Providing High Availability for SAP Resources with Oracle Clusterware 11 Release 2

An Oracle White Paper
January 2013

Document Version 7.0
Providing High Availability for SAP Resources

Overview of High Availability for SAP Resources .................................................... 3
New functionality ....................................................................................................... 3
SAP Support for High Availability ............................................................................. 5
Installation and Management ...................................................................................... 8
Overview Of Installation and Configuration ............................................................... 9
Functionality ............................................................................................................. 13
Conclusion ................................................................................................................ 16
Worked Example ...................................................................................................... 17
Appendix 1 – Sample profile scripts ....................................................................... 22
Appendix 2 – Troubleshooting and Log Files ........................................................... 24
Appendix 3 – SAPCTL Bill of Materials ................................................................... 25
Appendix 4 – CRS resources and types ................................................................... 25
Appendix 5 – Multiple application VIP’s for SAP resources ................................. 26
Providing High Availability for SAP Resources

OVERVIEW OF HIGH AVAILABILITY FOR SAP RESOURCES

Oracle Clusterware can provide high availability for SAP resources just as it does for Oracle resources. Oracle has created an Oracle Clusterware tool, SAP Control (SAPCTL), to enable you to easily manage SAP high availability resources.

SAPCTL provides an easy-to-use interface to administer the resources, scripts, and dependencies of Oracle Clusterware and SAP high availability components. SAPCTL consolidates the functionality of the Oracle command-line tools by enabling you to easily manage the SAP Enqueue Service for ABAP and JAVA, the SAP Replication Service for ABAP and JAVA, and the additional virtual IP addresses used by the SAP Enqueue Service for ABAP and/or JAVA.

NEW FUNCTIONALITY

An interface for SAP HA monitoring is implemented. Start and stop of SAP instances protected by SAPCTL can now be controlled by SAP monitoring tools, e.g. SAP MMC.

In addition to the critical SAP high availability components, namely the SAP Enqueue and SAP Replication Service, SAPCTL Version 6 provides an interface for the protection of arbitrary number of SAP application instances.

The SAP Central Instance (CI) or SAP application instances (DV) are possible candidates to run under SAPCTL supervision.

The SAPCTL tool supports SAP Standalone Gateway (GW) and SAP WebDispatcher (W) as independent SAP instances as well.

For all supported SAP instance types, SAPCTL in addition monitors the associated SAP Start Service (sapstartsrv daemon process) with an always co-located CRS resource. The SAP Start Service process is started automatically by SAPCTL if the SAP instance gets started on a cluster node or relocated to another node in the cluster. There is no need to bring up all SAP Start Processes on system boot by sapinit procedure as this task is performed by SAPCTL after startup of Oracle Clusterware.

Multiple networks for the application VIPs are supported. All defined application VIPs for a SAP instance are always co-located on the same node in the cluster. If an instance must be relocated to another node, all application VIPs for the SAP instance are relocated to this node as well. Configuration of additional application VIPs is not done automatically during resource creation and need to be defined by manual steps. Templates are provided together with the SAPCTL scripts. See appendix 4 for more information.

If SAP executables are located on an Oracle ACFS filesystem, this is detected by SAPCTL and the resource dependencies are automatically created.

SAPCTL is available for the LINUX, Solaris and AIX operating systems.

Oracle Clusterware 11g Release 2 is required for this version of SAPCTL.

All SAP kernels starting with kernel version 7.00 are supported by SAPCTL.

The Oracle Exadata Database Machine is certified to run SAP Central Services (SCS and ASCS instances and the ER instances) and SAPCTL supports these services when running on Oracle Exadata. For information on SAP components and configuration for

SAPCTL can also be used for the Central Services on the Oracle SPARC SuperCluster and the Oracle Database Appliance.
SAP SUPPORT FOR HIGH AVAILABILITY

The SAP Enqueue Service provides distributed lock management for SAP application server instances. The SAP Enqueue Service must be available whenever an SAP installation uses more than one SAP instance. In order to support High Availability, the Enqueue Service together with the Message Service builds up the “Standalone Enqueue Service” (ASCS) for SAP WebAS ABAP. In case of SAP WebAS JAVA, the Enqueue Service together with the Message Service forms the “Central Services Instance” (SCS). This SCS instance is always required for the JAVA stack of SAP.

SAP uses an active/passive approach to enable high availability for the respective Enqueue Service. There is exactly one Enqueue Service for application type ABAP and one for application type JAVA running in the SAP system. The SAP system comprises multiple application instances on different nodes. One or more SAP Replication Services run on the node on which the active Enqueue Service for ABAP and/or JAVA fails over in the event of a failure or planned outage.

Each Replication Service of type ABAP or JAVA maintains a copy of the Enqueue Service’s lock table; this copy is kept in a shared memory segment during runtime. The node that supports the Enqueue Service for ABAP and/or JAVA also hosts an additional virtual IP address (VIP) for communication. Whenever the active node fails or is shut down, the VIP and Enqueue Services fail over to the next node. The available node supports the Replication Service of type ABAP and/or JAVA.

The term ‘service’ is used to signify the SAP process that provides either Enqueue or Replication.

The nodes that host these services are termed the ‘Enqueue Server’ and the ‘Replication Server’.

SAP Enqueue and Replication Server Nodes
Oracle Clusterware, formerly known as Cluster Ready Services (CRS), is an integrated cluster management solution that enables linking multiple servers so that they function as a single system. While continuing to be required for Oracle RAC databases, Oracle Clusterware can also support non-Oracle applications.

OVERVIEW OF THE SAPCTL IMPLEMENTATION

This solution uses Oracle Clusterware to enable high availability for the SAP Enqueue Service. The implementation uses Oracle Clusterware modeling features so that each managed entity is represented as a resource. SAPCTL implements unique Oracle Clusterware resources, one each for the Enqueue Service of type ABAP or JAVA, the Replication Service for ABAP or JAVA, and the unique VIP resources for both types of Enqueues Service. For all additional SAP Application Instances of type CI, GW or DV, a pair of Oracle Clusterware resources for the instance itself and the accompanying unique VIP is used respectively.

The management policy for the Enqueue Service and VIP are configured so that the two are co-located, while the policy for the Replication Service resource ensures that it is never running on the same node as the associated Enqueue Service. The failover policy for the Enqueue Service resource guarantees that upon failure, the Enqueue Service is restarted on the node that is currently hosting the belonging Replication Service, if any. The Replication Service will be subsequently relocated to a different node if one is available. This applies to both the ABAP and JAVA application server type of SAP WebAS.

All resources for SAP Application Instances of type CI or DV have a co-located VIP resource assigned. The failover policy for this set of resources is to relocate the application service together with the associated VIP to an available node in the cluster in case of any error. There is no restart attempt by default.

For all types of supported SAP Instances, e.g. ASCS, ERS, CI or DV, an additional resource for the SAP Start Service is defined in CRS. Every SAP Instance has a dependency on the associated SAP Start Service and is always co-located if the SAP instance is running. The SAP Start Service for an SAP Instance should be always running on one node in the cluster, so the SAPCTL command line interface does not provide a function to start or stop the SAP Start Service.

If you need to stop a running SAP Start Service, you can stop it by Oracle Clusterware command crsctl.
SUPPORTED CONFIGURATIONS

The Enqueue and Replication Services for ABAP and/or JAVA can run on any node that supports SAP R/3 or SAP WebAS which is limited to the following two configurations:

- All solutions require shared disk for the Oracle Cluster Registry (OCR) and voting disk device.
- All solutions assume that SAP runs in an Oracle RAC environment.

Co-location of Enqueue and Replication Services with the Oracle database. For example, a two-node Linux cluster running Oracle RAC and R/3 in a co-located configuration.

Co-location of Enqueue and Replication Services in the same Oracle RAC environment as the Oracle database. For example, a four-node Linux cluster running Oracle RAC on two nodes and R/3 on two nodes of the same cluster.
INSTALLATION AND MANAGEMENT

You must install the Oracle Database software that you are using with your SAP installation in the operating system user oracle and primary group oinstall. In addition, the users <sid>adm from SAP as well as the Oracle Clusterware owner (oracle) must belong to the dba group. For more information about groups and users for SAP environments, refer to the SAP documentation.

REQUIREMENTS FOR RUNNING SAPCTL

Before running SAPCTL, your environment must meet the following conditions:

• You must run 'create' and 'remove' commands as the 'root' user.
• You must run 'start' and 'stop' commands as 'sapuser', this is the software owner of your SAP installation.
• You may run 'status' command as any user.

These requirements describe the default Oracle permission scheme. You can, however, modify the permission scheme with the crsctl setperm command.
OVERVIEW OF INSTALLATION AND CONFIGURATION

The procedures for installing this solution comprise the following tasks:

- Appropriately configuring the SAP application
- Registering the SAP Enqueue and Replication Services with the Oracle Clusterware.

CONFIGURING SAP

The configuration changes that you need to make to enable SAPCTL are not automated.

A few minor modifications to your existing configuration, namely the adaptation of parameters within the profiles is required.

The use of startsap / stopsap scripts is no longer supported with this version of SAPCTL. Instead, the SAP Start Service sapstartsrv is used to start, stop and monitor the SAP instances. The sapstartsrv service for every instance must be up and running on one node in the cluster to allow SAP Tools monitoring of SAP Instances. The SAP Start Service sapstartsrv should not run simultaneously on multiple nodes, especially if the SAP installation in the cluster is on a shared cluster filesystem. This is the default for Oracle RAC installations with SAP.

With the introduction of Oracle Clusterware resources for SAP Start Services, SAPCTL will automatically bring up the SAP Start Service if you start the SAP Instance by SAPCTL command.

Please note that the use of sapstart services requires unique instance numbers for all SAP instances. It is no longer possible to use the same number twice. Therefore you probably need to change the instance numbers for all replication instances (ABAP or JAVA). E.g. ASCS01 and ERS03 is a valid combination, whereas ASCS01 and ERS01 is not allowed.

Also note that using an Oracle RAC database requires additional profile changes. The configuration steps for this are documented in the white paper 'Configuration of SAP NetWeaver for Oracle 11g Release 2 Real Application Clusters Unix and Linux'.
INSTANCE PROFILE: CENTRAL SERVICES INSTANCE (ASCS, SCS), ENQUEUE AND MESSAGE SERVICE FOR ABAP AND JAVA

You must adapt the profiles for use by the standalone enqueue to the virtual node in the cluster on which the service runs. The virtual node name is part of the filename. Therefore, you must rename or create a copy of the files with the appropriate name, if not already done by sapinst with environment variable SAPINST_USE_HOSTNAME.

The virtual hostname must resolve to the VIP IP address associated with the SAP instance (ASCS, SCS, DVEBMGS, etc.) by either /etc/hosts resolution or DNS lookup. E.g., if the virtual hostname for ABAP Standalone Enqueue is sap_abapvip, this hostname must resolve to the IP address you specify during creation of Oracle Clusterware resources in –abapvip parameter of sapctl create command. Same applies to the –javavip and all –asinstvip parameters.

The directory holding the instance profile and the start commands should reside on a shared cluster file system. The default path is /usr/sap/<SID>/SYS/profile.

An example of the parameter changes for a standalone Enqueue Service for ABAP and JAVA on a cluster protected by Oracle Clusterware is available in Appendix 1.

INSTANCE PROFILE: ENQUEUE REPLICAION INSTANCE (ERS)

The Enqueue Replication instance has an instance and a start profile similar to any other SAP instance. The required modifications to these profiles are the same as those that this paper described for the standalone enqueue instance.

The Enqueue Replication instance does not require a VIP. So the hostname extension in the instance profile does not matter. Nevertheless it’s a good practice to use the same virtual hostname as the associated Enqueue instance.

See Appendix 1 for examples.

INSTANCE PROFILE: CENTRAL INSTANCE (CI) OR APPLICATION SERVER INSTANCES (DV)

The SAP application server instances, either configured as a dialog or dialog and update instances, are also eligible to be under the protection of Oracle Clusterware. In case of the ABAP application stack, it is still required to configure a Central instance (CI) even if the central services like message and enqueue service are part of the standalone ASCS instance type. Beside of the SAP Central Instance, it is possible to have additional SAP server instances (type DV) protected by SAPCTL. To configure additional Oracle Clusterware resources for the SAP server instances, run the sapctl create command once for every additional SAP server instance.

An example of a SAP Central Instance (CI) with failover capabilities on an Oracle CRS cluster is available in Appendix 1. Note that the instance name of a SAP central instance is DVEBMGS.

The Central Instance and all instances of type DV all have a unique VIP associated, if protected by Oracle Clusterware and running in the cluster.

For the relationship between virtual hostname to use and IP addresses for the VIPs the same rules as with SAP Central Services Instances (ASCS, SCS) apply.
All virtual hostnames must resolve by either DNS or /etc/host lookup to the IP address given during SAPCTL resource creation.

**INSTALLATION OF SAPCTL**

The installation must be performed as user root.

Create the following directories:

```
# mkdir /usr/sap/sapctl/bin
# mkdir /usr/sap/sapctl/log
# mkdir /usr/sap/sapctl/perl
```

Untar the SAPCTL scripts to directory /usr/sap/sapctl/bin

```
# cd /usr/sap/sapctl/bin
# tar –xvf sapctl.tar .
```

In directory /usr/sap/sapctl/bin, create 2 symbolic links to CRS executables installed in your CRS installation.

```
# ln –s /oracle/GRID/11202/bin/crsctl   /usr/sap/sapctl/bin/crsctl
# ln –s /oracle/GRID/11202/bin/crsctl.bin   /usr/sap/sapctl/bin/crsctl.bin
```

Recursively copy the perl subdirectory from your CRS installation to directory /usr/sap/sapctl/perl

```
# cp –r /oracle/GRID/11202/perl/*   /usr/sap/sapctl/perl
```

Recursively change ownership and permission for sapctl subdirectories

```
# chown –r  <sid>adm:sapsys /usr/sap/sapctl
# chmod –r 775 /usr/sap/sapctl
```

With these permissions all users with primary group sapsys have permission to use SAPCTL. You can use this setup for multiple SAP installations / multiple SID’s.

**UPRGADE FROM PREVIOUS VERSION**

Note: This section only applies if your current sapctl version is not sapctl version 6
Upgrades of sapctl version 6 is covered by separate upgrade tool

Delete all scripts installed from older versions of SAPCTL installed in directory /usr/sap/<SID>/SYS/exe/run. The names of the script files are:

- sapctl
- sapctl.pl
- sapwrap
- crssapactions.pl

Upgrade of CRS resources defined with older versions of SAPCTL is not possible. You must delete already defined CRS resources for SAP before you proceed with (re-) registering with Oracle Clusterware. Use command

```
# /usr/sap/sapctl/bin/sapctl remove all –sapsid <SID>
```

to delete the resources. Repeat this for all SAP systems protected by SAPCTL. After that, continue with registration as shown below.

**UPRGADE FOR SAPCTL VERSION 6**
Note: This section only applies if your current sapctl version is already version 6 and you plan to upgrade resource definition for application vip’s used for SAP.

Replace the scripts in /usr/sap/sapctl/bin with the scripts delivered with this version. Change owner and group for all scripts in /usr/sap/sapctl/bin to <sid>:sapsys.

```
# chown -r <sid>:sapsys /usr/sap/sapctl/bin
# chmod -r 775 /usr/sap/sapctl/bin
```

Start SAP resources with sapctl afterwards.

**REGISTERING WITH ORACLE CLUSTERWARE**

Registration with Oracle Clusterware, as well as the subsequent management of the resources, is done by using SAPCTL. An overview of SAPCTL functionality follows and a 'Worked Example' appears later in this paper.

**Interface for SAP HALIB library**

SAP provides a platform dependend library for vendor specific cluster solutions. This library is dynamically loaded by executable sapstartsrv during startup of the process. You must download the appropriate library for your platform / version from SAP service marketplace. SAPCTL provides a script “socc” (SAP Oracle Cluster Connector) which implements the functions required by SAP for interaction / monitoring with cluster solutions of different vendors. Script “socc” is platform independent.

Activation / load of the SAP halib is controlled with profile parameters in the instance profile. See Appendix 1 – Sample Profile Scripts for example. Also refer to SAP documentation for SAP high availability library for your platform.

Hint: Depending on your SAP kernel version, the SAP halib may require the script interface to reside in “/usr/local/bin/sap_cluster_connector”. Create a symbolic link to “/usr/sap/sapctl/bin/socc” to adhere to the regulations.

```
# ln -s /usr/local/bin/sap_cluster_connector /usr/sap/sapctl/bin/socc
```
MANAGEMENT INTERFACE

The management of the Enqueue and Replication Service as well as the VIP resource should be done using SAPCTL. This command line interface can manage any number of the installations on an Oracle Clusterware managed environment.

FUNCTIONALITY

SAPCTL performs the following:

• Creates and removes the Oracle Clusterware configuration of the SAP high availability subsystem.

• Starts the SAP high availability subsystem that is wrapped with Oracle Clusterware as a whole as well as its individual components.

• Stops the SAP high availability subsystem that is protected by the Oracle Clusterware as a whole as well as its individual components.

• Queries the state (status) of the SAP high availability subsystem as a whole as well as that of the components that comprise it.

• Relocates a resource to other hosting members on demand.
**USAGE**

    sapctl <command> [<object>] -sapsid <SAP_SID> [options]

    Command: start|stop|status|create|remove|relocate
    Objects: abapvip|javavip|abapenq|javaenq|
             abaprep|javarep|abapall|javaall|
             asinstall|asinstvip|asinst|
             gwvip|gwinst|wdspvip|wdspinst|all

**NOTE:**
Create and remove commands work on all objects only.
For detailed help about each command and object and its options use:
    sapctl <command> -h

**<Objects>**

abapvip: Refers to the VIP for the Enqueue Service of Web AS ABAP
javavip: Refers to the VIP for the Enqueue Service of Web AS JAVA
asinstvip: Refers to the VIP for SAP Web AS instances (type CI, DV)
abapenq: Refers to the Enqueue Service of Web AS ABAP
javaenq: Refers to the Enqueue Service of Web AS JAVA
abaprep: Refers to the Replication Service for Web AS ABAP
javarep: Refers to the Replication Service for Web AS JAVA
asinst: Refers to the SAP Web AS instances (type CI, DV)
abapall: Includes all resources associated with the Web AS ABAP
javaall: Includes all resources associated with the Web AS JAVA
asinstall: Include all resources associated with the SAP Web AS instances (type CI, DV)
gwvip: Refers to the VIP for SAP Standalone Gateway instances (Type GW)
gwinst: Refers to the SAP Standalone Gateway instance
wdspvip: Refers to the VIP for SAP WebDispatcher instances
wdspinst: Refers to the SAP WebDispatcher
all: Includes all of the resources associated with the solution for the specified SAP SID.

**<Commands>**

- **start**
  Starts the resources associated with the object.

- **stop**
  Stops the resources associated with the object.

- **status**
  Displays the current state or status of resources that are associated with the object.
  This includes the processed output of the resource state as reported by “crsctl
  status resource <resource_name>”

- **create**
  Creates the Oracle Clusterware resources for the SAP high availability subsystem.
  This command is only supported for a single object 'all' and must be run as
  the 'root' user because this operation must create a VIP resource. The following
  parameter switches are supported:
The VIP resource will be created as being owned by the 'root' user, while other resources are created as being owned by the SAP user. All of the generated Oracle Clusterware action scripts are owned by the SAP user.

- **remove**

Removes the Oracle Clusterware resources and Oracle Clusterware action scripts for the SAP high availability subsystem. This operand is only supported for a single object 'all' and must be run as the 'root' user because this operation must delete a VIP resource. The remove command deletes Oracle Clusterware type definitions for SAP resources as well if no other SAP system has dependencies on this type. E.g. if you have SAP systems <SID1> and <SID2> protected by SAPCTL, the type definition for SAP resources will only be deleted if the last SAP system is removed from SAPCTL protection.

- **relocate**

Relocate a running resource to other host from the list of valid hosting members. VIP resources are not allowed to be relocated, as VIP resources have dependent resources defined. If a resource with dependency to a VIP is specified, then the associated VIP resource gets relocated as well.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-abaprep &lt;Instance name&gt;</td>
<td>Name of the AB AP Replication instance.</td>
</tr>
<tr>
<td>-javarep &lt;Instance name&gt;</td>
<td>Name of the JAVA Replication instance.</td>
</tr>
<tr>
<td>-abapenq &lt;Instance name&gt;</td>
<td>Name of the AB AP Replication instance. The VIP resource for ABAP is relocated as well.</td>
</tr>
<tr>
<td>-javapenq &lt;Instance name&gt;</td>
<td>Name of the JAVA Replication instance. The VIP resource for JAVA is relocated as well.</td>
</tr>
<tr>
<td>-asinst &lt;Instance name&gt;</td>
<td>Name of the CI or DV instance. The VIP resource for the instance is relocated as well.</td>
</tr>
<tr>
<td>-gwinst &lt;Instance name&gt;</td>
<td>Name of the Standalone Gateway instance. The VIP resource for the instance is relocated as well.</td>
</tr>
<tr>
<td>-wdspinst &lt;Instance name&gt;</td>
<td>Name of the WebDispatcher instance. The VIP resource for the instance is relocated as well.</td>
</tr>
<tr>
<td>-to &lt;node&gt;</td>
<td>Tells the node where the resource should run after relocation.</td>
</tr>
</tbody>
</table>

**CONCLUSION**

This paper has shown how to use Oracle Clusterware to enable protection of both the SAP Enqueue Service and the SAP Replication Service.

Oracle Clusterware can protect SAP installation types SAP Web AS ABAP, SAP Web AS JAVA or both.

Oracle Clusterware provides an easy way to protect SAP Services from failure. More information about Oracle Clusterware can be found at:


And as a reminder - For details about configuring SAP with Oracle RAC refer to the white paper ‘Configuration of SAP NetWeaver for Oracle Grid Infrastructure 11.2 with Oracle Real Application Clusters 11g Release 2’ in key topics folder ‘SAP on Oracle Real Application Clusters (RAC)’ on the SAP SDN website at:

### WORKED EXAMPLE

This worked example assumes that you have completed the following steps.

- Oracle Clusterware is installed.
- Oracle RAC is installed.
- You have installed SAP and the SAPCTL package

You need to know the following information about your SAP installation:

<table>
<thead>
<tr>
<th>Required Information</th>
<th>Description</th>
<th>Typical Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP_SID</td>
<td>The name of the SAP Instance containing the SAP services that require protection</td>
<td>RAC</td>
</tr>
<tr>
<td>interface</td>
<td>The name of the public network interface</td>
<td>eth0</td>
</tr>
<tr>
<td>netmask</td>
<td>The IP network Netmask</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>network address</td>
<td>The address of the network</td>
<td>140.86.242.0</td>
</tr>
<tr>
<td>hosting_members</td>
<td>The list of nodes that are capable of running the SAP services</td>
<td>sapnode1,sapnode2</td>
</tr>
<tr>
<td>abap_enqueue</td>
<td>The name of the SAP enqueue service of Web AS ABAP</td>
<td>ASCS01</td>
</tr>
<tr>
<td>abap_replication</td>
<td>The name of the SAP replication service of Web AS ABAP</td>
<td>ERS03</td>
</tr>
<tr>
<td>abap_msport</td>
<td>The portnumber of message server for ABAP (SAP profile parameter rdisp/msserv)</td>
<td>3601</td>
</tr>
<tr>
<td>IP address ABAP</td>
<td>A new IP address that will be bound to the public network interface for Web AS ABAP</td>
<td>140.86.242.63</td>
</tr>
<tr>
<td>java_enqueue</td>
<td>The name of the SAP enqueue service of Web AS JAVA</td>
<td>SCS02</td>
</tr>
<tr>
<td>abap_replication</td>
<td>The name of the SAP replication service of Web AS ABAP</td>
<td>ERS04</td>
</tr>
<tr>
<td>java_msport</td>
<td>The portnumber of message server for JAVA (SAP profile parameter j2ee/ms/port)</td>
<td>3902</td>
</tr>
<tr>
<td>IP address JAVA</td>
<td>A new IP address that will be bound to the public network interface for Web AS JAVA</td>
<td>140.86.242.64</td>
</tr>
<tr>
<td>Instance name</td>
<td>The name of the SAP instance of type CI or DV</td>
<td>DVEBMGS00</td>
</tr>
<tr>
<td>IP address Instance</td>
<td>A new IP address that will be bound to the public network interface for SAP instance</td>
<td>140.86.242.65</td>
</tr>
</tbody>
</table>
The SAPCTL utility

```
[root@oracx2 bin]# sapctl
sapctl version 7.0 Patch 0    Production Copyright 2013 Oracle. All rights reserved
usage:
sapctl <command> [object] -sapsid <SAP_SID> [options]
  command : start|stop|status|create|remove|relocate
  objects : abapvip|javavip|abapenq|javaenq|abaprep|javarep
            |abapall|javaall|asinstall|asinstvip|asinst|all
NOTE:
Create and remove commands work on all objects only
For detailed help on each command and object and its options use:
  sapctl <command> -h
```

Enable Oracle Clusterware Protection for your SAP resources

```
[root@oracx2 bin]# sapctl create -sapsid RAC -if eth0
  -nodes oracx1,oracx2,oracw1,oracw2
  -abapvip 140.86.242.63 -abapmsport 3601 -abapenq ASCS01 -abaprep ERS03
  -javavip 140.86.242.64 -javamsport 3902 -javaenq SCS02 -javarep ERS04
  -asinstvip 140.86.242.65 -asinst DVEBMGS00

Creating SAP ABAP VIP
Creating SAP ABAP Enqueue resource
Creating SAP ABAP Replication resource
Creating SAP JAVA VIP
Creating SAP JAVA Enqueue resource
Creating SAP JAVA Replication resource
Creating SAP INST VIP for DVEBMGS00
Creating SAP INST resource for DVEBMGS00
Done
```

In this case, the SAP Central Services for ABAP and JAVA of the SAP system RAC is going to be protected:

The SAP enqueue service for Web AS ABAP called ASCS01 and the Replication service called ERS03 will be allowed to run on separate nodes from the following node list "oracx1 oracx2 oracw1 oracw2". A new virtual IP address 140.86.242.63 with a subnet of 255.255.255.0 will be loaded onto the eth0 network adapter.

Network adapter eth0 has associated network (-net parameter) 140.86.242.0.

The SAP enqueue service for Web AS JAVA called SCS02 and the Replication service called ERS04 will be allowed to run on separate nodes from the following node list "oracx1 oracx2 oracw1 oracw2". A new virtual IP address 140.86.242.64 with a subnet of 255.255.255.0 will be loaded onto the eth0 network adapter.

Network adapter eth0 has associated network (-net parameter) 140.86.242.0.

The SAP Central Instance called DVEBMGS00 is allowed to run on separate nodes from the following node list "oracx1 oracx2 oracw1 oracw2". A new virtual IP address 140.86.242.65 with a subnet of 255.255.255.0 will be loaded onto the eth0 network adapter.

Network adapter eth0 has associated network (-net parameter) 140.86.242.0.

The script does not automatically start these resources.
To see which SAP Instances have been protected by Oracle Clusterware

```
[root@oracx2 bin]# sapctl config
```

SAP_SIDs protected by the Oracle Clusterware are:

- RAC

In this case only one SAP Instance called 'RAC' is being protected by Oracle Clusterware.

Check the Status of the Oracle Clusterware Protected SAP resources

```
[root@oracx2 bin]# sapctl status all -sapsid RAC
```

SAP ABAP Enqueue service is OFFLINE
SAP JAVA Enqueue service is OFFLINE
SAP ABAP Replication service is OFFLINE
SAP JAVA Replication service is OFFLINE
SAP instance DVEBMGS00 is OFFLINE

To check the status of the Oracle Clusterware resources that protect the SAP services use the status command. Please note that the status of all defined SAP instances is reported. The status of associated VIPs and the SAP Start Services for the SAP instances will not be displayed.

Start the Oracle Clusterware Protected SAP resources

```
[root@oracx2 bin]# sapctl start all -sapsid RAC
```

Starting SAP ABAP Enqueue service
Starting SAP JAVA Enqueue service
Starting SAP ABAP Replication service
Starting SAP JAVA Replication service
Starting SAP instance DVEBMGS00
SAP ABAP Enqueue service is ONLINE on oracx1
SAP JAVA Enqueue service is ONLINE on oracx1
SAP ABAP Replication service is ONLINE on oracx2
SAP JAVA Replication service is ONLINE on oracx2
SAP instance DVEBMGS00 is ONLINE on oracx1
Done

Notice that you can use this command to start either the enqueue or replication services. However you should start the enqueue service first.

Check the Status after the start

```
[root@oracx2 bin]# sapctl status all -sapsid RAC
```

SAP ABAP Enqueue service is ONLINE on oracx1
SAP JAVA Enqueue service is ONLINE on oracx1
SAP ABAP Replication service is ONLINE on oracx2
SAP JAVA Replication service is ONLINE on oracx2
SAP instance DVEBMGS00 is ONLINE on oracx1

The resource may take a short while to start. Therefore, some resources may indicate that they are OFFLINE while they are being started.

Stop the Enqueue Service for Web AS ABAP (and associated VIP)

```
[root@oracx2 bin]# sapctl stop abapenq -sapsid RAC
```

Providing High Availability for SAP Resources   Page 19
Stopping SAP ABAP Enqueue service
SAP ABAP Enqueue service is OFFLINE
Done

You can use the stop command to stop the Enqueue service. Note that this will not stop the associated VIP.

Start the Enqueue Service for Web AS ABAP (and associated VIP)

```
[root@oracx2 bin]# sapctl start abapenq -sapsid RAC
```

Starting SAP ABAP Enqueue service
SAP ABAP Enqueue service is ONLINE on oracx2
Done

You can use the start command to start the Enqueue service. This will also start the associated VIP. Notice that the Enqueue service now runs on the node that hosted the Replication service before.

Relocate SAP resources to other hosting member

```
[root@oracx2 bin]# sapctl relocate -sapsid RAC -abaprep ERS03 -to oracw1
```

Stopping SAP ABAP Replication service
Relocating SAP ABAP Replication Service to node oracw1
Done

You can use the relocate command to move SAP resources to other hosting members. You cannot relocate any VIP resource with this command. Valid SAP resources are the Enqueue service, the Replication service or any SAP instance of type CI or DV.

Stop all of the Oracle Clusterware managed SAP resources

```
[root@oracx2 bin]# sapctl stop all -sapsid RAC
```

Stopping SAP instance for DVEBMGS00
Stopping SAP JAVA Replication service
Stopping SAP ABAP Replication service
Stopping SAP JAVA Enqueue service
Stopping SAP ABAP Enqueue service
SAP ABAP Enqueue service is OFFLINE
SAP JAVA Enqueue service is OFFLINE
SAP ABAP Replication service is OFFLINE
SAP JAVA Replication service is OFFLINE
SAP instance DVEBMGS00 is OFFLINE
Done

You can choose to not stop all of the Oracle Clusterware resources by replacing all with either 'enqueue' or 'replication' but you should stop the replication service first.

Remove Oracle Clusterware protection for SAP resources

```
[root@oracx2 bin]# sapctl remove all -sapsid RAC
```

Removing resource SAP ABAP Enqueue service
Removing resource SAP ABAP VIP
Removing resource SAP JAVA Enqueue service
Removing resource SAP JAVA VIP
Removing resource SAP ABAP Replication service
Removing resource SAP JAVA Replication service
Removing resource SAP instance DVEBMGS00
Removing resource SAP instance VIP for DVEBMGS00
Done
Note you cannot remove clusterware protection for a single resource. You must remove protection for all of the resources within an SAP instance in a single operation. This command must be run as the 'root' user.
APPENDIX 1 – SAMPLE PROFILE SCRIPTS

Start or Restart of SAP processes by sapstartsrv

The daemon process sapstartsrv for a specific SAP instance starts, stops and monitors the different processes. SAP has implemented a restart capability, enabling an automatic restart of crashed processes by sapstartsrv in case of a failure within the application server process. E.g., sapstartsrv will restart a broken message server process if the Restart_Program_xx directive is used in the Instance- or START-profile of the Central Services Instance ASCS or SCS.

It is not recommended to use this feature in combination with SAPCTL. For certain process types it is not supported at all. You cannot use the Restart_Program_xx directive for the enqueue server process (enq). For this process, you must use the Start_Program_xx directive in the Instance- or START-profile. You may use Restart_Program_xx for the message server (msg).

If you use Restart_Program_xx for the enqueue replication process, this process will not be moved to another node after automatic restart by sapstartsrv process. It is strongly recommended that you use Start_Program_xx directive on the profile of Enqueue Replication Instance ERS.

Check in all profiles of SAP instances protected by SAPCTL that only Start_Program_xx directive is used. Replace Restart_Program_xx by Start_Program_xx for all critical process types. E.g. for enq process:

```
Restart_Program_01 = local $(EN) pf=$(PF)
```

Replace with

```
Start_Program_01 = local $(EN) pf=$(PF)
```

Best practice with SAPCTL is not to use restart capabilities of sapstartsrv daemon. This contributes to the observation that it is better to do an immediate failover instead of probably unsuccessful retry to bring service up again on same computing node.

Note: In all examples given here only the required entries for SAP Standalone Enqueue and Enqueue Replication are shown.

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS ABAP on a cluster. The virtual hostname of the cluster nodes hosting the ASCS instance is sap_abapvip in this example.

**Script: DEFAULT.PFL**

```
#----------------------------------------------------------------------
# rdisp/mshost must resolve to the IP address given as –abapvip
# parameter in sapct1 create command
#----------------------------------------------------------------------
rdisp/mshost = sap_abapvip
rdisp/msserv = sapmsRAC
rdisp/msserv_internal = 3901
enque/process_location = REMOTESA

#----------------------------------------------------------------------
# enque/serverhost must resolve to the IP address given as –abapvip
# parameter in sapct1 create command
#----------------------------------------------------------------------
enque/serverhost = sap_abapvip
enque/serverinst = 01
```

Providing High Availability for SAP Resources  Page 22
Script: **RAC_ASCS01_sap_abapvip**

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS ABAP on a cluster. The virtual hostname of the cluster nodes hosting the ASCS instance is sap_abapvip in this example.

```bash
enque/table_size=4096
enque/process_location=LOCAL
enque/server/replication=true
enque/server/internal_replication=true
enque/enrep/keepalive_count=0

# SAP HA interface for cluster

service/halib=/usr/sap/RAC/ASCS01/exe/saphascriptco.so
service/halib_cluster_connector=/usr/sap/sapctl/bin/socc
service/protectedwebmethods=NONE

```

Script: **RAC_ERS03_sap_abapvip**

The following is an example of the profile scripts for a standalone Replication Service for SAP Web AS ABAP on a cluster.

```bash
enque/table_size=4096
enque/process_location=LOCAL
enque/server/replication=true
enque/server/internal_replication=true
enque/enrep/keepalive_count=0

# SAP HA interface for cluster

service/halib=/usr/sap/RAC/ERS01/exe/saphascriptco.so
service/halib_cluster_connector=/usr/sap/sapctl/bin/socc
service/protectedwebmethods=NONE

```

Script: **RAC_SCS02_sap_javavip**

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS JAVA on a cluster. The virtual hostname of the cluster nodes hosting the SCS instance is sap_javavip in this example.

```bash
enque/table_size=4096
enque/process_location=LOCAL
enque/server/replication=true
enque/server/internal_replication=true
enque/enrep/keepalive_count=0

# SAP HA interface for cluster

service/halib=/usr/sap/RAC/SCS02/exe/saphascriptco.so
service/halib_cluster_connector=/usr/sap/sapctl/bin/socc
service/protectedwebmethods=NONE

```

Script: **RAC_ERS04_sap_javavip**

The following is an example of the profile scripts for a standalone Replication Service for SAP Web AS JAVA on a two-node cluster.

```bash
enque/table_size=4096
enque/process_location=LOCAL
enque/server/replication=true
enque/server/internal_replication=true
enque/serverinst = 02
enque/enrep/keepalive_count=0
rdisp/msserv = 0

```

---

Script: **RAC_ASCS01_sap_abapvip**

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS ABAP on a cluster. The virtual hostname of the cluster nodes hosting the ASCS instance is sap_abapvip in this example.

```bash
j2ee/scs/host = sap_abapvip
j2ee/scs/system = 02
j2ee/ms/port = 3902

```

---

Script: **RAC_SCS02_sap_javavip**

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS JAVA on a cluster. The virtual hostname of the cluster nodes hosting the SCS instance is sap_javavip in this example.

```bash
j2ee/scs/host = sap_javavip
j2ee/scs/system = 02
j2ee/ms/port = 3902

```
rdisp/msserv_internal = 3902
ms/standalone = 1
ms/server_port_0 = PROT=HTTP,PORT=81$$
#---------------------------------------------
# SAP HA interface for cluster
#---------------------------------------------
service/halib_cluster_connector=/usr/sap/sapctl/bin/socc
service/halib=$user/sap/RAC/ERS04/exe/saphascriptco.so
service/protectedwebmethods=NONE

Script: RAC_DVEBMGS00_sap_instvip
The following is an example of the profile scripts for a SAP Central Instance on a
cluster. The virtual hostname of the cluster nodes hosting the instance is
sap_instvip in this example.

#---------------------------------------------
# SAP HA interface for cluster
#---------------------------------------------
service/halib=$user/sap/RAC/DVEBMGS00/exe/saphascriptco.so
service/halib_cluster_connector=/usr/sap/sapctl/bin/socc
service/protectedwebmethods=NONE

APPENDIX 2 – TROUBLESHOOTING AND LOG FILES
The troubleshooting content in this section assumes that you are familiar with the
Oracle Clusterware and SAP R/3 commands. You can troubleshoot SAPCTL by
examining the log files as described in this section.
The following is the log of the actions that SAPCTL performs. You should also
test to see which commands were executed. If you cannot resolve your problem,
then examine this log file to determine why resource actions do not perform as
expected:

/usr/sap/sapctl/log/sapctl.log

In addition to the sapct1.log file, every resource for SAP instances and startupservice
will use a separate log file. The name of the resource is part of the logfile name:

/usr/sap/sapctl/log/sap.<SID>.<INST>.<type>.log

E.g., the logfile for the central services instance ABAP ASCS01 is

/usr/sap/sapctl/log/sap.RAC.ASCS01.abapenq.log

You can also examine the content of the CRSD log file:

${ORA_CRS_HOME}/log/<hostname>/crsd/crsd.log

A detailed log output from actions performed by SAPCTL per node for every SAP
SID is in directory:

${ORA_CRS_HOME}/log/<hostname>/agent/crsd/scriptagent_<sidadm>

All commands passed from SAP HALIB saphascriptco.so are logged in file

/usr/sap/sapctl/log/socc.log
APPENDIX 3 – SAPCTL BILL OF MATERIALS

This section describes the SAPCTL bill of materials (BOM). The un-compressed tar file contains the following items:

- ./bin/sapctl – This is a Perl-based SAPCTL management utility implementation.
- ./bin/crssapactions.pl – The action script called by CRS to start, stop and monitor all defined SAP resources.
- ./bin/socc – The SAP Oracle cluster connector. A script implementing SAP interface for cluster vendors providing HA functionality for SAP resources.
- ./doc/sapctl.pdf – This is a copy of this document.

APPENDIX 4 – CRS RESOURCES AND TYPES

The SAPCTL create command creates all CRS resources and type definition for SAP instances. Every SAP instance type has specialized CRS resource type. Below is the list with all resources and types used by SAPCTL. Depending on the SAP instance types protected by SAPCTL you may find only a subset in an actual installation.

List of CRS resources used by SAPCTL

```
sap.<SID>.ASCS<NR>.startsrv
sap.<SID>.ASCS<NR>.abapenq
sap.<SID>.ASCS<NR>.abaprep
sap.<SID>.ASCS<NR>.abapvip
sap.<SID>.ASCS<NR>.javaeq
sap.<SID>.ASCS<NR>.javavip
sap.<SID>.DVERBMS<NR>.startsrv
sap.<SID>.DVERBMS<NR>.sapinst
sap.<SID>.DVERBMS<NR>.instvip
sap.<SID>.DV<NR>.startsrv
sap.<SID>.DV<NR>.sapinst
sap.<SID>.DV<NR>.instvip
sap.<SID>.G<NR>.startsrv
sap.<SID>.G<NR>.sapgw
sap.<SID>.G<NR>.gvip
sap.<SID>.W<NR>.startsrv
sap.<SID>.W<NR>.wdspinst
sap.<SID>.W<NR>.wdspvip
ora.net<x>.network
```

For detailed information of all resource attributes use CRS command “crsctl status resource <resource_name> -f”.

List of CRS Types used for SAPCTL

For implementation of the CRS resources defined for SAP, some specific resource types will get defined by sapctl during resource creation.

```
sap.abapenq.type
sap.abaprep.type
sap.javaeq.type
sap.javarep.type
sap.sapinst.type
sap.startsrv.type
app.appvip.type
```
For detailed information of resource types use CRS command “crsctl status type <type_name> -f”

Serverpool used for SAPCTL
If not already existing; sapctl will create a serverpool resource in CRS repository.

SAP_SP_<SID>
Information on nodes defined for the Serverpool can be obtained by CRS command “crsctl status serverpool <serverpool_name> -f”.

APPENDIX 5 – MULTIPLE APPLICATION VIP’S FOR SAP RESOURCES
The Oracle Clusterware Framework supports multiple application VIPs on different networks for SAP application resources. If a SAP resource has a dependency on multiple application VIPs, these VIPs are always co-located on the same host. Configuration of additional application VIPs is not supported by SAPCTL create command, so the required modifications must be done manually.

The following example script shows how to set up a second application VIP for SAP ASCS and SCS resource. You can use this script as template.

```bash
# Template for creation of second cluster resource VIP for
# SAP instance types ASCS, SCS and DVEBMGS
#
# Note that there are no VIP resources defined for ERS instances.
# Also the ERS instances and related resources for sapstartsrv processes
# must NOT be modified. Value for dispersion attribute is to one VIP only.
#
# Settings used in this example which needs to be adopted to your needs:
#
#   ABAP Central services Instance ASCS:    ASCS10
#   ABAP Replication Instance:              ERS20
#   JAVA Central Services Instance SCS:     SCS11
#   JAVA Replication Instance:              ERS21
#   ABAP Central Instance DVEBMGS:          DVEBMGS12
#   <sid>adm user for this installation:    kb1adm
#   <SID> for this installation:            KB1
#   Netmask for second network:             255.255.255.0
#   Subnet used for 2nd VIP resources:      192.168.20.0
#
# 1. Create additional network resource "ora.net3.network"
#    for the VIP’s on the second network.
#    Change ACL permissions of resource "ora.net3.network" to allow
#    user <sid>adm and user oracle to operate on this resource
#
/oracle/GRID/11202/bin/crsctl add resource ora.net3.network -type ora.network.type -attr
"USR_ORA_IF=eth1, USR_ORA_NETMASK=255.255.255.0, USR_ORA_SUBNET=192.168.20.0"
/oracle/GRID/11202/bin/crsctl setperm resource ora.net3.network -u user:kb1adm:r-x
/oracle/GRID/11202/bin/crsctl setperm resource ora.net3.network -u user:oracle:r-x
/oracle/GRID/11202/bin/crsctl setperm resource ora.net3.network -u group:sapsys:r-x
#
# 2. Create application VIP for ASCS and/or SCS instance on the second network.
#    Change ACL permissions of resources "sap.<SID>.[abapvip2|javavip2]" to allow
#    user <sid>adm and user oracle to operate on this resource
#    ABAP ASCS:
/oracle/GRID/11202/bin/crsctl add resource sap.KB1.abapvip2 -type app.appvip.type -attr
"START_DEPENDENCIES=hard(ora.net3.network) pullup(ora.net3.network) attraction(sap.KB1.ERS20.abaprep),STOP_DEPENDENCIES=hard(ora.net3.network)"
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.abapvip2 -u user:kb1adm:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.abapvip2 -u user:oracle:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.abapvip2 -u group:sapsys:r-x
```

Providing High Availability for SAP Resources
# JAVA SCS:
/oracle/GRID/11202/bin/crsctl add resource sap.KB1.javavip2 -type app.appvip.type -attr "RESTART_ATTEMPTS=0,START_TIMEOUT=100,STOP_TIMEOUT=100,USR_ORA_VIP=192.168.20.101,START_DEPENDENCIES=hard(ora.net3.network) pullup(ora.net3.network) attraction (sap.KB1.ERS21.javarep),STOP_DEPENDENCIES=hard(ora.net3.network)"
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.javavip2 -u user:kb1adm:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.javavip2 -u user:oracle:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.javavip2 -u group:sapsys:r-x

# ABAP CI DVEBMGS:
/oracle/GRID/11202/bin/crsctl add resource sap.KB1.DVEBMGS12.instvip2 -type app.appvip.type -attr "RESTART_ATTEMPTS=0,START_TIMEOUT=100,STOP_TIMEOUT=100,USR_ORA_VIP=192.168.20.102,START_DEPENDENCIES=hard(ora.net3.network) pullup(ora.net3.network),STOP_DEPENDENCIES=hard(ora.net3.network)"
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.DVEBMGS12.instvip2 -u user:kb1adm:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.DVEBMGS12.instvip2 -u user:oracle:r-x
/oracle/GRID/11202/bin/crsctl setperm resource sap.KB1.DVEBMGS12.instvip2 -u group:sapsys:r-x

3. Modify resource for ASCS and/or SCS instance and all related resources for the related sapstartsrv processes. Add START and STOP dependency on both application VIP's on backend and frontend network.
Note the single quotation mark for attribute START_DEPENDENCIES, required if there is a list.

# ABAP ASCS START_DEPENDENCIES and STOP_DEPENDENCIES for instance and startsapsrv resource:
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.ASCS10.abapenq -attr "STOP_DEPENDENCIES='hard(sap.KB1.abapvip,sap.KB1.abapvip2,sap.KB1.ASCS10.startsrv)'"'
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.ASCS10.startsrv -attr "START_DEPENDENCIES='hard(sap.KB1.abapvip,sap.KB1.abapvip2) pullup(sap.KB1.abapvip,sap.KB1.abapvip2) attraction(sap.KB1.ERS20.abaprep)'"'
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.ASCS10.startsrv -attr "STOP_DEPENDENCIES='hard(sap.KB1.abapvip,sap.KB1.abapvip2)'"'

# JAVA SCS START_DEPENDENCIES and STOP_DEPENDENCIES for instance and startsapsrv resource:
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.SCS11.javaenq -attr "START_DEPENDENCIES='hard(sap.KB1.javavip,sap.KB1.javavip2,sap.KB1.SCS11.startsrv) pullup(sap.KB1.javavip,sap.KB1.javavip2,sap.KB1.SCS11.startsrv) attraction(sap.KB1.ERS21.javarep)'"'
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.SCS11.javaenq -attr "STOP_DEPENDENCIES='hard(sap.KB1.javavip,sap.KB1.javavip2,sap.KB1.SCS11.startsrv)'"'
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.SCS11.startsrv -attr "START_DEPENDENCIES='hard(sap.KB1.javavip,sap.KB1.javavip2) pullup(sap.KB1.javavip,sap.KB1.javavip2) attraction(sap.KB1.ERS21.javarep)'"'
/oracle/GRID/11202/bin/crsctl modify resource sap.KB1.SCS11.startsrv -attr "STOP_DEPENDENCIES='hard(sap.KB1.javavip,sap.KB1.javavip2)'"'

# ABAP CI DVEBMGS START_DEPENDENCIES and STOP_DEPENDENCIES for instance and startsapsrv resource:

(Note that there are no CR/LF in the command lines shown)