

Exadime DBA Scripts Manual

Version 1.0.0

Database Administration

Revision History

Date	Version	Description	Author	Reviewer
Apr 12, 2016	1.0.0	Draft	EXADIME LLC	
Jul 06, 2017	1.0.0	Troubleshooting	EXADIME LLC	
Aug 08, 2018	1.0.0	Installation – UNIX	EXADIME LLC	
Mar 25, 2019	1.0.0	Software	EXADIME LLC	
Apr 01, 2019	1.0.0	Libraries	EXADIME LLC	
May 16, 2019	1.0.0	Golden Gate Parameters	EXADIME LLC	
May 04, 2020	1.0.0	Monitoring Scripts	EXADIME LLC	
Sep 03, 2020	1.0.0	Sample Cron Entry	EXADIME LLC	
Jan 13, 2020	1.0.0	Data Pump Export	EXADIME LLC	
Jan 15, 2020	1.0.0	Golden Gate Monitor	EXADIME LLC	
Feb 01, 2021	1.0.0	Removed title.ini and esp.ini	EXADIME LLC	
Feb 02, 2021	1.0.0	Host Ping Monitoring	EXADIME LLC	
Feb 07, 2021	1.0.0	Data Guard Monitoring	EXADIME LLC	
Feb 11, 2021	1.0.0	Directory Navigation	EXADIME LLC	
Apr 30, 2021	1.0.0	SQL Scripts	EXADIME LLC	
May 18, 2022	1.0.0	Installation & Navigation	EXADIME LLC	
Sep 12, 2023	1.0.0	TDE Wallet Backup	EXADIME LLC	
Nov 09, 2023	1.0.0	ASH Files	EXADIME LLC	
Dec 18, 2023	1.0.0	Java Files	EXADIME LLC	
May 20, 2024	1.0.0	Health Check	EXADIME LLC	

Contents

INTRODUCTION	6
INSTALLATION	6
UNIX	6
LINUX	7
WINDOWS	7
CONFIGURATION	8
1. Oracle files and directories.	8
2. Title.....	8
3. Notifications	8
4. Defaults	9
5. Oracle Cron.....	11
6. Job Schedule.....	12
7. Auto Start and Stop.....	12
GETTING STARTED.....	13
DATABASE ENVIRONMENT.....	13
DIRECTORY NAVIGATION.....	14
SETUP	15
1. DATABASE.....	15
1.1. stp_cr_db.ksh.....	15
1.2. stp_rm_db.ksh.....	15
MAINTENANCE	17
1. ENV SETTINGS.....	17
1.1. mnt_sw_db.ksh.....	17
2. LISTENER.....	17
2.1. mnt_st_lis.ksh	17
2.2. mnt_sp_lis.ksh	18
3. INSTANCE	19
3.1. mnt_st_db.ksh	19
3.2. mnt_sp_db.ksh	19
4. AGENT.....	20
4.1. mnt_st_agt.ksh.....	20
4.2. mnt_sp_agt.ksh	20
5. OSWATCHER.....	21
5.1. mnt_st_osw.ksh.....	21
5.2. mnt_sp_osw.ksh.....	21
6. BACKUPS.....	22
6.1. mnt_exp_dmp.ksh	22
6.2. mnt_imp_dmp.ksh.....	22
6.3. mnt_col_bkp.ksh	23
6.4. mnt_hot_bkp.ksh.....	24
6.5. mnt_tts_bkp.ksh	24
6.6. mnt_ctl_bkp.ksh.....	25
6.7. mnt_dp_exp.ksh	26
6.8. tde_wall_bkp.ksh	26

7. HOUSEKEEPING	27
7.1. <i>mnt_exa_hkf.ksh</i>	27
7.2. <i>mnt_ora_hkf.ksh</i>	27
7.3. <i>mnt_arc_hkf.ksh</i>	28
7.4. <i>mnt_aud_hkf.ksh</i>	28
7.5. <i>mnt_old_hkf.ksh</i>	29
7.6. <i>mnt_asm_aud.ksh</i>	30
8. PERFORMANCE	30
8.1. <i>mnt_anz_db.ksh</i>	30
MONITORING	31
1. STATUS	31
1.1. <i>mtr_ins_sts.ksh</i>	31
1.2. <i>mtr_lis_sts.ksh</i>	32
1.3. <i>mtr_host_ping.ksh</i>	32
2. SPACE	32
2.1. <i>mtr_tbs_spa.ksh</i>	33
2.2. <i>mtr_asm_disk.ksh</i>	33
2.3. <i>mtr_host_disk.ksh</i>	34
3. LOG	35
3.1. <i>mtr_alt_log.ksh</i>	35
3.2. <i>mtr_sys_log.ksh</i>	35
4. REPLICATION	36
4.1. <i>mtr_dg_gap.ksh</i>	36
4.2. <i>mtr_gg_lag.ksh</i>	36
REPORTS	38
1. <i>dar_mem_pad.ksh</i>	38
SUPPORT	38
1. <i>run_crn_job.ksh</i>	38
2. <i>run_sql_file.ksh</i>	39
3. <i>run_she_file.ksh</i>	39
4. <i>run_aut_ssh.ksh</i>	39
RECOVERY	40
1. <i>rco_mng_file.ksh</i>	40
2. <i>rco_syn_stb.ksh</i>	41
MISCELLANEOUS	42
1. <i>sho_file_cnt.ksh</i>	42
2. <i>sho_file_sum.ksh</i>	42
3. <i>sho_disk_sum.ksh</i>	42
4. <i>sho_crc_sum.ksh</i>	42
5. <i>sho_host_disk.ksh</i>	43
6. <i>sho_pat_con.ksh</i>	44
7. <i>sho_aix_node.ksh</i>	44
TROUBLESHOOTING	44
Problem #1	44
SQL SCRIPTS	45
REFERENCE	52
1. Interpretations	52



SOFTWARE.....	52
<i>OSWatcher.....</i>	<i>52</i>
<i>SQLT.....</i>	<i>52</i>
CONTACT US.....	52

Introduction

This document is a manual source for the usage of Exadime DBA Scripts for Oracle databases maintenance, monitoring, reporting, support and recovery.

Installation

UNIX

1. Make the base directory for Exadime scripts in ~oracle/scripts.

```
$mkdir -p ~oracle/scripts/exa
```

2. Transfer Exadime scripts into the above directory.

S. No.	Transfer Mode	Directory/File
1	Binary	doc lib plb unix
2	Text/ASCII	lib/exa.lib
3	Text/ASCII	unix/Sora; mnt_sw_db.ksh; run_crn_job.ksh
4	Text/ASCII	All other folders.

3. Set RWX permission.

```
chmod -R 750 ~oracle/scripts/exa
```

4. Specify Exadime scripts path in the following files if required.

S. No.	File Name	Directory	Comments
1	exa.ini	\${exa_home}/ini	Initialization parameters.
2	unix.ini	\${exa_home}/ini	Unix scripts initialization.
3	oracle.ini	\${exa_home}/ini	Oracle scripts initialization.
4	exa.env	\${exa_home}/env	Environment setup.
5	run_crn_job.ksh	\${exa_home}/unix	Cron job execution.
6	Sora	\${exa_home}/unix	Database auto start/stop.

Note: Default scripts path is “~oracle/scripts/exa” and ORACLE_BASE is “~oracle”.

5. Lib folder contains Korn Shell libraries for each O/S. Use the appropriate KSH file.

```
$cd lib  
$cp <ksh_os> ksh  
$chmod 750 ksh
```

6. Add the following line into .bash_profile (Linux)/.profile (AIX/Solaris).

```
. ~oracle/scripts/exa/env/exa.env
```

LINUX

In case of LINUX, install the following rpm.

```
#yum install ksh
```

```
#yum install bc
```

Windows

1. Create folder "scripts" in c:\oracle and extract scripts.

2. Set the values for SQLPATH and TNS_ADMIN.

1. Open command prompt.

```
Start>Run>cmd.exe
```

2. Open system properties.

```
C:\>control.exe %windir%\system32\sysdm.cpl
```

3. Select Advanced>Environment Variables>System Variables>New.

4. Enter the name and value as shown below.

```
SQLPATH=c:\oracle\scripts\exa\ini;c:\oracle\scripts\exa\sql
```

```
TNS_ADMIN=c:\oracle\scripts\exa\tns
```

3. Make necessary environment changes to the following file.

```
dba.env
```

Note: This is required only if the host is a database server.

Configuration

1. Oracle files and directories.

Specify the values for the following entries in exa.ini file.

- ✓ ora_tab
- ✓ ORACLE_BASE
- ✓ TNS_ADMIN

2. Title

Specify the following information in cnf.ini file.

- ✓ Company Name
- ✓ Company Address
- ✓ Author

3. Notifications

The following notifications can be set in cnf.ini file.

- ✓ email
- ✓ sms
- ✓ pager

There are two types of notifications.

- ✓ Default
- ✓ Ad-hoc

Default

Default notifications are regular notifications that will be sent to DBA group when production is live.

Ad-hoc

Ad-hoc notifications are used when notifications are to be sent to a specific DBA while performing maintenance or support activities.

The anatomy of -esp (email, sms and pager) value is as shown below.

-esp|-e [{0|1}{0|1}{0|1}[a-z]].

0-Turnoff

1-Turnon

Contents of cnf.ini file related to notifications are shown below.

```
#Default DBA Notifications
dba_email=<email>
dba_sms=<sms>
dba_pager=<pager>
return_recipient=<email>

#Ad-hoc DBA Notifications
dba_email_x=<email>
dba_sms_x=<sms>
dba_pager_x=<pager>
return_recipient=<email>
```

Example

1.To send notifications to default group, specify at -esp|-e as below.

- e 100 for email.
- e 110 for email and mobile sms.
- e 111 for email, mobile sms and pager.

2.To send notifications to ad-hoc group, specify at -esp|-e as below.

- e 100x for email.
- e 110x for email and mobile sms.
- e 111x for email, mobile sms and pager.

4. Defaults

Default values for different inputs in the scripts can be specified in cnf.ini file. The following are the list of entries that can be found in cnf.ini file. These are self-explanatory.

```
#File Permissions
umask 026

#Tablespace Threshold Limits
#1.Regular
```

rts_lmt=90

#2.Fatal
fts_lmt=95

#File System Threshold Limits
#1.Regular
rfs_lmt=90

#2.Fatal
ffs_lmt=95

#Default Notification Codes
stp_esp="000"
mnt_esp="000"
mtr_esp="000"
dar_esp="000"
run_esp="000"
rco_esp="000"
trs_esp="000"
sho_esp="000"

5. Oracle Cron

It is recommended to set the following jobs in Oracle cron as described below.

Job#	Category	Description	Frequency	esp		Script	Environment				Comments
				P&S	U&D		PRD	STB	UAT	DEV	
1.1	MNT	Listener shutdown.	Weekly	100	100	mnt_sp_lis.ksh	Y	N/A	Y	Y	Green Zone
1.2	MNT	Listener startup.	Weekly	100	100	mnt_st_lis.ksh	Y	N/A	Y	Y	Green Zone
1.3	MNT	Database shutdown*.	Weekly	100	100	mnt_sp_db.ksh	Y	N/A	Y	Y	Green Zone
1.4	MNT	Database startup.	Weekly	100	100	mnt_st_db.ksh	Y	N/A	Y	Y	Green Zone
1.5	MNT	Cold backup.	Weekly	100	100	mnt_col_bkp.ksh	Y	N/A	Y	Y	Green Zone
1.6	MNT	Hot backup.	Daily/Weekly	100	100	mnt_hot_bkp.ksh	Y	N/A	Y	Y	
1.7	MNT	Control file backup.	Daily	100	100	mnt_ctl_bkp.ksh	Y	N/A	Y	Y	
1.8	MNT	Structure export.	Daily	100	100	mnt_exp_dmp.ksh	Y	N/A	Y	Y	
1.9	MNT	Data export.	Daily/Weekly	100	100	mnt_exp_dmp.ksh	Y	N/A	Y	Y	
1.10	MNT	File housekeeping.	Daily	100	100	mnt_exa_hkf.ksh	Y	Y	Y	Y	keepdays=30
1.11	MNT	Oracle housekeeping.	Daily	100	100	mnt_ora_hkf.ksh	Y	Y	Y	Y	keepdays=30
1.12	MNT	Audit housekeeping.	Daily	100	100	mnt_aud_hkf.ksh	Y	Y	Y	Y	keepdays=30
1.13	MNT	Arc housekeeping.	Daily	100	100	mnt_arc_hkf.ksh	Y	Y	Y	Y	keepdays=7
1.14	MNT	Block corruption.	Daily/Weekly	100	100	mnt_anz_db.ksh	Y	N/A	Y	Y	act=v:lev=d
1.15	MNT	Database Analyze.	Daily/Weekly	100	100	mnt_anz_db.ksh	Y	N/A	Y	Y	act=g:lev=d
2.1	MTR	Instance status.	10 Min	101	100	mtr_ins_sts.ksh	Y	Y	Y	Y	
2.2	MTR	Listener status.	10 Min	101	100	mtr_lis_sts.ksh	Y	Y	Y	Y	
2.3	MTR	Server status.	10 Min	101	100	mtr_svr_sts.ksh	Y	Y	Y	Y	
2.4	MTR	Regular ts space.	Daily	100	100	mtr_rts_spa.ksh	Y	N/A	Y	Y	rts_lmt=90
2.5	MTR	Fatal ts space.	1 Hour	101	100	mtr_fts_spa.ksh	Y	N/A	Y	Y	fts_lmt=95
2.6	MTR	Regular fs space.	Daily	100	100	mtr_rfs_spa.ksh	Y	Y	Y	Y	rfs_lmt=90
2.7	MTR	Fatal fs space.	1 Hour	101	100	mtr_ffs_spa.ksh	Y	Y	Y	Y	ffs_lmt=95
2.8	MTR	Alert log.	1 Hour	101	100	mtr_alt_log.ksh	Y	Y	Y	Y	fire=hour
2.9	MTR	Sys log.	1 Hour	101	100	mtr_sys_log.ksh	Y	Y	Y	Y	fire=hour

* Database shutdown job should be avoided if cold backup job is in place.

Example

```
#####
####                      OS Watcher Monitoring                      #####
#####
00 * * * * /home/oracle/scripts/exa/unix/run_crn_job.ksh -f "mtr_osw_sts.ksh -a y -r 728
-e 100" > /dev/null 2>&1
```

6. Job Schedule

If possible, it is recommended to schedule the above daily jobs as described below except in Green Zone.

1.00:00 - 06:00	- Backup
2.06:00 - 12:00	- Monitoring (Regular)
3.18:00 - 00:00	- Housekeeping

Schedule weekly jobs during the time when there is minimal business activity in the database.

7. Auto Start and Stop

Create the following soft links in /etc/rc?.d directory. These will perform auto start and stop of databases and listeners.

S.No.	Link	Description
1	ln -s ~oracle/scripts/exa/unix/Sora Sora	Auto Start
2	ln -s ~oracle/scripts/exa/unix/Sora Kora	Auto Stop

Once the link is created, specify the database role in oratab file at the end of every entry separated by colon (:).

S.No.	Server	Database Role
1	Primary	P
2	Standby	S
3	Data Guard	D

Syntax

<SID>:<ORACLE_HOME>:<AUTO_START>:<DATABASE_ROLE>

Example

ORCL:~oracle/product/10.2.0.1:N:P
STBY:~oracle/product/10.2.0.1:N:D

Note: 1. Standard oratab file does not contain <DATABASE_ROLE>.
2. Always specify AUTO_START=N.

Getting Started

Database Environment

After scripts setup is completed, run exa.env script to prepare the environment.

```
$. ~oracle/scripts/exa/env/exa.env
```

Note: If scripts are copied in different location, then specify the appropriate path.

When you run exa.env, it will set to first database environment listed in oratab file.

```
$Switched to db <DB Name> environment on host <Hostname>!
```

If no entries exist in oratab, it leaves message as shown below.

```
$oratab file is blank!
```

Now you are ready to run any unix script from any directory. To see the help of the script of your choice, simply type the script name with -h or -help argument.

```
$ksh run_sql_file.ksh -help
```

Directory Navigation

You can navigate to the following directories just by typing following short name at unix prompt.

S.No.	Directory Name	Short Name	Description
0	exa	exa	Scripts
1	alt	alt	Alert Files
2	ash	ash	ASH Files
3	awr	awr	AWR Files
4	bat	bat	Windows Scripts
5	crn	crn	Cron Backup
6	doc	doc	Documents
7	env	en	Environment Files
8	ini	ini	Ini Files
9	java	jav	Java Files
10	job	job	User Cron Job Files
11	lib	lib	Library Files
12	log	log	Log Files
13	mas	mas	Master Files
14	oby	oby	Golden Gate Obey Files
15	ora	ora	Oracle Support Scripts
16	osw	osw	OSWatcher
17	out	out	Troubleshooting Logs*
18	pal	pal	3 rd Party Scripts
19	par	par	Parameter Files
20	pie	pie	Unix Pipes
21	plb	plb	Encrypted PL/SQL Code
22	pls	pls	PL/SQL Scripts
23	prm	prm	Golden Gate Parameters
24	psu	psu	PSU Patching
25	rda	rda	Remote Diagnostic Agent
26	rmn	rmn	RMAN Scripts
27	rpt	rpt	Report Files
28	rsp	rsp	Response Files
29	shc	shc	Health Check
30	spl	spl	Special Files
31	sql	sql	SQL Scripts
32	sqt	sqt	SQLT
33	tmp	tmp	Temp Files
34	tns	tns	tnsnames.ora
35	unix	unx	Unix Scripts

*As out files are intended for troubleshooting purpose e.g. Oracle Support, housekeeping scripts do not remove files from this folder.

Setup

1. Database

1.1. stp_cr_db.ksh

This script creates database.

Syntax

```
stp_cr_db.ksh -n <db name> -v <db ver> -m <ora home> -x <mp pfx> -p <par file> -l <label> -t <title> -e <esp>
```

Parameters

- n Database Name.
- v Database Version(9i|10g|11g|12c).
- m Oracle Home.
- lp [o] Listener Port.
- x [o] Mount Point Directory prefix.
- p [o] Name of par file.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Defaults

- x "u"
- p \${par_dir}/cr_db_{-v}.par

Example

```
stp_cr_db.ksh -n ORCL -v 10g -m ~oracle/product/10.2.0.1 -x u -p  
${par_dir}/cr_db_9i.par -l ORCL -t ORCL-CREATE-DB -e 100x
```

1.2. stp_rm_db.ksh

This script removes database.

Syntax

```
stp_rm_db.ksh -s <sid> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance Name.

-x [o] Mount Point Directory prefix.
-p Parameters (act={run|gen}:adm=[y|n]).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.
 run -Run.
 gen -Generates script.
adm Remove admin directories.
 y [d] -Removes.
 n -Does not remove.

Defaults

-x "u"

Example

stp_rm_db.ksh -s ORCL -p {act=run} -l ORCL -t ORCL-REM-DB -e 100x

Maintenance

1. Env Settings

1.1. mnt_sw_db.ksh

This script sets database environment of the given sid or selected sid. This script can be used either in interactive mode or non-interactive mode.

Syntax

1. Interactive Mode
 mnt_sw_db.ksh
2. Non-Interactive Mode
 mnt_sw_db.ksh <sid>

Alias

swdb

Example

1. Interactive Mode
 mnt_sw_db.ksh or swdb
2. Non-Interactive Mode
 mnt_sw_db.ksh smpl

Output

\$ swdb

Here are the databases on <hostname>, your sid is currently ORCL

- 1) ORCL
- 2) GNRL

Which database would you like to switch to?

2. Listener

2.1. mnt_st_lis.ksh

This script starts listener(s).

Syntax

```
mnt_st_lis.ksh -i <lis> -s <sid> -l <label> -t <title> -e <esp>
```

Parameters

- i Listener name|ALL.
- s Instance name (Only for single listener).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_st_lis.ksh -i ORCL_LISTENER -s ORCL -l ORCL -t ORCL-LIS-START -e 100  
mnt_st_lis.ksh -i ALL -l ORCL -t ORCL-LIS-START -e 100
```

2.2. mnt_sp_lis.ksh

This script stops listener(s).

Syntax

```
mnt_sp_lis.ksh -i <lis> -s <sid> -l <label> -t <title> -e <esp>
```

Parameters

- i Listener name|ALL.
- s Instance name (Only for single listener).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_sp_lis.ksh -i ORCL_LISTENER -s ORCL -l ORCL -t ORCL-LIS-STOP -e 100  
mnt_sp_lis.ksh -i ALL -l ORCL -t ORCL-LIS-STOP -e 100
```

3. Instance

3.1. mnt_st_db.ksh

This script starts database(s).

Syntax

```
mnt_st_db.ksh -s <sid> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- p [o] Parameters (opt=[pri|stb|odg]) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

- opt Startup option.
 - pri [d] -Primary database.
 - stb -Standby database.
 - odg -Oracle Data Guard.

Example

```
mnt_st_db.ksh -s ORCL -p {opt=pri} -l ORCL -t ORCL-START -e 100
mnt_st_db.ksh -s ORCL -p {opt=odg} -l ORCL -t ORCL-START -e 100
```

3.2. mnt_sp_db.ksh

This script stops database(s).

Syntax

```
mnt_sp_db.ksh -s <sid> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- p [o] Parameters (opt=[nor|imm|abo]) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.

-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

opt Shutdown option.
nor [d] -Normal.
imm -Immediate.
abo -Abort.

Example

mnt_sp_db.ksh -s ORCL -p {opt=imm} -l ORCL -t ORCL-STOP -e 100

4. Agent

4.1. mnt_st_agt.ksh

This script starts agent.

Syntax

mnt_st_agt.ksh -d <agent home directory> -l <label> -t <title> -e <esp>

Parameters

-d EM Agent home directory.
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

mnt_st_agt.ksh -d ~oracle/product/12.1.0.5/agent/agent_inst -e 100

4.2. mnt_sp_agt.ksh

This script stops agent.

Syntax

mnt_sp_agt.ksh -d <agent home directory> -l <label> -t <title> -e <esp>

Parameters

-d EM Agent home directory.
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)

- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_sp_agt.ksh -d ~oracle/product/12.1.0.5/agent/agent_inst -e 100
```

5. OSWatcher

5.1. mnt_st_osw.ksh

This script starts OSWatcher.

Syntax

```
mnt_st_osw.ksh -d <OSWatcher directory> -l <label> -t <title> -e <esp>
```

Parameters

- d OSWatcher directory.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_st_osw.ksh -d ~oracle/scripts/osw -e 100
```

5.2. mnt_sp_osw.ksh

This script stops OSWatcher.

Syntax

```
mnt_sp_osw.ksh -d <OSWatcher directory> -l <label> -t <title> -e <esp>
```

Parameters

- d OSWatcher directory.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_sp_osw.ksh -d ~oracle/scripts/osw -e 100
```

6. Backups

6.1. mnt_exp_dmp.ksh

This script does logical export of the given database.

Syntax

```
mnt_exp_dmp.ksh -s <sid> -o <loc sid> -d <dmp directory> -p <par file> -c <cut files> -z <file size> -n <db env> -k <keep days> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name.
- o Local instance name. Use this if connected from remote machine.
- d Dump file directory.
- p Par file name or parfile contents enclosed in curling braces ({}).
- c [o] Split option [y/n].
- z Split file size. Required only if split option is "y".
- n [o] Database environment report [y/n].
- k [o] # of days to keep old backups.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Defaults

- c "n"
- n "n"

Example

```
mnt_exp_dmp.ksh -s ORCL -d /u02/oradata/ORCL/expbkp -p  
~oracle/scripts/exa/par/exp_cmpd_9i.par -c y -z 8192m -n n -k 2 -l ORCL-cmpd -t  
ORCL-DATA-EXP -e 100
```

6.2. mnt_imp_dmp.ksh

This script does logical import from the given compressed dump file/directory.

Syntax

```
mnt_imp_dmp.ksh -s <sid> -o <loc sid> -f <dmp file> -d <dmp directory> -p  
<par file> -c <cut files> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name.
- o Local instance name. Use this if connected from remote machine.

-f Dump file name.
-d Dump file directory. Required only if split option is "y".
-p Par file name or parfile contents enclosed in curling braces ({}).
-c [o] Split option [y/n].
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Defaults

-c "n"

Example

```
mnt_imp_dmp.ksh -s ORCL -d /u02/oradata/ORCL/expbkp-p "{fromuser=pie  
touser=pie ignore=y}" -c y -l ORCL-PIE-IMP -t ORCL-PIE-IMP -e 100
```

6.3. mnt_col_bkp.ksh

This script does cold backup of the given database.

Syntax

```
mnt_col_bkp.ksh -s <sid> -d <bkp directory> -p <parameters> -l <label> -t  
<title> -e <esp>
```

Parameters

-s Instance name.
-d Backup file directory.
-p Parameters (act={run|gen}:bsz=[n]:up=[y|n]) separated by colon (:) and enclosed in curling braces ({}).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act	Action.
run	-Run.
gen	-Generates script.
bsz	Backup set size.
up	Startup option after backup.
y [d]	-Brings database up after backup.
n	-Does not bring database up after backup.

Example

```
mnt_col_bkp -s ORCL -d /u02/oradata/ORCL/colbkp -p {act=run:bsz=10} -l  
ORCL -t ORCL-COL-BKP -e 100
```

6.4. mnt_hot_bkp.ksh

This script does hot backup of the given database.

Syntax

```
mnt_hot_bkp.ksh -s <sid> -d <bkp directory> -p <parameters> -l <label> -t  
<title> -e <esp>
```

Parameters

-s Instance name.
-d Backup file directory.
-p Parameters (act={run|gen}) separated by colon (:) and enclosed in curling braces ({}).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act	Action.
run	-Run.
gen	-Generates script.

Example

```
mnt_hot_bkp -s ORCL -d /u02/oradata/ORCL/hotbkp -p {act=run} -l ORCL -t  
ORCL-HOT-BKP -e 100
```

6.5. mnt_tts_bkp.ksh

This script does transportable tablespaces backup of the given database.

Syntax

```
mnt_tts_bkp.ksh -s <sid> -d <bkp directory> -p <parameters> -l <label> -t  
<title> -e <esp>
```

Parameters

-s Instance name.
-d Backup file directory.
-p Parameters (act={run|gen}:bsz=[n]) separated by colon (:) and enclosed in curling braces ({}).

-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)

-t [o] Title for report/alert.

-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.

run -Run.

gen -Generates script.

bsz Backup set size.

Example

```
mnt_tts_bkp -s ORCL -d /u02/oradata/ORCL/ttsbkp -p {act=run:bsz=10} -l ORCL -t ORCL-TTS-BKP -e 100
```

6.6. mnt_ctl_bkp.ksh

This script does controlfile backup of the given database.

Syntax

```
mnt_ctl_bkp.ksh -s <sid> -d <bkp directory> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

-s Instance name.

-d Backup file directory.

-p Parameters (act={run|gen}) separated by colon (:) and enclosed in curling braces ({}).

-l [o] Label for rpt/alt/log file.

(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)

-t [o] Title for report/alert.

-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.

run -Run.

gen -Generates script.

Example

```
mnt_ctl_bkp -s ORCL -d /u02/oradata/ORCL/ctlbkp -p {act=run} -l ORCL -t ORCL-CTL-BKP -e 100
```

6.7. mnt_dp_exp.ksh

This script does data pump export of the given database.

Syntax

```
mnt_dp_exp.ksh -s <sid> -o <loc sid> -d <dmp directory name> -p <par file> -r  
<data pump recovery template> -k <keep days> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name.
- o Local instance name. Use this if connected from remote machine.
- d [o] Data Pump directory name.
- p [o] Par file name or parfile contents enclosed in curling braces ({}).
- r [o] Data pump recovery template [y/n].
- k [o] # of days to keep old backups.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Defaults

- d DATA_PUMP_DIR
- p dp.par
- r "y"

Example

```
mnt_dp_exp.ksh -s ORCL -l ORCL -t ORCL-DATA-PUMP-EXP -e 100
```

6.8. tde_wall_bkp.ksh

This script does backup of TDE wallet files.

Syntax

```
tde_wall_bkp.ksh -s <sid> -d <backup directory name> -k <keep days> -t <title>  
-e <esp>
```

Parameters

- s Instance name.
- d Backup directory name.
- k # of days to keep old backups.
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
tde_wall_bkp.ksh -s ORCL -d /tmp -k 10 -t ORCL-TDE-WALL-BKP -e 100
```

7. Housekeeping

7.1. mnt_exa_hkf.ksh

This script removes files present in rpt, alt, log, tmp, spl and pie directories.

Syntax

```
mnt_exa_hkf.ksh -k <keep days> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

-k # of days to keep files.
-p Parameters (act={run|gen}:hst={day|mon}).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.
run -Run.
gen -Generates script.
hst History of mnt_exc_crn.log.

Example

```
mnt_exa_hkf.ksh -k 30 -p {act=run:hst=mon} -l ATLANTIC -t ATLANTIC-  
EXA-FILES -e 100
```

7.2. mnt_ora_hkf.ksh

This script removes files present in bdump, udump and cdump directories.

Syntax

```
mnt_ora_hkf.ksh -k <keep days> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

-k # of days to keep files.
-p Parameters (act={run|gen}:hst={n}:arc=[y|n]).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act	Action.	
	run	-Run.
	gen	-Generates script.
hst	History of alert log in days.	
arc	Archival of history.	
	y [d]	-Archives history.
	n	-Does not archive history.

Example

```
mnt_ora_hkf.ksh -k 30 -p {act=run:hst=7:arc=n} -l ATLANTIC -t ATLANTIC-ORA-FILES -e 100
```

7.3. mnt_arc_hkf.ksh

This script compresses and removes Oracle archive log files.

Syntax

```
mnt_arc_hkf.ksh -k <keep days> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

-k	# of days to keep files.
-p	Parameters (act={run gen}).
-l [o]	Label for rpt/alt/log file. (Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o]	Title for report/alert.
-e [o]	email/sms/pager notification [{0 1}{0 1}{0 1}[a-z]].

Brace Parameters

act	Action.	
	run	-Run.
	gen	-Generates script.

Example

```
mnt_arc_hkf.ksh -k 7 -p {act=run} -l ATLANTIC -t ATLANTIC-ARC-FILES -e 100
```

7.4. mnt_aud_hkf.ksh

This script compresses and removes Oracle audit files.

Syntax

```
mnt_aud_hkf.ksh -k <keep days> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- k # of days to keep files.
- p Parameters (act={run|gen}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

- | | |
|-----|--------------------|
| act | Action. |
| run | -Run. |
| gen | -Generates script. |

Example

```
mnt_aud_hkf.ksh -k 30 -p {act=run} -l ATLANTIC -t ATLANTIC-AUD-FILES  
-e 100
```

7.5. mnt_old_hkf.ksh

This script removes old files.

Syntax

```
mnt_old_hkf.ksh -d <dir> -x <file ext> -k <keep days> -p <parameters> -l  
<label> -t <title> -e <esp>
```

Parameters

- d Directory of files.
- x Extension of file names. Specify "ALL" for all files.
- k # of days to keep files.
- p Parameters (act={run|gen}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

- | | |
|-----|--------------------|
| act | Action. |
| run | -Run. |
| gen | -Generates script. |

Example

```
mnt_rm_fls.ksh -d ~oracle -x ALL -k 30 -p {act=run} -l ATLANTIC -t  
ATLANTIC-OLD-FILES -e 100
```

7.6. mnt_asm_aud.ksh

This script removes old ASM audit files.

Syntax

```
mnt_asm_aud.ksh -k <keep days> -l <label> -t <title> -e <esp>
```

Parameters

- k # of days to keep files.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mnt_asm_aud.ksh -k 90 -l ATLANTIC -t ATLANTIC-ASM-AUD-FILES -e 100
```

8. Performance

8.1. mnt_anz_db.ksh

This script analyzes the given database(s).

Syntax

```
mnt_anz_db.ksh -s <sid> -o <loc sid> -p <par file> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- o Local instance name. Use this if connected from remote machine.
- p Par file name or parfile contents separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mmt_anz_db.ksh -s ORCL -p ~oracle/scripts/exa/par/anz_db.par -l ORCL -t
ORCL-ANZ-DB -e 100
mmt_anz_db.ksh -s ORCL -p {act=a:lev=d:pct=20:opt=4} -l ORCL -t ORCL-
ANZ-DB -e 100
```

Brace Parameters

act	Action.	
	v	-Validate structure.
	g	-Gather statistics.
	b	-Both.
lev	Level.	
	d	-Database.
	s	-Schema.
pct [o]	Percent.	
opt [o]	Option.	
	1 [d]	-Gather.
	2	-Gather empty.
	3	-Gather stale.
	4	-Gather auto.

Monitoring

1. Status

1.1. mtr_ins_sts.ksh

This script checks instance status.

Syntax

```
mtr_ins_sts.ksh -s <sid> -l <label> -t <title> -e <esp>
```

Parameters

-s Instance name|ALL.
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mtr_ins_sts.ksh -s ORCL -l ORCL -t ORCL-INSANCE-STATUS -e 101
```

1.2. mtr_lis_sts.ksh

This script checks listener status.

Syntax

```
mtr_lis_sts.ksh -i <listener> -l <label> -t <title> -e <esp>
```

Parameters

- i Listener Name|ALL.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mtr_lis_sts.ksh -i ORCL_LISTENER -l ORCL -t ORCL-LISTENER-STATUS -e 100
```

1.3. mtr_host_ping.ksh

This script pings the given host(s) and alerts if response not received from the host(s).

Syntax

```
mtr_host_ping.ksh -p <par file> -l <label> -t <title> -e <esp>
```

Parameters

- p [o] Par file name.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Default

Parameter file is ping.ini.

Example

```
mtr_host_ping.ksh -l smg -t SMG-DOMAIN-SERVER(S)-STATUS -e 101
```

2. Space

2.1. mtr_tbs_spa.ksh

This script checks tablespace space and reports the tablespace(s) of each or individual database(s) that are exceeding the regular space limit.

Syntax

```
mtr_tbs_spa.ksh -s <sid> -o <loc sid> -p <parameters> -l <label> -t <title> -e  
<esp>
```

Parameters

- s Instance name|ALL.
- o Local instance name. Use this if connected from remote machine.
- p Parameters (rts_lmt={nn}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

rts_lmt Threshold limit (% used).

Example

```
mtr_tbs_spa.ksh -s ORCL -p {rts_lmt=90} -l ORCL -t ORCL-TABLESPACES-  
SPACE -e 100
```

2.2. mtr_asm_disk.ksh

This script checks asm file system space and reports the file systems that are exceeding the specified threshold.

Syntax

```
mtr_asm_disk.ksh -p <par file> -l <label> -t <title> -e <esp>
```

Parameters

- p [o] Par file name.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Default

Parameter file is asm.ini.

Example

```
mtr_asm_disk.ksh -l ATLANTIC -t ATLANTIC-FILE-SYSTEM-SPACE -e 111
```

2.3. mtr_host_disk.ksh

This script checks file system space and reports the file systems that are exceeding the specified threshold.

Syntax

```
mtr_host_disk.ksh -p <par file> -l <label> -t <title> -e <esp>
```

Parameters

- p [o] Par file name.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Default

Parameter file is disk.ini.

Example

```
mtr_host_disk.ksh -l ATLANTIC -t ATLANTIC-FILE-SYSTEM-SPACE -e 100
```

3. Log

3.1. mtr_alt_log.ksh

This script reports errors that occur in database alert log. It can be fired on each day or each hour.

Syntax

```
mtr_alt_log.ksh -s <sid> -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- p Parameters (fire={ day|hour }) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

- fire Error checking interval.
 - day -Checks errors once in a day.
 - hour -Checks errors once in an hour.

Example

```
mtr_alt_log.ksh -s ORCL -p {fire=day} -l ORCL -t ORCL-ALT-LOG -e 111
```

3.2. mtr_sys_log.ksh

This script reports errors that occur in server log. It can be fired on each day or each hour.

Syntax

```
mtr_sys_log.ksh -p <parameters> -l <label> -t <title> -e <esp>
```

Parameters

- p Parameters (fire={ day|hour }) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

fire Error checking interval.
 day -Checks errors once in a day.
 hour -Checks errors once in an hour.

Example

```
mtr_sys_log.ksh -s ORCL -p {fire=day} -l ATLANTIC -t ATLANTIC-SYS-LOG  
-e 111
```

4. Replication

4.1. mtr_dg_gap.ksh

To monitor Data Guard gap. Run on primary database.

Syntax

```
mtr_dg_gap.ksh -s <sid> -g <lag> -l <label> -t <title> -e <esp>
```

Parameters

-s Instance name.
-p Process names separated by colon (:) and enclosed in curling braces ({}).
-g Gap.
-l [o] Label for rpt/alt/log file.
 (Should be alpha numeric with special characters - and _ only. Blank
space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mtr_dg_gap.ksh -s ORCL -g 20 -e 100
```

Defaults

-g 10

4.2. mtr_gg_lag.ksh

This script monitors Golden Gate processes and replication lag.

Syntax

```
mtr_gg_lag.ksh -s <sid> -p <process> -g <lag> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- p Process names separated by colon (:) and enclosed in curling braces ({}).
- g Lag in minutes.
- d Golden Gate home directory.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
mtr_gg_lag.ksh -s ORCL -p ALL -g 30 -d /home/oracle/product/ggs/19.1.0 -e 100
```

Defaults

- p ALL
- g 60 Minutes

Reports

1. dar_mem_pad.ksh

This script provides memory stats.

Syntax

```
dar_mem_pad.ksh -s <sid> -o <loc sid> -l <label> -t <title> -e <esp>
```

Parameters

- s Instance name|ALL.
- o Local instance name. Use this if connected from remote machine.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
dar_mem_pad.ksh -s ORCL -l ORCL -t ORCL-MEM-PAD -e 100
```

Support

1. run_crn_job.ksh

This script executes the given script from cron. This script does not use .profile of the cron user.

Syntax

```
run_crn_job.ksh -a <scripts_home> -f <script_name>
```

Parameters

- a [o] Scripts home path.
- f Script file name.

Example

```
12 19 * * * ~oracle/scripts/exa/unix/run_crn_job.ksh -f "mnt_exp_dmp.ksh -s ORCL -d /u02/oradata/ORCL/expbkp -p ~oracle/scripts/exa/par/exp_cmpd_9i.par -c y -z 8192m -n n -k 2 -l ORCL-cmpd -t ORCL-DATA-EXP -e y" > /dev/null 2>&1
```

2. run_sql_file.ksh

This script runs the given SQL file in one or all databases.

Syntax

```
run_sql_file.ksh -s <sid> -o <loc sid> -f <sql file> -l <label> -t <title> -e <email>
```

Parameters

- s Instance name|ALL.
- o Local instance name. Use this if connected from remote machine.
- f SQL file name.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
run_sql_file.ksh -s ALL -f $ora_dir/db_env.sql -l ATLANTIC -t DB-ENV -e 100
```

3. run_she_file.ksh

This script runs the given shell file.

Syntax

```
run_she_file.ksh -f <sql file> -l <label> -t <title> -e <email>
```

Parameters

- f Shell file name.
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
run_sql_file.ksh -f $unx_dir/svr_stat.sh -l stats -t SERVER-STATS -e 100
```

4. run_aut_ssh.ksh

This script transfers files via ssh.

Syntax

```
run_aut_ssh.ksh -rs <rmt svr> -sd <src dir> -td <tar dir> -x <file ext> -p  
<parameters> -l <label> -t <title> -e <esp>
```

Parameters

- rs Remote server.
- sd Source directory.
- td Target directory.
- x Extension of file names. Specify "ALL" for all files.
- p Parameters (act={run|gen}:tsz=[n]) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
- t [o] Title for report/alert.
- e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

- act Action.
 - run -Run.
 - gen -Generates script.
- tsz Transfer set size.

Example

```
run_aut_ssh.ksh -rs 150.110.177.219 -sd /u02/oradata/ORCL/colbkp -td  
/u02/oradata/ORCL/colbkp -x ALL -p {act=run:tsz=20} -l ORCL -t ORCL-COL-AUT-  
SSH -e 111
```

Recovery

1. rco_mng_file.ksh

This script manages file(s).

Syntax

```
rco_mng_file.ksh -sd <src dir> -td <tar dir> -x <file ext> -p <parameters> -l  
<label> -t <title> -e <esp>
```

Parameters

- sd Source directory.
- td Target directory.
- x Extension of file names. Specify "ALL" for all files.
- p Parameters (act={run|gen}:cmd={c|u|p|m}:fsz=[n]) separated by colon (:) and enclosed in curling braces ({}).
- l [o] Label for rpt/alt/log file.

(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)

-t [o] Title for report/alert.

-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.

run -Run.

gen -Generates script.

cmd Command.

c -Compress

u -Uncompress

p -Copy

m -Move

fsz File set size.

Example

```
rco_mng_file.ksh -sd /u03/oradata/ORCL -x ALL -p {act=run:cmd=u:fsz=10} -l  
ORCL -t ORCL-UNC-ARC -e 100x
```

2. rco_syn_stb.ksh

This script sync standby database.

Syntax

```
run_syn_stb.ksh -s <sid> -rs <rmt svr> -sd <src dir> -td <tar dir> -p  
<parameters> -l <label> -t <title> -e <esp>
```

Parameters

-s Instance name.

-rs Remote server.

-sd Source directory.

-td Target directory.

-p Parameters (act={run|gen}:fsz=[n]) separated by colon (:) and enclosed in curling braces ({}).

-l [o] Label for rpt/alt/log file.

(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)

-t [o] Title for report/alert.

-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act Action.

run -Run.

gen -Generates script.

fsz File set size.

Example

```
rco_syn_stb.ksh -s ORCL -rs gwmrutidbpd1 -sd /u03/oradata/ORCL -td  
/u03/oradata/ORCL -p {act=run:fsz=10} -l ORCL -e 100
```

Miscellaneous

1. sho_file_cnt.ksh

Shows count of files in current working directory.

Syntax

```
sho_file_cnt.ksh
```

Example

```
sho_file_cnt.ksh
```

2. sho_file_sum.ksh

Shows sum of file sizes in current working directory in mega bytes.

Syntax

```
sho_file_sum.ksh
```

Example

```
sho_file_sum.ksh
```

3. sho_disk_sum.ksh

Shows sum of file system sizes in the server in giga bytes.

Syntax

```
sho_disk_sum.ksh
```

Example

```
sho_disk_sum.ksh
```

4. sho_crc_sum.ksh

Shows cyclical redundancy check sum.

Syntax

```
sho_crc_sum.ksh -d <file dir> -o <file order> -p <parameters> -l <label> -t  
<title> -e <esp>
```

Parameters

-d Directory of file(s).
-o [o] Order of files.
-p Parameters (act={run|gen}:csz=[n]) separated by colon (:) and enclosed in curling braces ({}).
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Brace Parameters

act	Action.	
	run	-Run.
	gen	-Generates script.
csz	Crc set size.	

Example

```
sho_crc_sum.ksh -d /u02/oradata/ORCL/colbkp -p {act=run:csz=10} -l ORCL -t  
ORCL-COL-BKP-CRC-SUM -e 100
```

5. sho_host_disk.ksh

Shows file system space.

Syntax

```
sho_host_disk.ksh -n <nas > -l <label> -t <title> -e <esp>
```

Parameters

-n Include NAS.
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
sho_host_disk.ksh -n y -l <hostname> -t HOST-DISK-REPORT -e 100
```

6. sho_pat_con.ksh

To check one off patch conflict.

Syntax

```
sho_pat_con.ksh -d <patch directory> -l <label> -t <title> -e <esp>
```

Parameters

-d Patch directory.
-l [o] Label for rpt/alt/log file.
(Should be alpha numeric with special characters - and _ only. Blank space is not allowed.)
-t [o] Title for report/alert.
-e [o] email/sms/pager notification [{0|1}{0|1}{0|1}[a-z]].

Example

```
sho_pat_con.ksh -d ~oracle/patches -l <hostname> -t PATCH-CONFLICT -e 100
```

7. sho_aix_node.ksh

Shows nodes in AIX cluster.

Syntax

```
sho_aix_node.ksh
```

Example

```
sho_aix_node.ksh
```

Troubleshooting

Problem #1

Error: ": command not found"

Resolution: 1. Copy the directory contents in ASCII format. env, ini, lib/exa.lib, par, unix/mnt_sw_db.ksh.

2. Make sure ksh is present in lib folder. If not, copy the respective ksh located in lib folder.

SQL Scripts

The following SQL queries included in the scripts to support Oracle databases.

#	Name	Purpose
0	exa_info.sql	Shows Exadime scripts release.
1	act_ses_info.sql	Shows all session details.
2	app_user_free.sql	To show the space used by app users in the database.
3	ash_pga_sql.sql	Shows details of top PGA consuming SQLs captured in ASH.
4	asm_disk_list.sql	shows ASM disks.
5	asm_free.sql	Shows free space in ASM disk groups.
6	awr_base_line.sql	To create baseline for good SQL plan captured in AWR.
7	awr_bw_hist.sql	Shows bandwidth usage history captured in awr.
8	awr_info.sql	Shows AWR configuration details.
9	awr_cpu_hist.sql	CPU usage history captured in awr.
10	awr_io_hist.sql	Shows I/O usage history captured in awr.
11	awr_iops_hist.sql	Shows IOPS history captured in awr.
12	awr_load_sts.sql	Loads SQL Tuning Set from AWR.
13	awr_inst_hist.sql	Instance usage history captured in awr.
14	awr_plan_hist.sql	To show AWR plan changes for SQL Id.
15	awr_set_ret.sql	To set AWR retention period.
16	awr_snap_list.sql	Shows list of snapshots in awr.
17	awr_sql_list.sql	Shows captured SQL in awr.
18	awr_sql_plan.sql	Shows SQL Plan for the given SQL ID.
19	awr_sql_text.sql	Shows SQL text for the given sql_id captured in awr.
20	awr_sys_hist.sql	Shows sysmetric history captured in awr.
21	awr_tab_io.sql	Table i/o history captured in awr.
22	awr_tbs_hist.sql	Shows tablespace growth statistics captured in AWR.
23	awr_top_wait.sql	Top wait events captured in awr.
24	awr_tps_hist.sql	Shows transactions per second history captured in awr.
25	awr_uns_plan.sql	To show unstable plans.
26	cr_ctl.sql	To unload the data into a flat file. This generates SQL Loader control file & data file.
27	cr_pf.sql	Generates profile creation script.
28	cr_rl.sql	Generates non system roles (12c).
29	cr_sch.sql	Generates schema creation script for the given schema.
30	cr_seed_ts.sql	Generates non system permanent seed tablespace creation script with auto extend.
31	cr_ts.sql	Generates non system permanent tablespace creation script.
32	cr_us.sql	Generates user creation script (12c).
33	cre_tune_set.sql	To create SQL Tuning Set.

34	db_info.sql	Shows database information.
35	del_dat.sql	Deletes rows in the current schema.
36	del_dup.sql	The following script deletes the duplicate records in the given table.
37	dg_err.sql	Checks Data Guard error in primary.
38	dg_gap.sql	To display data guard archive log gap.
39	dg_info.sql	To display data guard information.
40	dg_mrp_sts.sql	Shows MRP process status in standby. Run in primary database.
41	dg_mtr.sql	Data Guard Monitoring. This shows ARC, MRP and RFS processes status.
42	dg_set_ma.sql	Sets standby db to max availability.
43	dg_set_mp.sql	Sets standby db to max protection.
44	dg_sp_mrp.sql	To stop managed recovery process.
45	dg_sp_stb.sql	To stop MRP and standby database.
46	dg_st_mrp.sql	To start managed recovery process.
47	dg_st_stb.sql	To start standby database and MRP.
48	dg_start.sql	To start Data Guard.
49	dg_stop.sql	To stop Data Guard.
50	dg_sts.sql	Data Guard Status
51	dg_sw_log.sql	To switch logfile.
52	dg_swo_pri.sql	To switchover standby db to primary.
53	dg_swo_stb.sql	To switchover primary db to standby.
54	dg_swo_sts.sql	Shows data guard switch over status.
55	dp_job_prg.sql	To check the data pump job progress.
56	dp_reco.sql	The following script shows the tablespace & user space utilization and their mapping to help during data pump restore.
57	fra_free.sql	Shows free space in flash recovery area.
58	gen_sch_ro.sql	This script makes schema read only.
59	gen_sch_rw.sql	This script makes schema read write.
60	ggp.sql	This script spools the output to a file named ggp.out
61	ggs.sql	To create Golden Gate User.
62	ggt.sql	To create Golden Gate Tablespace.
63	inst_info.sql	To show instance details.
64	last_acc_tab.sql	Shows the last accessed tables.
65	list_base_line.sql	Lists all baselines.
66	load_sql_plan.sql	Loads desired SQL plan into SQL Plan Baseline from SQL Tuning Set.
67	pin_tune_task.sql	To load the given SQL profile.
68	rmn_bct_info.sql	Shows RMAN block change tracking information.
69	rmn_bkp_hist.sql	Shows RMAN backup details.
70	rmn_cfg.sql	RMAN Configuration in 10g.
71	rmn_cmp_hot_bkp.sql	RMAN complete hot backup.

72	rmn_col_bkp.sql	RMAN Cold Backup.
73	rmn_hkf_cat.sql	RMAN with catalog housekeeping.
74	rmn_hkf_ctl.sql	RMAN without catalog housekeeping.
75	rmn_inc_hot_bkp.sql	RMAN incremental hot backup.
76	rmn_job_prg.sql	Shows RMAN job progress.
77	rmn_rco_dsk_bkp.sql	RMAN restoration from disk.
78	rmn_rco_fra_bkp.sql	RMAN restoration from FRA.
79	rmn_ses_info.sql	Shows RMAN session details.
80	rmn_stb_bkp.sql	RMAN backup for standby database.
81	rmn_stb_stp.sql	Standby setup using RMAN backup.
82	run_sql_mtr.sql	Run SQL Monitor to for given SQL Id.
83	sch_aud_act.sql	Shows audit actions in the database.
84	sch_dis_con.sql	To disable constraints in the given schema.
85	sch_dis_rcon.sql	To disable referencial constraints in the given schema.
86	sch_drop_obj.sql	To drop all existing objects in the given schema.
87	sch_ena_con.sql	To enable constraints in the given schema.
88	sch_ena_rcon.sql	To enable referencial constraints in the given schema.
89	sch_obj_gra.sql	This script provides object grants from source schema to the given target schema.
90	sch_row_cnt.sql	To count the number of rows of each table in the given schema.
91	sch_ses_info.sql	Shows session details for the given schema.
92	sch_tab_frag.sql	Shows table fragmentation for the given schema.
93	sch_tab_gra.sql	This script provides table grants from source schema to the given target schema.
94	ses_info.sql	Shows all session details.
95	ses_long_ops.sql	Shows long running sessions.
96	ses_sp_dtrc.sql	This script stops deep tracing the sql session for the given sid, serial#.
97	ses_sp_trc.sql	The following script stop tracing the sql session for the given sid, serial#.
98	ses_st_dtrc.sql	This script starts deep tracing the sql session for the given sid, serial#.
99	ses_st_trc.sql	The following script start tracing the sql session for the given sid, serial#.
100	set_my_tfi.sql	Sets my trace file identifier.
101	sho_act_sort.sql	To display the active sorts information.
102	sho_all_ses.sql	Shows all user session details.
103	sho_alt_log.sql	Shows alert log errors.
104	sho_arc_log.sql	Shows arch log switches (1296168.1).
105	sho_asm_hwm.sql	High water mark in the data files for the given ASM disk group.
106	sho_aud_act.sql	Shows audit actions in the database.
107	sho_aux_occ.sql	Shows sysaux occupants.

108	sho_bc_obj.sql	This script shows objects in b/c.
109	sho_bit.sql	Shows Oracle Bit (32-Bit or 64-Bit).
110	sho_blk_ses.sql	Shows blocked sessions including SQL text.
111	sho_col_com.sql	This script shows column comments.
112	sho_dat_dic.sql	This script shows data dictionary.
113	sho_db_comp.sql	To display database components.
114	sho_db_hwm.sql	High water mark in all data files.
115	sho_db_io.sql	This script shows the I/O generated per day.
116	sho_db_lock.sql	This script shows locks in the database.
117	sho_db_prop.sql	To display database properties.
118	sho_db_psu_11g.sql	To display database patchset update in 11g.
119	sho_db_psu_12c.sql	To display database patchset update in 12c.
120	sho_db_psu_19c.sql	To display database patchset update in 19c.
121	sho_db_scn.sql	Shows database SCN.
122	sho_db_time.sql	shows the database current time in 100th's of a second.
123	sho_dba_dir.sql	Shows DBA directories.
124	sho_dead_lock.sql	This script shows dead locks in the database.
125	sho_dep_tab.sql	This script shows child/dependent tables for the given table.
126	sho_df_hwm.sql	High water mark in given data file.
127	sho_eg.sql	Show script example.
128	sho_exc_pls.sql	To list the sessions that are currently executing stored code.
129	sho_fk_col.sql	This script shows the Foreign Key column(s) in the given table.
130	sho_gg_cap.sql	Shows Golden Gate capture details.
131	sho_gg_scn.sql	Shows SCN that can be used to start Golden Gate extract.
132	sho_hit_ratio.sql	To identify the hit ratio.
133	sho_host_cpu.sql	Show host cpu utilization.
134	sho_idx_col.sql	This query shows the indexed columns of the given table.
135	sho_inv_obj.sql	This script shows invalid objects.
136	sho_keep_obj.sql	This script shows objects in k/p.
137	sho_mem_dtl.sql	Query v\$memory_dynamic_components. Show sizes in KB.
138	sho_my_ses.sql	Shows my session details.
139	sho_my_trc.sql	Shows my session trace files.
140	sho_nsp.sql	Shows Non System Profiles (11g).
141	sho_nsr.sql	Shows Non System Roles (11g).
142	sho_nsu.sql	Shows Non System Users (12c).
143	sho_obj_cnt.sql	This script shows object count.
144	sho_obj_lock.sql	This script shows sessions info for the given locked object.

145	sho_ora_home.sql	Shows Oracle Home directory.
146	sho_pid.sql	Shows process details.
147	sho_pk_col.sql	This script shows the primary key column(s) in the given table.
148	sho_rac_con.sql	This script shows RAC connections.
149	sho_sch_bw.sql	Schema's current bandwidth usage.
150	sho_sch_lock.sql	This script shows sessions info for the given schema.
151	sho_sch_ses.sql	Shows session details for the given schema.
152	sho_sch_size.sql	Shows size of the given schema.
153	sho_scn_log.sql	Shows log seq# for the given SCN.
154	sho_ses_dtl.sql	Shows session details.
155	sho_ses_lock.sql	This script shows locked objects for the given session.
156	sho_ses_trc.sql	Shows session trace files.
157	sho_sid.sql	Shows session details.
158	sho_sps.sql	Statspack Snapshots
159	sho_sql_plan.sql	Shows SQL Plan for the given SQL ID.
160	sho_su.sql	Shows System Users (12c)
161	sho_sys_ses.sql	Shows system/background session details.
162	sho_tab_com.sql	This script shows table comments.
163	sho_tab_size.sql	This script shows the table size.
164	sho_tde_wall.sql	To display TDE wallet details.
165	sho_top_sql.sql	Shows top ten resource consuming sql(s).
166	sho_tpd.sql	Shows transactions per day.
167	sho_ts_hwm.sql	High water mark in given tablespace.
168	sho_tx_sql.sql	Shows sql that causes enq: TX - row lock contention wait.
169	user_info.sql	Shows details of the given user.
170	shrink_db.sql	To deallocate the space above high watermark in all the data files.
171	shrink_ts.sql	To deallocate the space above high watermark in the given tablespace.
172	sid_ses_info.sql	Shows session details for the given sid.
173	sql_ses_info.sql	Shows session details for the given sql id.
174	sql_svr_bkp.sql	This script does SQL Server backup.
175	run_tune_task.sql	Creates SQL Tuning Task and reports advisor recommendations.
176	stat_job_dtl.sql	Shows auto statistics gathering job configuration details.
177	stat_job_hist.sql	Shows auto statistics gathering job history.
178	stat_job_off.sql	To disable statistics gathering job.
179	stat_job_on.sql	To enable statistics gathering job.
180	stat_job_reset.sql	To reset statistics gathering job.
181	stat_job_run.sql	To run statistics gathering job.
182	stat_job_sch.sql	To gather schema statistics.
183	stat_job_ses.sql	Shows auto statistics gathering job sessions.

184	stat_job_set.sql	To adjust statistics gathering job settings.
185	stat_job_stop.sql	To stop statistics gathering job.
186	stat_job_tab.sql	To gather table statistics.
187	stp_ora_jvm.sql	To setup JVM feature in Oracle.
188	stp_ora_xml.sql	To setup XML feature in Oracle.
189	stp_pvf.sql	Custom Password Verify Function.
190	stp_rco.sql	Database Recovery Setup.
191	stp_sec.sql	Database Security Setup
192	syn_trg.sql	Synonym trigger to avoid synonyms.
193	tab_part_col.sql	Shows table partition columns.
194	tab_part_stat.sql	Shows table partition statistics.
195	temp_free.sql	Shows free space in all temp tablespaces.
196	top_big_idx.sql	Shows top big indexes in the database.
197	top_big_tab.sql	Shows top big tables in the database.
198	top_cur_dtl.sql	Shows details of top open cursors.
199	top_cur_ses.sql	Shows details of top open cursor sessions.
200	top_pga_ses.sql	Shows details of top PGA consuming sessions.
201	ts_frag.sql	This script shows fragmentation at tablespace level.
202	ts_free.sql	Shows free space in all permanent tablespaces.
203	sho_data_file.sql	Datafile info for the given tablespace.
204	ts_us.sql	The following script shows mapping between tablespaces and users.
205	tts_cfg_10g.sql	Generates TTS migration scripts.
206	tts_gen_scr.sql	Generates TTS migration scripts.
207	undo_free.sql	Shows free space in all undo tablespaces.
208	us_ts.sql	The following script shows mapping between users and tablespaces.
209	user_free.sql	The following script shows the space used by each user in the database.
210	sch_inv_obj.sql	This script shows invalid objects for the given schema.
211	sch_obj_cnt.sql	This script shows object count for the given schema.
212	flush_sq.sql	To flush SQL from the shared pool.
213	awr_data_snap_hist.sql	Shows database growth by each snap captured in AWR.
214	awr_data_day_hist.sql	Shows database growth by each day captured in AWR.
215	awr_data_week_hist.sql	Shows weekly database growth captured in AWR.
216	rmn_reco_time.sql	Shows the last successful RMAN backup time.
217	stat_job_win.sql	Shows stat job scheduler windows.
218	list_sql_prof.sql	Shows SQL Profiles.
219	sys_inv_obj.sql	This script shows system invalid objects.
220	awr_undo_hist.sql	Shows undo tablespace usage captured in awr.
221	sho_idx_size.sql	This script shows the index size for the given table.
222	sho_db_rp.sql	Shows database restore points.
223	sga_free.sql	Shows free space in SGA.
224	awr_tune_task.sql	Creates SQL Tuning Task for SQL captured in AWR

		and reports advisor recommendations.
225	drop_tune_task.sql	To drop SQL Tuning task.
226	drop_sql_prof.sql	To drop SQL profile.
227	dis_sql_prof.sql	To disable SQL profile.
228	ena_sql_prof.sql	To enable SQL profile.
229	ash_blk_ses.sql	Shows blocked sessions captured in ASH.
230	sho_last_sql.sql	To display last sql ran in the current session.
231	dg_mrp_info.sql	Shows Data Guard managed recovery process (MRP) information. Run in any standby node.
232	awr_wait_hist.sql	History of given wait event captured in awr.
233	run_expl_plan.sql	Run explain plan for the given SQL text.
234	sho_expl_plan.sql	Shows explain plan.
235	cr_tab.sql	Generates table creation script for the given table.
236	cr_seq.sql	Generates sequence creation script for the given sequence.
237	sho_temp_file.sql	Temp file info for the given temp tablespace.
238	sho_sga_adv.sql	Shows SGA Advisory recommendations.
239	sho_imp_rpm.sql	To monitor insert/import speed.
240	sho_tde_tbs.sql	Shows encrypted tablespaces details.
241	ssl_info.sql	To verify if SSL is being used in the connection.
242	stat_win_on.sql	To enable statistics gathering job window.
243	stat_win_off.sql	To disable statistics gathering job window.
244	awr_tbs_day_hist.sql	Tablespace growth by each day captured in AWR.

Reference

1. Interpretations

- [o] -Optional.
- [d] -Default.
- P&S -Production and Standby
- U&D -UAT and Development

Software

The following 3rd party software included in the scripts.

OSWatcher

Oracle OSWatcher Black Box (OSWbb) collects and archives operating system and network metrics that you can use to diagnose performance issues. OSWbb operates as a set of background processes on the server and gathers data on a regular basis, invoking such Unix utilities as **vmstat**, **netstat**, **iostat**, and **top**.

SQLT

SQLT is a tool to diagnose the performance of queries by their sql_ids. It suggests the improvement areas of an sql query by inputting the sql id.

Contact Us

For any issues, queries or comments please contact us at support@exadime.com.