IBM PowerHA SystemMirror for AIX

Standard Edition

Version 7.2

Smart Assists for PowerHA SystemMirror

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 171.

Contents

iv

About this document

You can use Smart Assist to create a highly available instance of a supported application in PowerHA SystemMirror. Smart Assist manages a collection of PowerHA SystemMirror components that you identify to support a particular application. You view these collection of PowerHA SystemMirror components as a single entity, and in PowerHA SystemMirror that entity is represented by an application name.

Highlighting

The following highlighting conventions are used in this document:

Bold	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Also identifies graphical objects such as buttons, labels, and icons that the user selects.
Italics	Identifies parameters whose actual names or values are to be supplied by the user.
`Monospace`	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you should actually type.

Case-sensitivity in AIX

Everything in the AIX operating system is case-sensitive, which means that it distinguishes between uppercase and lowercase letters. For example, you can use the `ls` command to list files. If you type LS, the system responds that the command `is not found`. Likewise, **FILEA**, **FiLea**, and **filea** are three distinct file names, even if they reside in the same directory. To avoid causing undesirable actions to be performed, always ensure that you use the correct case.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Related information

- The PowerHA SystemMirror Version 7.2 for AIX PDF documents are available in the PowerHA SystemMirror 7.2 PDFs topic.
- The PowerHA SystemMirror Version 7.2 for AIX release notes are available in the PowerHA SystemMirror 7.2 release notes topic.

Smart Assists for PowerHA SystemMirror

Smart Assist manages a collection of PowerHA SystemMirror components that you identify to support a particular application. You can view these collections of PowerHA SystemMirror components as a single entity, and in PowerHA SystemMirror that entity is represented by an application name.

What's new in Smart Assists for PowerHA SystemMirror

Read about new or significantly changed information for the Smart Assists for PowerHA SystemMirror topic collection.

How to see what's new or changed

In this PDF file, you might see revision tags (>| and |<) surrounding new and changed information.

December 2023

- Updated information about WebSphere® MQSeries®, IBM Tivoli® Directory Server, IBM Lotus® Domino® Server, SAP liveCache Hot Standby, and IBM Tivoli Storage Manager for AIX version 7.2 and 7.3 in the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

December 2022

- Updated information about WebSphere® MQSeries®, MaxDB, and IBM Tivoli® Storage Manager in the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

December 2021

- Added information about WebSphere MQSeries in the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

December 2020

- Added information about IBM Tivoli Storage Manager in the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.
- Added a topic about "Migration support for Oracle database instance " on page 29.

Support matrix for PowerHA SystemMirror Smart Assists

You can use PowerHA SystemMirror Smart Assists to configure and manage middleware applications for high availability.

The following table describes the versions of middleware that can be configured and managed with PowerHA SystemMirror Smart Assists. For information about supported PowerHA SystemMirror version, see PowerHA SystemMirror for AIX Reference Information.

Table 1. Support matrix for PowerHA SystemMirror Smart Assists. The table describes the middleware application versions that are supported on PowerHA SystemMirror with different versions of the AIX operating system.		
Middleware application	**AIX 7.2**	**AIX Version 7.3**
AIX print subsystem, DHCP, and DNS	7.2	7.3
Oracle Database Server	19C	19C

Table 1. Support matrix for PowerHA SystemMirror Smart Assists. The table describes the middleware application versions that are supported on PowerHA SystemMirror with different versions of the AIX operating system. (continued)		
Middleware application	**AIX 7.2**	**AIX Version 7.3**
SAP Netweaver	7.52	7.52
Db2®	11.5	11.5
WebSphere MQSeries	9.3	9.3
IBM Tivoli Directory Server	6.4	6.4
IBM Lotus® Domino® Server **Note:** Support is withdrawn for this product from PowerHA SystemMirror 7.2.8 or later.	NA	NA
SAP liveCache Hot Standby **Note:** Support is withdrawn for this product from PowerHA SystemMirror 7.2.8 or later.	NA	NA
MaxDB	7.9.08	7.9.08
IBM Tivoli Storage Manager **Note:** The product is now known as IBM Spectrum® Protect™.	8.1.8	8.1.8

Installing PowerHA SystemMirror Smart Assist

You can install PowerHA SystemMirror Smart Assist from media, a hard drive, or an installation server.

The media for installing PowerHA SystemMirror Smart Assist contains the following files sets:

Table 2. Contents of the installation media	
Fileset	**Description**
`cluster.es.assist.filenet`	PowerHA SystemMirror Smart Assist for FileNet® P8
`cluster.es.assist.ihs`	PowerHA SystemMirror Smart Assist for IBM HTTP Server
`cluster.es.assist.maxdb`	PowerHA SystemMirror Smart Assist for SAP MaxDB
`cluster.es.assist.oraappsrv`	PowerHA SystemMirror Smart Assist for Oracle Application Server
`cluster.es.assist.oracle`	PowerHA SystemMirror Smart Assist for Oracle
`cluster.es.assist.printServer`	PowerHA SystemMirror Smart Assist for Print Subsystem
`cluster.es.assist.sap`	PowerHA SystemMirror Smart Assist for SAP
`cluster.es.assist.tds`	PowerHA SystemMirror Smart Assist for IBM Tivoli Directory Server
`cluster.es.assist.tsmadmin`	PowerHA SystemMirror Smart Assist for IBM TSM Admin center

Table 2. Contents of the installation media (continued)	
Fileset	**Description**
`cluster.es.assist.tsmclient`	PowerHA SystemMirror Smart Assist for IBM TSM Client
`cluster.es.assist.tsmserver`	PowerHA SystemMirror Smart Assist for IBM TSM Server
`cluster.es.assist.websphere`	PowerHA SystemMirror Smart Assist for WebSphere

PowerHA SystemMirror Smart Assist installation

You must verify that your environment meets specific requirements before you can install the various PowerHA SystemMirror Smart Assist applications.

You must meet the following requirements before you install any of the PowerHA SystemMirror Smart Assist applications:

- The node has 2.5 MB of space in the **/usr** directory.
- The systems are running PowerHA SystemMirror 7.1, or later.
- The environment is using AIX Version 6.1 with the 6100-06 Technology Level, or later.

Installing PowerHA SystemMirror Smart Assist from a server

To install PowerHA SystemMirror Smart Assist software in a cluster environment, you can create a PowerHA SystemMirror Smart Assist installation server (containing the PowerHA SystemMirror Smart Assist installable images) on one node and then load the images onto the remaining cluster nodes.

Installing from the server is the fastest way to install PowerHA SystemMirror Smart Assist. PowerHA SystemMirror Smart Assist supports the Network Installation Management program and Alternate Disk Migration.

Installing PowerHA SystemMirror Smart Assist from a hard drive

To install PowerHA SystemMirror Smart Assist from your hard disk, copy the software from the installation medium to the hard disk prior to installation.

To copy the PowerHA SystemMirror Smart Assist software to your hard disk, complete the following steps:

1. Place the product CD into the CD-ROM drive.
2. Enter `smit bffcreate`

   PowerHA SystemMirror displays the **Copy Software to Hard Disk for Future Installation** screen.
3. Enter the name of the CD-ROM drive in the **INPUT device/directory for software** field and press Enter.

   If you are unsure of the input device name, press F4 to list available devices. Select the proper drive and press Enter. That value is entered into the **INPUT device/directory** field as the valid input device.
4. Press Enter to display the **Copy Software to Hard Disk for Future Installation** screen.
5. Enter field values as follows:

Table 3. Copy Software to Hard Disk for Future Installation fields	
Field	**Value**
SOFTWARE name	Press F4 for a software listing. Install the images for PowerHA SystemMirror Smart Assist. For a list of the PowerHA SystemMirror Smart Assist images, see the section Contents of the installation media.

Table 3. Copy Software to Hard Disk for Future Installation fields (continued)	
Field	**Value**
DIRECTORY for storing software	Change the value to the storage directory accessed by all nodes using PowerHA SystemMirror Smart Assist.

6. Enter values for the other fields as appropriate for your site.

7. When you are satisfied with the entries, press Enter.

   SMIT responds:

   ```
   Are you sure?
   ```

8. Press Enter again to copy the software.

9. After the PowerHA SystemMirror Smart Assist software has been copied to your system, install the software by following the instructions in the "Installing PowerHA SystemMirror Smart Assist from the CD-ROM" on page 4 topic.

Installing PowerHA SystemMirror Smart Assist from the CD-ROM

To install PowerHA SystemMirror Smart Assist from the CD-ROM, install the software directly onto each cluster node.

To install the PowerHA SystemMirror Smart Assist software on a cluster node, complete the following steps:

1. Insert the product CD into the CD-ROM drive.

2. Enter smit install_all

   PowerHA SystemMirror displays the first **Install and Update from ALL Available Software** panel.

3. Enter the device name of the installation medium or install directory in the **INPUT device/directory for software** field and press Enter.

   If you are unsure about the input device name or about the install directory, press F4 to list available devices. Then select the proper device or directory and press Enter. The correct value is entered into the **INPUT device/directory** field as the valid input device.

4. Accept the default settings for the following fields. If you need to change a setting, you can do so.

   Use the context-sensitive (F1) Help to get more information about the field.

Table 4. INPUT device/directory for software fields	
Field	**Value**
INPUT device / directory for software	This field shows the device or directory you specified earlier.
SOFTWARE to install	Press F4 for a software listing. In the software list, use the arrow keys to locate all software filesets associated with an image. For a list of filesets, see the section Contents of the installation media earlier in this chapter. Next press F7 to select either an image or a fileset. Then press Enter after making all selections. Your selections appear in this field.
PREVIEW only?	If set to **yes,** the preview option checks and verifies that installation prerequisites are met, for instance that required software is installed and sufficient disk space is available. Press F1 for details. When you are ready to perform the actual installation, set this field to **no**.
COMMIT software updates?	This field applies only when installing software updates (PTFs).

Table 4. INPUT device/directory for software fields (continued)	
Field	**Value**
SAVE replaced files?	This field applies only when installing software updates (PTFs). If you select **no** to **commit software updates?** you must select **yes** for this field.
AUTOMATICALLY install requisite software	Set this field to **no** if the prerequisite software is already installed or if the **OVERWRITE same or newer versions?** field is set to **yes**; otherwise, set this field to **yes** to install required software.
EXTEND filesystems if space needed?	Select **yes** if the system has adequate hard disk space. Select **no** if the system has limited disk space.
OVERWRITE same or newer versions?	For normal new installations, leave this field set to **no**. Set it to **yes** if you are reinstalling the software. If you set this field to **yes**, you must set the **Automatically install requisite software** field to **no**.
VERIFY install and check file sizes?	Select **yes** if you want the system to perform some checks on the software you installed.
DETAILED output?	Select **yes** if you want a detailed log of all installation messages.
Process multiple volumes?	Select this option if you want to enable the processing of multiple-volume CDs.

5. Accept the license agreement and preview the license. Enter field values as follows:

Table 5. License agreement fields	
Field	**Value**
ACCEPT new license agreements?	Select **yes** to proceed with installation. If you select **no**, installation may stop with a warning that one or more filesets require software license agreements. You accept the license agreement only once for each cluster node.
Preview new license agreements?	Select **yes** to view the text of the license agreements. The text displays in the current window according to the language defined on your system.

6. When you are satisfied with the entries, press Enter. SMIT prompts you to confirm your choices.

7. Press Enter again to install the software.

For more information, read the PowerHA SystemMirror Smart Assist Release Notes, which reside in the **/usr/es/sbin/cluster/haws/release_notes_haws** directory.

Removing PowerHA SystemMirror Smart Assist

If you remove an installation of PowerHA SystemMirror Smart Assist, removing the software does not remove the files generated by running the commands and scripts within PowerHA SystemMirror Smart Assist, and does not affect the PowerHA SystemMirror configuration. If you want to remove the files generated by PowerHA SystemMirror Smart Assist, delete the files from AIX.

To remove PowerHA SystemMirror Smart Assist, complete the following steps:

1. Stop cluster services:
   a) Enter `smit cl_admin`
   b) Select **PowerHA SystemMirror Services** > **Stop Cluster Services** and press Enter.
   c) Select **Stop now** and press Enter.
2. Enter `smit install_remove`. SMIT displays the **Install/Remove** panel.

3. Enter field values as follows:

Table 6. Install/Remove fields

Field	Value
SOFTWARE name	Use the arrow keys to locate all software you want to remove; then press F7 to select it. Remove the filesets listed in the section Contents of the installation media. Press Enter after making all selections. Your selections appear in this field.
REMOVE dependent software?	Select **no**.
EXTEND filesystems if space needed?	Select **yes**.
DETAILED output?	Select **no**.

Smart Assist for Oracle

Use this information for planning and configuring a PowerHA SystemMirror environment for Oracle components.

Before continuing with PowerHA SystemMirror Smart Assist for Oracle, familiarize yourself with Oracle Application Server and Oracle Relational Database Management System (RDBMS) instances from a conceptual and functional standpoint, and IBM AIX and PowerHA SystemMirror.

Getting started

PowerHA SystemMirror Smart Assist for Oracle discovers pre-installed Oracle Application Servers and Oracle database instances. In addition, it helps you to integrate PowerHA SystemMirror and Oracle Application Server clusters to monitor the Oracle processes and ensure that the applications remain available.

However, it does not suggest the best high availability configuration for your environment. For information on choosing the optimum high availability configuration for your Oracle environment, see the Oracle documentation listed on the Oracle documentation Web site.

You must install the Oracle and PowerHA SystemMirror software, and create and configure Oracle Application Server and database instances on all nodes before running PowerHA SystemMirror Smart Assist for Oracle.

Oracle installations are required to be in the OFA (Optimal Flexible Architecture) form. If the Oracle installation is not in an OFA directory structure, the discovery component will not function properly.

PowerHA SystemMirror Smart Assist for Oracle supplements the *Oracle Application Server Installation Guide*.

To use PowerHA SystemMirror Smart Assist for Oracle, install the software listed in the section Software installation prerequisites in on all nodes in the Oracle-related resource groups (which may be a subset of all cluster nodes).

Related information
Oracle Documentation

Enhancing PowerHA SystemMirror and Oracle integration

PowerHA SystemMirror Smart Assist for Oracle is an extension of the PowerHA SystemMirror software that helps you to make your Oracle Application Server cluster configuration and Oracle database instances highly available using the features provided by PowerHA SystemMirror.

PowerHA SystemMirror Smart Assist for Oracle lets you:

- Discover and configure the Oracle Application Server (AS) components, make the AS components highly available, and monitor those components for failure.
- Integrate PowerHA SystemMirror and Oracle database for a Cold Failover Cluster (CFC) environment and monitor the database for failure.
- Make the resources specific to the Oracle Infrastructure highly available, such as a shared volume group, file systems, and the service IP label associated with the Oracle application.

This lets you create a mutual takeover cluster configuration and a cluster configuration with multiple nodes and resource groups with Oracle instances. PowerHA SystemMirror ensures the availability of these instances by moving resource groups from one node to another when the conditions in the cluster change.

- Start and stop the Oracle Application Server business applications on the nodes automatically, by the means of a PowerHA SystemMirror application server (a collection of start and stop scripts in PowerHA SystemMirror) created for the Oracle instance in the PowerHA SystemMirror cluster.
- Automatically monitor the Oracle database instance(s) running on the nodes as well as Oracle Application Server.
- Verify the existing configuration of the Oracle components to ensure that the Oracle and PowerHA SystemMirror configuration is valid.

Keeping Oracle Application server highly available

PowerHA SystemMirror increases the availability of Oracle Application Server instance by eliminating single points of failure. A *single point of failure* exists when a critical function relies on a single component in a configuration. If that component fails, the application dependent on that component becomes unavailable.

The primary components required to deploy a PowerHA SystemMirror cluster for the Oracle Application Server are:

- *Middle Tier Application Server*. The Middle Tier Application Server communicates with the cluster through the virtual IP address (VIP).

  The middle tier hosts most of the Oracle Application Server business applications. These applications rely on Oracle AS Infrastructure for security and metadata support. The middle tier also includes a Web caching sub-tier (Oracle Application Server Web Cache), and a Web server sub-tier (Oracle HTTP Server).

- *Infrastructure Tier*. Consists of two parts: Oracle AS Metadata Repository and Oracle Identity Management (IM). Together, they provide centralized metadata, management, and security services for Oracle Application Server components.

  A highly available Oracle AS deployment requires a highly available Infrastructure service. Uninterrupted access to Oracle Identity Management, installed as part of the infrastructure, is in the critical path to the availability of other application services.

The Oracle AS infrastructure tier components are grouped into the following tiers:

- *OID tier*. Provides Oracle Internet Directory (OID) services including directory services, directory integration services to integrate OID with third-party directories, etc. The main processes in this tier are the OID (Oracle Internet Directory) and Oracle Directory Integration and Provisioning (DIP) processes.
- *SSO tier*. Provides Single Sign On (SSO) and Delegation Administration services (DAS). The main processes in this tier are the Oracle HTTP Server (OHP) and OC4j instances hosting SSO and DAS applications.

  The OID tier and SSO tier together provide the Identity Management services.

- *Database Tier*. An Oracle database serves as the metadata repository (MR). The same database may contain the metadata repository and the schemas used to hold application data. The processes in this tier are the database instance processes and the database listener.

You may group Middle-tier components and Infrastructure tier components on two separate clusters. Usually, middle tier components are configured with AFC and infrastructure components with CFC.

We recommend that you have all three tiers of Infrastructure components (OID tier, SSO tier and Database tier) in one cluster. Each tier will have its own resource group. These tiers can be distributed among multiple nodes.

PowerHA SystemMirror views the application server as the Oracle Infrastructure Tier. PowerHA SystemMirror Smart Assist for Oracle provides support for both the Oracle Application Server configured for a Cold Failover Cluster (CFC) (the PowerHA SystemMirror two-node cluster with a hot standby configuration), and the Oracle Application Server configured for an Active Failover Cluster (AFC) (the PowerHA SystemMirror two-node configuration for concurrent access).

To protect Oracle Application Servers and eliminate single points of failure, each Oracle Application Server instance that you want to make highly available must be able to run on at least two nodes. Clustering provides high availability for Oracle Application Server middle tier by uniting various Oracle Application Server components in certain permutations. This clustering offers scalable and unified functionality and redundancy should any of the individual components fail.

You can configure each of the three core Oracle Application Server components in multiple permutations (Application Server middle tier, Identity Management, metadata repository) as follows:

- All components installed on the same node
- All components installed on separate nodes
- Two components on one node, one on another node.
- The node mentioned in the list previously is not required to be a PowerHA SystemMirror-managed node. However, PowerHA SystemMirror only discovers and makes highly available those components that are installed on PowerHA SystemMirror managed AIX systems.

PowerHA SystemMirror Smart Assist for Oracle configures each of the three Oracle Application Server components in their own resource group. You can configure each of the tiers as either a Cold Failover Cluster (CFC) or Active Failover Cluster (AFC) as follows:

- *Cold Failover Cluster*. The participating nodes share the same set of resources (volume groups, and logical volumes and file systems). PowerHA SystemMirror ensures the availability of cluster resources by moving resource groups from one node to another when the conditions in the cluster change.
- *Active Failover Cluster*. Each node has its own resource group. PowerHA SystemMirror monitors the resource groups for application server failure and restarts the application server on that node - no movement of resources takes place in an AFC configuration.

Note that some components may be configured in an AFC configuration and some may be configured in a CFC configuration, for example, an Oracle Application Server middle tier configured in an AFC configuration, with the metadata repository and identity management components operating on one node in a CFC configuration.

Increasing availability for an Oracle Application server

PowerHA SystemMirror Smart Assist for Oracle provides several availability features for the Oracle Application server.

These features include:

- Monitors for the Oracle database instance, verifies background processes, listener-related processes, and the state of each instance.
- Monitors for Oracle Application Server use of the Oracle Process Manager and Notification (OPMN) utility. When a failure is detected by any of the components, an attempt is made to restart (up to 3 times by default).

For AFC each node has its own resource group (SSO, OID or MR) to monitor for application server failure and to restart the application server on that node - no movement of resources takes place in an AFC configuration. If any component fails, PowerHA SystemMirror attempts to restart it on the same node. Therefore, for all three resource groups when restart fails (up to 3 times by default), PowerHA SystemMirror logs the failure information and makes no further restart attempts.

Supported Oracle Application server fallover configurations

PowerHA SystemMirror Smart Assist for Oracle supports several cluster configurations.

These configurations are:

- *Hot standby*. The Oracle Application Server CFC configuration, where one node performs the work while the other node waits as a fallover location.
- *Mutual takeover*. The Oracle Application Server AFC Standard Edition configured with PowerHA SystemMirror provides each node of the two-node cluster with a hot standby simultaneous access to process incoming requests.
- *Multiple node*. The Oracle Application Server AFC Enterprise Edition configured with PowerHA SystemMirror allows for a maximum of 32 active/active nodes (limited by your hardware) with simultaneous access to the shared disks with all nodes owning the same disk resources.

Related concepts

Configuring a highly available Application server infrastructure
You can configure PowerHA SystemMirror and Oracle Application server infrastructure in different cluster configurations.

Overview of Oracle database

Oracle database is comprised of an instance and data storage. An instance is a set of operating-system processes and memory structures that interact with the storage.

These processes are also called as background processes or detached processes. Typical background processes include:

- DBWn (Database Writer Process)
- LGWR (Log Writer Process)
- CKPT (Checkpoint Process)
- PMON (Process Monitor)
- SMON (System Monitor).

Oracle refers to the server-side memory structure as the SGA (System Global Area). The SGA typically holds cache information like data buffers, SQL commands and user information. In addition to storage, the database consists of online redo logs (which hold transactional history). You can activate background processes to archive the online redo logs into archive logs (offline redo logs), which provide the basis for data recovery.

The PowerHA SystemMirror Smart Assist for Oracle assists you in configuring Oracle database in a Cold Failover Cluster (CFC) environment.

Configuring a highly available Application server infrastructure

You can configure PowerHA SystemMirror and Oracle Application server infrastructure in different cluster configurations.

PowerHA SystemMirror Smart Assist for Oracle helps you to integrate PowerHA SystemMirror and your existing Oracle Application Server configured for CFC or AFC, and Oracle database instances configured for CFC. The PowerHA SystemMirror Smart Assist for Oracle assumes these configurations are configured and running. For all other configurations listed, define the resource groups according to implementation requirements.

Basic Cold Failover Cluster

PowerHA SystemMirror Smart Assist for Oracle supports the Oracle Application Server configured for Cold Failover Cluster (CFC).

The Oracle Application Server configured for CFC is the PowerHA SystemMirror two-node with a hot standby configuration, where one node performs the work while the other node waits as a fallover

location. The PowerHA SystemMirror nodes and a shared storage device are the primary components of the cluster. By including the Oracle application in your two-node cluster with hot standby, PowerHA SystemMirror ensures high availability for the Identity Management (IM) and Metadata Repository (MR) components of an Oracle AS deployment.

In this CFC configuration, any one node of the cluster provides the infrastructure service at any given time through a virtual hostname. The virtual hostname is bound to any one node of the cluster at a time but can float over to any other node. PowerHA SystemMirror automatically manages failover of the service across the nodes of the cluster.

The following figure displays a basic Oracle Application Server for CFC configuration.
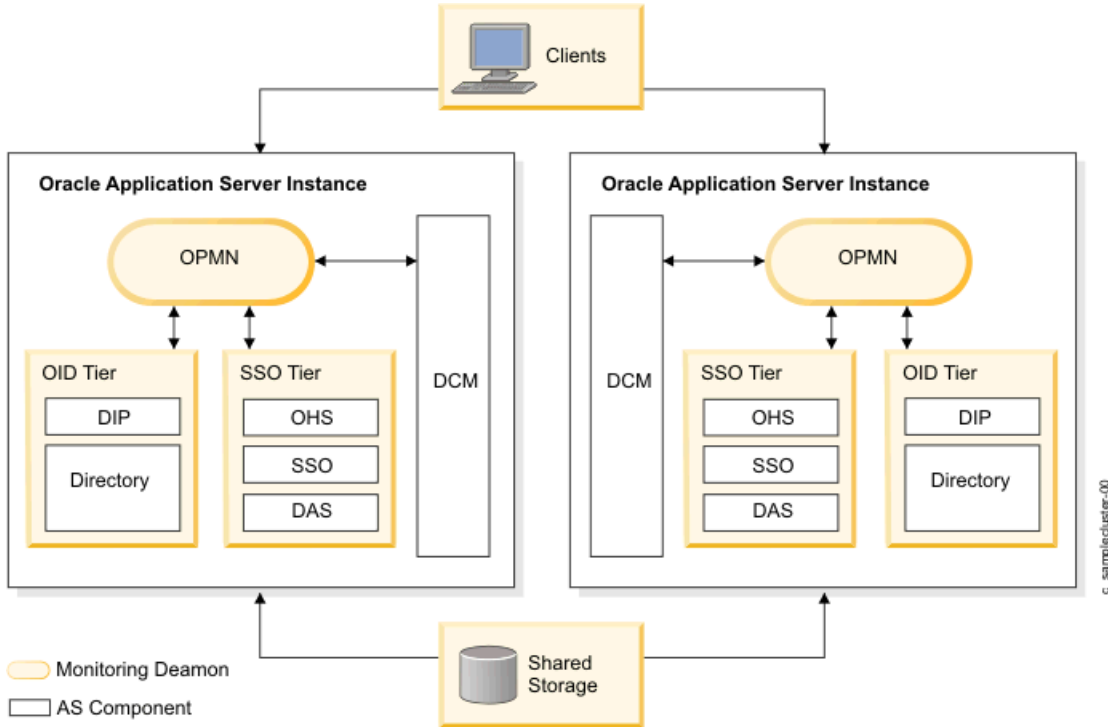
Figure 1. Basic Oracle Application Server for CFC configuration

Cold Failover Cluster infrastructure components
Cold Failover Cluster has several different infrastructure components.

The following table lists the infrastructure components for the Cold Failover Cluster:

Infrastructure Component	Configuration
Database Tier	Active-Passive
OID Tier	Active-Passive
SSO Tier	Active-Passive
Hardware Requirements	• Shared storage (direct attached/NAS/SAN) • No load balancer is required.

Cold Failover Cluster resource groups
There are three types of resource groups for the Cold Failover Cluster (CFC) configuration.

The PowerHA SystemMirror Smart Assist for Oracle names these resource groups as follows:
- <App_Name>_CFC_RG_SSO
- <App_Name>_CFC_RG_OID

- <App_Name>_CFC_RG_MR.

For RDBMS DB, the **spfile** or **pfile** is used to form the application name, for AFC/CFC Application Server, the name is derived from the **opmn.xml ias-instance** name. *<App_Name>* is user-modifiable; short names are recommended. In one node there could be two instances of Oracle. Typically one Oracle instance for production and one instance for testing. This *<App_Name>* distinguishes the multiple instances.

Cold Failover Cluster resource group for SSO component
There are several attributes related to the SSO components in the Cold Failover Cluster configuration.

The following table shows the SSO components for the resource group attributes in a CFC configuration:

Attribute Name	Attribute Value
Resource Group Name	<App_Name>_CFC_RG_SSO
Startup Policy	Online On Home Node Only
Fallover Policy	Fallover to Next Priority Node in the List
Fallback Policy	Never Fallback
Service IP Label	IP address used by this component
Volume Groups	Volume groups used by this component
Filesystem Recovery Method	Sequential

The following table shows PowerHA SystemMirror Application Server name, start and stop scripts and monitor to handle the application server SSO component:

Attribute Name	Attribute Value
Server Name	<App_Name>_CFC_AP_<ias-comp>
Start Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStartAS**
Stop Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStopAS**
AS Infrastructure Monitors	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraMonitorAS**

Cold Failover Cluster resource group for OID components
Cold Failover Cluster (CFC) has several OID components.

The following table shows the OID components for the resource group attributes in a CFC configuration:

Attribute Name	Attribute Value
Resource Group Name	<App_Name>_CFC_RG_OID
Startup Policy	Online on Home Node Only
Fallover Policy	Fallover to Next Priority Node (In a two-node configuration this is the standby node.)
Fallback Policy	Never Fallback
Service IP Label	IP address used by this component
Volume Groups	Volume groups used by this component
Filesystem Recovery Method	Sequential

The following table shows PowerHA SystemMirror Application Server name, start and stop scripts, and monitor used to handle the application server OID component:

Attribute Name	Attribute Value
Server Name	<App_Name>_CFC_AP_<ias-comp>
Start Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStartAS**
Stop Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStopAS**
AS Infrastructure Monitor	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraMonitorAS**

Cold Failover Cluster resource group for MR components
A Cold Failover Cluster (CFC) configuration has several attributes for MR components.

The following table lists the resource group for MR component of AS in a CFC configuration:

Attribute Name	Attribute Value
Resource Group Name	<App_Name>_CFC_RG_MR
Startup Policy	Online on Home Node Only
Fallover Policy	Fallover to Next Priority Node (In a two-node configuration this is the standby node.)
Fallback Policy	Never Fallback
Service IP Label	IP address used by this component
Volume Groups	Volume groups used by this component
Filesystem Recovery Method	Sequential

The following table shows PowerHA SystemMirror Application Server name, start and stop scripts and monitor to handle the application server MR component:

Attribute Name	Attribute Value
Server Name	<App_Name>_CFC_AP_MR
Start Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStartDB**
Stop Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStopDB**
AS Infrastructure Monitor	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraMonitorDB**

Cold Failover Cluster middle tier components
The Application Server CFC configurations of any middle tier components have resource group attributes that are almost identical to that of Application Server CFC infrastructure related resource groups.

This is displayed in the following table, where they can be one of OC4J, Webcache, Portal, Forms and Oracle Business Intelligence Beans (BI Beans):

Attribute Name	Attribute Value
Resource Group Name	<App_Name>_CFC_RG
Startup Policy	Online on Home Node Only
Fallover Policy	Fallover to Next Priority Node (In a two-node configuration this is the standby node.)

Attribute Name	Attribute Value
Fallback Policy	Never Fallback
Service IP Label	IP address used by this component
Volume Groups	Volume groups used by this component
Filesystem Recovery Method	Sequential

Distributed Cold Failover Cluster

The distributed Cold Failover Cluster (CFC) configuration is based on the Application Server Basic Cold Failover Cluster, but pulls out the SSO tier into the Demilitarized Zone (DMZ) - outside of the firewall.

The database tier and OID tier are co-located under PowerHA SystemMirror in the two-node cluster configuration. The SSO tier is located on multiple non-clustered boxes. PowerHA SystemMirror automatically manages failover of the service across the nodes of the cluster.

Distributed Cold Failover Cluster infrastructure components

The following table lists the infrastructure components for the Distributed Cold Failover Cluster:

Infrastructure Component	Configuration
Database Tier	Active-Passive
OID Tier	Active-Passive
SSO Tier	Active-Active
Hardware Requirements	• Shared storage (direct attached/NAS/SAN) • Multiple non-clustered servers for SSO tier • Load balancer is required for SSO tier

Basic Active Failover Cluster

The Oracle Application Server configured for Active Failover Cluster (AFC) in an active-active environment is the basic two-node mutual takeover configuration for concurrent access.

In the basic AFC configuration, all nodes of cluster are in service. Multiple active instances provide continued availability in case of one of the instances fails. The database instance processes and application server processes run on both nodes. The cluster is front-ended by a load balancer.

Active Failover Cluster infrastructure components
Active Failover Cluster (AFC) has several different infrastructure components.

The following table lists the infrastructure components for the Basic Active Failover Cluster:

Infrastructure Component	Configuration
Database Tier	Active-Active (RAC)
OID Tier	Active-Active
SSO Tier	Active-Active
Hardware Requirements	• Shared storage • Load balancer is required to front-end OID and SSO tier on the cluster.

Active Failover Cluster resource groups

There are several types of resource groups for the Active Failover Cluster (AFC) configuration.

The following lists the three types of resource groups for the AFC configuration, as named by PowerHA SystemMirror Smart Assist for Oracle:

- <App_Name>_AFC_RG_SSO
- <App_Name>_AFC_RG_OID
- <App_Name>_AFC_RG_MR.

For RDBMS DB, the name from **spfile** or **pfile** is used to form the application name. For AFC/CFC Application Server the name is derived from the **opmn.xml ias-instance** name.

For AFC configurations, if any component fails, PowerHA SystemMirror attempts restart it on the same node but does not move resources to another node. Therefore, the SSO, OID and MR resource groups can use identical resource group policies as shown in the following table:

Attribute Name	Attribute Value
Resource Group Name	<App Name>_AFC_RG
Startup Policy	Online All Available Nodes
Fallover Policy	Bring Offline (on Error Node Only)
Fallback Policy	Never Fallback
Service IP Label	No need for Service IP label.
Volume Groups	Any log volume groups that exist for AS component
Filesystem Recovery Method	Sequential
Participating Nodes	"Local" node only

The following table shows PowerHA SystemMirror Application Server name, start and stop scripts and monitor for the application server components (these can be one of either SSO, OID, or MR):

Attribute Name	Attribute Value
Server Name	<App_Name>_AFC_AP_<ias-comp>
Start Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStartAS**
Stop Script	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStopAS**
AS Infrastructure Monitor	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraMonitorAS**

Active Failover Cluster middle tier components

PowerHA SystemMirror can monitor the OC4J, Webcache, Portal, Forms or BI Beans middle tier components. The resource groups for Active Failover Cluster (AFC) configurations for these components have identical attributes with respect to infrastructure components (OID, SSO and MR).

The following table shows attributes for the AFC middle tier components. The AFC middle tier component can be one of OC4J, Webcache, Portal, Forms or BI Beans.

Attribute Name	Attribute Value
Resource Group Name	<App Name>_AFC_RG
Startup Policy	Online On All Available Nodes
Fallover Policy	Bring Offline (on Error Node only)
Fallback Policy	Never Fallback

Attribute Name	Attribute Value
Service IP Label	No need for Service IP label.
Volume Groups	Any log volume groups that exist for AS component
Filesystem Recovery Method	Sequential
Participating Nodes	Local node only

Distributed Active Failover cluster

This configuration is based on the distributed two-node mutual takeover configuration for concurrent access. It is based on the Oracle Application Server configured for AFC, but pulls out the SSO tier into DMZ - outside of the firewall.

The database tier and OID tier are co-located. A load balancer front-ends this tier. The SSO tier is deployed on multiple non-clustered boxes. A load balancer front-ends this tier as well.

Distributed Active Failover Cluster infrastructure components

The following table lists the infrastructure components for the Distributed Active Failover Cluster:

Infrastructure Component	Configuration
Database Tier	Active-Active (RAC)
OID Tier	Active-Active
SSO Tier	Active-Active
Hardware Requirements	• Shared storage • Multiple non-clustered servers for SSO and load balancers.

Cold Failover Cluster IM against repCA database

The Database tier and Identity Management tier are separated in this architecture.

The Database tier is typically created with repCA (Repository Creation Agent) or it may be Metadata Repository only installed. The database tier may be Active-Passive or Active-Active (RAC). The Identity Management tier is installed separately.

Cold Failover Cluster IM against repCA database infrastructure components

The following table lists the infrastructure components for the Cold Failover Cluster IM against repCA database:

Infrastructure Component	Configuration
Database Tier	Active-Passive or Active-Active (RAC)
OID Tier	Active-Passive
SSO Tier	Active-Passive
Hardware Requirements	• Shared storage • No load balancer is required to configure this model.

Distributed Cold Failover Cluster IM against repCA database

Distributed Cold Failover Cluster IM against repCA database is similar to the Basic Cold Failover Cluster IM against repCA database configuration.

In this case, the SSO tier is pulled out and deployed in multiple non-clustered servers. A load balancer front-ends this tier.

Distributed Cold Failover Cluster IM against repCA database infrastructure components

The following table lists the infrastructure components for the distributed Cold Failover Cluster IM against repCA database:

Infrastructure Component	Configuration
Database Tier	Active-Passive or Active-Active (RAC)
OID Tier	Active-Passive
SSO Tier	Active-Active
Hardware Requirements	• Shared storage • Multiple non-clustered servers for SSO tier • Load balancer is required to configure this model for SSO tier.

Basic Multiple Node Cluster configuration

This configuration splits the Database tier and the IM tier.

The Database tier is typically created with repCA (Repository Creation Agent) or it may be Metadata Repository only installed. It may be Active-Passive or Active-Active (RAC). The IM tier is installed separately but is collocated on the same PowerHA SystemMirror cluster as the database (in a separate $ORACLE_HOME). One IM installation is performed on each node in the cluster.

Basic Multiple Node Cluster infrastructure components

The following table lists the infrastructure components for the Basic Multiple Node Cluster:

Infrastructure Component	Configuration
Database Tier	Active-Passive or Active-Active (RAC)
OID Tier	Active-Active
SSO Tier	Active-Active
Hardware Requirements	• Shared storage • Load balancer is required for IM (OID and SSO tier).

Distributed Multiple Node Cluster configuration

This architecture is similar to Basic Multiple Node Cluster configuration, but can be set up in different distributed configurations.

SSO tier can be pulled out and put in non-clustered hardware in the DMZ with a front-end load balancer or the OID tier can be pulled out.

Distributed Multiple Node Cluster infrastructure components

The following table lists the infrastructure components for the Distributed Multiple Node Cluster:

Table 7. Distributed Multiple Node Cluster components	
Infrastructure Component	**Configuration**
Database Tier	Active-Passive or Active-Active (RAC)
OID Tier	Active-Active
SSO Tier	Active-Active
Hardware Requirements	• Shared storage • Multiple non-clustered servers • Load balancers.

Replicated IM and distributed replicated IM

PowerHA SystemMirror is not required for these configurations.

Note that this configuration provides high availability for IM-related data of the Metadata Repository. It supports geographically distributed instances of IM for localized access. However, this solution does not provide high availability of metadata repository (other than IM related).

Monitoring Application server components

The Oracle Application Server components are managed using OPMN. In order to use OPMN, the monitoring daemon - **$ORACLE_HOME/opmn/bin/opmn** - is started on the system.

Use the `startall` argument with the **$ORACLE_HOME/opmn/bin/opmn** command to start all the Application Server components being monitored by OPMN.

You can start individual components using the following command:

```
$ORACLE_HOME/opmn/bin/opmnctl startproc process-type=<process>
```

Where <process> is the component such as HTTP_Server.

You can monitor the application server processes using the **opmnctl** command with status argument. To get the status of all AS processes, use the command:

```
$ORACLE_HOME/opmn/bin/opmnctl status
```

Executing the previous command display results similar to the following:

Table 8. $ORACLE_HOME/opmn/bin/opmnctl status command output			
ias component	**process type**	**pid**	**status**
OID	OID	479045	Alive
OC4J	home	483582	Alive
HTTP_Server	HTTP_Server	450574	Alive
dcm-daemon	dcm-daemon	470923	Alive
LogLoader	logloaderd	460932	Alive

Cluster monitor scripts

PowerHA SystemMirror provides custom monitor scripts for Oracle AS in the form cl_oraMonitorAS.

All such scripts have certain common properties that can be listed in a tabular form, as shown in the following table:

Attribute Name	Attribute Value
Method	**/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraMonitorAS**
Monitor Mode	LONG-RUNNING MONITORING
Stabilization Interval	180 (- 3minutes)
Restart Count	3
Restart Interval	1 minute or 60 seconds
Action On Application Failure	Notify for AFC, fallover for CFC
Cleanup Method	/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStopAS
Restart Method	/usr/es/sbin/cluster/sa/oracle/sbin/cl_oraStartAS

Planning your configuration

The PowerHA SystemMirror Smart Assist for Oracle discovers the Oracle components.

It also assists in integrating PowerHA SystemMirror to monitor and manage Oracle Application Server configured for CFC or AFC and Oracle database instances configured for CFC. However, it does not suggest the best high availability configuration for a user environment and automatically configure that high availability configuration.

Related information
Oracle Documentation

Hardware system requirements

Make sure that your system meets requirements for a successful installation.

These requirements include:

• Two IBM System p 650 servers (minimum). You need at least two nodes in the Infrastructure Tier for a highly available environment - if one node fails, the second node takes over for the node that fails.
• Memory:
  – 1536 MB Swap space (paging space)
  – 400 MB **/tmp** space
  – 25 MB **/var** space for cluster verification
• Disk Storage: 4.3 GB Shared Disk Storage.
• Shared disk storage is not required for an AFC configuration.

Supported software

PowerHA SystemMirror Smart Assist for Oracle extends and improves upon the high availability solutions that were previously available in PowerHA SystemMirror.

To view what versions of Oracle, Oracle Application Server, and Oracle Database Server, are supported see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

License and user privilege

PowerHA SystemMirror Smart Assist for Oracle requires specific license and user privileges for installation.

You must meet the following requirements to install PowerHA SystemMirror:

- Each cluster node has its own PowerHA SystemMirror software license.
- A user with root privileges to perform the installation.

Limitations

PowerHA SystemMirror Smart Assist for Oracle has certain limitations

These limitations include:

- Oracle installations are required to be in the OFA (Optimal Flexible Architecture) form.
- The monitor scripts for Oracle database attempt to restart the instance if the database instance state is not OPEN. Therefore, you must suspend the instance monitor before closing and/or dismounting a particular instance for any administrative reason (such as file system backup). Otherwise, the monitor script for the database instance unnecessarily tries to restart and re-open the instance.

Configuring Oracle in a PowerHA SystemMirror cluster

PowerHA SystemMirror Smart Assist for Oracle helps you to integrate PowerHA SystemMirror and your existing Oracle Application Server configured for CFC or AFC, and Oracle database instances configured for CFC. The PowerHA SystemMirror Smart Assist for Oracle assumes these applications are configured and running.

This Smart Assist for Oracle also sets up monitoring for Oracle AS and Oracle database instances and processes. In order to bring the Application Server and database under the control of PowerHA SystemMirror, you must have pre-installed the respective Oracle components.

Once the PowerHA SystemMirror Smart Assist for Oracle integrates PowerHA SystemMirror and your existing Oracle Application Server configured for CFC or AFC, and Oracle database instances configured for CFC, you can extend the configuration using the regular SMIT panels.

Configuration overview

PowerHA SystemMirror Smart Assist for Oracle runs the discovery process and sets up your PowerHA SystemMirror resource configuration for you. It uses information it gathers from the application configuration, the system configuration, and information you enter to create a PowerHA SystemMirror resource group.

PowerHA SystemMirror resource groups contain a set of resources such as cluster nodes, volume groups, and IP labels (used by server applications to communicate with clients) that PowerHA SystemMirror manages as a unit. PowerHA SystemMirror uses the resource group information to protect the Oracle application and keep it highly available.

In addition to creating a resource group for the Oracle application to be made highly available, PowerHA SystemMirror Smart Assist for Oracle performs the following tasks:

- Helps you to create the PowerHA SystemMirror application server (a collection of the Oracle instance start and stop scripts), shared volume group and resource group, and configures the two-node cluster with shared disk(s).
- When PowerHA SystemMirror Smart Assist for Oracle creates the PowerHA SystemMirror resource configuration, it updates the PowerHA SystemMirror configuration but does not change any of the supported application configurations.
- Helps you to integrate PowerHA SystemMirror and Oracle Application Server clusters to monitor the Oracle processes and ensure that the applications remain available.

- Helps you to integrate PowerHA SystemMirror for the Oracle database instance configured for a Cold Failover Cluster.
- Verifies the configuration, and optionally tests the cluster, and starts PowerHA SystemMirror cluster services.
- Stores the generated PowerHA SystemMirror configuration in the PowerHA SystemMirror Configuration Database.

Prerequisites

Before configuring the PowerHA SystemMirror two-node cluster using the Smart Assist for Oracle, ensure that you have these prerequisites completed.

Before reading this topic, you should plan your configuration and install PowerHA SystemMirror Smart Assist for Oracle.

1. Ensure your system meets the Hardware system requirements.
2. Install AIX, PowerHA SystemMirror and the PowerHA SystemMirror Smart Assist for Oracle filesets on all cluster nodes.
3. Before using the Smart Assist to add or modify an application, appropriate file systems must be mounted, and the **ORACLE_HOME** and **ORACLE_SID** shell variables must be set and exported. Otherwise, the pick list for certain field values might be unavailable.

   In an Active Failover Clusters (AFC) configuration, the **ORACLE_HOME** and **ORACLE_SID** shell variables must be identical across all AFC nodes.
4. Ensure that the database and listener are active before running discovery. If discovery is run without the database and listener active, the instance is found, but the version cannot be determined. If the version is unknown, the application monitor might not be able to check for all the required processes.

   If the database has already been made highly available and the version is missing from PowerHA SystemMirrorODM class HACMPsa_metadata, then the simplest solution is to remove the application, make the database and instance active, and then re-add the application.
5. Add the PowerHA SystemMirror service IP label and the standby node hostname to **/etc/hosts** on all nodes. The cluster event scripts use the **/etc/hosts** file for name resolution. You must add all cluster node IP interfaces (boot and service) to this file on each node. Note that if you do not add the service IP labels to the **/etc/hosts** file on any cluster node, then PowerHA SystemMirror automatically adds them to this file as part of the automatic corrective action of the cluster verification process.
6. Ensure the service labels are resolvable on the local node.
7. Ensure that all participating nodes in the Oracle Application Server or database instance are accessible (running, and communication interfaces are available).
8. If PowerHA SystemMirror cluster services are running, stop them before proceeding:

   a. Enter the fastpath `smit cl_admin`

   b. Select **PowerHA SystemMirror Services** > **Stop Cluster Services**

   c. Select **Stop now** and press Enter.

Oracle database listener

If Oracle database listener is configured for Oracle database, PowerHA SystemMirror manages it along with the Oracle database instance.

The **SID_NAME** parameter in the `listener.ora` file must be set to the **ORACLE_SID** of the Oracle database instance to be discovered by the Smart Assist for Oracle software. Multiple Oracle database listeners can point to the Oracle database instance; however, you must ensure that all Oracle database listeners fall over correctly since Smart Assist for Oracle software can currently detect only one `listener.ora` file. An Oracle database listener that listens for more than one Oracle instance is not supported by the Smart Assist for Oracle software.

With PowerHA SystemMirror Version 7.2.2, or later, each Oracle database listener thread has its own monitor, which ensures that high availability of each Oracle database listener is managed by an independent monitor. The monitor performs the following tasks:

- If a failure is detected in any Oracle database listener thread, the associated monitor restarts only the specific Oracle listener thread without disturbing other Oracle database listeners and other Oracle database instances.
- The main listener monitor checks only the Oracle database instance. But upon failure of Oracle database instance, the main listener monitor restarts all Oracle database listeners that are associated with it.

Note: PowerHA SystemMirror Version 7.2.2, or later, supports monitoring of multiple Oracle database listeners. When you migrate to PowerHA SystemMirror Version 7.2.2, or later, from an earlier release, and if the Smart Assist applications for Oracle database is configured, additional monitors are automatically added to the configuration.

When a single Oracle database instance is configured with many Oracle database listeners, the number of associated monitors also increase. This might cause problems while starting the cluster, unless adequate stabilization interval is specified for initialization of all monitors. You might need to adjust the stabilization interval of monitors such that all monitors can initialize before starting the monitoring of the Oracle database instance. Consider the following points during the migration process to avoid problems that are caused by an inadequate stabilization interval:

- Before you upgrade to PowerHA SystemMirror Version 7.2.2 or PowerHA SystemMirror Version 7.2.3 by using the rolling migration method or by using the **cl_ezupdate** command, and if Smart Assist applications are configured for Oracle database, you can add the CL_LM_RDBMS_STAB_INT_MIN environment variable to the /etc/environment file, which changes the stabilization interval temporarily without modifying the cluster configuration. The stabilization interval must be configured depending on the number of Oracle database listeners and ensure that the stabilization interval is adequate for initializing all monitors. The value of the CL_LM_RDBMS_STAB_INT_MIN environment variable can be calculated by using the following formula:

```
CL_LM_RDBMS_STAB_INT_MIN=180+(30*number_of_listeners)
```

If you are migrating to PowerHA SystemMirror Version 7.2.2 by using the rolling migration method, ensure that you apply the interim fix for APAR IJ18784 before you start cluster services. If you are migrating to PowerHA SystemMirror Version 7.2.3 by using the rolling migration method, you must apply the interim fix for APAR IJ20653.

During migration, PowerHA SystemMirror uses the value that is defined for the CL_LM_RDBMS_STAB_INT_MIN environment variable in the /etc/environment file instead of the value that was specified during the cluster configuration. After the migration operation completes, you can permanently change the value for the **Stabilization Interval** property in the cluster configuration to the value that is defined for the CL_LM_RDBMS_STAB_INT_MIN variable in the /etc/environment file, and then remove the entry for the CL_LM_RDBMS_STAB_INT_MIN variable from the /etc/environment file.

- When you are migrating from PowerHA SystemMirror Version 7.2.2 to PowerHA SystemMirror Version 7.2.3, or later, and if you used Smart Assist for Oracle to configure applications for Oracle database, the following problems might occur:

Duplicate Oracle listener monitors are created during the migration process
If you have a Smart Assist configuration for Oracle database with more than 60 Oracle database listeners, duplicate Oracle listener monitors are created during the migration process, resulting in 120 Oracle listener monitors. Because of these duplicate Oracle listener monitors, cluster services might fail to start or the cluster might display an error state. To avoid the problem of duplicate Oracle listener monitors, you can perform an offline migration or a snapshot migration.

The log file shows that Oracle listener monitors are not created during the migration process

The `clconvert.log` file that is created during the migration process displays the following message:

```
Unexpected error encountered while creating oracle listener monitors.
```

This message does not impact the functioning of PowerHA SystemMirror. Therefore, you can ignore this message.

Note: In either scenario, you must apply the relevant interim fix and update the stabilization interval as mentioned in the subsequent paragraph after the migration operation is complete.

• After you migrate to PowerHA SystemMirror Version 7.2.2, and if Smart Assist applications are configured for Oracle database, you must apply the interim fix for APAR IJ17664 to ensure that sufficient stabilization interval is configured for each monitor. If you migrate to PowerHA SystemMirror Version 7.2.3, you must apply the interim fix for APAR IJ17904. The stabilization interval must be configured depending on the number of Oracle database listeners. To update the stabilization interval, complete the following steps:

1. Run the following script on any node in the cluster:

```
/usr/es/sbin/cluster/utilities/cl_update_oracle_listener_monitor
```

2. Verify and synchronize the cluster. For more information, see PowerHA SystemMirror verification and synchronization.

3. Ensure that the stabilization interval is adequate to avoid false errors while starting resource group. You must remove previously added entry for the CL_LM_RDBMS_STAB_INT_MIN variable from the `/etc/environment` file.

Configuring PowerHA SystemMirror for Oracle Application server

PowerHA SystemMirror Smart Assist for Oracle assumes you have pre-configured your Oracle Application Server. PowerHA SystemMirror Smart Assist for Oracle discovers and helps you to configure the Oracle application server components, make those components highly available, and monitor those components for failure.

PowerHA SystemMirror Smart Assist for Oracle configures one resource group for each of the components outlined in the following list:

• Middle Tier

• Oracle Internet Directory (OID) Tier

• Single Sign On Tier.

You can configure each of these tiers as either a Cold Failover Cluster (CFC) or an Active Failover Cluster (AFC) and set up the Oracle Application Server Middle Tier components for high availability in two configurations - Active-Passive and Active-Active.

You can configure each of the tiers as either CFC or AFC. The PowerHA SystemMirror Smart Assist for Oracle helps you to add the necessary PowerHA SystemMirror cluster components, such as a resource group and its associated resources. Configuring an Active Failover Cluster is similar to setting up Cold Failover Clusters, except with AFC where the infrastructure is active on all nodes cluster at the same time.

Related tasks
Testing the availability of Oracle applications
Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Related information
Oracle Documentation

Discovering and configuring Oracle components

You can use the SMIT interface to configure and discover Oracle components.

The Smart Assist discovers the applications and prompts you for information it needs to configure them for you.

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of the applications installed on the cluster.
3. Select the **Oracle Smart Assist** application.

Configuring Oracle Application server configured for CFC

The PowerHA SystemMirror Smart Assist for Oracle assumes you have pre-configured the Oracle Application Server. PowerHA SystemMirror performs discovery and creates one resource group for the Middle Tier, Oracle Internet Directory (OID) Tier, and Single Sign On Tier.

PowerHA SystemMirror Smart Assist for Oracle provides the add, change/show, and remove options for modifying the PowerHA SystemMirror configuration to work with pre-installed Oracle Application Server configured for CFC environments.

Adding an Oracle Application server configured for CFC

An Oracle Application Server configured for CFC must be installed for PowerHA SystemMirror to discover the installation. After you enter the primary node, and secondary node, PowerHA SystemMirror Smart Assist for Oracle performs discovery on the primary node in Oracle home directory. PowerHA SystemMirror determines the Oracle Application Server processes that require monitoring and the instances that are accessible within the Oracle directory structure.

With Oracle Application Server configured for CFC - the PowerHA SystemMirror hot standby cluster configuration - the Oracle instance is hosted on one node, with the other node serving as a standby node. In case of a cluster failure, the resource group managing the Oracle instance falls over to the standby node.

To add a new Oracle Application Server configured for CFC:

1. Set and export the ORACLE_HOME shell variable on the primary node.
2. On the primary node, enter `smit hacmp`
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
4. If discovery finds a two-node cluster configured, PowerHA SystemMirror Smart Assist for Oracle presents the Oracle Application Server Configuration panel. If discovery does not find a two-node cluster, PowerHA SystemMirror Smart Assist for Oracle displays the Two-Node Cluster Configuration Assistant SMIT panel.
5. Select **Oracle CFC Application Server** from the list of discovered applications and press Enter.
6. Select **CFC Instance** to bring online with the Oracle Application Server configured for CFC. The default is the first Oracle instance found in the Oracle home directory.
7. The **Add an Oracle CFC Application** server panel appears. Enter field values as follows:

Table 9. Add an Oracle CFC Application fields

Field	Value
Oracle AS CFC Instance Name	The name of the Oracle instance to bring online with the Oracle Application Server configured for CFC.
	The default displays the first Oracle instance found in the Oracle home directory and inserts the volume group for the instance in the Shared Volume Group field. If you select an instance different from the default, you must also specify the appropriate shared volume group for that instance in the Add an Oracle Application Server panel.
Application Name	The current Oracle Application name is discovered. If the name is longer than 32 characters, shorten it.
Primary Node	Select a node from a list of predefined cluster nodes. The PowerHA SystemMirror Smart Assist for Oracle discovers the nodes this instance is accessible on and provides those node names. You can remove or exclude nodes from this list.
Takeover Node	Select a node from a list of predefined cluster nodes.
	The takeover node acquires the Oracle instance in the event of an instance failure on the primary node. You can specify one or more takeover nodes.
	The system uses the local cluster node and an additional node name provided by you to form the PowerHA SystemMirror cluster topology configuration with two nodes.
Oracle Metadata Repository	Select the Oracle RDBMS to serve as the metadata repository (MR) from the picklist. You may use the same database to contain the metadata repository and the schemas used to hold application data.
Service IP Label	Enter one service IP label to use for both the Oracle listener and the Apache HTTP server. You may either:
	• Select from the picklist of pre-configured service IP labels that are not configured to a PowerHA SystemMirror resource group.
	or
	• Specify a service IP label that is not configured to PowerHA SystemMirror; Smart Assist for Oracle adds that service IP label to the PowerHA SystemMirror configuration and to the resultant PowerHA SystemMirror resource group.
Shared Volume Groups	Select from the picklist the shared volume group(s) that contain the tablespaces, logs, etc., required for the selected Oracle SID/Instance.
	By default, the picklist provides those volume groups associated with the first Oracle instance (displayed in the Oracle instance field). You must change the volume groups if you selected an Oracle instance different from the default.
Oracle OID and SSO Components	Select from the picklist the OID and SSO components for the Oracle AS infrastructure tier. You may select one or more components.

8. Press Enter.

Smart Assist for Oracle constructs the resource group, application server, custom application monitors, and service IP label and associates the shared volume groups with the constructed resource group. A summary report appears indicating the Oracle Application Server components discovered, and the processes that PowerHA SystemMirror monitors.

9. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

PowerHA SystemMirror propagates your changes to all cluster nodes.

10. Test your configuration.

Related concepts

Configuring a highly available Application server infrastructure

You can configure PowerHA SystemMirror and Oracle Application server infrastructure in different cluster configurations.

Basic Active Failover Cluster

The Oracle Application Server configured for Active Failover Cluster (AFC) in an active-active environment is the basic two-node mutual takeover configuration for concurrent access.

Getting started

PowerHA SystemMirror Smart Assist for Oracle discovers pre-installed Oracle Application Servers and Oracle database instances. In addition, it helps you to integrate PowerHA SystemMirror and Oracle Application Server clusters to monitor the Oracle processes and ensure that the applications remain available.

Related tasks

Testing the availability of Oracle applications

Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Changing or showing an Oracle Application server configured for CFC

Using the **Change/Show a CFC Oracle Application Server SMIT panel**, you may view or change the Oracle instance to bring online, the service IP label, and the shared volume groups associated with the instance. Changing the Oracle instance causes PowerHA SystemMirror to remove the existing configuration, resource group, Oracle Application Server, and custom monitors before recreating the new Oracle Application Server.

To change or show an existing Oracle Application Server configured for CFC:

1. Set and export the ORACLE_HOME shell variable on the primary node before making any changes.
2. On the primary node, enter `smit hacmp`
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.
4. Select an Oracle Application Server CFC Configuration to change or show from the picklist.
5. The **Modify an Oracle CFC Application Server** panel appears. To change the configuration, enter field values as shown:

Table 10. Modify an Oracle CFC Application Server fields

Field	Value
Oracle AS CFC Instance Name	The name of the Oracle instance to bring online with the Oracle Application Server configured for CFC. The default displays the first Oracle instance found in the Oracle home directory and inserts the volume group for the instance in the Shared Volume Group field. If you select an instance different from the default, you must also specify the appropriate shared volume group for that instance in the Add an Oracle Application Server panel.
Application Name	The current Oracle Application name is discovered. If the name is longer than 32 characters, shorten it.

Table 10. Modify an Oracle CFC Application Server fields (continued)	
Field	**Value**
Primary Node	Select a node from a list of predefined cluster nodes. The PowerHA SystemMirror Smart Assist for Oracle discovers the nodes this instance is accessible on and provides those node names. You can remove or exclude nodes from this list.
Takeover Node	Select a node from a list of predefined cluster nodes. The takeover node acquires the Oracle instance in the event of an instance failure on the primary node. You can specify one or more takeover nodes. The system uses the local cluster node and an additional node name provided by you to form the PowerHA SystemMirror cluster topology configuration with two nodes.
Oracle Metadata Repository	Select the Oracle RDBMS to serve as the metadata repository (MR) from the picklist. You may use the same database to contain the metadata repository and the schemas used to hold application data.
Service IP Label	Enter one service IP label to use for both the Oracle listener and the Apache HTTP server. You may either: • Select from the picklist of pre-configured service IP labels that are not configured to a PowerHA SystemMirror resource group. or • Specify a service IP label that is not configured to PowerHA SystemMirror; Smart Assist for Oracle adds that service IP label to the PowerHA SystemMirror configuration and to the resultant PowerHA SystemMirror resource group.
Shared Volume Groups	Select from the picklist the shared volume group(s) that contain the tablespaces, logs, etc., required for the selected Oracle SID/Instance. By default, the picklist provides those volume groups associated with the first Oracle instance (displayed in the Oracle instance field). You must change the volume groups if you selected an Oracle instance different from the default.
Oracle OID and SSO Components	Select from the picklist the OID and SSO components for the Oracle AS infrastructure tier. You may select one or more components.

6. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)**, and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.

7. (Highly recommended) Test your configuration.

Related tasks

Testing the availability of Oracle applications
Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Removing an existing Application server CFC configuration

Removing an Oracle Application Server CFC configuration removes all of the PowerHA SystemMirror resource groups and PowerHA SystemMirror application servers from the PowerHA SystemMirror cluster.

Remember to either stop cluster services or bring the resource group offline before removing the configuration.

To remove an Application Server configured for CFC from the PowerHA SystemMirror configuration:

1. Enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.

    SMIT displays a list of available applications.
3. Select the Oracle Application Server configured for CFC to remove from the picklist.

    SMIT displays: `Are you sure?`
4. Press Enter to remove the Oracle Application Server.

Configuring the Oracle Application server for AFC

In an Oracle Application Server configured for an Active Failover Cluster, each node has its own resource group. PowerHA SystemMirror monitors the resource groups for application server failure and restarts the application server on that node - no movement of resources takes place in an AFC configuration.

PowerHA SystemMirror Smart Assist for Oracle provides the add, change/show and remove options for modifying the PowerHA SystemMirror configuration to work with pre-installed Oracle Application Server configured for AFC installations.

Adding an Oracle Application server configured for AFC
An Oracle Application Server configured for AFC must be installed for PowerHA SystemMirror to discover the installation. PowerHA SystemMirror determines the Oracle Application Server processes that require monitoring and the instances that are accessible within the Oracle directory structure.

To add an Oracle Application Server configured for AFC, specify all nodes that participate in the AFC configuration. Each node must have its own resource group, with application start and stop scripts, and a monitor to determine if the AFC processes have died on the node and require a restart. If a failure occurs, PowerHA SystemMirror restarts the Oracle application server on that node. After the restart count has been exhausted the resource group goes into the error state, the resources do not move to another node in an AFC configuration.

To add a new Oracle Application Server AFC configuration:

1. Set and export the ORACLE_HOME shell variable on the primary node.
2. On the primary node, enter `smit hacmp`
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.

    Note: If the cluster is not configured, PowerHA SystemMirror displays the Two-Node Cluster Configuration Assistant window. By using this tool, you can supply the minimum information that is required to define a cluster, and PowerHA SystemMirror discovers the remainder of the information for you.
4. Select the application from the **Oracle AFC Application Server** from the list of discovered applications and press Enter.

    The Add an Oracle AFC Application server panel appears.
5. Enter field values as follows:

Table 11. Add an Oracle AFC Application fields	
Field	**Value**
Application Name	The name of the Oracle AFC application. Use less than 32 characters for the name.

Table 11. Add an Oracle AFC Application fields (continued)	
Field	**Value**
Participating Node or Communication Paths	Select all cluster node names participating in the Oracle Application Server AFC configuration, or alternatively, one or more communication paths. When you specify a communication path, the node is discovered and added to the cluster configuration.
	You may select a combination of communication paths (previously defined to PowerHA SystemMirror) and pre-existing PowerHA SystemMirror cluster node names:
	• An existing PowerHA SystemMirror cluster node name
	• An IP address
	• A fully qualified domain name
	• A resolvable IP label.

6. Press Enter.

7. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)**, and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.

8. Test your configuration.

Related tasks

Testing the availability of Oracle applications
Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Changing or showing an existing Application server AFC configuration

When you change or show an application configuration using the **Change/Show an Application's PowerHA SystemMirror Configuration SMIT panel**, PowerHA SystemMirror Smart Assist for Oracle adds or changes the appropriate resource group in the PowerHA SystemMirror cluster configuration.

To change or show an existing Oracle Application Server configured for AFC:

1. Set and export the ORACLE_HOME shell variable on the primary node.

2. On the primary node, enter `smit hacmp`

3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

4. Select an Oracle Application Server AFC Configuration to change or show from the picklist.

5. The Modify an Oracle AFC Application Server panel appears. To change the configuration, enter field values as shown in:

Table 12. Modify an Oracle AFC Application Server fields	
Field	**Value**
Application Name	The name of the Oracle AFC application. Use less than 32 characters for the name.

Table 12. Modify an Oracle AFC Application Server fields (continued)	
Field	**Value**
Participating Node or Communication Paths	Select all cluster node names participating in the Oracle Application Server AFC configuration, or alternatively, one or more communication paths. When you specify a communication path, the node is discovered and added to the cluster configuration. You may select a combination of communication paths (previously defined to PowerHA SystemMirror) and pre-existing PowerHA SystemMirror cluster node names: • An existing PowerHA SystemMirror cluster node name • An IP address • A fully qualified domain name • A resolvable IP label.

6. Press Enter.

7. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.

8. Test your configuration.

Related tasks

Testing the availability of Oracle applications
Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Migration support for Oracle database instance

If you update the ORACLE_HOME directory during Oracle database migration, you must update PowerHA SystemMirror configuration.

To update the new Oracle home directory in PowerHA SystemMirror Smart Assist for Oracle, you must first remove an existing configuration and then reconfigure the Oracle database instance. However, in PowerHA SystemMirror 7.2.5, or later, you can update the *${ORACLE_HOME}* environment variable without reconfiguring the Oracle database instance by using the following command:

```
/usr/es/sbin/cluster/sa/oracle/sbin/cl_osa_modifyInstance
        -a <ApplicationID> -H <NewOracleHomeDirectory>
```

The command also collects the Oracle database version internally based on the specified Oracle home directory.

Note: You must run this command only when the cluster services are active on all nodes or are inactive on all nodes.

Removing an existing Application server AFC configuration

Removing an Oracle Application Server AFC configuration removes all of the PowerHA SystemMirror resource groups and PowerHA SystemMirror application servers from the PowerHA SystemMirror cluster.

Remember to either stop cluster services or bring the resource group offline before removing the configuration.

To remove a Application Server configured for AFC from the PowerHA SystemMirror configuration:

1. Enter `smit hacmp`

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications.
3. Select an Oracle Application Server AFC configuration to remove from the picklist.

   SMIT displays: `Are you sure?`
4. Press Enter to remove the Oracle Application Server.

Configuring the Oracle database instance

The PowerHA SystemMirror Smart Assist for Oracle helps you to configure the database instances as CFC components.

You may configure the service IP label (virtual IP), shared volumes, primary and takeover nodes for each defined Oracle database instance. You may also specify more than one Oracle database instance per cluster. However, the participating nodes cannot be the same as those nodes combined in another Oracle database instance.

The Oracle CFC database instance component may be used without an Oracle Application Server, or may be integrated with an Oracle Application Server configured for CFC or AFC.

The Cold Failover Cluster database can use the same machines as the Oracle Application Server configured for Cold Failover Cluster environment. In this scenario, use an extra pair of virtual hostname and virtual IP for the database Cold Failover Cluster.

Adding an Oracle database instance

To add an Oracle database instance, you specify the primary and takeover nodes, a maximum of one node per field (hot-standby), and the Oracle home directory where the OFA directory structure resides.

PowerHA SystemMirror discovers the **spfiles** or **pfiles**. The discovery process determines which Oracle database instances are accessible and displays a list of those instances.

If there are no accessible instances, or no **pfiles** or **spfiles** discovered, then an error displays and the process of adding an instance fails.

The default value for the Oracle home Directory is the value of the environment variable *${ORACLE_HOME}*. This variable must be set and exported before adding or modifying an instance.

To add a new Oracle database instance to the PowerHA SystemMirror configuration:

1. On the primary node, enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. Select **Oracle RDBMS Database** from the list of discovered applications and press Enter.
4. Select the Oracle **RDBMS Instance** to make highly available from the list of discovered Oracle RDBMS Instance/SID and press Enter.

   The instances available in the selector previously mentioned are only those instances that are available via the OFA directory structure in **pfile** or **spfiles** on the primary node specified in the first SMIT screen. The secondary node is not searched.
5. Enter field values as follows:

Table 13. Oracle database instance fields	
Field	**Value**
Application Name	The name of the Oracle RDBMS Instance.
Primary Node	Select a node from the list of predefined PowerHA SystemMirror cluster nodes.

Field	Value
Table 13. Oracle database instance fields (continued)	
Field	**Value**
Takeover Node(s)	Select a node from the list of predefined PowerHA SystemMirror cluster nodes If the node where the Oracle database instance resides is not currently part of the cluster configuration, you must add the node to the cluster so that it appears in this picklist.
Service IP label	Specify the service IP label used by the Oracle listener to communicate with remote clients and other application tiers. If the service IP label does not already exist, the specified IP label must be resolvable. Please provide one of the following for the service IP label: • An existing PowerHA SystemMirror service IP label • An IP address • A resolvable IP label.
Shared Volume Group	Select the shared volume from the list of discovered shared volumes that are accessible between the primary and takeover node. These are the shared volumes required for the database instance to function properly. All file systems that reside on the specified shared volumes will also be made highly available.
Network Name	Specify the network for the service IP alias
Netmask(IPv4)/ Prefix Length(IPv6)	For the configuration of the IP version 4 service interface, enter the network mask for the address. For the configuration of IP version 6 service interface, enter the prefix length for the address. This field is not a required field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

Table 13. Oracle database instance fields (continued)	
Field	**Value**

DB Instance Shutdown Option	• "Allows the user to change Shutdown option for the Oracle Database instance. Choose any one of the available options:
	
	  a. **immediate (default):** Performs an immediate, consistent shutdown of the target database, with the following implications:
	    – Current client SQL statements being processed by the database are allowed to complete.
	    – Uncommitted transactions are rolled back.
	    – All connected users are disconnected.
	
	  b. **abort:**Performs an inconsistent shutdown of the target instance, with the following implications:
	    – All current client SQL statements are immediately terminated.
	    – Uncommitted transactions are not rolled back until next startup.
	    – All connected users are disconnected.
	    – Instance recovery will be performed on the database at next startup.
	
	  c. **normal:** Performs a consistent shutdown of the target database with normal priority, which means:
	    – No new connections are allowed after the statement is issued.
	    – Before shutting down, the database waits for currently connected users to disconnect.
	    – The next startup of the database will not require instance recovery.
	
	  d. **transactional:** Performs a consistent shutdown of the target database while minimizing interruption to clients, with the following implications:
	    – Clients currently conducting transactions are allowed to complete, that is, either commit or terminate before shutdown.
	    – No client can start a new transaction on this instance; any client attempting to start a new transaction is disconnected.
	    – After all transactions have either committed or terminated, any client still connected is disconnected."

6. Press Enter. The PowerHA SystemMirror discovery process executes on the primary node and determines all of the available Oracle database instances using the OFA directory structure to search for **pfile** and **spfiles**. If no **pfile** or **spfiles** are found, an error message is issued and the SMIT session terminates.

7. Press Enter to add the Oracle database instance to the PowerHA SystemMirror configuration.

8. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.

9. Test your configuration.

Related tasks

Testing the availability of Oracle applications

Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Changing or showing an Oracle database instance

The **Change/Show an Oracle RDBMS Instance** SMIT panel for the Oracle database instance allows you to view or change certain parameters of a configured database instance.

These parameters include:

- Service IP label
- Shared Volume Groups.
- DB Instance Shutdown Option

The default value for the Oracle home Directory is the value of the environment variable *$ {ORACLE_HOME}*. This variable must be set and exported before adding or modifying an instance.

To change or show an existing Oracle database instance:

1. On the primary node, enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's Configuration** and press Enter.
3. The **Modify an Oracle RDBMS Instance** panel appears. To change the configuration, enter field values as:

Table 14. Modify an Oracle RDBMS Instance fields	
Field	**Value**
Application Name	The name of the Oracle RDBMS Instance.
Primary Node	Select a node from the list of predefined PowerHA SystemMirror cluster nodes.
Takeover Node(s)	Select a node from the list of predefined PowerHA SystemMirror cluster nodes If the node where the Oracle database instance resides is not currently part of the cluster configuration, you must add the node to the cluster so that it appears in this picklist.
Service IP label	Specify the service IP label used by the Oracle listener to communicate with remote clients and other application tiers. If the service IP label does not already exist, the specified IP label must be resolvable. Please provide one of the following for the service IP label: • An existing PowerHA SystemMirror service IP label • An IP address • A resolvable IP label.
Shared Volume Group	Select the shared volume from the list of discovered shared volumes that are accessible between the primary and takeover node. These are the shared volumes required for the database instance to function properly. All file systems that reside on the specified shared volumes will also be made highly available.
Network Name	Specify the network for the service IP alias

Table 14. Modify an Oracle RDBMS Instance fields (continued)	
Field	**Value**
Netmask(IPv4)/ Prefix Length(IPv6)	For the configuration of the IP version 4 service interface, enter the network mask for the address. For the configuration of IP version 6 service interface, enter the prefix length for the address. This field is not a required field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

4. Select the **Oracle database instance** to change or show from the Oracle RDBMS Instance/SID picklist. Enter field values as follows:

Table 15. Oracle database instance fields	
Field	**Value**
Service IP label	Specify the service IP label used by the Oracle listener to communicate with remote clients and other application tiers. If the service IP label does not already exist, you must go back and add it to the cluster before you can select it here.
Shared Volume Groups	Select the shared volume from the list of discovered shared volumes that are accessible between the primary and takeover node. These are the shared volumes required for the database instance to function properly. All file systems that reside on the specified shared volumes will also be made highly available.

Table 15. Oracle database instance fields (continued)	
Field	**Value**
DB Instance Shutdown Option	• "Allows the user to change Shutdown option for the Oracle Database instance. Choose any one of the available options:
	a. **immediate (default):** Performs an immediate, consistent shutdown of the target database, with the following implications:
	– Current client SQL statements being processed by the database are allowed to complete.
	– Uncommitted transactions are rolled back.
	– All connected users are disconnected.
	b. **abort:** Performs an inconsistent shutdown of the target instance, with the following implications:
	– All current client SQL statements are immediately terminated.
	– Uncommitted transactions are not rolled back until next startup.
	– All connected users are disconnected.
	– Instance recovery will be performed on the database at next startup.
	c. **normal::** Performs a consistent shutdown of the target database with normal priority, which means:
	– No new connections are allowed after the statement is issued.
	– Before shutting down, the database waits for currently connected users to disconnect.
	– The next startup of the database will not require instance recovery.
	d. **transactional:** Performs a consistent shutdown of the target database while minimizing interruption to clients, with the following implications:
	– Clients currently conducting transactions are allowed to complete, that is, either commit or terminate before shutdown.
	– No client can start a new transaction on this instance; any client attempting to start a new transaction is disconnected.
	– After all transactions have either committed or terminated, any client still connected is disconnected."

5. Press Enter to change the Oracle database instance.

6. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.

7. Test your configuration.

Related tasks

Testing the availability of Oracle applications
Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

Removing an Oracle RDBMS instance

You can remove all of the constructed PowerHA SystemMirror components by removing an Oracle database instance from the PowerHA SystemMirror configuration. This includes the resource group, service IP label, application server and application monitor.

Stop cluster services or move the resource group offline before removing the instance.

To remove a Oracle database instance from the PowerHA SystemMirror configuration:

1. Enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications.
3. Select an Oracle database instance to remove from the picklist.

   SMIT displays: `Are you sure?`
4. Press Enter to remove the Oracle RDBMS instance.

PowerHA SystemMirror displays a summary indicating all of the components that were removed from the PowerHA SystemMirror configuration.

Verifying and completing your configuration setup

When you have finished configuring your Oracle database instance, you need to verify and complete your setup.

To confirm your PowerHA SystemMirror Smart Assist for Oracle configuration is valid:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.
3. Start cluster services.

   a) Enter the fastpath `smit cl_admin`

   b) Select **PowerHA SystemMirror** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.
4. Review the cluster configuration from SMIT.
5. Make sure that the cluster is stable and running as expected.

Related information

Verifying and synchronizing a PowerHA SystemMirror cluster

Monitoring a PowerHA SystemMirror cluster

Starting and stopping cluster services

Testing the availability of Oracle applications

Testing the availability of an application after you have it configured is optional but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

To test your PowerHA SystemMirror Smart Assist for Oracle configuration, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Test Your Application for Availability** and press Enter.

   SMIT displays a list of applications.
3. Select the application you want to test and press Enter.

   You are prompted to confirm the action by pressing Enter again.
4. Press Enter to continue with the test of the selected application.

SMIT displays output to the display as the Cluster Test Tool runs.

Changing or showing Oracle PowerHA SystemMirror resource

You can display or change an Oracle application server or Oracle RDBMC instance configured as a PowerHA SystemMirror resource.

1. Enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications.
3. Select the Oracle Application Server or Oracle RDBMS instance to change or show and press Enter.
4. Make the changes as needed in the field(s) for the selected application and press Enter.

Changing or showing the PowerHA SystemMirror resources associated with your application

You may want to add resources such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, create separate resource groups for each set.

To change or show the resources associated with your application, complete the following steps:

1. Enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications.
3. Select the application and press Enter.

   SMIT displays the **Change/Show Resources Associated with your Application** for the application's resource group.
4. Make the changes as needed and press Enter.

Related information
Managing resource groups in a cluster

Managing your applications

If the Smart Assist used to make your application highly available has unique configuration tasks, you can do them from the **Make Applications Highly Available** SMIT menu.

Note: This menu currently has no options.

To manage the PowerHA SystemMirror Oracle Application Server or Oracle RDBMS instance:

1. Enter `smit hacmp`
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Manage Your Applications** and press Enter.

   SMIT displays a list of available applications.
3. Select the Oracle Application Server or Oracle RDBMS instance and press Enter.

Troubleshooting the installation

Use this topic to find troubleshooting information directly related to the PowerHA SystemMirror Smart Assist for Oracle.

Related information

Troubleshooting guide

Using the log files

The hacmp.out file is the most helpful log file (once the cluster is running). The hacmp.out file captures information on the activity and location of resource groups after cluster events. For instance, the hacmp.out file captures details of resource group parallel processing that other logs (such as the cluster history log) cannot report. The event summaries included in this log make it easier to quickly see the events that have recently occurred in the cluster.

The following list describes the log files into which the PowerHA SystemMirror software writes messages and the types of cluster messages they contain. The list also provides recommendations for using the different log files. Note that the default log directories are listed here; you have the option of redirecting log files to a chosen directory.

Table 16. PowerHA SystemMirror log files	
Log files	**Description**
/var/hacmp/log/ oraclesa.log	The PowerHA SystemMirror Smart Assist for Oracle software writes the messages it generates to the **oraclesa.log**.
	The **oraclesa.log** on the primary node contains **results of internal commands and variables used by the Smart Assist for Oracle. The oraclesa.log** on the secondary node contains only basic checks and information about that node.
	This file contains information you can use to remove each part of the configuration individually.
/var/hacmp/clverify/ clverify.log	The **/var/hacmp/clverify/clverify.log** file contains the verbose messages output by the cluster verification. The messages indicate the node(s), devices, command, and such, in which any verification error occurred.
/var/hacmp/log/ cspoc.log	Contains time-stamped, formatted messages generated by PowerHA SystemMirror C-SPOC commands. The **/tmp/cspoc.log** file resides on the node that invokes the C-SPOC command.
	Recommended Use: Use the C-SPOC log file when tracing a C-SPOC command's execution on cluster nodes.
/var/hacmp/log/ hacmp.out	Contains time-stamped, formatted messages generated by PowerHA SystemMirror scripts on the current day.
	In verbose mode (recommended), this log file contains a line-by-line record of every command executed by scripts, including the values of all arguments to each command. An event summary of each high-level event is included at the end of each event's details.

Related information

Verifying and synchronizing a PowerHA SysteMirror cluster

Understanding the hacmp.out log file

Miscellaneous problems

This topic collection helps you understand, analyze and resolve miscellaneous problems that you are having with your PowerHA SystemMirror Smart Assist for Oracle installation.

Unable to start the Oracle interactive installer
This topic explains what to do if you are unable to start the Oracle interactive installer.

Problem

The Smart Assist for Oracle fails to start and displays the following error message:

Unable to start an interactive install session because of the following error: Can't connect to X11 window server using '9.49.162.47:0.0' as the value of the DISPLAY variable. The DISPLAY environment variable should be set to <hostname or IP address>:<screen>, where the <screen> is usually '0.0'.

Solution

Depending on the UNIX shell, use one of the following commands as examples to set the DISPLAY environment variable:

```
For csh: % setenv DISPLAY 192.168.1.128:0.0

For sh, ksh and bash:$ DISPLAY=192.168.1.128:0.0; export DISPLAY
```

Use the following command to see the shell being used:

```
echo $SHELL
```

Use the following command to view the current DISPLAY environment variable setting:

```
echo $DISPLAY
```

Make sure that client users are authorized to connect to the X Server.

To enable client users to access the X Server, open an **xterm**, **dtterm**, or **xconsole** session as the user that started the session and type the following command:

```
% xhost Name
 # Where Name defines the host name to be added to the X Server
access list.
```

To test that the DISPLAY environment variable is set correctly, run a X11-based program that comes with the native operating system, such as **xclock**:

```
% <full path to xclock... see below>
```

If you are not able to run **xclock** successfully, please refer to your PC-X Server or OS vendor for further assistance.

Typical path for **xclock**: **/usr/bin/X11/xclock**

Oracle database shared memory clean-up

The shared memory that is associated with the Oracle database instance is cleaned up before you start the Oracle database. The Oracle resource group that is created by using the PowerHA SystemMirror Smart Assist for Oracle database is made available and the Oracle database instance is started. The clean-up operation is performed only if the shared memory segments associated with Oracle database exist and if Oracle processes are not running before starting the Oracle database.

Smart Assist for WebSphere

This information describes how to use the PowerHA SystemMirror Smart Assist for WebSphere tool to support WebSphere Application Server, WebSphere Deployment Manager, WebSphere Recovery Log, IBM HTTP Server, and Tivoli Directory Server.

Getting started

PowerHA SystemMirror Smart Assist for WebSphere extends an existing PowerHA SystemMirror configuration to include monitoring and recovery support for the components in a Base configuration of WebSphere.

These components include:

- WebSphere Application Server
- WebSphere Deployment Manager
- WebSphere Transaction Log
- IBM HTTP Server
- Tivoli Directory Server

Keeping applications highly available

PowerHA SystemMirror Smart Assist for WebSphere extends an existing PowerHA SystemMirror configuration, enabling you to increase the availability of WebSphere-related applications by eliminating their inherent single points of failure.

A *single point of failure* exists when a critical function is provided by a single component. If that component fails, the application dependent on that component becomes unavailable.

PowerHA SystemMirror Smart Assist for WebSphere includes monitoring and recovery support for the components described within this guide. You should be familiar with WebSphere, Tivoli, and other products external to PowerHA SystemMirror; for information about these products, refer to their documentation.

Note: Some terms have different meanings, depending on the product. For example, a *PowerHA SystemMirror node* (also referred to as a *cluster node*) is the system that is running an application protected by PowerHA SystemMirror. A *WebSphere node* is a group of application servers on a system.

To protect applications in a WebSphere environment, PowerHA SystemMirror Smart Assist for WebSphere requires each application that you want to make highly available to run on both:

- A *primary* cluster node. The primary cluster node is the node from which the application runs unless a condition forces a fallover.
- A *takeover* cluster node. The takeover cluster node is the node from which the application runs after a fallover from the primary cluster node.

  In a configuration created by PowerHA SystemMirror Smart Assist for WebSphere, typically each application has its own takeover cluster node.

If an application becomes unavailable, PowerHA SystemMirror tries to restart the application on the primary cluster node three times. In general, if the application does not start after three attempts, PowerHA SystemMirror starts the application on the takeover node to continue service.

Available fallover configurations

PowerHA SystemMirror Smart Assist for WebSphere resource components support several high availability fallover configurations.

These configurations include:

- **Hot standby**.

In a hot standby configuration, one node performs the work while the other node waits as a fallover location.

- **Mutual takeover**.

  In a mutual takeover configuration, each node runs an application while also providing a fallover location for the application of the other node.

- **Multiple nodes**.

  Configuration of three or more nodes in which one node provides a fallover location for the applications running on the other nodes in the cluster.

For a comparison of the different fallover configurations available for each application supported by PowerHA SystemMirror Smart Assist for WebSphere, see the following table:

Different fallover configurations

Component to Keep Highly Available	Hot Standby	Mutual Takeover	Multiple Node
IBM HTTP Server	X	X	X
WebSphere Application Server Node	X	X	X
WebSphere Application Server Deployment Manager	X	X	X
Tivoli Directory Services	X		
WebSphere Application Server Cluster Transaction Log		X	

Configuring PowerHA SystemMirror with PowerHA SystemMirror Smart Assist for WebSphere

PowerHA SystemMirror Smart Assist for WebSphere sets up your PowerHA SystemMirror resource configuration for you. It gathers information from the application configuration, the system configuration, and from you, and creates a PowerHA SystemMirror resource group to protect a WebSphere application.

A PowerHA SystemMirror resource group contains a set of resources, such as cluster nodes, volume groups, and IP labels (used by server applications to communicate with clients) that PowerHA SystemMirror manages as a unit.

Note: The term *application server* can have different meanings. A *WebSphere Application Server* makes Web-based applications available to users. A *PowerHA SystemMirror application server* identifies the application start and stop scripts that are used in conjunction with PowerHA SystemMirror.

In addition to creating a resource group for each application to be made highly available, PowerHA SystemMirror Smart Assist for WebSphere performs the following tasks:

- Creates a PowerHA SystemMirror application server for the application
- Ensures the application has a service IP label that can be transferred to another system, thus keeping the application highly available
- Creates PowerHA SystemMirror application monitors to detect failures in application processes
- Provides start and stop scripts for the application (that is, application servers)
- Stores the generated PowerHA SystemMirror configuration in the PowerHA SystemMirror configuration database (ODM).

When PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror resource configuration, it updates the PowerHA SystemMirror configuration and changes the IP address (for IHS, WAS, and Deployment Manager), or the Transaction Log directory (for the WebSphere cluster).

Sample configuration

In a Network Deployment configuration, you install the Deployment Manager from one node (for example, **regaa10**) onto the shared disk. If the application becomes unavailable, PowerHA SystemMirror starts the application on the takeover node (for example, **regaa11**) to continue service.

The following figure shows a Network Deployment configuration with the WebSphere Deployment Manager residing on a shared disk:
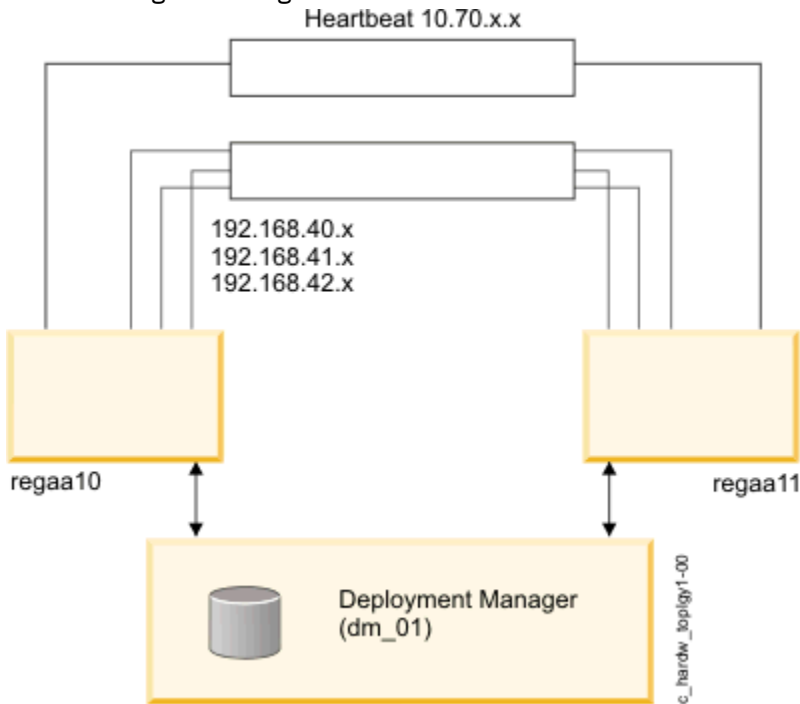

Figure 2. Network Deployment configuration with the WebSphere Deployment Manager residing on a shared disk

The following figure shows the highly available Transaction Log on shared storage. WebSphere handles fallover of the application servers, while PowerHA SystemMirror handles the Transaction Log fallover:
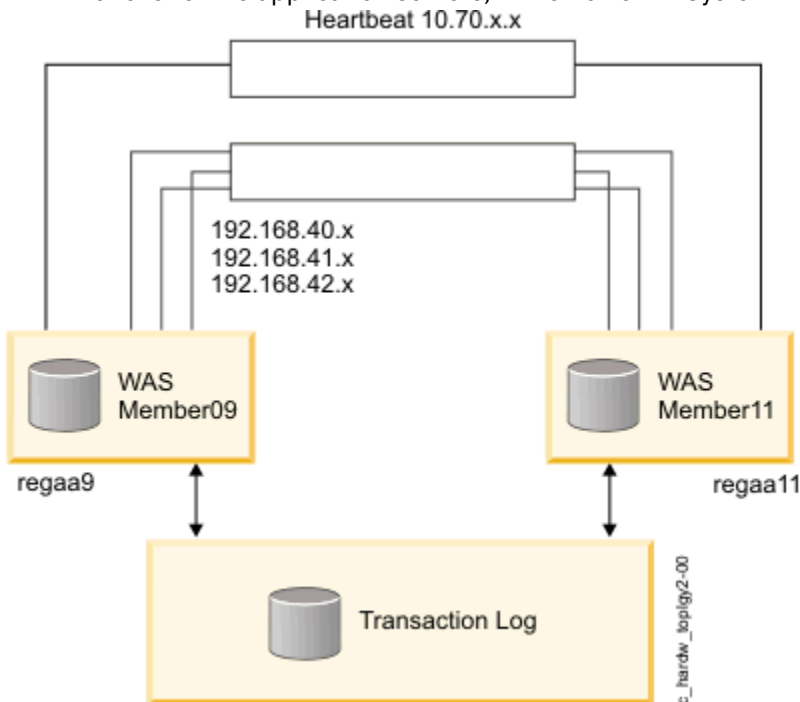

Figure 3. Highly available Transaction Log on shared storage

Resource group behavior

PowerHA SystemMirror Smart Assist for WebSphere creates a separate PowerHA SystemMirror resource group for each application. The resource group includes a list of all the cluster nodes that can control the resource group; that is, the list of nodes on which the application can run, with a priority assigned to each node.

The policies for the resource group are set to:

- Startup: online on home node
- Fallover: fallover to next priority node in the list
- Fallback: Never Fallback.

  This means that the application:
- Runs on the primary node
- Falls over to the takeover node if the primary node is unavailable
- Falls back to higher priority node in the list only when manually triggered.

Each resource group includes:

- PowerHA SystemMirror application nodes

  Note: Each resource group contains a primary node and one or more takeover nodes even though more than one application may use the same takeover node.
- The service IP addresses for a specified application
- Volume groups and file systems associated with a specified application available to each node in the resource group
- Application servers that contain the application start and stop scripts (created by PowerHA SystemMirror Smart Assist for WebSphere)
- Application monitors (created by PowerHA SystemMirror Smart Assist for WebSphere).

Related concepts
PowerHA SystemMirror configuration summary
These topics contain detailed configuration information about several different PowerHA SystemMirror Smart Assist for WebSphere resources and default setting of these resources.

Planning your configuration

These topics describe the tasks to complete and the information to gather before running the PowerHA SystemMirror Smart Assist for WebSphere application.

Supported software

The necessary software must be installed and configured before using PowerHA SystemMirror Smart Assist for WebSphere.

To view what versions of WebSphere are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Limitations

PowerHA SystemMirror Smart Assist for WebSphere has some limitations.

These limitations include:

- It supports only fallover of WebSphere Application Server in a Base installation; if you have a Network Deployment (ND) installation, use the high availability features provided by ND, and use PowerHA SystemMirror Smart Assist for WebSphere to make the transaction log highly available. Express and Enterprise installations are *not* supported.

- PowerHA SystemMirror does *not* support fallover management of application servers on *federated nodes* (nodes that are part of a Network Deployment environment).
- PowerHA SystemMirror supports only WebSphere installations that have one profile per PowerHA SystemMirror node. The use of multiple profiles is *not* supported. (Multiple installations of WebSphere on unique volume groups is supported.)
- PowerHA SystemMirror does *not* support multiple installations of IBM HTTP Server on the same node.
- PowerHA SystemMirror Smart Assist for WebSphere requires that WAS operates in a PowerHA SystemMirror environment where nodes share resources used by WAS (for example, volume groups and file systems).
- PowerHA SystemMirror Smart Assist for WebSphere does *not* support PowerHA SystemMirror geographic sites.

Planning a PowerHA SystemMirror cluster

Before planning your PowerHA SystemMirror implementation for a WebSphere environment, you should understand both PowerHA SystemMirror and the application to be made highly available.

PowerHA SystemMirror Smart Assist for WebSphere automates many of the PowerHA SystemMirror cluster configuration activities; however, you need to plan and configure your basic environment before you run PowerHA SystemMirror Smart Assist for WebSphere. While planning your environment, record planning information as noted.

Step 1: Planning for applications
The first step is to plan for applications.

If you want to run WebSphere on more cluster nodes than you can list in the following table, print an additional copy of this form and use it to record information.

In the following table, list information about the WebSphere Application Server, IBM HTTP Server, or Deployment Manager that applies to your environment.

Table 17. Information about your server

Application	Name of Primary Node	Service IP Label on Primary Node	Name of Standby Cluster Node
WebSphere Application Server			
WebSphere Application Server			
IBM HTTP Server			
IBM HTTP Server			
Tivoli Directory Server			
WebSphere Deployment Manager			

For your WebSphere Clusters, list the information specific to your environment in the following table.

Table 18. Information about your environment

Name of WebSphere Cluster	Service IP Label	Name of Shared Volume Group	Name of Transaction Log Directory

Step 2: Planning the cluster topology

After you have planned for applications, plan the cluster topology.

Complete the planning activities described in Planning cluster network connectivity in the *Planning Guide*.

Related information

Planning cluster network connectivity

Step 3: Planning shared volume groups

For PowerHA SystemMirror Smart Assist for WebSphere to manage the availability of the WebSphere components, they must be installed on a shared volume group.

PowerHA SystemMirror Smart Assist for WebSphere does not move WebSphere components from their original installation location. An exception to this is Transaction Log Recovery. For the Transaction Log Recovery, WebSphere manages the availability of the WAS application servers and PowerHA SystemMirror Smart Assist for WebSphere moves only the log files to a shared volume group.

For each WebSphere Deployment Manager, IBM HTTP Server, and WebSphere Application Server, plan a shared volume group (a concurrent access volume group) to contain the application files.

For each Tivoli Directory Server, plan which volumes groups are to store data files.

The primary and takeover cluster nodes for an application require access to the same volume group and file system that stores the data for the application. This way, if the application falls over from the primary cluster node to the takeover cluster node, the application continues to run from the shared volume group. For PowerHA SystemMirror Smart Assist for WebSphere to collect configuration information for an application, any cluster node that might run the application must have access to the volume groups and file systems that store the application files.

PowerHA SystemMirror requires that a volume group be configured as a resource in only one resource group. This means that different applications require different volume groups. Volume groups must be varied on and file systems mounted before you run the PowerHA SystemMirror Smart Assist for WebSphere application.

List the volume groups that you plan to use with each application. Note that the list of volume groups is only for your planning. During the configuration process, PowerHA SystemMirror Smart Assist for WebSphere automatically discovers information about the volume groups available.

Table 19. Volume groups for each application	
Application	**Shared Volume Group**
WebSphere Application Server	
WebSphere Application Server	
IBM HTTP Server	
IBM HTTP Server	
Tivoli Directory Server	
WebSphere Deployment Manager	
Transaction Log Directory	

Related information

Planning shared disk and tape devices

Planning shared LVM components

Step 4: Installing the PowerHA SystemMirror software

Install the PowerHA SystemMirror software on each primary cluster node and all takeover cluster nodes.

Install the following:

- Deployment Manager
- Tivoli Directory Server
- IBM HTTP Server
- WebSphere Application Server.

Related information
Installation guide

Step 5: Planning security

Only an administrator with root privileges can run the configuration utilities included in PowerHA SystemMirror Smart Assist for WebSphere. An **/.rhosts** file is needed when using PowerHA SystemMirror Smart Assist for WebSphere to configure the Tivoli Directory Server, but you can remove the file after the configuration is complete.

PowerHA SystemMirror, WebSphere, and Tivoli Directory Server each have their own security mechanism. Those security mechanisms remain in place; PowerHA SystemMirror Smart Assist for WebSphere does not make any changes.

Related information
Managing user and groups
Managing cluster security

Planning for application installation and configuration

Before you run the PowerHA SystemMirror Smart Assist for WebSphere, install and configure the applications it supports.

Because PowerHA SystemMirror Smart Assist for WebSphere automates resource configuration for supported applications, it has configuration requirements for each application.

Note: If you are using DB2®, determine if DB2 requires any specific planning before continuing with this guide.

Planning for WebSphere Deployment Manager

When you install the WebSphere Deployment Manager, ensure that the application executable files reside on a shared volume group. From the primary cluster node, mount all of the file systems in the shared volume groups.

The WebSphere Deployment Manager uses a WebSphere cell to control all of the WebSphere components. PowerHA SystemMirror Smart Assist for WebSphere enables the following availability behavior:

- Monitors the Deployment Manager
- If the Deployment Manager becomes unavailable, attempts to restart the application three times on the primary node. If it fails to restart the application, starts the Deployment Manager on the takeover node that you have defined

Note: When the primary node becomes available again, manual intervention is required for fallback.

Planning for transaction log

In a Network Deployment (ND) installation, you can use the high availability features provided by ND, and use PowerHA SystemMirror Smart Assist for WebSphere to make the Transaction Log highly available.

WebSphere handles fallover of the application servers, and PowerHA SystemMirror handles the Transaction Log fallover.

It is important to understand why making Transaction Logs highly available is useful. Although WebSphere alone has built-in performance balancing and high availability capabilities, it does not allow a situation where an application instance is "locked" permanently.

Smart Assist for WebSphere makes Transaction Logs highly available with the use of NFS cross-mounts. This allows each member of the WebSphere Cluster access to the other members' transaction logs. If a WebSphere Cluster member should, another member can roll back any unfinished transactions by accessing the failed member's transaction log.

PowerHA SystemMirror Smart Assist for WebSphere does the following to keep the Transaction Log highly available:

- Stops the WebSphere application servers
- Creates the Transaction Log path on the shared volume group, if necessary
- Copies the Transaction Log files to the subdirectory in the path that you specify
- Creates NFS cross mounts so that all nodes can access all Transaction Logs at the new location
- Updates the PowerHA SystemMirror Smart Assist for WebSphere to access the Transaction Log at the new location.

The following figure shows the highly available Transaction Log on shared storage.
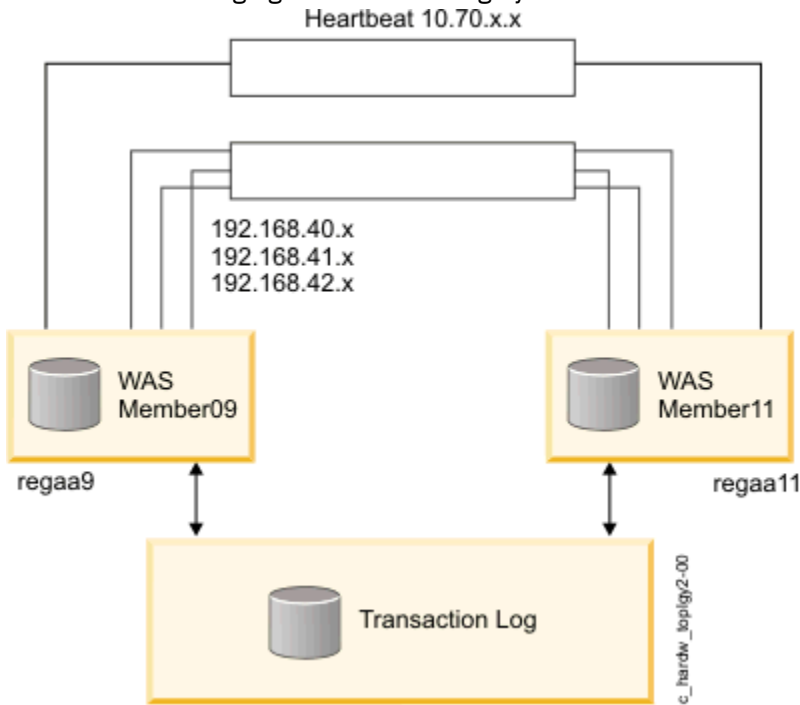
Figure 4. Highly available Transaction Log on shared storage

When PowerHA SystemMirror Smart Assist for WebSphere adds a Transaction Log as a resource to its resource group, it also automatically adds NFS cross-mounts that are necessary to make Transaction Logs highly available in the cluster. For this operation to succeed, prior to integrating Transaction Logs into a PowerHA SystemMirror cluster, add persistent IP labels to the nodes.

Related information

Administration guide

Planning for WebSphere Application Server
PowerHA SystemMirror Smart Assist for WebSphere supports several different installation and configurations.

PowerHA SystemMirror Smart Assist for WebSphere supports:

- One WebSphere cell
- One WebSphere node (which represents a group of application servers) on a system
- Multiple installations of WebSphere Application Server on a unique shared volume group per PowerHA SystemMirror node.

Planning for WebSphere Application Server installation and configuration

PowerHA SystemMirror Smart Assist for WebSphere enables the following availability behavior:

- Monitors the WebSphere Application Server
- If the WebSphere Application Server becomes unavailable, attempts to restart the application three times on the primary node. If it fails to restart the application, starts the WebSphere Application Server on the takeover node you have defined

Note: When the primary node becomes available again, manual intervention is required for fallback.

PowerHA SystemMirror Smart Assist for WebSphere does not modify the **plugin_cfg.xml** file, which is automatically generated by WebSphere and used by the IBM HTTP Server (IHS) to monitor WebSphere Application Server.

Note: You may be required to generate a new **plugin_cfg.xml** file and propagate it to the location of the IBM HTTP Servers.

The first time you create a WebSphere Application Server resource group, it is created in a *hot standby* cluster configuration. If you want a different configuration, such as *mutual takeover* or *multiple nodes cluster*, plan to create additional WebSphere Application Server resources and specify the applicable primary and takeover nodes.

Installation notes

When installing the WebSphere Application Server, store executable files on a shared volume group. From the primary cluster node, mount all of the file systems in the shared volume groups.

For information about these files, see the WebSphere Application Server Network Deployment *Getting Started* guide from the following URL:

http://www.ibm.com/software/webservers/appserv/was/library/

Planning for IBM HTTP Server configuration

When you are installing the IBM HTTP Server, store the IHS executable files on a shared volume group.

If you created a plugin to enable IHS to communicate with an WebSphere Application Server, store it in the same volume group as the IBM HTTP Server.

The IBM HTTP Server requires its own resource group, since it is typical, but not required, for an HTTP server to be on a different node than the application servers to which it sends requests. PowerHA SystemMirror Smart Assist for WebSphere enables the following availability behavior:

- Monitors the IBM HTTP Server
- If the IBM HTTP Server becomes unavailable, attempts to restart the application three times on the primary node, and if it fails to restart the application, starts the IBM HTTP Server on the takeover node you have defined

Note: When the primary node becomes available again, manual intervention is required for fallback.

PowerHA SystemMirror Smart Assist for WebSphere does not modify the **plugin_cfg.xml** file, which is automatically generated by WebSphere and used by the IBM HTTP Server to monitor WebSphere Application Server.

Note: You may be required to generate a new **plugin_cfg.xml** file and propagate it to the location of the IBM HTTP Servers.

Planning for the Takeover Cluster Node (for IHS)

Only for IHS, make sure that the **gskta.rte** fileset is installed on the takeover cluster node(s). You can run the following AIX command to check if this fileset is installed:

```
lslpp -l gskta.rte
```

Note: This fileset is included in the IBM HTTP Server installation filesets. Also, when you add an IBM HTTP Server using the WebSphere Smart Assist, the list generated for takeover nodes will only show nodes that have this fileset installed.

Related information

WebSphere Application Server

Planning for Tivoli Directory Server

PowerHA SystemMirror Smart Assist for WebSphere enables several availability behaviors for Tivoli Directory Server

These behaviors include:

- Monitors the Tivoli Directory Server
- If the Tivoli Directory Server becomes unavailable, attempts to restart the application three times on the primary node, and if it fails to restart the application, starts the Tivoli Directory Server on the takeover node you have defined

Note: When the primary node becomes available again, manual intervention is required for fallback.

PowerHA SystemMirror Smart Assist for WebSphere supports a Tivoli Directory Server that has:

- A single directory server
- DB2 as the data repository.

Planning Tivoli Directory Server installation and configuration
During installation and configuration for Tivoli Directory Server on the primary cluster node and on the takeover cluster node, you need to verify several aspects of the installation.

Make sure that:

- The application software files are stored on each local node and that data files, stored in a DB2 database instance, are stored on a shared volume group.
- The Tivoli Directory Server retains the default user and associated home directory created by the Tivoli Directory Server installation:

  – The user is **ldapadm**.

    Both the primary and the takeover cluster nodes require the same user access configuration.
  – The home directory for **ldapadm** is a shared file system.

    This is the directory where the DB2 data files for Tivoli Directory Server are stored.

PowerHA SystemMirror Smart Assist for WebSphere requires these default settings.

- Change the port number for the embedded WebSphere Application Server from the default setting of 9090 to a value that does not conflict with other port numbers on the system.
- On the takeover cluster node, add the following entries to the **/etc/services** file. These entries are added to the **/etc/services** file on the primary cluster node during the Tivoli Directory Server instance creation:

  – `ldapdb2svc 3700/tcp`
  – `ldapdb2svci 3701/tcp`

For information on making configuration changes, see the *IBM Directory Server Version 5.1: Administration Guide* at the following URL:

http://publib.boulder.ibm.com/tividd/td/IBMDS/IDSadmin51/en_US/PDF/admin_gd.pdf

Adding the Service IP label as a managed service
From the IBM Directory Server Web Administration Tool on the primary cluster node, add a managed console server with the service IP address as the hostname that Tivoli Directory Server uses. Repeat this process on the takeover cluster node.

Any time you make a change to the configuration of the managed console server on one cluster node, make the same configuration change on the other cluster node. The IBM Directory Server Web Administration Tool runs independently on each node, and does *not* synchronize information between the two nodes.

For more information about adding a managed server, see the *IBM Directory Server Version 5.1: Administration Guide* at the following URL:

http://publib.boulder.ibm.com/tividd/td/IBMDS/IDSadmin51/en_US/PDF/admin_gd.pdf

Turning off automatic start of the Tivoli Directory Server and the DB2 Fault Monitor Coordinator
The PowerHA SystemMirror configuration established by PowerHA SystemMirror Smart Assist for WebSphere requires that the system does *not* automatically start the DB2 Fault Monitor Coordinator, or automatically start the Tivoli Directory Server on server reboot.

After installing Tivoli Directory Server, the **/etc/inittab** file contains the following entries:

```
fmc:2:respawn:/usr/opt/db2_08_01/bin/db2fmcd #DB2 Fault Monitor Coordinator
ibmdir:2:once:/usr/ldap/sbin/rc.ibmdir > /dev/null 2>&1
```

To turn off automatic restart of the DB2 Fault Monitor Coordinator and the Tivoli Directory Server, enter the following commands:

```
# chitab "fmc:2:off:/usr/opt/db2_08_01/bin/db2fmcd #DB2 Fault Monitor Coordinator"
# chitab "ibmdir:2:off:/usr/ldap/sbin/rc.ibmdir > /dev/null 2>&1"
```

The **/etc/inittab** file should now contain the following entries when the DB2 Fault Monitor Coordinator and the Tivoli Directory Server are *not set* to automatically start:

```
fmc:2:off:/usr/opt/db2_08_01/bin/db2fmcd #DB2 Fault Monitor Coordinator
ibmdir:2:off:/usr/ldap/sbin/rc.ibmdir > /dev/null 2>&1
```

Recording information for PowerHA SystemMirror Smart Assist for WebSphere
If you plan to use PowerHA SystemMirror to make a Tivoli Directory Server highly available, collect information before you use PowerHA SystemMirror Smart Assist for WebSphere

Use the following table to record the information that you need:

Table 20. PowerHA SystemMirror Smart Assist for WebSphere information	
Environment information	**Value**
The Tivoli Directory Server password.	
The port that the Tivoli Directory Server uses to communicate over the network.	
The DB2 instance that is used to start, stop, and monitor the DB2 database used by Tivoli Directory Server.	
(*Optional*) The name of the WebSphere node which is a group of application servers on a system. You need this information only if there is more than one node specified in the *WAS_HOME* **/config/cells/** *WAS_CELL* **/nodes** directory.	

Configuring cluster resources

You can use PowerHA SystemMirror Smart Assist for WebSphere to discover and configure WebSphere support for your PowerHA SystemMirror cluster.

Prerequisites

Before using PowerHA SystemMirror Smart Assist for WebSphere to configure PowerHA SystemMirror for each application:

1. Ensure that the **/usr** directory has at least 5 MB of free space; 20 MB of free space is recommended.
2. If you have an existing PowerHA SystemMirror configuration, synchronize your cluster and take a snapshot.
3. If you are using PowerHA SystemMirror Smart Assist for WebSphere to make Tivoli Directory Server highly available, create an **/.rhosts** file on each cluster node.

   You can delete this file after your configuration is in place.
4. Ensure that volume groups are varied on and file systems mounted where the WebSphere components are installed.
5. If you are using PowerHA SystemMirror Smart Assist for WebSphere to make WebSphere Transaction Log highly available (by including it as a resource in the resource group), ensure that you have configured persistent IP labels on the nodes in the cluster so that PowerHA SystemMirror Smart Assist for WebSphere is able to correctly perform the necessary NFS cross mounting operations.

Related information
Verifying and synchronizing a PowerHA SystemMirror cluster
Saving and restoring cluster configurations

Configuring the PowerHA SystemMirror cluster and nodes

Configure the cluster and nodes before using the PowerHA SystemMirror Smart Assist for WebSphere to discover WebSphere components and configure them for you.

To configure the PowerHA SystemMirror cluster and nodes, complete the following steps:

1. From the control node, enter `smit hacmp`.
2. In SMIT, select **Custom Cluster Configuration** > **Cluster Nodes and Networks** > **Initial Cluster Setup (Custom)** and press Enter.

Note: If you skip this step and use the SMIT path **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, you will be directed back to this step as part of the process.

Discovering and configuring WebSphere components

SmartAssist uses the SMIT interface to discover the applications and prompt you for the information it needs to configure the application.

To discover and configure WebSphere applications, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of the applications, which are installed on the cluster.
3. Select the **WebSphere Smart Assistant** application.

   SMIT displays a list of possible configurations as follows:

   - WebSphere Application Server
   - WebSphere Cluster Transaction Log recovery

- IBM HTTP Server
- Deployment Manager
- Tivoli Directory Services

4. Select one of the components from the list and press Enter. SMIT displays a list of individual component installations.

PowerHA SystemMirror Smart Assist for WebSphere discovers the component and will not allow configuration when any of the following are true:

- WebSphere component is not installed on a shared volume group
- WAS applications for WAS Node are located on different volume groups
- WebSphere cluster is found and not all of its nodes are part of the PowerHA SystemMirror cluster
- WAS application is already managed in a ND (Network Deployment) environment
- **gskta.rte** fileset is not installed (this means the node cannot serve as a fallover node for IBM HTTP Server)

Adding a WebSphere Application Server resource

You can add a WebSphere Application Server resource to PowerHA SystemMirror.

To add a WebSphere Application Server resource to PowerHA SystemMirror, complete the following steps:

1. Verify that the volume group that stores the data for the WebSphere Application Server is varied on and the associated file systems mounted.
2. From the command line, enter `smit hacmp`.
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** list. SMIT displays a list of discovered applications.
4. Select **WebSphere Smart Assistant**, and press Enter.

   SMIT displays a list of discovered WebSphere components.
5. Select **WebSphere Application Server**, and press Enter.

   SMIT displays a list of available WebSphere Application Servers by WebSphere Cell and Node combinations.
6. Select the WebSphere Application Server that you want to add to your PowerHA SystemMirror cluster and press Enter. PowerHA SystemMirror displays the Add a WebSphere Application Server Node to the Cluster panel.
7. Enter field values as follows:

Table 21. Add a WebSphere Application Server Node to the Cluster fields

Field	Value
Application Name	A symbolic name for the collection of PowerHA SystemMirror resources that represent the WebSphere component.
WebSphere Application Server Cell	The name of the WebSphere Application Server cell. This field cannot be modified.
WebSphere Node	The name of the WebSphere node. This field cannot be modified.
Application Servers	A list of the application servers that run on the node. This field cannot be modified.
Primary Node	The name of the node where the WebSphere Application Server is installed. This field cannot be modified.
Takeover Node(s)	Enter (or select from the pick list) the name of one or more cluster nodes to which the application may fall over.

Table 21. Add a WebSphere Application Server Node to the Cluster fields (continued)	
Field	**Value**
Service Interface	This field defaults to the HTTP Transports Hosts of the application server and uses IPAT via IP Aliasing.

8. Press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

   PowerHA SystemMirror Smart Assist for WebSphere stops the WebSphere Application Server because it is now managed by PowerHA SystemMirror. It also modifies the IP address on which the WebSphere Application Server listens.

   Note: From this point onwards, do not stop or start the WebSphere Application Server manually. Use the SMIT panels for these tasks.

9. If you want a mutual takeover or multiple nodes cluster configuration for your WebSphere Application Server, create additional WebSphere Application Server resources.

   Return to previous SMIT panels to perform other configuration tasks.

Adding a Deployment Manager PowerHA SystemMirror resource

The first time you add a Deployment Manager resource, it is created in a hot standby cluster configuration. If you want a different configuration, such as mutual takeover or multiple nodes cluster, create additional Deployment Manager resources and specify the applicable primary nodes, takeover nodes, and service interfaces.

To add a deployment manager PowerHA SystemMirror resource, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror** > **Configure a Deployment Manager Resource** and press Enter.

   SMIT displays a list of discovered applications.
3. Select **WebSphere Smart Assistant**, and press Enter.

   SMIT displays a list of discovered WebSphere components.
4. Select **WebSphere Deployment Manager** and press Enter.

   SMIT displays a list of available WebSphere Deployment Managers.
5. Select a deployment manager from the list, and press Enter.

   PowerHA SystemMirror displays the Add a WebSphere Deployment Manager to the Cluster panel.
6. Enter field values as follows:

Table 22. Add a WebSphere Deployment Manager to the Cluster fields	
Field	**Value**
Application Name	A symbolic name for the collection of PowerHA SystemMirror resources that represent the WebSphere component.
WebSphere Application Server Cell	The name of the WebSphere Cell that you want to make highly available. This field cannot be modified.
Deployment Manager Name	The name of the server that manages the operation of a WebSphere Cell. This field cannot be modified.
Primary Node	The name of the node running the WebSphere Deployment Manager. This field cannot be modified.
Takeover Node(s)	Enter (or select from the pick list) the name of one or more nodes to which the application may fall over.

Table 22. Add a WebSphere Deployment Manager to the Cluster fields (continued)	
Field	**Value**
Service Interface	Enter (or select from the pick list) the value the Deployment Manager should use for listening on the service port. IPAT via IP Aliasing is used. The **clstat** utility displays the TCP/IP ports on which Smart Assist to WebSphere listens for the Deployment Manager.

7. Press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

   PowerHA SystemMirror Smart Assist for WebSphere stops the Deployment Manager because it is now managed by PowerHA SystemMirror. It also modifies the IP address on which the WebSphere Deployment Manager listens.

   Note: From this point onward, do not stop or start the Deployment Manager manually. Use the SMIT panels for these tasks.

8. Optional: If you want a mutual takeover or multiple node configuration for your Deployment Manager, create additional Deployment Manager resources.

9. Return to previous SMIT panels to perform other configuration tasks.

Adding an IBM HTTP Server PowerHA SystemMirror resource

The first time you add an IBM HTTP Server resource, it is created in a *hot standby* cluster configuration. If you want a different configuration, such as *mutual takeover* or *multiple node*, create additional IBM HTTP Server resources and specify the applicable primary nodes, takeover nodes, and service interfaces.

To add an IBM HTTP server resource to your PowerHA SystemMirror cluster, complete the following steps:

1. From the command line, enter smit hacmp.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror** > **Configure an IBM HTTP Server** and press Enter.

   SMIT displays a list of discovered applications.

3. Select WebSphere Smart Assistant and press Enter. SMIT displays a list of discovered WebSphere components.

4. Select IBM HTTP Server and press Enter.

   SMIT displays a list of available IBM HTTP Servers by installed node name.

5. Select an IBM HTTP Server from the list and press Enter.

   PowerHA SystemMirror displays the **Add an IBM HTTP Server to the Cluster** panel.

6. Enter field values as follows:

Table 23. Add an IBM HTTP Server to the Cluster fields	
Field	**Value**
Application Name	Displays a symbolic name for the collection of PowerHA SystemMirror resources that represent the WebSphere component.
IBM HTTP Server Name	Displays the name server where the IBM HTTP Server was installed. This field cannot be modified.
Primary Node	Displays the name of the node where the IBM HTTP Server has been installed. This field cannot be modified.
Takeover Node(s)	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fall over.

Table 23. Add an IBM HTTP Server to the Cluster fields (continued)	
Field	**Value**
Service Interface	Enter (or select from the picklist) the value IHS should use for the Listen parameter on port 80. The value you specify is written to the **httpd.conf** file and IPAT via IP Aliasing is used.

7. Press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

   PowerHA SystemMirror Smart Assist for WebSphere stops the IBM HTTP Server, as it is now managed by PowerHA SystemMirror. It also modifies the IP address the IBM HTTP Server listens on as you have specified.

   Note: From this point onward, do *not* stop or start the IBM HTTP Server manually; use the SMIT panels for these tasks.

8. If you want a *mutual takeover* or *multiple node* configuration for your IBM HTTP Server, create additional IBM HTTP Server resources.

9. Return to previous SMIT panels to perform other configuration tasks or complete the tasks in Verifying and completing your configuration setup.

Related tasks

Verifying and completing your configuration setup
After you have installed and configured PowerHA SystemMirror Smart Assist for WebSphere, verify that it is valid.

Adding a Tivoli Directory Server PowerHA SystemMirror Resource

The first time you add a Tivoli Directory Server resource, it is created in a hot standby cluster configuration. If you want a multiple node configuration, create additional Tivoli Directory Server resources and specify the applicable primary nodes, takeover nodes, and service interfaces.

Note: A mutual takeover configuration is not supported for the Tivoli Directory Server.

To create the PowerHA SystemMirror configuration for a Tivoli Directory Server, complete the following steps:

1. If you have changed the configuration for your Tivoli Directory Server, verify that you migrate changes in the **/usr/ldap/etc/ibmslapd.conf** file on the primary node, to the same file on the takeover node that you have defined. If the two files are not synchronized, the Tivoli Directory Server might not work as expected.

2. Ensure that the Tivoli Directory Server is running.

3. From the command line on the primary node that is running the Tivoli Directory Server, enter `smit hacmp`.

4. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror** > **Configure a Tivoli Directory** and press Enter.

   SMIT displays a list of discovered applications.

5. Select **WebSphere Smart Assistant** and press Enter.

   SMIT displays a list of discovered WebSphere components.

6. Select **Tivoli Directory Server** and press Enter.

   PowerHA SystemMirror displays the Add a Tivoli Directory Server to the Cluster panel.

7. Enter field values as follows:

Table 24. Add a Tivoli Directory Server to the Cluster fields

Field	Value
Application Name	A symbolic name for the collection of PowerHA SystemMirror resources that represent the WebSphere component.
Primary Node	Defaults to the name of the node that is running Tivoli Directory Server. This field cannot be modified.
Takeover Node	The name of a single cluster node to which the application might fallover.
Service IP Label	The service IP label (defined in PowerHA SystemMirror) that the application uses (IPAT via IP aliasing). If a PowerHA SystemMirror service IP label does not exist, this creates a service IP label according to the values that you specify. Or, select one service interface from the pick list and press Enter.
Tivoli Directory Server Password	The password for the Tivoli Directory Server. The password is stored in the **/usr/es/sbin/cluster/haws/config/ HAWS_CFG_tds_** *nodename***.cfg** file (where *nodename* identifies the name of the local node). Only users with root access have read access to this file.
Tivoli Directory Server Port	The number for the port that Tivoli Directory Server uses to communicate over the network and to stop and monitor the application.
DB2 Instance Name	The name of the DB2 instance that acts as the data repository for Tivoli Directory Server.

8. Press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

   PowerHA SystemMirror Smart Assist for WebSphere stops the Tivoli Directory Server because it is now managed by PowerHA SystemMirror. The application server is restarted for each application server whose HTTP transport has been changed.

   Note: From this point onwards, do not stop or start the Tivoli Directory Server manually. Use the SMIT panels for these tasks.

9. Optional: If you want a multiple node configuration for your Tivoli Directory Server (a mutual takeover configuration is not supported), create and configure additional Tivoli Directory Server resources. In the additional resources, specify the same takeover node as you originally specified in the first Tivoli Directory Server resource that you created.

10. Return to previous SMIT panels to perform other configuration tasks.

Related tasks

Verifying and completing your configuration setup
After you have installed and configured PowerHA SystemMirror Smart Assist for WebSphere, verify that it is valid.

Adding a transaction log recovery PowerHA SystemMirror resource

Ensure that you have configured persistent labels in your cluster configuration. This will enable the software to correctly mount the NFS file systems.

To add a transaction log recovery resource to your PowerHA SystemMirror cluster, complete the following steps:

1. From the command line, enter smit hacmp.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror** > **Configure WebSphere Cluster Transaction Log Recovery**, and press Enter.

SMIT displays a list of discovered applications.

3. Select **WebSphere Smart Assistant**, and press Enter.

   SMIT displays a list of discovered WebSphere components.

4. Select **WebSphere Cluster Transaction Log Recovery**, and press Enter.

   SMIT displays a list of available WebSphere Clusters by WebSphere Cell and WebSphere Cluster name.

5. Select a cluster from the list and press Enter.

   PowerHA SystemMirror displays the **Add WebSphere Cluster Transaction Log Recovery to the Cluster** panel.

6. Enter field values as follows:

Table 25. Add WebSphere Cluster Transaction Log Recovery to the Cluster fields

Field	Value
Application Name	A symbolic name for the collection of PowerHA SystemMirror resources that represent the WebSphere component.
WebSphere Application Server Cell	The name of the WebSphere Cluster that you want to keep highly available. This field cannot be modified.
WebSphere Cluster Name	The name of the WebSphere Cell, which manages the WebSphere Cluster that you want to keep highly available. This field cannot be modified.
Service Interface	The service IP label (defined in PowerHA SystemMirror) that the application uses (IPAT via IP aliasing). If a PowerHA SystemMirror service IP label does not exist, this creates a service IP label according to the values that you specify.
Volume Group	Enter (or select from the pick list) the name of the volume group that stores the transaction log.
Transaction Log Path	Enter the relative path to the transaction log file. If the path does not exist, PowerHA SystemMirror Smart Assist for WebSphere creates it for you. PowerHA SystemMirror Smart Assist for WebSphere also copies the transaction log of each WebSphere Cluster member to the path that you specify. For each member of the WebSphere cluster, a new transaction log volume group is mounted and the current Transaction Log files are copied to the new volume group. **Note:** To ensure that the mounting operation performs correctly, configure persistent IP labels in the cluster.

7. Press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

   PowerHA SystemMirror Smart Assist for WebSphere stops each application server in the cluster and changes the transaction log directory to the new location. After PowerHA SystemMirror performs the cross mount of the new location of the transaction logs, the WebSphere cluster members can be restarted.

8. Create additional transaction recovery log resources as needed for each WebSphere cluster found during discovery.

   Note: For each additional transaction recovery log resource, specify a separate volume group to contain the shared transaction logs and ensure that you have persistent labels that are configured.

9. Return to previous SMIT panels to perform other configuration tasks or complete the tasks in Verifying and completing your configuration setup.

   Note: After the resource group is online, use the Admin console to start each member of the WebSphere cluster.

Related tasks
Verifying and completing your configuration setup

After you have installed and configured PowerHA SystemMirror Smart Assist for WebSphere, verify that it is valid.

Verifying and completing your configuration setup

After you have installed and configured PowerHA SystemMirror Smart Assist for WebSphere, verify that it is valid.

To confirm that your PowerHA SystemMirror Smart Assist for WebSphere configuration is valid, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

   PowerHA SystemMirror propagates your changes to all cluster nodes.
3. Start cluster services by completing the following steps:

   a) From the command line, enter `smit cl_admin`.

   b) Select **PowerHA SystemMirror Services** > **Start Cluster Services** and press Enter.
4. Verify that the following applications perform as expected:

   - WebSphere Application Server
   - WebSphere Deployment Manager
   - WebSphere Cluster Transaction Log Recovery
   - IBM HTTP Server
   - Tivoli Directory Server
5. Review the cluster configuration from the System Management Interface Tool (SMIT).
6. Ensure that the cluster is stable and running as expected.
7. If you had created an **/.rhosts** file on each cluster node, you can now delete this file.

Related information

Verifying and synchronizing a PowerHA SystemMirror cluster
Starting and stopping cluster services
Monitoring a PowerHA SystemMirror cluster

Testing the availability of WebSphere applications

Testing the availability of an application after you have it configured is optional, but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

To test your PowerHA SystemMirror Smart Assist for WebSphere configuration, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Test Your Application for Availability** and press Enter. SMIT displays a list of applications.
3. Select the application you want to test and press Enter.

   SMIT prompts you to confirm the action by pressing Enter again.
4. Press Enter to continue with the test of the selected application.

SMIT displays output to the display as the Cluster Test Tool runs.

Changing or showing WebSphere PowerHA SystemMirror resources

Once a WebSphere application is configure as a PowerHA SystemMirror resource, you can change or display it.

To change or show a WebSphere application configured as a PowerHA SystemMirror resource, complete the following steps:

1. Enter `smit hacmp`

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration**, and press Enter.

   SMIT displays a list of available applications.

3. Select the WebSphere application to change or show and press Enter.

4. Make the changes as needed in the field(s) for the selected application and press Enter.

   a) Most WebSphere components allow the same fields to be edited during a change as was allowed during add, except for the Application Name which cannot be modified.

   b) The WebSphere Cluster Transaction Log Recovery does not allow the volume group hold the shared transaction logs to be modified.

Changing or showing the PowerHA SystemMirror resources associated with your application

You may want to add resources such as file systems and volume groups to the resource group. These resources will always be acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, create separate resource groups for each set.

To change or show the resources associated with your WebSphere application, complete the following steps:

1. Enter `smit hacmp`

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration**, and press Enter.

   SMIT displays a list of available applications.

3. Select the application and press Enter.

   SMIT displays the **Change/Show Resources and Attributes for a Resource Group** for the application's resource group.

4. Make the changes as needed and press Enter.

Related information

Administration guide

Managing your applications

If the Smart Assist used to make your application highly available has unique configuration tasks, you can do them from the Smart Assist SMIT menu.

To manage the PowerHA SystemMirror WebSphere applications, complete the following steps:

1. Enter `smit hacmp`

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Manage Your Applications** and press Enter.

   SMIT displays a list of available applications.

3. Select the WebSphere application and press Enter.

Removing a WebSphere application from PowerHA SystemMirror

When you remove a WebSphere application from PowerHA SystemMirror, PowerHA SystemMirror stops managing the availability of the resources you remove, since they are no longer part of the PowerHA SystemMirror cluster definition. The applications themselves are *not* removed.

To remove a WebSphere application from PowerHA SystemMirror, complete the following steps:

1. Enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** > and press Enter.

   SMIT displays a list of available applications.
3. Select the WebSphere application to remove and press Enter.

   Depending on the component you select, SMIT displays a list of that type of resource.
4. Select the resource you want to remove and press Enter to remove this resource from the PowerHA SystemMirror Configuration Database on the local node.

SMIT prompts you to confirm that you want to remove the component by pressing Enter again.

Removing WebSphere PowerHA SystemMirror resources

The following information describe particular effects for each type of WebSphere resource.

Removing a WebSphere Application Server PowerHA SystemMirror resource
Removing a WebSphere Application Server PowerHA SystemMirror resource removes all analogous resources from PowerHA SystemMirror, including application controllers, application monitors, service IP labels, and resource groups. In addition, any HTTP transport host that was modified by PowerHA SystemMirror Smart Assist for WebSphere is reset to its original value.

Removing a Deployment Manager PowerHA SystemMirror resource
When a Deployment Manager is removed, the removal includes all analogous resources from PowerHA SystemMirror, application controllers, application monitors, service IP labels, and resource groups. In addition, any modified HTTP transports are changed back to the original value.

Removing an IBM HTTP Server PowerHA SystemMirror resource
Removing an IBM HTTP server PowerHA SystemMirror resource resets the Listen parameter to its original value and removes all analogous resources from PowerHA SystemMirror, including application controllers, application monitors, service IP labels, and resource groups.

Removing a Tivoli Directory Server PowerHA SystemMirror resource
Make sure the Tivoli Directory server is running when you remove it from the cluster definition. All analogous resources from PowerHA SystemMirror, including application controllers, application monitors, service IP labels, and resource groups are removed.

Removing a Transaction Log Recovery PowerHA SystemMirror resource
The Transaction Log directories for the application controllers on the WebSphere cluster revert to their previous values. All analogous resources from PowerHA SystemMirror, including service IP labels, and resource groups are removed.

Monitoring and troubleshooting your configuration

You can troubleshoot PowerHA SystemMirror Smart Assist for WebSphere issues.

PowerHA SystemMirror Smart Assist for WebSphere writes configuration data into several files as follows.

Type of Data	Location
Discovery Data	PowerHA SystemMirror Smart Assist for WebSphere writes discovery data to the **/usr/es/sbin/cluster/sa/was/etc/was.disc** file on the control node.

Type of Data	Location
Configuration Data	PowerHA SystemMirror Smart Assist for WebSphere stores configuration data in the **HACMPsa_metadata** PowerHA SystemMirror Configuration Database (ODM).
Log Data	The **/var/hacmp/log/wassa.log** file contains debug information that is typically used only by technical support.

If you encounter a problem with the PowerHA SystemMirror resources created by PowerHA SystemMirror Smart Assist for WebSphere, follow the directions in this section.

To resolve configuration issues:

1. Review the following log files for information related to the time or type of event that occurred:

   - The log files for the specific application (for example, WebSphere Application Server or Tivoli Directory Server)
   - The **/usr/es/sbin/cluster/sa/was/etc/was.disc** file on the control node for information that PowerHA SystemMirror collects about WebSphere components when the PowerHA SystemMirror discovery process is run.
   - The **smit.log** file to examine the SMIT processes that had been invoked from the user interface
   - The **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application
   - The **/var/hacmp/log/wassa.log** file for information about PowerHA SystemMirror resources configuration and the application monitors
   - The **/var/hacmp/log/wassa.log** file for the WebSphere information that has been gathered and instance resource groups that have been created

2. Verify and synchronize your cluster.

3. f configuration problems persist, remove the resource configuration that PowerHA SystemMirror Smart Assist for WebSphere created.

Related information

Troubleshooting guide

Smart Assist for WebSphere start and stop scripts

Smart Assist for WebSphere uses information from an application configuration to generate the start and stop scripts for PowerHA SystemMirror.

The following list describes what the PowerHA SystemMirror start and stop scripts do:

Application	Start Script Description	Stop Script Description
WebSphere Application Server	Starts WebSphere Application Server by calling **startServer.sh** for a discovered WebSphere Application Server of a specified name.	Stops WebSphere Application Server by calling **stopServer.sh** for a discovered WebSphere Application Server of a specified name.
WebSphere Deployment Manager	Calls **startManager.sh** .	Calls **stopManager.sh** .
Tivoli Directory Server	Starts the Tivoli Directory Server by calling **ibmslapdn**. Starts WebSphere Application Server by calling **startServer.sh** for the discovered WebSphere Application Server associated with Tivoli Directory Server.	Stops the discovered WebSphere Application Server associated with Tivoli Directory Server. Calls **ibmdirctl** to stop the Tivoli Directory Server.

Application	Start Script Description	Stop Script Description
IBH HTTP Server	If an IBM HTTP server (IHS) is installed on the same node as the WebSphere Application Server, starts IHS by calling **apachectl start**.	If an IBM HTTP server (IHS) is installed on the same node as the WebSphere Application Server, stops IHS by calling **apachectl stop**.

PowerHA SystemMirror configuration summary

These topics contain detailed configuration information about several different PowerHA SystemMirror Smart Assist for WebSphere resources and default setting of these resources.

PowerHA SystemMirror Smart Assist for WebSphere resources

PowerHA SystemMirrorSmart Assist for WebSphere uses certain naming conventions when creating names for resources.

The name *PowerHA SystemMirror Smart Assist for WebSphere* contains a maximum of 32 characters. Any name longer than 32 characters is truncated to 32. While determining name syntax and length, PowerHA SystemMirror Smart Assist for WebSphere also ensures that the names it creates are unique.

Similar to any other PowerHA SystemMirror resource, you can change the default names that PowerHA SystemMirror Smart Assist for WebSphere creates by using the Extended Configuration path in SMIT.

Related information
Administration guide

WebSphere Application Server PowerHA SystemMirror resource
PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror components for a WebSphere Application Server

These components include:

PowerHA SystemMirror Component	Name
Resource group	**WAS_Node_***WASCell_WASNode***_rg** where *WASCell* is the name of the cell in which the WebSphere Application server resides and *WASNode* is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror application server	**WAS_Node_***WASCell_WASNode***_as** where *WASCell* is the name of the cell in which the WebSphere Application server resides and *WASNode* is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror custom monitor	**WAS_Node_***WASCell_WASNode***_monitor** where *WASCell* is the name of the cell in which the WebSphere Application server resides and *WASNode* is the name of the PowerHA SystemMirror node This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin**. Note that the name of the custom monitor includes the name of the PowerHA SystemMirror application server.
Start script	**cl_wasnodestart** This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin**.
Stop script	**cl_wasnodestop** This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin**.

Deployment Manager PowerHA SystemMirror resource
PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror components for the WebSphere Deployment Manager.

These components include:

PowerHA SystemMirror Component	Name
Resource group	**WAS_DM_**_WASCell_**_rg** where _WASCell_ is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror application server	**WAS_DM_**_WASCell_**_as** where _WASCell_ is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror custom monitor	**WAS_DM_**_WASCell_**_monitor** where _WASCell_ is the name of the primary PowerHA SystemMirror node This file is located in the directory **/usr/es/sbin/ cluster/sa/was/sbin**. Note that the name of the custom monitor includes the name of the PowerHA SystemMirror application server.
Start script	**cl_wasdmstart** This file is located in the directory **/usr/es/sbin/ cluster/sa/was/sbin**.
Stop script	**cl_wasdmstop** This file is located in the directory **/usr/es/sbin/cluster/sa/was/ sbin**.

IBM HTTP Server PowerHA SystemMirror resource
PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror components for the IBM HTTP Server.

These components include:

PowerHA SystemMirror Component	Name
Resource group	**WAS_IHS_**_IHSNode_**_rg** where _IHSNode_ is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror application server	**WAS_IHS_**_IHSNode_**_as** where _IHSNode_ is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror custom monitor	**WAS_IHS_**_IHSNode_**_monitor** where _IHSNode_ is the name of the primary PowerHA SystemMirror node This file is located in the directory **/usr/es/sbin/ cluster/sa/was/sbin**. Note that the name of the custom monitor includes the name of the PowerHA SystemMirror application server.
Start script	**cl_wasihsstart** This file is located in the directory **/usr/es/sbin/cluster/sa/was/ sbin**.
Stop script	**cl_wasihsstop** This file is located in the directory **/usr/es/sbin/cluster/sa/was/ sbin**.

Tivoli Directory Server PowerHA SystemMirror resource
PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror components for the Tivoli Directory Server

These components include:

PowerHA SystemMirror Component	Name
Resource group	**tdsrg_**_Node_ where _Node_ is the name of the PowerHA SystemMirror node where Tivoli Directory Server resides.

PowerHA SystemMirror Component	Name
PowerHA SystemMirror application server	**tdsam_**_Node_ where _Node_ is the name of the primary PowerHA SystemMirror node
PowerHA SystemMirror custom monitor	**HAWS_MON_tdsas_Primary**_Node_**.sh** where _Node_ is the name of the primary PowerHA SystemMirror node This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin**. Note that the name of the custom monitor includes the name of the PowerHA SystemMirror application server.
Start script	**startTDS** This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin**.
Stop script	**stopTDS** This file is located in the directory **/usr/es/sbin/cluster/sa/was/sbin** .

Transaction log recovery PowerHA SystemMirror resource

PowerHA SystemMirror Smart Assist for WebSphere creates the PowerHA SystemMirror components for the Transaction log recovery resource.

These components include:

PowerHA SystemMirror Component	Name
Resource group	**WAS_CLUS_**_Cell_Cluster_**_rg** where _Cell_ is the name of the WebSphere cell; _Cluster_ is the name of the primary PowerHA SystemMirror node.

Settings for application monitors

PowerHA SystemMirror Smart Assist for WebSphere enables PowerHA SystemMirror to monitor each application your have configured to be monitored.

The following table lists the default settings for the custom monitors for each application.

Settings for...	WebSphere Application Server	Deployment Manager	Tivoli Directory Server	IBM HTTP Server
Monitor Interval	120 sec.	120 sec.	120 sec.	120 sec.
Stabilization Interval	420 sec.	420 sec.	420 sec.	420 sec.
Restart Count	3	3	3	3
Restart Interval	1800 sec.	1800 sec.	1800 sec.	1800 sec.
Action on Application Failure	Fallover	Fallover	Fallover	Fallover

You can change the values set for an application monitor from SMIT.

Related information

Administration guide

Smart Assist for WebSphere MQSeries

WebSphere MQSeries is middleware software for messaging and queuing network communication. You can use WebSphere MQSeries to increase the speed of distributed applications by simplifying application development and testing.

WebSphere MQSeries runs on various platforms. WebSphere MQSeries applications use a consistent application programming interface (API) across all platforms. With WebSphere MQSeries your applications can communicate with each other across a network of different components, such as processors, subsystems, operating systems, and communication protocols.

In PowerHA SystemMirror Version 7.2.4, or later, WebSphere MQSeries listeners that are associated with WebSphere MQ queue manager are discovered by Smart Assist for WebSphere MQSeries, and each WebSphere MQSeries listener is configured with an individual application monitor. Thus, high availability of each listener is managed by PowerHA SystemMirror. If a WebSphere MQSeries listener fails, the corresponding WebSphere MQSeries listener is restarted without disturbing the other WebSphere MQSeries listeners that are associated with the WebSphere MQ queue manager.

WebSphere MQSeries resource groups have the following resources:

- Queue manager as an application controller and application monitor with the corresponding start and stop script.
- Shared volume group.
- Data file system and log file system that are created on the shared volume group.
- Service IP label.

Note: In PowerHA SystemMirror Version 7.2.4, or later, Smart Assist for WebSphere MQSeries supports configuration and monitoring of WebSphere MQSeries listeners. This feature support is not enabled through migration. To use this feature, you must first remove the Smart Assist application from PowerHA SystemMirror by using SMIT and configure the Smart Assist application again. For more information, see Removing an application and Configuring Smart Assist for WebSphere MQSeries.

Related information
WebSphere MQ documentation
IBM Redbooks MQSeries Primer

Planning for WebSphere MQSeries

You must install the operating system, PowerHA SystemMirror, and WebSphere MQSeries using normal procedures on all systems in the cluster.

Install WebSphere MQSeries onto internal disks (non-shared) on each of the nodes and do not share a single installation on shared disks. It is important that you are running identical versions of software on all cluster nodes, except during a rolling upgrade.

When installing WebSphere MQSeries in a cluster, the mqm username and the mqm groupname must have been created and have the same numeric value on all of the cluster nodes.

The smallest unit of failover for WebSphere MQSeries is a queue manager, because you cannot move part of a queue manager without moving all of it. Place each queue manager in a separate resource group, along with the resources it depends on. The resource group must contain the following resources:

- The shared volume group used by the queue manager.
- The IP address used to connect to the queue manager (the service address).
- The object that represents the queue manager.

Note: You can put multiple queue managers into the same resource groups, but they might fallover to another node together, even if the problem causing the fallover is confined to one queue manager. However, this process causes unnecessary disruption to applications that are using other queue managers.

A queue manager that is used in a PowerHA SystemMirror cluster must have its recovery logs and data on shared disks, so that they can be accessed by a surviving node in the event of a node failure. A node running a queue manager must also maintain a number of files on internal (non-shared) disks. These files include files that relate to all queue managers on the node, such as the `/var/mqm/mqs.ini` file, and queue manager-specific files that are used to generate internal control information. Files related to a queue manager are divided between internal disks and shared disks.

A queue manager can use a single-shared disk to store all log files and data files. However, for optimal performance, you can place log files and data files in a separate file system so that they can be separately tuned for disk I/O.

Software requirements

To view what versions of WebSphere MQSeries are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Related information
Using WebSphere MQ with a high availability cluster on UNIX

Installing WebSphere MQSeries

Before you install WebSphere MQSeries, verify that the mqm username and the mqm groupname are created and they have the same numeric value on all of the cluster nodes. Set the primary group of user mqm to mqm.

To make WebSphere MQSeries queue manager highly available on a node, install the following filesets:

- mqm.base.runtime
- mqm.base.samples
- mqm.base.sdk
- mqm.client.rte
- mqm.java.rte
- mqm.jre.rte
- mqm.keyman.rte
- mqm.msg.en_US
- mqm.server.rte
- mqm.txclient.rte
- mqm.base.runtime
- mqm.man.en_US.data

Related information
Installing a WebSphere MQ client
Installing a WebSphere MQ server
WebSphere MQ components and filesets

Configuring WebSphere MQSeries

Before you can configure WebSphere MQSeries, verify that PowerHA SystemMirror is installed and configured on all nodes in the cluster.

Configuring a shared disk for WebSphere MQSeries

A WebSphere MQSeries queue manager in a PowerHA SystemMirror cluster requires data files and log files to be in a common named remote file systems on a shared disk.

To configure a shared disk to work with WebSphere MQSeries, complete the following steps:

1. Create a fast disk takeover enabled volume group bey using the **cspoc** command. This volume group is used for the queue managers data and log files, and is managed by the PowerHA SystemMirror cluster in the same resource group as the queue manager.
2. Create a file system for data and another file system for logs by using the volume group created in step 1.
3. For each node in the cluster, import the volume group and set the volume group to vary on. Verify that the file system for the volume group can be mounted. The mgm user must own the mount points.
4. Unmount the file system and vary off the volume group.

Related information

PowerHA SystemMirror commands

Creating a queue manager for a PowerHA SystemMirror cluster

Before you can use a queue manager in a PowerHA SystemMirror cluster, you must create a queue manager on one of the nodes in the cluster.

To create a queue manager for a PowerHA SystemMirror cluster, complete the following steps:

1. Select one of the nodes in the cluster on which you want to create the queue manager and login as a root user.
2. Vary on the volume group and mount that data file system and log file system that you created for the shared disks.
3. Change the owner permissions of the data file system and log file systems by running the following commands:

```
chown -R mqm:mqm file_system_name
chmod -R 2775 file_system_name
```

where *file_system_name* is the name of the file system.

4. Create the queue manager as the mqm user by using the following command:

```
crtmqm -md  /MQHA/qmgrname/data -ld  /MQHA/qmgrname/log  qmgrname
```

where *qmgrname* is the name of the queue manager.

5. Display the **addmqinf** command as the mqm user by running the following command:

```
dspmqinf -o command qmgrnameTo
```

where *qmgrname* is the name of the queue manager.

6. Run the **addmqinf** command that is displayed as a result of running the **dspmqinf** command in step 5. The **addmqinf** command that you run is similar to the following example:

```
<p>  addmqinf -sQueueManager -vName=qmgrname -vDirectory=qmgrname
-vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
```

where *qmgrname* is the name of the queue manager.

7. Unmount the queue manager file systems and vary off the volume group.

Related information

Queue Manager Clusters

Adding a queue manager configuration to other nodes

After you have created a queue manager, you can add configuration from the queue manager that you created to other nodes in the PowerHA SystemMirror cluster.

To add the configuration information for the queue manager to each of other nodes in the PowerHA SystemMirror cluster, complete the following steps on each of the other nodes:

1. Vary on the volume group and mount the queue manager data file system and log file system.

2. Add the queue manager configuration information to the node by editing the `/var/mqs.ini` file, or by running the **addmqinf** command that is displayed when you run the **dspmqinf –o command qmgrnameTo** command, where, *qmgrname* is the name of the queue manager.

   Note: If you run the **addmqinf** command you must be logged in as the mqm user.
3. Start and stop the queue manager to verify that the configuration is working correctly.
4. Unmount the queue manager data file system and log file system.

Configuring Smart Assist for WebSphere MQSeries

After you configure WebSphere MQSeries for your environment, you can configure Smart Assist for WebSphere MQSeries.

Prerequisite for discovering WebSphere MQSeries listener

Smart Assist for WebSphere MQSeries discovers all WebSphere MQSeries listeners that are associated with WebSphere MQ queue manager. The default listener is SYSTEM.DEFAULT.LISTENER.TCP.

To discover WebSphere MQSeries listeners, you must ensure that the WebSphere MQ queue manager is up and running by performing the following steps:

1. Vary on the volume group.
2. Mount the data file system and the log file system of WebSphere MQ queue manager.
3. Start the WebSphere MQ queue manager.

During the configuration, Smart Assist for WebSphere MQSeries creates an application monitor for each WebSphere MQSeries listener and WebSphere MQ queue manager with the corresponding application start script and application stop script.

Note: After you configure Smart Assist for WebSphere MQSeries, you can stop the WebSphere MQ queue manager by performing the following steps:

1. Stop the WebSphere MQ queue manager.
2. Unmount the data file system and the log file system of WebSphere MQ queue manager.
3. Vary off the volume group.

Automatic discovery and configuration for Smart Assist for WebSphere MQSeries

Using SMIT you can set up Smart Assist for WebSphere MQSeries to automatically discover and WebSphere MQSeries instances running in the cluster along with its resources, such as shared volume groups and service IP addresses.

To set up automatic discover and configuration, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
3. From the list of applications, select **MQ Series Smart Assist**, and press Enter.
4. Select **Automatic Discovery and Configuration**, and press Enter.
5. Select the queue manager that you want to be automatically discovered and configured, and press Enter.
6. Enter the following information for the fields:

Table 26. Smart Assist for WebSphere MQSeries automatic discover and configuration fields	
Fields	**Values**
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the WebSphere MQSeries queue manager component.
WebSphere MQSeries server instance name	Displays the name of the queue manager that you selected in step 5. This field cannot be modified.
Primary Node	Displays the name of the node where the queue manager is currently active. This field cannot be modified.
Takeover Nodes	Enter (or select from the pick list) the name of one or more cluster nodes to which the application might fallover.
Service IP Label	Enter or select from the list the service IP label or service IP address that the clients use to communicate with the queue manager by using IPAT with IP aliasing.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This field is not a required field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

7. Verify that all fields are correct and press Enter.

   Note: Smart Assist for WebSphere MQSeries stops the queue manager because it is now managed by PowerHA SystemMirror.

Manual discovery and configuration for Smart Assist for WebSphere MQSeries

You can use an xml file to configure Smart Assist for WebSphere MQSeries.

To manually configure Smart Assist for WebSphere MQSeries, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
3. From the list of applications, select **MQ Series Smart Assist** > **Manual Configuration**, and press Enter.
4. In the File Name field, specify the path where the xml file is located, and press Enter. You can find a sample xml at the, `/usr/es/sbin/cluster/sa/wmq/config/cl_wmq_manual_config.xml` location.

   Note: You cannot modify the Smart Assist ID field. This ID is automatically generated.

Managing Smart Assist for WebSphere MQSeries

After you have configured WebSphere MQSeries,Smart Assist for WebSphere MQSeries, and verified that they are working, you can manage Smart Assist for WebSphere MQSeries in your environment.

Changing WebSphere MQSeries PowerHA SystemMirror resources

After WebSphere MQSeries components are configured as a PowerHA SystemMirror resource, you can change or display the resources that are being used.

To change the WebSphere MQSeries PowerHA SystemMirror resources, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Applications PowerHA SystemMirror Configuration**, and press Enter.
3. Select the required application that you want to change, and press Enter.
4. Change the fields that you want for the selected application, and press Enter. You can change the values for all fields except the **Application Name** field.

   Note: To add or remove an application monitor for a WebSphere MQ listener from the Smart Assist application configuration, you must first remove the Smart Assist application from PowerHA SystemMirror by using SMIT and configure the Smart Assist application again. For more information, see Removing an application and Configuring Smart Assist for WebSphere MQSeries.

Changing resources associated with your application

You can add resources, such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, create separate resource groups for each set.

To change resources associated with your application, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show the Resources Associated with Your Application**, and press Enter.
3. Select the application that you want to change, and press Enter.
4. Select the associated resource group, and press Enter. Modify the resources.

Removing an application

When you remove an application from the list of configured WebSphere MQSeries applications, PowerHA SystemMirror stops managing the availability of the resources you remove because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration**, and press Enter.
3. From the list, select the application that you want to remove, and press Enter.
4. From the list, select the resource that you want to remove, and press Enter.
5. Press Enter to remove the resource from the PowerHA SystemMirror Configuration Database on the local node.

   Note: Removing a resource from an application removes all analogous resources from PowerHA SystemMirror, which includes application controllers and monitors, service IP labels, and resource groups.

Smart Assist for WebSphere MQSeries start and stop scripts

Smart Assist for WebSphere MQSeries uses information from an application configuration to generate the start and stop scripts for PowerHA SystemMirror.

To start the queue manager use the **cl_wmq_start.sh** script.

To stop the queue manager use the **cl_wmq_stop.sh** script.

Smart Assist for WebSphere MQSeries resources

Smart Assist for WebSphere MQSeries uses standard naming convention to make it easy to identify various PowerHA SystemMirror resources that are created for Smart Assist for WebSphere MQSeries.

The following table lists PowerHA SystemMirror resources that are created for Smart Assist for WebSphere MQSeries.

Table 27. PowerHA SystemMirror resources created for Smart Assist for WebSphere MQSeries

PowerHA SystemMirror resources	Name
Resource Group	*Queue_mgr*_RG, where *Queue_mgr* is the queue manager name.
Application Controller	*Application_Name*, where *Application_Name* is the application controller name.
Custom monitor	*Application_name*_mon, where *Application_name* is the application name. The relevant script is the **cl_wmq_monitor** file that is located in the /usr/es/sbin/cluster/sa/wmq/sbin directory.
Start Script	The relevant script is the **cl_wmq_start** file that is located in the /usr/es/sbin/cluster/sa/wmq/sbin directory.
Stop Script	The relevant script is the **cl_wmq_stop** file that is located in the /usr/es/sbin/cluster/sa/wmq/sbin directory.

Custom application monitors

Smart Assist for WebSphere MQSeries configures a custom monitor for the WebSphere MQSeries applications that you configure in your PowerHA SystemMirror environment.

Custom application monitors check the health of an application at user-specified polling intervals.

The following table displays the default settings for the Smart Assist for WebSphere MQSeries custom monitor.

Table 28. Smart Assist for WebSphere MQSeries custom monitor settings

Field	Value
Name	*Application_name*_mon, where *Application_name* is the application name
Application controllers to monitor	*Application_controller*, where *Application_controller* is the name of the application controller
Monitor method	**/usr/es/sbin/cluster/sa/wmq/sbin/ cl_wmq_monitor -u** <*mqm_user*> -m <*MQM_NAME*>, where *mqm_user* is the user for the queue manager and *MQM_NAME* is the queue manager name.

Table 28. Smart Assist for WebSphere MQSeries custom monitor settings (continued)

Field	Value
Mode	Long running monitoring
Interval	120 sec
Hung monitor signal	9
Stabilization interval	180 sec
Restart interval	900 sec
Restart count	3
Action on application failure	Fallover
Cleanup method	**/usr/es/sbin/cluster/sa/wmq/sbin/cl_wmq_stop** **-u** *<mqm_user>* -m *<MQM_NAME>*, where *mqm_user* is the user for the queue manager and *MQM_NAME* is the queue manager name.
Restart method	**/usr/es/sbin/cluster/sa/wmq/sbin/cl_wmq_start** **-u***<mqm_user>* -m *<MQM_NAME>*, where *mqm_user* is the user for the queue manager and *MQM_NAME* is the queue manager name.

Troubleshooting Smart Assist for WebSphere MQSeries

To successfully troubleshoot Smart Assist for WebSphere MQSeries you must know where the different types of data are stored.

Use the following table to identify where Smart Assist for WebSphere MQSeries writes different types of data and the locations of the data.

Table 29. Smart Assist for Tivoli Directory Server data locations

Type of data	Location
Configuration data	Smart Assist for WebSphere MQSeries stores configuration data in the **HACMPsa_metadata** PowerHA SystemMirror Configuration Database (ODM).
Log data	The **/var/hacmp/log/wmqsa.log** file is used while Smart Assist for Tivoli Directory Server is configured and monitored.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   - View the log files in the Table 1 table.
   - The **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
   - The **hacmp.out** file for messages related to detection, migration, termination, and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to verify that all your clusters are using the latest configuration settings.

Smart Assist for DB2

Use this information for planning and configuring a PowerHA SystemMirror high available environment for DB2.

Getting started

PowerHA SystemMirror Smart Assist for DB2 extends an existing PowerHA SystemMirror configuration to include monitoring and recovery support for DB2 Universal Database (UDB) Enterprise Server Edition.

PowerHA SystemMirror Smart Assist for DB2 allows you to automatically configure PowerHA SystemMirror in an environment where DB2 is already configured to make non-partitioned DB2 instances highly available.

PowerHA SystemMirror Smart Assist cannot be used to configure a cluster in a partitioned (DB2 UDB DPF) environment.

To make use of PowerHA SystemMirror Smart Assist for DB2, the DB2 UDB software must be installed on all nodes that are going to be part of the cluster, and DB2 instances must be configured on some of the nodes. The shared physical disks that contain the DB2 instances must be configured on all cluster nodes that will serve as takeover nodes for the DB2 instances.

PowerHA SystemMirror Smart Assist automatically discovers your DB2 configuration and lets you create either one of the following cluster configurations:

- A hot standby two-node cluster
- A mutual takeover two-node cluster
- A cluster with more than two nodes and more than two instances where some of the nodes serve as takeover nodes for the configured DB2 instances.

Enhanced PowerHA SystemMirror and DB2 integration

PowerHA SystemMirror Smart Assist for DB2 is an extension of the PowerHA SystemMirror software that enables you to easily make your existing DB2 configuration highly available with PowerHA SystemMirror. You use PowerHA SystemMirror Smart Assist for DB2 after you have installed DB2 on the nodes.

Typically, DB2 runs on a single system and accesses data stored in volume groups. Although DB2 incorporates a number of application-specific availability features, the integration with PowerHA SystemMirror enhances application availability.

PowerHA SystemMirror Smart Assist for DB2:

- Automates the PowerHA SystemMirror cluster configuration process to minimize the time and effort required to integrate an application into a PowerHA SystemMirror cluster.
- Automatically detects how DB2 is configured. At startup, PowerHA SystemMirror verifies that the home directory for DB2 resides on shared physical volumes, that is, that DB2 uses shared non-concurrent volume groups. PowerHA SystemMirror Smart Assist for DB2 issues an error if DB2 is using non-shared volume groups.
- Makes highly available the resources specific to a particular DB2 instance, such as a shared volume group, file systems, and the aliased service IP label associated with the DB2 instance.
- Lets you easily add more DB2 instances to the PowerHA SystemMirror configuration. This lets you create a mutual takeover cluster configuration, or a cluster configuration with multiple nodes and resource groups with DB2 instances where these instances are recovered on other nodes by the means of fallover.
- Automatically starts and stops the DB2 instances on the nodes by the means of an application controller (a collection of start and stop scripts in PowerHA SystemMirror) that is created for each DB2 instance in the PowerHA SystemMirror cluster.
- Automatically monitors DB2 instances running on the nodes. If an application instance becomes unavailable, PowerHA SystemMirror tries to restart the application three times on the node. If the

application does not start after three attempts, PowerHA SystemMirror starts the application on another PowerHA SystemMirror cluster node. (This is the default behavior which you can change).

- Verifies the existing configuration of the DB2 components to ensure that the DB2 and PowerHA SystemMirror configuration is valid.

Keeping DB2 instances highly available

PowerHA SystemMirror increases the availability of DB2 components by eliminating single points of failure for DB2. A single point of failure exists when a critical function relies on a single component in a configuration. If that component fails, the application dependent on that component becomes unavailable.

To protect DB2 instances and eliminate single points of failure, PowerHA SystemMirror Smart Assist for DB2 requires that each database instance that you want to make highly available runs on at least two nodes.

In this guide and the PowerHA SystemMirror Smart Assist SMIT user interface, the following terminology is used, to define the node on which the DB2 instance may run in the cluster:

- A primary node for the DB2 instance. The *primary* cluster node is the node where the application runs unless a condition forces a fallover to another cluster node. It is also a home node for the resource group hosting the DB2 instance.

- A standby node for the DB2 instance. The *standby* cluster node is the node where the application runs after a fallover from the primary node.

  One of the typical configurations created by PowerHA SystemMirror Smart Assist for DB2 - a two-node hot standby configuration - consists of one primary node and one standby cluster node. The standby node acts as a standby for only one DB2 instance.

  You can specify location dependencies between resource groups in which your applications are included as resources. This allows you to keep the applications together on the same node, or to keep them always on separate nodes.

Related reference

Sample configurations
There are several different cluster configurations that you can have with DB2.

Related information

Administration guide

Increasing availability for a DB2 server

A DB2 server manages stored, related data. A DB2 server can provide back end storage for application data.

PowerHA SystemMirror Smart Assist for DB2 provides the following availability features for the DB2 server:

- Monitors the DB2 server

- If an application becomes unavailable, PowerHA SystemMirror tries to restart DB2 on its resource group's home node three times. If the application does not start after three attempts, PowerHA SystemMirror starts DB2 on the standby cluster node to continue service.

- Performs a fallover of the DB2 instance from one node to another node in the cluster, if necessary.

- W hen the primary node becomes available again the DB2 instance runs on the node on which it is currently hosted (this is the default), or moves the DB2 server and the DB2 database instance to its primary node. The fallback behavior depends on the fallback policy that you specify for the resource group hosting the DB2 instance. In PowerHA SystemMirror Smart Assist for DB2, the default resource group policy is Never Fallback.

Configuring PowerHA SystemMirror with PowerHA SystemMirrorSmart Assist for DB2

PowerHA SystemMirror Smart Assist for DB2 sets up your PowerHA SystemMirror resource configuration. It gathers information from the application configuration, the system configuration, and from you, and creates a PowerHA SystemMirror resource group to protect a specified database instance.

A PowerHA SystemMirror resource group contains a set of resources, such as cluster nodes, volume groups, and IP labels (used by server applications to communicate with clients), that PowerHA SystemMirror manages as a unit.

In addition to creating a resource group for each DB2 instance to be made highly available, PowerHA SystemMirror Smart Assist for DB2 performs the following tasks for each DB2 instance:

- Provides start and stop scripts for DB2 instances (a PowerHA SystemMirror application server).

  A PowerHA SystemMirror application server consists of the application start and stop scripts that are used in conjunction with PowerHA SystemMirror.

- Ensures the application has a service IP label that can be transferred to another system. This service IP label is the one configured in the **$INSTHOME/sqllib/db2nodes.cfg** file and the **$INSTHOME/.rhosts** file. You can create the **$INSTHOME/.rhosts** file when you configure the DB2 instance on the node.

- Ensures that the resource group contains all necessary shared volume groups as resources.

- Creates custom PowerHA SystemMirror process monitors to detect failures in application processes.

- Stores the generated PowerHA SystemMirror configuration in the PowerHA SystemMirror configuration database.

  When PowerHA SystemMirror Smart Assist for DB2 creates the PowerHA SystemMirror resource configuration it updates the PowerHA SystemMirror configuration but does not change the configuration for the DB2 instances.

Sample configurations

There are several different cluster configurations that you can have with DB2.

A two-node cluster with a hot standby configuration

Consider a single DB2 UDB instance in a single partition environment, where two cluster nodes use shared volume groups.

In this configuration, either node can have access to the shared volume group that contains the DB2 instance home file systems. In this environment, DB2 is installed on both nodes; the DB2 software is not shared. The following diagram shows the architecture necessary to support this configuration:

The following figure displays a two-node cluster configuration with a DB2 instance.
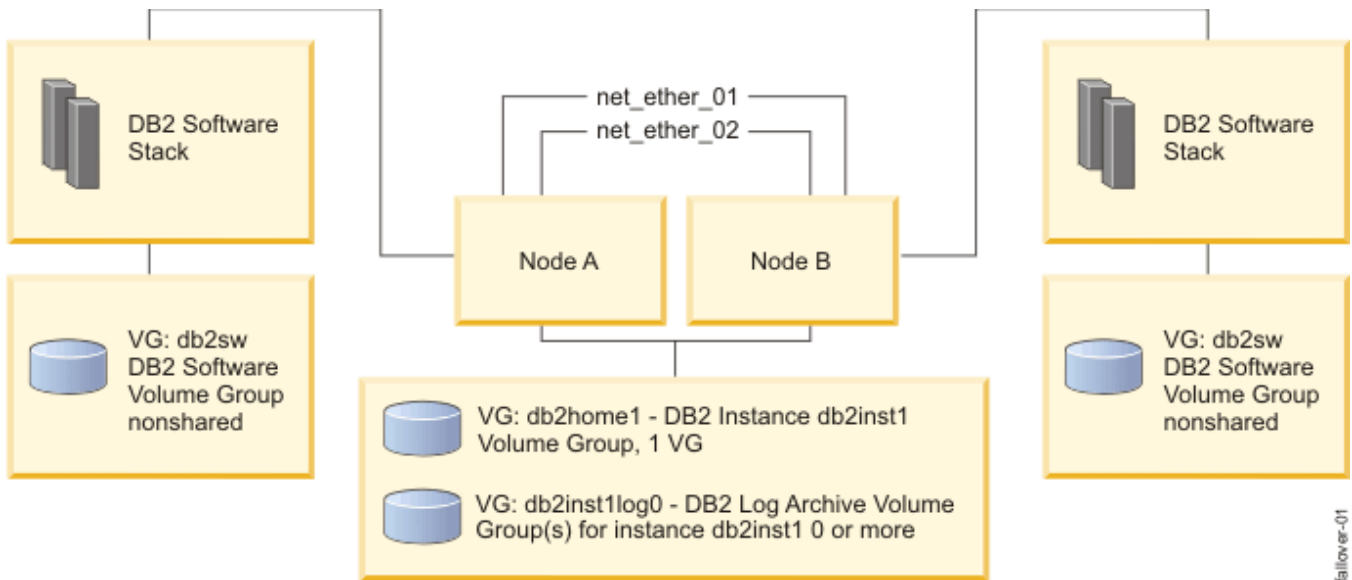
Figure 5. Two-node cluster configuration with a DB2 instance

Assume that in this configuration, node A owns the DB2 instance, that is, node A owns the resource group containing the application controller for the DB2 instance.

The shared volume group(s) for the DB2 instance contain the DB2 instance home and any number of volume groups containing copies of the archive and transaction logs for this instance. As part of this configuration, a service IP label is configured for the DB2 instance to communicate with other application tiers.

You can expand this two-node configuration by adding additional nodes to the nodelist for the resource group containing the DB2 instance application controller. The additional nodes can then serve as takeover nodes for the monitored DB2 instance. The DB2 software for the monitored instance will not be running on these additional takeover nodes until the PowerHA SystemMirror resource group that owns the instance falls over to the takeover node.

In PowerHA SystemMirror Smart Assist for DB2, a node can own one or more DB2 instances; additional resource groups must be added to the cluster to manage each new DB2 instance, that is, each DB2 instance must be included in a separate resource group.

A two-node cluster with a mutual takeover configuration
For better hardware utilization of DB2 UDB, you can easily expand the hot standby configuration described previously to configure a mutual takeover environment.

In a mutual takeover environment each DB2 instance has its own set of resources, shared volume groups, and service IP label as shown in the following graphic. In this cluster configuration that keeps several DB2 instances highly available, cluster resources are better utilized.

The following figure displays a two-node cluster environment with a DB2 mutual takeover configuration.
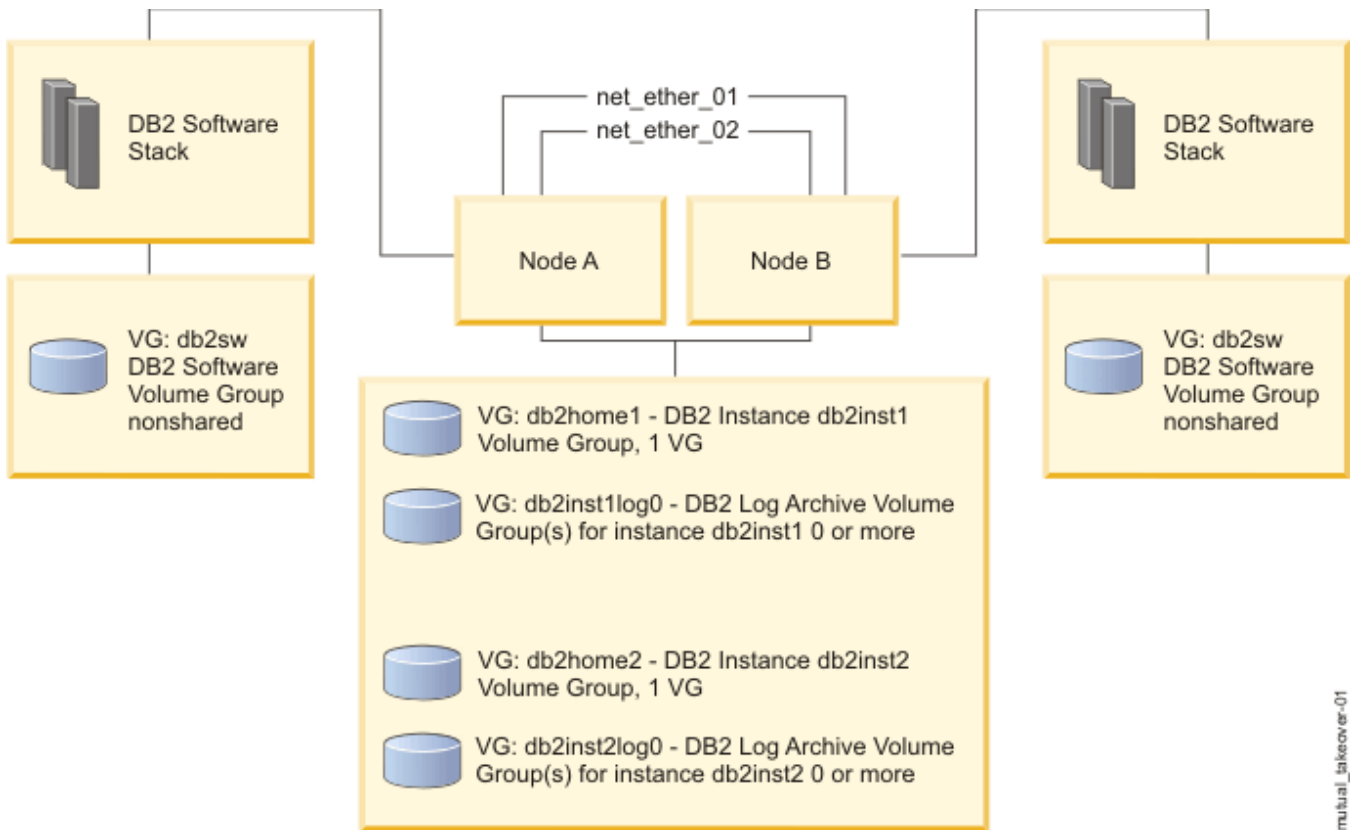
Figure 6. Two-node cluster environment with a DB2 mutual takeover configuration

In this diagram, let us assume the following configuration:

Node A owns DB2 instance: db2inst1, or is the home node for the resource group containing the resources associated with DB2 instance: db2inst1.

Node B is the takeover node for DB2 instance db2inst1 as it was in the diagram for the hot standby configuration. Node B is also the primary node for the database instance db2inst2, and its takeover node is Node A.

A separate resource group contains all the resources associated with DB2 instance db2inst2 as shown in the diagram. It is a requirement that each DB2 instance have a separate volume group for each DB2 instance monitored by PowerHA SystemMirror.

PowerHA SystemMirror Smart Assist for DB2 allows you to configure a two-node mutual takeover configuration such as this one using a one-step configuration process in SMIT. You can also expand the mutual takeover configuration.

Related reference

Multiple nodes cluster configuration

You can expand the mutual takeover configuration to more than two nodes and configure one or more takeover nodes for each DB2 instance. To configure more than two nodes in mutual takeover, you add each instance separately in PowerHA SystemMirror SMIT.

Multiple nodes cluster configuration

You can expand the mutual takeover configuration to more than two nodes and configure one or more takeover nodes for each DB2 instance. To configure more than two nodes in mutual takeover, you add each instance separately in PowerHA SystemMirror SMIT.

Planning your configuration

These topics describe the tasks you complete and the information you gather before running PowerHA SystemMirror Smart Assist for DB2. It is assumed that you are familiar with the DB2 application before you start planning for PowerHA SystemMirror Smart Assist for DB2.

Supported software

There are different versions of the software supported by PowerHA SystemMirror Smart Assist for DB2.

PowerHA SystemMirror Smart Assist for DB2 supports only a non-partitioned environment.

To view what versions of DB2 are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Limitations

PowerHA SystemMirror Smart Assist for DB2 has some limitations.

These limitations are:

- PowerHA SystemMirror configuration changes that affect your DB2 configuration, require that you update the DB2 configuration.
- Application configuration changes that can affect the PowerHA SystemMirror configuration, require you to update the PowerHA SystemMirror configuration.

PowerHA SystemMirror Smart Assist for DB2 supports several cluster configurations:

Recommended cluster configurations

There are several configurations that we recommend.

These configurations are:

- A hot standby two-node configuration
- A two-node mutual takeover configuration, in which each cluster node acts as a home node for one DB2 instance and at the same time serves as a standby cluster node for another DB2 instance
- Configurations with multiple nodes and multiple DB2 instances in which various nodes in the cluster serve as fallover nodes for the nodes that own the DB2 instances.

For a diagram and description of each configuration, see Sample configurations.

For planning requirements, see the section Planning for DB2 installation and configuration.

Related reference
Sample configurations
There are several different cluster configurations that you can have with DB2.

Planning DB2 installation and configuration

During installation and configuration for DB2, you need to consider some aspects.

Planning a PowerHA SystemMirror cluster

You must have a good working knowledge of the DB2 application and of PowerHA SystemMirror before planning your PowerHA SystemMirror implementation for a DB2 environment. PowerHA SystemMirror Smart Assist for DB2 takes care of many of the PowerHA SystemMirror cluster configuration activities for you; however, you need to plan and configure your basic environment before you run PowerHA SystemMirror Smart Assist for DB2.

As you plan your environment, record planning information as noted.

Complete the steps in this section for each application.

Step 1: Planning for a DB2 application

For each DB2 instance that you plan to keep highly available, you need to identify some information.

For each DB2 instance that you plan to keep highly available, identify:

- The primary node - the node on which the application usually runs
- The standby cluster node - the node on which DB2 runs after a fallover
- The service IP label uses the DB2 instance to communicate with clients. This must be the service IP label that is used in **$INSTHOME/sqllib/db2nodes.cfg** file and in the **$INSTHOME/.rhosts** file. You can create the **$INSTHOME/.rhosts** file when you configure the DB2 instance on the node.
- If you are planning to add more than one DB2 instance to the cluster, record the primary (home node for the resource group) and standby cluster nodes for each instance, along with the service IP label.

  You can have a number of DB2 instances running on different cluster nodes with one or more cluster nodes acting as fallover nodes for all of the instances. You can add nodes to the cluster to handle additional possible fallovers.

  If you are planning a two-node hot standby configuration, in the following table, list the home node, the standby node, and the service IP label for the DB2 application that you plan to protect with PowerHA SystemMirror.

Application	Name of Home Node	Service IP Label on Home Cluster Node	Name of Standby Cluster Node
DB2			

If you are planning a two-node mutual takeover configuration, expand the table for each DB2 instance.

If you are planning a multiple-node configuration, in case of fallover the DB2 instance will fallover to the node(s) listed in the resource group nodelist, that is, it will fall over to the first node in the list that is currently up and running.

Step 2: Planning the cluster topology

Plan for cluster nodes and cluster networks.

Complete the planning activities described in Planning cluster network connectivity in the *Planning Guide*.

Planning cluster nodes

Each node should own only one copy of the DB2 software stack, regardless of how many DB2 instances are supported by each node. It is required that only one software stack exists on each node, and that all nodes in the cluster are running the same version of DB2.

When you use PowerHA SystemMirror SMIT to add a resource group with the DB2 UDB instance, PowerHA SystemMirror automatically adds a DB2 instance to the PowerHA SystemMirror configuration, including any supporting topology configuration, such as nodes, networks and IP interfaces.

In addition, if you want to add a node to an already configured resource group containing a DB2 instance, you can do so using the PowerHA SystemMirror SMIT option **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration**.

Planning cluster networks

Before proceeding to install PowerHA SystemMirror Smart Assist for DB2, plan and configure the cluster networks:

- Connect and configure all IP network interfaces
- Add the DB2 service IP label to **/etc/hosts** on all nodes, or make the labels available via DNS.

In general, PowerHA SystemMirror uses the communication path for IP network discovery and automatic configuration of the cluster network topology. For this purpose, at configuration time, you will need to specify to PowerHA SystemMirror one of the following:

- A service IP address
- A fully-qualified domain name
- A resolvable IP label
- An already existing PowerHA SystemMirror node name.

Also, when you use PowerHA SystemMirror SMIT to add a DB2 UDB instance resource group, PowerHA SystemMirror automatically adds a DB2 instance to the PowerHA SystemMirror configuration, including any supporting topology configuration, such as nodes, networks and IP interfaces.

Related reference
Step 5: Configuring cluster topology
You need to configure the cluster topology to include several different items.

Step 6: Configuring service IP labels
The service IP label is a label for a network interface that clients use to communicate with the server application. PowerHA SystemMirror Smart Assist requires a service IP label for each application.

Related information
Planning cluster network connectivity

Step 3: Planning access to stored data
For each DB2 server, plan which volumes groups are to store data files.

In general, PowerHA SystemMirror Smart Assist for DB2 automatically discovers the components and resources that need to be made highly available for a DB2 cluster. This includes automatically identifying the available DB2 instances and volume groups associated with the specific instance. In SMIT, when adding DB2 instances to the configuration, you see the discovered resources and DB2 instances included in the picklists (accessible by pressing the F4 key).

Planning for storage entails the following:

- *For a two-node hot standby cluster configuration*, the primary and standby cluster nodes for a DB2 instance require access to the same volume group and file system that stores the data for the instance. This way if the instance falls over from the primary node to the standby node, it continues to access its data from the shared volume group. For PowerHA SystemMirror Smart Assist to collect configuration information for a DB2 instance, it must run on a cluster node that has access to the DB2 instance-specific volume groups and mounted file systems that store the application data.

  Configure a volume group as a resource in only one resource group. This means that you should configure different volume groups for different DB2 instances. Volume groups are varied on and file systems are mounted before you run the PowerHA SystemMirror Smart Assist for DB2 application.

- *For a two-node mutual takeover configuration,* configure shared non-concurrent volume groups on each cluster node. (A DB2 instance should be configured on each cluster node.)

List the volume groups that you plan to use with each DB2 instance. Note that the list of volume groups is only for your planning. PowerHA SystemMirror Smart Assist for DB2 discovers information about the volume groups used by the DB2 instances.

Application	Shared Volume Group
DB2	

During verification, PowerHA SystemMirror verifies the following aspects of your storage configuration:

- The DB2 instance home directory resides on shared storage.
- Shared volume groups are accessible on all nodes where a particular DB2 instance might reside in the cluster. This requires that all physical disks are defined on all participating cluster nodes.

  If the DB2 instance volume groups do not exist on a takeover node, then PowerHA SystemMirror automatically imports those volume groups as part of the verification automatic corrective action. For automatic import to be successful, the takeover nodes must already share the same set of disks.

Related information

Planning guide

Volume groups added to resource groups containing DB2 instances
When you add a DB2 instance to your configuration using the PowerHA SystemMirror SMIT option for the Smart Assist for DB2, the software adds volume groups to the resource group configuration. The volume groups added to the resource group are specific to the particular DB2 instance.

The volume groups that PowerHA SystemMirror adds automatically to the resource group:

- DB2 home volume group. This is the volume group that contains the file system with the DB2 instance data, such as the system database directory, database manager configuration, and DB2 instance registry variables. PowerHA SystemMirror adds the volume group that contains the file system for the particular instance to the DB2 instance resource group.
- Log volume groups that exist for the DB2 instance. These are the volume group(s) that contain the DB2 instance transaction logs.
- Tablespace Data volume group(s). These volume groups contain the tablespace file systems associated with each DB2 instance.

You can always add additional volume groups to the created resource group that contains the DB2 instance.

Note: Users and groups related to the DB2 instance(s) must have the same User ID and Group ID on all nodes.

Determining which volume groups will be added by PowerHA SystemMirror Smart Assist

You can determine which shared volume groups will participate in a particular DB2 instance resource group by running the following commands as the DB2 instance owner:

```
su - db2inst1 # become db2inst1 owner
db2 'connect to database' # connect to the particular DB
db2 'get database configuration'
```

PowerHA SystemMirror Smart Assist for DB2 uses the Path to log files entry, the OVERFLOWLOGPATH entry, and the FAILARCHPATH entry to determine which shared volume groups will be added to the resource group if they are available on multiple nodes.

The core binaries that run the DB2 database software must be installed on each node that participates in a PowerHA SystemMirror resource group.

Step 4: Installing the PowerHA SystemMirror software and Smart Assist filesets

Install the PowerHA SystemMirror software on all nodes in the cluster.

Related information

Installation guide

Step 5: Configuring cluster topology

You need to configure the cluster topology to include several different items.

Configure the cluster topology for PowerHA SystemMirror to include:

- Cluster nodes that you planned in the section Step 1: Planning for a DB2 application
- Network connectivity for the cluster.

  When you configure the network:

- Use discovery to gather information about network interfaces.

Related reference

Step 1: Planning for a DB2 application
For each DB2 instance that you plan to keep highly available, you need to identify some information.

Related information

Administration guide

Step 6: Configuring service IP labels

The service IP label is a label for a network interface that clients use to communicate with the server application. PowerHA SystemMirror Smart Assist requires a service IP label for each application.

Configure the IP labels in PowerHA SystemMirror SMIT menu for cluster topology to make them available to PowerHA SystemMirror Smart Assist for DB2. Configure the labels from the primary node and synchronize the configuration to the standby node.

Related reference

Step 1: Planning for a DB2 application
For each DB2 instance that you plan to keep highly available, you need to identify some information.

Related information

Planning cluster network connectivity
Configuring PowerHA SystemMirror cluster topology and resources (extended)

Step 7: Planning security

Only an administrator with root privileges can run the configuration utilities included in PowerHA SystemMirror Smart Assist for DB2. A **$INSTHOME/.rhosts** file, located in the DB2 instance home directory must be configured when using PowerHA SystemMirror Smart Assist for DB2.

PowerHA SystemMirror and DB2 each have their own security mechanism. Those security mechanisms remain in place; PowerHA SystemMirror Smart Assist for DB2 does not make any changes.

PowerHA SystemMirror Smart Assist for DB2 ensures that the **$INSTHOME/.rhosts** file is secure for the DB2 instances and is properly configured by running the cluster verification process. The automatic corrective action of the PowerHA SystemMirror verification utility notifies you of security errors and lets you correct them at the time during which they are discovered.

In addition, if the instance is monitored by PowerHA SystemMirror, the cluster verification ensures that the **/$INSTHOME/.rhosts** file in the home directory of the DB2 instance contains a proper entry, as follows:

```
<cluster service IP address> <DB2 instance owner user name>
```

Or as:

```
+ <DB2 instance owner user name>
```

In general, although the entry should only be used in trusted environments, the PowerHA SystemMirror verification does not flag an error if the **all hosts** entry appears in the **.rhosts** file, but issues a warning indicating that all remote hosts with the user name DB2 Instance owner name are granted access to the local host.

If PowerHA SystemMirror discovers that the IP addresses are missing in the **.rhosts** file for a particular resource group that contains a DB2 instance, it issues an error asking you to add one or more IP addresses to the **.rhosts** file in the DB2 instance home directory **.rhosts** file.

During verification, an automatic corrective action adds all known PowerHA SystemMirror service IP labels to the **.rhosts** file for the DB2 instance to ensure connectivity.

Related information
Managing cluster security
Managing user and groups

Planning for DB2 installation and configuration

Install and configure DB2 before you install PowerHA SystemMirror Smart Assist for DB2.

Because PowerHA SystemMirror Smart Assist for DB2 automates resource configuration for supported applications, it has configuration requirements for DB2. The following sections describe installation and configuration requirements and identify the information that you should collect before you run PowerHA SystemMirror Smart Assist for DB2.

Planning for DB2
Only one node accesses data at a time. The primary node for the DB2 instance accesses data when the server is running on the home node. At fallover, only the standby cluster node (which becomes the primary cluster node at fallover) accesses the shared volume group for the DB2 instance.

In two-node mutual takeover, two different DB2 instances must be configured, one on each node. The node that is the primary node for instance A is the takeover node for instance B, and, in reverse, the node that is the primary node for instance B is the takeover node for instance A.

Planning DB2 installation and configuration
During installation and configuration for DB2, you need to consider some aspects.

These aspect are:

- Make sure that DB2 installation path are same across all participating nodes in PowerHA SystemMirror cluster.
- The DB2 instance should be installed in the DB2 instance owner's home directory. PowerHA SystemMirror scripts (in particular, the discovery script) assume this.
- When you install the DB2 server, create a DB2 database instance only from the primary server.
- Make sure that there is an appropriate **.rhosts** file (as required by DB2), in the home directory of the database instance owner to start and stop the database instance.
- Specify the IP service label for the hostname in the **$INSTHOME/sqllib/db2nodes.cfg** file and in the **$INSTHOME/.rhosts** file in your environment.
- Ensure that the DB2 server on both the home and the standby nodes have the same configuration settings for:
  - User access

    You can configure user privileges through PowerHA SystemMirror, and synchronize those user accounts among the nodes in a cluster. The DB2 instance users and groups should have the same ID on all nodes in the cluster.
  - Port number assignments

    DB2 adds the port numbers to the **/etc/services** file on the primary cluster node for the DB2 instance when the database instance is created. Copy these port numbers to the **/etc/services** file on the other cluster nodes that will be included in the resource group containing this DB2 instance.

The following shows example entries in an **/etc/services** file:

```
DB2_db2inst1 60000/tcp
DB2_db2inst1_1 60001/tcp
DB2_db2inst1_2 60002/tcp
DB2_db2inst1_END 60003/tcp
db2c_db2inst1 50000/tcp
```

During PowerHA SystemMirror cluster verification, the cluster verification utility checks for this condition and, in the absence of a valid configuration, automatically corrects this condition for you.

In configurations that have more than one DB2 instance in the cluster, PowerHA SystemMirror adds the port entries in such a way as to avoid port conflicts.

Turning off automatic start of the DB2 Fault Monitor Coordinator
The PowerHA SystemMirror configuration established by PowerHA SystemMirror Smart Assist for DB2 requires that the system not automatically start the DB2 Fault Monitor Coordinator.

If the DB2 Fault Monitor Coordinator is set to automatically start, the **/etc/inittab** file contains the following entry:

```
fmc:2:respawn:<DB2DIR>/bin/db2fmcd #DB2 Fault Monitor Coordinator,
    Where DB2DIR is the installed path for DB2.
```

To turn off automatic restart of the DB2 Fault Monitor Coordinator, enter the following command:

```
"# chitab "fmc:2:off:<DB2DIR>/bin/db2fmcd #DB2 Fault Monitor Coordinator,
    Where DB2DIR is the installed path for DB2.
# chitab "ibmdir:2:off:/usr/ldap/sbin/rc.ibmdir > /dev/null 2>&1"
```

The **/etc/inittab** file contains the following entries when the DB2 Fault Monitor Coordinator is not set to automatically start:

```
fmc:2:off:<DB2DIR>/bin/db2fmcd #DB2 Fault Monitor Coordinator,
    Where DB2DIR is the installed path for DB2.
ibmdir:2:off:/usr/ldap/sbin/rc.ibmdir > /dev/null 2>&1
```

During the cluster verification, PowerHA SystemMirror checks to ensure each node that participates in a DB2 instance resource group has the Fault Monitor coordinator turned off. If the Fault Monitor coordinator is not disabled, an automatic corrective action during cluster verification allows you to disable it.

Turning off automatic startup of DB2 Instance on node reboot
Ensure that the DB2 instance is not set to automatically restart on system startup.

To turn off this function, you can run the command: db2iauto -off InstanceName as the DB2 instance owner.

Related reference
Creating the PowerHA SystemMirror configuration
You have several options for setting up a PowerHA SystemMirror configuration for DB2. You can create a two-node cluster with a hot standby configuration or with a mutual takeover configuration, or you can extend an existing cluster by adding more nodes and additional DB2 instances.

Recording information for PowerHA SystemMirror Smart Assist for DB2
If you plan to use PowerHA SystemMirror to make DB2 highly available, there is some information that you will need to collect before you use PowerHA SystemMirror Smart Assist for DB2.

The information you need to collect includes:

Table 30. PowerHA SystemMirror Smart Assist for DB2 information	
Environment information	**Value**
The name of the DB2 database	_____

Table 30. PowerHA SystemMirror Smart Assist for DB2 information (continued)	
Environment information	**Value**
The instance name for the DB2 database	_____

Recording information for PowerHA SystemMirror Smart Assist for DB2
If you plan to use PowerHA SystemMirror to make DB2 applications highly available, there is some information that you will need to collect before you use PowerHA SystemMirror Smart Assist for DB2.

The information you need to collect includes:

Table 31. PowerHA SystemMirror Smart Assist for DB2 information	
Environment information	**Value**
The DB2 instance that is used to start, stop, and monitor the DB2 database	_____

Planning backups, logging, and other DB2 parameters

When planning backups, logging, and other DB2 parameters there are several tips that you can use.

These tips are:

- If you already have or plan to create scheduled DB2 backup volume groups on local file systems, PowerHA SystemMirror Smart Assist for DB2 does not discover these volume groups. Thus, the volume groups may not be available after a fallover. To prevent this, if you have scheduled DB2 backup volume groups, manually add these volume groups to the DB2 instance resource groups.
- If you use a backup software, such as Tivoli Storage Manager (TSM), the software must also exist and be properly configured on the fallover node(s) in the cluster.
- If you use DB2 log shipping and standby database with *userexit* (especially using the pull method), PowerHA SystemMirror Smart Assist for DB2 might not be able to determine the location where the standby database pulls the archive logs. Thus, these file systems and volume groups must be manually added to the DB2 instance resource groups.

Configuring DB2 in a PowerHA SystemMirror cluster

These topics describe how to use PowerHA SystemMirror Smart Assist for DB2 to set up the PowerHA SystemMirror configuration for your DB2 servers.

Note: Be aware that if you must back off the configuration process or, for some reason, must retry the configuration process, you should first reset the IP label in each **$INSTHOME/sqllib/db2nodes.cfg** file to its initial home node IP label (the one that is initially set by the DB2 instance installation process).

Preparing to configure DB2 in anPowerHA SystemMirror cluster

Before you configure DB2 in a PowerHA SystemMirror cluster, there are several things you need to consider.

Before using PowerHA SystemMirror Smart Assist for DB2:

1. Make sure that the **/usr** directory has at least 5 MB of free space to run the PowerHA SystemMirror Smart Assist for DB2 utilities. It is recommended that the **/usr** directory has 20 MB of free space.
2. Synchronize your existing PowerHA SystemMirror configuration.
   a) Enter `smit hacmp`
   b) In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

3. Create a cluster snapshot to capture your PowerHA SystemMirror configuration before PowerHA SystemMirrorSmart Assist for DB2 makes changes to the PowerHA SystemMirror configuration.
4. If cluster services are running, stop them before proceeding:
   a) Enter the fastpath `smit cl_admin`
   b) Select **PowerHA SystemMirror Services** > **Stop Cluster Services** and press Enter.
   c) Select **Stop now** and press Enter.
5. Create a **$INSTHOME/.rhosts** file on the node where the volume group for the instance is currently active.

   PowerHA SystemMirror Smart Assist for DB2 uses this file during setup of the PowerHA SystemMirror resource configuration. You can delete this file after your configuration is in place.
6. Make sure that volume groups are varied on and file systems are mounted before you run the PowerHA SystemMirror Smart Assist for DB2 application.
7. Make sure that the DB2 instances for which you are configuring PowerHA SystemMirror are running to enable PowerHA SystemMirror to discover all configured resources.

Related information

Starting and stopping cluster services
Saving and restoring cluster configurations
Verifying and synchronizing a PowerHA SystemMirror cluster

PowerHA SystemMirror Smart Assist for DB2 configuration: Overview

Setting up PowerHA SystemMirror Smart Assist for DB2 configuration has many steps.

To set up a PowerHA SystemMirror Smart Assist for DB2 configuration, complete the following steps:

1. Configure volume groups and file systems on home node.
2. Mount file systems on home node.
3. Ensure that each **$INSTHOME/.rhosts** file is created and contains the service IP label and the instance name, for each instance.
4. Ensure that each **$INSTHOME/sqllib/db2nodes.cfg** file is created and contains a home node IP label that is initially set by the DB2 instance installation process.
5. Start the DB2 UDB Database Manager.
6. Configure the necessary service IP labels.
7. Configure a cluster with a standby configuration or a two-node mutual takeover cluster, or add multiple nodes and multiple DB2 instances. PowerHA SystemMirror Smart Assist stops the DB2 instances, after it includes them with all other related resources into the PowerHA SystemMirror cluster resource groups and completes the cluster configuration process. Review the contents of the created resource groups (they should contain DB2 instance-specific volume groups and the corresponding service IP label).
8. Change the service IP label to the instance service IP label in the **$INSTHOME/sqllib/db2nodes.cfg** file.

   Note: Once you perform this step, be aware that if you must go back and change the configuration or retry the configuration process, you should first perform step 4.
9. Verify and synchronize the cluster.
10. If necessary, unmount all file systems and vary off all volume groups.
11. Restart the cluster services.

Related tasks

Configuring the PowerHA SystemMirror cluster and nodes
Configure the cluster and nodes before using the Smart Assist to discover DB2 components and configure them for you.

Configuring a two-node hot standby cluster

When adding instances to the cluster, you can either add a DB2 instance that is currently running, or a DB2 instance that is stopped.

Configuring a two-node cluster with mutual takeover
You can configure a mutual takeover cluster in a single step. You configure two resource groups at a time, each containing a DB2 instance, where each node is both the primary and takeover node for a DB2 instance resource group.

Related reference

Reviewing the resource group configuration
Once you have created a cluster that contains DB2 instances, you should review your configuration.

Verifying a cluster with DB2 instances
PowerHA SystemMirror offers an extensive cluster verification function that allows you to verify all aspects of your configuration and allows you to let the system fix the errors for you at the same time.

Related information

Administration guide

Creating the PowerHA SystemMirror configuration

You have several options for setting up a PowerHA SystemMirror configuration for DB2. You can create a two-node cluster with a hot standby configuration or with a mutual takeover configuration, or you can extend an existing cluster by adding more nodes and additional DB2 instances.

Before proceeding to configure a cluster, ensure that you:

- Install AIX, PowerHA SystemMirror and DB2 filesets on both cluster nodes.
- Configure at least one DB2 UDB instance with shared volume groups on each cluster node.
- Check that the appropriate configuration entries are made in the AIX configuration files (in **/etc/services** and others), and add the DB2 service IP label to **/etc/hosts** on all nodes, or make the service IP label available via DNS.
- Configure the volume groups that contain the DB2 instance shared home directory on a volume group that is attached to both nodes. Do this for each DB2 instance that is configured.
- Turn off db2iauto. DB2 should not automatically restart on system restart. See Turning off automatic startup of DB2 instance on node reboot.
- Check that users and groups related to the DB2 instance(s) have the same User ID and Group ID on all nodes.
- Have configured PowerHA SystemMirror topology (nodes and networks) or resources (resource groups).
- If you are using the LOGRETAIN variable for the DB2 instance (to enable DB2 to save the archive logs and perform the Point in Time recovery), then the following should be set for the DB2 instance:

    LOGRETAIN = RECOVERY
    USEREXIT = ON

(USEREXIT is used to enable you to use the compiled script, **db2uext2**, to save to the disk, archive and retrieve the database log files.)

If these variables are set, you should also set the ARCHIVE_PATH, AUDIT_ERROR_PATH and RETRIEVE_LOG_PATH variables in the userprofile for the DB2 instance. Set the ARCHIVE_PATH, AUDIT_ERROR_PATH and RETRIEVE_LOG_PATH variables in such a way that they correctly point to the corresponding actual pathnames of these logs. If these variables are set correctly, the PowerHA SystemMirror Smart Assist for DB2 discovery process discovers them for the DB2 instance. This enables the script for log archives to run correctly.

- Also, see the section Planning Backups, Logging and Other DB2 Parameters. You may need to manually add scheduled backup volume groups to your resource groups, or install the backup software on all cluster nodes.

Note: The modifications to the AIX-level and DB2-level configuration files can also be corrected during the PowerHA SystemMirror synchronization phase by automatic corrective actions of the cluster verification utility, if you have not already provided the appropriate information.

Related reference

Turning off automatic startup of DB2 Instance on node reboot
Ensure that the DB2 instance is not set to automatically restart on system startup.

Configuring the PowerHA SystemMirror cluster and nodes

Configure the cluster and nodes before using the Smart Assist to discover DB2 components and configure them for you.

To configure the PowerHA SystemMirror cluster and nodes:

1. From the control node, enter `smit hacmp`
2. In SMIT, select **Cluster Nodes and Networks Configuration** > **Configure Nodes** and press Enter.

Table 32. Configure Node fields	
Field	**Value**
Cluster Name	Enter the name of the cluster.
New Nodes (via selected communication paths)	Enter the names of the nodes to add to your cluster.
Currently Configured Node(s)	Displays a list of the nodes that are currently configured. This field cannot be modified.

Discovering and configuring DB2 resources

After adding nodes, you can use the SMIT to configure PowerHA SystemMirror Smart Assist for DB2. The Smart Assist discovers the applications and prompts you for information it needs to configure them for you.

1. To use DB2 having V9.1 onwards, export a variable *DSE_INSTALL_DIR* to the installed path of DB2. From DB2 V9.1 onwards, DB2 installation images no longer uses **lpp** formats; therefore the **lslpp** command cannot be used to query DB2 installation.

   For example, if you use DB2 V9.1 and the installation path is **/opt/IBM/db2/V9.1**, then you need to export a variable called *DSE_INSTALL_DIR* as `export DSE_INSTALL_DIR=/opt/IBM/db2/V9.1`.

2. From the command line, enter `smit hacmp`.
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.

   SMIT displays a list of the applications that are installed on the cluster.

4. Select **DB2 UDB non-DPF Smart Assistant**.

   SMIT displays a list of possible configurations:

   - DB2 Mutual Takeover
   - DB2 Single Instance

5. Select one of the configurations, and press Enter.

   SMIT displays a panel for you to fill in information so the Smart Assist can configure the component.

Configuring a two-node hot standby cluster
When adding instances to the cluster, you can either add a DB2 instance that is currently running, or a DB2 instance that is stopped.

If you add an instance that is not currently running, PowerHA SystemMirror will start it as part of its normal cluster processing, when the cluster services are restarted. If you add a running DB2 instance (recommended), PowerHA SystemMirror discovers it and adds it to the configuration. PowerHA SystemMirror also temporarily stops the instance once it is added and restarts it as part of the cluster

event processing, when the cluster services are restarted. The DB2 instance is restarted as soon as the cluster services are started on the primary node for the instance and the resource group is acquired on this node.

To configure a two-node hot standby cluster with a single DB2 instance:

1. In SMIT, you selected the *DB2 Single Instance* configuration.

   An **Instance Owning Node** panel appears.

2. Select the local node name from the picklist of already defined nodes as the primary cluster node for the DB2 instance resource group and press Enter.

3. Select the DB2 instance to make highly available and press Enter.

   PowerHA SystemMirror Smart Assist for DB2 determines which of the specified entries are new communication paths and automatically adds one or more node(s), networks and IP labels to form the PowerHA SystemMirror topology configuration. After the cluster topology is added, the following screen appears.

4. Enter field values as follows:

Table 33. DB2 instance fields	
Field	**Value**
Application Name	The application name is the symbolic name for the set of resources such as application controllers, resource groups, and application monitors to make the DB2 instance highly available.
DB2 Instance Owning Node	The name of the primary owner node for the DB2 instance resource group that you selected.
Takeover Node(s)	Select a node from a list of predefined cluster nodes The takeover node(s) will acquire the DB2 instance in the event of an instance failure on the primary node. You can specify one or more takeover nodes.
	In the event of a failure of the DB2 instance, PowerHA SystemMirror uses the takeover nodes in the order in which you enter them in this field.
	The system uses the local cluster node and an additional node name provided by you to form the PowerHA SystemMirror cluster topology configuration with two nodes.
DB2 Instance Name	Press F4 to select from the picklist the name of the DB2 instance that is configured on the node with which you started (the primary owner node for the DB2 instance) and that you want to make highly available.
	Note: To appear in the picklist, the DB2 instance must be defined and its resources accessible on the remote takeover node. The local and the takeover nodes must utilize the same IDs for users and groups for each shared DB2 instance. For more information, see Planning your configuration.
DB2 Instance Database to Monitor	Specify one or more DB2 database instances that must be monitored. The monitored database instances are kept highly available.

Table 33. DB2 instance fields (continued)	
Field	**Value**
Service IP label	Specify the service IP label to be used by the DB2 instance. The service IP label must be defined in the **/etc/hosts** file on both the primary and takeover cluster nodes and cannot be a Fully Qualified Domain Name or IP address. PowerHA SystemMirror automatically adds this service IP label to the PowerHA SystemMirror resource group along with the associated DB2 resources, shared volume groups, associated DB2 instance file systems, and application monitors for the DB2 processes.

5. Press Enter. You have added a DB2 instance to a resource group in the PowerHA SystemMirror cluster that has two nodes and uses a hot standby configuration to recover the DB2 instance in case of a failure.

   Use the same procedure multiple times to extend the PowerHA SystemMirror cluster, by either adding additional DB2 instances or additional cluster nodes.

Related concepts

Planning your configuration
These topics describe the tasks you complete and the information you gather before running PowerHA SystemMirror Smart Assist for DB2. It is assumed that you are familiar with the DB2 application before you start planning for PowerHA SystemMirror Smart Assist for DB2.

Configuring a two-node cluster with mutual takeover
You can configure a mutual takeover cluster in a single step. You configure two resource groups at a time, each containing a DB2 instance, where each node is both the primary and takeover node for a DB2 instance resource group.

1. In SMIT, you select **DB2 Mutual Takeover** configuration.
2. Enter field values as follows:

Table 34. DB2 Mutual Takeover fields	
Field	**Value**
Application Name	The application name is the symbolic name for the set of resources such as application controllers, resource groups, and application monitors to make the DB2 instance highly available.
Node for First Instance	This field is already filled in with the name of the primary owner node for the first DB2 instance resource group.
First DB2 Instance Name	Specify the DB2 instance to make highly available. Press F4 to select from a list of available DB2 instances. The primary node for the specified instance is the node that appears directly before the instance name. **Note:** The first node and second node must utilize the same IDs for users and groups for each shared instance.
DB2 Instance Database to Monitor	Specify one or more DB2 database instances that must be monitored. The monitored database instances are kept highly available.

Table 34. DB2 Mutual Takeover fields (continued)	
Field	**Value**
Service IP Label	Specify the service IP label to be used by the first DB2 instance.
	The service IP label must be defined in the **/etc/hosts** file on both the local and takeover cluster nodes and cannot be a Fully Qualified Domain Name or IP address.
	PowerHA SystemMirror Smart Assist for DB2 adds this service IP label to the PowerHA SystemMirror resource group along with the associated DB2 resources, shared volume groups, associated DB2 instance file systems, and application monitoring for the DB2 processes.
Node for Second Instance	This field is already filled in with the name of the primary owner node for the second DB2 instance resource group.
Second DB2 Instance Name	Specify the second DB2 instance to make highly available. Press F4 to select from a list of available DB2 instances. The primary node for the specified instance is the node that appears directly before the instance name.
	Note: The first node and second node must utilize the same IDs for users and groups for each shared instance.
DB2 Instance Database to Monitor	Specify one or more DB2 database instances that must be monitored. The monitored database instances are kept highly available.
Service IP Label	Specify the service IP label to be used by the second DB2 instance. It must be a service IP label on a PowerHA SystemMirror network that uses IPAT via IP Aliasing.
	The service IP label must be defined in the **/etc/hosts** file on both the local and takeover cluster nodes and cannot be a Fully Qualified Domain Name or IP address.
	PowerHA SystemMirror Smart Assist for DB2 adds this service IP label to the PowerHA SystemMirror resource group along with the associated DB2 resources, shared volume groups, associated DB2 instance file systems, and application monitoring for the DB2 processes.

3. Press Enter. You have configured a two-node cluster with two DB2 instances that will use mutual takeover for keeping the DB2 instances highly available.

   Along with resource groups, PowerHA SystemMirror Smart Assist for DB2 also creates an application start and stop script for each DB2 instance and the application monitors. For more information, see Understanding the configuration for PowerHA SystemMirror start and stop scripts and Settings for application monitors.

Verifying and restarting the cluster

Once you have configured PowerHA SystemMirror, you should verify and restart the cluster.

To verify and synchronize the PowerHA SystemMirror configuration:

1. From the command line, enter `smit hacmp`
2. In SMIT, select **Custom Cluster Configuration** > **Verify and Synchronize Cluster Configuration (Advanced)** and press Enter.

To start PowerHA SystemMirror cluster services:

1. Enter the fastpath `smit cl_admin`
2. Select **PowerHA SystemMirror Services** > **Start Cluster Services** and press Enter.

Related information

Verifying and synchronizing a PowerHA SystemMirror cluster

Starting and stopping cluster services

Reviewing the resource group configuration

Once you have created a cluster that contains DB2 instances, you should review your configuration.

Do the following:

- Make sure that DB2 functions as expected.
- Review the cluster configuration from SMIT.
- Make sure that the PowerHA SystemMirror cluster is stable and running as expected.

PowerHA SystemMirror Smart Assist for DB2 creates a separate PowerHA SystemMirror resource group for each DB2 instance. The resource group includes a list of all the cluster nodes that can control the resource group, that is the list of nodes on which the DB2 instance can run, with a priority assigned to each node. The policies for the resource group are set to:

- Startup: Online on Home Node
- Fallover: Fallover to Next Priority Node in the List
- Fallback: Never Fallback.

  The fallback policy for the DB2 instance resource group ensures that the DB2 instance runs without unnecessary temporary outages, even in cases when the DB2 instance is hosted on cluster nodes other than its primary owner node.

This means that the application:

- Runs on the primary node
- Falls over to the standby node if the primary node is unavailable
- Does not fall back to the primary node when it is available again. You can manually move the resource group containing the DB2 instance to the preferred node or change the resource group policy.

Each resource group includes:

- Nodes on which the DB2 instance runs.
- The service IP addresses for the DB2 instance.
- Volume groups containing file systems associated with a specified DB2 instance available to each node in the resource group.
- PowerHA SystemMirror application servers that contain the application start and stop scripts (created by PowerHA SystemMirror Smart Assist for DB2 for the DB2 instances)
- PowerHA SystemMirror application monitors (created by PowerHA SystemMirror Smart Assist).

Related concepts

PowerHA SystemMirror configuration summary

These topics describe the PowerHA SystemMirror configuration that the PowerHA SystemMirror Smart Assist for DB2 software creates for the DB2 application.

Related tasks

Testing a DB2 instance resource group

Quickly test fallover and fallback functions in your cluster with DB2 instances by using the automated PowerHA SystemMirror cluster test tool.

Related information

Managing resource groups in a cluster

Monitoring a PowerHA SystemMirror cluster

Testing the availability of DB2 applications

Testing the availability of an application after you have it configured is optional, but is recommended. If there are problems with the configuration, the test tool gives you helpful information for making adjustments.

To test your PowerHA SystemMirror Smart Assist for DB2 configuration

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Test Your Application for Availability**and press Enter.

   SMIT displays a list of applications.
3. Select the application you want to test and press Enter.

   You are prompted to confirm the action by pressing Enter again.
4. Press Enter to continue with the test of the selected application.

   SMIT displays output to the display as the Cluster Test Tool runs.

Understanding the configuration for PowerHA SystemMirror start and stop scripts

PowerHA SystemMirror Smart Assist for DB2 uses information from the DB2 configuration to generate the start and stop scripts for PowerHA SystemMirror.

The following list describes the PowerHA SystemMirror start and stop scripts:

Application	Start Script Description	Stop Script Description
DB2	Calls **db2start** using the specified DB2 instance name.	Calls **db2stop** using the specified DB2 instance name.

It is important to realize that once a DB2 instance is added to the PowerHA SystemMirror cluster configuration, PowerHA SystemMirror manages it by starting and stopping it, if necessary.

For more information about the start and stop scripts, see the section PowerHA SystemMirror components created for DB2.

Related concepts

PowerHA SystemMirror components created for DB2
PowerHA SystemMirror Smart Assist for DB2 creates several different PowerHA SystemMirror components for the DB2 server.

Related tasks

Configuring a two-node cluster with mutual takeover
You can configure a mutual takeover cluster in a single step. You configure two resource groups at a time, each containing a DB2 instance, where each node is both the primary and takeover node for a DB2 instance resource group.

Related reference

Changing start and stop scripts

PowerHA SystemMirror Smart Assist for DB2 supplies the start and stop scripts that PowerHA SystemMirror uses to start and stop the PowerHA SystemMirror application servers for a cluster node.

Changing and testing the DB2 configuration in anPowerHA SystemMirror cluster

These topics provide information about working with the PowerHA SystemMirror configuration created by PowerHA SystemMirror Smart Assist for DB2.

Changing the DB2 instance cluster configurations

You can add a node to the cluster, and add that node to the participating nodelist of the selected DB2 instance resource group. Also, you can modify existing DB2 instance resources, such as the startup, fallover and fallback behavior of the resource group containing the DB2 instance, and modify other resource group resources. Finally, you can remove a node from the participating nodelist of the selected DB2 instance resource group.

You can create DB2 configurations while the PowerHA SystemMirror cluster services are running. When PowerHA SystemMirror creates a new resource group with the DB2 instance, if the DB2 instance is running on the node, it is stopped temporarily and then restarted, as part of the resource group. If PowerHA SystemMirror cannot stop the DB2 instance, it issues an error message.

Changing or showing the configuration of DB2 applications
Once your DB2 applications are configured, you can display the current configuration or change to a new configuration.

1. From the command line, enter smit hacmp.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications (for example, DB2 Hot Standby - Instance ABC01).
3. Select the application configured by DB2 Smart Assist to change or show and press Enter.
4. Make the changes as needed in the field(s) for the selected application and press Enter.

   See Configuring DB2 in a PowerHA SystemMirror cluster for information on the fields.

Related concepts
Configuring DB2 in a PowerHA SystemMirror cluster
These topics describe how to use PowerHA SystemMirror Smart Assist for DB2 to set up the PowerHA SystemMirror configuration for your DB2 servers.

Changing an existing DB2 instance resource group
You change the DB2 instance resource group in the same way you change a regular resource group in PowerHA SystemMirror.

Note: A resource group that contains a DB2 instance must have the following parameters and resources specified for it: an application controller, an aliased service IP label and instance-specific volume group(s). The PowerHA SystemMirror cluster verification utility issues a warning if it does not find an application controller, a service IP label or an instance-specific volume group specified for a resource group with a DB2 instance.

You can change the following for a configured resource group:

- The name of the resource group
- The nodes in the list of participating nodes
- The inter-site management policy of a resource group
- The priority of participating nodes (by changing their position in the list of participating nodes).

- Attributes of the resource group, such as a fallback timer, attributes of volume groups and file systems in the resource group, tape resources and Workload Manager classes.
- The startup, fallover and fallback policies for resource groups .

  You can change the resource group startup, fallover or fallback policy only for a resource group that has no resources in it, or after you remove all the resources from the resource group. To change the resource group's startup, fallover or fallback policies, remove the resources first. After you make the changes, add the resources manually to the resource group by selecting them from the picklists.

  You can change most of the attributes of a resource group in an active cluster without having to stop and then restart cluster services. However, to change the *name* of a resource group, you must stop and then restart the cluster to make the change part of the current cluster configuration.

To change attributes for a DB2 instance resource group:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show the Resources Associated with Your Application** and press Enter.

   A list of applications, including the DB2 instances and their associated resource groups appears.
3. Select a DB2 instance and its associated resource group from the list and press Enter.

   SMIT displays the **Change/Show Resources and Attributes for a Resource Group** for the application's resource group.
4. Change field values as needed.
5. Press Enter to change the resource group information stored in the PowerHA SystemMirror Configuration Database.
6. Return to previous SMIT panels to perform other configuration tasks.
7. Verify and synchronize the changes you made. From SMIT, select **Cluster Management** > **Cluster Services** and press Enter.

   If the Cluster Manager is running on the local node, synchronizing cluster resources triggers a dynamic reconfiguration event.

Related information
Administration guide

Removing a node from the DB2 instance resource group
You can remove one or more nodes at a time from the DB2 instance resource group.

To remove a node from the DB2 instance resource group:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show the Resources Associated with Your Application** and press Enter.

   A list of applications, including the DB2 instances and their associated resource groups appears.
3. Select a DB2 instance for which you want to remove a node and press Enter.

   SMIT displays the resource group for the chosen DB2 instance. You can remove one or more nodes at a time. If you remove all nodes from the DB2 instance resource group, the system removes the resource group from the cluster configuration.
4. Remove the node from the participating nodelist and press Enter.

Removing a DB2 Smart Assist application instance

You can remove the DB2 Smart Assist data, including thePowerHA SystemMirror resource group and associated application controller and application monitors.

The DB2 instance remains on the node; it is no longer monitored and managed by PowerHA SystemMirror.

To remove Smart Assist DB2 instance data from the cluster:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.

   SMIT displays a list of available applications
3. Select **DB2** from the list of applications.

   The list of DB2 instances and the associated resource groups appears.
4. Select the DB2 instance and resource group that you want to remove from the Smart Assist application and press Enter.

   The system removes the resource group, the application controller and the application monitor associated with the particular DB2 instance that was included in this resource group. The DB2 instance itself is not removed from the configuration.

Testing a DB2 instance resource group

Quickly test fallover and fallback functions in your cluster with DB2 instances by using the automated PowerHA SystemMirror cluster test tool.

To test your PowerHA SystemMirror Smart Assist for DB2 configuration, complete the following steps:

1. From the command line, enter `smit hacmp`.
2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Test Your Application for Availability** and press Enter.

   SMIT displays a list of applications.
3. Select the DB2 instance and resource group you want to test and press Enter.

   You are prompted to confirm the action by pressing Enter again.
4. Press Enter to continue with the test of the selected application.

   The Cluster Test Tool runs and displays status messages to the screen and stores output from the tests in the file **/var/hacmp/log/cl_testtool.log**. Messages indicate when a test starts and finishes and provide additional status information. More detailed information, especially when verbose logging is enabled, is stored in the log file that appears on the screen. Information is also logged to the **hacmp.out** file.

Related information

Administration guide

Managing users for DB2 in a PowerHA SystemMirror cluster

Use the C-SPOC (Cluster Single Point of Control) function of PowerHA SystemMirror to manage the DB2 instance users.

There are three different types of DB2 users:

• The instance owner. Often the user name is the same as the instance name, such as `db2inst1`.
• The owner or administrator of the UDF and stored procedures, such as `db2fenc1`, and associated group, `db2fgrp1`.
• The administration server user, such as `dasusr1` and the associated group `dasadm1`.

In general, DB2 relies upon several users whose group, user IDs and passwords should be synchronized across the nodes where the DB2 instance can reside.

Related information

Administration guide

Verifying a cluster with DB2 instances

PowerHA SystemMirror offers an extensive cluster verification function that allows you to verify all aspects of your configuration and allows you to let the system fix the errors for you at the same time.

The PowerHA SystemMirror Verification utility ensures that your DB2 and PowerHA SystemMirror integration goes smoothly.

Related information

Administration guide

Initial verification checks

Before allowing the addition of the DB2 instances to the PowerHA SystemMirror configuration and before making any changes to the DB2 configuration, PowerHA SystemMirror performs a set of initial checks to validate that the DB2 instance configuration on the remote nodes is a valid configuration.

PowerHA SystemMirror Smart Assist verifies the following:

- DB2 instances have a home directory for the DB2 instance primary node that resides on a shared volume group.
- The instance home volume group contains only one DB2 instance. Multiple DB2 instances in the same volume group are not supported.
- The physical disks that belong to the home volume group are accessible on all takeover nodes.
- The volume group is imported on the primary node for the DB2 instance. The volume group does not have to be imported on the takeover nodes, an automatic corrective action of the verification process imports the volume group on the takeover nodes if necessary.
- User and group names for the instance owner user are defined on all primary and takeover nodes.
- The user ID and group ID are identical on all nodes for the instance owner user and group.
- Supported DB2 version filesets are installed on the participating nodes of an instance resource group.
- DB2 DBF partition instances are not supported and if the system detects a partition instance, this instance cannot be added to the cluster.

Verification checks for storage configuration

During verification, PowerHA SystemMirror verifies the certain aspects of your storage configuration.

These aspects include:

- The DB2 instance home directory resides on shared storage.
- Shared volume groups are accessible on all nodes where a particular DB2 instance might reside in the cluster. This requires that all physical disks are defined on all participating cluster nodes.

  If the DB2 instance volume groups do not exist on a takeover node, PowerHA SystemMirror automatically imports those volume groups as part of the verification automatic corrective action. For automatic import to be successful, the takeover nodes must already share the same set of disks.

Verification checks for security and for PowerHA SystemMirror .rhosts entries

PowerHA SystemMirror ensures that the **$INSTHOME/.rhosts** file is secure for the DB2 instances that are properly configured by running the cluster verification process. The automatic corrective action of the PowerHA SystemMirror verification utility notifies you of security errors and lets you correct them at the time they are discovered.

In addition, verification ensures that the DB2 instance home directory **$INSTHOME/.rhosts** file contains an entry for the service IP aliased label used for DB2 instance communication to clients and other tiers for every DB2 instance that PowerHA SystemMirror monitors. Typically, the preferred entry in **.rhosts** may appear as follows:

```
<cluster IP address> <DB2 instance owner user name>
```

Or as:

```
+ <DB2 instance owner user name>
```

In general, although the latter entry should only be used in trusted environments, the PowerHA SystemMirror verification does not flag an error if the **all hosts** entry appears in the **.rhosts** file, but issues a warning indicating that all remote hosts with the user name DB2 Instance owner name are granted access to the local host.

An automatic corrective action of the verification process adds the appropriate entry if the field is missing from **$INSTHOME/.rhosts**.

Other verification checks

PowerHA SystemMirror Smart Assist verifies certain issues and provides an automatic corrective action to fix the errors.

These issues include:

- Each node that participates in a DB2 instance resource group has the Fault Monitor Coordinator (FMC) turned off. If the Fault Monitor coordinator is not disabled, an automatic corrective action prompts you to disable it.
- The port numbers are added to the **/etc/services** file on the primary cluster node for the DB2 instance. An automatic corrective action of the verification process adds the necessary port numbers. See Planning DB2 installation and configuration.
- The DB2 instances are not set to automatic start on system reboot. An automatic corrective action of the verification process prompts you to disable automatic startup of the DB2 instance managed by PowerHA SystemMirror.

Related reference
Planning DB2 installation and configuration
During installation and configuration for DB2, you need to consider some aspects.

Changing start and stop scripts

PowerHA SystemMirror Smart Assist for DB2 supplies the start and stop scripts that PowerHA SystemMirror uses to start and stop the PowerHA SystemMirror application servers for a cluster node.

Do not change to the PowerHA SystemMirror start and stop scripts. If you need to change how an application starts or stops, make changes to the application start or stop script that is inherent to the application (outside of PowerHA SystemMirror).

For more information about the PowerHA SystemMirror start and stop scripts, see the section Understanding the configuration for PowerHA SystemMirror start and stop scripts.

PowerHA SystemMirror configuration summary

These topics describe the PowerHA SystemMirror configuration that the PowerHA SystemMirror Smart Assist for DB2 software creates for the DB2 application.

Naming convention

The names of the components created use a standard naming convention to make it easy to identify the various PowerHA SystemMirror components.

The following list shows the naming conventions used:

Table 35. Naming conventions	
Short name	**Full name**
db2	DB2 server
rg	Resource group

Table 35. Naming conventions (continued)	
Short name	**Full name**
as	Application controller

PowerHA SystemMirror components created for DB2

PowerHA SystemMirror Smart Assist for DB2 creates several different PowerHA SystemMirror components for the DB2 server.

These components include:

PowerHA SystemMirror Component	**Name**
Resource group	**InstanceName_ResourceGroup** where *InstanceName* is the name of the instance of the DB2 database
PowerHA SystemMirror application server	**InstanceName_ApplicationServer** where *InstanceName* is the name of the instance of the DB2 database.
PowerHA SystemMirror custom monitor	***InstanceName*_SQLMonitor_*DatabaseName*** where *InstanceName* is the name of the DB2 database instance. The custom monitor file is located in the directory **/usr/es/sbin/cluster/sa/db2/sbin**
Start script	**cl_db2start** This file is located in the directory **/usr/es/sbin/cluster/sa/db2/sbin**
Stop script	**cl_db2stop** This file is located in the directory**/usr/es/sbin/cluster/sa/db2/sbin**

Settings for application monitors

PowerHA SystemMirror Smart Assist for DB2 configures a custom monitor and a process monitor for the DB2 application that you configure PowerHA SystemMirror in your environment.

Related tasks

Configuring a two-node cluster with mutual takeover
You can configure a mutual takeover cluster in a single step. You configure two resource groups at a time, each containing a DB2 instance, where each node is both the primary and takeover node for a DB2 instance resource group.

Custom monitor settings
Custom application monitoring checks the health of an application with a custom monitor method at user-specified polling intervals.

The following table lists the default settings for the custom monitor:

Customer Mmnitor setting	**Default value**
Name	***InstanceName*_SQLMonitor_*DatabaseName***
Application Controllers to Monitor	*InstanceName*_ApplicationController
Method	**/usr/es/sbin/cluster/sa/db2/sbin/cl_db2cmon -i <InstanceName> -A <ApplicationName> -d <DatabaseToMonitor>**

Customer Mmnitor setting	Default value
Mode	Long-running monitoring
Interval	120 sec.
Hung Monitor Signal	9
Stabilization Interval	240 sec.
Restart Count	3
Restart Interval	1440 sec.
Action on Application Failure	Fallover
Cleanup Method	*/usr/es/sbin/cluster/sa/db2/sbin/cl_db2stop InstanceName*
Restart Method	*/usr/es/sbin/cluster/sa/db2/sbin/cl_db2start InstanceName*

Note: If an application monitor is configured, after successful migration, the name of the application monitor is displayed as `InstanceName_SQLMonitor`. If multiple databases are monitored, the `DatabaseName` is appended to the name of the application monitor, `InstanceName_SQLMonitor` to differentiate between multiple application monitors.

You can change the default values set for an application monitor by using SMIT.

Related information

Administration guide

Process monitor settings

The process monitor for DB2 in a PowerHA SystemMirror cluster determines whether the parent process for the DB2 instance, db2sysc, is still running for the DB2 instance. Should this process terminate, the DB2 instance attempts to run a cleanup script and to restart up to three times before the DB2 instance falls over to the next node. After three attempts, the application controller associated with this monitor falls over to another node that participates in this resource group.

The following table lists the default settings for the process monitor:

Name	*InstanceName*_**ProcessMonitor**
Monitor Mode	Long-running monitoring
Process to Monitor	**db2sysc**
Process Owner	Instance Owner
Instance Count	1
Stabilization Interval	240 sec.
Restart Count	3
Restart Interval	1440 sec.
Action on Application Failure	Fallover
Cleanup Method	**/usr/es/sbin/cluster/sa/db2/sbin/cl_db2stop***InstanceName*
Restart Method	**/usr/es/sbin/cluster/sa/db2/sbin/cl_db2start***InstanceName*

Smart Assist for FileNet P8

IBM FileNet P8 Platform offers you enterprise-level scalability and flexibility to handle the most demanding content challenges, the most complex business processes, and integration with existing

systems in your environment. The main reason for using Smart Assist for FileNet P8 is to change all your unstructured data into structured data using Enterprise Content Management (ECM).

Smart Assist for FileNet P8 concepts

It is important that you have at least a basic knowledge of Smart Assist before you start configuring and using it.

The IBM FileNet P8 Platform is the unified enterprise foundation for the integrated IBM FileNet P8 products. It provides the core components that the add-onFileNet P8 products seamless interoperate, sharing a common information infrastructure and associated security model, taxonomy, and set of Application Programming Interfaces (APIs). FileNet P8 applications leverage the Java™ Enterprise Edition application model to build multi-tier applications that deliver the scalability, accessibility, and administration that enterprise applications require.

The core of FileNet P8 Platform is provided by the following three products:

- IBM FileNet Content Manager
- IBM FileNet Business Process Manager
- IBM FileNet Records Manager

Using FileNet P8 product provides you with the following benefits:

- Combines an enterprise content management reference architecture and core enterprise platform with comprehensive business process management and compliance capabilities.
- Includes a comprehensive set of content and process management business services that can be consumed and deployed in a service-oriented architecture.
- Supports a flexible API for Java, Microsoft .NET, and XML Web services application development for a rich and interactive user experience that is easily customized.
- Delivers data center manageability and support for enterprise system management tools and enterprise scalability and flexible system deployment in clustered and highly-available environments.
- Provides distributed content caching and network optimization features that provide optimal performance across geographically distributed sites and wide area networks.
- Includes multilingual system capabilities for decentralized, federated system architectures, advanced security services, comprehensive auditing, and a standards-based authentication framework.

Enterprise Content Management capabilities

You can use Enterprise Content Management (ECM) to manage, capture, organize, secure, search, and store unstructured content and semi-structured content within your environment.

ECM is not a product or a specific solution, it is a disciplined framework for identifying the sources and consumers of your organization's content. You apply controls to the contents creation, use, and handling, and make the content available to applications and users who need access it. The objective of ECM is to ensure that access to the content that is being shared across an organization environment is timely, accurate, and secure, and provides the required processes to execute key functions in support of strategic business goals. ECM is about empowering employees at all levels of an organization to make the proper decision at the correct time.

ECM represents an evolution and convergence of a variety of different technologies that include elements of document management, imaging, workflow, collaboration, knowledge management, and compliance. From a functional perspective, there are a number of fundamental capabilities that are necessary to support effective managing and sharing of content:

- Flexible authoring capabilities using common desktop authoring tools
- Support for common file formats
- Ability to allow or restrict specific access and action permissions
- Full text indexing of unstructured content and the ability to search on the content and its metadata

- Automated classification of existing and new content
- System-wide audit and tracking capability
- Versioning of content with check in and check out feature
- Integrated security and access management
- Open interfaces to integrate with other applications and systems and deliver highly-specialized applications
- Content transformation
- Enterprise-level availability and scalability
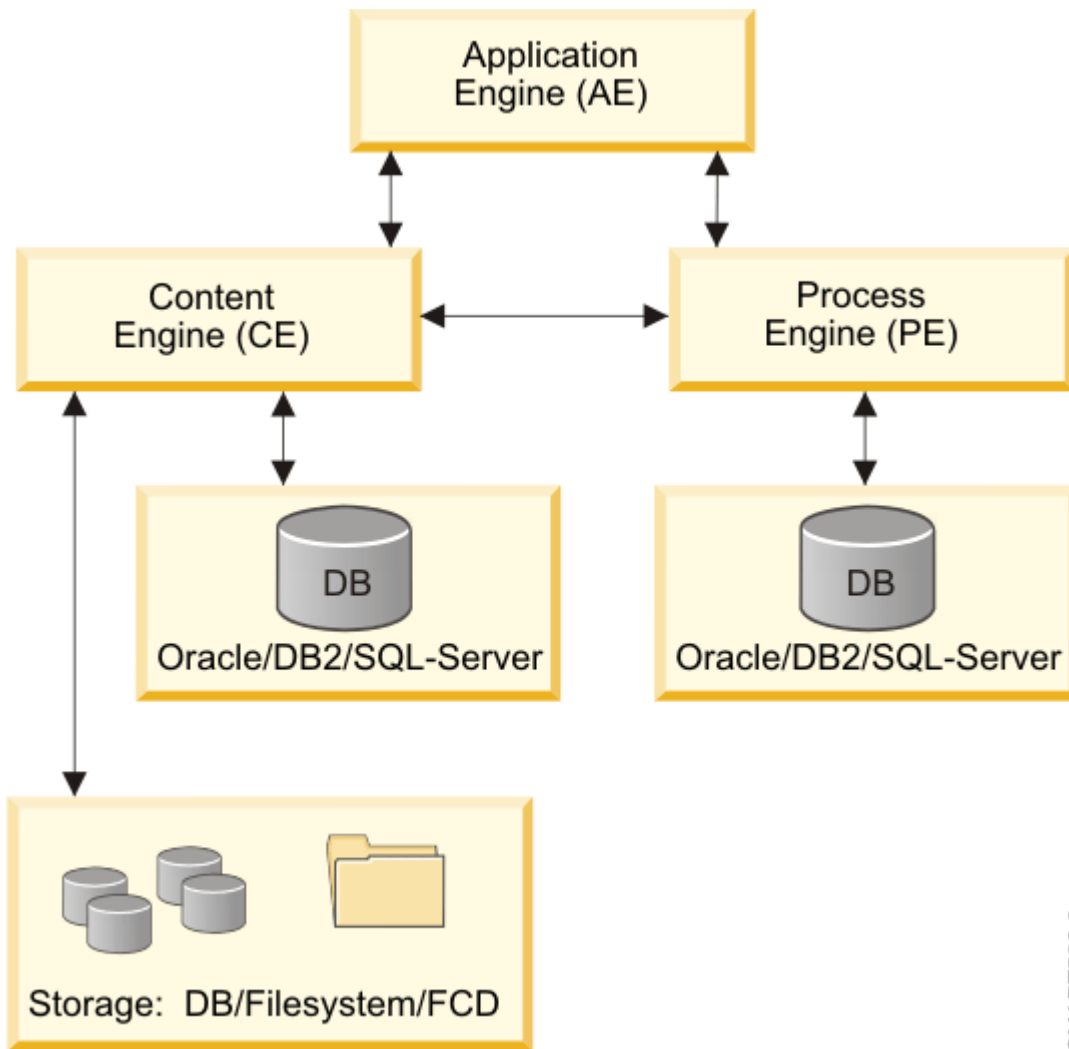
FileNet P8 architecture

FileNet P8 is built from the content engine, process engine, and application engine.

The main components that build the FileNet P8 core products are the following engines:

- Content Engine
- Process Engine
- Application Engine

Each engine consists of a collection of services and applications that perform well-defined sets of services and tasks.

The following figure illustrates the FileNet P8 architecture and how each engine interacts with each other and databases.

Figure 7. FileNet P8 architecture

Content Engine

Using FileNet P8 suite of products the content of each document is stored and managed by the Content Engine server. The Content Engine is implemented as a J2EE application, and so it runs within a J2EE application server. The Content Engine supports WebSphere, WebLogic, and JBoss application servers. The properties associated with each document comprise the document's metadata. Typical metadata properties include: creator of the document, creation time of the document, and the type of the document. The metadata is stored in a database that is known as the document catalog. Content Engine supports DB2, Oracle, and SQL Server databases. Searching for a particular document consists of querying the Content Engine database, and then retrieving the content corresponding to the matching documents

More than one piece of content, which is called a content element, can be associated with a single document. The content elements can be stored in any of the following locations:

- Database
- Conventional file system
- Fixed-content device

Although content can be stored in the Content Engine database, it is not recommended that you use this configuration, because the database can become too large and therefore difficult to manage if it holds all the content. To store content you should use either the file system or a fixed content device. If a file

system is used, it is most often stored on a network-attached storage (NAS) or storage-attached network (SAN) device. Implementing a Redundant Array of Independent Disks (RAID) system provides both higher performance and high availability of the data. Fixed content devices typically include specialized storage subsystems that meet certain compliance requirements, such as the ability to guarantee that content is never modified and that it cannot be deleted until a specified date in the future. The Content Engine supports user-extensible document classes, allowing users to define new types of documents that can be stored and what additional metadata these classes of documents will maintain.

The Content Engine also supports event processing, that is, the ability to perform a user-defined action whenever a chosen event happens, such as creating, deleting, checking in, and checking out documents. Security can be configured on document classes or on an individual document using an Access Control List (ACL), allowing the owner of a document to define precisely who can access or modify the document. Content can be versioned, so that revisions to the content of the document can be checked into the Content Engine, and the Content Engine maintains a list of all these versions over time. Documents within Content Engine can be arranged hierarchically by filing them into one or more folders.

The Content Engine uses any of a number of supported Lightweight Directory Access Protocol (LDAP) servers to perform user authentication. Using an LDAP server simplifies the installation and administration of the Content Engine system, because most corporations use an LDAP system for maintaining their user IDs and passwords. The Content Engine caches the responses from the LDAP server (keeps a copy for a period of time), which reduces the number of LDAP queries and reduces future response times.

The Content Engine uses TCP/IP to carry communication using the EJB communication protocol.

To use the Content Engine you must have setup one of following types of databases:

- DB2
- Oracle
- SQL

Process Engine

The Process Engine is a C++ based application that provides an enterprise-wide process management platform on which to build and deliver enterprise applications. The Process Engine allows you to create, modify, and manage automated business processes. The Process Engine provides software services, such as process execution and routing, integration of external rules engines, process analysis, and process simulation. These processes can be performed by applications, enterprise users, or external users, such as partners and clients. Processes run inside of an isolated region that acts as an individual processing space. The Process Engine uses the Process Engine database in which all process-related data is stored.

To use the Process Engine you must have setup one of following types of databases:

- DB2
- Oracle
- SQL

Process Engine uses Content Engine for user authentication, again simplifying its installation and administration. It also uses TCP/IP to carry communication using the IIOP communication protocol.

Application Engine

The Application Engine hosts the Workplace Web application, Workplace Java applets, and Application Programming Interfaces (APIs). It is the presentation tier for both content and process. The Application Engine also handles user authentication against the directory service. An Application Engine consists of an application server with one or more deployed applications.

Clients of Application Engine are Web browsers. They use the HTTP or HTTPS protocols in connecting to Application Engine. As a client of Content Engine and Process Engine, Application Engine uses their client protocols to connect to them.

Related tasks

Adding a Content Engine

To configure FileNet P8 correctly you need to add a Content Engine to your PowerHA SystemMirror environment.

Smart Assist for FileNet P8 planning

To implement FileNet P8 you must have a minimum of 2 nodes with an availability of minimum 3 shared volume groups.

FileNet P8 supports a high availability configuration for the following components:

- Content Engine (if running under Websphere Application Server)
- Application Engine (if running under Websphere Application Server)
- Process Engine
- Database (you must use a DB2 database)

Smart Assist for FileNet P8 does not support storage. It is your own responsibility to make the storage highly available.

Software Requirements

To view what versions of FileNet P8 are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Service IP labels

Before you deploy FileNet P8 you need to know the following service IP labels:

- One service IP label if the database for content and process engine is deployed under same DB2 instance.
- Two service IP labels if the database for content and process engine are deployed under different DB2 instance.
- One service IP label for the Process Engine.
- One service IP label if content and application engine is deployed under same Websphere Application Server.
- Two service IP labels if content and application engine is deployed under different Websphere Application Server.

Installing FileNet P8 for Smart Assist

If you reviewed the planning information you are now ready to install FileNet P8 for Smart Assist.

To install FileNet P8, complete the following steps:

1. Install DB2 as a single instance. For information on how to install DB2 as a single instance, see "Installing PowerHA SystemMirror Smart Assist" on page 2
2. Install WebSphere Application Server. For information on how to install WebSphere Application Server, see "Installing PowerHA SystemMirror Smart Assist" on page 2
3. Install and deploy a Content Engine on the WebSphere Application Server. If you want to install a Content Engine on a volume group different than the one where the WebSphere Application Server is installed it has to be a shared volume group. For information on how to install and deploy a Content Engine, see Installing a Content Engine
4. Install and deploy a highly available Process Engine on the WebSphere Application Server. For information on how to install and deploy a highly available Process Engine, see Installing a highly available Process Engine

5. Install and deploy a Application Engine on the WebSphere Application Server. If you want to install a Application Engine on a volume group different than the one where the WebSphere Application Server is installed it has to be a shared volume group. For information on how to install and deploy a Application Engine, see Installing a Application Engine

6. Install Smart Assist for FileNet P8

Example

The following figure is an example that explains how to set up a basic two node environment.
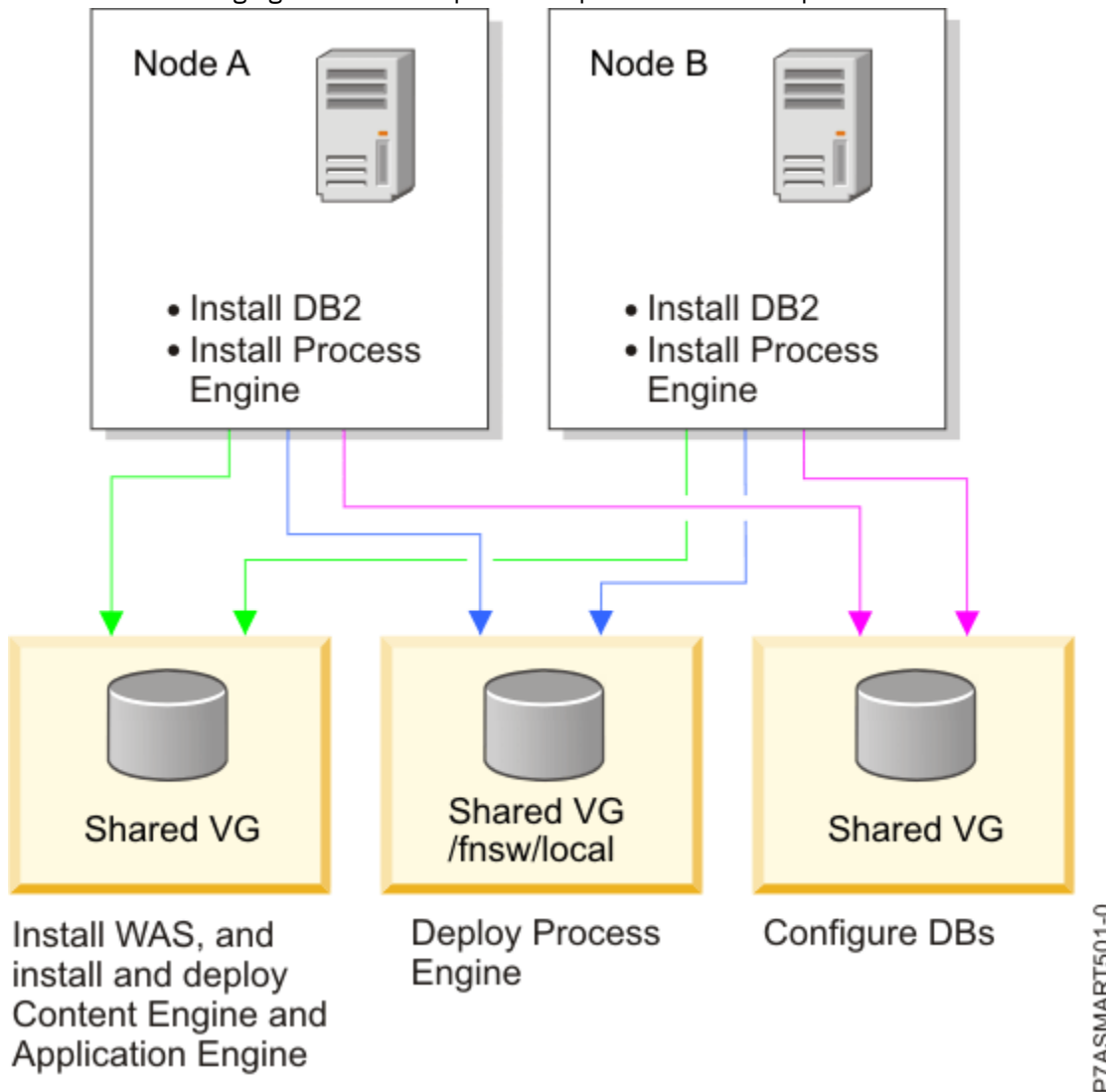

Figure 8. Example of two node installation and configuration

Even if you have a valid installation and configuration you can have problems. For example, you have a two node cluster with two shared volume groups and WebSphere Application Server (WAS) is installed on two shared volume groups that are used to deploy the Content Engine and the Application Engine on individual shared volume group. This example configuration is valid, however, you would need to resolve WAS port conflicts because WAS is installed on two shared volume groups. If you started WAS on same node then both instances of WAS will start listening on same port which would lead to problems. You would need to resolve these types of problems with underlying software products we would be using according to the setup and configuration.

Configuring Smart Assist for FileNet P8

A cluster has to be configured correctly before initiating Smart Assist for FileNet P8, otherwise you will be automatically redirected to the 2-node cluster configuration assistant interface to configure a 2-node cluster.

You must have all components active in your environment before initiating Smart Assist for FileNet P8.

If you have enabled security for a WebSphere Application Server cell, you must supply authentication information to communicate with other servers. You must update the authentication information in the <connector type>.client.props file according to the connector type being used. For example, if you use a SOAP connector, set the following properties in the soap.client.props file with appropriate values:

```
com.ibm.SOAP.securityEnabled=true
com.ibm.SOAP.loginUserid=
com.ibm.SOAP.loginPassword=
```

Adding a Content Engine

To configure FileNet P8 correctly you need to add a Content Engine to your PowerHA SystemMirror environment.

Before you can add a Content Engine you must verify that the volume group that stores the data for the Content Engine and the WebSphere Application Server (WAS) is varied on and the associated file systems mounted.

To add a Content Engine, complete the following steps:

1. If the shared volume group used for the Content Engine installation is not the WAS installed one, from the command line export the variable CE_INSTALL_PATH to the path where the Content Engine is installed. For example, CE_INSTALL_PATH=<xxxx> where xxxx is the location where the Content Engine is installed.
2. From the command line enter `smit sysmirror`.
3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
4. From the list of applications, select **FileNet P8 Smart Assistant**, and press Enter.
5. Select **Automatic Discovery and Configuration** > **Content Engine**, and press Enter.
6. Select the Cell name with the Node name for WebSphere Application Server where the Content Engine is deployed.
7. From the Add Content Engine configuration to the Cluster panel, complete the following fields:

Table 36. Content Engine fields

Fields	Values
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the WebSphere Application Server component.
WebSphere Application Server Cell	The name of the WebSphere Application Server cell. This field can not be modified.
WebSphere Application Server Node	The name of the WebSphere node. This field can not be modified
Application Servers	A list of the application servers that run on the node. This field can not be modified.
Primary Node	The name of the node where the WebSphere Application Server is installed. This field cannot be modified.

Table 36. Content Engine fields (continued)	
Fields	**Values**
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
Service Interface	Enter the Service Interface for Content Engine. The default for this field is the HTTP Transports Hosts of the application server and uses IPAT via IP Aliasing.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address.
	This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

8. Verify the fields are correct and press Enter.

Smart Assist for FileNet P8 stops the WebSphere Application Server, where the Content Engine is deployed, because the Content Engine is now managed by PowerHA SystemMirror. The IP address for the WebSphere Application Server is changed to setting you specified. Do not stop or start the WebSphere Application Server manually; use the SMIT interface to complete these tasks.

Related concepts

FileNet P8 architecture
FileNet P8 is built from the content engine, process engine, and application engine.

Adding a Process Engine

Before you can add a Process Engine you must verify that the volume group that has configuration for Process Engine is varied on and the associated file systems mounted.

To add a Process Engine, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. In SMIT, select **Initialization and Standard Configuration** > **Configuration Assistants** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **FileNet P8 Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **Process Engine** and press Enter.
5. Select the Process Engine user that you want to add and press Enter.
6. From the Add Process Engine configuration to the Cluster panel, complete the following fields:

Table 37. Process Engine fields	
Fields	**Values**
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the Process Engine component.

Table 37. Process Engine fields (continued)	
Fields	**Values**
Primary Node	This is the primary node to host the Process Engine resource group. This field is automatically populated by Smart Assist after discovering the node where process engine was up and running. This field can be modified by using a pick list.
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
Process Engine User Name	Displays the user name you selected in Step 5. This field can not be modified.
Process Engine Configuration VG	Displays the volume group discovered by Smart Assist. This volume group is where the Process Engine configuration data is stored. You can not modify this field.
Service Interface	Enter or select from the list the service IP label or service IP address the Process Engine uses to establish communication with its clients using IPAT with IP Aliasing.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address.
	This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

7. Verify the fields are correct and press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

Smart Assist for FileNet P8 stops the Process Engine because it is now managed by PowerHA SystemMirror. FileNet P8 also modifies the IP address of the Process Engine to the setting you specified.

Adding an Application Engine

Before you can add a Application Engine you must verify that the volume group that stores the data for Application Engine and WebSphere Application Server is varied on and the associated file systems mounted.

To add a Application Engine, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. In SMIT, select **Initialization and Standard Configuration** > **Configuration Assistants** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **FileNet P8 Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **Application Engine** and press Enter.
5. Select the Cell name with the Node name for the WebSphere Application Server where the Application Engine is deployed.

6. From the Add Application Engine configuration to the Cluster panel enter the following information for the fields:

Table 38. Application Engine fields

Fields	Values
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the WebSphere Application Server component.
WebSphere Application Server Cell	The name of the WebSphere Application Server cell. This field can not be modified.
WebSphere Application Server Node	The name of the WebSphere node. This field can not be modified
Application Servers	A list of the application servers that run on the node. This field can not be modified.
Primary Node	The name of the node where the WebSphere Application Server is installed. This field cannot be modified.
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
Service Interface	Enter the Service Interface for the Application Engine.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address.
	This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

7. Verify the fields are correct and press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

FileNet P8 stops the WebSphere Application Server, where the Application Engine is deployed because it is now managed by PowerHA SystemMirror.

Adding a database for the Content Engine

Once you have set up your Content Engine you need to add a database to hold all your Content Engine data.

Before you can add a database for the Content Engine you must verify that the volume group that stores the data for the databases under the required DB2 instance is varied on, that the associated file systems are mounted, and that the DB2 instance is up and running.

To add a database for the Content Engine, complete the following steps:

1. If the shared volume group used for the Content Engine installation is not the WAS installed one, from the command line export the variable CE_INSTALL_PATH to the path where the Content Engine is installed. For example, CE_INSTALL_PATH=<xxxx> where xxxx is the location where the Content Engine is installed.
2. From the command line enter `smit sysmirror`.

3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
4. From the list of applications, select **FileNet P8 Smart Assistant** and press Enter.
5. Select **Automatic Discovery and Configuration** > **DB for Content Engine** and press Enter.
6. Select the DB2 instance that you want to add your PowerHA SystemMirror cluster and press Enter.
7. From the Add DB2 single instance to the Cluster panel enter the following information for the fields:

Table 39. DB2 interface fields

Fields	Values
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the DB2 instance.
DB2 Instance Owning Node	Enter the name of the primary owner node for the DB2 instance resource group that you selected.
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
DB2 Instance Name	Displays the name of the DB2 instance you select in Step 5.
DB2 Instance Database to Monitor	Specify the DB2 database to monitor in order for PowerHA SystemMirror to determine whether the DB2 instance is unavailable.
Service IP Label	Specify the service IP label to be used by DB2 instance.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

8. Verify the fields are correct and press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

Smart Assist for FileNet P8 stops the DB2 instance responsible for the database for Content Engine because it is now managed by PowerHA SystemMirror. Do not stop or start the DB2 instance that is responsible for the database for Content Engine manually. You must use the SMIT interface to start or stop the DB2 instance.

Adding a database for the Process Engine

After you set up your Process Engine you need to add a database to hold all your Process Engine data.

Before you can add a database for the Process Engine you must verify that the volume group that stores the data for the databases under the required DB2 instance is varied on, that the associated file systems are mounted, and that the DB2 instance is up and running.

To add a database for the Process Engine, complete the following steps:

1. If the shared volume group used for the Content Engine installation is not the WAS installed one, from the command line export the variable CE_INSTALL_PATH to the path where the Content Engine is

installed. For example, CE_INSTALL_PATH=<xxxx> where xxxx is the location where the Content Engine is installed.

2. From the command line enter `smit sysmirror`.

3. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.

4. From the list of applications, select **FileNet P8 Smart Assistant** and press Enter.

5. Select **Automatic Discovery and Configuration** > **DB for Process Engine** and press Enter.

6. Select the DB2 instance that you want to add your PowerHA SystemMirror cluster and press Enter.

7. From the Add DB2 single instance to the Cluster panel enter the following information for the fields:

Table 40. DB2 interface fields

Fields	Values
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the DB2 instance.
DB2 Instance Owning Node	Enter the name of the primary owner node for the DB2 instance resource group that you selected.
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
DB2 Instance Name	Displays the name of the DB2 instance you select in Step 5.
DB2 Instance Database to Monitor	Specify the DB2 database to monitor in order for PowerHA SystemMirror to determine whether the DB2 instance is unavailable.
Service IP Label	Specify the service IP label to be used by DB2 instance.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

8. Verify the fields are correct and press Enter to add this information to the PowerHA SystemMirror Configuration Database on the local node.

Smart Assist for FileNet P8 stops the DB2 instance responsible for the database for the Process Engine because it is now managed by PowerHA SystemMirror. Do not stop or start the DB2 instance that is responsible for the database for the Process Engine manually. You must use the SMIT interface to start or stop the DB2 instance.

Manually configure FileNet P8

You can use a xml file to configure FileNet P8.

To manually configure FileNet P8, complete the following steps:

1. From the command line enter `smit sysmirror`.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.

3. From the list of applications, select **FileNet P8 Smart Assistant** > **Manual Configuration** and press Enter.

4. In the File Name field specify the path where the xml file is located and press Enter. You can find a sample xml here, `/usr/es/sbin/cluster/sa/filenet/config/cl_filenet_manual_config.xml`.

   Note: You can not modify the Smart Assist ID field. This ID is automatically generated.

Changing FileNet P8 PowerHA SystemMirror resources

Once FileNet P8 components are configured as a PowerHA SystemMirror resource, you can change or display the resources that are being used.

To change the FileNet P8 PowerHA SystemMirror resources, complete the following steps:

1. From the command line enter `smit sysmirror`.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

3. Select the application you want to change and press Enter.

4. Make the changes as needed in the fields for the selected application and press Enter. You can not change the Application Name field.

Changing resources associated with your application

You can add resources such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, create separate resource groups for each set.

To change resources associated with your application, complete the following steps:

1. From the command line enter `smit sysmirror`.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.

3. Select the application you want to change and press Enter.

4. Make the changes as needed in the fields for the selected application and press Enter. You can not change the Application Name field.

Related information

Adding resources and attributes to resource groups

Removing an application

When you remove an application from the list of configured FileNet P8 applications, PowerHA SystemMirror stops managing the availability of the resources you remove because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line enter `smit sysmirror`.

2. In SMIT, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.

3. From the list select the application you want to remove and press Enter.

4. From the list select the resource you want to remove and press Enter.

5. Press Enter to remove the resource from the PowerHA SystemMirror Configuration Database on the local node.

   Note: Removing a resource an from a application removes all analogous resources from PowerHA SystemMirror, including application controllers and monitors, service IP labels, and resource groups.

Smart Assist for FileNet P8 start and stop scripts

Smart Assist for FileNet P8 uses information from an application configuration to generate the start and stop scripts for PowerHA SystemMirror.

The following table describes what the PowerHA SystemMirror start and stop scripts do:

Table 41. Smart Assist for FileNet P8 start and stop scripts

Engine	Start Script Description	Stop Script Description
Content Engine	Starts the WebSphere Application Server by calling the `startServer.sh` script.	Stops the WebSphere Application Server by calling the `stopServer.sh` script.
Application Engine	Starts the WebSphere Application Server by calling the `startServer.sh` script.	Stops the WebSphere Application Server by calling the `stopServer.sh` script.
Process Engine	Starts by calling the `initfnsw -y start` command using the `fnsw` user.	Stops by calling the following commands using the `fnsw` user: • `initfnsw -y stop` • `killfnsw -DayS` • `killfnsw -r`
Database for Content Engine	Starts by calling the `db2start` script using the specific DB2 instance name.	Stops by calling the `db2stop` script using the specific DB2 instance name.
Database for Process Engine	Starts by calling the `db2start` script using the specific DB2 instance name.	Stops by calling the `db2stop` script using the specific DB2 instance name.

Smart Assist for FileNet P8 resources

Smart Assist for FileNet P8 uses standard naming convention to make it easy to identify various PowerHA SystemMirror resources.

The PowerHA SystemMirror resources created for the Content Engine and the Application Engine are same as the Smart Assist for WebSphere Application Server resource. For more information about WebSphere Application Server resource, see .

The PowerHA SystemMirror resources created for the Database Content Engine and the Database Process Engine are same as the Smart Assist for DB2 resource. For more information about DB2 resources, see .

PowerHA SystemMirror resources created for the Process Engine

The following table lists resources created for the Process Engine:

Table 42. Process Engine resources

PowerHA SystemMirror resources	Name
Resource Group	ProcessEngine_*PE_USER_NAME*_ResourceGroup, where *PE_USER_NAME* is the Process Engine user.

Table 42. Process Engine resources (continued)	
PowerHA SystemMirror resources	**Name**
Application Server	*PE_USER_NAME_*ApplicationServer, where *PE_USER_NAME* is the Process Engine user.
Custom Monitor	*PE_USER_NAME_SQLMonitor,* where *PE_USER_NAME* is the Process Engine user. The relevant script file **cl_filenet_process_engine_monitor** is located in the /usr/es/sbin/cluster/sa/filenet/ sbin directory.
Process Monitor	*PE_USER_NAME_*ProcessMonitor, where *PE_USER_NAME* is the Process Engine user.
Start Script	The relevant script file **cl_filenet_process_engine_start** is located in the /usr/es/sbin/cluster/sa/filenet/ sbin directory.
Stop Script	The relevant script file **cl_filenet_process_engine_stop** is located in the /usr/es/sbin/cluster/sa/filenet/ sbin directory.

Setting for application monitors

Smart Assist for FileNet P8 configures a custom monitor and a process monitor for the FileNet P8 applications that you configure in your PowerHA SystemMirror environment.

Custom application monitors

Custom application monitoring checks the health of an application with a custom monitor method at user-specified polling intervals.

The following table lists the default settings for the custom monitor:

Table 43. Custom application monitor default settings	
Field	**Value**
Name	*PE_USER_NAME_*SQLMonitor, where *PE_USER_NAME* is the Process Engine user.
Application Servers to Monitor	*PE_USER_NAME_*ApplicationServer, where *PE_USER_NAME* is the Process Engine user.
Monitor Method	/usr/es/sbin/cluster/sa/filenet/sbin/ cl_filenet_process_engine_monitor -u <*PE_USER_NAME*>, where *PE_USER_NAME* is the Process Engine user.
Mode	Long-running monitoring
Interval	120 sec
Hung Monitor Signal	9
Stabilization Interval	240 sec
Restart Count	3
Restart Interval	1440 sec

Table 43. Custom application monitor default settings (continued)	
Field	**Value**
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/filenet/sbin/ cl_filenet_process_engine_stop -u *<PE_USER_NAME>*, where *PE_USER_NAME* is the Process Engine user.
Restart Method	/usr/es/sbin/cluster/sa/filenet/sbin/ cl_filenet_process_engine_start -u *<PE_USER_NAME>*, where *PE_USER_NAME* is the Process Engine user.

Process monitor settings

The process monitor for the Process Engine in a PowerHA SystemMirror cluster determines whether the processes for the Process Engine, TM_daemon, vworbbroker and vwior are still running.

If the process monitor terminates, PowerHA SystemMirror attempts to run a cleanup script and to restart according to the settings listed in following table:

Table 44. Process monitor default settings	
Field	**Value**
Name	*PE_USER_NAME*_ProcessMonitor, where *PE_USER_NAME* is the Process Engine user.
Application Servers to Monitor	*PE_USER_NAME*_ApplicationServer, where *PE_USER_NAME* is the Process Engine user.
Processes to Monitor	TM_daemon, vworbbroker, vwior.
Process Owner	Process engine user name
Mode	Long-running monitoring
Instance Count	1
Stabilization Interval	240 sec
Restart Count	3
Restart Interval	1440 sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/filenet/sbin/ cl_filenet_process_engine_stop -u *<PE_USER_NAME>*, where *PE_USER_NAME* is the Process Engine user.
Restart Method	/usr/es/sbin/cluster/sa/filenet/sbin/ cl_filenet_process_engine_start -u *<PE_USER_NAME>*, where *PE_USER_NAME* is the Process Engine user.

Troubleshooting Smart Assist for FileNet P8

To successfully troubleshoot Smart Assist for FileNet P8 it is important to know where the different types of data are stored.

Smart Assist for FileNet P8 stores configuration data in the HACMPsa_metadata PowerHA SystemMirror Configuration Database (ODM).

Smart Assist for WebSphere log files are stored in different locations depending on the type of engine. The following table displays where the log files are stored:

Table 45. Smart Assist for WebSphere log file locations

Engine	Log files location
Application engine	/var/hacmp/log/wassa.log
Content engine	/var/hacmp/log/wassa.log
Process engine	/var/hacmp/log/filenetsa.log
Database for content engine	/var/hacmp/log/db2sa.log
Database for process engine	/var/hacmp/log/db2sa.log

If you are having trouble with your resources created by Smart Assist for WebSphere, complete the following steps:

1. View the log files in the table.
2. Examine the SMIT processes that had been invoked from the user interface by viewing the **smit.log** file.
3. Review messages related to detection, migration, termination and execution of a PowerHA SystemMirrorcluster for an application by viewing the **hacmp.out** file.
4. Verify and synchronize your cluster to make sure all your clusters are using the latest configuration settings.
5. Remove the resource configuration that Smart Assist for WebSphere created.

Related information

Troubleshooting PowerHA SystemMirror

Smart Assist for Lotus Domino Server

Smart Assist for Lotus Domino Server allows you to automatically configure PowerHA SystemMirror where Lotus Domino is already configured. There is a cluster present in lotus domino server. PowerHA SystemMirror configuration supports operating systems cluster across the nodes in the cluster.

Smart Assist for Lotus Domino Server server will support the following lotus domino configuration:

PowerHA SystemMirror cluster with active-passive lotus domino configuration (without domino clustering)
> The basic configuration for active-passive operating system clustering is having a IBM Lotus Domino Server running on Node 1. While Node 2 monitors Node 1 and waits for a failure to occur. When a failure occurs, Node 2 picks up the resources of Node 1 and takes over running the IBM Lotus Domino Server. Node 2 uses the same disk set and the same IP address for the IBM Lotus Domino Server that Node 1 used.

PowerHA SystemMirror cluster with active-active lotus domino configuration (without domino clustering)
> To run IBM Lotus Domino Server in an active-active cluster, you must use Domino partitioned servers on the nodes. This allows each node take over the tasks of the other node while also maintaining its own tasks.

The basic configuration for active-active operating system clustering is Node 1 and Node 2 each has IBM Lotus Domino Server running in the first partition. The second partition on each node duplicates the resources of the first partition on the other node. Each node has its own disk set, but both nodes have access to both disk sets in case failover occurs. Thus, the nodes monitor each other.

When Node 1 fails, Node 2 picks up the resources of Node 1 and runs the IBM Lotus Domino Server for both nodes. To use an active-active configuration, you must be sure that each node can handle the load of the other node if failover occurs.

PowerHA SystemMirror cluster with domino clustering
For a Domino cluster you can configure either partitioned domino servers or stand alone IBM Lotus Domino Server. Do not include partitioned servers that are on the same computer in the cluster. Doing so can cause significant failover and workload problems if that computer fails.

To configure operating cluster for a Lotus Domino cluster consider the following requirements:

- There is no failover on the IBM Lotus Domino Server in the Domino cluster.
- The shared volume groups between the IBM Lotus Domino Server in the Domino cluster is not required as they replicate the data for every few seconds.

Planning for Smart Assist for Lotus Domino Server

You must have IBM Lotus Domino Server in your environment to run Smart Assist for Lotus Domino Server.

For information about specific versions of IBM Lotus Domino Server that are supported on different versions of PowerHA SystemMirror, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

The default installation location of the IBM Lotus Domino Server program files are `/opt/ibm/lotus`. Smart Assist for Lotus Domino Server assumes you have exported the DOMINO_PROG_DIR to the installed location of the Domino program folder. All the nodes must have the same program directory path.

When you are configuring IBM Lotus Domino Server in PowerHA SystemMirror, IP address or hostname of the IBM Lotus Domino Server must be an aliased IP address which can be accessible to all nodes in the cluster. This IP address is configured as service IP address for PowerHA SystemMirror. The service IP address must be in the **notes.ini** file of respective IBM Lotus Domino Server, which is assigned to the variable TCPIP_TCPIPAddress(TCPIP_TCPIPAddress=0,9.126.125.96).

PowerHA SystemMirror cluster with active-passive Lotus Domino configuration
In this configuration the data files of the IBM Lotus Domino Server must be installed on shared location. One node will have only one domino server configuration. The IBM Lotus Domino Server User ID and Group ID must be same in all the nodes in the cluster. Each Data directory of the Lotus domino server must be on different Volume Group (VG).

PowerHA SystemMirror cluster with active-active Lotus Domino configuration
In this configuration the data files of the IBM Lotus Domino Server must be installed on shared location. Partitioned IBM Lotus Domino Server are required for the active-active domino configuration. TheIBM Lotus Domino Server User ID and Group ID must be same in all the nodes in the cluster. Each data directory of the IBM Lotus Domino Server must be on different Volume Group (VG).

PowerHA SystemMirror cluster with Domino clustering
While configuring a Lotus Domino cluster in IBM Lotus Domino Server environment the data files can be stored in any location in the node. Partitioned IBM Lotus Domino Server are required for Domino clustering. It is your responsibility to mount the file system if the data directory is present on the file system upon system restart or initial setup.

Configuring Smart Assist for Lotus Domino Server

After you have planned and installed Smart Assist for Lotus Domino Server you can start to configure Smart Assist for Lotus Domino Server.

Note: If the data files are present in the shared volume group then the volume group must be online and the file system must be mounted.

Automatic discovery and configuration for Smart Assist for Lotus Domino Server

With minimal input you can use Smart Assist for Lotus Domino Server to automatically discovery and configure nodes.

To automatically discover and configure nodes, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **Lotus Domino Server Smart Assist** and press Enter.
4. Select **Automatic Discovery and Configuration** and press Enter.
5. Enter the following information for the fields:

Table 46. Smart Assist for Lotus Domino Server automatic discover and configuration fields

Fields	Values
Application Name	Specify the name for the collection of PowerHA SystemMirror component that represent the IBM Lotus Domino Server. The name is limited to 64 characters and can not contain spaces.
Participated nodes in the Lotus Domino cluster	Select the nodes names on which you want the IBM Lotus Domino Server configuration to be highly available.
Lotus Domino node and server's directory path	Enter the home node where the data directory resides and the absolute path of the Lotus Domino Server's data directory, which will be kept highly available. For example, node1,/server1;node2,/lotus/server2.
Domino cluster between the domino servers	Specify **n** if the Domino cluster is not configured between the Domino servers. Specify **y** if Domino cluster is configured between the domino servers.

Manual configuration and discovery

To use manual configuration you must have an xml file with required information. The IBM Lotus Domino Server and Smart Assist will parse the input xml file and configure the required configuration settings.

To use manual configure nodes, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **Lotus Domino Server Smart Assist** and press Enter.
4. Select **Manual Configuration** and press Enter.

5. In the File Name field specify the path where the xml file is located and press Enter. You can find a sample xml here, `/usr/es/sbin/cluster/sa/domino/config/cl_domino_config.xml`.

   Note: You can not modify the Smart Assist ID field. This ID is automatically generated.

Changing the IBM Lotus Domino Server configuration for PowerHA SystemMirror

After a IBM Lotus Domino Server has been configured for PowerHA SystemMirror, you can easily change the properties of the application.

To change the IBM Lotus Domino Server configuration, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.
3. Select the required application you want to change and press Enter.
4. Make the changes as needed in the fields for the selected application and press Enter. The same fields that you were able to modify when you created the configuration are available when you make changes, except for the Application Name and Instance owning node fields.

Changing the resources associated with your application

You can use SMIT to change any of the resources associated with your application.

To change the resources associated with your application, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show the Resources Associated with your Application** and press Enter.
3. Select the applications resources you want to modify and press Enter.
4. Make the changes as needed in the fields for the selected application and press Enter.

Removing an application

When you remove an application from the list of configured Smart Assist for Lotus Domino Server applications, PowerHA SystemMirror stops managing the availability of the resources you remove, because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.
3. Select the resource you want to remove from the PowerHA SystemMirror configuration database on the local node and press Enter.
4. Confirm that you want to remove the application and press Enter.

   Note: Removing an application from PowerHA SystemMirror removes all analogous resources from PowerHA SystemMirror, including application servers and monitors, service IP labels, and resource groups.

Smart Assist for Lotus Domino Server resources

After the IBM Lotus Domino Server has been configured through Smart Assist for Lotus Domino Server, PowerHA SystemMirror creates resources.

The following table displays the PowerHA SystemMirror resources created from Smart Assist for Lotus Domino Server:

Table 47. PowerHA SystemMirror resources	
PowerHA SystemMirror resource	**Name**
Resource group	**Domino_RG_nodeName_userName**, where userName is the Domino server user name, nodeName is the PowerHA SystemMirror nodeName.
Backup archive client application server	**Domino_APP_nodename_userName**, where userName is the Domino server user name, nodeName is the PowerHA SystemMirror nodeName.
Backup archive client custom monitor	**Domino_APP_MON_nodename_userName**, where userName is the Domino server user name, nodeName is the PowerHA SystemMirror nodeName.
Backup archive client start script	**cl_dominostart**, This file is located in the directory `/usr/es/sbin/cluster/sa/domino/sbin`.
Backup archive client stop script	**cl_dominostop**, This file is located in the directory `/usr/es/sbin/cluster/sa/domino/sbin`.
Backup archive client monitor script	**cl_dominomonitor**, This file is located in the directory `/usr/es/sbin/cluster/sa/domino/sbin`.

The following table describe the default settings associated with Backup Archive client custom monitor (Domino_AC_APP_MON_nodename_userName) that is displayed in Table 1.

Table 48. Backup Archive client customer monitor default settings	
Field	**Value**
Name	Domino_APP_MON_nodename_userName
Application Servers to Monitor	Domino_APP_nodename_userName
Monitor Method	/usr/es/sbin/cluster/sa/domino/sbin/ cl_dominomonitor -d < Domino server data directory>
Mode	Long-running monitoring
Interval	120 Sec
Hung Monitor Signal	9
Stabilization interval	180 Sec
Restart Count	3
Restart Interval	900 Sec

Table 48. Backup Archive client customer monitor default settings (continued)	
Field	**Value**
Action on Application Failure	Fallover

Troubleshooting Smart Assist for Lotus Domino Server

To successfully troubleshoot Smart Assist for Lotus Domino Server you must know where the different types of data are stored.

Smart Assist for Lotus Domino Server writes different types of data into different log locations.

Table 49. Smart Assist for Lotus Domino Server data locations	
Type of data	**Location**
Configuration data	Smart Assist for Lotus Domino Server stores configuration data in the HACMPsa_metadata PowerHA SystemMirror Configuration Database (ODM).
Log data	The **/var/hacmp/log/domino_server.log** file is used while Smart Assist for Lotus Domino Server is configured and monitored.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   • The **smit.log** file to examine the SMIT processes that have run from the SMIT interface.

   • The **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to make sure all your clusters are using the latest configuration settings.

Smart Assist for Tivoli Storage Manager

IBM Tivoli Storage Manager is comprised of a server, client, and admin center components.

Smart Assist for Tivoli Storage Manager overview

Before you start planning you need to have a basic understanding of the different components of IBM Tivoli Storage Manager.

IBM Tivoli Storage Manager server

IBM Tivoli Storage Manager server provides backup, archive, and space management services to the clients. You can set up multiple servers in your environment to balance storage, processor, and network resources. Using Smart Assist for Tivoli Storage Manager you can configure the IBM Tivoli Storage Manager server instance for high availability.

IBM Tivoli Storage Manager client

A client node can be a workstation, a personal computer, a file server, or another IBM Tivoli Storage Manager server. The client node has IBM Tivoli Storage Manager client software installed and is registered with the server.

IBM Tivoli Storage Manager backup-archive client
With the backup-archive client you can maintain backup versions of files, which you can use to restore the original files if they are lost or damaged. You can also archive files for long-term storage and retrieve the archived files when necessary. You can register workstations and file servers as client nodes with the IBM Tivoli Storage Manager server.

IBM Tivoli Storage Manager for space management
IBM Tivoli Storage Manager for space management is also known as the space manager client, or the hierarchical storage management (HSM) client. IBM Tivoli Storage Manager for space management provides you with space management services for workstations. The space management function is essentially a more automated version of archive. Tivoli Storage Manager for Space Management automatically migrates files that are less frequently used to server storage, freeing space on the workstation. The migrated files are also called space-managed files. IBM Tivoli Storage Manager backup-archive client can be used for backup and archiving of migrated files using HSM client.

IBM Tivoli Storage Manager admin center

IBM Tivoli Storage Manager admin center and integrated solutions console, are included in the IBM Tivoli Storage Manager product distribution and are installed with the IBM Tivoli Storage Manager server, as an optional component.

Planning for Smart Assist for Tivoli Storage Manager

Before you can install Smart Assist for Tivoli Storage Manager you must have your environment setup correctly.

Planning for Smart Assist for Tivoli Storage Manager server

Before you can use Smart Assist for Tivoli Storage Manager server you must review the following information:

- The IBM Tivoli Storage Manager server instance must have the service IP which can be pinged by the Smart Assist for Tivoli Storage Manager client.

- IBM Tivoli Storage Manager server instance name and IBM Tivoli Storage Manager server user name must be identical and configured on all the nodes in the cluster.

- The following information is shared between the nodes in the cluster for IBM Tivoli Storage Manager server instance:

  - Database directories

  - Instance log volumes

  - Storage pool volumes

  - Instance directory

  - Instance user directory

- Any number of IBM Tivoli Storage Manager server instances can be configured using Smart Assist for Tivoli Storage Manager, but the volume groups used for each IBM Tivoli Storage Manager server instance must be different from other IBM Tivoli Storage Manager server instances.

- When you add a IBM Tivoli Storage Manager server instance, it must be defined in dsm.sys file and dsm.opt file in the `/opt/tivoli/tsm/server/bin/tsmdiag` directory on all nodes of the cluster that are used for starting, stopping and monitoring the IBM Tivoli Storage Manager server instance. For example, the dsm.sys file must contain the following:

```
servername tsminst8
commmethod tcpip
tcpport 1500
tcpserveraddress abc.us.com
```

The dsm.opt file must contain the following:

```
servername tsminst8
```

For more information about specific versions of IBM Tivoli Storage Manager that are supported on different versions of PowerHA SystemMirror, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

The following figure displays a typical two node configuration for a IBM Tivoli Storage Manager server configuration.

Figure 9. IBM Tivoli Storage Manager server configuration
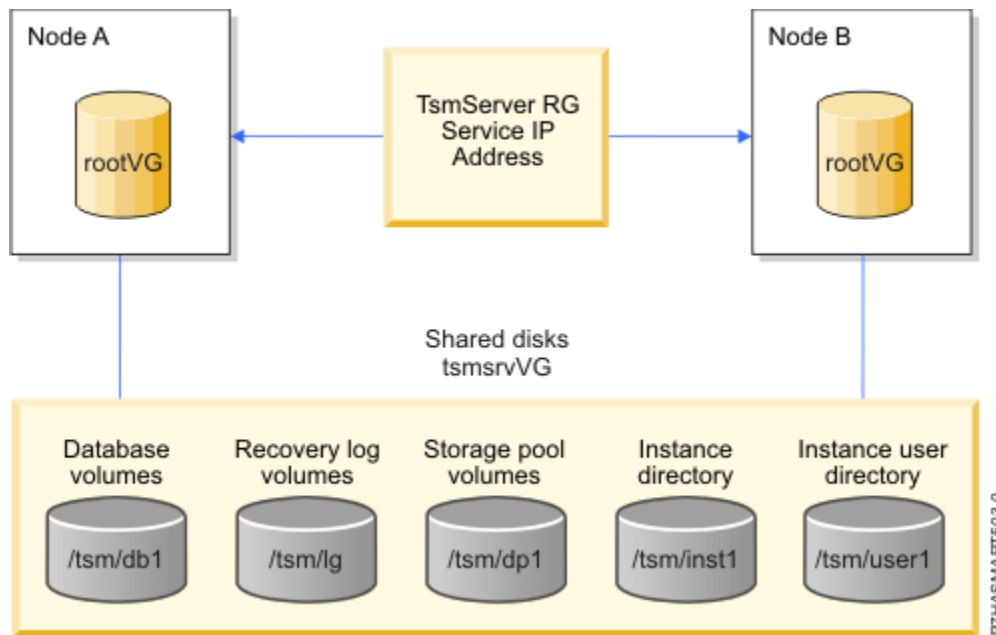
Planning for Smart Assist for Tivoli Storage Manager client

Before you can use Smart Assist for Tivoli Storage Manager client you must review the following information:

- The IBM Tivoli Storage Manager client must be installed in the default `/usr/tivoli/tsm/` directory on all the nodes in the cluster.
- The IBM Tivoli Storage Manager client must have the service IP aliased on one of the nodes from your initial configuration. You must register the service IP hostname as node to respective IBM Tivoli Storage Manager server instance.
- You must be able to ping service IP addresses from the IBM Tivoli Storage Manager server.
- The dsm.sys file must be the same on all the nodes in the cluster. The dsm.sys file is propagated to all node in the cluster from where you start the verify and synchronization. The dsm.sys file is located in the default location of IBM Tivoli Storage Manager client installation directory.
- The dsm.opt file and the tsm.pwd file are on shared location across all the nodes in the cluster.
- You must configure the dsm.opt file with the appropriate server name and domain.

For more information about specific versions of IBM Tivoli Storage Manager that are supported on different versions of PowerHA SystemMirror, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

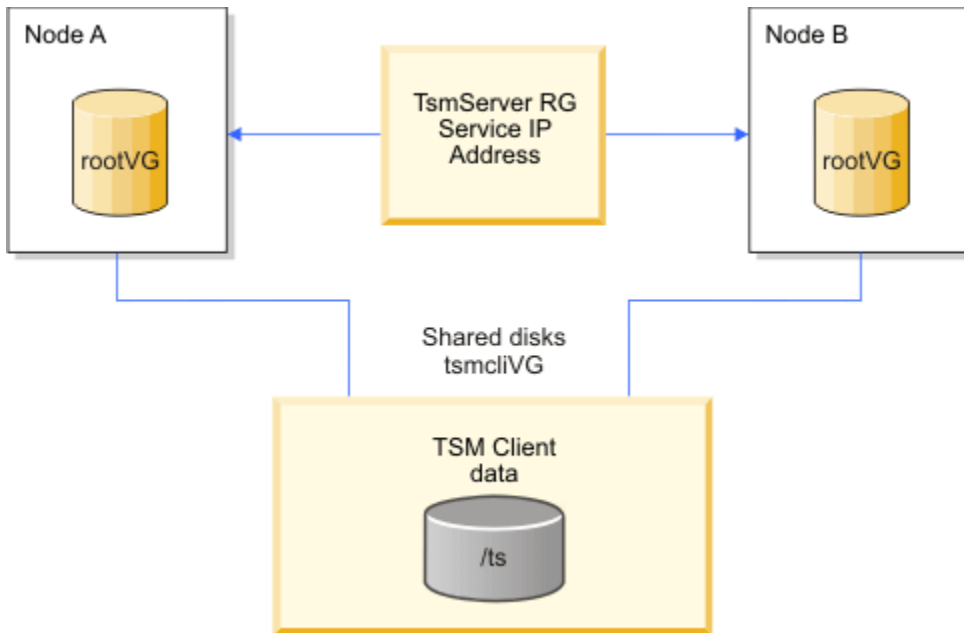The following figure displays a typical two node IBM Tivoli Storage Manager client configuration.

Figure 10. IBM Tivoli Storage Manager client configuration

Planning for Smart Assist for Tivoli Storage Manager admin center

Before you can use Smart Assist for Tivoli Storage Manager admin center you must review the following information:

- The IBM Tivoli Storage Manager admin center must be installed in the `/opt/tivoli/tsm` directory on all the nodes in the cluster.
- The IBM Tivoli Storage Manager admin center must have the service IP which will be aliased on the node where the admin center is configured.
- You must be able to ping service IP addresses from the IBM Tivoli Storage Manager server.

The following figure displays typical two node IBM Tivoli Storage Manager admin center configuration.
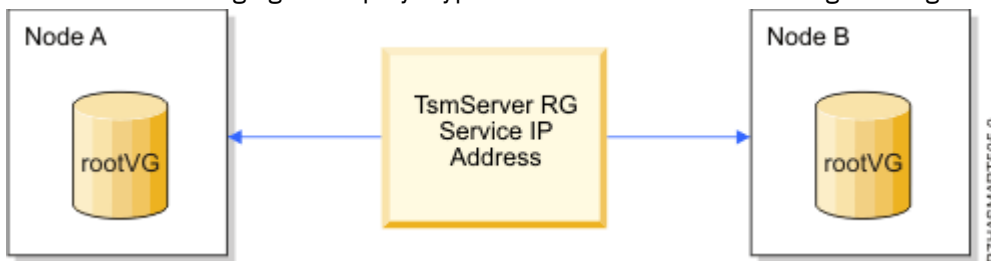

Figure 11. IBM Tivoli Storage Manager admin center configuration

Configuring Smart Assist for Tivoli Storage Manager

After you have planned and installed Smart Assist for Tivoli Storage Manager you can start to configure Smart Assist for Tivoli Storage Manager.

Automatic discovery and configuration for Smart Assist for Tivoli Storage Manager server

With minimal input you can use Smart Assist for Tivoli Storage Manager server to automatically discovery and configure nodes.

To automatically discovery and configure IBM Tivoli Storage Manager server instances, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **TSM server smart assist** and press Enter.
4. Select **Automatic Discovery and Configuration** and press Enter.
5. Select the server instance you want to discover and configure and press Enter.
6. Enter the following information for the fields:

Table 50. IBM Tivoli Storage Manager server automatic discover and configuration fields

Fields	Values
Application Name	Specify the name for the collection of PowerHA SystemMirror component that represent the IBM Tivoli Storage Manager server. The name is limited to 64 characters and can not contain spaces.
IBM Tivoli Storage Manager server instance owning node	Specify the highest priority node that can own the resource group created by the Smart Assist for Tivoli Storage Manager server.
Take over nodes	Specify the lowest priority node in order that can own the resource group that the IBM Tivoli Storage Manager server configuration assistant creates. You must leave a space between the node names.
IBM Tivoli Storage Manager server instance name	Displays the selected IBM Tivoli Storage Manager server instance will be made highly available by IBM Tivoli Storage Manager server configuration assistant.
IBM Tivoli Storage Manager server instance directory	Specify the IBM Tivoli Storage Manager directory where the IBM Tivoli Storage Manager instance logs are stored. This directory should be present in the shared disk.
Shared volume groups for IBM Tivoli Storage Manager server instance	Specify the shared volume groups where IBM Tivoli Storage Manager servers log files and configuration information is stored.
Service IP label	Specify the service IP label to be used by the IBM Tivoli Storage Manager server instance. The IBM Tivoli Storage Manager server instance is accessed on the server IP label.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.

Table 50. IBM Tivoli Storage Manager server automatic discover and configuration fields (continued)	
Fields	**Values**
IBM Tivoli Storage Manager server user ID	Specify the IBM Tivoli Storage Manager server instance administrator user ID that is used for monitoring server instances. **Note:** The IBM Tivoli Storage Manager admin can set up a user ID and password with limited authority.
IBM Tivoli Storage Manager server password	Specify the IBM Tivoli Storage Manager server password that is used for monitoring server instances. **Note:** The IBM Tivoli Storage Manager admin can set up a user ID and password with limited authority, or you can create them when you are configuring the IBM Tivoli Storage Manager server instance using the IBM Tivoli Storage Manager admin center.

Automatic discovery and configuration for Smart Assist for Tivoli Storage Manager client

With minimal input you can use Smart Assist for Tivoli Storage Manager client to automatically discovery and configure nodes.

The shared volume group must be online while configuring the IBM Tivoli Storage Manager client.

To automatically discovery and configure IBM Tivoli Storage Manager clients, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **TSM client smart assist** and press Enter.
4. Select **Automatic Discovery and Configuration** and press Enter.
5. Specify the location of the IBM Tivoli Storage Manager shared configuration directory which contains the dsm.opt file and press Enter.
6. Enter the following information for the fields:

Table 51. IBM Tivoli Storage Manager client automatic discover and configuration fields	
Fields	**Values**
Application Name	Specify the name for the collection of PowerHA SystemMirror component that represent the IBM Tivoli Storage Manager client. The name is limited to 64 characters and can not contain spaces.
IBM Tivoli Storage Manager client configuration directory	Displays the IBM Tivoli Storage Manager client shared configuration directory you already specified in Step 5.
IBM Tivoli Storage Manager client owning node	This is the highest priority node that can own the resource group created by IBM Tivoli Storage Manager client.

Table 51. IBM Tivoli Storage Manager client automatic discover and configuration fields (continued)	
Fields	**Values**
Take over nodes	Specify the lowest priority node in order that can own the resource group that the IBM Tivoli Storage Manager client configuration assistant creates. You must leave a space between the node names.
Shared file systems of the client to be backed up	Select the file systems from the list that you want to be backed up. These file systems are in the dsm.opt file.
Service IP label	Specify the service IP label to be used by the IBM Tivoli Storage Manager client. The IBM Tivoli Storage Manager client is accessed on the server IP label.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.
IBM Tivoli Storage Manager space management (HSM) client	Specify **y** if you want to configure the IBM Tivoli Storage Manager space management with the IBM Tivoli Storage Manager client. Specify **n** if you do not want to configure the IBM Tivoli Storage Manager space management with the IBM Tivoli Storage Manager client.

Automatic discovery and configuration for Smart Assist for Tivoli Storage Manager admin center

With minimal input you can use Smart Assist for Tivoli Storage Manager admin center to automatically discovery and configure nodes.

To automatically discovery and configure IBM Tivoli Storage Manager admin center, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **TSM admin smart assist** and press Enter.
4. Select **Automatic Discovery and Configuration** and press Enter.
5. Select the admin center node you want to discover and configure and press Enter.
6. Enter the following information for the fields:

Table 52. IBM Tivoli Storage Manager automatic discover and configuration fields

Fields	Values
Application Name	Specify the name for the collection of PowerHA SystemMirror component that represent the IBM Tivoli Storage Manager admin center. The name is limited to 64 characters and can not contain spaces.
IBM Tivoli Storage Manager admin center owning node	This is the highest priority node that can own the resource group created by IBM Tivoli Storage Manager admin center.
Take over nodes	Specify the lowest priority node in order that can own the resource group that the IBM Tivoli Storage Manager admin center configuration assistant creates. You must leave a space between the node names.
Shared file systems of the client to be backed up	Select the file systems from the list that you want to be backed up. These file systems are in the dsm.opt file.
Service IP label	Specify the service IP label to be used by the IBM Tivoli Storage Manager admin center. The IBM Tivoli Storage Manager admin center is accessed on the service IP label.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address.
	This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.
IBM Tivoli Storage Manager admin center user ID	Specify the IBM Tivoli Storage Manager admin center user ID.
IBM Tivoli Storage Manager admin center user password	Specify the IBM Tivoli Storage Manager admin center user password.

Manual configuration and discovery

You can use a xml file to configure IBM Tivoli Storage Manager.

To manually configure IBM Tivoli Storage Manager, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **TSM Server Smart Assist** > **Manual Configuration** and press Enter.
4. In the File Name field specify the path where the xml file is located and press Enter. You can find a sample xml in the following locations:

IBM Tivoli Storage Manager server
> `/usr/es/sbin/cluster/sa/tsmserver/config/cl_tsmserver_config.xml`

IBM Tivoli Storage Manager client
> `/usr/es/sbin/cluster/sa/tsmclient/config/cl_tsmclient_config.xml`

IBM Tivoli Storage Manager admin center
> `/usr/es/sbin/cluster/sa/tsmclient/config/cl_tsmadmin_config.xml`

Note: You can not modify the Smart Assist ID field. This ID is automatically generated.

Changing the IBM Tivoli Storage Manager configuration for PowerHA SystemMirror

After a IBM Tivoli Storage Manager has been configured for PowerHA SystemMirror, you can easily change the properties of the application.

To change the IBM Tivoli Storage Manager configuration, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Application's HACMP Configuration** and press Enter.
3. Select the required application you want to change and press Enter.
4. Make the changes as needed in the fields for the selected application and press Enter. The same fields that you were able to modify when you created the configuration are available when you make changes, except for the Application Name, Instance owning node, and IBM Tivoli Storage Manager server instance name fields.

   Note: You can not modify all of the fields.

Changing the resources associated with your application

You can use SMIT to change any of the resources associated with your Smart Assist for Tivoli Storage Manager instance.

To change the resources associated with your Smart Assist for Tivoli Storage Manager instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show the Resources Associated with your Application** and press Enter.
3. Select the applications resources you want to modify and press Enter.
4. Make the changes as needed in the fields for the selected application and press Enter.

Removing an application

When you remove an application from the list of configured Smart Assist for Tivoli Storage Manager applications, PowerHA SystemMirror stops managing the availability of the resources you remove, because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.
3. Select the resource you want to remove from the PowerHA SystemMirror configuration database on the local node and press Enter.
4. Confirm that you want to remove the application and press Enter.

Note: Removing an application from PowerHA SystemMirror removes all analogous resources from PowerHA SystemMirror, including application controllers and monitors, service IP labels, and resource groups.

Smart Assist for Tivoli Storage Manager resources

After IBM Tivoli Storage Manager has been configured through Smart Assist for Tivoli Storage Manager, PowerHA SystemMirror creates resources.

IBM Tivoli Storage Manager server resources

Review the PowerHA SystemMirror resources created from IBM Tivoli Storage Manager server.

Table 53. IBM Tivoli Storage Manager server resources	
PowerHA SystemMirror resource	**Name**
Resource group	**TSM_SERV_RG_instanceName**, where instanceName is the name of the IBM Tivoli Storage Manager server instance name.
Backup archive client application server	**TSM_SERV_APP_instanceName**, where instanceName is the name of the IBM Tivoli Storage Manager server instance name.
Backup archive client custom monitor	**TSM_SERV_APP_MON_instanceName**, where instanceName is the name of the IBM Tivoli Storage Manager server instance name.
Backup archive client start script	**cl_tsmserverstart**, this file is located in the directory /usr/es/sbin/cluster/sa/tsmserver/sbin.
Backup archive client stop script	**cl_tsmserverstop**, this file is located in the directory /usr/es/sbin/cluster/sa/tsmserver/sbin.
Backup archive client monitor script	**cl_tsmservermonitor**, this file is located in the directory /usr/es/sbin/cluster/sa/tsmserver/sbin.

The following table describes the default settings associated with the backup archive custom monitor (TSM_SERV_APP_MON_instanceName) that is displayed in Table 1.

Table 54. Backup archive customer monitor (TSM_SERV_APP_MON_instanceName) default settings	
Field	**Value**
Name	TSM_SERV_APP_MON_instanceName
Application Server(s) to Monitor	TSM_SERV_APP_instanceName
Monitor Method	/usr/es/sbin/cluster/sa/tsmserver/sbin/ cl_tsmservermonitor –i <Instance Name>
Mode	Long-running monitoring
Interval	180 Sec
Hung Monitor Signal	9
Stabilization interval	180 Sec
Restart Count	3

Table 54. Backup archive customer monitor (TSM_SERV_APP_MON_instanceName) default settings (continued)

Field	Value
Restart Interval	1200 Sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/tsmserver/sbin/ cl_tsmserverstop -i <instance Name>
Restart Method	/usr/es/sbin/cluster/sa/tsmserver/sbin/ cl_tsmserverstart -i <instance Name> -d <instance Directory>

IBM Tivoli Storage Manager client resources

Review the PowerHA SystemMirror resources created from IBM Tivoli Storage Manager client.

Table 55. IBM Tivoli Storage Manager client resources

PowerHA SystemMirror resource	Name
Resource group	**TSM_CLI_RG_tsmclient_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client application server	**TSM_CLI_APP_tsmclient_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client custom monitor	**TSM_CLI_APP_MON_tsmclient_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client start script	**cl_tsmBAclientstopt**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.
Backup archive client stop script	**cl_tsmserverstop**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.
Backup archive client monitor script	**cl_tsmBAclientmonito**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.
HSM client application server	**TSM_HSM_APP_tsmclient_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
HSM client start script	**cl_tsmHSMclientstart**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.
HSM client stop script	**cl_tsmHSMclientstop**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.
HSM client monitor script	**cl_tsmHSMclientmonitor**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmclient/sbin.

The following table describes the default settings associated with backup archive client custom monitor (TSM_CLI_APP_MON_tsmclient_nodename) that is displayed in Table 1.

Table 56. Backup archive client customer monitor (TSM_CLI_APP_MON_tsmclient_nodename)

Field	Value
Name	TSM_CLI_APP_MON_tsmclient_nodename
Application Server(s) to Monitor	TSM_CLI_APP_tsmclient_nodename
Monitor Method	/usr/es/sbin/cluster/sa/tsmclient/sbin/ cl_tsmBAclientmonitor –a <Application Name>
Mode	Long-running monitoring
Interval	180 Sec
Hung Monitor Signal	9
Stabilization interval	180 Sec
Restart Count	5
Restart Interval	1200 Sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/tsmclient/sbin/ cl_tsmBAclientstop
Restart Method	/usr/es/sbin/cluster/sa/tsmclient/sbin/ cl_tsmBAclientstart –a <Application name>

IBM Tivoli Storage Manager admin center resources

Review the PowerHA SystemMirror resources created from IBM Tivoli Storage Manager admin center.

Table 57. IBM Tivoli Storage Manager admin center resources

PowerHA SystemMirror resource	Name
Resource group	**TSM_AC_RG_tsmadmin_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client application server	**TSM_AC_APP_tsmadmin_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client custom monitor	**TSM_AC_APP_MON_tsmadmin_nodename**, where nodename is the name on which the PowerHA SystemMirror configuration is configured.
Backup archive client start script	**cl_tsmadminstart**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmadmin/sbin.
Backup archive client stop script	**cl_tsmadminstop**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmadmin/sbin.
Backup archive client monitor script	**cl_tsmadminmonitor**, this file is located in the directory /usr/es/sbin/cluster/sa/ tsmadmin/sbin.

The following table describes the default settings associated with backup archive client custom monitor (TSM_AC_APP_tsmadmin_nodename) that is displayed in Table 1.

Table 58. Backup archive client customer monitor (TSM_AC_APP_tsmadmin_nodename)

Field	Value
Name	TSM_AC_APP_MON_tsmadmin_nodename
Application Server(s) to Monitor	TSM_AC_APP _tsmadmin_nodename
Monitor Method	/usr/es/sbin/cluster/sa/tsmadmin/sbin/ cl_tsmadminmonitor –a <Application Name>
Mode	Long-running monitoring
Interval	120 Sec
Hung Monitor Signal	9
Stabilization interval	180 Sec
Restart Count	3
Restart Interval	900 Sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/tsmclient/sbin/ cl_tsmadminstop -a <Application name>
Restart Method	/usr/es/sbin/cluster/sa/tsmclient/sbin/ cl_tsmadminstart

Troubleshooting Smart Assist for Tivoli Storage Manager

To successfully troubleshoot Smart Assist for Tivoli Storage Manager you must know where the different types of data are stored.

Smart Assist for Tivoli Storage Manager writes different types of data into different log locations.

Table 59. Smart Assist for Tivoli Storage Manager data locations

Type of data	Location
Configuration data	Smart Assist for Tivoli Storage Manager stores configuration data in the HACMPsa_metadata PowerHA SystemMirror Configuration Database (ODM).
Smart Assist for Tivoli Storage Manager server log data	The **/var/hacmp/log/tsm_server.log** file is used while Smart Assist for Tivoli Storage Manager server is configured and monitored.
Smart Assist for Tivoli Storage Manager client log data	The **/var/hacmp/log/tsm_client.log** file is used while Smart Assist for Tivoli Storage Manager client is configured and monitored.
Smart Assist for Tivoli Storage Manager admin center log data	The **/var/hacmp/log/tsm_admin.log** file is used while Smart Assist for Tivoli Storage Manager admin center is configured and monitored.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   • View the log files in the table.

- The **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
- The **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to make sure all your clusters are using the latest configuration settings.

Smart Assist for Tivoli Directory Server

With Smart Assist for Tivoli Directory Server you can automatically configure PowerHA SystemMirror where Tivoli Directory Server is already installed.

Related information

IBM Security Directory Server

Smart Assist for Tivoli Directory Server concepts

Smart Assist for Tivoli Directory Server supports stand-alone directory configurations, distributed directory configurations, and peer-to-peer directory configurations.

Stand-alone directory configuration

In a stand-alone directory configuration there are multiple Tivoli Directory Server instances that are running on a single node (Node 1). When a failure occurs on Node 1, another node (Node 2) obtains the resources of Node 1 and takes over running the Tivoli Directory Server instance. Node 2 uses the same disk set and the same IP address for the Tivoli Directory Server instance that Node 1 used.

When Node 1 fails, a database instance is created on a shared volume group that Node 2 can access. Also, when Node 1 fails its service IP label is taken over by Node 2.

Distributed directory configuration

In a distributed directory configuration there are multiple Tivoli Directory Server instances running on each node in a PowerHA SystemMirror cluster. Each Tivoli Directory Server instance has its peer servers and replica servers that are running on a node in the cluster.

In a distributed directory configuration, client requests are routed to the proper Tivoli Directory Server by a proxy server that is running on a node in the cluster. The proxy server is not controlled by PowerHA SystemMirror.

When a failure occurs on any of the Tivoli Directory Server instances, the instance is restarted by PowerHA SystemMirror. If failure occurs on a node in the cluster, PowerHA SystemMirror is not supported because fallover and fallback are handled by the Tivoli Directory Server.

When a failure occurs for a node and data is replicated onto different a Tivoli Directory Server, there is no need for a shared volume group or service IP label in the cluster. Therefore, the node taking over for the failed node does not need to access the shared volume group or service IP label.

Peer-to-peer directory configuration

In a peer-to-peer directory configuration, the proxy server is not available, and there is no data partition and data distribution between the Tivoli Directory Server instances. However, the Tivoli Directory Server instances have access to peer server instances and replica server instances on a node in the cluster.

When a failure occurs on any of the Tivoli Directory Server instances, the instance is restarted by PowerHA SystemMirror. If failure occurs on a node in the cluster, PowerHA SystemMirror is not supported because fallover and fallback are handled by the Tivoli Directory Server.

When a failure occurs for a node and data is replicated onto different a Tivoli Directory Server, there is no need for a shared volume group or service IP label in the cluster. Therefore, the node taking over for the failed node does not need to access the shared volume group or service IP label.

Planning Smart Assist for Tivoli Directory Server

You must install all Tivoli Directory Server software on all nodes in the cluster, before you install Smart Assist for Tivoli Directory Server.

You must install the Tivoli Directory Server files in the `/opt/IBM/ldap/V6.3` default directory, because Smart Assist looks for the Tivoli Directory Server installation in the default directory during the discovery process.

If you use a stand-alone directory configuration you must create every database instance on a shared volume group. If you use a peer-to-peer directory configuration or a distributed directory configuration you do not need to create any database instances on a shared volume group, you can create the database instance on a local node.

Software requirements

To view what versions of Tivoli Directory Server are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Related information

IBM Sercurity Directory Server

Installing Tivoli Directory Server

Tivoli Directory Server system requirements

Configuring Smart Assist for Tivoli Directory Server

After you have planned and installed Smart Assist for Tivoli Directory Server, you can start to configure Smart Assist for Tivoli Directory Server. Configuring Smart Assist for Tivoli Directory Server is different for distributed directory and stand-alone Tivoli Directory Server configurations.

Automatic discovery and configuration of Smart Assist for Tivoli Directory Server

You can use automatic discovery to configure Smart Assist for Tivoli Directory Server with minimal input.

To automatically discover and configure Smart Assist for Tivoli Directory Server, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and ResourcesMake Applications Highly Available (use Smart Assists) Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **Tivoli Directory Server Smart Assist** and press Enter.
4. Select **Automatic Discovery and Configuration** and press Enter.
5. Enter the following information for the type of Tivoli Directory Server configuration that is running in your cluster.

Table 60. Smart Assist for Tivoli Directory Server configuration fields	
Field	**Value**
Application name	Specify an application name. A unique application name is required for every Tivoli Directory Server instance because a different resource group is created for every Tivoli Directory Server instance.
Primary node	Specify the primary node for the Smart Assist. This field is only available for stand-alone configurations.

Table 60. Smart Assist for Tivoli Directory Server configuration fields (continued)	
Field	**Value**
Takeover node	Specify the xml file that contains information about the Smart Assist for Tivoli Directory Server environment. This field is only available for stand-alone configurations.
Service interface	Specify the service IP label that is used with the Tivoli Directory Server client. This field is only available for stand-alone configurations.
Tivoli Directory Server password	Specify the password for the Tivoli Directory Server.
Tivoli Directory Server port	Specify the port that is configured with Tivoli Directory Server.
DB2 instance name	Specify the DB2 instance name that is used with the Tivoli Directory Server. This field is only available for stand-alone configurations.
Number of Tivoli Directory Server instances	Specify the number of instances on the node. This field is only available for distributed directory configurations.
Tivoli Directory Server instance name	Specify the name of the Tivoli Directory Server instance. This field is only available for stand-alone configurations.

6. Verify and synchronize your cluster to make the configuration changes on all the nodes.

Manual configuration of Smart Assist for Tivoli Directory Server

For manual configuration you specify an xml file that Smart assist with use to to create the Tivoli Directory Server.

A template configuration file is located in the `/usr/es/sbin/cluster/sa/tds/config/tds_config.xml` directory.

To manual configure Smart Assist for Tivoli Directory Server, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and ResourcesMake Applications Highly Available (use Smart Assists) Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **Tivoli Directory Server Smart Assist** and press Enter.
4. Select **Manual Configuration** and press Enter.
5. In the **File Name** field specify the xml file that contains the required information to configure Tivoli Directory Server with Smart Assist.

   Note: You cannot modify the **Smart Assist ID** field.
6. Verify and synchronize your cluster to make the configuration changes on all the nodes.

Managing Smart Assist for Tivoli Directory Server

You can change the Smart Assist for Tivoli Directory Server configuration or resources, and remove Smart Assist for Tivoli Directory Server application with the SMIT interface.

Changing the Smart Assist for Tivoli Directory Server configuration

After you have configured Smart Assist for Tivoli Directory Server, you can change the properties of the application.

To change the configuration of Smart Assist for Tivoli Directory Server, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Applications PowerHA SystemMirror Configuration**, and press Enter.
3. Select the required application that you want to change, and press Enter.
4. Make the changes as needed in the fields for the selected application, and press Enter. The same fields that you were able to modify when you created the configuration are available when you make changes, except for the **Application name** field and the **Primary node** filed.

Changing the resources of Smart Assist for Tivoli Directory Server

You can change the resources if you want to replace a takeover node or change the service IP label to an existing service IP label.

To change the resource of the Smart Assist for Tivoli Directory Server, complete the following steps.

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show the Resources Associated with yoru Application Configuration**, and press Enter.
3. Select the applications resources that you want to modify, and press Enter.
4. Change the fields that you want for the selected application, and press Enter.

Removing a Smart Assist for Tivoli Directory Server application

When you remove an application from the list of configuredSmart Assist for Tivoli Directory Server applications, PowerHA SystemMirror stops managing the availability of the resources you remove, because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove a Smart Assist for Tivoli Directory Server application, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration**, and press Enter.
3. Select the application that you want to remove from the PowerHA SystemMirror configuration database on the local node, and press Enter.
4. Confirm that you want to remove the application, and press Enter.

   Note: Removing an application from PowerHA SystemMirror removes all analogous resources from PowerHA SystemMirror, which includes application controllers and monitors, service IP labels, and resource groups.

Troubleshooting Smart Assist for Tivoli Directory Server

To successfully troubleshoot Smart Assist for Tivoli Directory Server you must know where the different types of data are stored.

Use the following table to identify where Smart Assist for Tivoli Directory Server writes different types of data and the locations of the data.

Type of data	Location
Configuration data	Smart Assist for Tivoli Directory Server stores configuration data in the **HACMPsa_metadata** PowerHA SystemMirror Configuration Database (ODM).
Log data	The **/var/hacmp/log/tdssa.log** file is used while Smart Assist for Tivoli Directory Server is configured and monitored.

Table 61. Smart Assist for Tivoli Directory Server data locations

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:
   - View the log files in the Table 1 table.
   - View the **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
   - View the **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application.
2. Verify and synchronize your cluster to verify that all your clusters are using the latest configuration settings.

Smart Assist for SAP Netweaver

Smart Assist for SAP Netweaver supports multiple SID installations in which resources are not shared among instances of different SIDs.

In a typical SAP Netweaver installation, the following single point of failures must be protected.

- SAP global file system
- SAP ABAP central services instance
- SAP Java central service instance
- SAP enqueue replication server instance for ABAP
- SAP enqueue replication server instance for Java
- Optional: SAP application server instance. If you have only one SAP application server instance, you can set up multiple AS/Dialog instances with SAP application server.

Apart from these SAP Netweaver internal single points of failures, the following components are also important to achieve high availability of SAP Netweaver:

- Database instance (powered by either IBM DB2, Oracle database, or SAP MaxDB)
- SAP liveCache Hot Standby (optional)

If you are using Network File System version 4 (NFSv4), the application monitor for the SAP Central Service (SCS) instance might time out if the NFS performance is slow. This time out issue can cause an unwanted failover of the SCS instance. To prevent this problem, you can change the lease time for NFSv4 by running the /usr/sbin/chnfs -L'20' '-B' command. You must restart the node for the lease time change to take effect.

The PowerHA SystemMirror base version still supports Network File System if it is used to make any application highly available by using a traditional method.

Smart Assist for SAP Netweaver helps you identify a single point of failure with the following features:

Support for standalone Enqueue Server 2

The Central Services Instance ABAP (ASCS) instance and Enqueue Replicator Server (ERS) instance might not run on the same node simultaneously when you are using a standalone Enqueue Server.

During failover of the ASCS instance, if you move the ASCS instance to a host where ERS is running, you must restart all ERS instances. This process results in two recovery actions. The services are unavailable during these recovery actions, which decrease the overall availability of the solution.

With the new standalone Enqueue Server2 (ENSA2) feature, the ASCS can failover to a host where ERS is not running, thus avoiding the need for the ERS instance to be moved.

The ASCS instance can start on any other host in the cluster and retrieve the lock table.

PowerHA SystemMirror supports both standalone Enqueue Server (ENSA) and ENSA2.

Discovery

Discovers all the configured SAP Netweaver instances such as central services instances and enqueue replication instances that are running in the cluster (collection of nodes) and it's associated resources such as volume groups and logical host names.

Optionally, this feature can be used to configure the SAP Global directory or SAP Transport directory, based on NFS4 and SAP Application Server instances.

Configuration

Configures different SAP Netweaver instances by creating appropriate PowerHA SystemMirror objects such as resource groups, application servers, application monitor, and service IP addresses.

Management

You can use start and stop monitor scripts to effectively manage SAP Netweaver instances while protecting SAP global file systems through NFS.

Planning for Smart Assist for SAP Netweaver

To use Smart Assist for SAP Netweaver, you must install and tune the AIX operating system according to the SAP OSS Notes and the official SAP installation guides.

To view what versions of SAP Netweaver are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

The following table describes the storage and IP requirements while installing the different single point of failures for SAP Netweaver 2004s system. These requirements apply for those IP addresses that must be discovered by Smart Assist.

Note: All single point of failures in the infrastructure up to the application layer must be set up highly available to meet the business availability requirements.

In the table, SID represents a three letter SAP system ID, <nn> represents a SAP instance number, and ERS represents an enqueue replication server.

Table 62. Storage and IP requirements

SI Number	Single point of failure	Volume group requirement	File system mount point	IP address requirement
1	SAP global file system	A minimum of 1 volume group for each, attached to both nodes.	/export/sapmnt/<SID> /export/usr/sap/trans	One free IP

Table 62. Storage and IP requirements (continued)

SI Number	Single point of failure	Volume group requirement	File system mount point	IP address requirement
2	ABAP central services instance	A minimum of 1 volume group for each, attached to both nodes, or a local file system that is maintained by the user on both nodes.	/usr/sap/<SID>/ ASCS<nn>	Installed with a dedicated virtual IP
3	Java central services instance	A minimum of 1 volume group for each, attached to both nodes, or a local file system that is maintained by the user on both nodes.	/usr/sap/<SID>/ SCS<nn>	Installed with a dedicated virtual IP
4	ABAP ERS instance	A minimum of 1 volume group for each, attached to both nodes, or a local file system that is maintained by the user on both nodes.	/usr/sap/<SID>/ ERS<nn>	Installed with a dedicated virtual IP
5	Java ERS instance	A minimum of 1 volume group for each, attached to both nodes, or a local file system that is maintained by the user on both nodes.	/usr/sap/<SID>/ ERS<nn>	Installed with a dedicated virtual IP
6	Application server instance	A minimum of 1 volume group for each, attached to both nodes, or a local file system that is maintained by the user on both nodes.	/usr/sap/<SID>/ DVEBMGS<nn>	Installed with a dedicated virtual IP

Note:

1. The volume group names that are displayed in Table 1 are just for demonstration, and you can replace them with a valid volume group name.

2. To support older system installations ASCS and SCS instances can share a virtual IP. Also, both ERS instances can share a virtual IP. This setup is not recommended and applies only for Dual Stacks.

Before you can start Smart Assist for SAP Netweaver, review the following information:

• Ensure that your hardware infrastructure is set up for redundancy.

- When using shared disks, you must verify that all volume groups that become part of a resource group are attached to both nodes. You can manage local and shared volume groups by using C-SPOC.
- Configure and tune all nodes as described in the SAP installation guide of your release and version of SAP.
- Ensure that user IDs and group IDs of the SAP OS-users are the same on both nodes. You can manage users, groups, and password with C-SPOC (smit cspoc).
- Create the basic two node PowerHA SystemMirror cluster by using SMIT or the **clmgr** command.
- Install the SAP Instances on your primary node. All instances must be installed with their own virtual IP by using the **./sapinst SAPINST_USE_HOSTNAME=<IP alias>** command (exceptions see Table 1).
- Prepare the second node. Ensure that the `/usr/sap/sapservices` file is the same on both nodes.
- Ensure that sapcpe program is configured such that all instances can operate independent from their availability of NFS during runtime.
- The service sapstartsrv of all instances must be running so that they can be discovered properly by the smart assist. This setting is the SAP default after the installation.
- The Sap Global directory must be accessible from all the nodes and setup to be highly available.
- The SAP Admin user <sid>adm must be the SAP default.
- Apply the SAP HA licenses to run SAP Netweaver instances on the failover nodes as required by SAP.
- Verify that SAP Netweaver instances can run successfully on all the failover nodes by completing the following steps:
  1. Set up the failover nodes for accessing the SAP global file system.
  2. Create SAP users on all failover nodes with same uid and gid values.
  3. Run the SAP Netweaver instances on failover nodes by manually varying on the volume group, which contain the file system for the instance directory (on a shared disk) and set up the logical IP addresses as an IP alias by using the **ifconfig** command.
- Remove the MAIL variable information from the .login file that is set by SAP. You must have SAP administrator authority to remove the MAIL variable.
- When you have mail there is additional output from commands that are run by PowerHA SystemMirror Smart Assists that can cause the parsing of the command output to fail. To fix this issue, add the following code at the start of the the the .cshrc file:

```
setenv MAIL "/var/spool/mail/$LOGNAME"
setenv MAILMSG "[YOU HAVE NEW MAIL]"

if ( $?prompt == 0 ) then
# This is a shell script .... do nothing
else
# This is interactive
if ( -f "$MAIL" && ! -z "$MAIL") then
        echo "$MAILMSG"
endif
endif
```

Note: If you are not using the csh shell, then use the appropriate variable for the prompt.

- You must verify that the .cshrc and .login files do not produce any output text.
- To perform SAP administrative functions, add the SAP admin user to the hacmp group by using the SMIT interface (smitty chuser) or the **chuser** command. This process allows the SAP admin user to access the PowerHA SystemMirror binary files and ODMs.

Before installing any SAP Netweaver instances, you must set up the global file systems on all the cluster nodes by using NFS cross mounts.

If you plan to also add the SAP Global directory and the SAP Transport directory to this cluster, the following items apply in addition:

- Set up the following mounts on a node where the volume group is hosting file system that is varied on:

- /export/sapmnt/<SAP System ID>
- /export/usr/sap/trans

- Set up the following NSF cross mounts on all nodes, for example:
  - mount <nfsip>:/export/sapmnt/<SAP System ID> /sapmnt/<SAP System ID>
  - mount <nfsip>:/export/usr/sap/trans /usr/sap/trans

  Note: Smart Assist for SAP cross mounts file systems related to SAP global file systems such as, /export/sapmnt/<SAP System ID> and /export/usr/sap/trans use NFS version 4 on all cluster nodes.

  NFS is used for the SAP global file system. If the /sapmnt directory is inaccessible due to NFS failure caused by a network or interface failure on a particular node, PowerHA SystemMirror tries to move any of the SAP instances running on that node to another node. Therefore, the SAP instances might fail to stop on that node, resulting in a cluster event script error. If this problem occurs, you must manually complete the fallover by using the SMIT utility Recover From PowerHA SystemMirror Script Failure.

- Verify that ABAP and Java central services instances as well as all ABAP and Java enqueue replication server instances do not run on the same node while using Smart Assist for SAP Netweaver.

- The ERS is cluster controlled (no polling) and is installed with a virtual IP.

- To configure a SAP instance run Smart Assist for SAP Netweaver on the node where that particular instance is running.

- Verify that SAP Netweaver 2004s instances can run successfully on all the failover nodes by completing the following steps:
  1. Set up the failover nodes for accessing SAP global file system.
  2. Create SAP users on all failover nodes with same uid and gid values
  3. Run the SAP Netweaver 2004s instances on failover nodes by manually varying on the volume group that contains file system for the instance directory (on a shared disk) and set up the logical IP addresses by using IP alias.
  4. Apply any required licenses to run different SAP Netweaver 2004s instances on the failover nodes.

  Now the nodes are configured to run Smart Assist for SAP Netweaver on the node where that particular instance's sapstartsrv process is running.

Related information

Recovering from PowerHA SystemMirror script failure

IBM Techdocs White Paper: Invincible Supply Chain - SAP APO Hot Standby liveCache on IBM Power Systems

Configuring Smart Assist for SAP Netweaver

Before you configure any SAP Netweaver instances for high availability, you must verify that a PowerHA SystemMirror cluster is created.

Automatic discovery and configuration for global file system instance

With minimal input you can use Smart Assist for SAP to automatically discovery and configure a global file system.

To automatically discover and configure a global file system instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **SAP Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **SAP NW 7.0 Global Filesystem** and press Enter.
5. Enter the following information for the fields:

Table 63. Global file system instance automatic discovery and configuration fields

Fields	Values
SAP System ID	Displays the value for this field. This value is pre-populated with the SAP system ID and it can not be modified.
Application Name	Specify the application name for the PowerHA SystemMirror resource groups, application servers, and other cluster components. This value must be fewer than 64 characters and follow PowerHA SystemMirror restrictions in application names.
SAP Global File System Owning Node	Displays the node where SAP global file system is presently available in the cluster. This value automatically discovered.
Take over nodes	Specify the node that you want the SAP global file system to make available through NFS cross mounts.
Service IP label	Specify the service IP label used for NFS cross mounts.
Shared volume groups	Specify the shared volume groups, which include /export/sapmnt/<SAP System ID> and /export/usr/sap/trans files systems.
File systems / Directories to Export	Specify the SAP global filesystem mount points.
File systems / Directories to NFS Mount	Specify the NSF cross mounts. Use the following syntax for specifying an NFS cross mounts `NFS_Mount_Point;Local_Filesystem`.

Automatic discovery and configuration for a central services instance

With minimal input, you can use Smart Assist for SAP Netweaver to automatically discover and configure a SAP center services instance.

To automatically discover and configure a central service instance, complete the following steps that are based on the /usr/sap/sapservices file and the sapstartsrv service of the instances:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **SAP Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **SAP NW 7.0 (A)SCS Instance** and press Enter.
5. Select the central service instance name that you want to discover and configure.
6. Enter the following information for the fields in the table:

Table 64. Central services instance automatic discovery and configuration fields

Fields	Values
SAP System ID	Displays the value for this field that is pre-populated with the SAP system ID. This value cannot be modified.

Table 64. Central services instance automatic discovery and configuration fields (continued)

Fields	Values
SAP SCS/ASCS Instance Name	Displays the name of central services instances that are selected in Step 5. This value cannot be modified.
Application Name	Specify the application name for the PowerHA SystemMirror resource groups, application servers, and other cluster components. This value must be fewer than 64 characters and follow PowerHA SystemMirror restrictions in application names.
Primary Node	Specify the node where the central services instance is presently running. Changing this value to a different node might cause the central services instances to stop on the original node and restart on changed node.
Take over nodes	Specify the take over node for this central services instance.
Network name	Specify the network for the service IP alias.
Service IP label	Specify the logical host name that is used to configure this central services instance. It must be mandatory to configure the central services instance as Service IP alias valid on all nodes.
Volume groups	Specify the shared volume groups that contain the file system that is used as instance directory for the central services instance. If it is a local set up, **LOCAL** must be configured.
Database resource group	In case the database of this SAP system is clustered in the same cluster, a startup dependency can be configured.

Automatic discovery and configuration for an Enqueue replication server instance

With minimal input, you can use Smart Assist for SAP Netweaver to automatically discover and configure an Enqueue replication server (ERS) instance.

To automatically discover and configure an Enqueue replication server instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **SAP Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **SAP NW 7.0 ERS Instance** and press Enter.
5. Select the unique replication server instance name that you want to discover and configure.
6. Enter the following information for the fields in the table:

Table 65. Enqueue replication server instance discovery and configuration fields

Fields	Values
SAP System ID	Displays the value for this field that is pre-populated with the SAP system ID. This value cannot be modified.
SAP ERS Instance Name	Displays the name of Enqueue replication server instance that is selected in Step 5. This value cannot be modified.
Application Name	Specify the application name for the PowerHA SystemMirror resource groups, application servers, and other cluster components. This value must be fewer than 64 characters and follow PowerHA SystemMirror restrictions in application names.
Primary Node	Specify the node where the Enqueue replication server instance is presently running. Changing this value to a different node might cause the Enqueue replication server instance to stop on the original node and restart on changed node.
Take over nodes	Specify the take over node for this Enqueue replication server instance.
Network name	Specify the network for the service IP alias.
Service IP label	Specify the logical host name that is used to configure this Enqueue replication server instance. It must be mandatory to configure the Enqueue replication server instance as Service IP alias valid on all nodes.
Volume groups	Specify the shared volume groups that contain the file system that is used as instance directory for the Enqueue replication server instance. If it is a local setup, **LOCAL** must be configured.

Automatic discovery and configuration for an application server instance

With minimal input, you can use SAP Netweaver to automatically discover and configure an application server instance.

To automatically discover and configure an application server instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **SAP Smart Assistant** and press Enter.
4. Select **Automatic Discovery and Configuration** > **SAP NW 7.0 AS Instance** and press Enter.
5. Select the application server instance name that you want to discover and configure.
6. Enter the following information for the fields:

Table 66. Application server instance discovery and configuration fields

Fields	Values
SAP System ID	Displays the value for this field that is pre-populated with the SAP System ID. This value cannot be modified.
SAP Application Server Instance Name	Displays the name of application server instances that are selected in Step 5. This value cannot be modified.
Application Name	Specify the application name for the PowerHA SystemMirror resource groups, application servers, and other cluster components. This value must be fewer than 64 characters and follow PowerHA SystemMirror restrictions in application names.
Primary Node	Specify the node where the application server instance is presently running. Changing this value to a different node might cause the application server instances to stop on the original node and restart on changed node.
Take over nodes	Specify the take over node for this application server instance.
Network name	Press F4 to select a network name from a list. To keep PowerHA SystemMirror highly available, select one of the PowerHA SystemMirror networks from the list. The IP label is added as a service IP label in the PowerHA SystemMirror configuration.
	If you do not want PowerHA SystemMirror to keep the IP label highly available, select **LOCAL** from the list. This option is useful when the IP label is bound to a particular node in the cluster and it cannot be moved to any other node. If you select **LOCAL**, the IP label is not added as a service IP and is not part of the resource group.
IP label	Specify the logical host name that is used to configure this application server instance. This IP label is added as a service IP alias to PowerHA SystemMirror based on the network selection.
Volume groups	Specify the shared volume groups that contain the file system that is used as instance directory for the application server instance. If it is a local setup, **LOCAL** must be configured.
Database resource group	In case the database of this SAP system is clustered in the same cluster, a startup dependency can be configured.

Manual configuration and discovery

You can use a xml file to configure different SAP Netweaver instance.

Before you manual configure a SAP Netweaver instance, you must modify the `/usr/es/sbin/cluster/sa/sap/config/cl_sap_manual_config.xml` file correctly depending on the following SAP Netweaver instance:

Global file system
> Create a section with the tag name **SAP_Global_FileSystem** with the required values and change the value for the name attribute from **configure** to **true**.

Central services
> Create two sections with the tag name **ABAP_SCS_Instance** and **Java_SCS_Instance** with the required values, and change the value for the name attribute from **configure** to **true**.

Enqueue replication server
> Create two sections with the tag name **ABAP_ERS_Instance** and **Java_ERS_Instance** with the required values, and change the value for the name attribute from **configure** to **true**.

Application Server
> Create a section with the tag name **AS_Instance** with the required values, and change the value for the name attribute from **configure** to **true**.

To manually configure different SAP Netweaver instances, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** > **SAP Smart Assistant** > **Manual Configuration** and press Enter.
3. In the File Name field specify the path where the xml file is located and press Enter. You can find a sample xml here, `/usr/es/sbin/cluster/sa/filenet/config/cl_filenet_manual_config.xml`.

    Note: You can not modify the Smart Assist ID field. This ID is automatically generated.

Changing PowerHA SystemMirror resources for different SAP instances

You can use SMIT to change any of the resources associated with your different SAP Netweaver instances.

To change the resources associated with your different SAP Netweaver instances, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.
3. Select the resources for the SAP instance you want to modify and press Enter.
4. Make the changes as needed in the fields for the selected resources and press Enter. The same fields that you were able to modify when you created the configuration are available when you make changes, except for the Application Name field.

Changing the resources associated with your application

You can add resources such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, create separate resource groups for each set.

To change the resources associated with your SAP Netweaver applications, complete the following steps:

1. From the command line enter `smit sysmirror`.

2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show the Resources Associated with your Application** and press Enter.
3. Select the applications resources you want to modify and press Enter.
4. Make the changes as needed in the fields for the selected application and press Enter.

Removing an application

When you remove an application from the list of configured Smart Assist for SAP Netweaver applications, PowerHA SystemMirror stops managing the availability of the resources you remove, because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.
3. Select the resource you want to remove from the PowerHA SystemMirror configuration database on the local node and press Enter.
4. Confirm that you want to remove the application and press Enter.

   Note: Removing an application from PowerHA SystemMirror removes all analogous resources from PowerHA SystemMirror, including application controllers and monitors, service IP labels, and resource groups.

Smart Assist for SAP Netweaver resources

After SAP Netweaver has been configured through Smart Assist for SAP Netweaver, PowerHA SystemMirror creates resources.

Global file system resources

Review the PowerHA SystemMirror resources created from SAP Netweaver global file system resources.

Table 67. Global file system resources	
PowerHA SystemMirror resource	**Name**
Resource group	SAP_RG_NFS

Central service instance resources

Review the PowerHA SystemMirror resources that are created from SAP Netweaver central service instance resources.

Table 68. Central service instance resources

PowerHA SystemMirror resource	**Name**
Resource group	**SAP_<SID>_<INST_NAME>_RG**, where SID is the SID for this instance and INST_NAME is the name of the instance.
Application Server	**<application name>_AP**, where application name is the name of the central service instance application.
Custom Monitor	**<application name>_AM**, where application name is the name of the central service instance application.

Table 68. Central service instance resources (continued)

PowerHA SystemMirror resource	Name
Start Script	**cl_sapStart**, this file is in the `/usr/es/sbin/ cluster/sa/sap/sbin` directory.
Stop Script	**cl_sapStop**, this file is in the `/usr/es/sbin/ cluster/sa/sap/sbin` directory.
Monitor Script	**cl_sapMonitor**, this file is in the `/usr/es/sbin/ cluster/sa/sap/sbin` directory.
Startup Policy	Online on first Available Node
Fallover Policy	Fallover To Next Priority Node In The List
Fallback Policy	Never fallback

The following table describes the default settings associated with custom monitor (<application name>_AM) for central services instances that are displayed in Table 1.

Table 69. Central services instances custom monitor (<application name>_AM) default settings

Field	Value
Name	**<application name>_AM**
Application Servers to Monitor	**<application name>_AP**
Monitor Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapMonitor –a <application name>
Mode	Long-running monitoring
Interval	60 sec
Hung Monitor Signal	9
Stabilization interval	240 sec
Restart Count	0
Restart Interval	0
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStop –a <application name>
Restart Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStart –a <application name>

Enqueue replication server instance resources

Review the PowerHA SystemMirror resources that are created from SAP Netweaver Enqueue replication server (ERS) instance resources.

Table 70. Enqueue replication server instance resources

PowerHA SystemMirror resource	Name
Resource group	SAP_<SID>_<INST_NAME>_RG
Application Server	**<application name>_AP**, where application name is the name of the Enqueue replication server instance application.

Table 70. Enqueue replication server instance resources (continued)

PowerHA SystemMirror resource	Name
Custom Monitor	**<application name>_AM**, where application name is the name of the Enqueue replication server instance application.
Start Script	**cl_sapStart**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Stop Script	**cl_sapStop**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Monitor Script	**cl_sapMonitor**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Startup Policy	Online On Home Node Only
Fallover Policy	Fallover To Next Priority Node In The List
Fallback Policy	Never fallback

The following table describes the default settings associated with the custom monitor (<application name>_AM) for the Enqueue replication server that is displayed in Table 1.

Table 71. Enqueue replication server instances (<application name>_AM) default settings

Field	Value
Name	<application name>_AM
Application Servers to Monitor	sap_ers_appserver_AP
Monitor Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapMonitor –a <application name>
Mode	Long-running monitoring
Interval	60 sec
Hung Monitor Signal	9
Stabilization interval	240 sec
Restart Count	3
Restart Interval	990 sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStop –a <application name>
Restart Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStart –a <application name>

Application server instance resources

Review the PowerHA SystemMirror resources that are created from SAP Netweaver application server instance resources.

Table 72. Application server instance resources

PowerHA SystemMirror resource	Name
Resource group	SAP_<SID>_<INST_NAME>_RG

Table 72. Application server instance resources (continued)

PowerHA SystemMirror resource	Name
Application Server	**<application name>_AP**, where application name is the name of the application server instance application.
Custom Monitor	**<application name>_AM**, where application name is the name of the application server instance application.
Start Script	**cl_sapStart**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Stop Script	**cl_sapStop**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Monitor Script	**cl_sapMonitor**, this file is in the `/usr/es/sbin/cluster/sa/sap/sbin` directory.
Startup Policy	Online on first Available Node
Fallover Policy	Fallover To Next Priority Node In The List
Fallback Policy	Never fallback

The following table describes the default settings associated with the custom monitor (<application name>_AM) for application server instance that is displayed in Table 1.

Table 73. Application server instances custom monitor (<application name>_AM) default settings

Field	Value
Name	<application name>_AM
Application Servers to Monitor	<application name>_AP
Monitor Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapMonitorAS –a <application name>
Mode	Long-running monitoring
Interval	60 sec
Hung Monitor Signal	9
Stabilization interval	240 sec
Restart Count	1
Restart Interval	990 sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStop –a <application name>
Restart Method	/usr/es/sbin/cluster/sa/sap/sbin/cl_sapStart –a <application name>

Troubleshooting Smart Assist for SAP Netweaver

To successfully troubleshoot Smart Assist for SAP Netweaver you must know where the different types of data are stored.

Smart Assist for SAP Netweaver writes different types of data into different log locations.

Table 74. Smart Assist for SAP Netweaver data locations

Type of data	Location
Configuration data	Smart Assist for SAP stores configuration data in the HACMPsa_metadata PowerHA SystemMirror Configuration Database (ODM).
Log data	The **var/hacmp/log/sapsa.log** file is used while Smart Assist for SAP Netweaver configures different SAP instances.
	The **/var/hacmp/log/<rgname>_<application name>.log** file is used by the start and stop monitor scripts.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   - View the log files in the Table 74 on page 153 table.
   - The **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
   - The **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to make sure all your clusters are using the latest configuration settings.

Smart Assist for SAP MaxDB

Use Smart Assist for SAP MaxDB to setup MaxDB/LiveCache database instances for high availability.

With Smart Assist for SAP MaxDB you can do the following:

- Configure a MaxDB/LiveCache database instance by creating the appropriate PowerHA SystemMirror objects such as resource groups, application servers, application monitor, and service IPs.
- Discover all the configured MaxDB/LiveCache database instances running in the cluster and its associated resources such as volume groups and logical host names.
- Manage all MaxDB/LiveCache database instance for high availability by staring or stopping monitor scripts.

Smart Assist for SAP MaxDB planning

Before you can install Smart Assist for SAP MaxDB you must have your environment setup correctly.

To correctly configure a SAP MaxDB/LiveCache database instance for high availability using Smart Assist for SAP MaxDB, review the following information:

- SAP MaxDB must be installed with its own logical hostname. If you install the SAP MaxDB as part of any SAP Software, for example SAP SCM, you must use SAP_USE_HOSTNAME environment variable when running the **sapinst** installation executable. If SAP MaxDB is going to be used outside of SAP applications, see the SAP MaxDB Libary for instructions.
- The SAP MaxDB software can be installed on any node in the cluster, however, the installation path and related files must reside on a file system that is setup as a shared disk.
- The administrator User ID for the SAP MaxDB database instance must be setup on all failover nodes with the same user ID and group ID.
- Verify that all SAP MaxDB database instances can run successfully on all the failover nodes by completing the following tasks:

  – Run the SAP MaxDB instance, by manually varying on all the shared disks used for storing SAP MaxDB program files and database instance files, and set up the logical IP addresses.

– Resolve any possible licensing issues by running the SAP MaxDB software.

The following figure displays the setup for a typical two node cluster running a SAP MaxDB database instance for High Availability.
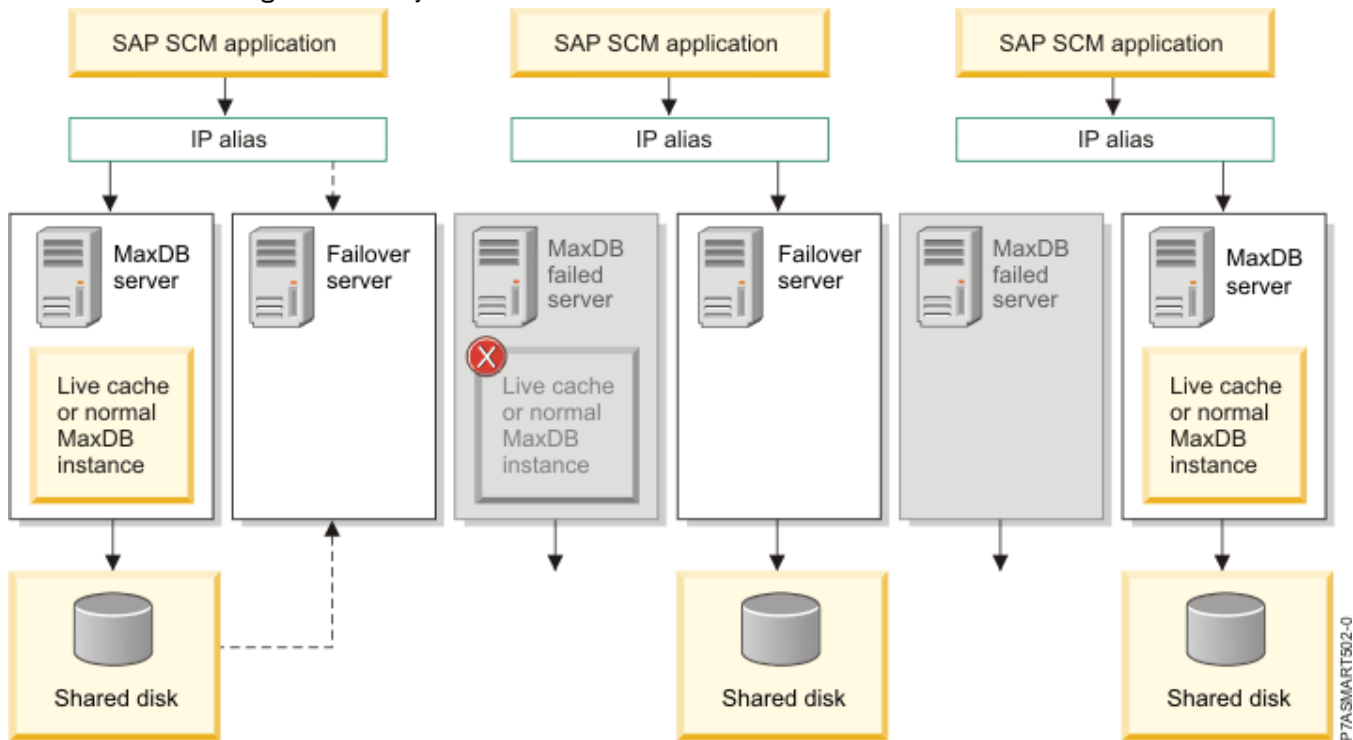

Figure 12. Typical two node cluster running a SAP MaxDB database instance for High Availability

Software Requirements

To view what versions of SAP MaxDB are supported, see the "Support matrix for PowerHA SystemMirror Smart Assists" on page 1 topic.

Configuring Smart Assist for SAP MaxDB

You must initiate Smart Assist for SAP MaxDB from a node where the SAP MaxDB database instance is already online and running.

A cluster has to be configured correctly before initiatingSmart Assist for SAP MaxDB, otherwise you will be automatically redirected to the 2-node cluster configuration assistant interface to configure a 2-node cluster.

Automatic discovery and configuration

Using SMIT you can setup Smart Assist for SAP MaxDB to automatically discover and configure database instances running in the cluster along with its resources like volume groups and service IP addresses.

To setup automatic discover and configuration, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
3. From the list of applications, select **SAP MaxDB Smart Assistant**, and press Enter.
4. Select **Automatic Discovery and Configuration** > **SAP MaxDB/LiveCache Instances**, and press Enter.
5. Select the database instance that you want to be automatically discovered and configured, and press Enter.

6. Enter the following information for the fields:

Table 75. Smart Assist for SAP MaxDB automatic discover and configuration fields

Fields	Values
Application Name	Enter the name for the collection of PowerHA SystemMirror resources that represent the SAP MaxDB database instance.
MaxDB Instance Name	Displays the name of the database instance you selected in Step 5. This field can not be modified.
MaxDB Admin User	Enter the administrator user name for the database instance.
MaxDB Admin Password	Enter the administrator password for the database instance.
Primary Node	Displays the name of the node where the database instance is installed. This field cannot be modified.
Takeover Nodes	Enter (or select from the picklist) the name of one or more cluster nodes to which the application may fallover.
Service Interface	Enter the logical hostname used to configure the previously listed database instance.
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.
Shared Volume Group	Enter the volume group associated to store the SAP MaxDB program files and instance related files.

7. Verify the fields are correct and press Enter.

Manual discovery and configuration

You can use a xml file to configure Smart Assist for SAP MaxDB.

To manually configure Smart Assist for SAP MaxDB, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications, select **SAP MaxDB/LiveCache Instances** > **Manual Configuration** and press Enter.
4. In the File Name field specify the path where the xml file is located and press Enter. You can find a sample xml here, `/usr/es/sbin/cluster/sa/filenet/config/cl_filenet_manual_config.xml`.

Note: You can not modify the Smart Assist ID field. This ID is automatically generated.

Changing SAP MaxDB PowerHA SystemMirror resources

Once a MaxDB/LiveCache database instance has been configured for PowerHA SystemMirror, you can change the properties of the application.

To change the SAP MaxDB PowerHA SystemMirror resources, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show an Application's PowerHA SystemMirror Configuration** and press Enter.
3. From the list of applications select the application you want to make changes to and press Enter.
4. Make the changes you want in the fields and press Enter. You can change the values for all fields except the Application Name field.

Changing resources associated with your MaxDB/LiveCache database instance

You can add resources such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want a set of resources to be acquired by one node and another set acquired by a different node, you can create separate resource groups for each set.

To change the resources associated with your MaxDB/LiveCache database instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Change/Show the Resources Associated with your Application** and press Enter.
3. From the list of applications select the application you want to make changes to and press Enter.
4. Make the changes you want in the fields and press Enter.

Related information
Managing resource groups

Removing an application

When you remove an application from the list of configured Smart Assist for SAP MaxDB applications, PowerHA SystemMirror stops managing the availability of the resources you remove, because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove an application, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration** and press Enter.
3. From the list select the application you want to remove and press Enter.
4. From the list select the resource you want to remove and press Enter.
5. Press Enter to remove the resource from the PowerHA SystemMirror Configuration Database on the local node.

   Note: Removing a resource an from a application removes all analogous resources from PowerHA SystemMirror, including application controllers and monitors, service IP labels, and resource groups.

Smart Assist for SAP MaxDB resources

Once a MaxDB/LiveCache database instance has been configured through Smart Assist for SAP MaxDB, PowerHA SystemMirror will create specific resources.

The following table displays the PowerHA SystemMirror resources created from Smart Assist for SAP MaxDB.

Table 76. PowerHA SystemMirror resources

PowerHA SystemMirror resource	Name
Resource Group	**maxdb_instanceName**, where instanceName is the name of the MaxDB/LiveCache database instance
Application Server	**maxdb_instanceName_APPSERV**, where instanceName is the name of the MaxDB/ LiveCache database instance
Custom Monitor	**maxdb_instanceName_APPMON**, where instanceName is the name of the MaxDB/ LiveCache database instance
Start Script	**cl_maxdb_startDB** This file is located in the `/usr/es/sbin/cluster/sa/maxdb/sbin`
Stop Script	**cl_maxdb_stopDB** This file is located in the `/usr/es/sbin/cluster/sa/maxdb/sbin`
Monitor Script	**cl_maxdb_monitorDB,** This file is located in the `/usr/es/sbin/cluster/sa/maxdb/sbin`

The following table displays the default settings for the PowerHA SystemMirror Custom Monitor resource.

Table 77. PowerHA SystemMirror Custom Monitor resource

Field	Value
Name	maxdb_instanceName_APPMON
Application Servers to Monitor	maxdb_instanceName_APPSERV
Monitor Method	usr/es/sbin/cluster/sa/maxdb/sbin/ cl_maxdb_monitorDB –a <Application name> -I <Instance Name>
Mode	Long-running monitoring
Interval	120 Sec.
Hung Monitor Signal	9
Stabilization interval	240 Sec
Restart Count	3
Restart Interval	1440 Sec
Action on Application Failure	Fallover
Cleanup Method	/usr/es/sbin/cluster/sa/maxdb/sbin/ cl_maxdb_stopDB –a <Application name> -I <Instance Name>

Table 77. PowerHA SystemMirror Custom Monitor resource (continued)	
Field	**Value**
Restart Method	/usr/es/sbin/cluster/sa/maxdb/sbin/ cl_maxdb_startDB –a <Application name> -I <Instance Name>

Troubleshooting Smart Assist for SAP MaxDB

To successfully troubleshoot Smart Assist for SAP MaxDB it is important to know where the different types of data are stored.

Smart Assist for SAP MaxDB writes different types of data into different log locations.

Table 78. Smart Assist for SAP MaxDB data locations	
Type of data	**Location**
Configuration data	HACMPsa_medata PowerHA SystemMirror Configuration Database (ODM)
Log data	The **/var/hacmp/log/maxdbsa.log** file is used while Smart Assist for SAP MaxDB configures the database instance.
	The **/var/hacmp/log/maxdb_<application name>.log** file is used by the start monitor and stop monitor scripts.
	The logs specified contain information that is usually only used by IBM Technical Support.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   - View the log files in the <u>Table 78 on page 158</u> table.
   - The **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
   - The **hacmp.out** file for messages related to detection, migration, termination and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to make sure all your clusters are using the latest configuration settings.

Smart Assist for SAP liveCache Hot Standby

Use Smart Assist for SAP liveCache Hot Standby to set up SAP liveCache Hot Standby instances for high available environment.

Before you can use Smart Assist for SAP liveCache Hot Standby, you must have already installed the Smart Assist for SAP MaxDB filesets. There is not a Smart Assist for SAP liveCache Hot Standby menu in the SMIT interface. To configure, discover, and manage SAP liveCache Hot Standby instances you must use the Smart Assist for SAP MaxDB menu in the SMIT interface.

You can complete following tasks with Smart Assist for SAP liveCache Hot Standby:

- Configure a liveCache Hot Standby database instance by creating the appropriate PowerHA SystemMirror objects, such as resource groups, application controllers, application monitors, and service IPs.
- Discover all the configured liveCache Hot Standby database instances that are running in the cluster and its associated resources, such as volume groups and logical hostnames.

- Manage a liveCache Hot Standby database instance for high availability by staring or stopping monitor scripts to control flow of data from the SAP Advanced Planning and Optimizations (SAP APO) application.
- Not to impact the performance of the primary instance of IBM System Storage with FlashCopy, functionality is used and triggered during starts and moves to sync the standby data.

Related information

SAP MaxDB Documentation

IBM Techdocs White Paper: Invincible Supply Chain - SAP APO Hot Standby liveCache on IBM Power Systems

Smart Assist for SAP liveCache Hot Standby concepts

It is important that you have at least a basic knowledge of Smart Assist for SAP liveCache Hot Standby before you start configuring and manage it.

SAP liveCache

SAP liveCache is a database which resides in memory that is used to speed up material planning scenarios in SAP Supply Chain Management (SCM).

With SAP liveCache you can complete resource-intensive material planning activities, such as demand planning (DP) as a forecast up to on demand available to promise (ATP) scenarios.

The typical sizes of the memory cache for large implementations of SAP liveCache are 50 - 250 GB. The SAP liveCache Hot Standby solution does prevent you from the following outages:

- Rebuild of large (~ 50 -100 GB) memory structure
  - The time involved in the rebuild increases with the data size of the liveCache
- Rollback of open transactions and then a resynchronization at application level
- Resync time increases with the data size of the SCM system
- Manual corrections for miss matches can take days to resolve

With the SAP liveCache Hot Standby, the takeover still provides full data consistency of the liveCache to an already build memory structure just waiting to be activated. By using IBM System Storage with FlashCopy the primary instance does not get impacted in the performance

To minimize the recovery time and data loss resulting in the need to resynchronize from the application level use the SAP liveCache Hot Standby function.

SAP liveCache Hot Standby

With SAP liveCache Hot Standby you can maintain a standby instance of an SAP liveCache environment that when a failure occurs the standby instance can take over the SAP service immediately.

Using SAP liveCache Hot Standby provides your applications with the following features:

- Speed of recovery and return to production
- Coverage of server outage
- Coverage of database outage
- Coverage of data disk failures
- Automatic failover and fallback
- Easy management of databases

The implementation of liveCache Hot Standby is only supported on a pair of servers. These servers access a single storage system. The data-volumes for each database are separate and the log volume is shared. The liveCache Hot Standby function provides a duplicate cache filled with the most relevant data and removes the need to move physical resources other than the IP service address. The following figure

displays the setup for a typical two node cluster running a SAP liveCache Hot Standby database instance for a high available environment.
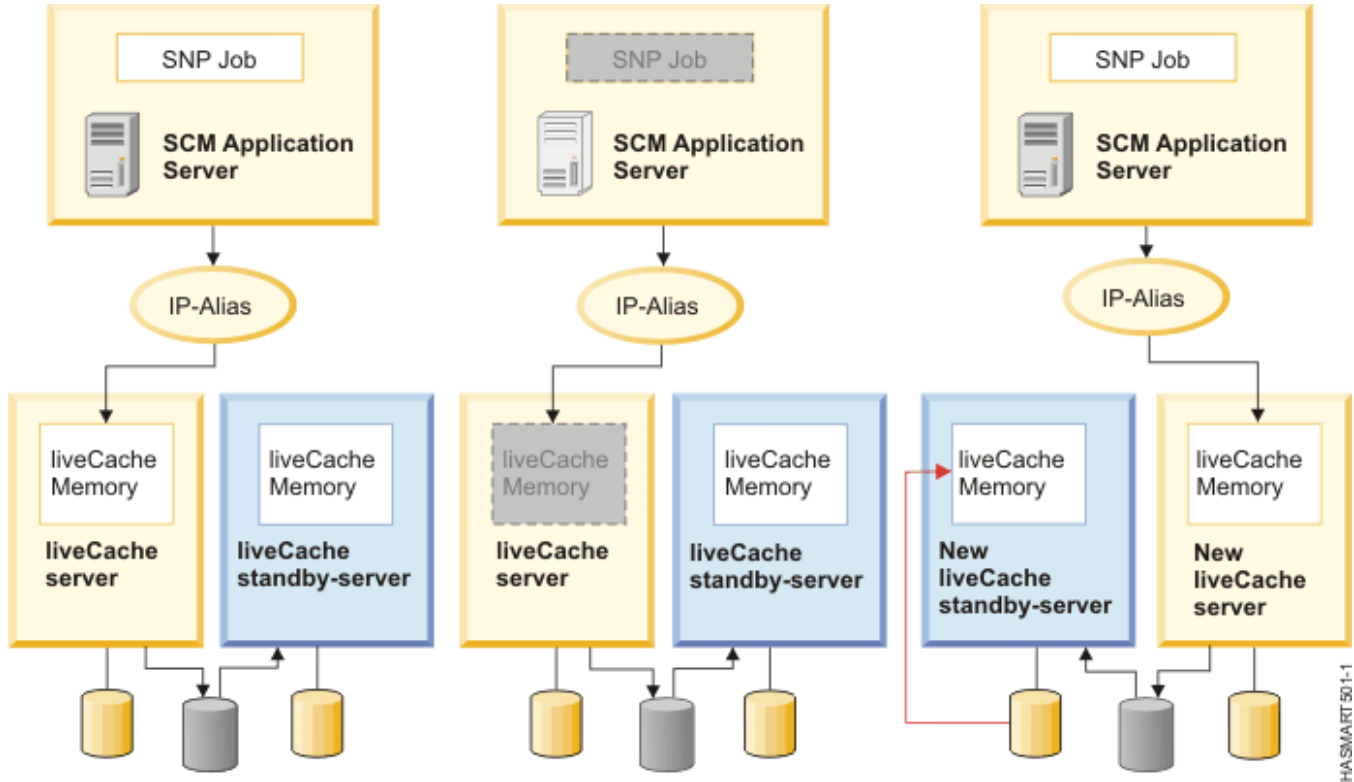

Figure 13. Two node cluster running SAP liveCache Hot Standby

Planning for Smart Assist for SAP liveCache Hot Standby

Before you can install Smart Assist for SAP liveCache Hot Standby, you must verify that your environment is working correctly.

To correctly plan for an SAP liveCache Hot Standby database instance for high availability by using Smart Assist for SAP liveCache Hot Standby, review the following information:

- All PowerHA SystemMirror file sets and PowerHA SystemMirror Smart Assists are installed and configured corrected.
- The libHSS library, which is installed with the Smart Assist for SAP MaxDB file set, exists on all clusters and nodes and is owned by the SdbOwner user and the SdbGroup group
- The SAP liveCache software is installed on the same path for all nodes in the cluster. You must have the same **SdbOwner** user name and **SdbGroup** group name on all the nodes with same numeric values for user ID and group ID on all of the cluster nodes.
- The SAP liveCache log volume group or groups must be created with raw logical volumes with a maximum of 128 GB of space. The volume group must be an enhanced concurrent volume in concurrent mode that is activated on both nodes.
- The SAP liveCache data volume group must be created and imported on all node in the cluster and must be active. The raw logical volumes that belong to the SAP liveCache data volume group must be controlled by the SdbOwner user and the SdbGroup group.
- The global file system must be accessible to all the nodes in the cluster to be used as the SAP liveCache lock directory.
- A Secure Shell (SSH) connection is configured correctly for the **SdbOwner** user, if an IBM SAN Volume Controller (SVC) storage disk is used for an SAP liveCache Hot Standby database instance.
- DSCLI software is installed in the /opt/ibm/dscli (default location) directory if an IBM System Storage® DS8000® disk is used for the SAP liveCache Hot Standby configuration.

- All the nodes have the same independent data path and independent program path that is identified in the `/etc/opt/sdb` file, and the same dependent path. The dependent path is identified by running the `dbmcli db_enum INSTANCENAME <instance_name> PATH` command.
- Verify that all SAP MaxDB primary and standby database instances are running successfully on all the failover nodes by completing the following tasks:
  - Run the SAP liveCache Hot Standby instance from SAP Advanced Planner and Optimizer (APO), by manually varying on all the disks that are used for storing SAP liveCache Hot Standby program files and database instance files. You must also configure the logical IP addresses.
  - Resolve any possible licensing issues by running SAP liveCache software.

When you add a disk to the AIX operating system, some of the storage subsystems enable an SCSI-2 reservation policy by default. This reservation policy might cause problems to the shared disk across the cluster, and can cause problems when you use the SAP liveCache Hot Standby wizard. You must verify that the disk and the related volume group do not have any reservation policy settings before you start the SAP liveCache Hot Standby wizard.

Before you can use the SAP liveCache Hot Standby wizard, you must manually change the value of the reservation policy by running the **chdev -l hdisk1 -a reserve_policy=no_reserve** command, where `hdisk1` is the name of the storage device. To check whether the reservation policy is already configured to run the SAP liveCache Hot Standby wizard, run the **lsattr -El hdisk1 -a reserve_policy** command, where `hdisk1` is the name of the storage device.

PowerHA SystemMirror SAP liveCache Hot Standby is supported with FlashCopy® mechanisms IBM System Storage DS8000 and SVC storage subsystems. You can only configure the SAP liveCache Hot Standby wizard to work with one of these storage subsystems. The SAP liveCache Hot Standby wizard will not work with any other storage subsystem type. The SAP liveCache Hot Standby will work with direct attached storage devices or N_Port ID Virtualization (NPIV) with Virtual I/O Server (VIOS) configured disks. The automatic configuration option of The SAP liveCache Hot Standby wizard will not work for virtual Small Computer System Interface (vSCSI) with VIOS configured disks. For vSCSI disks, you can use manual configuration.

To manually configure SAP liveCache Hot Standby using SAP liveCache Hot Standby wizard, complete the following steps:

1. From the command line, enter **smit sysmirror**.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **SAP liveCache Hot Standby Configuration Wizard**, and press Enter.
3. From the list of node names, select both the nodes and press Enter.
4. Select the option `Manual Configuration` and press enter.
5. In the `File Name` field specify the path where the xml file is located and press Enter. You can find a sample xml at the, `/usr/es/sbin/cluster/sa/hswizard/config/hswizard_config.xml` location.

Note: Multiple SAP liveCache Hot Standby database instances for each primary database instance are not supported for the current releases of SAP liveCache Hot Standby and Smart Assist for SAP liveCache Hot Standby.

Software requirements

To view what versions of SAP liveCache Hot Standby are supported, see the topic.

Configuring Smart Assist for SAP liveCache Hot Standby

A cluster has to be configured correctly before initiating Smart Assist for SAP liveCache Hot Standby.

You must have both SAP liveCache primary database instance and SAP liveCache Hot Standby database instance online and running before you start Smart Assist for SAP liveCache Hot Standby.

You must start Smart Assist for SAP liveCache Hot Standby from a node where the SAP liveCache primary database instance is online and running.

Automatic discovery and configuration for Smart Assist for SAP liveCache Hot Standby

By using SMIT you can setup Smart Assist for SAP liveCache Hot Standby to automatically discover and configure database instances that are running in the cluster along with its resources such as volume groups and service IP addresses.

To setup automatic discover and configuration, complete the following steps:

1. Export the *LC_XUSER=<XUSERname>* variable.
2. From the command line enter `smit sysmirror`.
3. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
4. From the list of applications, select **SAP MaxDB Smart Assist**, and press Enter.
5. Select **Automatic Discovery and Configuration** > **SAP liveCache Hot Standby Database Instance**, and press Enter.
6. Select the database instance that you want to be automatically discovered and configured, and press Enter.
7. Enter the following information for the fields:

Table 79. Smart Assist for SAP liveCache Hot Standby automatic discover and configuration fields

Field	Value
liveCache Instance Name	Displays the name of the database instance that you selected in step 5. This field cannot be modified.
SAP liveCache Hot Standby Instance DBM user XUSER	Enter the name of the XUSER, created with DBM user credentials.
SAP liveCache Hot Standby Global Filesystem Mount Point	Enter a global file system mount point configure PowerHA SystemMirror. The file system is used as a lock directory for SAP liveCache Hot Standby. The mount point must always be available on all nodes in the cluster.
Primary Node	Displays the name of the node where the database instance is installed. This field cannot be modified.
Takeover Nodes	Enter (or select from the pick list) the name of one or more cluster nodes to which the application might fallover.
Service Interface	Enter the logical hostname used to configure the previously mentioned database instance.

Table 79. Smart Assist for SAP liveCache Hot Standby automatic discover and configuration fields (continued)

Field	Value
Netmask (IPv4) / Prefix Length (IPv6)	For the configuration of IPv4 service interface, enter the network mask for the address. For the configuration of IPv6 service interface, enter the prefix length for the address. This is not a mandatory field. If you do not enter a value, the prefix length or netmask of the underlying network is used. If a prefix length value or netmask value is specified, it is checked for compatibility with the underlying network.
Data Volume Group	Enter the volume group associated with the SAP liveCache Hot Standby data volumes.
Log Volume Group	Enter the volume group associated with the SAP liveCache Hot Standby log volumes.

8. Verify that the fields are correct and press Enter.

Manual discovery and configuration for Smart Assist for SAP liveCache Hot Standby

You can use an xml file to configure Smart Assist for SAP liveCache Hot Standby.

To manually configure Smart Assist for SAP liveCache Hot Standby, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (Use Smart Assists)** > **Add an Application to the PowerHA SystemMirror Configuration**, and press Enter.
3. From the list of applications, select **SAP MaxDB Smart Assist** > **Manual Configuration**, and press Enter.
4. In the **File Name** field specify the path where the xml file is located and press Enter. You can find a sample xml at the, `/usr/es/sbin/cluster/sa/maxdb/config/maxdb_hs_config.xml` location.

   Note: You cannot modify the Smart Assist ID field. This ID is automatically generated.

Managing Smart Assist for SAP liveCache Hot Standby

You can change the Smart Assist for SAP liveCache Hot Standby configuration or resources, and remove Smart Assist for SAP liveCache Hot Standby applications with the SMIT interface.

Changing SAP liveCache Hot Standby PowerHA SystemMirror resources

After a SAP liveCache Hot Standby database instance has been configured for PowerHA SystemMirror, you can change the properties of the application.

To change the SAP liveCache Hot StandbyPowerHA SystemMirror resources, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show an Applications PowerHA SystemMirror Configuration**, and press Enter.
3. Select the required application that you want to change, and press Enter.

4. Change the fields that you want for the selected application, and press Enter. You can change the values for all fields except the **Application Name** field.

   Note: To add or remove an application monitor for a WebSphere MQ listener from the Smart Assist application configuration, you must first remove the Smart Assist application from PowerHA SystemMirror by using SMIT and configure the Smart Assist application again. For more information, see Removing an application and Configuring Smart Assist for WebSphere MQSeries.

Changing resources associated with the SAP liveCache Hot Standby database instance

You can add resources, such as file systems and volume groups to the resource group. These resources are always acquired and released as a single entity. If you want to a set of resources to be acquired by one node and another set acquired by a different node, you can create separate resource groups for each node.

To change the resources associated with your SAP liveCache Hot Standby database instance, complete the following steps:

1. From the command line enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Change/Show the Resources Associated with your Application Configuration**, and press Enter.
3. Select the applications resources that you want to modify, and press Enter.
4. Change the fields that you want for the selected application, and press Enter.

Removing a Smart Assist for SAP liveCache Hot Standby application

When you remove an application from the list of configured Smart Assist for SAP liveCache Hot Standby applications, PowerHA SystemMirror stops managing the availability of the resources that you remove because they are no longer part of the PowerHA SystemMirror cluster definition.

To remove a Smart Assist for SAP liveCache Hot Standby application, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. From the SMIT interface, select **Cluster Applications and Resources** > **Make Applications Highly Available (use Smart Assists)** > **Remove an Application from the PowerHA SystemMirror Configuration**, and press Enter.
3. Select the application that you want to remove from the PowerHA SystemMirror configuration database on the local node, and press Enter.
4. Confirm that you want to remove the application, and press Enter.

   Note: Removing an application from PowerHA SystemMirror removes all analogous resources from PowerHA SystemMirror, which includes application controllers and monitors, service IP labels, and resource groups.

Smart Assist for SAP liveCache Hot Standby resources

After a SAP liveCache Hot Standby database instance has been configured through Smart Assist for SAP liveCache Hot Standby, PowerHA SystemMirror creates specific resources.

PowerHA SystemMirror resources for SAP liveCache Hot Standby log volumes

Resource groups are created by Smart Assist for SAP liveCache Hot Standby for managing SAP liveCache Hot Standby log volumes.

The following table displays the resource group properties for SAP liveCache Hot Standby log volumes.

Table 80. Resource group properties for SAP liveCache Hot Standby log volumes

Field	Value
Name	RG_Log_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby instance.
Startup policy	Online on all available nodes
Fallover policy	Bring offline (on error node only)
Fallback policy	Never fallback
Volume groups	Displays the log volumes for the volume group of the instance
Automatically import volume groups	False
Application controllers	as_log_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby instance.

The following table displays the application monitor properties for SAP liveCache Hot Standby log volumes.

Table 81. Application monitor properties for SAP liveCache Hot Standby log Volumes

Field	Value
Name	mon_log_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Application controllers to monitor	as_log_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Method	The `/usr/es/sbin/cluster/sa/maxdb/sbin/cl_hotstandby_logmonitor`
Monitor mode	Long-running monitoring
Monitor interval	60
Hung monitor signal	30
Stabilization interval	180
Restart count	0
Restart interval	0
Action on application failure	fallover
Cleanup method	The `/usr/es/sbin/cluster/sa/maxdb/sbin/cl_hotstandby_cleanDB` *STOP instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Restart method	The `/usr/es/sbin/cluster/sa/maxdb/sbin/cl_hotstandby_cleanDB` *START instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

PowerHA SystemMirror resources for the SAP liveCache Hot Standby primary instance

Resource groups, application controllers, and application monitors are created by Smart Assist for SAP liveCache Hot Standby for managing the SAP liveCache Hot Standby primary instance.

The following table displays the resource group properties for the SAP liveCache Hot Standby primary instance.

Table 82. Resource group properties for the SAP liveCache Hot Standby primary instance.

Field	Value
Name	RG_Primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Startup policy	Online on first available node
Fallover policy	Fallover to next priority node in the list
Fallback policy	Never fallback
Service IP label	IP labels, which are used by the primary instance are displayed.
Application controllers	as_primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

The application controller is responsible for starting and stopping the SAP liveCache Hot Standby primary instance. The following table displays the application controller properties for SAP liveCache Hot Standby primary instances.

Table 83. Application controller properties for the SAP liveCache Hot Standby primary instance.

Field	Value
Name	as_primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Start script	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_startDB "Primary" "instance name"` script is used to start the primary instance. Where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Stop script	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_stopDB "Primary" "instance name"` script is used to stop the primary instance. Where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

The following table displays the startup application monitor properties for SAP liveCache Hot Standby primary instances.

Table 84. Startup application monitor properties for the SAP liveCache Hot Standby primary instance.	
Field	**Value**
Name	mon_primary_startup_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Application controllers to monitor	as_primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Method	The `/usr/es/sbin/cluster/sa/maxdb/ cl_hotstandby_startupmonitor "Primary" "instance name"`, where *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Monitor mode	Startup monitoring
Monitor interval	10
Hung monitor signal	9
Stabilization interval	120
Restart count	0
Restart interval	77
Action on application failure	Notify
Cleanup method	The `/usr/es/sbin/cluster/sa/maxdb/ cl_hotstandby_stopDB "Primary" "instance name"`, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Restart method	The `/usr/es/sbin/cluster/sa/maxdb/ cl_hotstandby_startDB "Primary" "instance name"`, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

The following table displays the application monitor properties for SAP liveCache Hot Standby primary instances.

Table 85. Application monitor properties for the SAP liveCache Hot Standby primary instance.	
Field	**Value**
Name	mon_primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Application controllers to monitor	as_primary_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Method	The `/usr/es/sbin/cluster/sa/maxdb/ cl_hotstandby_monitorDB "Primary" "instance name"`, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

Table 85. Application monitor properties for the SAP liveCache Hot Standby primary instance. (continued)	
Field	**Value**
Monitor mode	Long-running monitoring
Monitor interval	20
Hung monitor signal	9
Stabilization interval	180
Restart count	0
Restart interval	0
Action on application failure	Fallover
Cleanup method	The `/usr/es/sbin/cluster/sa/maxdb/` `cl_hotstandby_stopDB "Primary"` `"instance name"`, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.
Restart method	The `/usr/es/sbin/cluster/sa/maxdb/` `cl_hotstandby_startDB "Primary"` `"instance name"`, where, *instance name* is the name for the SAP liveCache Hot Standby primary instance.

PowerHA SystemMirror resources for the SAP liveCache Hot Standby standby instance

Resource groups, application controllers, and application monitors are created by Smart Assist for SAP liveCache Hot Standby for managing the SAP liveCache Hot Standby standby instance.

The following table displays the resource group properties for the SAP liveCache Hot Standby standby instance.

Table 86. Resource group properties for the SAP liveCache Hot Standby standby instance.	
Field	**Value**
Name	RG_standby_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Startup policy	Online on home node only
Fallover policy	Fallover to next priority node in the list
Fallback policy	Never fallback
Application controllers	as_standby_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.

The application controller is responsible for starting and stopping the SAP liveCache Hot Standby standby instance. The following table displays the application controller properties for SAP liveCache Hot Standby standby instances.

Table 87. Application controller properties for the SAP liveCache Hot Standby standby instance.

Field	Value
Name	as_standby_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Start script	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_startDB "standby" "`*instance name*`"` script is used to start the standby instance. Where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Stop script	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_stopDB "standby" "`*instance name*`"` script is used to stop the standby instance. Where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.

The following table displays the application monitor properties for SAP liveCache Hot Standby standby instances.

Table 88. Application monitor properties for the SAP liveCache Hot Standby standby instance.

Field	Value
Name	mon_standby_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Application controllers to monitor	as_standby_*instance name*, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Method	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_monitorDB "standby" "`*instance name*`"`, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.
Monitor mode	Long-running monitoring
Monitor interval	10
Hung monitor signal	9
Stabilization interval	120
Restart count	3
Restart interval	429
Action on application failure	Fallover
Cleanup method	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_stopDB "standby" "`*instance name*`"`, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.

Table 88. Application monitor properties for the SAP liveCache Hot Standby standby instance. (continued)	
Field	**Value**
Restart method	The `/usr/es/sbin/cluster/sa/maxdb/cl_hotstandby_startDB "standby" "`*instance name*`"`, where, *instance name* is the name for the SAP liveCache Hot Standby standby instance.

Troubleshooting Smart Assist for SAP liveCache Hot Standby

To successfully troubleshoot Smart Assist for SAP liveCache Hot Standby you must know where the different types of data are stored.

Use the following table to identify where Smart Assist for SAP liveCache Hot Standby writes different types of data and the locations of the data.

Table 89. Smart Assist for SAP liveCache Hot Standby data locations	
Type of data	**Location**
Configuration data	Smart Assist for SAP liveCache Hot Standby stores configuration data in the **HACMPsa_metadata** PowerHA SystemMirror Configuration Database (ODM).
Log data	The `/var/hacmp/log/maxdbsa.log` file is used by SAP liveCache Hot Standby Smart Assist to log messages. The logs contain information that is used by IBM Technical Support.

To troubleshoot configuration issues, complete the following steps:

1. Review the following log files for information related to the time or type of event that occurred:

   - View the log files in the Table 1 table.
   - View the **smit.log** file to examine the SMIT processes that have run from the SMIT interface.
   - View the **hacmp.out** file for messages related to detection, migration, termination, and execution of a PowerHA SystemMirror cluster for an application.

2. Verify and synchronize your cluster to verify that all your clusters are using the latest configuration settings.

Notices

This information was developed for products and services offered in the US.

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

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

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.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

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

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

171

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as the customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Microsoft is a trademark 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.