security/limits.conf*：

```
sybase hard nofile 8096  
sybase soft nofile 8096
```

## 更改字符集或排序顺序

若要更改字符集或排序顺序：

- 1 使用 VCS 命令关闭 Adaptive Server。
- 2 将成员资格模式更改为本地：

```
qrmutil -Q quorum_file --membership-mode="native"
```

手动或通过 sybcluster 重新启动集群中的单个实例。

- 3 若要更改字符集，请运行字符集。例如：

```
$SYBASE/ASE-15_0/bin/charset -Usa -P nocase.srt utf8
```

- 4 若要更改排序顺序，请重置缺省的 `sortorder id` 配置参数。例如：

```
isql -Usa -P  
sp_configure 'default sortorder id', 101, 'utf8'
```

- 5 从 `isql` 关闭集群

- 6 重新启动实例。Adaptive Server 重新配置字符集和排序顺序，并且关闭。

- 7 将成员资格模式改回 VCS：

```
qrmutil -Q quorum_file --membership-mode='vcs'
```

- 8 使用 VCS 命令重新启动集群。

## 成员资格模式

Cluster Edition 15.0.3 版及更高版本包含两个成员资格模式，即 `native` 和 `vcs`。成员资格模式指示 Cluster Edition 的内部成员资格管理器 (CMS) 是自主运行（在 `native` 模式下，缺省设置），还是与 Veritas 集群件协同工作（在 `vcs` 模式下）。

## 确定成员资格模式

使用 `@@membershipmode` 全局变量和 `sybcluster show membership mode` 来确定当前的成员资格模式。如果 Cluster Edition 与 VCS 集群子系统一起运行，则 `@@membershipmode` 和 `sybcluster` 返回“`vcs`”，如果 Cluster Edition 未与 VCS 集群子系统一起运行，则返回“`native`”。例如：

```
select @@membershipmode  
vcs
```

## 更改成员资格模式

通常，无需更改成员资格模式，因为创建 VCS 服务组时会自动将实例更改为 vcs 模式。

使用 `qrmutil --membership-mode` 选项可更改成员资格模式。必须关闭集群。语法为：

```
qrmutil --[-Q | quorum-dev=quorum_device --membership-mode=mode
```

其中：

- `vcs` — 指示 Cluster Edition 与 VCS 集群子系统一起运行
- `native` — 指示 Cluster Edition 不与 VCS 集群子系统一起运行

例如：

```
qrmutil --quorum-dev=/dev/vx/rdisk/quorum_dg/quorumvol --membership-mode=vcs
```

若要手动将 Cluster Edition 配置为装载 VCMP 并且激活 VCS 集成，请向集群配置文件中添加 `membership mode = vcs`。在添加此行之后，Cluster Edition 将作为 VCS 服务组的一部分启动。若要将 Cluster Edition 配置为不与 Veritas 集群子系统一起运行，请将 `membership mode = native` 输入到集群配置文件。

此示例将成员资格模式设置为 vcs：

```
[cluster]
name = isles
max instances = 4
primary protocol = udp
master device = /sybase_cfs/data/master.dat
config file = /sybase_cfg/ASE-15_0/ase.cfg
interfaces path = /sybase_cfg/ASE-15_0/ase.cfg
membership mode = vcs
```

如果您不将 `vcs` 或 `native` 作为 `membership mode =` 行中的参数包含在内（或者不将该行包含在内），则 Cluster Edition 将以 `native` 模式启动。

在生成 master 数据库之后，更改集群配置文件中的成员资格模式不会产生任何作用。在生成 master 数据库之后，请使用 `qrmutil` 更改成员资格模式：

```
qrmutil --quorum-dev=quorum_path --membership-mode=[vcs|native]
```

## 了解故障情况

由于 Cluster Edition 的成员资格更改与 VCS 协调进行并且 Cluster Edition 依赖于 SF for Sybase CE 执行 I/O 防护，因此 Cluster Edition 成员资格管理器需等待 VCS 重新配置其成员资格以及执行 I/O 防护。但是，在出现某些故障时，即使情况并不会导致 VCS 成员资格重新配置，Adaptive Server 也需要进行 I/O 防护。为了获取 I/O 防护，Adaptive Server 会通过仲裁设备向要防护的节点上的 VCS 代理传递一条消息。VCS 代理在收到该消息后会将该节点置于应急状态，这样就会触发 VCS 重新配置。

在出现故障时 VCS 系统如何响应取决于以下情况：

- 主机节点崩溃 — 由 VCS 检测到。VCS 从该发生故障的节点重新配置集群成员资格并执行 I/O 防护。Cluster Edition 从 VCS 接收新的成员资格并执行恢复。
- 主机节点挂起 — 导致 VCS LLT 超时以及 Cluster Edition 心跳超时。VCS 重新配置并执行 I/O 防护。Cluster Edition 从 VCS 接收新的成员资格并执行恢复。正在挂起的节点无法写入共享存储并最终转入应急状态。
- 实例失败或挂起 — 其余的实例检测到心跳故障并执行成员资格仲裁。集群向实例失败的主机节点上的 VCS 代理传递一条消息，这样会触发主机节点出现应急状态。从此时开始，VCS 将以上述“主机节点崩溃”情况中描述的方式工作。
- 实例 shutdown with nowait — Cluster Edition 不会在 VCS 成员资格模式下触发 VCS 成员资格更改。
- 无法访问存储 — 如果您配置了多路径，并且至少有一个指向存储的路径保持可用，则 Cluster Edition 不会受到影响。不过，如果指向存储的所有路径都已丢失，则 VCS 会导致主机应急，集群以上述“主机节点崩溃”中描述的方式处理该主机应急。
- 互连断开 — 基于替代通信通道的可用性进行响应。Adaptive Server 和 VCS 均能将通信切换到替代通道。如果 Adaptive Server 没有可行通道而 VCS 有可行通道，则集群会以上述“实例失败或挂起”中描述的方式进行响应。如果 VCS 没有可行通道，则它以上述“主机节点崩溃”情况中描述的方式工作。

## VCS 故障排除

错误日志中的信息可帮助您确定问题的原因，并找到可能的解决方案。错误日志的缺省位置为：

- Cluster Edition — `$$SYBASE/$$SYBASE_ASE/install/instance_name.log`
- Veritas — `/var/VRTSvcs/log/engine_A.log`
- Linux — `/var/log/messages`
- Solaris — `/var/adm/messages`

---

**注释** 下面列出的许多问题可通过 Veritas 命令（例如 `hagrp` 和 `hares`）解决。有关语法和用法信息，请参见 Veritas 文档。

---

## Cluster Edition 启动失败

很多问题可能导致 Cluster Edition 无法启动。

如果 VCS 无法启动 Cluster Edition 或检测到它已在运行，则 Cluster Edition 会创建一个资源故障。使用 `hagrp -clear` 命令可清除该资源故障。此示例将清除名为“`asegrp1`”的 VCS 组的资源故障：

```
hagrp -clear asegrp1
```

VCS 使用 `ps` 命令定位 `dataserver` 进程：不要重命名 `dataserver` 二进制文件。

使用 `hares -display` 验证 Cluster Edition 资源的 `Home` 属性。`Home` 必须指向 `$$SYBASE`，`dataserver` 二进制文件必须位于 `$$SYBASE/ASE-15_0/bin` 中。如果 `Home` 未指向正确的值，则使用 `hares -modify` 命令更改该值。

VCS 使用 `RUN_server_name` 文件启动实例。如果 `RUN_server_name` 文件不存在，则 VCS 使用 `Home`、`Server` 和 `Quorum_dev` 变量中定义的参数来构建启动实例的执行字符串：

```
HOME/ASE-15_0/bin/dataserver -sserver_name -QQuorum_dev
```

如果您收到资源故障，则应核实以下事项：

- 对于集群中的每个实例，都有一个单独的 `RUN_server_name` 文件。
- `RUN_server_name` 文件包括实例名称作为后缀。
- 每个 `RUN_server_name` 文件都是可执行文件。
- 每个 `RUN_server_name` 文件都包含指向 `dataserver` 二进制文件的正确路径。
- `Quorum_dev` 属性值使用指向现有仲裁设备的正确路径。使用 `hares -display` 可查看 `Quorum_dev` 值。此示例假定有一个名为 “`aseres1`” 的 VCS 资源：

```
hares -display aseres1 -attribute Quorum_dev
```

使用 `hares -modify` 修改路径：

```
hares -modify aseres1 Quorum_dev /sybase/quorum
```

- `Server` 属性对应于实例名称。使用 `hares -display` 命令可验证 `Server` 属性。此示例假定有一个名为 “`aseres1`” 的 VCS 资源：

```
hares -display aseres1 -attribute Server
```

使用 `hares -modify` 修改值：

```
ohares -modify aseres1 Server asel -sys hpcbladel
```

- 您已经使用 `hares -display` 启用了 VCS 资源：

```
hares -display aseres1 -attribute Enabled
```

如有必要，请使用 `hares -modify` 更改它：

```
hares -modify aseres1 Enabled 1
```

## Veritas 日志：“Sybase home directory does not exist”

由于 Sybase 主目录 (`$SYBASE`) 必须存在于 CFS 安装目录中，因此，如果 CFS 安装目录不可用，则实例无法启动。

若要验证 `$SYBASE` 目录是否可用，请使用 `hagrp -dep` 验证 `ASE` 组是否依赖于 `cfsmount` 组：

```
hagrp -dep
```

如果 `ASE` 组当前不依赖于 `cfsmount` 组，请使用 `hagrp -link` 重新建立依赖性：

```
hagrp -link asegrp1 cfsmount1 online local firm
```

## 实例日志：“failed to bind to socket”

当实例存在 VCMP 套接字错误时出现此错误，“sybase”用户必须拥有 VCMP 套接字 (*/tmp/vcmp\_socket*)，然后才能读取和写入套接字并与 VxFEN 通信。

使用 `ls -l` UNIX 命令可验证“sybase”用户是否拥有 */tmp/vcmp\_socket*：

```
ls -l /tmp/vcmp_socket
```

如果“sybase”用户未拥有 */tmp/vcmp\_socket*，请使用 `chown` 命令更改成员资格：

```
chown sybase /tmp/vcmp_socket
```

## 实例日志：“Membership service failed to receive initial message from Veritas cluster membership after 301 seconds. Exiting...”

由于实例存在 VCMP 超时，因此发生该错误。原因及相应的解决方案包括：

- 如果您使 ASE 组联机，则 VxFEND 资源将变得不可用。验证 VxFEND 资源可用性：

a 使 VxFEND 资源脱机：

```
hares -offline vxfend -sys <server name>
```

b 使 ASE 组联机：

```
hagrp -online ASE -sys <server name>
```

- 另一个实例正在使用套接字。使用 UNIX `ps` 命令验证是否在节点上没有其它 `dataserver` 进程在运行：

```
ps -ef | grep dataserver
```

- VxFEND 资源未启用。使用 `hares -display` 验证 VxFEND 是否已启用：

```
hares -display vxfend -attribute Enabled
```

如果 VxFEND 未启用，请使用 `hares -modify`：

```
hares -modify vxfend Enabled 1
```

- ASE 资源不依赖于 VxFEND 资源。使用 `hares -dep` 验证 ASE 资源依赖性：

```
hares -dep
```

如果 ASE 资源不依赖于 VxFEND 资源，请使用 `hares -link` 重新建立依赖性：

```
hares -link aseres1 vxfend
```

- 另一个 Adaptive Server 或 VxFEND 资源与资源启动冲突。使用 `hares -display` 验证是否未启用其它 Adaptive Server 或 VxFEND 资源：

```
hares -display
```

使用 `hares -modify` 纠正任何问题：

```
hares -modify ASE2 Enabled 0
```

### 实例日志：“Failed to open quorum device '*device\_path*'. OS error 13, 'Permission denied'”

由于在对 VCS 原始卷的权限中不包括 “sybase” 用户的访问权限，因此发生错误。向 “sybase” 用户授予对 VCS 卷的读取和写入权限。

```
vxedit -g sybasedg set user=sybase group=sybase mode=660 quorum
vxedit -g sybasedg set user=sybase group=sybase mode=660 master
vxedit -g sybasedg set user=sybase group=sybase mode=660 sybasehome
```

### 实例日志：“basis\_dsizecheck: attempt to open device '*device\_path*' failed, system error is: Permission denied”

因一些与上述错误相同的原因而出错。按照相同的步骤纠正错误。

## 实例日志：“The configuration area in master device appears to be corrupt.”

您可能还会在 Linux 日志中看到以下消息：

```
Linux log: READ CAPACITY failed
Linux log: reservation conflict
Linux log: attempt to access beyond end of device
```

在 master 数据库损坏或集群文件系统导致挂起时发生错误。由于没有针对 Veritas 正确配置磁盘阵列，因此可能会发生此问题。若要纠正此问题：

- 验证磁盘阵列固件是否为最新。
- （仅限 Linux）验证 Veritas Volume Manager 中的磁盘初始化格式是否为分片的。
- 更正磁盘阵列连接模式配置设置，从而使为一个主机保留的逻辑单元号 (LUN) 对其它主机可见，并且验证是否对该阵列正确启用了 SCSI-3 防护。通常，这些设置为：
  - Set Host Mode Option = 19 （启用）
  - Set Host Mode Option = 254 （启用）
  - Set Host Mode Option = 186 （启用）
  - Set Host Mode Option = 60 （启用）
  - Host Mode = 0A
  - Path Switch Mode (Active/Active) = 1 （启用）
  - No\_RSV\_Conf Mode = 1 （启用）
  - Persistent RSV Cluster Mode = 1 （启用）
  - Unique Reserve Mode = 1 （启用）

有关特定于操作系统和硬件的设置，请参见 Veritas 文档。

## Veritas 日志：“Path not found”

由于 VCS 找不到 UNIX sh 命令，无法启动 `dataserver` 二进制文件，因此发生错误。Veritas 会截断 `$PATH` 环境变量的开头部分，因此请验证 Veritas 是否找到了 sh 命令。检查 `$PATH` 的值并重新启动 Veritas：

- 1 检查 `$PATH` 的值

```
echo $PATH
```

- 2 更改 `$PATH` 的值以包含 sh 命令：

```
export PATH=/bin:/sbin:/usr/bin:/usr/sbin
```

- 3 停止 Veritas：

```
Issue hastop -local
```

- 4 重新启动 Veritas：

```
hastart
```

## VCS 在实例启动之后关闭实例并引发资源故障

使用 `hagrp -clear` 可清除资源故障。此命令将清除名为 “asegrp1” 的 VCS 组的资源故障：

```
hagrp -clear asegrp1
```

此错误可能会导致实例向错误日志写入大量错误：

- Failed to identify instance — VCS 启动实例时使用的名称不正确。若要纠正此问题：
  - 验证每个 `RUN_server_name` 文件是否包含仲裁设备的正确路径，以及 `server_name` 的值是否与实例名称相同。
  - 使用 `qrmutil` 验证仲裁配置，尤其是实例和主机名称：

```
qrmutil -Qquorum_path --display=all
```

- Shutdown with nowait detected — VCS 无法确定实例是否正在运行。VCS 使用执行字符串中含有服务器名称的 UNIX `ps` 命令验证实例是否正在运行。确保 `ps` 命令可以找到每个实例：

- （仅限 Solaris）验证 `RUN_server_name` 文件中的实例名称是否显示在前 80 个字符之内
- 使用 `hares -display` 验证 `ASE` 资源的 `Server` 属性。`Server` 属性必须与实例名称相对应。此示例显示 “`aseres1`” 资源的实例名称：

```
hares -display aseres1 -attribute Server
```

如有必要，请使用 `hares -modify` 更改值：

```
hares -modify aseres1 Server asel -sys hpcbladel
```

如果实例的启动需要很长时间，则 Veritas 可能会在启动过程中停止 `dataserver` 进程并引发资源故障。若要防止这些资源故障，请通过调整下列 Sybase VCS 监控参数以允许更长的实例启动时间：

- 调整监控器检查 `ASE` 资源的时间间隔。例如：

```
hatype -modify Sybase MonitorInterval 60
```

- 调整监控器检查以确认 `ASE` 资源是否联机的次数。例如：

```
hatype -modify Sybase OnlineWaitLimit 10
```

- 调整在发出 `Sybase` 资源的资源故障之前允许出现的监控器检查失败次数。例如：

```
hatype -modify Sybase ToleranceLimit 2
```

- Engine exited with signal 11 — 实例在 Linux 平台上失败。如要纠正此问题，请在 `RUN_server_name` 文件中重置 `LD_POINTER_GUARD` 环境变量：

- 在 SUSE 平台上，输入以下命令：

```
export LD_POINTER_GUARD=1
```

- 在 RedHat 平台上，发出以下命令：

```
export LD_POINTER_GUARD=0
```

## VCS 无法关闭实例资源

如果 VCS 在使用 `isql` 连接到实例时出错，则会发生此问题。VCS 使用 `isql` 连接并关闭实例。`isql` 使用 `interfaces` 文件获取连接参数。请验证以下各项：

- 在版本目录 (`$SYBASE`) 中定义了 `interfaces` 文件并对其进行了正确配置。
- `interfaces` 文件包含每个已定义实例的定义。
- 版本目录包含 `SYBASE.sh` 或 `SYBASE.csh` 文件，并且进行了正确配置。

VCS 使用 `SA` 和 `SAPswd` 命令以及 `isql` 连接和关闭实例。使用 `hares -display` 可验证 ASE 资源的 `SA` 和 `SAPswd` 属性：

```
hares -display aseres1 -attribute SA
hares -display aseres1 -attribute SAPswd
```

如有必要，请使用 `hares -modify` 更改这些属性：

```
hares -modify aseres1 SA sa
hares -modify aseres1 SAPswd ""
```

## VCS 组的资源故障

如果您在 `cfsmount` 组中发现资源故障：

- 使用 `hagrp -dep` 验证 `cfsmount` 组是否依赖于 `cvm` 和 `cvmvoldg` 组：

```
hagrp -dep
```

使用 `hagrp -link` 修复任何依赖性：

```
hagrp -link cfsmount1 cvm online local firm
hares -link cfsmount1 cvmvoldg1
```

- 使用 `vxdg list` 验证是否在所有节点上启用了 VCS 安装和卷磁盘组：

```
vxdg -o alldgs list
```

使用 `vxdg deport` 和 `import` 重新启用：

```
vxdg deport disk_group_name
vxdg -s import disk_group_name
```

- 使用 `vxdisk list` 验证 VCS 安装和卷磁盘是否联机：

```
vxdisk list
hastop -local
```

如有必要，请在每个节点上重新启动 VCS：

```
/etc/init.d/vxfen stop
/etc/init.d/vxfen start
hastart
```

- 由于正在使用 VCS 卷，因此 CFS 安装可能会失败。您可能需要重新启动集群中的所有节点以清除任何冲突。请参见操作系统文档。

如果您在 *cvm\_clus* 组中发现资源故障，请使用 *hares -dep* 验证此资源是否依赖于 *cvm\_vxconfigd* 资源：

```
hares -dep
```

使用 *hares -link* 修复任何依赖性：

```
hares -link cvm_clus cvm_vxconfigd
```

如果您在 *vxfscd* 组中发现资源故障，请使用 *hares -dep* 验证此资源组是否依赖于 *cvm\_clus* 资源：

```
hares -dep
```

使用 *hares -link* 修复任何依赖性：

```
hares -link vxfscd cvm_clus
```

## VCS 启动失败

在节点无法加入 VCS 集群时发生。

如果您在 Solaris 或 Linux 错误日志中看到此错误，则节点将因 VCS 无法启动 I/O 防护而无法加入集群：

```
CVMcluster:???:monitor:node - state: out of cluster
```

- 使用 *vx dg list* 验证是否在所有节点上都启用了 *vxfencoordg* 磁盘组：

```
vx disk list
```

如果 *vxfencoordg* 磁盘组未启用，请移出并重新导入该组：

```
vx dg deport vxfencoordg
vx dg -t import vxfencoordg
```

- 使用 *vx disk list* 验证 *vxfencoordg* 磁盘是否联机：

```
vx disk list
```

如果 `vxfcntl` 磁盘未联机，请在每个节点上重新启动 VCS：

```
hastop -local
/etc/init.d/vxfen stop
/etc/init.d/vxfen start
hastart
```

- 使用 `vxfcntladm -g` 验证 `vxfcntl` 磁盘上是否存在保留：

```
vxfcntladm -g
```

如果不存在保留，请停止 VCS，使用 `vxfcntlclearpre` 清除 VCS 保留，并且重新启动 VCS：

```
hastop -local
/etc/init.d/vxfen stop
vxfcntladm -g all -f /etc/vxfentab
vxfcntlclearpre
/etc/init.d/vxfen start
hastart
```

如果有新的节点加入现有集群，则可能需要使用 `gapconfig -x` 重新应用组成员资格配置：

```
gapconfig -x
```

如果在 Solaris 或 Linux 错误日志中看到此消息，则 Veritas 配置守护程序无法启动：

```
ERROR: IPC Failure: Configuration daemon is not
accessible
```

清除并重新启动配置守护程序：

```
vxconfigd -k
"vxiod set 10
"vxconfigd -m disable
"vxdctl init
"vxdctl initdmp
"vxdctl enable
```

如果您在 VCS 错误日志中看到此错误，则在运行 `vxldg init` 时发生了磁盘组初始化失败：

```
Device sda cannot be added to a CDS disk group
```

缺省情况下，`vxldg` 将磁盘初始化为 CDS。重新运行带有 `cds=off` 的 `vxldg` 命令：

```
vxldg init disk_group cds=off
```



## 故障排除

本章提供了针对常见错误进行故障排除的说明。

主题	页码
验证集群环境	190
使用来自早期版本的 <code>dataserver</code> 二进制文件重新启动集群	191
访问磁盘设备时出错	192
验证集群是否已关闭	193
使用 <code>sybcluster</code> 创建集群失败，并出现错误 -131	193
集群创建失败，文件留在 <code>\$\$SYBASE</code> 目录中	194
<code>Unified Agent</code> 已启动，但 <code>sybcluster connect</code> 失败	194
磁盘设备被占用	195
实例无法加入集群	195
私有互连失败	195
客户端连接故障切换失败	196
客户端无法重新连接到备用高可用性服务器	196
如果所有连接都使用 SSL，则 <code>sybcluster</code> 无法连接	197
<code>jConnect</code> 样本禁用 HA	197
PC 客户端安装 — <code>java.lang.NoClassDefFound</code> 错误	198
集群条目 “name” 不包含任何服务器	198
更改口令后， <code>sybcluster</code> 无法管理集群	199
“找不到”代理	200
<code>Sybase Central</code> 无法注册 AMCP 插件	200
UAF 插件注册错误	201
磁盘上的数据不可用：影响数据库创建的问题	202
启用 I/O 防护后，设备访问权限被拒绝	202
<code>sybcluster</code> 无法找到 <code>interfaces</code> 文件	202
IBM 错误	203

## 验证集群环境

使用 Cluster Edition 时发生的许多错误都是由于集群环境中的配置问题所引起的。Sybase 建议，在配置集群之前：

- 验证是否已通过获取 `$SYBASE` 的 `SYBASE.csh` 或 `SYBASE.sh` 文件来设置环境变量。
- 从每个节点运行 `dataserver -v` 以验证主机上是否安装所有必需的库。

如果缺少任何系统库，您将看到操作系统错误，并且数据服务器版本号不会出现。请更正此问题再继续。如果 `dataserver` 显示了版本字符串，且未出错，则可以假定所有必需的系统库均已安装。

- 验证集群中的每个节点是否均可读取并写入每个数据库设备。使用操作系统 `ls -l` 命令测试是否能够使用 `dd` 操作系统实用程序读取和写入这些设备。

测试设备的可读性：

```
dd if=<device path> of=/dev/null count=x
```

您应获得类似于如下内容的结果：

```
%dd if=/dev/raw/raw123 of=/dev/null
count=10
10+0 records in
10+0 records out
```

也可以使用 `dd` 实用程序测试设备的可写入性。但是，只有在设备中没有要保留的数据的情况下，您才应执行此操作。

- 运行 `ping` 实用程序验证所有节点之间的连接。尝试从每个节点对主机名或其它每个节点的网络地址执行 `ping` 操作。请对每个要使用的网络执行此操作。例如，如果您的配置使用一个公共网络和两个专用网络，请验证对节点和网络地址的所有组合执行 `ping` 操作是否成功。

使用 `sybcluster 'show cluster config'` 参数确定每个实例使用的私有互连地址。例如，如果您的集群包含节点 `node1` 和 `node2`，则 `sybcluster` 将显示类似于如下内容的信息：

```
SYBCE> show cluster config
** Cluster configuration for "SYBCE" **
  Interface Path "/sybce"
  Trace Flags:
  There were are no trace flags.
  Maximum Instances "4"
  Quorum "/dev/raw/raw23"
  Master Device "/dev/raw/ra24"
  logfile INSTANCE1 /sybce/ASE15_0/install/GATEST_INSTANCE1.log
```

```

run_parameters INSTANCE1
logfile INSTANCE2 /sybce/ASE-15_0/install/GATEST_INSTANCE2.log
run_parameters INSTANCE2
Primary Interconnect "udp"
Server[1] INSTANCE1 node1_priv 49152 49171
Server[2] INSTANCE2 node2_priv 49172 4919

```

此集群包括互连网络地址 `node1_priv` 和 `node2_priv`。从 `node1` 执行 `ping node2_priv` 可验证 `node2` 上的专用网络地址是否可从 `node1` 访问。从 `node2` 执行 `ping node1_priv` 可验证 `node1` 上的专用网络是否可从 `node2` 访问。

如果 `ping` 命令失败或错误消息表明专用网络有问题，请检查：

- 包含在 `/etc/hosts` 文件中的信息
- 专用网络使用的网线、路由器或交换机的状况
- 在 `sybcluster` “`show cluster config`” 命令报告的集群配置中指定的名称或 IP 地址

## 使用来自早期版本的 *dataserver* 二进制文件重新启动集群

如果存在以下情形，Cluster Edition 将无法启动，并将此消息写入错误日志 “Cluster is running with message version *y*. This version of ASE requires message version *x*”：

- 尝试在同一正在运行的集群中使用不同版本的 *dataserver* 二进制文件。
- 在未正常关闭使用之前版本的 *dataserver* 二进制文件的集群的情况下，应用 EBF 或 ESD。

### 解决问题

使用旧版本的 *dataserver* 二进制文件启动一个集群实例，然后发出 `shutdown cluster`。

使用新版本的 *dataserver* 二进制文件重新启动集群。

如果之前版本的 *dataserver* 二进制文件不可用，您可以通过重新创建仲裁设备解决此错误消息。

---

**警告！** 这些步骤绕过了 Adaptive Server 的安全检查，您必须确保在执行这些步骤时没有实例在运行。

---

- 1 验证集群的所有实例是否均已关闭。
- 2 使用 `qrmutil` 提取集群输入文件信息：

```
$SYBASE/ASE-15_0/bin/qrmutil --quorum_dev=path_to_quorum  
--extract_config=quorum.out
```

- 3 使用新的 `dataserver` 二进制文件启动 Cluster Edition，然后重建仲裁设备：

```
dataserver --quorum_dev=path_to_quorum --instance=instance_name  
--buildquorum=force --cluster_input=quorum.out
```

- 4 关闭集群
- 5 使用常规过程启动集群。

---

**警告！** `--buildquorum` 或 `--cluster_input` `dataserver` 参数仅用于这些步骤。在后续的集群或实例重新启动期间不要使用这些参数。

---

## 访问磁盘设备时出错

Cluster Edition 使用的所有磁盘设备都必须配置为可从集群中的所有节点访问。这些设备在所有节点上的路径都必须是相同的，并且用于启动集群的帐户必须拥有读取和写入所有磁盘设备的权限。

如果 Cluster Edition 报告无法访问任一设备，请从集群的每个节点验证以下几点：

- 配置集群时指定的设备路径从集群中的所有节点都可访问。
- 用于启动集群的帐户拥有读取和写入这些设备的权限。
- 设备路径在集群中的所有节点上都是相同的。
- 用以指向设备的任何符号链接都是正确的。

使用 UNIX `ls` 和 `ls -l` 命令可验证路径和文件的权限。可以使用 UNIX `dd` 实用程序验证这些设备是否可以通过 Sybase 帐户读取和写入。

请参见第 190 页的“验证集群环境”。

## 验证集群是否已关闭

如果集群已崩溃并在仲裁设备中留下了一些不一致的状态信息，则 `sybcluster` 实用程序可能无法确定集群是否正在运行。如果不确定集群的状态，请分析集群中的每个实例以确定集群的状态。

- 1 使用 `isql` 登录到集群中的每个实例。
- 2 使用 `sybcluster 'show cluster status'` 查看集群中每个实例的状态。
- 3 如果 `sybcluster` 没有报告所有实例的状态均为 'Up'，且 `heartbeat` 为 'Yes'，则相应实例可能已关闭。
- 4 在集群中的每个节点上使用 UNIX 'ps' 命令确定是否正在针对为相应节点配置的每个实例运行表示 `dataserver` 程序的进程。
- 5 启动第一个实例后，发出 `sybcluster start cluster` 启动集群中所有剩余的实例。

如果您确定这些实例没有在运行，请解锁集群并重新启动：

```
start instance instance_name unlock
```

## 使用 `sybcluster` 创建集群失败，并出现错误 -131

`sybcluster create cluster` 命令可能失败，出现错误 -131 并且发出消息指出无法访问定义裸设备的父目录：

```
INFO - Choosing the first instance to be created using the connected agent...
ERROR - Parent directory access error. The parent directory /dev/rdsd for the
device can not be accessed. Please change the protection on the device and try
again.
INFO - Create cluster error: -131
```

裸设备的名称可能拼写有误。检查每个裸设备的完整名称并验证其是否正确。

## 集群创建失败，文件留在 \$SYBASE 目录中

如果 `sybcluster create cluster` 命令或 Sybase Central Adaptive Server 插件的创建集群向导因出现错误情况而终止，则某些文件可能会留在集群安装目录中。请在尝试再次创建集群前删除这些文件：

- 1 删除安装目录内 `interfaces` 文件中集群的任何条目。
- 2 从这些位置删除每个节点的 Unified Agent 插件目录：

```
$SYBASE/UAF-2_5/nodes/node_name/plugins/cluster_name
```

- 3 从 \$SYBASE 目录中删除名称以 `*.res` 结尾的资源文件。
- 4 从 \$SYBASE 目录中删除名称以 `*.inp` 结尾的集群定义文件。
- 5 从 \$SYBASE/ASE-15\_0/install 中删除 `RUN_instance_name` 中间文件。
- 6 从 \$SYBASE/ASE-15\_0/install 中删除任何错误日志文件。
- 7 从 \$SYBASE 中删除任何实例配置文件。

只需执行步骤 1 和 2，便可开始创建另一集群。不过，需要执行其它步骤来减少安装区域的文件数量。

## Unified Agent 已启动，但 `sybcluster connect` 失败

`sybcluster 'connect'` 命令可能失败，原因是在 Unified Agent 启动过程中或执行 `sybcluster` 命令时网络配置信息不正确。

- 请检查集群中每个代理的 Unified Agent 日志文件，以确定在为代理打开 RMI 监听器时是否报告了任何错误。代理日志文件为：

```
$SYBASE/UAF-2_5/nodes/node_name/log/agent.log
```

- 在启动 `sybcluster` 时指定 `-F` 参数的正确节点名称和监听端口号：

```
sybcluster -U uafadmin -P -C MYCLUSTER  
-Fnode1:1234,node2:1234
```

## 磁盘设备被占用

在 Linux 上，某些设备在缺省情况下绑定到磁盘。如果您试图创建一个集群并无意中指定了其中一个此类设备，则 `disk init` 将失败，且创建集群向导或 `sybcluster` 无法创建集群。

启动创建集群向导或 `sybcluster` 会话前，请检查以确保相应的磁盘设备可用。

请参见第 190 页的“验证集群环境”。

## 实例无法加入集群

实例可能因为用于支持互连通信的专用网络存在问题而无法加入集群。如果集群中的一个或多个实例无法启动：

- 请检查那些无法加入的实例的错误日志，查看是否有任何消息指示网络通信已失败。错误日志位于 `$$SYBASE/$$SYBASE_ASE/install/instance_log`。
- 验证集群所使用的节点上使用的专用网络是否均可从集群中的所有节点访问。使用 `ping` 实用程序执行此操作。请参见第 190 页的“验证集群环境”。
- 检查 UAF 代理日志中是否存在错误。

## 私有互连失败

如果私有互连未正确配置，集群将无法启动。使用操作系统 `ping` 命令验证私有互连是否正在工作。如果 `ping` 失败，请联系系统管理员以启用各个节点之间的互连通信。

请参见第 190 页的“验证集群环境”。

## 客户端连接故障切换失败

当客户端连接的集群所用的 IP 地址和节点不包括在本地 */etc/hosts* 或 DNS 中时，则客户端连接故障切换会失败，从而发生此错误。

Adaptive Server 可发送列在集群的 *interfaces* 文件中的故障切换实例地址。如果集群的 *interfaces* 文件中列出的实例网络地址为主机名称，则 Adaptive Server 会将相应的主机名称返回到客户端应用程序。不过，客户端应用程序使用 DNS 或 */etc/hosts* 文件解析集群实例的主机名称，因此，如果客户端的 */etc/hosts* 文件或 DNS 服务器中没有主机名称，则故障切换不会成功。

确保集群中的所有节点是否均已在客户端系统的 DNS 服务器或客户端系统的 */etc/hosts* 文件中列出。

请尝试在集群的 *interfaces* 文件中使用 IP 地址，而非主机名称。

## 客户端无法重新连接到备用高可用性服务器

当客户端连接到的实例出现故障时，有 DNS 名称解析问题的客户端可能无法重新连接到备用高可用性服务器，并生成以下消息：

```
Connection to Sybase server has been lost, connection
to the next available HA server also failed. All active
transactions have been rolled back.
```

当客户端的主机无法正确地解析备用实例的 *hostname* 时，将发生 DNS 名称解析问题。若要确定此错误是否是由 DNS 名称解析问题引起的，请对高可用性列表中的每个备用实例发出 `ping hostname` 命令，其中 *hostname* 为备用服务器所驻留的主机。如果无法解析 *hostname*，或者解析为错误的 IP 地址，请按照相应系统管理文档纠正名称解析问题。

## 如果所有连接都使用 SSL，则 sybcluster 无法连接

如果所有 Adaptive Server 监听端口都使用 SSL，则 Adaptive Server 将发出以下错误消息：

```
2008-03-20 10:42:46,260 ERROR [Timer-6]
GA1:GA1_1:SQLConnect:270:Login Failure - The user "sa"
and the entered password is not authorized to connect
to the cluster.
```

验证集群中的每个实例是否都有一个非 SSL 连接，并且该连接包含在 Unified Agent 使用的 interfaces 文件中。

## jConnect 样本禁用 HA

在适用于 Adaptive Server 15.0 版的 Open Client、Open Server、ODBC、jConnect 和 ADO.NET 的 What's New in ESD #12（《ESD #12 中的新增功能》）文档中，所包含的针对 Cluster Edition 的样本 jConnect 连接字符串不正确。

文档中的该样本连接字符串为：

```
URL="jdbc:sybase:Tds:server1:port1,server2:port2,...,s
erverN:portN/mydb?JCONNECT_VERSION=6&PACKETSIZE=1024&D
YNAMIC_PREPARE=true&REQUEST_HA_SESSION=true"
```

此字符串会失败，因为 JCONNECT\_VERSION=6 参数模仿高可用性协同服务器功能。当客户端尝试使用 jConnect 驱动程序的故障切换属性，但在该连接字符串中没有指定 hafailever 服务器时，JDBC 驱动程序会导致客户端 Java 异常。

下面是正确的连接字符串：

```
URL="jdbc:sybase:Tds:server1:port1,server2:port2,...,s
erverN:portN/mydb?&PACKETSIZE=1024&DYNAMIC_PREPARE=tru
e&REQUEST_HA_SESSION=true"
```

## PC 客户端安装 — java.lang.NoClassDefFound 错误

当您在 Windows 平台上使用 MKS tar 实用程序而不是 Winzip 实用程序解压 PC 客户端的 tar 文件时，会发生该错误。在 Windows 平台上，MKS tar 实用程序会截断完整的路径名，造成文件丢失。

在 UNIX 平台上，如果不使用 GNU tar 实用程序来解包 Adaptive Server 安装程序，也会发生此问题。

对于 Windows 平台，请始终使用 Winzip 实用程序解压安装程序。对于 UNIX 平台，请始终使用 GNU tar 实用程序解包安装程序。

## 集群条目 “name” 不包含任何服务器

如果 UAF 代理无法访问仲裁设备，当您发出 `sybcluster start cluster` 和 `start instance` 命令时，可能会发生以下错误：

```
start cluster
ERROR - The cluster entry SDCDEMO did not contain any servers
start instance INSTANCE1
ERROR - The cluster entry SDCDEMO did not contain any servers
```

- 相应的 UAF 代理可能没有读取仲裁设备的权限。验证您是否已使用正确的用户帐户启动相应的 UAF 代理，并且该帐户具有读取和写入仲裁设备的权限。
- 在集群的一个或所有节点上，`sybcluster` 和相应的 UAF 代理使用的仲裁设备路径不正确。如果使用符号链接，请确保所有节点上的各个链接都正确。
- 另一个进程使 UAF 代理无法读取仲裁设备。这可能是由于磁盘存储系统的功能问题或配置错误所引起的。

## 更改口令后，sybcluster 无法管理集群

sybcluster 使用 UAF 代理进行连接和执行针对集群的操作（例如，shutdown cluster）。UAF 代理必须登录到集群才能执行此操作。若要登录，UAF 代理必须使用正确的 sa 登录名和口令。

sa 登录名和口令采用加密形式存储在集群的 UAF 插件 *configure* 文件中。使用 `sybcluster set cluster login` 可更改 UAF 用于连接集群的登录名和口令。

无论何时更改 Adaptive Server 口令，都必须同时更改 UAF 代理登录名。

若要更改 UAF 用于连接到集群的登录名和口令，请执行以下操作之一：

- 使用 `sybcluster set cluster login` 设置 UAF 登录名和/或口令。在发出此命令之前连接到集群。

语法为：

```
set cluster login sa-login [password sa-password
```

- 使用 Adaptive Server 插件修改 UAF 插件：
  - a 连接到集群。
  - b 从“服务器实例”中，左键单击树视图窗格中列出的一个实例。
  - c 从右窗格中选择“代理属性” (Agents Attributes)。
  - d 单击“口令”行。
  - e 为管理当前实例的代理输入新的口令。
  - f 对集群中的每个实例重复步骤 c – e。
- 编辑 UAF 配置文件。

---

**注释** 编辑 *agent-plugin.xml* 文件之前，必须关闭所有集群 UAF 代理。

---

UAF Adaptive Server 登录名和口令存储于：

```
$SYBASE/UAF-2_5/nodes/<node name>/plugins/  
<cluster name>/agent-plugin.xml
```

例如：

```
<set-property property="ase.user" value="sa" /><set-property  
property="ase.password"  
value="REVTe1NZVUFGfWNvbsS5zdW4uY3J5cHRvLnByb3ZpZGVyLlN1bkpDRXtTWVVBn1j  
UEZPSkJoTTZ2QT0=" />
```

---

**注释** 如果口令标记丢失，请执行以上步骤 a – f。

---

以明文形式输入口令。 `passencrypt` 将生成加密的文本字符串。在 `agent-plugin.xml` 文件中 `password` 标记的引号间输入此整个字符串。

生成加密的口令：

```
$$SYBASE/UAF-2_5/bin/passencrypt
```

## “找不到”代理

`sybcluster show agents` 不会显示所有代理。

通常，集群中的主机系统名称很相似。如果使用 `sybcluster -F` 指定与集群中每个代理的连接，请确保每个主机系统的拼写正确，并且已指定端口号且这些端口号无误

## Sybase Central 无法注册 AMCP 插件

如果在尝试注册 AMCP 插件 (`amcplugin.jpr`) 时收到以下消息：

```
Could not read manifest.file
```

安装程序已将 `amcplugin.jar` 文件重命名为 `amcplugin.jar.installed` 以避免覆盖现有文件。

在尝试从命令行运行 `registerAMCP` 时，您将看到：

```
Error: Unable to find the AMC Plugin binary.  
Please check that $$SYBROOT has been set correctly and  
that the file  
'/AMCP/lib/amcplugin.jar' exists.\n
```

请将 `amcplugin.jar.installed` 重命名为 `amcplugin.jar`。

## UAF 插件注册错误

可能会由于现有的集群参数值与较新的配置参数不一致而看到以下错误：

```
2008-06-16 14:05:16,051 ERROR [main] Failed to register plugin
com.sybase.ase.cluster_15.0.1. Class
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin not found. Ignored.
java.lang.ClassNotFoundException:
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin
2008-06-16 14:05:16,052 INFO [main] Finished loading primordial services.
2008-06-16 14:05:16,063 WARN [main] Bootstrap completed with 1 error(s):
2008-06-16 14:05:16,064 WARN [main] Failed to register plugin
com.sybase.ase.cluster_15.0.1. Class
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin not found. Ignored.
java.lang.ClassNotFoundException:
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin
```

如果收到此错误，则很可能存在以下情形：

- UAF 代理将不会启动，并且在 *agent.log* 文件的接近末尾处显示 Bootstrap completed with x error(s):。
- 您已多次创建、删除和重新配置同一集群。

如果您没有将 *uafstartup.sh* 作为后台进程执行，则在 *agent.log* 文件中或终端窗口中会出现此错误消息。请执行以下操作之一：

- 清除可能引入错误的旧参数，或者
- 删除 *\$SYBASE UA/nodes/<node\_hostname>/plugins* 中的 *cluster* 目录。您可以永久删除该文件夹或将该文件夹重新分配到该目录结构以外。

---

**注释** 不要删除 *\*snmp* 和 *\*sysam* 文件夹。

---

执行以上任一步骤后，重新启动 UAF 代理。您可能必须手动重新配置该插件。

---

**注释** 如果使用上述步骤从某一节点删除了集群的 *plugin* 目录，请使用 *deploy plugin* 重新创建此目录。对于集群，要求其中的每个节点都包含一个 *plugin* 目录。

---

## 磁盘上的数据不可用：影响数据库创建的问题

磁盘卷标存储在每个磁盘的块 0 中。如果用于设备创建的原始数据切片在块 0 上开始，则它们可能会覆盖磁盘卷标并使磁盘上的数据不可访问。

不要创建在块 0 上开始的原始磁盘切片。

## 启用 I/O 防护后，设备访问权限被拒绝

在 Solaris 系统上，启动 Cluster Edition 的 UNIX 用户必须具有 SYS\_DEVICES 可继承特权，才能访问用于数据库和仲裁的裸设备（`/dev/raw/raw#` 或 `/dev/rdsk/c##d##s#`）。

SYS\_DEVICES 特权可授予 Adaptive Server 执行用于 I/O 防护的 SCSI-3 PGR 命令的权限。可授予 UNIX 用户临时或永久的 SYS\_DEVICES 特权：

- 若要授予临时 SYS\_DEVICES 特权，请将 SYS\_DEVICES 特权添加到当前用户的 shell 进程的可继承特权集中：

```
ppriv -s l+sys_devices $$
```

- 若要授予永久的 SYS\_DEVICES 特权，请使用 `usermod` UNIX 命令将 SYS\_DEVICES 添加到任何用户的可继承特权集中：

```
usermod -K defaultpriv=basic,sys_devices user_login
```

## sybcluster 无法找到 interfaces 文件

sybcluster 可能报告一条错误消息，指出在一个不存在的目录中无法找到 interfaces 文件：

```
ERROR - The interfaces file /remote/ase_cluster/sdclinux/UAF-2_5/bin/interfaces  
could not be found.
```

当一个或多个 UAF 代理无法访问仲裁设备时，通常发生此错误。发生此错误的可能原因如下：

- 仲裁设备不存在于所配置的位置。
- 仲裁设备的权限不允许相应 UAF 代理读取或写入数据。

若要解决此问题，请验证以下各项：

- 1 在集群中的每个节点上启动 UAF 代理前，您是否已获取了 *SYBASE.sh* 或 *SYBASE.csh* 文件。
- 2 仲裁设备是否存在并且用于启动相应 UAF 代理的登录名具有对仲裁设备的读写权限。

## IBM 错误

本节描述在 IBM AIX 操作系统上运行 Cluster Edition 时可能遇到的错误。

### 未启用异步 I/O

如果在尝试启动 Cluster Edition 时，IBM AIX 上没有启用异步 I/O，则 **dataserver** 二进制文件将发出以下错误消息，并且 Cluster Edition 无法启动：

```
exec(): 0509-036 Cannot load program dataserver because of the following errors:
0509-130 Symbol resolution failed for /usr/ccs/lib/libc.a[aio_64.o] because:
0509-136 Symbol kaio_rdwr64 (number 1) is not exported from dependent module
/unix.
0509-136 Symbol listio64 (number 2) is not exported from dependent module
/unix.
0509-136 Symbol acancel64 (number 3) is not exported from dependent module
/unix.
0509-136 Symbol iosuspend64 (number 4) is not exported from dependent module
/unix.
0509-136 Symbol aio_nwait (number 5) is not exported from dependent module
/unix.
0509-136 Symbol aio_nwait64 (number 6) is not exported from dependent module
/unix.
0509-136 Symbol aio_nwait_timeout (number 7) is not exported from dependent
module /unix.
0509-136 Symbol aio_nwait_timeout64 (number 8) is not exported from dependent
module /unix.
0509-026 System error: Error 0
0509-192 Examine .loader section symbols with the 'dump -Tv' command.
```

有关启用异步 I/O 的信息，请参见 IBM AIX 操作系统文档。

## 设备权限不正确

如果您无权管理 IBM AIX 上的裸设备，则在您尝试启动 Cluster Edition 时，操作系统会发出以下消息，并且集群不会启动：

```
dopen: open '/dev/device_name', Not owner
```

- 授予正确的权限（以下权限之一：PV\_ROOT、PV\_SU\_ 或 PV\_KER\_RAS）：

```
setsecattr -p iprivs=+PV_KER_RAS $$
```

这些权限会在 Cluster Edition 启动时由 Cluster Edition 进程继承。

- 授予未联网的 (NIS) 用户运行管理设备的进程的权限：

- a 创建用户：

```
mkuser sybase
```

- b 创建角色：

```
mkrole authorizations=aix.device.manage.change  
role_disk_access
```

- c 为用户指派角色：

```
chuser roles=role_disk_access  
default_roles=role_disk_access sybase
```

## 正在使用设备的另一计算机

如果因为另一计算机上的进程正在使用设备，且该设备一次只能由一台计算机使用，而导致 Cluster Edition 无权访问该设备，则操作系统将发出以下消息：

```
The IBM AIX SDC dataserver may fail to run with one of the following errors:  
Quorum library error 1: Failed to open quorum device '/dev/disk_name'. OS error  
16, 'Device busy'
```

或：

```
dopen: open '/dev/disk_name', Device busy
```

设备必须允许多个服务器同时访问。数据库设备必须具有共享预留，在这里，预留键是在集群配置文件中定义的实例 ID（例如，where ID = 1）：

在集群中的每台计算机上更改仲裁设备的设备访问限制：

```
chdev -l device_name -a reserve_policy=no_reserve
```

例如，如果磁盘设备名称为 */dev/hdisk1*：

```
chdev -l /dev/hdisk1 -a reserve_policy=no_reserve
```

在集群中的每个实例上，对每个数据库设备运行此命令，以更改数据库设备的设备访问限制：

```
chdev -l device_name -a PR_key_value=instance_ID -a reserve_policy=PR_shared
```

例如，若要在实例 1 上更改 */dev/rhdisk2* 的设备访问限制：

```
chdev -l hdisk2 -a PR_key_value=1 -a reserve_policy=PR_shared
```

## 运行 chdev 时出错

运行 `chdev` 命令时，您可能会看到以下错误：

```
Method error (/usr/lib/methods/chgdisk):  
0514-047 Cannot access a device.
```

相应设备当前正在使用。请关闭所有访问该设备的进程。



## 使用 Adaptive Server 插件管理集群

使用用于 Sybase Central 的 Adaptive Server 插件，可以执行针对集群的管理任务。例如，创建集群，添加实例，启动和停止集群或实例，创建或修改逻辑集群，管理负载管理。

Adaptive Server 插件用于取代命令行方法。

主题	页码
<a href="#">管理共享磁盘集群</a>	207
<a href="#">管理多个临时数据库</a>	219
<a href="#">管理负载</a>	223
<a href="#">管理路由</a>	236

有关使用 Sybase Central 和 Adaptive Server 插件的详细信息，请参见《系统管理指南》中的第 4 章“用于 Sybase Central 的 Adaptive Server 插件简介”。

### 管理共享磁盘集群

Adaptive Server 插件使您可以在 Sybase Central 中管理共享磁盘集群环境。

必须将 Unified Agent 与集群代理 Adaptive Server 插件一起运行，才能通过 Adaptive Server 插件访问所有集群管理功能。请参见第 209 页的“在 Adaptive Server 插件中启用 Unified Agent 功能”。

## 连接到集群

当启动 Sybase Central 时，主窗口打开，其中显示了 Adaptive Server 插件以及您以前连接到的集群和实例的图标列表。如果集群正在运行，则集群名称旁出现一个绿色三角形。

如果 Adaptive Server 插件管理的集群未运行，则服务器图标中出现一个红色方块。如果未显示三角形或方块指示符，请参见第 215 页的“启动集群”上的有关配置 Adaptive Server 插件和 Unified Agent 的说明。

连接到列表中的运行集群的最快方法是右键单击集群名称，然后选择“连接”。Adaptive Server 插件使用以前的连接数据进行连接。如果树视图中未显示集群，您可以使用 Server Discovery 查找它，或者提供集群的主机和端口、登录名和口令信息。任何一种方法都可以通过单击 Sybase Central 窗口顶部附近工具栏中的“连接”图标来启动。如果您知道所需的连接信息，请在“连接”窗口中的相应字段中输入此信息。如果您没有集群或集群节点的主机和端口号，请输入登录名和口令，然后单击“查找”按钮。Unified Agent 搜索集群，并提供可用集群的列表。如果列表不包含您要搜索的集群，请参见第 209 页的“更改服务器发现设置”。

### ❖ 连接到集群

- 1 选择“工具”|“连接”。

如果运行多个注册的 Sybase Central 插件，请选择 Adaptive Server 插件。

- 2 输入要用于连接到实例的登录名。
- 3 输入登录口令。
- 4 从“服务器名”列表中选择集群名称（该列表中填充了 Linux、Solaris、IBM AIX 和 HP-UX 的 `interfaces` 文件中的条目以及 Windows 的 `sql.ini` 文件中的条目），或者键入集群节点的主机和端口。
- 5 （可选）指定集群中实例的主机和端口。
- 6 单击“确定”。

### ❖ 快捷方式

- 右键单击集群图标，然后选择“连接”。

一旦连接到集群，集群图标即从灰色变为蓝色。

## 使用工具栏断开与集群的连接

- 1 选择要与其断开连接的集群的图标。
- 2 选择“文件” | “断开连接”。

### 快捷方式

- 右键单击集群的快捷方式图标，然后选择“断开连接”。
- 选择要断开连接的集群。从工具栏中选择“断开连接”。

## 在 Adaptive Server 插件中启用 Unified Agent 功能

- 1 选择“工具” | “Adaptive Server Enterprise” | “首选项”。在“首选项”选项卡上，选择：“启用 Unified Agent 功能”(Enable Unified Agent features)、“检查服务器状态”(Check Server Status)、“使用代理端口号”(Use Agent Port Number)。缺省 UAF 端口为 9999。如果需要检查另一个端口上的 UAF 代理，请在此处输入此值

可以更改 UAF 端口号，缺省值为 9999。

- 2 单击“确定”。如果集群由 UAF 代理监控，则当集群未运行时，集群图标中会出现一个红色方块。如果集群正在运行，则集群图标上出现一个绿色三角形。

## 更改服务器发现设置

您不需要选择单一发现方法。服务器发现会搜索所有指定的发现方法。

- 1 右键单击集群的名称。
- 2 选择“连接”。
- 3 从“连接”对话框，选择“设置”。
- 4 选择“服务器发现”(Server Discovery)选项卡。

- 5 选择发现方法。
  - JINI — 一个开放体系结构，使开发人员可以创建以网络为中心的非常适合更改的服务。JINI 针对发现提供了标准的查找服务。  
请参见 Unified Agent and Agent Management Console Version 2.0 for Windows and UNIX (《适用于 Windows 和 UNIX 的 Unified Agent 和 Agent Management Console 2.0 版》) 中的第 2 章 “Installing and Configuring Unified Agent and Agent Management Console” (安装和配置 Unified Agent 和 Agent Management Console)。
  - 用户数据报协议 (UDP) — 一种网络协议，它为应用程序提供一种过程，以使用最简单的协议机制将消息发送到其它程序。

---

**注释** 如果仅使用 UDP，则仅发现与运行 Sybase Central 的服务器位于同一子网中的服务器。

---

- 6 单击 “添加”。
- 7 如果在上一步中选择了 JINI:
  - 选择 JINI 服务器的主机
  - 选择缺省主机和端口或输入新的主机和端口
- 8 单击 “确定”。
- 9 要添加或编辑发现过滤器，请单击 “过滤器”。Server Discovery 仅使用选定的用于搜索的过滤器来指定过滤器：
  - a 单击 “添加”。
  - b 选择 “启用此过滤器”。
  - c 选择要过滤的目标、主机、名称、操作系统、平台、端口、版本类型、状态、版本、生成日期。
  - d 选择条件：包含 (contains)、不包含 (does not contain)、是 (is)、不是 (is not)、开头是 (starts with)、结尾是 (ends with)。
  - e 输入要过滤的条件字符串。
  - f 单击 “确定”。

- 10 配置 Adaptive Server 插件以发现系统上当前运行的集群。请使用
  - 删除 — 从列表中删除发现服务。
  - 编辑 — 编辑当前发现服务的设置。
  - 向上 — 在列表中将选定发现服务向上移动。
  - 向下 (Down) — 在列表中将所选发现服务向下移动。
- 11 如果使用的是 LDAP 服务器，请选择 LDAP 窗格：
  - a 选择 LDAP 服务器名称。
  - b 使用条件来设置搜索超时时间。
  - c 输入用户名和口令以登录节点。
  - d 在“选择集群节点”框中选择要登录到的节点。
  - e 单击“确定”。

## 显示集群属性

要查看集群属性，请右键单击集群名称，然后选择“属性”。Adaptive Server 插件显示“服务器属性”对话框，该对话框中包括“常规”、“配置”、“日志空间”、“Job Scheduler 服务器”、“代理”、“服务器日志”、“集群”和“本地化”选项卡。

### “常规属性”选项卡

“常规属性”选项卡显示有关 Cluster Edition 的下列信息：

- 类型 — Adaptive Server 的版本。
- 版本 (Version) — 软件的版本。
- 版本类型 — 该版本是软件的 Beta 版本还是 Production 版本。
- 平台 — 节点上运行的计算机的平台。
- 主机名 — 此节点上运行的计算机的名称。
- 操作系统 — 节点上运行的操作系统。
- 生成选项 — 特定于当前运行的 Adaptive Server 版本的选项。
- 生成日期 — 生成 `dataserver` 二进制文件的日期。
- 版本 (Edition) — 当前运行的 Adaptive Server 的版本。

- 许可证 — Adaptive Server 许可证的当前状态。选择“详细信息”可获取详细信息。
- 字符集 — 当前配置的缺省字符集。
- 语言 — 当前配置的语言。
- 排序顺序 — 当前配置的排序顺序。
- 状态 — 服务器的状态（正在运行或关闭）。
- ASE 日志文件 — 日志文件的位置。

## “集群”选项卡

“集群”选项卡显示有关集群的一般信息，包括：

- 最大实例数 — 此集群允许的最大实例数
- 安装模式 — 共享或私有
- 成员资格模式 — vcs 或 native
- 仲裁设备 — 仲裁设备的位置
- 主设备 — 主设备的位置

“集群”选项卡包含可用实例列表，包括其节点名称和端口号。

## “配置”选项卡

在“集群创建向导”中，每个实例使用相同的服务器配置文件 (*server\_name.cfg*) 确定其配置。缺省情况下，集群中的所有实例将 *cluster\_name.cfg* 文件用于集群配置。但是，当配置实例时可以指定另一个配置文件，从而使您可以为不同实例配置不同配置值。

可以将“配置”面板用于集群和实例。

要查看当前配置设置，请右键单击集群或实例名称，然后选择“配置”。也可以选择“属性”，然后选择“配置”面板。

有关配置参数的详细信息，请参见《系统管理指南》。有关在确定最佳设置时要考虑的配置问题的讨论，请参见 *Performance and Tuning Guide: Basics*（《性能和调优指南：基础知识》）。以下规则控制哪些用户可以设置配置参数：

- 指派了系统安全员（“sso\_role”）角色的登录可以重新设置下列参数：
  - allow updates
  - audit queue size
- default character set id 参数是在集群安装期间自动设置的，无法从 Sybase Central 中重新设置该参数。
- 指派了系统管理员（“sa\_role”）角色的登录可以重新设置所有其它参数。

### 需要重新启动的参数

某些配置参数值是动态的，这意味着只要您重新设置值，参数就会生效。其它参数在您重新启动集群之前不会更改（这些参数称作静态参数）。Adaptive Server 插件指示当您选择参数名称时参数是否需要重新启动。

### 删除特定于实例的配置参数

集群实例的“配置”面板包括一个“删除”按钮，仅针对实例级别的配置值启动该按钮，该按钮不出现在集群配置属性选项卡上。删除参数意味着您将其作为与集群范围值无关的、特定于实例的设置来删除。

“删除”一直处于禁用状态，直到您更改某值并选择“应用”为止。下一次选择此配置参数时，将启用“删除”，您可以选择它来删除配置参数。

### 设置配置参数

Cluster Edition 包括全局和实例配置参数。全局配置参数影响整个集群，而实例配置参数仅影响设置这些参数时所用的实例。要设置全局配置参数，请打开集群的“配置”选项卡，选择集群名称，然后选择要重新设置的配置参数。

缺省情况下，实例使用全局配置值，除非实例设置覆盖了它们。

❖ **设置实例的配置参数**

- 1 右键单击要配置的实例名称。
- 2 选择“配置”（或选择“文件”|“属性”，然后单击“配置”选项卡）。
- 3 选择要显示的功能组，或者选择“全部”（All）。
- 4 选择要更新的参数。有关选定参数的简要说明，请查看“解释”框。
- 5 在表的“值”列中输入新的值。
- 6 单击“确定”（如果更改了多个配置值，请单击“应用”）。
  - 如果参数立即生效，则在“值”列中列出该参数。
  - 如果参数要求重新启动 Adaptive Server，则在“待定值”列中列出该参数。

**“日志空间”选项卡**

“日志空间”面板显示有关集群的当前日志空间的下列信息：

- 数据库（实例）— 日志空间的名称。如果日志特定于某实例，实例名称将显示在括号中。如果没有括号，则日志针对的是集群。
- 总计 (MB) — 可用日志空间总量，以兆字节计。
- 已使用 (MB) — 当前使用的日志空间量，以兆字节计。
- 可用 (MB) — 日志的可用空间量，以兆字节计。
- 已使用 — 当前使用的空间总量的百分比。

**“Job Scheduler 服务器”选项卡**

指定为 Job Scheduler 服务器的服务器必须安装了 Job Scheduler。有关安装 Job Scheduler 的详细信息，请参见《Job Scheduler 用户指南》。

**“本地化”选项卡**

“本地化”选项卡显示缺省语言、字符集和排序顺序的当前值。您可以更改缺省值以及添加或删除语言。

## 启动集群

Adaptive Server 插件必须启用统一代理功能才能启动集群。由代理管理的集群在集群名称旁的服务器图标上显示一个红色方块（如果集群未运行）或绿色三角形（如果集群运行）。启动未运行的集群：

- 1 在树视图的左窗格中，右键单击集群快捷方式图标，然后选择“启动”。
- 2 输入具有管理访问权的 Unified Agent 登录名和口令以启动集群。
- 3 代理启动服务器时，将打开一个消息日志窗口。当集群启动过程完成时，将启用“确定”按钮。
- 4 集群图标上的红色方块变为绿色三角形，表示集群正在运行。

## 关闭集群

关闭未连接的集群：

- 1 右键单击要关闭的集群图标。
- 2 选择“关闭”。
- 3 单击“是”确认关闭。

关闭已连接的集群：

- 1 在树视图的左窗格中，右键单击集群图标，然后选择“关闭”。
- 2 选中框以：
  - 在过程完成后关闭集群，如果关闭花费的时间大于一分钟、五分钟或十分钟，您将得到通知；或者
  - 立即关闭集群。
- 3 单击“是”关闭集群。

## 删除集群

删除集群允许用户撤销创建集群时执行的所有步骤。删除集群不同于从服务器组中删除集群。您只能删除关闭的集群。

---

**注释** 当删除集群时，会将其完整删除。它不再能够重新启动。

---

删除集群：

- 右键单击要删除的集群的名称。
- 从列表中选择“删除集群”。
- 输入管理代理的登录名（通常为 `uafadmin`）。
- 选择“确定”。

## 删除服务器组

删除服务器组会从 Adaptive Server 插件中删除服务器组的集群条目。集群不受影响。

从缺省组删除集群：

- 1 右键单击集群名称，然后选择“从缺省值删除” (Remove from Default)。
- 2 通过从“确认删除”对话框中选择集群名称，然后选择“是”来确认删除。

## 显示集群的状态

如果集群由统一代理管理，则该代理提供您是否已连接的状态，但是连接的集群将显示详细信息。

在树视图中，单击“服务器实例”文件夹，以在右窗格中查看实例。状态详细信息包括：

- 实例 — 集群中实例的名称。
- ID — 集群中实例的编号顺序。
- 状态 — 集群的当前状态（联机或脱机）。
- 地址 — 集群中实例的地址。

- 开始时间 — 启动集群的时间。
- 活动连接 — 连接数量。
- 联机引擎 — 引擎数量。

## 管理集群实例

该功能启用对集群中实例的管理。

### 向集群中添加实例

在添加实例之前，`max instances` 参数必须有空间来容纳更多实例，集群支持的代理必须在创建实例时所在的节点上运行。您必须知道统一代理 (UA) 的主机名和端口号。

- 1 打开左侧窗格中的“服务器实例”文件夹，以在右窗格中显示服务器实例和选项。
- 2 选择“添加集群服务器实例”以打开“添加集群实例服务器实例”向导。
- 3 按照向导中的步骤向集群中添加实例。
- 4 单击“完成”。新实例在“服务器实例”视图下列出。
- 5 启动实例。

### 从集群中删除实例

必须先关闭实例，才能将其从集群中删除。删除实例后，将删除该实例的所有临时数据库定义，包括绑定和组成员资格。

---

**注释** 不能删除集群中剩余的最后一个实例。

---

- 1 在右窗格中，右键单击要删除的实例，然后选择“删除”。
- 2 单击“是”。实例即从集群中删除。

## 显示实例属性

实例属性对话框包含有关 Sybase 版本目录 (\$SYBASE)、\$SYBASE\_ASE 和接口目录的信息。

## 启动实例

右键单击实例图标，然后选择“启动”。将显示一个标题为“正在启动”(Start in Progress) 的栏。

实例一旦启动，状态即显示为“联机”。

---

**注释** 如果实例启动花费的时间比预计长，您可能需要手动刷新文件夹，以更新实例的状态。

---

## 关闭实例

- 1 右键单击实例图标，然后选择“关闭”。
- 2 选择是否执行下列操作：
  - 在所有过程完成后关闭实例，如果关闭花费的时间大于一分钟、五分钟或十分钟，您将得到通知。
  - 立即关闭实例。
- 3 单击“是”。当关闭完成时，将显示实例的新状态。

## 创建共享数据库设备

共享数据库设备可供所有集群实例访问。在树视图中选择“数据库设备”文件夹

- 选择“添加数据库设备”以启动“添加数据库设备”向导。
- 按照向导中的说明进行操作。
- 单击“完成”。该设备显示在右窗格中的设备列表下。

## 管理多个临时数据库

在 Cluster Edition 中，有四个不同的临时数据库：全局系统、本地系统、全局用户创建和本地用户创建。

要查看临时数据库，请选择“临时数据库”文件夹：

- 组视图 — 列出临时数据库组。只有本地临时数据库可以加入临时数据库组。
- 列表视图 — 显示全局临时数据库（而不是本地临时数据库）。

## 管理本地临时数据库

本地临时数据库只能由本地实例访问。集群中的其它服务器实例不能访问此临时数据库。

查看临时数据库：

- 1 选择“服务器实例”文件夹
- 2 选择实例的名称。
- 3 选择本地临时数据库

右键单击临时数据库名称，以配置和维护本地临时数据库。选择选项：

- 打开 Interactive SQL (Open Interactive SQL) — 使用 Interactive SQL 启动会话。
- 检查一致性 — 按照向导中的说明，对临时数据库运行数据库一致性检查程序 (dbcc)。
- 检查点 — 选择“是”在此数据库上运行检查点。选择“预览”以查看当前运行的 SQL。
- 显示统计信息 — 按照向导中的说明，对临时数据库运行 optdiag。
- 生成 DDL — 选择“创建数据库 DDL”以查看当前运行的 DDL。选择“排除 DDL” (Exclude DDL) 以查看可以从 DDL 中排除的对象的列表。选中要排除的对象，然后选择“确定”。
- 删除 — 选择“删除”以删除此临时数据库。选择“是”进行确认。由于不能删除最后一个系统临时数据库，如果这是剩余的唯一临时数据库，将禁用“删除”选项。

- 属性 — 选择适当的窗格，以获取下列信息：
  - 常规 — 描述数据库类型、数据库所有者（选择“更改”可更改数据库所有者）、创建数据库的日期、上次转储事务日志的时间、数据库是否具有 `guest` 用户、数据缓冲区的类型、缺省数据库位置以及是否重新同步代理表。
  - 设备 — 列出当前配置的数据库设备。选择：
    - 添加 — 添加设备。在“设备大小”窗口中选择“数据”或“事务日志”，然后指定设备大小。单击“确定”进行确认。
    - 删除 — 删除设备。
    - 编辑 — 配置设备。“设备大小”窗口列出当前配置的名称、大小、未使用的部分、当前分配、空间分配总计，并允许您向设备中添加空间。
    - 移动日志 — 指定日志设备的新位置。
    - 属性 — 包括四个窗格，分别介绍常规信息、镜像设备状态、位于此设备上的数据库以及段信息。
  - 用法 — 详述数据库使用的空间。选择显示信息时使用的单位：页、KB、MB 或 GB。
  - 事务日志 — 允许您配置绑定、日志 IO 大小和使用最后机会阈值的段。
  - 选项 — 列出可以为此数据库设置的数据库选项。选中或取消选中选项，然后单击“确定”。
- 活动会话 — 显示向此数据库指派的会话的 SPID 和登录信息。

## 系统临时数据库

当创建集群或实例时，创建系统临时数据库。不能删除全局或本地系统临时数据库。但是，当您删除实例时，将自动删除本地系统临时数据库。

## 添加用户创建的全局临时数据库

在 Cluster Edition 中，可以创建全局临时数据库（可用于全集群范围的所有实例）和本地临时数据库（可用于单个实例）。

- 1 导航到“数据库” | “临时数据库” | “列表视图”。
- 2 选择“添加临时数据库”向导。
- 3 按照向导中的说明操作，单击“下一步”继续下一组说明，单击“上一步”返回到上一说明。
- 4 完成时，单击“完成”。临时数据库显示在右窗格中的数据库列表下。

## 添加用户创建的本地临时数据库

用户创建的本地临时数据库只能由拥有它的实例访问：

- 1 在树视图的右窗格中，导航到“服务器实例” | “Instance\_name” | “本地临时数据库”。
- 2 选择“添加本地临时数据库”。
- 3 按照向导中的说明创建本地临时数据库。

## 向组中添加临时数据库

要查看数据库组，请从“临时数据库”文件夹选择“组视图”。

### 添加组

管理员创建包含本地临时数据库的组。缺省情况下，创建 default 组。Sybase 建议您在组上创建绑定（而不是在单个临时数据库上创建绑定），以便于管理。

从“临时数据库”文件夹选择“组视图”以查看“组”文件夹。

- 1 从树视图，导航到“数据库” | “临时数据库” | “组视图”
- 2 选择“添加临时数据库组”
- 3 Adaptive Server 插件启动“添加临时数据库组”向导。按照向导提供的说明创建本地临时数据库组

## 组属性

右键单击组名称，然后选择“属性”。Sybase Central 将显示“绑定”和“数据库”窗格。

### “数据库”窗格

“数据库”窗格在组中显示所有当前临时数据库。

向组中添加临时数据库：

- 1 选择“添加”。
- 2 从“添加临时数据库”屏幕，选择要添加的临时数据库的名称。
- 3 单击“确定”。

从组中删除临时数据库：

- 1 选择临时数据库名称。
- 2 选择“删除”。
- 3 单击“是”。

### “绑定”窗格

“绑定”窗格显示当前组的应用程序和登录绑定。

绑定新应用程序。

- 1 选择“绑定应用程序”。
- 2 输入应用程序名称，然后单击“确定”。

绑定登录。

- 1 选择“绑定登录”。
- 2 从“新建绑定登录”屏幕，选择要绑定的登录。
- 3 单击“确定”。

删除当前绑定的应用程序或登录。

- 1 选择应用程序或登录名。
- 2 选择“解除绑定”。
- 3 单击“是”进行确认。

可以将应用程序或登录绑定到临时数据库或临时数据库组。

查看当前绑定。

- 1 右键单击临时数据库或临时数据库组。
- 2 选择“绑定”选项卡。Adaptive Server 插件列出当前绑定。

解除登录或应用程序的绑定。

- 1 从列表中选择登录或应用程序。
- 2 单击“解除绑定”。
- 3 选择“是”进行确认。

解除所有登录和应用程序的绑定。

- 1 单击“解除所有绑定”。
- 2 选择“是”进行确认。

## 管理负载

使用负载管理器可以查看、创建和处理逻辑集群、负载配置文件、负载得分和路由。

## 负载配置文件

负载配置文件允许您定义逻辑集群的操作条件。这些条件通常称为“负载得分指标”，对于使用负载配置文件的逻辑集群中的每个实例，每个条件的关联值加入到“得分”中。可以定期比较逻辑集群中不同实例的负载得分，以检测负载何时意外倾斜到一个或多个实例，或者确定实例是否未得到充分利用。

包括在多个逻辑集群中的实例可能受多个负载配置文件影响，因此，当将实例与多个逻辑集群关联以及当定义和应用负载配置文件时，一定要加以小心。

---

**注释** Cluster Edition 包括两个系统负载配置文件：`sybase_profile_oltp`（针对 OLTP 环境）和 `sybase_profile_dss`（针对 DSS 环境）。不能修改或删除系统负载配置文件。但是，可以复制它们，然后修改副本以创建您自己的负载配置文件。

---

负载配置文件状态报告：

- 名称 — 负载配置文件配置的名称。
- 类型 — 负载配置文件类型：系统或用户。
- 指标权值 — 为负载配置文件中的每个指标指派的相对权值。指标包括：
  - 用户连接 — 显示连接到特定负载配置文件的用户的权值。
  - CPU 繁忙 — 显示当前繁忙的 CPU 的权值。
  - 运行队列长度 — 显示运行队列的权值。
  - IO 负载 — 显示 I/O 负载的权值。
  - 引擎差值 — 显示引擎差值的权值。
  - 用户 — 显示用户选择测量的指标的权值。
- 阈值 — 逻辑集群中两个实例之间的负载的配置差异（以百分比计），在此处可能发生下列操作：
  - 登录重定向 — 用于连接时间负载平衡和将连接路由到逻辑集群。如有必要，实例指示客户端停止当前登录尝试，并指示客户端尝试连接到它以可用网络地址列表形式提供的实例。
  - 动态迁移 —（又称为停滞值）显示动态迁移配置。
- 最小负载得分 — 触发登录重定向和动态迁移所需的负载得分。

#### 添加负载配置文件

- 1 从“工作负载管理”文件夹中选择“负载配置文件”，然后选择“添加负载配置文件”。
- 2 输入配置文件的名称。
- 3 选择“下一步”(Next)。
- 4 调整负载配置文件指标权值。

如果负载配置文件与逻辑集群关联，则负载管理器计算逻辑集群中每个实例的负载得分。它是使用为每个指标输入的权值、实例的每个指标的原始值以及负载测量算法计算得出的。请参见第 233 页的“查看负载状态”。

服务器测量的指标有：

- 用户连接 — 实例基于资源可用性接受新连接的能力。
- CPU 利用率 — 实例接受额外工作的能力。
- 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。
- I/O 负载 — 未完成的异步 I/O。
- 引擎差值 — 集群中各实例之间联机引擎数的差异。

---

**注释** 仅当集群中的实例具有数量不等的引擎时，才会测量引擎差值。引擎差值会在负载得分中添加一个衡量最大相对能力的指标。

---

- 用户指标 — 特定于用户环境的客户提供的可选指标。

确保指定的负载得分总计为 100。如果它们的总计不为 100，则负载管理器使用得分来创建总计为 100 的比例值。

5 选择“下一步”(Next)。

6 输入以下参数的值：

- 最小负载得分 — 该负载得分不是比例，而是负载管理器将工作重定向到其它实例之前所需的最小得分。在使用负载配置文件与逻辑集群中其它实例的负载得分进行比较时，最小负载得分很有意义。
- 登录重定向 (%) — 用于确定如何以最佳方式分配传入连接的负载阈值。
- 动态连接迁移 (%) — 用于确定是否分配现有连接的负载阈值。

负载阈值是当前实例上的负载与当前参与逻辑集群的最小负载实例上的负载之间的百分比差异。在 Cluster Edition 重定向登录或迁移现有连接之前，该值必须得到满足。

---

**注释** 登录重定向和动态连接迁移的百分比是独立的百分比，其总计不需要为 100。

---

7 选择“完成”以创建负载配置文件。

## 删除负载配置文件

删除负载配置文件：

- 1 从“工作负载管理” | “负载配置文件”文件夹，右键单击负载配置文件名称。
- 2 选择“删除” (Delete)。

---

**注释** 只能删除用户创建的负载配置文件。

---

## 将负载配置文件与逻辑集群关联

- 1 从“工作负载管理” | “逻辑集群”文件夹，右键单击逻辑集群名称。
- 2 选择“属性”。
- 3 选择“负载配置文件”选项卡。
- 4 单击“更改”。Adaptive Server 插件显示可用负载配置文件列表。
- 5 (可选) 选择“预览配置文件”以显示一个窗口，可以通过该窗口选择配置文件和查看它如何影响逻辑集群中实例的加权指标值。
  - a 突出显示要与此逻辑集群关联的负载配置文件。
  - b 选择“关闭”。
  - c 选择“确定”以退出“属性”对话框。
- 6 选择负载配置文件。
- 7 选择“确定”。
- 8 选择“确定”或“应用”将新的负载配置文件与逻辑集群关联。

## 负载配置文件的“常规”选项卡

查看负载配置文件属性：

- 1 从“工作负载管理”文件夹选择“负载配置文件”。
- 2 右键单击负载配置文件名称，然后选择“属性”。

“常规”选项卡描述负载配置文件，包括其名称和类型（它是系统负载配置文件还是用户负载配置文件）。

请参见第 223 页的“负载配置文件”。

## “指标权值”选项卡

“指标权值”选项卡描述应用于测量指标的当前权值。如果权值总计不为 100，则负载管理器使用比例，将值调整成总计为 100。

- 用户连接 — 实例基于资源可用性接受新连接的能力。
- CPU 利用率 — 实例接受额外工作的能力。
- 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。
- I/O 负载 — 未完成的异步 I/O。
- 引擎差值 — 集群中各实例之间联机引擎数的差异。

---

**注释** 仅当集群中的实例具有数量不等的引擎时，才会测量引擎差值。引擎差值会在负载得分中添加一个衡量最大相对能力的指标。

---

- 用户指标 — 特定于用户环境的客户提供的可选指标。  
确保指定的负载得分总计为 100。

## “阈值”选项卡

使用“阈值”选项卡查看和更改负载配置文件的阈值设置。输入以下参数的值：

- 最小负载得分 — 该负载得分不是比例，而是负载管理器将工作重定向到其它实例之前所需的最小得分。在使用负载配置文件与逻辑集群中其它实例的负载得分进行比较时，最小负载得分很有意义。
- 登录重定向 (%) — 用于确定如何以最佳方式分配传入连接的负载阈值。
- 动态连接迁移 (%) — 用于确定是否分配现有连接的负载阈值。

## 管理逻辑集群

“逻辑集群”文件夹包含已定义逻辑集群的列表，还包含帮助您添加新逻辑集群的向导。右窗格显示下列详细信息：

- 集群 ID — 显示每个逻辑集群的关联 ID 号。
- 逻辑集群的当前状态 — 逻辑集群的连接状态（联机或脱机）。
- 连接 — 逻辑集群中的活动连接数量。
- 基本实例 — 逻辑集群中基本实例的数量。
- 活动基本实例 — 显示逻辑集群中当前活动的基本实例的数量。
- 故障切换实例 — 为故障切换配置的实例的数量。
- 活动故障切换实例 — 显示活动故障切换实例的数量。
- 向下路由模式 — 显示向下路由配置。以下状态之一：
  - **system** — 将不可路由的连接发送到系统逻辑集群。由于系统逻辑集群对于每个实例均始终处于联机状态，因此 **system** 可确保最高的可用性。这是缺省设置。
  - **open** — 将不可路由的连接发送到开放式逻辑集群。如果无法将连接发送到开放式逻辑集群，或者连接不支持重定向，则实例将应用开放式逻辑集群的向下路由模式。
  - **disconnect** — 断开不可路由的连接。通过断开目标逻辑集群中的实例无法提供服务的客户端的连接，此设置可强制实现资源保留。
- 故障切换模式 — 显示故障切换模式配置。选项为实例或组。
- 启动模式 — 显示启动模式配置：自动（该选项处于未选中状态）或手动。
- 系统视图 — 显示系统视图配置（实例或集群）。
- 角色 — 只有系统创建的逻辑集群具有系统角色。缺省情况下，开放角色指派给系统逻辑集群。当开放角色被指派给新逻辑集群时，该开放角色从以前拥有该值的逻辑集群中删除。
- 负载配置文件 — 显示用于集群的负载配置文件的名称。

## 添加逻辑集群

- 1 右键单击“添加逻辑集群”。
- 2 选择“打开”。
- 3 输入集群名称，然后单击“下一步”。
- 4 单击“添加”选择要加入逻辑集群中的服务器实例。单击“下一步”。
- 5 为逻辑集群添加故障切换服务器实例。单击“下一步”。
- 6 输入路由的应用程序、登录名和别名。您还可以先选择路由来删除路由。
- 7 要选择除缺省负载配置文件之外的负载配置文件，请单击“更改”，然后单击“下一步”。
- 8 设置新逻辑集群的选项。单击“下一步”。
- 9 Adaptive Server 插件显示了摘要。要进行更改，请使用“上一步”按钮。完成时，单击“完成”。

## 删除逻辑集群

要删除逻辑集群，它必须处于脱机状态。

- 1 右键单击逻辑集群，然后选择“删除”。
- 2 单击“是”。

## 逻辑集群属性

右键单击逻辑集群名称，然后选择“属性”以查看逻辑集群的配置。

请参见第 223 页的“负载配置文件”。

## “常规”选项卡

“常规”选项卡中显示了下列选项：

- 系统视图 — 确定逻辑集群用户查看 Cluster Edition 的方式（是以整个集群还是以单个实例的形式）。这将影响一些查询和存储过程。  
选择“实例”或“集群”。
- 自动启动逻辑集群 — 如果希望此逻辑集群在集群启动时启动，则选择此选项。
- 故障切换模式 — 确定逻辑集群是故障切换到另一个实例还是故障切换到一个组：
  - 实例 — 逻辑集群一次故障切换一个实例。
  - 组 — 指定只有当所有基本实例均失败时才替换基本实例，并且所有故障切换实例随后将进入联机状态。例如“SalesLC”的故障切换模式为“group”。如果基本实例“ase1”失败，则集群将继续在基本实例“ase2”上运行。没有故障切换实例将进入联机状态。但是，如果“ase1”和“ase2”均失败，则集群将在故障切换实例“ase3”和“ase4”上运行。
  - 任何实例都提供故障切换支持 — 允许逻辑集群即使在没有可用故障切换实例时也进行故障切换。逻辑集群根据负载，故障切换到集群中的任何可用物理实例。  
  
“任何实例都提供故障切换支持”将逻辑集群配置为故障转移到任何可用实例，即使它未定义为该逻辑集群中的故障切换实例。
- 向下路由模式 — 显示向下路由配置。以下状态之一：
  - **system** — 将不可路由的连接发送到系统逻辑集群。由于系统逻辑集群对于每个实例均始终处于联机状态，因此 **system** 可确保最高的可用性。这是缺省设置。
  - **open** — 将不可路由的连接发送到开放式逻辑集群。如果无法将连接发送到开放式逻辑集群，或者连接不支持重定向，则实例将应用开放式逻辑集群的向下路由模式。
  - **disconnect** — 断开不可路由的连接。通过断开目标逻辑集群中的实例无法提供服务的客户端的连接，此设置可强制实现资源保留。
- 逻辑集群角色 — 系统角色自动设置为“系统逻辑集群”，您不能更改此设置。

缺省情况下，开放角色指派给系统逻辑集群。您不能授予开放角色，只有当另一个逻辑集群假设该角色时才能将其删除。

## “基本实例”选项卡

“基本实例”选项卡列出当前配置的逻辑集群实例。

- 选择“添加”以向逻辑集群中添加实例。
- 从逻辑集群删除实例：
  - a 突出显示要删除的实例。此实例必须处于脱机状态，然后才能选择它。
  - b 选择“删除”。
  - c 选择“是”确认删除。
- 选择“脱机”使实例脱机。可以指定使实例立即脱机；也可以指定使其逐步脱机，但在指定的时间段后提醒您。
- 选择“联机”使脱机的实例联机。
- 选择“故障切换”以从此实例故障切换到另一个实例。

## 向逻辑集群中添加实例

向逻辑集群中添加实例：

- 1 单击“添加”。
- 2 突出显示要添加的实例
- 3 单击“确定”。直到您单击“应用”或“确定”后，这些更改才会生效。

## “故障切换实例”选项卡

列出有关当前配置的故障切换实例的信息，包括：

- 名称
- ID
- 状态
- 故障切换组

## 添加故障切换实例

添加故障切换实例：

- 1 单击“添加”。
- 2 从“添加故障切换实例到逻辑集群”窗口选择要配置的故障切换组。  
故障切换组允许您指定故障切换实例在故障切换时变为活动的顺序。  
组可以有一个或多个实例。
- 3 在此列表中突出显示要配置为故障切换实例的实例，然后单击“确定”

## “负载配置文件”选项卡

列出有关与逻辑集群关联的负载配置文件的信息：

- 名称 — 与此逻辑集群关联的负载配置文件的名称。单击“更改”可将另一个负载配置文件与此逻辑集群关联。
- 类型 — 显示负载配置文件类型：系统或用户。
- 最小负载得分 — 用于激活登录重定向或迁移的最小负载得分。
- 指标 — 包含有关负载配置文件的各种统计信息，包括
  - 用户连接 — 实例基于资源可用性接受新连接的能力。
  - CPU 利用率 — 实例接受额外工作的能力。
  - 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。
  - I/O 负载 — 未完成的异步 I/O。
  - 引擎差值 — 集群中各实例之间联机引擎数的差异。
  - 用户指标 — 特定于用户环境的客户提供的可选指标。
- 权值 — 指示指标对于负载得分的重要性。这是相对测量值，且上限为 100。权值为 0 的指标没有值，在计算负载得分时不使用；权值为 50 的指标对负载得分有一半的影响。如果 5 个指标的权值都为 20，则在计算负载得分时每个指标的重要性是等同的。
- 阈值 — 当前配置的与此负载配置文件关联的阈值。

## “路由”选项卡

此选项卡描述指派给此逻辑集群的应用程序、登录名和别名的路由。

### 添加应用程序路由

- 1 单击“添加应用程序路由”。
- 2 输入应用程序名称，然后单击“确定”。

### 添加登录路由

- 1 单击“添加登录路由”。
- 2 选择路由的登录名（或多个登录名），然后单击“确定”。

### 添加别名路由

- 1 单击“添加别名路由”。
- 2 输入别名名称。
- 3 单击“确定”。

### 删除路由

- 1 选择路由。
- 2 单击“删除路由”。
- 3 选择“是”确认删除。

## 查看负载状态

“工作负载”文件夹在下列两个选项卡上显示了负载指标的全集群范围视图：“加权得分”(Weighted Scores)和“基本指标值”。

从树视图中的“工作负载管理”文件夹选择“工作负载”。负载得分报告加权得分和基本得分。

### “加权得分”(Weighted Scores)选项卡

每个实例跟踪一组负载指标。将为每个实例和逻辑集群的组合计算负载得分和指标，并通过将逻辑集群的负载配置文件应用于实例的负载统计信息来确定负载得分和指标。结果将是总体负载得分和一组加权得分，这些得分表示了对特定实例属性的相对影响。

“加权得分” (Weighted Scores) 显示每个实例和逻辑集群组合的负载得分和加权指标值。如果某实例与两个逻辑集群关联，则详细信息选项卡中将存在该实例的两个条目。

“加权得分” (Weighted Scores) 选项卡包括：

- 实例 — 表示其负载的实例的名称。
- 逻辑集群 — 与实例关联的逻辑集群的名称。
- 负载配置文件 — 指派给逻辑集群的负载配置文件。
- 负载得分 — 表示实例上的总体负载的计算值。可以将此无单位数字在整个实例中进行比较，以作为比较负载的手段。
- 用户连接 — 实例基于资源可用性接受新连接的能力。
- CPU 繁忙 — 对引擎繁忙程度进行的测量，它提供了与 `sp_sysmon` 同样的信息。用于确定实例接受额外工作的能力。
- 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。
- IO 负载 — 测量未完成的异步 IO，它指示了实例之间的相对 IO 饱和。
- 引擎差值 — 测量实例之间联机引擎数的差异。在所有实例的引擎数都相同的集群中，没有引擎差异。但是，例如在由两个实例组成的集群中，“实例 1”有四个引擎，“实例 2”有两个引擎，“实例 1”没有引擎差异，但是“实例 2”有 50% 的差异，因为它的引擎数为“实例 1”的一半。
- 用户 — 您在负载配置文件中指定的指标的加权值。

---

**注释** 由于每个实例可以包括在多个逻辑集群中，每个实例对于它所属的逻辑集群，都有一组指标值。

---

“基本指标值”选项卡

“基本指标值”选项卡显示集群中每个实例的所有负载值。由于每个实例只有一组值，无论它加入了多少个逻辑集群，都为每个实例显示一组值。

- 实例 — 实例的名称。
- 用户连接百分比 (% User Connections) — 正在使用的已配置用户连接的百分比。
- CPU 繁忙百分比 (% CPU busy) — 实例忙于执行任务的时间的百分比。该值为一分钟，是从系统上所有引擎汇总的移动平均值。
- 运行队列长度百分比 (% Run queue length) — 系统上可运行任务的基本百分比。运行队列长度衡量处理积压，可以很好地指示相对响应时间。

- IO 负载百分比 (% IO load) — 未完成异步 IO 的基本百分比。
- 引擎差值百分比 (% Engine deficit) — 集群中各实例之间联机引擎数的差异基本百分比。
- 用户百分比 (% User) — 为负载配置文件中指定的用户指标提供的百分比值。

## 查看实例的负载状态

负载状态基于与实例关联的负载配置文件。该状态显示原始值，以指示实例相对于每个指标区域的负载程度。

查看负载状态：

- 1 单击“服务器实例”选项卡以在右窗格中显示实例列表。
- 2 右键单击实例名称，然后选择“工作负载状态”。负载状态显示原始值，这些值指示了每个指标区域的每个实例执行的工作量。
  - 用户连接 — 实例基于资源可用性接受新连接的能力。
  - CPU 繁忙 — 对引擎繁忙程度进行的测量，它提供了与 `sp_sysmon` 同样的信息。用于确定实例接受额外工作的能力。该值测量为一分钟移动平均值。
  - 运行队列长度 — 系统上可运行任务的数量。运行队列长度衡量处理积压，可以很好地指示相对响应时间。该值测量为一分钟移动平均值。
  - IO 负载 — 测量未完成的异步 I/O，它指示了实例之间的相对 IO 饱和。该值测量为一分钟移动平均值。
  - 引擎差值 — 测量实例之间联机引擎数的差异。在所有实例的引擎数都相同的集群中，没有引擎差异。但是，例如在由两个实例组成的集群中，“实例 1”有四个引擎，“实例 2”有两个引擎，“实例 1”没有引擎差异，但是“实例 2”有 50% 的差异，因为它的引擎数为“实例 1”的一半。
  - 用户 — 您使用 `workload_metric` 函数指定的指标的加权值。

---

**注释** 由于每个实例可以包括在多个逻辑集群中，每个实例对于它所属的逻辑集群，都有一组指标值。

---

## 管理路由

路由允许您将客户端连接定向到特定逻辑集群。

### 路由属性

查看当前路由：

- 1 从“工作负载管理” | “逻辑集群”文件夹，右键单击逻辑集群名称。
- 2 选择“属性”。
- 3 选择“路由”选项卡。

“路由常规信息” (Routes General) 选项卡报告：

- 路由的名称。
- 路由类型 — 显示路由的类型：应用程序、登录名或别名。
- 逻辑集群 — 此路由关联的逻辑集群的名称。

要将此路由与另一个逻辑集群关联，请选择另一个逻辑集群的名称，然后单击“确定”。

### 创建路由

使用“创建路由向导”创建路由。

启动该向导：

- 从“工作负载管理”文件夹选择“路由”。
- 双击“添加路由”。
- 按照向导说明添加路由。

## 用 sybcluster 管理集群

sybcluster 是一种交互式实用程序，能让您从命令行执行集群的管理任务。

主题	页码
<a href="#">使用 sybcluster</a>	238
<a href="#">sybcluster 和 Unified Agent Framework</a>	240
<a href="#">启动 sybcluster</a>	240
<a href="#">创建集群</a>	240
<a href="#">连接到集群</a>	241
<a href="#">启动集群</a>	244
<a href="#">管理集群</a>	244
<a href="#">管理实例</a>	250
<a href="#">在手动创建集群后启用 sybcluster</a>	254
<a href="#">创建和管理辅助服务器</a>	255
<a href="#">升级服务器</a>	257

可以使用 sybcluster 来创建集群，添加实例，启动和停止集群或实例，显示状态信息，等等。也可以使用适用于 Sybase Central 的 Adaptive Server 插件，按照第 14 章“使用 Adaptive Server 插件管理集群”中的说明来执行这些任务。

不能使用 sybcluster 来管理临时数据库或逻辑集群。

- 要管理临时数据库，请使用 Adaptive Server 插件或按照第 8 章“使用临时数据库”中的说明执行操作。
- 要管理逻辑集群，请使用 Adaptive Server 插件或 sp\_cluster logical 和 sp\_cluster profile 存储过程（如第 6 章“管理负载”中所述）。

有关 sybcluster 命令的完整语法和用法信息，请参见《实用程序指南》。

## 使用 sybcluster

有些 sybcluster 命令在连接到集群之前有效，其它命令则只有在连接到集群后才有效。请参见表 15-1。

sybcluster 提示能让您知道 sybcluster 是否与集群连接，以及缺省集群和实例（如果已经设置的话）。

- 当 sybcluster 未连接到任何集群时，提示为：

```
>
```

- 在连接到集群后，提示变为：

```
cluster_name>
```

例如：

```
mycluster>
```

- 如果您声明了缺省实例，则提示为：

```
cluster_name instance_name>
```

例如：

```
mycluster asel>
```

**表 15-1: sybcluster 交互式命令**

命令名	说明	在连接到集群之前还是之后有效
add backupserver	在当前还未进行 Backup Server 配置的集群中的节点上配置一个或多个新 Backup Server。	之后
add instance	向集群中添加一个新实例。	之后
connect	连接到一个现有集群。	之前
create backupserver	创建 Backup Server。	之后
create cluster	创建一个新的集群。	之前
create xpserver	创建 XP Server。	之后
deploy plugin	将集群的单个实例的配置信息配置到 Unified Agent 中。	之前
diagnose { cluster   instance }	执行一组检查以确保集群或实例正常运行。	之后
disconnect	关闭与当前集群的所有连接并使集群返回到未连接状态。	之后
drop backupserver	删除 Backup Server。	之后
drop cluster	删除集群中的每个实例，并删除集群配置文件中的集群定义。	之后
drop xpserver	删除 XP Server。	之后
drop instance	删除集群中的一个实例。	之后

命令名	说明	在连接到集群之前还是之后有效
exit	退出 sybcluster。	之前或之后
help	列出当前可用的 sybcluster 交互式命令。	之前或之后
localize	显示缺省语言、字符集和排序顺序的当前值。能让您更改缺省值以及添加或删除语言。	之后
quit	退出 sybcluster。	之前或之后
set backupserver	更改一个或多个节点上的 Backup Server 监听端口号。	之后
set cluster	设置集群的属性。	之后
set instance	设置实例的属性。	之后
set xpserver	更改一个或多个节点上的 XP Server 监听端口号。	之后
show agents	显示有关可用 UAF 代理的信息。	之前
show backupserver config	显示配置 Backup Server 的节点的名称以及关联的监听端口号。	之后
show cluster	显示集群的配置、日志和状态值。	之后
show instance	显示有关实例的信息。	之后
show membership mode	显示集群的当前成员资格模式，该模式指定集群是否支持 Veritas Cluster Server 集成。	之后
show session	显示当前代理和发现信息。	之后
show xpserver config	显示配置 XP Server 的实例和节点的名称以及关联的监听端口号。	之后
shutdown cluster	通过对集群中的每个实例执行一个 Transact-SQL shutdown 命令，来关闭集群。	之后
shutdown instance	通过执行 Transact-SQL shutdown 命令，来关闭实例。	之后
start cluster	启动集群中的所有实例。	之后
start instance	启动集群中的一个实例。	之后
upgrade server	将 Adaptive Server 更新为 Adaptive Server Cluster Edition。	之前
use	设置缺省实例。	之后

## sybcluster 和 Unified Agent Framework

sybcluster 使用 Unified Agent Framework (UAF) 来提供远程管理功能，以运行时服务的形式来管理分布式 Sybase 资源。它提供一组通用服务，并启用 sybcluster 和 Adaptive Server 插件之类的 Sybase 进程，从而连接到一个代理来管理服务器资源和执行各种操作。

Unified Agent 服务器可以使用 UDP 在子网上广播自己，或向 Jini 或 LDAP 之类的查找服务器注册自己。

有关 UAF 的详细信息，请参见 Users Guide to the Unified Agent and Management Console（《Unified Agent 和 Management Console 用户指南》）。

## 启动 sybcluster

从命令行启动 sybcluster。最简单的方法是输入：

```
sybcluster -U uafadmin -P
```

“uafadmin”是缺省用户名，缺省口令是 Null 或空白。还可以使用经过鉴定的 Adaptive Server 数据库用户名和操作系统用户名。请参见第 241 页的“鉴定用户”。

还可以在启动 sybcluster 的同时指定集群，标识缺省实例，连接到集群中一个或多个节点上的 Unified Agent。例如：

```
sybcluster -U uafadmin -P -C mycluster -I asel  
-F "blade1:9999,blade2:9999,blade3:9999"
```

在本例中，-F 选项标识集群中每个 Unified Agent 的节点和监听端口。如果您经常使用此命令，可以为它创建一个简单的别名。例如：

```
sybcluster_mycluster
```

## 创建集群

有关使用 sybcluster 创建集群的说明，请参见所用平台的安装指南。

## 连接到集群

可以在启动 `sybcluster` 时通过指定所需的信息来连接到集群；也可以在启动 `sybcluster` 后再使用 `connect` 命令连接到集群。只有连接到集群后才能启动集群。

当您安装和配置集群时，还会在集群中的每个节点上安装和配置一个 Unified Agent。当您启动 `sybcluster` 时，必须提供经过鉴定的用户名和口令，并使用直接连接或发现方法在集群的一个或多个节点上标识 Unified Agent。然后，`sybcluster` 可以连接到 Unified Agent 并执行命令、控制和发现操作。

## 鉴定用户

可以使用任何 UAF 鉴定的用户名和口令来连接到集群。缺省情况下，用户名为“`uafadmin`”，缺省口令是 `Null` 或空白字符串。例如：

```
sybcluster -U uafadmin -P -C mycluster
```

## 设置用户名和口令

Sybase 建议您更改缺省用户名和口令，并对新口令进行加密。UAF 的用户名和口令信息存储在集群中每个节点上的 `csi.properties` 文件中。

### ❖ 设置用户名

- 在每个 `$SYBASE/UAF-2_5/nodes/<node_name>/conf/csi.properties` 文件的 Simple Log Module 部分的“`username`”属性中输入新的用户名。例如：

```
# Simple Login Module
...
CSI.loginModule.2.options.username=newusername
CSI.loginModule.2.options.password=
CSI.loginModule.2.options.encrypted=false
CSI.loginModule.2.options.roles=uaAgentAdmin,uaPluginAdmin
```

---

**注释** 确保在集群中的每个节点上都编辑 `csi.properties` 文件。

---

❖ **加密和设置口令**

- 1 运行 `passencrypt`（位于 `$$SYBASE/UAF-2_5/UAF-2_5/bin` 中），从而生成加密口令。
- 2 在每个 `$$SYBASE/UAF-2_5/nodes/<node_name>/conf/csi.properties` 文件的 Simple Log Module 部分：
  - 1 将加密值粘贴到口令属性中。
  - 2 将加密属性设置为“true”。例如：

```
# Simple Login Module
...
CSI.loginModule.2.options.username=newusername
CSI.loginModule.2.options.password=REVTelNZVUFGfWNvbS5zdW4uY3J5cHRvLnByb3ZpZGVyLlN1bkpDRXtTWVVRn1nTUJacVh5R3pnN09RSDJDN1NPUXhBPT0=
CSI.loginModule.2.options.encrypted=true
CSI.loginModule.2.options.roles=uaAgentAdmin,uaPluginAdmin
```

---

**注释** 确保在集群中的每个节点上都编辑 `csi.properties` 文件。

---

❖ **激活新的用户名和口令**

- 要使新用户名和口令生效，请在集群中的每个节点上关闭并重新启动 Unified Agent。

## 标识 Unified Agent

可以使用直接连接或发现方法来标识 Unified Agent，这可以在启动 `sybcluster` 时进行，也可以在启动后，使用 `connect to` 交互式命令进行。

---

**注释** 如果当您与集群连接时集群已在运行，则通常仅需在集群中的一个节点上标识一个 Unified Agent 即可。但若启动集群或 XP Server，则必须在集群中的每个节点上都标识代理。

如果需要提供额外的信息，`sybcluster` 会提示您。

---

如果您不知道集群的 Unified Agent 指定情况，请使用 `sybcluster show agents` 命令来查找子网上的可用 Unified Agent 和集群。

## 使用直接连接方法

可以直接与一个或多个代理连接，方法是负责管理集群的 Unified Agent 指定集群节点和端口号。下面是一些可能的代理指定：

- 集群的节点名称（一个或多个）和可选端口号 — 能让您精确地指定地址。如果不包括端口号，**sybcluster** 就会采用缺省值 9999。例如，要在 “mycluster” 的 “blade1”、“blade2” 和 “blade3” 上指定代理，请输入：

```
sybcluster -U uafadmin -P -C mycluster
-F "blade1:1234,blade2:2345,blade3:3456"
```

要在启动 **sybcluster** 后，通过使用 **connect** 来指定节点和端口号，请输入：

```
connect to mycluster -U uafadmin -P
-F "blade1:1234,blade2:2345,blade3:3456"
```

- 节点的域 — 能让您通过域名精确地指定地址。例如：

```
sybcluster -U uafadmin -P -C mycluster
-F "blade1.mydomain.com"
```

## 使用发现方法

**sybcluster** 支持以下三种用于确定代理位置的发现方法以及发现顺序：

- 用户数据报协议 (UDP) — 是一种广播发现方法，其工作原理是：**sybcluster** 将请求广播出去，位于同一子网上的代理进行响应。例如，要使用 UDP 来查找 “mycluster” 中某个节点的位置，请输入：

```
sybcluster -U uafadmin -P -C mycluster
-d "udp()"
```

要在启动 **sybcluster** 后，通过使用 **connect** 来执行发现，请输入：

```
connect to mycluster login uafadmin password " "
discovery "udp(),
jini(myjinihost1:5678;myjinihost2:1234)"
```

- Jini 服务器技术 — 提供查找功能。每个代理都向 Jini 服务器注册，该服务器中存储每个节点的位置以及状态信息。要在 “mycluster” 中查找某个代理的位置，请输入：

```
sybcluster -U uafadmin -P -C mycluster
-d "jini(myjiniserver:4564)"
```

要在启动 **sybcluster** 后，通过使用 **connect** 来执行发现，请输入：

```
connect to mycluster discovery
"jini(myjiniserver:4564)"
```

- LDAP 技术 — 提供查找功能。每个代理都向 LDAP 服务器注册，该服务器中存储每个节点的位置以及状态信息。

要使用这三种发现方法来查找 “mycluster” 中 Unified Agent 的位置，请输入：

```
sybcluster -U uafadmin -P -C mycluster  
-d "udp(),jini(myjiniserver:4123),  
ldap(myldapservers:6123)
```

## 启动集群

在启动集群前，必须先与目标集群中每个节点上的 Unified Agent 连接。请参见第 241 页的“[连接到集群](#)”。然后，要启动集群，请输入：

```
start cluster
```

`sybcluster` 会在启动集群中的所有实例时打印集群说明。

## 管理集群

本节介绍如何执行任务来帮助您管理集群及其环境。

## 创建集群

可以使用 `sybcluster` 或 Adaptive Server 插件来创建集群。这两种方法都会提示您提供所需信息，然后为您创建集群。您还可以手动创建集群，这需要您自己执行所有任务。

请参见《安装指南》中有关使用上述每种方法创建集群的分步说明。

## 检验集群

`diagnose cluster` 会执行一组检查以确保集群正常运行。请输入：

```
diagnose cluster
```

`diagnose cluster` 会显示集群信息并检查：

- Unified Agent 是否在集群中的所有节点上运行。
- 集群中的节点数是否未超过集群中的最大实例数。
- 是否存在仲裁设备，如果不存在，请检查目录是否具有写权限。
- 是否存在包含所有节点的 `interfaces` 文件，而且节点名和端口号是否不冲突。
- 主协议和辅助协议规范是否不重叠。
- 每个节点上的 Sybase 主目录是否都共享。

## 显示有关可用 Unified Agent 的信息

使用 `show agents` 可标识子网上配置的所有 Unified Agent，也可缩小搜索范围以显示有关特定 Unified Agent 的信息。

例如，要标识所有 Unified Agent，请输入：

```
show agents
```

`sybcluster` 会显示每个 Unified Agent 的直接连接地址、其节点和集群名称，以及其它相关信息。

可以通过限制发现或标识所需代理来显示有关特定 Unified Agent 的信息。例如，要查看有关 “mycluster” 中 “blade2” 上的 Unified Agent 的信息，请输入：

```
show agents agent "blade2:9999"
```

## 显示集群信息

本节介绍如何使用 `sybcluster` 来显示有关集群、集群中的实例以及集群环境的信息。有关完整的语法和用法信息，请参见《实用程序指南》。

可以使用 `Adaptive Server` 插件来执行相同的任务。请参见第 14 章“使用 `Adaptive Server` 插件管理集群”。

- 要显示配置信息、主协议和辅助协议的值、所用的跟踪标志，以及仲裁设备和主设备的地址，请输入：

```
show cluster config
```

要查看带格式的集群配置信息，请输入：

```
show cluster config template
```

- 要显示集群中每个实例的状态和心跳信息，请输入：

```
show cluster status
```

状态值有：

- Up（已启动）
  - Down（已关闭）
  - Undefined（未定义）
  - Invalid（无效）
  - Start（启动）
  - Init（初始化）
  - Quiesce（停顿）
- 要显示所有日志信息，请输入：

```
show cluster log
```

可以通过指定如下内容来限制输出：

- 错误严重性，例如：

```
show cluster log minseverity 5
```

- 日志条目的日期范围：

```
show cluster log startdate 03:31:08  
enddate 04:30:08
```

- 显示错误日志中的多行（从最近的开始往后）：

```
show cluster log last 25
```

- 要显示与集群的所有 UAF 和 JDBC 连接，请输入：  

```
show cluster connection
```
- 要显示有关集群的常规信息以及有关代理连接的详细信息，请输入：  

```
show session
```
- Symantec Veritas Cluster Server (VCS) 管理集群环境中的应用程序服务。如果您的站点支持 VCS，则可以在集群上启用 VCS。请参见 Veritas 文档。要查明集群上是否启用了 VCS，请显示成员资格模式。请输入：  

```
show membership mode
```

成员资格模式值有：

  - vcs — 集群支持 VCS 集成。
  - native — 集群不支持 VCS。

请参见第 249 页的“更改成员资格模式”。

## 更改集群配置值

可以使用 `set cluster` 以及其它命令来更改集群的某些配置值。通过发出 `show cluster status` 命令来检验集群的状态。

当集群已关闭时，您可以更改：

- 最大实例数
- 处于活动状态的跟踪标志
- 主协议或辅助协议
- 成员资格模式

当集群在运行时，您可以更改：

- Unified Agent 用来登录到集群的登录名或口令
- 集群的缺省语言、字符集和排序顺序

例如，要将“mycluster”的最大实例数更改为 4，请输入：

```
set cluster maxinst 4
```

要将集群的主协议重新设置为“udp”，请输入：

```
set cluster primary protocol udp
```

## 更改用户名或口令

`sybcluster` 是一种客户端程序，与被配置以用来运行集群的 Unified Agent 连接。当启动 `sybcluster` 时，您需要提供登录名和口令，才能让 `sybcluster` 登录到 Unified Agent。要更改 Unified Agent 登录名或口令的值，请使用 Agent Management Console Sybase Central 插件。要对口令进行加密，请参见第 241 页的“设置用户名和口令”。

对于某些操作，Unified Agent 必须登录到集群。当 `sybcluster` 或 Sybase Central 发出 `shutdown` 命令时或者当 Unified Agent 执行集群心跳任务来确定集群状态时，就会发生这种情况。对于这些任务，Unified Agent 必须使用具有 `sa_role` 的登录名。缺省情况下，Unified Agent 使用登录名“`sa`”，该登录名没有口令。要更改该口令，请使用 `sybcluster set cluster login` 命令。

例如，要将登录名“`sa`”的口令更改为“`newpassword`”，请输入：

```
set cluster login sa password newpassword
```

只有在集群在运行时，才能执行此命令。

有关完整的语法和用法信息，请参见《实用程序指南》。

## 更改本地化值

使用 `sybcluster localize` 命令可查看语言、字符集和排序顺序的当前值。显示当前缺省值后，`sybcluster localize` 会提示您接受每个值或更改它们。例如，要查看当前值而不更改它们，请输入：

```
localize
```

```
Current default locale properties are:
```

```
Default Language - portuguese
```

```
Default Charset - mac
```

```
Default SortOrder - Binary ordering, for use with the  
Macintosh character set(mac).
```

```
Options for default Language are:
```

1. spanish
2. portuguese
3. german
4. us\_english
5. thai
6. french
7. japanese
8. chinese
9. korean

```
10. polish
Enter the number representing the language to be set as
defaults: [2]

Options for default charsets are:
1. gb18030
2. eucgb
3. uttf8
Enter the number representing the charset to be set as
default: [1]

Options for sort orders are:
1. Binary ordering, for the EUC GB2312-80 character set
(eucgb).
Enter the number representing the sort order to be set
as default [1]

Do you want to install any language? [Y] n
Do you want to remove any language? [N ]
The cluster mycluster was successfully localized with
default language portuguese, charset gb18030, sortorder
bin_eucgb.
```

为确保整个集群的一致性，请在更改本地化值后关闭并重新启动集群。

## 更改成员资格模式

如果您的站点支持 VCS，则可以在创建集群时启用 VCS。可以使用 `qrmutil` 或 Veritas 实用程序来更改成员资格模式。请参见《实用程序指南》和 Veritas 文档。

## 断开与集群的连接

要关闭所有与当前集群的连接，请输入：

```
disconnect
```

## 关闭集群

可以正常关闭集群，这样便可在 Adaptive Server 按照集群配置文件中指定的顺序关闭每个实例以前，先完成事务。

```
shutdown cluster
```

要立即关闭集群而不等待事务完成，请输入：

```
shutdown cluster nowait
```

---

**注释** 如果集群在 Veritas Cluster Server (VCS) 模式下运行，请确保使用 VCS 关闭和启动机制来关闭或启动服务器和集群。不要使用 `sybcluster shutdown` 命令。

---

## 删除集群

在删除集群前，应先确保集群的状态为 Down（已关闭），而且 Unified Agent 在运行。然后，输入：

```
drop cluster
```

Adaptive Server 会从 `interfaces` 文件中删除集群和实例条目，删除集群配置文件，将仲裁磁盘标记为未使用，删除日志文件，关闭并删除集群的 Unified Agent 插件。对每个删除都必须进行确认。

## 管理实例

本节介绍如何执行任务来帮助您管理集群中的实例。

## 显示有关实例的信息

与 `show cluster` 类似，`show instance` 可显示有关实例的配置、状态和日志信息。

- 要显示配置信息（包括主机节点的名称、主网络与辅助网络信息，以及日志文件的路径），请输入：

```
show instance instance_name config
```

- 要显示状态信息，请输入：

```
show instance instance_name status
```

此命令可显示命名实例的状态信息：

- Up（已启动）
  - Down（已关闭）
  - Undefined（未定义）
  - Invalid（无效）
  - Start（启动）
  - Init（初始化）
  - Quiesce（停顿）
- 要显示所有日志信息，请输入：

```
show instance instance_name log
```

可以通过指定如下内容来限制输出：

- 错误严重性，例如：

```
show instance instance_name log minseverity 5
```

- 日志条目的日期范围：

```
show instance instance_name log startdate  
03:31:08 enddate 04:30:08
```

- 显示错误日志中的多行（从最近的开始往后）：

```
show instance instance_name log last 25
```

## 添加实例

可以采用两种方式向集群中添加实例，一是交互式，由 `sybcluster` 提示您输入所需信息，二是使用输入文件。

---

**注释** `add instance` 会为新实例创建一个本地系统临时数据库。请确保设备上有足够的空间。

---

如果以交互方式添加实例，`Adaptive Server` 会提示您输入：

- 实例名称（如果未在命令语句中指定实例名称）
- 承载实例的节点的名称
- 节点上 `Unified Agent` 的端口号
- 查询端口号
- 节点的主地址和辅助地址
- 主端口和辅助端口指定

如果是使用输入文件来添加实例，请确保文件与集群输入文件的格式形成镜像（请参见所用平台的安装指南），但仅需包含新实例的定义即可。如果定义辅助服务器，则应包括端口以及为新实例配置辅助服务器所需的其它信息。

例如，要使用输入文件来添加实例，请输入：

```
add instance new_instance file /$SYBASE/myfile
```

要以交互方式添加实例，请输入：

```
add instance new_instance
```

## 检验实例

`diagnose instance` 会执行一组检查以确保实例配置得当。例如，要检验“`ase1`”的配置，请输入：

```
diagnose instance ase1
```

`diagnose instance` 会显示实例的配置信息并检验：

- 查询端口
- 是否有 `JDBC` 连接
- 实例是否在公共网络上可用

- 最小和最大端口数
- 主协议与辅助协议的端口范围

## 更改缺省实例

使用 `use` 命令可设置或更改在 `sybcluster` 命令行中指定的缺省实例。设置缺省实例后，就不必在任何交互式命令的命令行中进行指定了。例如，可输入：

```
use ase1
```

可以通过在交互式命令中包含实例名称来替换缺省实例。但这样做不会更改缺省指定。

要删除缺省指定，请省略实例名称。请输入：

```
use
```

## 更改实例属性

可以使用 `set instance` 来更改实例的某些属性。只有在实例处于 Down（已关闭）状态时，才能使用 `set instance`。通过发出 `show cluster status` 来检验状态。

可以更改的实例属性有：

- 日志路径
- 用于启动实例的参数
- 实例的主地址或辅助地址
- 实例所用的主端口或辅助端口的范围

例如，要重新设置主端口范围 6123 到 6126，请输入：

```
set instance primary port 6123 6126
```

## 关闭实例

可以正常关闭实例，这样便可让事务进行完毕。例如，要关闭“ase1”，请输入：

```
shutdown instance ase1
```

要立即关闭实例而不等待事务完成，请输入：

```
shutdown instance ase1 nowait
```

如果关闭集群中的最后一个实例，集群状态就会变为 **Down**（已关闭）。如果关闭承载集群协调器的节点上的实例，就会由另一个节点来承载协调器。

## 删除实例

在删除实例前，应先确保实例处于 **Down**（已关闭）状态，而且集群处于 **Up**（已启动）状态。然后，输入：

```
drop instance instance_name
```

Adaptive Server 会从 `interfaces` 文件中删除该实例对应的条目并删除仲裁设备，而且将拓扑更改告知集群。对每个删除都要进行确认。

---

**注释** 不能使用 `drop instance` 来删除集群中的最后一个实例。而应该使用 `drop cluster`。

---

## 在手动创建集群后启用 sybcluster

通常，集群是使用 `sybcluster` 或 Adaptive Server 插件创建的。在这两种情况下，Adaptive Server 会自动添加配置信息，以使 `sybcluster` 或 Adaptive Server 插件能够与每个节点上的 Unified Agent 连接。如果您手动配置集群（如《安装指南》中所述），就必须先向 Unified Agent 中添加该配置信息，然后才能使用 `sybcluster` 或 Adaptive Server 插件来管理集群。

必须先配置插件配置信息。

- 1 如果还未配置该信息，请在集群上启动 Unified Agent。请参见所用平台的安装指南中的第 3 章“安装服务器和启动集群”。
- 2 配置插件。例如，要在缺省集群“mycluster”上配置集群代理插件信息，请输入：

```
deploy plugin agent "blade1,blade2,blade3"
```

可以使用任何直接连接或发现方法来指定 Unified Agent（如第 242 页的“标识 Unified Agent”中所述）。

指定代理后，Adaptive Server 会提示您输入到以下指定内容的路径：

- 仲裁设备
- 环境 shell 脚本
- ASE 主目录

---

**注释** 在将插件配置到集群中的单个节点上后，可以使用 sybcluster 或 Adaptive Server 插件来管理集群。但若要启动集群，则必须将插件配置到集群中的所有节点上。

---

还可以使用 `deploy plugin` 来更新现有插件的值。

## 创建和管理辅助服务器

可以使用 `sybcluster` 来创建、删除、配置以下辅助服务器的端口号，以及显示其当前端口号：

- Backup Server
- XP Server

## 创建辅助服务器

请参见《安装指南》中有关使用 `sybcluster` 来创建 Backup Server 和 XP Server 的说明。还可以使用 Adaptive Server 插件来创建辅助服务器。

有关语法和用法信息，请参见《实用程序指南》。

如果创建多个 Backup Server，则集群中的所有实例都必须具有一个 Backup Server。可以为一个或多个节点仅创建一个 Backup Server。

如果创建 XP Server，则必须针对集群中的所有实例创建它们。

## 删除辅助服务器

可以使用 `drop backupserver`、`drop monitorserver` 或 `drop xpserver` 来删除集群中的辅助服务器。Cluster Edition 会在删除服务器前提示您确认。

如果为集群配置了多个 Backup Server，则必须将所有 Backup Server 都删除。如果仅配置了单个 Backup Server，则可以删除一个或全部。但当您使用 `drop monitorserver` 或 `drop xpserver` 时，会从集群中删除所有 XP Server。

要从“mycluster”中删除所有 XP Server，请输入：

```
drop xpserver
```

```
Are you sure you want to drop the XP Servers from cluster "mycluster"? (Y or N): [N] y
```

```
The XP Servers have been dropped for all instances.
```

要从“mycluster”的“blade2”中删除 Backup Server，请输入：

```
drop backupserver
```

```
Do you want to drop the Backup Server from:
```

```
1. Selected nodes
```

```
2. Cluster
```

```
Enter choice: 1
```

```
Do you want to drop Backup Server from node "blade1"? [N] n
```

```
Do you want to drop Backup Server from node "blade2"? [N] y
```

```
Do you want to drop Backup Server from node "blade3"? [N] n
```

```
The Backup Server has been dropped.
```

## 显示监听端口信息

要显示 Backup Server 或 XP Server 的当前监听端口号，请使用：

- `show backupserver config`
- `show xpserver config`

例如，要显示 “mycluster” 的 Backup Server 监听端口信息，请输入：

```
show backupserver config
```

```
Backup Server is configured on the following nodes:  
  1. blade1:5001  
  3. blade3: 5003
```

## 更改监听端口信息

要更改辅助服务器的监听端口，请使用：

- `set backupserver`
- `set xpserver config`

例如，要更改 “blade3” 上的实例 “ase3” 的 Backup Server 监听端口号，请输入：

```
set backupserver
```

```
Enter the Backup Server port number for instance "blade1": [6011] <CR>  
Enter the Backup Server port number for instance "blade2": [6012] <CR>  
Enter the Backup Server port number for instance "blade3": [6013] 6666
```

## 升级服务器

请参见所用平台的安装指南中有关将 Adaptive Server 升级到当前版本 Cluster Edition 的说明。



# 常规配置问题

《集群用户指南》的本部分讲述 Cluster Edition 的常规配置问题。

---

## 配置操作系统

本章论述操作系统配置设置，您可以在安装或升级 Cluster Edition 之后调整这些设置。除非另行说明，否则此信息适用于所有支持的 UNIX 平台。

主题	页码
<a href="#">使用 stty 设置</a>	261
<a href="#">恢复正确权限</a>	261
<a href="#">文件描述符和用户连接</a>	262
<a href="#">调整客户端连接的超时期限</a>	265
<a href="#">检查硬件错误</a>	266
<a href="#">监控操作系统资源的使用情况</a>	267
<a href="#">C Shell 维护脚本样本</a>	268

### 使用 stty 设置

设置 `stty tostop` 选项会导致后台服务器在尝试向终端写入时停止。为了避免此错误，请在启动 Cluster Edition 之前执行以下命令：

```
stty -tostop
```

如果将所有 Cluster Edition 输出重定向到文件，则不必更改 `stty` 设置。

### 恢复正确权限

Sybase 软件文件和目录在安装时便设置了正确的访问权限。如果发现权限不再正确，您可以使用位于 `$SYBASE/$SYBASE_ASE/install` 目录中的脚本 `setperm_all` 恢复正确的权限。

## 文件描述符和用户连接

Cluster Edition 所使用的用户连接数不能超过操作系统上 Cluster Edition 可用的文件描述符数量。在 Cluster Edition 上配置用户连接时，系统管理员应考虑每个进程中可用的文件描述符的数量。尽管大多数打开文件描述符都可用于用户连接，但 Cluster Edition 只使用一小部分来打开文件和设备。

### 对于 Linux

每个进程的文件描述符数仅限于 10,000。可以使用 `ulimit` 来设置文件描述符数。

### 对于 Sun Solaris

对于 Sun Solaris，可以同时为文件描述符设置硬限制和软限制。用户可将软限制一直增加到硬限制，但硬限制只能由拥有“root”权限的人员来增加。软限制确定 Cluster Edition 引擎可用的打开文件描述符的数量。限制为 10,000。

尽管大多数打开文件描述符都可用于用户连接，但 Cluster Edition 引擎只使用一小部分来打开文件和设备。

有关用户连接的其它信息，请参见《系统管理指南》。

### 对于 HP-UX

内核参数 `maxfiles` 和 `maxfiles_lim` 控制任何一个进程可用的文件描述符的数量。对于 32 位和 64 位 HP-UX 系统，文件描述符的最大数量分别为 10,000 和 60,000。

要显示文件描述符的当前值，请使用 Korn 或 Bourne shell `ulimit` 命令：

```
ulimit -n
```

## 显示当前的软限制和硬限制

要显示 C shell 的当前软限制，请输入：

```
limit descriptors
```

对于 Bourne shell，输入：

```
ulimit -n
```

若要显示 C shell 的当前硬限制，请输入：

```
limit -h descriptors
```

对于 Bourne shell，输入：

```
ulimit -Hn
```

## 增加软限制

若要增加 C shell 的软限制，请输入：

```
limit descriptors n
```

对于 Bourne shell，输入：

```
ulimit -Sn new_value
```

其中，*n* 是软限制的当前值，*new\_value* 是要将软限制增加到的值。

---

**注释** 可以在 *RUN\_server\_name* 文件中使用上述命令，以便增加硬限制和软限制。*RUN\_server\_name* 文件是一个 Bourne shell 脚本，因此，请确保在 *RUN\_server\_name* 文件中使用这些命令的 Bourne shell 版本。

---

## 增加硬限制

若要增加硬限制，请使用与第 264 页的“示例程序”中所示的样本程序类似的程序。

### ❖ 设置示例程序以增加硬限制

- 1 使用 ASCII 编辑器创建 *file\_name.c*（其中，*file\_name* 是您为文件指定的名称）。键入第 264 页的“示例程序”的样本中所示的文本。
- 2 编译该文件：

```
cc file_name.c -o program_name
```

其中，*file\_name* 是您创建的源文件名，*program\_name* 是要为程序指定的名称。

- 3 更改程序的权限和所有权，使其可作为“root”来执行：

```
chmod 755 program_name  
chown root program_name
```

其中，*program\_name* 是已编译程序的名称。

- 4 通过在操作系统提示符处键入以下命令，“root”用户可以用该程序来启动已增加了用户连接的 Cluster Edition：

```
# program_name dataserver -d master_device_name
```

其中 *program\_name* 是已编译程序的名称，*master\_device\_name* 是 Cluster Edition 主设备的完整路径。可以在 Cluster Edition 的 *RUN\_server\_name* 文件中的 *dataserver* 命令行前面添加 *program\_name*，而不是在操作系统提示符处键入命令。

## 示例程序

---

**注释** 这是样本脚本；请根据需要进行修改。

---

以下示例显示可用于增加硬限制的源代码：

```
#include <sys/time.h>  
#include <sys/resource.h>  
#include <sys/types.h>  
/*  
** define MAX_CONNECTIONS to a number less than  
** 10000. The number defined will then become the maximum  
** number of connections allowed by an Adaptive Server.
```

```

*/
#define MAX_CONNECTIONS 9999
extern int errno;

main(argc,argv)
char **argv;
{
    struct rlimit rlp;
    uid_t uid;

    rlp.rlim_cur = MAX_CONNECTIONS;
    rlp.rlim_max = MAX_CONNECTIONS;
    /* set the number of open file descriptors to
    MAX_CONNECTIONS */
    if (setrlimit (RLIMIT_NOFILE,&rlp) == -1)
    {
        perror("setrlimit");
        exit(1);
    }

    /* reset the user id to disable superuser
    privileges */
    uid = getuid();
    setuid(uid);
    /* run the program indicated as arguments to
    this program */
    execv(++argv, argv);
}

```

有关用户连接的其它信息，请参见《系统管理指南》。

## 调整客户端连接的超时期限

Cluster Edition 使用 TCP/IP 协议的 `KEEPALIVE` 选项来检测已不再活动的客户端。如果与客户端的连接在一段时间（*超时期限*）内处于非活动状态，操作系统将每隔一定时间发送 `KEEPALIVE` 包。如果没有收到客户端计算机对任意这些包的响应，操作系统将通知 Cluster Edition 该客户端已不再响应。然后，Cluster Edition 将终止该客户端的连接。

`KEEPALIVE` 的缺省超时期限为 2 小时（7,200,000 毫秒）。若要显示当前时间值，请按以下各节所示使用适用于您的平台的命令。

## 对于 Sun Solaris

若要显示超时值，请输入：

```
/usr/sbin/ndd -get /dev/tcp tcp_keepalive_interval
```

若要将超时期限减少至 15 分钟（900,000 毫秒），请输入：

```
/usr/sbin/ndd -set /dev/tcp tcp_keepalive_interval 900000
```

## 对于 Linux

若要显示超时值，请输入：

```
/sbin/sysctl -e net.ipv4.tcp_keepalive_time
```

若要将超时期限减至 15 分钟（900 秒），请输入：

```
/sbin/sysctl -w net.ipv4tcp_keepalive_time=900
```

## 对于 HP-UX

要显示当前的超时期限值，请输入：

```
/usr/contrib/bin/nettune -l
```

**tcp-keepstart** 参数指定一段时间长度（以秒为单位），此时间段内空闲连接将保持为活动状态，此后系统才会检查连接是否不再活动。

要更改超时期限，请使用 **nettune -s** 命令：

## 检查硬件错误

以下类型的硬件错误消息指示可能导致数据库损坏的问题：

- 磁盘读、写或重试错误
- 超时
- 系统应急
- 任何类型的内存问题

## 对于 Sun Solaris

定期检查 `/var/adm/messages` 文件。如果出现本节开始处所描述的任何类型的硬件错误，请使用 Sun Microsystems 诊断工具 `sundiag` 来检查内存和磁盘。请参见操作系统文档。

## 对于 Linux

定期检查 `/var/log/messages` 文件。请参见操作系统文档。

## 对于 HP-UX

定期检查 `/var/adm/syslog/syslog.log` 文件。您可以直接查看该文件，也可以使用 HP-UX 的 `dmesg` 命令。请参见操作系统文档。

## 监控操作系统资源的使用情况

《系统管理指南》讨论了为负载和系统配置维护最佳数量的服务器引擎。要确定最佳数量，请监控系统 and CPU 的使用情况。

Sun Solaris 和 Linux 提供了以下工具，用于帮助监控性能：

- `iostat` 命令报告终端和硬盘的 I/O 总量以及 CPU 时间的使用情况。
- `vmstat` 命令监控虚拟内存的使用情况。
- `netstat` 命令监控网络状态。
- `ps` 命令提供 CPU 累计时间和单个进程的使用情况的精确快照。这有助于确定特定于数据服务器、引擎和进程的装载。
- `time` 命令有助于确定一次完整运行所使用的各种用户资源、系统资源和实时资源。

有关这些工具的详细信息，请参见操作系统文档。

## C Shell 维护脚本样本

通过运行 `dbcc` 检查并执行数据库备份，可以保护 Cluster Edition 数据库的完整性和可恢复性。以下 C shell 脚本样本调用几个 `isql` 脚本来帮助用户完成此操作：

```
#!/bin/csh -f
if ( -e dbcc_mail.out) then
    rm dbcc_mail.out
endif
foreach i (*.dbcc)
    isql -Usa -Ppassword < $i > dbcc_out
    if ( `Grep -c 'Msg 25[0-9][0-9]' dbcc_out` ) then
        echo "There are errors in" $i >> dbcc_mail.out
        cat dbcc_out >> dbcc_mail.out
    else
        echo "Backing up " $i:r >> dbcc_mail.out
        isql -Usa -Ppassword < $i:r.backup
    endif
end
mail -s "Backup Report" jjones < dbcc_mail.out
```

第一组脚本（每个数据库一个，文件名中附加 `.dbcc`）为每个数据库运行 `dbcc checkalloc` 和 `dbcc checkdb`，并将消息发送到名为 `dbcc_out` 的输出文件中。

例如，脚本 `master.dbcc` 运行 `dbcc` 以检查 `master` 数据库：

```
dbcc checkalloc (master)
go
dbcc checkdb (master)
go
```

然后，C shell 脚本运行 `grep` 命令，以便在 `dbcc` 输出中查找 2500 级错误消息。`grep` 命令的结果将存入名为 `dbcc_mail.out` 的输出文件中。

接着，该脚本将为每个没有出现 2500 级别错误的数据库调用 `isql` 备份脚本，并将 “Backing up `database_name`” 行添加到 `dbcc_mail.out`。例如，脚本 `master.backup` 将备份 `master` 数据库：

```
use master
go
dump database master to master_dump
go
```

您可能需要将相应的 `dump transaction` 命令添加到脚本。

如果有 2500 级错误消息，脚本将不会备份数据库。在脚本末尾，*dbcc\_mail.out* 被邮递给系统管理员 “jjones”，“jjones” 将获得一个关于 dbcc 致命错误和成功备份的记录。

可以定制此 shell 样本和 isql 脚本，以便满足安装的需要。

要使脚本自动执行，请编辑 *crontab* 文件，并添加以下类似条目：

```
00 02 * * * /usr/u/sybase/dbcc_ck 2>&1
```

此示例将在每天凌晨 2:00 执行名为 dbcc\_ck 的 C shell 脚本。



本章提供关于国际安装的 Sybase 本地化支持的信息，包括配置语言、字符集和排序顺序方面的信息。有关详细信息，请参见《系统管理指南》。

主题	页码
<a href="#">本地化支持概述</a>	271
<a href="#">字符集转换</a>	277
<a href="#">排序顺序</a>	278
<a href="#">语言模块</a>	282
<a href="#">本地化</a>	283
<a href="#">更改本地化配置</a>	286

## 本地化支持概述

本地化是一个设置应用程序以使其能够在特定的语言或国家/地区环境中运行的过程，包括翻译的系统消息以及日期、时间和货币的正确格式。Cluster Edition 支持国际客户和异构环境客户的本地化。

这种支持包括：

- 数据处理支持 — Cluster Edition 附带字符集和排序顺序定义文件，用于处理不同语言中使用的各种字符。

Sybase 对以下地区的主要语言提供支持：

- 西欧
- 东欧
- 中东
- 拉丁美洲
- 亚洲

- 已翻译的系统信息 — Cluster Edition 包括以下语言模块：
  - 巴西葡萄牙语
  - 简体中文
  - 法语
  - 德语
  - 日语
  - 朝鲜语
  - 波兰语
  - 西班牙语
  - 泰语

## 语言模块

Cluster Edition 将本地化软件消息存储在独立的语言模块中。

当安装某个语言模块时，安装程序在正确的位置装载支持新语言的消息、字符集和排序顺序文件。

安装 Cluster Edition 和 Backup Server 时，缺省情况下安装使用英语版的系统消息。

## 服务器的缺省字符集

缺省字符集是用于编码数据并将数据存储存储在 Cluster Edition 数据库中的字符集。

## 更改缺省语言和字符集

---

**警告！** 对于新的服务器，必须在创建任何用户数据库或更改 Sybase 提供的数据库之前，完成对字符集和排序顺序的所有更改。在将数据或数据结构添加到 Cluster Edition 之后，更改字符集和排序顺序可能需要执行其它步骤。若要在添加数据后更改字符集或排序顺序，请参见《系统管理指南》。

---

sybcluster 和 Adaptive Server 插件创建具有以下缺省值的实例：

- us\_english 语言
- iso\_1 字符集（HP-UX 平台上，使用 Roman8）
- 二进制排序顺序

## 更改服务器的缺省字符集

可以选择任一字符集作为 Cluster Edition 的缺省字符集，包括非所用平台缺省字符集的字符集。选择新的缺省字符集时谨记以下指南：

- 为避免转换错误或开销，要根据客户端使用的字符集来确定缺省字符集。

例如，如果大多数客户端使用 ISO 8859-1，则通过指定 ISO 8859-1，可将必须进行的数据转换的工作量降至最低。

- 如果服务器在异构语言环境中运行，则应选择可与所需的全部字符集一起使用的字符集。通常为 Unicode (UTF-8)。

---

**警告！** 对于新的实例，必须在创建任何用户数据库或更改 Sybase 提供的数据库之前，完成对缺省字符集和排序顺序的所有更改。在将数据或数据结构添加到实例之后，更改字符集和排序顺序可能会引发错误行为。若要在添加数据后更改字符集或排序顺序，请参见《系统管理指南》。

---

## 支持的字符集

Cluster Edition 支持以下语言、脚本和字符集：

- 阿拉伯语 — 请参见第 274 页的表 17-1。
- 波罗的语 — 请参见第 274 页的表 17-2。
- 简体中文 — 请参见第 274 页的表 17-3。
- 繁体中文 — 请参见第 274 页的表 17-4。
- 古斯拉夫语 — 请参见第 275 页的表 17-5。
- 东欧 — 请参见第 275 页的表 17-6。
- 希腊语 — 请参见第 275 页的表 17-7。
- 希伯来语 — 请参见第 275 页的表 17-8。

- 日语 — 请参见第 276 页的表 17-9。
- 朝鲜语 — 请参见第 276 页的表 17-10。
- 泰语 — 请参见第 276 页的表 17-11。
- 土耳其语 — 请参见第 276 页的表 17-12。
- Unicode（支持超过 650 种语言）— 请参见第 276 页的表 17-13。
- 越南语 — 请参见第 277 页的表 17-14。
- 西欧 — 请参见第 277 页的表 17-15。

以下表格定义了每种字符集。

请参见第 277 页的“字符集转换”。

表 17-1 列出了阿拉伯语字符集：

**表 17-1：阿拉伯语字符集**

字符集	说明
cp864	PC 阿拉伯语
cp1256	Microsoft Windows 阿拉伯语
iso88596	ISO 8859-6 拉丁语/阿拉伯语

表 17-2 列出了波罗的语字符集：

**表 17-2：波罗的语字符集**

字符集	说明
cp1257	Microsoft Windows 波罗的语

表 17-3 列出了简体中文字符集：

**表 17-3：简体中文字符集**

字符集	说明
eucgb	EUC GB 编码 = 简体中文字符集
cp936	Microsoft 简体中文字符集
gb18030	PRC 18030 标准

表 17-4 列出了繁体中文字符集：

**表 17-4：繁体中文字符集**

字符集	说明
cp950	PC (Microsoft) 繁体中文
euccns	EUC CNS 编码 = 带扩展的繁体中文
big5	Big 5 繁体中文
big5hk	带 HKSCS 扩展的 Big 5

表 17-5 列出了古斯拉夫语字符集：

**表 17-5: 古斯拉夫语字符集**

字符集	说明
cp855	IBM PC 古斯拉夫语
cp866	PC 俄语
cp1251	Microsoft Windows 3.1 古斯拉夫语
iso88595	ISO 8859-5 拉丁语/古斯拉夫语
koi8	KOI-8 古斯拉夫语
mac_cyr	Macintosh 古斯拉夫语

表 17-6 列出了东欧字符集：

**表 17-6: 东欧字符集**

字符集	说明
cp852	PC 东欧
cp1250	Microsoft Windows 3.1 东欧
iso88592	ISO 8859-2 拉丁语 -2

表 17-7 列出了希腊语字符集：

**表 17-7: 希腊语字符集**

字符集	说明
cp869	IBM PC 希腊语
cp1253	MS Windows 希腊语
greek8	HP GREEK8
iso88597	ISO 8859-7 拉丁语/希腊语
macgrk2	Macintosh 希腊语

表 17-8 列出了希伯来语字符集：

**表 17-8: 希伯来语字符集**

字符集	说明
cp1255	Microsoft Windows 希伯来语
iso88598	ISO 8859-8 希伯来语

表 17-9 列出了日语字符集：

**表 17-9: 日语字符集**

字符集	说明
cp932	IBM J-DBCS:CP897 + CP301 (Shift-JIS)
eucjis	EUC-JIS 编码
sjis	Shift-JIS (没有扩充)
deckanji	DEC Kanji

表 17-10 列出了朝鲜语字符集：

**表 17-10: 朝鲜语字符集**

字符集	说明
eucksc	EUC KSC 朝鲜语编码 = CP949
cp949	Ms Windows 朝鲜语

表 17-11 列出了泰语字符集：

**表 17-11: 泰语客户端字符集**

字符集	说明
tis620	TIS-620 泰语标准
cp874	Microsoft Windows 泰语

表 17-12 列出了土耳其语字符集：

**表 17-12: 土耳其语字符集**

字符集	说明
cp857	IBM PC 土耳其语
cp1254	Microsoft Windows 土耳其语
iso88599	ISO 8859-9 拉丁语 -5 土耳其语
turkish8	HP TURKISH8
macturk	Macintosh 土耳其语

表 17-13 列出了 Unicode 字符集：

**表 17-13: Unicode 字符集**

字符集	说明
utf8	Unicode UTF-8 编码

表 17-14 列出了越南语字符集：

**表 17-14：越南语字符集**

字符集	说明
cp1258	Microsoft Windows 越南语

表 17-15 列出了西欧字符集：

**表 17-15：西欧字符集**

字符集	说明
ascii8	具有 8 位数据的 US ASCII, ISO 646
cp437	IBM CP437 — 美国代码集
cp850	IBM CP850 — 欧洲代码集
cp860	PC 葡萄牙语
cp858	带欧元支持的 cp850
cp1252	Microsoft Windows US (ANSI)
iso_1	ISO 8859-1 拉丁语 -1
roman8	HP ROMAN8
iso15	带欧元支持的 ISO 8859-15 拉丁语 -1
roman9	带欧元支持的 HP ROMAN8
mac	Macintosh 罗马语
mac_euro	带欧元支持的 Macintosh 罗马语

## 字符集转换

Backup Server 使用客户端语言和 Cluster Edition 字符集将消息传递给 Cluster Edition。然后 Cluster Edition 使用客户端的语言和字符集转换并发出消息。选择字符集时谨记以下要求：

- 在异构环境中，Cluster Edition 和 Backup Server 可能需要与运行在不同平台上和使用不同字符集的客户端通信。为了保持数据完整性，服务器在字符集之间转换代码。
- Unicode 转换可应用于所有本地字符集。在两种本地字符集之间进行转换时，Unicode 转换方式将 Unicode 作为中间字符集。例如，要在服务器的缺省字符集 (CP 437) 和客户端的字符集 (CP 860) 之间转换，先是将 CP 437 转换为 Unicode，然后将 Unicode 转换为 CP 860。缺省情况下，会将 Unicode 转换用于字符集转换。

- Adaptive Server 直接转换支持同一语言组内两种本地字符集之间的转换。例如，Adaptive Server 支持 CP 437 与 CP 850 之间的转换，因为它们都属于第 1 语言组。若要使用直接转换，必须在服务器上为客户端使用的所有字符集安装字符集定义文件。要启用直接转换，必须运行 `sp_configure 'enable unicode conversion', 0` 以禁用 Unicode 转换。

如果 Cluster Edition 或 Backup Server 不支持客户端的语言或字符集，该服务器将发出警告消息。Backup Server 字符集与 Cluster Edition 字符集不兼容时，也会出现错误。

有关所支持转换的详细信息，请参见《系统管理指南》。

## 服务器与客户端之间的转换

如果 Cluster Edition 不支持客户端的语言或字符集，则客户端可以与服务器连接，但不会发生字符转换。

当已本地化的客户端应用程序连接到 Cluster Edition 时，服务器检查其是否支持客户端的语言和字符集。

- 如果 Cluster Edition 支持该语言，将自动执行所有字符集转换，并用客户端的语言和字符集显示消息。
- 如果 Cluster Edition 不支持该语言，则使用用户的缺省语言或 Cluster Edition 的缺省语言。
- 如果 Cluster Edition 不支持该字符集，会向客户端发出警告，关闭转换并将语言设置为美国英语。

## 排序顺序

每个字符集都带有一种或多种排序顺序（归类序列），它们位于排序顺序定义文件（.srt 文件）中或安装在系统中（如果是 Unicode 排序顺序）。这些文件随字符集定义文件一起提供，并可在同一目录中找到。

可以根据您站点的需要为数据选择排序顺序。但服务器一次仅能支持一种排序顺序，因此，请选择适用于所有客户端的排序顺序。

**警告！** 对于新的服务器，必须在创建任何用户数据库或更改 Sybase 提供的数据库之前，完成对缺省字符集和排序顺序的所有更改。在将数据或数据结构添加到 Cluster Edition 之后，更改字符集和排序顺序可能会引发错误行为。若要在添加数据后更改字符集或排序顺序，请参见《系统管理指南》。

## 可用的排序顺序

排序顺序决定了 Cluster Edition 用于排序、比较字符数据和对其编制索引的归类序列。每种字符集都带有一种或多种排序顺序。

排序顺序位于排序顺序定义文件（.srt 文件）中，该文件随字符集定义文件一起提供。

**注释** 依据安装在 Cluster Edition 上的字符集，可用的排序顺序会有所不同。

通过查看所用语言的 .srt 文件，可以看到所用字符集的可用排序顺序。排序顺序存储在：

```
$SYBASE/charsets/<charset_name>/*.srt
```

要查看可用的 Unicode 排序顺序，请运行 sp\_helpsort。请参见第 283 页的“本地化目录”。

表 17-16 描述了可用的排序顺序。

**表 17-16: Cluster Edition 中可用的排序顺序**

排序顺序名	说明
二进制顺序	根据该字符集的字节数字值对所有数据进行排序。二进制顺序将所有 ASCII 大写字母排在小写字母之前。加变音或表意（多字节）字符以其各自的标准顺序排序，这可能为任意顺序。 所有字符集的缺省排序顺序都是二进制顺序。如果二进制顺序不能满足需要，可以在安装时或安装后使用 charset 实用程序指定其它排序顺序之一。
字典顺序，区分大小写，区分变音	区分大小写。将每个大写字母排在相应的小写字母之前，包括加变音的字符。识别字母的各种加变音的形式，并将它们排在相关联的未加变音字母之后。
字典顺序，不区分大小写，区分变音	不区分大小写字典排序顺序。大写字母与相应的小写字母等同，在排序结果中大小写字母混合在一起。

排序顺序名	说明
字典顺序，不区分大小写，不区分变音	不区分大小写字典排序顺序。变音标记被忽略。
字典顺序，不区分大小写，具有优先级	不区分大小写字典排序顺序，带有用于归类的大小写优先级。用大写字母书写的字与用小写字母书写的同一字等同。 只有使用 <code>order by</code> 子句时才区分大小写字母。 <code>order by</code> 子句先对大写字母排序，然后再对小写字母排序。
	<b>注释</b> 不要选择此排序顺序，除非安装的版本要求在 <code>order by</code> 子句的字符串中，大写字母排在小写字母之前（其余字符串部分相同）。当 <code>order by</code> 子句中指定的列与表的聚簇索引键相匹配时，使用这种排序顺序可能会降低大表的性能。
替代字典顺序，区分大小写	将小写变量排在大写变量之前的区分大小写的替代字典排序顺序。 用于几种西欧语言。
替代字典顺序，不区分大小写，不区分变音	不区分大小写和变音的替代字典排序顺序。 用于几种西欧语言。
替代字典顺序，不区分大小写，大写优先	不区分大小写的替代字典排序顺序，但大写优先。 用于几种西欧语言。
西班牙语字典顺序，区分大小写	区分大小写的西班牙语字典排序顺序。 用于西班牙和大多数拉丁美洲区域设置。
不区分大小写的西班牙语字典顺序	不区分大小写的西班牙语字典排序顺序。 用于西班牙和大多数拉丁美洲区域设置。
西班牙语字典顺序，不区分大小写，不区分变音	不区分大小写和变音的西班牙语字典排序顺序。 用于西班牙和大多数拉丁美洲区域设置。
斯堪的纳维亚字典顺序，区分大小写	区分大小写的字典排序顺序。 用于斯堪的纳维亚语言。
斯堪的纳维亚字典顺序，不区分大小写，大写优先	不区分大小写和变音的字典排序，但大写优先。 用于斯堪的纳维亚语言。

要查看可用的排序顺序，请使用 `charset` 实用程序显示计划使用的字符集的排序顺序。有关 UTF-8 的 `unicode` 排序顺序的详细信息，请参见《系统管理指南》中的“配置字符集、排序顺序和语言”。

表 17-17 列出了可用的缺省 Unicode 排序顺序。

**表 17-17: 缺省的 Unicode 排序顺序**

名称	ID	说明
defaultml	20	缺省的 Unicode 多语言排序
thaidict	21	泰语字典顺序
iso14651	22	按照每个 ISO14651 标准进行排序
utf8bin	24	为与 UTF-8 二进制相匹配的 UTF-16 排序
binary	25	二进制排序
altnoacc	39	替代不区分变音
altdict	45	替代字典排序
altnocsp	46	替代不区分大小写且具有优先级
scandict	47	斯堪的纳维亚语字典排序
scannocp	48	斯堪的纳维亚语不区分大小写且具有优先级
bin_utf8	50	UTF-8 二进制排序顺序
dict	51	通用字典顺序
nocase	52	不区分大小写的通用字典顺序
nocasep	53	通用的不区分大小写且具有优先级
noaccent	54	通用的不区分变音的字典顺序
espdict	55	西班牙语字典顺序
espnoacs	56	不区分大小写的西班牙语字典顺序
espnoac	57	不区分变音的西班牙语字典顺序
rusnoacs	59	不区分大小写的俄语字典顺序
cyrnoacs	64	不区分大小写的古斯拉夫语字典顺序
elldict	65	希腊语字典顺序
hundict	69	匈牙利语字典顺序
hunnoacs	70	不区分变音的匈牙利语字典顺序
hunnoacs	71	不区分大小写的匈牙利语字典顺序
turknoacs	73	不区分变音的土耳其语字典顺序
cp932bin	129	CP932 二进制排序顺序
dynix	130	GB 拼音排序顺序
gb2312bn	137	GB2312 二进制排序顺序
cyrdict	140	古斯拉夫文字典排序顺序
turdict	155	土耳其文字典排序顺序
euckscbn	161	EUCKSC 二进制排序顺序
gbpinyin	163	GB 拼音排序顺序
rusdict	165	俄文字典排序顺序
sjisbin	179	SJIS 二进制排序顺序
eucjisbn	192	EUCJIS 二进制排序顺序
big5bin	194	BIG5 二进制排序顺序

## 语言模块

如果要用美国英语 (us\_english) 以外的其它语言显示 Cluster Edition 错误消息，则必须安装相应的语言模块。

在安装新的语言模块时，安装程序自动将语言装载到 Sybase 安装目录下，以支持新的语言。请参见第 283 页的“本地化目录”。

## 安装新的语言模块

完整的服务器安装会自动包括所有语言组件。如果未选择完全安装，则需要手动安装其它语言模块。

若要安装新的语言模块：

- 1 从分配介质上装载语言模块软件。必须将此软件装载到装载 Cluster Edition 的同一目录下。
- 2 为 Cluster Edition 重新配置语言，如有必要，还要重新配置字符集和排序顺序。请参见第 286 页的“更改本地化配置”。

## 消息语言

对于消息，美国英语作为缺省语言安装在 Cluster Edition 中。以下规则适用于语言模块：

- 在 Cluster Edition 安装或重新配置期间，可以指定除美国英语以外的缺省语言。但是，您必须已经安装了所指定语言的语言模块。
- 如果客户端要求 Cluster Edition 以非美国英语的语言显示消息，则必须装载这些语言的语言模块。然后，可以将 Cluster Edition 配置成客户端所使用的语言。
- 如果 Cluster Edition 不支持使用客户端语言的消息，则这些客户端将接收到使用服务器缺省语言的消息。

例如，如果客户端的语言是拉丁语，而 Cluster Edition 安装了西班牙语模块并将西班牙语指定为缺省语言，则客户端将接收到西班牙语消息。

## 本地化

缺省情况下，Cluster Edition 和 Backup Server 配置使用英语地区设置，包括：

- 西欧字符集的字符集定义文件
- 西欧字符集的排序顺序定义文件
- 美国英语系统消息文件

在安装过程中或重新配置期间，可以指定不同的语言、字符集和排序顺序。

## 本地化目录

Sybase 本地化配置涉及以下目录：

- *locales*
- *charsets*

下表说明了本地化文件的结构。它没有显示所有文件的完整列表。

	<i>charset_name</i>	*.srt 文件
<i>\$\$SYBASE/charsets</i>	<i>charset_name...</i>	<i>charset.loc</i>
	<i>unicode</i>	*.uct 文件
<i>\$\$SYBASE/\$SYBASE_ASE/locales</i>	<i>language_name</i>	<i>charset_name</i>
	<i>language_name...</i>	<i>charset_name...</i>
<i>%SYBASE/locales</i>	<i>language.dat</i>	<i>language_name</i>
	<i>message</i>	<i>language_name...</i>

## 关于目录

*\$\$SYBASE/locales* 和 *\$\$SYBASE/\$SYBASE\_ASE/locales* 目录针对每种可用语言包含一个子目录。每个语言子目录又针对该语言可用的每个字符集包含一个子目录。

- 这些子目录中的 *.loc* 文件使 Cluster Edition 或 Backup Server 能够在特定字符集中编码的特定语言报告错误。

在每个子目录下有多种 *.loc* 文件。其中的大多数文件都包含特定产品或实用程序的翻译后的错误消息。

- 每个子目录下的 *common.loc* 文件都包含所有产品都要使用的已本地化的信息，如本地日期、时间和货币格式。
- *locales.dat* 文件包含的条目将平台特定的地区名与 Sybase 语言及字符集组合相关联。

## 关于 *charsets* 目录

*\$\$SYBASE/charsets/charset\_name* 中的文件包含与每种特定字符集相关的信息，如字符集的定义以及该字符集可用的任何排序顺序。

## 关于 *locales.dat* 文件

可以编辑 *locales.dat* 文件达到以下目的：

- 更改平台的缺省语言或字符集，或
- 添加平台的地区名与 Sybase 语言及字符集名之间的新关联。

### *locales.dat* 文件条目的格式

*locales.dat* 文件中的每一条目都可将平台特定的区域设置定义链接到一个 Sybase 语言和字符集组合。每个条目都有以下格式：

```
locale = platform_locale, syb_language, syb_charset
```

其中：

- *platform\_locale* 是区域设置的特定于平台的关键字。有关可接受值，请参见操作系统文档。  
如果定义的地区是站点的缺省地区，则 *platform\_locale* 为 “default”。
- *syb\_language* 是在 *\$\$SYBASE/locales/language\_name* 中使用的语言目录的名称。
- *syb\_charset* 是 *\$\$SYBASE/locales/language\_name/charset\_name* 中的字符集名，它为客户端确定字符集转换方法并标识消息文件的目录位置。

例如，以下条目指定缺省区域设置使用 *us\_english* 作为语言和使用 *iso\_1* 作为字符集：

```
locale = default, us_english, iso_1
```

## 客户端应用程序如何使用 *locales.dat*

客户端应用程序使用 *locales.dat* 文件来标识要使用的语言和字符集。连接过程遵照以下步骤进行：

- 1 客户端应用程序启动时，它将检查操作系统的地区设置，然后检查 *locales.dat* 文件来判定设置是否适合于 Cluster Edition。例如，法语的区域设置条目如下所示：

```
locale = fr_FR, french, iso_1
```
- 2 当客户端连接到 Cluster Edition 时，语言和字符集信息被传递到登录记录中的 Cluster Edition。
- 3 然后 Cluster Edition 使用：
  - 字符集信息（如 iso\_1），以便标识客户端的字符集并检验其能否将字符数据转换成该字符集
  - 语言（上面示例中为法语）和字符集信息，以便查看是否有以客户端语言表示的消息

---

**注释** Cluster Edition 软件包含一些已在 *locales.dat* 文件中定义的地区条目。如果这些条目不符合需要，您可以修改这些条目或添加新的区域设置条目。

---

## 编辑 *locales.dat* 文件

在开始进行编辑之前，请复制原始文件，以防编辑后的文件出现问题。

若要编辑 *locales.dat* 文件：

- 1 在文本编辑器中打开 *locales.dat* 文件副本。
- 2 查找用括号括起来的部分：
  - 对于 Sun Solaris, [*sun\_svr4*]
  - 对于 HP, [*hp ux*]
  - 对于 IBM, [*aix*]

- 3 确保该部分包含一个条目，用于要使用的语言 (*syb\_language*) 和字符集 (*syb\_charset*) 组合。

---

**注释** *platform\_locale* 的值必须与操作系统所需的值相匹配。如果系统配置文件中的区域设置定义与 Sybase 区域设置定义不匹配，应用程序将不能正确运行。

---

例如，如果要以法语显示 Open Client 消息，而 Cluster Edition 使用 ROMAN8 字符集，则需要检查所用平台的 *locales.dat* 条目，查找如下条目：

```
locale = fr_FR, french, roman8
```

- 4 添加所需条目或修改现有条目。
- 5 如果进行了更改，保存更改并退出文本编辑器。

## 更改本地化配置

缺省情况下，Cluster Edition 和 Backup Server 配置使用英语地区设置本地化，包括：

- 西欧字符集的字符集定义文件
- 西欧字符集的排序顺序定义文件
- *us\_english* 系统消息文件

在安装过程中和重新配置期间，可以指定另一种语言、字符集和排序顺序。

## Cluster Edition 本地化

每一语言的每一模块大约占用 2MB 的数据库空间。如有必要，在添加另一种语言之前，使用 `alter database` 命令增加 *master* 数据库的大小。

---

**注释** 如果要在 Cluster Edition 上安装多种语言，而 *master* 数据库中的空间不足以管理多种语言，事务日志可能变得过满。只能在主设备上扩展 *master* 数据库。请参见《系统管理指南》。

---

- 1 如果尚未设置 Sybase 环境变量，则获取 *SYBASE.csh* 或 *SYBASE.sh*。
- 2 使用 `langinstall` 实用程序来配置 Cluster Edition 的本地化：

```
$SYBASE/$SYBASE_ASE/bin/langinstall
```

下面是 `langinstall` 的语法：

```
langinstall [-S server_name] [-U user_name] [-P password]
[-R release_number] [-I path_to_interfaces] [-v] language character_set
```

例如，要安装缺省字符集为 `iso_1` 的法语语言，请输入：

```
langinstall -Usa -P -Sserver_name french iso_1
```

## Backup Server 本地化

可以通过修改 `RUN_<backup_server_name>` 文件来更改 Backup Server 语言和字符集。有关 `backupsaver` 命令参数的详细信息，请参见《实用程序指南》。

## 配置 Cluster Edition 使用其它字符集

要配置 Cluster Edition 使用所用语言的字符集和排序顺序，请完成以下步骤。系统消息以缺省语言（英语）显示。

- 1 使用 `charset` 实用程序装载缺省字符集和排序顺序。

若要使用 `charset`，服务器必须正在运行，并且用户必须具有系统管理员特权。使用排序顺序的 *file name*：

```
$SYBASE/$SYBASE_ASE/bin/charset -Usa -Ppassword
-Sserver_name sort_order_file character_set
```

用排序顺序文件名替换 `sort_order_file`。请参见第 288 页的表 17-18。用字符集的 Sybase 名称替换 `character_set`。请参见第 290 页的表 17-19。

- 2 使用 `charset` 实用程序来装载任何其它字符集。请参见第 291 页的“`charset` 实用程序”。

若要使用 Cluster Edition 内置字符集转换，必须为客户端平台上的所有字符集装载字符集定义文件。如果使用 Unilib 字符集转换，则不必执行此操作。

- 使用 `isql`，以 “sa” 登录到服务器，然后选择 `master` 数据库。

```
1> use master
2> go
```

- 使用排序顺序的 ID 配置服务器的新字符集和排序顺序。

```
1> sp_configure "default sortorder_id",
2> sort_order_id, "character_set"
3> go
```

用排序顺序 ID 替换 `sort_order_id`。请参见第 288 页的表 17-18。用字符集的 Sybase 名称替换 `character_set`。请参见第 290 页的表 17-19。

- 关闭集群。可以使用 `sybcluster`、Adaptive Server 插件，如果是手动配置了集群，还可以使用命令行选项。
- 使用 `sybcluster` 或 Adaptive Server 插件重新启动集群中的任何实例。

**注释** 如果是手动配置了集群，则使用 UNIX 系统上的常规过程便可重新启动实例。通常，这意味着包括以下 `dataserver` 命令：

```
$$SYBASE/$SYBASE_ASE/BIN/dataserver
--quorum_dev=quorum_path --instance=instance_name
```

- 实例启动，重新建立所有系统索引，然后关闭。再次重新启动，以使实例进入稳定状态。
- 检查集群日志文件，以检验字符集和排序顺序更改是否已成功完成。

## 排序顺序

表 17-18 描述了可用的排序顺序。如果未显示用户的语言，说明没有特定于该语言的排序顺序 — 将使用二进制排序顺序。

**表 17-18: 可用的排序顺序**

语言或脚本	排序顺序	文件名	ID
所有语言	二进制顺序	<code>binary.srt</code>	50
中欧捷克语、斯洛伐克语 这些排序顺序只适用于 CP 852、CP 1250 和 ISO 8859-2	字典顺序，区分大小写，区分变音	<code>czedit.srt</code>	80
	字典顺序，不区分大小写，区分变音	<code>czeocs.srt</code>	82
	字典顺序，不区分大小写，不区分变音	<code>czenoac.srt</code>	81
古斯拉夫语	字典顺序，区分大小写，区分变音	<code>cyrdict.srt</code>	63
	字典顺序，区分大小写，区分变音	<code>cyrnocs.srt</code>	64

语言或脚本	排序顺序	文件名	ID
英语	字典顺序, 区分大小写, 区分变音	dictiona.srt	51
法语	字典顺序, 不区分大小写, 区分变音	nocase.srt	52
德语	字典顺序, 不区分大小写, 区分变音, 具有优先级	nocasepr.srt	53
这些排序顺序与所有西欧字符集一起使用。	字典顺序, 不区分大小写, 不区分变音	noaccent.srt	54
英语	替代字典顺序, 区分大小写	altdict.srt	45
法语	替代字典顺序, 区分大小写, 不区分变音	altnoacc.srt	39
德语	替代字典顺序, 区分大小写, 具有优先级	altnocsp.srt	46
这些排序顺序只与 CP 850 一起使用。			
希腊语	字典顺序, 区分大小写, 区分变音	elldict.srt	65
此排序顺序只与 ISO 8859-7 一起使用。			
匈牙利语	字典顺序, 区分大小写, 区分变音	hundict.srt	69
这些排序顺序只与 ISO 8859-2 一起使用。	字典顺序, 不区分大小写, 区分变音	hunnoac.srt	70
	字典顺序, 不区分大小写, 不区分变音	hunnocs.srt	71
俄语	字典顺序, 区分大小写, 区分变音	rusdict.srt	58
这些排序顺序与除 CP 855 以外的所有古斯拉夫语字符集一起使用。	字典顺序, 不区分大小写, 区分变音	rusnoes.srt	59
斯堪的纳维亚语	字典顺序, 区分大小写, 区分变音	scandict.srt	47
这些排序顺序只与 CP 850 和 CP858 一起使用	字典顺序, 不区分大小写, 具有优先级	scannoep.srt	48
西班牙语	字典顺序, 区分大小写, 区分变音	espdict.srt	55
	字典顺序, 不区分大小写, 区分变音	espnoes.srt	56
	字典顺序, 不区分大小写, 不区分变音	espnoac.srt	57
泰语	字典顺序	dictionary.srt	51
土耳其语	字典顺序, 区分大小写, 区分变音	turdict.srt	72
这些排序顺序只与 ISO 8859-9 一起使用。	字典顺序, 不区分大小写, 不区分变音	turnoac.srt	73
	字典顺序, 不区分大小写, 区分变音	turnoes.srt	74

## 字符集

表 17-19 列出了支持的字符集及其 Sybase 名称。

**表 17-19: Sybase 字符集名称**

字符集	Sybase 名称
ASCII 8	ascii_8
Big 5	big5
Big 5HK	big5hk
CP 437	cp437
CP 850	cp850
CP 852	cp852
CP 855	cp855
CP 857	cp857
CP 858	cp858
CP 860	cp860
CP 864	cp864
CP 866	cp866
CP 869	cp869
CP 874	cp874
CP 932	cp932
CP 936	cp936
CP 949	cp 949
CP 950	cp950
CP 1250	cp1250
CP 1251	cp1251
CP 1252	cp1252
CP 1253	cp1253
CP 1254	cp1254
CP 1255	cp1255
CP 1256	cp1256
CP 1257	cp1257
CP 1258	cp1258
DEC Kanji	deckanji
EUC-CNS	euccns
EUC-GB	eucgb
EUC-JIS	eucjis
EUC-KSC	eucksc
GB 18030	gb18030
GREEK8	greek8

字符集	Sybase 名称
ISO 8859-1	iso_1
ISO 8859-2	iso88592
ISO 8859-5	iso88595
ISO 8859-6	iso88596
ISO 8859-7	iso88597
ISO 8859-8	iso88598
ISO 8859-9	iso88599
ISO 8859-15	iso15
Koi8	koi8
MAC	mac
MAC_CYR	mac_cyr
MAC_EE	mac_ee
MAC_EURO	mac_euro
MACGRK2	macgrk2
MACTURK	macturk
ROMAN8	roman8
ROMAN9	roman9
Shift-JIS	sjis
TIS 620	tis620
TURKISH8	turkish8
UTF-8	utf8

## charset 实用程序

使用 `charset` 实用程序可将字符集和排序顺序装载到 Cluster Edition 中。如果使用 `charset` 装载缺省字符集和排序顺序，此操作应该仅在安装时进行。

若要更改 Cluster Edition 的缺省字符集和排序顺序，请参见《系统管理指南》。

语法

```
charset
[ -U username ]
[ -P password ]
[ -S server ]
[ -I interfaces ]
[ -v version ]
sort_order
[ charset ]
```

表 17-20: *charset* 的关键字和选项

关键字和选项	说明
-U	如果尚未以“sa”身份登录到操作系统，则必须在命令行中指定“-Usa”或“/username = sa”。
-P	在命令行中指定“sa”口令。如果未指定，将提示用户输入“sa”口令。
-S	指定服务器名。如果未指定， <i>charset</i> 会使用 DSQUERY 环境变量来标识服务器名。如果没有 DSQUERY 环境变量， <i>charset</i> 会尝试连接到名为“SYBASE”的服务器。
-I	指定要使用的 interfaces 文件。如果未指定， <i>charset</i> 会使用 SYBASE 目录中的 interfaces 文件。
-v	输出 Sybase 版本字符串，然后退出。使用时不指定其它选项。
<i>sort_order</i>	使用 <i>charset</i> 装载缺省字符集和排序顺序时， <i>sort_order</i> 是必需的参数，它用于指定 Cluster Edition 要使用的排序顺序文件的名称。装载其它字符集时，用 <i>charset.loc</i> 指示字符集文件的名称。
<i>charset</i>	指定 Cluster Edition 要使用的字符集的目录。

本章提供有关为 Cluster Edition 增加可选功能的说明：

主题	页码
<a href="#">添加审计</a>	293
<a href="#">安装 Transact-SQL 语法的联机帮助</a>	301

在系统上安装了 Sybase 产品后，请参见产品文档以了解有关配置和管理的问题。

## 添加审计

审计是数据库管理系统中安全性的重要部分。与安全性相关的系统活动记录在审计追踪中，审计追踪可用于检测系统渗透和资源误用情况。通过检查审计追踪，系统安全员可以对数据库中对象的访问模式进行检查，并监控特定用户的活动。可以对特定用户的审计记录进行跟踪，从而使审计系统能够防止用户试图误用系统。

系统安全员负责管理审计系统，并且只有系统安全员才能启动和停止审计、设置审计选项和处理审计数据。

## 审计系统设备和数据库

审计系统包括多个组件。主要组件有：

- `sybsecurity` 设备和 `sybsecurity` 数据库，该数据库用于存储审计信息
- 审计追踪，由配置时确定的多个审计设备和表组成
- `syslogs` 事务日志设备，用于存储事务日志

sybsecurity 设备和数据库

sybsecurity 设备存储 sybsecurity 数据库。sybsecurity 数据库是在审计配置过程中创建的。该数据库包含 model 数据库中的所有系统表，并包含用于跟踪服务器范围内审计选项的系统表和审计追踪的系统表。

用于审计追踪的表和设备

Cluster Edition 将审计追踪存储在系统表中，命名为 sysaudits\_01 到 sysaudits\_08。在任何指定的时间，都只有一个审计表为 *当前* 审计表。Cluster Edition 将所有审计数据都写入当前审计表中。系统安全员可以使用 sp\_configure 来设置或更改哪个审计表为当前表。

当您配置 Cluster Edition 以便进行审计时，需要确定安装的审计表数量。最多可以指定八个系统表（sysaudits\_01 到 sysaudits\_08）。应计划使用至少两个或三个审计追踪系统表，并将每个系统表放置在它自己的设备上，与主设备分开。如果这样做，可以使用阈值过程，在当前审计表填满并切换到新的空审计表以存储随后的审计记录之前，该过程会将当前审计表自动存档。

用于 syslogs 系统表的设备

在审计配置期间，您必须为 syslogs 系统表指定一个单独的设备，该系统表包含事务日志。每个数据库中都存在的 syslogs 表包含数据库中所执行事务的日志。

## 在 Cluster Edition 中运行 auditinit

如果在 Adaptive Server 的非聚簇版本中使用 auditinit，auditinit 将启动服务器（如果服务器未运行）。但是，Cluster Edition 要求在使用 auditinit 实用程序之前必须先运行一个实例。如果该实例未启动，auditinit 则不会启动该实例。如果您尝试在 auditinit 未启动的情况下登录 Cluster Edition，auditinit 将显示以下警告消息：

```
Can not login to server because it is not running.
Please manually start the server and retry
```

## 审计设备的预安装任务

确定 sybsecurity、syslogs 和 sysaudits 表设备的裸设备位置。稍后将需要提供这些信息。

Sybase 建议：

- 为系统配置所需的最小数量审计设备 — 但至少必须配置三个设备。以后可以使用 sp\_addauditable 添加更多的审计设备。有关信息，请参见《参考手册：过程》。

- 按一对一的比例安装审计表和设备。共享相同设备的表将共享相同的阈值上限值。当设备已满时将无法按顺序使用这些表，因为它们驻留在同一设备上。
- 将每个审计表安装在它自己的设备上。这样，您将可以设置一个平稳运行的审计系统，而不会丢失审计记录。通过使用两个审计表，您可以在一个审计表已满时切换到另一个。通过使用第三个审计表，如果一个设备出现故障，系统安全员可以安装一个新的阈值过程，该过程将更改设备轮换以跳过损坏的设备，直到该设备修好为止。
- 使设备的大小大于表的大小。如果仅使用三个审计表和设备，表的大小和设备大小可以接近，因为您可以通过添加其它审计表和设备（最多八个）来获得更强的审计能力。如果使用的表和设备达到了上限值（六到八个），则可能需要使设备比表大很多。这样，如果以后需要更强大的审计能力，但只有少数附加设备或没有附加设备可用，则您可以扩展表大小，使其接近设备的上限大小。

## 安装审计

### ❖ 配置 Cluster Edition 以进行审计

- 1 如果尚未设置 Sybase 环境变量，则获取 *SYBASE.csh* 或 *SYBASE.sh* 文件。
- 2 在 UNIX 提示符下启动 *auditinit*（*auditinit* 实用程序位于 *\$\$SYBASE/\$SYBASE\_ASE/install* 中）：

```
$$SYBASE/$SYBASE_ASE/install/auditinit
```

*auditinit* 将显示以下菜单：

```
AUDITINIT
1. Release directory: /usr/u/sybase
2. Configure a Server product
```

- 3 选择“配置服务器产品” (Configure a Server Product)。
- 4 选择 Adaptive Server。
- 5 选择“配置现有 Sybase 服务器” (Configure an Existing Sybase Server)。
- 6 选择要配置的服务器。
- 7 提供所选服务器的 SA 口令。

- 8 从“Sybase 服务器配置” (Sybase Server Configuration) 屏幕中，选择“配置审计” (Configure Auditing)。

继续进行 `auditinit` 中的菜单时，您可以更改显示的任何缺省值。完成每个菜单时，按 `Ctrl+A` 接受缺省值或更改的值，然后移动到下一个菜单。

```
CONFIGURE AUDITING
```

1. Configure auditing: no
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry

```
List of devices for the audit tables:
```

Logical name	Physical name	Segment name	Table name	Size
--------------	---------------	--------------	------------	------

```
Device for the audit database transaction log:
```

Logical name	Physical name	Segment name	Table name	Size
--------------	---------------	--------------	------------	------

- 9 从“配置审计” (Configure Auditing) 屏幕中，选择“配置审计” (Configure Auditing)。

`auditinit` 将再次显示“配置审计” (Configure Auditing) 菜单，并且“配置审计” (Configure Auditing) 的值显示为“是” (yes)。

- 10 重新启动 Cluster Edition，以使更改生效。

#### ❖ 创建用于审计表的设备

- 1 从“配置审计” (Configure Auditing) 屏幕中，选择“增加用于审计表的设备” (Add a Device for Audit Table(s))。

`auditinit` 将显示以下菜单：

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
```

1. sybsecurity physical device name:
2. Logical name of the device:
3. Size of the device (Meg):
4. Device size for auditing:

- 2 选择“Sybsecurity 物理设备名” (Sybsecurity Physical Device Name)。

若要创建用于审计表的设备：

- 1 输入您在第 294 页的“审计设备的预安装任务”中找到的物理设备（裸分区）的完整路径，其中 *path\_to\_partition* 是设备的裸分区的路径。

```
Enter the physical name of the device to use for the
audit database (default is " "):
```

```
/dev/path_to_partition
```

如果指定了操作系统文件，将出现以下警告：

```
WARNING: '/secret1/sybase_dr/install/aud1.dat' is a
regular file which is not recommended for a Server
device.
```

- 2 按回车键确认警告。

`auditinit` 将再次显示“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单，该菜单将显示设备的物理名称：

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device
name: /secret1/sybase_dr/install/aud1.dat
2. Logical name of the device:
3. Size of the device:
4. Device size for auditing:
```

- 3 继续进行此菜单中剩余的各项。

---

**注释** “设备大小” (Size of the Device) 的值必须等于或大于“用于审计的设备大小” (Device Size for Auditing) 的值。“用于审计的设备大小” (Device Size for Auditing) 的值必须等于设备大小。如果按照 Sybase 审计准则进行操作，则无需更改“用于审计的设备大小” (Device Size for Auditing) 中显示的值。

---

- 4 按 Ctrl+A 接受这些设置。`auditinit` 将返回到“配置审计” (Configure Auditing) 菜单并显示已创建的设备。

```
CONFIGURE AUDITING
```

- ```
1. Configure auditing: yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry
```

```
List of devices for the audit tables:
```

```
Logical name   Physical name   Segment name   Table name   Size
6.Audit_01' secret1/sybase_dr/install/aud1.dat' sysaudits_01 5
```

5 若要添加多个审计设备，请重复步骤 1–6。

可以添加多达八个设备。Sybase 建议添加三个或更多的审计表设备。

添加一个设备后，`auditinit` 将返回到“配置审计” (Configure Auditing) 菜单，并显示已创建的所有设备。

```
CONFIGURE AUDITING
```

1. Configure auditing: yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry

```
List of devices for the audit tables:
```

```
Logical name   Physical name   Segment name   Table
name          Size
6. Audit_01'   /secret1/sybase_dr/install/aud1.dat' sysaudits_01 5
7. Audit_02'   /secret1/sybase_dr/install/aud2.dat' sysaudits_02 5
```

#### ❖ 创建用于审计数据库事务日志的设备

- 1 从“配置审计” (Configure Auditing) 菜单中，选择“添加用于审计数据库事务日志的设备” (Add a Device for the Audit Database Transaction Log)。

`auditinit` 将显示“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单。

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device name:
2. Logical name of the device:
3. Size of the new device (Meg):
4. Device size for auditing:
```

- 2 选择“Sybsecurity 物理设备名” (Sybsecurity Physical Device Name)。

`auditinit` 将提示输入物理名称并提供一个缺省值（如果有）：

```
Enter the physical name of the device to use for the
sybsecurity database (default is''):
/dev/path_to_partition
```

其中 `path_to_partition` 为设备裸分区的路径。

- 3 输入物理设备的完整路径名。

如果输入了操作系统文件名，将出现以下警告：

```
WARNING: '/secret1/sybase_dr/install/auditlog' is a
regular file, which is not recommended for a Server
device.
```

- 4 按回车键确认此警告。

`auditinit` 将显示“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单，并显示为设备物理名所选的值。

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1.sybsecurity physical device name:
   /secret1/sybase_dr/install/auditlog.dat
2.Logical name of the device:
3.Size of the device:
4.Device size for auditing:
```

- 5 继续进行此菜单中剩余的各项。进行这些操作时，请注意以下几点：
  - Sybase 建议事务日志的最小大小为 2MB。
  - `auditinit` 在“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单中的“设备大小” (Size of the Device) 和“用于审计的设备大小” (Device Size for Auditing) 中都显示该大小。
  - 基于您可能想将整个设备都用于审计任务日志的假设，“用于审计的设备大小” (Device Size for Auditing) 的缺省值等于设备的大小。如果希望仅使用设备的一部分，可以编辑“设备大小” (Size of the Device) 的值。
- 6 按 Ctrl+A 接受“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单中显示的设置。

`auditinit` 将返回到“配置审计” (Configure Auditing) 菜单并显示创建的所有设备。

```
CONFIGURE AUDITING
1. Configure auditing: yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry
```

```
List of devices for the audit tables:
Logical name  Physical name  Segment name  Table
name         Size
```

```
6. Audit_01' /secret1/sybase_dr/install/aud1.dat' sysaudits_01 5
7. Audit_02' /secret1/sybase_dr/install/aud2.dat' sysaudits_02 5
8. auditlog /secret1/.../auditlog.dat logsegment syslogs 2
```

7 准备好执行审计配置后，按 Ctrl+A。auditinit 将返回到“Sybase 服务器配置” (Sybase Server Configuration) 屏幕。

8 再次按 Ctrl+A。auditinit 将提示：

```
Execute the Sybase Server Configuration now?
```

9 输入“y” (是)。

auditinit 将执行这些任务以安装审计。安装成功完成后，将显示以下消息：

```
Running task: install auditing capabilities.
.....Done
Auditing capability installed.
Task succeeded: install auditing capabilities.
Configuration completed successfully.
Press <return> to continue.
```

### 启用审计

安装审计后，只有在系统安全员使用 sp\_configure 启用审计后才能进行审计。请参见《系统管理指南：第一卷》。

#### ❖ 删除设备条目

- 1 从“配置审计” (Configure Auditing) 菜单中选择“删除设备条目” (Delete a Device Entry)。
- 2 输入要删除的设备的编号。
- 3 按回车键。

#### ❖ 更改设备条目

- 1 从“配置审计” (Configure Auditing) 菜单中选择“更改设备条目” (Change a Device Entry)。
- 2 输入要更改的设备的编号。

auditinit 将显示“添加/更改新的审计设备” (Add/Change a New Device for Auditing) 菜单，并显示有关所选设备的信息：

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device name:
    /secret1/sybase_dr/install/auditlog
2. Logical name of the device: aud.log
3. size of the new device (Meg): 5
4. Device size for auditing:5
```

- 3 选择要更改的每个剩余条目。
- 4 按 Ctrl+A 保存新的条目。

## 安装 Transact-SQL 语法的联机帮助

本节提供有关安装 Transact-SQL 语法联机帮助の説明。

### 联机语法帮助: *sp\_syntax*

`$$SYBASE/$SYBASE_ASE/scripts` 目录中包含用于安装语法帮助数据库 (sybsyntax) 的脚本。可以使用 `sp_syntax` 检索此数据。有关 `sp_syntax` 的详细信息, 请参见《参考手册: 过程》。

`scripts` 目录包含一个或多个表 18-1 中所示的 `sp_syntax` 脚本, 具体情况取决于服务器所包含的 Sybase 产品:

**表 18-1: *sp\_syntax* 安装脚本**

| 脚本                        | 产品                          |
|---------------------------|-----------------------------|
| <code>ins_syn_cl</code>   | Open Client Client-Library™ |
| <code>ins_syn_esql</code> | Embedded SQL™               |
| <code>ins_syn_os</code>   | Open Server                 |
| <code>ins_syn_sql</code>  | Transact-SQL                |

所有 Cluster Edition 安装均包含 `ins_syn_sql` 脚本。该脚本包括关于 Transact-SQL、系统过程和 Sybase 实用程序的语法信息。执行该脚本时, 将安装 sybsyntax 数据库的 SQL 部分。

取决于服务器上对 Sybase 信息的需求情况, 您可以安装这些脚本中的任意一种。执行的第一个脚本创建 sybsyntax 数据库和所需的表及索引。在第一个脚本后执行的任何脚本将会添加数据库中的现有信息。如果执行以前执行过的脚本, 则会以前安装的信息行从数据库表中删除, 然后重新安装。

---

**警告!** `ins_syn_cl` 和 `ins_syn_os` 脚本有冲突。如果同时执行两个脚本, 则会发生错误。

---

## sybsyntax 数据库的缺省设备

sybsyntax 数据库在数据库设备上要求有 3MB 的空间。缺省情况下，sybsyntax 安装脚本将 sybsyntax 数据库安装在指定的缺省数据库设备上。

如果尚未使用 sp\_diskdefault 来更改主设备（安装为缺省磁盘）的状态或指定另一个缺省设备，则脚本会将 sybsyntax 安装在主设备上。Sybase 建议不要使用这种配置，因为最好把 sybsyntax 占用的宝贵空间预留出来，以供将来扩展 master 数据库之用。

若要避免将 sybsyntax 安装在主设备上，请执行以下操作之一：

- 使用 sp\_diskdefault 将除主设备之外的设备指定为缺省设备。有关 sp\_diskdefault 的信息，请参见《参考手册：过程》。
- 按照下一节中的说明，修改打算执行的每个 sybsyntax 安装脚本，以便指定其它设备。

## 安装 sybsyntax

对于打算执行的每个 sybsyntax 安装脚本：

- 1 确定要在其中存储 sybsyntax 数据库的设备的类型（裸分区、逻辑卷、操作系统文件等）和位置。稍后将需要提供这些信息。
- 2 将原始脚本复制一份。确保可以访问该副本，以防在编辑的脚本中出现错误。
- 3 如有必要，可使用文本编辑器来编辑该脚本，将缺省设备由主设备改为在步骤 1 中创建的设备。请参见第 302 页的“sybsyntax 数据库的缺省设备”。

- 注释掉以下指定缺省设备的部分：

```
/* create the database, if it does not exist */
if not exists (select name from sysdatabases
where name = "sybsyntax")
begin
    /* create the sybsyntax table if it doesn't exist */
    /* is the space left on the default database
    devices > size of model? */
    if (select sum (high-low +1) from sysdevices where status
    & 1 = 1) - (select sum(size) from sysusages, sysdevices
    where vstart >= sysdevices.low
    and vstart <= sysdevices.high
    and sysdevices.status &1 = 1) >
    (select sum(sysusages.size) from sysusages
```

```

        where dbid = 3)
begin
    create database sybsyntax
end
else
begin
    print "There is not enough room on the default
        devices to create the sybsyntax database."
    return
end
end
end

```

- 注释掉上面整个部分后，请向脚本中添加如下所示的行：

```
create database sybsyntax on device_name
```

其中 *device\_name* 为要在其中安装 **sybsyntax** 的设备的名称。

- 4 使用如下所示的命令执行该脚本：

```
isql -Usa -Ppassword -Sservername <
$SYBASE/$SYBASE_ASE/scripts/ins_syn_sql
```

其中 *sa* 为系统管理员的用户 ID，*password* 为系统管理员的口令，*servername* 为要在其中安装数据库的 Cluster Edition。

如果已将 **DSQUERY** 环境变量设置为 *servername*，则可以使用 **SDSQUERY** 替换服务器名称。

- 5 若要确保已安装了 **sybsyntax** 数据库并且它能正确工作，请使用 **isql** 登录到安装该数据库的服务器，并执行 **sp\_syntax**。例如：

```
isql -Usa -Ppassword -Sservername

1> sp_syntax "select"
2> go
```

Cluster Edition 显示包含 “select” 单词或单词片断的命令列表。



本章描述如何使用 Cluster Edition 的错误记录功能。

| 主题                                   | 页码  |
|--------------------------------------|-----|
| <a href="#">Cluster Edition 错误记录</a> | 305 |
| <a href="#">设置错误日志的路径</a>            | 306 |
| <a href="#">管理消息</a>                 | 307 |

## Cluster Edition 错误记录

Cluster Edition 每次启动时，会将信息写入称为 Adaptive Server 错误日志的本地错误日志文件中：

`$SYBASE/$SYBASE_ASE/install/instance_name.log`

此文件：

- 存储有关每次启动尝试成功或失败的信息。
- 记录运行期间由服务器生成的错误和信息性消息。
- 在停止服务器进程之前一直保持打开状态。
- 包含 Cluster Edition 的启动信息

Cluster Edition 按以下格式在错误日志标头的前面包含实例 ID：

```
instance id: engine number : family id: process id: date time
```

---

**注释** 如果想通过减少错误日志的大小来增加可用磁盘空间，请在删除已记录消息之前停止实例。日志文件不能释放其内存空间，直到实例停止。

---

## 启用和禁用错误记录

向 Cluster Edition 错误日志添加记录这一功能始终启用。但是，当创建或修改一个特定的用户定义消息时，可以将其设置为不向日志中添加。请参见第 307 页的“记录用户定义的消息”。

## 设置错误日志的路径

在配置新的 Cluster Edition 时，安装程序会在 Sybase 安装目录中设置错误日志的位置。Backup Server 有自己的错误日志。

各服务器的错误日志的缺省位置是：

- Cluster Edition: `$$SYBASE/$$SYBASE_ASE/install/instance_name.log`
- Backup Server: `$$SYBASE/$$SYBASE_ASE/install/instance_name_back.log`

可以在启动时通过命令行重新设置 Cluster Edition 错误日志文件的名称和位置。在 `dataserver` 命令中使用 `-e` 启动参数和值来启动 Cluster Edition。但如果使用 `sybcluster` 来管理集群，就必须使用 `sybcluster 'set instance logpath'` 参数来更改每个实例的错误日志文件的位置。

---

**注释** 多个实例不能共享同一个错误日志。

---

## 设置 Cluster Edition 错误日志路径

---

**注释** Cluster Edition 安装程序不创建 `RUN_server` 文件。但如果您创建 `RUN_server` 文件，则可以为错误日志添加新位置。

如果不提供错误日志的路径，Cluster Edition 会根据以下内容添加错误日志：

- Cluster 输入文件错误日志位置。此信息存储在仲裁设备中，由 `dataserver` 命令使用。
  - 错误日志的缺省位置。如果不提供错误日志路径，则会在当前工作目录中为 `dataserver` 创建一个名为 `instance_name.log` 的日志文件。
-

可以使用 `qrmutil --errorlog` 参数或通过 `sybcluster 'set instance errorlog'` 来更改仲裁设备上存储的日志位置。

如果使用 `shell` 脚本来启动集群实例，则更改 `dataserver -e` 参数的值，以替换错误日志位置的其它设置。

## 管理消息

事件记录启用后，可以用以下方法管理其功能：

- 使用 `sp_addmessage` 或 `sp_altermessage` 控制是否将特定的用户定义消息记录在 Cluster Edition 错误日志中。  
有关 `sp_addmessage` 和 `sp_altermessage` 的完整语法，请参见《参考手册：过程》。
- 使用配置参数指定是否记录审计事件。有关用户登录 Cluster Edition 成功 (`log audit logon success`) 或失败 (`log audit logon failure`) 的审计事件。

## 记录用户定义的消息

可以指定是否将用户定义的消息记录在 Cluster Edition 错误日志中。Cluster Edition 能让您确定是否记录：

- 新消息 (`sp_addmessage`)。
- 已存在的消息 (`sp_altermessage`)。

有关这些命令及其参数的详细信息，请参见《参考手册：过程》中的 `sp_addmessage` 和 `sp_altermessage`。

## 新消息

在 `sysusermessages` 中添加一个新的用户定义消息时，需在 `sp_addmessage` 中包含 `with_log` 选项。此参数可将 Cluster Edition 设置为在消息每次出现时都将其记录下来。

## 已存在的消息

在 `sp_altermessage` 中包含 `with_log` 选项，以便更改现有的用户定义消息。此参数将变更该消息的报告状态：

- TRUE — 启用记录。
- FALSE — 禁用记录。

## 记录审计事件

缺省情况下，Cluster Edition 不记录审计事件。但是，可以使用 `sp_configure` 参数指定 Cluster Edition 是否在 Cluster Edition 错误日志中记录审计事件（如登录）。

可能的参数和值有：

- `log audit logon success` 为 1 — 允许记录成功的 Cluster Edition 登录：

```
sp_configure "log audit logon success", 1
```

- `log audit logon failure` 为 1 — 允许记录失败的 Cluster Edition 登录：

```
sp_configure "log audit logon failure", 1
```

- 两个参数之一为 0 — 禁用对该消息类型的记录：

```
sp_configure "log audit logon success", 0  
sp_configure "log audit logon failure", 0
```

有关 `sp_configure` 的详细信息，请参见《系统管理指南：第一卷》。

## 设置网络通信

集群中的每个实例都可以与网络上的其它 Adaptive Server、Open Server 应用程序和客户端软件通信。通过远程过程调用，客户端可与一台或多台服务器通信，服务器可与其它服务器通信。

| 主题                                             | 页码  |
|------------------------------------------------|-----|
| <a href="#">Cluster Edition 如何确定要使用的目录服务条目</a> | 310 |
| <a href="#">客户端如何使用目录服务</a>                    | 311 |
| <a href="#">创建目录服务条目</a>                       | 311 |
| <a href="#">支持的目录驱动程序</a>                      | 311 |
| <a href="#">interfaces 文件的内容</a>               | 312 |
| <a href="#">异构环境和同构环境</a>                      | 313 |
| <a href="#">了解 interfaces 文件的格式</a>            | 314 |
| <a href="#">创建主 interfaces 文件</a>              | 316 |
| <a href="#">配置多个网络的接口文件</a>                    | 318 |
| <a href="#">IPv6 支持</a>                        | 321 |
| <a href="#">故障排除</a>                           | 324 |

目录服务包含有关服务器网络位置的信息。目录服务包含网络中所有 Adaptive Server、Backup Server 和其它服务器产品的条目。

在 Sybase 客户端/服务器环境中，如果客户端知道服务器在网络中的驻留位置且服务器支持客户端的语言或字符集，则此客户端能够与实例连接。当客户端启动一个连接时，它查看其目录服务以确定目标服务器的网络位置。

目录服务列出每个服务器的名称和地址，包括 Backup Server 和 XP Server。如果正使用某客户端程序，并要与特定的服务器连接，则客户端程序会在目录服务中查找该服务器名称并连接到该服务器。

服务器也需要网络信息。服务器启动时，它查看其 interfaces 文件以确定在哪里监听客户端连接请求。此外，每个实例还能在对其它实例进行远程过程调用时充当客户端。

表 20-1: interfaces 文件任务和主题所在的位置

| interfaces 文件的类型 | 任务或主题                      | 请参见                                                                                                                         |
|------------------|----------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| UNIX 服务器或客户端     | 添加多个 Cluster Edition 安装的条目 | Adaptive Server Configuration Guide (《Adaptive Server 配置指南》)                                                                |
|                  | 为多个安装创建主 interfaces 文件     | <a href="#">第 316 页的“创建主 interfaces 文件”</a>                                                                                 |
|                  | 配置多个网络                     | <a href="#">第 318 页的“配置多个网络的接口文件”</a>                                                                                       |
|                  | 参考信息                       | <a href="#">第 314 页的“了解 interfaces 文件的格式”</a>                                                                               |
| PC 客户端           | 配置客户端                      | 所用平台的《安装指南》                                                                                                                 |
|                  | 有关高级任务的参考信息和说明             | 所用 PC 客户端平台的 Open Client and Open Server Programmer's Supplement (《Open Client 和 Open Server 程序员补充说明》), 或适当的 Open Client 文档 |
| 未列出的客户端平台        | 有关高级任务的配置、参考信息和说明          | 所用 PC 客户端平台的 Open Client and Open Server Programmer's Supplement (《Open Client 和 Open Server 程序员补充说明》), 或适当的 Open Client 文档 |

## Cluster Edition 如何确定要使用的目录服务条目

每个实例都使用目录服务来确定监听客户端的地址。启动实例时，它会执行以下步骤：

- 1 它查找在 `-s` 命令行选项中提供的服务器名。如果未在命令行中提供服务器名：
- 2 它通过检查 `DSLISTEN` 环境变量的值来确定自己的名称。如果未设置 `DSLISTEN` 环境变量，则它假定服务器名为 `SYBASE`。
- 3 在目录服务中查找名称与在以上步骤中找到的名称匹配的条目。
- 4 它使用所找到的目录服务条目提供的网络信息来监听客户端连接。

## 客户端如何使用目录服务

客户端连接到服务器时，它：

- 通过编程方式或引用 `DSQUERY` 环境变量来确定服务器的名称。如果应用程序的用户没有设置 `DSQUERY`，服务器名的运行时值会缺省为 `SYBASE`。
- 在目录服务中查找名称与服务器名相匹配的条目。
- 使用目录服务条目提供的网络信息连接服务器。如果客户端第一次未连接成功，则根据目录服务中给出的延迟和重试次数再进行尝试。如果没有找到任何匹配的条目，则将错误消息写入到客户端的标准错误文件中。如果支持多个网络，客户端使用服务器的第二个网络地址条目中的信息尝试连接。

`Open Client` 文档更详尽地讨论了客户端连接。请参见所用客户端平台的 `Open/Client Programmer's Supplement`（《`Open/Client` 程序员补充说明》）或相应的 `Open/Client` 文档。

## 创建目录服务条目

当您创建集群时，`sybcluster` 实用程序会为集群以及每个实例创建目录服务条目。您也可以使用以下 `Sybase` 实用程序编辑目录服务中的网络信息：

- `dsedit` — 一个 GUI 实用程序。
- `dscp` — 一个 UNIX 命令行实用程序。

有关使用这些实用程序的详细信息，请参见《实用程序指南》。

## 支持的目录驱动程序

UNIX 支持三种驱动程序：

- 接口驱动程序
- 轻量目录服务驱动程序。
- 分布式计算环境 (DCE) 提供的 Cell 目录服务 (CDS)

本章的这一剩余部分介绍 *interfaces* 文件，并为每个支持的 UNIX 平台提供具体的配置信息。有关 LDAP 驱动程序、Cell 目录服务的信息，以及有关 *interfaces* 文件和 LDAP 目录服务之间的比较，请参见所用平台的 Open Client/Server Configuration Guide（《Open Client/Server 配置指南》）。

## interfaces 文件的内容

*interfaces* 文件包含有关网络上所有服务器的网络信息，其中包括实例、Backup Server、XP Server 以及任何其它服务器应用程序，如 Replication Server 和任何其它 Open Server 应用程序。

文件中的网络信息包括服务器名、主机的网络名或地址，以及服务器监听查询时所在的端口、对象或套接字号（取决于网络协议）。请参见第 314 页的“了解 *interfaces* 文件的格式”。

*interfaces* 文件中的每个条目均可包含两种类型的行：

- **master** 行，服务器应用程序用它们在网络上监听查询。此信息称为 *监听器服务*。
- **query** 行，客户端应用程序用它们连接网络上的服务器。此信息称为 *查询服务*。

对于服务器来说，**master** 行和 **query** 行中包含的网络信息可能是一致的，因为服务器在客户端用来请求连接的同一端口上监听连接请求。

服务器在其 *interfaces* 文件中需要 **master** 行。当服务器充当其它服务器的客户端时，这些其它服务器就需要 **query** 行。

客户端的 *interfaces* 文件不需要 **master** 行。仅有 **query** 行即可正常工作。

### 如果站点有多个安装

如果您安装了多个 Adaptive Server 或 Cluster Edition，则每个服务器的 *interfaces* 文件都应包含有关网络上所有服务器的信息。

如果所有服务器产品都运行在同一个平台中，则可创建一个主 *interfaces* 文件，并将此文件复制到每台计算机中。请参见第 316 页的“创建主 *interfaces* 文件”。

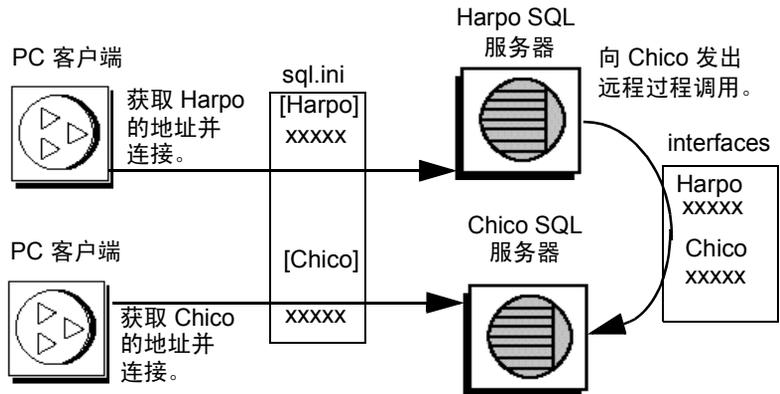
如果主机支持多个网络，则参见第 318 页的“配置多个网络的接口文件”。

## 异构环境和同构环境

可在同一平台上或不同平台上运行 Cluster Edition 和客户端。

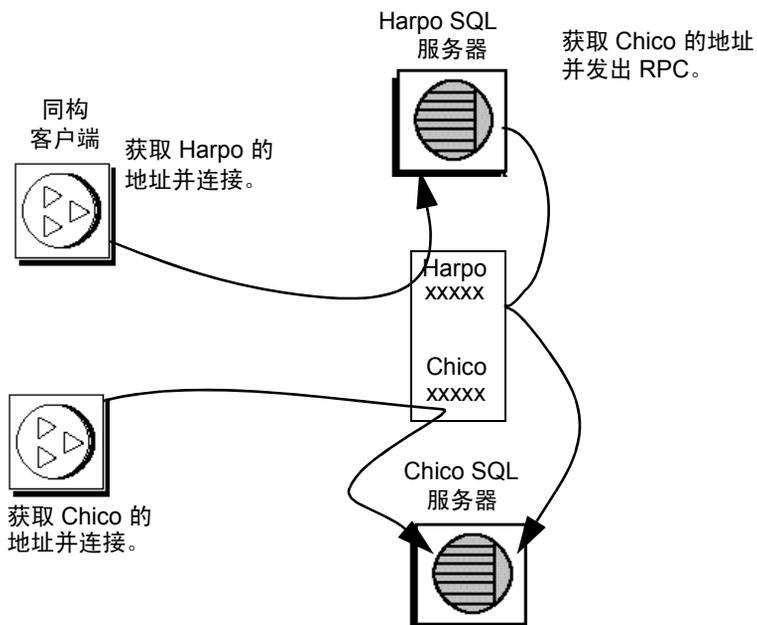
如果在不同的平台上运行，则每个平台可能要求其 *interfaces* 文件使用不同的格式和配置。图 20-1 说明了 PC 客户端如何使用其 *interfaces* 文件 (*sql.ini*) 中的网络信息来连接到在 UNIX 上运行的实例，并说明在远程过程调用期间，实例如何使用其 *interfaces* 文件来连接到其它服务器。

图 20-1: 在异构环境中建立网络连接



如果客户端和服务器都运行在 UNIX 下，则同一 *interfaces* 文件对它们都有效。图 20-2 说明运行在同构环境中的客户端和实例如何使用 *interfaces* 文件的副本来建立连接。由于这两个实例运行在同一操作系统下，因此它们可使用同一 *interfaces* 文件或同一文件的完全一致的副本。

图 20-2: 在同构环境中建立网络连接



## 了解 interfaces 文件的格式

下列规则适用于 interfaces 文件条目的格式：

- 每个实例仅有一个条目，但条目中可以有多行。
- 在 *servername* 行后面的每一行必须以空格或制表符开始。
- 必须用单个空格将行中的每个元素分开。
- 每个条目用空行分开。
- 可通过在行首添加磅字符 (#) 并在行尾添加分行符来为 interfaces 文件添加注释。

有两种 interfaces 文件条目格式：TLI 和 TCP。

TLI 样式的条目如以下所示：

```
servername retry_attempts delay_interval<newline>
<tab>service_type api protocol device address filter<newline>
<tab>ha_failover servername<newline>
```

TCP 样式的条目如以下所示：

```
servername retry_attempts delay_interval<newline>
<tab>service_type protocol network machine port filter<newline>
<tab>ha_failover servername<newline>
```

## interfaces 文件条目的组成部分

表 20-2 说明了 interfaces 文件条目的组成部分。

**表 20-2: interfaces 文件的组成部分**

| 组成部分                       | 值                                                                                                                                                                           |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>servername</i>          | 实例或 Backup Server 的名称。服务器名的要求为： <ul style="list-style-type: none"> <li>名称长度不能超过 30 个字符。</li> <li>第一个字符必须是字母（ASCII 的 a 到 z、A 到 Z）。</li> <li>随后的字符必须是字母、数字或下划线（_）。</li> </ul> |
| <i>retry_attempts</i> （可选） | 希望客户端在首次失败后尝试连接服务器的次数。缺省值为 0。                                                                                                                                               |
| <i>delay_interval</i> （可选） | 连接尝试间隔的秒数。缺省值为 0。                                                                                                                                                           |
| <i>service_type</i>        | 由条目定义的服务类型。必须为下列之一： <ul style="list-style-type: none"> <li>master</li> <li>query</li> </ul>                                                                                 |
| <i>api</i>                 | 可用于网络的应用程序编程接口。支持的值为 tli。                                                                                                                                                   |
| <i>protocol</i>            | 网络协议的名称。可用的协议类型有： <ul style="list-style-type: none"> <li>TCP/IP，用字母“tcp”表示</li> </ul>                                                                                       |
| <i>network</i>             | 网络的名称；当前未被 Cluster Edition 使用。sybcluster 输入“ether”作为占位符。                                                                                                                    |
| <i>host</i>                | 服务器主机的网络名称或地址。 <ul style="list-style-type: none"> <li>对于 TCP/IP，可使用主机名或 Internet 地址。条目的最大大小为 32 字节。若要确定计算机的主机名，请登录到该计算机并输入：<br/>/bin/hostname</li> </ul>                    |
| <i>machine</i>             | 服务器主机的网络名称或地址。 <ul style="list-style-type: none"> <li>可使用主机名或 Internet 地址。条目的最大大小为 32 字节。若要确定计算机的主机名，请登录到该计算机并输入：<br/>/bin/hostname</li> </ul>                              |

| 组成部分                         | 值                                                                                                                                                                                                                                                                                                                                                                                                                        |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>device</i>                | <p>网络设备的结束点。</p> <p>对于 TCP 网络，设备会因网络软件供应商的不同而不同。检查特定供应商文档中的设备名。您的网络可能会提供多个与 TCP 协议组中不同协议相对应的流设备。选择 TCP 流设备。常见的 TCP 流设备为 <i>/dev/tcp</i>。</p>                                                                                                                                                                                                                                                                             |
| <i>address</i> （用于 TLI 协议条目） | <p>地址由以下部分组成：</p> <ul style="list-style-type: none"> <li>• 地址前缀，“\x”代表 TLI。</li> <li>• 网络类型，始终为 0002。</li> <li>• 转换成四位数字的端口号，用十六进制表示。必须是 1025 与 65535 之间的唯一号码。检查网络中每台计算机上的 <i>/etc/services</i> 文件，查看正在使用的端口号。在标记为“Sybase specific services”（Sybase 特定服务）的 <i>/etc/services</i> 新部分中输入实例端口号。即使不创建此条目，操作系统也能正常工作，但如果文件中含有端口号，则可以防止其他用户使用该端口号。</li> <li>• 转换为八位数字的主机 IP 网络节点地址，用十六进制表示。</li> <li>• 尾随零，可选，16 位。</li> </ul> |
| <i>port</i>                  | <p>1025 与 65535 之间的唯一端口号。检查网络中每台计算机上的 <i>/etc/services</i> 文件，查看正在使用的端口号。在标记为“Sybase specific services”（Sybase 特定服务）的 <i>/etc/services</i> 新部分中输入实例端口号。即使不创建此条目，操作系统也能正常工作，但如果文件中含有端口号，则可以防止其他用户使用该端口号。</p>                                                                                                                                                                                                              |
| <i>ha_failover</i>           | <p>在目录服务或 <i>interfaces</i> 文件中为高可用性创建的条目。</p>                                                                                                                                                                                                                                                                                                                                                                           |
| <i>filter</i>                | <p>Cluster Edition 支持将安全套接字层 (SSL) 用作过滤器，它附加到目录服务的 <i>master</i> 行和 <i>query</i> 行。SSL 是用于保护敏感信息的传输的标准。有关 SSL 的详细信息，请参见第 35 页的“在集群环境中使用 SSL”。</p>                                                                                                                                                                                                                                                                        |

## 创建主 *interfaces* 文件

主 *interfaces* 文件包含网络中所有 Sybase 服务器的条目。它可用于连接到网络的每台服务器和客户端。通过分配主 *interfaces* 文件的副本，可确保网络中所有的 Sybase 产品能够互相交互。

在同构网络环境中，维护 *interfaces* 文件一致性的最简便方法是分配一个含有所有实例条目的 *interfaces* 文件（主文件）的副本。

可对文件的一个版本进行所有更改，然后将更新的主文件复制到所有相应的 Sybase 目录。

可采用下述两种方法之一生成主文件：

- 使用 `dsedit` 或 `dscp`
- 使用文本编辑器

## 使用 `dsedit` 或 `dscp` 创建主 `interfaces` 文件

可使用 `dsedit` 或 `dscp` 实用程序创建主 `interfaces` 文件，然后可将其分配到所有服务器。如果您不是经验丰富的 Sybase 用户，可能会感觉到使用 `dsedit` 或 `dscp` 比使用文本编辑器更容易。使用 `dsedit` 或 `dscp` 也能确保 `interfaces` 文件在格式上保持一致。

若要用 `dsedit` 或 `dscp` 创建主 `interfaces` 文件：

- 1 选择包含最完整、最新信息的 `interfaces` 文件。
- 2 在最新的 Sybase 安装中开始一个 `dsedit` 或 `dscp` 会话，以编辑此 `interfaces` 文件。
- 3 添加此文件中未列出的任何实例或 Backup Server 的条目。

有关使用这些实用程序的详细信息，请参见《实用程序指南》。

## 使用文本编辑器创建主 `interfaces` 文件

若要从几个单独的 `interfaces` 文件构造一个主 `interfaces` 文件：

- 1 并置单独的 `interfaces` 文件。
- 2 制作此文件的一份副本。
- 3 使用 ASCII 文本编辑器修改并置文件的副本。

---

**注释** 手动编辑 `interfaces` 文件时，确保每一条目中第一行后面的每行都以 `<tab>` 符开始。

---

下列元素在生成的文件中必须正确且唯一：

- `servername` — `interfaces` 文件中的每个服务器条目必须是唯一的。每个集群以及集群中的每个实例所对应的服务器名条目都必须在 `interfaces` 文件中是唯一的。
- 主机的网络名或地址与实例的端口号或对象号的组合。

- 如果在网络中只有一台计算机时创建原始 *interfaces* 文件，则其条目可能用文字 “loghost” 代替计算机名（地址）。如果 *loghost* 存在，则用计算机名替换它。

## 配置多个网络的 *interfaces* 文件

在某些平台中，Cluster Edition 可用于多个网络。这可以使实例通过多个网络接口监听客户端。必须将每个网络接口的条目添加到 *interfaces* 文件中。

### 配置多个网络处理程序的服务器

若要配置多个网络监听器：

- 1 在操作系统的主机数据库中为每个网络接口定义一个唯一的主机名。
- 2 在 *interfaces* 文件中，使用文本编辑器为实例添加 “master” 行的副本；您希望服务器监听的每个附加接口各有一个副本。
- 3 在每行中包含一个唯一的主机名，以便为每个网络接口配置网络处理程序。
- 4 接口内的端口号不必相同，但它们可以相同。它们的命名和数值范围的规则与主网络接口所遵循的规则相同。

### 多个网络处理程序的示例 *interfaces* 文件

下例说明了具有两个网络接口的实例的 *interfaces* 文件。在公司网络中，服务器主机称为 *SERV\_CORPNET*，在工程网络中称为 *SERV\_ENGNET*。

```
# PRODUCTION server with two network listeners
PRODUCTION<tab>3<tab>3<newline>
<tab>master tcp ether SERV_CORPNET 4559
<tab>query tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
```

当实例重新启动时，它为与服务器的 *DSLISTEN* 值对应的条目中的每个 *master* 行产生一个网络处理程序进程。将等同处理在每个接口上建立的连接，它们是对等的。

## 配置客户端连接

当实例的客户端扫描 `interfaces` 文件来获取服务器名时，客户端将它遇到的第一个“`query`”条目用于服务器条目。这使得将客户端配置成使用多个网络连接的操作没有配置服务器端口的操作那么简单。有两种选择：

- 对所有客户端使用相同的 `DSQUERY` 名称。不同计算机上的 `interfaces` 文件包含不同的网络名。
- 对所有客户端使用不同的 `DSQUERY` 名称。所有计算机上的 `interfaces` 文件都是相同的，但它们包含多个 `DSQUERY` 名称。

### 使用一个独立于网络的 `DSQUERY` 名称

如果统一的客户端 `DSQUERY` 命名很重要，可在 `interfaces` 文件中对客户端网络地址进行必要的更改。可在每个网络中的客户端文件服务器上安装单独的 Sybase 安装目录和不同的 `interfaces` 文件，以使用户能够连接到正确的网络地址。可在所有网络中为所有客户端保持一个 `DSQUERY` 名称，并相应更改每个网络 `interfaces` 文件，而不更改客户端使用的 `DSQUERY` 名称。

此方法假定：

- 您可对 Sybase 安装客户端在每个网络中能访问的内容进行完全控制。
- （最低限度）不会在不同网络中的 Sybase 安装之间共享或复制 `interfaces` 文件。

`interfaces` 文件与以下有关“工程”网络的示例类似：

```
PRODUCTION<tab>3<tab>3<newline>
<tab>query tcp ether SERV_ENGNET 5470
<tab>master tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
```

`interfaces` 文件与以下有关“公司”网络的示例类似：

```
PRODUCTION<tab>3<tab>3<newline>
<tab>query tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
```

每个文件名中的“`query`”行是不同的，这取决于所用的网络。

完整的“`master`”条目在两个文件中都存在。这是允许的，因为只有实例会使用“`master`”行。假定服务器主机可看到这两个网络（两个主机名可互换），则实例用哪一个 `interfaces` 文件启动都无关紧要。

## 使用不同的 DSQUERY 名称

若要对每个网络监听器使用不同的 DSQUERY 名称：

- 1 选择其它服务器名。

可并置原始服务器名和网络名。例如，如果您的服务器名为 PRODUCTION，则可选择名称 PRODUCTION\_network1 和 PRODUCTION\_network2。

- 2 执行下列操作之一：

- 对于 PC 客户端，使用 `sqledit` 为服务器创建多个 `sql.ini` 文件条目，每个网络一个。在下例中，为 PRODUCTION\_network1 和 PRODUCTION\_network2 各创建一个条目。请参见针对您的客户端平台的 Open Client 文档。
- 对于 UNIX 客户端，可用 ASCII 文本编辑器编辑 `interfaces` 文件。从服务器的 `interfaces` 文件中，将每个网络的服务器名行和 “master” 行复制到客户端 `interfaces` 文件中。为每个条目添加合适的服务器名，并将 “master” 更改为 “query”。

每个网络上的客户端必须使用与客户端所在网络相对应的 DSQUERY 值。在下例中，可使用 PRODUCTION\_network1 或 PRODUCTION\_network2。

```
# Client entry for PRODUCTION on network1
PRODUCTION_network1<tab>3<tab>3<newline>
<tab>query tcp ether serv_corpnet 4559
# Client entry for PRODUCTION on network2
PRODUCTION_network2<tab>3<tab>3<newline>
<tab>query tcp ether serv_engnet 5479
```

## 配置查询端口备份

多个网络接口的另一个用途是在网络发生故障时提供备份。如果客户端通过两个网络连接到服务器，则在第一个网络发生故障时，客户端可通过第二个网络建立连接。

若要对实例进行查询端口备份配置：

- 1 在 `interfaces` 文件的服务器条目中放置多个 “master” 行和 “query” 行。
- 2 一个实例在两个端口上监听连接。为了查找主机名和端口号以连接实例，客户端会按顺序尝试每个 “query” 行上的端口，直至建立连接。

下例说明如何配置仅在正常连接失败的情况下使用的备用网络。主网络是“公司网络”，备用网络是“工程网络”。

```
# PRODUCTION server with two network listeners
PRODUCTION<tab>3<tab>3<newline>
<tab>master tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
<tab>query tcp ether SERV_CORPNET 4559
<tab>query tcp ether SERV_ENGNET 5479
```

- 3 按 Open Client 文档中的说明，用适当的多个“query”条目配置 PC 客户端 `interfaces` 文件。对于同构环境中的客户端 `interfaces` 文件，可以将实例的全部 `interfaces` 文件条目复制到客户端 `interfaces` 文件中。
- 4 仅在禁用公司网络的情况下，或者在主机上的公司网络接口出现故障或因网络相关的故障造成接口关闭的情况下，才进行第二个端口的连接。

## IPv6 支持

Cluster Edition 支持 IPv6 技术。

### 了解 IPv6

IPv6 寻址术语：

- 链接本地地址 — 仅在单个链接上可用的 IPv6 地址。
- 站点本地地址 — 可在单个站点内使用的 IPv6 地址。
- 全球地址 — 可以在全球 Internet 上使用的 IPv6 地址。

IPv6 应用程序类型：

- IPv6 不识别 — 一种无法处理 IPv6 地址的应用程序。
- 可识别 IPv6 — 一种可以与没有 IPv4 地址的节点通信的应用程序。在某些情况（例如，API 隐藏实际地址的内容和格式时）下，这可能对应用程序是透明的。

- 支持 IPv6 — 一种除了可识别 IPv6 外还采用了一些 IPv6 功能的应用程序。
- IPv6 必需 — 一种要求一些 IPv6 功能且不能在 IPv4 上运行的应用程序。

IPv6 基础结构:

## IPv6 基础结构

双堆栈基础结构同时实现了 IPv4 和 IPv6。这是推荐的基础结构实现，以便将 Cluster Edition 用作可识别 IPv6 的服务器。

Sybase 应用程序可识别 IPv6。所有将 Cluster Edition 和 Open Client/Server 组件变为可识别 IPv6 的代码都是使用 IETF 设计的基元完成的，请参见“Creating or converting for IPv6-aware applications”（可识别 IPv6 的应用程序的创建或转换）。下表列出了平台运行时要求以及特定产品及其发行版本:

**表 20-3: IPv6 支持**

| 平台                       | Cluster Edition<br>对 IPv6 的识别 | Open Client/Server<br>对 IPv6 的识别 |
|--------------------------|-------------------------------|----------------------------------|
| Sun Solaris 8 32 位和 64 位 | 12.5.3a 和 15.0                | 12.5 和 15.0                      |
| HP-UX 11i(v1) 32 位和 64 位 | 12.5.3a 和 15.0                | 12.5 和 15.0                      |
| Microsoft Server 2003    | 12.5.3a 和 15.0                | 12.5 和 15.0                      |
| Linux RHEL 3.0           | 15.0                          | 12.5 和 15.0                      |

许多基于 Open Client/Server 的 Sybase 产品，如 XP Server、Backup Server、Replication Server 和 Open Switch，由于存在分层的 Open Client 传输控制层 (CTlib->NETlib)（这对于网络套接字操作而言是可识别 IPv6 的），因此它们都自动变为可以识别 IPv6。特别需要注意的是，任何基于 DBlib 的 Open Client 产品都不可识别 IPv6。

对于 Cluster Edition 来说，变为可识别 IPv6 是一个很复杂的问题，因为服务器中的某些组件是第三方组件，还不能识别 IPv6。为便于理解这种情况对 Cluster Edition 的影响，下表列出了就上述平台/发行版本而言可识别 IPv6 的 Cluster Edition 的所有功能机制:

- 连接处理程序
- RPC 机制
- Job Scheduler 任务/代理会话连接
- 网络主机 API

- 对 `sybsendmsg` 的 UDP Message 支持
- 组件集成服务连接
- 主机/名称解析
- XML URL 连接处理程序
- 客户端地址数据的审计

Cluster Edition 中的下列功能机制不支持 IPv6。Cluster Edition 中的这些机制不能识别 IPv6。

- Java 支持
- 许可证管理服务器
- LDAP 驱动程序
- 集群中多个实例之间的专用互连。

## 启动可识别 IPv6 的 Cluster Edition

缺省情况下，Cluster Edition 不能识别 IPv6。必须使用跟踪标志 7841 来启动 Cluster Edition，才能使其识别 IPv6。通过这种方式，Cluster Edition 便可以确定 IPv6 的可用性，并能够识别 IPv6。

请咨询您的网络或 IT 专家，以便正确配置您的平台和网络基础结构，以获得 IPv6 支持。

还有一个跟踪标志，即 7815，它可在启动 Cluster Edition 时设置，用于捕获并记录地址连接请求和主机/名称查找结果。

IPv6 Cluster Edition 跟踪标志：

- T7841 — 使 Cluster Edition 可识别 IPv6。
- T7815 — 报告所有 Cluster Edition IPv4 和 IPv6 客户端地址连接请求。

在启动 Cluster Edition 以执行可识别 IPv6 的操作之前，请确保基础结构已正确设置。在正确配置了操作系统之后，即可配置和启用 IPv6 connection handler。配置和启用 IPv6 connection handler 需要另外添加一个 DCL 条目。通常，单个 Cluster Edition 配置最多可在 DCL 内指派 32 个 connection handler。

例如，如果您的站点本地设置带有两个在名称服务器设置下管理的域名：

```
sybase.com - being responsible for all IPv4 networking applications
v6.sybase.com - being responsible for all IPv6 networking applications
```

如下所示的 DCL 条目能让 Cluster Edition 在主机 “revival” 端口 17100 上启动名为 “SYBASE” 的集群：

```
SYBASE
master tcp ether revival.sybase.com 17100
query tcp ether revival.sybase.com 17100
master tcp ether revival.v6.sybase.com 17100
query tcp ether revival.v6.sybase.com 17100
```

在上面的示例中，当启动可识别 IPv6 的 Cluster Edition 时，它会创建两个 **connection handler**。一个处理程序监听端口 17100 上是否有传入的 IPv4 客户端连接请求，另一个监听端口 17100 上是否有传入的 IPv6 客户端连接请求。

## 故障排除

本节介绍如何纠正一些可导致服务器无法启动的常见情况。

### 服务器启动失败

如果服务器启动失败并出现以下消息，则 `interfaces` 文件中指定的端口号可能正在使用中：

```
00:00000:00002:2003/09/22 12:37:23.63 kernel network name SERV_CORPNET, type
ether, port 4559, filter NONE
00:00000:00002:2003/09/22 12:37:23.65 kernel ninit: bind, Address already in
use
00:00000:00002:2003/09/22 12:37:23.68 server Error: 1602, Severity: 18, State:
2
00:00000:00002:2003/09/22 12:37:23.68 server Unable to initialize network 0
00:00000:00002:2003/09/22 12:37:23.68 kernel ninit: All master network
listeners have failed. Shutting down.
00:00000:00002:2003/09/22 12:37:23.68 kernel ueshutdown: exiting
00:00000:00016:2003/09/22 16:11:35.46 server SQL Server shutdown by request.
```

#### ❖ 研究端口的指派

- 1 查看 `interfaces` 文件以确定指派给服务器的端口号。
- 2 通过输入以下命令来确定是否有其它进程正在使用同一端口号：

```
netstat -a
```

如果端口号在 `netstat` 输出中显示为本机地址，则不能将此端口用于服务器。另一进程已在使用此端口。

- 3 若要检验服务器端口是否在使用，请手动启动服务器。

如果指派给服务器的端口号已在使用，则服务器不能启动。

有关手动启动服务器的信息，请参见所用平台的安装文档和《实用程序指南》。

#### ❖ 如果旧的服务器进程仍在使用此端口号

- 1 执行下列操作之一：
  - 使用操作系统的 `kill` 命令终止此进程。
  - 修改 `interfaces` 文件，将另一个端口号用于服务器。
- 2 手动启动服务器以确认端口号是否可用。

有关手动启动服务器的信息，请参见所用平台的安装文档和《实用程序指南》。

## 执行 ESP 时出错

如果尝试执行 ESP（扩展存储过程），则可能看到以下错误：

```
00:00000:00008:1997/09/10 12:52:53.03 kernel XP Server failed to start. Try bringing up XP Server manually. Check SQL Server documentation for more information on how to bring XP Server up.
```

由于端口号可能正被另一进程使用，XP Server 无法启动。使用上节中描述的 `netstat` 命令来确定为 XP Server 指定的端口号是否正在使用。

如果查明没有其它进程在使用相同的端口号，则执行早前尝试过的 ESP。XP Server 应自动启动。

如果您找到一个使用同一端口号的进程，则可以执行下列操作之一：

- 更改 `interfaces` 文件以将新的端口号用于 XP Server。
- 使用指派给 XP Server 的端口号停止此进程。

重新启动 Cluster Edition，然后执行早前尝试过的 ESP。XP Server 应自动启动。



# 索引

## 符号

`SSYBASE` 环境变量作为缺省服务器名 317  
`/etc/services` 文件 316

## 英文

### Adaptive Server

错误日志路径 306  
客户端通信 309  
排序顺序 273  
缺省排序顺序 273  
缺省字符集 273  
语言, 改变 273  
在 *interfaces* 文件中命名 315  
之间的转换, 与客户端 278  
字符集, 更改 272, 273  
`allow updates` 配置参数, 设置 213  
`alter database` 命令 158  
**alter database**, 用于私有设备 137  
ASE 插件  
JINI 发现方法 210  
UDP 发现方法 210  
创建共享数据库设备 218  
从集群中删除实例 217  
断开与集群的连接 209  
更改服务器发现设置 209  
故障切换实例 231  
关闭集群 215  
关闭实例 218  
管理本地临时数据库 219–220  
管理多个临时数据库 219–223  
管理负载 223–235  
管理共享磁盘集群 207–218  
管理逻辑集群 227–235  
集群属性 211

将负载配置文件与逻辑集群关联 226  
连接到集群 208  
连接属性 236  
逻辑集群的负载配置文件 232  
逻辑集群的负载状态 233–235  
逻辑集群的路由 233  
逻辑集群属性 229–233  
启动集群 215  
启动实例 218  
启用 **Unified Agent** 209  
删除服务器组 216  
删除负载配置文件 226  
删除集群 216  
删除逻辑集群 229  
设置配置参数 212  
添加负载配置文件 224  
添加故障切换实例 232  
添加逻辑集群 229, 231  
添加用户创建的本地临时数据库 221  
添加用户创建的全局临时数据库 221  
系统临时数据库 220  
显示集群的状态 216  
显示日志空间 214  
向集群中添加实例 217  
向组中添加临时数据库 221–223  
阈值 227  
指标权值 227  
ASE 插件中显示的日志空间 214  
`audit queue size`, 设置 213  
Backup Server  
配置 283, 286  
字符集 287  
`charsets` 目录 279, 283  
关于 284  
Cluster Edition  
displaying the log space in ASE 插件 214

- DSS/ 报告应用程序的方案 16
- OLTP 应用程序的方案 16
- 断开连接 209
- 更改服务器发现设置 209
- 故障切换方案 15
- 管理 207–218
- 配置 213
- 配置方案 14
- 使用 ASE 插件断开连接 209
- 使用 ASE 插件关闭集群 215
- 使用 ASE 插件管理本地临时数据库 219–220
- 使用 ASE 插件管理多个临时数据库 219–223
- 使用 ASE 插件连接到 208
- 使用 ASE 插件启动集群 215
- 使用 ASE 插件删除集群 216
- 使用 ASE 插件设置配置参数 212
- 显示集群属性 211
- 新增客户端技术 17
- 优点 11
- 在 ASE 插件中创建共享数据库设备 218
- 在 ASE 插件中从集群删除实例 217
- 在 ASE 插件中负载配置文件 223
- 在 ASE 插件中故障切换实例 231
- 在 ASE 插件中关闭实例 218
- 在 ASE 插件中管理负载 223–235
- 在 ASE 插件中管理逻辑集群 227–235
- 在 ASE 插件中将负载配置文件与逻辑集群关联 226
- 在 ASE 插件中路由的属性 236
- 在 ASE 插件中逻辑集群的负载配置文件 232
- 在 ASE 插件中逻辑集群的负载状态 233–235
- 在 ASE 插件中逻辑集群的路由 233
- 在 ASE 插件中逻辑集群属性 229–233
- 在 ASE 插件中启动实例 218
- 在 ASE 插件中删除服务器组 216
- 在 ASE 插件中删除负载配置文件 226
- 在 ASE 插件中删除逻辑集群 229
- 在 ASE 插件中系统临时数据库 220
- 在 ASE 插件中显示集群的状态 216
- 在 ASE 插件中阈值 227
- 在 ASE 插件中指标权值 227
- Cluster Edition 和非集群 Adaptive Server 11
- common.loc* 文件 284
- create database** 137
- create database**, 用于私有设备 137
- CS\_DS\_RAND\_OFFSET 属性 23
- CS\_HAFAILOVER 属性 21
- CS\_NOREDIRECT 属性 23
- CS\_PROP\_EXTENDEDFAILOVER 属性 28
- CS\_PROP\_MIGRATABLE 属性 20
- CS\_PROP\_REDIRECT 属性 23
- CS\_RET\_FAILOVER 属性 22
- CS\_RET\_HAFAILOVER 属性 28
- CTLIB API 调用, 为故障切换进行修改 21
- dbcc** 错误消息 268
- dbcc** 消息中的错误 268
- disk init** 命令 133
- disk refit**, 对集群中的私有设备执行 137
- disk refit**, 具有私有设备时的执行过程 137
- disk refit**, 用于私有设备 137
- disk reinit** 命令 133
- dscp** 317
- dscp** 实用程序
  - 创建主 *interfaces* 文件 316
- dsedit** 实用程序
  - 创建主 *interfaces* 文件 317
  - 添加 LDAP 服务器 43
- DSQUERY 环境变量
  - 多个网络, 使用不同的值 320
  - 客户端连接和 319
  - 描述的 311
  - 命名 319
- DTM 158
  - ASTC 机制 161
  - xact\_conmigrate\_check* 函数 163
  - xact\_owner\_instance* 函数 163
  - 处理错误 162
  - 处理实例故障 160
  - 非所有者实例 160
  - 集群作为资源管理器 159
  - 连接迁移 161
  - 配置问题 162
  - 特定于集群的问题 159
- HAFAILOVER 属性 20, 28
- HP-UX
  - 网络协议 315

- I/O 防护 166
- IBM RS/6000
  - 网络协议 315
- interfaces 文件
  - 描述的 309
- interfaces 文件
  - Adaptive Server, 命名 315
  - Adaptive Server, 使用者 312
  - API 组成部分 315
  - delay\_interval 组成部分 315
  - device 组成部分 316
  - ether 占位符 315
  - host 组成部分 315
  - loghost 占位符 318
  - machine 组成部分 315
  - master 服务类型 315
  - network 组成部分 315
  - port 组件 316
  - protocol 组成部分 315
  - query 服务类型 315
  - retry\_attempt 组成部分 315
  - servername 组成部分 315
  - service\_type 组成部分 315
  - 查询端口备份配置 320
  - 创建, 适用于初学者 316
  - 多个网络 311, 318
  - 多个网络监听器 318
  - 客户端, 使用者 309
  - 客户端版本和服务器版本, 差别 312
  - 空格位于 314
  - 内容 311
  - 缺省位置 309
  - 条目中的唯一元素 317
  - 调试服务类型 315
  - 同构环境和 311
  - 位置 309
  - 异构环境和 311
  - 用 **dsedit** 创建主文件 317
  - 用文本编辑器创建主文件 317
  - 由客户端使用的 310
  - 制表符 314
  - 自动创建 309
  - interfaces 文件中的 API 组成部分
    - 描述的 315
  - interfaces 文件中的 delay\_interval 组成部分 315
  - interfaces 文件中的 device 组成部分 316
  - interfaces 文件中的 ether 占位符 315
  - interfaces 文件中的 host 组成部分 315
  - interfaces 文件中的 loghost 318
  - interfaces 文件中的 machine 组成部分 315
  - interfaces 文件中的 network 组成部分 315
  - interfaces 文件中的 port 组成部分 316
  - interfaces 文件中的 retry\_attempts 组成部分 315
  - interfaces 文件中的 servername 组成部分 315
  - interfaces 文件中的 service\_type 组成部分 315
  - interfaces 文件中的空格 314
  - interfaces 文件中的制表符 314
- iosstat 命令
  - Sun Solaris 267
- iso-Latin1 字符集 273
- isql 实用程序 29
- JINI 发现方法 210
- Job Scheduler 139–141
  - 安装 139
  - 重定向作业 140
  - 关闭 140
  - 配置 139
  - 运行 139
- JVM 157
- KEEPALIVE 选项, TCP/IP 265
- LDAP
  - 定义的 37
  - 多目录服务 44
  - 访问限制 38
  - 服务器, 使用 **dsedit** 实用程序添加和修改 43
  - 库, 环境变量 42
  - 目录定义 39
  - 目录架构 40
  - 启用 41
  - 添加服务器 43
  - 样本条目 40
  - 与 *interfaces file* 38
  - 在 *libtcl.cfg* 中指定 41
- ldapurl
  - 定义的 41

- ldapurl
  - 关键字 42
  - 示例 41
- libtcl\*.cfg 文件 41
  - 格式 41
  - 口令 45
  - 目的 41
  - 位置 41
- loc 文件 283
- locales 目录 283
- locales.dat 文件 284
- master
  - interfaces 文件 311, 317
  - 服务类型 315
- mount database 命令 164
- native 成员资格模式 168
- netstat 命令
  - Sun Solaris 267
- Open Client 19
  - 支持级别 20
- protocol
  - interfaces 文件中的组成部分 315
  - SPX 315
  - TCP/IP 315
- ps 命令
  - Sun Solaris 267
- pwdcrypt 命令
  - 口令加密 45
  - 位置 45
- qrmutil 实用程序 155, 172, 176
- query 服务类型 312, 315
- RepAgent 线程 17
- Replication Server 17
- roman8 字符集 273
- setperm\_all
  - 权限 261
- SMP Adaptive Server 和 Cluster Edition 11
- sp\_bindcache 存储过程 113, 114
- sp\_cacheconfig 存储过程 99
- sp\_dropdevice 存储过程 134
- sp\_helpdevice 存储过程 134
- sp\_poolconfig 存储过程 109
- sp\_serveroption 存储过程 30
- SPX 网络协议 315
- srt 文件 278
- stty 设置 261
- Sun Solaris
  - ioostat 命令 267
  - netstat 命令 267
  - ps 命令 267
  - time 命令 267
  - vmstat 命令 267
  - 超时期限 266
  - 网络协议 315
- sundiag 系统诊断工具 267
- Sybase 95
- Sybase 全球化支持 271, 282, 286
- sybcluster
  - Unified Agent Framework 240, 245
  - 标识 Unified Agent 242
  - 创建 XP Server 256
  - 断开与集群的连接 249
  - 更改实例属性 253
  - 关闭集群 250
  - 关闭实例 254
  - 检验实例 252
  - 鉴定用户 241
  - 连接到集群 241
  - 命令列表 238
  - 启动 238, 240, 244
  - 删除集群 250
  - 删除实例 254
  - 设置缺省实例 253
  - 升级服务器 257
  - 添加实例 252
  - 显示会话信息 247
  - 显示集群信息 246
  - 显示配置值 247
  - 显示实例信息 251
  - 限制 237
  - 在手动创建后启用 sycluster 254
  - 执行集群检查 245
- sybcluster 实用程序 237–257
- sybcluster 中的故障切换
  - 配置 75
  - 启用 21
  - 修改 CTLIB API 调用 21
- sybsecurity 数据库 294

sybsyntax 数据库 301  
 TCP/IP 265, 315  
     **KEEPALIVE** 选项 265  
**time** 命令  
     Sun Solaris 267  
 TLI 协议 315  
 UDP 发现方法 210  
 Unicode  
     字符转换 274  
 Unified Agent Framework 240  
 Unified Agent, 启用 209  
 UNIX  
     网络协议 315  
     硬件错误消息 266  
 UnixWare  
     网络协议 315  
 unmount database 命令 164  
 us\_english 语言 273  
 vcs 成员资格模式 168  
     限制 169  
     要求 168  
 Veritas Cluster Membership 插件 (VCMP) 166  
 Veritas SF for Sybase CE 165  
 Veritas Storage Foundation 165  
 Veritas 和 Cluster Edition 165–187  
     I/O 防护 166  
     Sybase 组件 167  
     Veritas Sybase 组件 167  
     Veritas 组件 167  
     安装 169, 170  
     成员资格管理 166  
     重新配置互连 172  
     创建集群 170  
     故障排除 178  
     管理 173  
     了解故障情况 177  
     配置 169  
     优点 165  
     支持的平台 168  
     转换集群 172  
     组件 167  
 Veritas 和 Cluster Edition 支持的平台 168  
**vmstat** 命令  
     Sun Solaris 267

xact\_connmigrate\_check 函数 163  
 xact\_owner\_instance 函数 163  
 阿拉伯语字符集 274

## A

安全套接字层 (SSL) 35  
 安全性。请参见审计  
 安装 Job Scheduler 139

## B

绑定  
     对象, 语法 114  
     对象到命名高速缓存 113  
     删除高速缓存 115  
 绑定的高速缓存 113  
     获得有关信息 114  
     删除绑定 115  
     显示有关信息 115  
 保留锁 148  
 本地  
     命名高速缓存 116  
     命名高速缓存, 格式 116  
 本地高速缓存  
     条目中的额外行 117  
 本地化 271  
     更改配置 286  
     通用, 信息 284  
 本地集群高速缓存 98  
 本地临时数据库  
     创建 126  
     类型 122  
     使用 ASE 插件管理 219–220  
     在 ASE 插件中添加用户创建的 221  
 本地配置  
     创建, 在存在全局配置的情况下 119  
     命名高速缓存 118  
 本地日期、时间和货币格式 284  
 本地数据库  
     私有设备支持 132

## 索引

- 本地系统临时数据库 121, 122
  - 创建 125
- 本地用户临时数据库 122

## C

### 参数

- 功能组 213
- 需要重新启动 213
- 操作描述符 85
- 操作系统
  - 资源 267
- 查询端口备份配置 320
- 朝鲜语字符集 276
- 成员资格模式 168, 175
  - native 168
  - vcs 168
  - 更改 176
  - 确定 175

### 池

- 更改本地异步预取百分比 112
- 更改清洗大小 111
- 缓冲, 移动内存 110
- 删除缓冲 112
- 池的本地异步预取百分比, 更改 112
- 池的清洗大小, 更改 111
- 创建

- interfaces* 文件 311, 317
- interfaces* 文件 (自动) 309
- 多个缓冲池 109
- 私有设备 (使用 **disk init**) 133
- 用 **dscp** 实用程序创建主 *interfaces* 文件 317
- 用 **dsedit** 创建主 *interfaces* 文件 317
- 主 *interfaces* 文件 (使用文本编辑器) 317

### 创建 Backup Server 256

### 存储过程

- sp\_bindcache** 113
- sp\_dropdevice** 134
- sp\_helpdevice** 134
- sp\_poolconfig** 109
- sp\_serveroption** 30

### 存档数据库 158

- 错误日志的路径 306
  - 配置 306

## D

- “待定值”列 214
- 登录重定向 19, 22
  - 连接属性 23
  - 目录服务 23
- 端口号和 *interfaces* 文件 318
- 对称多重处理 (SMP) 11
- 对象一致性 7
- 多个安装
  - 创建一个 *interfaces* 文件 311, 317
  - 影响 *interfaces* 文件 312
- 多个缓冲池, 配置和使用 109
- 多个网络
  - interfaces* 文件 318
  - interfaces* 文件和 311
  - 在网络发生故障时用作备份 320
- 多目录服务
  - LDAP 44

## E

- 二进制排序顺序 279

## F

### 分布式检查点

- Cluster Edition 与 SMP 152

### 分布式事务管理

- 另请参见 DTM

### 服务类型

- master 315
- query 312, 315
- 监听器 312
- 调试 315

### 服务器

- 命名要求 315
- 服务器的命名要求 315
- 服务器发现, 更改设置 209

服务器配置文件 155  
 服务器组, 使用 ASE 插件删除 216  
 复制  
   集群支持 17  
 负载管理 88–95  
   负载配置文件 91  
   负载阈值 90  
   故障排除 95  
   样本负载配置文件 92  
   指标 89  
 负载管理器 7, 61–95  
   内存要求 77  
 负载配置文件 91  
   创建 93  
   负载分配阈值 94  
   建立 93  
   示例 92  
   修改 95  
   与逻辑集群关联 94  
   在 ASE 插件中 223  
 负载阈值 90  
   停滞值 91  
 负载指标 89  
   CPU 利用率 89  
   I/O 负载 89  
   加权 90  
   引擎差值 89  
   用户连接数 89  
   用户提供的指标 89, 90  
   运行队列长度 89

## G

高速缓存  
   绑定, 获取有关信息 114  
   本地, 额外行 117  
   本地命名, 格式 116  
   将对象绑定到命名 113  
   具有本地配置的命名 118  
   具有全局配置的命名, 已删除 117  
   删除绑定 115

高速缓存配置, 集群 97  
 格式  
   本地命名高速缓存 116  
 跟踪标志 95  
 更改  
   排序顺序 286  
   语言 286  
   字符集 272, 286  
 共享安装模式 155  
 共享数据库设备, 在 ASE 插件中创建 218  
 古斯拉夫语字符集 275  
 故障  
   由集群处理 4  
 故障切换资源 75  
   添加 76  
 故障切换组 75  
 管理 Veritas 和 Cluster Edition 173  
   启动和停止实例 173  
   添加和删除实例 174  
 归类序列。请参见排序顺序 278  
 国际系统  
   Sybase 支持 271  
   支持 271

## H

互连网络 12–14  
 环境变量  
   DSQUERY 311, 319  
 缓冲池  
   创建 109  
   更改清洗大小 111  
   更改预取百分比 112  
   删除 112  
   移动内存 110  
   转移内存 110  
 缓冲区高速缓存一致性 7  
 恢复 143  
   Cluster Edition 与 SMP 143

## J

- 基本实例 62
- 集群
  - DBMS 层 7
  - 成员资格服务 7
  - 定义 3
  - 互连网络 12-14
  - 空间和阈值 7
  - 日志和恢复 7
  - 数据库设备 8
  - 锁管理器 7
  - 信息的存储 8
- 集群高速缓存
  - 本地 98
  - 定义 97
  - 配置 97
  - 全局 97
- 集群环境中的 Java 157
- 集群件 6
  - 组件 6
- 集群进程间通信 (CIPC) 7, 152
- 集群事件服务 7
- 集群锁管理程序 (CLM) 148
- 集群协调器 7
- 集群中的数据库设备 8
- 计划
  - 停机时间 82, 86
- 加变音的字母 279
- 监控
  - 操作系统资源 267
- 监控表
  - 查询 78
- 监听器服务 312
- 脚本
  - C shell 267
  - 维护 267
  - 维护示例 267
- 角色
  - 重新设置配置选项 213
  - 配置实例 213
  - 系统安全员 213
  - 系统管理员 213

## 节点

- 定义 3
- 具有有效配置的已删除条目 118

## K

- 开放式逻辑集群 70
- 客户端
  - Adaptive Server 通信 309
  - DSQUERY 和 319
  - 缺省字符集 273
  - 文件服务器 319
  - 应用程序和 *locales.dat* 文件 285
  - 之间的转换, 与服务器 278
- 客户端 *interfaces* 文件
  - 客户端与服务器版本之间的差别 312
  - 同构 313
  - 异构的 313
- 客户端 / 服务器交互 22
- 客户端应用程序 19-31
- 客户端与 Adaptive Server 之间的通信 309
- 口令加密
  - pwdcrypt** 命令 45
  - 用于 *libtcl\*.cfg* 45
- 扩展的高可用性故障切换 19, 28
  - Open Client 支持级别 28
  - 目录服务 28
  - 应用程序更改 21
  - 与 HA 故障切换的区别 28

## L

- 拉丁语字符集 275
- 联机语法帮助 301
- 连接迁移 24
  - CS\_PROP\_MIGRATABLE 属性 20
  - 迁移 24
  - 迁移上下文 25
  - 迁移与故障切换 24
  - 条件 25
- 连接重定向 19

- 链路
    - 节点之间 12–14
    - 在实例之间监控 12
  - 临时数据库 121–132
    - 本地 121
    - 创建 125
    - 创建绑定 129
    - 创建组 127
    - 从 model 数据库继承 121
    - 定义 121
    - 管理绑定 129
    - 会话绑定 128
    - 将应用程序绑定到 126, 127
    - 将用户绑定到 126, 127
    - 全局 121
    - 删除 130
    - 使用 ASE 插件管理 219–223
    - 使用准则 131
    - 特性 124
    - 在 ASE 插件中向组中添加 221–223
  - 路径, 错误日志 306
  - 路由规则 68
    - 使用 SSL 时 68
    - 为登录 68
    - 用于别名 68
    - 用于应用程序 68
  - 逻辑集群 227–235
    - 创建 63, 80
    - 定义 61
    - 将负载配置文件关联 226
    - 路由规则 68
    - 启动 67
    - 删除、 80
    - 删除路由 81
    - 删除资源 80
    - 属性 69
    - 添加路由到 66, 81
    - 添加实例到 66
    - 添加资源 80
    - 显示有关信息 78
    - 在 ASE 插件中负载配置文件 232
    - 在 ASE 插件中故障切换实例 231
    - 在 ASE 插件中逻辑集群的负载状态 233–235
    - 在 ASE 插件中逻辑集群的路由 233
    - 在 ASE 插件中删除 229
    - 在 ASE 插件中属性 229–233
    - 在 ASE 插件中添加 229, 231
    - 在 ASE 插件中添加故障切换实例 232
  - 逻辑集群特性
    - fail\_to\_any 73, 75
    - system\_view 71
    - 登录分配 69, 73
    - 负载配置文件 69, 73
    - 故障切换 69, 72, 75
    - 开放式 69, 70
    - 启动 69, 72
    - 任何实例都提供故障切换支持 69
    - 系统视图 69
    - 向下路由模式 69, 70
  - 逻辑集群中的 SPID 77
  - 逻辑集群状态 82
    - 操作描述符 85
    - 定义 83
    - 更改 83
    - 命令影响 87
    - 全局 12, 82
    - 实例 12, 82
    - 异步命令 85
  - 逻辑集群资源
    - 故障切换 62
    - 实例 62
- ## M
- 命令
    - alter database** 137
    - create database** 137
    - disk init** 133
    - disk refit** 137
    - disk reinit** 133
  - 命名高速缓存
    - 本地 116
    - 将对象绑定到 113
    - 具有全局配置 118

- 具有全局配置的已删除 117
- 配置文件 115
- 命名数据高速缓存 98
  - 创建 98, 99
  - 分配空间 105
  - 更改高速缓存类型 107
  - 减小大小 106
  - 将对象绑定到 113
  - 配置替换策略 108
  - 删除 107
  - 显示有关信息 98
- 目录
  - charsets 279
  - charsets** 284
  - 本地化 283
- 目录架构, LDAP 40

## N

- 内存
  - 负载管理器的要求 77
  - 在缓冲池之间移动 110
- 内核 7
  - 集群组件 7

## P

- 排除 Veritas 和 Cluster Edition 的故障 178
  - Veritas 未启动 186
  - 集群未启动 178
  - 资源故障 183, 185
- 排除负载管理器的故障 95
- 排序顺序 278
  - Adaptive Server 的缺省值 273
  - 定义文件 278
  - 二进制 279
  - 更改 273, 286
  - 数据库和 279
  - 字典 279
  - 字符集和 279
  - 字母大小写 279

- 排序顺序中的字母大小写 279
- 配置
  - Backup Server 283, 286
  - Cluster Edition 213
  - 多个缓冲池 109
  - 立即和重新启动 213
  - 实例, 所需的角色 213
  - 系统管理员 213
  - 有效, 具有已删除的条目 118
  - 字符集 287
- 配置 Job Scheduler 139
- 配置参数, 使用 ASE 插件设置 212
- 配置方案 14
- 配置命名高速缓存 115
- 配置文件
  - 修改 115

## Q

- 启用 sybcluster 254
- 迁移上下文
  - 元素 25
- 权限
  - 恢复 261
- 全局集群高速缓存 97
- 全局临时数据库 121, 124
  - 创建 126
  - 在 ASE 插件中添加用户创建的 221
- 全局配置
  - 已删除命名高速缓存 117
  - 在存在的情况下创建本地配置 119
- 全球化支持, Sybase 271, 282, 286
- 缺省值
  - Adaptive Server 的语言 273
  - Adaptive Server 的字符集 273
  - 排序顺序 273
  - 语言, 改变 273
  - 字符集, 更改 273

## S

- 删除缓冲池 112
- 上下文迁移 26
- 设置配置参数 213
- 审计
  - 进程 294
  - 全局选项 294
  - 设备 294
  - 使用 脚本安装 294
  - 数据库 294
  - 用于跟踪的表 294
- 审计系统 293
- 审计追踪
  - 概述 293
  - 系统审计表 294
- 实例
  - 定义 3
  - 在 ASE 插件中从集群删除 217
  - 在 ASE 插件中关闭 218
  - 在 ASE 插件中启动 218
  - 在 ASE 插件中添加到集群 217
- 实例, 监控链路 12
- 使用 ASE 插件断开与集群的连接 209
- 使用 **disk reinit** 重新初始化私有设备 133
- 使用 **sp\_dropdevice** 删除私有设备 134
- 示例
  - sp\_poolconfig** 109
- 数据翻译 271
- 数据高速缓存
  - 分配空间 105
  - 更改高速缓存类型 107
  - 减小大小 106
  - 配置替换策略 108
  - 删除 107
  - 添加内存到 105
- 数据库 279
- 数据库设备
  - sybsyntax 302
- 斯基的纳维亚字典排序顺序 280
- 私有设备 132
- 锁 148
  - 保留锁 148
  - 集群锁管理程序 148

## T

- 泰语字符集 276
- 特定于平台的区域设置名 284
- 添加
  - 在 ASE 插件中负载配置文件 224
  - 在 ASE 插件中故障切换实例 232
  - 在 ASE 插件中逻辑集群 229, 231
  - 在 ASE 插件中向集群添加实例 217
  - 在 ASE 插件中向组中添加临时数据库 221–223
  - 在 ASE 插件中用户创建的本地临时数据库 221
  - 在 ASE 插件中用户创建的全局临时数据库 221
- 添加 LDAP 服务器 43
- 添加空间, 向存档数据库 158
- 条目
  - 本地高速缓存中的额外行 117
  - 删除具有有效配置的 118
- 调试服务类型 315
- 停机时间
  - 计划 82, 86
- 停滞值 91
- 同构环境
  - interfaces* 文件和 313
  - 描述的 313
- 土耳其语字符集 276

## W

- 网络
  - DSQUERY 和 319
  - interfaces* 文件 309
  - 备用连接和 321
  - 多重的, 多值 311
  - 故障 321
- 网络协议
  - Digital UNIX 315
  - HP-UX 315
  - IBM RS/6000 315
  - Sun Solaris 315
  - UnixWare 315
- 文件
  - common.loc* 284
  - locales.dat* 284
  - 本地化 272

## 索引

- 服务器 319
- 排序顺序定义 (.srt) 文件 278
- 配置, 修改 115
- 显示当前描述符 261
- 已本地化的错误消息 (.loc) 283
- 文件描述符的限制 262

## X

- 西班牙语字典排序顺序 280
- 希伯来语字符集 275
- 希腊语字符集 275
- 系统安全员
  - 配置实例 213
- 系统管理员
  - 配置集群 213
- 系统逻辑集群
  - open 属性 62
  - 定义 62
- 系统审计表 294
- 系统消息, 被翻译的 272
- 显示
  - 当前文件描述符 261
  - 私有设备信息 (使用 **sp\_helpdevice**) 134
  - 有关逻辑集群的信息 78
- 限制 120
- 向下路由模式 70
  - disconnect** 命令 71, 228, 230
  - open** 命令 70, 228, 230
  - system** 命令 70, 228, 230
  - 值 70
- 消息
  - 选择语言 282
  - 硬件错误 266

## Y

- 已翻译的消息
  - 错误 (.loc 文件) 283
  - 系统 272
- 异步命令 85
  - wait** 选项 85

- 异构环境 273, 277
  - interfaces* 文件和 313
  - 描述的 313
- 硬件错误消息 266
  - UNIX 266
- 用户定义的消息 307
- 用户连接数 262
- 用户数据报协议 (UDP) 210
- 用于本地日期、时间和货币的格式 284
- 语法
  - 将对象绑定到 114
- 语言
  - 翻译支持 271
  - 更改 286
  - 特定语言的错误报告 283
  - 选择消息 282
- 语言模块 272, 282
  - 安装新的 282
  - 本地化文件 272
  - 内存要求 286
- 远程过程调用 (RPC) 29

## Z

- 在 ASE 插件中负载管理器 223–235
- 在 ASE 插件中路由属性 236
- 在 ASE 插件中系统临时数据库 220
- 在 ASE 插件中阈值 227
- 在 ASE 插件中指标权值 227
- 在 Veritas 上安装 Cluster Edition 169
- 在 Veritas 上配置 Cluster Edition 169
- 在缓冲池之间移动内存 110
- 在集群中配置高速缓存 97
- 针对 Veritas 创建集群 170
- 针对 Veritas 重新配置互连 172
- 针对 Veritas 转换集群 172
- 中文字符集 274
- 仲裁设备
  - 说明 8
- 主机名
  - 确定 315
- 专用安装模式 155
- 转换, Unicode 字符 274
- 资源保留 71

- 字典排序顺序 279
  - 斯堪的纳维亚语 280
  - 西班牙语 280
- 字符集 278
  - 代码转换和 277
  - 更改 272, 273, 286
  - 客户端选择 273
  - 排序顺序和 279
  - 配置 287
  - 缺省值 272
  - 数据库和 279
  - 在异构环境中 277
  - 之间转换 277
- 字符集之间的转换 277

