



高可用性用户指南



高可用性用户指南

声明

在使用本资料及其支持的产品之前，请阅读第 179 页的『声明』中的信息。

本版本适用于 IBM solidDB（产品编号 5724-V17）和 IBM solidDB Universal Cache（产品编号 5724-W91）V6R3 及所有后续发行版和修订版，直到在新版本中另有声明为止。

© Solid Information Technology Ltd. 1993, 2008

目录

图	v
表	vii
关于本手册	ix
图解约定	ix
印刷约定	x
语法表示法约定	xi
1 IBM solidDB HotStandby 简介	1
HotStandby 的工作方式	2
HotStandby API (HSB 管理命令)	2
基本 HotStandby 服务器方案	3
脉动信号	3
事务日志和 HotStandby	4
服务器 HotStandby 状态	5
将 HotStandby 与高级复制技术结合起来使用	5
故障转移	6
服务器同步复制	7
HotStandby 中的复制方式	9
对服务器状态的描述	11
HotStandby 如何影响性能	14
自适应耐久性	15
高可用性控制器 (HAC)	17
常见的故障	19
控制数据库服务器进程	19
外部引用实体 (ERE)	20
在 HAC 中联网	21
高可用性管理器	22
HAC 日志记录	23
2 HotStandby 入门	25
开始之前	25
HotStandby 演示程序	25
HotStandby 快速启动过程	25
启动和停止 HA 控制器	27
使用了 HA 控制器的 HSB 的快速启动过程	28
启动顺序总结	29
3 管理和配置 HotStandby	33
您应当知道的信息	33
HotStandby 和 solidDB 配置文件	33
HotStandby 和访问权	34
solidDB 工具和 HotStandby	34
数据库迁移 (仅适用于基于磁盘的服务器)	34
互操作性	35
HotStandby 存在的局限性和与 HotStandby 有关的警告	35
内存表	35
网络分区和双重主服务器	35
用尽用于事务日志的空间	36

管理任务概述	36
执行 HotStandby 恢复和维护	37
特殊配置: 更低的成本与更高的安全性	38
降低成本: N + 1 个备用服务器和 N + M 个备用服务器方案	38
提高可靠性: 2N + 1 备用服务器和 2N + M 备用服务器方案	39
solidDB HSB 如何支持 N+1 (N+M) 和 2N+1 (2N+M) 方法	39
将 HAC 与备用服务器配合使用	40
为 HotStandby 配置 solidDB	40
定义辅助节点和主节点配置 ([Com] 节)	40
定义应用程序与服务器之间的超时 ([Com] 节)	41
事务耐久性	42
配置特定于 HotStandby 的参数	43
定义主服务器和辅助服务器 HotStandby 配置	43
设置 HotStandby 服务器等待时间以帮助检测已断开的连接或者不可用的连接	43
定义 HotStandby 数据库复制操作的名称和位置	45
定义主服务器在辅助服务器发生故障期间的行为	45
性能调整	46
使用安全级别和耐久性级别来调整复制性能	46
调整 netcopy 性能 ([General] 节)	46
配置 HA 控制器和 HA 管理器	47
[HAController] 节	47
[LocalDB] 节	48
[RemoteDB] 节	49
[ERE] 节	49
配置文件示例	49
solid.ini 配置文件	49
solidhac.ini 配置文件	50
HAManager.ini 配置文件	52
4 对应用程序使用 HotStandby	53
用于连接至 HotStandby 服务器的两种方法	53
选择连接类型	53
使用透明连接	54
TC 中的故障透明性	54
透明连接中的负载均衡	54
透明连接信息的语法	57
TC Info 属性组合	58
处理 TC Info 矛盾	59
在 JDBC 中建立透明连接	59
连接切换的编程	61
使用基本连接	65
从应用程序重新连接至主服务器	65
重新连接至辅助服务器	69
高级复制需求	69
5 使用 HotStandby API 命令	73
切换服务器状态	73

执行切换	73
验证切换	75
执行故障转移	75
运行新的处于 PRIMARY ALONE 状态的主服务器	76
使辅助服务器恢复联机状态	77
停止 HotStandby 操作	77
使主服务器与辅助服务器同步	78
同步复制	78
完全复制	79
验证复制	82
使用看守程序使服务器同步	82
通过网络将主数据库复制到辅助服务器	82
为辅助服务器创建新的数据库	83
替换辅助服务器上的现有数据库	84
验证 netcopy 状态	85
将数据库文件从主服务器复制到指定目录	85
连接 HotStandby 服务器	86
检查 HotStandby 状态	87
显示切换状态信息	87
显示连接状态信息	88
显示通信信息	88
显示角色的开始时间	89
验证 HotStandby 服务器状态	89
服务器状态组合情况	90
选择使哪个服务器成为主服务器	91
将 HotStandby 服务器更改为非 HotStandby 服务器	92

6 高可用性控制器在故障情况下的行为 . . . 93

主数据库失败	93
辅助数据库失败	93
主节点失败	94
辅助节点失败	95
HotStandby 链路失败	95

7 使用 HotStandby 对服务器进行升级 . . . 97

冷迁移和热迁移	97
在与 HSB 兼容的版本之间进行迁移	97
在与 HSB 不兼容的版本之间进行迁移	97
在与 HSB 不兼容的版本之间进行迁移之前的准备步骤	98
升级之后	100

附录 A. HotStandby 配置参数 103

确保主服务器和辅助服务器的参数值协调	103
确定主服务器的设置是否优先于辅助服务器的设置	104
查询 HotStandby 配置参数	104

修改 HotStandby 配置参数	105
访问方式	105
[Cluster] 节	106
[HotStandby] 节	106
高可用性控制器配置参数	109
高可用性管理器配置参数	112

附录 B. 错误代码 115

HotStandby 错误和状态码	115
高可用性控制器错误和状态码	122
solidDB 数据库错误	124
solidDB 表错误	126
solidDB 通信错误	126

附录 C. HotStandby 管理命令总结 . . . 127

HotStandby 命令	127
高可用性控制器命令	134

附录 D. 服务器状态过渡 135

HotStandby 状态过渡图	135
------------------	-----

附录 E. HotStandby 系统事件 141

附录 F. 看守程序样本 143

使用看守程序的 HotStandby 配置	143
看守程序的工作方式	144
系统设计问题	145
看守程序配置	146
使用样本看守程序	147
故障情况和看守程序操作	147
主服务器已关闭	148
辅助服务器已关闭	150
看守程序已关闭	153
主服务器和辅助服务器之间的通信链路已断开	156
看守程序和主服务器之间的通信链路已断开	158
看守程序和辅助服务器之间的通信链路已断开	161
看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开	163
看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开	166
solid.ini 配置文件的 [Watchdog] 节	169

索引 173

声明 179



1. HotStandby 体系结构	2	16. 使用了看守程序的异构 HotStandby 配置	146
2. HotStandby 服务器方案	3	17. 主服务器已关闭的情况和补救措施	149
3. 由 HotStandby 以及主服务器和副本服务器组成的方案	6	18. 辅助服务器已关闭的情况和补救措施	152
4. HotStandby 切换到新的主服务器（也就是旧的辅助服务器）	7	19. 看守程序已关闭的情况和补救措施	155
5. 服务器故障转移和同步复制示例	8	20. 主服务器与辅助服务器之间的链路已断开的情况和补救措施	157
6. 同步 HotStandby 配置	10	21. 看守程序与主服务器之间的链路已断开的情况和补救措施	160
7. 高可用性控制器体系结构	18	22. 看守程序与辅助服务器之间的链路已断开的情况和补救措施	162
8. “外部引用实体”组件	21	23. 看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况和补救措施	164
9. 高可用性管理器	22	24. 看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况和补救措施	167
10. 启动顺序总结	30		
11. 主数据库故障转移	70		
12. 副本故障转移	72		
13. 状态切换	74		
14. 手动完全复制过程	81		
15. HotStandby 服务器状态过渡	136		

表

1. HotStandby 图解约定	ix	18. 集群参数	106
2. 印刷约定	x	19. HotStandby 参数	106
3. 语法表示法约定	xi	20. HAC 配置参数: [HAController] 节	109
4. 对服务器状态的描述	12	21. HAC 配置参数: [LocalDB] 节	111
5. 安装顺序步骤	31	22. HAC 配置参数: [RemoteDB] 节	112
6. 管理任务	37	23. HAC 配置参数: [ERE] 节	112
7. 选择连接类型	53	24. 高可用性管理器配置参数	112
8. TC Info 缩写	57	25. HotStandby 错误和状态码	115
9. 可能存在的 TC Info 属性组合	58	26. 高可用性控制器错误和状态码	122
10. 连接请求错误	60	27. solidDB 数据库错误	124
11. 警告	61	28. solidDB 表错误	126
12. 连接切换请求	62	29. solidDB 通信错误	126
13. 通信链路故障	62	30. 高可用性控制器命令	134
14. 会话状态保留	63	31. 服务器状态过渡表	137
15. HOTSTANDBY_CONNECTSTATUS 状态值	68	32. HotStandby 事件	141
16. 连接状态返回值	88	33. 看守程序参数	169
17. 服务器状态	90		

关于本手册

IBM® solidDB® 高可用性 (HotStandby) 组件将提高数据库系统的可靠性, 同时可以缩短停机时间。HotStandby 使用一种“热备用”方法。在这种方法中, 第二个数据库服务器与主服务器并行运行并且保留数据的最新副本。如果主数据库服务器失败, 那么高可用性控制器 (HAC) 将以对于应用程序透明的方式切换到辅助服务器, 并且不会丢失已落实的事务, 对于性能的影响也最小。切换速度非常快 - 可以在几百毫秒之内完成切换, 这取决于您所在的硬件和软件环境的特征。

本指南只包含特定于 HotStandby 组件的信息。有关 solidDB 数据库的一般管理和维护信息, 请参阅《IBM solidDB 管理员指南》。

本指南假定读者已经具备 DBMS 的一般知识并且熟悉 SQL 和 solidDB。

图解约定

本文档中包含多个服务器图, 用于描述 HotStandby 环境中的不同情况。

下表提供了服务器图的图解要点:

表 1. HotStandby 图解约定

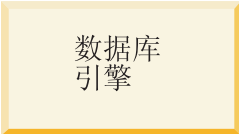


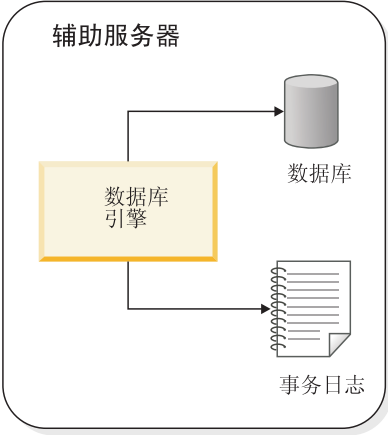
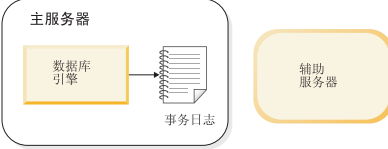
符号	描述
 数据库引擎	矩形表示正在执行的程序, 即, 数据库服务器 (引擎) 本身。
 数据库	柱面表示通常存储在磁盘上的数据。也有可能某些数据或所有数据存储在内存中。
 事务日志	此符号表示事务日志 (事务日志), 用于数据库恢复和 HotStandby 中。

表 1. HotStandby 图解约定 (续)

符号	描述
 <p>辅助服务器</p> <p>数据库引擎</p> <p>数据库</p> <p>事务日志</p>	<p>圆角矩形表示一个具有数据和事务日志的完整服务器。如果圆角矩形中存在辅助服务器或主服务器这两个词语，那么此服务器是 HotStandby 服务器。</p>
 <p>主服务器</p> <p>数据库引擎</p> <p>事务日志</p> <p>辅助服务器</p>	<p>在某些情况下，为了简单起见，将省略用于表示数据库中的数据的数据的柱面。在某些情况下，将更进一步简化此符号以只显示圆角矩形。两个图标都是服务器的简化表示法。</p>

印刷约定

solidDB 文档使用下列印刷约定:

表 2. 印刷约定

格式	适用于
数据库表	此字体用于所有普通文本。
NOT NULL	采用此字体的大写字母指示 SQL 关键字和宏名称。
solid.ini	这些字体指示文件名和路径表达式。
SET SYNC MASTER YES; COMMIT WORK;	此字体用于程序代码和程序输出。示例 SQL 语句也使用此字体。
run.sh	此字体用于样本命令行。
TRIG_COUNT()	此字体用于函数名。
java.sql.Connection	此字体用于接口名称。
LockHashSize	此字体用于参数名、函数自变量和 Windows® 注册表条目。
<i>argument</i>	此类强调词指示用户或应用程序必须提供的信息。
管理指南	这种样式用于引用其他文档或者同一文档中的章节。新术语和强调的问题也按此样式书写。

表 2. 印刷约定 (续)

格式	适用于
文件路径表示	文件路径按 UNIX® 格式提供。斜杠 (/) 字符表示安装根目录。
操作系统	如果文档包含有关操作系统之间的差别的内容，那么首先提到的是 UNIX 格式。Microsoft® Windows 格式位于 UNIX 格式之后并括在括号中。其他操作系统将单独列出。对于不同的操作系统还可能有不同的章节进行描述。

语法表示法约定

solidDB 文档使用下列语法表示法约定：

表 3. 语法表示法约定

格式	适用于
INSERT INTO <i>table_name</i>	语法描述采用此字体。可替换部分采用此字体。
solid.ini	此字体指示文件名和路径表达式。
[]	方括号指示可选项；如果是粗体文本，那么必须将方括号包含在语法中。
	竖线，用于将语法行中的两个互斥选项分隔开。
{ }	大括号用于对语法行中的一组互斥选项进行定界；如果是粗体文本，那么必须将大括号包括在语法中。
...	省略号指示可以多次重复使用自变量。
· · ·	由三个点组成的一列表示这是先前代码行的延续。

1 IBM solidDB HotStandby 简介

本章描述了 IBM® solidDB HotStandby 组件。HotStandby 使辅助服务器（它是一个热备用服务器）与主服务器并行运行并且保留主服务器中数据的最新副本。

服务器状态由一个称为看守程序的实体控制。solidDB HotStandby 组件包含一个称为 solidDB 高可用性控制器（HAC）的看守程序实现。

在内部，HAC 使用 HotStandby 命令来控制服务器状态。诸如此类的解决方案能够实现具有更高可靠性的系统。即使数据库服务器失败，也不会使您的站点完全停止。在 solidDB 支持的任何引擎配置中（例如，solidDB 主控引擎配置或副本引擎配置），HotStandby 允许辅助数据库在仅仅几百毫秒之内就可以替换失败的数据库。

注：

“热备用”这个术语（hot standby）是指一项常规技术，即，当第一台服务器发生故障时，另一台服务器可以立即接替工作。而“HotStandby”（它是一个单词，首字母大写）是指 solidDB 用于实现此常规技术的特定组件。通常将 HotStandby 缩写为 HSB。

同样，看守程序也是指一项技术，用于监控两个数据库的状态，必要时可以切换状态。HAC 就是 solidDB 看守程序的实现。其中一个附录中提供的 Watchdog 样本是一个编程示例，它说明了如何使用 HSB API 来编写您自己的看守程序。

请参阅下面提供的 HotStandby 体系结构的概念性图示：

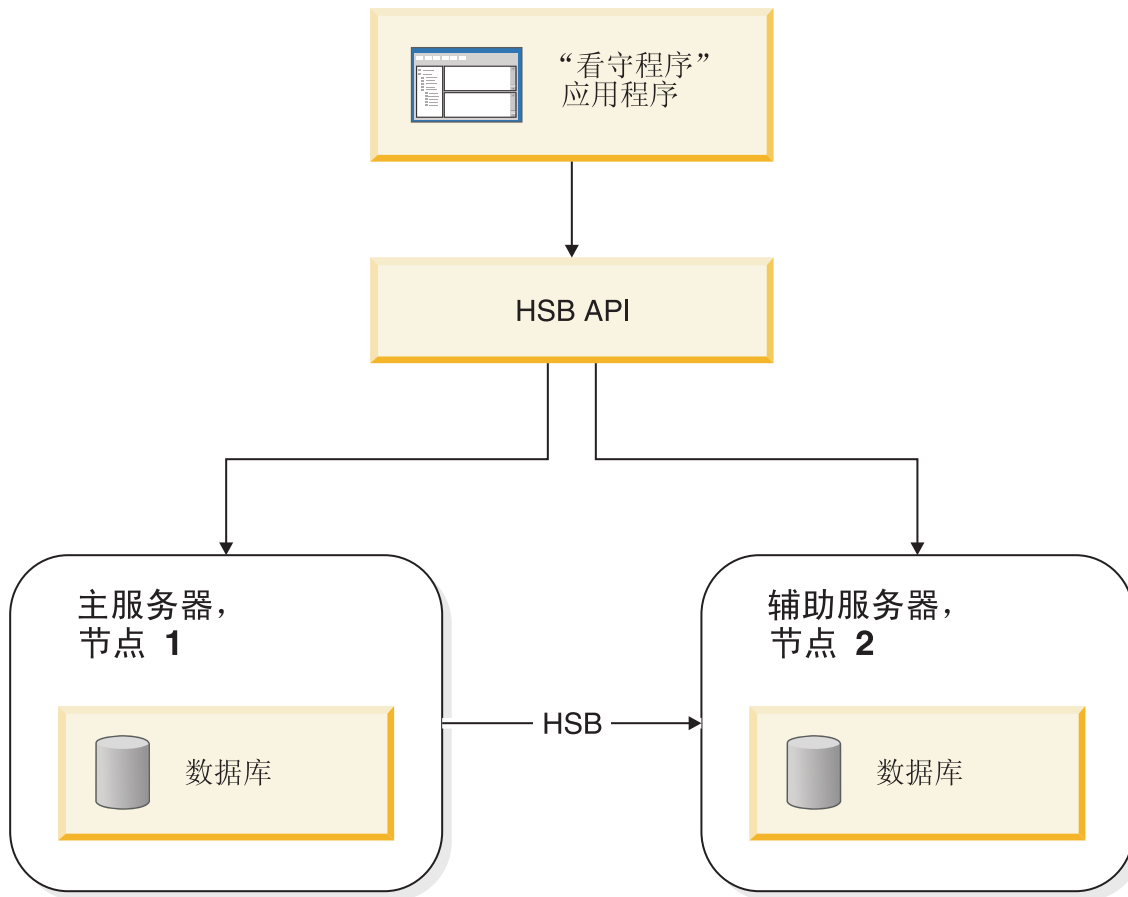


图 1. HotStandby 体系结构

从上图可以看出，HotStandby 提供了下列各项：

- 一个称为 solidDB 高可用性控制器（HAC）的看守程序。
- HotStandby API（HSB 管理命令）
- solidDB 服务器

HotStandby 的工作方式

HotStandby（HSB）在主服务器和辅助服务器这两个节点之间执行同步事务复制。

主服务器节点（主服务器）包含活动数据库。辅助服务器节点（辅助服务器）包含此活动数据库的准确而且是最新的副本。如果主服务器失败，它还可以接替主服务器。

辅助服务器将接收来自主服务器的更新。当主服务器失败时，辅助服务器可以接替它成为新的主服务器。提供辅助服务器的另一个好处在于，辅助服务器还可以对来自客户机的只读请求（例如，SELECT 语句）作出响应。这样就可以将一些工作负载分散在两台服务器上，而不是由一台服务器来承担。

HotStandby API（HSB 管理命令）

HotStandby 服务器的高可用性行为受基于一部分 solidDB 管理命令的 API 控制。

主命令将这一部分命令标识为 hotstandby（可以将 hotstandby 缩写为 hsb）。可以使用任何支持 SQL 的工具（例如，solsql）或者诸如 ODBC 或 JDBC 等程序化接口来发出这些命令。HotStandby 管理命令的语法如下：

```
admin command 'hotstandby hsb-command options';
```

或者

```
admin command 'hsb hsb-command options';
```

作为结果，它将说明您可以如何使用 hsb 管理命令来控制 solidDB HSB 服务器的高可用性状态或者检索状态信息。注意，可以手动发出或者通过程序发出这些命令。solidDB HA 管理工具“HA 控制器”和“看守程序”通过程序发出命令。

基本 HotStandby 服务器方案

在基本 HotStandby 服务器方案中，有两个数据库服务器 - 主服务器和辅助服务器，每个服务器都具有自己用于存储数据库的磁盘驱动器，并且都有自己的事务日志（Txn 日志）。

图 2 说明了基本 HotStandby 服务器方案。主服务器写入它的事务日志并转发至辅助服务器，以便辅助服务器可以对它的数据库副本进行相同更改。辅助服务器的事务日志不会主动涉及到 HSB，但是会维护事务日志，以便辅助服务器可以恢复已落实但是尚未写入主数据表的数据。（有关记录和恢复的更详细说明，请参阅《solidDB 管理员指南》。）

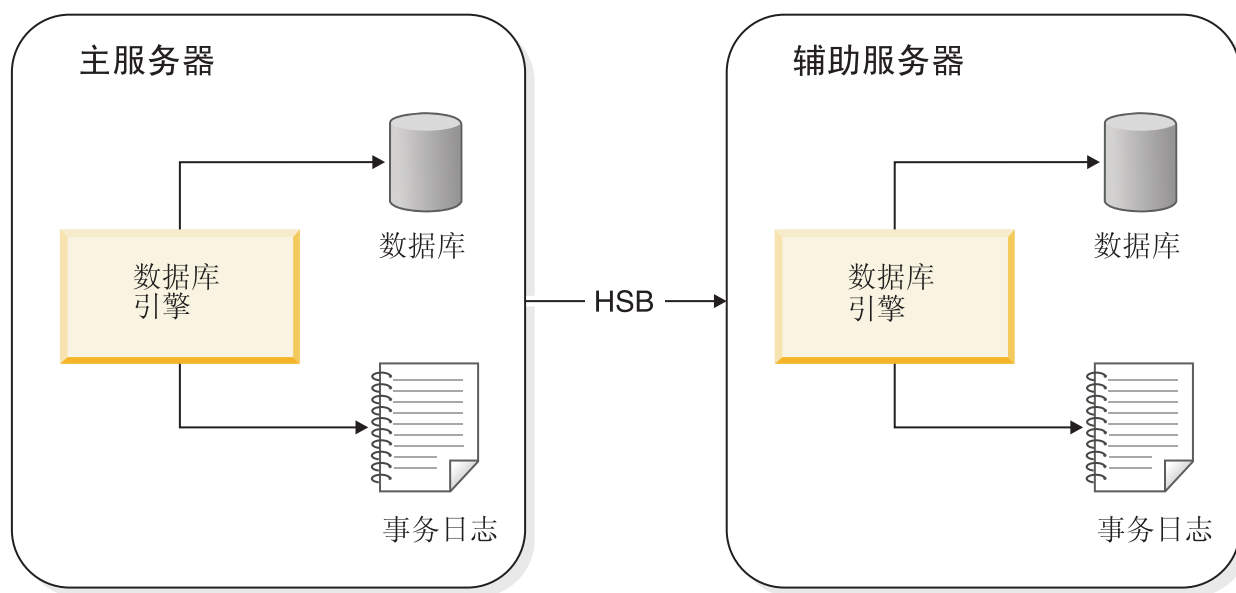


图 2. HotStandby 服务器方案

脉动信号

solidDB HSB 在内部使用一项称为脉动信号的技术来监视服务器之间的连接。

活动服务器与备用服务器之间将发送一系列保持活动消息。这两个服务器都会不断地向对方发送这些说明了位置的单向消息。将按固定的时间间隔发送这些消息。反过来，期望来自另一个服务器的消息在预定义的时间段内到达。在 solidDB 中，脉动信号技术称为 *ping*。

要点: 在 solidDB 中，脉动信号技术称为 *ping*，尽管没有发送 ping 请求。请不要将此技术与 TCP/IP 网络中使用的 Ping 协议混淆，两者是不同的概念。

事务日志和 HotStandby

HotStandby 使用主服务器的事务日志，此事务日志中包含此服务器上已落实事务的副本。在非 HotStandby 服务器中，当服务器异常关闭时，此事务日志可用来恢复数据。

在 HotStandby 主服务器中，也会将日志数据发送至辅助服务器以便让它知道要更新的数据。辅助数据库将运行连续前滚过程，此进程将接收日志数据并且使辅助服务器的数据副本与主服务器的数据副本保持同步。

如果主服务器失败，那么看守程序将让辅助服务器成为主服务器。一旦新的主服务器开始运行，客户机就可以与它连接并继续工作。客户机将看到在主服务器关闭之前已落实的所有数据。（客户机需要重新启动在原来的主服务器关闭时已启动但是未完成的所有事务。）

提供了一种称为透明连接（TC）的特殊类型的客户机连接，以便在进行故障转移和切换服务器时使客户机在 HSB 环境中运行。有关更多信息，请参阅第 53 页的 4 章，『对应用程序使用 HotStandby』。

如果辅助服务器失败，那么主服务器可以继续运行。它会继续将数据写入事务日志并且保留此事务日志，直到主服务器与辅助服务器互相重新建立连接，并且主服务器已将此日志发送至辅助服务器为止。¹

一旦失败的数据库服务器再次变得可用时，可以将它配置为新的辅助数据库服务器（当然，先前未失败的服务器目前已成为主服务器）。

如果主服务器失败，那么服务器之间的职责将互换，即，原来的辅助服务器将接替成为主服务器，而原来的主服务器在被修复之后将返回系统中成为新的辅助服务器。每当发生故障时，服务器之间的职责就会互换。正是因为任一服务器都可以作为主服务器这一事实，使得系统在经历多次故障之后仍能幸存，并且实际上将无限期地继续运行。

1.

主服务器保留日志的准确时间长度取决于 solid.ini 配置参数 **CheckpointDeleteLog** 和 **BackupDeleteLog** 的设置。

1. 如果 **CheckpointDeleteLog=Y**，那么主服务器将从辅助服务器关闭开始或者从最新检查点开始（以这两者中较早者为准）保留所有事务日志。（有关检查点的详细说明，请参阅《solidDB 管理员指南》。）
2. 如果 **CheckpointDeleteLog=N** 并且 **BackupDeleteLog=Y**，那么主服务器将从辅助服务器关闭开始或者从最新备份开始（以这两者中较早者为准）保留所有事务日志。
3. 如果 **CheckpointDeleteLog=N** 并且 **BackupDeleteLog=N**，那么服务器将无限期地保留日志。

注意:

如果主服务器在很长一段时间内都无法与辅助服务器联系, 那么事务日志可能会填满所有可用的磁盘空间。通过适当的配置参数设置可以避免发生这种情况。请参阅第 36 页的『用尽用于事务日志的空间』。

您甚至可以使用 HSB 来缩短对硬件和软件进行升级期间的停机时间。可以在将一台服务器作为主服务器运行的同时对另一台服务器进行升级。

还可以使用 HotStandby 来帮助选择速度与安全的定制均衡。HSB 参数 **SafenessLevel** 和 **2SafeAckPolicy** 用于控制辅助服务器确认事务的方式。与记录相关的 **DurabilityLevel** 参数与此参数组合使您可以指定速度与安全的组合。在非 HSB 服务器上, 某些参数设置实际上可以提高性能。(有关更多详细信息, 请参阅第 46 页的『性能调整』中对耐久性级别和安全性参数的讨论。)

另一方面, 可以将 **SafenessLevel** 参数设置为 auto 值, 以允许安全性级别随耐久性级别而动态更改。

服务器 HotStandby 状态

在 HotStandby 系统中, 每个服务器都处于其中一种可能存在的状态, 此状态描述了服务器的当前行为。

例如, 当主服务器和辅助服务器正在进行通信和同步时, 它们分别处于 PRIMARY ACTIVE 和 SECONDARY ACTIVE 状态。再举一个例子, 如果主服务器与辅助服务器失去联系, 那么主服务器将自动切换至 PRIMARY UNCERTAIN 状态。在此状态下, 它将不接受新的事务。用户或者 HAC (后者是更典型的情况) 可以将服务器切换至 PRIMARY ALONE 状态。在这种状态下, 服务器充当一个独立服务器, 它接受新事务并存储它们以便稍后发送至辅助服务器。

将 HotStandby 与高级复制技术结合起来使用

可以将 solidDB HotStandby 组件与 solidDB 高级复制技术结合起来使用。高级复制技术将定期进行双向数据同步, 从而允许您创建一个同时包含主服务器和副本服务器的分布式系统。借助 HotStandby, 可以使分布式系统的任何数据库服务器具备高可用性。

第 6 页的图 3 显示了一个简单的分布式系统, 它包含一个主数据库和两个副本数据库。每个副本数据库至少包含主数据库的一部分数据。借助 HotStandby 复制, 每个数据库服务器都已经具备容错能力。将在数据库服务器层次结构的主服务器之间进行高级复制。如果任何主数据库服务器发生问题, 那么失败的节点可以进行 HotStandby 故障转移, 从而使该节点的辅助服务器成为新的主服务器。现在就可以继续对新的主服务器进行高级复制。

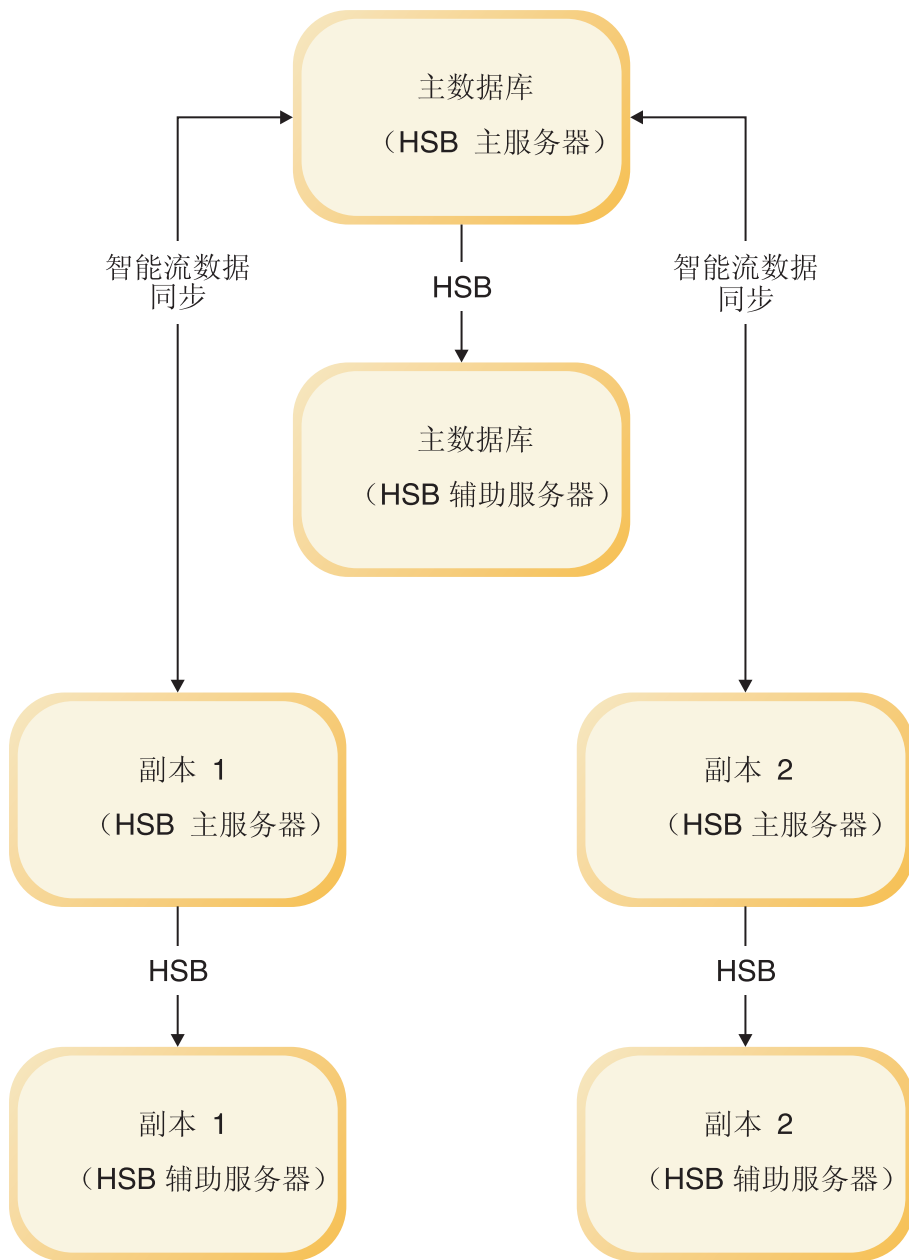


图 3. 由 HotStandby 以及主服务器和副本服务器组成的方案

故障转移

在故障转移中，辅助服务器将切换为新的主服务器。

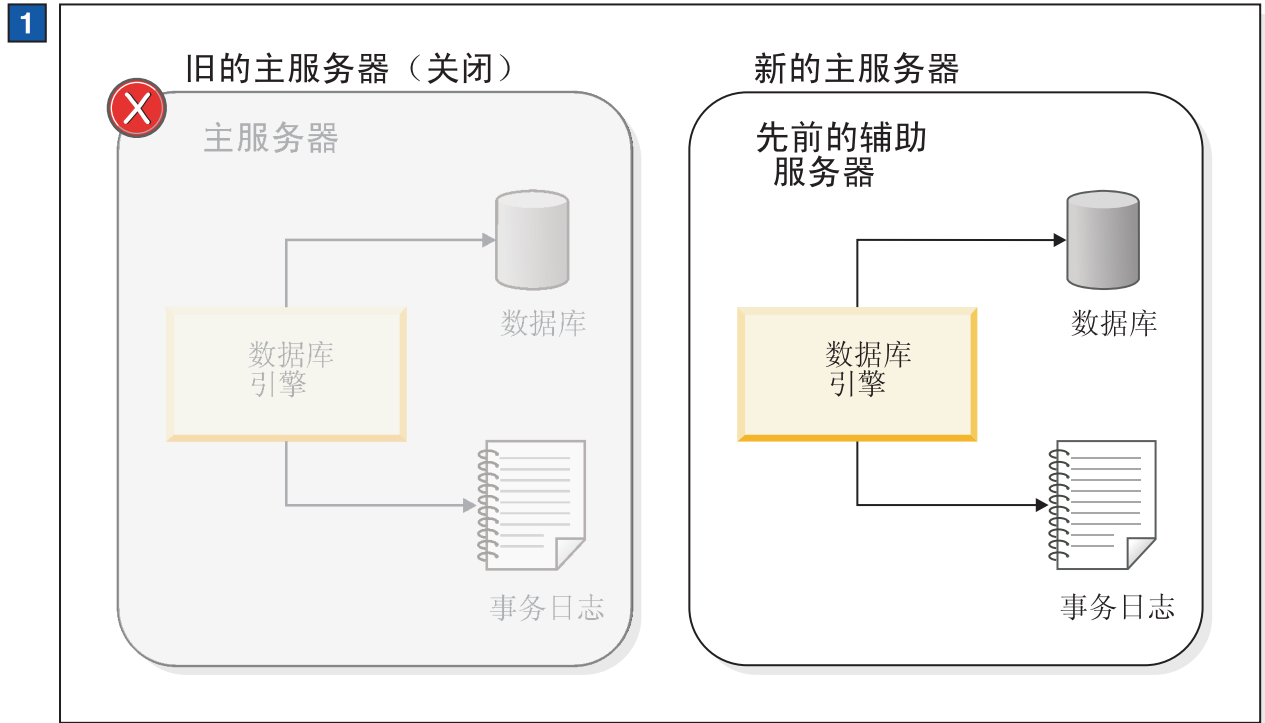
有多种原因会造成辅助服务器切换为新的主服务器：

1. 当主服务器失败时
2. 当您想管理主服务器时
3. 当系统上没有现成的主服务器，而您必须选择一个主服务器时。

通过在辅助服务器上发出以下命令使辅助服务器成为新的主服务器：

ADMIN COMMAND 'hotstandby set primary alone';

进行故障转移时，新的主服务器中将包含旧的主数据库中最新落实的数据。将保证可以在辅助数据库中找到主数据库中已落实的所有数据。如果使用了“透明连接”（TC），那么在故障转移时就不会丢失连接。但是，正在进行的事务将异常终止，必须重新执行此事务。有关更多信息，请参阅第 54 页的『TC 中的故障透明性』。新的主服务器可以单独运行，并继续将事务和数据写入它的数据库和事务日志。

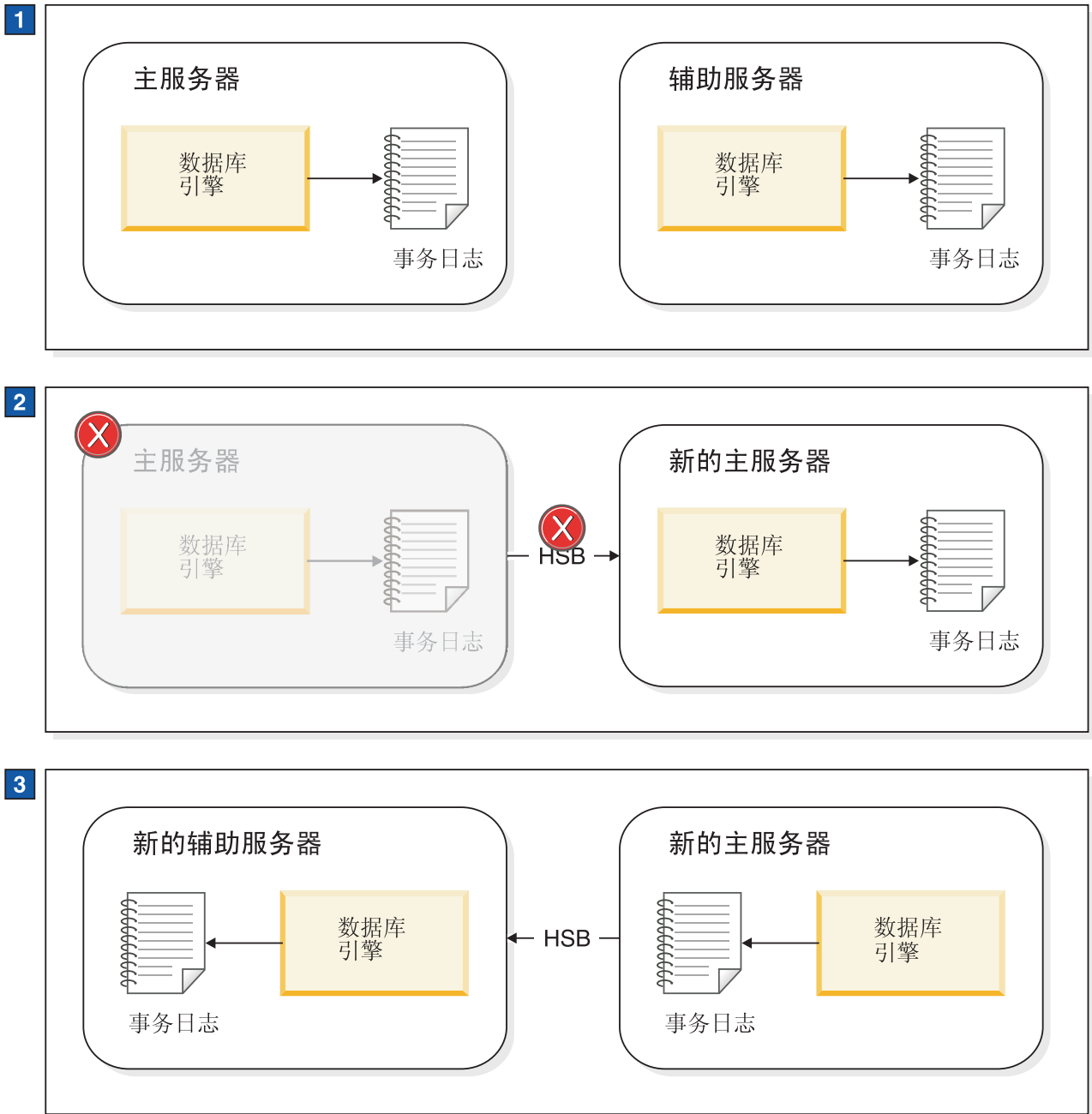


1. 在旧的主服务器失败之后，原来的辅助服务器变成了新的主服务器。

图 4. HotStandby 切换到新的主服务器（也就是旧的辅助服务器）

服务器同步复制

如果在旧的主服务器重新联机时已经有一个现成的主服务器，那么它将成为新的辅助服务器。此时，由于已经将新的事务落实到新的主数据库中，因此新的辅助服务器中的信息落后于新的主服务器中的信息。为了使新的辅助服务器中的信息也是最新的，在连接服务器之后将新的主服务器的事务日志数据自动发送至新的辅助服务器。所有暂挂的更改将从事务日志写入新的辅助服务器，以便辅助服务器可以与主服务器保持同步。第 8 页的图 5 中说明了服务器同步复制。



1. 正常操作：主服务器将数据发送至辅助服务器。
2. 当主服务器失败时，辅助服务器将接替它成为新的主服务器。新的主服务器会将事务信息保存在它的事务日志中，以便它稍后可以将数据发送至新的辅助服务器。
3. 在旧的主服务器作为新的辅助服务器重新运行之后，就会将新的主服务器的事务日志中的信息发送至新的辅助服务器，以便它可以执行同步复制。

图 5. 服务器故障转移和同步复制示例

HotStandby 中的复制方式

1-Safe 和 2-Safe 复制

solidDB 提供了各种选项来调整系统，使它的性能与持久性之间达到必需的均衡。这样的一个选项是所使用的复制协议选项。一个称为 **SafenessLevel** 的系统参数可用来确定复制协议是同步的 (*2-safe*) 还是异步的 (*1-safe*)。

- 1-safe: 首先在主服务器中落实事务，然后再将此事务发送到辅助服务器中。
- 2-safe: 在未得到辅助服务器确认之前不落实事务 (这是缺省值)。

可以在以下三个级别来控制安全级别: 全局 (服务器)、会话和事务。

使用 2-safe 复制的同步 HotStandby

solidDB 主要使用一种同步 *HotStandby* 模型来确保主服务器和辅助服务器具有完全相同的数据。这种方法称为“2-safe”复制方法; 在让用户知道已经落实了数据之前会将数据写入两个位置。

这就意味着, 当主服务器已经写入了数据时, 它不会立即让用户知道已经落实了事务; 主服务器将一直等到辅助服务器也已经落实了数据 (或者至少已经接收到数据) 之后, 才会让用户知道已经落实了数据。²

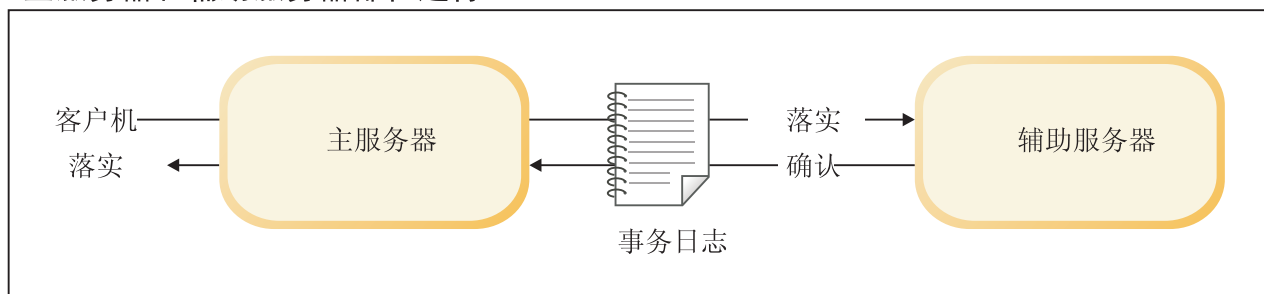
在落实对主数据库中的事务所作的更改之前, 主服务器会将事务数据发送至辅助服务器。辅助服务器必须向主服务器发送一条确认消息, 声明它已经落实了 (或者至少已经接收到) 数据。否则, 主服务器将超时, 并且其状态会从 **PRIMARY ACTIVE** 更改为 **PRIMARY UNCERTAIN** (在后续章节中将更详细地讨论这些状态)。在这种情况下, 主服务器既不能回滚也不能落实事务。HAC 可以将主服务器设置为 **PRIMARY ALONE** 状态。在这种状态下, 主服务器就可以继续接收事务并独立于辅助服务器运行。它将落实已发送至辅助服务器的暂挂事务并继续接受新的事务。

注:

一旦辅助服务器已经落实了 (或者至少已经接收到) 事务日志条目, 它就会发送一条确认消息。此配置可以防止在发生单一故障时丢失事务。此外, 可以选择保留基于文件的事务日志, 以便于在整个系统发生故障时恢复数据库。

2. 有关 2-safe 与 1-safe 算法的更多信息, 请参阅由 Jim Gray and Andreas Reuter 编写, Morgan Kaufman 出版社在 1993 年出版的《事务处理: 概念和技术》。

主服务器和辅助服务器都在运行



主服务器运行，辅助服务器关闭

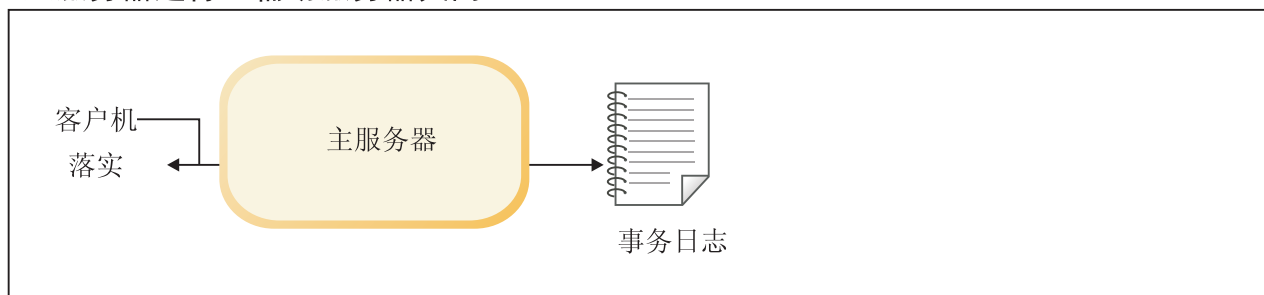


图 6. 同步 HotStandby 配置

发送数据时执行的基本步骤

下面是使用同步复制发送数据时将执行的步骤:

1. 主服务器将数据（采用记录级别格式）写入主节点中的事务日志。
2. 如果主服务器遇到一个 `commit` 语句，那么所有已更改的数据都会被发送至辅助服务器。

注：如果辅助服务器在事务启动之后到主服务器发送数据之前这段时间内失败，那么主服务器将回滚此事务。

3. 辅助服务器将确认落实消息。进行确认所需的时间取决于 **2SafeAckPolicy** 配置参数的设置。在最快的备用方式下（称为 *2-safe received*），辅助服务器一接收到落实消息就立即向主服务器发送确认消息。在最安全的备用方式下（称为 *2-safe durable*），辅助服务器将在执行并且持久地将事务写入辅助服务器自己的事务日志之后才发出确认消息。

当主服务器接收到辅助服务器的确认消息时，主服务器就会通知用户已经落实了数据。

4. 如果主服务器未接收到来自辅助服务器的确认消息（例如，因发生网络故障或节点故障而未接收到），那么主服务器将超时并且切换到 **PRIMARY UNCERTAIN** 状态。由于主服务器不知道辅助服务器中的最新事务的状态，因此无法回滚或落实事务。主服务器不知道发生了下面哪种情况：
 - 辅助服务器在落实事务之前已关闭
 - 辅助服务器已经落实了事务，但是主服务器未接收到确认消息（例如，由于发生网络故障而导致未接收到）。

当服务器处于 PRIMARY UNCERTAIN 状态时，用户试图落实的当前事务和新事务将被阻塞，而用户可能理解为服务器无响应。

5. 如果 HAC 检测到辅助服务器已关闭或者发生网络故障，那么它可以将主服务器切换为 PRIMARY ALONE 状态。一旦主服务器设置为 PRIMARY ALONE，它就会落实已发送至辅助服务器的暂挂事务并继续接受新的事务。
6. 更改将被累积到事务日志文件中，直到辅助服务器恢复运行或者主服务器用尽磁盘空间为止。（如果服务器用尽了用于保存事务日志的磁盘空间，那么主服务器将更改为只读方式。）
7. 如果您认为辅助服务器将停止运行较长时间，并且服务器可能用尽用于保存事务日志的磁盘空间，那么您可能希望将主服务器从 PRIMARY ALONE 状态切换为 STANDBY 状态。这就意味着事务日志将不会存储自从与辅助服务器失去联系以来发生的所有事务，因此，辅助服务器仅仅通过从主服务器中读取事务日志将无法进行“同步复制”。如果辅助服务器无法使事务日志是最新的，那么唯一的办法就是使辅助服务器与主服务器同步，以将主服务器的数据库文件复制到辅助服务器中。这可以通过执行 HotStandby copy 或 HotStandby netcopy 命令来实现。有关 copy 和 netcopy 命令的详细信息，请阅读第 82 页的『通过网络将主数据库复制到辅助服务器』和第 85 页的『将数据库文件从主服务器复制到指定目录』。
8. 要执行 copy 或 netcopy 命令，主服务器必须处于 PRIMARY ALONE 状态。执行 copy/netcopy 命令之后，无论此命令成功还是失败，主服务器仍将处于 PRIMARY ALONE 状态。
9. 为了使主服务器再次开始将它的事务发送至辅助服务器，主服务器必须使用 hotstandby connect 命令显式地连接至辅助服务器，第 86 页的『连接 HotStandby 服务器』中描述了此命令。在主服务器连接至辅助服务器之后，主服务器将在 PRIMARY ACTIVE 状态下运行。

一旦两个服务器之间建立了连接，它们就会开始执行同步复制（此时会自动将所有暂挂的更改从事务日志写入辅助服务器，以便与主服务器保持同步）。注意，在服务器执行同步复制之前，主服务器与辅助服务器将交换信息并确定在何处开始执行同步复制，以便在辅助服务器上不会将事务落实两次。

使用 1-safe 复制的异步 HotStandby

（可选）可以使用从主服务器异步复制到辅助服务器。这种复制方法称为 *1-safe* 复制。

采用 1-safe 复制时，一旦在主服务器中落实了事务就会立即确认这些事务。这将显著提高性能。落实事务之后，采用异步方式将它们发送至辅助服务器。这种方法也存在一定弊端，当主服务器中发生故障时，当时正在转移的一些事务可能会丢失。

可以动态选择这两种复制方法中的任一方法，甚至可以对每个会话或事务选择复制方法。也可以控制 1-safe 复制涉及到的复制延迟。

对服务器状态的描述

组成一个 HotStandby (HSB) 对的两个服务器都将呈现某些可以查询和控制的狀態。

下面列示了各种状态及其含义。

表 4. 对服务器状态的描述

状态	描述
PRIMARY ACTIVE	已连接服务器，并且此服务器（主服务器）正在接受读写事务并且正在将数据发送至辅助服务器。辅助服务器必须处于 SECONDARY ACTIVE 状态。
PRIMARY ALONE	<p>对等服务器之间未互相连接。对等服务器可能在运行，但是未建立连接，因此它未接受任何事务（即，它可能处于 SECONDARY ALONE 状态）。</p> <p>此服务器（即，主服务器）正在积极接受和执行读写事务，并且正在收集这些事务以便稍后发送至辅助服务器。</p>

表 4. 对服务器状态的描述 (续)

状态	描述
PRIMARY UNCERTAIN	<p>服务器之间已异常断开连接，并且 AutoPrimaryAlone 配置参数设置为“No”。在 PRIMARY UNCERTAIN 状态下，任何未经确认的事务将保持暂挂状态，这意味着在 HAC 将此服务器更改为另一种状态之前，此服务器将不会落实或回滚事务。</p> <p>操作员可以选择执行以下三项操作：将主服务器重新连接至辅助服务器；将主服务器设置为 PRIMARY ALONE 状态；将主服务器设置为 SECONDARY ALONE 状态。</p> <ol style="list-style-type: none"> 1. 如果主服务器已重新连接至辅助服务器，就会在主服务器上落实事务。 2. 如果状态更改为 PRIMARY ALONE，就会在主服务器上落实已打开的事务。 3. 如果状态更改为 SECONDARY ALONE，那么已打开的事务将保持暂挂状态。将在服务器更改为另一种状态之后最终解决这些事务。例如，如果服务器更改为 SECONDARY ACTIVE 状态，那么将根据同步复制结果异常终止或落实被阻塞的事务。如果服务器状态更改为 STANDALONE 或 PRIMARY ALONE，就会落实被阻塞的事务。 <p>如果您希望主服务器在与辅助服务器失去联系时就自动进入 PRIMARY ALONE 状态而不是 PRIMARY UNCERTAIN 状态，那么请阅读关于 AutoPrimaryAlone 配置参数的描述。</p> <p>注：</p> <p>HAC 可以始终使服务器从 PRIMARY UNCERTAIN 状态切换到 SECONDARY ALONE 状态，从而获得最高安全性。这样可以防止存在双重主服务器的可能性。但是，这样也会阻止用户更新服务器上的数据。（请参阅第 35 页的『网络分区和双重主服务器』。）</p>
SECONDARY ACTIVE	<p>对等服务器之间已互相连接，并且此服务器正在接受来自主服务器的人局事务日志数据。将在辅助服务器中执行这些事务，以使它与主服务器具有相同的数据（还会将这些事务写入辅助服务器自己的事务日志，以便在辅助服务器失败的情况下它自己就可以恢复数据）。此外，客户机可以执行处于 SECONDARY ACTIVE 状态的服务器上的只读事务。当服务器处于 SECONDARY ACTIVE 状态时，此服务器的对等服务器必须处于 PRIMARY ACTIVE 状态。</p>

表 4. 对服务器状态的描述 (续)

状态	描述
SECONDARY ALONE	辅助服务器已与其对等服务器断开连接。它仅接受读取请求。服务器可以通过在辅助服务器或主服务器发出“HotStandby connect”命令来连接至对等服务器。
STANDALONE	服务器没有 HSB 状态（主服务器或辅助服务器），且以常规独立服务器的运行方式运行。同时也按正常方式处理和除去事务日志；不会为辅助服务器保存事务日志。要继续执行 HSB 操作，必须将服务器设置为 PRIMARY ALONE 或 SECONDARY ALONE 状态，并且主服务器将必须执行 netcopy 或 copy 操作以将数据库的完整副本发送至辅助服务器。
OFFLINE	<p>服务器是以“netcopy 侦听方式”（也称为“备份服务器方式”）启动的。在此方式下，服务器将等待来自处于 PRIMARY ALONE 状态的服务器的人局 netcopy。在服务器成功完成 netcopy 之后，服务器将更改为 SECONDARY ALONE 状态。</p> <p>因为处于 OFFLINE 状态的服务器不接受客户机连接，所以您无法直接观察 OFFLINE 状态。如果您尝试连接至处于 OFFLINE 状态的服务器，那么将产生错误代码 14552，表示“服务器采用备份服务器方式，不允许建立连接”。处于 OFFLINE 状态的服务器将只对 netcopy 操作作出响应（稍后将描述此操作）。</p>

HotStandby 如何影响性能

尽管您可能预计 HotStandby (HSB) 会降低性能，但并不会始终都是这种情况。实际上，在某些配置中，HSB 甚至可以显著提高性能。

影响 HSB 性能的关键因素包括：

- 在需要对每个故障保留事务时使用自适应耐久性
- 在可接受故障发生次要事务丢失时使用 1-Safe 复制协议
- 2-Safe 确认策略（当使用 2-safe 复制时）- 即，辅助服务器是一旦从主服务器中接收到事务就立即确认它接收到了事务，还是等到它已经落实了事务之后才确认它接收到了事务
- 是否可以在辅助服务器上执行只读事务
- 服务器的内部并行性

总之，除非您需要尽可能获得最高级别的安全性，否则可以通过执行下列操作来提高性能：

- 使用自适应记录（即，设置 **DurabilityLevel=2**）
- 使用“2-safe received”方式（即，设置 **2SafeAckPolicy=1**）

如果您具有只读查询（例如，用于生成摘要报告的查询），那么您可能希望在辅助服务器上运行这些查询，以便由两台机器共同分担工作负载。

请注意，即使您使用此处指定的“不太安全的”设置（“自适应耐久性”和“2-safe received”方式），您仍然会受到 HSB 的保护，除非至少发生了两个故障。这样，您通过牺牲很小的安全性为代价，可以使性能获得大幅提高。solidDB HotStandby 同时为您提供了高性能和高安全性。

实际的吞吐量和响应时间取决于很多因素，其中包括（但不仅限于）网络速度、网络上的其他流量的大小、SQL 语句的复杂程度以及每个事务的 SQL 语句数。当然，一些常见因素（例如，内存量和磁盘速度）也会影响性能。

自适应耐久性

solid.ini 配置文件允许您指定您是想要宽松耐久性（快速）、严格耐久性（安全）还是称为“自适应耐久性”的第三个选项。

- **严格耐久性**：如果一旦落实了事务就将此事务写入事务日志，这就称为“严格耐久性”。这种耐久性提供了最高安全性。
- **宽松耐久性**：如果允许服务器延迟到它不太忙或者可以同时写入多个事务时才写入事务，这就称为“宽松耐久性”（或者“宽松记录”）。在一个不属于 HSB 对的服务器中，如果使用宽松耐久性，那么在服务器异常终止的情况下您就存在丢失最新的一些事务的风险。但是，如果服务器是 HSB 对的一部分，那么另一个服务器（即，辅助服务器）上存在事务的副本，即使在记录事务之前主服务器失败，也不会丢失事务。因此，对 HSB 使用宽松耐久性时，宽松耐久性将导致安全性稍微降低。另一方面，宽松耐久性可以提高系统的性能，尤其是在服务器的负载由许多较小的写入事务组成时。
- **自适应耐久性**：自适应耐久性仅适用于 HotStandby 主服务器（这是缺省情况）。自适应耐久性仅仅意味着，如果服务器处于 Primary Active 状态（将事务发送至辅助服务器），那么它将使用宽松耐久性；但是在任何其他状态下它将使用严格耐久性。因此，当 HSB 处于活动状态时，您将获得高性能（安全性稍微降低）；如果只有一个服务器在运行，您还将保持较高的安全性。仅当使用 2-Safe 复制时才能有效地实施自适应耐久性（这是缺省情况）。

自适应耐久性可以显著提高性能，同时又能在故障情况下保证具有较高的数据安全程度。它可以提高整个系统的吞吐量并且可以缩短等待时间，即，用户在被告知已经落实了事务之前必须等待的时间。

有关宽松记录和 **DurabilityLevel** 参数的更多详细信息，请参阅《solidDB 管理员指南》。

1-Safe 复制

使用 1-safe 复制时，一旦在主服务器中完成了落实处理，就会立即确认 commit 语句。在已将控制权返回给应用程序之后，会将已落实的事务异步传输至辅助服务器。

传输事务时涉及到的延迟可能从几毫秒到几百毫秒。由于主服务器中的等待时间急剧缩短，因此 1-safe 复制将显著提高性能。1-safe 复制的不足之处在于，在发生故障的情况下，某些事务可能会在故障转移时丢失。

对于服务器，可以使用以下参数来设置 1-safe 复制：

[HotStandby]
SafenessLevel=1safe; 值: 1safe、2safe 或 auto; 缺省值为 2safe

还可以使用 SET 命令来动态控制安全级别:

```
SET SAFENESS {1SAFE| 2SAFE| DEFAULT}
```

设置当前会话的安全级别, 直到它更改为止。

```
SET TRANSACTION SAFENESS {1SAFE| 2SAFE| DEFAULT}
```

设置当前事务的安全级别。落实之后, 安全级别可以重新恢复为会话设置的值、启动值或者系统缺省值 (即, 2-safe)。

选项 DEFAULT 表示会话的当前设置。

当 **SafenessLevel** 参数具有特殊值“auto” (即, “automatic”) 时, 可以使用程序耐久性控件 (例如, SET DURABILITY RELAXED) 来控制安全级别。在这种情况下, “严格”对应于“2-safe”, “宽松”对应于“1-safe”。

2-Safe 确认策略

当启用了 2-safe 复制时 (这是缺省情况), 主服务器要在接收到辅助服务器已具有事务的确认消息之后, 才会让客户机知道已经成功地落实了此事务。

solidDB 当前允许以下三种不同的确认策略:

- 2-safe received: 当辅助服务器接收到数据时, 它才会发送确认消息 (这是缺省情况)。
- 2-safe visible: 辅助服务器已更新它的数据副本, 并且更改现在“可视”。换句话说, 已连接至辅助服务器的客户机应用程序将能够查看更新。
- 2-safe durable: 辅助服务器将在它已使数据成为持久数据之后 (即, 在它已落实数据并且将数据写入磁盘之后) 进行确认。

使用 2-safe received 时速度将更快。使用 2-safe durable 时将更安全。请注意, 由于仅当主服务器和辅助服务器都处于活动状态时 (即, 都在应用事务) 这些确认策略才适用, 因此, 即使 2-safe received 都被认为是安全的。仅当两台服务器在一秒钟之内几乎同时失败, 才会存在丢失事务的风险。

通过使用“2-safe received”可以缩短等待时间, 等待时间就是从开始落实到用户接收到确认已落实这段时间。**2SafeAckPolicy** 对总体吞吐量的影响很小。

内部并行性

为了确保让系统利用并行性, 应考虑将事务分散在多个连接上, 而不是通过同一个连接来提交所有事务。

当您使用 HotStandby (HSB) 组件时, 每个包含写操作的事务都将执行两次 - 在主服务器上执行一次, 在辅助服务器上又执行一次。在某些情况下, 这可能意味着单个事务在使用 HSB 时所花的时间大约是未使用 HSB 时的两倍。但是, 这并不意味着总体吞吐量会降低 50%。服务器具有很高的并行度, 当辅助服务器在处理一个事务时, 主服务器将处理另一个事务。

为了确保让系统利用并行性，可以将事务分散在多个连接上。但是应注意，您并行运行的查询越多，服务器需要的内存就越多。因此，通过添加连接并且并行运行查询并不是始终都会增大吞吐量，尤其是在没有大量内存的系统中。您可能需要通过试验来找到一次运行的最佳查询数。

在辅助服务器上执行只读事务

允许客户机连接至辅助服务器并执行只读操作。

在某些情况下，可以通过将只读客户机连接至辅助服务器并在辅助服务器中执行读取操作，从而实现“分散负载”并提高系统的总体性能。这对于执行生成报告或“数据仓储”查询等工作特别有用。在这些工作中，您希望读取许多记录，但是又不希望更改它们。

其他安全因素

热备用方法旨在防止系统中的单个部件发生故障。但是，如果两个服务器都会受到同一问题的影响（例如，发生电源故障），那么热备用方法也无法保护您。

如果您将 **DurabilityLevel** 设置为“宽松”或者“自适应”，并且您采用的确认策略是“2-safe received”，那么在两个服务器同时停止的情况下可能会丢失事务。每个服务器至少应配备一个不间断电源（UPS）以防发生电源故障。而且，与任何数据库系统一样，应当备份重要数据并且可能应该将它们归档在一个单独的位置。**HotStandby** 并不能代替您备份数据。

提示:

可以在 HSB 对的任一服务器上（使用 **ADMIN COMMAND backup** 命令）运行备份。通常，辅助服务器上具有更多资源可用于创建备份。

高可用性控制器（HAC）

高可用性控制器（HAC）是用于 **solidDB HotStandby** 配置的一个自动冗余管理程序。它通过检测故障、执行故障转移到备用单元并在必要时重新启动失败进程来维护数据库服务的可用性。

数据库节点中发生硬件问题、数据库进程失败或者 **HSB** 链路断开都可能会导致故障。**HAC** 将监视服务器的 **HSB** 状态。当发生故障时，**HAC** 将确保未受此故障影响的服务器充当主服务器角色，并且此服务器已准备好接受事务负载。

换句话说，**HAC** 充当一个看守程序的角色。在 **HAC** 的实现中，**solidDB** 的事件机制用来监视服务器状态。每当 **HSB** 服务器的状态更改时，它就会向 **HAC** 发送一个事件，然后 **HAC** 将推断可能需要使用 **HotStandby** 管理命令 **API** 执行哪些操作。

下图说明了 **HAC** 体系结构:

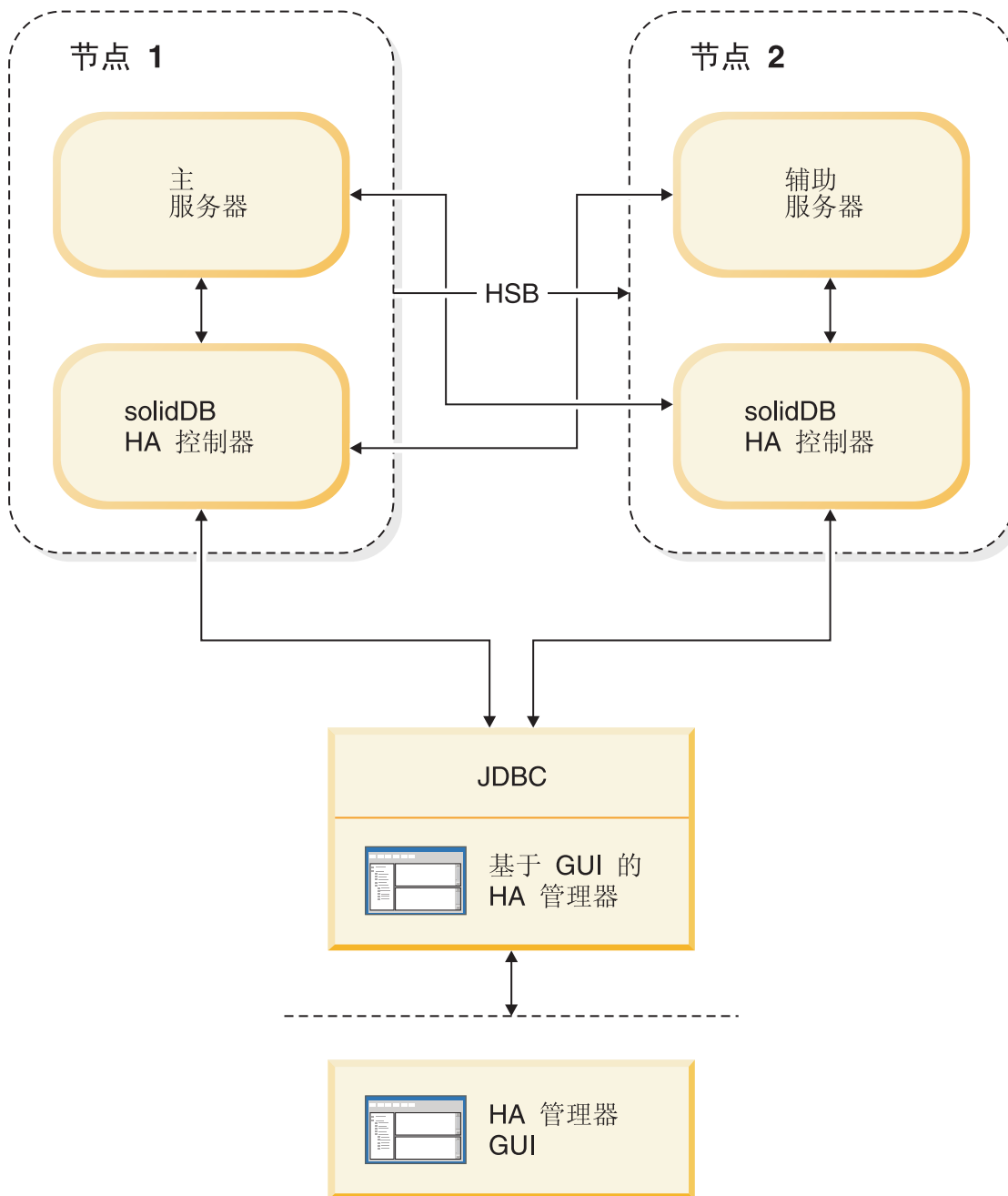


图 7. 高可用性控制器体系结构

HAC 有以下两个主要用途:

1. 可以将 HAC 用作看守程序来自动维护数据库服务的可用性。在此方式下（称为“自动”方式），HAC 将执行下列操作:
 - 启动、重新启动和终止数据库服务器进程（可选）
 - 监视服务器的状态以及它们之间的 HSB 链路
 - 推断必需执行的操作
 - 执行操作

2. HAC 还可以充当 HSB 系统监视器角色，用来监视 HSB 服务器状态的更改（由其他人触发）和报告系统状态。此方式称为“管理”方式。在此方式下，HAC 不会执行任何 HSB 状态过渡或修改 HSB 系统。

还为 HAC 提供了一个称为高可用性管理器（HAM）的图形用户界面（GUI）组件。它包含在样本目录“samples\hac\”中。

可通过 solidhac.ini 配置文件来配置高可用性控制器。在 HAC 启动之前，此文件可能位于 HAC 工作目录中。产品程序包中有一个配置文件模板，它包含所有可用的配置参数和注释以及示例。HA 管理器是以相似的方式配置的。

常见的故障

HAC 将监视 HotStandby 服务器的运行状况和状态。在故障情况下（例如，数据库进程故障或者计算机节点故障），HAC 将执行故障转移和其他必需的状态过渡以维护数据库服务的最佳可用性。

对于所考虑的所有故障，假定它们都是在一种正常并且完全可运行的状态下发生的，也就是用两个 HSB 服务器的 PRIMARY ACTIVE 和 SECONDARY ACTIVE 状态表示的状态。HAC 仅处理单个故障。换句话说，假定在系统从前一个故障恢复之前不能再发生故障。但是，HAC 可以处理许多预定义的多故障情况。

只要是考虑单个故障，HAC 就会维护几乎不间断的数据库服务。如果发生了多个故障，HAC 将尝试避免出现错误的系统状态（例如，双重主服务器）。

HAC 可以处理的故障包括：

- 单个故障
 - 主（ACTIVE）数据库服务器进程失败
 - 辅助（ACTIVE）数据库服务器进程失败
 - 主节点失败
 - 辅助节点失败
 - 如果使用了“外部引用实体”，那么 HAC 还可以处理 HotStandby 链路故障，即，两个 HotStandby 数据库进程之间的连接断开。有关“外部引用实体”的更多信息，请参阅第 20 页的『外部引用实体（ERE）』。
- 双重故障
 - 在从前一个故障恢复时，HAC 将识别主数据库与辅助数据库之间同步过程中发生的错误。
 - HAC 也会注意一些不常见的故障，例如，在前一个故障之后，当服务器建立 HSB 链路时服务器进程发生故障。

控制数据库服务器进程

可以配置 HAC 来启动数据库进程和重新启动失败进程。

当 HAC 实例与本地数据库服务器断开连接时，它将调用在 solidhac.ini 配置文件中指定的启动脚本。此脚本由用户提供。随程序包一起提供了一个示例脚本。

要点: HAC 实例假定服务器正在运行，并且在启动脚本终止时将作出响应。由于 HAC 不会处理服务器启动时发生的故障，因此，除非服务器接受连接，否则脚本不应退出。

除了以下两种情况之外，每当数据库服务器失败或者因其他某些原因而消失时，HAC 就会重新启动数据库服务器：

- 数据库进程尚未从操作系统的进程列表中消失；或者
- 使用 HA 管理器关闭了数据库服务器。

外部引用实体 (ERE)

要处理的更困难的一种故障情况是数据库节点之间的通信链路失败，而两个数据库服务器可能都认为对方已关闭。这可能会导致双重主服务器情况（这就是所谓的“裂脑”），并且稍后使数据库同步时可能会丢失事务。为了更好地避免执行 solidDB HAC 作出错误决定，建议您使用“外部引用实体”（ERE），它是一个用于检查网络运行状况的网络引用设备。例如，当一台计算机中的网络适配器发生故障时，HAC 可以检测到这种情况，并且能够设置正确的数据库节点继续作为主数据库（而另一个数据库节点继续作为辅助数据库）。

如果使用了 ERE，那么 HAC 将通过对 ERE 执行 ping 操作来检查 HotStandby 节点与 ERE 设备之间的物理链路的状态。如果与离得最近的 ERE 的物理链路不起作用，那么本地 HAC 会将本地服务器设置为 SECONDARY ALONE 状态。如果离得最近的物理链路起作用，并且没有连接可用于其他服务器，那么本地 HAC 将断定本地服务器就是继续提供服务的服务器并将它设置为 PRIMARY ALONE。因此，HotStandby 节点（与它相对应的 HotStandby 节点和离得最近的 ERE 之间的连接断开）成为辅助服务器。这样，就可以防止在发生网络故障的情况下出现双重主服务器（这就是所谓的“裂脑”）情况。

有关为 ERE 配置 HAC 的更多信息，请参阅第 47 页的『配置 HA 控制器和 HA 管理器』。

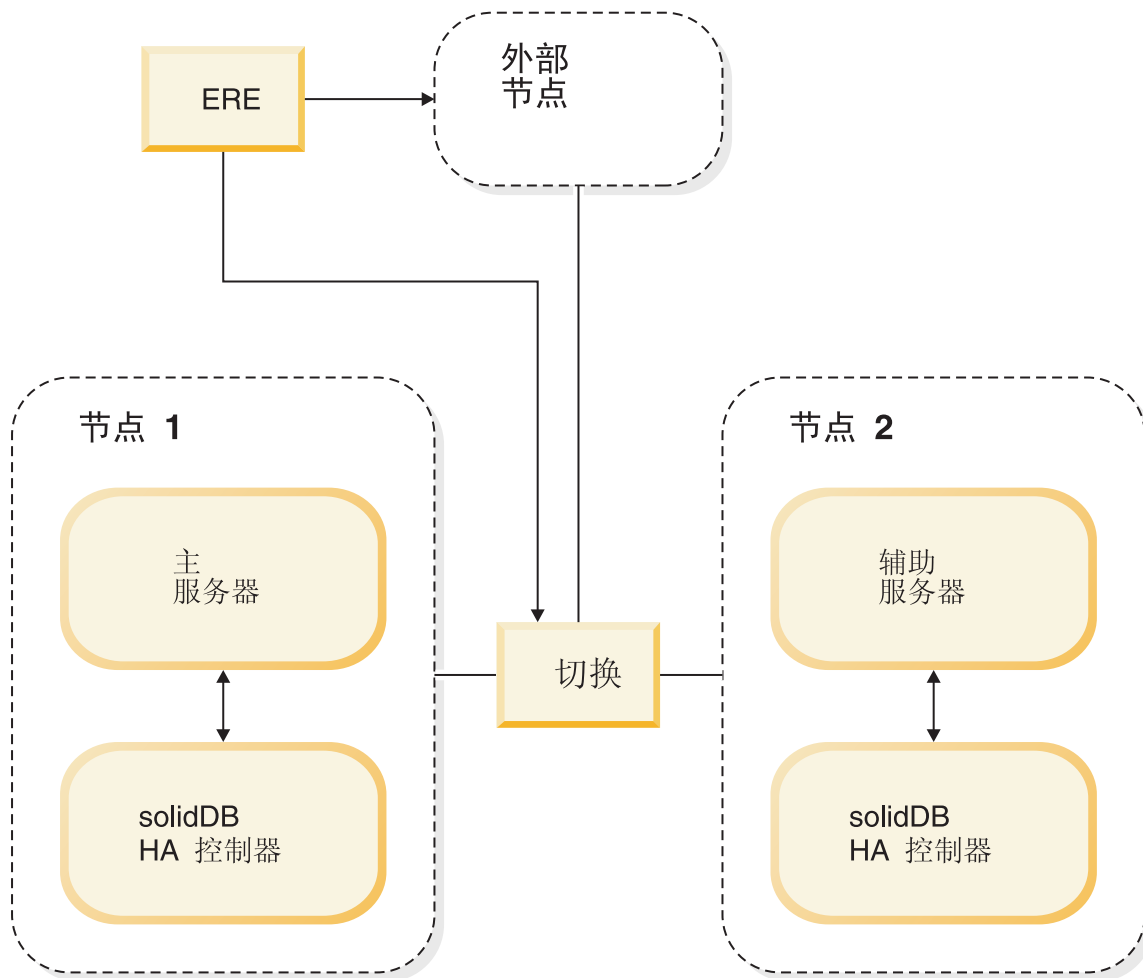


图 8. “外部引用实体”组件

上图说明了 ERE 可能用于的两个位置:

- 组开关
- 位于集群之外的网络中的任何计算机。如果在集群中使用了冗余网络（也就是说使用完全一样的两套网络控制器、电缆和交换机），那么建议您在集群外部定义 ERE。

要点: 如果认为 HotStandby 链路不可靠（包括所有使用了 ERE 的情况），那么必须将以下 HotStandby 服务器参数设置为出厂值:

HotStandby.AutoPrimaryAlone=no

ERE 必须使用保持活动消息使用的同一 HSB 链路。

在 HAC 中联网

建议在每个 HSB 服务器中使用单个逻辑网络接入（IP 地址）来完成所有通信。

此建议并不排除使用在网络接口 API 层透明的多个或者冗余的较低级别的网络组件（网络接口卡、电缆和交换机）。

服务器可以将不同的端口号用于不同用途。

当仅使用一个逻辑网络接入时，HAC（带有 ERE）将检测网络中断，网络中断将影响 HSB 事务复制和数据库客户机应用程序。由于 HAC 仅监视 HSB 链路的运行状况，因此，如果使用了独立的接口，HAC 将不会检测数据库客户机通信故障。

但是，无论底层的网络接入实现方案如何，如果网络接入因任何原因而被认为不可靠，就可以使用 HAC 的 ERE 功能部件。

高可用性管理器

高可用性管理器是一个基于 Java 的图形界面工具，用于显示 HotStandby 服务器状态和 HAC 状态。它还提供了用于管理 HAC 的基本功能，例如，通过切换 HotStandby 服务器的角色以及暂挂和恢复 HAC 来管理 HAC。

可通过 HAManager.ini 配置文件来配置高可用性管理器。

请参阅下面提供的高可用性管理器的屏幕快照：



图 9. 高可用性管理器

可以使用高可用性管理器的 GUI 来执行下列操作：

- 在 HotStandby 服务器之间执行切换
- 在“自动”与“管理”HAC 方式之间切换
 - 在“管理”方式下，HAC 将仅监视 HSB 集群，并且用户可以在 HSB 服务器上执行管理任务
 - 在“自动”方式下，HAC 充当看守程序，用于处理故障和维护数据库服务的可用性。

- 关闭 HotStandby 数据库服务器进程
- 启动 HotStandby 数据库服务器进程
- 暂挂高可用性控制器
- 恢复高可用性控制器

注:

仅当在 solidhac.ini 配置文件中设置了下列配置参数时, 才能关闭和启动数据库服务器进程:

HACcontroller.EnableDBProcessControl=Yes

HAC 日志记录

HAC 将日志记录写入 HAC 工作目录下的 hacmsg.out 文件中。

日志中包含有关下列各项的信息:

- 警告
- 致命错误
- 与配置相关的信息
- 与初始化相关的信息
- 所有输入事件
- HotStandby 状态更改
- 用户命令, 导致系统中发生更改
- HAC 状态更改
- HAC 方式更改 (自动/管理)
- 事件, 导致系统中发生状态更改

HAC 日志文件的最大大小为 64 MB。当超过了大小限制时, hacmsg.out 被重命名为 hacmsg.bak, 并且会创建一个新的 hacmsg.out。这些文件中最多包含 128 MB 的最新日志。

2 HotStandby 入门

本节提供了有关设置两个 solidDB HotStandby 服务器（一个主服务器和一个辅助服务器）的逐步指示信息。

假定您已经安装了 solidDB HotStandby。请务必遵循随此产品一起提供的安装指示信息。

第 33 页的 3 章，『管理和配置 HotStandby』中提供了有关设置高可用性控制器（HAC）的指示信息。

开始之前

关于此任务

在设置 HotStandby 之前，请注意以下信息：

1. 请阅读第 33 页的『您应当知道的信息』。本节中包含有关使用 solidDB HotStandby 组件的重要信息。
2. 要了解 HotStandby 功能部件，请运行 solidDB 程序包中包含的演示程序。有关详细信息，请阅读下面一节。
3. 请参阅『HotStandby 快速启动过程』以了解有关设置和配置 HotStandby 的逐步指示信息。
4. 对 solidDB 命令行 SQL 编辑器 solsql 有所了解。

HotStandby 演示程序

已安装的 solidDB 软件包中包含您在运行使用 HAC 或样本看守程序实现的 HotStandby 组件的演示程序时需要的所有文件。演示程序经过了简化，但是可以帮助您理解如何使用 HotStandby 组件。

可以在 samples\hac 目录下找到 HAC 演示程序。可以在 samples\hac\readme.txt 文件中找到有关 HSB/HAC 演示程序的详细信息。

有关设置 HAC 系统的指示信息，请参阅第 28 页的『使用了 HA 控制器的 HSB 的快速启动过程』。

samples\hsb\readme.txt 文件中包含有关看守程序演示程序的详细指示信息。

HotStandby 快速启动过程

本节描述 HotStandby 的快速启动过程。完成此过程之后，HSB 系统将可以开始为应用程序服务。

关于此任务

除了 HSB 服务器之外，此方法不假定任何其他 solidDB 组件。例如，不需要 HAC。第 27 页的『启动和停止 HA 控制器』中描述了用于 HAC 的类似逐步完成的过程。solidDB HotStandby 程序包中还包含一个样本看守程序。要使用此样本看守程序，需要为它提供配置设置。

要设置和运行 HSB 服务器对，需要两台互相联网的计算机。要设置 HotStandby 服务器（它们没有任何其他 solidDB 组件），请遵循以下过程。

1. 配置主节点和辅助节点。

HotStandby 要求您至少在 solid.ini 配置文件的 [HotStandby] 节中配置下列参数：

- **HSBEnabled=Yes**

如果您省略了 **HSBEnabled** 参数，或者在您打算用作 HotStandby 服务器的服务器中将此参数的值设置为“no”，那么当此服务器启动时将是一个非 HotStandby 服务器。

- **Connect=connect string for the opposite HSB server**

如果您省略 **Connect** 参数，那么服务器将作为 HotStandby 服务器启动，但是必须在您通过使用 ADMIN COMMAND 提供连接字符串之后才能连接服务器。

可以对 HotStandby 服务器启用或禁用事务日志记录以便用于本地恢复。如果禁用了日志记录，那么主服务器会将事务日志信息的必需部分保留在内存中以供复制。禁用事务日志记录可以提高写入事务的性能，但是会降低数据的安全程度。缺省情况下，已启用事务日志记录。可以通过在 solid.ini 配置文件中设置以下参数来禁用事务日志记录：

```
[Logging] LogEnabled=no
```

除了上述参数以外，还可以使用其他参数。有关这些参数的更多详细信息，请阅读第 43 页的『定义主服务器和辅助服务器 HotStandby 配置』。

如果您使用的是 TCP/IP 协议，那么您可能需要调整应用程序与 HotStandby 服务器之间的超时时间间隔。有关详细信息，请阅读第 41 页的『定义应用程序与服务器之间的超时（[Com] 节）』。

（可选）可以更改特定于 HotStandby 的配置参数选项的缺省设置。有关这些参数的详细信息，请阅读第 43 页的『配置特定于 HotStandby 的参数』。

2. 采用您将用来启动任何 solidDB 服务器的方式来启动两个 HSB 服务器。

这两个服务器将从它们自己的 solid.ini 文件中读取 HotStandby 配置信息。这两个服务器在启动之后的 HSB 状态为 SECONDARY ALONE。

3. 选择将成为主服务器的服务器，并通过发出以下命令将所选服务器的状态切换为 PRIMARY ALONE:

```
ADMIN COMMAND 'hotstandby set primary alone';
```

注：在这些管理命令中，可以使用缩写“hsb”来代替“hotstandby”，例如：

```
ADMIN COMMAND 'hsb set primary alone'
```

4. 通过在任一服务器中发出以下命令将主服务器连接至辅助服务器:


```
ADMIN COMMAND 'hsb connect'
```

要验证连接是否成功，请输入以下命令：

```
ADMIN COMMAND 'hsb state'
```

主服务器的状态应为“PRIMARY ACTIVE”。但是，如果主服务器的状态是预期状态之外的其他状态（例如，PRIMARY ALONE），那么可以通过输入以下命令来检查 hsb connect 的状态：

```
ADMIN          COMMAND 'hsb status connect'
```

如果获得的结果为 ACTIVE，那么连接进程仍然处于活动状态。但是，如果结果为 BROKEN，那么在连接服务器之前必须使它们的数据库同步。

5. 可以通过在主服务器中执行以下命令来使数据库同步：

```
ADMIN COMMAND 'hsb netcopy'
```

可以通过输入以下命令来检查数据库复制进程的状态：

```
ADMIN COMMAND 'hsb status copy'
```

6. 一旦结果为 SUCCESS，就说明数据库之间已同步，因此可以连接服务器了。在任一服务器中重新执行以下命令：

```
ADMIN COMMAND 'hsb connect'
```

验证连接是否成功，如步骤 4 中所述。

7. 开始使用应用程序。

启动和停止 HA 控制器

必须在正确填写了 HAC 工作目录下的 solidhac.ini 配置文件之后才能启动 HAC。有关配置 HAC 的简要描述，请参阅第 28 页的『使用了 HA 控制器的 HSB 的快速启动过程』。可以在第 47 页的『配置 HA 控制器和 HA 管理器』和第 50 页的『solidhac.ini 配置文件』中找到有关配置 HAC 的更完整描述。

注：

根据您使用的平台不同，HAC 二进制文件名为 solidhac 或 solidhac.exe。为了更清楚了，我们使用 solidhac.exe。同样，样本 IBM solidDB 启动脚本的名称为 start_solid.sh 或 start_solid.bat。在下面的内容中，我们将使用 start_solid.bat。

HA 控制器命令语法为：

```
solidhac.exe [-c working  
              directory | -?]
```

? 参数或者除了 **c** 之外的任何其他参数将打印用法消息。

启动 HAC

假定 solidhac.exe 位于 c:\solid\hac 目录下（这也是当前目录），并且使用 c:\solid\run\server1 作为 HAC 的工作目录。通过使用以下命令来启动 HA 控制器：

```
solidhac.exe -c c:\solid\run\server1
```

或者

```
solidhac.exe
-c ..\run\server1
```

HAC 启动之后，它就会开始侦听在 `solidhac.ini` 配置文件中指定的端口。此端口用于在 HAC 与 HA 管理器（Java™ GUI）之间传输命令。此外，HAC 接受通过使用 ADMIN COMMAND 发送至此端口的终止命令。

可以通过执行以下命令来停止（终止）HAC。要执行此命令，请使用 SQL 工具（例如，`solsql`）或者 ODBC 接口，并将此命令发送至 HAC 进程。使用 `solidhac.ini` 配置文件中的 **listen** 参数中所提到的端口。命令为：

```
ADMIN COMMAND      'hacontroller shutdown'
```

有关管理 HAC 的更多信息，请参阅第 22 页的『高可用性管理器』。

使用了 HA 控制器的 HSB 的快速启动过程

本节描述使用 HAC 的 HotStandby 的快速启动过程。

关于此任务

此过程与第 25 页的『HotStandby 快速启动过程』中的过程非常相似。但是，此过程不仅仅是设置两个 HSB 服务器，它还会指导您设置一个高可用 HSB 系统，其可用性由 HA 控制器来保证。通过遵循以下过程，您将准备好一个具有容错能力的 HSB 系统来为应用程序服务。

要设置和运行高可用 HSB 系统，需要两台互相联网的计算机。要设置 HotStandby 服务器和 HA 控制器（每个 HSB 服务器一个实例），请遵循以下过程。

1. 无论您是否使用了 HAC，请按相似的方式配置 HSB 服务器。有关更多信息，请参阅第 25 页的『HotStandby 快速启动过程』。
2. 配置两个 HAC。HAC 从位于工作目录下的 `solidhac.ini` 文件中读取其配置。下列列表中包含必需的配置参数。

在 [HAController] 节中：

- **Listen**=<listen address, 'tcp' and chosen port #>
- **Username**
- **Password**
- **DBUsername**
- **DBPassword**

在 [LocalDB] 节中：

- **Connect**=<connect address, protocol, ip/hostname, port #>
- **StartScript**（如果 **EnableDBProcessControl**=Yes（这是缺省值），那么此参数是必需的）

在 [RemoteDB] 节中：

- **Connect**=<connect address, protocol, ip/hostname, port #>

有关如何配置 HA 控制器的更多信息，请参阅第 47 页的『配置 HA 控制器和 HA 管理器』和第 50 页的『solidhac.ini 配置文件』。

3. 启动两个节点中的 HA 控制器。

HAC 将自动根据服务器先前的角色和日志位置来识别哪个服务器应成为新的主服务器。HAC 将使用第 91 页的『选择使哪个服务器成为主服务器』中所描述的机制。在某些特殊情况下（例如，当使用空数据库启动时），两个服务器都可以作为新的主服务器的候选者。在这些情况下，如果 `solidhac.ini` 的 `[LocalDB]` 节中的 **PreferredPrimary** 参数设置为“`Yes`”，那么 HAC 将选择它的本地服务器。

4. 在完成第二个步骤之后，应当有一个服务器处于 PRIMARY ACTIVE 状态，另一个服务器处于 SECONDARY ACTIVE 状态。必要时，可以通过按 HA 管理器中的**切换角色按钮**或者在辅助服务器上执行下列命令来切换角色：

```
ADMIN COMMAND 'hsb switch primary'
```

或者，在主服务器上执行以下命令：

```
ADMIN COMMAND 'hsb switch secondary'
```

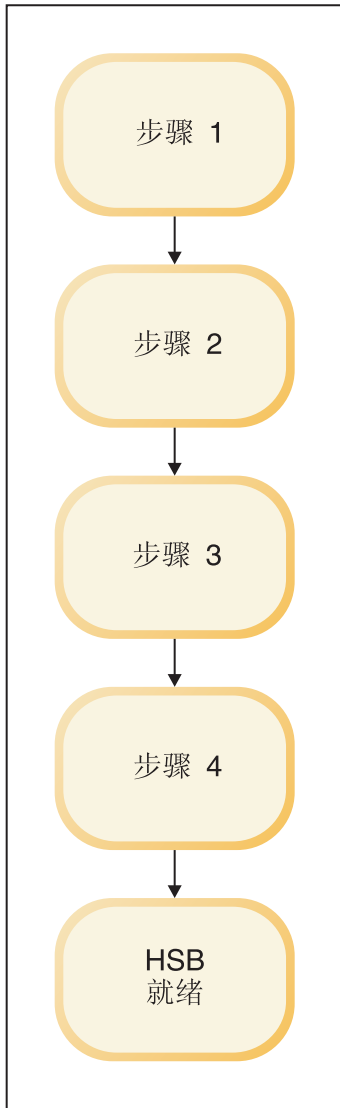
此后，HAC 就会在必要时切换服务器的角色。也就是说，它至少会保持让一个服务器执行读写事务操作。

5. 开始使用应用程序。

启动顺序总结

下图和下表以并排方式提供了带有 HAC 的安装和不带 HAC 的安装的顺序，以便进行比较。在下图中，带有 HAC 的安装顺序位于左边，不带 HAC 的安装顺序位于右边。图下方的表中说明了两种安装类型的图中的每个带有编号的步骤。

带有 HAC 的安装



不带 HAC 的安装

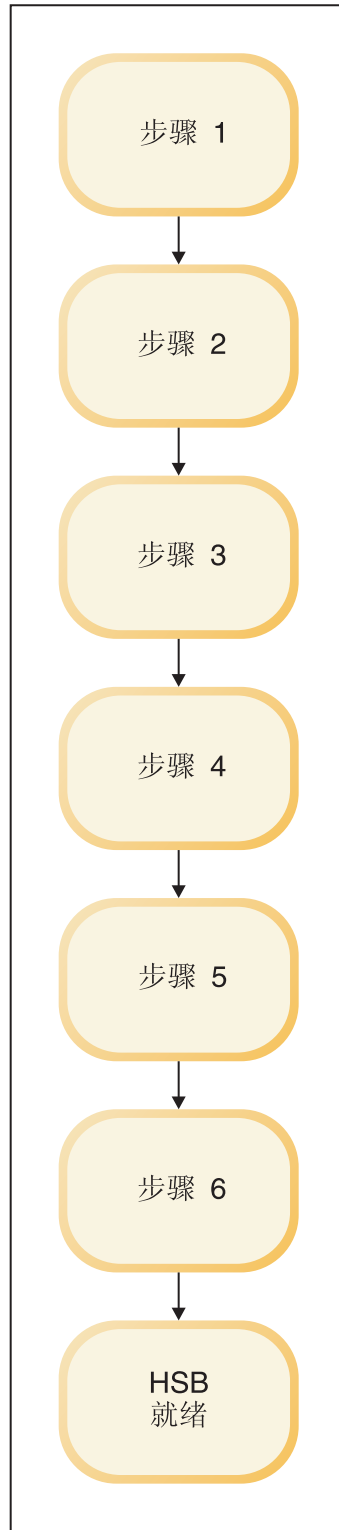


图 10. 启动顺序总结

表 5. 安装顺序步骤

带有 HAC 的安装	不带 HAC 的安装
步骤 1. 配置 HSB 服务器	
步骤 2. 配置 HA 控制器 步骤 3. 在两个节点中启动 HA 控制器 步骤 4. 必要时, 切换 HSB 角色	步骤 2. 在两个节点中启动 HSB 服务器 步骤 3. 将数据库服务器的状态切换到 PRIMARY ALONE 步骤 4. 将主服务器连接至辅助服务器 步骤 5. 如果连接失败, 那么请启动从主数据库复制到辅助数据库 步骤 6. 执行 netcopy 之后, 重新进行连接
HSB 已就绪。您可以开始使用应用程序了。	

3 管理和配置 HotStandby

本节描述如何维护 HotStandby 安装（包括 HSB 服务器、HA 控制器实例和 HA 管理器）。而且，本节还描述了用于实现和维护 HotStandby 功能的参数设置。

此描述是对《solidDB 管理员指南》中的『配置 solidDB』一节中信息的补充。

这些参数是按照 solid.ini 配置文件中的节类别进行分组的。当您使用 HotStandby 时，需要配置 solid.ini 配置文件的 [HotStandby] 节。

可以在 solidhac.ini 配置文件中找到高可用性控制器（HAC）配置。还可以按不同的节类别将 solidhac.ini 配置文件中的参数分组。

可以在 HAManager.ini 配置文件中找到高可用性管理器配置。HAManager.ini 中的参数主要用来标识 HA 控制器实例，以便高可用性管理器对它们具有访问权。

可以采用下列方法来更改配置参数：

- 手动编辑配置文件 solid.ini、solidhac.ini 和 HAManager.ini。请注意，服务器仅在启动期间才读取配置文件，因此，对任何配置文件所作的任何更改都将在下一次启动相应程序之后才会生效。
- 如果您要更改一个正在运行的 solidDB 服务器的设置，那么可以在 solidDB SQL 编辑器 solsql 中使用 ADMIN COMMAND 来进行更改。下面提供了 ADMIN COMMAND 语法：

```
ADMIN COMMAND 'parameter  
                section_name.param_name=value';
```

例如：

```
ADMIN COMMAND 'parameter hotstandby.2SafeAckPolicy=2'; ADMIN COMMAND  
                'parameter com.listen="tcp sf_server 1315";
```

注意：

当您使用 **ADMIN COMMAND** 来更改参数时，对某些参数（但不是所有参数）所作的更改将立即生效。

您应当知道的信息

本节描述在您开始管理和维护之前需要知道的有关 HotStandby 的信息。

HotStandby 和 solidDB 配置文件

要启用 HotStandby 功能，必须在 solidDB 配置文件（solid.ini）中提供一个特殊的 [HotStandby] 节，并且您的许可证文件必须允许使用 solidDB HotStandby 组件。

[HotStandby] 节中至少需要的配置信息是：

- 将 **HSBEnabled** 设置为“yes”。

- 设置服务器的 **Connect** 参数设置。此设置定义用于连接至其他服务器（主服务器或辅助服务器）的网络名。如果您在 `solid.ini` 文件中未设置此参数，那么服务器将不会作为 HotStandby 服务器启动。请注意，在服务器已启动之后，可以使用 ADMIN COMMAND 来设置或更改此参数。

请注意，每个服务器的 **Connect** 字符串必须与另一个服务器的 **Listen** 字符串相匹配。有关 **Connect** 参数和其他参数设置的详细信息，请阅读第 43 页的『配置特定于 HotStandby 的参数』。如果您是首次设置 HotStandby，请参阅第 25 页的 2 章，『HotStandby 入门』。

有关主服务器和辅助服务器上的 `solid.ini` 文件的示例，请参阅第 25 页的『HotStandby 演示程序』。

HotStandby 和访问权

管理员不需要特殊访问权就可以运行 HotStandby，具有正常的访问权就足够了。

要进行管理，需要具有 `SYS_ADMIN_ROLE` 或 `SYS_CONSOLE_ROLE` 才能执行 HotStandby 管理命令。可以使用以下 `solidDB SQL` 命令来执行这些命令：

```
ADMIN COMMAND 'hotstandby command_string';
```

例如，如果您使用 `solidDB SQL` 编辑器（`teletype` 工具），那么请输入：

```
ADMIN COMMAND 'hotstandby status connect';
```

当使用 `solidDB` 远程控制工具（`teletype` 工具）来输入 HotStandby 命令时，仅输入 `hotstandby` 命令字符串（不带引号），例如：

```
hotstandby status connect
```

solidDB 工具和 HotStandby

所有可用于对 `solidDB` 执行管理操作的工具也适用于 HotStandby。

可以分别在 `solidDB SQL` 编辑器（`solsql`）和 `solidDB` 远程控制（`solcon`）工具中发出特定于 HotStandby 的管理命令。此外，`solidDB` 快速装入器（`SOLLOAD`）、`solidDB` 导出工具（`SOLEXP`）和 `solidDB` 数据字典（`SOLDD`）可与 HotStandby 配合使用。有关这些工具的描述，请阅读《`solidDB` 管理员指南》中的『使用 `solidDB` 数据管理工具』。

可以使用高可用性管理器来监视 `solidDB HotStandby` 的状态以及控制 HSB 服务器和 HA 控制器。HA 管理器只能与 HA 控制器配合使用。有关高可用性管理器的描述，请参阅第 22 页的『高可用性管理器』。

数据库迁移（仅适用于基于磁盘的服务器）

可以使用下列其中一个命令行参数将 `solidDB 4.x` 数据库转换为最新的 `solidDB` 格式：

- **-xconvert**，用于将数据库文件转换为新结构并关闭服务器；或者
- **-xautoconvert**，用于转换数据库文件并继续运行。

将创建所有必需的系统表（包括用于 HotStandby 功能的那些系统表）。转换之后，已转换的数据库不再能够与更早的数据库版本配合使用。因此，在迁移到新的发行版之前，建议您备份数据库和文件。

注：

当 HotStandby 不再使用 IBM solidDB 数据库时，这些数据库仍然与 IBM solidDB 兼容。

互操作性

HSB 系统中的服务器应为与 HSB 兼容的版本。通常，相邻的版本就是与 HSB 兼容的版本。

有关与先前版本兼容的 HSB 的信息，请参阅《发行说明》。

HotStandby 存在的局限性和与 HotStandby 有关的警告

下列主题介绍了将内存表与 HotStandby 配合使用时存在的局限性。警告是为了避免出现双重主服务器的情况和用尽用于事务日志的空间。

内存表

如果您已连接至辅助服务器并且要从内存表中读取数据，那么会自动将事务隔离级别设置为“落实读”，即使您先前已指定了“可重复读”或“可序列化”也是如此（在主服务器或辅助服务器上，内存表根本不支持“可序列化”隔离级别）。

如果使用了负载均衡，那么隔离级别在缺省情况下为“落实读”。有关负载均衡和“透明连接”的更多信息，请参阅第 54 页的『使用透明连接』。

网络分区和双重主服务器

在某些情况下，可能会同时有两个服务器都在 PRIMARY ALONE 状态下运行。如果存在双重主服务器，可能会导致严重的无法恢复的错误。

在这种情况下，如果一个服务器落实了任何事务，而另一个服务器不落实这些事务，那么您将不能重新使服务器同步，这是因为没有方法来“合并”数据库以创建具有正确信息的单个数据库。实际上，在存在双重主服务器的情况下，在“错误主服务器”数据库中落实的事务将丢失。存在双重主服务器还可能会导致其他错误。

双重主服务器问题最有可能是由“网络分区”导致的 - 在这种情况下，某些网络连接已丢失（但不是所有网络连接都已丢失），并且您的单个网络实际上被分成多个单独的子网络，每个子网络允许在其内部进行通信，但是不与其他子网络进行通信。于是，两个服务器将互相失去连接，但是仍然在运行，并且在某些情况下可能仍然能够与某些客户机进行通信。

可以通过在位于 HSB 系统外部的节点中使用单个实例看守程序来避免出现双重主服务器情况。通过使用这样一个看守程序，就很容易决定应将哪个服务器设置为主服务器并确保客户机仅看到一个主服务器。

尽管 HAC 由两个与 HSB 服务器在相同节点中运行的实例组成，但是，当 HAC 与 ERE 配合使用时，实际上不可能出现双重主服务器情况。

即使您真的遇到了双重主服务器情况，您实际上也不会具有不一致数据，除非有人能够在原来的辅助服务器上执行写操作（在此服务器已切换到 PRIMARY ALONE 状态之后执行）。如果原来的辅助服务器与网络中的其他部分完全断开，那么没有人能够写入

此辅助服务器，并且原来的主服务器将成为辅助服务器的超集，而且（在重新连接服务器并且允许辅助服务器同步复制在原来的主服务器上所作的更改之后）您仍然能够获得单个一致数据集。

尽管很少出现双重主服务器的情况，但是如果真的出现这种情况则会非常危险，您必须格外小心地使用，以防止数据变得不一致。强烈建议您使用 ERE。

如果您在其中一个服务器或者两个服务器中的 `solid.ini` 文件中设置了配置参数 **AutoPrimaryAlone=Yes**，那么出现双重主服务器情况的机会就更大。使用 **AutoPrimaryAlone=Yes** 时，系统可能会更快对故障作出响应，但是它也意味着系统不再有任何独立的观察器（HAC、看守程序或人员）来防止发生双重主服务器情况。如果您对网络可靠性有任何顾虑，那么请让 **AutoPrimaryAlone** 参数保持使用出厂值（即，No）。

用尽用于事务日志的空间

使用 HotStandby 时，如果您使服务器处于 PRIMARY ALONE 状态，那么一定要小心不要让它用尽用于事务日志的磁盘空间。在一个非 HotStandby 服务器中，如果您频繁地执行检查点操作，那么事务日志不会变得非常大，这是因为每次执行检查点操作之后，服务器都会删除“旧的”事务日志。

尤其是，非 HotStandby 服务器将删除在执行检查点操作之前生成的日志和数据更改。有关执行检查点操作的更多信息，请参阅《solidDB 管理员指南》。

但是，在一个处于 PRIMARY ALONE 状态下运行的 HotStandby 服务器中，服务器必须保留自从主服务器与辅助服务器失去联系以来已累积的事务日志。如果辅助服务器已关闭很长时间，那么此服务器可能会保留许多事务日志数据；然而，如果服务器不处于这种状态，那么在每次执行检查点操作之后都会删除这些事务日志数据。在最糟糕的情况下，如果辅助服务器在合理时间内不能恢复正常，并且又没有足够的磁盘空间来存储发生的所有事务，那么主服务器的事务日志可能会填满所有可用磁盘空间。这将导致服务器切换为只读方式。

通过为 [HotStandby] 节中的 **MaxLogSize** 参数设置适当的值，就可以防止这种情况发生。达到所指定的日志总大小之后，服务器在下一个检查点将自动切换为 STANDALONE 状态。（在一个无盘服务器中，将继续保持 PRIMARY ALONE 状态，尽管根本没有磁盘写入操作。）

如果服务器设置为 STANDALONE 状态，那么它将不会保留自从主服务器与辅助服务器失去联系以来已累积的所有事务日志。当然，在没有完整事务日志的情况下，仅仅通过将主服务器连接至辅助服务器并允许辅助服务器通过读取旧日志来进行“同步复制”将无法使系统同步。您将需要使用 `copy` 或 `netcopy` 命令将整个数据库从主服务器复制到辅助服务器中。

如果使用了 HAC，那么它将识别上述情况，并通过自动执行必需的操作使服务器处于活动状态。

管理任务概述

本节描述您在使用 HotStandby 时可能需要执行的管理任务。

本节中包括下列主题:

表 6. 管理任务

主题	描述	页面
执行 HotStandby 恢复和维护任务	描述在发生系统故障时要执行的 HotStandby 任务（这些故障可能是由已中断的通信链路或者不可运行的 HotStandby 服务器导致的）。这些任务包括： <ul style="list-style-type: none">• 切换服务器状态• 停止 HotStandby 操作• 使主服务器与辅助服务器同步• 连接 HotStandby 服务器	『执行 HotStandby 恢复和维护』
通过网络将主数据库复制到新的辅助服务器	描述当 HotStandby 配置中新增了远程服务器（即，它是新的辅助服务器）或者远程服务器的数据被破坏并且必须替换时如何创建数据库的远程（网络）副本。	第 82 页的『通过网络将主数据库复制到辅助服务器』
检查 HotStandby 状态	描述如何检查主服务器和辅助服务器的 HotStandby 状态信息。	第 87 页的『检查 HotStandby 状态』
验证 HotStandby 服务器状态	描述如何检查 HotStandby 服务器的状态（是主服务器、辅助服务器还是独立服务器）。	第 89 页的『验证 HotStandby 服务器状态』
将 HotStandby 服务器更改为非 HotStandby 服务器	描述如何将为 HotStandby 配置的服务器设置为一个正常的非 HotStandby 服务器。	第 92 页的『将 HotStandby 服务器更改为非 HotStandby 服务器』

执行 HotStandby 恢复和维护

在发生系统故障（这些故障可能是由已中断的通信链路或者不可运行的 HotStandby 服务器导致的）或者维护服务器时，您可能需要执行一些 HotStandby 任务。这些任务包括切换服务器状态、停止 HotStandby 操作、使主服务器与辅助服务器同步以及连接 HotStandby 服务器。

1. 执行下面的某些或者所有操作：
 - a. 切换服务器状态。

这包括将主服务器设置为 PRIMARY ALONE 状态，它在这种情况下会继续将事务累积在事务日志中（以便稍后可以将它们发送至辅助服务器）或者关闭 HotStandby。

- b. 使服务器同步，以确保主数据库和辅助数据库完全相同。
 - c. 如果通信链路因某种原因而中断，那么请将主服务器连接至辅助服务器。
2. 可以按如下所示使用 HAC 和 HA 管理器来执行相同的步骤：
 - a. 按 HA 管理器中的切换按钮。如果需要关闭服务器，那么请按 HA 管理器中的关闭按钮。

- b. 通过按 HA 管理器中的**管理**按钮将 HAC 实例设置为“管理”方式。
- c. 通过按 HA 管理器中的**自动**按钮将 HAC 实例设置为“自动”方式。

下一步做什么

在下列各节中描述了这些主题。有关将应用程序重新连接至辅助数据库或主数据库的详细信息，请阅读第 65 页的『从应用程序重新连接至主服务器』。

要点:

如果使用了 HAC，那么使用 HA 管理器来执行管理步骤或者将 HAC 实例设置为“管理”方式，然后才启动管理。

特殊配置：更低的成本与更高的安全性

solidDB HotStandby 解决方案使用成对的主服务器和辅助服务器来提供真正的热备用功能。然而，使用成对的服务器并不是对于每个客户都是最佳的。如果不需要几乎即时进行故障转移，那么可能就没有必要为每个主服务器配备一个辅助服务器。另一种极端情况是，某些客户可能对可靠性要求特别高，可能要为备用服务器购买备用服务器。也就是说，不仅要为每个主服务器购买辅助服务器，而且还要额外购买一个或多个备用服务器，以便在主服务器失败而由其辅助服务器接替它时，如果此时无法快速修复原来的主服务器，那么可以将一个备用服务器用作新的辅助服务器。

为了使客户能够降低成本或者提高可靠性，solidDB HotStandby (HSB) 支持使用标准热备用模型的一些替代模型。有时候将标准模型称为“N+N”或者“2N”模型，这是因为主服务器与辅助服务器的数目相同（都为“N”）。替代模型包括：

- N + 1 个备用服务器或者 N + M 个备用服务器：这是用于独立服务器的“备用节点”方案。有 N 个“主”服务器以及一个或多个备用服务器，但是没有辅助服务器。失败的“主”服务器将被一个备用服务器替换。这不是一个真正的“热备用”方案，最好是将它称为“暖备用”，这是因为虽然提供了计算机，但是它没有数据库副本。
- 2N + 1 个备用服务器或者 2N + M 个备用服务器：这是用于 HotStandby 的“备用节点”方案。有 N 个 HotStandby 对，即，每个主服务器都有一个辅助服务器。此外，有 M 个备用服务器，其中 M 至少应为 1，但是通常小于 N。当一个主服务器或辅助服务器失败时，就会将一个备用服务器作为新的辅助服务器。因此，主服务器决不会长时间独立运行，即使它原来的伙伴服务器失败也是如此。

后续主题说明了 N+M（或者 N+1）和 2N+M（或者 2N+1）方法，以及可帮助您实现这些方法的 solidDB 功能部件。

降低成本：N + 1 个备用服务器和 N + M 个备用服务器方案

在这些方案中，有 N 个“主”服务器，每个主服务器都以 STANDALONE 状态运行，即，它们不连接至辅助服务器。此外，有 M 个备用服务器，其中 M 至少应为 1，但是通常小于 N。如果“主”服务器失败，其中一个备用服务器就会替换它。会将数据从原来的服务器复制到备用服务器，然后原来的服务器将脱机，并将备用服务器配置为替代原来的服务器。请注意，任何备用服务器可以替换任何主服务器，备用服务器并不专用于特定的主服务器。还要注意，不会即时进行故障转移。

我们将此方法称为“N + 1”（单个备用服务器）或者“N + M”（多个备用服务器）方案。

由于此方法要求您具有原来的服务器数据的副本，因此，如果原来的服务器的磁盘驱动器被损坏，从而没有数据备份，那么此方法将无效。此 N+M 方法在下列情况下最有用：

1. 您要使用备用节点来处理预定维护，而不是因意外故障而进行维护。
2. 您具有可靠的备份，可以将它快速复制到备用服务器中。
 - a. 您已将备份保存在磁带、RAID 驱动器或者其他某些安全位置；或者
 - b. 您要使用 solidDB 的高级复制技术，并且您可以通过从已失败的高级复制“主”或“副本”服务器中读取数据来复制或重新创建足够多的数据。
3. 单个数据块不是非常关键或者不是唯一的。
 - a. 例如，如果您真正需要的是“计算马力”（负载分布能力）而不是特定数据，那么通过复制标准数据库或“种子”数据库，或者通过从客户机中获取数据然后继续运行，就可以满足您的需要。
 - b. 同样，如果所有服务器几乎具有相同的数据，并且几乎完全是响应“读”请求，只有少量甚至没有“写”请求（例如，如果您正在运行许多服务器，它们都使用相同的因特网路由表或者电话簿信息），那么您就可以从任何一个计算机中复制有用的数据库。

提高可靠性：2N + 1 备用服务器和 2N + M 备用服务器方案

正常的 solidDBHotStandby 操作是非常可靠的。如果主服务器和辅助服务器都使用单独的可靠电源，那么它们几乎同时失败的可能性非常低。但是，如果您甚至希望将同时失败这种风险也降低，或者无法快速修复失败的服务器，那该怎么办呢？实际上，当主服务器失败并且您使用辅助服务器替换它时（或者当辅助服务器失败时），您肯定愿意使用一个新的辅助服务器来替换旧的辅助服务器，以便您可以使用一个完整的服务器对来继续运行。

这种情况称为 2N + 1 个备用服务器（或者 2N + M 个备用服务器）方案。您具有 N 个主服务器，N 个辅助服务器，并且至少还有一个备用服务器可用来替换任何已失败或者已转换为主服务器的辅助服务器。备用服务器并不专用于特定服务器（或者 HSB 服务器对），并且备用服务器需要具备某些配置之后才能替换失败的服务器。

solidDB HSB 如何支持 N+1 (N+M) 和 2N+1 (2N+M) 方法

必须使备用服务器看起来与它要替换的服务器一样。通常，这意味着：

1. 您必须将数据复制到备用服务器中。
2. 您必须让备用服务器与它要替换的服务器在同一网络地址或者在客户机程序知道用来通信的另一个地址进行“侦听”。
3. 此外，在 2N+1 (2N+M) 方案中，您还必须让新的辅助服务器和当前的主服务器知道如何互相进行通信。换句话说，您必须让每个服务器知道用来连接至对方的地址。

solidDB 有以下两项功能用来满足这些需要：

- solidDB 允许您在不关闭备用服务器的情况下将数据复制到备用服务器。
- solidDB 允许您动态设置某些配置参数。

下面更详细地说明了这些功能：

1. 尽管通常通过关闭服务器，更新 `solid.ini` 配置文件，然后重新启动服务器这种方法来设置 `solidDB` 配置参数，但是也可以通过执行如下所示的 `ADMIN COMMAND` 来更改某些配置参数（例如，“`com.listen`”和“`hotstandby.connect`”参数）：

```
ADMIN COMMAND 'parameter com.listen="tcp SpareServer1 1315"'; ADMIN COMMAND 'hsb
parameter connect "tcp srvr27 1316"';
```

这意味着可以在不关闭备用服务器的情况下动态配置备用服务器以替换另一个服务器。同样，主服务器可以知道其新的辅助服务器的连接字符串。

注意：

执行这些命令之后，不会将已更新的参数值写入 `solid.ini` 文件。因此，为了确保服务器在下次重新启动之后具有新值，您还应该更新 `solid.ini` 文件并且执行上述命令。

要点：

应使用 `-x backupserver` 命令行选项来启动备用服务器，以使它准备好接收来自主服务器的 `netcopy`。有关 `-x backupserver` 选项的更多信息，请参阅第 83 页的『为辅助服务器创建新的数据库』，同时还请参阅《`solidDB` 管理员指南》中对各个命令行选项的说明。

2. `solidDB` 的“`netcopy`”命令允许您将数据库复制到一个已启动并且正在运行的服务器中。

- a. 设置“`connect`”参数的新值：

```
ADMIN COMMAND 'hsb parameter
connect "tcp srvr27 1316"';
```

- b. 执行 `netcopy` 命令：

```
ADMIN COMMAND 'hsb
netcopy';
```

- c. 通过执行以下命令将当前主服务器与新的辅助服务器连接起来：

```
ADMIN COMMAND 'hsb connect';
```

将 HAC 与备用服务器配合使用

HAC 对于备用服务器方案提供了有限支持。可以将它用于备用服务器中，但是在主节点中的 HAC 能够开始监视备用服务器的 HSB 状态之前，必须更新主 HAC 的连接信息 **RemoteDB.Connect**。这要求更新 `solidhac.ini` 文件并重新启动主节点中的 HAC。

同样，备用节点中的 HAC 需要主服务器的连接信息。如果事先不知道主服务器的连接信息，那么必须将此信息添加到 `solidhac.ini` 文件中。将此信息添加到 `solidhac.ini` 文件之后，必须重新启动 HAC。

为 HotStandby 配置 `solidDB`

主节点和辅助节点中的 `solid.ini` 文件中包含在设置 HotStandby 系统时必需的参数。

定义辅助节点和主节点配置（[Com] 节）

主服务器或辅助服务器的网络名由一种通信协议和一个服务器名称组成。网络名是使用配置文件的 [Com] 节中的 **Listen** 参数定义的。`solid.ini` 文件可能位于 `solidDB` 程序的工作目录中，或者位于由 **SOLIDDIR** 环境变量设置的目录中。

一个主服务器或辅助服务器可以使用一个或多个网络名。请注意，网络名的所有部分都不区分大小写。

solid.ini 文件中的条目

主节点:

```
[Com]
;The Primary server listens to the network with this name
Listen = tcp 1320
```

辅助节点:

```
[Com]
;The Secondary server listens to the network with this name
Listen = tcp 1321
```

有关 **Listen** 参数的更多信息，请参阅《solidDB 管理员指南》。

定义应用程序与服务器的超时 ([Com] 节)

本节描述如何通过使用用于 ODBC 的 solid.ini **Connect** 参数或者 SQLConnect 函数的连接字符串来配置“应用程序读取超时”和“连接超时”设置。

这些超时值将应用于服务器与客户机应用程序（包括 solidDB SQL 编辑器（solsql）、solidDB 远程控制 teletype 工具（solcon）和 HA 管理器）之间的连接。

应用程序读取超时选项

此选项用于检测低级别网络 RPC 读取操作中发生的故障。其超时设置应用于物理网络中的读取操作，并且仅适用于 TCP/IP 协议。

此 RPC 读取超时（在 ODBC 和 JDBC 中称为连接超时）值是使用以下选项在连接字符串和侦听字符串中配置的：

```
TCP
  -rnumber_of_milliseconds
  [machine_name]
  port_number
```

要在 solid.ini 文件中指定此值，请在 solid.ini 的 [Com] 节中使用 **Connect** 参数来指定。例如：

```
[Com] ;Set RPC read timeout to 1000
      milliseconds (one second) Connect=TCP -r1000 1313
```

RPC 读取超时的缺省值为 60000 毫秒（即，60 秒）。如果值为 0，那么将设置无限期超时。

如果您要使用 ODBC，那么可以在 SQLConnect 函数的连接字符串中指定超时设置。例如：

```
SQLConnect (hdbc, "TCP -r1000
1313", SQL_NTS, "dba", SQL_NTS, "dba", SQL_NTS);
```

在上述示例中，常量 SQL_NTS 指示先前的字符串（服务器名称、用户名或密码）是作为以 null 结束的标准字符串传递的。

注：

对于客户机应用程序（例如，看守程序），很方便在 `connect` 参数中使用 `-r` 选项提供 RPC 读取超时（也称为连接超时）。否则，某些网络故障类型可能会导致无限期等待。

注:

[Com] 节、[Watchdog] 节和 [Hotstandby] 节中的 `Connect` 参数具有不同的用途。您务必要编辑正确的那个 `Connect` 参数。

在 `Connect` 参数中指定 `-C` 选项

可以在 `solid.ini` 文件的 [Com] 节和 [Watchdog] 节中使用的 `Connect` 参数中指定连接超时（也称为登录超时）值。此连接超时仅适用于 TCP/IP 协议。

语法为:

```
Parameter = tcp -cnumber-of-milliseconds  
            [machine name]  
            port_number
```

其中 `Parameter` 为 `Connect` 或 `Listen`。

如果没有为连接超时提供任何值，那么服务器将使用特定于操作系统的缺省值。

注:

对于客户机应用程序（例如，看守程序），很方便在 `Connect` 参数中使用 `-c` 选项提供连接超时值。否则，某些网络故障类型可能会导致等待很长时间之后才检测到故障。

例如:

应用程序节点:

```
[Com] ;The server listens to port 1320, and the Connection timeout  
      is 1000 ms. Listen = tcpip -c1000 1320
```

事务耐久性

DurabilityLevel

`DurabilityLevel` 参数同时适用于 HotStandby 服务器和非 HotStandby 服务器。对于“宽松”、“自适应”和“严格”耐久性，此参数具有不同的值。

“自适应”耐久性仅用于 HotStandby 服务器。“自适应”耐久性意味着:

- 如果主服务器和辅助服务器已经建立了连接并且都在正常运行（即，它们分别处于 PRIMARY ACTIVE 和 SECONDARY ACTIVE 状态），那么服务器将使用“宽松”耐久性。
- 在所有其他情况下（例如，PRIMARY ALONE 和 STANDALONE），服务器将使用“严格”耐久性。

有关“严格”耐久性与“宽松”耐久性之间的差别，或者有关 `DurabilityLevel` 参数的更多信息，请参阅《solidDB 管理员指南》。

配置特定于 HotStandby 的参数

在主服务器和辅助服务器节点中，solid.ini 文件包含 [HotStandby] 节以指定特定于 HotStandby 的配置参数。

定义主服务器和辅助服务器 HotStandby 配置

为了启用 HotStandby，您必须至少设置的一组 solid.ini 配置参数是：

- **HSBEnabled**。此参数用于打开或关闭 HSB。
- **Connect**。此参数定义用来定义主服务器或辅助服务器的网络名。网络名是主服务器用来连接至辅助服务器（或者辅助服务器用来连接至主服务器）的协议和服务器名称。（严格来说，**Connect** 参数并不是必须包含在 solid.ini 文件中。您可以在不使用此参数的情况下启动服务器，然后使用一个 ADMIN COMMAND 来指定连接字符串。如果未设置连接字符串，那么服务器只能在不需要连接的状态下运行（例如，PRIMARY ALONE、SECONDARY ALONE 和 STANDALONE 状态）。
- **LogEnabled**。如果设置了此参数，那么应将它设置为“yes”。请注意，此参数位于 solid.ini 文件的 [Logging] 节中，而不是位于 [HotStandby] 节中。

注：

如果 solid.ini 文件中不包含 [HotStandby] 节，或者 [HotStandby] 节中不包含 HSBEnabled=Yes，那么服务器将作为一个非 HotStandby 服务器启动并且不使用 HotStandby 复制。

部分 solid.ini 文件

节点 1

```
[HotStandby] HSBEnabled = yes ;The server connects to the opposite server
;using the following connect string.
Connect = tcp machine2 1321 [Logging] LogEnabled=yes
```

节点 2

```
[HotStandby] HSBEnabled = yes ;The server connects to the opposite server
;using the following connect string.
Connect = tcp machine1 1320 [Logging] LogEnabled=yes
```

设置 HotStandby 服务器等待时间以帮助检测已断开的连接或者不可用的连接

HotStandby 服务器使用超时参数来控制它在断定现有连接已断开或者无法建立新连接之前将等待的时间。

关于此任务

超时参数包括：

- **HotStandby.PingTimeout**
- **HotStandby.PingInterval**
- **HotStandby.ConnectTimeout**

当一个处于 PRIMARY ACTIVE 状态或 SECONDARY ACTIVE 状态的 HotStandby 服务器尝试连接至另一个服务器时，如果它在指定时间内未接收到应答，那么其状态将更改为 PRIMARY UNCERTAIN、PRIMARY ALONE 或 SECONDARY ALONE。

要控制服务器的等待时间，可以执行下列操作：

- 设置 **PingTimeout** 参数，以指定服务器在更改为 PRIMARY UNCERTAIN 状态之前将等待的时间。
- 设置 **PingInterval** 参数，以指定服务器两次发送“ping”消息（用于指示此服务器在正常运行）之间的时间间隔。
- 设置 **ConnectTimeout** 参数，以指定服务器在尝试与另一个服务器建立新连接（例如，通过 ADMIN COMMAND 'hotstandby connect' 操作来建立连接）时应等待的时间。

PingTimeout 和 PingInterval 参数 [HotStandby]

[HotStandby] 节中的可选参数 **PingTimeout** 和 **PingInterval** 用于控制 ping 操作。

“ping”操作实质上是由一个数据库服务器发送至另一个数据库服务器的一条表示“我还活着”的消息。（某些联网软件也具有“ping”操作，但是 [HotStandby] 节中的 solidDB**PingTimeout** 配置参数仅适用于 solidDB 服务器 ping 操作，而不适用于常规的网络 ping 操作。）换句话说，这指的是被动脉动信号系统。当设置了此参数时，主 HotStandby 服务器和辅助 HotStandby 服务器会定期互相发送“ping”消息。另请参阅第 3 页的『脉动信号』。

- **PingTimeout** 参数指定一个服务器在断定另一个服务器已关闭或者不可访问之前应等待的时间。缺省值为 4000（即，4 秒）。
- **PingInterval** 参数指定两次 ping 操作之间的时间间隔（按毫秒计）。缺省值为 1000（即，1 秒）。

例如，如果 **PingInterval** 为 10 秒，那么服务器之间每隔 10 秒钟就会互相发送 ping 消息。如果 **PingTimeout** 参数为 20 秒，而一个服务器（S1）在 20 秒之内未接收到来自另一个服务器（S2）的 ping 消息，那么 S1 就会断定 S2 已关闭或者不可访问。于是，服务器 S1 将切换到另一种状态，即，从“PRIMARY ACTIVE”切换到“PRIMARY UNCERTAIN”状态。

如果上述参数的值不相同，那么将优先采用在执行“hsb connect”命令期间在主服务器中设置的值。这些值在切换期间并不会更改。但是，可以使用 ADMIN COMMAND “parameter” 命令来动态更改这些值。

如果 **PingTimeout** 设置为 0，那么将禁用 ping 操作。

Ping 操作需要很少的开销，并且 solidDB 服务器设置为快速对丢失 ping 消息作出响应。可以将 **PingInterval** 值设置为一个相当短的时间间隔（例如，1 秒甚至更短）。

如果您需要在服务器失败时快速检测故障转移，那么请将 **PingTimeout** 值设置为一个相对短的时间。但是，此值越小，产生“误报警”的机会就越多。如果您的网络流量很大，从而导致在接收到 ping 消息之前产生延迟，那么您可能需要将 **PingTimeout** 参数设置为一个较大值以避免产生误报警。

ConnectTimeout 参数 [HotStandby]

在某些网络实现方案中，连接操作可能无限期地不作出响应。一种原因可能是，虽然远程机器是一个已知节点，但是在尝试连接期间不可用。通过指定连接超时值，可以设置 HotStandby 连接操作等待与远程机器建立连接的最长时间（按秒计）。

ConnectTimeout 参数仅与某些管理命令配合使用（此参数仅在某些平台上有用）。这些管理命令是：

- hotstandby connect
- hotstandby switch primary
- hotstandby switch secondary

使用 solid.ini 文件的 [HotStandby] 节中的 **ConnectTimeout** 参数来设置连接超时值，时间单位是毫秒。缺省值为 0，它意味着不存在超时。可以将它设置为其他值，例如：

```
[HotStandby]
; Set ConnectTimeout to 20 seconds (20000 milliseconds).
ConnectTimeout=20000
```

定义 HotStandby 数据库复制操作的名称和位置

[HotStandby] 节中的可选参数 **CopyDirectory** 定义 HotStandby 复制操作要复制到的目录的名称和位置。

HotStandby 复制操作是通过以下命令指定的：

```
ADMIN COMMAND 'hotstandby copy [directory_name]';
```

此参数没有缺省值，因此，如果在 solid.ini 文件中未指定目录，那么必须在 copy 命令中提供目录。如果您为 **CopyDirectory** 参数提供一个相对路径，那么此路径是相对于用来保存主服务器的 solid.ini 文件的目录的路径。

如果您使用 ADMIN COMMAND 'hotstandby netcopy' 命令来执行 HotStandby 数据库复制操作，那么不需要此参数。在这两个命令中，netcopy 提供了更灵活的功能，因此建议使用此命令。

定义主服务器在辅助服务器发生故障期间的行为

可以在 [HotStandby] 节中使用 **AutoPrimaryAlone** 参数来控制以下情况：主服务器在与辅助服务器失去联系之后是自动切换到 PRIMARY ALONE 状态还是继续保持 PRIMARY UNCERTAIN 状态。

如果 **AutoPrimaryAlone** 设置为 Yes，那么当主服务器与辅助服务器失去联系时，主服务器将自动切换到 PRIMARY ALONE 状态，从而允许它继续接受事务。如果 **AutoPrimaryAlone** 设置为 No，那么当主服务器与辅助服务器失去联系时，主服务器将自动切换到 PRIMARY UNCERTAIN 状态。

缺省情况下，**AutoPrimaryAlone** 设置为 No。

```
[HotStandby]
AutoPrimaryAlone = No
```

在 PRIMARY UNCERTAIN 状态下，主服务器将无法接受新的事务或者落实当前处于活动状态的事务。在 HAC、看守程序或者系统管理员告诉主服务器切换到 PRIMARY ALONE 之前，它不会切换到此状态。

如果 **AutoPrimaryAlone** 设置为 No，那么服务器可以通过执行 ADMIN COMMAND 'hotstandby set primary alone' 命令来设置为 PRIMARY ALONE 状态。请注意，此命令不会更改配置文件中的 **AutoPrimaryAlone** 值。

如果将缺省值更改为 Yes，那么主服务器的状态将从 PRIMARY ACTIVE 更改为 PRIMARY ALONE 而不是更改为 PRIMARY UNCERTAIN。

性能调整

使用安全级别和耐久性级别来调整复制性能

在执行正常操作期间的数据复制性能取决于耐久性级别和安全级别的设置。此外，当使用 2-safe 复制时，2-Safe 方式中使用的确认策略将影响等待时间，应用程序将识别这一点。有关更多信息，请参阅第 14 页的『HotStandby 如何影响性能』。

调整 netcopy 性能 ([General] 节)

HotStandby netcopy 命令允许将主服务器的数据库复制到远程辅助服务器。当其中一个或者两个都是无盘服务器时，还可以使用此命令将数据库从主服务器复制到辅助服务器。用来连接至辅助服务器以进行网络复制的连接字符串是在 solid.ini 文件的 [HotStandby] 节中指定的。

通过网络链路来复制主数据库文件。有关 netcopy 命令的更多详细信息，请阅读第 82 页的『通过网络将主数据库复制到辅助服务器』。

solid.ini 文件的 [General] 节中的下列参数可提高网络复制性能。

BackupBlockSize 参数 [General]

通过网络复制将主数据库文件复制到辅助服务器时，可使用 solid.ini 文件的 [General] 节中的 **BackupBlockSize** 参数通过增大或减小网络复制块大小来提高性能。一般来说，块大小越大，网络复制/备份的速度就更快；但是，在执行网络复制/备份时，服务器对其他请求作出响应的时间会变长。

缺省情况下，**BackupBlockSize** 参数设置为 64K。可以将它设置为其他值，例如：

```
[General]
BackupBlockSize = 32K
```

或者

```
[General]
BackupBlockSize = 32768
```

请注意，**BackupBlockSize** 的最小值是服务器块大小，最大值目前为 1GB（也可以采用“M”和“K”为单位；例如 32K 和 1M）。**BackupBlockSize** 的值应为服务器的数据库块大小的倍数。

调整数据库同步复制性能

在失败的辅助服务器恢复运行并且已连接至主服务器之后，HotStandby 将通过一个称为 *HotStandby 数据库同步复制* 的自动化进程继续将主服务器的 HotStandby 事务日志文件内容发送至辅助节点。

可使用 solid.ini 文件的 [HotStandby] 节中的 **CatchupSpeedRate** 参数通过调整服务器用于同步复制与用于处理当前客户机数据库查询的时间来调整数据库同步复制的性能。

如果将 **CatchupSpeedRate** 的值指定为 90，这就意味着服务器大约要花 90% 的时间来进行同步复制，大约要花 10% 的时间来执行用户查询。例如：

```
[HotStandby]
CatchupSpeedRate = 50
```

此值越大，执行同步复制的速度就会更快，但是也会更多地影响其他活动（例如，用户查询）。缺省情况下，**CatchupSpeedRate** 设置为 70。

配置 HA 控制器和 HA 管理器

高可用性控制器（HAC）部署在每个 HotStandby 服务器节点上。HAC 配置文件 `solidhac.ini` 必须位于 HAC 工作目录下。

`solidhac.ini` 配置文件中的参数归入下列各节：

- [HAController]
- [LocalDB]
- [RemoteDB]
- [ERE]

所有配置参数都显示在第 50 页的『`solidhac.ini` 配置文件』中的 `solidhac.ini` 示例文件中。

[HAController] 节

下面是对 [HAController] 节中各个参数的说明：

- **Listen**

Listen 参数的值指定 HAC 用于客户机通信的协议和端口。在启动期间，HAC 就开始侦听此端口。如果无法开始侦听（例如，如果此端口已被另一个进程使用），那么 HAC 将会终止然后将信息写入日志文件（`hacmsg.out`）。唯一受支持的协议是 TCP/IP（“tcp”）。HA 管理器使用 HAC 侦听端口与 HAC 进行通信。此外，还会将 HAC 关闭命令发送至此端口。

Listen 是一个必需参数。

- **StartInAutomaticMode**

一旦 HAC 在运行，它就可能处于下列方式之一：“自动”或“管理”。在“自动”方式下，HAC 将自动尝试通过更改服务器的 HSB 状态并在必要时重新启动服务器进程来使可用性达到最大值。相比之下，在“管理”方式下，HAC 将仅监视服务器的运行状况。

StartInAutomaticMode 指定 HAC 的初始方式。

StartInAutomaticMode 的缺省值为 “Yes”。

- **EnableDBProcessControl**

设置 **EnableDBProcessControl=Yes** 时，允许 HAC 通过自动启动服务器以及通过为用户提供用于关闭并重新启动数据库进程的命令来管理本地服务器进程。设置 **EnableDBProcessControl=Yes** 时，**StartScript** 参数是必需的。

缺省值为 “No”。

- **EnableAutoNetcopy**

设置 **EnableAutoNetcopy**=Yes 时，允许 HAC 在使用 hsb connect 命令无法建立 HSB 链路时启动 netcopy。

EnableAutoNetcopy 的缺省值为 “Yes”。

- **RequiredConnectFailures**

当服务器状态未知，或者因其他某些原因而使得 HAC 需要确保服务器状态正确时，请使用非分块 SQLConnect (check) 命令。如果在这种情况下执行非分块 SQLConnects 失败，那么请多次重复执行此操作，直到认为所讨论的服务器无法作出响应为止。连续尝试连接的最大次数等于 **RequiredConnectFailures** 的值。

RequiredConnectFailures 的缺省值为 2。

- **CheckTimeout**

连续非分块 SQLConnect 命令的超时（按毫秒计）。超时值太小可能会导致“假正”问题。换句话说，尽管服务器在运行，但是它未能在超时时间段内作出响应，看起来好像已经失败了一样。

CheckTimeout 的缺省值为 150（毫秒）。

- **CheckInterval**

两次连续执行非分块 SQLConnect 命令的时间间隔。此值不会影响故障转移时间。通常在发生故障之后或者在系统启动期间执行检查（轮询）。

CheckInterval 的缺省值为 1000（毫秒）。

- **Username**

HAC 用户名。

Username 是一个必需参数。

- **Password**

使用 **Username** 标识的用户的密码。

Password 是一个必需参数。

- **DBUsername**

HAC 要用来连接至本地 HSB 服务器的用户名。数据库用户应当具有 SYS_ADMIN_ROLE 或 SYS_CONSOLE_ROLE。

DBUsername 是一个必需参数。

- **DBPassword**

通过使用 **DBUsername** 标识的用户的密码。

DBPassword 是一个必需参数。

[LocalDB] 节

下面是对 [LocalDB] 节中各个参数的说明：

- **Connect**

本地数据库服务器的连接信息。HAC 在连接至本地服务器时使用此信息。

Connect 是一个必需参数。

- **StartScript**

如果配置了 HAC 以控制数据库进程 (**EnableDBProcessControl=Yes**)，那么 **StartScript** 是一个必需参数。如果 **EnableDBProcessControl=No**，那么此值无效。

- **PreferredPrimary**

如果任一服务器都可以作为主服务器启动，那么在设置了 **PreferredPrimary=Yes** 的情况下，会将本地 HSB 服务器作为主服务器。如果两个服务器在 **PreferredPrimary** 中具有相同值，那么第一个服务器将成为新的主服务器。

PreferredPrimary 的缺省值为 “No”。

[RemoteDB] 节

下面是对 [RemoteDB] 节中各个参数的说明：

- **Connect**

远程数据库服务器的连接信息。HAC 在连接至远程节点中的服务器时使用此信息。

Connect 是一个必需参数。

[ERE] 节

下面是对 [ERE] 节中各个参数的说明：

- **EREIP**

ERE 的 IP 地址。

- **RequiredPingFailures**

在 HAC 断定服务器已与 ERE 断开连接之前，必须连续失败的 ping 调用次数。

RequiredPingFailures 的缺省值为 3。

配置文件示例

下列各节提供了与 solidDB 相关的不同配置文件的示例。

solid.ini 配置文件

下面是第一个 HotStandby 服务器的 solidDB 配置文件 (solid.ini) 的样本摘录：

```
[Com]
; The first server listens to the network with this
; name
Listen = tcp 1320
[HotStandby]
HSBEnabled=yes
; The first server connects to the second server
; using the following connect string.
```

```

Connect = tcp 188.177.166.12 1321
AutoPrimaryAlone=No
[Logging]
LogEnabled=yes

```

下面是第二个 HotStandby 服务器的 solidDB 配置文件 (solid.ini) 的样本摘录:

```

[Com]
; The second server listens to the network using the following
; connect string.
Listen = tcp 1321
[HotStandby]
HSBEnabled=yes
; The second server connects to the first server
; using the following connect string.
Connect = tcp 188.177.166.11 1320
AutoPrimaryAlone=No
[Logging]
LogEnabled=yes

```

solidhac.ini 配置文件

高可用性控制器配置文件 (solidhac.ini) 的样本摘录。

```

;=====
; NOTE : Copy this file as solidhac.ini
;        to solidhac working directory
;
; solidDB High Availability Controller inifile
;=====

[HAController]
;** HAC connect info
;** HAC clients, HA Manager, for example, use this information.
;** Mandatory
;** Listen=tcp 3135
Listen=

;** Setting StartInAutomaticMode=Yes starts HAC in AUTOMATIC mode.
;** In AUTOMATIC mode, solidhac automatically tries
;** to maximize the availability by changing the HSB states of the
;** server, and restarting the server processes when necessary.
;** In contrast, it can be in ADMINISTRATIVE mode
;** in which HAC only monitors the health of the servers.
;**
;** This is dynamically changeable parameter.
;** Optional
;** Values : Yes/No, default = Yes
StartInAutomaticMode=

;** Setting EnableDBProcessControl=Yes allows solidhac
;** manage local db process by automatically starting
;** the db, and by providing the user with commands to
;** shutdown and restart db process.
;**
;** Optional
;** Effective only when HAC is in AUTOMATIC mode.
;** Values : Yes/No, default = No
EnableDBProcessControl=

;** Setting EnableAutoNetcopy=Yes allows solidhac to initiate
;** netcopy when HSB link cannot be established with 'hsb connect'.
;**
;** Optional
;** Effective only when HAC is in AUTOMATIC mode.
;** Values : Yes/No, default = Yes
EnableAutoNetcopy=

```



```

;** When server state is unknown, or HAC needs, for some other reason, to
;** ensure the state of server, non-blocking SQLConnect command is used.
;** If the execution of non-blocking SQLConnect in such a case fails,
;** it is repeated multiple (RequiredConnectFailures) times before
;** the server in question is considered as non-responsive.
;**
;** Optional
;** Values : 1..n, default=2
RequiredConnectFailures=

;** Timeout in milliseconds for non-blocking SQLConnect commands.
;** Too short interval can cause 'false positives', server seems
;** to be failed because it wasn't able to respond within the timeout period.
;**
;** Optional
;** Values : 1..n, default=150 (milliseconds)
CheckTimeout=

;** Interval between consecutive non-blocking SQLConnect commands.
;** The value doesn't affect on failover time. Checking (polling)
;** takes place typically after failure, or during system startup.
;**
;** Optional
;** default = 1000 (milliseconds)
CheckInterval=

;** HAC user identification
;** Mandatory
Username=
Password=

;** HSB server user identification
;** Mandatory
DBUsername=
DBPassword=

[LocalDB]
;** solidb connect info
;** Mandatory
;** Connect=tcp 2125
Connect=

;** The name of the script, which is used to initiate the db process.
;**
;** Optional, except if HAC controls db process (EnableDBProcessControl=Yes).
;** Value is not effective if EnableHACActions=No or EnableDBProcessControl=No
;** StartScript=/home/solid/start_solid.sh
StartScript=

;** Setting PreferredPrimary=Yes moves local HSB server to as Primary
;** in the case where either of the servers could start as Primary.
;** If both servers have PreferredPrimary=No, or no value, first
;** (new Primary) server wins.
;**
;** Optional
;** Value is not effective if EnableHACActions=No.
;** Values : Yes/No, default no
PreferredPrimary=

[RemoteDB]
;** solidb connect info
;** Mandatory
;** Connect=tcp 192.168.3.123 2125
Connect=

```

```

[ERE]
;** IP address of an ERE
;** Optional
;** Connect=192.168.3.1
EREIP=

;** The number of consecutive ping calls that must
;** fail before HAC concludes that the server is
;** disconnected from ERE.
;**
;** Optional
;** Values : 1..n, default=3
;** RequiredPingFailures=10
RequiredPingFailures=

```

HAManager.ini 配置文件

下面是高可用性管理器配置文件（HAManager.ini）的样本摘录：

```

=====
;solidDB High-Availability Manager
; Configuration file HAManager.ini
; V. 0.3
; 2008-21-01
=====
;** HA Controller connect info, e.g.
;Server1_name = Server 1
;Server1_host = node1.acme.com
;Server1_port = 2220
;Server2_name = Server 2
;Server2_host = node2.acme.com
;Server2_port = 2220

; All the following lines are mandatory.
Window_title = HA Manager
Header_text = SolidDB HA Manager

; Display names, host addresses and port numbers
; of the SolidHAC (HA Controllers) instances
;Server 1 HA Controller
;-----
Server1_name = Server 1
Server1_host = localhost
Server1_port = 1234
Server1_user = foo
Server1_pass = bar
;
;Server 2 HA Controller
;-----
Server2_name = Server 2
Server2_host = 192.168.0.1
Server2_port = 1234
Server2_user = foo
Server2_pass = bar

```

4 对应用程序使用 HotStandby

本章说明应用程序应当如何处理 HotStandby 配置中的故障和切换。

用于连接至 HotStandby 服务器的两种方法

在 HotStandby 环境中可以使用两种方法来编写应用程序。除了“基本连接”以外，还提供了“透明连接”（TC）。使用“基本连接”时，客户机必须显式连接至每个 HSB 服务器；客户机借助“透明连接”仅建立一个称为“TC 连接”的逻辑连接。

这两种连接类型在 solidDB ODBC 和 JDBC 驱动程序中都受支持。通过相应地明确阐述通用连接字符串（数据源信息）来选择连接类型。

基本连接

使用“基本连接”时，应用程序必须考虑通过使用特定服务器地址单独连接至 HotStandby 配置中的每个服务器。如果发生了故障转移，那么活动连接将断开，应用程序必须重新连接至新的主服务器。

透明连接

使用“透明连接”时，应用程序不必处理连接至任何特定服务器，或者在发生了故障转移的情况下重新建立连接。应用程序将保持一个称为 TC 连接的逻辑连接（句柄）。

只要所指定的服务器集中有任何服务器处于 PRIMARY ACTIVE、PRIMARY ALONE 或 STANDALONE 状态，那么在进行故障转移和切换之后仍会保持连接句柄。在进行故障转移和切换时，驱动程序将执行一项称为连接切换的内部操作。会将连接切换告知应用程序，因为应用程序必须根据故障透明性级别来重新构造某些会话状态。借助 TC，可以均衡主服务器与辅助服务器之间的只读负载。简单地说，“透明连接”使得应用程序不必考虑服务器的多重性及其地址。

要点:

solidDB 工具（例如 solsql）不支持 TC 连接。

选择连接类型

以下兼容性矩阵可以帮助您通过对所选择的连接信息指示受支持的功能来选择连接类型：

表 7. 选择连接类型

功能	独立配置	HSB 配置
基本连接	Yes (BC Info)	Yes (BC Info)
透明故障转移	No	Yes (TC Info)
负载均衡	No	Yes (TC Info)

使用透明连接

当使用 solidDB 透明连接时，驱动程序在一定程度上会对应用程序隐藏两个 HSB 服务器的存在。

驱动程序提供了已映射至内部活动连接的单个逻辑 TC 连接。在理想情况下，当主服务器和辅助服务器都在活动状态下运行时，驱动程序还会维护一个备用连接，即，与辅助服务器的连接。此连接将设置为事件等待方式，它已准备好接收 HSB 状态更改事件。这些事件是有关驱动程序将使用的故障转移和切换的主要信息源。在某些情况下（例如，PRIMARY ALONE 操作），将缺少备用连接，但是每当有可能，驱动程序就会尝试执行它。备用连接的处理对于客户机完全透明。另一方面，将通过特殊的错误代码向客户机通知所产生的任何连接切换（即，更改 TC 连接至内部活动连接的映射）。

要点:

IBM solidDB 工具（例如，solsql）不支持 TC 连接。

TC 中的故障透明性

故障透明性用于处理对故障的掩盖。故障透明性级别是使用 TC Info 的 TF_LEVEL 属性设置的。

有三个故障透明性级别:

1. NONE，它将禁用故障透明性。这是缺省值。
2. CONNECTION，它将保持服务器连接。也就是说，在进行故障转移或切换服务器时不需要重新连接。
3. SESSION，它将使大多数会话属性使用非缺省值。此外，还将保留预编译语句。但是，打开的游标会关闭，并且正在进行的事务会异常终止。有关保留的会话属性的列表，请参阅第 61 页的『连接切换的编程』。

透明连接中的负载均衡

透明连接驱动程序使用两种方法来管理事务负载；一种方法用来处理读取密集型负载，另一种方法用来处理写入密集型负载。为了实现负载均衡，会将逻辑 TC 连接映射至更低级别的、称为“工作负载连接”的服务器连接。工作负载连接可能会随时间而变化，它对于应用程序通常是无关紧要的。但是，如果工作负载连接是必需的，那么可以通过一种方法识别当前工作负载连接的内容。

静态负载均衡配置

负载均衡方法包括:

1.

PREFERRED_ACCESS=READ_MOSTLY。此方法用于处理读取密集型负载。可以在辅助服务器和主服务器中执行只读事务。如果 **Cluster.ReadMostlyLoadPercentAtPrimary** 参数设置为 0，那么只读负载全部都将在辅助服务器中执行。

2.

PREFERRED_ACCESS=WRITE_MOSTLY。这是缺省值。此方法用于处理写入密集型负载。所有事务都将在主服务器中执行。这对应于典型 HotStandby 操作。

通过设置 `PREFERRED_ACCESS=READ_MOSTLY`，主服务器告诉驱动程序为了工作负载连接而连接至哪个服务器。如果将负载转移至辅助服务器，并且发出了写操作，那么就会将负载移交给主服务器并在主服务器中执行事务。在落实事务之后，又会将负载转移回辅助服务器。如果辅助服务器失败，那么连接将从辅助服务器故障转移到主服务器。

此外，引入一个新的负载均衡配置参数。它允许将一定数量的只读负载转移到主服务器。

```
[Cluster]
ReadMostlyLoadPercentAtPrimary=<n>
```

其中 $n = [0 \dots 100]$ 。缺省值为 50。

此参数定义在主服务器中转移的主要读取负载总量所占的百分比。根据此值选择了一个已指定工作负载的服务器。缺省情况下，50% 的连接使用主服务器，50% 的连接使用辅助服务器。对于大多数混合型负载来说，这是更合适的值。如果此值设置为 0，那么所有负载都将转移到辅助服务器中。在读取密集型（或者只读）应用程序使用 `PREFERRED_ACCESS=READ_MOSTLY` 并且（同时）写入密集型应用程序使用 `PREFERRED_ACCESS=WRITE_MOSTLY` 时，将此值设置为 0 是合适的。

注：

仅在“落实读”隔离级别执行负载均衡操作。如果服务器的隔离级别（启动）缺省值设置为其他值，那么设置 `PREFERRED_ACCESS=READ_MOSTLY` 会将此会话的隔离级别强制设置为“落实读”。可以将隔离级别动态复位为更高级别，即“可重复读”，但是这会禁用负载均衡。

注：

如果会话设置为“自动落实”方式，那么将禁用负载均衡。

动态控制负载均衡

如果已指定工作负载的服务器为辅助服务器，那么可以按程序将它更改为主服务器。

在会话级别，可以使用下列语句将工作负载连接服务器更改为主服务器：

- `SET WRITE`
- `SET ISOLATION LEVEL REPEATABLE READ`
- `SET ISOLATION LEVEL SERIALIZABLE`

如果此语句是事务的第一个语句，那么它将立即生效，否则将从下一个事务开始生效。

在事务级别，下列语句会在一个事务的时间内将工作负载连接服务器更改为主服务器：

- `SET TRANSACTION WRITE`
- `SET TRANSACTION ISOLATION LEVEL REPEATABLE READ`
- `SET TRANSACTION ISOLATION LEVEL SERIALIZABLE`

受影响的事务是通过使用此语句启动的事务，否则将为下一个事务。在主服务器中执行事务之后，工作负载连接服务器将恢复为对会话指定的工作负载连接服务器。

使用 SET [TRANSACTION] READ WRITE (SQL:1999) 语句可以恢复 SET [TRANSACTION] WRITE 语句的效果。另外，隔离级别语句具有相同的效果：

- SET ISOLATION LEVEL READ COMMITTED
- SET TRANSACTION ISOLATION LEVEL READ COMMITTED

故障转移透明性和负载均衡

当设置了透明故障转移 (**TF_LEVEL** 的值不为 NONE) 并且启用了负载均衡 (**PREFERRED_ACCESS=READ_MOSTLY**) 时，应用的故障转移策略为如下所示：

1. 主服务器故障：所有负载都转移到新的处于 PRIMARY ALONE 状态的主服务器。
2. 辅助服务器故障：所有负载都转移到主服务器（处于 PRIMARY ALONE 状态）。
3. 服务器之间的连接断开；服务器处于 PRIMARY ALONE 和 SECONDARY ALONE 状态：如果有一个只读事务正在辅助服务器中执行，那么也会在辅助服务器中成功落实此事务。所有后续事务都将转移到主服务器中（处于 PRIMARY ALONE 状态）。

当恢复正常的热备用操作时（服务器分别处于 PRIMARY ACTIVE 和 SECONDARY ACTIVE 状态），就会在主服务器与辅助服务器之间重新均衡负载。

注：

即使对于 TF_LEVEL=NONE（无故障透明性），某些基本故障转移功能仍然可用，例如，当辅助服务器失败时，从辅助服务器故障转移到主服务器。所有其他故障都将导致通信链路失败。因此，对于 TF_LEVEL=NONE，在大多数故障情况下，应用程序需要使用相同的 TC Info 重新建立连接。为了避免重新建立连接，建议在使用了负载均衡的情况下始终启用故障透明性。

在实现了负载均衡的情况下执行过程

除非在过程声明中通过使用 SQL 标准子句 *SQL Data Access Indication* 将 SQL 存储过程指定为只读过程，否则所有 SQL 存储过程都将在主服务器中执行。

```
<SQL-data-access-indication> ::=  
    NO SQL |  
    READS SQL DATA |  
    CONTAINS SQL |  
    MODIFIES SQL DATA
```

为了避免不必要地移交只读过程和函数，可以声明下列值之一：

- NO SQL
- READS SQL DATA
- CONTAINS SQL

只有 MODIFIES SQL DATA（这是缺省值）才会进行事务移交。

此子句位于（可选的）RETURNS 子句与过程主体之间。例如：

```
"CREATE PROCEDURE PHONEBOOK_SEARCH  
(IN FIRST_NAME VARCHAR, LAST_NAME VARCHAR)  
RETURNS (PHONE_NR NUMERIC, CITY VARCHAR)  
READS SQL DATA  
BEGIN  
-- procedure_body  
END";
```

透明连接信息的语法

当使用 solidDB 透明连接时，客户机将仅建立一个称为“TC 连接”的逻辑连接。此连接是在 TC Info 中指定的。TC Info 对两个 HSB 配置都将执行透明故障转移和负载均衡。

solidDB TC Info 的完整语法为如下所示：

```
<solidDB TC Info> ::= {[<failure transparency level attribute>]
[<preferred access attribute>] <connect target list>} | <cluster info>

<failure transparency level attribute> ::= TF_LEVEL={NONE |
CONNECTION | SESSION}

<preferred access attribute> ::= PREFERRED_ACCESS={WRITE_MOSTLY |
READ_MOSTLY}

<connect target list> ::= [SERVERS=]<connection string>[, <connection string > ...]

<cluster info> ::= CLUSTER <connect string>[, <connect string>...]
```

此外，还可以使用下列缩写。

表 8. TC Info 缩写

缩写	相应的语法
TF	TF_LEVEL
CO	CONNECTION
SES	SESSION
PA	PREFERRED_ACCESS
RM	READ_MOSTLY
WM	WRITE_MOSTLY
S	SERVERS

故障透明性属性

由 TF_LEVEL 属性表示的故障透明性用于处理对故障的掩盖。有三个故障透明性级别：

1. NONE，它将禁用故障透明性。这是缺省值。
2. CONNECTION，它将保持服务器连接。也就是说，在进行故障转移或切换服务器时不需要重新连接。
3. SESSION，它将使某些会话属性保持具有非缺省值。此外，还将保留预编译语句。但是，打开的游标会关闭，并且正在进行的事务会异常终止。

负载均衡属性

首选访问（PREFERRED_ACCESS）属性指示是否应用了负载均衡。有两个级别可供选择：

1. WRITE_MOSTLY，会将工作负载全部转移到主服务器。这是缺省值。
2. READ_MOSTLY，缺省情况下，会将工作负载转移到辅助服务器。写入事务会移交给主服务器。

最后，solidDB TC Info 包含服务器地址列表。驱动程序将从左到右扫描整个列表，并设法找到主服务器和辅助服务器。因此，必须将首选配置放在列表开头。列表中的其余部分可能包含一些备用地址，可以在系统生存期内的某个时间激活这些备用地址。

应让此列表保持比较简短，因为在发生错误的情况下，要经过较长时间之后才会将错误返回给应用程序。将逐个尝试每个地址，这涉及到所指定的登录超时。列表中的地址数不受限制。

如果未指定 `TF_LEVEL` 或 `PREFERRED_ACCESS` 属性（或者 `TF_LEVEL=NONE`），那么连接行为将后退到使用“基本连接”时的状态。如果给出了多个连接字符串，那么将与列表中第一个接受连接请求的服务器建立连接。

`CLUSTER` 关键字与以下内容的作用相同：

```
TF_LEVEL=SESSION PREFERRED_ACCESS=READ_MOSTLY SERVERS=
```

例如，可以将以下 `TC Info`：

```
F_LEVEL=SESSION PREFERRED_ACCESS=READ_MOSTLY
SERVERS=tcp srv1.acme.com 1315, tcp srv2.acme.com 1315
```

替换为：

```
CLUSTER tcp srv1.acme.com 1315, tcp srv2.acme.com 1315
```

客户端 INI 文件

```
[Data Sources]
Cluster1=
  TF_LEVEL=SESSION
  PREFERRED_ACCESS=READ_MOSTLY
  SERVERS=
    tcp -c 1000 srv1.dom.acme.com 1315,
    tcp srv2.dom.acme.com 1315,
    tcp srv3.dom.acme.com 1316
```

ODBC 中的连接字符串

```
rc = SQLConnect(comHandle, "CLUSTER
  tcp -c 1000 srv1.dom.acme.com 1315,
  tcp srv2.dom.acme.com 1315,
  tcp srv3.dom.acme.com 1316", ...
```

TC Info 属性组合

下表总结了可能存在的 `TC Info` 属性组合并且提供了最终获得的连接能力：

表 9. 可能存在的 `TC Info` 属性组合

PREFERRED_ACCESS:	TF_LEVEL: 未指定 或者 NONE	TF_LEVEL: CONNECTION	TF_LEVEL: SESSION
未指定	<ul style="list-style-type: none"> 不支持故障转移或切换 不进行负载均衡 （基本连接）	<ul style="list-style-type: none"> 透明故障转移（不保留会话状态） 透明切换 仅适用于主服务器中的工作负载 不进行负载均衡 	<ul style="list-style-type: none"> 透明故障转移（保留会话状态） 透明切换 仅适用于主服务器中的工作负载 不进行负载均衡

表 9. 可能存在的 TC Info 属性组合 (续)

PREFERRED_ACCESS:	TF_LEVEL: 未指定或者 NONE	TF_LEVEL: CONNECTION	TF_LEVEL: SESSION
WRITE_MOSTLY (缺省值)	<ul style="list-style-type: none"> 不支持透明故障转移 透明切换 仅适用于主服务器中的工作负载 不进行负载均衡 	<ul style="list-style-type: none"> 透明故障转移 (不保留会话状态) 透明切换 仅适用于主服务器中的工作负载 不进行负载均衡 	<ul style="list-style-type: none"> 透明故障转移 (保留会话状态) 透明切换 仅适用于主服务器中的工作负载 不进行负载均衡
READ_MOSTLY	<ul style="list-style-type: none"> 不支持透明故障转移 透明切换 辅助服务器和主服务器中的工作负载 负载均衡 	<ul style="list-style-type: none"> 透明故障转移 (不保留会话状态) 透明切换 辅助服务器和主服务器中的工作负载 负载均衡 	<ul style="list-style-type: none"> 透明故障转移 (保留会话状态) 透明切换 辅助服务器和主服务器中的工作负载 负载均衡

处理 TC Info 矛盾

TC Info 的属性可能会与“使实际服务可用”产生矛盾。在这些情况下，将保证建立连接，但是会发出 SUCCESS_WITH_INFO 警告。

在下列情况下将处理矛盾：

- 指定了 PREFERRED_ACCESS，但是未启用 HSB。将启用基本连接。
- 指定了 TF_LEVEL，但是未启用 HSB。将启用基本连接。

在 JDBC 中建立透明连接

在 JDBC 中，使用两个非标准连接属性来启用透明连接。

故障透明性用于处理对故障的掩盖。它同样应用于 HotStandby 和集群透明方式。可使用 solid_tf_level 连接属性来启用故障透明性。可以采用助记符（例如，NONE）或者数字（0 代表 NONE）来提供此连接属性的值，但主要使用助记符。必须以字符串形式提供此值。有三个故障透明性级别：

- NONE。NONE 对应的数值为 0。
- CONNECTION。CONNECTION 对应的数值为 1。
- SESSION。SESSION 对应的数值为 3。

有关这些值的更多信息，请参阅第 54 页的『TC 中的故障透明性』。

首选访问属性指示是否分配了只读负载。使用 solid_preferred_access 连接属性来启用首选访问属性。可以采用助记符或者数字来提供此属性的值，但主要使用助记符。必须以字符串形式提供此值。有两个级别可供选择：

- WRITE_MOSTLY。WRITE_MOSTLY 还会将连接设置为 WRITE MOSTLY 方式。通过指定数值不能完成此任务。WRITE_MOSTLY 对应的数值为 0。
- READ_MOSTLY。READ_MOSTLY 对应的数值为 1。

还有一个特定于 TC/TF 的 `solid_tf1_reconnect_timeout` 属性，用于指定连接重新连接超时（按毫秒计）。缺省值为 10000 毫秒（也就是 10 秒）。

作为扩展 JDBC 连接字符串的一部分提供了服务器地址列表：

```
conStr= "jdbc:solid://host_name:port [,host_name:port].../user_name/password";
```

地址列表中的地址数不能超过 20。

注意：

在 **JDBC** 中使用透明连接时，必须处理显式地删除语句对象。垃圾回收器将不会检测未引用的语句对象。

在 JDBC 中使用透明连接

```
...
String conStr = "jdbc:solid://srv1.acme.com:1323,srv2-acme.com:1423/dba/dba";
Properties prop = new Properties();
prop.setProperty("solid_tf_level", "CONNECTION");
...
Connection c = DriverManager.getConnection(conStr, prop);
...
```

连接错误处理

对 TC 连接发出连接请求之后，如果至少找到了一个合适的服务器并且已连接至此服务器，那么就认为此连接请求成功。

服务器应处于下列任一状态：**PRIMARY ACTIVE**、**PRIMARY ALONE** 或 **STANDALONE**。否则，就认为连接失败。将扫描地址列表一次。

可能有各种不同的原因会造成连接请求失败。下列错误情况涵盖了大多数原因：

表 10. 连接请求错误

SQLSTATE	本机代码	消息文本和描述
08001	25217	客户机无法建立连接 描述：驱动程序已使用 TC 连接信息来查找合适的服务器并与它连接。但是，由于下列原因之一而失败： <ul style="list-style-type: none"> 找不到地址列表中所列示的主机 找到了主机，但是登录时发生超时 找到了主机，但是登录时被拒绝 找到了主机，但是不处于 PRIMARY/STANDALONE 状态
HY000	21307	连接信息无效... 描述：在基本连接字符串或者 TC 连接信息（数据源信息）中发现了语法错误。

表 10. 连接请求错误 (续)

SQLSTATE	本机代码	消息文本和描述
HY000	21300	协议 ... 不受支持。 描述: TC 连接信息开头的字符串“TC”存在拼写错误 (或者在基本连接字符串中提供的协议名称不正确)。

在某些情况下接受连接时发出了警告。

表 11. 警告

SQLSTATE	本机代码	消息文本和描述
0100	25218	已连接至 STANDALONE 或 PRIMARY ALONE 服务器。 已经尝试为 TF_LEVEL 或 PREFERRED_ACCESS 设置任何非缺省值, 并且只有一个服务器可用。在这种情况下, 故障透明性和负载均衡都不可用。

连接切换的编程

连接切换的处理原则

连接切换是指驱动程序更改活动服务器连接这种情况。通常, 在故障转移到辅助服务器或者在服务器之间切换时就会进行连接切换。

更具体地说, 从下列其中一个事件中检测到需要进行连接切换:

- 辅助服务器中使得状态更改为 PRIMARY ALONE (故障转移) 或 PRIMARY ACTIVE (切换) 的事件。这是执行连接切换的主要方式 (也是速度最快的方式)。
- 指示主服务器中发生了状态更改。
- 活动连接上发生链路故障。
- 活动连接上发生连接超时。

驱动程序通过以下两个步骤来执行连接切换:

1. 检测到需要进行连接切换。对于暂挂的请求或以下请求, 驱动程序将返回以下连接切换错误:

表 12. 连接切换请求

SQLSTATE	本机代码	消息文本和描述
HY000	25216	<p>连接切换，某些会话上下文可能已丢失</p> <p>描述：驱动程序已发现需要进行连接切换。期望客户机发出事务回滚调用以最终确定连接切换。在每个连续的网络请求调用时都将接收到此错误代码和消息，直到发出了回滚调用为止</p>

2. 客户机程序发出回滚命令（ODBC: `SQLEndTran()` with `SQL_ROLLBACK`; JDBC: `Connection.rollback()`）。如果成功完成了回滚，那么已将新的活动连接映射至可以使用的 TC 连接。

注：一些连续的 ODBC 调用可能会返回连接切换错误。因此，必须规定当任何 ODBC 网络请求返回此错误时始终都进行回滚。如果在执行事务过程中发生此错误，那么必须重新执行此事务。

另一方面，如果无法建立新的活动连接，那么将返回以下错误代码：

表 13. 通信链路故障

SQLSTATE	本机代码	消息文本和描述
08S01	14503	<p>通信链路故障</p> <p>描述：驱动程序未能建立新的活动连接。TF 连接已断开，客户机需要使用数据源信息重新建立连接才能进行后续操作。</p>

在接收到回滚调用之后，驱动程序将使用一些备用方法来查找新的活动连接。在最简单的情况下，它将使用备用连接。如果此连接处于不正确的状态，那么驱动程序将等待两秒钟以等待正确的事件达到。如果事件未达到以及其他情况下，驱动程序将后退到 TC 连接信息中的地址列表，并且最多将在 10 秒钟之内反复执行连接序列。如果所有操作都失败，那么驱动程序将返回上述错误。

此错误导致的结果就是连接断开，正如应用程序所看到的。对此连接发出的任何后续请求都将导致相同错误。

会话状态的保留

在驱动程序执行连接切换之后，某些会话上下文可能会丢失，客户机必须重新构造这些会话上下文。保留的状态数量由透明故障转移级别指定，用 TC Info 属性 `TF_LEVEL` 来表示透明故障转移级别。

实质上，对于 TF 级别 CONNECTION，不会保留任何状态；而对于 SESSION 级别，将保留大多数会话状态。会话状态的保留是通过将必需数据高速缓存在驱动程序中来实现的。要获得更高的透明性级别，需要进行高速缓存的请求的响应时间就会越长，驱动程序中的内存使用量也会增大。

无论对于哪个 TF 级别，进行故障转移时都会保留下列各项：

- 由于进行事务回滚而丢失了当前事务的更新
- 丢失了打开的游标及其位置。

下表总结了会话状态保留。

表 14. 会话状态保留

TF_LEVEL	保留的状态
CONNECTION	不保留会话状态。

表 14. 会话状态保留 (续)

TF_LEVEL	保留的状态
SESSION	<p>预编译语句</p> <ul style="list-style-type: none"> • 保留预编译状态。 <p>保留了下列语句的作用:</p> <ul style="list-style-type: none"> • SET CATALOG • SET SQL INFO • SET SQL SORTARRAYSIZE • SET SQL CONVERTORSTOUNIONS • SET SQL SORTEDGROUPBY • SET SQL { OPTIMIZEROWS OPTIMISEROWS } • SET SIMPLEOPTIMIZERRULES • SET LOCK TIMEOUT <seconds> • SET OPTIMISTIC LOCK TIMEOUT • LOCK_TIMEOUT • SET IDLE TIMEOUT • SET STATEMENT MAXTIME • SET ISOLATION LEVEL • SET DURABILITY • SET SAFENESS • SET SCHEMA • SET SQL JOINPATHSPAN • SET SYNC USER • SET SYNC MODE <p>保留了下列标准 ODBC 属性:</p> <ul style="list-style-type: none"> • SQL_ATTR_ACCESS_MODE <p>(SET READ ONLY 和 SET READ WRITE)</p> <ul style="list-style-type: none"> • SQL_ATTR_CURRENT_CATALOG <p>(与上面的 SET CATALOG 作用相同)</p> <ul style="list-style-type: none"> • SQL_ATTR_AUTOCOMMIT

其他私有 ODBC 属性

下列只读 ODBC 属性可供应用程序员用于任何格式的 TC Info。

- SQL_ATTR_PA_LEVEL

(integer, 首选访问级别: 0=WRITE_MOSTLY, 1=READ_MOSTLY)

此属性指示是否使用了负载均衡。

- SQL_ATTR_TC_WORKLOAD_CONNECTION

(string, 工作负载连接的服务器名称)

当前工作负载连接服务器; 如果在落实之前进行了查询, 那么此值指示用于落实事务的服务器。

- SQL_ATTR_TF_LEVEL

(integer, TF 级别: 0=NONE, 1=CONNECTION, 3=SESSION)

故障透明性级别。

- SQL_ATTR_TC_PRIMARY

(string, 主服务器连接字符串)

始终有一个值用来指示当前主服务器。

- SQL_ATTR_TC_SECONDARY

(string, 辅助服务器连接字符串)

如果满足下列条件, 那么此值指示已指定工作负载的服务器:

1. PA=READ_MOSTLY, 并且
2. 辅助服务器就是指定工作负载的服务器 (这是缺省情况)。

否则, 将返回一个空字符串。

注: 不能将私有 ODBC 属性与 Windows ODBC 驱动程序管理器配合使用。如果需要在 Windows 中使用私有 ODBC 属性, 那么必须将 Solid[®] ODBC 驱动程序导入库 (solidimpodbca.lib 或 solidimpodbcu.lib) 直接与应用程序进行链接。

使用基本连接

使用“基本连接”时, 应用程序必须处理通过使用特定服务器地址单独连接至 HotStandby 或集群配置中的每个服务器。如果发生了故障转移, 那么活动连接将断开, 应用程序必须重新连接至新的主服务器。

没有透明性的基本连接

```
Connect=tcp srv1.dom.acme.com 1315
```

从应用程序重新连接至主服务器

为 HotStandby 准备客户机应用程序

已经与主服务器断开连接的客户机程序必须能够重新连接至新的主服务器 (也就是旧的辅助服务器)。您必须编写客户机应用程序以便能够完成下列任务:

1. 识别主服务器不再可用于写入事务。
2. 连接至另一个服务器或者切换为使用先前创建的连接。
3. 考虑当前已中断的事务是否已丢失/异常终止, 并且必须在新的主服务器上重新执行。

获取辅助服务器地址

最容易获取辅助数据库服务器的连接信息的方法是使用 ADMIN COMMAND 'hotstandby cominfo' 命令，此命令将为 HSB 对中的另一个服务器提供连接信息。

关于此任务

1. 当应用程序首次连接至主服务器时，应用程序可以执行 ADMIN COMMAND 'hotstandby cominfo' 命令并存储结果。注意，当 cominfo 命令返回一个值时，它并不意味着主服务器与辅助服务器当前已建立连接。“cominfo”命令仅仅返回在 solid.ini 配置文件的 **Connect** 参数中指定的值，或者返回最近使用 hsb parameter connect 命令指定的值。如果需要检查主服务器与辅助服务器之间的连接状态，那么可以使用 ADMIN COMMAND 'hotstandby status connect'。
2. 稍后，如果主服务器失败，那么应用程序可以使用所存储的信息来连接至辅助服务器（新的主服务器）。

在客户机应用程序中检测 HotStandby 服务器故障

要使用 HotStandby (HSB) 组件，应用程序必须知道何时从失败的主服务器切换到辅助服务器（也就是新的主服务器）。这可以通过几种方法来实现。最佳方法是仅检查您调用函数之后所产生的返回码，以了解您是否接收到错误并指出您应当切换到另一个服务器。

还可以监视服务器的状态（例如，检查主服务器以了解其状态是否已更改为 PRIMARY UNCERTAIN）。

如果发生下列错误，那么您应当尝试切换到另一个服务器：

- 10013: 事务是只读的
- 10041: 数据库是只读的
- 10047: 复制事务异常终止
- 11002: 磁盘已满
- 11003: 超出了配置
- 14501: 操作失败
- 14502: RPC 参数无效
- 14503: 通信错误
- 14506: 服务器已关闭（例如，由于它当前是 HSB copy 或 netcopy 操作的目标而关闭）
- 14510: 通信写入操作失败
- 14511: 通信读取操作失败
- 14518: 连接已断开
- 14519: 用户被拒绝（例如，由于某些管理操作而被拒绝）
- 14529: 操作超时
- 20009: 会话错误，写入操作失败
- 21306: 未找到服务器，连接失败
- 21308: 连接已断开（写入失败并提供了代码...）
- 21318: 操作失败（不常见的返回码）

ODBC 应用程序

对于无法建立连接的 ODBC 应用程序（例如，由于数据库服务器无法运行而无法建立连接）将返回以下错误消息：

- SQLState = 08001 - 客户机无法建立连接

此外，将产生以下 solidDB 通信错误消息：

- 21306 - 未找到服务器 “*server_name*”，连接失败。

如果在双向操作（例如，执行查询并访存结果）中连接失败（例如，由于发生网络故障而连接失败），那么将返回以下错误消息：

- SQLState = 08S01 - 通信链路故障

JDBC 应用程序

对于无法建立连接的 JDBC 应用程序（例如，由于数据库无法运行而无法建立连接）将返回以下错误消息：

- SQLState = 08001 - 无法连接至数据源。

如果在双向操作（例如，执行查询并访存结果）中连接失败（例如，由于发生网络故障而连接失败），那么将返回以下错误消息：

- SQLState = 08S01 - 通信链路故障

注：

对于同一个错误代码（08001），ODBC 和 JDBC 将提供不同的错误消息。

将应用程序切换到新的主服务器

在应用程序检测到它无法将事务发送至“旧的主服务器”之后，应用程序必须轮询旧的主服务器和旧的辅助服务器，直到找到处于 PRIMARY ACTIVE、PRIMARY ALONE 或 STANDALONE 状态的服务器为止。

通过让应用程序尝试连接至服务器并在建立连接之后检查服务器状态来完成轮询。连接成功之后，客户机可以使用 SQL 函数 HOTSTANDBY_STATE 来请求服务器状态，在第 68 页的『使用 HOTSTANDBY_STATE 函数』这一节中描述了此函数。

注意：

切换之后，所有打开的数据库对象（例如，预编译语句）以及打开的游标和事务将不再处于活动状态。因此，必须再次初始化这些对象。此外，如果您先前使用的是临时表或瞬态表（solidDB 主内存引擎功能部件），那么这些表在新的主服务器上将为空。

使用 HOTSTANDBY_CONNECTSTATUS 函数

要在从应用程序重新连接至主服务器时验证连接状态信息，可以使用 HOTSTANDBY_CONNECTSTATUS 函数。此函数等价于管理命令 hotstandby status connect。

此函数没有自变量，并且将返回下列状态值之一：

表 15. *HOTSTANDBY_CONNECTSTATUS* 状态值

状态	描述
已连接	连接处于活动状态。从主服务器和辅助服务器中都将返回此状态。
正在连接	主服务器正在连接至辅助服务器。从主服务器和辅助服务器中都将返回此状态。
同步复制	主服务器已连接至辅助服务器，但是尚未完全复制事务日志。从主服务器和辅助服务器中都将返回此状态。
已断开连接	连接已断开。从主服务器和辅助服务器中都将返回此状态。

使用 *HOTSTANDBY_STATE* 函数

要实现使用应用程序来轮询主服务器和辅助服务器，可以使用 *HOTSTANDBY_STATE* 函数来完成。此函数等价于管理命令 *hotstandby state*。在当前 HotStandby 已连接至服务器的情况下，此函数允许应用程序请求当前 HotStandby 的状态。

注：此函数没有自变量。有关此函数可能返回的每种状态的描述，请参阅第 89 页的『验证 HotStandby 服务器状态』。

样本伪码

无论应用程序是否支持 HSB，它都应该具有错误处理方法，以允许此应用程序重放失败/异常终止的事务。

在非 HSB 环境中，事务可能会因并行冲突（乐观表）或死锁（悲观表）而异常终止。应用程序必须发现这些错误情况，并且自动重试此事务或者要求交互用户重新执行此事务。

如果应用程序中已经具有代码用来处理失败事务或者异常终止的事务，那么扩展此代码以利用 HSB 就相对比较容易。

在一个非常简单的示例中，具有不了解 HA 的应用程序处理的正确错误处理的应用程序伪码看起来类似于以下内容：

```
BEGIN TRANSACTION
EXECUTE APPLICATION LOGIC
PREPARE & EXECUTE STATEMENTS
COMMIT TRANSACTION
IF ERROR OCCURRED
    IF ERROR == concurrency conflict or deadlock
        GO TO BEGIN TRANSACTION
    END IF
    other error handling
END IF ;
```

很容易改进上述应用程序以使它了解 HA。您必须增加一些代码，以使应用程序可以完成下列任务：

- 连接至两个服务器中的任何一个，而不是只能连接至一个服务器；并且

- 发生错误时，找出当前处于下列其中一种状态的服务器：PRIMARY ACTIVE、PRIMARY ALONE 或 STANDALONE。

伪码看起来应该类似于以下内容：

```
BEGIN TRANSACTION
EXECUTE APPLICATION LOGIC
PREPARE & EXECUTE STATEMENTS
COMMIT TRANSACTION
IF ERROR OCCURRED
    IF ERROR == server unavailable for write transactions
        FIND CURRENT PRIMARY SERVER
        GO TO BEGIN TRANSACTION
    END IF
    IF ERROR == concurrency conflict or deadlock
        GO TO BEGIN TRANSACTION
    END IF
    IF ERROR == something else
        other error handling
    END IF
END IF
```

用于查找当前主服务器的逻辑也非常简单。只需同时检查两个服务器当前的状态（必要时尝试重新连接），如果其中任何一个服务器处于 PRIMARY ACTIVE、PRIMARY ALONE 或 STANDALONE 状态，那么将此服务器设置为当前的主服务器。如果两个服务器都不满足此条件，那么稍等一会儿，然后重新检查当前服务器状态。

重新连接至辅助服务器

在某些情况下，您可能需要连接至当前辅助服务器（如果它已经启动）。应用程序可以向辅助服务器提交只读查询；这在有时候可以帮助您均衡服务器之间的工作负载。

应用程序只能以只读方式连接至辅助数据库。请注意，客户机可以通过使用服务器的 solid.ini 配置文件的 [HotStandby] 节中的下列参数值来连接至辅助服务器（只能以只读方式进行连接）：

- 主服务器中的 **Connect** 参数
- 辅助服务器中的 **Listen** 参数

还可以使用以下命令来获取服务器的伙伴的连接信息：

```
ADMIN COMMAND 'hotstandby cominfo';
```

因此，如果您已连接至当前的主服务器，那么可以通过使用 cominfo 查询来获取当前辅助服务器的地址。

高级复制需求

通过 solidDB HotStandby 组件可以使高级复制系统的任何节点都具有高可用性。

当高级复制系统的主数据库和副本数据库正在使数据同步时，在数据库服务器对的主服务器之间也将进行同步。换句话说，主数据库的主服务器与副本数据库的主服务器进行通信。请参阅第 6 页的图 3。

数据库服务器在任何时候都可以故障转移到其辅助服务器，当数据库服务器与另一个使用高级复制的服务器同步数据时也是如此。如果在同步期间进行故障转移，那么将停止执行同步消息，必须在完成故障转移之后继续完成此进程。有关在发生错误之后如何进行同步的详细信息，请参阅 *Advanced Replication Guide*。

如果包含高级复制主数据库的服务器实现了与 solidDB HotStandby 之间容错，那么主数据库的副本必须知道用于两个主服务器的连接字符串。为此，在每个副本数据库中执行以下语句：

```
SET SYNC CONNECT 'connect_string_to_server_1, connect_string_to_server_2'  
TO MASTER master_nodename
```

在下图中，灰色箭头表示先前与原来的主服务器建立的连接；而黑色箭头表示与新的主服务器（也就是旧的辅助服务器）建立的新连接。如果与旧的主服务器的同步失败，那么将使用备用连接。

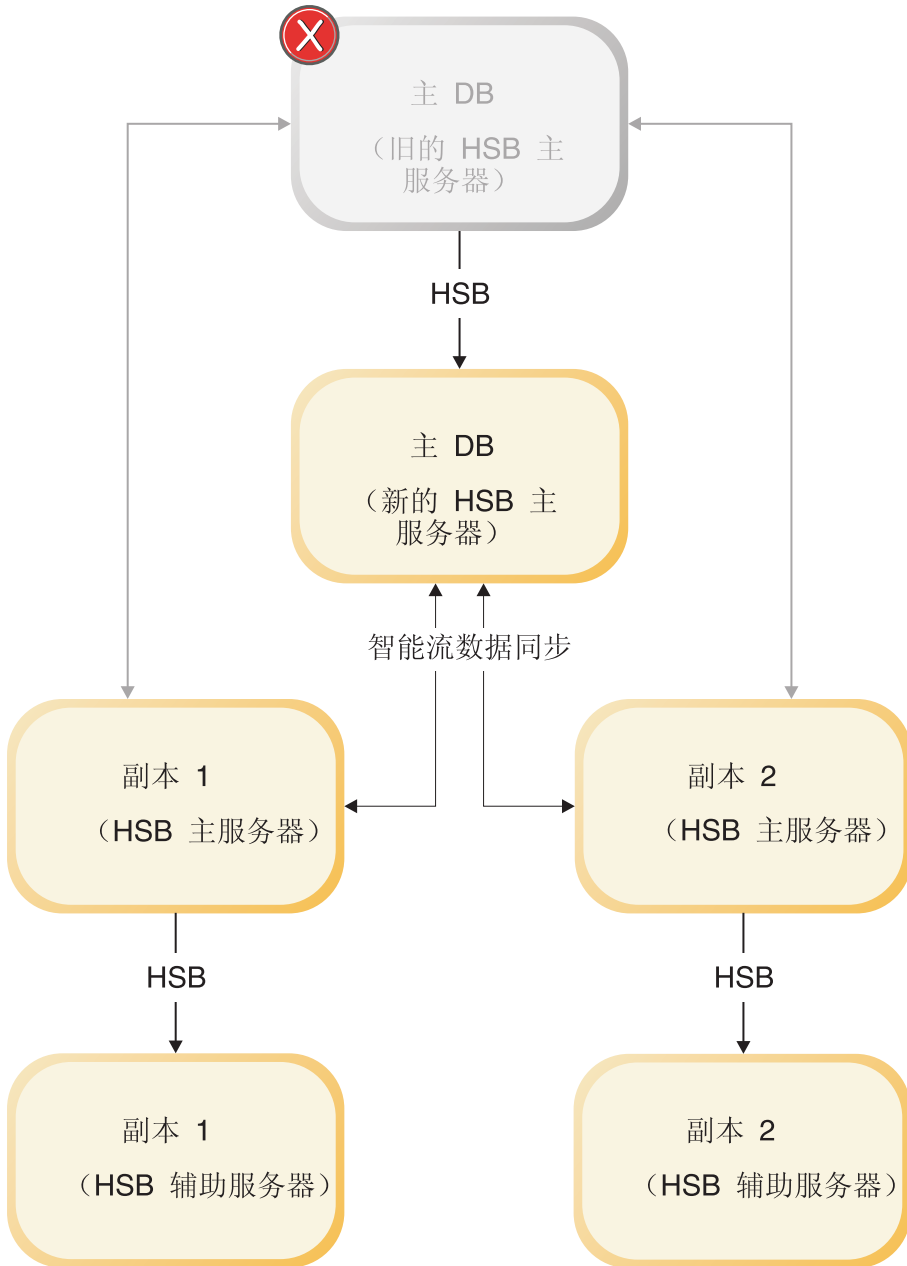


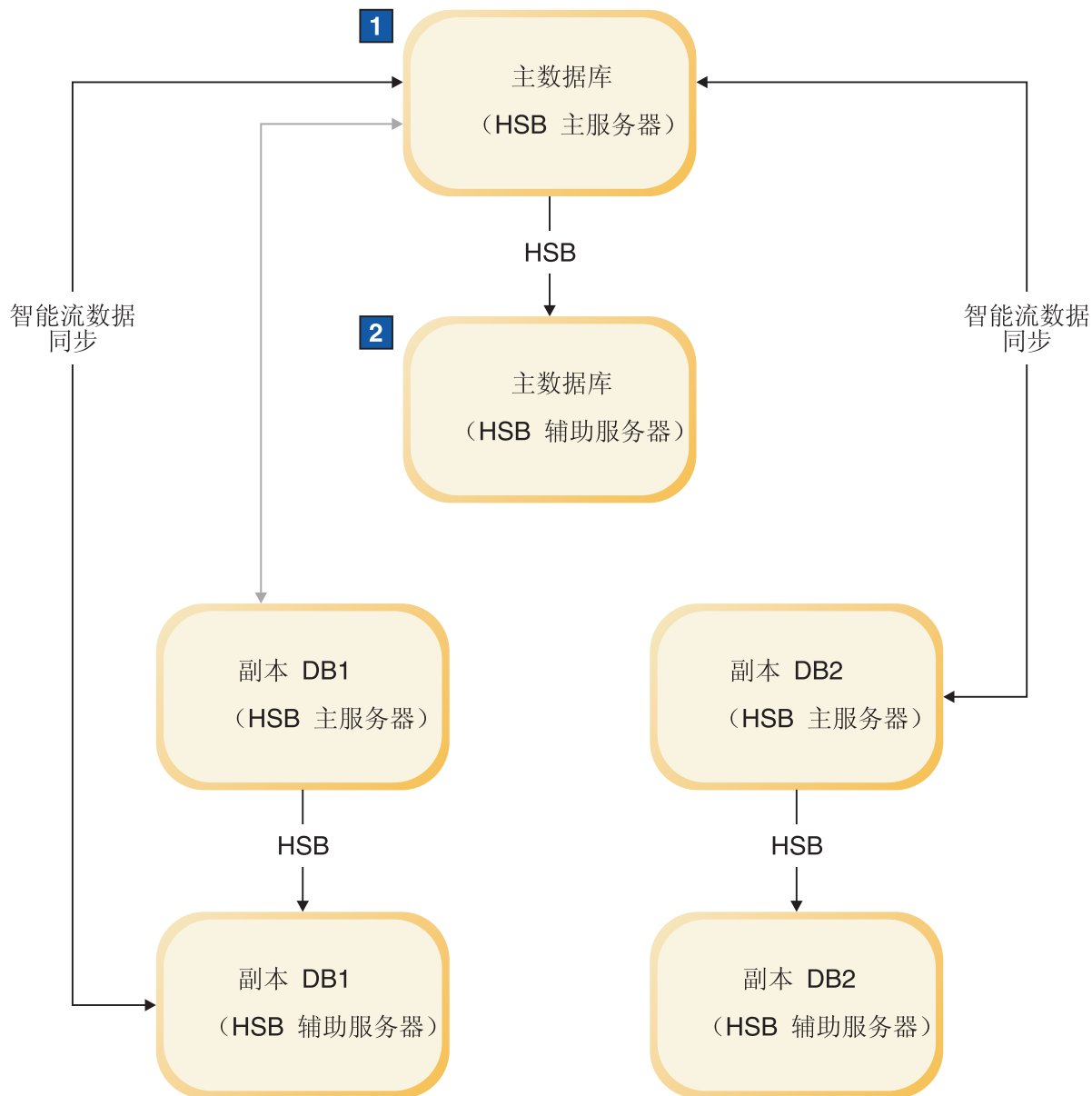
图 11. 主数据库故障转移

如果使用 solidDB HotStandby 的服务器是一个包含副本数据库的服务器，并且主服务器使用远程过程调用（在 `node_name` 中调用过程）在副本服务器中运行过程（例如，启动同步），那么必须将两个副本服务器的连接字符串都告知主服务器。通常，主服务器使用远程过程调用来启动与副本数据库同步。要将副本服务器对的连接字符串告知主服务器，请在主数据库中执行以下语句：

```
SET SYNC CONNECT 'connect_string_to_server_1,  
connect_string_to_server_2' TO REPLICa replica_nodename
```

或者，可以将此语句保存在副本服务器中，然后在下一次进行同步时将它传播至主服务器。在这种情况下，请使用以下语句：

```
SAVE SET SYNC CONNECT 'connect_string_to_server_1,  
connect_string_to_server_2' TO REPLICa replica_nodename
```



1. ReplicaDB 中的故障转移
2. SET SYNC CONNECT 'tcp machine4 1315' REPLICA TO replicaDB1

图 12. 副本故障转移

如果主服务器从不在副本服务器中执行远程过程调用，那么不需要上述语句。

5 使用 HotStandby API 命令

在任何用于管理 solidDB HotStandby 高可用性的软件中，都使用 HotStandby API (HSB API) 来监视和控制服务器进程。随此产品一起提供的 HA 控制器就是这种程序的一个示例。另一个示例是“看守程序”样本程序。

HSB API 是作为 SQL 的语法扩展提供的，它以非标准语句的形式出现：

```
ADMIN COMMAND hotstandby hsb-command options
```

或者

```
ADMIN COMMAND hsb hsb-command options
```

可以通过任何支持 SQL 的交互式工具（例如，solsql）来发出 HSB 命令，也可以通过 ODBC 或 JDBC 按程序发出 HSB 命令。

可以使用本节来编写您自己的应用程序，以管理 solidDB 中的高可用性。这可能是必需的，例如，为了实现与外部集群管理软件进行集成。

本节中包含下列主题：

切换服务器状态

HotStandby 组件要求在必要时自动切换或者由用户手动切换服务器状态。

在生产使用中，通过使用自动切换状态（即，执行故障转移）来选择服务器状态。这是 solidDB 中实现的自动高可用性控件的职责。在 solidDB 中，自动高可用性控件由高可用性控制器（HAC）来处理。

切换意味着颠倒主服务器与辅助服务器的角色（前提是它们都在运行）。进行各种维护时可能需要这样做。

另一方面，故障转移是在主服务器失败时由辅助服务器来替代主服务器角色这样一项操作。

执行切换

HAC 可以通过在辅助服务器中发出 ADMIN COMMAND 'hotstandby switch primary'；命令或者在主服务器中发出 ADMIN COMMAND 'hotstandby switch secondary'；命令来颠倒服务器的角色。

无论两个服务器是否建立了连接，都可以使用此命令。如果服务器之间已连接，那么它们的状态将颠倒；旧的辅助服务器将成为新的主服务器，而旧的主服务器将成为新的辅助服务器。如果服务器之间未连接，那么旧的辅助服务器将成为新的主服务器，而其他服务器的状态将不变。

下图说明了当服务器之间建立了连接的情况下发出 switch secondary 或 switch primary 命令时将发生的情况。请注意，仅在处于 SECONDARY 状态（例如，SECONDARY

ACTIVE) 的服务器上发出 `switch primary` 命令; 而 `switch secondary` 命令仅用于处于 PRIMARY 状态 (例如, PRIMARY ACTIVE) 的服务器上。

状态切换

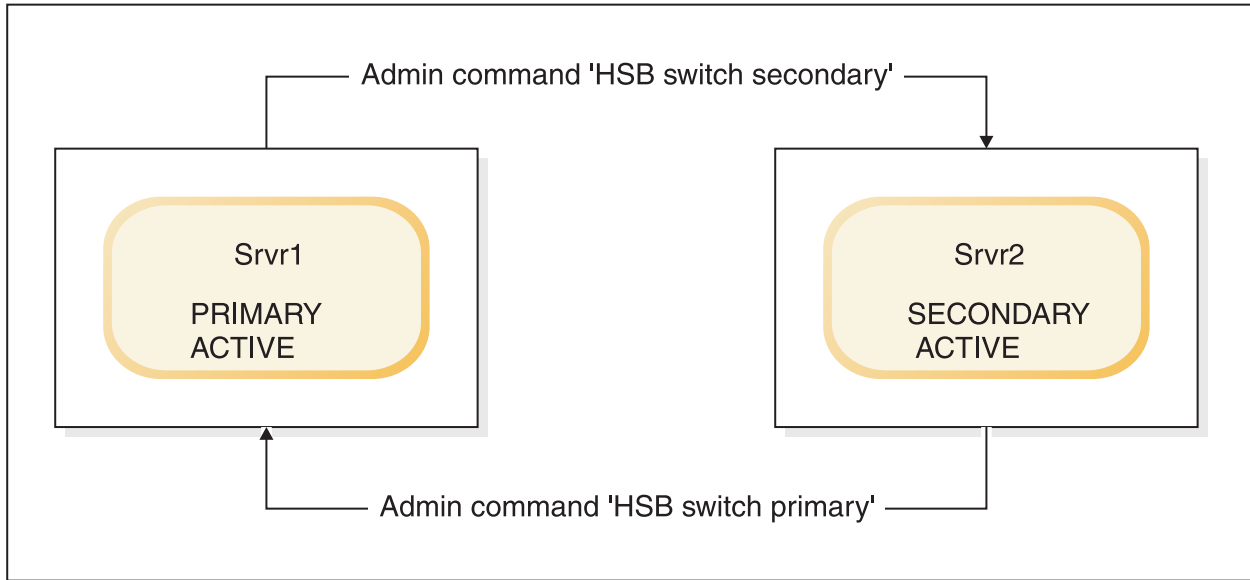


图 13. 状态切换

当执行 `hotstandby switch primary` 命令以将辅助服务器 (Srvr2) 切换为主服务器时, 如果辅助服务器 (Srvr2) 未连接至另一个服务器 (Srvr1), 那么将返回错误。

如果两个服务器之间建立了连接, 那么它们将切换状态。换句话说, 旧的主服务器 (Srvr1) 将成为新的辅助服务器, 而旧的辅助服务器 (Srvr2) 将成为新的主服务器。

如果旧的辅助服务器 (Srvr2) 无法连接至另一个服务器 (Srvr1), 那么这两个服务器都将切换为 `SECONDARY ALONE` 状态。(请注意, 即使 `AutoPrimaryAlone` 配置参数设置为 `yes`, 新的主服务器也将切换到 `SECONDARY ALONE` 状态而不是 `PRIMARY ALONE` 状态。)

将辅助服务器切换到主服务器

执行 `hotstandby switch primary` 命令时, 它将启动一个切换状态的进程。

如果成功启动了切换进程, 那么将显示以下消息: 已启动将角色切换为主服务器的进程。

在切换期间, 所有活动的写入事务都将异常终止。可以使用 `hotstandby status switch` 命令来监视切换状态。有关详细信息, 请阅读第 75 页的『验证切换』。

如果您在执行 `SWITCH` 命令之后发出 `COMMIT` 命令, 那么 `COMMIT` 将失败并产生错误: "复制的事务已异常终止"。

所有事务在切换期间都终止。但是请注意, 诸如“HSB switch”命令的 `ADMIN COMMAND` (管理命令) 并不是用于处理事务的命令, 无法回滚。

注:

如果尚未打开一个新事务，那么管理命令将强制启动一个新事务。为了避免留下打开的事务，或者让事务的开始时间不同于您期望的时间，那么您可能需要在执行管理命令之后执行 `COMMIT WORK`。

如果发生了配置错误并导致两个服务器的状态都为 `PRIMARY`（例如，都为 `PRIMARY ALONE` 状态），那么可以使用 `hotstandby switch secondary` 命令将其中一个服务器切换回 `SECONDARY` 状态。如果两个服务器具有相同的数据，那么在两个服务器可以继续执行正常操作。但是，如果两个服务器没有相同的数据，那么主服务器将拒绝来自辅助服务器的连接操作并且发出以下消息：14525：未正确地对 HotStandby 数据库进行同步。

未启动 HotStandby 复制。在这种情况下，需要在辅助服务器中完全复制主数据库。您首先需要决定哪个数据库是正确的。请注意，如果发生了 14525 错误，那么数据库状态不会更改；两个服务器都将处于在发出此命令之前的状态。

将主服务器切换到辅助服务器

可以通过发出以下命令将主服务器切换为 `SECONDARY` 状态：ADMIN COMMAND 'hotstandby switch secondary';

如果两个已经切换了状态，而您希望将它们切换回原来的状态，那么此命令特别有用。例如，当新的辅助服务器恢复工作时，可以将它的状态切换回主服务器，并将新的主服务器切换回辅助服务器。

执行 `hotstandby switch secondary` 时，如果服务器之间尚未互相连接，那么旧的主服务器将尝试连接至旧的辅助服务器。

如果两个服务器之间建立了连接，那么它们将切换状态。换句话说，旧的主服务器将成为新的辅助服务器，而旧的辅助服务器将成为新的主服务器。

执行 `hotstandby switch secondary` 命令时，它将启动一个切换状态的进程。如果成功启动了切换进程，那么将显示以下消息：已启动将角色切换为辅助服务器的进程。

可以检查任何 HotStandby 服务器的切换状态，以验证是否成功执行了切换。有关详细信息，请阅读第 87 页的『显示切换状态信息』。

验证切换

可以在主服务器或辅助服务器中检查切换进程的状态。

发出以下命令：ADMIN COMMAND 'hotstandby status switch';

此命令将显示一条状态消息，让您知道两个服务器之间是从未进行切换、切换成功、仍在进行切换还是切换失败。

执行故障转移

通过在辅助服务器中执行以下命令来执行故障转移：ADMIN COMMAND 'hotstandby set primary alone';

处理了先前从主服务器中接收到的所有暂挂事务之后，服务器将进入新的状态。这将保证不会丢失事务，并且数据库状态反映主服务器在发生故障之前的状态。但是，如果使用的安全级别为 1-safe，那么某些事务在故障转移期间可能会丢失。

运行新的处于 PRIMARY ALONE 状态的主服务器

尽管与辅助服务器的连接已断开，但是此状态允许您运行主服务器并连续更新事务日志。在辅助服务器恢复运行之后，处于 PRIMARY ALONE 状态的服务器可以继续将事务发送至辅助服务器。

可以使用以下三种方法将服务器设置为 PRIMARY ALONE 状态：

- 通过发出以下命令：

```
ADMIN COMMAND 'hotstandby set primary
alone';
```

- 通过在主服务器或辅助服务器中执行受控制的断开连接：

```
ADMIN COMMAND 'hotstandby disconnect';
```

请注意，如果您通过在辅助服务器上执行

```
ADMIN COMMAND 'shutdown';
```

来执行受控关闭，那么辅助服务器在关闭之前将隐式断开连接，而主服务器将安全地切换到 PRIMARY ALONE 状态。

- 通过将 solid.ini 配置文件的 [HotStandby] 节中的 **AutoPrimaryAlone** 配置参数设置为“yes”，来将服务器状态缺省设置为 PRIMARY ALONE 状态。

如果 PRIMARY ALONE 状态是缺省状态，那么在与辅助服务器断开连接时服务器将自动进入 PRIMARY ALONE 状态。否则，在服务器失败之后，除非 HAC、管理员或者看守程序发出了 ADMIN COMMAND 'hotstandby set primary alone' 命令，否则服务器将保持 PRIMARY UNCERTAIN 状态。缺省情况下，solid.ini 文件的 [HotStandby] 节中的 **AutoPrimaryAlone** 参数设置为“no”，它表示在辅助服务器失败的情况下，原来在 PRIMARY ACTIVE 状态下运行的主服务器将自动切换到 PRIMARY UNCERTAIN 状态。

结果

在发生下列其中一种情况之前，服务器将保持 PRIMARY ALONE 状态：

- 成功地连接至辅助服务器。
- 服务器用尽了用于保存事务日志的磁盘空间。
- 达到了日志大小限制 (**MaxLogSize**)。
- 另一个命令将服务器切换到另一种状态（例如，STANDALONE）。
- 主服务器已关闭。

注意:

用户应注意，不要在执行关闭主服务器的同时又命令辅助服务器切换到 **PRIMARY ALONE** 状态。这两项操作是互相冲突的，反而可能会导致辅助服务器进入 **SECONDARY ALONE** 状态。在实际的操作中几乎不会发生这种情况。但是，在测试系统时，用户可能想要通过关闭服务器来模拟主服务器故障。不应当这样做，不能用关闭来替代故障。关闭是一项复杂的、同时涉及到主服务器和辅助服务器的分布式操作。不要执行关闭的另一个原因是，在关闭之后作为新的辅助服务器启动的主服务器，将无法与新的主服务器进行同步复制。如果确实需要关闭主服务器，那么应遵循正确的顺序:

1. 执行切换
2. 关闭新的辅助服务器
3. 新的主服务器将自动切换到 **PRIMARY ALONE** 状态

使辅助服务器恢复联机状态

要使辅助服务器恢复联机状态，请将主服务器与辅助服务器连接起来。有关详细信息，请阅读第 86 页的『连接 HotStandby 服务器』。

一旦您使辅助节点恢复联机状态，它可能就需要进行同步复制。主服务器中的更改已经累积了一段时间。主服务器被设置为 **PRIMARY ALONE** 状态，主服务器将事务和数据写入事务日志。

在辅助服务器再次与主服务器建立连接之后，就会将主服务器的暂挂更改从事务日志写入辅助服务器中以保持同步。将更改写入辅助服务器之后，辅助服务器将处于 **SECONDARY ALONE** 状态，主服务器将处于 **PRIMARY ALONE** 状态。（如果您发出 `ADMIN COMMAND 'hsb status connect'` 命令，那么将产生一条消息告诉您服务器之间是否执行了同步复制。）

注：如果先前已使用 `hotstandby set standalone` 命令将主服务器设置为 **STANDALONE** 状态，那么必须将整个数据库从主服务器复制到辅助服务器之后，才能使辅助服务器进入 **SECONDARY ACTIVE** 状态。请阅读第 78 页的『使主服务器与辅助服务器同步』。

结果

在辅助服务器成功处理了这些暂挂更改之后，主服务器和辅助服务器的状态将分别自动更改为 **PRIMARY ACTIVE** 和 **SECONDARY ACTIVE** 状态。

停止 HotStandby 操作

有时候，可能需要临时停止主服务器中的 HotStandby 操作 - 例如，如果您要使辅助服务器停止运行以对它进行升级，而主服务器中没有足够的磁盘空间用来存储辅助服务器停止运行期间将累积的事务日志。

关于此任务

（有关更多详细信息，请参阅第 36 页的『用尽用于事务日志的空间』。）

要在主服务器中停止 HotStandby:

1. 如果服务器之间当前已建立连接，那么请将它们断开连接。
2. 将使用下面这一系列命令将主服务器设置为 **STANDALONE** 状态:

```
ADMIN COMMAND 'hotstandby disconnect'; -- 如果服务器之间已建立连接
ADMIN COMMAND 'hotstandby set standalone';
```

这允许主服务器继续运行，尽管它不是 HotStandby 服务器。

注:

一旦您停止存储事务日志以发送至辅助服务器，就不再能够仅仅通过再次连接主服务器和辅助服务器就在它们之间进行同步复制。而且当您继续执行 HotStandby 操作时，将需要以手动方式使两个服务器同步。有关详细信息，请阅读下面一节。

下一步做什么

如果您想永远都不再将此服务器用作 HotStandby 服务器，那么请参阅第 92 页的『将 HotStandby 服务器更改为非 HotStandby 服务器』。

使主服务器与辅助服务器同步

为了使服务器之间开始进行 HSB 复制，两个服务器的数据库必须完全相同。换句话说，辅助数据库必须是主数据库的精确副本。使一个 HotStandby 系统的数据库完全相同的过程称为 HotStandby 同步。

在下列情况下需要使主服务器与辅助服务器同步：

- 辅助服务器是新的，但是没有主服务器的数据库的副本而无法开始运行。
- 辅助服务器暂时未运行，并且其数据副本不是最新的。
- 主服务器和辅助服务器同时在 Primary Alone 状态下运行，因此具有相冲突的数据。
- 辅助服务器的磁盘驱动器已崩溃或者文件被破坏，必须更换。

可以通过以下两种主要方法使两个服务器上的数据同步：*同步复制*和*完整复制*。

同步复制

当且仅当主服务器中存储了服务器之间断开连接期间辅助服务器“已丢失”的所有事务的副本时，才能使用同步复制。如果主服务器中存储了所有这些事务，那么当它重新连接至辅助服务器时，它会自动将这些事务转发至辅助服务器，以便辅助服务器可以与它进行同步复制。

仅当 solidDB 服务器处于 PRIMARY ALONE 状态时（而不能是处于 STANDALONE 状态或者正在作为非 HotStandby 服务器运行），它才能存储事务以转发至辅助服务器。因此，如果主服务器自从上次与辅助服务器建立连接以来一直处于 STANDALONE 状态或者作为一个非 HotStandby 服务器运行，那么它就不具有所有事务，因此无法执行同步复制。这种情况下，您将需要执行完全复制（稍后描述）。

没有显式的“catchup”命令。当您使用

```
ADMIN COMMAND 'hotstandby connect';
```

连接服务器时，它们将自动尝试执行同步复制。

在主服务器与辅助服务器建立连接之后，它们将自动进行检查以了解主服务器的事务日志中是否有要发送至辅助服务器的数据。如果存在数据，那么服务器之间将自动尝试进行同步复制。

在同步复制过程中，主服务器和辅助服务器将分别保持 PRIMARY ALONE 和 SECONDARY ALONE 状态。客户机可以继续提交查询和落实事务。同步复制过程对于客户机应用程序是透明的。

如果服务器认为即使在将事务从主服务器复制到辅助服务器之后主数据库与辅助数据库也不完全相同，那么将产生一条错误消息。

如果同步复制失败（或者您提前知道将不会进行同步复制，因为主服务器处于 STANDBY 状态），那么您将需要执行完全复制。

仅当辅助服务器在某个时间已经在 SECONDARY ACTIVE 状态下运行时才会进行同步复制。如果您具有新的辅助服务器，那么即使主服务器先前是在 PRIMARY ALONE 状态下运行并且已经存储了自从它启动以来的所有事务，您也需要执行完全复制，以便为辅助服务器提供数据库的初始副本。

第 77 页的『使辅助服务器恢复联机状态』中提供了有关同步复制过程的更多信息。

完全复制

顾名思义，完全复制就是将主服务器中的所有数据复制到辅助服务器中。这是通过复制数据库文件本身来实现的。

完全复制用于下列情况：

- 辅助服务器是新的，并且要获取主服务器的数据库的初始副本。
- 主服务器在不处于 PRIMARY ALONE 状态的情况下写入了事务，因此无法进行同步复制。
- 辅助服务器的数据库被破坏或者丢失。
- 辅助服务器是一个无盘服务器并且发生了故障。当发生故障的无盘辅助服务器再次启动时，它将要求使用 `hotstandby netcopy` 命令来完整复制数据库。与基于磁盘的辅助服务器不同，无盘辅助服务器无法读取事务日志并应用在它无法运行期间进行的更改。
- 主服务器具有进行同步复制所需的所有数据，但是进行同步复制可能比只是复制当前数据文件要花更长时间。

注意：

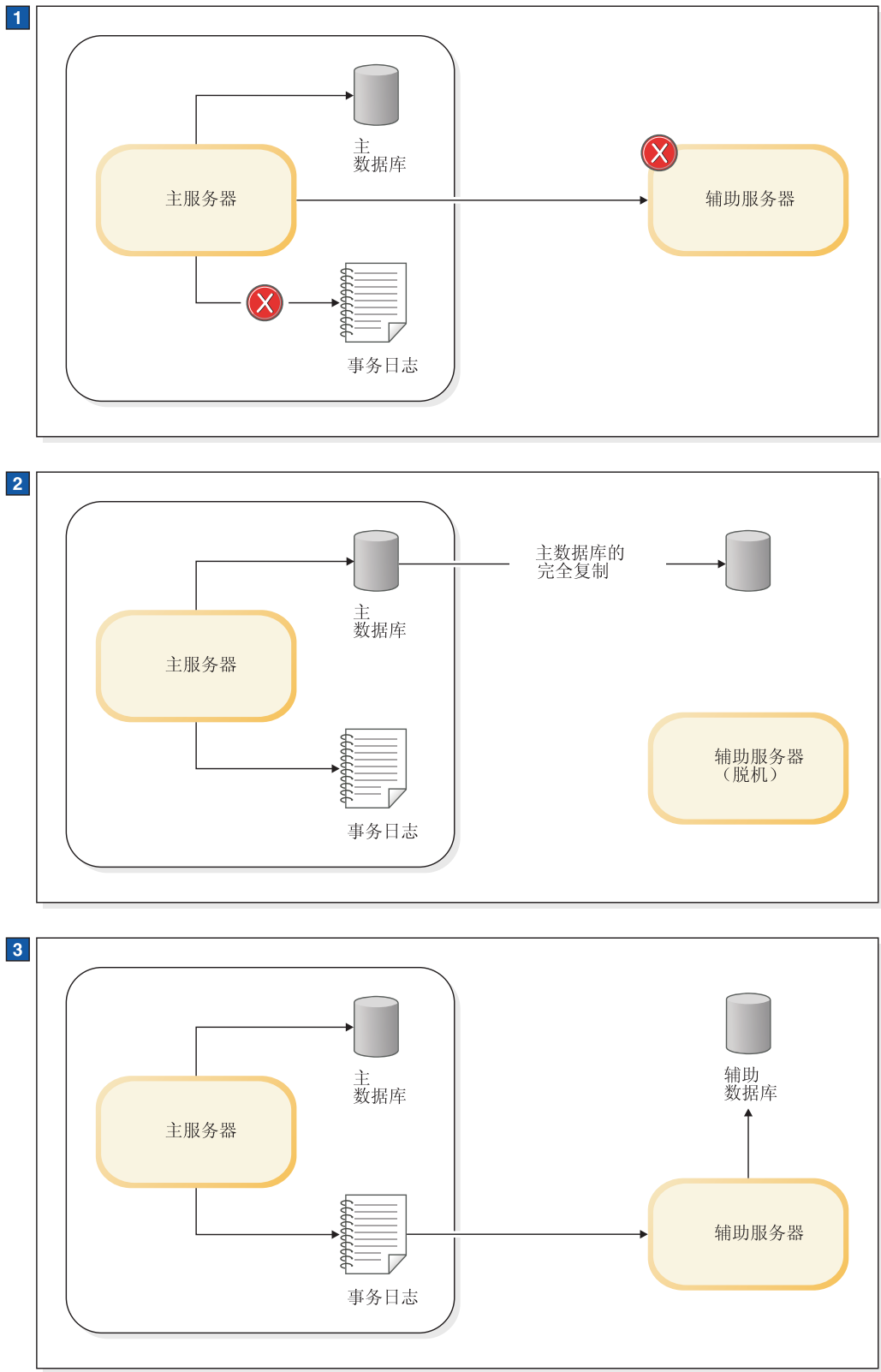
如果辅助服务器中具有旧的数据库文件，那么完全复制将覆盖这些旧文件。如果辅助服务器上的文件因任何原因而包含不是来源于主服务器的数据（例如，如果两个服务器同时都在 PRIMARY ALONE 状态下运行），那么此数据将丢失。

可以使用两个 HotStandby 命令来执行完全复制，即，它们将数据库文件从主服务器复制到辅助服务器。可以使用下列任一命令：

```
ADMIN COMMAND 'hotstandby netcopy'; ADMIN COMMAND 'hotstandby copy  
[<directory_name>]';
```

`netcopy` 操作通过网络将数据库复制到正在运行并且可以通过网络接收文件的辅助服务器。`copy` 操作将数据库文件复制到对于主服务器可视的指定磁盘驱动器目录。在执行 `copy` 操作期间，辅助服务器一定不能在运行。通常，使用 `netcopy` 命令比使用 `copy` 命令更合适。因此，大多数示例都将以 `netcopy` 为例，而不以 `copy` 为例。

第 85 页的『将数据库文件从主服务器复制到指定目录』和第 82 页的『通过网络将主数据库复制到辅助服务器』中描述了 `copy` 和 `netcopy` 命令。



1. 辅助服务器已停止很长时间，因此主服务器停止使用事务日志来存储用于辅助服务器的数据。日志仍然用于进行本地恢复。
2. 已将主数据库复制到辅助服务器节点，并且主服务器开始写入事务日志。
3. 复制了数据库，并且主服务器将对事务日志文件所作的更改发送至辅助服务器。

图 14. 手动完全复制过程

注:

上图非常简单地说明了事务日志的用法。在此图的第一部分中，当主服务器与辅助服务器未建立连接时，主服务器实际上会继续将数据写入事务日志，但是仅保留足够的数据库以执行恢复，但是不足以允许辅助服务器同步复制自从断开连接以来进行的所有更改。

验证复制

要验证 copy 或 netcopy 操作的状态，请在主服务器中发出以下命令:

```
ADMIN COMMAND 'hotstandby status copy';
```

请注意，即使执行的操作是 netcopy，也要使用关键字“copy”（而不是“netcopy”）。此命令将显示一条状态消息，它通过错误代码和错误消息让您知道 copy 操作是已成功完成、仍在执行还是已失败。

使用看守程序使服务器同步

看守程序还可以使用允许您已手动方式使服务器同步的命令来自动使服务器同步。

如果执行同步复制就足够了，那么看守程序只需要监视辅助服务器以了解它何时启动，然后执行此命令将主服务器连接至辅助服务器。如果需要执行完全复制，那么看守程序可以指示主服务器执行 netcopy（或者 copy）操作。请记住，完全复制将重写辅助服务器上的所有数据。

通过网络将主数据库复制到辅助服务器

要将主服务器中的数据库文件的副本发送至辅助服务器，请使用 netcopy 命令。辅助服务器必须已经在运行。

开始之前

要点: 要执行此命令，主服务器必须处于 PRIMARY ALONE 状态。

关于此任务

在以下两种主要情况下，应使用 netcopy 命令来为辅助服务器创建数据库的副本:

- 为以前从未有过的全新辅助服务器创建数据库。（将数据库复制到一个无盘辅助服务器时也使用此方法，这是因为发生故障之后，此服务器将失去它的整个数据库，必须将它看作一个全新的辅助服务器。）
- 替换现有的辅助数据库时（例如，已经被破坏的辅助数据库）

用来执行 netcopy 的命令是:

```
ADMIN COMMAND 'hotstandby netcopy';
```

当主服务器执行 netcopy 时，它将使用在 solid.ini 的 [HotStandby] 节中指定的连接字符串。

有关用于定义连接字符串的 **Connect** 参数的详细信息，请参阅第 43 页的『定义主服务器和辅助服务器 HotStandby 配置』。

当您执行 hotstandby netcopy 命令时，它将在执行数据库检查点之后才发送主数据库的副本。

在执行 netcopy 期间，主服务器将继续接受事务和连接（但是，将拒绝用于更改服务器状态的任何 ADMIN COMMAND。）辅助服务器将不再继续接受事务和连接。当开始执行 netcopy 时，如果辅助服务器有任何已打开的连接或事务，那么它将回滚这些已打开的事务并与它的客户机断开连接，然后再开始接收 netcopy。虽然辅助服务器将接收 netcopy，但是它将仅与主服务器通信。

成功完成 netcopy 之后，辅助服务器的状态将更改为 SECONDARY ALONE（如果它先前不是处于此状态）。

在执行 netcopy 操作期间，主服务器将保持处于 PRIMARY ALONE 状态。在完成 netcopy 之后，主服务器将继续保持同一状态。在可以继续执行完全热备用操作之前，必须连接主服务器和辅助服务器，这会将主服务器设置为 PRIMARY ACTIVE 状态。有关连接辅助服务器与主服务器的信息，请参阅第 86 页的『连接 HotStandby 服务器』。

为辅助服务器创建新的数据库

通常，当您启动 solidDB 服务器时，如果还没有数据库，那么它将询问您是否要创建新的数据库。但是，如果此服务器是一个辅助服务器，那么它应当使用主服务器数据库的副本而不是创建它自己的数据库。因此，当您启动一个没有现成数据库的辅助服务器时，必须为它提供一个命令行参数，以告诉它等待接收来自主服务器的数据库副本。此命令行参数为 **-x backupserver**。例如，应使用以下命令来启动辅助服务器：

```
solid -x backupserver
```

“-x”与“backupserver”之间也可以不保留空格。以下命令与上述命令具有相同作用：

```
solid -xbackupserver
```

-x backupserver 命令行参数让服务器进入“netcopy 侦听方式”（也称为“备份侦听方式”）。在此方式下，辅助服务器唯一可以执行的操作就是接收来自主服务器的数据库副本。辅助服务器将不会对任何其他命令作出响应，实际上它甚至不会接受来自客户机程序（例如，solsql、应用程序或者看守程序）的连接请求。

如果已经存在辅助数据库，那么可以采用正常方式来启动服务器，这将导致服务器处于 SECONDARY ALONE 状态。

一旦使用 **-x backupserver** 启动了辅助服务器，或者辅助服务器处于 SECONDARY ALONE 状态，就可以在主服务器上执行 netcopy 命令。

首先请确保主服务器处于 PRIMARY ALONE 状态，然后主服务器上发出以下命令：

```
ADMIN COMMAND 'hsb netcopy';
```

在主服务器上，hotstandby netcopy 命令使用由 solid.ini 配置文件中的 **connect** 参数定义的连接字符串来连接至辅助服务器。一旦连接之后，它就会通过网络链路来复制数据库文件。在 netcopy 侦听方式下，辅助服务器仅在接收到新的数据库副本（通过在主服务器中执行 hotstandby netcopy 命令）之后才会尝试打开辅助数据库。

按照以下过程来创建辅助数据库副本：

1. 确保您已经配置了 solid.ini 文件，以使它对于 HotStandby 配置有效。有关用于定义连接字符串的 **Connect** 参数的详细信息，请参阅第 43 页的『定义主服务器和辅助服务器 HotStandby 配置』。

此连接字符串将用来从主服务器连接至辅助服务器以及通过网络来复制数据库文件。

2. 启动主服务器。
3. 在 `netcopy` 侦听方式下通过执行以下命令来启动辅助服务器:

```
solid -x backupserver
```

或者, 启动具有现成数据库的辅助服务器。

4. 如果主服务器尚未处于 `PRIMARY ALONE` 状态, 那么请将它设置为此状态:
`ADMIN COMMAND 'hotstandby set primary alone';`
5. 在主服务器中发出以下命令:
`ADMIN COMMAND 'hotstandby netcopy';`
6. 完成 `netcopy` 命令之后, 可以通过发出以下命令来连接两个服务器并启动 (或者恢复) 完整热备用操作:
`ADMIN COMMAND 'hotstandby connect';`

当辅助服务器通过网络链路接收到数据库的新副本时, 它将打开辅助数据库。在使用 `hsb connect` 命令连接服务器之后, 辅助服务器将在正常的 `SECONDARY ACTIVE` 状态下运行, 并且可以接受来自主服务器的用户事务。

如果使用了 `HAC`, 那么将主服务器的数据库复制到辅助服务器的过程为如下所示:

1. 确保两个服务器具有正确的 `Connect` 参数。有关用于定义连接字符串的 **Connect** 参数的详细信息, 请参阅第 43 页的『定义主服务器和辅助服务器 HotStandby 配置』。
2. 确保主节点中的 `HAC` 在 `solidhac.ini` 中设置了 **PreferredPrimary=Yes** 参数值, 而辅助节点中的 `HAC` 未设置此参数值。有关配置 `HAC` 的更多信息, 请参阅第 47 页的『配置 HA 控制器和 HA 管理器』。
3. 启动 `HAC` 实例, 或者将 `HAC` 设置为“自动”方式 (后一项操作可选)。

注:

如果已将 `netcopy` 发送至处于 `SECONDARY ALONE` 状态的服务器, 那么将使用所复制的数据库来覆盖现有数据库。如果需要重新使数据库同步或者修复已被破坏的辅助数据库, 那么使用此选项就很容易完成任务。

替换辅助服务器上的现有数据库

尽管 `netcopy` 主要用来将数据库发送至一个以前从未具有数据库的辅助服务器, 但是也可以在其他情况下使用 `netcopy`。例如, 如果辅助服务器的数据库副本已被破坏 (例如, 由于磁盘驱动器崩溃而被破坏), 那么可以使用 `netcopy` 命令将主服务器的数据库副本发送至辅助服务器。

开始之前

如果您要替换现有数据库, 那么辅助服务器不需要处于“`netcopy` 侦听方式”; 换句话说, 您不需要使用 `-x backupserver` 来启动辅助服务器。

关于此任务

1. 请确保主服务器处于 `PRIMARY ALONE` 状态, 并且辅助服务器处于 `SECONDARY ALONE` 状态。

2. 对主服务器发出以下命令:

```
ADMIN COMMAND 'hotstandby netcopy';
```

请注意, 完成 netcopy 之后, 主服务器仍将处于 PRIMARY ALONE 状态, 而辅助服务器将自动进入 SECONDARY ALONE 状态 (如果它尚未处于此状态)。

如果您在辅助服务器处于 SECONDARY ALONE 状态时执行 netcopy, 并且所有客户机都已连接至辅助服务器 (以执行只读查询), 那么辅助服务器将回滚所有打开的事务并断开所有客户机连接。一旦完成了 netcopy, 辅助服务器仍将保持 SECONDARY ALONE 状态。

3. 服务器之间不会自动建立连接; 您仍然需要发出以下命令来建立连接:

```
ADMIN COMMAND 'hotstandby connect';
```

验证 netcopy 状态

当您启动 netcopy 命令时, 它将在后台异步运行。当完成 netcopy 时, 服务器不会显示一条消息。

关于此任务

实际上, 即使由于发生诸如网络错误之类的问题而导致 netcopy 失败, 服务器也不会显示一条消息。要了解是否成功完成了 netcopy, 您始终应验证 netcopy 的状态:

在主服务器中发出以下命令:

```
ADMIN COMMAND 'hotstandby status copy';
```

注:

此命令使用关键字“copy”而不是“netcopy”。使用同一个命令来执行 copy 和 netcopy 操作。

此命令将显示一条状态消息, 它通过错误代码和错误消息让您知道 netcopy 是已成功完成、仍在执行还是已失败。

将数据库文件从主服务器复制到指定目录

如果辅助服务器将用于存放数据库的目录对于主服务器可视, 那么可以使用 hotstandby copy 命令将此数据库从主服务器的目录复制到辅助服务器的目录。

开始之前

仅当主服务器和辅助服务器都能够查看某些磁盘驱动器, 从而都能够读写某些目录时, 才能执行此任务。

注意:

在使用 hotstandby copy 命令之前, 请务必关闭辅助服务器。当主服务器正在写入数据库文件时, 辅助服务器不能尝试访问此文件。

注: 当您发出 hotstandby copy 命令时, 主服务器应处于 PRIMARY ALONE 状态, 并且在执行此命令期间和执行之后, 主服务器仍将保持此状态。

关于此任务

hotstandby copy 命令与 hotstandby netcopy 命令的一个重要区别是：仅当辅助服务器正在运行时才能使用 netcopy 命令，而仅当辅助服务器未运行时才应使用 copy 命令。这两种数据库复制方法在性能方面没有重大区别。

1. 要使用 hotstandby copy 命令来复制文件，请在主服务器中发出以下命令：

```
ADMIN COMMAND 'hotstandby copy[directory_name]';
```

其中 *directory_name* 是您要将文件复制到的目录的名称。目录名的格式取决于操作系统。目录名是可选的。如果您未指定目录名，那么服务器将使用由 solid.ini 配置文件中的 **CopyDirectory** 参数指定的值。

当您执行 hotstandby copy 命令时，它将为数据库创建一个检查点，然后在生成主数据库的副本之后再将此副本发送至辅助服务器。

由于服务器处于 PRIMARY ALONE 状态，因此在执行 copy 命令期间主服务器通常继续进行事务处理，并且主服务器会将事务存储在事务日志中，以便稍后可以将它们转发至辅助服务器。

2. 执行复制操作之后，辅助服务器仍然处于关闭状态。必须启动辅助服务器，然后发出 hotstandby connect 命令才能将主服务器与辅助服务器连接起来。

通过使用管理命令 hotstandby connect 将主数据库与辅助服务器连接起来之后，主服务器和辅助服务器将自动执行“同步复制”以使辅助服务器中的信息也是最新的。

启动辅助服务器并执行同步复制

完成复制之后，必须使用新复制的数据库来启动辅助服务器。

1. 以正常方式启动服务器，即，通过在操作系统提示符处发出“solid”命令来启动。
2. 重新启动辅助服务器之后，在主服务器中使用 hotstandby connect 命令将主服务器连接至辅助服务器。

```
ADMIN COMMAND 'hotstandby connect';
```

在『连接 HotStandby 服务器』中更详细地讨论了 hotstandby connect 命令。

结果

在主服务器连接至辅助服务器之后，主服务器和辅助服务器将自动开始执行同步复制。这意味着主服务器将使辅助数据库也变成最新的，其步骤如下：将主服务器的事务日志复制到辅助服务器中，然后辅助服务器前滚事务日志并更新其数据库副本。

连接 HotStandby 服务器

关于此任务

主服务器用来连接至辅助服务器的连接字符串是使用 solid.ini 配置文件的 [hotstandby] 节中的 **Connect** 参数指定的。

可以在主节点和辅助节点中通过发出以下命令来查看当前的连接设置：

```
ADMIN COMMAND 'hotstandby cominfo';
```

如果主服务器与辅助服务器之间的连接已断开或者尚未建立，那么您需要在主节点或辅助节点中发出以下命令：

```
ADMIN COMMAND 'hotstandby connect';
```

例如，在执行 `netcopy` 之后，您通常将连接服务器。

由于 HotStandby 服务器中没有自动连接机制，因此，当服务器之间的连接断开之后，应当让高可用性控制应用程序执行此命令。

发出此命令之后，如果主服务器与辅助服务器之间的连接成功，那么将显示一条确认消息。请注意，如果主服务器与辅助服务器已建立连接，但是在辅助服务器中尚未完全复制事务日志，那么您将从主服务器中接收到以下消息：已启动用于连接服务器的进程。

如果当您执行此命令时主服务器的状态为 PRIMARY UNCERTAIN 或 PRIMARY ALONE 并且连接成功，那么主服务器的状态将更改为 PRIMARY ACTIVE。如果连接不成功，那么状态仍然为 PRIMARY UNCERTAIN 或 PRIMARY ALONE。

下一步做什么

有关查询主服务器和辅助服务器中的连接状态的详细信息，请阅读第 88 页的『显示连接状态信息』。有关将应用程序重新连接至主服务器的详细信息，请阅读第 65 页的『从应用程序重新连接至主服务器』。

检查 HotStandby 状态

本节描述您可以从主服务器和辅助服务器中请求的 HotStandby 状态信息。

要检查状态，请在主服务器或辅助服务器中发出以下命令：

```
ADMIN COMMAND 'hotstandby status option';
```

其中 *option* 可以是下列其中一项：

选项	描述
catchup	指示服务器是否正在执行同步复制。同步复制是在主服务器连接至辅助服务器之后执行的。在同步复制期间，主服务器将发送累积的事务日志，以便辅助服务器可以应用更改。此选项的值可以为“ACTIVE”和“NOT ACTIVE”。
connect	指出上次尝试连接服务器是否成功。
copy	指出上次尝试执行 <code>copy/netcopy</code> 是否成功。
switch	指出上次尝试将服务器切换到 PRIMARY ACTIVE 或 SECONDARY ACTIVE 状态是否成功。

示例

```
ADMIN COMMAND 'hotstandby status catchup';
```

显示切换状态信息

您可能需要验证两个 HotStandby 服务器之间是否发生了状态切换。

关于此任务

要检查 HotStandby 切换状态信息:

在主服务器或辅助服务器中发出以下命令:

```
ADMIN COMMAND 'hotstandby status switch';
```

- 当两个服务器之间先前未发生切换时, 将显示以下消息: 先前未发生服务器切换。
- 当切换进程仍然处于活动状态时, 将显示以下消息: 活动。
- 当成功完成了最近的切换进程时, 将显示以下消息: 成功。
- 当最新尝试切换失败时, 将显示以下消息: 错误 *number*, 其中 *number* 标识切换期间所发生错误的类型。

显示连接状态信息

可以查询主服务器与辅助服务器之间的连接状态信息。此功能等价于 SQL 函数 HOTSTANDBY_CONNECTSTATUS, 可以在应用程序代码中使用此函数。

要检查连接状态, 在主服务器或辅助服务器中发出以下命令:

```
ADMIN COMMAND 'hotstandby status connect';
```

返回值可能是:

表 16. 连接状态返回值

错误代码	文本	描述
0	已连接	连接处于活动状态。从主服务器和辅助服务器中返回。
14007	正在连接	主服务器正在连接至辅助服务器。从主服务器和辅助服务器中返回。
14008	同步复制	主服务器已连接至辅助服务器, 但是尚未完全复制事务日志。从主服务器和辅助服务器中返回。
14010	正在断开连接	服务器正在断开连接。
14537	已断开连接	连接已断开。从主服务器和辅助服务器中返回。

显示通信信息

可以查询用来连接至其他服务器的通信信息。这是 solidDB 配置文件 (solid.ini) 的 [HotStandby] 节中的 **Connect** 参数设置的值。可以在客户机应用程序中使用此信息来连接至其他服务器。

要显示通信信息, 在主服务器或辅助服务器中发出以下命令:

```
ADMIN COMMAND 'hotstandby cominfo';
```

显示角色的开始时间

有时候，知道服务器何时进入当前角色（即，成为主服务器或辅助服务器）非常重要。

可以使用 ADMIN COMMAND 'info' 的两个相应选项来检索此信息：

```
admin command 'info primarystarttime';
RC TEXT
-- ----
0 2005-06-09 14:22:18
admin command 'info secondarystarttime';
RC TEXT
-- ----
0 2005-06-09 18:24:44
```

所报告的时间就是此角色成为主服务器或辅助服务器的时间。STANDALONE 状态被认为是主服务器角色的一种状态。

具体来说，主服务器的开始时间是在发生下列状态过渡时设置的：

- SECONDARY ALONE => PRIMARY ALONE
- SECONDARY ALONE => STANDALONE
- SECONDARY ACTIVE => PRIMARY ACTIVE

辅助服务器的开始时间是在以下情况下设置的：

- 服务器是在 SECONDARY ALONE 状态下启动的
- OFFLINE（使用 -x backupserver 启动）=> SECONDARY ALONE
- PRIMARY ALONE => SECONDARY ALONE
- STANDALONE => SECONDARY ALONE
- PRIMARY ACTIVE => SECONDARY ACTIVE

如果当前角色与查询发生矛盾，那么查询将返回空字符串。例如，如果角色是“辅助服务器”，并且发出了 'info primarystarttime' 命令，那么此命令将返回一个空字符串。

验证 HotStandby 服务器状态

当管理和维护 HotStandby 时，通常需要检查 HotStandby 服务器的状态。

关于此任务

要检查 HotStandby 服务器的当前状态，请在此服务器中发出以下 HotStandby 命令：

```
ADMIN COMMAND 'hotstandby state';
```

下一步做什么

有关此命令可能返回的状态的描述，请参阅第 11 页的『对服务器状态的描述』。

注：

1.

如果配置了服务器的 solid.ini 配置文件以使此服务器成为 HotStandby 服务器，那么当此服务器启动时，此服务器将在 SECONDARY ALONE 状态下启动。

2.

请注意，前面的表中未列示 OFFLINE 状态；服务器无法返回状态名“OFFLINE”，这是因为当服务器处于 OFFLINE 状态时，您将无法与它建立连接并发出任何查询（例如，ADMIN COMMAND 'hotstandby state'）。

3.

如果在一个未对 HotStandby 配置的服务器上发出了 ADMIN COMMAND 'hotstandby state'，那么将返回以下错误消息：

14527: 这是一个非 HotStandby 服务器

如果使用了 HAC 和 HA 管理器，那么 HA 管理器将显示两个服务器的 HSB 状态。每秒钟将定期查询信息。

服务器状态组合情况

并不是所有服务器状态组合情况都可以实现。例如，如果主服务器处于 PRIMARY ACTIVE 状态，那么辅助服务器只能处于 SECONDARY ACTIVE 状态。

下表列出了当一个 HotStandby 服务器的相关联服务器处于特定状态时，此服务器可以具有的状态。

表 17. 服务器状态

服务器的状态	相关联的服务器可能处于的状态
PRIMARY ACTIVE	SECONDARY ACTIVE
PRIMARY ALONE	PRIMARY ALONE * PRIMARY UNCERTAIN SECONDARY ALONE STANDALONE *
PRIMARY UNCERTAIN	PRIMARY ALONE PRIMARY UNCERTAIN SECONDARY ALONE STANDALONE
SECONDARY ACTIVE	PRIMARY ACTIVE
SECONDARY ALONE	PRIMARY ALONE PRIMARY UNCERTAIN SECONDARY ALONE STANDALONE
STANDALONE	PRIMARY ALONE * PRIMARY UNCERTAIN SECONDARY ALONE STANDALONE *

* 如果一个服务器处于 PRIMARY ALONE 状态或 STANDALONE 状态，那么另一个服务器不应处于 PRIMARY ALONE 或 STANDALONE 状态。这是因为如果单独对两个服务器都进行了更改，那么将无法将两个数据库合并成一个数据库。

选择使哪个服务器成为主服务器

在某些情况下，当您尝试从两个数据库都失败的故障情况进行恢复时，您可能不知道应当使哪个服务器成为主服务器。在服务器之间互相失去联系之前是主服务器的服务器现在并不是一定成为主服务器。

关于此任务

要确定哪个服务器应成为主服务器，可以在每个服务器上使用以下命令来确定：

```
ADMIN COMMAND 'hsb logpos';
```

此函数将返回一个字符串值或二进制值。具有“更大”值的服务器（即，已接受更多事务的服务器）应当成为主服务器。如果两个服务器的 `logpos` 值相等，那么 **PreferredPrimary** 参数将定义本地服务器是否将成为主服务器。

要使用此命令，请遵循下列指示信息：

1. 两个服务器都应已启动且正在运行，并且处于 `SECONDARY ALONE` 状态。
2. 连接至两个服务器。
3. 在每个服务器中执行以下命令：

```
ADMIN COMMAND 'hsb logpos';
```

如果管理命令运行成功，那么将返回错误代码 0、一个字符串和服务器先前的角色。（注意：应用程序会认为字符串是一个不透明的值，它未定义结构。）

4. 对字符串值进行比较。

例如，在 C 语言中，使用 `strcmp()` 函数。应当选择已返回“更大”字符串的服务器作为新的主服务器。如果返回的字符串大小相等，那么 **PreferredPrimary** 参数将定义本地服务器是否将成为主服务器。

5. 通过在将成为主服务器的服务器上执行以下命令来选择主服务器：

```
ADMIN COMMAND 'hsb set primary alone';
```

6. 通过使用以下命令将 `HotStandby` 服务器互相连接起来：

```
ADMIN COMMAND 'hsb connect';
```

7. 如果上述命令成功，那么辅助服务器将与主服务器之间进行同步复制，并且 `HotStandby` 对将再次起作用。如果此命令失败，那么必须通过在主服务器上发出以下命令来单独使节点同步：

```
ADMIN COMMAND 'hsb netcopy';
```

`netcopy` 命令完成之后不会提供返回值。然而，您必须积极观察它。可以通过执行以下命令来完成此任务：

```
ADMIN COMMAND 'hsb status copy';
```

返回值可能是 `ACTIVE`、`SUCCESS` 或 `FAILED`。如果发生故障，那么还会输出故障原因。在完成同步之后，发出以下命令：

```
ADMIN COMMAND 'hsb          connect';
```

`HotStandby` 对将再次起作用。

注意:

此过程并不保证具有更大字符串值的服务器是另一个服务器的超集。在这两个服务器中，仍然有可能每个服务器都接受了对方未接受的事务 - 例如，两个服务器都可能已经在 **PRIMARY ALONE** 状态下运行。要检测这两个服务器都不是对方的超集这种可能性，服务器在执行“**connect**”命令时将比较信息。如果这两个服务器都不是超集，那么 **Connect** 命令将失败并提供适当的错误消息。

将 HotStandby 服务器更改为非 HotStandby 服务器

可以通过编辑 `solid.ini` 文件的 `[HotStandby]` 节将主服务器或辅助服务器更改为一个正常的非 HotStandby 服务器。

1. 除去 **HSBEnabled** 参数（或者将它设置为“no”）。
2. （可选）除去或者剔除 **Connect** 参数。
3. 更改 `solid.ini` 文件中的参数设置之后，必须重新启动服务器才能使更改生效。

示例

如果您想让服务器临时停止充当 HotStandby 服务器，但是您希望它稍后继续充当 HotStandby 服务器，那么可能需要让 `solid.ini` 文件保持不变，而只是将服务器的状态更改为 `STANDALONE`。

6 高可用性控制器在故障情况下的行为

本节描述了可能会发生的故障情况。高可用性控制器（HAC）将隐式处理故障情况。但是，不同的故障或初始化方案（简称管理方案）可由管理员或者任何称为“看守程序”的软件程序来处理。

看守程序是一个独立程序，用于监视主服务器和辅助服务器，并在必要时提供命令来更改这些服务器的状态。建议您使用 HAC，以便您可以确定主服务器或辅助服务器本身失败的时间或者这些服务器之间的通信链路的断开时间。

进行恢复的目的是为了使失败组件重新恢复运行。有时，恢复期间可能又会发生其他故障。这些故障通常会导致系统处于一种可用性受到限制的状态（只有一个服务器已启动），从而期待人员干预。下面是不会自动处理的典型恢复时故障：

- 失败的数据库被破坏，从而无法重新启动此数据库
- 没有足够的可用磁盘空间用来执行同步复制

主数据库失败

情况

节点 1 上的主数据库（处于 PRIMARY ACTIVE 状态）失败。

节点 2 上的辅助数据库（处于 SECONDARY ACTIVE 状态）与节点 1 上的主数据库建立连接时发生故障。

恢复

从主数据库故障进行恢复时，辅助服务器将替换主服务器。将按如下所示自动继续进行恢复：

1. 发生连接故障时，节点 2 上的辅助数据库将自动进入 SECONDARY ALONE 状态。
2. 节点 2 上的辅助数据库上的 HAC 实例断定节点 1 上的主数据库已失败，并将节点 2 上的辅助数据库设置为 PRIMARY ALONE 状态。
3. 在执行上述任务的同时，节点 1 上的主数据库上的 HAC 实例将重新启动主数据库，主数据库将进入 SECONDARY ALONE 状态。
4. 节点 2 上的辅助数据库上的 HAC 实例将启动连接主数据库与辅助数据库的过程。
5. 执行了同步复制。

（可选）连接过程中包括一项 netcopy 操作，它会将节点 2 上的辅助数据库复制到节点 1 上的主数据库。

辅助数据库失败

情况

节点 2 上的辅助数据库（处于 SECONDARY ACTIVE 状态）失败。

节点 1 上的主数据库（处于 PRIMARY ACTIVE 状态）与节点 2 上的辅助数据库建立连接时发生故障。

恢复

从辅助数据库故障进行恢复时，辅助服务器将重新启动。将按如下所示自动继续进行恢复：

1. 发生连接故障时，节点 1 上的主数据库将自动进入 PRIMARY UNCERTAIN 状态；如果启用了 **AutoPrimaryAlone** 参数，那么此数据库将进入 PRIMARY ALONE 状态。
2. 节点 1 上的主数据库上的 HAC 实例断定节点 2 上的辅助数据库已失败。
3. 如果在步骤 1 中已将主数据库设置为 PRIMARY UNCERTAIN 状态，那么 HAC 现在会将它设置为 PRIMARY ALONE 状态。
4. 在执行上述任务的同时，节点 2 上的辅助数据库上的 HAC 实例将重新启动辅助数据库，辅助数据库将进入 SECONDARY ALONE 状态。
5. 节点 1 上的主数据库上的 HAC 实例将启动连接主数据库与辅助数据库的过程。
6. 执行了同步复制。

（可选）连接过程中包括一项 netcopy 操作，它会将主数据库复制到辅助数据库。

主节点失败

情况

主节点（节点 1）失败。

节点 2 上的辅助数据库（处于 SECONDARY ACTIVE 状态）与节点 1 上的主数据库（处于 PRIMARY ACTIVE 状态）建立连接时发生故障。

恢复

从主节点故障进行恢复时，主服务器将重新启动。将按如下所示自动继续进行恢复：

1. 发生连接故障时，辅助数据库将自动进入 SECONDARY ALONE 状态。
2. 节点 2 上的辅助数据库上的 HAC 实例断定节点 1 上的主数据库已失败。
3. 节点 2 上的辅助数据库上的 HAC 实例将辅助数据库设置为 PRIMARY ALONE 状态。
4. 重新启动了主节点（节点 1）。
5. 重新启动了主数据库（节点 1）上的 HAC 实例。
6. 主数据库（节点 1）上的 HAC 实例断定主数据库未运行。
7. 主数据库（节点 1）上的 HAC 实例将重新启动主数据库并将它设置为 SECONDARY ALONE 状态。
8. 节点 2 上的辅助数据库上的 HAC 实例将启动连接主数据库与辅助数据库的过程。
9. 执行了同步复制。

（可选）连接过程中包括一项 netcopy 操作，它会将主数据库复制到辅助数据库。

辅助节点失败

情况

辅助节点（节点 2）失败。

节点 1 上的主数据库（处于 PRIMARY ACTIVE 状态）与节点 2 上的辅助数据库（处于 SECONDARY ACTIVE 状态）建立连接时发生故障。

恢复

从辅助节点故障进行恢复时，辅助服务器将重新启动。将按如下所示自动继续进行恢复：

1. 发生连接故障时，节点 1 上的主数据库将自动进入 PRIMARY UNCERTAIN 状态。
2. 节点 1 上的主数据库上的 HAC 实例断定节点 2 上的辅助数据库已失败。
3. 节点 1 上的主数据库上的 HAC 实例将主数据库设置为 PRIMARY ALONE 状态。
4. 重新启动了辅助节点（节点 2）。
5. 重新启动了辅助数据库（节点 2）上的 HAC 实例。
6. 辅助数据库（节点 2）上的 HAC 实例断定辅助数据库未运行。
7. 辅助数据库（节点 2）上的 HAC 实例将重新启动辅助数据库并将它设置为 SECONDARY ALONE 状态。
8. 节点 1 上的主数据库上的 HAC 实例将启动连接主数据库与辅助数据库的过程。
9. 执行了同步复制。

（可选）连接过程中包括一项 netcopy 操作，它会从主数据库复制到新的辅助数据库。

HotStandby 链路失败

情况

HotStandby 链路失败。

节点 1 上的主数据库（处于 PRIMARY ACTIVE 状态）和节点 2 上的辅助数据库（处于 SECONDARY ACTIVE 状态）互相建立连接时都发生了故障。

恢复

从 HotStandby 链路故障进行恢复时，HAC 实例将对“外部引用实体”（ERE）执行 Ping 操作，以便弄清楚是发生了网络故障还是相对应的服务器已失败。将按如下所示自动继续进行恢复：

1. 发生连接故障时，主数据库和辅助数据库将分别自动进入 PRIMARY UNCERTAIN（节点 1）和 SECONDARY ALONE（节点 2）状态。
2. 两个 HAC 实例直接连接至远程服务器都失败。
3. 两个 HAC 实例都使用操作系统的 Ping 实用程序对 ERE 执行 Ping 操作。
4. 重新启动了节点 2 上的辅助数据库。

5. 如果 Ping 操作失败，那么本地服务器将保持或者设置为 SECONDARY ALONE 状态。
6. 如果 Ping 操作成功，那么成功的 HAC 将尝试连接至远程数据库服务器。
7. 如果连接至远程数据库服务器失败，那么 HAC 将断定它的网络连接是可操作的，并将本地服务器设置为 PRIMARY ALONE 状态。
8. 主数据库上的 HAC 实例将尝试重新与辅助数据库建立连接。
9. 一旦网络变得可操作并且连接成功，那么主数据库和辅助数据库将分别自动进入 PRIMARY ACTIVE 和 SECONDARY ACTIVE 状态。

7 使用 HotStandby 对服务器进行升级

冷迁移和热迁移

迁移过程涉及到更新软件版本。对于一个高可用系统（例如，solidDB HotStandby），可以执行“冷”迁移或“热”迁移。

冷迁移是传统迁移方法。关闭整个系统（两个服务器），然后使用新的软件和配置数据重新启动。

solidDB 高可用性设计允许您对 solidDB 服务器进行升级，并且在对服务器进行升级期间不会使整个系统脱机。一个服务器可以在另一个服务器进行升级期间继续运行。这种迁移称为热迁移。

注意:

尽管整个系统将不会关闭，但是用户或应用程序可能需要与一个服务器断开连接，然后连接至另一个服务器（如果您使用的是基本连接）。如果您使用的是透明连接，那么可以采用对用户和应用程序透明的方式执行热迁移。

在与 HSB 兼容的版本之间进行迁移

当软件版本与 HSB 兼容时，每个服务器可能使用不同的版本，但它们仍然能够互相通信。例如，solidDB 的任何主要版本的所有发行版都与 HSB 兼容。

冷迁移

冷迁移是普通的。您将关闭整个系统，然后使用与关闭之前具有相同角色的新软件重新启动服务器。

热迁移

热迁移的基本过程是:

1. 断开连接并关闭辅助服务器，然后对辅助服务器进行升级。
2. 将旧的辅助服务器设置为主服务器（处于 PRIMARY ALONE 状态）；关闭旧的主服务器，然后对旧的主服务器进行升级。
3. 使旧的主服务器作为新的辅助服务器运行。连接新的主服务器和新的辅助服务器，并在新的辅助服务器与新的主服务器之间进行同步复制。

请注意，此过程将“颠倒”服务器。在这一系列步骤结束之后，原来的主服务器将成为辅助服务器，而原来的辅助服务器将成为新的主服务器。

在与 HSB 不兼容的版本之间进行迁移

当版本之间不兼容时，在系统升级过程中必须遵循某些情况。

具体来说，必须使用一些转换命令行参数。

可以迁移、但是与此版本不能实现 HSB 兼容的版本是：3.1、3.7 和 4.0。不支持从 3.1 之前的版本进行迁移。

下面显示了更详细的逐步完成的过程。

在与 HSB 不兼容的版本之间进行迁移之前的准备步骤

此示例假定应用程序将通过检测每个连接的状态来自动故障转移到新的主服务器。因此，服务器之间的受控切换将不会扰乱应用程序，只不过打开的事务可能会在切换期间异常终止。

1. 如果应用程序尚未设计为自动进行故障转移，那么您可能需要通知用户这些应用程序将断开它们的连接，并且需要重新连接至新的主服务器。
2. 准备系统和软件以进行升级。您可能需要执行下列任务：
 - a. 由于每个 solidDB 服务器在部分升级操作期间将单独运行（尤其是在 PRIMARY ALONE 状态下），因此，应当确保两台计算机都处于“健康”状态，例如，它们都具有足够的可用磁盘空间、可靠的网络连接以及在发生电源故障时可以使用的 UPS。
 - b. 每个服务器至少有一小段时间（当另一个服务器进行升级时）将在 PRIMARY ALONE 状态下运行。当服务器处于 PRIMARY ALONE 状态时，它会将事务存储在事务日志中。必须具有足够的磁盘空间可供日志文件用来存储在另一个服务器进行升级期间产生的所有事务（其中包括另一个服务器在重新启动之后进行同步复制时产生的事务）。
 - c. 确保每台计算机上都具有升级软件的副本或者很容易获得此副本。
3. 注意：

如果您具有看守程序，那么应当暂时关闭此看守程序，以使它发出的命令与您在升级过程中发出的命令不会冲突。

例如，在将主服务器与辅助服务器断开连接之后，在对辅助服务器进行升级之前，您不会希望看守程序尝试重新连接这两个服务器。

冷迁移过程

关于此任务

迁移步骤如下：

1. 将服务器断开连接并关闭它们。
2. 安装新的软件版本。
3. 更新 solid.ini 文件。
4. 使用命令行参数 `-x autoconvert` 启动主服务器，此参数将指示服务器将现有数据库转换为新格式。
5. 将主服务器设置为 PRIMARY ALONE 状态。
6. 执行从主服务器到辅助服务器的“`hsb copy`”或“`hsb netcopy`”。
7. 连接服务器。

热迁移过程

关于此任务

在下列步骤中，S1（“OP”）和 S2（“OS”）表示原来的主服务器和辅助服务器。当您完成此过程时，每个服务器的状态都会更改。

1. S1: 使主服务器与辅助服务器断开连接:

```
ADMIN COMMAND 'hsb set broken';  
ADMIN COMMAND 'hsb status connect ping';
```

2. S2 OS: 关闭服务器 S2（辅助服务器）:

```
ADMIN COMMAND 'shutdown force';
```

3. S1 OP: 如果服务器 S1（主服务器）尚未自动切换到 PRIMARY ALONE 状态，那么让此服务器在此状态下运行。验证服务器 S1（主服务器）是否处于 PRIMARY ALONE 状态。

```
ADMIN COMMAND 'hsb set primary alone';  
ADMIN COMMAND 'hsb state';
```

4. S2 OS: 对服务器 S2（原来的辅助服务器）进行升级

您应当更新 solid.ini 文件中的配置参数以及更新软件。

5. S2 OS: 使用 -x migratehsbg2 命令行开关启动服务器 S2（原来的辅助服务器）:

```
solid -x migratehsbg2
```

此命令行开关有两个作用。它指示服务器接受并转换现有数据库（其作用与 -x autoconvert 参数相同）。另外，它使新的辅助服务器能够使用旧的复制协议与旧的主服务器通信。服务器应进入 SECONDARY ALONE 状态。

6. S1 OP: 检查服务器 S1（主服务器）以确保它仍处于 PRIMARY ALONE 状态:

```
ADMIN COMMAND 'hsb state';
```

7. S1 OP: 执行 hsb connect 命令以将主服务器连接至辅助服务器:

```
ADMIN COMMAND 'hsb connect';
```

请注意，如果这种运行更高版本的服务器，那么不能从辅助服务器进行连接。

此步骤将从以下进程开始：辅助服务器同步复制在辅助服务器关闭期间产生的数据更改。

8. S1 OP: 等到同步复制完成之后才继续执行后续操作。如果同步复制失败，那么您将需要执行下列操作:

- a. 关闭服务器 S2（辅助服务器）。
- b. 在 S1（主服务器）中执行 hsb copy 以将整个数据库复制到服务器 S2 中。

（注意：您必须使用 hsb copy 而不能使用 hsb netcopy，这是因为 hsb netcopy 不能在不同的服务器版本之间进行复制。

- c. 使用旧版本的服务器（S2）来恢复复制。
 - d. 关闭 S2（辅助服务器）。
 - e. 后退到上一步。
9. S1 OP: 在服务器之间已建立连接并进行同步复制之后，执行以下命令:

```
ADMIN COMMAND 'shutdown force';
```

注意:

force 选项将异常终止打开的任何事务。

在服务器 S2 (辅助服务器) 进行同步复制之后, 使它成为新的主服务器。关闭服务器 S1 (先前的主服务器) 以对它进行升级。

10. S2 OS: 如果新的主服务器 S2 (旧的辅助服务器) 尚未自动切换到 PRIMARY ALONE 状态, 那么将此服务器设置为在此状态下运行。验证服务器 S2 是否处于 PRIMARY ALONE 状态。

```
ADMIN COMMAND 'hsb set primary alone';  
ADMIN COMMAND 'hsb state';
```

11. S1 OP: 对服务器 S1 (原来的主服务器) 进行升级。

您应当更新 solid.ini 文件中的配置参数以及更新软件。

12. S1 OP: 在处于 OFFLINE 状态的服务器 S1 上重新启动 solidDB 服务器。

```
solid -x backupserver
```

13. S2 OS: 检查新的主服务器 S2 (旧的辅助服务器) 以确保它仍处于 PRIMARY ALONE 状态:

```
ADMIN COMMAND 'hsb state';
```

14. S2 OS: 通过执行 Netcopy 操作将数据库从新的主服务器 (S2) 复制到新的辅助服务器 (S1) 中:

```
ADMIN COMMAND 'hsb netcopy';
```

15. S2 OS: 验证 netcopy 操作是否成功:

```
ADMIN COMMAND 'hsb status copy';  
ADMIN COMMAND 'hsb connect';
```

hsb connect 命令会在新的主服务器与新的辅助服务器之间建立连接, 并且将启动以下进程: 新的辅助服务器同步复制在它关闭期间产生的数据更改。

如果此步骤失败, 那么使用 hsb copy 命令将整个数据库复制到辅助服务器, 然后从步骤 11 开始继续执行操作。

升级之后

在新的辅助服务器与新的主服务器之间进行同步复制之后, 系统应彻底恢复正常。新的主服务器和新的辅助服务器都将进行升级, 并且都将具有最新数据。您可能需要运行某些测试查询以确保每个对象都正确运行。

1. 测试主服务器和辅助服务器是否都在正确运行。例如, 您可以选择下面这一系列操作:

在主服务器上:

```
ADMIN COMMAND 'hsb state';  
ADMIN COMMAND 'hsb status catchup';
```

发出某些类型的只读查询。

在辅助服务器上:

```
ADMIN COMMAND 'hsb state';  
ADMIN COMMAND 'hsb status catchup';
```

发出某些类型的只读查询。

2. 如果您使用了看守程序，请将它重新启动。

注:

无论您是想对硬件、操作系统还是 solidDB 进行升级，此方法都适用。

附录 A. HotStandby 配置参数

本节讨论 solid.ini 配置文件中的服务器端配置参数。服务器端参数包括 solid.ini 配置文件中的 [Cluster] 节和 [HotStandby] 节中的参数。

而且，本节还讨论了 solidhac.ini 配置文件中的高可用性控制器（HAC）配置参数以及 HAManager.ini 配置文件中的高可用性管理器配置参数。这些参数都是特定于 solidDB HotStandby 的参数。有关其他 solid.ini 参数的讨论，请参阅《solidDB 管理员指南》。有关如何设计参数名和节标题的格式的信息，请参阅《solidDB 管理员指南》，此指南提供了有关这些配置参数的附录。此附录说明了在设计参数名和值的格式时必须遵循的规则。solid.ini 文件的 [HotStandby] 节也遵循相同的规则。有关可以在 solid.ini 文件的 [Watchdog] 节中使用的参数的信息，请参阅『看守程序样本』。请注意，[HotStandby] 节之外的其他节中的某些参数也会影响 HotStandby 功能。在第 43 页的『配置特定于 HotStandby 的参数』和《solidDB 管理员指南》中说明了其他这些参数。

确保主服务器和辅助服务器的参数值协调

本节说明哪些参数在主服务器和辅助服务器上应当相同，哪些参数应当不同。

某些参数在主服务器和辅助服务器上应当相同。这是因为在进行故障转移之后，原来的辅助服务器成为新的主服务器，它应当与旧的主服务器具有相同的行为。请注意，并不是绝对需要使用相同的值；即使您使用不同的值，服务器也不会失败，但是客户机可能会发现服务器具有不同的行为。

不包含在 [HotStandby] 节中、但是与此节间接相关的某些参数在主服务器和辅助服务器也应当相同。例如，**DurabilityLevel** 参数在主服务器和辅助服务器上通常就应当相同。

某些参数在主服务器和辅助服务器上应当不相同，这样才能唯一地标识每个服务器并且服务器之间互相进行通信。

下列 HotStandby 参数在主服务器和辅助服务器上应当相同：

- [HotStandby]
 - **2SafeAckPolicy**
 - **AutoPrimaryAlone**
 - **ConnectTimeout**
 - **HSBEnabled**
 - **PrimaryAlone**（不推荐使用此参数；但是如果使用了此参数，那么就应当相同）
- [IndexFile]
 - **FileSpec** 应当兼容，也就是说 **FileSpec** 参数的数目应当相同，相应的 **FileSpec** 参数的大小应当匹配。
 - **BlockSize**
- [Logging]

- **BlockSize**

下列参数应当不相同:

- [HotStandby]

- **Connect**

下列参数可以相同也可以不相同, 这取决于所在的环境(例如, 计算机上的磁盘驱动器配置):

- [General]

- **BackupDirectory**

- [HotStandby]

- **CopyDirectory**

还有一些非 HSB 参数的设置会影响 HSB 性能。例如, solid.ini 文件的 [Logging] 节中的 **DurabilityLevel** 参数的设置允许您优化 HotStandby 的性能。请参阅第 15 页的『自适应耐久性』以及《solidDB 管理员指南》中对 **DurabilityLevel** 参数的描述。

确定主服务器的设置是否优先于辅助服务器的设置

某些参数在主服务器和辅助服务器上应当相同。如果未将同一个参数的值设置为相同, 那么您可能期望每个服务器使用在该服务器的 solid.ini 文件中定义的值。但是, 并不是必需这样做。

即使对于一些用于控制辅助服务器行为的参数(例如, **2SafeAckPolicy**), 这些参数在主服务器上的值也是用于确定行为的值。总的原则是, 所有安全和耐久性参数都由主服务器控制。例如, 主服务器读取它的 **2SafeAckPolicy** 值, 并将此值发送至辅助服务器供辅助服务器使用。仅当辅助服务器成为主服务器时, 才会使用存储在辅助服务器的 solid.ini 文件中的值。

主服务器中的值优先于辅助服务器中的值的参数包括:

- **HotStandby.SafenessLevel**
- **HotStandby.2SafeAckPolicy**
- **Logging.DurabilityLevel**
- **HotStandby.NetcopyRpcTimeout**

执行 'hsb connect' 命令时, 位于主服务器中的下列参数优先:

- **HotStandby.PingTimeout**
- **HotStandby.PingInterval**

查询 HotStandby 配置参数

可以使用标准参数处理命令来查询 HotStandby 参数的值和属性。

命令为:

```
ADMIN COMMAND '[describe] parameter[section_name[.parameter_name]]';
```

例如:

```
ADMIN COMMAND 'parameter logging.durabilitylevel';
RC TEXT
-- ----
0 Logging DurabilityLevel 3 3 2

ADMIN COMMAND 'parameter hotstandby.MaxLogSize';
RC TEXT
-- ----
0 HotStandby MaxLogSize 10000000 0 0
```

结果行中显示的三个值按从左到右的顺序依次为:

- **当前值** - 动态设置此值, 或者继续使用缺省值或出厂值。
- **缺省值** - 最初是从 `solid.ini` 文件中读取的, 或者继续使用出厂值。
- **出厂值** - 是在产品中提供的。

修改 HotStandby 配置参数

通常, 要更改某个参数的值, 可以通过在 `solid.ini` 配置文件中更改此值, 然后重新启动服务器来完成更改。但是, 还可以使用 `ADMIN COMMAND` 来更改大多数 HotStandby 参数。

关于此任务

命令为:

```
ADMIN COMMAND 'parameter section_name.parameter_name=value [temporary]';
```

注: 还可以使用 `ADMIN COMMAND "hotstandby parameter ..."` 命令来修改 HotStandby 参数, 但是不推荐您使用此命令。其语法如下:

```
ADMIN COMMAND 'hotstandby parameter parameter_name value';
```

访问方式

参数的访问方式指出是否可以动态更改此参数以及更改何时生效。

当使用 `ADMIN COMMAND` 更改了参数的值时, 可能会也可能不会立即应用更改, 并且当服务器下一次启动时可以应用也可以不应用这些更改。如果已将一个参数值写入 `solid.ini` 文件, 那么它将在服务器下一次启动时生效。

访问方式值

访问方式可能是:

- **RO (只读):** 不能更改值; 当前值始终与启动值完全相同。
- **RW:** 可以通过 `ADMIN COMMAND` 来更改值, 并且更改将立即生效。
- **RW/Startup:** 可以通过 `ADMIN COMMAND` 来更改值, 并且更改将在服务器下一次启动时生效。
- **RW/Create:** 可以通过 `ADMIN COMMAND` 来更改值, 并且将在创建了新数据库时应用更改。

保存参数更改

除非使用了 **temporary** 选项，否则，对参数所作的所有更改都将保存在下一个检查点的 `solid.ini` 文件。还可以使用以下命令来快速完成保存：

```
ADMIN COMMAND 'save parameters [file_name]';
```

缺省情况下，此命令将重写缺省 `solid.ini` 文件。通过使用 `file_name` 选项，可以将输出转移到另一个位置。

[Cluster] 节

可以在 `solid.ini` 配置文件的 [Cluster] 节中设置下列配置参数：

表 18. 集群参数

参数名称	描述	出厂值	访问方式
ReadMostlyLoadPercentAtPrimary	转移给主服务器的读取负载所占的百分比。	50	RW/Startup

[HotStandby] 节

可以在 `solid.ini` 配置文件的 [HotStandby] 节中设置下列配置参数：

表 19. HotStandby 参数

参数名称	描述	出厂值	访问方式
1SafeMaxDelay	在 1-Safe 复制中，此参数是指将已落实的事务发送至辅助服务器之前的最大延迟时间（按毫秒计）。	1000	RW
2SafeAckPolicy	<p>此参数指定辅助服务器从主服务器中接收到事务时进行确认的计时。</p> <p>有效值为：</p> <ul style="list-style-type: none">• 1 = 2-safe received。辅助服务器接收到数据时就进行确认。• 2 = 2-safe visible。辅助服务器将在数据“可视”时（即，当辅助服务器执行了事务时）进行确认。• 3 = 2-safe durable。辅助服务器将在它已使数据成为持久数据之后（即，在它已落实数据并且将数据写入磁盘之后）进行确认。 <p>当然，2-safe durable 是最安全的方法，2-safe received 的响应时间最短。但是，2-safe received 方式实际上在大多数情况下都足以保证数据安全，因此，在数据安全性与响应速度之间达到了最佳平衡。</p> <p>仅当服务器正在使用 2-safe 复制时，此参数才适用。</p> <p>注： 尽管此参数可控制辅助服务器的行为，但此参数是在主服务器设置的。将忽略在辅助服务器的 <code>solid.ini</code> 文件中的值。</p> <p>请参阅第 104 页的『确定主服务器的设置是否优先于辅助服务器的设置』和第 103 页的『确保主服务器和辅助服务器的参数值协调』。</p>	1	RW

表 19. HotStandby 参数 (续)

参数名称	描述	出厂值	访问方式
AutoPrimaryAlone	如果此参数设置为 Yes, 那么在与辅助服务器断开连接时服务器将自动进入 PRIMARY ALONE 状态 (而不是 PRIMARY UNCERTAIN 状态)。如果您打算将此参数设置为“yes”, 那么请阅读第 35 页的『网络分区和双重主服务器』中非常重要的警告信息。	No	RW
CatchupSpeedRate	当服务器正在执行同步复制时, 它还会继续处理来自客户机的数据库请求。可以通过 CatchupSpeedRate 参数来提高“对应用程序请求作出响应”的重要性, 同时降低同步复制的优先级; 或者降低“对应用程序请求作出响应”的重要性, 同时提高同步复制的优先级。 使用由链路和辅助服务器吞吐量指定的最大可用速度所占的百分比来表示速率。此数目越大, 表示更注重同步复制, 处理客户机请求的重要性就会降低。允许值为 1 - 99。	70	RW
Connect	Connect 参数指示服务器对中另一个 HotStandby 服务器所在的“地址”。 [HotStandby] 节中连接字符串的格式与 [Com] 节中 Listen 参数的格式相同 (有关更多详细信息, 请参阅《IBM solidDB 管理员指南》)。 如果您在打算用于 HotStandby 的服务器中省略此参数, 那么可以使用 ADMIN COMMAND 来动态设置此参数。在服务器具有连接字符串之前, 服务器只能在不涉及到连接的状态下运行 (例如, PRIMARY ALONE、SECONDARY ALONE 和 STANDALONE 状态)。 除非 HSBEnabled 参数设置为“yes”, 否则将忽略 Connect 参数。	无出厂值。	RW
ConnectTimeout	通过指定连接超时值, 可以设置 HotStandby 连接操作等待与远程机器建立连接的最长时间 (按秒计)。 ConnectTimeout 参数仅与某些管理命令配合使用 (此参数仅在某些平台上有用)。这些管理命令是: <ul style="list-style-type: none"> • hotstandby connect • hotstandby switch primary • hotstandby switch secondary 例如, 要将超时值设置为 30 秒 (30000 毫秒): [HotStandby] ConnectTimeout=30000 另请参阅 PingTimeout 。	0 (无超时) 单位: 1 毫秒	RW

表 19. HotStandby 参数 (续)

参数名称	描述	出厂值	访问方式
CopyDirectory	<p>[HotStandby] 节中的 CopyDirectory 参数定义当用户执行以下命令时所执行的 HotStandby 复制操作的名称和位置:</p> <pre>ADMIN COMMAND 'hotstandby copy';</pre> <p>例如, 此参数看起来可能为如下所示:</p> <pre>[HotStandby] CopyDirectory=C:\solidDB\secondary\dbfiles</pre> <p>如果您为 CopyDirectory 参数提供一个相对路径, 那么此路径是相对于用来保存主服务器的 solid.ini 文件的目录的路径。</p> <p>此参数没有出厂值, 因此, 如果在 solid.ini 文件中未指定目录, 那么必须在 copy 命令中提供目录。</p> <p>请注意, ADMIN COMMAND 'hotstandby netcopy' 是更灵活的解决方案, 建议使用它来复制数据库。</p>	无出厂值	RW
HSBEnabled	<p>如果此参数设置为 yes, 那么服务器可以充当 HotStandby 主服务器或辅助服务器。如果此参数设置为 no, 那么服务器不能充当 HotStandby 服务器。</p> <p>将此参数设置为 Yes 时, 当服务器首次启动时, 会将服务器的缺省初始状态隐式定义为 SECONDARY ALONE。有效值为“yes”和“no”。</p> <p>要使用 HotStandby, 还必须通过在 solid.ini 文件中设置 Connect 参数或者使用 ADMIN COMMAND 来设置此参数的方式来指定此参数。</p>	no	RO
MaxLogSize	基于磁盘的 HSB 日志的最大大小。出厂值: 无限制	0 单位: 1 字节 k=KB m=MB	
MaxMemLogSize	当禁用基于文件的记录 (Logging.LogEnabled=No) 时, 在将内存日志发送至辅助服务器之前, 会将事务保存在内存日志中。此值将影响在内存日志变满之前服务器可能保持处于 PRIMARY ALONE 状态的时间。	8M 单位: 1 字节 k=KB m=MB	RO
NetcopyRpcTimeout	netcopy 操作的数据传输确认超时 (按毫秒计)	30000 单位: 1 毫秒	RW
PingInterval	主服务器和辅助服务器会定期互相发送“ping”消息以确保它们仍然互相保持连接。(这些 ping 消息与主服务器发送至辅助服务器的事务信息无关。) 此值等于服务器发送的两个连续 ping 操作之间的时间间隔 (按毫秒计)。	1000 (1 秒) 单位: 1 毫秒	RW
PingTimeout	此参数指定一个服务器在断定另一个服务器已关闭或者不可访问之前应等待的时间。 经过所指定的时间 (按毫秒计) 之后, 服务器将断定连接已断开并且将相应地更改状态。 请参阅第 44 页的「PingTimeout 和 PingInterval 参数 [HotStandby]」。 另请参阅 ConnectTimeout 。	4000 (4 秒) 单位: 1 毫秒	RW

表 19. HotStandby 参数 (续)

参数名称	描述	出厂值	访问方式
PrimaryAlone	不推荐使用此参数。请使用 AutoPrimaryAlone 参数。	No	RW
SafenessLevel	<p>此参数用于设置复制协议的安全级别。</p> <p>通过使用“auto”值，可以根据耐久性级别来动态更改安全级别。如果将 SafenessLevel 设置为“auto”，并且使用 SET DURABILITY 命令或者 DurabilityLevel 参数将耐久性设置为“宽松”，那么安全级别将设置为 1-safe；当您将耐久性级别设置为“严格”时，安全级别将设置为 2-safe。但是，如果 DurabilityLevel 设置为 2（自适应耐久性），那么“auto”设置将不起作用 - 安全级别始终将为 2-safe。</p>	可以使用的值为： 1safe、2safe 和 auto	RW

高可用性控制器配置参数

下列各表说明了 solidhac.ini 配置文件中的高可用性控制器配置参数。配置文件分成不同的节；各个表分别位于相应的节下面。各个参数是按照它们在配置文件中的相同顺序提供的。

[HAController] 节

表 20. HAC 配置参数: [HAController] 节

参数名称	描述	出厂值
Listen	HAC 侦听端口的协议和端口。高可用性管理器使用此端口将命令发送至 HAC。	tcp 3135
StartInAutomaticMode	<p>此参数用于定义 HAC 是否以自动方式开始执行。这是一个可动态更改的参数。</p> <p>在“自动”方式下，HAC 将自动尝试通过监视服务器的状态并在必要时执行与数据库相关的操作来维护数据库的可用性。相比之下，在“管理”方式下，HAC 将仅监视系统组件的状态。</p> <p>另请参阅 PreferredPrimary 参数。</p> <p>值可以是“Yes”和“No”。</p>	Yes

表 20. HAC 配置参数: [HAController] 节 (续)

参数名称	描述	出厂值
EnableDBProcessControl	<p>此参数允许 HAC 通过自动启动数据库以及通过为用户提供用于关闭并重新启动数据库进程的命令来管理本地数据库进程。</p> <p>仅当 HAC 处于“活动”方式时此值才有效。</p> <p>值可以是“Yes”和“No”。</p> <p>这是一个可选参数。</p>	No
EnableAutoNetcopy	<p>此参数用于定义 HAC 在必要时是否可以启动网络复制操作。</p> <p>值可以是“Yes”和“No”。</p> <p>这是一个可选参数。</p>	No
RequiredConnectFailures	<p>此参数用于定义在认为服务器已失败之前需要的连接尝试连续失败的次数。</p> <p>值可以是大于 1 的任何数值，或者是无限制。</p> <p>这是一个可选参数。</p>	1
CheckTimeout	<p>此参数用于定义 SQL 命令在“检查”方式下的超时（按毫秒计）。</p> <p>值可以是大于 1 的任何毫秒数，或者是无限制。</p>	50
CheckInterval	<p>此参数用于定义两次连续检查服务器状态之间的时间间隔（按毫秒计）。</p> <p>值可以是大于 1 的任何毫秒数，或者是无限制。</p> <p>这是一个可选参数。</p>	1000
Username	<p>HAC 的用户名。</p> <p>这是一个必需参数。</p>	
Password	<p>HAC 的密码。</p> <p>这是一个必需参数。</p>	
DBUsername	<p>HotStandby 服务器的用户名。</p> <p>这是一个必需参数。</p>	

表 20. HAC 配置参数: [HAController] 节 (续)

参数名称	描述	出厂值
DBPassword	HotStandby 服务器的密码。 这是一个必需参数。	

[LocalDB] 节

表 21. HAC 配置参数: [LocalDB] 节

参数名称	描述	出厂值
Connect	此参数允许 HAC 连接至本地数据库。本地数据库被定义为一个网络名。网络名由一种通信协议和一个服务器名称组成。 这是一个必需参数。	tcp 2125
StartScript	此参数声明用来启动数据库进程的脚本的名称。此参数是可选的。但是，如果 HAC 将控制数据库进程，那么建议您设置此参数。如果 HAC 采用“活动”方式并且 EnableDBProcessControl 参数设置为“ Yes ”，那么 HAC 将使用给定的脚本。 这是一个可选参数。	/home/solid/start_solid.sh
PreferredPrimary	如果两个服务器的 logpos 值相等，那么此参数将定义本地服务器是否将成为主服务器。如果两个服务器都具有值“ No ”或者 no ，那么第一个服务器将成为主服务器。 仅当 StartInActiveMode 参数的值为“ Yes ”时此值才有效。 值可以是“ Yes ”和“ No ”。 这是一个可选参数。	No

[RemoteDB] 节

表 22. HAC 配置参数: [RemoteDB] 节

参数名称	描述	出厂值
Connect	<p>此参数允许 HAC 连接至远程数据库。本地数据库是通过提供通信协议、远程服务器 IP 地址及其端口来定义的。</p> <p>这是一个必需参数。</p>	tcp 192.168.3.123 2125

[ERE] 节

表 23. HAC 配置参数: [ERE] 节

参数名称	描述	出厂值
EREIP	<p>此参数用于标识外部引用实体的 IP 地址。</p> <p>另请参阅 RequiredPingFailures 参数。</p> <p>这是一个可选参数。</p>	EREIP=192.168.3.1
RequiredPingFailures	<p>此参数定义在 HAC 断定服务器已与 ERE 断开连接并与客户机网络分隔开之前，必须连续失败的 ping 调用次数。</p> <p>只能对 ERE 使用此参数。</p> <p>值可以是大于 1 的任何数值，或者是无限制。</p> <p>这是一个可选参数。</p>	3

高可用性管理器配置参数

HAManager.ini 配置文件中的高可用性管理器配置参数。

表 24. 高可用性管理器配置参数

参数名称	描述
Header_text	<p>此参数用于为 HA 管理器定义标题文本。此值显示在用户界面中。</p> <p>这是一个必需参数。</p>
Server1_host Server2_host	这两个参数用于定义 HAC 实例的主机名。

表 24. 高可用性管理器配置参数 (续)

参数名称	描述
Server1_name Server2_name	这两个参数用于定义 HAC 实例的名称。
Server1_pass Server2_pass	这两个参数用于定义 HAC 实例的密码。
Server1_port Server2_port	这两个参数用于定义 HAC 实例的端口。
Server1_user Server2_user	这两个参数用于定义 HAC 实例的用户名。
Window_title	此参数用于为 HA 管理器定义窗口标题。此值显示在用户界面中。 这是一个必需参数。

附录 B. 错误代码

本节说明了与 HotStandby 相关的错误代码。在《solidDB 管理员指南》中说明了其他大多数服务器错误代码。

本节中说明的某些错误是 ADMIN COMMAND 结果集的 RC 列的值，而其他某些错误是作为 ODBC 或 JDBC 驱动程序的错误代码返回的。例如，所有“solidDB HotStandby 错误”和“solidDB 高可用性控制器错误”是 ADMIN COMMAND 结果集的值，而所有“solidDB 通信错误”是由驱动程序返回的。

本节中包含的各个表中涵盖的错误类别为：

HotStandby 错误和状态码

使用特定 HotStandby 命令时发生 solidDB HotStandby 错误（14009 - 147xx, 307xx），它们是 solidDB SQL 扩展。

表 25. HotStandby 错误和状态码

错误或状态码	描述
14003	活动 在命令的结果集中可能会返回此状态的 ADMIN COMMAND: <ul style="list-style-type: none">• ADMIN COMMAND 'hotstandby status switch'• ADMIN COMMAND 'hotstandby status catchup'• ADMIN COMMAND 'hotstandby status copy' 含义：切换进程、同步复制进程、复制或网络复制进程仍然处于活动状态。
14007	正在连接 在命令的结果集中可能会返回此状态的 ADMIN COMMAND: <ul style="list-style-type: none">• ADMIN COMMAND 'hotstandby status connect' 含义：主服务器和辅助服务器正在建立连接。
14008	同步复制 在命令的结果集中可能会返回此状态的 ADMIN COMMAND: <ul style="list-style-type: none">• ADMIN COMMAND 'hotstandby status connect' 含义：主服务器已连接至辅助服务器，但是尚未完全复制事务日志。仅从主服务器中返回此消息。
14009	先前未进行服务器切换。 在命令的结果集中可能会返回此状态的 ADMIN COMMAND: <ul style="list-style-type: none">• ADMIN COMMAND 'hotstandby status switch' 含义：两个服务器之间从未发生切换进程。

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14501	<p>操作失败。</p> <p>含义: 操作失败, 服务器即将关闭。故障原因可能是由于对非 HotStandby 服务器发出了命令, 或者是对命令不适用于的主服务器或辅助服务器发出了命令。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch primary' • ADMIN COMMAND 'hotstandby switch secondary' • ADMIN COMMAND 'hotstandby cominfo' • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby status switch' • ADMIN COMMAND 'hotstandby set standalone' • ADMIN COMMAND 'hotstandby copy' • ADMIN COMMAND 'hotstandby netcopy'
14502	<p>RPC 参数无效</p> <p>含义: HSB 连接字符串中提供的某些连接信息是错误的, 并且连接至另一个服务器失败。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby status connect'
14503	<p>通信错误, 丢失连接。</p> <p>含义: 发生了通信错误, 未找到另一个服务器。连接至另一个服务器时发生故障。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby status connect'
14520	<p>服务器是 HotStandby 辅助服务器, 不允许建立连接。</p>
14522	<p>未指定 HotStandby 复制目录。</p> <p>含义: 未指定复制目录。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby copy' <p>要解决此问题, 要么在命令中指定目录, 例如:</p> <p>ADMIN COMMAND 'hotstandby copy \Secondary\dbfiles\'</p> <p>要么在 solid.ini 配置文件中设置 CopyDirectory 参数。</p>

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14523	<p>切换进程已处于活动状态。</p> <p>含义: 在 HotStandby 服务器中, 切换进程已处于活动状态。如果您只需要完成当前切换, 那么请等待。如果您要尝试再次进行切换 (即, 切换回最初配置), 那么必须等待第一次切换完成之后才能开始第二次切换。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch primary' • ADMIN COMMAND 'hotstandby switch secondary' • ADMIN COMMAND 'hotstandby status switch'
14524	<p>HotStandby 数据库具有不同的基本数据库, 数据库时间戳记是不同的。</p> <p>含义: 数据库来源于不同的种子数据库。您必须使这些数据库同步。可能需要执行从主服务器的数据库网络复制到辅助服务器。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby status switch'
14525	<p>HotStandby 数据库之间未正确地同步。</p> <p>含义: 数据库之间未正确地同步。您必须使这些数据库同步。您可能需要使用命令行参数 -x backupserver 启动其中一个数据库服务器 (您打算使它成为辅助服务器的数据库服务器), 然后将主服务器的数据库网络复制到辅助服务器。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby status switch'
14526	<p>参数无效。</p> <p>含义: HotStandby ADMIN COMMAND 中使用的参数未知或者无效。</p> <p>所有 HotStandby 命令都可能会在 ADMIN COMMAND 的结果集中返回此错误。</p> <p>注意: 在下列 HotStandby 命令中, “参数无效”错误是一个语法错误, 这时, 无法将指定的主服务器或辅助服务器应用于切换:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch primary' • ADMIN COMMAND 'hotstandby switch secondary'

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14527	<p>这是一个非 HotStandby 服务器。</p> <p>含义: 在一个未对 HotStandby 配置的服务器上执行了此命令。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby status switch' • ADMIN COMMAND 'hotstandby switch primary' • ADMIN COMMAND 'hotstandby switch secondary' • ADMIN COMMAND 'hotstandby state'
14528	<p>两个 HotStandby 数据库都是主数据库。</p> <p>含义: 两个数据库都是主数据库。这是一个致命错误, 这是因为可能有互相冲突的更改。两个数据库都会被系统自动降级为辅助数据库。您必须决定哪个数据库是真正的主数据库, 然后使两个数据库同步。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby status switch'
14535	<p>服务器已经是主服务器。</p> <p>含义: 您尝试切换为主服务器的服务器已经处于 PRIMARY 状态。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> •
14536	<p>服务器已经是辅助服务器。</p> <p>含义: 您尝试切换为辅助服务器的服务器已经处于 SECONDARY 状态。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch secondary'
14537	<p>HotStandby 连接已断开。</p> <p>含义: 从主服务器和辅助服务器中都将返回此命令。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby status connect' <p>可能会导致此问题的一个原因是辅助服务器的 solid.ini 文件中具有不正确的连接字符串。如果 netcopy 操作成功, 但是 connect 命令失败, 那么请检查连接字符串。(Netcopy 操作不需要辅助服务器与主服务器建立单独的连接, 因此, 即使辅助服务器上的连接字符串错误, netcopy 操作也可能会成功。)</p>

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14538	<p>服务器不是 HotStandby 主服务器。</p> <p>含义: 要发出此命令, 服务器必须是一个 HotStandby 主服务器。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby copy copy_directory' • ADMIN COMMAND 'hotstandby netcopy' • ADMIN COMMAND 'hotstandby connect' • ADMIN COMMAND 'hotstandby set primary alone' • ADMIN COMMAND 'hotstandby set standalone'
14539	<p>操作被拒绝。</p> <p>当发生下列情况之一时就会提供此错误代码:</p> <ul style="list-style-type: none"> • 用户对主服务器发出了 netcopy 命令, 但是应当作为辅助服务器的服务器实际上并不处于 SECONDARY 状态, 或者未采用“netcopy 侦听方式”。(主服务器和“辅助”服务器可能都处于 PRIMARY ALONE 状态。) <p>要解决此问题, 可使用 -x backupserver 命令行选项重新启动“辅助”服务器, 然后再次尝试对主服务器发出 netcopy 命令。</p> <p>警告: 如果两个服务器先前都处于 PRIMARY ALONE 状态, 并且两个服务器都在它们处于 PRIMARY ALONE 状态的情况下执行了事务, 那么每个服务器可能都具有另一个服务器不具有的数据。这是一个严重错误, 并且通过执行 netcopy 使它们恢复同步将导致覆盖已经在“辅助”服务器中落实的一些事务。</p> <ul style="list-style-type: none"> • 当您使用回调函数时, 如果此回调函数拒绝关闭或接受 backup 或 netcopy 命令, 就会生成此消息。 <p>当您使用链接库访问时, 可以通过使用 SSCSetNotifier 函数来提供“回调”函数。当已经命令服务器关闭或者执行 netcopy 操作时, 将通知回调函数。如果您的应用程序因某种原因不希望遵从此命令, 那么回调函数可能会返回一个用于取消此命令的值。在这种情况下, 将产生错误 14539。</p> <p>要解决此问题, 请等到客户机代码完成它不希望中断的操作之后, 重试此命令(例如, 关闭或 netcopy)。</p>
14540	服务器已经是一个非 HotStandby 服务器。
14541	solid.ini 中的 HotStandby 配置与 ADMIN COMMAND 'HSB SET STANDALONE' 相冲突。
14542	服务器采用备份服务器方式。操作被拒绝。
14543	命令无效。数据库是一个 HotStandby 数据库, 但是在 solid.ini 配置文件中未找到 [HotStandby] 节。
14544	操作失败。此命令在无盘服务器上不受支持。

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14545	当主服务器的角色为 PRIMARY BROKEN 时，只能将它设置为 PRIMARY ALONE。
14546	<p>切换失败。服务器或远程服务器无法从 PRIMARY ALONE 切换为辅助服务器。在切换之前首先应执行同步复制。</p> <p>含义：当在处于 PRIMARY ALONE 状态的本地或远程主服务器中执行了将状态切换为 SECONDARY，并且它检测到主服务器与辅助服务器不同步时，就会返回此命令。必须将主服务器连接至辅助服务器并且等待同步复制进程完成之后才将辅助服务器切换为主服务器。</p> <p>将返回此错误的 HotStandby 命令：</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch secondary'
14547	-R 选项（读取超时）的值缺少或者无效。
14548	<p>切换失败。无法将处于 STANDALONE 状态的服务器切换到 SECONDARY 状态。</p> <p>含义：当在处于 STANDALONE 状态的本地或远程主服务器中执行了将状态切换为 SECONDARY，并且它检测到主服务器与辅助服务器不同步时，就会返回此命令。必须将主服务器连接至辅助服务器并且等待同步复制完成之后才将辅助服务器切换为主服务器。</p> <p>将返回此错误的 HotStandby 命令：</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hotstandby switch secondary'
14549	<p>HotStandby 事务处于活动状态。</p> <p>含义：如果 HotStandby 连接已断开，那么在关闭主服务器之前必须将它设置为 STANDALONE 方式或者切换到 SECONDARY 方式。</p>
14550	仅当主服务器未连接至辅助服务器时，才能更改 HotStandby 连接参数。
14551	已达到 START AFTER COMMIT 语句的最大数目。
14552	服务器采用备份服务器方式，不允许建立连接。
14700	<p>拒绝了连接，两个服务器都具有 PRIMARY 角色。</p> <p>含义：如果两个节点都具有同一角色，那么 'hsb connect' 命令将返回此错误。</p>
14701	<p>14701: 拒绝了连接，两个服务器都具有 SECONDARY 角色。</p> <p>含义：如果两个节点都具有同一角色，那么 'hsb connect' 命令将返回此错误。</p>

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
14702	<p>14702: 操作失败, 同步复制处于活动状态。</p> <p>含义: 当两个服务器正在执行同步复制时, 如果在主服务器上发出下列任何命令, 就会产生此错误: 'hsb switch secondary'、'hsb set secondary alone'、'hsb set standalone'、'hsb connect'、'hsb copy' 或 'hsb netcopy'。</p> <p>当两个服务器正在执行同步复制时, 如果在辅助服务器上发出下列任何命令, 就会产生此错误: 'hsb switch primary'、'hsb set secondary alone'、'hsb set primary alone'、'hsb set standalone' 或 'hsb connect'。</p>
14703	<p>操作失败, 复制处于活动状态。</p> <p>含义: 当主服务器正在执行复制或网络复制时, 下列命令将返回此错误: 'hsb switch secondary'、'hsb set secondary alone'、'hsb set standalone'、'hsb connect'、'hsb disconnect'、'hsb copy' 或 'hsb netcopy'。</p>
14704	<p>仅当主服务器处于 PRIMARY ALONE 状态时才允许执行 HotStandby 复制或网络复制。</p> <p>含义: 如果服务器处于 PRIMARY ACTIVE 状态并且发出了 'hsb copy' 或 'hsb netcopy' 命令, 就会返回此错误。</p>
14705	<p>在此状态下, 不允许设置为 STANDALONE。</p> <p>含义: 如果服务器处于 PRIMARY ACTIVE 状态并且您发出了 'hsb set standalone' 命令, 那么将产生此消息。</p>
14706	HotStandby 的读取线程方式无效, 仅支持方式 2。
14707	不允许在 STANDALONE 状态下执行操作。
14708	同步复制失败, 在日志文件中找不到同步复制位置。
14709	启用了 HotStandby, 但是未定义连接字符串。
14710	HotStandby 管理命令与人局管理命令冲突。
14711	由于服务器即将关闭而失败。
14712	服务器是辅助服务器。请使用主服务器来完成此操作。
30787	<p>pri_dologskip: 类型、日志位置和日志大小错误。</p> <p>此错误是由 HSB 主服务器上的失败操作导致的。此错误将返回失败的操作及其在日志中的位置以及日志大小。跳过了复制日志中的操作。</p>
30788	<p>pri_hsblogcopy_write: 类型、日志位置和日志大小错误。</p> <p>此错误是由 HSB 主服务器上的失败操作导致的。写入复制日志文件失败。此错误将返回失败的操作及其在日志中的位置以及日志大小。</p>
30789	打开热备用复制日志文件失败。

表 25. HotStandby 错误和状态码 (续)

错误或状态码	描述
30790	<p>为 HotStandby 日志分配内存失败。最大日志大小为 <i>logsize</i>。</p> <p>此错误与使用 HotStandby 的无盘数据库有关。在这些系统中，HotStandby 日志被写入内存中。如果为日志文件分配更多内存失败，就会产生此错误。</p>

高可用性控制器错误和状态码

当使用特定的高可用性控制器命令时发生 solidDB 高可用性控制器错误（17xxx）。

表 26. 高可用性控制器错误和状态码

错误或状态码	描述
17501	<p>HAC 即将关闭。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> ADMIN COMMAND 'hac shutdown' <p>含义: 高可用性控制器即将关闭。</p>
17502	<p>命令失败, HAC 已被暂挂。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> ADMIN COMMAND 'hac suspend' ADMIN COMMAND 'hac getsbdstate' ADMIN COMMAND 'hac getdbstate' ADMIN COMMAND 'hac shutdowndb' ADMIN COMMAND 'hac restartdb' ADMIN COMMAND 'hac switchdb' ADMIN COMMAND 'hac statemachinestate' ADMIN COMMAND 'hac getereip' ADMIN COMMAND 'hac pingere' <p>含义: 命令执行失败, 高可用性控制器已被暂挂。</p>
17503	<p>未知命令。请输入 'hac commands' 以获取帮助。</p> <p>含义: ADMIN COMMAND 语法不正确。</p>
17504	<p>HAC 已经在运行。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> ADMIN COMMAND 'hac resume'

表 26. 高可用性控制器错误和状态码 (续)

错误或状态码	描述
17506	<p>HSB 状态不允许进行切换。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac switchdb'
17507	<p>无法执行命令。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac shutdowndb'
17509	<p>重新启动数据库服务器失败。有关详细信息, 请参阅 solmsg.out。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac restartdb'
17510	<p>无法连接至数据库服务器。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac switchdb'
17511	<p>数据库服务器未关闭。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac restartdb'
17513	<p>切换失败。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac switchdb'
17514	<p>在配置文件中未指定 ERE IP。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac getereip' • ADMIN COMMAND 'hac pingere'
17516	<p>HAC 已处于活动状态。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> • ADMIN COMMAND 'hac setactive' • ADMIN COMMAND 'hac pingere'

表 26. 高可用性控制器错误和状态码 (续)

错误或状态码	描述
17517	<p>HAC 已处于被动状态。</p> <p>在命令的结果集中可能会返回此状态的 ADMIN COMMAND:</p> <ul style="list-style-type: none"> ADMIN COMMAND 'hac setpassive'

solidDB 数据库错误

solidDB 数据库错误 (10002 - 10050) 由 solidDB 检测到并发送至客户机应用程序。它们可能要求执行管理操作。

表 27. solidDB 数据库错误

错误代码	描述
数据库错误 10002	<p>操作失败。</p> <p>含义: 连接操作因发生意外错误而失败。很可能是因为服务器之间未正确地同步。</p>
数据库错误 10013	<p>事务是只读事务。</p> <p>含义: 您已尝试在设置为只读的事务中写入, 或者服务器临时 (例如, 在状态切换期间) 设置为只读方式。不允许使用可更新的事务。</p>
数据库错误 10019	<p>备份已处于活动状态。</p> <p>含义: 当一个备份或副本已经在运行时, 您又尝试启动一个备份或副本。</p>
数据库错误 10024	<p>备份目录“<i>directory_name</i>”非法。</p> <p>含义: 备份或副本目录是一个空字符串或者是一个句点 (表示将在当前目录中创建备份或副本)。</p>
数据库错误 10030	<p>备份或副本目录“<i>directory_name</i>”不存在。</p> <p>含义: 未找到备份或副本目录。请检查备份或副本目录的名称是否正确。</p>
数据库错误 10045	<p>不能在 HotStandby 辅助服务器上执行此操作。</p> <p>含义: 不能在 HotStandby 辅助服务器上执行此操作。</p> <p>为了成功执行所请求的操作, 服务器必须是主服务器。</p>
数据库错误 10046	<p>操作失败, 数据字典操作处于活动状态。</p> <p>含义: 当前正在进行数据字典操作。</p>

表 27. solidDB 数据库错误 (续)

错误代码	描述
数据库错误 10047	复制的事务异常终止。 含义: 事务异常终止, 例如, 在状态切换时异常终止。当服务器状态从主服务器切换为辅助服务器时, 所有活动的事务都将异常终止。
数据库错误 10048	复制的事务包含数据字典更改, 不允许执行正常的更新操作。 含义: HotStandby 方式将限制执行数据字典操作; 例如, 不能将 CREATE TABLE 与正常的更新操作混合在一起执行。 此消息在版本 4.1 和更高版本中已过时, 这些版本允许您在使用 HSB 时在事务中执行 DML 和 DDL 操作。
数据库错误 10049	远程服务器不是辅助服务器。 含义: 您在命令中指定的服务器不处于 SECONDARY 状态。
数据库错误 10050	复制操作更新了 BLOB 列。 含义: 无法将 BLOB 列复制到辅助服务器。
数据库错误 10078	用户回滚了事务。
数据库错误 10079	无法除去文件规范。已经在使用文件。
数据库错误 10080	HotStandby 辅助服务器无法执行从主服务器中接收到的操作。 含义: 发生此错误的原因可能是未使用 HotStandby copy 或 netcopy 命令复制主服务器中的数据库。
数据库错误 10081	数据库文件不完整或者被破坏。 含义: 如果此文件位于 HotStandby 辅助服务器上, 那么请再次使用 hotstandby copy 或 hotstandby netcopy 命令从主服务器中发送此文件。
数据库错误 10082	备份异常终止。
数据库错误 10083	由于已经将落实操作发送至辅助服务器, 因此未能异常终止 HSB 事务。
数据库错误 10084	表未被锁定。
数据库错误 10085	已禁止执行检查点操作。
数据库错误 10087	不允许将主内存表用于 HotStandby。
数据库错误 10088	指定的锁定超时太长。
数据库错误 10089	操作失败, 服务器处于 HSB PRIMARY UNCERTAIN 状态。

solidDB 表错误

solidDB 数据库表错误（13xxx）是由错误的 SQL 语句导致的，solidDB 将检测到这些错误。不需要执行管理操作。

表 28. solidDB 表错误

错误代码	描述
表错误 13068	服务器正在关闭 含义：由于服务器正在关闭，因此您无法完成此操作。
表错误 13123	“ <i>table_name</i> ”表不是空的 含义：只有在表为空的情况下才能执行此操作。例如，只有在表为空的情况下才能将基于磁盘的表更改为内存表（或者将内存表更改为基于磁盘的表）。
表错误 13167	只有 M 表才能是瞬态表 含义：不能创建一个基于磁盘的瞬态表。例如，以下 SQL 语句将产生此错误消息： CREATE TRANSIENT TABLE t1 (i INT) STORE DISK;
表错误 13170	只有 M 表才能是临时表 含义：不能创建一个基于磁盘的临时表。例如，以下 SQL 语句将产生此错误消息： CREATE TEMPORARY TABLE t1 (i INT) STORE DISK;

solidDB 通信错误

solidDB 通信错误（21306 和 21308）是由网络错误导致的。这些错误要求执行管理操作。

表 29. solidDB 通信错误

错误代码	描述
通信错误 21306	未找到服务器“ <i>server_name</i> ”，连接失败。 含义：未找到辅助服务器。 <ul style="list-style-type: none">• 检查服务器是否正在运行。• 检查网络名是否有效。• 检查服务器是否在侦听所指定的网络名。
通信错误 21308	“连接已断开（%s '%s' 操作失败，错误代码为 %d）”。 例如， "连接已断开（TCP/IP"写入"操作失败，错误代码为 7）。" 请参阅针对错误 21306 提供的建议操作。

附录 C. HotStandby 管理命令总结

本节总结了可用于 HotStandby 的管理命令。

您使用的工具（例如，solidDB SQL 编辑器（solsql）或者 solidDB 远程控制（solcon））会影响您必须如何输入命令。

使用 solidDB SQL 编辑器（teletype 工具）时，请按照下面所显示的语法来输入命令：

```
ADMIN COMMAND 'hotstandby switch primary';
```

如果您在 solidDB 远程控制（teletype 工具）中输入这些命令，请务必仅指定命令（不带引号，也不带“ADMIN COMMAND”）；例如：

```
hotstandby switch primary
```

还应注意，可以将“hotstandby”缩写为“hsb”，例如：

```
hsb switch primary
```

在下面的表中，使用了可以采用的最短格式：省略了“ADMIN COMMAND”和引号，并且使用了缩写“hsb”和“hac”。

有关 solidDB 远程控制（teletype 工具）或 solidDB SQL 编辑器的更多信息，请参阅《solidDB 管理员指南》中的『使用 solidDB 数据管理工具』。

请注意，如果此命令中没有语法错误，那么 ADMIN COMMAND 始终将返回一个成功的返回码（0）。结果集的“RC”字段中包含此命令的实际结果代码。

HotStandby 命令

命令	说明
hsb cominfo	<p>返回用于连接至另一个服务器的通信信息（连接字符串）。这通常是 solid.ini 配置文件的 [HotStandby] 节中的 Connect 参数的值，但是也可能使用以下命令设置了此值：</p> <pre>ADMIN COMMAND 'hsb parameter connect connect_string';</pre> <p>可以在应用程序中使用此信息来连接至其他服务器。</p>
hsb connect	<p>如果主服务器与辅助服务器之间的连接已断开或者尚未建立连接，那么此命令会将主服务器连接至辅助服务器并启动 HotStandby 复制。由于没有在服务器之间建立自动连接的机制，因此，此命令始终需要连接服务器。在成功连接之后，主服务器的状态会自动从 PRIMARY ALONE 设置为 PRIMARY ACTIVE。如果连接不成功，那么状态仍然为 PRIMARY ALONE。</p> <p>可以在主服务器或辅助服务器上执行此命令。</p> <p>请注意，当您执行此命令时，如果主服务器与辅助服务器已建立连接，但是尚未将事务日志完全复制到辅助服务器中，那么将显示以下消息：同步复制处于活动状态。</p>

命令	说明
<p>hsb copy [<i>directory_name</i>]</p>	<p>注意: 不推荐使用此命令。建议您使用 <code>hsb netcopy</code> 命令。</p> <p>可以使用 <code>hsb copy</code> 命令根据主数据库来创建初始辅助数据库。此命令会将数据库复制到主节点本地的一个目录（当然也会复制到辅助节点本地的一个目录）。完成复制之后，您就可以启动辅助服务器。将主服务器连接至辅助服务器之后，主服务器会将事务日志复制到辅助服务器，从而使辅助服务器也变成最新的。</p> <p>还可以使用此命令使主数据库与位于主节点本地的一个目录中的辅助数据库同步（要在辅助数据库相当长一段时间内处于脱机状态才进行此操作）。请阅读第 78 页的『使主服务器与辅助服务器同步』。</p> <p>如果指定了可选的 <i>directory_name</i>，那么会将数据库文件复制到此目录中；否则，会将它复制到使用 <code>solid.ini</code> 配置文件的 <code>[hotstandby]</code> 节中的 copydirectory 参数指定的目录中。因为 <code>hsb copy</code> 命令不会复制 <code>solid.ini</code> 配置文件或者日志文件，所以建议您使此目录不同于正常的备份目录。</p> <p>主服务器只有处于 <code>PRIMARY ALONE</code> 状态时才能执行 <code>hsb copy</code> 命令。在执行此命令期间和执行之后，服务器将保持 <code>PRIMARY ALONE</code> 状态。完成此命令之后，可以启动辅助服务器，然后连接这两个服务器。</p>
<p>hsb disconnect</p>	<p>此命令让服务器与 <code>HSB</code> 对的另一个成员适时断开连接。此命令在主服务器或辅助服务器上都有效。使用此命令的一个典型原因是在对其中一个服务器升级之前将两个服务器断开连接。（可以将另一个服务器设置为 <code>PRIMARY ALONE</code> 状态，以便它可以继续对客户机请求作出响应。）</p> <p>此命令通常会导致两个服务器都进入“ALONE”方式；即，主服务器从 <code>PRIMARY ACTIVE</code> 切换为 <code>PRIMARY ALONE</code>；而辅助服务器从 <code>SECONDARY ACTIVE</code> 切换为 <code>SECONDARY ALONE</code>。</p> <p>此命令在主服务器和辅助服务器上都有效。</p> <p>请注意，使用 <code>shutdown</code> 命令</p> <pre>ADMIN COMMAND 'shutdown';</pre> <p>会导致服务器在关闭之前会执行受控制的断开连接。如果辅助服务器已关闭（并且已断开连接），那么主服务器知道进入 <code>PRIMARY ALONE</code> 状态是安全的，并且它也会进入此状态。</p>
<p>hsb logpos</p>	<p>看守程序（或者用户）可以使用这两个命令来确定两个服务器中的哪个应切换为主服务器，哪个应切换为辅助服务器。（在服务器之间互相失去联系之前是主服务器的服务器现在并不是一定成为主服务器。）此方法用于检测哪个服务器更“优先”（也就是说，哪个服务器接受了更多事务），从而在建立 <code>HSB</code> 连接之前应当将此服务器作为主服务器。</p> <p>有关如何使用此命令的详细说明，请参阅第 91 页的『选择使哪个服务器成为主服务器』。</p> <p>下面显示了此命令的典型输出：</p> <pre>ADMIN COMMAND 'hsb logpos'; RC TEXT -- ---- 0 0000000000000000000000000871:PRIMARY</pre> <p>输出中包含日志操作标识和服务器先前具有的角色，在此示例中为“主服务器”。</p>

命令	说明
hsb netcopy	<p>此命令用来通过使用在 solid.ini 的 [HotStandby] 节中指定的 Connect 参数将主数据库或者无盘内存数据复制到辅助服务器。一旦使用连接字符串来连接至辅助服务器，就会通过网络链路来复制数据库文件。</p> <p>可以使用此命令使主数据库与已经长时间保持脱机状态的辅助数据库同步。请阅读第 78 页的『使主服务器与辅助服务器同步』。</p> <p>还可以使用此命令来创建新的辅助数据库。执行此操作的原因可能是辅助数据库已被破坏、为新的 HotStandby 配置创建初始辅助数据库或者向现有配置添加新的辅助数据库。有关详细信息，请阅读第 82 页的『通过网络将主数据库复制到辅助服务器』。</p> <p>主服务器必须处于 PRIMARY ALONE 状态才能发出此命令。</p> <p>完成此命令之后（无论成功还是失败），主服务器的状态将保持不变。</p> <p>如果成功完成了复制，那么辅助服务器将自动切换为 SECONDARY ALONE 状态。</p> <p>执行 netcopy 命令之后通常应接着执行“connect”命令以连接主服务器和辅助服务器。将主服务器连接至辅助服务器之后，主服务器会通过复制事务日志来自动使辅助服务器也变成最新的。</p>
hsb parameter	<p>（不推荐使用）此命令允许您设置特定于 HSB 的参数，例如，AutoPrimaryAlone、Connect 和 PingTimeout。有关每个这些参数的完整描述，请参阅第 103 页的附录 A，『HotStandby 配置参数』。</p> <p>请注意，当您设置其中某个参数的值时，此命令将立即生效，但是在执行 shutdown 命令之前不会写入 solid.ini 配置文件。</p> <p>此命令的语法为：</p> <pre>ADMIN COMMAND 'hsb parameter param_name param_value';</pre> <p>请注意，此命令不使用等号。因此，它不同于其他类似的命令（建议您这样做）：</p> <pre>ADMIN COMMAND 'parameter hotstandby.param_name = param_value';</pre>

命令	说明
hsb role	<p>注意: 不推荐使用此命令。请使用 <code>hsb state</code>。</p> <p>在结果集中返回下列其中一个角色:</p> <ul style="list-style-type: none"> • PRIMARY, 如果所连接的服务器是一个正常的主服务器。在此角色下, 会将主服务器中的事务发送至辅助服务器。 • PRIMARY NOHSBLOG, 表示主服务器将接受事务并将它们存储在数据库中。但是, 它不会将事务存储在日志中以便稍后可以将它们发送至辅助服务器。要使辅助服务器与主服务器重新同步, 必须将主服务器中的整个数据库复制到辅助服务器。 • PRIMARY BROKEN, 如果主服务器已与辅助服务器断开连接。在主服务器中只能执行只读事务。 • PRIMARY ALONE, 如果主服务器在独自运行。与辅助服务器的连接已断开, 但是主服务器接受了事务并且已添加至事务日志, 以便稍后可以将它们发送至辅助服务器。 • PRIMARY CATCHUP, 如果正在进行同步复制。在同步复制期间, 在主服务器中发出 <code>'hsb connect'</code> 命令之后, 主服务器会自动将事务日志更改发送至辅助服务器。完成同步复制进程之后, 服务器的角色将自动切换为 PRIMARY。如果在连接之前主服务器的角色为 PRIMARY ALONE, 那么主服务器将继续接受事务。 • SECONDARY, 如果所连接的服务器是一个正常的辅助服务器。这意味着此服务器将从主服务器中接收并应用事务。 • SECONDARY BROKEN, 如果辅助服务器已与主服务器断开连接。 • SECONDARY CATCHUP, 如果在主服务器中发出 <code>'hsb connect'</code> 命令之后, 辅助服务器正在同步复制主服务器中所作的更改。完成同步复制进程之后, 辅助服务器的角色将自动切换为 SECONDARY。 <p>如果在一个未对 HotStandby 配置的服务器上发出了 ADMIN COMMAND <code>'hsb role'</code>, 那么将返回以下错误消息: 14527: 这是一个非 HotStandby 服务器。</p> <p>此命令将与 SQL 函数 <code>HOTSTANDBY_ROLE</code> 返回相同的信息。</p>

命令	说明
hsb set primary alone	<p>无条件地将 HotStandby 主服务器设置为 PRIMARY ALONE 状态。此命令在下列状态下是合法的: PRIMARY ACTIVE、SECONDARY ACTIVE、SECONDARY ALONE 和 STANDALONE。</p> <p>可以使用此命令来实现快速故障转移。当辅助服务器处于 SECONDARY ACTIVE 状态时, 如果接收到此命令, 那么此服务器将不会尝试与主服务器进行通信。它将立即切换为 PRIMARY ALONE 状态。如果在辅助服务器检测到故障之前, 就已将有关主服务器故障的信息传送到 HAC 中(延迟时间由 PingTimeout 和 PingInterval 参数指定), 就会产生此行为。</p> <p>但是, 如果在辅助服务器中执行此命令时主服务器是活动的并且处于 PRIMARY ACTIVE 状态(例如, 不正确的故障检测就会导致这种情况), 那么将强制使主服务器进入 PRIMARY UNCERTAIN 状态。然后, 可以将它过渡到 SECONDARY ALONE 状态并重新进行连接, 而不会丢失任何事务。 注: 用于执行故障转移的另一种方法是使用 hsb switch primary 命令。</p> <p>在 PRIMARY ALONE 状态下, 与辅助服务器的连接已断开, 但是此状态允许您运行主服务器并连续更新事务日志。PRIMARY ALONE 状态将一直保持到主服务器已关闭、成功地连接至辅助服务器或者服务器用尽了用于保存事务日志的磁盘空间为止。</p> <p>请注意, 当您将服务器设置为 PRIMARY ALONE 状态时, 它并不会自动尝试与其他服务器重新建立连接。</p> <p>要点: 在服务器上执行此命令之前, 应尽力确保服务器对中的另一个服务器尚未处于 PRIMARY ALONE 状态(或 STANDALONE 状态)。避免存在“双重主服务器”是非常重要的(请参阅第 35 页的『网络分区和双重主服务器』)。</p> <p>另请参阅 'hsb switch primary' 命令。</p>
hsb set secondary alone	<p>此命令将服务器状态设置为 SECONDARY ALONE。如果服务器当前处于下列状态之一, 就可以使用此命令: PRIMARY ALONE、PRIMARY UNCERTAIN 和 STANDALONE。</p>
hsb set standalone	<p>发出此命令时, 主服务器的状态将变为 STANDALONE。服务器将停止为辅助服务器存储事务。主服务器(处于 STANDALONE 状态)可以继续接受读/写事务。当辅助服务器在相当长一段时间内处于脱机状态, 并且事务日志可能变得太大时, 在主服务器中使用此选项就很有用。如果服务器当前处于下列状态之一, 就可以使用此命令: PRIMARY ALONE 或 SECONDARY ALONE。</p>

命令	说明
hsb state	<p>在结果集中返回下列其中一种状态:</p> <ul style="list-style-type: none"> • PRIMARY ACTIVE, 如果所连接的服务器是一个正常的主服务器。在此状态下, 会将主服务器中的事务发送至辅助服务器。 • STANDALONE, 表示主服务器将接受事务并将它们存储在数据库中。但是, 它不会存储事务以将它们转发至辅助服务器。 • PRIMARY UNCERTAIN, 如果主服务器已与辅助服务器断开连接, 且尚未切换到另一种状态 (例如, PRIMARY ALONE)。在主服务器中只能执行只读事务。 • PRIMARY ALONE, 如果主服务器在独自运行。与辅助服务器的连接已断开, 但是主服务器接受了事务并且已存储在事务日志, 以便可以将它们转发至辅助服务器。 • SECONDARY ACTIVE, 如果所连接的服务器是一个正常的辅助服务器。这意味着此服务器将从主服务器中接收并应用事务。 • SECONDARY ALONE, 如果辅助服务器已与主服务器断开连接。 <p>如果在一个未对 HotStandby 配置的服务器上发出了 ADMIN COMMAND 'hsb state', 那么将返回以下错误消息: 14527: 这是一个非 HotStandby 服务器。</p> <p>此命令将与 SQL 函数 HOTSTANDBY_STATE 返回相同的信息。有关此函数的详细信息, 请阅读第 67 页的『将应用程序切换到新的主服务器』中使用 HOTSTANDBY_STATE 函数一节。</p> <p>有关在执行管理操作和故障诊断操作时发生的 HotStandby 状态过渡的概述, 请参阅第 135 页的附录 D, 『服务器状态过渡』。</p>
hsb status option	<p>返回 HotStandby 状态信息。option 可以是下列任何一项:</p> <ul style="list-style-type: none"> • catchup • connect • copy • switch <p>有关更多详细信息, 请参阅下面对各个命令/选项 (例如, 'hsb status catchup') 的描述。</p> <p>status 命令的意图是提供有关成功启动之后花了很长时间执行的操作的结果信息。此命令将返回最后一个成功启动的操作的状态。如果启动某项操作失败 (例如, 由于状态不正确而失败), 那么 status 命令将不会返回此操作的状态, 而是返回所执行的前一个命令的状态。</p>
hsb status catchup	<p>指示服务器是否正在执行同步复制, 即, 辅助服务器在读取主服务器的事务日志并应用更改。</p> <p>值可能是:</p> <ul style="list-style-type: none"> • ACTIVE • NOT ACTIVE

命令	说明
hsb status connect	<p>返回的状态信息:</p> <ul style="list-style-type: none"> • CONNECTED (已连接) - 连接处于活动状态。从主服务器和辅助服务器中都将返回此信息。 • CONNECTING (正在连接) - 主服务器与辅助服务器正在互相连接。从主服务器和辅助服务器中都将返回此信息。 • CATCHUP (同步复制) - 主服务器已连接至辅助服务器, 但是尚未将 HotStandby 主数据库日志完全复制到辅助服务器。从主服务器和辅助服务器中都将返回此信息。 • • BROKEN (已断开) - 主服务器与辅助服务器之间的连接已断开。从主服务器和辅助服务器中都将返回此信息。 <p>请注意: 此命令将与 SQL 函数 HOTSTANDBY_CONNECTSTATUS 返回相同的信息。有关此函数的详细信息, 请阅读第 67 页的『将应用程序切换到新的主服务器』中使用 HOTSTANDBY_CONNECTSTATUS 函数一节。</p>
hsb status copy	<p>此命令允许您检查上一个 hsb copy 或 hsb netcopy 命令的结果。请注意, 即使您正在检查 netcopy 的状态而不是 copy 的状态, 此 status 命令也将使用“copy”关键字。</p> <p>返回的状态信息:</p> <ul style="list-style-type: none"> • SUCCESS (成功) - 成功完成了复制。 • ACTIVE (活动) - 仍然在完成复制进程。 • ERROR <i>number</i> (错误) - 复制失败, 并返回了错误代码 <i>number</i>。
hsb status switch	<p>返回 HotStandby 切换状态信息。返回的状态信息:</p> <ul style="list-style-type: none"> • ACTIVE (活动) - 仍然在完成复制进程。 • SUCCESS (成功) - 成功完成了复制。 • ERROR <i>number</i> (错误) - 复制失败, 并返回了错误代码 <i>number</i>。 • NO SERVER SWITCH OCCURRED BEFORE (先前没有发生服务器切换) - 先前没有发生切换。

命令	说明
hsb switch primary	<p>将数据库服务器切换为 PRIMARY。此命令将启动一个切换进程，可以使用 <code>hsb status switch</code> 命令来监视此进程。</p> <p>如果当您执行此命令时服务器之间已建立连接，那么服务器之间将只是颠倒状态 - 即，旧的主服务器将从 PRIMARY ACTIVE 更改为 SECONDARY ACTIVE 状态，而辅助服务器将从 SECONDARY ACTIVE 切换为 PRIMARY ACTIVE 状态。</p> <p>如果服务器之间未连接，并且服务器处于 SECONDARY ALONE 状态，那么当您将此服务器切换为主服务器时，它最终将处于 PRIMARY ALONE 状态。新的主服务器将不会自动尝试连接至另一个服务器并切换为 PRIMARY ACTIVE 状态。</p> <p>因为在 SECONDARY ACTIVE 和 SECONDARY ALONE 状态下都可以执行此命令，所以可以使用它来执行故障转移。但是，因为此服务器始终将尝试与主服务器通信，所以可能会发生网络超时。因此，使用此方法比使用 <code>'hsb set primary alone'</code> 命令时的速度更慢。另一方面，此方法可以更好地防止存在“双重主服务器”这种可能性。</p> <p>另请参阅 <code>'hsb set primary alone'</code> 命令。</p>
hsb switch secondary	<p>将数据库切换为辅助数据库状态。所有活动的写入事务都将终止。</p> <p>注： 如果所连接的数据库服务器是主服务器，那么它将成为辅助服务器。如果旧的辅助服务器可用，那么此服务器将切换为主服务器（请参阅 <code>hsb switch primary</code> 命令）。</p> <p>注： 如果 <code>switch</code> 命令是在一个打开的事务中发出的（Microsoft Windows，是在启动此事务之后，但是在执行 <code>COMMIT</code> 语句之前发出的），那么当您发出 <code>COMMIT</code> 语句时，<code>COMMIT</code> 将失败并产生一个错误：“重复的事务已异常终止”。将回滚在执行 <code>switch</code> 命令期间产生的所有事务，其中包括执行此 <code>switch</code> 语句的事务。<code>switch</code> 本身是成功的（即，未被回滚），这是因为 <code>ADMIN COMMAND</code> 不是针对事务的命令。（注意：如果尚未打开一个新事务，那么管理命令将强制启动一个新事务。）</p>

高可用性控制器命令

表 30. 高可用性控制器命令

命令	说明
hac shutdown 缩写: hac sd	此命令将终止 HAC 进程。

附录 D. 服务器状态过渡

本章描述可能会发生的状态过渡（例如，从 OFFLINE 过渡到 SECONDARY ALONE）。

第 11 页的『对服务器状态的描述』中对每种服务器状态进行了描述。

HotStandby 状态过渡图

本节中的图显示了可能会发生的状态过渡以及在哪些情况下可能会发生这些状态过渡。

例如，可以通过执行 'hsb set primary alone' 命令将服务器的状态从 PRIMARY UNCERTAIN 更改为 PRIMARY ALONE:

```
ADMIN COMMAND 'hsb Set Primary Alone';
```

当您使用此图时，请记住:

1. 未显示命令的完整语法。例如，它显示:

```
'hsb set primary alone'
```

而不是:

```
ADMIN COMMAND 'hsb Set Primary Alone';
```

2. 为 'hsb copy' 显示的状态过渡路径也适用于 'hsb netcopy'。
3. 执行某些命令时可能会失败。当一个命令既可能成功也有可能失败时，将同时显示这两种可能性。如果打算使用分支来描述命令失败时将发生的情况，那么将使用“失败”这个词: 'Disconnect' 失败。
4. 在某些情况下，行为取决于称为 **AutoPrimaryAlone** 的 solid.ini 配置参数的设置。通常使用缩写“APA”来表示此参数。
5. 当图中提到“事件”时，它指的是在内部生成的通知。这些事件与用户可以公告和等待的“事件”不同，如用于 CREATE EVENT 的 SQL 命令中所述。
6. 靠近此图的左上角，您将看到使用 "-x backupserver" 启动的文本。如果您要启动新的辅助服务器，并且您希望它通过“netcopy”命令从主服务器中获取数据库的副本，那么请从操作系统命令行中使用命令行选项 -x backupserver 来启动此服务器。此命令将让服务器等待从主服务器中发出的 netcopy。请注意，当服务器在等待接收 netcopy 时，此服务器将不会对有关其状态（或角色）的查询作出响应。例如，如果您发出以下命令:

```
ADMIN COMMAND 'hsb state';
```

那么此服务器将不会作出响应，因此您实际上看不到它返回“OFFLINE”状态。

7. “rpc”代表“远程过程调用”。“rpc 已断开”意味着主服务器与辅助服务器之间在未执行显式断开连接的情况下互相失去了连接。如果网络失败或者其中一个服务器已崩溃，就可能会失去连接。
8. 当一个箭头指向它的起始状态时，表示状态不变。例如，如果一个服务器处于 PRIMARY ALONE 状态，并且它尝试连接至另一个服务器但失败了，那么它将保持 PRIMARY ALONE 状态。

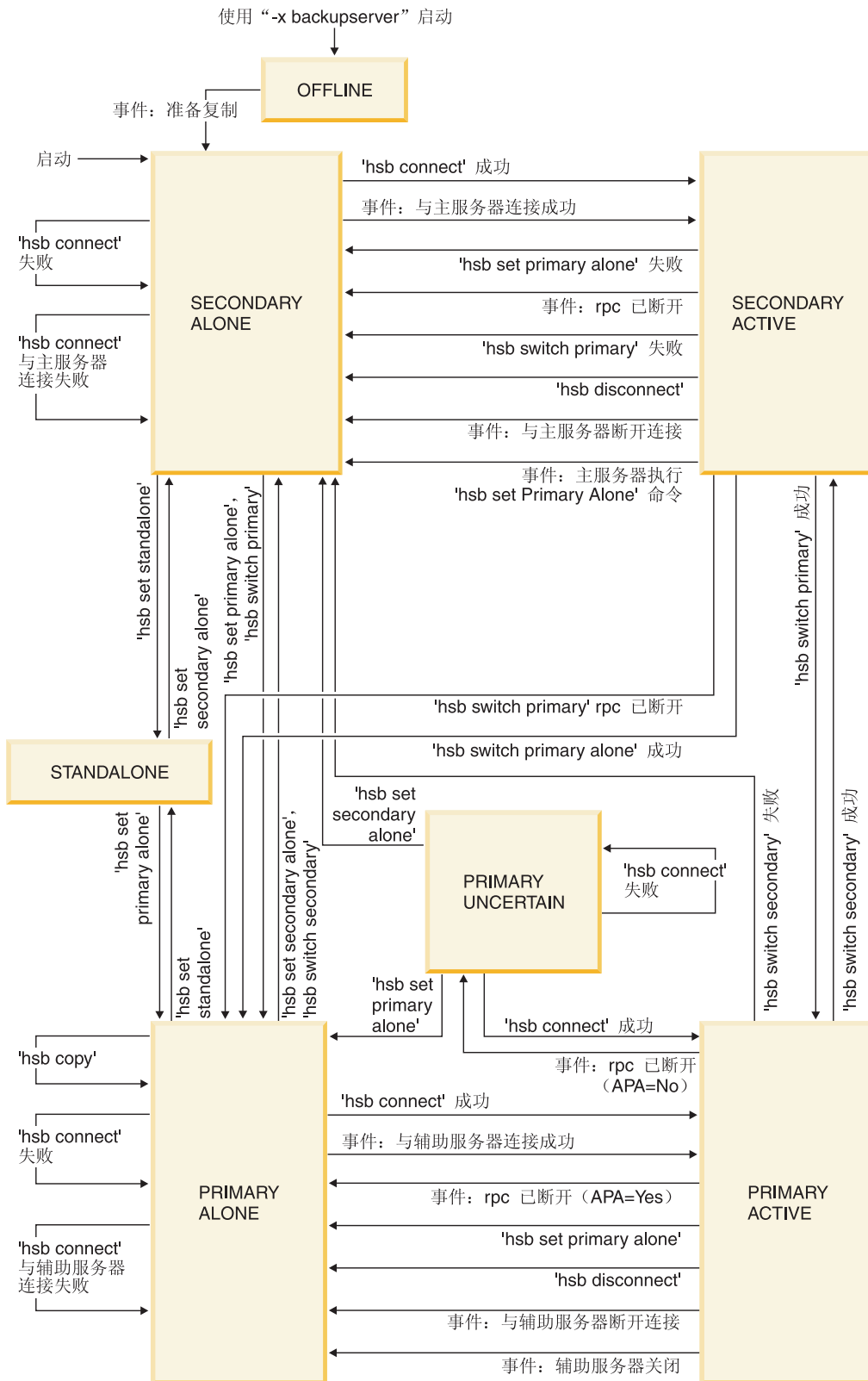


图 15. HotStandby 服务器状态过渡

下表显示了服务器状态以及 HotStandby 命令可以用来更改服务器状态的方式。

表 31. 服务器状态过渡表

服务器状态	如果发生此情况, 或者如果发出了此 HSB 命令...	那么服务器状态将变为...	如果此命令不成功, 那么状态为...
OFFLINE	如果主服务器执行 ADMIN COMMAND 'hotstandby netcopy', 那么在复制数据库之后, 辅助服务器的状态将更改为 SECONDARY ALONE。	SECONDARY ALONE	保持不变
PRIMARY ACTIVE	当 AutoPrimaryAlone = Yes 时发生了 HotStandby 超时 (自动)。 注意: 当辅助服务器关闭或者主服务器与辅助服务器之间的连接断开时, 就会自动产生 HSB 超时。	PRIMARY ALONE	(不适用)
PRIMARY ACTIVE	当 AutoPrimaryAlone = No 时发生了 HotStandby 超时 (自动)。 注意: 当辅助服务器关闭或者主服务器与辅助服务器之间的连接断开时, 就会自动产生 HSB 超时。	PRIMARY UNCERTAIN	(不适用)
PRIMARY ACTIVE	在主服务器中发出 ADMIN COMMAND 'hotstandby set standalone' 命令	STANDALONE	保持不变
PRIMARY ACTIVE	在主服务器中发出 ADMIN COMMAND 'hotstandby switch secondary' 命令, 或者在辅助服务器中发出 ADMIN COMMAND 'hotstandby switch primary'。	SECONDARY ACTIVE	SECONDARY ALONE

表 31. 服务器状态过渡表 (续)

服务器状态	如果发生此情况, 或者如果发出了此 HSB 命令...	那么服务器状态将变为...	如果此命令不成功, 那么状态为...
PRIMARY ACTIVE	在主服务器中发出 ADMIN COMMAND 'hotstandby disconnect'.	PRIMARY ALONE	PRIMARY ALONE
PRIMARY ALONE	<p>在主服务器中发出 ADMIN COMMAND 'hotstandby copy' 或 ADMIN COMMAND 'hotstandby netcopy' 命令。</p> <p>请注意, 主服务器的状态将保持不变。它将保持 PRIMARY ALONE 状态。要将状态更改为 PRIMARY ACTIVE, 必须发出“connect”命令: ADMIN COMMAND 'hotstandby connect';</p> <p>注意: 如果您使用的是无盘服务器, 但是您对辅助服务器没有文件访问权, 那么必须使用 netcopy 而不使用 copy。</p>	PRIMARY ALONE	PRIMARY ALONE
PRIMARY ALONE	<p>在主服务器中发出 ADMIN COMMAND 'hotstandby connect' 命令</p> <p>注意: 以上命令用来连接至现在已修正的辅助服务器, 或者连接至除了失败的辅助服务器之外的其他服务器。</p>	PRIMARY ACTIVE (在完成同步复制之后)	保持不变
PRIMARY ALONE	<p>在主服务器中发出 ADMIN COMMAND 'hotstandby set standalone' 命令, 或者发生事务日志已满的情况。</p>	STANDALONE	保持不变

表 31. 服务器状态过渡表 (续)

服务器状态	如果发生此情况, 或者如果发出了此 HSB 命令...	那么服务器状态将变为...	如果此命令不成功, 那么状态为...
PRIMARY ALONE	在主服务器中发出 ADMIN COMMAND 'hotstandby set secondary alone' 或者 ADMIN COMMAND 'hotstandby switch secondary'。	SECONDARY ALONE	SECONDARY ALONE
PRIMARY UNCERTAIN	在主服务器中发出 ADMIN COMMAND 'hotstandby set primary alone' 命令	PRIMARY ALONE	保持不变
PRIMARY UNCERTAIN	在主服务器中发出 ADMIN COMMAND 'hotstandby connect' 命令。 注: 以上命令用来连接至现在已修正的辅助服务器, 或者连接至除了失败的辅助服务器之外的其他服务器。	PRIMARY ACTIVE	保持不变
PRIMARY UNCERTAIN (连接至辅助服务器时发生了 HSB 超时)	在主服务器中发出 ADMIN COMMAND 'hotstandby set standalone' 命令	STANDALONE	保持不变
PRIMARY UNCERTAIN	在主服务器中发出 ADMIN COMMAND 'hotstandby set secondary alone' 或者 ADMIN COMMAND 'hotstandby switch secondary'。	SECONDARY ALONE	保持不变
SECONDARY ACTIVE	HSB 超时 (自动) 注: 当辅助服务器关闭或者主服务器与辅助服务器之间的连接断开时, 就会自动产生 HSB 超时。	SECONDARY ALONE	(不适用)

表 31. 服务器状态过渡表 (续)

服务器状态	如果发生此情况, 或者如果发出了此 HSB 命令...	那么服务器状态将变为...	如果此命令不成功, 那么状态为...
SECONDARY ACTIVE	在主服务器中发出 ADMIN COMMAND 'hotstandby switch secondary' 命令, 或者在辅助服务器中发出 ADMIN COMMAND 'hotstandby switch primary'。	PRIMARY ACTIVE	保持不变
SECONDARY ACTIVE	在辅助服务器中发出 ADMIN COMMAND 'hotstandby set primary alone' 命令。	PRIMARY ALONE	保持不变
SECONDARY ACTIVE	在辅助服务器或主服务器中发出 ADMIN COMMAND 'hotstandby disconnect' 命令。	SECONDARY ALONE	SECONDARY ALONE
SECONDARY ALONE	在辅助服务器或主服务器中发出 ADMIN COMMAND 'hotstandby connect' 命令。	SECONDARY ACTIVE	保持不变
SECONDARY ALONE	在辅助服务器中发出 ADMIN COMMAND 'hotstandby set standalone' 命令。	STANDALONE	保持不变
SECONDARY ALONE	在辅助服务器中发出 ADMIN COMMAND 'hotstandby set primary alone' 或者 ADMIN COMMAND 'hotstandby switch primary' 命令。	PRIMARY ALONE	保持不变

附录 E. HotStandby 系统事件

本附录只包含特定于 HSB 的事件。有关其他类型的事件的讨论，请参阅其他手册，例如《solidDB SQL 指南》。

每个 HotStandby 操作都将生成一个事件。可以使用诸如看守程序等应用程序来监视这些事件。

事件通过其名称来反映在服务器中发生了特定操作。需要在存储过程中使用特殊语句来接收事件。HotStandby 事件与 solidDB 创建和支持的其他事件没有区别。这些事件将发送给已注册为将在存储过程中接收事件的用户。有关公布、注册和等待事件的详细信息，请阅读《solidDB SQL 指南》中的『存储过程、事件、触发器和序列』以及『solidDB SQL 语法』。

下表列示了当前可用于 HotStandby 的事件。请注意，大多数事件都包含五个参数，但并不是必需使用所有这些参数。

表 32. HotStandby 事件

HSB 事件	事件参数	事件原因
SYS_EVENT_HSBCONNECTSTATUS	ENAME WVARCHAR、POSTSRVTIME TIMESTAMP、UID INTEGER、NUMDATAINFO INTEGER 和 TEXTDATA WVARCHAR 对于 TEXTDATA，有效值可能是： TEXTDATA = { CONNECTED CONNECTING CATCHUP BROKEN}	主服务器与辅助服务器之间的连接状态更改
SYS_EVENT_HSBSTATESWITCH	ENAME WVARCHAR、POSTSRVTIME TIMESTAMP、UID INTEGER、NUMDATAINFO INTEGER 和 TEXTDATA WVARCHAR 对于 TEXTDATA，有效值可能是： TEXTDATA = { PRIMARY ACTIVE PRIMARY ALONE PRIMARY UNCERTAIN SECONDARY ACTIVE SECONDARY ALONE STANDALONE }	每次状态切换都会发送一个状态切换事件。

表 32. HotStandby 事件 (续)

HSB 事件	事件参数	事件原因
SYS_EVENT_NETCOPYEND	ENAME WVARCHAR、POSTSRVTIME TIME STAMP、UID INTEGER、NUMDATAINFO INTE- GER 和 TEXTDATA WVARCHAR 不使用参数。	HotStandby NETCOPY 操作已结束。 仅当用户使用链接库访问时，用户才能 捕获到此事件。
SYS_EVENT_NETCOPYREQ	ENAME WVARCHAR、POSTSRVTIME TIME STAMP、UID INTEGER、NUMDATAINFO INTE- GER 和 TEXTDATA WVARCHAR 不使用参数。	请求了 HotStandby NETCOPY。 如果用户应用程序回调函数返回非零 值，那么将不执行网络复制。 仅当用户使用链接库访问时，用户才能 捕获到此事件。

附录 F. 看守程序样本

本节讨论 solidDB 安装中包含的样本中提供的看守程序样本应用程序。

看守程序是一个独立程序，用于监视和控制主服务器和辅助服务器。看守程序将同时监视两个热备用服务器，并在必要时切换它们的状态。这就减轻了数据库管理员需要监视服务器的负担。

solidDB 提供了一个样本看守程序，可以将它用作构建能够满足您需要的定制看守程序的基础。此样本应用程序称为“看守程序”。在开始编程之前，请注意“看守程序”样本的下列功能。

- 打算将“看守程序”用作看守程序示例
- 看守程序使用轮询来保持它本身对于服务器状态是最新的
- 看守程序是一个单线程程序
- 看守程序通过 ODBC 使用 HSB API。可以将此 API 实现用作您自己的看守程序的模型。
- 看守程序没有用户界面。

如果您要使用看守程序，那么需要配置 solidDB 配置文件 (solid.ini) 中的 [WatchDog] 节，此文件位于看守程序的当前工作目录中。如果看守程序正在与主服务器或辅助服务器在同一目录中运行，那么您将只有一个 solid.ini 文件，服务器和看守程序将共享此文件。如果看守程序在一个单独的目录中运行，那么看守程序将具有它自己的 solid.ini 文件。

而且，此附录中还说明了特定于看守程序的 solid.ini 配置参数。这些参数是在 solid.ini 配置文件的 [WatchDog] 节中设置的。如果您编写自己的看守程序，那么不需要使用任何这些参数。

有关其他 solid.ini 参数的讨论，请参阅《solidDB 管理员指南》。

使用看守程序的 HotStandby 配置

HotStandby 配置允许主服务器、辅助服务器和看守程序位于不同机器中并且使用不同的操作系统和 API，如第 145 页的『系统设计问题』中的示例所示。有关实现异构配置的详细信息，请阅读第 145 页的『系统设计问题』。

主数据库与辅助数据库之间的所有通信（包括使失败的系统恢复运行并且使主数据库与辅助数据库重新同步）都是在现有通信层（例如，TCP/IP）进行的。HotStandby 不需要使用辅助存储器或传输方法（例如，共享磁盘或 FTP 传输）。

要点：如果您正在辅助服务器所在机器上运行看守程序，那么务必将 **AutoPrimaryAlone** 参数设置为 no。在这种情况下，将 **AutoPrimaryAlone** 设置为 no 非常重要，这是因为它可以防止发生存在两个主服务器这种错误。主服务器可能处于 PRIMARY ALONE 状态，并且在发生服务器故障时看守程序可以将辅助服务器切换为

PRIMARY ALONE 状态。如果用户碰巧将旧的辅助服务器设置为新的主服务器，那么也可能发生此错误。有关双重主服务器的更多信息，请参阅第 35 页的『网络分区和双重主服务器』。

看守程序的工作方式

当主服务器关闭时，看守程序样本应用程序将通知您。在正常方式下，看守程序将在主服务器和辅助服务器中使用 `hotstandby status connect` 命令来检查服务器的连接状态。

看守程序定期在服务器之间执行此检查。时间间隔是在看守程序的 `solid.ini` 配置文件中的 **PingInterval** 参数设置的。

在尝试轮询给定次数之后，如果看守程序未从主服务器或/和辅助服务器节点中接收到响应，它就会认为 HotStandby 系统中发生了问题。尝试轮询的次数是在看守程序配置文件（`solid.ini` 文件中的 [Watchdog] 节）中的 **NumRetry** 参数中设置的。

看守程序还会观察主服务器和辅助服务器之间是否相连。如果主服务器或辅助服务器向看守程序返回成功连接状态，那么意味着主服务器和辅助服务器仍然相连。另一方面，如果返回一个错误，那么意味着主服务器和辅助服务器不再相连。

如果看守程序配置文件中的 **AutoSwitch** 参数设置为 YES，那么看守程序还将负责在主服务器发生故障时自动切换服务器的状态。例如，当主服务器关闭时，看守程序就会切换辅助服务器以使它成为新的主服务器并将它置于 PRIMARY ALONE 状态。如果 **AutoSwitch** 参数设置为 NO，看守程序本身不会更改服务器状态，但它会将一条消息写入看守程序日志以通知用户切换服务器状态。

要继续监视，看守程序将切换到故障方式，这意味着它将连续检查失败的服务器是否具有可以工作的连接。

故障方式

当看守程序样本应用程序知道 HotStandby 主服务器和辅助服务器已经相连时，看守程序将保持正常方式。如果其中一个服务器已失败，或者服务器之间的通信链路已失败，那么看守程序就会执行一些操作。如果执行操作之后未能使服务器相连，那么看守程序将进入故障方式。

在看守程序进入故障方式之后，它将等待系统管理员解决主服务器和辅助服务器存在的问题。同时，如果再次发生故障，看守程序不会处理此故障。看守程序中的这种局限性是故意这样设计的。在某些情况下，一系列故障甚至表面上看起来合适的响应都可能导致“具有两个主服务器”（处于 PRIMARY ALONE 或 STANDALONE 状态）这种错误。如果网络中短时间内发生故障，但是数据库服务器本身未发生故障，就特别容易产生这种错误。第 145 页的『编写用于处理多个故障的看守程序』中提供了产生两个主服务器的示例。

在故障方式下，看守程序将同时轮询主服务器和辅助服务器。当它能够连接至这两个服务器时，就会对这两个服务器发送 `hotstandby state` 命令，以了解它是否可与这两个服务器通信以及每个服务器所处的状态。

一旦看守程序能够与这两个服务器通信，它就会根据 `solid.ini` 参数 **DualSecAutoSwitch** 来决定下一步要执行的操作。如果 **DualSecAutoSwitch** = Yes 并且两个服务器都是辅助服务器，那么看守程序将自动选择其中一个辅助服务器作为新的主服务器并将它切

换为主服务器。如果 **DualSecAutoSwitch** = No, 那么系统管理员必须将一个服务器切换为主服务器。请注意, 无论看守程序处于“正常”方式还是“故障”方式, **DualSecAutoSwitch** 参数都适用。

编写用于处理多个故障的看守程序

看守程序中可以采用两种方法来处理多个故障。您可以:

- 在每次发生故障(并由看守程序自动作出响应)之后, 需要手动(人为)干预以检查所发生的情况。手动干预时可能需要执行一些操作, 例如, 重新启动服务器或者解决网络问题。看守程序通常也是采用这种方法, 因为这样做可以降低存在两个主服务器的风险。
- 编写一个看守程序, 可以用来处理随着时间推移而产生的多个故障。

此方法确实存在产生两个主服务器的风险, 如以下示例中所示。

双重主服务器

在此示例中, Server1 最初是主服务器, Server2 最初是辅助服务器。

1. 发生了网络故障, Server1 变得不可访问。
2. 看守程序将 Server2 从 SECONDARY 状态切换为 PRIMARY ALONE 状态。
3. 发生了第二个网络故障, Server2 变得不可访问。
4. 修复了第一个网络故障, Server1 再次变得可访问。
5. 由于 Server1 可访问, 而 Server2 不可访问, 因此看守程序将 Server1 切换为 PRIMARY ALONE 状态。
6. 修复了第二个网络故障, Server2 再次变得可访问。
7. 此时, Server1 和 Server2 都处于 PRIMARY ALONE 状态。

系统设计问题

您用于配置 HotStandby 的方式(本地或远程配置, 在一个或多个不同位置配置, 通过因特网配置, 使用看守程序来配置)会影响系统的可靠性和效率。本节的重点就是解决这些问题。

下图显示了一个异构系统的示例。在此系统中, 主服务器和辅助服务器甚至不使用同一类型的硬件和操作系统。

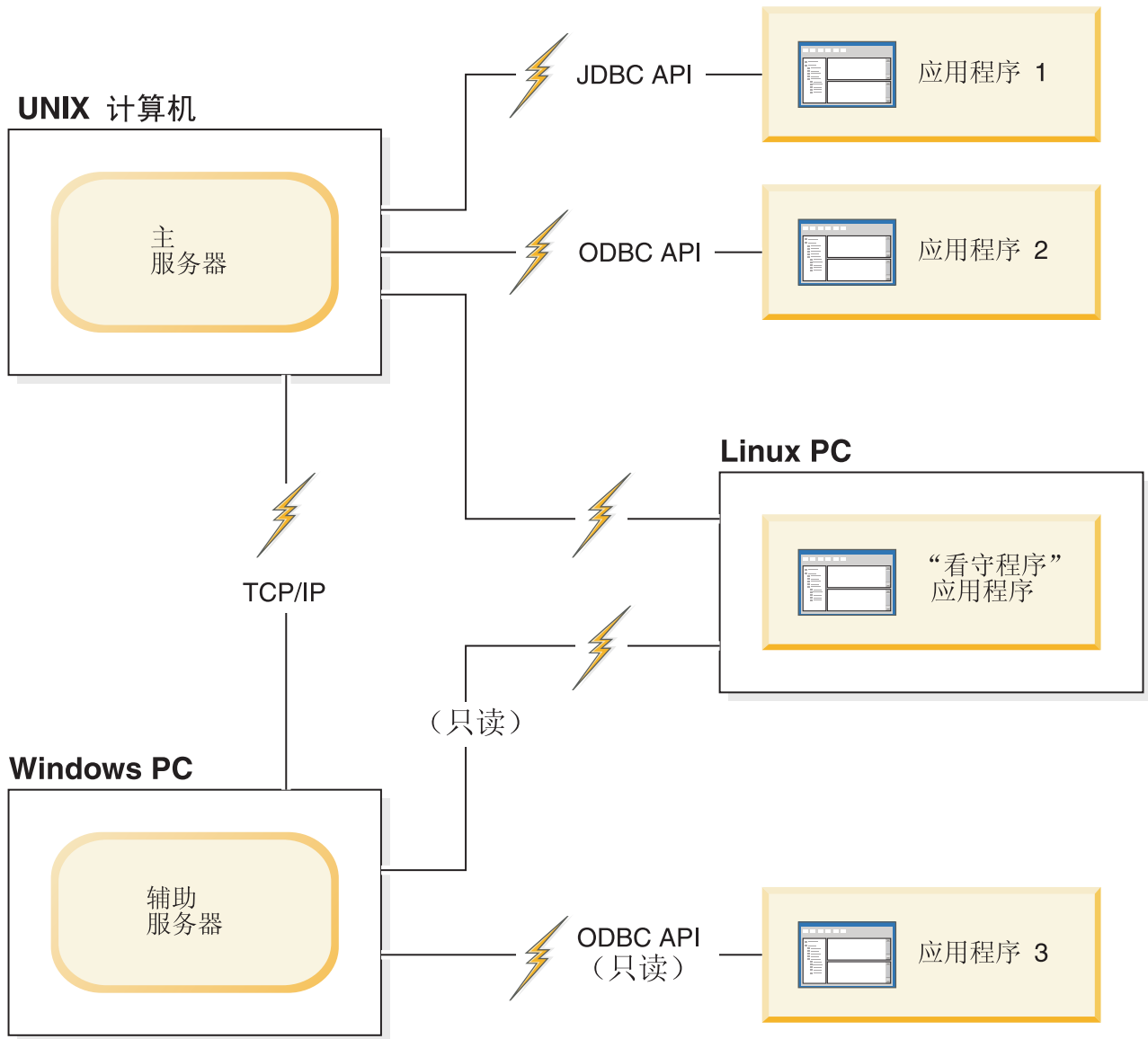


图 16. 使用了看守程序的异构 HotStandby 配置

看守程序配置

为了更有效率并且更准确地监视服务器状态，建议将看守程序作为任何 HotStandby 配置中的独立组件。

如果只有两台机器可用，那么不可能单独使用一台机器来运行看守程序。因此，在辅助服务器所在的机器上运行看守程序，并在主服务器和辅助服务器的 `solid.ini` 配置文件中将 **AutoPrimaryAlone** 参数设置为 `no`。请注意，将此参数设置为“no”非常重要，因为它可以防止发生存在两个主服务器的错误。

注意:

如果两个服务器都处于一种允许写入的状态（**PRIMARY ALONE** 或 **STANDALONE** 状态），并且两个服务器的数据库是单独更新的，那么将无法重新使这两个数据库同步。请确保看守程序不允许两个服务器同时处于 **PRIMARY ALONE** 或 **STANDALONE** 状态。请参阅第 35 页的『网络分区和双重主服务器』。

如果主服务器确实失败了，那么看守程序能够将辅助服务器切换为新的主服务器。

将看守程序与辅助服务器放在同一台机器中存在一些缺点。这些缺点包括:

- 如果只是看守程序与主服务器之间的通信链路断开了，那么此配置可能会导致主服务器与辅助服务器之间进行假切换。
- 通信链路成为“单一故障点”，即，可能会禁用整个系统的单一故障。（在大多数 HotStandby 配置中，除非至少发生了两个故障，否则不会禁用整个系统。）
- 即使发生了网络故障并且辅助服务器所在的机器无法与主服务器所在的机器通信，用户和应用程序仍然能够访问主服务器，并且从理论上讲可以继续对主服务器执行操作。但是，主服务器将停止接受事务，这是因为看守程序无法通知主服务器继续运行，例如，通过切换为 **PRIMARY ALONE** 状态而继续运行。

使用样本看守程序

关于此任务

最初，应当在两个服务器都已启动并且相连的情况下启动看守程序。

要启动看守程序，切换至看守程序的当前工作目录，并在命令提示符处发出以下命令:

```
watchdog
```

如果您在 `solid.ini` 文件尚未对 `connect1` 和 `connect2` 服务器（它们都能够用作主服务器和辅助服务器）指定用户名和密码，那么看守程序将提示您指定这些用户名和密码。

结果

启动看守程序之后，它就会对两个服务器执行 `ping` 操作以检查哪个服务器是主服务器。除非在重试次数超过指定次数之后看守程序检测到服务器故障，否则看守程序将保持正常方式。如果在看守程序最后一次向服务器发送重试尝试之后还是发生了故障，那么看守程序就会切换到故障方式。一旦主服务器和辅助服务器都已启动并且已重新连接，看守程序就会切换到正常方式。

故障情况和看守程序操作

本节描述典型的看守程序应当如何处理通常遇到的特定故障情况。

这些情况是发生服务器故障或者主服务器与辅助服务器之间或者其中一个服务器与看守程序之间的通信链路断开时发生的。

尽管管理员或软件程序都可以发出这些命令，但是为了简单起见，我们假定这些命令是由看守程序样本发出的。

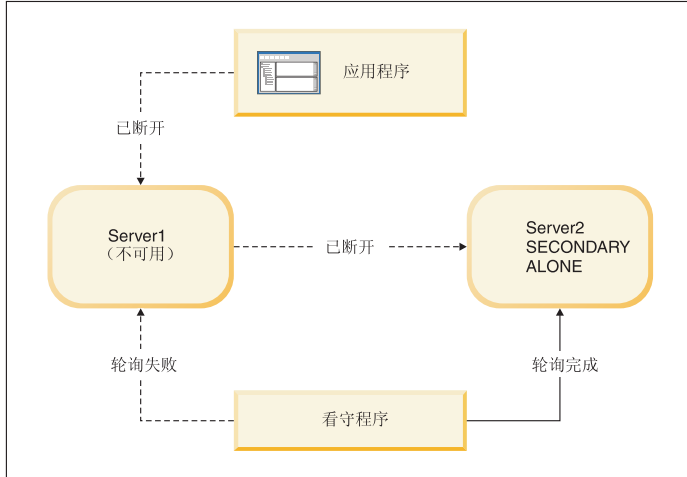
主服务器已关闭 情况

与主服务器的所有连接都已断开。

补救措施

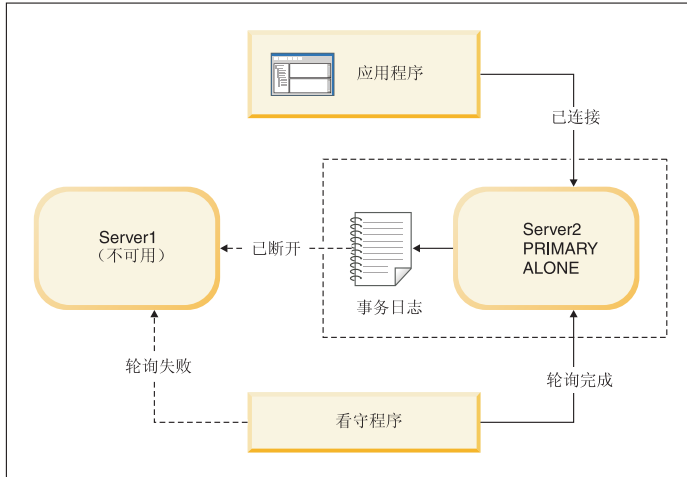
当主服务器已关闭时，将辅助服务器切换为新的主服务器并将新的主服务器设置为 PRIMARY ALONE 状态。然后，旧的主服务器可以成为新的辅助服务器。

故障情况



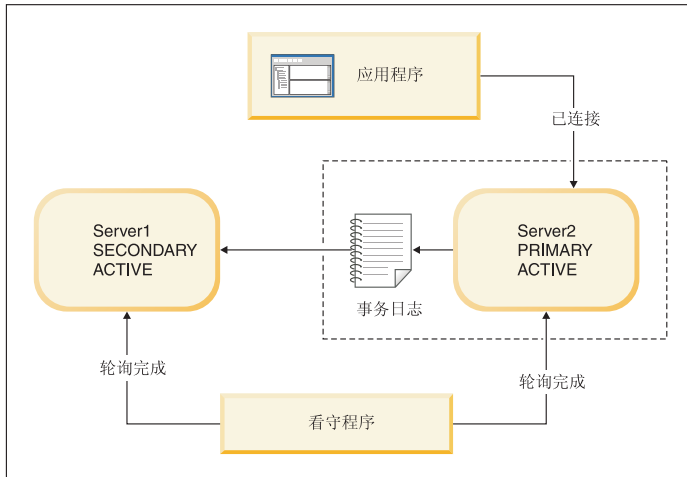
1

故障转移/补救措施



2

解决方案



3

图 17. 主服务器已关闭的情况和补救措施

1. 看守程序将指示 Server2 执行以下操作:

HSB SET PRIMARY ALONE

应用程序从 Server1 切换到 Server2。

HSB SET STANDALONE

2. 在修复 Server1 之后，Server1 将作为辅助服务器运行。

看守程序将指示 Server2 执行以下操作：HSB COPY 或 NETCOPY HSB CONNECT

3. 如果事务日志已填满，那么可能需要将 PRIMARY ALONE 服务器切换为 STANDALONE。在这种情况下，在重新连接服务器之前您还需要执行 HSB COPY 或 HSB NETCOPY。如果事务日志未填满，那么您必须跳过 COPY/NETCOPY 命令。

症状

应用程序无法连接至主服务器。此外，看守程序在主服务器上轮询失败。辅助服务器的 HSB 状态为 SECONDARY ALONE。

在主服务器已关闭的情况下如何恢复

关于此任务

为了允许“HotStandby”（辅助服务器）替换主服务器，请执行下列操作：

1. 使用以下命令将新的主服务器设置为 PRIMARY ALONE 状态：

```
ADMIN COMMAND 'hotstandby set primary alone';
```

2. 将应用程序重新连接至新的主服务器。
3. 开始使用应用程序。
4. 修复旧的主服务器，并将它作为新的辅助服务器来启动。
5. 必要时，使用以下命令将数据库从新的主服务器复制到新的辅助服务器：

```
ADMIN COMMAND 'hotstandby netcopy';
```

有关详细信息，请阅读第 78 页的『使主服务器与辅助服务器同步』。

6. 使用以下命令将新的主服务器重新连接至新的辅助服务器：

```
ADMIN COMMAND 'hotstandby connect';
```

辅助服务器已关闭

情况

与辅助服务器的所有连接都已断开。这可能是由于辅助服务器中发生故障，或者网络故障使得主服务器或者看守程序无法与辅助服务器通信而导致的。在本节中，认为辅助服务器已失败，但实际上问题可能与辅助服务器或网络有关。

补救措施

标准补救措施是将主服务器切换为 PRIMARY ALONE 状态。在辅助服务器再次启动之后，使它与主服务器同步。

当发现与辅助服务器的连接存在问题时，主服务器将执行下列操作：

1. 暂挂已打开的任何事务，既不落实它们，也不回滚它们（主服务器不会向客户机发送错误消息或者“成功”消息）；并且

2. 自动将它自己的状态从 PRIMARY ACTIVE 切换为 PRIMARY UNCERTAIN。

通常，在确定辅助服务器不可用之后，看守程序会将主服务器从 PRIMARY UNCERTAIN 状态切换为 PRIMARY ALONE 状态。在主服务器切换为 PRIMARY ALONE 状态之后，它可以继续接受事务并保持它们以便发送至辅助服务器。稍后，当辅助服务器再次运行时，可以向辅助服务器发送事务日志，以使它可与主服务器之间进行同步复制。

将主服务器设置为 PRIMARY ALONE 状态之后，主服务器将落实打开的事务。为了避免在辅助服务器尚未落实事务的情况下主服务器就落实事务这种可能性，会将事务保留在事务日志中，就好像从未将它们发送至辅助服务器一样。当辅助服务器重新运行并且开始进行同步复制时，主服务器将发送该事务日志，并且辅助服务器将检查每个事务。如果任何事务是重复的（即，在辅助服务器失败之前，辅助服务器已经落实了此事务），那么在辅助服务器上不会重新执行重复的事务。

看守程序或系统管理员在选择是否使主服务器处于 PRIMARY ALONE 状态或者选择替代操作时必须小心。如果看守程序或系统管理员选择的不是将主服务器切换为 PRIMARY ALONE 状态，那么必须考虑辅助服务器和主服务器可能不具有相同的数据，即，它们可能并不是都已回滚了事务。有可能失败的辅助服务器实际上已经落实了数据，但是在落实数据之后、在向主服务器发送确认信息之前崩溃了，而主服务器从未落实数据。在这种情况下，辅助服务器实际上可能比主服务器先落实数据，而不是在它之后落实数据。

同样，看守程序或管理员还必须小心，请不要允许两个服务器同时都进入 PRIMARY ALONE 状态。

下图分成三个框。第一个框显示了发生的情况，也就是主服务器和看守程序已经与辅助服务器失去联系。下一个框显示了如何让系统继续保持工作，直到彻底解决了问题为止。第三个框显示了解决问题之后（即，在已经修复了已断开的服务器之后，或者在恢复通信之后）的最终状态。

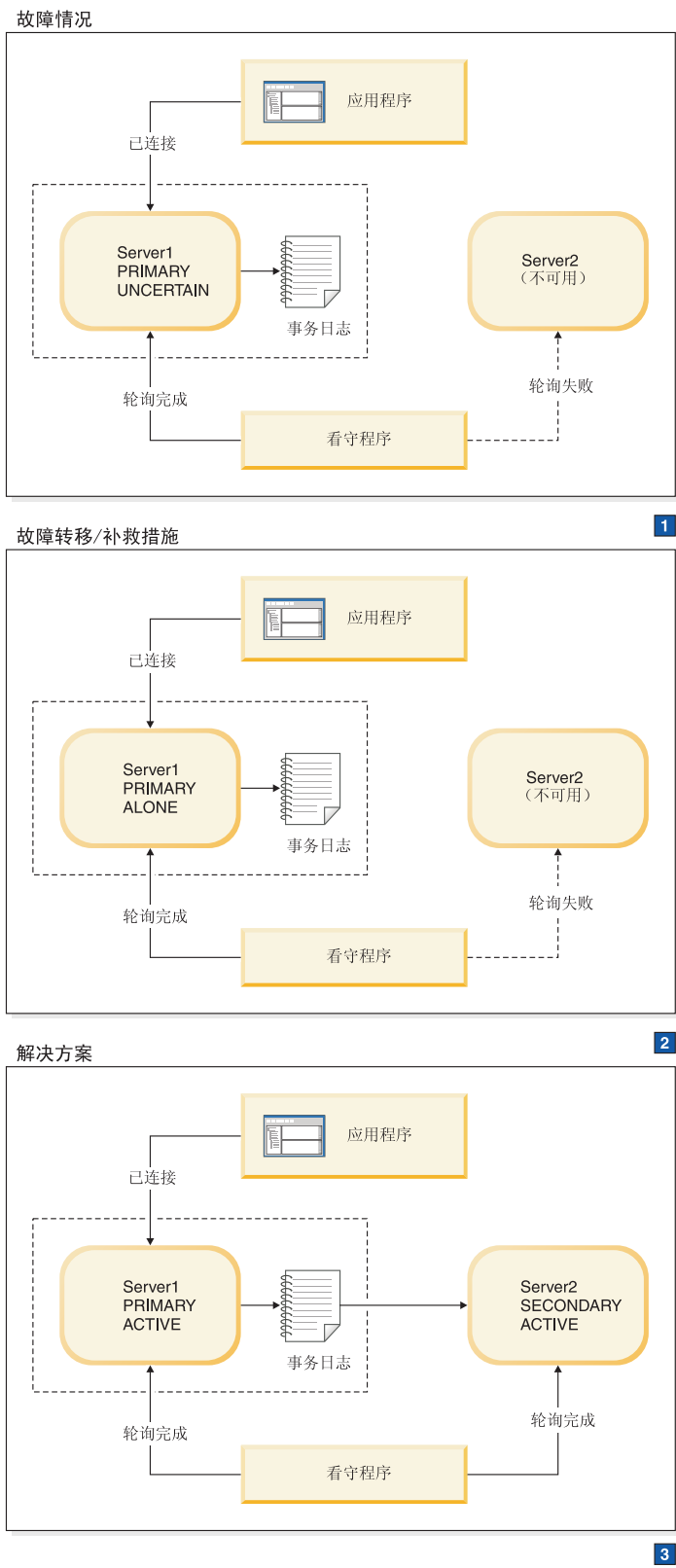


图 18. 辅助服务器已关闭的情况和补救措施

1. 看守程序将指示 Server1 执行以下操作:

HSB SET PRIMARY ALONE

HSB SET STANDALONE

- 在 Server2 重新开始运行之后，看守程序将指示 Server1 执行以下操作:

HSB COPY 或者

NETCOPY HSB CONNECT

- 如果事务日志已填满，那么可能需要将 PRIMARY ALONE 服务器切换为 STANDALONE。在这种情况下，在重新连接服务器之前您还需要执行 HSB COPY 或 HSB NETCOPY。如果事务日志未填满，那么您必须跳过 COPY/NETCOPY 命令。

症状

看守程序在辅助服务器上轮询失败。主服务器的状态为 PRIMARY ALONE 或 PRIMARY UNCERTAIN。

在辅助服务器已关闭的情况下如何恢复

关于此任务

为了允许主服务器继续接收事务并且独立于辅助服务器运行，请执行下列操作:

- 如果主服务器处于 PRIMARY UNCERTAIN 状态，那么使用以下命令将主服务器设置为 PRIMARY ALONE 状态:

```
ADMIN COMMAND 'hotstandby set primary alone';
```

- 在已经修复并且重新启动辅助服务器以及/或者重新建立了辅助服务器的网络连接之后，请使用以下命令检查主服务器的状态:

```
ADMIN COMMAND 'hotstandby state';
```

- 如果主服务器的状态为 PRIMARY ALONE，那么请使用以下命令将主服务器重新连接至辅助服务器:

```
ADMIN COMMAND 'hotstandby connect';
```

- 如果主服务器的状态先前已更改为 STANDALONE，那么请执行下列操作:

- 使用以下命令将数据库从新的主服务器复制到辅助服务器:

```
ADMIN COMMAND 'hotstandby netcopy';
```

- 有关详细信息，请阅读第 78 页的『使主服务器与辅助服务器同步』。

- 使用以下命令将主服务器重新连接至辅助服务器:

```
ADMIN COMMAND 'hotstandby connect';
```

当辅助服务器已关闭时的其他情况

如果应用程序从主服务器中接收到错误消息 10047 或 14537:

- 尝试连接至辅助服务器以检查它的状态是否已切换为新的主服务器。
- 如果它的状态不是 PRIMARY ACTIVE 或 PRIMARY ALONE，那么请参阅第 148 页的『主服务器已关闭』中的情况。

看守程序已关闭

本节说明看守程序失败时发生的情况。

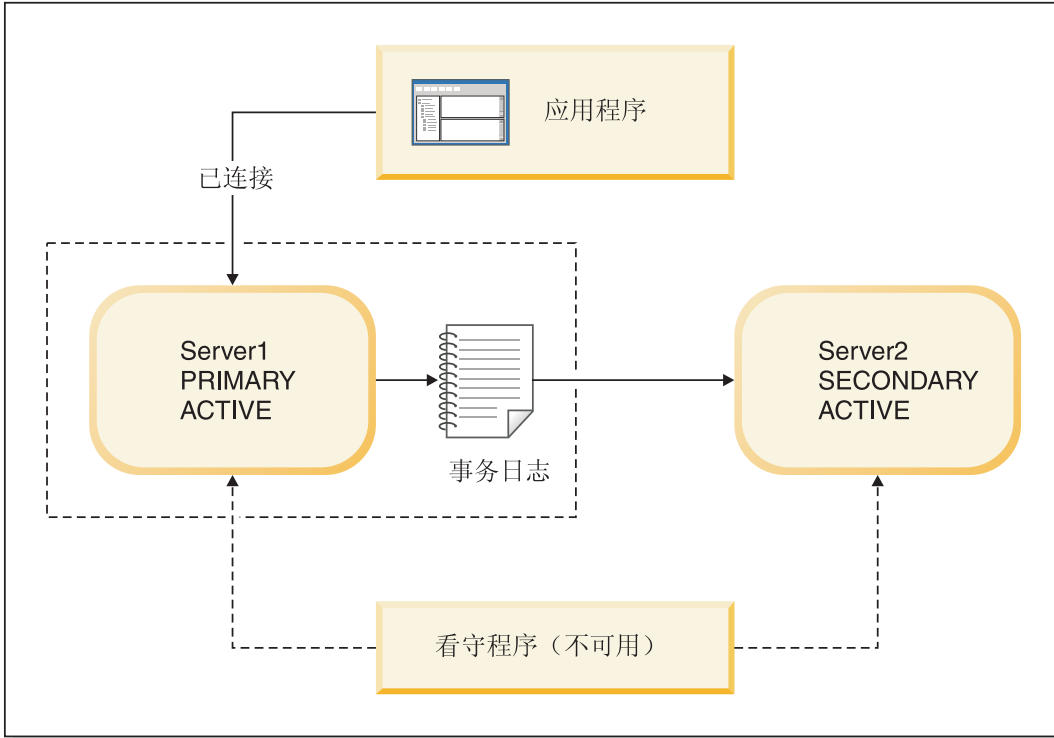
情况

与看守程序的所有连接都已断开。

补救措施

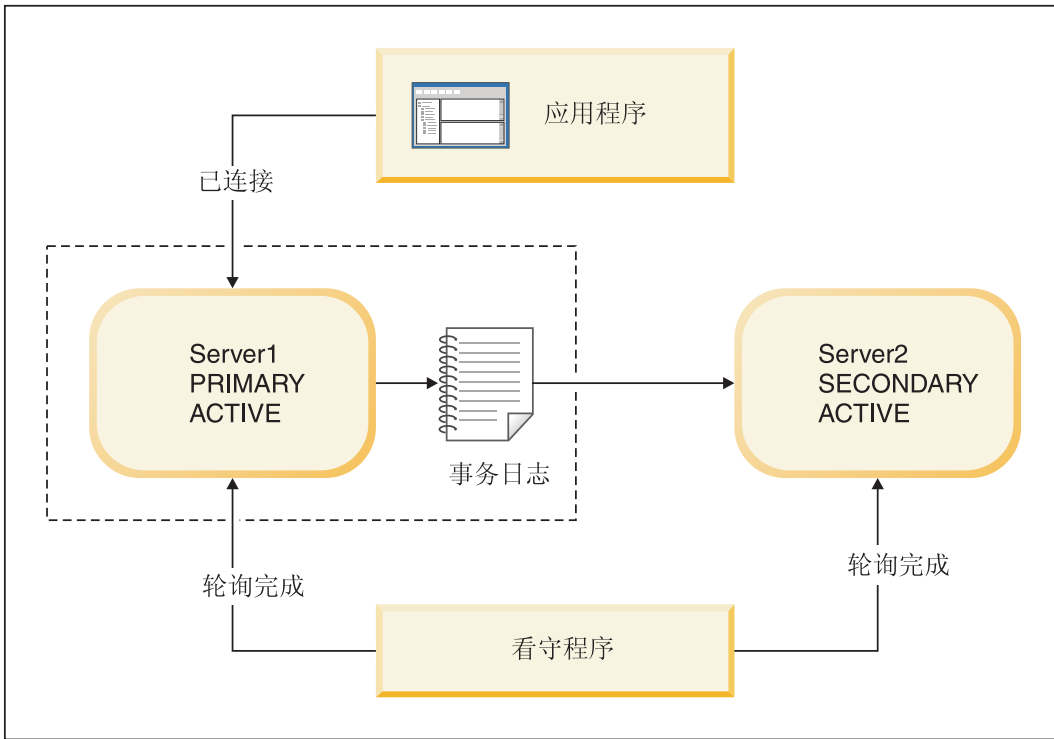
需要手动干预。当看守程序启动之后，务必要检查主服务器和辅助服务器以确认它们的状态。

故障情况



解决方案

1



1. 使看守程序重新开始运行或者修复网络。

图 19. 看守程序已关闭的情况和补救措施

症状

看守程序进程已关闭，或者看守程序与两个服务器的网络连接不可用。

其他情况

如果两个服务器已更改状态并且一个服务器不再工作，那么请参阅本节中的适当情况以获取指示信息。

在看守程序已关闭的情况下如何恢复

关于此任务

要从与看守程序的所有连接都已断开的情况下恢复：

1. 允许主服务器和辅助服务器继续正常操作。
2. 一旦看守程序已启动，就让它通过以下命令检查每个服务器的状态：

```
ADMIN COMMAND 'hotstandby state';
```

主服务器和辅助服务器之间的通信链路已断开情况

主服务器与辅助服务器之间的连接已断开。

主服务器会将它本身切换为 PRIMARY UNCERTAIN 状态。（如果 **AutoPrimaryAlone** 设置为 Yes，那么服务器会将它本身切换为 PRIMARY ALONE 状态。）

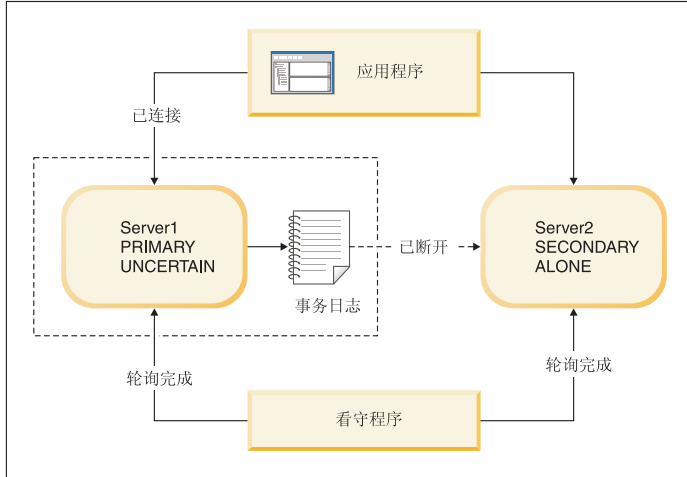
注：如果主服务器将落实消息发送至辅助服务器，然后检测辅助服务器的故障，那么主服务器将依赖看守程序或管理员来指示主服务器如何继续操作。这是因为在辅助服务器失败之前，主服务器将无法检测在辅助服务器中是否落实或回滚了事务。

在主服务器接收到来自看守程序或管理员的命令之前，它不再接受事务。在此阶段，为了让主服务器继续运行，看守程序或管理员可以将主服务器设置为 PRIMARY ALONE 状态。

补救措施

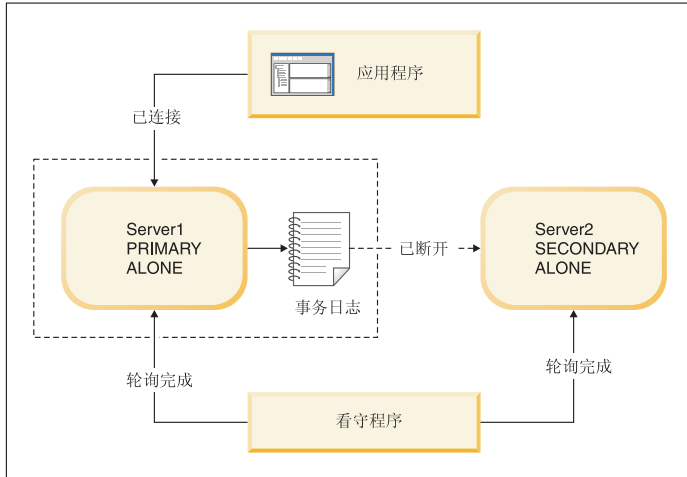
即使主服务器与辅助服务器之间的链路已断开，主服务器仍然可以继续运行。如果主服务器尚不处于 PRIMARY ALONE 状态，那么请将主服务器切换为 PRIMARY ALONE 状态。一旦恢复了主服务器与辅助服务器之间的链路，就使数据库同步。

故障情况



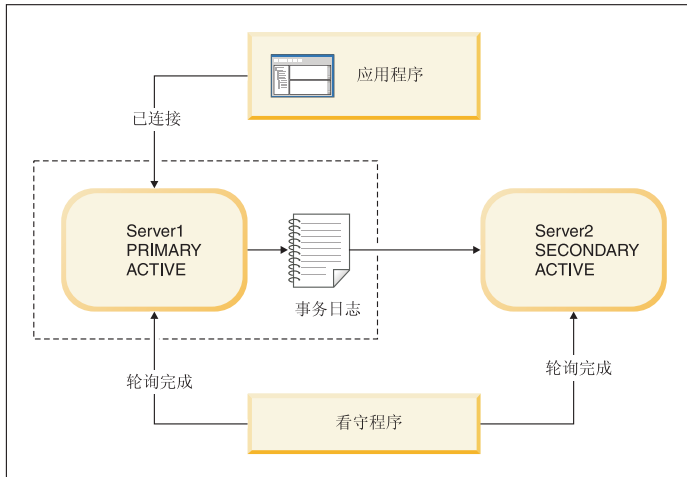
1

故障转移/补救措施



2

解决方案



3

图 20. 主服务器与辅助服务器之间的链路已断开的情况和补救措施

1. 看守程序将指示 Server1 执行以下操作:

HSB SET PRIMARY ALONE

HSB SET STANDALONE

2. 在修复主服务器与辅助服务器之间的连接之后，看守程序将指示 Server1 执行以下操作：

HSB COPY 或者

NETCOPY HSB CONNECT

3. 如果事务日志已填满，那么可能需要将 PRIMARY ALONE 服务器切换为 STANDALONE。在这种情况下，在重新连接服务器之前您还需要执行 HSB COPY 或 HSB NETCOPY。如果事务日志未填满，那么您必须跳过 COPY/NETCOPY 命令。

症状

主服务器未与辅助服务器连接，并且状态为 PRIMARY UNCERTAIN 或 PRIMARY ALONE。

当主服务器与辅助服务器之间的通信链路已断开的情况下如何恢复 关于此任务

要从主服务器与辅助服务器之间的连接已断开的情况下恢复：

1. 修复主服务器与辅助服务器之间的网络连接。
2. 使用以下命令检查主服务器的状态：

```
ADMIN COMMAND 'hotstandby state';
```
3. 如果主服务器的状态为 PRIMARY ALONE，那么请使用以下命令将主服务器重新连接至辅助服务器：

```
ADMIN COMMAND 'hotstandby connect';
```
4. 如果主服务器的状态为 STANDALONE，那么请执行以下操作：
 - a. 将数据库从主服务器复制到辅助服务器。有关详细信息，请阅读第 78 页的『使主服务器与辅助服务器同步』。

在使用 ADMIN COMMAND 'hotstandby netcopy'; 命令之前，请确保辅助服务器已启动并且正在运行，并且已准备好接收 netcopy 命令。另外，务必将主服务器的状态设置为 PRIMARY ALONE。

- b. 使用以下命令将主服务器重新连接至辅助服务器：

```
ADMIN COMMAND 'hotstandby connect';
```

当主服务器与辅助服务器之间的通信链路已断开时的其他情况

如果应用程序从主服务器中接收到错误消息 10047 或 14537：

- 尝试连接至辅助服务器以检查它是否已切换为新的主服务器。
- 如果旧的辅助服务器未切换为新的主服务器，那么请参阅第 148 页的『主服务器已关闭』中的情况。

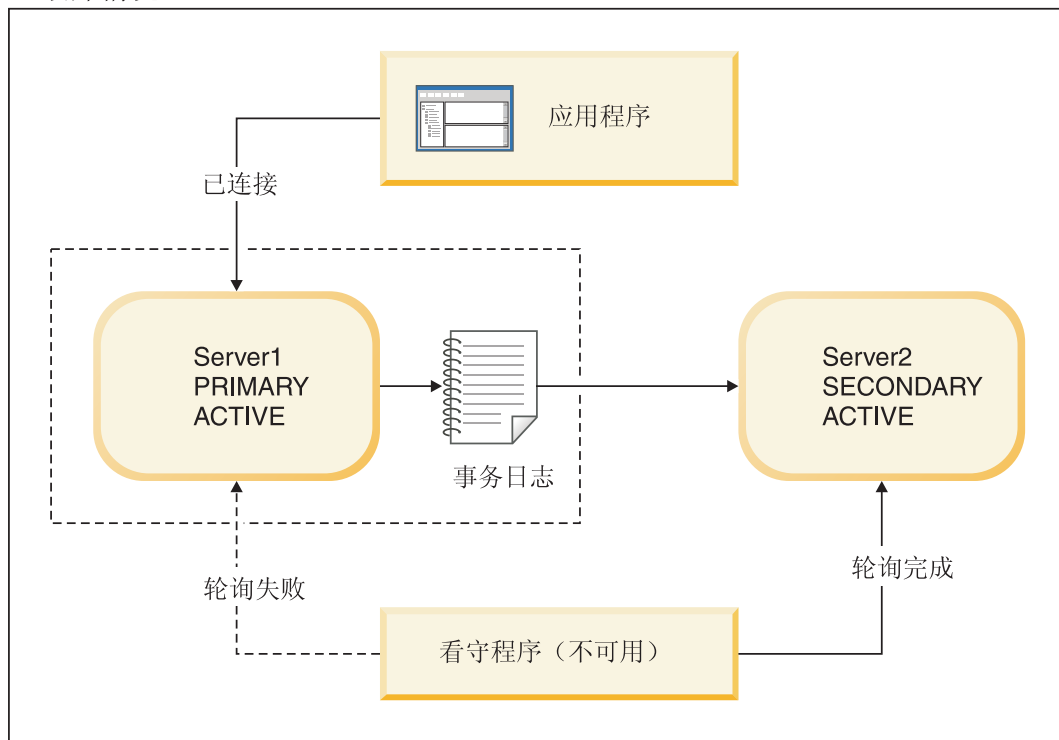
看守程序和主服务器之间的通信链路已断开 情况

看守程序与主服务器之间的连接已断开。

补救措施

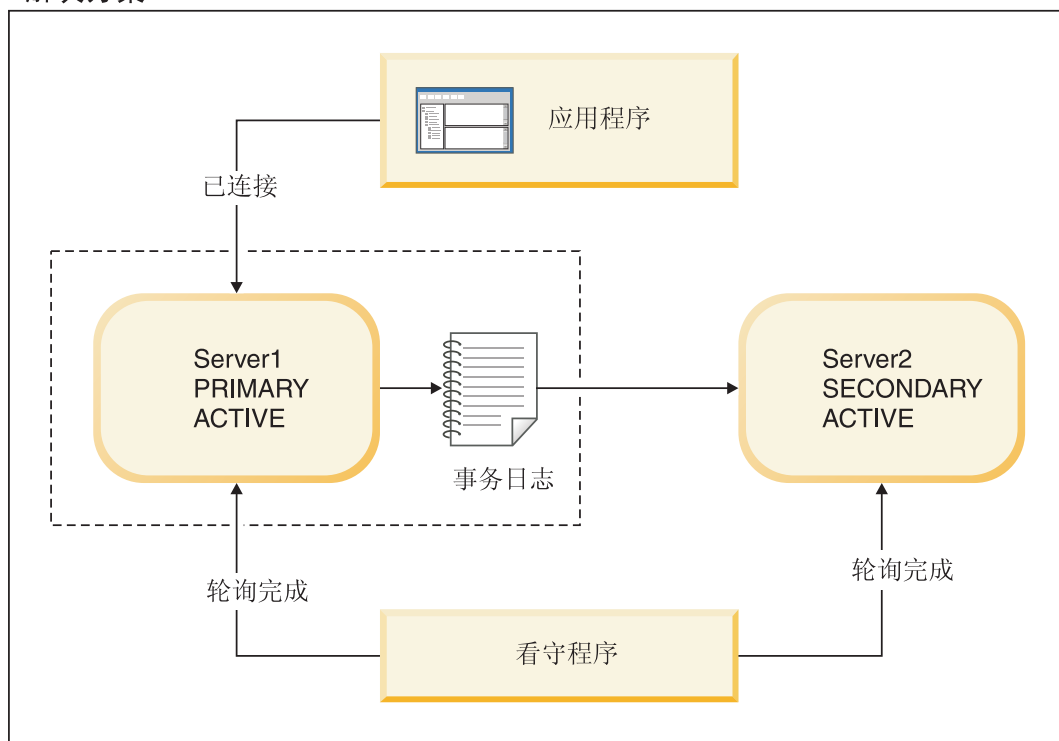
即使看守程序与主服务器之间的链路已断开，主服务器和辅助服务器仍然可以继续运行。修复了看守程序与主服务器之间的链路之后，务必检查主服务器和辅助服务器的状态。

故障情况



解决方案

1



1. 修复看守程序与 Server1 的网络连接。

图 21. 看守程序与主服务器之间的链路已断开的情况和补救措施

症状

看守程序在主服务器中轮询失败。但是，辅助服务器状态被报告为 `SECONDARY ACTIVE`。这意味着主服务器很可能处于正常状态，并且看守程序只是与主服务器失去联系而已。

其他情况

如果两个服务器的状态已更改并且一个服务器不再工作，那么请参阅本节中的适当情况以获取指示信息。

当看守程序与主服务器之间的通信链路已断开的情况下如何恢复

关于此任务

要从看守程序与主服务器之间的连接已断开的情况下恢复：

1. 允许主服务器和辅助服务器继续正常操作。
2. 修复看守程序与主服务器之间的网络连接。
3. 一旦建立了网络连接之后，让看守程序使用以下命令来检查每个服务器的状态：

```
ADMIN COMMAND 'hotstandby state';
```

看守程序和辅助服务器之间的通信链路已断开

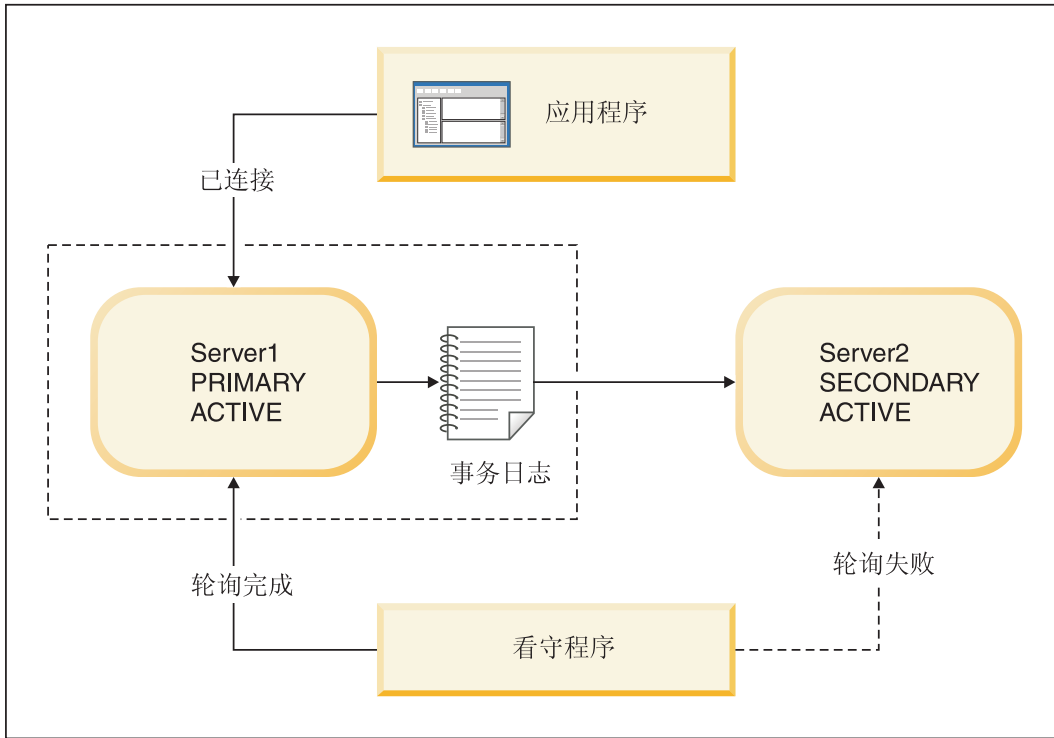
情况

看守程序与辅助服务器之间的连接已断开。

补救措施

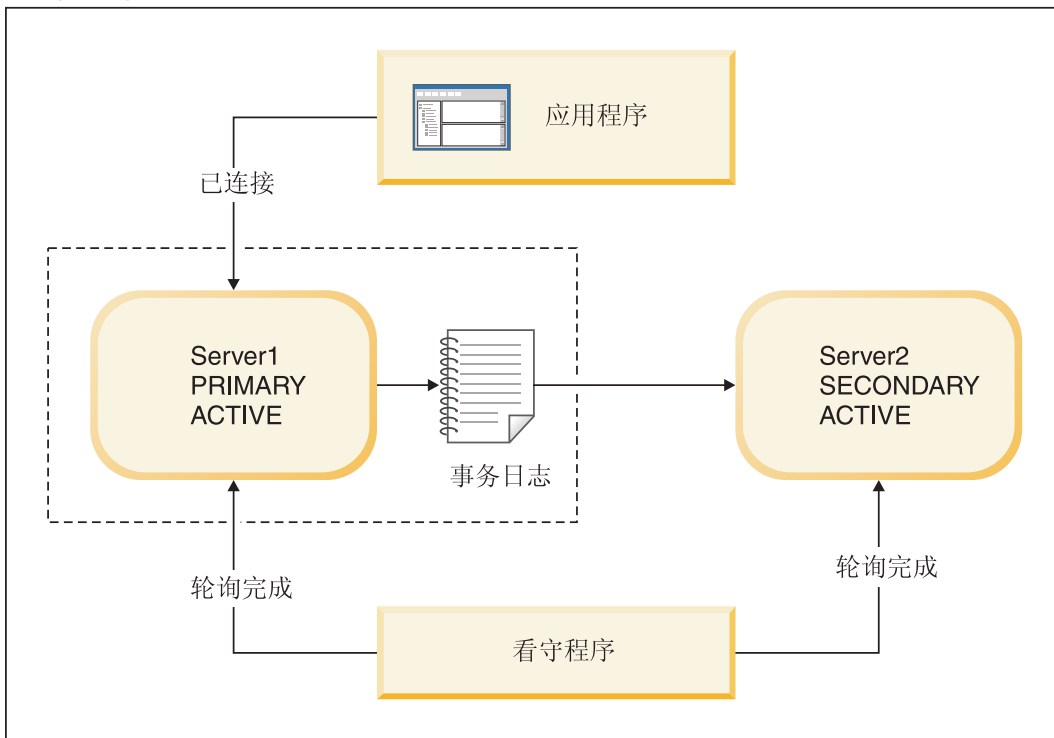
即使看守程序与辅助服务器之间的链路已断开，主服务器和辅助服务器仍然可以继续运行。修复了看守程序与辅助服务器之间的链路之后，务必要检查主服务器和辅助服务器以确认它们的状态。

故障情况



解决方案

1



1. 修复看守程序与 Server2 的网络连接。

图 22. 看守程序与辅助服务器之间的链路已断开的情况和补救措施

症状

看守程序在辅助服务器中轮询失败。

其他情况

如果两个服务器的状态已更改并且一个服务器不再工作，那么请参阅本节中的适当情况以获取指示信息。

当看守程序与辅助服务器之间的通信链路已断开的情况下如何恢复 关于此任务

要从看守程序与辅助服务器之间的连接已断开的情况下恢复：

1. 允许主服务器和辅助服务器继续正常操作。
2. 修复看守程序与辅助服务器之间的网络连接。
3. 一旦建立了网络连接之后，让看守程序使用以下命令来检查每个服务器的状态：

```
ADMIN COMMAND 'hotstandby state';
```

看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开

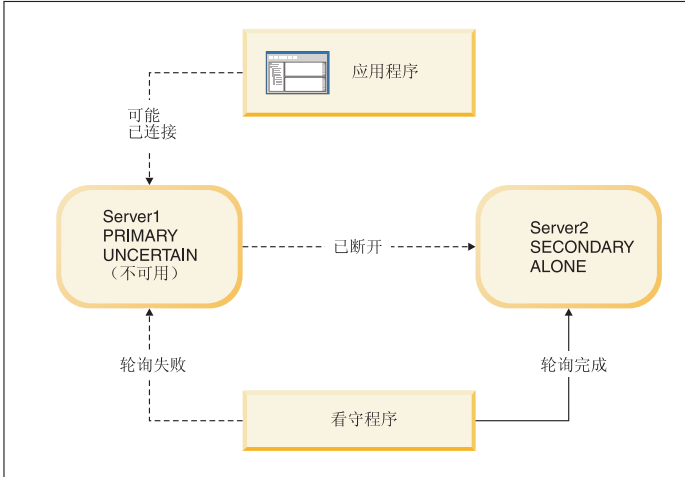
情况

看守程序与主服务器之间以及主服务器与辅助服务器之间的连接已断开。

补救措施

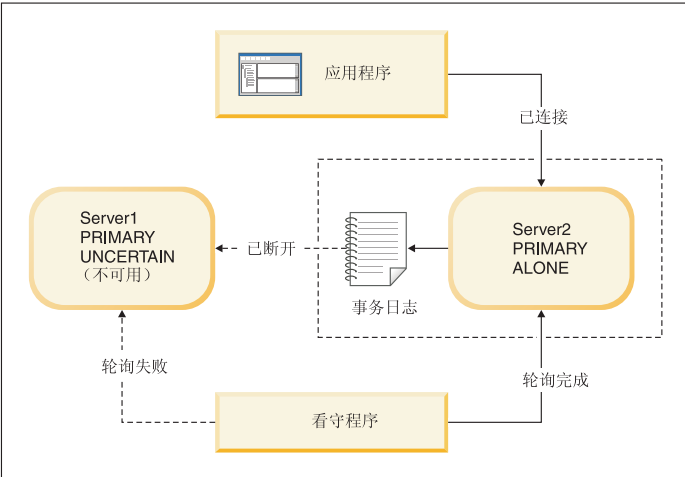
为了让看守程序继续监视主服务器，请将辅助服务器切换为新的主服务器并将此新的主服务器设置为 PRIMARY ALONE 状态。然后，设置新的辅助服务器并使它与主服务器同步。

故障情况



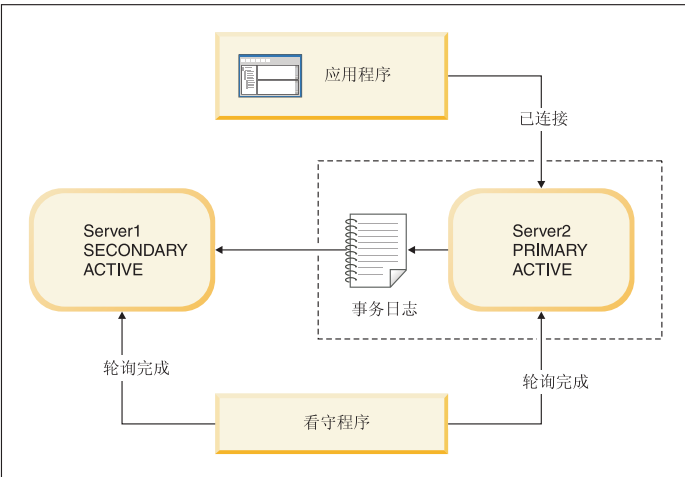
1

故障转移/补救措施



2

解决方案



3

图 23. 看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况和补救措施

1. Server1 的角色是 PRIMARY UNCERTAIN。但是，从看守程序的角度来看，Server1 是不可用而不是 PRIMARY UNCERTAIN。

看守程序将指示 Server2 执行以下操作:

```
HSB SET PRIMARY ALONE
```

应用程序从 Server1 切换到 Server2。

2. 两个服务器都认为它们自己是主服务器。如果程序或管理员（通过手动干预）将 Server1 从 PRIMARY UNCERTAIN 状态切换为 PRIMARY ALONE 状态，那么将有两个处于活动状态的主服务器，它们都可能在更新数据，将无法解决它们之间的差别。

在修复网络连接之后，看守程序将指示 Server1 执行以下操作:

```
HSB SWITCH SECONDARY
```

看守程序将指示 Server2 执行以下操作:

```
HSB COPY 或者
```

```
NETCOPY HSB CONNECT
```

3. 如果事务日志已填满，那么可能需要将 PRIMARY ALONE 服务器切换为 STANDALONE。在这种情况下，在重新连接服务器之前您还需要执行 HSB COPY 或 HSB NETCOPY。如果事务日志未填满，那么您必须跳过 COPY/NETCOPY 命令。

症状

看守程序在主服务器中轮询失败。辅助服务器与主服务器之间已经断开连接；因此，Server2 处于 SECONDARY ALONE 状态，如果可与主服务器联系，那么主服务器将报告其状态为 PRIMARY UNCERTAIN 或 PRIMARY ALONE。

这种情况开始假定应用程序可能已连接至旧的主服务器。但是，由于旧的主服务器处于 PRIMARY UNCERTAIN 状态，因此应用程序无法执行更新。请注意，连接至 Server1 的应用程序也有可能已经失去了它们的通信链路，因此不再知道旧的主服务器是否存在。

当看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况下如何恢复

要从看守程序与主服务器之间以及主服务器与辅助服务器之间的连接已断开的情况下恢复，请执行必需的步骤以使用热备用服务器（辅助服务器）来替换主服务器。

关于此任务

为了允许辅助服务器替换主服务器，请执行下列操作:

1. 如果旧的主服务器处于 PRIMARY UNCERTAIN 状态或者与应用程序和辅助服务器断开连接，那么使用以下命令将辅助服务器设置为 PRIMARY ALONE 状态:

```
ADMIN COMMAND 'hotstandby set primary alone';
```
2. 将应用程序重新连接至新的主服务器。
3. 修复与旧的主服务器的网络连接或者已断开的连接。
4. 检查服务器状态。两个服务器现在必须正在运行。

5. 如果新的主服务器处于 STANDALONE 状态（例如，由于在修复连接期间新的主服务器的事务日志已填满）：
 - a. 使用以下命令将新的主服务器设置为 PRIMARY ALONE 状态：

```
ADMIN COMMAND 'hotstandby set primary alone';
```
 - b. 将数据库从新的主服务器复制到新的辅助服务器。有关详细信息，请阅读第 78 页的『使主服务器与辅助服务器同步』。
6. 如果新的主服务器处于 PRIMARY ALONE 状态：
 - a. 使用以下命令将旧的主服务器切换到新的辅助服务器：

```
ADMIN COMMAND 'hotstandby switch secondary';
```
7. 使用以下命令将新的主服务器重新连接至新的辅助服务器：

```
ADMIN COMMAND 'hotstandby connect';
```

看守程序与主服务器之间以及主服务器与辅助服务器之间的通信链路已断开的其他情况

如果应用程序从新的主服务器中接收到错误消息 10047 或 14537:

- 尝试连接至旧的辅助服务器以检查它是否已切换为新的主服务器。
- 如果旧的辅助服务器未切换为新的主服务器，那么请对处于 PRIMARY ALONE 状态的原来的主服务器重新执行事务。

看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开

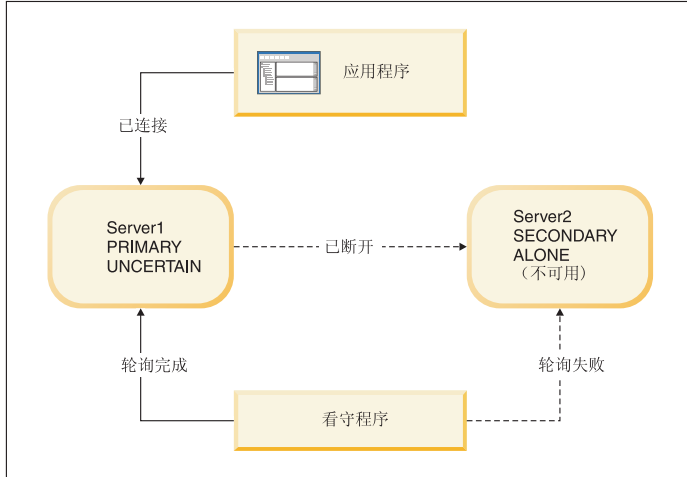
情况

看守程序与辅助服务器之间的连接以及主服务器与辅助服务器之间的连接已断开。

补救措施

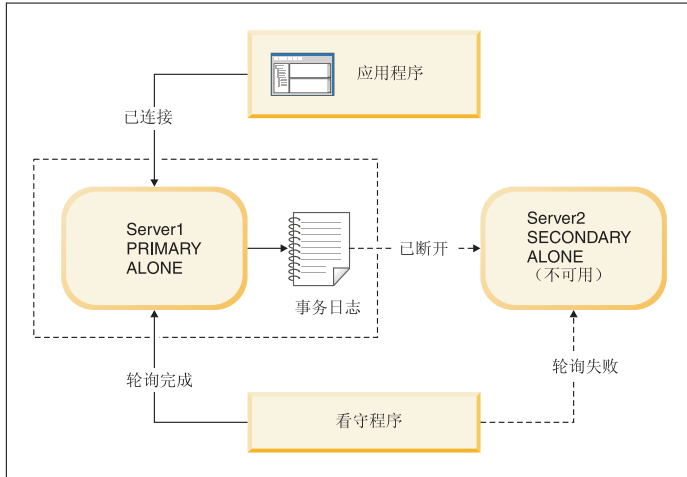
即使主服务器与辅助服务器和看守程序之间的链路已断开，主服务器仍然可以继续运行。如果主服务器尚未处于 PRIMARY ALONE 状态，那么请将它切换为此状态。然后，在辅助服务器再次启动之后，使它与主服务器同步。

故障情况



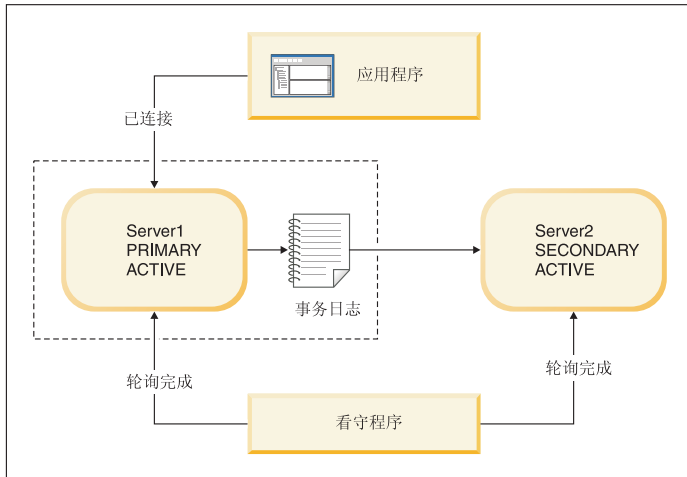
1

故障转移/补救措施



2

解决方案



3

图 24. 看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况和补救措施

1. Server2 看到它自己的角色为 SECONDARY ALONE，但是看守程序看不到 Server2，因为它认为 Server2 不可用。

看守程序将指示 Server1 执行以下操作:

```
HSB SET PRIMARY ALONE
```

- 在修复与辅助服务器的连接之后, 看守程序将指示 Server1 执行以下操作:

```
HSB COPY
```

```
NETCOPY HSB CONNECT
```

- 如果事务日志已填满, 那么可能需要将 PRIMARY ALONE 服务器切换为 STANDALONE。在这种情况下, 在重新连接服务器之前您还需要执行 HSB COPY 或 HSB NETCOPY。如果事务日志未填满, 那么您必须跳过 COPY/NETCOPY 命令。

症状

看守程序在辅助服务器中轮询失败; 主服务器未与辅助服务器连接, 并且已切换为 PRIMARY UNCERTAIN 或 PRIMARY ALONE 状态。

当看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开的情况下如何恢复

关于此任务

要从看守程序与辅助服务器之间的连接以及主服务器与辅助服务器之间的连接已断开的情况下恢复:

- 尝试修复连接。
- 在修复连接之后, 使用 ADMIN COMMAND 'hotstandby state' 命令检查主服务器的状态。
- 如果主服务器的状态为 STANDALONE:
 - 确保两个服务器都在运行。
 - 使用以下命令将主服务器的状态设置为 PRIMARY ALONE:

```
ADMIN COMMAND 'hotstandby set primary alone';
```
 - 使用以下命令将数据库从主服务器复制到辅助服务器:

```
ADMIN COMMAND 'hotstandby netcopy';
```

有关详细信息, 请阅读第 78 页的『使主服务器与辅助服务器同步』。

- 使用以下命令将主服务器重新连接至辅助服务器:

```
ADMIN COMMAND 'hotstandby connect';
```

看守程序与辅助服务器之间以及主服务器与辅助服务器之间的通信链路已断开的其他情况

如果应用程序从主服务器中接收到错误消息 10047 或 14537:

- 尝试连接至辅助服务器以检查它是否已切换为主服务器。
- 如果辅助服务器未切换为新的主服务器, 那么请对处于 PRIMARY ALONE 状态的原来的主服务器重新执行事务。

solid.ini 配置文件的 [Watchdog] 节

看守程序的 solid.ini 文件包含 [Watchdog] 配置节以指定特定于看守程序的参数。

要点: solid.ini 文件的 [WatchDog] 节中的参数并非都是由 solidDB 预定义的。根据您的如何编写看守程序以及您是否希望它从 solid.ini 文件中读取参数信息，可以将此处定义的参数与您已经定义的参数混合使用。您也可以忽略这些参数。此处所显示的参数适用于 solidDB 提供的用 C 语言编写的样本看守程序。

表 33. 看守程序参数

[Watchdog]	描述	出厂值
AutoSwitch	<p>如果 AutoSwitch 参数设置为 yes，那么看守程序将自动执行下列操作：</p> <ol style="list-style-type: none">1. 如果辅助服务器失败，那么看守程序会让主服务器切换为 PRIMARY ALONE 状态（而不是保持 PRIMARY UNCERTAIN 状态）。2. 如果主服务器失败，那么看守程序将自动发送下列命令： <pre>'hsb switch primary' 'hsb set primary alone'</pre> <p>以将原来的辅助服务器切换为新的主服务器。</p> <p>例如： [Watchdog] AutoSwitch = NO</p> <p>此参数是可选的。</p>	Yes
Connect1	<p>[Watchdog] 节中的 Connect1 参数使看守程序能够连接至主服务器或辅助服务器。这是一个必需参数，用于定义 Connect1 服务器的协议和网络地址。</p> <p>例如： connect1 = tcp primarymachine 1313</p>	None
Connect2	<p>[Watchdog] 节中的 Connect2 参数使看守程序能够连接至主服务器或辅助服务器。这是一个必需参数，用于定义 Connect2 服务器的协议和网络地址。</p> <p>例如： connect2 = tcp secondarymachine 1313</p>	None
DualSecAutoSwitch	<p>如果 DualSecAutoSwitch = Yes 并且两个服务器都是辅助服务器，那么看守程序将自动选择其中一个辅助服务器作为新的主服务器并将它切换为主服务器。如果 DualSecAutoSwitch = No，那么系统管理员必须将一个服务器切换为主服务器。请注意，无论看守程序处于“正常”方式还是“故障”方式，DualSecAutoSwitch 参数都适用。</p>	Yes

表 33. 看守程序参数 (续)

[Watchdog]	描述	出厂值
NumRetry	<p>[Watchdog] 节中的 NumRetry 参数允许您指定在认为连接尝试发生响应故障或错误之前看守程序可以尝试连接至辅助服务器或主服务器的次数。</p> <p>例如:</p> <pre>[Watchdog] NumRetry = 3</pre> <p>重试次数就是除初始尝试之外的尝试次数。如果重试次数设置为 3, 那么总的尝试次数就是 4。请注意, 重试是立即进行的。当发生故障时, 看守程序不会等待两次重试之间的时间间隔 (例如, PingTimeout) 这段时间。</p> <p>此参数是可选的。</p>	0
Password1 Password2	请参阅下面对 Username1 和 Username2 参数的描述。	无出厂值。
Pessimistic	<p>将此参数设置为 Yes 可以提高看守程序的反应速度。</p> <p>当 Pessimistic = No 时, 看守程序会检查它与服务器之间的连接, 但是要等到其中一个服务器检测到有一个问题并更改其状态 (例如, 更改为 PRIMARY UNCERTAIN) 之后, 看守程序才会真正执行操作 (例如, 将服务器的状态更改为 PRIMARY ALONE)。</p> <p>当 Pessimistic = Yes 时, 一旦看守程序与其中一个服务器失去联系, 它就会立即执行操作; 看守程序不会等到另一个服务器更改状态才执行操作。这可以提高反应速度, 但是也可能会提高假报警 (例如, 由于发生网络问题而发出的报警) 的可能性。</p> <p>当 Pessimistic = Yes 时, 看守程序将按如下所示作出反应: 如果看守程序已经与主服务器失去联系, 那么看守程序会将辅助服务器切换为主服务器; 如果看守程序已经与辅助服务器失去联系, 那么它就会将主服务器设置为 PRIMARY ALONE。</p> <p>注意: 设置 Pessimistic = Yes 可能会导致额外的切换甚至导致双重主服务器。除非网络比服务器更不可能失败, 否则不应将此参数设置为 Yes。</p> <p>还可以使用可选的命令行开关“-p”来打开悲观行为。</p>	No
PingInterval	<p>[Watchdog] 节中的 PingInterval 参数允许您指定两次以正常的看守程序方式查询状态连接信息之间的时间间隔 (按毫秒计)。要检测服务器故障, 每隔 PingInterval 毫秒之后, 看守程序就会对主服务器和辅助服务器都发送 hotstandby status connect 命令。</p> <p>例如:</p> <pre>[Watchdog] PingInterval = 5000</pre> <p>此参数是可选的。</p> <p>请注意, 看守程序的 PingInterval 参数不同于服务器的 PingTimeout 参数。</p> <p>注意: 先前的样本看守程序要求按秒 (而不是毫秒) 指定 PingInterval。如果您要使用更旧的 solid.ini 文件, 那么应对它进行更新。</p>	1000 (1 秒)

表 33. 看守程序参数 (续)

[Watchdog]	描述	出厂值
Username1 Username2	<p>[Watchdog] 节中的 Username 和 Password 参数是可选的。这两个参数指定有权使用 connect1 服务器的用户名和密码。</p> <p>例如:</p> <pre>[Watchdog] Username1 = Tom Password1 = dr17xy Username2 = Jerry Password2 = M89tvt</pre> <p>如果由于安全原因而在 solid.ini 配置文件中未指定这些参数, 那么看守程序启动时将提示您输入用户名和密码。</p>	无出厂值。
WatchdogLog	<p>[Watchdog] 节中的 WatchdogLog 参数允许您指定看守程序日志的文件名。将在当前工作目录中创建看守程序日志。它用来记录看守程序消息, 提醒管理员需要发出看守程序命令。</p> <p>例如:</p> <pre>[Watchdog] WatchdogLog = Watchdog.log</pre> <p>请注意, 只要文件名中不包含特殊字符 (例如, 空白或某些标点符号), 就不需要用引号将文件名引起来。</p> <p>此参数是可选的。</p>	Watchdog.log

当使用

[Logging]
DurabilityLevel

参数时, **DurabilityLevel** 参数值将只影响主服务器。辅助服务器的记录方式由 [HotStandby] 节中的 **2SafeAckPolicy** 参数指定。

索引

[B]

备份 4

备份侦听方式

请参阅 netcopy 侦听方式

[C]

参数

高可用性控制器

CheckInterval 109

CheckTimeout 109

Connect [LocalDB] 109

Connect [RemoteDB] 109

DBPassword 109

DBUsername 109

EnableAutoNetcopy 109

EnableDBProcessControl 109

EREIP 109

Listen 109

Password 109

PreferredPrimary 109

RequiredConnectFailures 109

RequiredPingFailures 109

StartInAutomaticMode 109

StartScript 109

Username 109

AutoPrimaryAlone 45, 73, 76

AutoSwitch 169

BackupBlockSize 46

BackupDeleteLog 4

CatchupSpeedRate 46

CatchupStepsToSkip 46

CheckInterval 47, 109

CheckpointDeleteLog 4

CheckTimeout 47, 109

Connect 33, 43, 48, 49, 92

Connect [LocalDB] 109

Connect [RemoteDB] 109

Connect1 169

Connect2 169

ConnectTimeout 43, 45

CopyDirectory 45

DBPassword 47, 109

DBUsername 47, 109

DualSecAutoSwitch 169

DurabilityLevel 42

EnableAutoNetcopy 47, 109

EnableDBProcessControl 47, 109

EREIP 49, 109

Header_text 112

参数 (续)

HSBEnabled 33, 43, 92

Listen 41, 47

LogEnabled 43

NumRetry 169

Password 47, 109

Password1 169

Password2 169

Pessimistic 169

PingInterval 43, 44, 169

PingTimeout 43, 44

PreferredPrimary 48, 109

ReadMostlyLoadPercentAtPrimary 54

RequiredConnectFailures 47, 109

RequiredPingFailures 49, 109

Server1_host 112

Server1_name 112

Server1_pass 112

Server1_port 112

Server1_user 112

Server2_host 112

Server2_name 112

Server2_pass 112

Server2_port 112

Server2_user 112

StartInAutomaticMode 47, 109

StartScript 48, 109

Username 47, 109

Username1 169

Username2 169

WatchdogLog 169

Window_title 112

查看当前的连接设置 86

出厂值 104

从应用程序重新连接至主服务器 65

存储方式 105

[D]

当前值 104

等号

设置参数值时使用 33

定义主服务器在辅助服务器发生故障期间的行为 45

对服务器进行升级

冷迁移 98

冷迁移和热迁移 97

热迁移 99

在与 HSB 不兼容的版本之间进行迁移 97

在与 HSB 兼容的版本之间进行迁移 97

准备 98

[F]

- 访问方式
 - RO (只读) 105
 - RW (读写) 105
 - RW/Create 105
 - RW/Startup 105
- 访问权 34
- 分区
 - 网络 35
- 服务器
 - 连接 86
- 服务器名称 网络名 41
- 服务器状态
 - 切换服务器状态 73
 - 验证 89
 - OFFLINE 12
 - PRIMARY ACTIVE 12
 - PRIMARY ALONE 12, 76
 - PRIMARY UNCERTAIN 12
 - SECONDARY ACTIVE 12
 - SECONDARY ALONE 12
 - STANDALONE 12
- 辅助服务器
 - 恢复联机状态 77
- 负载均衡
 - 动态控制 55
 - 透明连接 54
- 负载均衡方法
 - PREFERRED_ACCESS=READ_MOSTLY 54
 - PREFERRED_ACCESS=WRITE_MOSTLY 54
- 复制
 - 从主服务器复制到本地辅助服务器 85
 - 数据库的内容 83, 84, 85
 - 验证过程 82, 85
- 复制数据库的内容 83, 84, 85

[G]

- 高可用性管理器
 - 配置 47
 - 屏幕快照 22
 - 已定义 22
- 高可用性控制
 - 原则 73
- 高可用性控制器
 - 配置 27, 47
 - 启动 27
 - 日志记录 23
 - 设置 27
 - 停止 27
 - 样本 25
 - 在故障情况下的行为 93
 - solidhac.ini 27
- 工具
 - solidDB 数据管理工具 34

- 故障透明性
 - 选择连接类型
 - CONNECTION 54
 - NONE 54
 - SESSION 54
- 管理 HotStandby 88
 - 切换服务器状态 73

[H]

- 恢复
 - 维护 37

[J]

- 基本连接 65
- 检查点 4

[K]

- 看守程序样本 25
- 可重复读
 - 事务隔离级别 35
- 可序列化
 - 事务隔离级别 35

[L]

- 连接
 - 基本 65
 - 透明故障转移 54
 - 选择连接类型 53
- 连接切换
 - 透明连接 61
- 连接设置
 - 查看 86
- 落实读
 - 事务隔离级别 35

[N]

- 内存表 35

[P]

- 配置
 - 辅助节点和主节点配置 41
 - 应用程序与服务器之间的超时 41
- 配置 HotStandby
 - netcopy 性能 46

[Q]

- 启动顺序 29

切换
 显示连接状态信息 88
 显示切换状态信息 88
切换状态
 验证 75
权限
 访问 34
缺省值 104

[R]

日志记录
 请参阅 高可用性控制器

[S]

使辅助服务器恢复联机状态 77
使用 HAC 的 HotStandby
 快速启动 28
 配置 28
 设置 28
使主服务器与辅助服务器同步 78
事务隔离级别
 内存表 35
事务日志
 用尽空间 36
数据管理工具 34
数据库
 复制内容 83, 84, 85
 内存表 35
 验证复制过程 85
双重主服务器 35

[T]

停止 HotStandby 操作 77
通过网络将主数据库复制到辅助服务器 82
同步复制 78
透明连接 54

[W]

外部引用实体
 配置 109
 已定义 20
网络分区 35
网络分区和双重主服务器 35
网络名 41
为辅助服务器创建新的数据库 83

[X]

显示连接状态信息 88
显示切换状态信息 88

显示通信信息 88
选择使哪个服务器成为主服务器 91

[Y]

验证复制过程 85
验证连接状态信息
 HOTSTANDBY_CONNECTSTATUS 67
验证切换进程
 ADMIN COMMAND 'hotstandby status switch' 75
验证状态切换 75
样本
 高可用性控制器 25
 看守程序 25
应用程序
 切换到新的主服务器 67
 使用基本连接 65
用尽用于事务日志的空间 36
运行处于 PRIMARY ALONE 状态的服务器 76

[Z]

执行恢复和维护 37
状态 87
 列表 60, 61, 88
 显示连接状态信息 88
 显示切换状态信息 88
 显示通信信息 88
 验证服务器状态 89
 HotStandby 87
 OFFLINE 135
 PRIMARY ACTIVE 135
 PRIMARY ALONE 135
 PRIMARY UNCERTAIN 135
 STANDALONE 77, 92, 135
状态切换
 验证 75

[数字]

1SafeMaxDelay (参数) 106
2SafeAckPolicy (参数) 106

A

ADMIN COMMAND 'hotstandby cominfo'
 查看连接设置 86
ADMIN COMMAND 'hotstandby connect'
 连接 HotStandby 服务器 86
ADMIN COMMAND 'hotstandby copy'
 复制数据库的内容 85
ADMIN COMMAND 'hotstandby netcopy'
 复制到辅助服务器 83
 复制数据库的内容 83, 84

ADMIN COMMAND 'hotstandby set primary alone'
运行处于 PRIMARY ALONE 状态的服务器 76
ADMIN COMMAND 'hotstandby set standalone'
停止 HotStandby 操作 77
ADMIN COMMAND 'hotstandby state'
验证服务器状态 89
ADMIN COMMAND 'hotstandby status connect'
显示连接状态信息 88
ADMIN COMMAND 'hotstandby status copy'
验证复制过程 82, 85
ADMIN COMMAND 'hotstandby status switch'
验证切换进程 75
ADMIN COMMAND 'hotstandby status'
查询 HotStandby 状态 87
ADMIN COMMAND 'hotstandby switch primary'
切换服务器状态 73
ADMIN COMMAND 'hotstandby switch secondary'
切换服务器状态 73
autoconvert 99
AutoPrimaryAlone (参数) 45, 76, 106, 146
和 'hotstandby switch' 命令 73
AutoSwitch (参数) 169

B

BackupBlockSize (参数) 46
BackupDeleteLog (参数) 4

C

CatchupSpeedRate (参数) 46, 106
CatchupStepsToSkip (参数) 46
CheckInterval (参数) 109
CheckpointDeleteLog (参数) 4
CheckTimeout (参数) 109
CLUSTER 57
Connect (参数) 33, 43, 92, 106
Connect [LocalDB] (参数) 109
Connect [RemoteDB] (参数) 109
Connect1 (参数) 169
Connect2 (参数) 169
ConnectTimeout (参数) 43, 45, 106
CopyDirectory (参数) 45, 106

D

DBPassword (参数) 109
DBUsername (参数) 109
DualSecAutoSwitch (参数) 169
DurabilityLevel (参数) 42

E

EnableAutoNetcopy (参数) 109
EnableDBProcessControl (参数) 109

ERE
请参阅 外部引用实体
EREIP (参数) 109

G

GUI
请参阅 高可用性管理器

H

HA 管理器参数
Header_text 112
Server1_host 112
Server1_name 112
Server1_pass 112
Server1_port 112
Server1_user 112
Server2_host 112
Server2_name 112
Server2_pass 112
Server2_port 112
Server2_user 112
Window_title 112
HA 控制器 73
HAC 参数
HACController 47, 48, 49
HAC 故障情况
辅助节点失败 95
辅助数据库失败 93
主节点失败 94
主数据库失败 93
HotStandby 链路失败 95
Header_text (参数) 112
HotStandby
关闭 92
管理 33
快速启动 26
配置 26, 33, 40, 47
设置 26
停止操作 77
HotStandby 事件
SYS_EVENT_HSBCONNECTSTATUS 141
SYS_EVENT_HSBSTATESWITCH 141
SYS_EVENT_NETCOPYEND 141
SYS_EVENT_NETCOPYREQ 141
HotStandby 状态 87
hotstandby copy 135
hotstandby netcopy 135
HOTSTANDBY_CONNECTSTATUS
SQL 函数 88
HOTSTANDBY_CONNECTSTATUS (SQL 函数) 67
HOTSTANDBY_STATE (SQL 函数) 67
hsb status catchup 127
hsb status connect 127

hsb status copy 127
hsb status switch 127
HSBEnabled (参数) 33, 43, 92, 106

L

Listen (参数) 41, 109
LogEnabled (参数) 43
logpos (管理命令) 127
logpos (HotStandby 命令) 91

M

MaxLogSize (参数) 106
MaxMemLogSize (参数) 106
migratesbg2 99

N

netcopy 135
 主服务器必须处于 PRIMARY ALONE 状态 9
netcopy 侦听方式 83
 调整性能 46
 ADMIN COMMAND 'hotstandby netcopy' 83
netcopy netcopy 侦听方式 46
NetcopyRpcTimeout (参数) 106
NumRetry (参数) 169

O

OFFLINE (状态) 135

P

Password (参数) 109
Password1 (参数) 169
Password2 (参数) 169
Pessimistic (参数) 169
ping 44
PingInterval (参数) 43, 44, 106, 169
PingTimeout (参数) 43, 44, 106
PreferredPrimary (参数) 109
PRIMARY ACTIVE (状态) 135
PRIMARY ALONE 76
PRIMARY ALONE (状态) 135
PRIMARY ALONE 状态
 运行新的处于 PRIMARY ALONE 状态的主服务器 76
PRIMARY UNCERTAIN (状态) 135
PrimaryAlone (参数) 106

R

ReadMostlyLoadPercentAtPrimary (参数) 54, 106
RequiredConnectFailures (参数) 109

RequiredPingFailures (参数) 109
RO

 访问方式 105

RW

 访问方式 105

RW/Create

 访问方式 105

RW/Startup

 访问方式 105

S

Server1_host (参数) 112
Server1_name (参数) 112
Server1_pass (参数) 112
Server1_port (参数) 112
Server1_user (参数) 112
Server2_host (参数) 112
Server2_name (参数) 112
Server2_pass (参数) 112
Server2_port (参数) 112
Server2_user (参数) 112
SET TRANSACTION WRITE 55
SET WRITE 55
solidDB 导出
 已定义 34
solidDB 快速装入器
 已定义 34
solidDB 数据管理工具 34
solidDB 数据字典
 已定义 34
solidhac.ini 27
 [ERE] 节 49
 [HAController] 节 47
 [LocalDB] 节 48
 [RemoteDB] 节 49
SQL 函数
 HOTSTANDBY_CONNECTSTATUS 67, 88
 HOTSTANDBY_STATE 67
STANDALONE (状态) 77, 92, 135
StartInAutomaticMode (参数) 109
StartScript (参数) 109
SYS_EVENT_HSBCONNECTSTATUS (事件) 141
SYS_EVENT_HSBSTATESWITCH (事件) 141
SYS_EVENT_NETCOPYEND (事件) 141
SYS_EVENT_NETCOPYREQ (事件) 141

T

TC 连接 54
TC Info 57
 处理矛盾 59
 属性组合 58
 私有 ODBC 属性 64
 语法 57

TC Info (续)
 JDBC 语法 59
TF 连接 54

U

Username (参数) 109
Username1 (参数) 169
Username2 (参数) 169

W

WatchdogLog (参数) 169
Window_title (参数) 112

[特别字符]

-x autoconvert 99
-x backupserver (命令行选项) 83
-x backupserver (命令) 83
-x migratehsbg2 99
=
 设置参数值时使用等号 33

声明

Copyright © Solid Information Technology Ltd. 1993, 2008

All rights reserved.

除非经过 Solid Information Technology Ltd. 或者 International Business Machines Corporation 书面授权，否则不能以任何方式使用本产品中的任何部分。

本产品受美国专利 6144941、7136912、6970876、7139775、6978396 和 7266702 的保护。

为此产品指定的美国出口管制分类编号是 ECCN=5D992b。

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

有关双字节（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区：INTERNATIONAL BUSINESS MACHINES CORPORATION“按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和/或程序进行改进和/或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：（i）允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及（ii）允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

所有关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，示例中可能会包括个人、公司、品牌和产品的名称。所有这些名字都是虚构的，若现实生活中实际业务企业使用的名字和地址与此相似，纯属巧合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。

凡这些实例程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：

©（贵公司的名称）（年）。此部分代码是根据 IBM Corp. 公司的样本程序衍生出来的。

© Copyright IBM Corp.（输入年份）。All rights reserved.

商标

IBM、IBM 徽标、ibm.com[®]、Solid、solidDB、InfoSphere[™]、DB2[®]、Informix[®] 和 WebSphere[®] 是 International Business Machines Corporation 在美国和/或其他国家或地区的商标或注册商标。如果这些商标和其他 IBM 注册商标在本资料中第一次出现时带有商标符号（[®] 或 [™]），那么这些符号表示它们是发布本资料时归 IBM 所有的经过美国政府注册的商标或普通法商标。这些商标也可能是在其他国家或地区的注册商标或普通法商标。在 Web 上的版权和商标信息（www.ibm.com/legal/copytrade.shtml）处提供了 IBM 商标的最新列表。

Java 和所有基于 Java 的商标和徽标是 Sun Microsystems, Inc. 在美国和/或其他国家或地区的商标。

Linux[®] 是 Linus Torvalds 在美国和/或其他国家或地区的注册商标。

Microsoft and Windows 和 Microsoft Corporation 在美国和/或其他国家或地区的注册商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

其他公司、产品或服务名称可能是其他公司的商标或服务标记。



中国印刷

S151-1151-00

