



# **Automated SAP HANA System Replication with Pacemaker on RHEL**

## **Setup Guide**

Frank Danapfel (Red Hat)

Dieter Thalmayr (Magnum Opus GmbH)

Dieter Jäger (Magnum Opus GmbH)

Version 1.2

June 2015





100 East Davie Street  
Raleigh NC 27601 USA  
Phone: +1 919 754 3700  
Phone: 888 733 4281  
Fax: +1 919 754 3701  
PO Box 13588  
Research Triangle Park NC 27709 USA

.Linux is a registered trademark of Linus Torvalds. Red Hat, Red Hat Enterprise Linux and the Red Hat "Shadowman" logo are registered trademarks of Red Hat, Inc. in the United States and other countries.

AMD is a trademark of Advanced Micro Devices, Inc.

SAP, SAP NetWeaver and SAP HANA are registered trademarks of SAP AG in Germany and several other countries.

ABAP is a trademark of SAP AG in Germany and several other countries.

HANA is a trademark of SAP AG in Germany and several other countries.

UNIX is a registered trademark of The Open Group.

Intel, the Intel logo and Xeon are registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

POSIX is a registered trademark of IEEE.

Oracle is a registered trademark of Oracle Corporation.

IBM is a registered trademark of International Business Machines in many countries worldwide.

VMware, ESX, ESXi, and vSphere, are registered trademarks of VMware, Inc.

All other trademarks referenced herein are the property of their respective owners.

© 2015 by Red Hat, Inc. This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, V1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

The information contained herein is subject to change without notice. Red Hat, Inc. shall not be liable for technical or editorial errors or omissions contained herein.

Distribution of modified versions of this document is prohibited without the explicit permission of Red Hat Inc.

Distribution of this work or derivative of this work in any standard (paper) book form for commercial purposes is prohibited unless prior permission is obtained from Red Hat Inc.

The GPG fingerprint of the [security@redhat.com](mailto: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



# Table of Contents

1 Executive Summary.....	1
2 Concepts and Planning.....	2
2.1 How it works: overview of the basic setup.....	2
2.2 Supported scenarios.....	3
2.3 Resource Agents.....	4
2.3.1 SAPHana Resource Agent.....	4
2.3.2 SAPHanaTopology Resource Agent.....	5
2.4 High Availability Add-On Overview.....	6
2.4.1 CMAN.....	6
2.4.2 Pacemaker.....	6
2.4.3 Fencing/STONITH.....	6
3 Setup and Configuration .....	7
3.1 Introduction.....	7
3.1.1 Parameters used for example configuration.....	7
3.2 Prerequisites.....	8
3.2.1 General preparations.....	8
3.2.2 Storage requirements.....	8
3.2.3 Operating System Installation.....	8
3.2.4 Cluster package installation.....	8
3.3 SAP HANA Installation.....	9
3.4 SAP HANA System Replication Setup.....	10
3.4.1 Primary node.....	10
3.4.2 Secondary Node.....	10
3.4.3 Create a monitoring account in SAP HANA.....	11
3.5 Cluster Setup.....	12
3.5.1 Initialize the cluster.....	12
3.5.2 Configure Fencing/STONITH.....	12
3.5.3 Virtual IP addresses and hostnames.....	13
3.5.4 Create the SAPHanaTopology resource.....	13
3.5.5 Create clone resource for SAPHanaTopology.....	13
3.5.6 Create SAPHana resource.....	14



3.5.7 Create Master/Slave resource for SAPHana.....	14
3.5.8 Set up constraints.....	14
Appendix A: Reference Documentation.....	15
Appendix B: Example Cluster Configuration.....	16
Appendix C: Revision History.....	18



# 1 Executive Summary

As SAP HANA moves to take on a more central function as the primary database platform for SAP landscapes, requirements for stability and reliability increase dramatically. Red Hat Enterprise Linux for SAP HANA with High Availability and Smart Management meets those requirements by enhancing native SAP HANA replication and fail-over technology to automate the takeover process. During a fail-over in a typical SAP HANA System Replication deployment, a system administrator must manually instruct the application to perform a takeover to the secondary server in case there is an issue on the primary server. Red Hat Enterprise Linux for SAP HANA with High Availability and Smart Management provides pacemaker resource agents to automate the takeover from primary to secondary server.

Red Hat Enterprise Linux for SAP HANA with High Availability and Smart Management consist of these components:

- Red Hat Enterprise Linux for SAP HANA
- Red Hat Enterprise Linux High Availability Add-on

This document list the technical prerequisites for Pacemaker clustering and provides a step-to-step guide on how to set up a working solution using Red Hat Enterprise Linux for SAP HANA with High Availability and Smart Management together with SAP HANA.

Although the solution shown here will be a working one the challenges of a real implementation are usually more complex than what this document can cover. Red Hat therefore recommends that setting up and subsequently servicing such a solution should be guided by a certified consultant familiar with both SAP HANA and the Pacemaker based RHEL HA add-on.



## 2 Concepts and Planning

SAP HANA already provides some of the most crucial cluster components of system replication and auto fail-over of the software, with one active node, and one standby node, as well as storage replication and system replication. The host system needs to provide a mechanisms for monitoring the SAP processes and network resources and to initiate and control the fail-over process in case of a problem. It also will provide a fencing mechanism to make sure, that in case of a fail-over, the cluster will end up in a well defined state.

In this document we will focus on the second part, the host system cluster setup. For the setup of the SAP HANA components, please refer to the SAP HANA How To-Guide (see page 15, Appendix A).

### ***2.1 How it works: overview of the basic setup***

SAP HANA provides different kinds of high availability, aiming for a simple but highly customizable setup.

In this document we describe how to use Pacemaker to automate a SAP HANA Scale-Up setup with SAP HANA System Replication between two nodes. All data is preloaded (in-memory-database). All changes are constantly replicated from the master node to the slave node, so that in case of a fail-over recovery time can be minimized.



## 2.2 Supported scenarios

For the first version of the SAP HANA System Replication HA solution the support is limited to the following scenarios and parameters:

- Both SAP HANA SPS8 as well as SPS9 are supported
- Two-Node SAP HANA Scale-Up (single-box to single-box) System Replication only, no support for HANA Scale-Out SR or HANA installations configured with a standby node
- Both SAP HANA instances must have the same SAP Identifier (SID) and Instance-Number
- No other SAP HANA system (like QAS or TST) on the replicating node which needs to be stopped during takeover.
- Only one system replication for the SAP HANA database, no "Multitier System Replication"/"replication chain" (the resource agent is able to detect if Multitier System Replication is configured but it can't act in this case since it could possibly result in an unsupported configuration for the HANA System Replication (star topology))
- No support for SAP HANA System Replication "Multiple components One Database (MCOD)" (<http://scn.sap.com/docs/DOC-59893>) setups; MCOS is only supported if all databases running on the hosts are replicated and the replication is always to the same secondary node
- All nodes must be in the same network segment (layer 2)
- Technical users and groups such as "<SID>adm" must be identically defined on all cluster nodes
- Name resolution of the cluster nodes and the virtual IP address can be done locally on all cluster nodes
- Time on all cluster nodes must be in sync (using NTP or some other time synchronization method)
- If the cluster nodes are installed in different data centers or data center areas, the environment must match both the requirements defined by SAP for HANA System Replication (see chapter "4.2 Distance between data centers" in the SAP "[How to Perform System Replication for SAP HANA](#)" guide) and also the RHEL HA add-on stretch cluster requirements, specifically the network latencies between the nodes and the recommended maximum distance



## 2.3 Resource Agents

The central part of the Pacemaker cluster setup are two Open Cluster Framework (OCF) resource agents:

- SAPHana
- SAPHanaTopology

### 2.3.1 SAPHana Resource Agent

The SAPHana resource agent runs on both nodes in a master/slave configuration. The master assumes responsibility for the SAP HANA databases running in primary mode, and the slave is responsible for instances that are operated in synchronous (secondary) status.

The SAPHana resource agent understands the following mandatory parameters:

Attribute	Default	Description
SID	none	The unique SAP system identifier (e.g. COR)The SAP System Identifier (SID) of the SAP HANA installation (must be identical for the SAP HANA installations on all nodes) Example: RH1
InstanceNumber	none	The Instance Number of the SAP HANA installation (must be identical for the SAP HANA installations on all nodes) Example:00
PREFER_SITE_TAKEOVER	none	Should resource agent prefer to switch over to slave instance instead of restarting master locally? true: do prefer takeover to remote site false: do prefer restart locally
AUTOMATED_REGISTER	false	Whether a former primary instance should be registered automatically as a secondary by the resource agent during cluster/resource start, if a takeover event has occurred and the DUPLICATE_PRIMARY_TIMEOUT is expired
DUPLICATE_PRIMARY_TIMEOUT	7200	Time difference needed between two primary time stamps, if a dual-primary situation occurs. If the time difference is less than the time gap, than the cluster hold one or both instances in a "WAITING" status. This is to give a admin a chance to react on a takeover. If AUTOMATED_REGISTER is set to "true" a failed former primary will be registered after the time difference is passed. After this





Attribute	Default	Description
		registration as a secondary to the new primary all data will be overwritten by the system replication.
SAPHanaFilter	ra-act-d ec-lpa	Define SAPHana resource agent messages to be printed. Possible Values: any combination of “ra-act-lpa-dec-flow-dbg-dbg2” or “all” or “none” The default is sufficient for normal operation, and should only be changed for debugging purposes

## 2.3.2 SAPHanaTopology Resource Agent

The SAPHanaTopology resource agent runs as a stateless cloned agent on all nodes and gathers information about the status and configuration of all SAP HANA system replications.

The SAPHanaTopology agent understands the following mandatory parameters:

Attribute	Default	Description
SID	none	The unique SAP system identifier (e.g. COR)The SAP System Identifier (SID) of the SAP HANA installation (must be identical for the SAP HANA installations on all nodes) Example: RH1
InstanceNumber	none	The Instance Number of the SAP HANA installation (must be identical for the SAP HANA installations on all nodes) Example:00
SAPHanaFilter	ra-act-d ec-lpa	Define SAPHanaTopology resource agent messages to be printed. Possible Values: any combination of “ra-act-lpa-dec-flow-dbg-dbg2” or “all” or “none” The default is sufficient for normal operation, and should only be changed for debugging purposes



## 2.4 High Availability Add-On Overview

The Red Hat Enterprise Linux for SAP High Availability add-on provides the following components to set up HA environments:

- Cluster Manager (CMAN)
- Pacemaker
- STONITH

### 2.4.1 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.

### 2.4.2 Pacemaker

Pacemaker is the Cluster Resource Manager. Clustered services and resources are defined there. Pacemaker can control user requests like *service start*, *restart*, *disable* or *relocate*, as well as provide fail-over capabilities as necessary.

Pacemaker uses OCF-compliant resource agents to control and monitor required resources.

### 2.4.3 Fencing/STONITH

All cluster relevant components are installed locally on all participating nodes so both of the cluster member nodes can perform all services. In case of switching services, the now-disabled node has to be switched off, which is why there has to be a working power fencing mechanism in place. In our example setup, we will describe an *ipmi* fencing. But Pacemaker supports several different fencing methods, and every one of them may be applied. Since all storage components are exclusively local, there is no need for storage fencing.



# 3 Setup and Configuration

## 3.1 Introduction

This part of the document describes the entire installation process based on an example configuration. It provides some background information, describes the necessary steps and the commands to use.

Setups will hardly ever be performed exactly the same way as they are described here; it is the consultant's job to understand and modify the configuration and commands as appropriate to the respective context.

Basically, the whole process includes the following steps:

- Installing and configuring Red Hat Enterprise Linux for SAP HANA on all nodes according to the SAP guidelines
- Installing SAP HANA on all nodes using the same SID and instance number
- Manually configuring SAP HANA System Replication and verifying that it works according to SAP guidelines
- Installation of the RHEL HA add-on and configuration of the cluster
- Performing cluster tests to verify that fail-over works under all circumstances

### 3.1.1 Parameters used for example configuration

These are the parameter values that are used for the example setup described in the following chapters:

Parameter	Value
SID	RH1
Instance Number	00
Primary HANA Site name	DC1
Secondary HANA Site name	DC2
Public network	192.168.1.0/24
Cluster network	192.168.2.0/24
Virtual IP for clients to connect to	192.168.1.10
Primary Node public hostname/IP	node1/192.168.1.11
Primary Node cluster hostname/IP	node1hb/192.168.2.11
Primary Node cluster hostname/IP	node2/192.168.1.12
Secondary Node cluster hostname/IP	node2hb/192.168.2.12



## 3.2 Prerequisites

### 3.2.1 General preparations

Make sure your environment meets the following requirements:

- A working Fencing/STONITH mechanism must be available for all cluster nodes; HA setups without working Fencing/STONITH are not supported by Red Hat
- The time on all nodes must be in sync. This can be achieved by using a time synchronization mechanism like NTP

### 3.2.2 Storage requirements

SAP HANA is an in-memory database. All cluster relevant components are kept in memory and are replicated to the cluster nodes using replication mechanisms provided by SAP HANA, so there is no need for shared storage, which simplifies the storage setup very much.

### 3.2.3 Operating System Installation

Please make sure that the OS on all nodes is configured according to the SAP guidelines for installation of SAP HANA on RHEL (see Appendix A on page 15).

### 3.2.4 Cluster package installation

In addition please make sure that all cluster nodes are subscribed to the "RHEL Server High Availability" channel and install the cluster packages:

```
[root]# yum install cman pacemaker pcs resource-agents-sap-hana
```

Make sure the cluster will not be started at boot time by switching off the Corosync framework daemon:

```
[root]# chkconfig corosync off
```



### 3.3 SAP HANA Installation

For SAP HANA System Replication to work SAP HANA must be installed using the same SID and Instance Number on all nodes. The installation of SAP HANA is not covered in this document. Please refer to the setup manuals available at the [SAP Service Marketplace](#), download the SAP HANA software and perform the SAP HANA installation on all nodes following the instructions in the SAP Installation Guides.

After successful installation of the database, verify that SAP HANA is running on all nodes by using the SAP HDB tool:

```
rh1adm@node1:/usr/sap/RH6/HDB00> HDB info
USER          PID  PPID %CPU  VSZ   RSS  COMMAND
rh1adm        4949  4948  1.5 110452 3924 -bash
rh1adm        5032  4949  0.0 114064 1924
\_ /bin/sh /usr/sap/RH6/HDB00/HDB info
rh1adm        5059  5032  0.0 118036 1508      \_ ps fx -U rh6adm -o
user,pid,ppid,pcpu,vsz,rss,args
rh1adm        16918  1  0.0 22088 1496 sapstart
pf=/usr/sap/RH1/SYS/profile/RH6_HDB00_node1
rh1adm        16927 16918  0.0 896912 301692
\_ /usr/sap/RH1/HDB00/node1/trace/hdb.sapRH1_HDB00 -d -nw
-f /usr/sap/RH1/HDB00/node1/daemon.ini
pf=/usr/sap/RH1/SYS/profile/RH1_HDB00_node1
rh1adm        16947 16927  3.5 9589920 1925852      \_ hdbnameserver
rh1adm        16998 16927  0.0 7842256 238936      \_ hdbpreprocessor
rh1adm        17001 16927  0.0 7620992 235508      \_ hdbcompileserver
rh1adm        17093 16927  4.4 12878916 3763036      \_ hdbindexserver
rh1adm        17096 16927  4.4 12475796 4040556      \_ hdbstatisticsserver
rh1adm        17099 16927  3.5 11216184 2070452      \_
hdbxsengine
rh1adm        17535 16927  0.0 930408 72500      \_ sapwebdisp_hdb
pf=/usr/sap/RH1/HDB00/node1/wdisp/sapwebdisp.pfl
-f /usr/sap/RH1/HDB00/node1/trace/dev_webdisp
rh6adm        23804  1  0.0 678508
75720 /usr/sap/RH1/HDB00/exe/sapstartsrv
pf=/usr/sap/RH1/SYS/profile/RH1_HDB00_node1 -D -u rh1adm
```

For the system replication to work, the SAP HANA "log\_mode" variable must be set to "normal". This can be verified after installation with the following command:

```
[rh1adm]# hdbsql -u system -i 00 "select value from
\"SYS\".\"M_INIFILE_CONTENTS\" where key='log_mode'"
Password:
VALUE "normal"
1 row selected (overall time 191.885 msec; server time 188.704 msec)
```

Also, please make sure that SAP HANA is not configured to automatically start after system boot, since startup and shutdown of SAP HANA will be controlled by the cluster (must be done on all nodes):

```
[root]# grep Autostart /usr/sap/RH1/SYS/profile/*
/usr/sap/RH1/SYS/profile/RH1_HDB00_node1:Autostart = 0
```



## 3.4 SAP HANA System Replication Setup

If SAP HANA System Replication uses a separate network for its traffic, you need to configure the System Replication hostname resolution as described in [Host Name Resolution for System Replication](#)

### 3.4.1 Primary node

The first step is to prepare the primary node. The SAP HANA system replication will only work after an initial backup has been performed:

```
[rh1adm]# hdbsql -i 00 -u system -p <pw of user system> "BACKUP DATA USING
FILE ('/tmp/foo')"
```

0 rows affected (overall time 35.458355 sec; server time 35.426372 sec)

After successful backup, replication can be initialized:

```
[rh1adm]# hdbnsutil -sr_enable --name=DC1
checking for active nameserver ...
nameserver is active, proceeding ...
successfully enabled system as system replication source site
done.
```

As a final step, check replication status and make sure it both shows the current node is running and the current node as primary.

```
[rh1adm]# hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
-----
mode: primary
site id: 1
site name: DC1
Host Mappings:
```

### 3.4.2 Secondary Node

The second step is to initialize the secondary node. This has only one step: register the second node to the SAP HANA cluster

```
[rh1adm]# hdbnsutil -sr_register --remoteHost=node1 --remoteInstance=00
--mode=syncmem --name=DC2
adding site ...
checking for inactive nameserver ...
nameserver node2:30001 not responding.
collecting information ...
updating local ini files ...
done.
```

Make sure that everything has worked as expected by checking the replication status. You should now see both nodes, node1 still as primary; the secondary node should show up in “syncmem” mode.

```
[rh1adm]# hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
-----
```



```
mode: syncmem
site id: 2
site name: DC2
active primary site: 1
re-setup replication: hdbnsutil -sr_register --name=DC2 -mode=syncmem
--remoteHost=node1 --remoteInstance=00
Host Mappings:
-----
node2 -> [DC1] node1
node2 -> [DC2] node2
```

### 3.4.3 Create a monitoring account in SAP HANA

A technical user with "DATA ADMIN" privileges must exist in SAP HANA for the resource agents to be able to run queries on the system replication status. Run the following commands on primary node to create a technical user "rhelhasync" inside the HANA DB (replace <password> with your own password):

```
[rh1adm]# hdbsql -i 00 -u system -p <pw of system user> "create user
rhelhasync password <password>"
[rh1adm]# hdbsql -i 00 -u system -p <pw of system user> "grant DATA ADMIN to
rhelhasync"
[rh1adm]# hdbsql -i 00 -u system -p <pw of system user> "ALTER USER
rhelhasync DISABLE PASSWORD LIFETIME"
```

Then add a userkey "SAPHANA<SID>SR" for this user to the local HANA user store for the root user. Run the following command on all nodes:

```
[root]# hdbuserstore SET SAPHANA<SID>SR localhost:30015 rhelhasync
<password>
```

(replace <SID> with the SID of the SAP HANA installation; for example if the SID of your HANA installation is "RH1"; the userkey therefore must be "SAPHANARH1SR")

Please verify that it is possible to run hdbsql commands as root using the SAPHANA<SID>SR userkey without being prompted for a password, e.g.:

```
[root]# /usr/sap/RH1/HDB00/exe/hdbsql -U SAPHANARH1SR -i 00 "select distinct
REPLICATION_STATUS from SYS.M_SERVICE_REPLICATION"
```

If you get an error message about issues with the password or are being prompted for a password please verify with HANASTudio that the password for the user "rhelhaloc" is not configured to be changed on first login or that the password has expired.



## 3.5 Cluster Setup

### 3.5.1 Initialize the cluster

Run the following command on each cluster node to configure the cluster infrastructure and start the cluster:

```
[root]# pcs cluster setup --name hanasr --local node1hb node2hb
[root]# service pacemaker start
```

Check if your cluster node is running and connected with the following command:

```
root]# pcs status
Cluster name: hanasr
Last updated: Thu Dec 4 12:14:34 2014
Last change: Mon Dec 1 17:05:31 2014
Stack: cman
Current DC: node1 - partition with quorum
Version: 1.1.11-97629de
2 Nodes configured
0 Resources configured
Online: [ node1hb node2hb ]
Full list of resources:
[...]
```

The following commands need only to be issued on one of the cluster nodes, as Pacemaker propagates configuration to all cluster nodes automatically the moment they are committed.

Configure some basic cluster parameters. As this is a two-node cluster you have to set “no-quorum-policy” to “ignore”. Adjust resource stickiness and default timeout to values which are manageable with the SAP HANA system:

```
[root]# pcs property set no-quorum-policy="ignore"
[root]# pcs resource defaults default-resource-stickiness=1000
[root]# pcs resource defaults default-migration-threshold=5000
[root]# pcs resource op defaults timeout=600s
```

### 3.5.2 Configure Fencing/STONITH

The following is just an example of how to configure fencing using the ipmi fencing method. Please choose the correct fencing method for your environment.

To use IPMI for fencing you can set up STONITH with the following commands:

```
[root]# pcs stonith create st_ipmi_node1 fence_ipmilan \
ipaddr=<ip of management card of node1> \
login="<user of management card of node1>" \
passwd="<password of user of management card of node1>" \
pcmk_host_list="node1hb"
[root]# pcs stonith create st_ipmi_node2 fence_ipmilan \
ipaddr=<ip of management card of node2> \
login="<user of management card of node2>" \
passwd="<password of user of management card of node2>" \
pcmk_host_list="node2hb"
[root]# pcs constraint location st-ipmi-node1 avoids node2
[root]# pcs constraint location st-ipmi-node2 avoids node1
```





Unfortunately "pcs stonith" currently does not provide any information on which node the fencing agent is running, so you have to use "pcs status" to get this information:

```
[root]# pcs status
Cluster name: hanasr
Last updated: Thu Dec 4 12:14:34 2014
Last change: Mon Dec 1 17:05:31 2014
Stack: cman
Current DC: node1 - partition with quorum
Version: 1.1.11-97629de
2 Nodes configured
2 Resources configured
Online: [ node1 node2 ]
Full list of resources:
st-ipmi-node1 (stonith:fence_ipmilan): Started node2hb
st-ipm-node2 (stonith:fence_ipmilan): Started node1hb
```

To check if fencing actually works, test the fencing by initializing fencing on node2. Verify if fencing was successful by monitoring node2 activity:

```
pcs stonith fence node2
```

### 3.5.3 Virtual IP addresses and hostnames

Now set up the virtual IP address of the cluster. This address will always be connected to the active node. Make sure that your switches and network will permit the ip takeover:

```
[root]# pcs resource create rsc_ip_SAPHana_RH1_HDB00 IPaddr2 \
ip="192.168.1.10"
```

### 3.5.4 Create the SAPHanaTopology resource

Now create the primary SAPHanaTopology resource:

```
[root]# pcs resource create rsc_SAPHanaTopology_RH1_HDB00 SAPHanaTopology \
SID=RH1 \
InstanceNumber=00 \
op start timeout=600 \
op stop timeout=300 \
op monitor interval=10 timeout=600
```

### 3.5.5 Create clone resource for SAPHanaTopology

Create a clone of the primary resource, making sure that SAP HANA will be active on both nodes:

```
# pcs resource clone rsc_SAPHanaTopology_RH1_HDB00 \
meta is-managed=true clone-node-max=1 target-role="Started" interleave=true
```



### 3.5.6 Create SAPHana resource

Create the SAP HANA resource. This master/slave resource will make sure that the SAP HANA system is running on both nodes, where the master will be the primary active node<sup>1</sup>:

```
[root]# pcs resource create rsc_SAPHana_RH1_HDB00 SAPHana \  
SID=RH1 \  
InstanceNumber=00 \  
PREFER_SITE_TAKEOVER=true \  
DUPLICATE_PRIMARY_TIMEOUT=7200 \  
AUTOMATED_REGISTER=false \  
op start timeout=3600 \  
op stop timeout=3600 \  
op promote timeout=3600 \  
op demote timeout=3600 \  
op monitor interval=59 role="Master" timeout=700 \  
op monitor interval=61 role="Slave" timeout=700
```

### 3.5.7 Create Master/Slave resource for SAPHana

Make the SAPHana resource a master/slave resource:

```
root]# pcs resource master msl_rsc_SAPHana_RH1_HDB00 rsc_SAPHana_RH1_HDB00 \  
meta is-managed=true notify=true clone-max=2 clone-node-max=1 \  
target-role="Started" interleave=true
```

### 3.5.8 Set up constraints

Both resources, SAPHanaTopology and SAPHana, need to run on the same node. They have to be started and activated/deactivated in the correct order:

```
[root]# pcs constraint colocation add rsc_ip_SAPHana_RH1_HDB00 with master \  
msl_rsc_SAPHana_RH1_HDB00 2000 \  
[root]# pcs constraint order rsc_SAPHanaTopology_RH1_HDB00-clone then \  
msl_rsc_SAPHana_RH1_HDB00 symmetrical=false
```

---

1 The timeouts shown below for the start, stop, promote and demote operations are only examples and should be adjusted on the local SAP HANA setup (e. g. large HANA DBs can take longer to start up therefore the start timeout might have to be increased)



# Appendix A: Reference Documentation

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

- [SAP Note 2009879 - SAP HANA Guidelines for RedHat Enterprise Linux \(RHEL\) Operating System](#)
- [SAP Note 2013638 - SAP HANA DB: Recommended OS settings for RHEL 6.5](#)
- [SAP Note 2136965 - SAP HANA DB: Recommended OS settings for RHEL 6.6](#)
- [SAP Note 2001528 - Linux: SAP HANA Database SPS 08 revision 80 \(or higher\) on RHEL 6 or SLES 11](#)
- [SAP Note 2063657 - HANA System Replication takeover decision guideline](#)
- [Availability and Scalability - SAP HANA Administration Guide](#)
- [How to Perform System Replication for SAP HANA](#)
- [HANA System Replication - Take-over process](#)
- [Configuring the Red Hat High Availability Add-On with Pacemaker](#)



# Appendix B: Example Cluster Configuration

Here is the example configuration we used in this document:

```
Cluster Name: hanasr
Corosync Nodes:
node1hb node2hb
Pacemaker Nodes:
node1hb node2hb

Resources:
Clone: rsc_SAPHanaTopology_RH1_HDB00-clone
  Meta Attrs: is-managed=true clone-node-max=1 target-role=Started
  Resource: rsc_SAPHanaTopology_RH1_HDB00 (class=ocf provider=heartbeat
type=SAPHanaTopology)
  Attributes: SID=RH1 InstanceNumber=00
  Operations: start interval=0 timeout=600
(rsc_SAPHanaTopology_RH1_HDB00-start-timeout-600)
  stop interval=0 timeout=300
(rsc_SAPHanaTopology_RH1_HDB00-stop-timeout-300)
  monitor interval=10 timeout=600
(rsc_SAPHanaTopology_RH1_HDB00-monitor-interval-10-timeout-600)
Master: msl_rsc_SAPHana_RH1_HDB00
  Meta Attrs: is-managed=true notify=true clone-max=2 clone-node-max=1
  Resource: rsc_SAPHana_RH1_HDB00 (class=ocf provider=heartbeat
type=SAPHana)
  Attributes: SID=RH1 InstanceNumber=00 PREFER_SITE_TAKEOVER=true
DUPLICATE_PRIMARY_TIMEOUT=7200 AUTOMATED_REGISTER=false
  Operations: start interval=0 timeout=3600
(rsc_SAPHana_RH1_HDB00-name-start-timeout-3600)
  stop interval=0 timeout=3600
(rsc_SAPHana_RH1_HDB00-name-stop-timeout-3600)
  promote interval=0 timeout=3600
(rsc_SAPHana_RH1_HDB00-name-promote-timeout-3600)
  demote interval=0s timeout=3600
(rsc_SAPHana_RH1_HDB00-name-demote-timeout-3600)
  monitor interval=59 role=Master timeout=700
(rsc_SAPHana_RH1_HDB00-name-monitor-interval-59-role-Master-timeout-700)
  monitor interval=61 role=Slave timeout=700
(rsc_SAPHana_RH1_HDB00-name-monitor-interval-61-role-Slave-timeout-700)
  Resource: rsc_ip_SAPHana_RH1_HDB00 (class=ocf provider=heartbeat
type=IPAddr2)
  Attributes: ip=192.168.1.10
  Operations: start interval=0s timeout=20s
(rsc_ip_SAPHana_RH1_HDB00-start-timeout-20s)
  stop interval=0s timeout=20s
(rsc_ip_SAPHana_RH1_HDB00-stop-timeout-20s)
  monitor interval=10s timeout=20s
(rsc_ip_SAPHana_RH1_HDB00-monitor-interval-10s)
```



#### Stonith Devices:

```
Resource: st-ipmi-nodel (class=stonith type=fence_ipmilan)
  Attributes: ipaddr=nodel-drac login=xxxx passwd=yyyy
pcmk_host_list=nodelhb
  Operations: monitor interval=60s (st-ipmi-nodel-monitor-interval-60s)
Resource: st-ipmi-node2 (class=stonith type=fence_ipmilan)
  Attributes: ipaddr=node2-drac login=xxxx passwd=yyyy
pcmk_host_list=node2hb
  Operations: monitor interval=60s (st-ipmi-node2-monitor-interval-60s)
Fencing Levels:
```

#### Location Constraints:

```
Resource: st-ipmi-nodel
  Disabled on: nodelhb (score:-INFINITY)
(id:location-st-ipmi-nodel-nodelhb--INFINITY)
Resource: st-ipmi-node2
  Disabled on: node2hb (score:-INFINITY)
(id:location-st-ipmi-node2-node2hb--INFINITY)
```

#### Ordering Constraints:

```
start rsc_SAPHanaTopology_RH1_HDB00-clone then start
msl_rsc_SAPHana_RH1_HDB00 (kind:Mandatory) (non-symmetrical)
(id:order-rsc_SAPHanaTopology_RH1_HDB00-clone-msl_rsc_SAPHana_RH1_HDB00-mand
atory)
```

#### Colocation Constraints:

```
rsc_ip_SAPHana_RH1_HDB00 with msl_rsc_SAPHana_RH1_HDB00 (score:2000)
(rsc-role:Started) (with-rsc-role:Master)
(id:colocation-rsc_ip_SAPHana_RH1_HDB00-msl_rsc_SAPHana_RH1_HDB00-2000)
```

#### Cluster Properties:

```
cluster-infrastructure: cman
dc-version: 1.1.11-97629de
last-lrm-refresh: 1418642423
no-quorum-policy: ignore
```



# Appendix C: Revision History

Revision 1.0	Friday, May 29, 2015	Dieter Jäger, Dieter Thalmayr, Frank Danapfel
Initial Release		
Revision 1.1	Tuesday, June 2, 2015	Frank Danapfel
Corrections and Clarifications		
Revision 1.2	Monday, June 8, 2015	Lis Strenger
Formatting Correction		

