

## PDB backup machine - general information

PDB backup machine (currently itrac330 alias pdb-backup) hosts several services critical for the PDB Database Service operation. Those services include:

- Legacy database monitoring
- RACMon - new database monitoring
- RMAN backup scheduler (for both PSS and DES databases)
- PDB recovery catalog export scheduler
- Backup reporting utility (used by DES exclusively)
- Backup validation utility
- Database Access Manager (DAM) for PSS
- Alert log agregation tool

The machine is also used as a central repository for the Oracle binaries and several scripts being used during database configuration. There are also scripts simplifying DAD and public TNS edition.

Because of its important role in the service the PDB backup machine requires reliable and robust hardware. At the same time the software running there requires quite significant disk space and CPU resource. Therefore the current choice is to run all the stuff mentioned about an a mid-range server similar to the ones used as RAC nodes with a small dedicated disk array configured with RAID5 and mounted under /data directory.

### Legacy database monitoring

PDB backup machine still hosts legacy monitoring tools used in the past for re-active database monitoring. Those tools are able to connect to a given database using SQL\*Plus and in case SQL\*Plus connection is not possible they try to connect to the machine hosting the database via SSH. When there is a problem with the database, the listener or the host the scripts do sent e-mail and GSM notifications.

Files:

```
$HOME/db_monitoring_scripts/script/monitor/*
$HOME/db_monitoring_scripts/script/logs/monitor/
$HOME/edit_monitoring
/etc/cron.d/pdb-monitoring.cron
```

### RACMon

RACMon is a new tool devoted to comprehensive monitoring of clustered Oracle databases. Similarly to the legacy monitoring tool it can connect both to a database and to the host hosting it but additionally it is also able to talk to ASM instances and Oracle clusterware. In case of problems the tool sends e-mail and GSM notifications.

Files:

```
$HOME/rac_mon/*
$HOME/rac_mon/conf/*
$HOME/rac_mon/logs/
$HOME/rac_mon/tmp/
/etc/cron.d/pdb-monitoring.cron
```

### RMAN backup scheduler

This tool allows convenient scheduling and performing of RMAN on-tape and on-disk backups. There are several versions of the scripts used to backup different database. The tool requires a quite complicated setup

consisting of inittab entries, crontab entries and appropriate per-database directory structure.

Files:

```
/backup5/scripts/*  
/backup5/scripts/etc/*  
/backup5/$DB_NAME/etc/access_rman_targ  
/backup5/$DB_NAME/etc/orauser  
/backup5/$DB_NAME/etc/RMAN.COPYTODISK.TAG  
/backup5/$DB_NAME/logs/archived/  
/etc/cron.d/pdb-backup5.cron  
/etc/inittab
```

## PDB recovery catalog export scheduler

This tool does daily exports of PDB's catalog which can be used in case of loss of the database where this catalog exists. Exports are scheduled with crontab and last seven successful exports are kept. Together with the tool there is also a list of DBIDs store.

Files:

```
/backup5/scripts/rman_catalog_export/rman_catalog_export.sh  
backup5/scripts/rman_catalog_export/rman_catalog_export_wrapper.sh  
/backup5/scripts/rman_catalog_export/dumpfiles/  
/backup5/scripts/rman_catalog_export/logs/  
/etc/cron.d/pdb-backup5.cron  
/backup5/scripts/rman_catalog_export/Example_DBID_list.txt
```

## Backup reporting utility

It is another tool being run from crontab. The tool is capable to analyze RMAN logs and to produce and publish HTML reports. For the time being it is used only for DES 9i backups as it has not been adjusted to 10g backup scripts, yet.

Files:

```
/backup5/scripts/rman_logs_parser  
/backup5/scripts/rman_summary  
/backup5/scripts/html/*  
/etc/cron.d/rman-backup-summary.cron
```

## Backup validation utility

The tool uses the 'restore database check logical validate' RMAN command to validate on-tape backups. It needs to be scheduled using crontab for each database separately. It requires the same directory structure under '/backup5/' directory as RMAN backup scripts.

Files:

```
/backup5/scripts/pdb-run-validate.sh  
/etc/cron.d/pdb-backup5.cron
```

## Database Access Manager (DAM)

The tool used for administering access privileges to PSS machines.

Files:

```
$HOME/dam/*
```

## Alert log agregation tool

The tool used to periodically retrieve and sent around the list of errors found in the alert logs on different instances of RAC databases.

Files:

```
$HOME/production/alert_log_merge.sh
/etc/cron.d/pdb-monitoring.cron
```

## Installation

- Choose 2 nodes and 1 disk array where to install pdb-backup cluster.
- Install appropriate OS version on the nodes. Install following extra RPMs:

```
openldap, openldap-clients, openldap-servers, perl-Convert-ASN1, perl-LDAP, cvs, wassh, wa
oracle-instantclient-basic, perl-MailTools, perl-DBD-Oracle, perl-Tk, perl-X11-Keyboard, p
```

- ◆ Modify appropriately Quattor profiles.
- The disk array configure with 1 big RAID 5 volume and create ext3 file system there. On the storage level as usual create a 1GB partition for clusterware registry and voting disk.
- Go ahead with cluster configuration and clusterware installation as described in the installation instructions.
- Install RAC software on the cluster nodes. The version of the installed RAC software should meet the following constraint:

```
RAC_version_on_pdb-backup <= min(RAC_version_of_existing_databases)
```

- ◆ Do not create a listener.
- Create an ext3 file system on the RAID5 device.

```
mkfs.ext3 /dev/mpath/itstorXXXXp1
```

- Create mount points for the file system created on the RAID5 device.

```
# on both nodes
sudo mkdir /data
sudo chown oracle:ci /data
```

- On both nodes unregister CRS targets (ONS,GSD,VIP) created during clusterware installation:

```
srvctl stop nodeapps -n <nodename>
sudo crs_unregister ora.<nodename>.ons
sudo crs_unregister ora.<nodename>.gsd
sudo crs_unregister ora.<nodename>.vip
```

- Configure the application VIP: create the VIP for pdb-backup (called pdbbackupvip) + register it + set it to run as root + allow oracle to start it + start it

```
# as root on both nodes:
crs_profile -create pdbbackupvip -t application -a $ORA_CRS_HOME/bin/usrvip -o oi=eth0,ov=
# as root on the first node:
```

## PdbBackupRecovery < PSSGroup < TWiki

```
crs_register pdbbackupvip
crs_setperm pdbbackupvip -o root
crs_setperm pdbbackupvip -u user:oracle:r-x
# as oracle on the first node:
crs_start pdbbackupvip
```

- Write an action script for filesystem /data see: action\_PDB\_data.scr: action script for Oracle CRS - filesystem handler.
- Place the action script in the \$ORA\_CRS\_HOME/crs/public/ directory on both nodes of the cluster.
- Make sure that the action script is owned by root and its primary group is also root
- Create profile and register CRS target for filesystem /data

```
# as root on both nodes:
crs_profile -create fs_data -t application -d "Filesystem data" -r pdbbackupvip -a $ORA_CR
# as root on the first node:
crs_register fs_data
crs_setperm fs_data -o root
crs_setperm fs_data -u user:oracle:r-x
# as oracle on the first node:
crs_start fs_data
```

- On both nodes in .bashrc files set the following extra environment variables:

```
export TNS_ADMIN=/ORA/dbs01/oracle/admin/network
export JAVA_HOME=$ORACLE_HOME/jdk/jre
```

- On the shared storage create the following directories:

```
# from the node that has the /data file system mounted as oracle
mkdir -p /data/admin/network
mkdir -p /data/backup5
mkdir -p /data/etc/cron.d
mkdir -p /data/etc/init.d
mkdir -p /data/home/dam
mkdir -p /data/home/db_monitoring_scripts
mkdir -p /data/home/oracle_binaries
mkdir -p /data/home/production
mkdir -p /data/home/rac_mon
mkdir -p /data/home/scripts
mkdir -p /data/home/secscan
mkdir -p /data/home/streams
mkdir -p /data/home/strmmon
mkdir -p /data/home/tns_download
mkdir -p /data/home/work
# on both nodes
mkdir -p /ORA/dbs01/oracle/admin/network
```

- Populate create directories either by copying their contents over from the old pdb-backup machine or by restoring from TSM.
- On both nodes create symbolic links:

```
# as oracle
ln -s /data/admin/network/tnsnames.ora /ORA/dbs01/oracle/admin/network/tnsnames.ora
ln -s /ORA/dbs01/oracle/admin/network/tnsnames.ora $ORACLE_HOME/network/admin/tnsnames.ora
ln -s /data/home/dam $HOME/dam
ln -s /data/home/db_monitoring_scripts $HOME/db_monitoring_scripts
ln -s /data/home/oracle_binaries $HOME/oracle_binaries
ln -s /data/home/production $HOME/production
ln -s /data/home/rac_mon $HOME/rac_mon
ln -s /data/home/scripts $HOME/scripts
ln -s /data/home/secscan $HOME/secscan
ln -s /data/home/streams $HOME/streams
ln -s /data/home/strmmon $HOME/strmmon
ln -s /data/home/tns_download $HOME/tns_download
```

## PdbBackupRecovery < PSSGroup < TWiki

```
ln -s /data/home/work $HOME/work
```

```
sudo ln -s /data/backup5 /backup5
sudo chown root:root /data/etc/cron.d/*
sudo ln -s /data/etc/cron.d/damrefresh.cron /etc/cron.d/damrefresh.cron
sudo ln -s /data/etc/cron.d/pdb-backup5.cron /etc/cron.d/pdb-backup5.cron
sudo ln -s /data/etc/cron.d/pdb-monitoring.cron /etc/cron.d/pdb-monitoring.cron
sudo ln -s /data/etc/init.d/dsmcad /etc/init.d/dsmcad
sudo service crond restart
```

- Again either using a backup or a legacy pdb-backup machine populate on both nodes the `/etc/inittab` file with entries that will start backup daemons.
- Copy over or restore from a backup the following scripts:

```
/data/home/check_running_backups.sh
/data/home/edit_dad
/data/home/edit_monitoring
/data/home/edit_tnsnames
```

- On both nodes create symlinks pointing to those scripts

```
ln -s /data/home/check_running_backups.sh $HOME/check_running_backups.sh
ln -s /data/home/edit_dad $HOME/edit_dad
ln -s /data/home/edit_monitoring $HOME/edit_monitoring
ln -s /data/home/edit_tnsnames $HOME/edit_tnsnames
```

- Install and configure EM client

- ◆ go to <https://oms.cern.ch/em/console/emcli/download> (authentication required)
- ◆ download `emclikit.jar`
- ◆ put the file on all nodes of pdb-backup
- ◆ install it:

```
java -jar emclikit.jar client -install_dir=/ORA/dbs01/oracle/product/10.2.0/emcli
```

- ◆ configure:

```
emcli setup -url="https://oms.cern.ch/em/" -username="pdbadminoem" -trustall
```

- Configure LDAP

- ◆ Place `ldap` action script (`action_ldap.scr`) in the `$ORA_CRS_HOME/crs/public/` directory on both nodes of the cluster.
- ◆ Make sure that the action script is owned by `root` and its primary group is also `root`
- ◆ Create profile and register CRS target for LDAP

```
# as root on both nodes:
sudo crs_profile -create pdb_ldap -t application -d "PDB LDAP" -r "pdbbackupvip fs_c
# as root on the first node:
sudo crs_register pdb_ldap
sudo crs_setperm pdb_ldap -o root
sudo crs_setperm pdb_ldap -u user:oracle:r-x
# as oracle on the first node:
crs_start pdb_ldap
```

- ◆ Reinitialize and prepare LDAP data:

```
cd ~/production/ldap
./reinitialize.sh
```

- Configure TSM backups of PDB-BACKUP:

- ◆ Make sure that TSM-related RPMs are installed on both nodes of the cluster:

## PdbBackupRecovery < PSSGroup < TWiki

```
sudo rpm -qa|grep TIV
# the ouptup should be similar to the following:
TIVsm-API64-5.3.4-0
TIVsm-BA-5.3.4-0
TIVsm-API-5.3.4-0
```

- ◆ PDB-BACKUP is being backed up to TSM31 (pdb-backup node name, x1.... password). On both cluster nodes deploy appropriate `dsm.sys`, `dsm.opt` and `backup.excl` files in the `/opt/tivoli/tsm/client/ba/bin` directory. The files are attached to this page.
- ◆ On both nodes create symbolic links in the `/usr` directory:

```
sudo ln -s /opt/tivoli/tsm/client/ba/bin /usr/dsm
```

- ◆ From both nodes of the cluster try to connect to the TSM server:

```
sudo /opt/tivoli/tsm/client/ba/bin/dsmc
# use pdb-backup as userid and x1.... password
```

### \*NOTE\*

This should create a password file in the `/etc/adsm` directory

- ◆ Configure automatic backup scheduling:

```
# on both nodes
sudo chmod +x /etc/init.d/dsmcad
sudo vi /etc/inittab
# add:
cad:345:once:/etc/init.d/dsmcad start # Start TSM scheduler
# on the active node only:
sudo /sbin/telinit q
```

## Recovery from CPU node loss

This recovery scenario covers a situation when the mid-range server used to run backup and monitoring software fails and has to be replaced with a spare node. At the same time the disk array attached to the machine is intact.

- Find a replacement node and install it with the proper, up to date version of OS (RHEL 4.0 32bit at the moment)
- Using PDB inventory [application](#) identify the name of the disk array attached to the old PDB backup machine
- Change FC zoning, so the new node could see the disk array identified in the previous step.
- Connect to the node and:
  - ◆ configure multipathing as described in the 'Setup storage' section of the Database installation instruction; the devmapper device can be named `itstorXXX_1p1`
  - ◆ create a `/data` directory and change it ownership to `oracle:ci`
  - ◆ mount the attached disk array (`sudo mount /dev/mpath/itstor330_1p1 /data`)
  - ◆ modify the `/etc/fstab` file in order to have the disk array mounted after the reboot; add `'/dev/mpath/itstor330_1p1 /data ext3 defaults 1 2'` to this file.
- Check contents of the disk array. It should contain at least the following directories:
  - ◆ `backup` - where backup copies of important scripts and programs are stored
  - ◆ `backup5` - directory structure and scripts used by RMAN backup scheduler, PDB recovery catalog export scheduler, Backup reporting utility and Backup validation utility
  - ◆ `oracle_binaries` - repository of all used Oracle installers.
- Install Oracle software (9iR2 and 10gR2)
- Restore and restart monitoring tools:

!!! public key

```
cp -rp /data/backup/rac_mon $HOME
cp -rp /data/backup/db_monitoring_scripts $HOME
cp -rp /data/backup/production $HOME
sudo cp /data/backup/pdb-monitoring.cron /etc/cron.d/
```

- Restore and restart backups:

```
sudo ln -s /data/backup5 /backup5
sudo cp /data/backup/pdb-backup5.cron /data/backup/rman-backup-summary.cron /etc/cron
```

- Restore DAM
- Restart autobackups
- Configure TSM backups.
- Other tasks:

```
ln -s $HOME/oracle_binaries /data/oracle_binaries
cp -rp /data/backup/work $HOME
cp -rp /data/backup/scripts $HOME
cp /data/backup/.bashrc $HOME
cp /data/backup/.bash_profile $HOME
```

## Recovery from disk array loss

### Disaster recovery

- action\_ldap.scr: action\_ldap.scr
- backup.excl: backup.excl
- dsm.opt: dsm.opt
- dsm.sys: dsm.sys
- dsmcad: dsmcad
- action\_PDB\_data.scr: action\_PDB\_data.scr

---

This topic: PSSGroup > PdbBackupRecovery  
Topic revision: r20 - 2009-03-17 - DawidWojcik



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback