

NAME

orahot.ksh – make Oracle database hot (online) backup.

SYNOPSIS

```
oracold.ksh help=y
oracold.ksh MySID |sid:=MySid [debug=y]
```

DESCRIPTION

orahot.ksh first generates a list of *file_names* and *tablespace names* ordered by *used space*. Put one or more tablespace in *begin backup* mode as needed so that *NUM_PROCS* (specified in configuration file) of files can be compressed simultaneously to *multiple backup directories* (*BKUP_DIRS*, specified in configuration file). *As soon as* all of the files within a tablespace are backed up, the tablespace is put back to *end backup* mode.

ALTER SYSTEM SWITCH LOGFILE statement is issued and the archive log file up to the last inactive logfile is compressed to the backup directories. The original archive log files are *not* removed. Backing up and removing the archive log files is the job of *oraredo.ksh*, not *orahot.ksh*.

initSID.ora file is backed to one of the backup directories with *unix copy*.

The *control file* is backed to one of the backup directories with *ALTER DATABASE CONTROLFILE TO <file>* statement, and *unix copy*.

The *text copy of the control file* is generated with *ALTER DATABASE CONTROLFILE TO TRACE* statement in *user_dump_dest* directory. This file is backed up to one of the backup directories with *unix copy*, and the file is renamed *gen_controlfile.sql*. After verification of successful copying, the original file is removed.

The backup file systems are expected to be backed up via data center backup service.

Features include

- Support parallel backup processes (*NUM_PROCS*). Tablespaces will be put “begin backup” mode and “end backup” mode as needed.
- Choice of compress, gzip, bzip2.
- Large file support (it is not required to have a large file capable compress, gzip, bzip2).
- Support multiple backup directories (*BKUP_DIRS*). It can be used for I/O and capacity load balancing. Backup files are dynamically allocated to directories with largest free space.
- Remote backup directory support. (Currently coded for rsh, it can be expanded to support a choice of rsh, or ssh).
- Manage the number of backup copies online (*NUM_COPIES*).
- Put tablespace in *end backup* mode and clean up processes when the backup program encounters an error.
- Program terminates itself after maximum allowed time (*MAX_SECS* in configuration file). If *MAX_SECS* is 0, the backup is skipped.
- Prevent duplicate hot backup processes from running.
- Cooperate with *oraredo.ksh* backup. *orahot.ksh* and *oraredo.ksh* wait for each other.
- Backup files under *DIRECTORY_PATH*.

INSTALLATION

To install, follow these easy 1–2–3 steps:

1. Create an install directory for hot backup *<job_dir>/ora_hot*, and under *ora_hot*, create following sub-directories:

```
cd <job_dir>/ora_hot &&
mkdir -p script etc operlog/<SID> temp/<SID> ../ora_redo/operlog/<SID>
```

Where *SID* is the actual *ORACLE_SID* name. If you have multiple *SIDs*, then just create them under

operlog and temp.

Create ../ora_redo/operlog/<SID> even if you do not use ora_redo.ksh as ora_hot.ksh places a lock there to cooperate with redo backup.

2.

```
cp orahot.ksh script/orahot.ksh
chmod 755 script/orahot.ksh
```

3. Edit etc/orahot_SID.cf from the sample configuration file shown below. If you have multiple SIDs, then just create multiple files.

```
# *
# * Where do you want the hot backup files to go?
# * It can be a directory on a remote server.
# *
BKUP_DIRS="/bkup01/hot oracle@castor:/bkup02/hot"
# *
# * How many simutaneous compress processes you want to run?
# * Rule of thumb is to set the number to be the same as the n of processors.
# * Default is 3.
# *
NUM_PROCS=3
# *
# * How many copies you want to keep online?
# * ( If you keep only 1 copy, make sure the file system backup
# * occurs after the Oracle backup finishes. )
# *
NUM_COPIES=2
# *
# * What is the maximum time you allow the backup to run?
# * Backup will kill itself exceeding the maximum allowed time.
# * Default is 999999.
# *
MAX_SECS=21600
# *
# * Where is the redo backup script? (Hot backup and redo backup work together.)
# * If you use the standard naming conventions (ie <install_dir>/ora_redo,
# * <install_dir>/ora_hot), then it will derived.
# *
# * New version (5.0+) does not rely on redo backup script anymore.
# * REDO_LOCK directory is used to prevent redo backup from running during
# * the hot backup. If you use the standard naming conventions, this
# * variable is derived (<install_dir>/ora_redo/operlog/$ORACLE_SID/lock).
# *
# * If you are sure redo backup is not run and no lock is needed, then you can
# * define REDO_WAIT_MINS (see below) as a negative number to bypass the locking
# * mechanism.
# *
# REDO_PROG="/ds1-export/apps/jobs/ora_redo/script/oraredo.ksh" (deprecated)
# REDO_LOCK="/ds1-export/apps/jobs/ora_redo/operlog/$ORACLE_SID/lock" (derivable)
```

```

# *
# * Where does redo backup go?
# * If you use the standard naming conventions (ie <install_dir>/ora_redo,
# * <install_dir>/ora_hot), then it will derived.
# *
# * This variable is not used in new version (5.0+).
# *
# REDO_BKUP_DIRS="/bkup01/redo /bkup02/redo" (deprecated)
# *
# * How long do you want to wait when another process is running redo backup?
# * Default is 60.
# *
REDO_WAIT_MINS=60
# *
# * What cat command do you use to cat the data files (pipe to
# * compress utility). Default is "cat", and if "cat" can not
# * open first controlfile due to restricted permission ocfs,
# * then cat is defined as $TMPDIR/cat which is:
# * "dd o_direct=yes ${1:+if=$1} bs=8k 2>/dev/null". If this logic
# * does not work, define CAT.
# *
CAT="cat"
# CAT="/path/to/special/cat" where cat is something like:
# "dd o_direct=yes ${1:+if=$1} bs=8k 2>/dev/null"
# *
# * What command do you use to compress a file to standard output?
# * Default is "compress -c".
# *
COMPRESS_C="/usr/bin/compress -c"
# COMPRESS_C="/usr/local/bin/gzip -c"
# COMPRESS_C="/usr/local/bin/bzip2 -c"
# *
# * Whether to copy files under each DIRECTORY_PATH
# * from dba_directories.
# *
COPY_DIRECTORY_PATH=Y
# *
# * FILTER_DIRECTORY_PATH: to include or exclude certain
# * OWNER, DIRECTORY_NAME, or DIRECTORY_PATH from DBA_DIRECTORIES.
# *
FILTER_DIRECTORY_PATH="where DIRECTORY_NAME not in ('AUDIT_DIR')"
# *
# * Where can you find submit?
# * /usr/local/bin/submit is the default.
# *
SUBMIT=<job_dir>/common/submit
# SUBMIT=/usr/local/bin/submit

```

```

# *
# * What is the directory where you can find oracle-SID?
# * /etc/init.d is the default.
# *
# * The oracle-SID is used to verify if Oracle is up before and after
# * the backup. If you want to bypass the tests, then define INIT_D as
# * "NA" (Not Apply).
# *

INIT_D=<job_dir>/common
# INIT_D=/etc/init.d

# *
# * You can define pre and post processing command.
# *

# PRE_PROCESSING="/bin/true"
# POST_PROCESSING="/bin/true"

```

RESTORE

The `ora_hot.ksh` copies Oracle data files, control files, and archive log files generated *during* the hot backup. The text copy of the control file `gen_controlfile.sql`, and parameter file `initSID.ora` are also backed up gratuitously.

A sample hot backup file set is shown below:

```

/bkup01/hot/ora901/ora10/oradata/ora901/arch/1_63.dbf
/bkup01/hot/ora901/ora10/oradata/ora901/arch/1_64.dbf
/bkup01/hot/ora901/ora10/oradata/ora901/control01.ctl
/bkup01/hot/ora901/ora10/oradata/ora901/control02.ctl
/bkup01/hot/ora901/ora10/oradata/ora901/control03.ctl
/bkup01/hot/ora901/ora10/oradata/ora901/cwmlite01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/drsys01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/example01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/indx01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/system01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/tools01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/undotbs01.dbf.Z
/bkup01/hot/ora901/ora10/oradata/ora901/users01.dbf.Z
/bkup01/hot/ora901/gen_controlfile.sql
/bkup01/hot/ora901/initora901.ora

```

To restore, first copy Oracle data files, control files, and archive log files to where they were. As the original file path are recorded in the backup set, it is very easy to automate the task. The following *ad hoc* script does this.

```

#!/bin/ksh

find /bkup01/hot/ora901/ora* -type f | while read i; do
    j=${i#*ora901}
    dir=${j%/*}
    file=${j%.Z}

    CAT=cat
    [[ $i = *.Z ]] && CAT=zcat

    mkdir -p $dir
    $CAT $i > $file
done

```

What it does is something like,

```
mkdir -p /ora10/oradata/ora901
zcat /bkup01/hot/ora901/ora10/oradata/ora901/system01.dbf.Z \
> /ora10/oradata/ora901/system01.dbf
```

Once all the files are in place, we can go ahead to recover and open the database. A sample session is shown below,

```
SQL> startup mount

ORACLE instance started.
Total System Global Area  63734856 bytes
Fixed Size                 279624 bytes
Variable Size             46137344 bytes
Database Buffers         16777216 bytes
Redo Buffers              540672 bytes
Database mounted.

SQL> recover database using backup controlfile until cancel
ORA-00279: change 1329616 generated at 05/02/2002 13:30:39
+ needed for thread 1
ORA-00289: suggestion : /ora10/oradata/ora901/arch/1_63.dbf
ORA-00280: change 1329616 for thread 1 is in sequence #63
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
ORA-00279: change 1329687 generated at 05/02/2002 13:40:42
+ needed for thread 1
ORA-00289: suggestion : /ora10/oradata/ora901/arch/1_64.dbf
ORA-00280: change 1329687 for thread 1 is in sequence #64
ORA-00278: log file '/ora10/oradata/ora901/arch/1_63.dbf'
+ no longer needed for this recovery
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
ORA-00279: change 1329692 generated at 05/02/2002 13:40:46
+ needed for thread 1
ORA-00289: suggestion : /ora10/oradata/ora901/arch/1_65.dbf
ORA-00280: change 1329692 for thread 1 is in sequence #65
ORA-00278: log file '/ora10/oradata/ora901/arch/1_64.dbf'
+ no longer needed for this recovery
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
CANCEL
Media recovery cancelled.

SQL> alter database open RESETLOGS
2 /

Database altered.
```

That is it, happy restore!

SEE ALSO

submit help=y, rc.oracle help=y, oraredo.ksh help=y.

AUTHOR

Michael Wang <xw73@columbia.edu> with help from many others acknowledged in version notes.

This is free software. You may copy or redistribute it under the same terms as Perl. However, if you modify it, you need to send the modification to the author via email.