

# Configuring DB2 pureScale for disaster recovery using DS8700 Metro Mirror

Skill Level: Introductory

[Aslam Nomani \(aslam@ca.ibm.com\)](mailto:aslam@ca.ibm.com)  
Quality Assurance Manager  
IBM

[Madhusudan K J \(madhusuk@in.ibm.com\)](mailto:madhusuk@in.ibm.com)  
DB2 Software Development  
IBM

[Santhees Kodi \(satheesk@us.ibm.com\)](mailto:satheesk@us.ibm.com)  
IT Specialist  
IBM

27 May 2010

The DB2® pureScale Feature builds on familiar and proven design features from the IBM DB2 for z/OS® database software. This article describes how to deploy the DB2 pureScale Feature in a disaster recovery configuration. This configuration leverages the IBM DS8700 Metro Mirror functionality to replicate changes made on the source site to the target site.

## Introduction

In today's highly competitive marketplace, it is important to deploy a data processing architecture that not only meets your immediate tactical needs but that also provides the flexibility to grow and change to adapt to your future strategic requirements. In December 2009, IBM introduced the DB2 pureScale Feature for Enterprise Server Edition (DB2 pureScale Feature). The DB2 pureScale Feature leverages an active-active shared-disk database implementation based on the DB2 for z/OS data-sharing architecture. It leverages proven technology from the DB2 database software on the mainframe to bring the active-active shared-disk technology to open systems.

The DB2 pureScale Feature meets the needs of many customers by providing the following key benefits:

### **Virtually unlimited capacity**

The DB2 pureScale Feature provides practically unlimited capacity by enabling the addition and removal of members on demand. The DB2 pureScale Feature can scale to 128 members. It has a highly efficient, centralized management facility that allows for very efficient scale out capabilities. The DB2 pureScale Feature also leverages a technology called Remote Direct Memory Access (RDMA) that provides a highly efficient inter-node communication mechanism to augment its scaling capabilities

### **Application transparency**

An application that runs in a DB2 pureScale environment does not need to have any knowledge of the different members in the cluster or of partitioning data. The DB2 pureScale Feature automatically routes applications to the members deemed most appropriate. The DB2 pureScale Feature also provides native support for syntax that other database vendors use, which enables those applications to run in a DB2 pureScale environment with minimal or no changes.

### **Continuous availability**

The DB2 pureScale Feature provides a fully active-active configuration such that if one member goes down, processing continues at the remaining active members. During a failure, only data being modified on the failing member is temporarily unavailable until database recovery completes for that set of data, which is very quick. This is an advantage over some competing solutions in which an entire system freeze occurs as part of the database recovery process.

### **Reduced total cost of ownership**

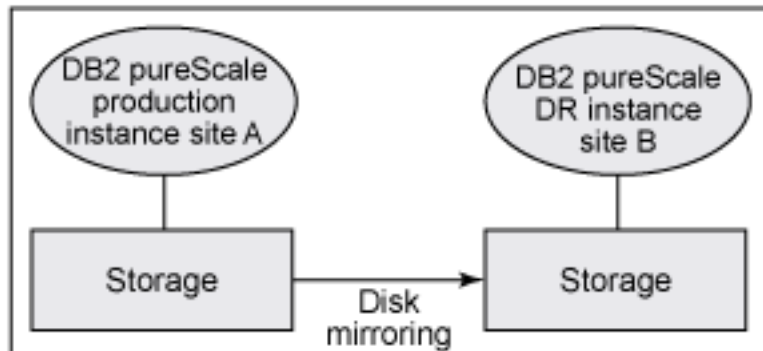
The DB2 pureScale interfaces easily handle the deployment and maintenance of components integrated within the DB2 pureScale Feature. This helps reduce what might amount to steep learning curves associated with deploying and maintaining with some competing technologies.

The DB2 pureScale Feature provides a local, high-availability solution, but many customers also require a disaster recovery solution to meet their business continuity requirements. The DB2 pureScale Feature leverages remote disk-mirroring technology, and it is designed to work with database replication products. If the entire primary site that is running a DB2 pureScale instance fails, the remote site can be leveraged to enable business operations to continue. The DB2 pureScale Feature can also leverage the traditional database backup, restore, and roll-forward functionality to enable disaster recovery solutions.

This article describes the steps required to deploy a Metro Mirror disk-mirroring solution with the IBM DS8700 storage subsystem. Disk mirroring can also be

deployed using disk-mirroring technology from other storage vendors. Figure 1 shows a typical configuration with storage mirrored to a DB2 pureScale disaster recovery instance.

**Figure 1. Disk mirroring**



The IBM System Storage® DS8700 is currently the most advanced model in the IBM DS8000® lineup. The DS8700 is designed to support the most demanding business applications with its superior data throughput and a design for continuous data availability. DS8700 can meet the needs of the most critical business applications with its scalability, new storage tier optimization, and broad server support. See [Resources](#) for more details about the DS8000 family of products.

## Understanding the solution configuration

In the example disaster recovery solution presented in this article, a DB2 pureScale instance is active at the primary site. Transactions are running against the database at the primary site, while Metro Mirror synchronously mirrors the DB2 pureScale Feature data and logs to the target site. If a complete failure occurs on the primary site, the DB2 pureScale instance at the target site can be brought online to handle application requests while the primary site is unavailable. When a site failure occurs on the primary site, a failover is invoked so that transactions are run against the database at the target site.

Figure 2 shows a basic topology of the configuration used in this example disaster recovery configuration.

**Figure 2. Topology overview**

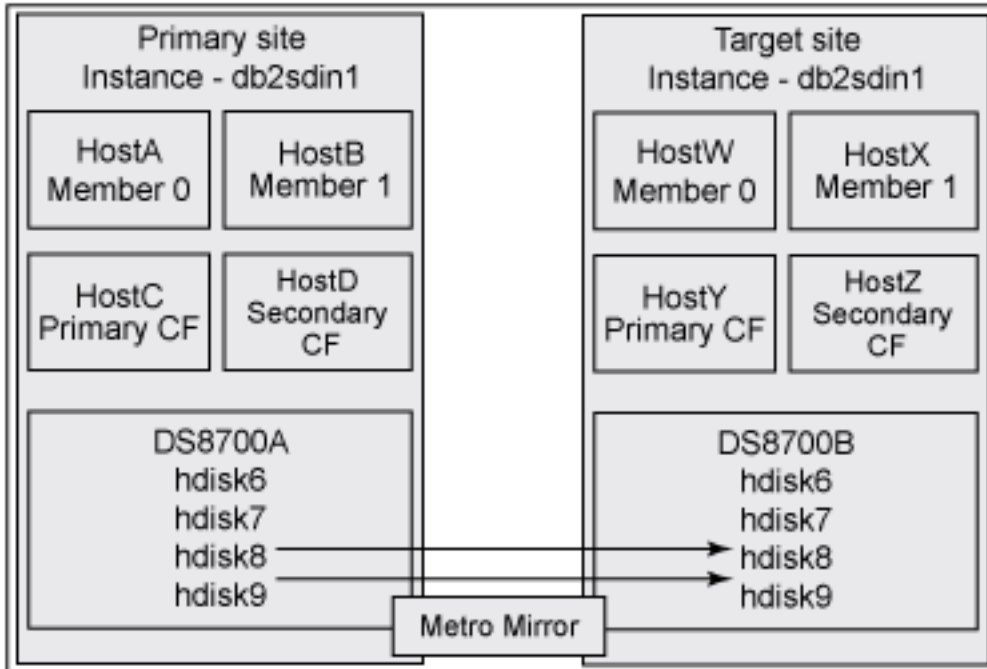


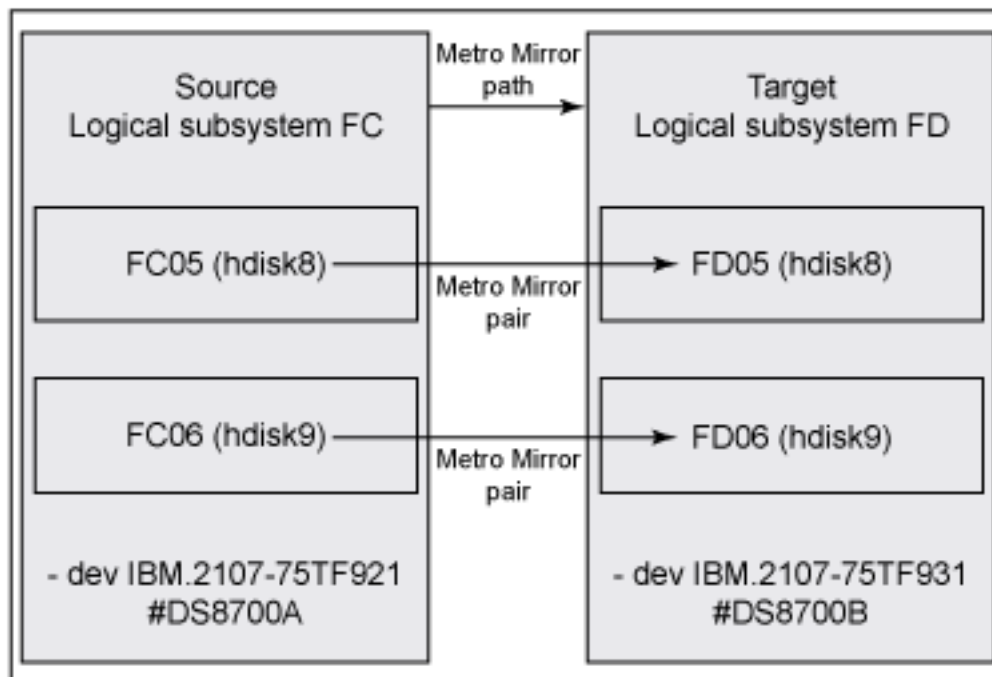
Figure 2 shows the following:

- An identical DB2 instance called db2sdin1 is created at both the primary site and the target site
- The DB2 instance at each site consists of two members and two PowerHA pureScale servers (CFs)
- A DS8700 storage subsystem is deployed at each site
- hdisk6 stores instance files in the home directory of the DB2 instance owner. This disk is not mirrored across sites.
- hdisk7 is leveraged by DB2 pureScale cluster services. This disk is not mirrored across sites.
- hdisk8 stores the data associated with a DB2 database. The file system created on this disk is called /db2fs/db2datafs. This disk is mirrored across sites.
- hdisk9 stores the transaction logs associated with a DB2 database. The file system created on this disk is called /db2fs/db2logfs. This disk is mirrored across sites.
- A storage-level consistency group is defined across both hdisk8 and hdisk9 to guarantee that a single-point-in-time copy of the data is seen at the target site upon any failure.

Figure 3 identifies more specifics of the two storage subsystems used in the

example scenario.

**Figure 3. Storage subsystem topology**



## Deploying the DB2 pureScale Feature with Metro Mirror

The following steps explain how to deploy the Metro Mirror functionality with the DB2 pureScale Feature so that hdisk8 and hdisk9 are being mirrored from the primary site to the target site.

1. From the DS8700 command line interface (dscli), determine the path status between logical subsystem FC and logical subsystem FD using the `lspprcpath` command, as shown in Listing 1.

### Listing 1. Determine the path status

```
DS8700A dscli> lspprcpath FC
DS8700A dscli> mkpprcpath -remotedev IBM.2107-75TF931
-remotewwn 5005076309FFC676 -srclss FC -tgtlss FD
-consistgrp IO302:IO302
```

Note that the consistency group option is used in the `mkpprcpath` command to ensure a consistent point-in-time copy across all disks that are being mirrored. The output provides the input/output (IO) port and worldwide node name (WWNN) to make a path between the two logical subsystems.

2. Create the Metro Mirror pairs, as shown in Listing 2.

### Listing 2. Creating Metro Mirror pairs

```
DS8700A dscli> mkpprc -remotedev IBM.2107-75TF931 -type mmir FC05-FC06:FD05-FD06
```

3. Check the status of the Metro Mirror pairs, as shown in Listing 3.

### Listing 3. Checking Metro Mirror pairs

```
DS8700A dscli> lsprrc -remotedev IBM.2107-75TF931 FC05-FC06:FD05-FD06
```

4. Create the db2sdin1 instance on the primary site with two DB2 members and two PowerHA pureScale servers. Listing 4 shows how to do this using the command line, but you can complete the same task from the DB2 graphical installation.

### Listing 4. Creating the db2sdin1 instance on the primary site

```
# db2icrt -d instance_shared_dev /dev/hdisk6 -tbdev  
/dev/hdisk7 -m hosta:hosta-ib0 -cf hostc:hostc-ib0 - u  
db2sdin1 db2sdin1  
# db2iupdt -d -add -m hostb:hostb-ib0 db2sdin1  
# db2iupdt -d -add -cf hostd:hostd-ib0 db2sdin1
```

5. Create the instance db2sdin1 on the target site with two DB2 members and two PowerHA pureScale servers, as shown in Listing 5.

### Listing 5. Creating the db2sdin1 instance on the target site

```
# db2icrt -d instance_shared_dev /dev/hdisk6  
-tbdev /dev/hdisk7 -m hostw:hostw-ib0 -cf hosty:hosty-ib0 - u  
db2sdin1 db2sdin1  
# db2iupdt -d -add -m hostx:hostx-ib0 db2sdin1  
# db2iupdt -d -add -cf hostz:hostz-ib0 db2sdin1
```

Note that any DB2 configuration parameter changes made at the primary site must be manually applied to the target site. Also, any required files that do not reside on the mirrored disks, including binary files from the creation of stored procedures, must be manually copied to the target site.

6. Create the file system for the DB2 data and transaction logs at the primary site, and then change the owner to the DB2 instance owner, as shown in Listing 6.

### Listing 6. Creating file system at the primary site

```
# db2cluster -cfs -create -filesystem db2datafs -disk /dev/hdisk8
# db2cluster -cfs -create -filesystem db2logfs -disk /dev/hdisk9
# chown db2sdin1:db2adm /db2fs/db2datafs
# chown db2sdin1:db2adm /db2fs/db2logfs
```

7. Configure passwordless rsh for root for computers between the primary site and the target site.
8. Synchronize the file system definition from the primary site to the target site by running the code in Listing 7 from HostA on the primary site. This synchronization only needs to occur when the definition of the file system changes.

### Listing 7. Synchronizing the file system definition

```
> db2stop force
# /usr/lpp/mmfs/bin/mmshutdown -a
```

9. Create a file called remotenodefile with the entries in Listing 8.

### Listing 8. Creating remotenodefile

```
hostw
hostx
hosty
hostz
```

10. Run the commands in Listing 9 to export the file system definitions from the primary site to the target site.

### Listing 9. Exporting the file system definitions

```
# /usr/lpp/mmfs/bin/mmfsctl db2datafs syncFSconfig -n remotenodefile
# /usr/lpp/mmfs/bin/mmfsctl db2logfs syncFSconfig -n remotenodefile
```

11. Validate the propagation of the file system definitions to the target site by running the commands in Listing 10 from HostW. The `mmlsnsd` command shows that the file system definitions for the file systems `db2datafs` and `db2logfs` now exist on the target site. Once the validation is complete, stop the file system.

#### Listing 10. Validating the propagation of the file system definitions

```
# /usr/lpp/mmfs/bin/mmstartup -a
# /usr/lpp/mmfs/bin/mmlsnsd
# /usr/lpp/mmfs/bin/mmshutdown -a
```

On the primary site, an application workload can now run against the database, and the Metro Mirror function automatically mirrors changes at the disk level to the target site.

12. Upon failure of the primary site, perform the disk failover as shown in Listing 11 such that the data on the target site can be used to continue database processing at the target site.

#### Listing 11. Performing the disk failover

```
DS8700B dscli> failoverpprc -remotedev IBM.2107-75TF921
-type mmir FD05-FD06:FC05-FC06
```

13. Start the file systems at the target site, as shown in Listing 12.

#### Listing 12. Starting the file systems at the target site

```
# /usr/lpp/mmfs/bin/mmstartup -a
```

14. The first time the database is accessed at the target site, catalog the database by user `db2sdin1` so that the instance on the target site can see it, as shown in Listing 13.

#### Listing 13. Cataloging the database

```
> db2 catalog db sample on /db2fs/db2datafs
```



15. Start DB2 on the target site with the db2sdin1 user ID, as shown in Listing 14.

#### Listing 14. Starting DB2 on the target site

```
> db2start
```

The database at the target site is now fully accessible and contains all committed updates that occurred on the primary site.

When the primary site comes back online, the disk-mirroring direction must be reversed for the primary site to once again become the database used for processing.

16. Reverse the disk-mirroring direction such that the disks on the target site are now being mirrored to disks on the primary site, as shown in Listing 15.

#### Listing 15. Reversing the disk-mirroring direction

```
DS8700B dscli> mkpprcpath -remotedev IBM.2107-75TF921 -remotewwnn  
5005076309FFC61A -srclss FD -tgtlss FC -consistgrp IO302:IO302  
DS8700B dscli> failbackpprc -remotedev IBM.2107-75TF921 -type mmir  
FD05-FD06:FC05-FC06
```

17. Validate that the primary site is back online by using the lspprc command.
18. When you want to reintegrate the original primary site as the location for data processing, stop the db2 instance and the file system from HostW at the target site, as shown in Listing 16.

#### Listing 16. Stopping the db2 instance and the file system

```
> db2stop force  
# /usr/lpp/mmfs/bin/mmshutdown -a
```

19. Ensure all data is synchronized between the disk volume pairs and are in a full-duplex state, as shown in Listing 17.

#### Listing 17. Checking for synchronized data

```
DS8700A dscli> lspprc -remotedev IBM.2107-75TF931 FC05-FC06:FD05-FD06
```

20. Perform the disk failover such that database processing can continue at the primary site, as shown in Listing 18.

### Listing 18. Performing disk failover to the primary site

```
DS8700A dscli> failoverpprc -remotedev IBM.2107-75TF931  
-type mmir FC05-FC06:FD05-FD06
```

21. Start the file system and the DB2 instance at the primary site from HostA, as shown in Listing 19.

### Listing 19. Starting the file system and instance

```
# /usr/lpp/mmfs/bin/mmstartup -a  
> db2start
```

22. Initiate disk failback such that the disks are now being mirrored from the primary site to the target site, in case any future primary site failures occur, as shown in Listing 20.

### Listing 20. Initiate disk failback

```
DS8700A dscli> mkpprcpath -remotedev IBM.2107-75TF931  
-remotewwnn 5005076309FFC676 -srclss FC -tgtlss FD -consistgrp IO302:IO302  
  
DS8700A dscli> failbackpprc -remotedev IBM.2107-75TF931  
-type mmir FC05-FC06:FD05-FD06
```

Tip: You can leverage the DB2 automatic client-reroute feature to automatically reconnect the application across sites once a site failure or reintegration has occurred.

## Conclusion

The DB2 pureScale Feature for Enterprise Server Edition provides a database solution that meets the needs of the most demanding users. It is designed for high availability to allow for business continuity in the event of planned and unplanned outages. You can deploy a disaster recovery solution to allow customers to continue to run their businesses even in the unfortunate circumstance of a full site failure. The DB2 pureScale Feature combined with a solution like the DS8700 Metro Mirror

described in this article provide a resilient solution designed to meet the high-availability and disaster-recovery requirements of the most demanding customer environments.

# Resources

## Learn

- Refer to "[Deploying the DB2 pureScale Feature](#)" (developerWorks, May 2010) to follow an example scenario based on a configuration using two p6 550s servers.
- Check out "[What is pureScale?](#)" (Data Management Magazine, Jan 2010) for the basics of DB2 pureScale from a technology perspective, showing how DB2 pureScale delivers both transparent application scalability and extreme availability.
- Find more details about the DS8000 family of products at [IBM System Storage DS8000 Turbo](#).
- Learn more about Information Management at the [developerWorks Information Management zone](#). Find technical documentation, how-to articles, education, downloads, product information, and more.
- Stay current with [developerWorks technical events and webcasts](#).

## Get products and technologies

- Build your next development project with [IBM trial software](#), available for download directly from developerWorks.
- Download a [free trial version of DB2 9.7 for Linux, UNIX, and Windows](#).
- Download [DB2 Express-C 9.7](#), a no-charge version of DB2 Express database server for the community.

## Discuss

- [Participate in the discussion forum for this content](#).
- Check out the [developerWorks blogs](#) and get involved in the [developerWorks community](#).

## About the authors

Aslam Nomani

Aslam Nomani has been with IBM for 14 years with most of that time spent with the Quality Assurance team. His main area of focus has been on high availability and disaster recovery solutions. Over the past several years Aslam has been the Quality Assurance Architect for the DB2 pureScale Feature.

---

### Madhusudan K J

Madhusudan K J has been working in System Verification Testing for DB2 since 2006 at IBM. He is an IBM DB2 Certified Advanced DBA. He has special interest in DB2 high availability, backup, and recovery solutions.

---

### Santhees Kodi

Sathees Kodi is a Certified IT Specialist at IBM and has been working on storage solutions since 2001. He has authored and published many white papers on backup and recovery, disaster recovery, cloning, high availability, and business continuity solutions for SAP and DB2.