

# Setting Up IBM Data Server Manager as a Highly Available Service

IBM® Data Server Manager is a web-based, integrated database management tools platform that manages the following databases:

- IBM Db2® for Linux, UNIX, Windows
- IBM Db2 for z/OS® databases
- IBM dashDB® for Analytics
- IBM Db2 on Cloud (formerly IBM dashDB for Transactions)
- IBM dashDB Local

With IBM Data Server Manager, you can monitor, analyze, tune, and administer Db2 databases. This document describes how to build the Data Server Manager high availability (HA) environment, based on IBM Tivoli® System Administration for Multiplatforms (Tivoli SA MP). Two servers with the Linux operating system are needed. The operating system used in this document is Red Hat Enterprise Linux Server Release 6.7.

Figure 1 shows the Data Server Manager HA structure. In the figure, TSA identifies the IBM Tivoli SA MP product in the environment.

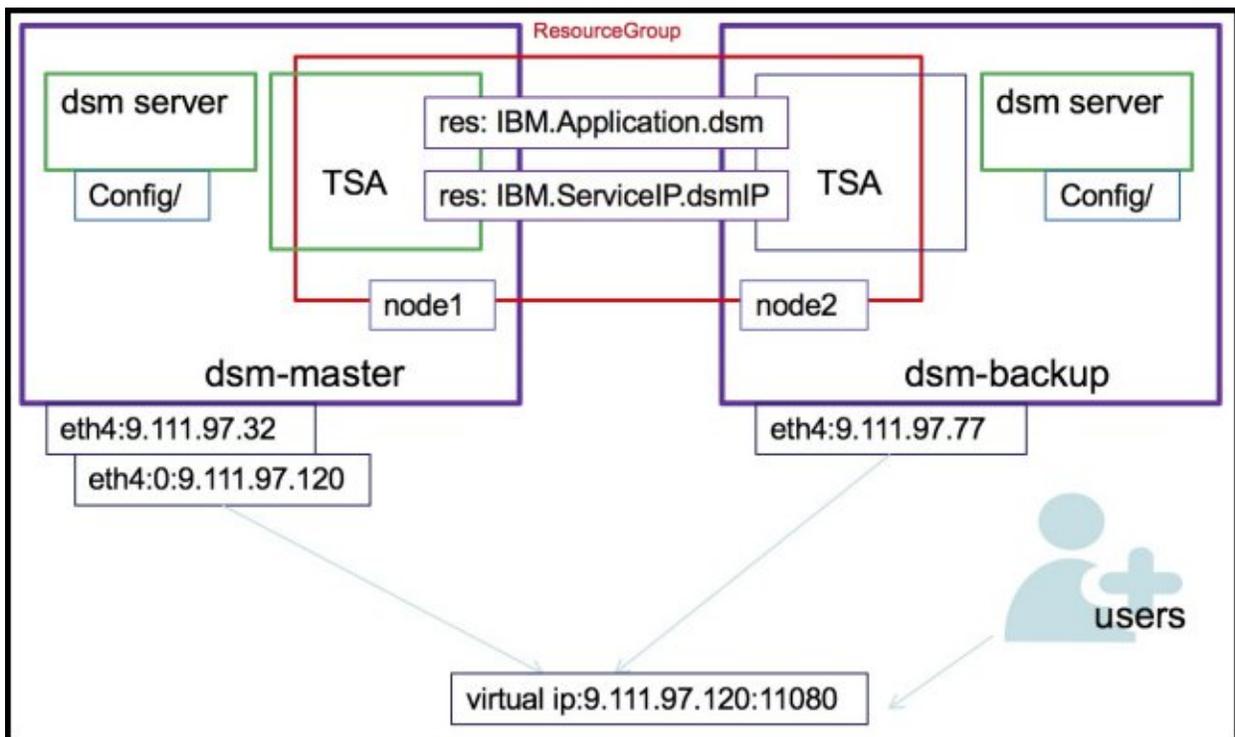


Figure 1. Data Server Manager HA structure

## Install Tivoli SA MP

Generally, the Tivoli SA MP installation package is bound together with the Db2 installation package. You need to manually install Tivoli SA MP on both the master and backup nodes. The Tivoli SA MP installation package can be obtained from the Db2 installation package directory `server/db2/linuxamd64/tsamp`. Perform the following steps to install Tivoli SA MP:

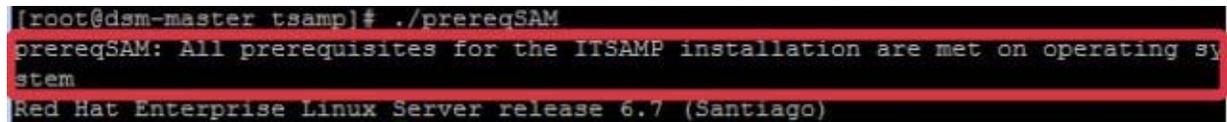
1. Install the prerequisite components by running the following commands (you must have root authority).

```
[root@dsm-master tsamp]# yum install ksh libstdc++.i686
compat-libstdc++-33.x86_64 compat-libstdc++-33.i686 -y
[root@dsm-master tsamp]# echo 'multilib_policy=all' >> /etc/yum.conf
[root@dsm-master tsamp]# yum install pam.i686
```

2. Run the following Tivoli SA MP command, to verify that Tivoli SA MP is ready to be installed:

```
[root@dsm-master tsamp]# ./prereqSAM
```

Figure 2 shows the output of the `prereqSAM` command.



```
[root@dsm-master tsamp]# ./prereqSAM
prereqSAM: All prerequisites for the IISAMP installation are met on operating system
Red Hat Enterprise Linux Server release 6.7 (Santiago)
```

Figure 2. Command `prereqSAM` output

3. Install Tivoli SA MP (root authority needed) using the following command:

```
[root@dsm-master tsamp]# ./installSAM
```

4. Enter `y` to accept the license agreement. Figure 3 shows the `installSAM` command.



```
D/N: L-TKAI-8GWJPQ
P/N: L-TKAI-8GWJPQ
installSAM: To accept all terms of the preceding License Agreement and License Information type
'y', anything else to decline
y
```

Figure 3. The `installSAM` command

Figure 4 shows the results of executing the `installSAM` command.

```
installSAM: The following license is installed
Product: IBM Tivoli System Automation for Multiplatforms 3.2
Creation date: Wed 19 Aug 2009 12:00:01 AM CST
Expiration date: Thu 31 Dec 2037 12:00:01 AM CST

Subsystem      Group      PID      Status
ctrmc          rsct       26132    active
installSAM: Warning: Must set CT_MANAGEMENT_SCOPE=

installSAM: All packages were installed successfully
[root@redhat1 tsamp]#
```

Figure 4. Results of the `installSAM` command

## Install `rsync` and `inotify` tools

Data Server Manager user data in the `ibm-datasrvrmgr/Config` folder is sensitive. The Data Server Manager HA cluster keeps this data safe by synchronizing backups across nodes. With the installation of `rsync` and `inotify` tools, any changes in this folder are synchronized from one node to the other node. If Data Server Manager is failing on a node, the other node starts with the same configuration data.

The `rsync` and `inotify` tools need to be installed on both master and backup nodes. Run the following commands on both nodes. These tools monitor and synchronize `ibm-datasrvrmgr/Config` file changes that are on the two nodes (root authority is needed):

```
[root@dsm-master tsamp]# yum install rsync
[root@dsm-master tsamp]# yum install inotify-tools
```

## Creating a Tivoli SA MP cluster

Two nodes act as a Tivoli SA MP cluster. For the remainder of this document, the host names for the two nodes are *dsm-master* and *dsm-backup*. You can use different host names, but be sure to replace the host names in subsequent steps where the names are used. Complete these steps.

1. On both master and backup nodes, update the `/etc/hosts` file with the fully qualified domain name (root authority needed). Figure 5 shows the file contents.

```
[root@dsm-master tsamp]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1        localhost localhost.localdomain localhost6 localhost6.localdomain6
#ITaaS Puppet Server:
9.123.108.166 puppet.cn.ibm.com puppet
10.0.10.215 dsm-master
10.0.10.216 dsm-backup
```

Figure 5. The contents of `/etc/hosts`

2. (Optional) Modify the host name value on both nodes by editing the `/etc/sysconfig/network` system file. Set `HOSTNAME` to `dsm-master` on the master node (Figure 6) and `dsm-backup` on the backup node (Figure 7); root authority is needed:

```
[root@dsm-master ~]# cat /etc/sysconfig/network
```

```
[root@dsm-master tsamp]# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=dsm-master
```

Figure 6. Set hostname for `dsm-master`

```
[root@dsm-backup ~]# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=dsm-backup
```

Figure 7. Set hostname for `dsm-backup`

3. On the master node, run the following command:

```
[root@dsm-master ~]# hostname dsm-master
```

4. On the backup node, run the following command:

```
[root@dsm-master ~]# hostname dsm-backup
```

5. Reboot the master and backup nodes after these configuration changes are complete.

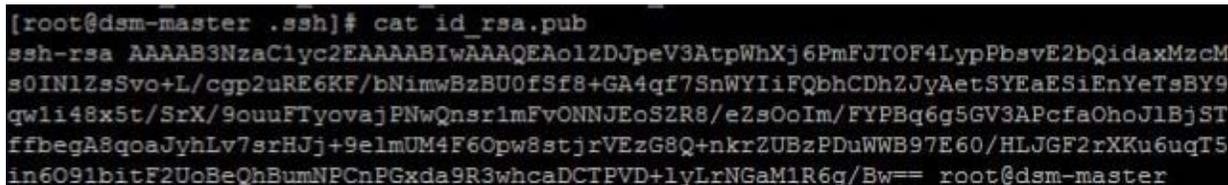
## Configuration on both nodes

Configure Secure Shell (SSH) on both nodes so that file changes can be synchronized between these two nodes:

1. Run the `ssh-keygen` command on both the master and backup nodes:

```
[root@dsm-master .ssh]# ssh-keygen -t rsa
```

2. Copy the contents of the `id_rsa.pub` file (Figure 8) on the master node and append it to the `/root/.ssh/authorized_keys` file on the backup node. Also copy the contents of `id_rsa.pub` on the backup node and append it to the `/root/.ssh/authorized_keys` file on the master node.



```
[root@dsm-master .ssh]# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAo1ZDJpeV3AtpWhXj6PmFJTOf4LypPbsvE2bQidaxMzcM
s0IN1ZsSvo+L/cgp2uRE6KF/bNimwBzBU0fSf8+GA4qf7SnWYIiFQbhCDhZJyAetSYEaESiEnYeTsBY9
qwli48x5t/SrX/9ouuFTyovajPNwQnsrlmFvONNJEoSZR8/eZsOoIm/FYPBq6g5GV3APcfaOhoJlBjST
ffbega8qoaJyhLv7srHJj+9elmUM4F6Opw8stjrVEzG8Q+nkrZUBzPDuWwB97E60/HLJGF2rXKu6uqT5
in6O91bitF2UoBeQhBumNPCnPGxda9R3whcaDCTPVD+lyLrNGaM1R6g/Bw== root@dsm-master
```

Figure 8. Content of the `id_rsa.pub` file

3. Run the Tivoli SA MP `preprnode` command on both nodes before creating a domain (root authority needed):

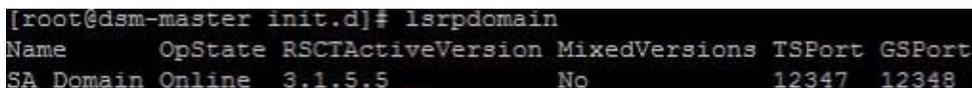
```
[root@dsm-master .ssh]# preprnode dsm-master dsm-backup
```

## Creating a domain for two nodes

Run the Tivoli SA MP commands to create a domain for Data Server Manager HA cluster only on the master node. To start the domain and check its status, make `dsm` a resource on the master node (root authority needed) by using the following commands:

```
[root@dsm-master init.d]# mkrpdomain SA_Domain dsm-master dsm-backup
[root@dsm-master init.d]# startdomain SA_Domain
[root@dsm-master init.d]# lsrpdomain
```

Figure 9 shows the `lsrpdomain` command execution.



```
[root@dsm-master init.d]# lsrpdomain
Name      OpState RSCActiveVersion MixedVersions TSPort GSPort
SA_Domain Online  3.1.5.5         No           12347 12348
```

Figure 9. Executing the `lsrpdomain` command

## Creating dsm resource

To create a resource, you need a script that includes start, stop, and status commands. You also need a definition file. Create a resource for Data Server Manager as follows:

1. On both the dsm-master and dsm-backup nodes, create a script with the name **dsm**. The script includes the Tivoli SA MP definition of start, stop, and status, and aligns with Data Server Manager start, stop, and status scripts. Save the script in a directory, such as **/etc/init.d/**, illustrated in this example. The directory highlighted in bold is the directory where Data Server Manager is installed (root authority might be needed).

```
#!/bin/bash
OPSTATE_ONLINE=1
OPSTATE_OFFLINE=2
Action=${1}
case ${Action} in
  start)
    /root/ibm-datasrvrmgr/bin/start.sh >/dev/null 2>&1
    logger -i -t "SAM-dsm" "Apache started"
    RC=0
    ;;
  stop)
    /root/ibm-datasrvrmgr/bin/stop.sh >/dev/null 2>&1
    logger -i -t "SAM-dsm" "Apache stopped"
    RC=0
    ;;
  status)
    active=`/root/ibm-datasrvrmgr/bin/status.sh | grep " ACTIVE" | wc -l`
    if [ "$active" -eq "1" ]
    then
      RC=${OPSTATE_ONLINE}
    else
      RC=${OPSTATE_OFFLINE}
    fi
    ;;
esac
echo "RC:"${RC}
exit $RC
```

Figure 10 shows the content of the **dsm** script.

```

[root@dsm-master init.d]# cat dsm
#!/bin/bash

OPSTATE_ONLINE=1
OPSTATE_OFFLINE=2

Action=${1}

case ${Action} in
    start)
        /root/ibm-datasrvrmgr/bin/start.sh >/dev/null 2>&1
        logger -i -t "SAM-dsm" "Apache started"
        RC=0
        ;;
    stop)
        /root/ibm-datasrvrmgr/bin/stop.sh >/dev/null 2>&1
        logger -i -t "SAM-dsm" "Apache stopped"
        RC=0
        ;;
    status)
        active=`/root/ibm-datasrvrmgr/bin/status.sh | grep " ACTIVE" | wc -l`
        if [ "$active" -eq "1" ]
        then
            RC=${OPSTATE_ONLINE}
        else
            RC=${OPSTATE_OFFLINE}
        fi
        ;;
esac
echo "RC:"`${RC}`
exit $RC

```

Figure 10. Contents of the dsm script

2. On the dsm-master node, create a definition file named `dsm.def`. The directory highlighted in bold is the directory where the `dsm` script is created. Save the file in the same directory as the `dsm` script, which is `/etc/init.d/`.

```

PersistentResourceAttributes::
    Name="dsm"
    StartCommand="/etc/init.d/dsm start"
    StopCommand="/etc/init.d/dsm stop"
    MonitorCommand="/etc/init.d/dsm status"
    MonitorCommandPeriod=5
    MonitorCommandTimeout=5
    NodeNameList={"dsm-master", "dsm-backup"}
    StartCommandTimeout=90
    StopCommandTimeout=90
    UserName="root"
    ResourceType=1

```

Figure 11 shows the contents of the `dsm.def` file.

```
[root@dsm-master init.d]# cat dsm.def
PersistentResourceAttributes::
  Name="dsm"
  StartCommand="/etc/init.d/dsm start"
  StopCommand="/etc/init.d/dsm stop"
  MonitorCommand="/etc/init.d/dsm status"
  MonitorCommandPeriod=5
  MonitorCommandTimeout=5
  NodeNameList={"dsm-master","dsm-backup"}
  StartCommandTimeout=90
  StopCommandTimeout=90
  UserName="root"
  ResourceType=1
```

Figure 11. Content of dsm.def file

3. Run the **export** command on the master node (root authority needed):

```
[root@dsm-master init.d]# export CT_MANAGEMENT_SCOPE=2
[root@dsm-master init.d]# echo 'export CT_MANAGEMENT_SCOPE=2' >>
~/.bash_profile
```

4. Generate the dsm resource on the master node by running the following command in the directory where **dsm.def** is located (root authority needed):

```
[root@dsm-master init.d]]# mkrsrc -f dsm.def IBM.Application
[root@dsm-master init.d]]# lsrsrc -s "Name='dsm'" IBM.Application
```

Figure 12 shows generating the dsm resource on the master node.

```
[root@dsm-master init.d]# lsrsrc -s "Name='dsm'" IBM.Application
Resource Persistent Attributes for IBM.Application
resource 1:
  Name = "dsm"
  ResourceType = 0
  AggregateResource = "0x2028 0xffff 0x6ab27f3c 0xc98619cd 0x94895419
0x8ef2e060"
  StartCommand = "/etc/init.d/dsm start"
  StopCommand = "/etc/init.d/dsm stop"
  MonitorCommand = "/etc/init.d/dsm status"
  MonitorCommandPeriod = 5
  MonitorCommandTimeout = 5
  StartCommandTimeout = 90
  StopCommandTimeout = 90
  UserName = "root"
  RunCommandsSync = 1
  ProtectionMode = 0
  HealthCommand = ""
  HealthCommandPeriod = 10
  HealthCommandTimeout = 5
  InstanceName = ""
  InstanceLocation = ""
  SetHealthState = 0
  MovePrepareCommand = ""
  MoveCompleteCommand = ""
  MoveCancelCommand = ""
  CleanupList = {}
  CleanupCommand = ""
  CleanupCommandTimeout = 10
  ProcessCommandString = ""
  ResetState = 0
  ReRegistrationPeriod = 0
  CleanupNodeList = {}
  MonitorUserName = ""
  ActivePeerDomain = "SA_Domain"
  NodeNameList = ("dsm-backup")
resource 2:
  Name = "dsm"
  ResourceType = 0
```

Figure 12. Creating dsm resource

## Creating a virtual Internet Protocol (IP) resource

Choose an available IP address as the virtual IP for the Data Server Manager service. Using this virtual IP makes Data Server Manager switching between the master and backup nodes transparent to users. Regardless of the Data Server Manager service active on either node, users see only the virtual IP. Complete these steps:

1. Choose an available IP (9.111.97.120 is used in this example) as the virtual IP. Next create the virtual IP resource on the master node (root authority needed):

```
[root@dsm-master init.d]]# mkrsrc IBM.ServiceIP
NodeNameList="{ 'dsm-master', 'dsm-backup' }" Name="dsmIP"
NetMask=255.255.255.0 IPAddress=9.111.97.120 ResourceType=1
[root@dsm-master init.d]]# lsrsrc -s "Name='dsmIP'" IBM.ServiceIP
```

Run the **lsrsrc IBM.NetworkInterface** command. This command shows that the IP resource is bound on `eth4` (this variable is used later).

```
[root@dsm-master init.d]]# lsrsrc IBM.NetworkInterface
```

Figure 13 shows the **lsrsrc IBM.NetworkInterface** command output.

```

resource 3:
  Name           = "eth4"
  DeviceName     = ""
  IPAddress      = "9.111.97.77"
  SubnetMask     = "255.255.255.0"
  Subnet         = "9.111.97.0"
  CommGroup      = "CG1"
  HeartbeatActive = 1
  Aliases        = {}
  DeviceSubType  = 1
  LogicalID      = 0
  NetworkID      = 0
  NetworkID64    = 0
  PortID         = 0
  HardwareAddress = "00:50:56:b2:6f:3c"
  DevicePathName = ""
  IPVersion      = 4
  Role           = 0
  ActivePeerDomain = "SA_Domain"
  NodeNameList   = {"dsm-backup"}
resource 4:
  Name           = "eth4"
  DeviceName     = ""
  IPAddress      = "9.111.97.32"
  SubnetMask     = "255.255.255.0"
  Subnet         = "9.111.97.0"
  CommGroup      = "CG1"
  HeartbeatActive = 1
  Aliases        = {}
  DeviceSubType  = 1
  LogicalID      = 0
  NetworkID      = 0
  NetworkID64    = 0
  PortID         = 0
  HardwareAddress = "00:50:56:b2:5e:fd"
  DevicePathName = ""

```

Figure 13. Output of the `lsrsrc IBM.NetworkInterface` command

2. Run the following command to bind the virtual IP to the two physical IPs for the master and backup nodes. The `eth4` variable was determined in step 1 of this section (root authority needed):

```

[root@dsm-master init.d]# mkequ netequ
IBM.NetworkInterface:eth4:dsm-master,eth4:dsm-backup

```

3. Display the Equivalency information by using the following command:

```

[root@dsm-master init.d]# lsequ -e netequ

```

Figure 14 shows the output of the `lsequ -e netequ` command.

```
[root@dsm-master ~]# lsequ -e netequ
Displaying Equivalency information:
For Equivalency "netequ".

Equivalency 1:
  Name                = netequ
  MemberClass          = IBM.NetworkInterface
  Resource:Node[Membership] = {eth4:dsm-master,eth4:dsm-backup}
  SelectString         = ""
  SelectFromPolicy     = ANY
  MinimumNecessary    = 1
  Subscription        = {}
  Color               = 0
  ActivePeerDomain    = SA_Domain
  ConfigValidity      =
```

Figure 14. Output of the `lsequ -e netequ` command

## Creating and starting a resource group

Add the previous resources to a resource group. Start the resource group and check its status with the following steps:

1. Create a resource group on the master node. Include the application resource "dsm" and the service IP resource "dsmIP" (root authority needed):

```
[root@dsm-master init.d]# mkrsg dsmsg
[root@dsm-master init.d]# lsrg -g dsmsg
```

2. Add the two resources to a resource group (root authority needed):

```
[root@dsm-master init.d]# addrgmbr -g dsmsg IBM.Application:dsm
[root@dsm-master init.d]# addrgmbr -g dsmsg IBM.ServiceIP:dsmIP
[root@dsm-master init.d]# lsrg -m
```

Figure 15 shows the output of the `lsrg -m` command.

```
[root@dsm-master ~]# lsrg -m
Displaying Member Resource information:
Class:Resource:Node[ManagedResource] Mandatory MemberOf OpState WinSource Location
IBM.ServiceIP:dsmIP True dsmsg Offline
IBM.Application:dsm True dsmsg Offline
```

Figure 15. Output of the `lsrg -m` command, for dsm-master

3. Define the dependencies between the resources on the master node (root authority needed):

```
root@dsm-master init.d]# mkrel -p DependsOn -S IBM.Application:dsm -G
IBM.ServiceIP:dsmIP dsm_dependson_dsmIP
[root@dsm-master init.d]# mkrel -p DependsOn -S IBM.ServiceIP:dsmIP -G
IBM.Equivalency:netequ dsmIP_dependson_netequ
[root@dsm-master haoshinit.dwei]# lsrel
```

Figure 16 shows the output of `lsrel` command.

```
[root@dsm-master ~]# lsrel
Displaying Managed Relations :

Name                               Class:Resource:Node[Source] ResourceGroup[Source]
dsmIP_dependson_netequ             IBM.ServiceIP:dsmIP      dsmrg
dsm_dependson_dsmIP                IBM.Application:dsm       dsmrg
```

Figure 16. Output of the lsrel command, for dsm-master managed relationships

4. Start the resource group on the master node (root authority needed):

```
[root@dsm-master init.d]# chrg -o online dsmrg
[root@dsm-master init.d]# lssam -V
```

Figure 17 shows the resource group status.

```
[root@dsm-master ~]# lssam -V
Online IBM.ResourceGroup:dsmrg Nominal=Online
  |- Online IBM.Application:dsm          -
    |- Offline IBM.Application:dsm:dsm-backup |
    ' - Online IBM.Application:dsm:dsm-master DO
  ' - Online IBM.ServiceIP:dsmIP IP=9.111.97.120 <' -
    |- Offline IBM.ServiceIP:dsmIP:dsm-backup |
    ' - Online IBM.ServiceIP:dsmIP:dsm-master DO
Online IBM.Equivalency:netequ
  |- Online IBM.NetworkInterface:eth4:dsm-backup
  ' - Online IBM.NetworkInterface:eth4:dsm-master
```

Figure 17. Resource group status

## Adding a notification resource

To synchronize data between the master and backup nodes, notification resources can be added on both nodes as follows:

1. Create a script named `syncup.sh` and save it on both the dsm-master and the dsm-backup nodes. Save `syncup.sh` in a directory of your choosing, but be sure to refer to the proper location in subsequent steps. In these examples, the script is saved in `/root/syncup`. Replace the value that is assigned to `inotifyDir` with the directory where the `inotifywait` command resides on the master and backup nodes. The following command can be used to locate the directory:

```
[root@dsm-master bin]# locate inotifywait
```

Figure 18 shows the output of the `locate inotifywait` command.

```
[root@dsm-master ~]# locate inotifywait
/usr/bin/inotifywait
```

Figure 18. Output of the locate inotifywait command

- a. Replace the value that is assigned to `destIP` with the IP address of the pair server. On the master

node, destIP is the IP address of backup node; and on the backup node, destIP is the IP address of master node.

*b.* Replace the value that is assigned to srcDir with the directory where Data Server Manager is installed on the master node. Replace the value that is assigned to destDir with the directory where Data Server Manager is installed on the backup node.

The Data Server Manager folder Config/ is defined to synchronize the master and backup nodes with the specified policy in this script:

```
[root@dsm-master syncup]# cat syncup.sh
#!/bin/bash
inotifyDir="/usr/bin"
srcDir="/root/ibm-datasrvrmgr/"
destIP="9.111.97.77"
destDir="/root/ibm-datasrvrmgr/"
scriptDir="/root/syncup"
dir=""
action=""
file=""
folderDir=""
rm -f ${scriptDir}/*.log
$inotifyDir/inotifywait -rmq -e modify,create,delete,attrib,move
${srcDir}Config | while read event
do
    dir=$(echo ${event}|cut -d\ -f1)
    action=$(echo ${event}|cut -d\ -f2)
    file=$(echo ${event}|cut -d\ -f3)
    echo "$(date) $event" >> ${scriptDir}/event.log
    folderDir=${dir}*Config/
    if [[ $file == "" || $file == .* ]]
    then
        continue
    elif [[ $action == DELETE* ]]
    then
        echo -e "$(date) \n Warning: You've tried to delete
important file $file. It has been recovered from standby server." >>
${scriptDir}/ha_config.log
        rsync -avzP root@$destIP:$destDir Config/$folderDir/$file
$srcDir Config/$folderDir >> ${scriptDir}/rsync.log
2>&1
    else
        echo -e "$(date) \n Backup change of $file to standby
server." >> ${scriptDir}/ha_config.log
        rsync -avzP $srcDir Config/$folderDir/$file
root@$destIP:$destDir Config/$folderDir >> ${scriptDir}/rsync.log
2>&1
    fi
done
```

2. On both the dsm-master and dsm-backup nodes, create a script with the name **inotify**. The script contains start, stop, and status information. Save the script in a directory of your choosing. In this example, the script is saved in the /etc/init.d directory. The /root/syncup directory is where **syncup.sh** was created in the previous step.

```
[root@dsm-master init.d]# cat inotify
#!/bin/bash
OPSTATE_ONLINE=1
```

```

OPSTATE_OFFLINE=2
Action=${1}
case ${Action} in
    start)
        nohup bash /root/syncup/syncup.sh >> syncup.log 2>&1 &
        logger -i -t "SAM-inotify" "inotify started"
        RC=0
        ;;
    stop)
        killall inotifywait
        logger -i -t "SAM-inotify" "inotify stopped"
        RC=0
        ;;
    status)
        ps ax |grep -v "grep"| grep inotifywait>/dev/null
        if [ $? == 0 ]
        then
            RC=${OPSTATE_ONLINE}
        else
            RC=${OPSTATE_OFFLINE}
        fi
        ;;
esac
exit $RC

```

3. Create a definition file with name `inotify.def` on the dsm-master node. The directory highlighted in bold in the following example is the directory where `inotify` is created. Save the file in the same directory where the `inotify` script was saved in a previous step. In this example, that directory is `/etc/init.d`.

```

[root@dsm-master init.d]# cat inotify.def
PersistentResourceAttributes::
    Name="inotify"
    StartCommand="/etc/init.d/inotify start"
    StopCommand="/etc/init.d/inotify stop"
    MonitorCommand="/etc/init.d/inotify status"
    MonitorCommandPeriod=5
    MonitorCommandTimeout=5
    NodeNameList={"dsm-master"}
    StartCommandTimeout=10
    StopCommandTimeout=10
    UserName="root"
    ResourceType=0

```

4. Create a definition file with name `inotify2.def` on the dsm-backup node. The directory in bold is where `inotify` is created. Save the file in the same directory where the `inotify` script was saved in a previous step. In this example, that directory is `/etc/init.d`.

```

[root@dsm-master init.d]# cat inotify2.def
PersistentResourceAttributes::
    Name="inotify2"
    StartCommand="/etc/init.d/inotify start"
    StopCommand="/etc/init.d/inotify stop"
    MonitorCommand="/etc/init.d/inotify status"
    MonitorCommandPeriod=5
    MonitorCommandTimeout=5
    NodeNameList={"dsm-backup"}
    StartCommandTimeout=10

```

```
StopCommandTimeout=10
UserName="root"
ResourceType=0
```

5. Make `inotify` a resource on the master node by running the following command in the directory where `inotify.def` is created (root authority needed):

```
[root@dsm-master init.d]# mkrsrc -f inotify.def IBM.Application
```

Make `inotify2` a resource on the backup node by running the following command in the directory where `inotify.def` is created (root authority needed):

```
[root@dsm-master init.d]# mkrsrc -f inotify2.def IBM.Application
```

Make a resource group on the master node:

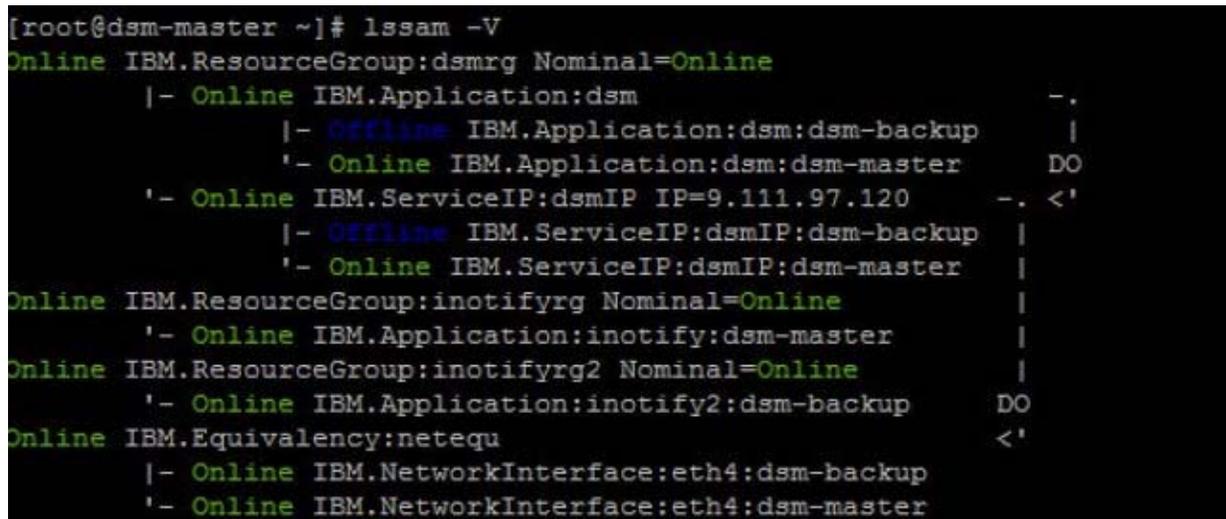
```
[root@dsm-master init.d]# mkrgrp inotifyrg
[root@dsm-master init.d]# addrgmbr -g inotifyrg
IBM.Application:inotify:dsm-master
[root@dsm-master init.d]# chrg -o online inotifyrg
[root@dsm-master init.d]# mkrgrp inotifyrg2
[root@dsm-master init.d]# addrgmbr -g inotifyrg2
IBM.Application:inotify2:dsm-backup
[root@dsm-master init.d]# chrg -o online inotifyrg2
```

### Checking resource status

Use the `lssam` command to display the status of the cluster; this command shows which node is online:

```
[root@dsm-master init.]# lssam -V
```

Figure 19 shows the output of `lssam -v` command .



```
[root@dsm-master ~]# lssam -V
Online IBM.ResourceGroup:dsmrg Nominal=Online
  |- Online IBM.Application:dsm
    |- Offline IBM.Application:dsm:dsm-backup
    '- Online IBM.Application:dsm:dsm-master DO
  '- Online IBM.ServiceIP:dsmIP IP=9.111.97.120 -' <'
    |- Offline IBM.ServiceIP:dsmIP:dsm-backup
    '- Online IBM.ServiceIP:dsmIP:dsm-master
Online IBM.ResourceGroup:inotifyrg Nominal=Online
  '- Online IBM.Application:inotify:dsm-master
Online IBM.ResourceGroup:inotifyrg2 Nominal=Online
  '- Online IBM.Application:inotify2:dsm-backup DO
Online IBM.Equivalency:netegu <'
  |- Online IBM.NetworkInterface:eth4:dsm-backup
  '- Online IBM.NetworkInterface:eth4:dsm-master
```

Figure 19. Output of the `lssam -V` command to display status of the cluster

## How HA works while Data Server Manager is failing

While the Data Server Manager is failing, HA works in the following ways:

- If Data Server Manager is stopped on the active node, Tivoli SA MP will try to restart it.
- If Data Server Manager is restarted successfully, it will continue to run on the same node.
- When Tivoli SA MP fails to restart Data Server Manager after several attempts (the number of attempts can be configured), Data Server Manager HA will fail over to another node. For example, Tivoli SA MP will switch from the master node to the backup node or from the backup node to the master node. To observe this behavior, you can temporarily move (or rename) some critical Data Server Manager installation files. With critical files missing, Tivoli SA MP cannot restart Data Server Manager on the master node. Tivoli SA MPs will then switch to the backup node and start Data Server Manager there. You can run the `lssam -v` command, which indicates that the backup node is online and the master node is offline.

In the following example, `ibm-datasrvrmgr/bin` is renamed to `ibm-datasrvrmgr/bin_bak`. With this change, Tivoli SA MP cannot start Data Server Manager.

```
[root@dsm-backup ibm-datasrvrmgr]# mv bin bin_bak
```

Figure 20 shows the output of these file commands.

```
[root@dsm-master ibm-datasrvrmgr]# ls
bin          dscomponents_with_rep  java    samples  setup.conf  support  work
Config      dsutil                 license setup    setup.conf.sample  undo_setup.bat
dscomponents iso-swid               logs    setup.bat setup.sh     wlp
[root@dsm-master ibm-datasrvrmgr]# mv bin bin_bak
[root@dsm-master ibm-datasrvrmgr]# ls
bin_bak     dscomponents_with_rep  java    samples  setup.conf  support  work
Config      dsutil                 license setup    setup.conf.sample  undo_setup.bat
dscomponents iso-swid               logs    setup.bat setup.sh     wlp
```

Figure 20. Output of the `mv bin bin_bak` command

Run the `lssam -v` command. Note that the `dsm-master` node is now in a "Pending online" state. Tivoli SA MP is trying to restart Data Server Manager on the master node.

Figure 21 shows the output of command `lssam -v` at this point.

```
[root@dsm-master ibm-datasrvrmgr]# lssam -V
Pending online IBM.ResourceGroup:dsmrg Nominal=Online
|- Pending online IBM.Application:dsm -
|   |- Offline IBM.Application:dsm:dsm-backup |
|   '- Pending online IBM.Application:dsm:dsm-master DO
'- Online IBM.ServiceIP:dsmIP IP=9.111.97.120 - <'
|   |- Offline IBM.ServiceIP:dsmIP:dsm-backup |
|   '- Online IBM.ServiceIP:dsmIP:dsm-master |
Online IBM.ResourceGroup:inotifyrg Nominal=Online
|- Online IBM.Application:inotify:dsm-master |
Online IBM.ResourceGroup:inotifyrg2 Nominal=Online
|- Online IBM.Application:inotify2:dsm-backup DO
Online IBM.Equivalency:netegu <'
|   |- Online IBM.NetworkInterface:eth4:dsm-backup
|   '- Online IBM.NetworkInterface:eth4:dsm-master
```

Figure 21. Output of the `lssam -v` command, with `dsm-master` in pending online status

After several failed attempts to start Data Server Manager on the master node, Tivoli SA MP starts Data Server Manager on the backup node. Run the command `lssam -v` to see that the dsm-backup node is online now.

Figure 22 shows the output of command `lssam -v` at this point.

```
[root@dsm-master ibm-datasrvrmgr]# lssam -v
Online IBM.ResourceGroup:dsmrg Control=MemberInProblemState Nominal=Online
|- Online IBM.Application:dsm Control=MemberInProblemState
|- Online IBM.Application:dsm dsm-backup
|- Offline IBM.Application:dsm:dsm-master
'- Online IBM.ServiceIP:dsmIP IP=9.111.97.120
|- Online IBM.ServiceIP:dsmIP:dsm-backup
'- Offline IBM.ServiceIP:dsmIP:dsm-master
Online IBM.ResourceGroup:inotifyrg Nominal=Online
'- Online IBM.Application:inotify:dsm-master
Online IBM.ResourceGroup:inotifyrg2 Nominal=Online
'- Online IBM.Application:inotify2:dsm-backup
Online IBM.Equivalency:netegu
|- Online IBM.NetworkInterface:eth4:dsm-backup
'- Online IBM.NetworkInterface:eth4:dsm-master
```

Figure 22. Output of the `lssam -v` command to see that the dsm-backup node is online

Fix the issue on the master node by renaming `ibm-datasrvrmgr/bin_bak` back to `ibm-datasrvrmgr/bin`.

Reset the dsm-master node resource by running the following command:

```
[root@dsm-backup ibm-datasrvrmgr]# resetrsrc -s 'Name="dsm" && NodeNameList={"dsm-master"}' IBM.Application
```

Run the `lssam -v` command to see that the dsm-master is offline now.

Figure 23 shows the output of command `lssam -v` at this point.

```
[root@dsm-master ibm-datasrvrmgr]# lssam -v
Online IBM.ResourceGroup:dsmrg Nominal=Online
|- Online IBM.Application:dsm
|- Online IBM.Application:dsm:dsm-backup
|- Offline IBM.Application:dsm:dsm-master
'- Online IBM.ServiceIP:dsmIP IP=9.111.97.120
|- Online IBM.ServiceIP:dsmIP:dsm-backup
'- Offline IBM.ServiceIP:dsmIP:dsm-master
Online IBM.ResourceGroup:inotifyrg Nominal=Online
'- Online IBM.Application:inotify:dsm-master
Online IBM.ResourceGroup:inotifyrg2 Nominal=Online
'- Online IBM.Application:inotify2:dsm-backup
Online IBM.Equivalency:netegu
|- Online IBM.NetworkInterface:eth4:dsm-backup
'- Online IBM.NetworkInterface:eth4:dsm-master
```

Figure 23. Output of `lssam -v` command to see the dsm-master is offline

Users do not need to do anything manually during failover. They still visit the same website regardless of which node is running Data Server Manager. This transparency is the result of configuring virtual IP resources and binding them with Data Server Manager resources.

Figure 24 shows that the Data Server Manager website with a virtual IP is always accessible during failover, except the intervals that Tivoli SA MP attempts to restart Data Server Manager on the master node.

```
[root@dsm-master ibm-datastrmgr]# curl -v http://9.111.97.120:11080
* About to connect() to 9.111.97.120 port 11080 (#0)
* Trying 9.111.97.120... connected
* Connected to 9.111.97.120 (9.111.97.120) port 11080 (#0)
> GET / HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.18 Basic ECC zlib/1.2.3 libidn/
1.18 libssh2/1.4.2
> Host: 9.111.97.120:11080
> Accept: */*
>
< HTTP/1.1 302 Found
< X-Powered-By: Servlet/3.0
< Location: http://9.111.97.120:11080/console/dswebcustomauth/pages/login.jsp
< Content-Language: en-US
< Set-Cookie: dsweb11082=0000pbdwxNRCSvYh4Lrol3tbIQZ:306b357b-d04e-4b53-a11c-87a0312cbc35; Path=/; HttpO
nly
< Transfer-Encoding: chunked
< Date: Fri, 03 Mar 2017 11:18:34 GMT
< Expires: Thu, 01 Dec 1994 16:00:00 GMT
< Cache-Control: no-cache="set-cookie, set-cookie2"
<
* Connection #0 to host 9.111.97.120 left intact
* Closing connection #0
```

Figure 24. Access Data Server Manager via virtual IP

## Related information

For more information, see the following topics in IBM Knowledge Center:

- Documentation for Data Server Manager:  
<https://ibm.biz/Bdi6R2>
- IBM Tivoli System Automation for Multiplatforms:  
<https://ibm.biz/Bdi6RP>

# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

**© Copyright International Business Machines Corporation 2017. All rights reserved.**

This document was created or updated on July 11, 2017.

Send us your comments in one of the following ways:

- Use the online **Contact us** review form found at:  
[ibm.com/redbooks](http://ibm.com/redbooks)
- Send your comments in an e-mail to:  
[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)
- Mail your comments to:  
IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400 U.S.A.

This document is available online at <http://www.ibm.com/redbooks/abstracts/tips1352.html> .

## Trademarks

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

dashDB®	IBM®	Redbooks (logo)®
Db2®	pureScale®	Tivoli®
DB2®	Redbooks®	z/OS®

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.