Providing High Availability for SAP Resources

An Oracle White Paper
July 2008

Document Version 3.0

Providing High Availability for SAP Resources

Providing High Availability for SAP Resources

Providing High Availability is an essential component in the world of business continuity

This paper explains how to use Oracle Clusterware to enable 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.

In addition to the critical SAP high availability components, namely the SAP Enqueue and SAP Replication Service, SAPCTL 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.

SAP SUPPORT FOR HIGH AVAILABILITY

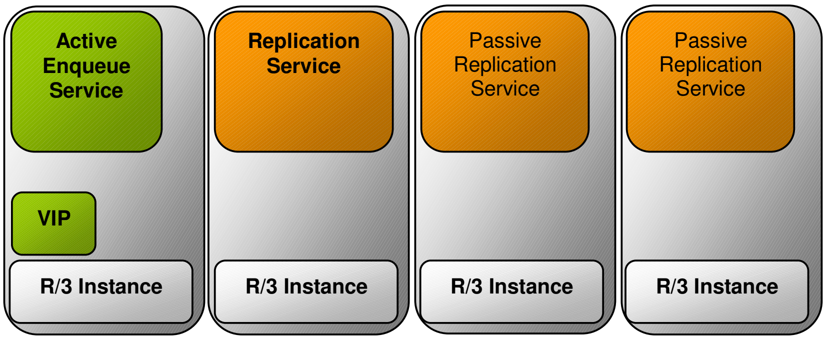
The SAP Enqueue Service provides distributed lock management for SAP application server instances. The SAP Enqueue Service must be available whenever an SAP installation uses more than one SAP instance. In order to support High Availability, the Enqueue Service together with the Message Service builds 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 falls over in the event of a failure or planned outage.

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'

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.



SAP Enqueue and Replication Server Nodes

ORACLE CLUSTERWARE

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 acompanying unique VIP is used respectivly.

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.

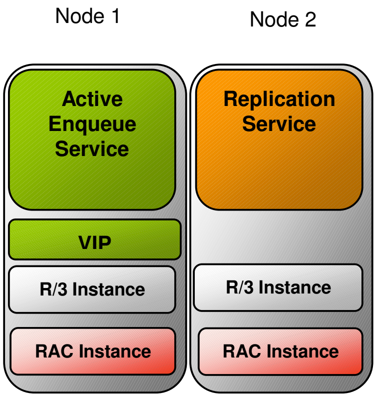
All resources for SAP Application Instances of type CI or DV have a co-located VIP resource assigned. The failover policy for these set of resources is to relocate the application service together with the associated VIP to an available node in the cluster in case of any error. There is no restart attempt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

SUPPORTED CONFIGURATIONS

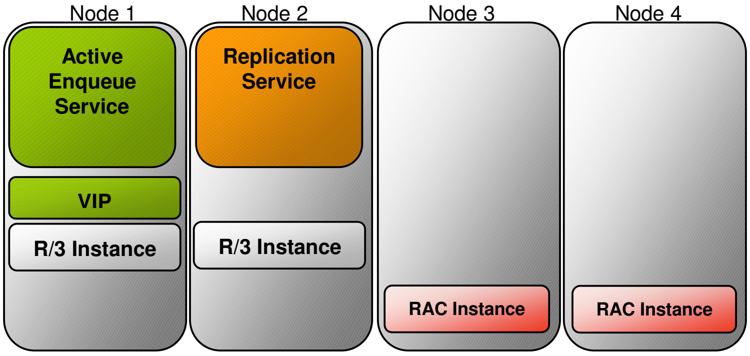
The Enqueue and Replication Services for ABAP and/or JAVA can run on any node that supports SAP R/3 or SAP WebAS which is limited to the following two configurations:

- 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 default operating system group, OSDBA (dba). In addition, the users <sid>adm, ora<sid> from SAP as well as the Oracle Clusterware owner (oracle) must belong to the dba group. The dba group is also the primary group for the Oracle Clusterware owner (oracle) and ora<sid>. 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:

- The ORA_CRS_HOME environment variable must be set to point at the Oracle Clusterware installation.
- You must run 'create', 'remove', and 'update' 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' and 'config' commands as any user.

These requirements describe the default Oracle permission scheme. You can, however, modify the permission scheme with the crs_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. Instead, you must manually configure the changes. The required changes are, however, minimal and affect only two shell scripts, namely startsap and stopsap. The changes to these two scripts are documented in Appendix 1 of this paper.

A few other minor modifications to your existing configuration, namely the generation of nearly identical profiles with different file names, are also required. This is because Oracle Clusterware does not change the hostname of the cluster node if the VIP fails over to another node. In addition, the SAP high availability agent configuration must consider that SAP queries the hostname or uses the cluster node's hostname to identify the profile and parameter file sets to run on the node.

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 10g Release 2 Real Application Clusters Unix and Linux'.

STARTSAP/STOPSAP SCRIPTS

Use the SAP 'startsap' and 'stopsap' scripts to start and stop SAP R/3 application instances. You must adapt these scripts to interact with Oracle Clusterware and to enable the use of the SAP high availability implementation with a standalone Enqueue Service and Enqueue Replication Service.

The modifications required for interaction with Oracle Clusterware are simple: After a successful start or stop of the SAP application instances, run a command with parameters to notify the Oracle Clusterware framework about the change of state. All other modifications in the startsap and stopsap for SAP R/3 scripts before SAP kernel version 700 enable SAPCTL to distinguish the two instance types, ASCS and ERS, from the other application instance types such as dialog (D), batch (B), gateway (GW), spool (S) and update (V). These modifications are no longer required when SAP kernel version 700 or higher is used.

ASCS is the default name for the standalone Enqueue Service. This ASCS instance contains the Enqueue Service and the message service, with a small number of dialog work processes. ERS is the instance name for the Enqueue Replication service instance, which is actually the replication process.

Appendix 1 shows a small portion of the 'startsap' and 'stopsap' scripts with the modifications denoted in boxes to highlight the required changes. This example is from SAP NW2004s using the 700 kernel. You can use this example as a template.

USER PROFILE SID<ADM>

The home directory for the SAP user sid<adm> contains profile scripts for the c-shell and the Bourne or bash shell that are commonly used in UNIX environments. There are four sets of scripts and the script sets have the following purposes:

- For the environment of the SAP application.
- To ensure that the Oracle database uses the required environment settings.
- For the SAP Java virtual machine.
- For SAP APO, an SAP tool that requires a specialized environment setup.

The appropriate script set is called during the execution of .login (csh) or .profile (sh). A common attribute of these scripts is the hostname used as the last portion of the script name. Because Oracle Clusterware can start the SAP Enqueue Service and the SAP Enqueue Replication Service on all of the hosts in the cluster with arbitrary hostnames, you must duplicate these scripts with as many different names as there are hosts in the cluster. You must do this on all hosts that are eligible for running the respective programs.

The important requirement of SAP for Oracle RAC is to have the Oracle RAC database's home directory reside on a shared cluster file system.

If the home directory is on a cluster file system, then create links with the different hostnames in the cluster to the various profile scripts. In the following example, SAP software has initially been installed on node1 of a four-node cluster. Node2, node3, and node4 are the other cluster members.

```
# su - sid<adm>
# for TYPENAME in "sapenv sapsrc dbenv dbsrc"
do
        for NODENAME in "node2 node3 node4"
        do
                ln .$TYPENAME_node1.sh .$TYPENAME_$NODENAME.sh
                ln .$TYPENAME_node1.csh .$TYPENAME_$NODENAME.csh
        done
done
```

In case of a .j2eee or .apo profile script, create the appropriate links in the same manner if these SAP applications are also controlled by Oracle Clusterware. Further discussion of this topic is outside of the scope of this white paper.

Most likely, the SAP Replicated Enqueue implementation is part of an Oracle RAC configuration that enables an SAP system. Refer to the change in the startsap script explained in Appendix 1. A short reminder: Setting the environment variable dbs_ora_tnsname overrides all of the subsequent settings for the selection of the database connection. Therefore, load balancing and failover to different database instances would not be possible. For details about configuring SAP with Oracle RAC databases, refer to the white paper "Configuration of SAP NetWeaver for Oracle 10g Release 2 Real Application Clusters Unix and Linux' on the SAP marketplace website at: http://service.sap.com/dbaora.

USER PROFILE ORA<SID>

For administrative purposes, a special Oracle database user is created by SAP. This user is known as ora<sid> in SAP installations. The user ora<sid> is the Oracle software owner for the database belonging to the SAP system. For this user account, the same changes to the profiles as for an SAP user account <sid>adm are required. Refer to the previous section, 'USER PROFILE <SID>ADM' for more information about the required changes.

INSTANCE PROFILE: CENTRAL SERVICES INSTANCE (ASCS, SCS), ENQEUE AND MESSAGE SERVICE FOR ABAP AND JAVA

See Appendix for the changes that are required to these profiles:

Appendix 2 shows the ABAP stack

Appendix 4 shows the JAVA stack

You must adapt the profiles for use by the standalone enqueue to the actual node in the cluster on which the service runs. The node name is part of the filename. Therefore, you must create a copy of the files with the appropriate name. 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 for a standalone Enqueue Service for ABAP on a two-node cluster is available in Appendix 2. The example for JAVA is in Appendix 4.

INSTANCE PROFILE: ENQUEUE REPLICATION INSTANCE (ERS)

See Appendix 3 for the changes that are required to these profiles

Appendix 3 shows the ABAP stack

Appendix 5 shows the JAVA stack

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.

For every node in the cluster, the last part of the file names must match the hostname. In addition, in the instance profile, you must change some of the parameters accordingly as shown in the example in Appendix 3 and Appendix 5.
An example for a standalone Enqueue Replication Instance for ABAP on a two-node cluster is available in Appendix 3.
An example for a standalone Enqueue Replication Instance for JAVA on a two-node cluster is available in Appendix 5.

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

See Appendix 6 for the changes that are required to these profiles

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. The restriction in this case is the fact that these instances must run on the cluster nodes. This is not valid in three-tier configurations. 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 a two-node cluster is available in Appendix 6. Note that the instance name of a SAP central instance is DVEBMGS.

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.

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.

SAP LEGACY MANAGEMENT INTERFACES

In addition to SAPCTL and the other conventional tools for managing SAP Enqueue and Replication Services, you can also use the startsap and stopsap scripts. However, Oracle recommends that you use only one management interface at the same time. As documented in this paper, these scripts need slight modifications to interact with Oracle Clusterware. Third party software for management of SAP systems will probably require additional configuration changes.

FUNCTIONALITY

A 'worked example' using SAPCTL appears later in this paper

SAPCTL performs the following:

- Updates the Oracle Clusterware configuration of the SAP high availability subsystem.
- 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.
- Queries the configuration of the SAP high availability subsystem.

- Relocates a resource to other hosting members on demand.

USAGE

```
sapctl <command> [<object>]  -sapsid <SAP_SID> [options]
        Command:
start|stop|status|config|create|remove|update|relocate
        Objects: abapvip|javavip|abapenq|javaenq|
                 abaprep|javarep|abapall|javaall|
                 asinstall|asinstvip|asinst|all

NOTE:
Create, remove, and update 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)

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 crs_stat

- config

Displays the configuration of the resources associated with all registered objects. Any valid object name is accepted, but information abaut all objects will be listed. This includes the following:

- The location of the Oracle Clusterware Home
- Resource identifier of the SAP Enqueue Service for Web AS ABAP
- The location of the SAP scripts for Web AS ABAP
- Resource identifier of the VIP of Web AS ABAP
- The Hosting Members for Web AS ABAP
- The Public Network Interface for Web AS ABAP
- Netmask for Web AS ABAP
- The Virtual IP Address for Web AS ABAP
- Resource identifier of the SAP Replication Service for Web AS ABAP
- Resource identifier of the SAP Enqueue Service for Web AS JAVA
- The location of the SAP scripts for Web AS JAVA
- Resource identifier of the VIP of Web AS JAVA
- The Hosting Members for Web AS JAVA
- The Public Network Interface for Web AS JAVA
- Netmask for Web AS JAVA
- The Virtual IP Address for Web AS JAVA
- Resource identifier of the SAP Replication Service for Web AS JAVA
- Resource identifier of SAP Web AS instances of type CI, DV
- The location of the SAP scripts for Web AS instance of type CI, DV
- Resource identifier of the VIP of SAP Web AS instances of type CI, DV
- The Hosting Members for Web AS instances of type CI, DV
- The Public Network Interface for Web AS instances of type CI, DV
- Netmask for Web AS instances of type CI, DV
- The Virtual IP Address for Web AS instances of type CI, DV

- 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:

Parameter	Description
-sapsid	The SID for the SAP instance
-if <adapter name>	The name of the public network adapter
-nm <netmask>	The network mask for the public network interface
-nodes <node1, … ,node(n)>	List of nodes in the cluster acting as hosting members
-abapenq <EnqInst>	The Enqueue Service identifier for Web AS ABAP
-abaprep <RepInst>	The Replication Service identifier for Web AS ABAP
-abapvip <IP Address>	A New IP Address for Web AS ABAP
-javaenq <EnqInst>	The Enqueue Service identifier for Web AS JAVA
-javarep <RepInst>	The Replication Service identifier for Web AS JAVA
-javavip <IP Address>	A New IP Address for Web AS JAVA
-asinst <SAP Instance>	The Instance name of SAP Central instance
-asinstvip <IP Address>	A New IP Address for SAP Central Instance

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.

- update

Updates the configuration of the SAP high availability subsystem. This operand is only supported for a single object 'all'. As VIP resources (abapvip, javavip or asinstvip) are possible targets the update command must be executed by user 'root' because of required privileges. Changes to the hosting members are valid for all registered resources. The following options are supported:

Parameter	Description
-abapvip <IP Address>	Updates the VIP value for the VIP used by the Enqueue Service of Web AS ABAP.
-javavip <IP Address>	Updates the VIP value for the VIP used by the Enqueue Service of Web AS ABAP.
-asinstvip <IP Address>	Updates the VIP value for the VIP used by the SAP instance (Type CI or DV)
-a <node>	Adds one node as placement candidates for SAPCTL's Oracle Clusterware resources. This option enables you to add nodes to the cluster after the installation.
-d <node>	Removes one node as placement candidates for SAPCTL's Oracle Clusterware resources. This option enables you to remove nodes from the cluster after installation.

- 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 dependend resources defined. If a resource with dependency to a VIP is specified, then the associated VIP resource gets relocated as well.

Parameter	Description
-abaprep <Instance name>	Name of the AB AP Replication instance.
-javarep <Instance name>	Name of the JAVA Replication instance.
-abapenq <Instance name>	Name of the AB AP Replication instance. The VIP resource for ABAP is relocated as well.
-javapenq <Instance name>	Name of the JAVA Replication instance. The VIP resource for JAVA is relocated as well.
-asinst <Instance name>	Name of the CI or DV instance. The VIP resource for the instance is relocated as well.
-to <node>	Tells the node where the resource should run after relocation.

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:
http://www.oracle.com/technology/products/database/clustering

And as a reminder - For details about configuring SAP with Oracle RAC refer to the white paper 'Configuration of SAP NetWeaver for Oracle Database 10g Real Application Clusters' in folder media library on the SAP marketplace website at: http://service.sap.com/dbaora .

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
- ORA_CRS_HOME environment variable is set to your Oracle Clusterware home.

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

Required Information	Description	Typical Example
SAP_SID	The name of the SAP Instance containing the SAP services that require protection	RAC
interface	The name of the public network interface	eth0
netmask	The IP network Netmask	255.255.240.0
hosting_members	The list of nodes that are capable of running the SAP services	sapnode1,sapnode2
abap_enqueue	The name of the SAP enqueue service of Web AS ABAP	ASCS01
abap_replication	The name of the SAP replication service of Web AS ABAP	ERS01
IP address ABAP	A new IP address that will be bound to the public network interface for Web AS ABAP	212.16.1.233
java_enqueue	The name of the SAP enqueue service of Web AS JAVA	SCS02
abap_replication	The name of the SAP replication service of Web AS ABAP	ERS02
IP address JAVA	A new IP address that will be bound to the public network interface for Web AS JAVA	212.16.1.234
Instance name	The name of the SAP instance of type CI or DV	DVEBMGS00
IP address Instance	A new IP address that will be bound to the public network interface for SAP instance	212.16.1.235

The SAPCTL utility

```
[root@oracx2 bin]# sapctl
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
USAGE:
sapctl <command> [<object>]  -sapsid <SAP_SID> [options]
        command : start|stop|status|config|create|remove
                 |update|relocate
        objects : abapvip|javavip|abapenq|javaenq|abaprep|javarep
                 |abapall|javaall|asinstall|asinstvip|asinst|all
NOTE:
create, remove, and update 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 -nm
255.255.255.0 -nodes oracx1,oracx2,oracw1,oracw2 -abapvip
140.86.242.63 -abapenq ASCS01 -abaprep ERS01 -javavip 140.86.242.64 -
javaenq SCS02 -javarep ERS02 -asinstvip 140.86.242.65 -asinst
DVEBMGS00
Production Copyright 2005, 2006, Oracle.  All rights reserved
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 **ERS01** 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.

The SAP enqueue service for Web AS JAVA called **SCS02** and the Replication service called **ERS02** 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.

The SAP Central Instance called **DVEBMGS00** be 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.

Running this command creates eight Oracle Clusterware resources:
Two application VIP´s for ABAP and JAVA, two Enqueue resources and two Replication resources. For the SAP Central instance tere is a instance resorce and an associated VIP resource. The script does not automatically start these resources.

To see which SAP Instances have been protected by Oracle Clusterware

```
[root@oracx2 bin]# sapctl config
Production Copyright 2005, 2006, Oracle.  All rights reserved
SAP_SIDs protected by the Oracle Clusterware are :
RAC
```

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

Modify the Standard SAP scripts

The sample SAP scripts must be modified according to the details in Appendix 1 of this paper

You must now modify the standard SAP scripts to enable them to work with Oracle Clusterware. See 'Appendix 1– Standard SAP Script Modifications'.

You must ensure that you complete this procedure before attempting to start the SAP resources that the Oracle Clusterware protects.

Check the Configuration

```
[root@oracx2 bin]# sapctl config all -sapsid RAC
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
CRS_HOME=/oracle/CRS/1020_32
Resource SAP ABAP Enqueue service = sap.RAC.ASCS01.abap.enq
Location of CRS SAP scripts = /usr/sap/RAC/SYS/exe/run
Resource SAP ABAP VIP = sap.RAC.abap.vip
Hosting Members = oracx1 oracx2 oracw1 oracw2
Interface = eth0
Netmask = 255.255.255.0
IP Address = 140.86.242.63
Resource SAP ABAP Replication service = sap.RAC.ERS01.0.abap.rep
Location of CRS SAP scripts = /usr/sap/RAC/SYS/exe/run
Resource SAP JAVA Enqueue service = sap.RAC.SCS02.java.enq
Location of CRS SAP scripts = /usr/sap/RAC/SYS/exe/run
Resource SAP JAVA VIP = sap.RAC.java.vip
Hosting Members = oracx1 oracx2 oracw1 oracw2
Interface = eth0
Netmask = 255.255.255.0
IP Address = 140.86.242.64
Resource SAP JAVA Replication service = sap.RAC.ERS02.0.java.rep
Location of CRS SAP scripts = /usr/sap/RAC/SYS/exe/run
Resource SAP instance DVEBMGS00 = sap.RAC.DVEBMGS00.inst
Location of CRS SAP scripts = /usr/sap/RAC/SYS/exe/run
Resource SAP instance VIP for DVEBMGS00 = sap.RAC.DVEBMGS00.inst.vip
Hosting Members = oracx1 oracx2 oracw1 oracw2
Interface = eth0
Netmask = 255.255.255.0
IP Address = 140.86.242.65
```

You can check the configuration values for a protected SAP instance using the config parameter.

Check the Status of the Oracle Clusterware Protected SAP resources

```
[root@oracx2 bin]# sapctl status all –sapsid RAC
Production Copyright 2005, 2006, Oracle.  All rights reserved
SAP ABAP Enqueue service is OFFLINE
SAP ABAP VIP is OFFLINE
SAP JAVA Enqueue service is OFFLINE
SAP JAVA VIP is OFFLINE
SAP ABAP Replication service is OFFLINE
SAP JAVA Replication service is OFFLINE
SAP instance DVEBMGS00 is OFFLINE
SAP instance VIP is OFFLINE
```

To check the status of the Oracle Clusterware resources that protect the SAP services use the status command.

Start the Oracle Clusterware Protected SAP resources

```
[root@oracx2 bin]# sapctl start all –sapsid RAC
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
Starting SAP ABAP Enqueue service
Starting SAP ABAP VIP
Starting SAP JAVA Enqueue service
Starting SAP JAVA VIP
Starting SAP ABAP Replication service
Starting SAP JAVA Replication service
Starting SAP instance DVEBMGS00
Starting SAP instance VIP for DVEBMGS00
SAP ABAP Enqueue service is ONLINE on oracx1
SAP ABAP VIP is ONLINE on oracx1
SAP JAVA Enqueue service is ONLINE on oracx1
SAP JAVA VIP is ONLINE on oracx
SAP ABAP Replication service is ONLINE on oracx2
SAP JAVA Replication service is ONLINE on oracx2
SAP instance DVEBMGS00 is ONLINE on oracx1
SAP instance VIP for 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
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
SAP ABAP Enqueue service is ONLINE on oracx1
SAP ABAP VIP is ONLINE on oracx1
SAP JAVA Enqueue service is ONLINE on oracx1
SAP JAVA VIP is ONLINE on oracx
SAP ABAP Replication service is ONLINE on oracx2
SAP JAVA Replication service is ONLINE on oracx2
SAP instance DVEBMGS00 is ONLINE on oracx1
SAP instance VIP for 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
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
Stopping SAP ABAP Enqueue service
Stopping SAP ABAP VIP
SAP ABAP Enqueue service is OFFLINE
SAP ABAP VIP is OFFLINE
Done
```

You can use the stop command to stop the Enqueue service. This will also stop the associated VIP.

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

```
[root@oracx2 bin]# sapctl start abapenq –sapsid RAC
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
Starting SAP ABAP Enqueue service
Starting SAP ABAP VIP
SAP ABAP Enqueue service is ONLINE on oracx2
SAP ABAP VIP 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.

Change the Config

```
[root@oracx2 bin]# sapctl update
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
USAGE:
sapctl update –sapsid <SAP_SID> [-a <node>] [-d <node>] [-abapvip
<IP_Address>] [-javavip <IP_Address>]
sapctl update –sapsid <SAP_SID> –asinst <SAP instancename> [-a <node>]
[-d <node>] [-asipaddress <IP_Address>]
```

You can use the update command to modify the nodes that the Oracle Clusterware protected SAP services are allowed to run on. You can also use these commands either to add or to delete nodes or to modify the IP address of the Virtual IP resource.

Relocate SAP resources to other hosting member

```
[root@oracx2 bin]# sapctl relocate –sapsid RAC –abaprep ERS01 –to
oracw1
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
Stopping SAP ABAP replication service
Relocating SAP ABAP Replication Service to node oracw1
Done
```

You can use the reolcate command to move SAP resources to other hosting members. You can´t 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
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
Stopping SAP instance VIP for DVEBMGS00
Stopping SAP instance for DVEBMGS00
Stopping SAP JAVA Replication service
Stopping SAP ABAP Replication service
Stopping SAP JAVA VIP
Stopping SAP JAVA Enqueue service
Stopping SAP ABAP VIP
Stopping SAP ABAP Enqueue service
SAP ABAP Enqueue service is OFFLINE
SAP ABAP VIP is OFFLINE
SAP JAVA Enqueue service is OFFLINE
SAP JAVA VIP is OFFLINE
SAP ABAP Replication service is OFFLINE
SAP JAVA Replication service is OFFLINE
SAP instance DVEBMGS00 is OFFLINE
SAP instance VIP for 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.

Check the Status after the stop

```
[root@oracx2 bin]# sapctl status all -sapsid RAC
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
SAP ABAP Enqueue service is OFFLINE
SAP ABAP VIP is OFFLINE
SAP JAVA Enqueue service is OFFLINE
SAP JAVA VIP is OFFLINE
SAP ABAP Replication service is OFFLINE
SAP JAVA Replication service is OFFLINE
SAP instance DVEBMGS00 is OFFLINE
SAP instance VIP for DVEBMGS00 is OFFLINE
```

After stopping all of the Oracle Clusterware resources, you will see that they are all marked as 'offline'.

Remove Oracle Clusterware protection for SAP resources

```
[root@oracx2 bin]# sapctl remove all -sapsid RAC
Production Copyright 2005, 2006, 2007, 2008 Oracle.  All rights
reserved
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 – STANDARD SAP SCRIPT MODIFICATIONS

There are two scripts that require changes before the Oracle Clusterware protected resources can start.

Script Name	Location
startsap	/usr/sap/<SID>/SYS/exe/run
stopsap	/usr/sap/<SID>/SYS/exe/run

Copies of the original scripts should be made prior to these changes.

Prior to modifying these scripts copies of original startsap and stopsap scripts should be made. The original scripts should be copied to startsap.orig and stopsap.orig.

Script: startsap

Location: /usr/sap/<SID>/SYS/exe/run directory.

```
#!/bin/sh
#----------------------------------------------------------------------
-----
#
# @(#) $Id: //bc/700-
1_REL/src/ins/SAPINST/impl/tpls/ind/ind/startsap#30 $
#
# NAME :
#      startsap
#
# PURPOSE :
#      Start SAP Database and SAP Instances
#
(---- cut here ----)


#=====================================================================
#
# FUNCTION: start_instance_do
#
# PURPOSE: starts the SAP instance doing
#

start_instance_do()
{
  setTrace "##### start_instance_do() start"
  setTrace "start_instance_do ${1}"

# Set START_PROFILE
  START_PROFILE=START_${1}_${HOSTNAME}
  setTrace "START_PROFILE=$START_PROFILE";

# Set instance specific sapstart
  SAPSTART_INSTANCE_SRV="/usr/sap/$SAPSYSTEMNAME/${1}/exe/sapstartsrv"
  SAPSTART_INSTANCE="/usr/sap/$SAPSYSTEMNAME/${1}/exe/sapstart"
  SAPSTART_INSTANCE_PATH="/usr/sap/$SAPSYSTEMNAME/${1}/exe"

  setTrace "SAPSTART_INSTANCE=$SAPSTART_INSTANCE"
  setTrace "SAPSTART=$SAPSTART"
  setTrace "SAPSTART_INSTANCE_PATH=$SAPSTART_INSTANCE_PATH"

# SET LOGFILE
  LOGFILE="${R3S_LOGDIR}/`basename ${0}`_${1}.log"
  setTrace "LOGFILE=$LOGFILE";

  init_log $0 $*;

  printf "\nStarting SAP Instance ${1}\n" | tee -a $LOGFILE
  echo "------------------------------" | tee -a $LOGFILE
  echo " Startup-Log is written to $LOGFILE"

  save_path;

  start_instance_service
```

Add this :

```
  if [ -x $SAPSTART_INSTANCE ]; then
      $SAPSTART_INSTANCE pf=$PROFILE_DIR/$START_PROFILE  >> $LOGFILE
2>&1
  else
      $SAPSTART pf=$PROFILE_DIR/$START_PROFILE  >> $LOGFILE 2>&1
  fi
  returncode=$?
  restore_path;

  case $returncode in
  0) echo " Instance on host $LOCAL_HOST started" | tee -a $LOGFILE
     sap2crs start  $SAPSYSTEMNAME ${1};;
  1) echo " Instance already running"             | tee -a $LOGFILE;;
  *) echo " Startup of Instance failed"           | tee -a $LOGFILE
    printf " See $LOGFILE for details\n\n"
    exit 5;;
  esac

  setTrace "##### start_instance_do() done"
}
(---- cut here )
```

Script: stopsap

Location: /usr/sap/<SID>/SYS/exe/run directory.

```
#!/bin/sh
#----------------------------------------------------------------------
-----
#
# @(#) $Id: //bc/700-
1_REL/src/ins/SAPINST/impl/tpls/ind/ind/stopsap#25 $
#
# NAME :
#      stopsap
#
# PURPOSE :
#      Stop SAP Database and SAP Instances
#
(---- cut here ----)

#=====================================================================
#
# FUNCTION: stop_instance_do
#
# PURPOSE:  stop instance doing
#
#
stop_instance_do()
{
  setTrace "##### stop_instance_do() start"
  setTrace "stop_instance_do ${1}"

  # Set START_PROFILE
  START_PROFILE=START_${1}_${HOSTNAME}
  setTrace "START_PROFILE=$START_PROFILE";

  # SET LOGFILE
  LOGFILE="${R3S_LOGDIR}/`basename ${0}`_${1}.log"
  setTrace "LOGFILE=$LOGFILE";

  init_log $0 $*;

  printf "\nStopping the SAP instance $1 \n"  | tee -a $LOGFILE
  echo "----------------------------------"  | tee -a $LOGFILE
  echo " Shutdown-Log is written to $LOGFILE"

  # STOP INSTANCE
  # check for TRX installation
  TTRX=`echo "$1" | awk '/^TRX[0-9][0-9]$/{print $1}'`

  if [ -n "$TTRX" ]; then
    if [ -f /usr/sap/$SAPSYSTEMNAME/$1/$HOSTNAME/trace/kill.sap ];
then
      SAP_PID=`awk '{print $1}'
/usr/sap/$SAPSYSTEMNAME/${1}/$HOSTNAME/trace/sapstart.sem`
      SAP_SHM=`awk '{print $2}'
/usr/sap/$SAPSYSTEMNAME/${1}/$HOSTNAME/trace/sapstart.sem`
      eval sh /usr/sap/$SAPSYSTEMNAME/${1}/$HOSTNAME/trace/kill.sap >>
$LOGFILE 2>&1
      returncode=$?
    else
      echo " Instance ${1} was not running!" | tee -a $LOGFILE
    fi
  else
    if [ -f /usr/sap/$SAPSYSTEMNAME/${1}/work/kill.sap ]; then
      SAP_PID=`awk '{print $1}'
/usr/sap/$SAPSYSTEMNAME/${1}/work/sapstart.sem`
      SAP_SHM=`awk '{print $2}'
/usr/sap/$SAPSYSTEMNAME/${1}/work/sapstart.sem`
      eval sh /usr/sap/$SAPSYSTEMNAME/$i/work/kill.sap >> $LOGFILE
2>&1
      returncode=$?

      setTrace " start /usr/sap/$SAPSYSTEMNAME/${1}/work/kill.sap"
      setTrace "returncode=$returncode"

    else
      echo " Instance ${1} was not running!" | tee -a $LOGFILE
    fi
  fi
```

Add this :

```
  case $returncode in
    0) echo " Instance on host $LOCAL_HOST stopped" | tee -a $LOGFILE
       printf " Waiting for cleanup of resources....."  | tee -a
$LOGFILE
       case `uname` in
         Linux*)
           ps_command="ps ax"
           ;;
         *)
           ps_command="ps -e"
           ;;
       esac
       while $ps_command | awk '{print $1}' | grep "^$SAP_PID$"
>/dev/null || \
         ipcs |awk '/^m / {print $2}' | grep "^$SAP_SHM$" >/dev/null;
do
         printf "."
         sleep 2
       done
       echo ""  | tee -a $LOGFILE
       sap2crs stop $SAPSYSTEMNAME ${1}
       ;;
    *) echo " Shutdown of instance failed" | tee -a $LOGFILE
        exit 7;;
  esac

  setTrace "##### stop_instance_do() done"

(---- cut here ----)
```

APPENDIX 2 – SAMPLE PROFILE SCRIPTS - ENQUEUE SERVICE FOR WEB AS ABAP (ASCS)

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS ABAP on a two-node cluster. The hostnames of the cluster nodes are oracx1 and oracx2 in this example.

Script Name	Location
RAC_ASCS01_oracx1	/usr/sap/RAC/SYS/profile
RAC_ASCS01_oracx2	/usr/sap/RAC/SYS/profile
START_ASCS01_oracx1	/usr/sap/RAC/SYS/profile
START_ASCS01_oracx2	/usr/sap/RAC/SYS/profile

Script: DEFAULT.PFL

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPSYSTEMNAME = RAC
SAPGLOBALHOST = sap_abapvip
#----------------------------------------------------------------------
# rdisp/mshost must resolve to the IP address given as –abapvip
# parameter in sapctl 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 sapctl create command
#----------------------------------------------------------------------
enque/serverhost = sap_abapvip
enque/serverinst = 01
login/system_client = 001
#-----------------------------------------------------------------------
# SAP Central Service Instance for J2EE
#-----------------------------------------------------------------------
#----------------------------------------------------------------------
# j2ee/scs/host must resolve to the IP address given as –javavip
# parameter in sapctl create command
#----------------------------------------------------------------------
j2ee/scs/host = sap_javavip
j2ee/scs/system = 02
j2ee/ms/port = 3902
#-----------------------------------------------------------------------
# SAP Central Service Instance for J2EE
#-----------------------------------------------------------------------
SAPDBHOST = ora_sap_vip
j2ee/dbtype = ora
j2ee/dbname = NW4
j2ee/dbhost = ora_sap_vip
rdisp/bufrefmode = sendon,exeauto
DIR_PUT = /usr/sap/$(SAPSYSTEMNAME)/put
```

Script: RAC_ASCS01_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx1
SAPLOCALHOSTFULL=oracx1.de.oracle.com
INSTANCE_NAME=ASCS01
SAPSYSTEM=00
SAPSYSTEMNAME=RAC
rdisp/myname=oracx1_RAC_01
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: RAC_ASCS01_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx2
SAPLOCALHOSTFULL=oracx2.de.oracle.com
INSTANCE_NAME=ASCS01
SAPSYSTEM=00
SAPSYSTEMNAME=RAC
rdisp/myname=oracx2_RAC_01
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: START_ASCS01_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Start profile START_ASCS01_oracx1
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEM          =00
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ASCS01

#-----------------------------------------------------------------------
# start SCSA administration
#-----------------------------------------------------------------------

Execute_00 =local $(DIR_EXECUTABLE)/sapmscsa -n
pf=$(DIR_PROFILE)/RAC_ASCS01_oracx1

#-----------------------------------------------------------------------
# start syslog send daemon
#-----------------------------------------------------------------------

_SE                =se.sapRAC_ASCS01_oracx1
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/rslgsend $(_SE)
Start_Program_01   =local $(_SE) -F pf=$(DIR_PROFILE)/RAC_ASCS01_oracx1

#-----------------------------------------------------------------------
# start syslog collector daemon
#-----------------------------------------------------------------------

_CO                =co.sapRAC_ASCS01_oracx1
Execute_02         =local ln -s -f $(DIR_EXECUTABLE)/rslgcoll $(_CO)
Start_Program_02   =local $(_CO) -F pf=$(DIR_PROFILE)/RAC_ASCS01_oracx1

#-----------------------------------------------------------------------
# start message server
#-----------------------------------------------------------------------

_MS                =ms.sapRAC_ASCS01_oracx1
Execute_03         =local ln -s -f $(DIR_EXECUTABLE)/msg_server $(_MS)
Start_Program_03   =local $(_MS) pf=$(DIR_PROFILE)/RAC_ASCS01_oracx1

#-----------------------------------------------------------------------
# start enqueue server
#-----------------------------------------------------------------------

_EN                =en.sapRAC_ASCS01_oracx1
Execute_04         =local ln -s -f $(DIR_EXECUTABLE)/enserver $(_EN)
Start_Program_04   =local $(_EN) pf=$(DIR_PROFILE)/RAC_ASCS01_oracx1

#-----------------------------------------------------------------------
```

Script: START_ASCS01_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Start profile START_ASCS01_oracx2
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEM          =00
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ASCS01

#-----------------------------------------------------------------------
# start SCSA administration
#-----------------------------------------------------------------------

Execute_00 =local $(DIR_EXECUTABLE)/sapmscsa -n
pf=$(DIR_PROFILE)/RAC_ASCS01_oracx2

#-----------------------------------------------------------------------
# start syslog send daemon
#-----------------------------------------------------------------------

_SE                =se.sapRAC_ASCS01_oracx2
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/rslgsend $(_SE)
Start_Program_01   =local $(_SE) -F pf=$(DIR_PROFILE)/RAC_ASCS01_oracx2

#-----------------------------------------------------------------------
# start syslog collector daemon
#-----------------------------------------------------------------------

_CO                =co.sapRAC_ASCS01_oracx2
Execute_02         =local ln -s -f $(DIR_EXECUTABLE)/rslgcoll $(_CO)
Start_Program_02   =local $(_CO) -F pf=$(DIR_PROFILE)/RAC_ASCS01_oracx2

#-----------------------------------------------------------------------
# start message server
#-----------------------------------------------------------------------

_MS                =ms.sapRAC_ASCS01_oracx2
Execute_03         =local ln -s -f $(DIR_EXECUTABLE)/msg_server $(_MS)
Start_Program_03   =local $(_MS) pf=$(DIR_PROFILE)/RAC_ASCS01_oracx2

#-----------------------------------------------------------------------
# start enqueue server
#-----------------------------------------------------------------------

_EN                =en.sapRAC_ASCS01_oracx2
Execute_04         =local ln -s -f $(DIR_EXECUTABLE)/enserver $(_EN)
Start_Program_04   =local $(_EN) pf=$(DIR_PROFILE)/RAC_ASCS01_oracx2

#-----------------------------------------------------------------------
```

APPENDIX 3 – SAMPLE PROFILE SCRIPTS - REPLICATION SERVICE FOR WEB AS ABAP (ERS)

The following is an example of the profile scripts for a standalone Replication Service for SAP Web AS ABAP on a two-node cluster. The hostnames of the cluster nodes are oracx1 and oracx2 in this example.

Script Name	Location
RAC_ERS01_oracx1	/usr/sap/RAC/SYS/profile
RAC_ERS01_oracx2	/usr/sap/RAC/SYS/profile
START_ERS01_oracx1	/usr/sap/RAC/SYS/profile
START_ERS01_oracx2	/usr/sap/RAC/SYS/profile

Script: RAC_ERS01_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx1
SAPLOCALHOSTFULL=oracx1.de.oracle.com
INSTANCE_NAME=ERS01
SAPSYSTEMNAME=RAC
rdisp/myname=oracx1_ERS_01
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: RAC_ERS01_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx2
SAPLOCALHOSTFULL=oracx2.de.oracle.com
INSTANCE_NAME=ERS01
SAPSYSTEMNAME=RAC
rdisp/myname=oracx2_ERS_01
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: START_ERS01_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Startprofil START_ERS01_oracx1
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ERS01

#-----------------------------------------------------------------------
# start enqueue replication server
#-----------------------------------------------------------------------

_ERS               =enr.sapRAC_ERS01_oracx1
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/enrepserver $(_ERS)
Start_Program_01   =local $(_ERS) pf=$(DIR_PROFILE)/RAC_ERS01_oracx1

#-----------------------------------------------------------------------
```

Script: START_ERS01_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Startprofil START_ERS01_oracx2
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ERS01

#-----------------------------------------------------------------------
# start enqueue replication server
#-----------------------------------------------------------------------

_ERS               =enr.sapRAC_ERS01_oracx2
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/enrepserver $(_ERS)
Start_Program_01   =local $(_ERS) pf=$(DIR_PROFILE)/RAC_ERS01_oracx2

#-----------------------------------------------------------------------
```

APPENDIX 4 – SAMPLE PROFILE SCRIPTS - ENQUEUE SERVICE FOR WEB AS JAVA (SCS)

The following is an example of the profile scripts for a standalone Enqueue Service for the SAP Web AS JAVA on a two-node cluster. The hostnames of the cluster nodes are oracx1 and oracx2 in this example.

Script Name	Location
RAC_SCS02_oracx1	/usr/sap/RAC/SYS/profile
RAC_SCS02_oracx2	/usr/sap/RAC/SYS/profile
START_SCS02_oracx1	/usr/sap/RAC/SYS/profile
START_SCS02_oracx2	/usr/sap/RAC/SYS/profile

Script: RAC_SCS02_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx1
SAPLOCALHOSTFULL=oracx1.de.oracle.com
INSTANCE_NAME=SCS02
SAPSYSTEM=00
SAPSYSTEMNAME=RAC
rdisp/myname=oracx1_SCS_02
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: RAC_SCS02_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx2
SAPLOCALHOSTFULL=oracx2.de.oracle.com
INSTANCE_NAME=SCS02
SAPSYSTEM=00
SAPSYSTEMNAME=RAC
rdisp/myname=oracx2_SCS_02
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/enrep/keepalive_count=0
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: START_SCS02_oracx1 and START_SCS02_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPSYSTEMNAME = RAC
SAPSYSTEM = 00
INSTANCE_NAME = SCS02
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = ora_sap_vip
DIR_PROFILE = $(DIR_INSTALL)/profile
_PF = $(DIR_PROFILE)/RAC_SCS02_ora_sap_vip
SETENV_00 = LD_LIBRARY_PATH=$(DIR_LIBRARY):%(LD_LIBRARY_PATH)
SETENV_01 = SHLIB_PATH=$(DIR_LIBRARY):%(SHLIB_PATH)
SETENV_02 = LIBPATH=$(DIR_LIBRARY):%(LIBPATH)
#-----------------------------------------------------------------------
# Copy SAP Executables
#-----------------------------------------------------------------------
_CPARG0 = list:$(DIR_CT_RUN)/scs.lst
Execute_00 = immediate $(DIR_CT_RUN)/sapcpe$(FT_EXE) pf=$(_PF) $(_CPARG0)
OS_UNICODE = uc
#-----------------------------------------------------------------------
# Start SAP messaging service
#-----------------------------------------------------------------------
_MS = ms.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_01 = local rm -f $(_MS)
Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/msg_server$(FT_EXE) $(_MS)
Start_Program_00 = local $(_MS) pf=$(DIR_PROFILE)/RAC_SCS02_ora_sap_vip
#-----------------------------------------------------------------------
# Start SAP locking service
#-----------------------------------------------------------------------
_EN = en.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_03 = local rm -f $(_EN)
Execute_04 = local ln -s -f $(DIR_EXECUTABLE)/enserver$(FT_EXE) $(_EN)
Start_Program_01 = local $(_EN) pf=$(DIR_PROFILE)/RAC_SCS02_ora_sap_vip
```

APPENDIX 5 – SAMPLE PROFILE SCRIPTS - REPLICATION SERVICE FOR WEB AS JAVA (ERS)

The following is an example of the profile scripts for a standalone Replication Service for SAP Web AS JAVA on a two-node cluster. The hostnames of the cluster nodes are oracx1 and oracx2 in this example.

Script Name	Location
RAC_ERS02_oracx1	/usr/sap/RAC/SYS/profile
RAC_ERS02_oracx2	/usr/sap/RAC/SYS/profile
START_ERS02_oracx1	/usr/sap/RAC/SYS/profile
START_ERS02_oracx2	/usr/sap/RAC/SYS/profile

Script: RAC_ERS02_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx1
SAPLOCALHOSTFULL=oracx1.de.oracle.com
INSTANCE_NAME=ERS02
SAPSYSTEMNAME=RAC
rdisp/myname=oracx1_ERS_02
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/serverinst = 02
enque/enrep/keepalive_count=0
rdisp/msserv = 0
rdisp/msserv_internal = 3902
ms/standalone = 1
ms/server_port_0 = PROT=HTTP,PORT=81$$
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: RAC_ERS02_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPLOCALHOST=oracx2
SAPLOCALHOSTFULL=oracx2.de.oracle.com
INSTANCE_NAME=ERS02
SAPSYSTEMNAME=RAC
rdisp/myname=oracx2_ERS_02
enque/table_size=4096
enque/process_location=LOCAL
enque/server/internal_replication=true
enque/server/replication=true
enque/serverinst = 02
enque/enrep/keepalive_count=0
rdisp/msserv = 0
rdisp/msserv_internal = 3902
ms/standalone = 1
ms/server_port_0 = PROT=HTTP,PORT=81$$
#for performance tuning
enque/server/threadcount=1
# these are needed to prevent the pools from being created
ipc/shm_psize_16=0
ipc/shm_psize_24=0
ipc/shm_psize_34=0
ipc/shm_psize_66=0
```

Script: START_ERS02_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Startprofil START_ERS02_oracx1
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ERS02

#-----------------------------------------------------------------------
# start enqueue replication server
#-----------------------------------------------------------------------

_ERS               =enr.sapRAC_ERS02_oracx1
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/enrepserver $(_ERS)
Start_Program_01   =local $(_ERS) pf=$(DIR_PROFILE)/RAC_ERS02_oracx1

#-----------------------------------------------------------------------
```

Script: START_ERS02_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Startprofil START_ERS02_oracx2
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME      =RAC
INSTANCE_NAME      =ERS02

#-----------------------------------------------------------------------
# start enqueue replication server
#-----------------------------------------------------------------------

_ERS               =enr.sapRAC_ERS02_oracx2
Execute_01         =local ln -s -f $(DIR_EXECUTABLE)/enrepserver $(_ERS)
Start_Program_01   =local $(_ERS) pf=$(DIR_PROFILE)/RAC_ERS02_oracx2

#-----------------------------------------------------------------------
```

APPENDIX 6 – SAMPLE PROFILE SCRIPTS – SAP CENTRAL INSTANCE

The following is an example of the profile scripts for a SAP Central Instance on a two-node cluster. The hostnames of the cluster nodes are oracx1 and oracx2 in this example.

Script Name	Location
RAC_DVEBMGS00_oracx1	/usr/sap/RAC/SYS/profile
RAC_DVEBMGS00_oracx2	/usr/sap/RAC/SYS/profile
START_DVEBMGS00_oracx1	/usr/sap/RAC/SYS/profile
START_DVEBMGS00_oracx2	/usr/sap/RAC/SYS/profile

Script: RAC_DVEBMGS00_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPSYSTEMNAME = RAC
SAPSYSTEM = 00
INSTANCE_NAME = DVEBMGS00
dbs/ora/tnsname=NW4_DVEBMGS00_ora_sap_vip
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = oracx1
jstartup/trimming_properties = off
jstartup/protocol = on
jstartup/vm/home = /ocfs2_gp/JAVA/j2sdk1.4.2_15
jstartup/max_caches = 500
jstartup/release = 700
jstartup/instance_properties =
$(jstartup/j2ee_properties):$(jstartup/sdm_properties)
j2ee/dbdriver = /oracle/client/10x_32/instantclient/ojdbc14.jar
PHYS_MEMSIZE = 512
exe/saposcol = $(DIR_CT_RUN)/saposcol
rdisp/wp_no_dia = 6
rdisp/wp_no_btc = 3
exe/icmbnd = $(DIR_CT_RUN)/icmbnd
rdisp/j2ee_start_control = 1
rdisp/j2ee_start = 1
rdisp/j2ee_libpath = $(DIR_EXECUTABLE)
exe/j2ee = $(DIR_EXECUTABLE)/jcontrol$(FT_EXE)
rdisp/j2ee_timeout = 600
rdisp/frfc_fallback = on
icm/HTTP/j2ee_0 = PREFIX=/,HOST=localhost,CONN=0-500,PORT=5$$00
icm/server_port_0 = PROT=HTTP,PORT=80$$
rdisp/wp_no_vb = 1
rdisp/wp_no_vb2 = 1
rdisp/wp_no_spo = 1
ipc/shm_psize_10 = 76000000
ipc/shm_psize_40 = 62000000


#------------------------------------------------------------
# Jcontrol: Migrated Profile Parameter
#            create at Mon Aug 27 12:04:53 2007
#------------------------------------------------------------
j2ee/instance_id = ID1261296
#------------------------------------------------------------
```

Script: RAC_DVEBMGS00_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
SAPSYSTEMNAME = RAC
SAPSYSTEM = 00
INSTANCE_NAME = DVEBMGS00
dbs/ora/tnsname=NW4_DVEBMGS00_ora_sap_vip
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = oracx2
jstartup/trimming_properties = off
jstartup/protocol = on
jstartup/vm/home = /ocfs2_gp/JAVA/j2sdk1.4.2_15
jstartup/max_caches = 500
jstartup/release = 700
jstartup/instance_properties =
$(jstartup/j2ee_properties):$(jstartup/sdm_properties)
j2ee/dbdriver = /oracle/client/10x_32/instantclient/ojdbc14.jar
PHYS_MEMSIZE = 512
exe/saposcol = $(DIR_CT_RUN)/saposcol
rdisp/wp_no_dia = 6
rdisp/wp_no_btc = 3
exe/icmbnd = $(DIR_CT_RUN)/icmbnd
rdisp/j2ee_start_control = 1
rdisp/j2ee_start = 1
```

```
rdisp/j2ee_libpath = $(DIR_EXECUTABLE)
exe/j2ee = $(DIR_EXECUTABLE)/jcontrol$(FT_EXE)
rdisp/j2ee_timeout = 600
rdisp/frfc_fallback = on
icm/HTTP/j2ee_0 = PREFIX=/,HOST=localhost,CONN=0-500,PORT=5$$00
icm/server_port_0 = PROT=HTTP,PORT=80$$
rdisp/wp_no_vb = 1
rdisp/wp_no_vb2 = 1
rdisp/wp_no_spo = 1
ipc/shm_psize_10 = 76000000
ipc/shm_psize_40 = 62000000


#------------------------------------------------------------
# Jcontrol: Migrated Profile Parameter
#             create at Mon Aug 27 12:04:53 2007
#------------------------------------------------------------
j2ee/instance_id = ID1261296
#------------------------------------------------------------
```

Script: START_DVEBMGS00_oracx1

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Start profile START_DVEBMGS00_oracx1
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME = RAC
SAPSYSTEM = 00
INSTANCE_NAME = DVEBMGS00
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = oracx1
DIR_PROFILE = $(DIR_INSTALL)/profile
_PF = $(DIR_PROFILE)/NW4_DVEBMGS12_oracx1
SETENV_00 = LD_LIBRARY_PATH=$(DIR_LIBRARY):%(LD_LIBRARY_PATH)
SETENV_01 = SHLIB_PATH=$(DIR_LIBRARY):%(SHLIB_PATH)
SETENV_02 = LIBPATH=$(DIR_LIBRARY):%(LIBPATH)
SETENV_03 = dbs_ora_tnsname=NW4_DVEBMGS12_ora_sapvip
#-----------------------------------------------------------------------
# Copy SAP Executables
#-----------------------------------------------------------------------
Execute_00 = immediate $(DIR_CT_RUN)/sapcpe$(FT_EXE) pf=$(_PF)
#-----------------------------------------------------------------------
# Start SCSA administration
#-----------------------------------------------------------------------
Execute_01 = local $(DIR_EXECUTABLE)/sapmscsa pf=$(_PF) -n
#-----------------------------------------------------------------------
# Start application server
#-----------------------------------------------------------------------
_DW = dw.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_02 = local rm -f $(_DW)
Execute_03 = local ln -s -f $(DIR_EXECUTABLE)/disp+work$(FT_EXE) $(_DW)
Start_Program_00 = local $(_DW) pf=$(_PF)
#-----------------------------------------------------------------------
# Start syslog collector daemon
#-----------------------------------------------------------------------
_CO = co.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_04 = local rm -f $(_CO)
Execute_05 = local ln -s -f $(DIR_EXECUTABLE)/rslgcoll $(_CO)
Start_Program_01 = local $(_CO) pf=$(_PF) -F
#-----------------------------------------------------------------------
# Start syslog send daemon
#-----------------------------------------------------------------------
_SE = se.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_06 = local rm -f $(_SE)
Execute_07 = local ln -s -f $(DIR_EXECUTABLE)/rslgsend $(_SE)
Start_Program_02 = local $(_SE) pf=$(_PF) -F
#-----------------------------------------------------------------------
# Start internet graphics server
#-----------------------------------------------------------------------
_IG = ig.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_08 = local rm -f $(_IG)
Execute_09 = local ln -s -f $(DIR_EXECUTABLE)/igswd_mt $(_IG)
Start_Program_03 = local $(_IG) -mode=profile pf=$(_PF)
```

Script: START_DVEBMGS00_oracx2

Location: /usr/sap/RAC/SYS/profile directory.

```
#.******************************************************************************
#.*                                                                            *
#.*       Start profile START_DVEBMGS00_oracx2
#.*                                                                            *
#.******************************************************************************
#-----------------------------------------------------------------------
SAPSYSTEMNAME = RAC
SAPSYSTEM = 00
INSTANCE_NAME = DVEBMGS00
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = oracx2
DIR_PROFILE = $(DIR_INSTALL)/profile
_PF = $(DIR_PROFILE)/NW4_DVEBMGS12_oracx2
SETENV_00 = LD_LIBRARY_PATH=$(DIR_LIBRARY):%(LD_LIBRARY_PATH)
SETENV_01 = SHLIB_PATH=$(DIR_LIBRARY):%(SHLIB_PATH)
SETENV_02 = LIBPATH=$(DIR_LIBRARY):%(LIBPATH)
SETENV_03 = dbs_ora_tnsname=NW4_DVEBMGS12_ora_sapvip
#-----------------------------------------------------------------------
# Copy SAP Executables
#-----------------------------------------------------------------------
Execute_00 = immediate $(DIR_CT_RUN)/sapcpe$(FT_EXE) pf=$(_PF)
#-----------------------------------------------------------------------
# Start SCSA administration
#-----------------------------------------------------------------------
Execute_01 = local $(DIR_EXECUTABLE)/sapmscsa pf=$(_PF) -n
#-----------------------------------------------------------------------
# Start application server
#-----------------------------------------------------------------------
_DW = dw.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_02 = local rm -f $(_DW)
Execute_03 = local ln -s -f $(DIR_EXECUTABLE)/disp+work$(FT_EXE) $(_DW)
Start_Program_00 = local $(_DW) pf=$(_PF)
#-----------------------------------------------------------------------
# Start syslog collector daemon
#-----------------------------------------------------------------------
_CO = co.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_04 = local rm -f $(_CO)
Execute_05 = local ln -s -f $(DIR_EXECUTABLE)/rslgcoll $(_CO)
Start_Program_01 = local $(_CO) pf=$(_PF) -F
#-----------------------------------------------------------------------
# Start syslog send daemon
#-----------------------------------------------------------------------
_SE = se.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_06 = local rm -f $(_SE)
Execute_07 = local ln -s -f $(DIR_EXECUTABLE)/rslgsend $(_SE)
Start_Program_02 = local $(_SE) pf=$(_PF) -F
#-----------------------------------------------------------------------
# Start internet graphics server
#-----------------------------------------------------------------------
_IG = ig.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_08 = local rm -f $(_IG)
Execute_09 = local ln -s -f $(DIR_EXECUTABLE)/igswd_mt $(_IG)
Start_Program_03 = local $(_IG) -mode=profile pf=$(_PF)
```

APPENDIX 7 – 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 check 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:

```
<homedir sapadm>/.sapctl.log
```

You can also examine the following file to troubleshoot issues with VIP resources:

```
${ORA_CRS_HOME}/log/<hostname>/racg/sap.${SAP_SID}.<abap|java>.
vip.log
```

In addition to the vip.log file, you can also examine the content of the CRSD log file:

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

APPENDIX 8 - 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 shell script that runs bin/sapctl.pl; it is included for convenience.
- ./bin/sapctl.pl – This is a Perl-based SAPCTL management utility implementation.
- ./doc/sapctl.doc – This is a copy of this document.
- ./templates/crssapactions.pl.sbs – This is a template for the action perl script used internally by SAPCTL.
- ./templates/sap2crs.sbs – This is a template for the 'SAP to crs' proxy script used internally by SAPCTL.
- ./templates/sapwrap.sbs – This is a template for the action script wrapper used internally by SAPCTL.

Providing High Availability for SAP Resources
July 2008
Version 3.0

Authors: David Austin, Mark Bauer, Andrey Gusev, Kurt Broeg, and Rajiv Jayaraman
Contributing Authors: Philip Newlan, Jan Klokkers

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com