HADR with TSA
using
db2haicu
(A DB2 v9.7 utility)
on AIX/Linux

Date : 04 Apr 2010
Version : 2.0
Author : Neeraj Sharma (neeraj.sharma@in.ibm.com)
Abhishek Iyer (abhishekiyer@in.ibm.com)

Abstract: This is a step by step guide for setting up end to end HADR with TSA using db2haicu utility.

Audience: The target audience for this document is customer.

Acronyms

Text in RED indicates user inputs.
Text in BLUE indicates questions prompted by system/utility.
Text in BLACK indicates information message by system/utility.
Above nomenclature does not apply to “1ssam” output in section 11, where-in GREEN indicates online node, BLUE indicates offline node (in warm mode) and RED indicates failed node.
Table of contents

Part 1: DB2 configuration
1. Check/Update /etc/hosts file ................................................................. 4
2. Check/Update /etc/services file ............................................................. 4
3. Enable HADR on databases ................................................................... 5
4. Take database backup ........................................................................... 5
5. Restore database backup on standby node ............................................ 6
6. Check database states ........................................................................... 6
7. Start HADR on databases ...................................................................... 6

Part 2: TSA Cluster setup
8. Prepare cluster nodes first ..................................................................... 7
9. On Standby node (should be executed on standby node first) .......... 7
10. Execute db2haicu on Primary node .................................................... 9
11. Check cluster status ............................................................................ 10

Part 3: Miscellaneous tasks / Diagnostics
12. Disable TSA cluster ............................................................................ 11
13. Enable TSA cluster ............................................................................. 11
14. db2haicu – maintenance mode ............................................................ 12
15. HADR automated failover resource listing using db2pd ................. 12
16. HADR status using db2pd ................................................................. 14

Part 4: Remove TSA/HADR configuration
17. Remove TSA cluster ........................................................................... 15
18. Stop HADR .......................................................................................... 15

Part 5: Automatic client reroute (ACR)
19. Add ACR configuration ...................................................................... 16
20. Remove ACR configuration .................................................................. 16

Appendix
Why database creation is required on both nodes? ............................... 17
Quorum device ............................................................................................. 17
Cluster node name resolution ..................................................................... 17
Cluster Public/Private networks ................................................................ 17
HADR States ............................................................................................... 18
Peer ............................................................................................................ 18
LocalCatchup ............................................................................................... 18
RemoteCatchupPending ............................................................................ 19
RemoteCatchup ......................................................................................... 19
Disconnected ............................................................................................. 19
Manual takeover ......................................................................................... 19
Definite NO NO for HADR setup ............................................................... 19
DB2_LOAD_COPY_NO_OVERRIDE ......................................................... 20
Application impact .................................................................................... 20
Upgrading DB2 ......................................................................................... 21
References ............................................................................................... 21
Part 1 : DB2 configuration

Overall architecture

Create database on both database servers [yes, creation on both servers is required, why its required is explained in appendix]

$ db2 create database hadrd

Configuration setting on both databases
Create database layout and other database objects on just one database which initially will act as primary node (say hadrnode01)

1. Check/Update /etc/hosts file
Make sure following entries exits on both the nodes

```
9.1.2.3     hadrnode01
9.1.2.4     hadrnode02
```

2. Check/Update /etc/services file
Make sure following entries exits on both the nodes

```
DB2_HADR_PORT: 55555/tcp
db2c_db2instl: 50000/tcp
DB2_db2instl: 60000/tcp
DB2_db2instl_END: 60003/tcp
```
3. Enable HADR on databases

hadrnode01

Log retain for recovery enabled
(LogREtain) = RECOVERY
Index re-creation time and redo index build (INDEXREC) = RESTART
Log pages during index build (LOGINDEXBUILD) = ON
HADR local host name (HADR_LOCAL_HOST) = hadrnode01
HADR local service name (HADR_LOCAL_SVC) = DB2_HADR_PORT
HADR remote host name (HADR_REMOTE_HOST) = hadrnode02
HADR remote service name (HADR_REMOTE_SVC) = DB2_HADR_PORT
HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1
HADR timeout value (HADR_TIMEOUT) = 30
HADR log write synchronization mode (HADR_SYNCMODE) = NEARSYNC
HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 60

hadrnode02

Log retain for recovery enabled
(LogREtain) = RECOVERY
Index re-creation time and redo index build (INDEXREC) = RESTART
Log pages during index build (LOGINDEXBUILD) = ON
HADR local host name (HADR_LOCAL_HOST) = hadrnode02
HADR local service name (HADR_LOCAL_SVC) = DB2_HADR_PORT
HADR remote host name (HADR_REMOTE_HOST) = hadrnode01
HADR remote service name (HADR_REMOTE_SVC) = DB2_HADR_PORT
HADR instance name of remote server (HADR_REMOTE_INST) = db2inst1
HADR timeout value (HADR_TIMEOUT) = 30
HADR log write synchronization mode (HADR_SYNCMODE) = NEARSYNC
HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 60

Values HADR_TIMEOUT and HADR_PEER_WINDOW should be tuned for a particular setup. Standard recommended values of these parameters are 120 and 300 respectively.

$ db2 update db cfg for hadrdb using LOGREtain RECOVERY

Once LOGREtain is set to RECOVERY, database will move to “backup pending” state.

hadrnode01

$ db2 get db cfg for hadrdb | grep -i ”Backup pending”
Backup pending = YES

hadrnode02

$ db2 get db cfg for hadrdb | grep -i ”Backup pending”
Backup pending = YES

4. Take database backup

hadrnode01

$ db2 Backup database hadrdb
Backup successful. The timestamp for this backup image is : 20081223202401

This will create a file with name HADRDB.0.db2.NODE0000.CATN0000.20081223202401.001 in current directory. Make this file available on hadrnode02 for restoration during step 5.

hadrnode02

$ db2 Backup database hadrdb to /dev/null
Backup successful. The timestamp for this backup image is : 20081223202501
The HADR DB CFG parameters mentioned in section 3 (4th to 11th) should be set after doing the Backup. The first 3 parameters should be set before Backup itself.

5. Restore database backup on standby node

```
$ db2 restore database hadrdb taken at 20081223202401
SQL2539W Warning! Restoring to an existing database that is the same as the
backup image database. The database files will be deleted.
Do you want to continue ? (y/n) y
DB20000I The RESTORE DATABASE command completed successfully.
```

The HADR DB CFG parameters mentioned in section 3 (4th to 11th) should be set after doing the Restore.

6. Check database states

```
$ db2 get db cfg for hadrdb | grep -i "Rollforward pending"
Rollforward pending = NO
```

```
$ db2 get db cfg for hadrdb | grep -i "Rollforward pending"
Rollforward pending = YES
```

7. Start HADR on databases

```
$ db2 start hadr on db hadrdb as standby
DB20000I The START HADR ON DATABASE command completed successfully.
```

```
$ db2 start hadr on db hadrdb as primary
DB20000I The START HADR ON DATABASE command completed successfully.
```
Part 2 : TSA Cluster setup
RSCT version 2.5.3.3 and TSA version 3.1.0.0 was used for this implementation

8. Prepare cluster nodes first

Execute following commands as root on both nodes, all other commands should be executed as instance owner (db2inst1 in current example).

$ prepnnode hadrnode01
$ prepnnode hadrnode02

9. On Standby node (should be executed on standby node first)

bash-3.00$ db2haicu
Welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).

You can find detailed diagnostic information in the DB2 server diagnostic log file called db2diag.1og. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called 'DB2 High Availability Instance Configuration Utility (db2haicu)' in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

db2haicu is collecting information on your current setup. This step may take some time as db2haicu will need to activate all databases for the instance to discover all paths ... when you use db2haicu to configure your clustered environment, you create cluster domains. For more information, see the topic 'Creating a cluster domain with db2haicu' in the DB2 Information Center. db2haicu is searching the current machine for an existing active cluster domain ...

db2haicu did not find a cluster domain on this machine. db2haicu will now query the system for information about cluster nodes to create a new cluster domain ...

db2haicu did not find a cluster domain on this machine. To continue configuring your clustered environment for high availability, you must create a cluster domain; otherwise, db2haicu will exit.

Create a domain and continue? [1]
1. Yes
2. No
1

Create a unique name for the new domain:

hadr_domain
Nodes must now be added to the new domain.

How many cluster nodes will the domain hadr_domain contain?
2

Enter the host name of a machine to add to the domain:

hadrnode01
Enter the host name of a machine to add to the domain:

hadrnode02

db2haicu can now create a new domain containing the 2 machines that you specified. If you choose not to create a domain now, db2haicu will exit.

Create the domain now? [1]
1. Yes
2. No
1

Creating domain hadr_domain in the cluster ...

Creating domain hadr_domain in the cluster was successful.

You can now configure a quorum device for the domain. For more information, see the topic "Quorum devices" in the DB2 Information Center. If you do not configure a quorum device for the domain, then a human operator will have to manually intervene if subsets of machines in the cluster lose connectivity.
Configure a quorum device for the domain called hadr_domain? [1]
1. Yes
2. No
1

The following is a list of supported quorum device types:
1. Network Quorum
Enter the number corresponding to the quorum device type to be used: [1]
1
Specify the network address of the quorum device:
9.1.2.3 landlords refer to appendix for details on Quorum device
Configuring quorum device for domain hadr_domain ...
Configuring quorum device for domain hadr_domain was successful.
The cluster manager found 4 network interface cards on the machines in the domain. You
can use db2haicu to create networks for these network interface cards. For more
information, see the topic ’Creating networks with db2haicu’ in the DB2 Information
Center.
Create networks for these network interface cards? [1]
1. Yes
2. No
1
Enter the name of the network for the network interface card: eth10 on cluster node:
hadnode01
1. Create a new public network for this network interface card.
2. Create a new private network for this network interface card.
Enter selection:
1 landlords refer to Appendix for details on this option selection (Cluster Public/Private
networks)
Are you sure you want to add the network interface card eth10 on cluster node hadrnode01
to the network db2_public_network_0? [1]
1. Yes
2. No
1
Entering network interface card eth10 on cluster node hadrnode01 to the network
db2_public_network_0 ...
Enter the name of the network for the network interface card: eth10 on cluster node:
hadnode02
1. db2_public_network_0
2. Create a new public network for this network interface card.
3. Create a new private network for this network interface card.
Enter selection:
1
Are you sure you want to add the network interface card eth10 on cluster node hadrnode02
to the network db2_public_network_0? [1]
1. Yes
2. No
1
Entering network interface card eth10 on cluster node hadrnode02 to the network
db2_public_network_0 ...
Enter the name of the network for the network interface card: eth3 on cluster node:
hadnode02
1. db2_public_network_0
2. Create a new public network for this network interface card.
3. Create a new private network for this network interface card.
Enter selection:
2
Are you sure you want to add the network interface card eth3 on cluster node hadrnode02
to the network db2_public_network_1? [1]
1. Yes
2. No
1
Entering network interface card eth3 on cluster node hadrnode02 to the network
db2_public_network_1 ...
Enter the name of the network for the network interface card: eth3 on cluster node:
hadnode01
1. db2_public_network_1
2. db2_public_network_0
3. Create a new public network for this network interface card.
4. Create a new private network for this network interface card.
Enter selection:
1
Are you sure you want to add the network interface card eth3 on cluster node hadrnode01 to the network db2_public_network_1? [1]
1. Yes
2. No
1
Adding network interface card eth3 on cluster node hadrnode01 to the network db2_public_network_1 ...
Adding network interface card eth3 on cluster node hadrnode01 to the network db2_public_network_1 was successful.
Retrieving high availability configuration parameter for instance db2inst1 ...
Retrieving high availability configuration parameter for instance db2inst1 was successful.
Adding DB2 database partition 0 to the cluster ...
Adding DB2 database partition 0 to the cluster was successful.
Retrieving high availability configuration parameter for instance db2inst1 ...
The cluster manager configuration parameter (high availability configuration parameter) is not set. For more information, see the topic "cluster_mgr - Cluster manager name configuration parameter" in the DB2 Information Center. Do you want to set the high availability configuration parameter?
The following are valid settings for the high availability configuration parameter:
1. TSA
2. Vendor
Enter a value for the high availability configuration parameter: [1]
1
Do you want to validate and automate HADR failover for the HADR database HADRDB? [1]
1. Yes
2. No
1
Adding HADR database HADRDB to the domain ...
The HADR database HADRDB has been determined to be valid for high availability. However, the database cannot be added to the cluster from this node because db2haicu detected this node is the standby for the HADR database HADRDB. Run db2haicu on the primary for the HADR database HADRDB to configure the database for automated failover.
All cluster configurations have been completed successfully. db2haicu exiting ...
bash-3.00$

10. Execute db2haicu on Primary node

bash-3.00$ db2haicu
Welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).
You can find detailed diagnostic information in the DB2 server diagnostic log file called db2diag.log. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called 'DB2 High Availability Instance Configuration Utility (db2haicu)' in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

db2haicu is collecting information on your current setup. This step may take some time as db2haicu will need to activate all databases for the instance to discover all paths ...
when you use db2haicu to configure your clustered environment, you create cluster domains. For more information, see the topic 'Creating a cluster domain with db2haicu' in the DB2 Information Center. db2haicu is searching the current machine for an existing active cluster domain ...
db2haicu found a cluster domain called hadr_domain on this machine. The cluster configuration that follows will apply to this domain.

Retrieving high availability configuration parameter for instance db2inst1 ...
The cluster manager name configuration parameter (high availability configuration parameter) is not set. For more information, see the topic "cluster_mgr - Cluster manager name configuration parameter" in the DB2 Information Center. Do you want to set the high availability configuration parameter?
The following are valid settings for the high availability configuration parameter:
1. TSA
2. Vendor
Enter a value for the high availability configuration parameter: [1]
1
Setting a high availability configuration parameter for instance db2inst1 to TSA. Adding DB2 database partition 0 to the cluster ...
Adding DB2 database partition 0 to the cluster was successful.
Do you want to validate and automate HADR failover for the HADR database HADRDB? [1]

1. Yes
2. No

1

Adding HADR database HADRDB to the domain ...

The cluster node 9.1.2.3 was not found in the domain. Please re-enter the host name: hadnode02

The cluster node 9.1.2.4 was not found in the domain. Please re-enter the host name: hadnode04

- refer to Appendix for more details on cluster node name resolution

Adding HADR database HADRDB to the domain was successful.

Do you want to configure a virtual IP address for the HADR database HADRDB? [1]

1. Yes
2. No

1

Enter the virtual IP address:

9.1.2.5

Enter the subnet mask for the virtual IP address 9.1.2.5: [255.255.255.0]

255.255.255.0

Select the network for the virtual IP 9.1.2.5:

1. db2_public_network_0
2. db2_public_network_1

Enter selection:

1

Adding virtual IP address 9.1.2.5 to the domain ...

Adding virtual IP address 9.1.2.5 to the domain was successful.

All cluster configurations have been completed successfully. db2haicu exiting ...

bash-3.00$

11. Check cluster status

On successful completion of commands, cluster domain and nodes status can be checked
by following commands [the command output should same across all nodes in cluster]

bash-3.00$ lsrpdomain
Name OpState RSCTActiveVersion MixedVersions TSPort GSPort
hadr_domain Online 2.5.3.3 No 12347 12348

bash-3.00$ lsrpnode
Name OpState RSCTVersion
hadrnode01 Online 2.5.3.3
hadrnode02 Online 2.5.3.3

bash-3.00$ lssam
Online IBM.ResourceGroup:db2_db2inst1_hadrnode01_0-rg Nominal=Online
  - Online IBM.Application:db2_db2inst1_hadrnode01_0-rs
  - Online IBM.Application:db2_db2inst1_hadrnode01_0-rs:hadrnode01
Online IBM.ResourceGroup:db2_db2inst1_hadrnode02_0-rg Nominal=Online
  - Online IBM.Application:db2_db2inst1_hadrnode02_0-rs
  - Online IBM.Application:db2_db2inst1_hadrnode02_0-rs:hadrnode02
Online IBM.ResourceGroup:db2_db2inst1_hadrnode02_1-rg Nominal=Online
  - Online IBM.Application:db2_db2inst1_hadrnode02_1-rs
  - Online IBM.Application:db2_db2inst1_hadrnode02_1-rs:hadrnode02
  - Online IBM.ServiceIP:db2ip_9_1_2_5-rs
  - Online IBM.ServiceIP:db2ip_9_1_2_5-rs:hadrnode01
  - Offline IBM.ServiceIP:db2ip_9_1_2_5-rs:hadrnode02

Now the HADR setup is all set for automatic failover in case of disaster. TSA cluster
starts automatically with absolutely no manual intervention.
Part 3 : Miscellaneous tasks / Diagnostics

12. Disable TSA cluster

This option is for scheduled maintenance of setup. It only stops TSA, HADR will remain active (but failover will not happen). Execute this on primary node as instance owner.

bash-3.00$ db2haicu -disable
Welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).

You can find detailed diagnostic information in the DB2 server diagnostic log file called db2diag.log. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called "DB2 High Availability Instance Configuration Utility (db2haicu)" in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

db2haicu is collecting information on your current setup. This step may take some time as db2haicu will need to activate all databases for the instance to discover all paths ... Are you sure you want to disable high availability (HA) for the database instance db2inst1. This will lock all the resource groups for the instance and disable the HA configuration parameter. The instance will not failover if a system outage occurs while the instance is disabled. You will need to run db2haicu again to enable the instance for HA.

Disabling HA for the instance db2inst1? [1]
1. Yes
2. No
1. Yes
Disabling high availability for instance db2inst1 ...
Locking the resource group for HADR database HADRDB ...
Locking the resource group for HADR database HADRDB was successful.
Locking the resource group for DB2 database partition 0 ...
Locking the resource group for DB2 database partition 0 was successful.
Locking the resource group for DB2 database partition 0 ...
Locking the resource group for DB2 database partition 0 was successful.
Disabling high availability for instance db2inst1 was successful.
All cluster configurations have been completed successfully. db2haicu exiting ...

13. Enable TSA cluster

Launch this command from primary node as instance owner.

bash-3.00$ db2haicu
Welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).

You can find detailed diagnostic information in the DB2 server diagnostic log file called db2diag.log. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called "DB2 High Availability Instance Configuration Utility (db2haicu)" in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

db2haicu is collecting information on your current setup. This step may take some time as db2haicu will need to activate all databases for the instance to discover all paths ... When you use db2haicu to configure your clustered environment, you create cluster domains. For more information, see the topic 'Creating a Cluster Domain with db2haicu' in the DB2 Information Center. db2haicu is searching the current machine for an existing active cluster domain ...
db2haicu found a cluster domain called hadr_domain on this machine. The cluster configuration that follows will apply to this domain.

db2haicu has detected that high availability has been disabled for the instance db2inst1. Do you want to enable high availability for the instance db2inst1? [1]
1. Yes
2. No
1
Retrieving high availability configuration parameter for instance db2inst1 ...
The cluster manager name configuration parameter (high availability configuration parameter) is not set. For more information, see the topic "cluster_mgr - Cluster manager name configuration parameter" in the DB2 Information Center. Do you want to set the high availability configuration parameter?
The following are valid settings for the high availability configuration parameter:
1. TSA
2. Vendor
Enter a value for the high availability configuration parameter: [1]
1
Setting a high availability configuration parameter for instance db2inst1 to TSA.
Enabling high availability for instance db2inst1 ...
Enabling high availability for instance db2inst1 was successful.
All cluster configurations have been completed successfully. db2haicu exiting ...

14. db2haicu – maintenance mode

db2haicu will have following output if its already active. This mode is called maintenance mode. You can add/remove nodes, databases, virtual IPs, network interfaces and quorum device. This command should be executed from primary database node.

bash-3.00$ db2haicu
welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).
You can find detailed diagnostic information in the DB2 server diagnostic log file called db2diag.log. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called 'DB2 High Availability Instance Configuration Utility (db2haicu)' in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

db2haicu is collecting information on your current setup. This step may take some time as db2haicu will need to activate all databases for the instance to discover all paths ...
when you use db2haicu to configure your clustered environment, you create cluster domains. For more information, see the topic 'Creating a cluster domain with db2haicu' in the DB2 Information Center. db2haicu is searching the current machine for an existing active cluster domain ...
db2haicu found a cluster domain called hadr_domain on this machine. The cluster configuration that follows will apply to this domain.

Select an administrative task by number from the list below:
1. Add or remove cluster nodes.
2. Add or remove a network interface.
3. Add or remove HADR databases.
4. Add or remove an IP address.
5. Move DB2 database partitions and HADR databases for scheduled maintenance.
6. Create a new quorum device for the domain.
7. Destroy the domain.
8. Exit.
Enter your selection:

15. HADR automated failover resource listing using db2pd

This command can be issued from any of the database nodes.

bash-3.00$ db2pd -ha
DB2 HA Status
Instance Information:
Instance Name = db2inst1
Number Of Domains = 1
Number Of RGS for instance = 2

Domain Information:
Domain Name = hadr_domain
Cluster Version = 2.5.3.3
Cluster State = Online
Number of nodes = 2

Node Information:
<table>
<thead>
<tr>
<th>Node Name</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadrnnode01</td>
<td>Online</td>
</tr>
<tr>
<td>hadrnnode02</td>
<td>Online</td>
</tr>
</tbody>
</table>

Resource Group Information:
<table>
<thead>
<tr>
<th>Resource Group Name</th>
<th>Resource Group LockState</th>
<th>Resource Group OpState</th>
<th>Resource Group Nominal OpState</th>
<th>Number of Group Resources</th>
<th>Number of Allowed Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_db2inst1_db2inst1_HADRDB-rg</td>
<td>Locked</td>
<td>Online</td>
<td>Online</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Member Resource Information:
<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Resource State</th>
<th>Resource Type</th>
<th>HADR Primary Instance</th>
<th>HADR Secondary Instance</th>
<th>HADR DB Name</th>
<th>HADR Primary Node</th>
<th>HADR Secondary Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_db2inst1_db2inst1_HADRDB-rg</td>
<td>Online</td>
<td>HADR</td>
<td>db2inst1</td>
<td>db2inst1</td>
<td>HADRDB</td>
<td>hadrnnode02</td>
<td>hadrnnode01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Resource State</th>
<th>Resource Type</th>
<th>DB2 Partition Number</th>
<th>Number of Allowed Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_db2inst1_hadrnode02_0-rg</td>
<td>Online</td>
<td>IP</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Member Resource Information:
<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Resource State</th>
<th>Resource Type</th>
<th>DB2 Partition</th>
<th>Number of Allowed Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_db2inst1_hadrnode02_0-rg</td>
<td>Online</td>
<td>DB2</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Network Information:
<table>
<thead>
<tr>
<th>Network Name</th>
<th>Number of Adapters</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_public_network_1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Adapter Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadrnnode02</td>
<td>eth3</td>
</tr>
<tr>
<td>hadrnnode01</td>
<td>eth3</td>
</tr>
</tbody>
</table>

Network Name | Number of Adapters |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_public_network_0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Adapter Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadrnnode01</td>
<td>eth10</td>
</tr>
<tr>
<td>hadrnnode02</td>
<td>eth10</td>
</tr>
</tbody>
</table>

Quorum Information:
<table>
<thead>
<tr>
<th>Quorum Name</th>
<th>Quorum State</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2_Quorum_Network_9_1_2_1:18_3_46</td>
<td>Online</td>
</tr>
<tr>
<td>Operator</td>
<td>Offline</td>
</tr>
<tr>
<td>Fail</td>
<td>Offline</td>
</tr>
</tbody>
</table>
16. **HADR status using db2pd**

This command can be issued from any of the database nodes.

bash-3.00$ db2pd -hadr -db hadrdb

Database Partition 0 -- Database HADRDB -- Active -- Up 0 days 03:33:33

HADR Information:

<table>
<thead>
<tr>
<th>Role</th>
<th>State</th>
<th>SyncMode</th>
<th>HeartBeatsMissed</th>
<th>LogGapRunAvg (bytes)</th>
<th>Nearsync</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Peer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectStatus</th>
<th>ConnectTime</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected</td>
<td>Mon Dec 22 20:15:14 2008 (1229955314)</td>
<td>30</td>
</tr>
<tr>
<td>PeerwindowEnd</td>
<td>Mon Dec 22 23:36:21 2008 (1229967381)</td>
<td>60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LocalHost</th>
<th>LocalService</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadrnode01</td>
<td>DB2_HADR_PORT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RemoteHost</th>
<th>RemoteService</th>
<th>RemoteInstance</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadrnode02</td>
<td>DB2_HADR_PORT</td>
<td>db2inst1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PrimaryFile</th>
<th>PrimaryPg</th>
<th>PrimaryLSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>S0037993.LOG</td>
<td>1406</td>
<td>0x000001BEE25065FD</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>StandByFile</th>
<th>StandByPg</th>
<th>StandByLSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>S0037993.LOG</td>
<td>1406</td>
<td>0x000001BEE25065FD</td>
</tr>
</tbody>
</table>

db2pd-hadr.out: END
Part 4 : Remove TSA/HADR configuration

In order to remove TSA configuration settings (cluster and resource groups) completely from the cluster,

17. Remove TSA cluster

bash-3.00$ db2haicu -delete
Welcome to the DB2 High Availability Instance Configuration Utility (db2haicu).

You can find detailed diagnostic information in the $DB2 server diagnostic log file called db2diag.log. Also, you can use the utility called db2pd to query the status of the cluster domains you create.

For more information about configuring your clustered environment using db2haicu, see the topic called 'DB2 High Availability Instance Configuration Utility (db2haicu)' in the DB2 Information Center.

db2haicu determined the current DB2 database manager instance is db2inst1. The cluster configuration that follows will apply to this instance.

When you use db2haicu to configure your clustered environment, you create cluster domains. For more information, see the topic 'Creating a cluster domain with db2haicu' in the DB2 Information Center. db2haicu is searching the current machine for an existing active cluster domain ...

db2haicu found a cluster domain called hadr_domain on this machine. The cluster configuration that follows will apply to this domain.

Removing HADR database HADRDB from the domain ...
Removing HADR database HADRDB from the domain was successful.
Removing DB2 database partition 0 from the cluster ...
Removing DB2 database partition 0 from the cluster was successful.
Deleting the domain hadr_domain from the cluster ...
Deleting the domain hadr_domain from the cluster was successful.
All cluster configurations have been completed successfully. db2haicu exiting ...

18. Stop HADR

hadrnode02
$ db2 stop hadr on db hadrdb
DB20000I The STOP HADR ON DATABASE command completed successfully.

hadrnode01
$ db2 deactivate database hadrdb
DB20000I The DEACTIVATE DATABASE command completed successfully.
$ db2 stop hadr on db hadrdb
DB20000I The STOP HADR ON DATABASE command completed successfully.
Part 5: Automatic client reroute (ACR)

Automatic client reroute is a DB2 Database for Linux, UNIX, and Windows feature that allows client applications to recover from a loss of communication with the server so that the application can continue its work with minimal interruption. Automatic client reroute can be accomplished only if an alternate server has been specified prior to the loss of communication. Client should at least connect to server once in order to be able to use ACR feature. No configuration changes on the client side are required.

19. Add ACR configuration

```
hadrnode01
$ db2 update alternate server for database hadrdb using hostname hadrnode02 port 50000
DB200000I The UPDATE ALTERNATE SERVER FOR DATABASE command completed successfully.
DB21056W Directory changes may not be effective until the directory cache is refreshed.

hadrnode02
$ db2 update alternate server for database hadrdb using hostname hadrnode01 port 50000
DB200000I The UPDATE ALTERNATE SERVER FOR DATABASE command completed successfully.
DB21056W Directory changes may not be effective until the directory cache is refreshed.
```

20. Remove ACR configuration

```
hadrnode01
$ db2 update alternate server for database hadrdb using hostname NULL port NULL
DB200000I The UPDATE ALTERNATE SERVER FOR DATABASE command completed successfully.
DB21056W Directory changes may not be effective until the directory cache is refreshed.

hadrnode02
$ db2 update alternate server for database hadrdb using hostname NULL port NULL
DB200000I The UPDATE ALTERNATE SERVER FOR DATABASE command completed successfully.
DB21056W Directory changes may not be effective until the directory cache is refreshed.
```
Appendix

*Why database creation is required on both nodes?*

Database creation is required on both db nodes so that LOGRETAIN can be set to recovery successfully. If backup from hadrnode01 is restored on hadrnode01 before enabling LOGRETAIN, database will have “Backup pending” and “Rollforward pending” flags “ON”. In this state database can not be backed up because its in Rollforward pending state and HADR standby can not be started on this database because database is in “Backup pending” state.

*Quorum device*

A *quorum device* helps a cluster manager make cluster management decisions when the cluster manager's normal decision process does not produce a clear choice. When a cluster manager has to choose between multiple potential actions, the cluster manager counts how many cluster domain nodes support each of the potential actions; and then cluster manager chooses the action that is supported by the majority of cluster domain nodes. If exactly the same number of cluster domain nodes supports more than one choice, then the cluster manager refers to a quorum device to make the choice.

A network quorum device is an IP address to which every cluster domain node can connect (ping) at all times.

In current implementation network route IP is used assuming that as long as network segment is UP and RUNNING, router will always be available. No special software is required to be installed in this quorum device. It should be reachable (ping-able) to both the nodes all the time. During the disaster/node crash event, this device should be reachable to other node which is working fine (either as primary/standby).

*Cluster node name resolution*

This resolution is required only when IP address are using during HADR configuration instead of hostname. Automatic name resolution will not happen using /etc/hosts file. Right hostname should be provided at this prompt. Names provided at this prompt should match the ones provided to db2haicu while configuration in step 9. This prompt will not appear if HADR configuration uses hostname instead of IP addresses.

*Cluster Public/Private networks*

Typically servers will have multiple network interfaces. In current implementation we have 2 network interfaces on each node [eth3 and eth10]. Public network on both nodes are using eth3 network adapter and private network uses eth10 network adapter. Network
interfaces with compatible IP address should be grouped together. In current implementation eth3 on both nodes are in `db2_public_network_1` and eth10 are in `db2_public_network_0`.

**HADR States**

![HADR States Diagram]

**Peer**
This state indicates that primary and standby nodes are in sync and standby is ready to takeover as primary database.

**LocalCatchup**
This state indicates that database has just started and is reading local log files from log paths, archive paths. Immediately after this state, node will enter into RemoteCatchupPending state.


**RemoteCatchupPending**
This state indicates standby node is waiting for Primary to come up in order to receive any log updates. Node can enter into this state if during a RemoteCatchup connection between Primary and standby breaks.

**RemoteCatchup**
This state indicates standby is applying the log gap between Primary and standby onto standby node.

**Disconnected**
This state indicates standby node is not available

---

**Manual takeover**
To make standby node as primary manually, issue following command at standby node

```
$ db2 takeover hadr on db hadrdb
DB20000I The TAKEOVER HADR ON DATABASE command completed successfully.
```

If for some reason, automatic failover fails, issue following command to force manual takeover.

```
$ db2 takeover hadr on db hadrdb by force
DB20000I The TAKEOVER HADR ON DATABASE command completed successfully.
```

---

**Definite NO NO for HADR setup**

Never use “nonrecoverable” load on HADR setup. It will stop log shipping to standby and the only option left to make HADR work again is to backup Primary (full backup) node and restore it onto standby node. This might not be an option at all if database size is of the order of terabytes; because backup and restore will take several hours to complete.
**DB2_LOAD_COPY_NO_OVERRIDE**

HADR replicates load with certain restrictions. Because load data is not embedded in the log stream, the standby can only get the data from a load copy. Thus load is replicated only if there is a load copy. The copy must be available to the standby when it replays the load. The standby may attempt to access the copy any time after the primary completes the load. It is recommended that a shared copy device such as NFS is used. If you transfer the copy by means like physically transferring a tape, it is recommended that the standby be stopped ("db2 deactivated database") before the primary starts the load. Once the primary finishes the load and the copy is available to the standby, restart the standby ("db2 activate database"). The standby will then reconnect to the primary and replay the load.

When the primary does a load with COPY NO option, by default the load is automatically converted to NONRECOVERABLE. When standby replays the load, the table is marked as invalid. COPY NO loads can be optionally converted to COPY YES via the DB2_LOAD_COPY_NO_OVERRIDE registry variable.

**Application impact**

In the even of failover from hadrnode01 to hadrnode02 (either due to disaster or manual takeover), application will receive following message once after successful takeover.

```
SQL30108N A connection failed but has been re-established. The hostname or IP address is "hadrnode02" and the service name or port number is "50003". Special registers may or may not be re-attempted (Reason code = "1"). SQLSTATE=08506
```

In order to make this failover look transparent to users, application need to catch and handle above SQL code and retry internally or request user to try after few minutes. Here is one sample code

```java
try {
    // transaction code
} catch (SQLException sqlEx) {
    if(sqlEx.getCannotProceed() == false) {
        // execute the transaction again.
    }else{
        throw sqlEx; // release unhandled exceptions for other handlers
    }
}
```
**Upgrading DB2**

In case of Rolling Upgrade (DB2 patch application), both the nodes should be brought on to the same DB2 patch level as soon as possible. In other words, it is not possible to apply patch on one node and then after few days apply the patch onto the second node. As soon as primary node will know that standby is on a different db2 level, HADR state will become “Disconnected”.

### Rolling Upgrade for DB2 fix pack, OS

1. **Both P and S on fpN**
   - Both P and S on fpN
2. **Shutdown S. Upgrade S, Restart S, S on fpN+1**
   - Shutdown S. Upgrade S, Restart S, S on fpN+1
3. **Connection is closed at end of takeover**
   - Takeover. P on fpN+1
4. **Shutdown S. Upgrade S, Restart S, Both on fpN+1**
   - Optional 2nd takeover
   - Both on fpN+1

### References