SAP HANA Business Continuity with SAP HANA System Replication and SUSE Cluster on Cisco Unified Computing System

Based on SAP HANA 2.0 with SUSE Linux Enterprise for SAP Applications 12 SP2 and Cisco UCS 3.1

Contents

Trademarks and disclaimers

Cisco's trademarks signify Cisco's high-quality products and services, and they are valuable assets of the company. Cisco's trademark and brand policies can be found at http://www.cisco.com/go/logo. The following is a list of Cisco's trademarks and registered trademarks in the United States and certain other countries. Please note, however, that this listing is not all-inclusive, and the absence of any mark from this list does not mean that it is not a Cisco trademark.

http://www.cisco.com/web/siteassets/legal/trademark.html

Document history

Version	Date	Author	Document change
1.0	March 5, 2015	Ralf Klahr	Final version
1.1	March 26, 2015	Ralf Klahr	Cluster administration and switchback
2.0	May 2017	Ralf Klahr Matt Schlarb	Technology update
2.1	September 2017	Ralf Klahr Matt Schlarb	IPMI watchdog

Overview

This document is not a step-by-step installation guide but instead provides guidance for setting up SAP HANA system replication. This document refers to specific vendor information, such as the SUSE SAP HANA replication white paper and other SAP HANA documentation listed at the end of this document.

At this point, automatic failover is available only with scale-up (single-system) architecture. Cisco is working with SUSE to enable automatic failover for scale-out (multiple-system) designs.

SAP HANA is SAP's implementation of in-memory database technology.

The SAP HANA database takes advantage of the low cost of main memory (RAM), the data processing capabilities of multicore processors, and the fast data access of solid-state disk (SSD) drives relative to traditional hard-disk drives (HDDs) to deliver better performance for analytical and transactional applications. Its multiple-engine query processing environment allows it to support relational data (with both row- and column-oriented physical representations in a hybrid engine) as well as graph and text processing for management of semistructured and unstructured data within the same system. The SAP HANA database is 100 percent compliant with the atomicity, consistency, isolation, and durability (ACID) model.

For more information about SAP HANA, see the SAP help portal, at http://help.sap.com/hana/.

Audience

The intended audience for this document includes sales engineers, field consultants, professional services staff, IT managers, partner engineering staff, and customers deploying SAP HANA.

Architecture

The solution presented in this document is based on the Cisco Unified Computing System™ (Cisco UCS®) and FlexPod, but it could also be configured with any other storage system or internal storage. The solution provides a general approach to a single system (scale-up design) with system replication and automated failover.

All Cisco UCS hardware listed in the Certified and Supported SAP HANA Hardware Directory can be used for SAP HANA System Replication scenarios.

Network recommendations

In general, SAP does not give a network bandwidth recommendation for the replication network.

The bandwidth required between the two sites depends by the change rates of the database. Therefore, you should start with a 10-Gbps Layer 2 connection for one scale-up system. If you configure a scale-out system, the bandwidth can be increased to the number of nodes times 10 Gbps.

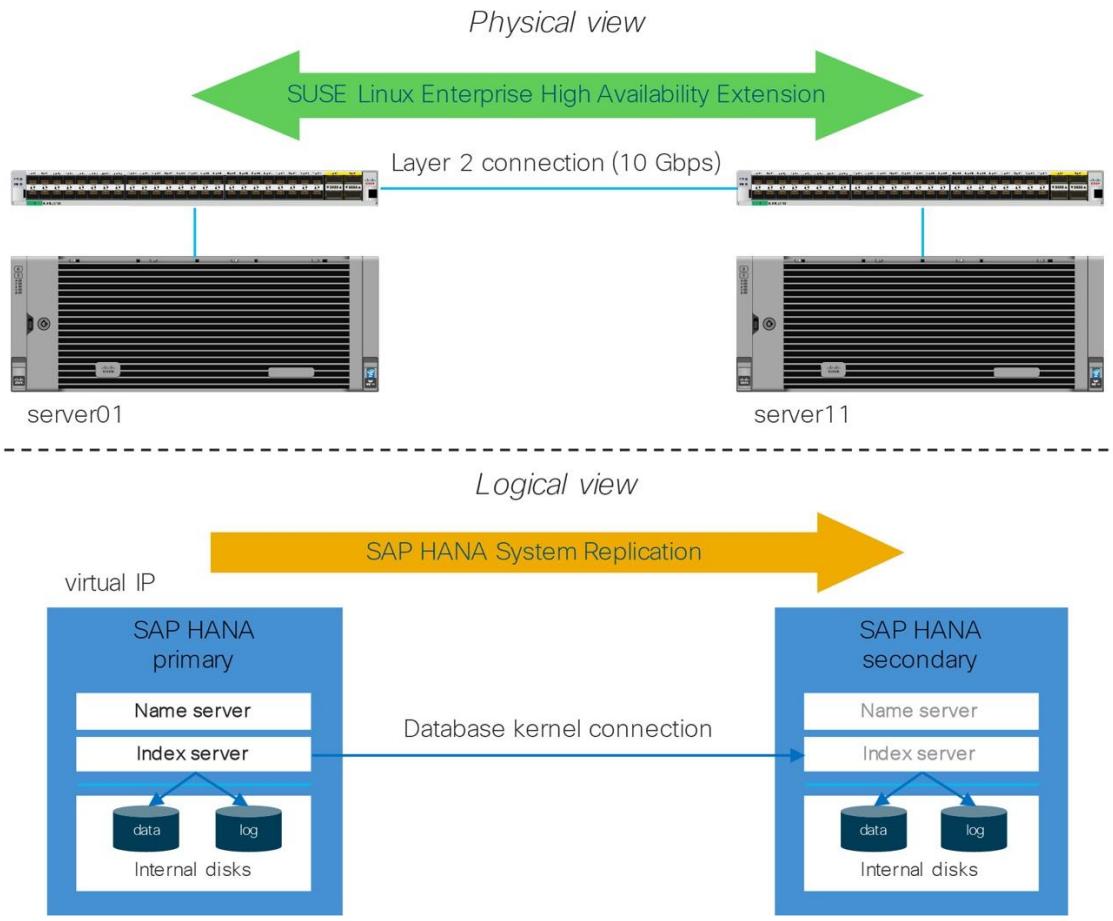
If the change rate is not high, less bandwidth may be required. In this case, even a switched (Layer 3) connection is possible.

Single-node disaster-tolerance configuration

The single-node disaster-tolerance configuration is built on SLES for SAP Applications 12 SP2 with installed SUSE Linux Enterprise High Availability Extension (SLE HAE) packages. SUSE provides a step-by-step description of how to set up the SLE HAE cluster. Please visit https://www.suse.com/products/sles-for-sap/resource-library/sap-best-practices/ to get the latest copy of it.

Figure 1 shows the design of the current version of the solution.

Figure 1. SAP HANA System Replication physical and logical view



Limitations of the current resource agent release

The SAP HANA system replication resource agent (SAPHanaSR) supports SAP HANA for system replication beginning with SAP HANA Version 1.0 SPS07. For the current version of the resource agent software package, support is limited to certain scenarios. Please visit https://wiki.scn.sap.com/wiki/display/ATopics/SAP+on+SUSE to get the latest information about what is supported.

Prepare for installation

Both nodes for system replication must be installed identically. SAP Notes 1944799 and 2205917 provide installation guidance.

After the installation is complete, update the nodes with the latest available kernel and OS patches for SLES for SAP 12 SP2.

```
server01:~ # uname -a
Linux server01 4.4.38-93-default #1 SMP Wed Dec 14 12:59:43 UTC 2016 (2d3e9d4) x86_64 x86_64
x86_64 GNU/Linux


server11:~ # uname -a
Linux server11 4.4.38-93-default #1 SMP Wed Dec 14 12:59:43 UTC 2016 (2d3e9d4) x86_64 x86_64
x86_64 GNU/Linux
```

Install SAP HANA

SAP HANA must be installed on both systems with the exact same system ID (SID) and instance number.

server01 (primary database)

```
server11:/ #  su - anaadm
server01:/usr/sap/ANA/HDB00> sapcontrol -nr 00 -function GetProcessList
 name, description, dispstatus, textstatus, starttime, elapsedtime, pid
 hdbdaemon, HDB Daemon, GREEN, Running, 2015 01 14 18:40:39
 hdbnameserver, HDB Nameserver, GREEN, Running, 2015 01 14 18:40:45
 hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2015 01 14 18:40:57
 hdbindexserver, HDB Indexserver, GREEN, Running, 2015 01 14 18:41:08
 hdbstatisticsserver, HDB Statisticsserver, GREEN, Running, 2015 01 14 18:41:08
 hdbxsengine, HDB XSEngine, GREEN, Running, 2015 01 14 18:41:08
 sapwebdisp_hdb, SAP WebDispatcher, GREEN, Running, 2015 01 14 18:42:31
 hdbcompileserver, HDB Compileserver, GREEN, Running, 2015 01 14 18:40:57

 server01:/usr/sap/ANA/HDB00> HDB info
 USER       PID  PPID %CPU    VSZ   RSS COMMAND
 anaadm   13025 13024  0.0  13884  2788 -sh
 anaadm   13119 13025  0.3  12900  1720  \_ /bin/sh /usr/sap/ANA/HDB00/HDB info
 anaadm   13142 13119  3.2   4944   876      \_ ps fx -U anaadm -o
user,pid,ppid,pcpu,vsz,rss,args
 anaadm   11858     1  0.0  22024  1508 sapstart pf=/hana/shared/ANA/profile/ANA_HDB00_server01
 anaadm   11870 11858  0.0 825768 301404  \_ /usr/sap/ANA/HDB00/server01/trace/hdb.sapANA_HDB00
-d -nw -f /usr/s
 anaadm   11889 11870  1.1 23685688 2208920      \_ hdbnameserver
 anaadm   12068 11870 33.1 16865944 6288988      \_ hdbpreprocessor
 anaadm   12071 11870  0.0 12361768 295608      \_ hdbcompileserver
 anaadm   12101 11870  8.8 20706496 7320808      \_ hdbindexserver
 anaadm   12104 11870  2.7 15484116 2771180      \_ hdbstatisticsserver
 anaadm   12107 11870  2.1 16653840 2314620      \_ hdbxsengine
 anaadm   12394 11870  0.0 271080 72324      \_ sapwebdisp_hdb
pf=/usr/sap/ANA/HDB00/server01/wdisp/sapwebdisp.p
 anaadm   11770     1  0.0 152580 74932 /usr/sap/ANA/HDB00/exe/sapstartsrv
pf=/hana/shared/ANA/profile/ANA_HDB00
 server01:/usr/sap/ANA/HDB00>
```

server11 (secondary database)

```
server11:/ #  su - anaadm
 server11:/usr/sap/ANA/HDB00> sapcontrol -nr 00 -function GetProcessList
 name, description, dispstatus, textstatus, starttime, elapsedtime, pid
 hdbdaemon, HDB Daemon, GREEN, Running, 2015 01 15 03:11:34
 hdbnameserver, HDB Nameserver, GREEN, Running, 2015 01 15 03:11:41
 hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2015 01 15 03:11:55
 hdbindexserver, HDB Indexserver, GREEN, Running, 2015 01 15 03:12:07
 hdbstatisticsserver, HDB Statisticsserver, GREEN, Running, 2015 01 15 03:12:07
 hdbxsengine, HDB XSEngine, GREEN, Running, 2015 01 15 03:12:07
 sapwebdisp_hdb, SAP WebDispatcher, GREEN, Running, 2015 01 15 03:13:57
 hdbcompileserver, HDB Compileserver, GREEN, Running, 2015 01 15 03:11:55

server11:/usr/sap/ANA/HDB00>  HDB info
 USER       PID  PPID %CPU    VSZ   RSS COMMAND
 anaadm   31848 31847  0.3  13884  2772 -sh
 anaadm   31907 31848  0.6  12900  1720  \_ /bin/sh /usr/sap/ANA/HDB00/HDB info
 anaadm   31930 31907 13.3   4944   876      \_ ps fx -U anaadm -o
user,pid,ppid,pcpu,vsz,rss,args
 anaadm   30631     1  0.0  22024  1508 sapstart pf=/hana/shared/ANA/profile/ANA_HDB00_server11
 anaadm   30643 30631  0.0 825796 301416  \_ /usr/sap/ANA/HDB00/server11/trace/hdb.sapANA_HDB00
-d -nw -f /usr/s
 anaadm   30663 30643  1.9 17904260 2165336      \_ hdbnameserver
 anaadm   30867 30643  134 16770920 7402644      \_ hdbpreprocessor
 anaadm   30870 30643  0.1 10406304 293128      \_ hdbcompileserver
 anaadm   30899 30643 21.2 19714576 6921608      \_ hdbindexserver
 anaadm   30902 30643  5.6 16580252 2822236      \_ hdbstatisticsserver
 anaadm   30905 30643  4.5 16402576 2332228      \_ hdbxsengine
 anaadm   31246 30643  0.0 271080 72380      \_ sapwebdisp_hdb
pf=/usr/sap/ANA/HDB00/server11/wdisp/sapwebdisp.p
 anaadm   30541     1  0.0 152580 74936 /usr/sap/ANA/HDB00/exe/sapstartsrv
pf=/hana/shared/ANA/profile/ANA_HDB00
 server11:/usr/sap/ANA/HDB00>
```

Prepare SAP HANA for the cluster user

Set up two-node system replication

Primary node

Enable the primary node. You must start the SAP HANA system.

```
server01:~ # su - anaadm
 server01:/usr/sap/ANA/HDB00> hdbnsutil -sr_enable --name=SJC-Rack-1

Verify the status.
server01:/usr/sap/ANA/HDB00> hdbnsutil -sr_state
checking for active or inactive nameserver ...

System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
online: true

mode: primary
operation mode:
site id: 1
site name: SJC-Rack-1

is source system: true
is secondary/consumer system: false
has secondaries/consumers attached: false
is a takeover active: false

Host Mappings:
...

done.
```

Secondary node

The SAP HANA database instance on the secondary side must be stopped before the instance can be registered for system replication.

Enable the secondary node.

```
server11:/usr/sap/ANA/HDB00> sapcontrol -nr 00 -function StopSystem

 15.01.2015 06:17:03
 StopSystem
 OK
server11:/usr/sap/ANA/HDB00> hdbnsutil -sr_register --remoteHost=server01 --
remoteInstance=00 --replicationMode=sync --operationMode=logreplay --name=SJC-
Rack-2
 adding site ...
```

```
checking for inactive nameserver ...
nameserver server11:30001 not responding.
collecting information ...
updating local ini files ...
done.
```

Start the secondary site.

```
server11:/usr/sap/ANA/HDB00> sapcontrol -nr 00 -function StartSystem

15.01.2015 06:35:17
StartSystem
OK
server11:/usr/sap/ANA/HDB00>
```

Verify the configuration on the primary node.

```
server01:/usr/sap/ANA/HDB00> hdbnsutil -sr_state
checking for active or inactive nameserver ...

System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
online: true

mode: primary
operation mode: primary
site id: 1
site name: SJC-Rack-1

is source system: true
is secondary/consumer system: false
has secondaries/consumers attached: true
is a takeover active: false

Host Mappings:
~~~~~~~~~~~~~~

server01 -> [DC1] server01
server01 -> [DC2] server11


done.
```

Verify the configuration on the secondary node.

```
server11:/usr/sap/ANA/HDB00> hdbnsutil -sr_state
checking for active or inactive nameserver ...
```

```
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
online: true

mode: syncmem
operation mode: logreplay
site id: 2
site name: SJC-Rack-2

is source system: false
is secondary/consumer system: true
has secondaries/consumers attached: false
is a takeover active: false
active primary site: 1


Host Mappings:
~~~~~~~~~~~~~~

server11 -> [DC1] server01
server11 -> [DC2] server11

primary masters:server01

done.
```

Test the system replication status on the primary node.

```
su - anaadm
python /usr/sap/ANA/HDB00/exe/python_support/systemReplicationStatus.py

| Database | Host    | Port  | Service Name | Volume ID | Site ID | Site Name | Secondary |
Secondary | Secondary | Secondary | Secondary     | Replication | Replication | Replication    |
|          |         |       |              |           |         |           | Host      | Port
| Site ID   | Site Name | Active Status | Mode        | Status      | Status Details |
| -------- | ------ | ----- | ------------ | --------- | ------- | --------- | --------- | -----
---- | --------- | --------- | ------------- | ------------ | ------------ | -------------- |
| SYSTEMDB | server01 | 30001 | nameserver   |         1 |       1 | SJC-Rack-1  | server11  |
30001 |         2 | SJC-Rack-2   | YES           | SYNCMEM     | ACTIVE      |                |

status system replication site "2": ACTIVE
overall system replication status: ACTIVE

Local System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

mode: PRIMARY
site id: 1
site name: SJC-Rack-1
```

Set up network for system replication

If nothing is configured, SAP HANA uses the access network to synchronize the systems. The solution in this document uses a separate network for system replication, so you should configure the network.

For more information, see SAP Note 2407186: How-To Guides & Whitepapers For SAP HANA High Availability.

Change the configuration as shown here.

```
server01:/usr/sap/ANA/HDB00> netstat -ia
 Kernel Interface table
 Iface   MTU Met    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
 access  1500   0    17116      0  17085      0        7      0      0      0 BMRU
 appl    9000   0        0      0      0      0        0      0      0      0 BM
 backup  9000   0    17099      0  17085      0        7      0      0      0 BMRU
 datasrc 9000   0    17116      0  17085      0        8      0      0      0 BMRU
 eth0    1500   0 96907431      0  17085      0 115121556      0      0      0 BMRU
 lo     16436   0 48241697      0      0      0 48241697      0      0      0 LRU
 mgmt    1500   0    41529      0  17085      0     2043      0      0      0 BMRU
 nfsd    9000   0  9547248      0  17085      0 10164551      0      0      0 BMRU
 nfsl    9000   0  1088756      0  17087      0  1515500      0      0      0 BMRU
 server  9000   0  1296168      0  17085      0 10105637      0      0      0 BMRU
 sysrep  9000   0    17118      0  17085      0        8      0      0      0 BMRU
 server01:/usr/sap/ANA/HDB00>
```

server01

```
server01:/usr/sap/ANA/HDB00> cdglo
 server01:/usr/sap/ANA/SYS/global> cd hdb/custom/config
 server01:/usr/sap/ANA/SYS/global/hdb/custom/config> cat global.ini
 [persistence]
 basepath_datavolumes = /hana/data/ANA
 basepath_logvolumes = /hana/log/ANA

 [system_replication]
 mode = primary
 actual_mode = primary
 site_id = 1
 site_name = SJC-Rack-1
```

server11

```
server11:/usr/sap/ANA/HDB00> cdglo
server11:/usr/sap/ANA/SYS/global> cd hdb/custom/config/
server11:/usr/sap/ANA/SYS/global/hdb/custom/config> cat global.ini
[persistence]
basepath_datavolumes = /hana/data/ANA
basepath_logvolumes = /hana/log/ANA

[system_replication]
site_id = 2
mode = sync
actual_mode = sync
site_name = SJC-Rack-2

[system_replication_site_masters]
1 = server01:30001
```

Reroute the network traffic from the access network to the system replication network.

server01

```
server01:/usr/sap/ANA/SYS/global/hdb/custom/config> cat global.ini
[persistence]
basepath_datavolumes = /hana/data/ANA
basepath_logvolumes = /hana/log/ANA

[system_replication]
mode = primary
actual_mode = primary
site_id = 1
site_name = SJC-Rack-1

[communication]
listeninterface=.internal

[internal_hostname_resolution]
192.168.220.101 = server01
192.168.220.111 = server11

[system_replication_hostname_resolution]
192.168.222.101 = server01
192.168.222.111 = server11
```

server11

```
server11:/usr/sap/ANA/SYS/global/hdb/custom/config> cat global.ini
 [persistence]
 basepath_datavolumes = /hana/data/ANA
 basepath_logvolumes = /hana/log/ANA

 [system_replication]
 site_id = 2
 mode = sync
 actual_mode = sync
 site_name = SJC-Rack-2

 [system_replication_site_masters]
 1 = server01:30001

 [communication]
 listeninterface=.internal

 [internal_hostname_resolution]
 192.168.220.101 = server01
 192.168.220.111 = server11

 [system_replication_hostname_resolution]
 192.168.222.101 = server01
 192.168.222.111 = server11
```

Restart SAP HANA.

Choose a SAP HANA synchronization mode

Several log replication modes are available to send log information to the secondary instance. You need to decide which mode to use.

- Synchronous (replicationMode=sync): In this mode, the log write operation is considered successful when the log entry has been written to the log volume of the primary and secondary systems. If the connection to the secondary system is lost, the primary system continues transaction processing and writes the changes only to the local disk. No data loss occurs in this scenario as long as the secondary system is connected. Data loss can occur if takeover is performed while the secondary system is disconnected.
- Synchronous in memory (replicationMode=syncmem): In this mode, the log write operation is considered successful when the log entry has been written to the log volume of the primary system and transmission of the log has been acknowledged by the secondary system after the log has been copied to memory. If the connection to the secondary system is lost, the primary system continues transaction processing and writes only the changes to the local disk. Data loss can occur if the primary and secondary systems fail at the same time when the secondary system is connected or takeover is performed when the secondary system is disconnected. This option provides better performance, because it is not necessary to wait for disk I/O on the secondary system, but it is more vulnerable to data loss.
- Asynchronous (replicationMode=async): In this mode, the primary system sends a redo log buffer to the secondary system asynchronously. The primary system commits a transaction when it has been written to the log file of the primary system and sent to the secondary system through the network. It does not wait for confirmation from the secondary system. This option provides better performance because it is not necessary to wait for log I/O on the secondary system. Database consistency across all services on the secondary system is guaranteed. However, this option is more vulnerable to data loss. Data changes may be lost when takeover occurs.

To set up system replication, you need to perform the configuration steps on the secondary system. You can complete this configuration using the hdbnsutil tool, which initializes the topology of the database during installation, or exports, imports, and converts the topology of an existing database. You also can use SAP HANA Studio.

Three operation modes are available:

- Delta Data Shipping (operationMode=delta_datashipping): This mode establishes system replication in which occasionally (by default, every 10 minutes) delta data shipping occurs in addition to continuous log shipping. The shipped redo log is not replayed at the secondary site. During takeover, the redo log needs to be replayed up to the last arrived delta data shipment.
- Log Replay (operationMode=logreplay): In this operation mode, redo log shipping occurs after system replication is initially configured with one full data shipping. The redo log is continuously replayed at the secondary site immediately after arrival, making an addition redo log replay operation superfluous during takeover. This mode does not require delta data shipping, therefore reducing the amount of data that needs to be transferred to the secondary system.
- Log Replay with read access (operationMode=logreplay_readaccess): This mode, also known as active-active read-only mode, is similar to log replay operation mode in its continuous log shipping, the redo log replay on the secondary system, as well as the required initial full data shipping and the takeover. The two modes differ is that in log replay mode with read access, the secondary system is read enabled. By establishing a direct connection to the secondary database or by providing a SELECT statement from the primary database with a hint, read access is possible on the active-active (read enabled) secondary system. For more information, see Client Support for Active-Active (Read Enabled).

Note: The operation mode logreplay_readaccess is not supported for systems with Dynamic Tiering services.

Enable full synchronization for system replication

When activated, the full synchronization (fullsync) option for system replication helps ensure that a log buffer is shipped to the secondary system before a commit operation occurs on the local primary system.

As of SPS08, the fullsync option can be enabled for SYNC replication (that is, not for SYNCMEM). With the fullsync option activated, transaction processing cannot be performed on the primary blocks when the secondary database is currently not connected, and newly created log buffers cannot be shipped to the secondary site. This behavior helps ensure that no transaction can be locally committed without shipping the log buffers to the secondary site.

The fullsync option can be switched on and off using the following command:

```
hdbnsutil -sr_fullsync --enable|--disable
```

The fullsync option is not supported for clusters because it makes no sense in a cluster environment.

Set up SUSE Linux Enterprise High Availability Extension

For more information about SLE HAE, see the SUSE cluster documentation at https://www.suse.com/products/sles-for-sap/resource-library/sap-best-practices/.

Install HAE.

```
zypper in -t pattern ha_sles
zypper in SAPHanaSR SAPHanaSR-doc
```

Make sure that the password and hosts files are identical for the two nodes.

Set up the cluster

Perform basic cluster setup

To set up a cluster, you first need to decide what cluster interconnect networks to use:

- eth0: ring0
- mgmt.: ring1 (optional)

This configuration does not include a shared block device (SBD), so you cannot use an SBD for the cluster. If a small device is available for use as an SBD, then an SBD is preferred.

- NetApp Small Computer System Interface over IP (iSCSI) or Fibre Channel (a separate license is required)
- EMC iSCSI or Fibre Channel

In this case, the Cisco® Integrated Management Controller (IMC) interface is configured to enable STONITH.

To get a detailed overview of all possible settings for the Intelligent Platform Management Interface (IPMI) resource, enter the command shown here.

```
server11:~ # crm ra info stonith:external/ipmi
```

To test the IPMI connection and function, request the power status.

```
server01:~ # ipmitool -I lanplus -H server01-ipmi -U sapadm -P cisco power status
 Chassis Power is on
```

```
server01:~ # ipmitool -I lanplus -H server11-ipmi -U sapadm -P cisco power status
 Chassis Power is on
```

Initialize the cluster

Initialize the cluster configuration.

```
ha-cluster-init
```

This command configures the basic cluster framework, including:

- Secure Shell (SSH) keys
- csync2 to transfer configuration files
- SBD (at least one device); note that the configuration presented here skips this step
- corosync (at least one ring)
- HAWK web interface

The cluster should not be allowed to start during bootup. Make sure that the systemctl setting for the cluster is disabled.

```
server01:~ # systemctl disable corosync
server01:~ # systemctl disable pacemaker
```

Configure the corosync.conf file for User Datagram Protocol (UDP).

```
totem {
        crypto_hash:    none
        rrp_mode:       passive
        token_retransmits_before_loss_const:    10
        join:   60
        max_messages:   20
        vsftype:        none
        token:  5000
        crypto_cipher:  none
        cluster_name:   hacluster
        ip_version:     ipv4
        secauth:        off
        version:        2
        clear_node_high_bit:    yes

        interface {
                bindnetaddr:    192.168.213.0
                mcastport:      5405
                ringnumber:     0
                ttl:    1
        }
        interface {
                bindnetaddr:    172.21.0.0
                mcastport:      5406
                ringnumber:     1
```

```
        }

        consensus:      6000
        transport:      udpu
}
nodelist {
        node {
        ring0_addr:     192.168.213.126
        ring1_addr:     172.21.0.126
        }
        node {
        ring0_addr:     192.168.213.127
        ring1_addr:     172.21.0.127
        }
}
logging {
        to_logfile:     no
        logfile:        /var/log/cluster/corosync.log
        timestamp:      on
        syslog_facility:        daemon
        logger_subsys {
                debug:  off
                subsys: AMF
        }
        to_syslog:      yes
        debug:  off
        to_stderr:      no
        fileline:       off
}
quorum {
        two_node:       0
        provider:       corosync_votequorum
}
```

Configure csync2 to keep the cluster configuration synchronized.

To initially synchronize all files once, enter the following command on the device from which you want to copy the configuration:

```
root # csync2 -xv
```

This command synchronizes all the files once by pushing them to the other nodes. If all files are synchronized successfully, csync2 will finish with no errors. If one or several files that are to be synchronized have been modified on other nodes (not only on the current node), csync2 reports a conflict. You will see output similar to the following:

```
While syncing file /etc/corosync/corosync.conf:
ERROR from peer hex-14: File is also marked dirty here!
Finished with 1 errors.
```

If you are sure that the file version on the current node is the “best” one, you can resolve the conflict by forcing this file and resynchronizing:

```
root # csync2 -f /etc/corosync/corosync.conf
root # csync2 -x
```

The second node joins the cluster.

```
server11:~ # ha-cluster-join
```

Restart the cluster on both nodes.

```
server01:~ # systemctl stop pacemaker

server11:~ # systemctl stop pacemaker

server01:~ # systemctl start pacemaker

server11:~ # systemctl start pacemaker
```

The actual status of the cluster is reported.

```
server01:~ # crm_mon -1 -r
 Last updated: Mon Jan 26 01:27:41 2015
 Last change: Mon Jan 26 01:20:10 2015 by root via crm_attribute on server11
 Stack: classic openais (with plugin)
 Current DC: server11 - partition with quorum
 Version: 1.1.11-3ca8c3b
 2 Nodes configured, 2 expected votes
 0 Resources configured


 Online: [ server01 server11 ]

 Full list of resources:
```

Test the basic cluster function and the cluster information base database

Test the cluster ring status.

```
server01:~ # corosync-cfgtool -s
 Printing ring status.
 Local node ID 1084784485
RING ID 0
        id      = 192.168.213.126
        status  = ring 0 active with no faults
RING ID 1
        id      = 172.21.0.126
        status  = ring 1 active with no faults
```

```
server11:~  # corosync-cfgtool -s
 Printing ring status.
 Local node ID 1084784495
RING ID 0
        id      = 192.168.213.127
        status  = ring 0 active with no faults
RING ID 1
        id      = 172.21.0.127
        status  = ring 1 active with no faults
```

Test the cluster information database (CIB) consistency.

```
server01:~ # crm_verify -LV
server11:~ # crm_verify -LV
```

Configure STONITH IPMI

Configure STONITH IPMI as shown here.

```
server01:~ # crm configure
crm(live)configure# primitive STONITH-Server01 stonith:external/ipmi op monitor interval="0"
timeout="60s" op monitor interval="300s" timeout="60s" on-fail="restart" op start interval="0"
timeout="60s" on-fail="restart" params hostname="server01" ipaddr="server01-ipmi"
userid="sapadm" passwd="cisco" interface="lanplus"


crm(live)configure# primitive STONITH-Server11 stonith:external/ipmi op monitor interval="0"
timeout="60s" op monitor interval="300s" timeout="60s" on-fail="restart" op start interval="0"
timeout="60s" on-fail="restart" params hostname="server11" ipaddr="server11-ipmi"
userid="sapadm" passwd="cisco" interface="lanplus"


crm(live)configure# location LOC_STONITH_Server01 STONITH-Server01 inf: server11
crm(live)configure# location LOC_STONITH_Server11 STONITH-Server11 inf: server01


 crm(live)#
```

With **inf:server0x**, you specify that the server runs only on this node. For IPMI, the service must always run on the other node: for instance, STONITH-Server01 must run on server11.

```
 server01:~ # crm configure property no-quorum-policy="ignore"
 server01:~ # crm configure property stonith-action="reboot"
 server01:~ # crm configure property startup-fencing="false"
 server01:~ # crm configure property stonith-timeout="30s"
```

Restart the cluster.

The result is shown here.

```
server01:~ # crm_mon -1 -r
 Last updated: Mon Jan 26 23:32:46 2015
 Last change: Mon Jan 26 23:26:52 2015 by root via cibadmin on server11
 Stack: classic openais (with plugin)
```

```
Current DC: server01 - partition with quorum
Version: 1.1.11-3ca8c3b
2 Nodes configured, 2 expected votes
2 Resources configured


Online: [ server01 server11 ]

Full list of resources:

 STONITH-Server01       (stonith:external/ipmi):        Started server11
 STONITH-Server11       (stonith:external/ipmi):        Started server01
server01:~ #
```

Configure the basic settings for the cluster

Configure the basic settings for the cluster as shown here.

```
server01:~ # vi crm-base.txt

property $id="cib-bootstrap-options" \
no-quorum-policy="ignore" \
stonith-enabled="true" \
stonith-action="reboot" \
startup-fencing="true" \
stonith-timeout="150s" \
rsc_defaults $id="rsc-options" \
resource-stickiness="1000" \
migration-threshold="5000"
op_defaults $id="op-options" \
timeout="600"
```

For testing purposes, startup-fencing can be set to false. After the cluster function has been verified, this option should be set to true.

```
server01:~ # crm configure load update crm-base.txt
```

Install the SAP HANA–specific properties.

```
server01:~ #  vi saphanatop.txt
primitive rsc_SAPHanaTopology_ANA_HDB00 ocf:suse:SAPHanaTopology \
operations $id="rsc_sap2_ANA_HDB00-operations" \
op monitor interval="10" timeout="600" \
op start interval="0" timeout="600" \
op stop interval="0" timeout="300" \
params SID="ANA" InstanceNumber="00"
clone cln_SAPHanaTopology_ANA_HDB00 rsc_SAPHanaTopology_ANA_HDB00 \
meta clone-node-max="1" interleave="true"
  server01:~ # crm configure load update saphanatop.txt
```

Configure the SAP HANA–specific parameters.

```
server01:~ # vi saphana.txt
primitive rsc_SAPHana_ANA_HDB00 ocf:suse:SAPHana \
operations $id="rsc_sap_ANA_HDB00-operations" \
op start interval="0" timeout="3600" \
op stop interval="0" timeout="3600" \
op promote interval="0" timeout="3600" \
op monitor interval="60" role="Master" timeout="700" \
op monitor interval="61" role="Slave" timeout="700" \
params SID="ANA" InstanceNumber="00" PREFER_SITE_TAKEOVER="true" \
DUPLICATE_PRIMARY_TIMEOUT="7200" AUTOMATED_REGISTER="false"
ms msl_SAPHana_ANA_HDB00 rsc_SAPHana_ANA_HDB00 \
meta clone-max="2" clone-node-max="1" interleave="true"
  server01:~ # crm configure load update saphana.txt
```

Configure the virtual IP address of the customer access LAN or application LAN.

```
server01:~ # vi crm-ip.txt
primitive rsc_ip_ANA_HDB00 ocf:heartbeat:IPaddr2 \
operations $id="rsc_ip_ANA_HDB00-operations" \
op monitor interval="10s" timeout="20s" \
params ip="192.168.220.200"
  server01:~ # crm configure load update crm-ip.txt
```

Configure the constraints.

server01:~ **# vi crm-constraints.txt**

colocation col_saphana_ip_ANA_HDB00 2000: rsc_ip_ANA_HDB00:Started \

msl_SAPHana_ANA_HDB00:Master

order ord_SAPHana_ANA_HDB00 Optional: cln_SAPHanaTopology_ANA_HDB00 \

msl_SAPHana_ANA_HDB00

server01:~ # **crm configure load update crm-constraints.txt**

The result is shown here.

```
Server01:/ # /usr/share/SAPHanaSR/tests/show_SAPHanaSR_attributes
 Host \ Attr clone_state remoteHost roles site srmode sync_state vhost  lpa_ana_lpt
-----------------------------------------------------------------------------
 server01 PROMOTED server11 4:P:master1:master:worker:master SJC-Rack-1 sync PRIM       server01
1422591580
 server11 DEMOTED server01 4:S:master1:master:worker:master SJC-Rack-2 sync  SOK        server11
30
 server11:/usr/sap/hostctrl/exe #

server01:/ # crm_mon -r -1
```

```
 Last updated: Thu Jan 29 20:23:10 2015
 Last change: Thu Jan 29 20:21:46 2015 by root via crm_attribute on server01
 Stack: classic openais (with plugin)
 Current DC: server11 - partition with quorum
 Version: 1.1.11-3ca8c3b
 2 Nodes configured, 2 expected votes
 7 Resources configured

 Online: [ server01 server11 ]

 Full list of resources:

  STONITH-Server01       (stonith:external/ipmi):        Started server01
  STONITH-Server11       (stonith:external/ipmi):        Started server11
  Clone Set: cln_SAPHanaTopology_ANA_HDB00 [rsc_SAPHanaTopology_ANA_HDB00]
      Started: [ server01 server11 ]
  Master/Slave Set: msl_SAPHana_ANA_HDB00 [rsc_SAPHana_ANA_HDB00]
      Masters: [ server01 ]
      Slaves: [ server11 ]
  rsc_ip_ANA_HDB00       (ocf::heartbeat:IPaddr2):       Started server01

server01:/ #
```

Configure the cluster resource manager

Configure the cluster resource manager (CRM) as shown here.

```
server01:~ # crm configure show
node 1084806526: server01 \
        attributes lpa_ana_lpt=10 hana_ana_vhost=server01 hana_ana_srmode=syncmem
hana_ana_remoteHost=server02 hana_ana_site=DC1 hana_ana_op_mode=logreplay
node 1084806527: server02 \
        attributes lpa_ana_lpt=1496409145 hana_ana_vhost=server02 hana_ana_remoteHost=server01
hana_ana_site=DC2 hana_ana_srmode=syncmem hana_ana_op_mode=logreplay
primitive rsc_SAPHanaTopology_ANA_HDB00 ocf:suse:SAPHanaTopology \
        operations $id=rsc_sap2_ANA_HDB00-operations \
        op monitor interval=10 timeout=600 \
        op start interval=0 timeout=600 \
        op stop interval=0 timeout=300 \
        params SID=ANA InstanceNumber=00
primitive rsc_SAPHana_ANA_HDB00 ocf:suse:SAPHana \
        operations $id=rsc_sap_ANA_HDB00-operations \
        op start interval=0 timeout=3600 \
        op stop interval=0 timeout=3600 \
        op promote interval=0 timeout=3600 \
        op monitor interval=60 role=Master timeout=700 \
        op monitor interval=61 role=Slave timeout=700 \
        params SID=ANA InstanceNumber=00 PREFER_SITE_TAKEOVER=true
DUPLICATE_PRIMARY_TIMEOUT=7200 AUTOMATED_REGISTER=.false.
primitive rsc_server01_stonith stonith:external/ipmi \
        params hostname=server01 ipaddr=172.21.1.247 userid=sapadm passwd=cisco
interface=lanplus \
        op monitor interval=1800 timeout=30
primitive rsc_server02_stonith stonith:external/ipmi \
        params hostname=server02 ipaddr=172.21.1.237 userid=sapadm passwd=cisco
interface=lanplus \
        op monitor interval=1800 timeout=30
primitive rsc_ip_ANA_HDB00 IPaddr2 \
        meta target-role=Started is-managed=true \
        operations $id=rsc_ip_ANA_HDB00-operations \
        op monitor interval=10s timeout=20s \
        params ip=172.21.0.131
ms msl_SAPHana_ANA_HDB00 rsc_SAPHana_ANA_HDB00 \
        meta is-managed=true notify=true clone-max=2 clone-node-max=1 target-role=Started
clone cln_SAPHanaTopology_ANA_HDB00 rsc_SAPHanaTopology_ANA_HDB00 \
        meta is-managed=true clone-node-max=1 target-role=Started
colocation col_saphana_ip_ANA_HDB00 2000: rsc_ip_ANA_HDB00:Started msl_SAPHana_ANA_HDB00:Master
location loc_server01_stonith rsc_server01_stonith -inf: server01
location loc_server02_stonith rsc_server02_stonith -inf: server02
order ord_SAPHana_ANA_HDB00 Optional: cln_SAPHanaTopology_ANA_HDB00 msl_SAPHana_ANA_HDB00
```

```
property cib-bootstrap-options: \
        have-watchdog=false \
        dc-version=1.1.15-21.1-e174ec8 \
        cluster-infrastructure=corosync \
        cluster-name=hacluster \
        stonith-enabled=true \
        placement-strategy=balanced \
        no-quorum-policy=ignore \
        stonith-action=reboot \
        stonith-timeout=30s \
        rsc_defaults \
        id=rsc-options \
        resource-stickiness=1000 \
        migration-threshold=5000 \
        last-lrm-refresh=1496234562
rsc_defaults rsc-options: \
        resource-stickiness=1 \
        migration-threshold=3
op_defaults op-options: \
        timeout=600 \
        record-pending=true
```

In some cases, instance monitoring may return errors as shown here.

```
Failed actions:
     rsc_SAPHana_ANA_HDB00_monitor_60000 on server11 'ok' (0): call=47, status=complete, last-
rc-change='Thu Jan 29 20:11:35 2015', queued=0ms, exec=27433ms
     rsc_SAPHana_ANA_HDB00_monitor_61000 on server01 'not running' (7): call=31,
status=complete, last-rc-change='Thu Jan 29 20:07:34 2015', queued=0ms, exec=0ms
```

If errors occur, tune the op monitor parameter at the primitive rsc_SAPHana_ANA_HDB00.

```
primitive rsc_SAPHana_ANA_HDB00 ocf:suse:SAPHana \
         operations $id="rsc_sap_ANA_HDB00-operations" \
         op start interval="0" timeout="3600" \
         op stop interval="0" timeout="3600" \
         op promote interval="0" timeout="3600" \
         op monitor interval="60" role="Master" timeout="1400" \
         op monitor interval="61" role="Slave" timeout="1400" \
   params SID="ANA" InstanceNumber="00" PREFER_SITE_TAKEOVER="true"
DUPLICATE_PRIMARY_TIMEOUT="7200" AUTOMATED_REGISTER="false"
```

Use 1400 or higher for the value.

Perform cluster switchback after a failure

After the cluster detects an error at the primary site, the cluster activates the secondary site automatically.

The cluster cannot switch back the SAP HANA database after the primary system is repaired. This process must be performed manually, as described in the following sections.

The results after cluster switchover are shown here:

- Server01: SAP HANA is down.
- Server11: SAP HANA active.

```
server01:~ # crm_mon -1 -r
 Last updated: Wed Mar 25 14:50:39 2015
 Last change: Wed Mar 25 14:49:45 2015 by root via crm_attribute on server11
 Stack: classic openais (with plugin)
 Current DC: server11 - partition with quorum
 Version: 1.1.11-3ca8c3b
 2 Nodes configured, 2 expected votes
 7 Resources configured


 Online: [ server01 server11 ]

 Full list of resources:

  STONITH-Server01       (stonith:external/ipmi):        Started server11
  STONITH-Server11       (stonith:external/ipmi):        Started server01
  Clone Set: cln_SAPHanaTopology_ANA_HDB00 [rsc_SAPHanaTopology_ANA_HDB00]
      Started: [ server01 server11 ]
  Master/Slave Set: msl_SAPHana_ANA_HDB00 [rsc_SAPHana_ANA_HDB00]
      Masters: [ server11 ]
      Slaves: [ server01 ]
  rsc_ip_ANA_HDB00       (ocf::heartbeat:IPaddr2):       Started server11

 Failed actions:
     rsc_SAPHana_ANA_HDB00_monitor_61000 on server01 'not running' (7): call=46,
status=complete, last-rc-change='Tue Mar 24 15:53:16 2015', queued=0ms, exec=20239ms


 server01:~ #
```

Resynchronize the primary database

The primary SAP HANA database is down.

```
server01:~ # su - anaadm
 server01:/usr/sap/ANA/HDB00> HDB info
 USER       PID  PPID %CPU    VSZ   RSS COMMAND
 anaadm   24542 24541 12.3 7857084 144916 hdbnsutil -sr_state --sapcontrol=1
 anaadm   24646 24542  3.8   3396   192  \_ /bin/sh -c PATH="/bin:/sbin:/usr/bin:/usr/sbin:/etc"
sysctl -n "kernel.shmall"
 anaadm   24392 24391  0.1  13884  2740 -sh
 anaadm   24592 24392  0.3  12900  1720  \_ /bin/sh /usr/sap/ANA/HDB00/HDB info
 anaadm   24662 24592  5.7   4944   876      \_ ps fx -U anaadm -o
user,pid,ppid,pcpu,vsz,rss,args
 anaadm    9840     1  0.0 218884 75584 /usr/sap/ANA/HDB00/exe/sapstartsrv
pf=/usr/sap/ANA/SYS/profile/ANA_HDB00_server01 -D -u anaadm
server01:/usr/sap/ANA/HDB00>
```

Enable resynchronization (now server01 becomes secondary).

```
server01:~> hdbnsutil -sr_register --remoteHost=server11 --remoteInstance=00 --mode=sync --
name=SJC-Rack-1
 adding site ...
 checking for inactive nameserver ...
 nameserver server01:30001 not responding.
 collecting information ...
 registered at 192.168.222.111 (server11)
 updating local ini files ...
 done.
```

Start the database now.

```
server01:~> sapcontrol -nr 00 -function StartSystem
 25.03.2015 15:42:40
 StartSystem
 OK

server01:~> HDB info
 USER       PID  PPID %CPU    VSZ   RSS COMMAND
 anaadm   39342 39341  5.1  50740  4804 python exe/python_support/landscapeHostConfiguration.py
 anaadm   39211 39210 14.4 1079000 104080 python
/usr/sap/ANA/HDB00/exe/python_support/landscapeHostConfiguration.py
 anaadm   39402 39211  3.1    300     4  \_ sysctl -n kernel.shmall
 anaadm   24392 24391  0.0  13884  2780 -sh
 anaadm   39261 24392  0.3  12900  1720  \_ /bin/sh /usr/sap/ANA/HDB00/HDB info
 anaadm   39404 39261  5.5   4944   876      \_ ps fx -U anaadm -o
user,pid,ppid,pcpu,vsz,rss,args
 anaadm   33980     1  0.0  22024  1512 sapstart pf=/usr/sap/ANA/SYS/profile/ANA_HDB00_server01
```

```
 anaadm   34060 33980  0.0 825832 196608  \_ /usr/sap/ANA/HDB00/server01/trace/hdb.sapANA_HDB00
-d -nw -f /usr/sap/ANA/HDB00/server01/daemon.ini pf=/usr/sap/ANA/SYS/profile/ANA_HDB00_server01
 anaadm   34145 34060  1.4 13646420 1079384      \_ hdbnameserver
 anaadm   34290 34060  0.6 9653164 278408      \_ hdbpreprocessor
 anaadm   34295 34060  0.4 9460648 267816      \_ hdbcompileserver
 anaadm   34441 34060  2.3 10835544 1776224      \_ hdbindexserver
 anaadm   34444 34060  2.1 9828516 1481288      \_ hdbstatisticsserver
 anaadm   34447 34060  1.3 12396176 715808      \_ hdbxsengine
 anaadm   34652 34060  0.1 205464 33000      \_ sapwebdisp_hdb
pf=/usr/sap/ANA/HDB00/server01/wdisp/sapwebdisp.pfl -f
/usr/sap/ANA/HDB00/server01/trace/dev_webdisp
 anaadm    9840     1  0.0 219400 75704 /usr/sap/ANA/HDB00/exe/sapstartsrv
pf=/usr/sap/ANA/SYS/profile/ANA_HDB00_server01 -D -u anaadm
 server01:~>
```

Clean up the cluster status.

```
server01:~ # crm resource cleanup msl_SAPHana_ANA_HDB00
 Cleaning up rsc_SAPHana_ANA_HDB00:0 on server01
 Cleaning up rsc_SAPHana_ANA_HDB00:0 on server11
 Cleaning up rsc_SAPHana_ANA_HDB00:1 on server01
 Cleaning up rsc_SAPHana_ANA_HDB00:1 on server11
 Waiting for 4 replies from the CRMd.... OK
 server01:~ #

server01:~ # crm_mon -1 -r
 Last updated: Wed Mar 25 15:58:36 2015
 Last change: Wed Mar 25 15:58:01 2015 by hacluster via crmd on server01
 Stack: classic openais (with plugin)
 Current DC: server11 - partition with quorum
 Version: 1.1.11-3ca8c3b
 2 Nodes configured, 2 expected votes
 7 Resources configured

 Online: [ server01 server11 ]

 Full list of resources:

  STONITH-Server01       (stonith:external/ipmi):        Started server11
  STONITH-Server11       (stonith:external/ipmi):        Started server01
  Clone Set: cln_SAPHanaTopology_ANA_HDB00 [rsc_SAPHanaTopology_ANA_HDB00]
      Started: [ server01 server11 ]
  Master/Slave Set: msl_SAPHana_ANA_HDB00 [rsc_SAPHana_ANA_HDB00]
      Masters: [ server11 ]
      Slaves: [ server01 ]
  rsc_ip_ANA_HDB00       (ocf::heartbeat:IPaddr2):       Started server11
 server01:~ #
```

Activate the original master as the primary database.

Switchover the IP address to server01.

```
server01:~ # crm resource migrate rsc_ip_ANA_HDB00 server01
```

Switchover the database.

```
server01:~ # su - anaadm
 server01:/usr/sap/ANA/HDB00> hdbnsutil -sr_takeover
 checking local nameserver ...

 done.
 server01:/usr/sap/ANA/HDB00>
```

The source database (server11) will be shut down now by the takeover process.

Re-enable the secondary site

Use the following configuration to re-enable the secondary site.

```
server11:/var/log # su - anaadm
 server11:/usr/sap/ANA/HDB00> hdbnsutil -sr_register --remoteHost=server01 --remoteInstance=00 --mode=sync --name=SJC-Rack-2
 adding site ...
 checking for inactive nameserver ...
 nameserver server11:30001 not responding.
 collecting information ...
 registered at 192.168.222.101 (server01)
 updating local ini files ...
 done.
 server11:/usr/sap/ANA/HDB00> sapcontrol -nr 00 -function StartSystem
 25.03.2015 16:41:23
 StartSystem
 OK
 server11:/usr/sap/ANA/HDB00> exit
```

Clean up the cluster resources.

```
server11:/var/log # crm resource cleanup msl_SAPHana_ANA_HDB00
 Cleaning up rsc_SAPHana_ANA_HDB00:0 on server01
 Cleaning up rsc_SAPHana_ANA_HDB00:0 on server11
 Cleaning up rsc_SAPHana_ANA_HDB00:1 on server01
 Cleaning up rsc_SAPHana_ANA_HDB00:1 on server11
 Waiting for 4 replies from the CRMd.... OK
server11:/var/log # crm_mon -1 -r
 Last updated: Wed Mar 25 16:46:12 2015
 Last change: Wed Mar 25 16:45:04 2015 by root via crm_attribute on server01
 Stack: classic openais (with plugin)
 Current DC: server11 - partition with quorum
 Version: 1.1.11-3ca8c3b
 2 Nodes configured, 2 expected votes
 7 Resources configured
 Online: [ server01 server11 ]
 Full list of resources:
  STONITH-Server01       (stonith:external/ipmi):        Started server11
  STONITH-Server11       (stonith:external/ipmi):        Started server01
  Clone Set: cln_SAPHanaTopology_ANA_HDB00 [rsc_SAPHanaTopology_ANA_HDB00]
      Started: [ server01 server11 ]
  Master/Slave Set: msl_SAPHana_ANA_HDB00 [rsc_SAPHana_ANA_HDB00]
      Masters: [ server01 ]
      Slaves: [ server11 ]
  rsc_ip_ANA_HDB00       (ocf::heartbeat:IPaddr2):       Started server01
```

Disable system replication

Before disabling system replication, verify that it is enabled.

```
hdbsql -U slehaloc 'select distinct REPLICATION_STATUS from SYS.M_SERVICE_REPLICATION'

REPLICATION_STATUS
 "ACTIVE"
```

Now stop the secondary site.

server11

```
sapcontrol -nr 00 -function StopSystem

25.01.2015 22:26:43
StopSystem
OK
```

server01

```
hdbsql -U slehaloc 'select distinct REPLICATION_STATUS from
SYS.M_SERVICE_REPLICATION'

REPLICATION_STATUS
 "ERROR"
```

server11

```
server11:/ > hdbnsutil -sr_unregister
 unregistering site ...
 nameserver server11:30001 not responding.
 nameserver server11:30001 not responding.
 checking for inactive nameserver ...
 nameserver server11:30001 not responding.
 nameserver is shut down, proceeding ...
 opening persistence ...
 run as transaction master
 updating topology for system replication takeover ...
 mapped host server01 to server11
 sending unregister request to primary site (2) ...
##########################################################################################
# CAUTION: You must start the database in order to complete the unregistration!
##########################################################################################

 done.
 server11:/usr/sap/ANA/SYS/global/hdb/custom/config>
```

server01

```
server01:/ > hdbnsutil -sr_disable
 checking local nameserver:
 checking for inactive nameserver ...
 nameserver is running, proceeding ...
 done.
 server01:/ > hdbnsutil -sr_state
 checking for active or inactive nameserver ...

 System Replication State
 ~~~~~~~~~~~~~~~~~~~~~~~~

 mode: none

 done.
 server01:/usr/sap/ANA/SYS/global/hdb/custom/config>
```

Configure the IPMI watchdog timer

This section describes how to configure the IPMI watchdog timer in an SLES12 SP2 environment. For a Linux cluster setup with certain fencing methods, you should use a watchdog timer that can directly interact with the baseboard management controller (BMC) of a server to reset the system in the event of a kernel hang or failure.

Note: The cluster setup described previously in this document uses an IPMI STONITH method. Therefore, you don't need to activate the IPMI watchdog timer. The IPMI watchdog timer is required for other fencing mechanisms: for example, SBD node fencing.

Baseboard management controller

The BMC, also called the service processor, resides on each blade to allow OS independence and pre-OS management. The focus of the BMC is on monitoring and managing a single blade in the Cisco UCS chassis. Cisco UCS Manager talks to the BMC and configures the blade. The BMC controls processes such as BIOS upgrade and downgrade, assignment of MAC addresses, and assignment of worldwide names (WWNs).

The BMC runs independent of the processor on the blade. Thus, in the event of a processor, memory, or other hardware failure, the BMC will still provide services. The BMC starts running as soon as the blade is inserted into the chassis. The power button on the front of the blade does not turn the BMC on and off. The BMC is connected through two 100BASE-T interfaces on the blade to the chassis internal management. It is not connected through the network interface cards (NICs) or the mezzanine cards used by the blade to send traffic. However, at any given time, the BMC is connected only through one of the two 100BASE-T interfaces, and the other one is redundant.

The BMC maintains two IP instances per 100BASE-T Ethernet interface. One IP instance connects to the chassis left and right infrastructure VLAN for communication with the chassis management controller (CMC). The other IP instance connects to the fabric A and B infrastructure VLAN for communication with endpoints external to the chassis. Thus, the BMC has a total of four IP addresses on two physical Ethernet interfaces. The BMC firmware is stored on separate flash chips, and multiple versions of firmware can reside on the flash chip at the same time. The BMC gets an external IP address from a pool created through Cisco UCS Manager. This IP address is then used to access BMC services such as Kernel-based Virtual Machine (KVM). The BMC communicates with the BIOS running on the blade using IPMI.

BMC integration into Cisco UCS

The BMC for Cisco UCS is tightly integrated with Cisco UCS Manager and uses a policy-based approach to configure, patch, and update its associated storage, network, and server resources. The BMC communicates with Cisco UCS Manager through a set of XML interfaces that directly expose the hardware service profiles from Cisco UCS Manager. It integrates with and builds on Cisco UCS Manager to automate the full server stack provisioning of Cisco UCS, including the virtual hardware and network resources, base OS, drivers, disk layouts, system software, application middleware, and custom code and content at the business-application layer. It also provisions supporting configurations specific to Cisco UCS as well as any user-defined custom configurations.

For more information, see https://supportforums.cisco.com/document/29806/what-bmc-and-what-are-tasks-bmc.

Set up the IPMI watchdog timer

By default, the Linux watchdog timer installed during the installation is the Intel TCO (iTCO) watchdog timer, which is not supported by Cisco UCS or by the Cisco C880 M4 Server. Therefore, this watchdog must be disabled.

The status information when the iTCO watchdog timer is loaded on Cisco UCS and the C880 server is shown here. This timer does not work in this scenario.

```
server01:~ # ipmitool mc watchdog get
Watchdog Timer Use:     BIOS FRB2 (0x01)
Watchdog Timer Is:      Stopped
Watchdog Timer Actions: No action (0x00)
Pre-timeout interval:   0 seconds
Timer Expiration Flags: 0x00
Initial Countdown:      0 sec
Present Countdown:      0 sec
```

The default device /dev/watchdog will not be created.

As shown here, no watchdog device was created.

```
server01:~ # ls -l /dev/watchdog
ls: cannot access /dev/watchdog: No such file or directory
```

As shown here, the wrong watchdog timer is installed and loaded on the system.

```
server01:~ # lsmod |grep iTCO
iTCO_wdt               13480  0
iTCO_vendor_support    13718  1 iTCO_wdt
```

Unload the incorrect driver from the environment.

```
server01:~ # modprobe -r iTCO_wdt iTCO_vendor_support
```

To make sure that the driver is not loaded during the next system boot, the driver must be blacklisted.

To blacklist the iTCO modules, add them to the blacklist file.

```
server01:~ # vi /etc/modprobe.d/50-blacklist.conf
...
# unload the iTCO watchdog modules
blacklist iTCO_wdt
blacklist iTCO_vendor_support
```

Now install and configure the supported watchdog timer.

Install and update the IPMI packages.

```
server01:~ # zypper update OpenIPMI ipmitool
```

Tested package version are:

For SLES12SP1

```
OpenIPMI-2.0.21-2.13.x86_64
ipmitool-1.8.13-9.1.x86_64
```

Configure the IPMI watchdog.

```
server01:~ # mv /etc/sysconfig/ipmi /etc/sysconfig/ipmi.org
server01:~ # vi /etc/sysconfig/ipmi

IPMI_SI=yes
DEV_IPMI=yes
IPMI_WATCHDOG=yes
IPMI_WATCHDOG_OPTIONS="timeout=20 action=reset nowayout=0 panic_wdt_timeout=15"
IPMI_POWEROFF=no
IPMI_POWERCYCLE=no
IPMI_IMB=no
```

If a kernel core dump capture is active, you need to set panic_wdt_timeout to a higher value. Otherwise, the core dump doesn't have enough time to be written to the disk.

Enable the system BMC watchdog.

```
server01:~ # vi /etc/systemd/system.conf
...
RuntimeWatchdogSec=10
ShutdownWatchdogSec=60
...
```

Restart the systemd daemon.

```
server01:~ # systemctl daemon-reexec
```

Enable the IPMI watchdog during boot time.

```
server01:~ # systemctl enable ipmi
server01:~ # systemctl start ipmi
```

Here are the results after the IPMI watchdog starts.

```
server01:~ # lsmod |grep ipmi
ipmi_watchdog          24912  1
ipmi_devintf           17572  0
ipmi_si                57482  1
ipmi_msghandler        49676  3 ipmi_devintf,ipmi_watchdog,ipmi_si
```

Verify that the watchdog is active.

```
server01:~ # ipmitool mc watchdog get
Watchdog Timer Use:     SMS/OS (0x44)
```

```
Watchdog Timer Is:      Started/Running
Watchdog Timer Actions: Hard Reset (0x01)
Pre-timeout interval:   0 seconds
Timer Expiration Flags: 0x10
Initial Countdown:      10 sec
Present Countdown:      6 sec
```

Now test the watchdog configuration.

Trigger a kernel crash.

```
server01:~ # echo c > /proc/sysrq-trigger
```

The system must reboot after five minutes (BMC timeout) or after the time set in **panic_wdt_timeout** in the /etc/sysconfig/ipmi config file.

For more information

For additional information, see:

- Introduction to SAP HANA High Availability
- How to Perform System Replication for SAP HANA
- SUSE Best Practices

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-739685-00 10/17