Red Hat Reference Architecture Series

Deploying Highly Available SAP NetWeaver-based Servers Using Red Hat Enterprise Linux HA add-on with Pacemaker

Dieter Thalmayr (Magnum Opus GmbH)
Dieter Jäger (Magnum Opus GmbH)
Frank Danapfel (Red Hat)

Version 1.0
March 2014

redhat.

100 East Davie Street
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701

The GPG fingerprint of the security@redhat.com key is:
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Send feedback to refarch-feedback@redhat.com

Comments and Feedback

In the spirit of open source, we invite anyone to provide feedback and comments on any reference architectures. Although we review our papers internally, sometimes issues or typographical errors are encountered. Feedback allows us to not only improve the quality of the papers we produce, but allows the reader to provide their thoughts on potential improvements and topic expansion to the papers.

Feedback on the papers can be provided by emailing refarch-feedback@redhat.com. Please refer to the title within the email.

Staying In Touch

Join us on some of the popular social media sites where we keep our audience informed on new reference architectures as well as offer related information on things we find interesting.

Like us on Facebook:

https://www.facebook.com/rhrefarch

Follow us on Twitter:

https://twitter.com/RedHatRefArch

Plus us on Google+:

https://plus.google.com/u/0/b/114152126783830728030/

Table of Contents

1 Introduction

1.1 Executive Summary

SAP NetWeaver-based systems (like SAP ERP, SAP CRM) play an important role in many business processes today. This makes it critical for many companies that servers running SAP NetWeaver-based systems are highly available. Upholding computers systems' availability to 99.99+ percent of the time is what HA(High Availability) Clustering can achieve. The underlying idea of Clustering is a fairly simple one: Not a single large machine bears all of the load and risk; but rather one or more machines automatically drop in as an instant full replacement for the one service or even the whole machine that failed. In the best of cases, this process is seamless to the systems' users who are unaware of any change. .

While companies' usually opt for “100 percent availability”, real-world scenarios teach us that this is not achievable. Double redundancy would provide for 99.99 percent availability, every increase in that matter adds another 9 at the end of the numbers behind the comma. But even 99.999% availability means up to 5.5 min downtime a year.

SAP, being the ERP developer, leaves Redundancy and Clustering to Infrastructure developers. Every High-Availability solution for SAP is a Third-Party Solution as a result. In fact, several approaches may lead to a clustered SAP system, which may become vast and very complicated according to the customers' needs.

This document describes how to set up a two-node-cluster solution that conforms to the guidelines for high availability that have been established by by both SAP and Red Hat. It is based on SAP NetWeaver on top of Red Hat Enterprise Linux 6.5 with RHEL HA Add-on. In addition to describing the the technical prerequisites for setting up a highly available SAP NetWeaver system on RHEL, it also provides provides an example showing a working SAP HA solution using Red Hat Enterprise Linux HA add-on with Pacemaker. It is also possible to make other SAP environments (e. g. SAP simple stack with only one instance containing all SAP services)) highly available, however we highly recommend to have such setups reviewed by Red Hat support (via an Architecture Review) or by an experienced consultant.

The reference architecture described in this whitepaper has passed all tests of the SAP High Availability Interface certification in December 2013. (see http://scn.sap.com/docs/DOC-31701)

SAP® Certified
Integration with SAP NetWeaver®

1.2 Audience

This document is intended for SAP and Red Hat certified and trained administrators and consultants who already have experience setting up high available solutions using the RHEL HA add-on or other clustering solutions. Access to both SAP Service marketplace and Red Hat Customer Portal. is required to be able to download software and additional documentation.

Customers may find that the solution their enterprise demand is more complex than the working solution in this document. As a result, it is recommended that Clustering Setup and subsequent Servicing should always be accompanied by an able Red Hat consultant.

2 Concepts and Planning

2.1 Central Components of an SAP HA-Cluster

To achieve the highest availability both hardware and software components must be kept redundant.

All of the following hardware components must be duplicated (minimum requirement):

- Server hardware
- Power supply per server hardware, each connected to a different power circuit
- Network interface card per network connection
- Network Router resp. Switches
  - in case of SAN/FC, two FC-Adaptors
  - in case of SAN/FC, two FC-Switches

Neither planning nor configuration or installation of said components is the matter of this Whitepaper, nevertheless it is crucial to have them working for an availability of 99,99% or higher.

2.2 Red Hat Enterprise Linux

Red Hat Enterprise Linux is a supported and certified Open Source GNU/Linux distribution. It provides a state-of-the-art Unix System V compatible user environment, including built-in multiuser support and network redundancy. The operating system may be installed from the provided bootable DVD, or by one of the Red Hat Datacenter Deployment technologies.

2.2.1 Red Hat Enterprise Linux HA Add-On

The RHEL HA Add-On provides the following components:

- the cluster infrastructure manager (CMAN), providing communication between the cluster nodes. This communication interface is provided by the corosync daemon. Corosync is responsible for distributing information between the cluster nodes.
- The cluster manager uses CMAN to monitor the status of all cluster nodes and also monitors the status of all configured cluster resources. In case of status change pacemaker decides about any necessary changes in the cluster, based on the rules provided by the administrator. It then calculates a migration path and in close communication with CMAN, performs all migrations to the new status. Starting with Red Hat Enterprise Linux 6.5 the RHEL HA Add-on supports Pacemaker as a new cluster resource manager in addition to the already existing rgmanager.
- Resource agents which are responsible for managing specific resources, like IP addresses, filesystems, databases or SAP Instances.

- Fencing Agents which enable the cluster to control the status of the servers itself running the cluster nodes
- The “sap_redhat_cluster_connector” script which allows the integration of the RHEL HA add-on components with SAP management tools like SAP MMC

2.3 Implementing SAP services in a cluster

2.3.1 SAP components' overview

In an SAP NetWeaver environment, these services must be considered:

- Database (DBMS)
- SAP (A)SCS (System Central Services)
  - Enqueue Server: Database Lock management
  - Message Server: communication between the SAP instances
- Enqueue Replication Server (ERS): provides a duplicate of the lock table of the enqueue server
- SAP Application Servers

According with SAP architecture and capabilities, high availability for each component included in a SAP system can be achieved by different strategies. Some components, considered *Single Points of Failure* for the whole system require a infrastructure cluster. Availability for other components can be provided by using several active instances.

The following table shows the main SAP components for ABAP systems and how high availability may be achieved.

Component	Number of components	High Availability
DBMS	1 for SAP System	Infrastructure Cluster
Enqueue Server	1 for SAP System (included in ASCS instance). Enqueue Replication Server provides enqueue server resilience.	Infrastructure Cluster
Message Server	1 for SAP System (included in ASCS instance)	Infrastructure Cluster
Dialog work process	1 or more for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
Update work process	1 or more for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
Batch work process	0 or more for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
Spool work process	1 or more for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
Gateway	1 for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
SAP System Mount Directory	1 for SAP System	Infrastructure Cluster (Highly Available NFS Service) or GFS2
ICM	1 for ABAP instance	Infrastructure Cluster or Several Active ABAP Instances
Web Dispatcher	1 or several Web Dispatcher processes	Infrastructure Cluster or Several Active Instances with load balancing

Table 2.3.1.1: Critical components in ABAP stack

The following table shows the main SAP components for JAVA systems and how high availability may be achieved.

Component	Number of components	High Availability
DBMS	1 for SAP System	Infrastructure Cluster
Enqueue Server	1 for SAP System (included in SCS instance). Enqueue Replication Server provides further enqueue server resilience.	Infrastructure Cluster
Message Server	1 for SAP System (included in SCS instance)	Infrastructure Cluster
Java Dispatcher	1 for Java instance	Infrastructure Cluster or Several Active Java Instances
Java Server Process	1 for Java instance	Infrastructure Cluster or Several Active Java Instances
ICM (NW 7.1)	1 for Java instance	Infrastructure Cluster or Several Active Java Instances

Table 2.3.1.2: Critical components in Java stack

Figure 2.3.2.1 shows details about the implementation of SAP components in a single stack mutual failover cluster in a two node cluster including DBMS, Central Services, Enqueue Replication Server and Application instance.

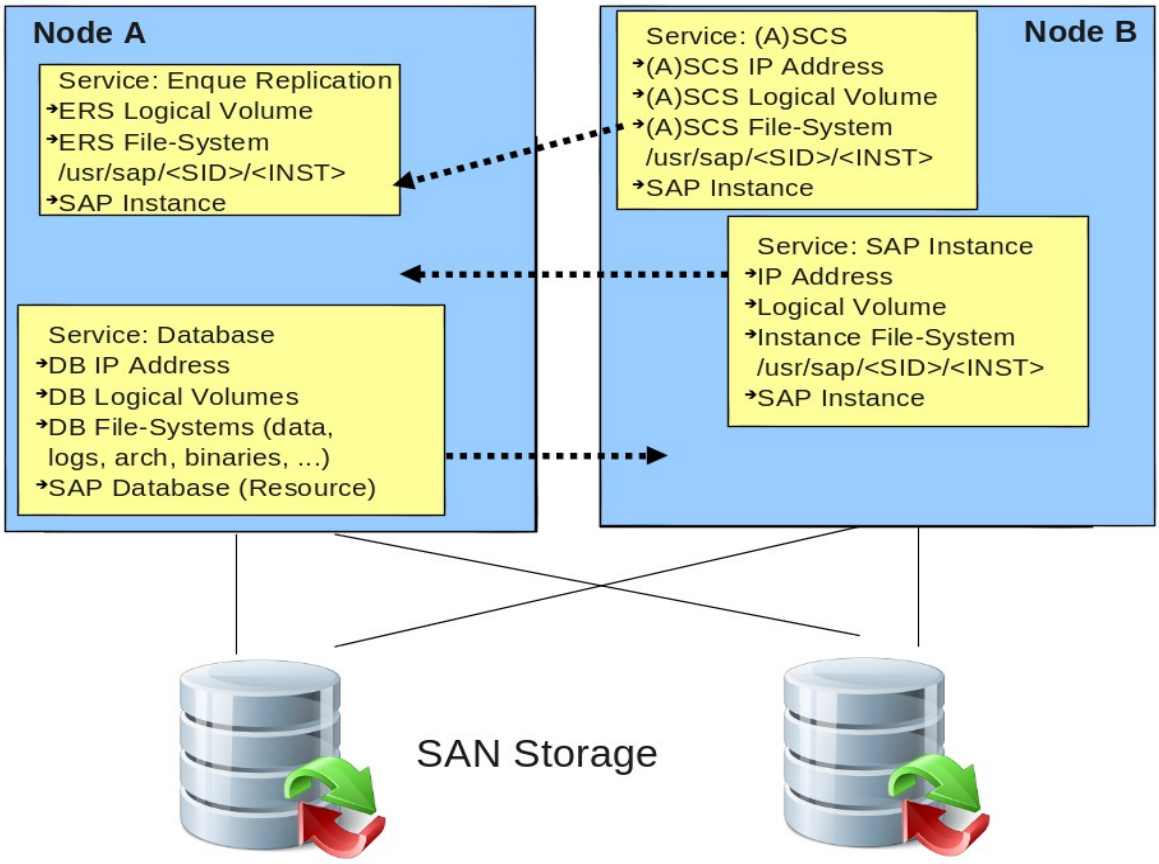

Figure 2.3.2.1: SAP Cluster

2.3.2 Enqueue Replication Server

Since the lock table of the enqueue server is the most critical piece in a SAP environment beside the database SAP has developed the “Enqueue Replication Server” (ERS) which maintains a backup copy of the lock table which can be used by the enqueue server in case it has to be restarted because of a failure. While the (A)SCS is running on one node, the ERS always needs to provide a copy of the current enqueue table on the second node. When the system needs to migrate the (A)SCS to the second node, it first starts on the second node, and then shuts down the ERS, taking over its' shared memory segment and thereby acquiring an up-to-date enqueue table. At this time the replicated enqueue table becomes the new primary enqueue table. As a last step the ERS will start on the now second node, which now provides the replicated enqueue table.

For this mechanism to work, both (A)SCS and ERS must be controlled by the cluster software. In normal mode, cluster rules ensure that ERS and (A)SCS will always run on different nodes. In the event of migration the (A)SCS needs to “follow” the ERS. (A)SCS

needs to switch to the node were ERS is running. Cluster rules have to ensure that (A)SCS starts up while the ERS is already (or still) running, because it needs to take over the replicated enqueue table.

3 Requirements

3.1 Server Hardware

Almost any contemporary enterprise grade server hardware is up to the requirements clustering demands. Both single or multi-core x86_64 processors are supported. Typically, SAP servers are equipped with a fair amount of memory starting at 8 gigabytes and are usually limited only by the hardware specification. The cluster nodes need to be attached to a fencing mechanism. Please refer to chapter 4.4.1 for further information.

In addition to physical hardware the cluster nodes can also be set up on VMs on the Virtualization platforms currently supported by SAP for running SAP on Linux (see SAP Note 1122387 - Linux: SAP Support in virtualized environments).

3.2 Network

The communication between the cluster nodes is the most crucial and sensitive part in a HA cluster environment. Therefore, the setup of the network must be done with foremost priority and care.

Red Hat recommends to separate network communication into two segments (minimum):

- a public network for Client-to-Cluster communication, laid out redundantly by using two NICs in each server and at best routing each NIC to separate switches.
- a private network, dedicated exclusively for inter-node communication, also redundant as described above.

For RHEL 6.5, Red Hat also recommends the use of IP multicast for cluster infrastructure traffic. Some network switches may require special configuration settings to enable multicast operations. Please refer to the hardware vendor's configuration guide for correct multicast configurations. Broadcast and UDP unicast are fully supported as alternatives to multicast in situations where multicast can not be implemented.

Public/Private Networks

At least two network interfaces are recommended for clustering. The reason for this is to separate cluster traffic from all other network traffic. Availability and cluster file system performance is dependent on the reliability and performance of the cluster communication network (private network).

Therefore, all public network load must be routed through a different network (public network).

Bonding

In high availability configurations, at least the private or cluster interconnect network setup, preferably both, must be fully redundant. Network Interface Card (NIC) bonding is the only method to provide NIC failover for the cluster communication network.

3.3 File Systems

The following different types of file systems are used in a SAP environment:

Shared Storage File systems

The following file systems should be created on a shared SAN LUN:

- /db2
- */usr/sap/<SID>/DVEBMGS0*
- */usr/sap/<SID>/<InstanceNo>*

for production environments the directories below /db2 should reside each on a separate LUN for performance reasons. Please refer to the SAP Installation Guide for details.

Since these file systems should always only be accessed from one cluster node it is recommended to create these file systems as HA-LVM volumes. Please refer to What is a Highly Available LVM (HA-LVM) configuration and how do I implement it? For instructions on how to setup a HA-LVM.

Local File Systems

The following directories can be created locally on each node.

- */usr/sap*
- */sapmnt*
- */db2*
- */usr/sap/<SID>/SYS (linking directory, that* can also reside locally. Provides links to */sapmnt/<SID>*)
- */usr/sap/<SID>/SYS* (directory to all cluster nodes)
- */usr/sap/tmp*

Specific directories for SAP agents such as */usr/sap/ccms*, */usr/sap/<SID>/ccms* or */usr/sap/SMD* must be configured according to your SAP landscape.

Follow the database file system configuration recommendations from the SAP installation guide. It is recommended to have physically different mount points for the program files and for *origlog*, *mirrlog*, log archives and each sapdata.

NOTE: The configuration process gets more complex when multiple database instances of the same type run within the cluster. The program files must be accessible for every instance. The mounts from shared storage must be added to the cluster configuration as file system resources to the failover service.

Shared File Systems

The following file systems need to be available on all servers where instances of a SAP system are running (including application servers that are not part of the cluster):

- */sapmnt/<SID>*

- */usr/sap/trans*

Since simultaneous read and write accesses can occur to those file systems from different servers those file systems should either reside on a high available NFS server or NFS exporting storage array. Alternatively a GFS2 formatted SAN LUN that is mounted on all servers can be used for each file system.

If NFS is used to provide access to these file systems the NFS server can not be part of the cluster (see https://access.redhat.com/site/solutions/22231 for more information).

4 Red Hat Enterprise Linux HA add-on overview

The RHEL HA add-on provides all components needed to set up high availability environments:

4.1 CMAN

Cluster Manager (CMAN) is a Red Hat specific service module that manages the cluster communication infrastructure. It provides a user API that is used by Red Hat layered cluster components. CMAN also provides additional functionality such as APIs for a distributed lock manger, clustered lvm, conditional shutdown, and barriers.

4.2 Cluster Resource Manager

The Cluster Resource Manager (**Pacemaker**) manages the resources and provides failover capabilities for cluster. It also controls the handling of user requests like service start, restart, disable, and relocate.

The service manager daemon also handles restarting and relocating services in the event of failures. **`Pacemaker`** uses Open Cluster Framework (OCF) compliant resource agents to control and monitor required resources. *SAPInstance* and *SAPDatabase* are OCF compliant resource agents provided by Red Hat.

4.3 Resource Agents

Resource agents provide the interface between the cluster resource manager and the services that should be managed in a cluster. In addition to resource agents for managing basic functionality like configuring IP-addresses, mounting file systems etc. the RHEL HA add-on also provides the resource agents for managing SAP instances and associated databases:

4.3.1 SAPInstance

The SAPInstance resource agent can manage most SAP instance types, like (A)SCS instances, dialog instances, etc.

All operations of the SAPInstance resource agent are performed by using the SAP startup framework, that was introduced with SAP kernel release 6.40. Reference additional information regarding the SAP Management Console in SAP Note 1014480.

Using this framework defines a clear interface for the cluster heartbeat and how it views the SAP system. The monitoring options for the SAP system are far superior than other methods such as monitoring processes with the *ps* command or pinging the application.

Since the “sapstartsrv” process used by the SAPInstance resource agent is part of the SAP kernel it is important to always keep the SAP kernel of all SAP instances up-to-date.

Supported SAP NetWeaver releases

The following releases of SAP NetWeaver are currently supported:

- SAP NetWeaver ABAP Release 6.40 – 7.x0
- SAP NetWeaver Java Release 6.40 – 7.x0
- SAP NetWeaver ABAP + Java Add-In Release 6.40 - 7.x0 (Java is not monitored by the cluster in that case)

When using SAP NetWeaver 6.40 please make sure to follow SAP Note 99516 – Backward porting of sapstartsrv for earlier releases

4.3.2 SAPDatabase

The purpose of the SAPDatabase resource agent is to start, stop, and monitor the database instance of an SAP system. Together with the RDBMS system, it also controls the related network service for the database (such as the Oracle Listener or the MaxDB xserver). The resource agent expects a standard SAP installation and therefore requires less parameters.

The monitor operation of the resource agent can test the availability of the database by using SAP tools (**`R3trans`** or **`jdbcconnect`**). With that, it ensures that the database is truly accessible by the SAP system.

After an unclean exit or crash of a database, require a recover procedure to restart can be required. The resource agent has a procedure implemented for each database type. If preferred, the attribute AUTOMATIC_RECOVER provides this functionality.

Supported Databases

Although the example later on in this document describes the setup of a high available SAP NetWeaver environment with an IBM DB2 10.1FP3 database, the SAPDatabase resource agent supports other databases as well:

- IBM DB2 LUW 9.7, 10.5
- Oracle 11.2.0.3
- SAP ASE 15.7
- SAP MaxDB 7.x

4.4 Fencing

When one system goes offline and another one takes over, there might be a moment when two (or more) systems simultaneously try to write onto the same shared datastore - destroying everything. This can be prevented with a number of measures that are grouped under the name of Fencing. All of them may work, but most well known is Power Fencing.

4.4.1 Power Fencing Systems

The power fencing subsystem allows operational cluster nodes to control the power of failed nodes to ensure that they do not access storage in an uncoordinated manner. Most power control systems are network based. They are available from system vendors as add-in cards or integrated into the motherboard. External power fencing devices are also available. These are typically rack or cabinet mounted power switches that can cut the power supply on any given port.

Note that this fencing method requires a working “admin” network connecting to the fence device to successfully trigger the fence action. Fencing devices are recommended to be on the same network that is used for cluster communication.

If the power fencing method uses a remote console (IBM RSA: Remote Supervisor Adapter, Fujitsu iRMC: Integrated Remote Management Controller, HP iLO: integrated lights-out, etc.) extensive testing of the fencing mechanism is recommended. These fencing mechanisms have a short time gap between issuing the “reset” command on the remote console and the actual reset taking place. In a situation when both nodes of a 2-node cluster are trying to fence each other, sometimes the time gap is long enough for both nodes to successfully send the “reset” command before the resets are executed. This results in a power cycle of both nodes.

4.4.2 SAN Switch Based Fencing

While it is preferable to employ a power fencing solution for the robustness a system reboot provides, SAN switch fencing is also possible. Fencing protects shared data storage from two (or more) systems that write simultaneously. SAN switch fencing works by preventing access to storage LUNs on the SAN switch.

4.4.3 SCSI Fencing

SCSI-3 persistent reservations can also be used for I/O fencing. All nodes in the cluster must register with the SCSI device to be able to access the storage. If a node has to be fenced, the registration is revoked by the other cluster members.

Please reference the fence_scsi(8) manpage for further details. The SCSI fencing mechanism requires SCSI-3 write-exclusive, registrants-only persistent reservation as well as support of the preempt-and-abort command on all devices managed or accessed by the cluster. Please contact Red Hat technical support to determine if your software and hardware configuration supports persistent SCSI reservations.

The How to Control Access to Shared Storage Devices Using SCSI Persistent Reservations with Red Hat Enterprise Linux Clustering and High Availability technical brief discusses this more.

4.5 Storage Protection

Fencing already protects storage from being destroyed. But there are more techniques that were developed to keep data consistent in combination with Clustering.

4.5.1 HA-LVM, CLVM

Logical volume configurations can be protected by the use of HA-LVM or CLVM. CLVM is an extension to standard Logical Volume Management (LVM) that distributes LVM metadata updates to the cluster. The *CLVM DAEMON* (**clvmd**) must be running on all nodes in the cluster and produces an error if any node in the cluster does not have this daemon running. HA-LVM imposes the restriction that a logical volume can only be activated exclusively; that is, active on only one machine at a time. Red Hat Enterprise Linux 5.6 introduced the option to use HA-LVM with CLVMD, which implements exclusive activation of logical volumes. Previous releases did implement HA-LVM without CLVMD using LVM tags filtering.

4.5.2 GFS2

Red Hat's Global File System (GFS2) is a POSIX compliant, shared disk cluster file system. GFS lets servers share files with a common file system on a SAN.

With local file system configurations such as ext3, only one server can have access to a disk or logical volume at any given time. In a cluster configuration, this approach has two major drawbacks. First, active/active file system configurations cannot be realized, limiting scale out ability. Second, during a failover operation, a local file system must be be unmounted from the server that originally owned the service and must be remounted on the new server.

GFS creates a common file system across multiple SAN disks or volumes and makes this file system available to multiple servers in a cluster. Scale out file system configurations can be easily achieved. During a failover operation, it is not necessary to unmount the GFS file system because data integrity is protected by coordinating access to files so that reads and writes are consistent across servers. Therefore, availability is improved by making the file system accessible to all servers in the cluster. GFS can be used to increase performance, reduce management complexity, and reduce costs with consolidated storage resources. GFS runs on each node in a cluster. As with all file systems, it is basically a kernel module that runs on top of the Virtual File System (VFS) layer of the kernel. It controls how and where the data is stored on a block device or logical volume. By utilizing the Linux distributed lock manager to synchronize changes to the file system, GFS is able to provide a cache-coherent, consistent view of a single file system across multiple hosts.

4.5.3 Storage Mirroring

In disaster tolerant configurations, storage mirroring techniques are used to protect data and ensure availability in the event of a storage array loss. Storage mirroring is normally performed in two different ways.

Enterprise storage arrays typically offer a mechanism to mirror all data from one storage array to one or more other arrays. In the case of a disaster, remote data copies can be used. When using array-based-mirroring (also known as SAN-based-mirroring) the cluster nodes need access to all mirrors (usually through multipath configurations). However, only one array is used by the cluster at one time (active array); the other array is used for replication and site failover purposes (passive array). If the active storage array fails, cluster services halt and the cluster must be manually stopped and reconfigured to use passive array.

Red Hat Enterprise Linux offers the possibility to create host-based-mirroring configurations with the logical volume manager (LVM). Servers in the cluster are able to assemble

independent storage devices (LUNs) on separate storage arrays to a soft-raid logical volume in order to prevent data loss and ensure availability in case of a failure on one of the physical arrays. LVM simultaneously reads and writes to two or more LUNs on separate storage arrays and keeps them synchronized.

4.6 Stretch Cluster

RHEL HA Add-On can be used to provide disaster recovery capabilities in order to minimize service downtime in physical site failure scenarios. Stretch clusters span exactly two sites and have LAN-like latency between sites via site-to-site interlink. Red Hat supports different stretch cluster architectures depending on the the storage infrastructure requirements and data replication technologies as shown in Knowledge Base article "Support for Red Hat Enterprise Linux Cluster and High Availability Stretch Architectures". This section will focus on the "Fully Interconnected SAN with LVM Mirroring" use case, described in following diagram:

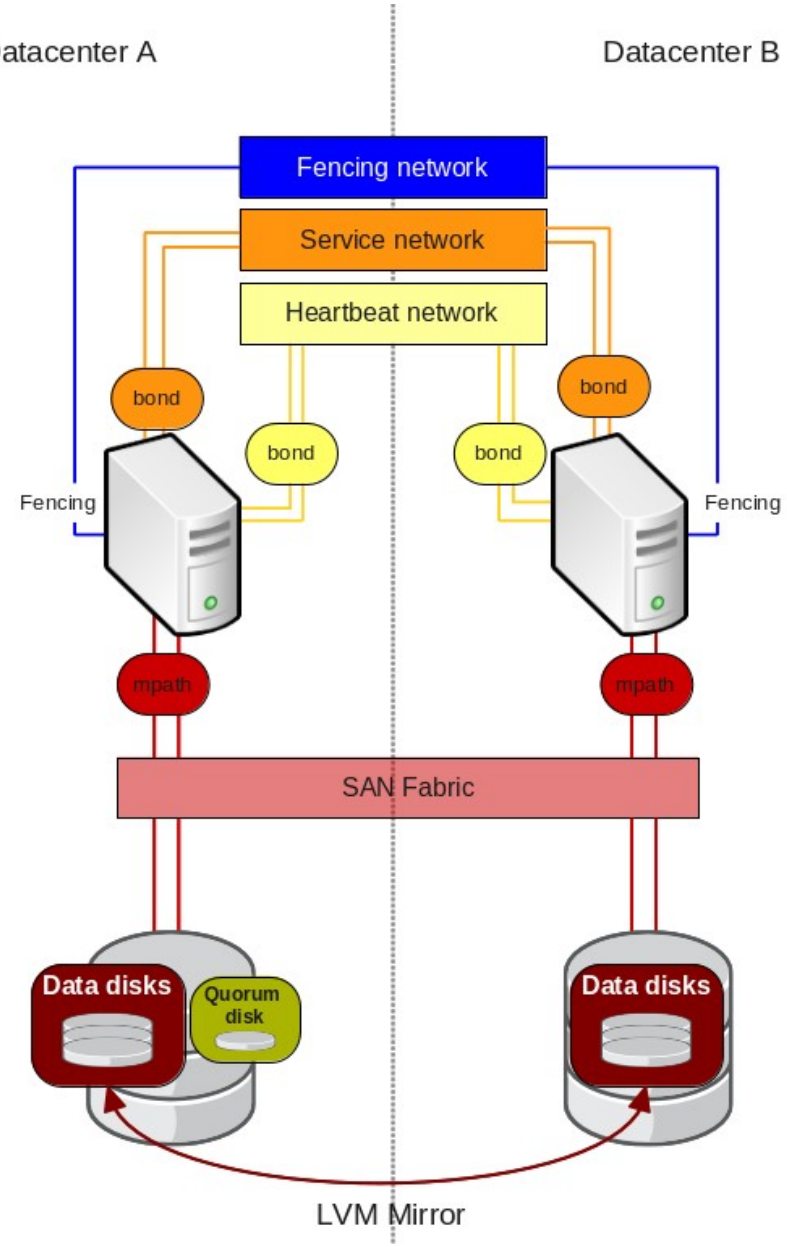

Figure 4.6.1: Fully Interconnected SAN with LVM Mirroring

4.6.1 Network infrastructure requirements

In this configuration cluster nodes are distributed between two sites. Both sites must be connected by a network interconnect that provides a LAN-like latency (<= 2ms round trip ping time), and share logical networks. Multicast or broadcast must work between the nodes in the different sites.

4.6.2 Storage requirements

A Stretch Cluster setup requires at least two storage arrays, one at each physical site, with full SAN connectivity between all arrays and all nodes at each site.

4.6.3 Data replication with LVM

LVM mirror is used to synchronously replicate storage between the two arrays. Several points must be considered when configuring LVM mirror in stretch clusters.

- LVM tags based HA-LVM is required in this case to ensure data consistency.
- When creating LVM mirrors, the mirror legs must be created in storage arrays at different physical sites.
- To avoid mirror resynchronization on reboot, disk based mirrorlog must be used. Since RHEL 5.6, mirrorlog can be mirrored itself.

4.6.4 Stretch cluster limitations

The use of stretch clusters impose some restrictions when configuring Red Hat Clustering.

- A maximum of two sites are supported.
- CLVMD is not supported in stretch clusters. HA-LVM with tags must be used to ensure data consistency in shared volumes.
- Cluster aware mirroring (cmirror), GFS, and GFS2 are not supported in stretch cluster environments.

4.6.5 Stretch clusters architecture review

Stretch clusters must be carefully designed and implemented to ensure proper behavior in all failure scenarios. Stretch clusters require obtaining a formal review from Red Hat Support as described in knowledge base article “Architecture Review Process for Red Hat Enterprise Linux High Availability, Clustering, and GFS/GFS2” to ensure that the deployed cluster meets established guidelines.

5 Example Configuration

5.1 Overview

In this part of the document we will go through the whole installation and configuration process of an example setup. It will give background were necessary, describe the necessary steps and if possible give you examples on which commands to use. The examples will hardly every be used the exact way they are written here, it is the responsibility of the consultant to understand and modify the commands in a way to apply them to the current context. So you will hardly every be able to copy/paste the commands and use without modifications.

The whole process consists of the following steps:

1. OS installation and configuration on all nodes
2. SAP installation and configuration on 1st node
3. Verification that SAP Installation also works on 2nd node
4. Cluster configuration
5. Test of cluster functionality

Each step will be explained in detail in the following chapters.

5.2 Configuration of the Test Environment

5.2.1 SAP system configuration

- SID: RH2
- SAP Instances and Instance numbers:
  - ASCS00
  - ERS10 (see High Availability - Frequently Asked Questions&... | SCN for more information about why an Enqueue Replication Server is needed in a SAP HA environment)
  - DVEBMGS01 (PAS)
- Used SAP NetWeaver release: 7.30

5.2.2 Used Database

IBM DB2 LUW 10.1 FP3

(the cluster setup described in this whitepaper was also tested with SyBase ASE 15.7, Oracle 11.2.02 and MaxDB 7.x)

5.2.3 Network setup

IP addresses and hostnames of Cluster nodes on production network

- 10.17.140.1 node1.example.com node1

- 10.17.140.2 node2.example.com node2

IP addresses and hostnames for separate heartbeat network

- 192.168.20.1 node1hb
- 192.168.20.2 node2hb

Virtual IP addresses and hostnames for SAP Instances (on production network)

- 10.17.140.54 rh2ascs.example.com rh2ascs (ASCS)
- 10.17.140.55 rh2db.example.com rh2db (database)
- 10.17.140.56 rh2ers.example.com rh2ers (ERS)
- 10.17.140.57 rh2pas.example.com rh2pas (PAS)

5.2.4 File System Setup

Shared SAN volumes for DB filesystem and SAP PAS instance filesystem (as separate VGs/LVs with EXT4 filesystems)

- /dev/vg_db/lv_db mounted as /db2 (80GB)
- /dev/vg_sap/lv_sap mounted as /usr/sap/RH2/DVEBMGS01 (10GB)

NFS shares

The following SAP global file systems are mounted via NFS on all cluster nodes:

- nfsserver:/shares/lnxha-sapmnt mounted as /samnt
- nfsserver:/shares/lnxha-usr-sap-trans mounted as /usr/sap/trans

Local File Systems

Instance filesystems for ASCS (/usr/sap/RH2/ASCS00) and ERS (/usr/sap/RH2/ERS10) are local on both cluster nodes

5.3 Operating System Installation

Reference the Red Hat Enterprise Linux 6 Release Notes and Installation Guides for the specific details regarding the acquisition and installation of Red Hat Enterprise Linux.

Once the hardware configuration has been performed to accommodate a cluster, install Red Hat Enterprise Linux 6.5 on the servers using the preferred method.

The installation assistant guides the user through the OS installation. The Red Hat Enterprise Linux Installation Guide provides details regarding each of the screens presented during the installation process.

5.4 Operating System Customizations

5.4.1 SAP specific OS customization

Please make sure that all OS customizations required for SAP have been applied:

SAP note 1496410 - “Red Hat Enterprise Linux 6.x: Installation and upgrade”

5.4.2 NTP

The synchronization of system clocks in a cluster becomes infinitely more important when storage is shared among members. System times should be synchronized against a network time server via the Network Time Protocol (NTP) by using the ntpd service.

5.4.3 ACPI

Please reference the *Configuring ACPI For Use with Integrated Fence Devices* section in “Configuring and Managing a Red Hat Cluster”. As described there, disabling ACPI Soft-Off allows an integrated fence device to shut down a server immediately rather than attempting a clean shutdown.

Soft-Off allows some components to remain powered so the system can be roused from input from the keyboard, clock, modem, LAN, or USB device and subsequently takes longer to fully shutdown. If a cluster member is configured to be fenced by an integrated fence device, disable ACPI Soft-Off for that node. Otherwise, if ACPI Soft-Off is enabled, an integrated fence device can take several seconds or more to turn off a node since the operating system attempts a clean shutdown. In addition, if ACPI Soft-Off is enabled and a node panics or freezes during shutdown, an integrated fence device may not be able to power off the node. Under those circumstances, fencing is delayed or unsuccessful.

Use the following commands to switch off ACPI Soft-Off:

```
# service acpid stop
# chkconfig acpid off
```

5.4.4 OS Dependencies

All changes to the operating system environment have to be rolled out to all nodes participating in the cluster. This includes changes to

- configuration files:
  - */etc/lvm/lvm.conf*
  - /etc/cluster/mdadm.conf
  - */etc/services*
  - */etc/hosts*
  - */etc/multipath.conf*
  - */etc/multipath.binding*

- os packages and patches
- os users and groups
    - home-directories
    - user settings and logon scripts

Virtual IP Addresses and Hostnames

SAP NetWeaver is typically installed via the graphical installation tool **sapinst**. Before beginning the installation, determine which IP addresses and hostnames are needed for use during the SAP installation:

- each node requires a static IP address and an associated host name. This address is also referred to as the physical IP address.
- each database and SAP instance (ASCS/SCS, ERS, PAS) requires a virtual IP address and virtual hostname.

The virtual addresses must not be configured at the operating system level because they are under the control of the Clustering. Those addresses are referred to as the virtual IP addresses. The virtual IP address and virtual hostname guarantee that a database or SAP instance is always accessible under the same name no matter which physical cluster node currently hosts the service.

Local dialog instances, which are not part of the cluster, use a virtual host name as an alias to the physical host name so those SAP instances are not failed over by RHEL HA Add-On. The virtual host name is used to start the instances manually via the **`sapstart`** command and to distinguish their profile names from physical host names.

Edit the */etc/hosts* file on all nodes and add the virtual host names and their associated IP addresses or add them to your DNS server. Additionally, add any other cluster relevant host name and address (e.g., the physical host names or addresses of the nodes) to /etc/hosts so the DNS server is no longer a possible single point of failure.

5.5 SAP Installation

5.5.1 SAP Installation

Preparations

Before installing SAP NetWeaver, mount all the necessary file systems (either through the cluster or manually) in the correct order.

Enable the virtual IP addresses (either through the cluster or manually) for all SAP instances. This is necessary because the SAP installation program will try to start each newly installed instance during its post-processing.

Run the SAP installer for each instance

Change to a directory with enough free space (e. g. /tmp) and run /usr/sap/swpm/sapinst to install DB and SAP instances. Use 'Distributed Installation' or "High availability' option in *sapinst* to install each parts of the SAP system.

Please follow the SAP Installation Guide for your SAP release for detailed SAP installation instructions.

Important: make sure to call sapinst with the option SAPINST_USE_HOSTNAME so that all instances get installed with the correct virtual hostname:

ASCS: /usr/sap/swpm/sapinst SAPINST_USE_HOSTNAME=rh2ascs

DB: /usr/sap/swpm/sapinst SAPINST_USE_HOSTNAME=rh2db

PAS: /usr/sap/swpm/sapinst SAPINST_USE_HOSTNAME=rh2pas

ERS: /usr/sap/swpm/sapinst SAPINST_USE_HOSTNAME=rh2ers

5.5.2 Installation Post-Processing

Modify (A)SCS profile

Since the start/stop of the SAP Enqueue server process is managed by the cluster, it is necessary to disable the automatic restart of the Enqueue server process in the SAP profile of the (A)SCS instance:

```
#-----------------------------------------------------------------------
# Start SAP enqueue server
#-----------------------------------------------------------------------
_EN = en.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_04 = local rm -f $(_EN)
Execute_05 = local ln -s -f $(DIR_EXECUTABLE)/enserver$(FT_EXE) $(_EN)
#Restart_Program_01 = local $(_EN) pf=$(_PF)  # <---- original line
Start_Program_01 = local $(_EN) pf=$(_PF)       # <---- modified line ...
```

Users, Groups and Home Directories

Create users and groups on the second node as they were created by the SAP installation on the first node. Use the same user and group IDs.

Depending on the Installation Master CD that was used for the SAP installation, the login profiles for the SAP administrator user (<sid>adm) and the database administrator user could differ. In older and non-HA installations, the user login profiles look similar to *.sapenv_hostname.csh.*

Using the host name in the user login profiles is a problem in an HA environment. By default, the profiles *.login*, *.profile* and *.cshrc* search for two types of user login profiles: first for the one including the local host name (e.g., *.dbenv_hostname.csh*) and then for a version without the host name included. The latest versions of the InstMaster CDs install both versions of the user login profiles. This could lead to some confusion for the administrator with regard to which version is used in which case. The removal of all user login profiles (that include a host name in the file name) is recommended. Do this for both the SAP administrator, <sid>adm, as well as the database administrator user.

Synchronizing Files and Directories

Copy */etc/services* or its values that were added by **`sapinst`** (see SAP related entries at the end of the file) to all other nodes.

Copy any database specific files and directories (to the other nodes.

The most important SAP profile parameter for a clustered SAP system is **`SAPLOCALHOST`**. After the installation with **`sapinst`**, ensure that all SAP instance profiles contain this parameter. The value of the parameter must be the virtual host name specified during the installation.

In the START profiles, the parameter **`SAPSYSTEM`** must be set (default since 7.00).

As a general recommendation, the SAP parameter **`es/implementation`** should be set to "std" in the SAP *DEFAULT.PFL* file. See SAP Note 941735. The SAPInstance resource agent cannot use the AUTOMATIC_RECOVERY function for systems that have this parameter set to "map".

Verify that SAP system is able to start on other cluster nodes

If the DB and SAP instances start without problems on all cluster nodes, please verify that the system is accessible by trying to log in remotely via the SAPGUI.

Install correct SAP license keys

For improved SAP hardware key determination in high-availability scenarios it might be necessary to install several SAP license keys based on the hardware keys of each cluster node. Please see SAP Note 1178686 - Linux: Alternative method to generate a SAP hardware key for more information.

Update SAP HostAgent on all nodes

Since the SAP resource agents for Pacemaker depend on SAP HostAgent it is important to always have the latest version of SAP Hostagent installed on all cluster nodes. At least version 138 should be used, for SAP ASE the minimal required version of SAP HostAgent is 146.

See SAP Note 1031096 - Installing Package SAPHOSTAGENT for more information about installing or upgrading SAP HostAgent.

To see which version of SAP Hostagent is currently installed, please run the following command:

```
$ /usr/sap/hostctrl/exe/saphostexec -version
```

Before starting to configure the Cluster

In some cases sapinst doesn't start the freshly installed instance and leaves an empty work directory (*/usr/sap/<SID>/<Instance><Number>/work*) which results in a monitoring error of the SAPInstance resource agent.

In that case the instance must be started manually in order for the correct entries to be written to the work directory. After a manual shutdown of the instances, the cluster agent can be used. Remember that the virtual IP addresses for the SAP instances you wish to start must be active. They can be started manually (e.g., with the Linux command **`ip`**) and then stopped again after shutting down the SAP instances.

```
When using IBM DB2 inside the cluster some additional configuration is
necessary to allow the database to start on all cluster nodes. Please see
Appendix A: Fix db2nodes.cfg file to enable startup of IBM DB2 database on
all cluster nodes for more information
```

5.6 Configuring the cluster

5.6.1 Install cluster software and start the cluster

1. Make sure that both cluster nodes are named correctly and that the node names resolve to the correct network interfaces.
2. Subscribe all cluster nodes to the “RHEL Server High Availability (v. 6 for 64-bit x86_64)” and “RHEL for SAP (v. 6 for 64-bit x86_64)” channels on RHN or your local RHN Satellite Server
3. Install the required RPMs with the following **yum** command:

```
$ yum install cman
$ yum install pacemaker
$ yum install resource-agents-sap
$ yum install pcs
```

4. install a skeleton cluster configuration in /etc/cluster/cluster.conf

   Run the following command **on each cluster node** to configure the cluster infrastructure (CMAN):

```
$ pcs cluster setup --name sap_pacemaker node1hb node2hb
```

   Alternatively you can also manually configure the /etc/cluster/cluster.conf file **on each cluster node** according to the following example:

```
<?xml version="1.0"?>
 <cluster config_version="1" name="sap_pacemaker">
 <logging debug="off"/>
 <cman two_node="1" expected_votes="1"/>
 <clusternodes>
 <clusternode name="node1hb" nodeid="1">
 <fence>
 <method name="pcmk-redirect">
 <device name="pcmk" port="node1hb"/>
 </method>
 </fence>
 </clusternode>
 <clusternode name="node2hb" nodeid="2">
 <fence>
 <method name="pcmk-redirect">
 <device name="pcmk" port="node2hb"/>
 </method>
 </fence>
 </clusternode>
 </clusternodes>
 <fencedevices>
 <fencedevice name="pcmk" agent="fence_pcmk"/>
 </fencedevices> </cluster>
```

Now start the cluster

```
$ service pacemaker start
```

or

```
$ pcs cluster start
```

Required minimal cluster package versions

Please verify that at least the following versions of the cluster packages are installed:

- cman: 3.0.12.1-58.el6.x86_64
- libqb-0.16.0-2.el6.x86_64
- pacemaker: 1.1.10-14.1.el6.x86_64
- pcs-0.9.90-2.el6.noarch
- resource-agents-3.9.2-40.el6_5.4.x86_64
- resource-agents-sap-3.9.2-41.el6_5.4.x86_64
- fence-agents-3.1.5-34.el6.x86_64

(please check Support for the Pacemaker resource manager in RHEL 6 High Availability clusters for further details on required package versions)

5.6.2 Cluster Resources and Services

There are many types of cluster resources that can be configured. Resources are bundled together to highly available services while a service consists of one or more cluster resources. The database service for example consists of these resources:

- virtual IP address (IP resource)
- volume groups (LVM resource)
- filesystems for DB executables, datafiles, logs, etc. (FS resource)
- database application start/stop/monitor (SAPDatabase resource)

Resources can be assigned to any cluster service (resource groups). Once associated with a cluster service, it can be relocated by the cluster transition engine if it deems it necessary, or manually through a GUI interface, a web interface (conga) or via the command line. If any cluster member running a service becomes unable to do so (e.g. due to hardware or software failure, network/connectivity loss, etc.), the service with all its resources are automatically migrated to an eligible member (according to failover domain rules).

Reference the Adding a Cluster Service to the Cluster section of “Configuring and Managing a Red Hat Cluster” for more information.

Resource Agents used for this setup

There are many types of configurable cluster resources. Reference the Adding a Cluster Service to the Cluster section of Configuring and Managing a Red Hat Cluster for more information.

The following resource types are used in this example to provide the high availability functionality for the database and SAP instances:

IPaddr2	manages virtual IP addresses
LVM	manages LVM activations
Filesystem	mounts file-systems
SAPDatabase	starts / stops / monitors the Database
SAPInstance	starts / stops / monitors a SAP instance (ABAP or java)

5.6.3 Initial cluster configuration

For initial setup there may be no power fencing device configured. Also, as we are running pacemaker in a two-node-cluster, we need to switch off the quorum management:

```
$ pcs property set stonith-enabled="false"
$ pcs property set no-quorum-policy="ignore"
$ pcs resource rsc defaults default-resource-stickiness=100
```

5.6.4 Setup the A(SCS) and ERS resource-dependency

As described in the design part of this document, A(SCS) and ERS need to run on either node, never on the same. For that we create a Master-/Slave-Resource with according constraints:

```
$ pcs resource create prm_RH2_ascs_IP IPaddr2 ip="10.17.140.54"
$ pcs resource create prm_RH2_ers_IP IPaddr2 ip="10.17.140.56"
$ pcs resource create prm_RH2_SCS SAPInstance \
    InstanceName="RH2_ASCS00_rh2ascs" DIR_PROFILE="/sapmnt/RH2/profile" \
    START_PROFILE="/sapmnt/RH2/profile/RH2_ASCS00_rh2ascs" \
    ERS_InstanceName="RH2_ERS10_rh2ers" \
    ERS_START_PROFILE="/sapmnt/RH2/profile/RH2_ERS10_rh2ers"
$ pcs resource op add prm_RH2_SCS monitor interval="31" role="Slave" \
    timeout="60"
$ pcs resource op add prm_RH2_SCS \
    monitor interval="30" role="Master" timeout="60"
$ pcs resource master ms_RH2_SCS prm_RH2_SCS \
    meta master-max="1" clone-max="2" notify="true" target-role="Started"
$ pcs constraint colocation add prm_RH2_ascs_IP with master ms_RH2_SCS 2000
$ pcs constraint colocation add prm_RH2_ers_IP with slave ms_RH2_SCS 2000
```

For SAP Dual-Stack systems that have both a ABAP and a Java SCS instances it is recommended to set up separate Master/Slave resources using separate IP addresses and enqueue replication servers.

To see the full list of options supported by the SAPDatabase resource agent you can run the following command:

```
$ pcs resource describe SAPInstance
```

5.6.5 Database resource group

here an example how to setup a cluster database group:

```
$ pcs resource create prm_RH2_db_IP IPaddr2 ip="10.17.140.55"
$ pcs resource create prm_RH2_db_LVM LVM volgrpname="vg_db" exclusive="true"
$ pcs resource create prm_RH2_db_FS Filesystem device="/dev/mapper/vg_db-
lv_db" directory="/db2" fstype="ext4"
$ pcs resource create prm_RH2_db_DB SAPDatabase \
    AUTOMATIC_RECOVER="TRUE" DBTYPE="DB6" SID="RH2" STRICT_MONITORING="TRUE"
$ pcs resource op add prm_RH2_db_DB start timeout="1800"
$ pcs resource op add prm_RH2_db_DB stop timeout="1800"
$ pcs resource group add grp_RH2_DB prm_RH2_db_IP prm_RH2_db_LVM
prm_RH2_db_FS prm_RH2_db_DB
```

To see the full list of options supported by the SAPDatabase resource agent you can run the following command:

```
$ pcs resource describe SAPDatabase
```

5.6.6 Primary Application Server group (PAS)

```
$ pcs resource create prm_RH2_pas_IP IPaddr2 ip="10.17.140.57"
$ pcs resource create prm_RH2_pas_LVM LVM volgrpname="vg_sap"
exclusive="true"
$ pcs resource create prm_RH2_pas_FS Filesystem \
    device="/dev/mapper/vg_sap-lv_sap" directory="/usr/sap/RH2/DVEBMGS01"
fstype="ext4"
$ pcs resource create prm_RH2_pas_SAP SAPInstance \
    InstanceName="RH2_DVEBMGS01_rh2pas" DIR_PROFILE="/sapmnt/RH2/profile" \
    START_PROFILE="/sapmnt/RH2/profile/RH2_DVEBMGS01_rh2pas"
$ pcs resource op add prm_RH2_pas_SAP start timeout="180"
$ pcs resource op add prm_RH2_pas_SAP stop timeout="240"
$ pcs resource group add grp_RH2_PAS prm_RH2_pas_IP prm_RH2_pas_LVM \
    prm_RH2_pas_FS prm_RH2_pas_SAP
$ pcs constraint order grp_RH2_DB then grp_RH2_PAS symmetrical=false
```

See man ocf_heartbeat_SAPInstance for the full list of options supported by the SAPInstance resource agent.

to make sure that the PAS will only be started AFTER after promoting the node to A(SCS), you can add the following constraint:

```
$ pcs constraint order promote ms_RH2_SCS then grp_RH2_PAS
```

5.6.7 Fencing

Example for a fencing setup

In order for a production cluster to be supported by Red Hat it is required that a working Fencing setup for all cluster nodes is implemented. Especially in a two node cluster fencing is an important part of the setup to make sure that you never create a “split brain” situation. The fencing configuration consists of two parts. The first is the configuration of the fencing daemon (fenced) itself. The second is the configuration of the fencing agents that the daemon

uses to fence each cluster node.

As an example we show here the setup of a VMware host fencing device. This is most probably not what you will have in your setup, but it gives you an idea how a fencing device would be set up. The parameters given to the stonith:... resource will be highly dependent on the sort of fencing device you will be using. Please refer to the pacemaker documentation for you specific fencing device for details.

```
$ pcs resource create st-vmware-node1hb stonith:fence_vmware_soap \
    action="reboot" ipaddr="vsphere.example.com" login="<user>" \
    passwd="<password>" ssl="1" port="rhel6_cluster_1" \
    pcmk_host_check="static-list" pcmk_host_list="node1hb" \
    pcmk_host_map="node1hb:rhel6_cluster_1" \
    op monitor interval="60s"
$ pcs resource create st-vmware-node2hb stonith:fence_vmware_soap \.
    action="reboot" ipaddr="vsphere.example.com" login="<user>" \
    passwd="<password>" ssl="1" port="rhel6_cluster_2" \
    pcmk_host_check="static-list" pcmk_host_list="node2hb" \
    pcmk_host_map="node2hb:rhel6_cluster_2" \
    op monitor interval="60s"
$ pcs constraint location add loc-st-node1hb st-vmware-node1hb node1hb
-INFINITY
$ pcs constraint location add loc-st-node2hb st-vmware-node2hb node2hb
-INFINITY
$ pcs property set stonith-enabled="true"
```

Configure the SAP HAlib

to activate the optional use of the SAP HAlib, you can use the following commands:

first switch on the recording of pending commands in the cluster:

```
$ pcs resource op defaults record-pending=true
```

next add <SID>adm user to "haclient" group on each node

```
$ useradd -a -G haclient rh2adm
```

Make sure saphascriptco.so (see SAP Note 1693245 - SAP HA Script Connector Library) and /usr/sbin/ sap_redhat_cluster_connector (delivered as part of the resource-agents-sap RPM) are present on the system. Since the saphascriptco.so is only included in newer SAP Kernel releases it can be necessary to do an SAP Kernel update to get this library installed.

Add the following to the instance profile of each SAP instance that should be managed via the SAP HA interface:

```
service/halib = /usr/sap/RH2/<Instance (e.g. DVEBMGS01)>/exe/saphascriptco.so
service/halib_cluster_connector = /usr/sbin/sap_redhat_cluster_connector
service/halib_tmp_prefix = <prefix (e.g. /tmp)>
```

6 Testing the setup

Before putting a new cluster setup in production please conduct thorough testing of all failover / split-brain scenarios.

It is very important to verify the continued availability by simulating:

- power outages (pull the plugs! Don't just use the soft-power-off)
- network failures (un-plug the cables)
- SAN failures (depending on configuration)

Consider that the fencing devices (network-based, remote console based) might not work during a power outage however Service failovers depend on a successful fence attempt..

In addition to testing the general cluster functionality with the tests mentioned above it is also important to verify that the cluster reacts correctly when there is a failure on the application level . This can be simulated with the following tests:

- kill the enqueue server
- kill the processes of the DB server

7 Useful commands

7.1 Cluster Management

Show the cluster configuration

```
$ pcs config
```

Monitor the cluster status

```
$ pcs status
```

to monitor the cluster continuously you can use:

```
$ crm_mon
```

Start/Stop a resource

```
$ pcs resource <enable|disable> <resource-name>
```

Switch a resource to unmanaged

an unmanaged resource will still be present in the cluster, but the cluster will no more react on any state changes of that resource. This is very useful while setting up or testing the cluster, or for problem management.

```
$ pcs resource unmanage <resourcename>
```

to switch back the resource to managed mode, use

```
$ pcs resource manage <resourcename>
```

Manage resource failcounts

Pacemaker may not immediately switch on a first problem with the resource. Dependent on the configuration it may first try to restart the resource. Only after a certain limit of retries has been attempted, a cluster switch will take place. The number of retries is stored in the resource failcount.

to view a resource failcount, use

```
$ pcs resource failcount show <resource>
```

to reset the failcount of a resource, use

```
$ pcs resource failcount reset <resource>
```

to cleanup a resource, use

```
$ pcs resource cleanup <resource>
```

a cleanup means, that not only the failcount is reset, but the whole resource history will be deleted. The status the resource is in after a cleanup should be the same like at the moment when the resource had just been created.

7.2 SAP mangement

Check VG tags

```
$ vgs -o +vg_tags
```

Manually test SAP RAs

```
$ OCF_RESKEY_InstanceName="RH2_DVEBMGS01_rh2pas" \
  OCF_RESKEY_DIR_PROFILE="/sapmnt/RH2/profile" \
  OCF_RESKEY_START_PROFILE="/sapmnt/RH2/profile/RH2_DVEBMGS01_rh2pas" \
  OCF_ROOT="/usr/lib/ocf" \
  /usr/lib/ocf/resource.d/heartbeat/SAPInstance status

$ OCF_RESKEY_AUTOMATIC_RECOVERY="TRUE" OCF_RESKEY_DBTYPE="DB6" \
  OCF_RESKEY_SID="RH2" OCF_RESKEY_STRICT_MONITORING="TRUE"
  OCF_ROOT="/usr/lib/ocf" \
  /usr/lib/ocf/resource.d/heartbeat/SAPDatabase status
```

Check status of a db instance

```
$ /usr/sap/hostctrl/exe/saphostctrl -function ListDatabases
```

Manually Start/Stop Database via SAP HostAgent

start:

```
$ /usr/sap/hostctrl/exe/saphostctrl -function StartDatabase -dbname RH2
  -dbtype DB6 -service
```

stop:

```
$ /usr/sap/hostctrl/exe/saphostctrl -function StopDatabase -dbname RH2
  -dbtype DB6 -force -service
```

Check and manage SAP instance manually

For the following commands to work the "sapstartsrv" process for each instance must already be running (can be stated via "service sapinit start" if SAP Instances are defined in /usr/sap/sapservices)

checking the status of an instance:

```
$ LD_LIBRARY_PATH=/usr/sap/RH2/ASCS00/exe/ /usr/sap/RH2/ASCS00/exe/sapcontrol \
   -nr 00 -function GetProcessList

$ LD_LIBRARY_PATH=/usr/sap/RH2/ERS10/exe/ /usr/sap/RH2/ERS10/exe/sapcontrol \
   -nr 10 -function GetProcessList

$ LD_LIBRARY_PATH=/usr/sap/RH2/DVEBMGS01/exe/ \
   /usr/sap/RH2/DVEBMGS01/exe/sapcontrol \
   -nr 01 -function GetProcessList
```

starting a SAP instance:

```
$ LD_LIBRARY_PATH=/usr/sap/RH2/ASCS00/exe/ /usr/sap/RH2/ASCS00/exe/sapcontrol \
  -nr 00 -function Start
```

```
$ LD_LIBRARY_PATH=/usr/sap/RH2/ERS10/exe/ /usr/sap/RH2/ERS10/exe/sapcontrol \
  -nr 10 -function Start

$ LD_LIBRARY_PATH=/usr/sap/RH2/DVEBMGS01/exe/ \
  /usr/sap/RH2/DVEBMGS01/exe/sapcontrol -nr 01 -function Start
```

stop an SAP instance:

```
$ LD_LIBRARY_PATH=/usr/sap/RH2/DVEBMGS01/exe/ \
  /usr/sap/RH2/DVEBMGS01/exe/sapcontrol -nr 01 -function Stop

$ LD_LIBRARY_PATH=/usr/sap/RH2/ERS10/exe/ /usr/sap/RH2/ERS10/exe/sapcontrol \
  -nr 10 -function Stop

$ LD_LIBRARY_PATH=/usr/sap/RH2/ASCS00/exe/ \
  /usr/sap/RH2/ASCS00/exe/sapcontrol -nr 00 -function Stop
```

get information about a SAP instance:

```
$ LD_LIBRARY_PATH=/usr/sap/RH2/ERS10/exe/ /usr/sap/RH2/ERS10/exe/sapcontrol \
  -nr 10 -function GetVersionInfo
```

get a user's SAP environment:

```
$ su - <SID>adm
$ sapcontrol -debug -nr 00 -function GetEnvironment
```

Appendix A: Fix db2nodes.cfg file to enable startup of IBM DB2 database on all cluster nodes

Note: This step is only required for IBM DB2, for other databases like Sybase ASE, Oracle or SAPDB/MaxDB it is not necessary.

As described in SAP cluster and issue with db2nodes.cfg - Red Hat Customer Portal some additional configuration is necessary to allow a DB2 database to run on multiple nodes:

- Edit /etc/hosts on both nodes and add an additional alias 'db2host' to the entries for the real hostnames of both cluster nodes:

```
...
# real system IPs and hostnames on prod network
10.17.140.1   node1.example.com node1 db2host
10.17.140.2   node2.example.com node2 db2host
```

- update `/db2/db2<SID>/sqllib/db2nodes.cfg` to contain the hostname alias instead of the real hostname:
  before:

```
$ cat /db2/db2rh2/sqllib/db2nodes.cfg
0 node1 0
```

  after:

```
$ cat /db2/db2rh2/sqllib/db2nodes.cfg
0 db2host 0
```

Alternatively you can also replace /db2/db2<SID>/sqllib/db2nodes.cfg with a symlink to a local copy of the db2nodes.cfg file on each node containing the hostname of each node, as described in http://scn.sap.com/message/9291463#9291463.

Appendix B: Reference Documentation

The following list includes the existing documentation and articles referenced by this document.

General:

1. **Red Hat Enterprise Linux 6 Release Notes and Installation Guides**

   https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/

2. **Configuring the Red Hat High Availability Add-On with Pacemaker**

   https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Configuring_the_Red_Hat_High_Availability_Add-On_with_Pacemaker/index.html

3. **Pacemaker explained**

   http://clusterlabs.org/doc/en-US/Pacemaker/1.1-pcs/html/Pacemaker_Explained/index.html

4. **SAP Installation Guides**

   http://service.sap.com/instguides

5. **SAP Technical Infrastructure Guide (high-availability)**

   http://scn.sap.com/docs/DOC-7848

6. **SAP High Availability FAQ**

   http://scn.sap.com/docs/DOC-25454

Red Hat Customer Portal KnowledgeBase Articles:

7. **Red Hat Enterprise Linux Cluster, High Availability Knowledge Base Index**

   https://access.redhat.com/kb/docs/DOC-48718

8. **Support for the Pacemaker resource manager in RHEL 6 High Availability clusters**

   https://access.redhat.com/site/solutions/509783

9. **What are the feature differences between `rgmanager` and `pacemaker` in a Red Hat Enterprise Linux 6 High Availability cluster?**

   https://access.redhat.com/site/articles/509563

10. **Virtualization Support for High Availability in Red Hat Enterprise Linux 5 and 6**

    https://access.redhat.com/site/articles/29440

11. **Support for Red Hat Enterprise Linux Cluster and High Availability Stretch Architectures**

    https://access.redhat.com/kb/docs/DOC-58412

12. **Architecture Review Process for Red Hat Enterprise Linux High Availability, Clustering, and GFS/GFS2**

    https://access.redhat.com/kb/docs/DOC-53348

13. **Mounting NFS over loopback results in hangs in Red Hat Enterprise Linux**

    https://access.redhat.com/knowledge/solutions/22231

SAP Notes

14. **SAP software on Linux: General information**

    http://service.sap.com/sap/support/notes/171356

15. **Linux: SAP Support in virtualized environments**

    http://service.sap.com/sap/support/notes/1122387

16. **Linux: High Availability Cluster Solutions**

    http://service.sap.com/sap/support/notes/1552925

17. **Red Hat Enterprise Linux 6.x: Installation and Upgrade**

    http://service.sap.com/sap/support/notes/1496410

18. **Support details for Red Hat Enterprise Linux HA Add-On**

    http://service.sap.com/sap/support/notes/1908655

19. **SAP memory management for 64-bit Linux systems**

    http://service.sap.com/sap/support/notes/941735

20. **License key for high availability environment**

    http://service.sap.com/sap/support/notes/181543

21. **Linux: Alternative method to generate a SAP hardware key**

    http://service.sap.com/sap/support/notes/1178686

22. **SAP HA Script Connector Library**

    http://service.sap.com/sap/support/notes/1693245

Appendix C: Acronyms

AAS	SAP Additional Application Server
ADA	SAP Database Type MaxDB
ABAP	**A**dvanced **B**usiness **A**pplication **P**rogramming, the programming language of SAP
API	Application Programming Interface
ASCS	SAP ABAP Central Services Instance[1]
(C)LVM	(Cluster) Logical Volume Manager
CMAN	Cluster Manager
CRM	Cluster Resource Manager
DB6	SAP Database Type DB2 on Linux
DBMS	DataBase Management System
DLM	Distributed Lock Manager
ERS	SAP Enqueue Replication Server
GFS	Global File System
HA	High-Availability
ICM	Information Chain Management
IP	Internet Protocol
NAS	Network Attached Storage
NFS	Network File Server
NIC	Network Interface Card
NTP	Network Time Protocol
OCF	Open Cluster Framework
ORA	SAP Database Type Oracle
OS	Operating System
PAS	SAP Primary Application Server
RHEL	Red Hat Enterprise Linux
RHN	Red Hat Customer Portal (http://access.redhat.com)
SAN	Storage Area Network
SCS	SAP Central Services Instance (for Java)[1]
SID	System Identification Number
SPOF	Single Point Of Failure
VFS	Virtual File System

1 (A)SCS used in the text to refer to both SCS and ASCS