数据是现代随需应变业务的血液；存储和[移动](http://tech.ddvip.com/comm/mobile/index.html" \t "_blank)数 据的系统（[服 务器](http://tech.ddvip.com/server/index.html" \t "_blank)、网络、数据库）是这个系统的心脏。但是如果没有 heartbeat —— 对这些数据具有可靠而快速的访问，且宕机时间最少 —— 那么这两者都是惰性组件。

　　简介

　　本系列的第一篇文章 Linux 上的高可用中间件，第 1 部分：Heartbeat 和 [Apache](http://tech.ddvip.com/server/apache/index.html) Web 服务器 简要介绍了高可用（HA）的概念，以及如何安装并配置 heartbeat。本篇文章是本系列的最后一篇文章，它将介绍如何在一个 冷备份（cold standby） 配置中使用 heartbeat 为 DB2 UDB 8.1 实现一个 HA 方案。

　　关于 heartbeat

　　Heartbeat 是 Linux-HA 项目中提供的一个公用包。Heartbeat 提供了 HA 系统所需要的基本功能，例如启动/停止资源，监视集群中系统的可用性，在集群节点之间切换共享的 IP 地址信息。Heartbeat 还可以通过一个串口线或以太网接口来监视特定服务（或多个服务）的健康信息。当前的版本支持一个两节点的配置，其中使用特殊的 heartbeat "pings" 来检查服务的状态和可用性。

　　在这种实现中，heartbeat 会检测主节点的失效情况，并发起故障迁移的过程：

　 　在主节点上停止 DB2 进程

　　在主节点上释放共享磁盘

　　在主节点上释放该服务的 IP 地址

　 　将这个服务的 IP 地址添加到备用节点上

　　在备用节点上加载这个共享磁盘

　　在备用机器上重新启动 DB2 进程

　　为了最好地理解本文的内容，您需要对 DB2 UDB 和高可用集群有一个基本的理解。本系列的第一篇文章 介绍了对于[软件](http://down.ddvip.com/" \t "_blank)的高可用性来说这意味着什么，以及如何在一 个两节点的系统上使用 High-Availability Linux 项目安装并设置 heartbeat 软件。

　　DB2 UDB 和 HA 基础

　　在 heartbeat 集群中使用的任何 DB2 UDB 都必须将数据全部存放在共享磁盘上，这样在发生节点失效的情况时，就可以在依然存活的机器上访问这些数据。运行数据库实例的节点还必须在内部磁盘上维护很 多文件。这些文件包括与节点上的所有数据库有关的文件。

　　与数据库实例有关的文件会被分别存放在内部磁盘和外部磁盘上。图 1 详细介绍了 DB2 文件系统的组织，它介绍了在我们的测试中针对实例 db2inst1 和数据库 hadb 的设置。

图 1. DB2 对实例 db2inst1 和数据库 hadb 的高可用设置  
Linux 上的高可用中间件，第 5 部分: IBM DB2 Universal
 Database

[查看原图（大图）](http://img.ddvip.com/2010_04_19/1271664883_ddvip_8757.gif)

　　在该设置中：

　　机器 ha1 用作主 DB2 UDB 数据库机器。

　　机器 ha2 用作节点 ha1 的备用机器。

　　每个节点都有所安装的 DB2 UDB 8.1 的一个本地副本。

　　数据库 hadb 特定的目录（db2inst1/NODE0000/SQL00001 和 db2inst1/NODE0000/sqldbdir）将保留在共享文件系统（/ha）上。

　　安装数据库

　　按照本节 中介绍的步骤在主节点和备用节点上安装 DB2 UDB 8.1。更多信息，请参考 DB2 Information Center：

　 　以 root 用户身份登录。

　　使用下面的命令解压 DB2 UDB 8.1 的安装映像文件：

rm 　-rf　/tmp/db28.1-install　  
mkdir　/tmp/db28.1-install　  
tar　xf　 C48THML.tar　-C　/tmp/db28.1-install

这里的 C48THML.tar 是安装的 tar 文件。

　　设置内核级别：export LD\_ASSUME\_KERNEL=2.4.19。

　 　不要使用 IBM Developer Kit for Linux，即 DB2 安装光盘中提供的 Java 2 Technology Edition。使用 IBM 1.4.2 JDK 来替换 DB2 中提供的 JDK。

cd　 /tmp/db28.1-install/009\_ESE\_LNX\_32\_NLV　  
mv　./db2/linux/java　 ./db2/linux/java.db2　  
ln　-s　/opt/IBMJava2-142　./db2/linux/java

　 　使用下面的命令启动 DB2 安装向导：./db2setup。

　　在这个向导中，使用下面的信息：

　　对于 Product to install，请使用 DB2 UDB Enterprise Server Edition。

　　对于 Group and User IDs，组 ID（gid）和用户 ID（uid）域的值在两台机器上必须匹配。我们使用表 1 中给出的 ID 值。

　 　对于 Partition，请选择 single-partition instance。

　　对于 DB2 Instance Name，请选择 db2inst1。

　　表 1. 安装 DB2 使用的组名/ ID、用户名/ ID

|  |  |  |  |
| --- | --- | --- | --- |
| 组 名 | GID | 用户名 | UID |
| dasadm1 | 2001 | dasusr1 | 2001 |
| db2grp1 | 2002 | db2inst1 | 2002 |
| db2fgrp1 | 2003 | db2fenc1 | 2003 |

　 　创建一个高可用的数据库

　　按照下面的步骤创建高可用的数据库 hadb：

　　以 db2inst1 用户的身份在主节点（ha1）和备用节点（ha2）上登录：su - db2inst1。

　　确保 DB2(R) 实例在 ha1 和 ha2 两个节点上都不会在启动时（以 db2inst1 的身份）使用 db2iauto 工具启动：

cd 　sqllib/bin　  
./db2iauto　-off　db2inst1

　　修改 /etc/inittab 文件，以正确运行 DB2 HA。在节点 ha1 和 ha2 上注释掉在系统启动时启动 DB2 的那一行，如下所示：#fmc:2345:respawn:/opt/IBM/db2/V8.1/bin/db2fmcd #DB2 Fault Monitor Coordinator。

　　在主节点 ha1 上启动 DB2：db2start。

　 　在备用节点（ha2）上以 root 用户的身份使用下面的命令挂载文件系统 /ha：mount /ha。

　　在节 点 ha1 上使用下面的命令创建数据库 hadb：db2 create database hadb on /ha。

　 　在节点 ha1 上使用下面的命令确保可以连接到数据库 hadb 上：db2 connect to hadb。如果成功，就使用这个命令断开连接：db2 connect reset。

　　在节点 ha1 上使用 db2stop 命令停止 DB2。

　　在备用节点（ha2）上以 root 用户的身份使用下面的命令挂载文件系统 /ha： mount /ha。

　　在备用节点上使用下面的命令启动 DB2：db2start。

　 　以 db2instl 用户的身份在节点 ha2 上执行下面的命令，对数据库 hadb 进行 catalog 操作：db2 catalog database hadb on /ha。

　　在节点 ha2 上使用 db2 connect to hadb 命令，确保可以连接到数据库 hadb 上。如果成功，就使用 db2 connect reset 命令断开连接。

　 　在节点 ha2 上使用 db2stop 命令停止 DB2。

　　配置 heartbeat 来管理 DB2

　　现在配 置 /etc/ha.d/haresources 文件（在主节点和备用节点上都要进行），使其包括管理 DB2 进程的脚本。这个脚本是由 heartbeat 提供的。

　　修改文件的一部分如下所示：

ha1.haw2.ibm.com　 9.22.7.46　  
　　　Filesystem::hanfs.haw2.ibm.com:/ha::/ha::nfs::rw,hard　 db2::db2inst1

　　这一行说明在启动 heartbeat 时，hal 使用集群的 IP 地址，挂载共享文件系统，并启动数据库服务器。在停止服务器时，heartbeat 首先要停止数据库服务器，然后卸载共享文件系统，最后放弃 IP 地址。

　　测试 DB2 UDB 的故障迁移

　　本节将介绍如何对高可用的 DB2 数据库 hadb 进行测试。这可能是本文中所介绍的最为棘手的一项工作，因此要仔细阅读以下的内容。

　　在主节点上启动 heartbeat 服务，然后在备用节点上也启动 hartbeat 服务。您可以以 root 用户的身份执行 /etc/rc.d/init.d/heartbeat start 命令。

　　在成功启动 heartbeat 之后，您应该会看到一个新的接口，它使用了您在 ha.cf 文件中配置的 IP 地址。在启动 heartbeat 之后，可以看一下主节点上的日志文件（默认是 /var/log/ha-log），并确保它正在进行 IP 接管，然后又启动了 DB2。使用 ps 命令确保 DB2 进程正在主节点上运行。heartbeat 将不能在备用节点上启动任何上述进程，这只能在主节点失效之后才会发生。

　　在 ha1 节点上以 db2instl 的身份检查数据库的状态：db2 -tf /ha/hahbcode/db2/listdb.sql。这个命令的输出如下所示：

　 　　　　　　　　　　　　　Active　Databases　  
Database　name　　　　　　　　　　　　　　　=　HADB　  
Applications 　connected　currently　　　　　　=　0　  
Database　path　　　　　　　　　　　　　　　=　 /ha/db2inst1/NODE0000/SQL00001/

　　在节点 ha1 上以 db2instl 的身份创建一个测试表（hadb.TestHATable），方法如下：db2 -tf /ha/hahbcode/db2/createdb.sql。

　　现在，在节点 ha1 上以 db2instl 的身份向测试表中插入一行数据：db2 -tf /ha/hahbcode/db2/insertdb.sql。

　　在节点 ha1 上以 db2instl 的身份查看测试表的内容：db2 -tf /ha/hahbcode/db2/selectdb.sql。您应该可以看到在上一个步骤中插入的数据行。运行的结果如下：

[db2inst1@ha1 　db2inst1]$　db2　-tf　/ha/hahbcode/db2/selectdb.sql　  
　　Database　 Connection　Information　  
　Database　server　　　　=　DB2/LINUX　8.1.0　  
　 SQL　authorization　ID　　=　DB2INST1　  
　Local　database　alias　　=　HADB　  
COL1 　　　　COL2　  
-----------　------------　  
　　　　　10　Hello　  
　1　record(s) 　selected.　  
DB20000I　The　SQL　command　completed　successfully.

　 　为了模拟故障迁移的情况，我们只需要在主节点上以 root 用户的身份来停止 heartbeat 即可：/etc/rc.d/init.d/heartbeat stop。您应该会看到在备用节点上，所有的服务都在一分钟之内启动起来了。您可以通过检查 /var/log/ha-log 文件的内容并在备用节点上使用 ps 命令来确认 DB2 正在备用节点上运行。

　　在节点 ha2 上以 db2instl 的身份检查数据库的状态：db2 -tf /ha/hahbcode/db2/listdb.sql。这个命令的输出如下所示：

　 　　　　　　　　　　　　　Active　Databases　  
Database　name　　　　　　　　　　　　　　　=　HADB　  
Applications 　connected　currently　　　　　　=　0　  
Database　path　　　　　　　　　　　　　　　=　 /ha/db2inst1/NODE0000/SQL00001/

　　在节点 ha2 上以 db2instl 的身份查看测试表（hadb.TestHATable）的内容：db2 -tf /ha/hahbcode/db2/selectdb.sql。您应该可以看到当 DB2 在节点 ha1 上运行时所插入的数据。我们运行的结果如下：

[db2inst1@ha2　db2inst1]$　db2　-tf　 /ha/hahbcode/db2/selectdb.sql　  
　　Database　Connection　Information　  
　 Database　server　　　　=　DB2/LINUX　8.1.0　  
　SQL　authorization　ID　　=　 DB2INST1　  
　Local　database　alias　　=　HADB　  
COL1　　　　COL2　  
----------- 　------------　  
　　　　　10　Hello　  
　1　record(s)　selected.　  
DB20000I　 The　SQL　command　completed　successfully.

这说明在从主节点到备用节点进 行故障迁移时，数据得以维护下来了。

　　在节点 ha2 上以 db2instl 的身份向测试表中插入一行数据：db2 -tf /ha/hahbcode/db2/insertdb.sql。

　　您 可以在主节点上以 root 用户的身份启动 heartbeat 来启动主节点：/etc/rc.d/init.d/heartbeat start。您应该会看到所有的服务器在一分钟之内都在主节点上启动起来了。您可以通过检查 /var/log/ha-log 文件的内容并在主节点上使用 ps 命令来确认 DB2 正在主节点上运行。

　　在节点 ha1 上以 db2instl 的身份检查测试表的内容：db2 -tf /ha/hahbcode/db2/selectdb.sql。您会看到这个表中包含两行数据了。我们运行的结果如下：

[db2inst1@ha1 　db2inst1]$　db2　-tf　/ha/hahbcode/db2/selectdb.sql　  
　　Database　 Connection　Information　  
　Database　server　　　　=　DB2/LINUX　8.1.0　  
　 SQL　authorization　ID　　=　DB2INST1　  
　Local　database　alias　　=　HADB　  
COL1 　　　　COL2　  
-----------　------------　  
　　　　　10　Hello　  
　　　　　10　 Hello　  
　2　record(s)　selected.　  
DB20000I　The　SQL　command　completed　 successfully.

　　这说明在从备用节点到主节点进行故障迁移时，数据得以维护下来了。

　　结 束语

　　关键的数据库应用程序都需要一种健壮的策略来防止数据丢失，并保证数据存储的高可用性。DB2 UDB 简化了具有高可用特性的大型数据库的管理。

　　在这篇文章中，您已经看到了如何使用开源软件和廉价的硬件来简单而且低成本地为 DB2 UD 数据库实现一个 HA 的方案。