Oracle DB Daily maintenance Manual (Scripts)

START/STOP CODA SERVICES:

Run following scripts to start coda service on each partition

1. Login coda Webservers (codaweb1 and codaweb2) as codaops

   $ su – codaops

   An operation menu will be started automatically as:
   ```
   *********************************************************
   ***                                                 ***
   ***      CODA Environment Recycle Menu              ***
   ***                                                 ***
   *********************************************************


    ========   Stop Steps   ============

    101 => Stop CODA Web Services

    ========   Startup Steps ==========

    102 => Start CODA Web Services

    ==================================

    108 => CODA WebService Status

    ==================================
    q => Exit the menu
   ```

   select 102 => Start CODA Web Services

   After this menu appears again, **select** 108 => CODA WebService Status to confirm all CODA WebService processes are started.

2. Login coda application server (codaapp1_svc) as codaops

   $ su – codaops

   ```
   An operation menu will be started automatically as:
   *********************************************************
   ***                                                 ***
   ***      CODA Environment Recycle Menu              ***
   ***                                                 ***
   *********************************************************


    ========   Stop Steps   ============

    201 => Stop CODA Application Server

    ========   Startup Steps  ==========
   ```

```
 202 => Start CODA Application Server

 ====================================

 208 => CODA Application Server Status

 ====================================

 q => Exit the menu



select 202 => Start CODA Application Server
```

After this menu appears again, **select** 208 => CODA Application Server Status to confirm all CODA Application processes are started.

3. Login coda DB server (codadb1_svc) as codaops

    $ su – codaops

```
An operation menu will be started automatically as:
 *********************************************************
 ***                                                 ***
 ***     CODA Environment Recycle Menu               ***
 ***                                                 ***
 *********************************************************

 ========    Stop Steps    ============

 301 => Stop Oracle Database

 ========     Startup Steps   =========

 302 => Start Oracle Database

 ========    Process Verify   =========

 308 => Oracle Database Status

 ====================================

 q => Exit the menu



select 302 => Start Oracle Database

After this menu appears again, select 308 => Oracle Database Status to confirm
Oracle Database is started.
```

SCHEDULED JOBS:

Codaweb1/codaweb2

Job	Frequency	Location	
NIM MKSYSB	Weekly	/usr/local/scripts/client.mksysb.ksh	
CFG2HTML report	Weekly	/usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1	
Rsync to DR	Daily	/usr/local/scripts/codaweb1_rsync_dr	
CODAWEB1 system error monitor	Daily	/coda/codaweb1/monitor/alertDog.ksh	
CODA Subsystem monitor	Daily	/coda/codaweb1/monitor/monProcess.ksh	
CODA Clean	Daily	/coda/codaweb1/script/codaClean.ksh	
Tomcat Recycle	Weekly	/coda/codaweb1/script/tomcat_recycle.ksh	
nmon Report	Daily	/usr/local/scripts/nmon.ksh	

```
codaweb1:lchen@/home/codaops>crontab -l
# @(#)08      1.15.1.3  src/bos/usr/sbin/cron/root, cmdcntl, bos610 2/11/94 17:19:47
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
# bos610 src/bos/usr/sbin/cron/root 1.15.1.3
#
# Licensed Materials - Property of IBM
#
# COPYRIGHT International Business Machines Corp. 1989,1994
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# IBM_PROLOG_END_TAG
#
# COMPONENT_NAME: (CMDCNTL) commands needed for basic system needs
#
# FUNCTIONS:
```

```
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989,1994
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#0 3 * * * /usr/sbin/skulker
#45 2 * * 0 /usr/lib/spell/compress
#45 23 * * * ulimit 5000; /usr/lib/smdemon.cleanu > /dev/null
0 11 * * * /usr/bin/errclear -d S,O 30
0 12 * * * /usr/bin/errclear -d H 90
0,5,10,15,20,25,30,35,40,45,50,55 * * * * /usr/sbin/dumpctrl -k >/dev/null 2>/dev/null
0 15 * * *  /usr/lib/ras/dumpcheck >/dev/null 2>&1


55 23 * * * /var/perf/pm/bin/pmcfg  >/dev/null 2>&1     #Enable PM Data Collection


#CGI
# Weekly NIM MKSYSB
0 12 * * 0 /usr/local/scripts/client.mksysb.ksh > /dev/null 2>&1


# Weekly CFG2HTML report
0 7 * * 0 /usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1


# MPG Power Navigator data collector Version 6.9
1,16,31,46 * * * * cd /usr/local/mpg; /usr/local/mpg/start_nmon_now -f -q
25 4 * * * cd /usr/local/mpg; /usr/local/mpg/cend.sh -c /usr/local/mpg/pn.config
35 4 * * * cd /usr/local/mpg; /usr/local/mpg/pndcm.sh -c /usr/local/mpg/pn.config
clean
30 4 * * * cd /usr/local/mpg; /usr/local/mpg/rackdata.sh -c /usr/local/mpg/pn.config
30 5 2 * * cd /usr/local/mpg; /usr/local/mpg/pndchk.sh
0 5 * * * cd /usr/local/mpg; ./pngetinfo.sh -qa


#Rsync to DR
00 03 * * * /usr/local/scripts/codaweb1_rsync_dr > /dev/null 2>&1
0 0 * * 0 /usr/local/scripts/rsync_log_cleanup.ksh > /dev/null 2>&1


# ===  LII Starts ====
#CODAWEB1 system error monitor;
1 * * * 1-5 /coda/codaweb1/monitor/alertDog.ksh > /coda/codaweb1/monitor/alertDog.out
2>&1


#CODA Subsystem monitor;
1,6,11,16,21,26,31,36,41,46,51,56 * * * * /coda/codaweb1/monitor/monProcess.ksh >
/coda/codaweb1/monitor/monProcess.out 2>&1
1 6,9,12,15,18 * * * /coda/codaweb1/monitor/monSpace.ksh >/dev/null 2>&1


#CODA Clean ;
1 0 * * * /coda/codaweb1/script/codaClean.ksh >/dev/null 2>&1


#Tomcat Recycle
```

```
1 22 * * 0
/coda/codaweb1/script/tomcat_recycle.ksh>>/coda/codaweb1/log/tomcat_recycle.out 2>&1

#nmon
#1 0 * * 1-5 /coda/nmon/coda-nmon.ksh >>/coda/nmon/coda-nmon.out 2>&1
# ===  LII ENDS ====

#Daily error report redirection
0 0 * * * /usr/local/scripts/DailyError_Archive.ksh >
/usr/local/logs/DailyError_Archive.log 2>&1

#Daily nmon Report
0 0 * * * /usr/local/scripts/nmon.ksh > /usr/local/logs/nmon.out 2>&1
```

codaweb1:lchen@/coda/codaweb1/script>cat codaClean.ksh

```
#!/usr/bin/ksh

TODAY="`date +%Y`-`date +%m`-`date +%d`"

echo $TODAY

#cd /usr/apache-tomcat-6.0.28/logs
#cp -p catalina.out catalina.out.$TODAY
#cat /dev/null>catalina.out

#find . -name "catalina*.log" -mtime +3 -exec rm -f {} \;
#find . -name "*.out.*" -mtime +3 -exec rm -f {} \;

cd /usr/local/apache2/logs
cp -p access_log access_log.$TODAY
cat /dev/null>access_log

find . -name "*_log.*" -mtime +3 -exec rm -f {} \;
```

codaweb1:lchen@/coda/codaweb1/script>cat tomcat_recycle.ksh

```
#!/usr/bin/ksh

export JAVA_HOME=/usr/java6_64/jre

PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr/vac/bin
:$JAVA_HOME/bin:.

export PATH

echo "Stop WebService @ `date` ....\n"
/coda/codaweb1/bin/stopcodaweb1.ksh

sleep 300

echo "\n Clean the log files ....\n"
cd /usr/apache-tomcat-6.0.28/logs/archive
rm -rf *
mv ../*.log .
mv ../*.out .
```

```
        echo "\nStart WebService @ `date` ....\n"
        /coda/codaweb1/bin/startcodaweb1.ksh

        sleep 120
        for process in DSS http tomcat
        do
          proNo=`ps -ef|grep $process|wc -l`
          if [[ $proNo -eq 0 ]]
            then
              mail -s "ALERT:$process is not running on `hostname` @ `date`"
        ""</dev/null
          fi
        done
```

lchen@codaweb2:/home/lchen># cat /coda/codaweb2/bin/startcodaweb2.ksh

```
        #!/usr/bin/ksh

        Log=/coda/codaweb2/bin/startcodaweb2.out
        LsvLog=/coda/codaweb2/bin/coda-dssstrlsv.out

        date > $Log

        #Start apache HTTP Server;
        cd /usr/local/apache2/bin
        echo "\n == Start Apache HTTP Server ..." >>$Log
        ./apachectl start
        if [ $? -eq 1 ]
         then
           echo "Problem to start HTTP Server ...quit\n" >>$Log
           exit 1
         else
           echo "HTTP Server has been started\n" >>$Log
        fi

        #Start apache TomCat Server;
        echo "\n == Start TomCat ..." >>$Log
        cd /usr/apache-tomcat-6.0.28/bin
        ./startup.sh|tee -a $Log
        if [ $? -eq 1 ]
         then
           echo "Problem to start TomCat ...quit\n" >> $Log
           exit 1
         else
           echo "TomCat has been started\n" >>$Log
        fi

        #Start CODA web application;
        echo "\n == Start CODA-DSSSTRLSV Service..." >>$Log
        cd /usr/coda/instances/CODAPROD/DSS
        nohup ./dssstrlsv >$LsvLog 2>&1 &
        if [ $? -eq 1 ]
         then
           echo "Problem to start CODA web application ...\n" >>$Log
           echo "quit.....\n"
           exit 1
```

```
         else
            echo "Coda web application has been started\n" >>$Log
         fi
```

codaweb1:lchen@/coda/codaweb1/bin>cat stopcodaweb1.ksh

```
         #!/usr/bin/ksh

         Log=/coda/codaweb1/bin/stopcodaweb1.out

         echo "\n===============================================\n">>$Log

         date > $Log

         #Stop CODA web application;
         echo "\n == Stop Coda Web Process .....\n" >>$Log
         for pid in `ps -ef|grep "/usr/coda/instances/CODAPROD/DSS"|grep -v "grep"|awk
         '{print $2}'`
         do
         echo "Here is the PID for web Application to kill : $pid\n" >> $Log
         kill $pid
         done

         #Stop apache TomCat Server;
         echo "\n == Stop TomCat .....\n" >>$Log
         cd /usr/apache-tomcat-6.0.28/bin
         ./shutdown.sh
         if [ $? -eq 1 ]
          then
            echo "\n Problem to stop TomCat ...quit\n" >>$Log
            exit 1
          else
            echo "\n TomCat has been stopped\n" >>$Log
         fi

         #Stop apache HTTP Server;
         echo "\n == Stop Apache HTTP Server .....\n" >>$Log
         cd /usr/local/apache2/bin
         ./apachectl stop
         if [ $? -eq 1 ]
          then
            echo "\n Problem to stop HTTP Server ...quit\n" >>$Log
            exit 1
          else
            echo "\n HTTP Server has been stopped\n" >>$Log
         fi

         echo "===================================================\n" >>$Log
```

codaweb1:lchen@/coda/codaweb1/monitor>cat monProcess.ksh

```
         #!/bin/ksh
         #############################################################################
         #
         # Name:        monProcess.ksh
         #
         # Reference:   n/a
```

```
        #
        # Description: monitor the CODA Sub System process;
        #
        # Parameters:  None
        #
        # Modification History:
        #
        #                Date            Name            Description
        #                ------------------------------------------------------
        #                2010-10-06      ""
        #
        ###########################################################################
        set -v
        set -x

        # log and reference files
        msgLog=/coda/codaweb1/log/monitor/monProcess.log

        # email user list
        set -A AlertList ""

        date >> $msgLog
        echo "\n=============================================================\n" >>$msgLog
        #Check HTTP process;
        ps -ef|grep http
        if (( $? == 1 ))
        then
          echo "Alert: HTTP Server is not running ....\n"|tee -a $magLog
          mail -s "No HTTP Process is running on `hostname`" ${AlertList[*]} </dev/null
        fi

        #Check TOMCAT process;
        ps -ef|grep tomcat
        if (( $? == 1 ))
        then
          echo "Alert: TOMCAT is not running ....\n"|tee -a $magLog
          mail -s "TOMCAT is not running on `hostname`" ${AlertList[*]} </dev/null
        fi

        #Check CODA WebService;
        ps -ef|grep DSS
        if (( $? == 1 ))
        then
          echo "Alert: DSS is not running ....\n"|tee -a $magLog
          mail -s "DSS is not running on `hostname`" ${AlertList[*]} </dev/null
        fi

        echo "\n=============================================================\n" >>$msgLog

        exit 0

codaweb1:lchen@/coda/codaweb1/monitor>cat /usr/local/scripts/codaweb1_rsync_dr
        #!/usr/bin/ksh93

        #   Script written by Dave Benvie CGI
```

```
#    This script rsync's file systems/directories identified by
#    Livingston Internation to be kept in syncronization with their current
#    DR partitions.
#    codaweb1

RSYNC_LIST='/coda /usr/coda'

PDC_IP='10.153.255.210'

LOG=/usr/local/logs/rsync/codaweb1_rsync_log`date +%Y"-"%m"-"%d`

for i in $RSYNC_LIST
do
   echo "Starting rsync for file system $i `date`" >> $LOG
   echo "Statistics of syncronization below" >> $LOG
   echo >> $LOG
   rsync -acvps --del $i/ root@$PDC_IP:$i >> $LOG
   RESULT=${?}
   echo "${RESULT}" >/tmp/rsyncStatus.$(uname -n)
   echo >> $LOG
   echo "Finished syncronizarions for file system $i." >> $LOG
   echo "####################################################" >> $LOG
   echo  >> $LOG
done
```

root@codaweb1:/home/codaops># cat .profile
```
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.

export PATH

if [ -s "$MAIL" ]           # This is at Shell startup.  In normal
then echo "$MAILMSG"        # operation, the Shell checks
fi                          # periodically.

alias rm='rm -i'

trap 'exit' INT

/home/codaops/bin/codarecycle.ksh

exit
```

Codaapp1/codaapp2

Job	Frequency	Location	email
NIM MKSYSB	Weekly	/usr/local/scripts/client.mksysb.ksh	""
CFG2HTML report	Weekly	/usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1	""
Rsync to DR	Daily	/usr/local/scripts/codaapp1_rsync_dr	""
system error monitor	Daily	//coda/codaapp1/monitor/alertDog.ksh	""

Monitor Space	Daily	/coda/codaapp1/monitor/monSpace.ksh	""
Monitor App	Daily	/coda/codaapp1/script/monProcess.ksh	""
CODA Clean	Daily	/coda/codaapp1/script/codaClean.ksh	""
nmon Report	Daily	/usr/local/scripts/nmon.ksh	""
ftp	Daily	/coda/codaapp1/script/ftp_Cheques.ksh /coda/codaapp1/script/ftp_EFT.ksh /coda/codaapp1/script/ftp_eStatement.ksh	""

root@codaapp1:/># crontab -l
```
# @(#)08        1.15.1.3  src/bos/usr/sbin/cron/root, cmdcntl, bos610 2/11/94 17:19:47
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
# bos610 src/bos/usr/sbin/cron/root 1.15.1.3
#
# Licensed Materials - Property of IBM
#
# COPYRIGHT International Business Machines Corp. 1989,1994
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# IBM_PROLOG_END_TAG
#
# COMPONENT_NAME: (CMDCNTL) commands needed for basic system needs
#
# FUNCTIONS:
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989,1994
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#0 3 * * * /usr/sbin/skulker
#45 2 * * 0 /usr/lib/spell/compress
#45 23 * * * ulimit 5000; /usr/lib/smdemon.cleanu > /dev/null
0 11 * * * /usr/bin/errclear -d S,O 30
0 12 * * * /usr/bin/errclear -d H 90
0,5,10,15,20,25,30,35,40,45,50,55 * * * * /usr/sbin/dumpctrl -k >/dev/null 2>/dev/null
0 15 * * *  /usr/lib/ras/dumpcheck >/dev/null 2>&1
```

```
0 0 * * * /usr/es/sbin/cluster/utilities/clcycle 1>/dev/null 2>/dev/null # HACMP for
AIX Logfile rotation

#CODAAPP1 system error monitor;
1,31 * * * * /coda/codaapp1/monitor/alertDog.ksh >/coda/codaapp1/monitor/alertDog.out
2>&1

#Coda Log Clean
58 23 * * * /coda/codaapp1/script/codaClean.ksh >/dev/null 2>&1

#Monitor Space
1 6,9,12,15,18 * * * /coda/codaapp1/monitor/monSpace.ksh >/dev/null 2>&1

#nmon
#1 0 * * 1-5 /coda/nmon/coda-nmon.ksh >> /coda/nmon/coda-nmon.out 2>&1

55 23 * * * /var/perf/pm/bin/pmcfg  >/dev/null 2>&1      #Enable PM Data Collection

# Weekly NIM MKSYSB
0 12 * * 0 /usr/local/scripts/client.mksysb.ksh > /dev/null 2>&1

# Weekly CFG2HTML report
0 7 * * 0 /usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1

# MPG Power Navigator data collector Version 6.9
1,16,31,46 * * * * cd /usr/local/mpg; /usr/local/mpg/start_nmon_now -f -q
25 4 * * * cd /usr/local/mpg; /usr/local/mpg/cend.sh -c /usr/local/mpg/pn.config
35 4 * * * cd /usr/local/mpg; /usr/local/mpg/pndcm.sh -c /usr/local/mpg/pn.config
clean
30 4 * * * cd /usr/local/mpg; /usr/local/mpg/rackdata.sh -c /usr/local/mpg/pn.config
30 5 2 * * cd /usr/local/mpg; /usr/local/mpg/pndchk.sh
0 5 * * * cd /usr/local/mpg; ./pngetinfo.sh -qa

#Rsync to DR
00 03 * * * /usr/local/scripts/codaapp1_rsync_dr > /dev/null 2>&1
0 0 * * 0 /usr/local/scripts/rsync_log_cleanup.ksh > /dev/null 2>&1

# ===== LII Starts =====
#daily ftp job;
1,6,11,16,21,26,31,36,41,46,51,56 * * * * /coda/codaapp1/script/ftp_Cheques.ksh >>
/coda/codaapp1/log/ftp/ftp_Cheques.log 2>&1
1 12,15,17,18 * * * /coda/codaapp1/script/ftp_EFT.ksh >>
/coda/codaapp1/log/ftp/ftp_EFT.log 2>&1
#1,16,31,46 8-12 * * * /coda/codaapp1/script/ftp_eStatement.ksh >>
/coda/codaapp1/log/ftp/ftp_eStatement.log 2>&1
1,16,31,46 8-17 * * * /coda/codaapp1/script/ftp_eStatement.ksh >>
/coda/codaapp1/log/ftp/ftp_eStatement.log 2>&1

#Application Monitor
1,6,11,16,21,26,31,36,41,46,51,56 * * * * /coda/codaapp1/script/monProcess.ksh >>
/coda/codaapp1/log/monProcess.log 2>&1
# ===== LII Ends =====

#Daily error report redirection
```

```
0 0 * * * /usr/local/scripts/DailyError_Archive.ksh >
/usr/local/logs/DailyError_Archive.log 2>&1

#Daily nmon Report
0 0 * * * /usr/local/scripts/nmon.ksh > /usr/local/logs/nmon.out 2>&1
```

codaapp1:lchen@/coda/codaapp1/script>cat codaClean.ksh

```
#!/usr/bin/ksh

TODAY=`date +%Y%m%d`

cd /home/codauser/bin

mkdir $TODAY

for log in $(ls coda*.out)
 do
   cp -p $log ${log}.${TODAY}
   cat /dev/null>$log
   mv ${log}.${TODAY} $TODAY
done

mv $TODAY /home/codauser/bin/log.bak
cd /home/codauser/bin/log.bak
#find . -name "coda*.out.*" -mtime +7 -exec rm -f {} \;
find . -mtime +8 -type d -exec rm -rf {} \;
```

codaapp1:lchen@/coda/codaapp1/script>cat ftp_Cheques.ksh

```
#!/usr/bin/ksh

echo "\n====== Start cheque ftp process @ `date` on `hostname` ======\n"

set -A AlertList ""

#set -A AlertList "" \
#                 computerops@"".com


lsvg -o|grep codaapp1_codavg

if [[ $? == 1 ]]
 then
   echo "*** Alert: Resource is not available on node `hostname`..."
   exit 1
fi

if [[ -f /coda/codaapp1/status/ftp_ePay.status ]]
then
 echo "\n `date`  ** ALERT : Previous job is still running on `hostname` **\n"
 mail -s "`hostname`: Previous ePay FTP is still running" ""</dev/null
 exit 0
fi

cd /codafiles/ePay/output
```

```
fileNo=`ls|grep -v "EFT"|wc -l`
if [[ $fileNo -eq 0 ]]
then
  echo "There is no file to grab ...."
  exit
fi

DATE=`date +%Y%m%d`

if [[ ! -d "/codafiles/ePay/archive/$DATE" ]]
then
   mkdir /codafiles/ePay/archive/$DATE
fi

touch /coda/codaapp1/status/ftp_ePay.status

#Get the files;
#mv /codafiles/ePay/output/* /codafiles/ePay/temp
for file in `ls|grep -v "EFT"`
do
#  mv $file /codafiles/ePay/temp
  proNo=`fuser test1.out 2>/dev/null|awk '{print $1}'|wc -l`
  if [[ $proNo -eq 0 ]]
    then
       mv $file /codafiles/ePay/temp
  fi
done

cd /codafiles/ePay/temp
for file in `ls /codafiles/ePay/temp`
do
echo "\n\n `date` -- Put file to Bottomline Server ...\n"
ftp -v -n 10.253.128.12 <<!
user codaupload Coda123
prompt
bin
put "$file"
quit
!
done

#mv /codafiles/ePay/temp/* /codafiles/ePay/archive/$DATE
#Move the files;
cd /codafiles/ePay/temp
timestamp=`date +%Y%m%d%H%M%S`
for file in `ls`
do
  mv $file /codafiles/ePay/archive/$DATE/$file.$timestamp
done

cd /codafiles/ePay/archive

find . -mtime +35 -type d -exec rm -rf {} \;

rm -rf /coda/codaapp1/status/ftp_ePay.status
```

```ksh
#!/usr/bin/ksh

echo "\n ===== Start EFT ftp process @ `date` on `hostname` =====\n"

set -A AlertList ""

#set -A AlertList "" \
#                  computerops@"".com

lsvg -o|grep codaapp1_codavg

if [[ $? == 1 ]]
 then
   echo "****** Alert: Resource is not available on node `hostname` ...\n"
   exit 1
fi

if [[ -f /coda/codaapp1/status/ftp_EFT.status ]]
then
 echo "\n `date`  ** ALERT : Previous job is still running on `hostname` **\n"
 mail -s "`hostname`: Previous EFT FTP is still running" ""</dev/null
 exit 0
fi

cd /codafiles/ePay/output

fileNo=`ls|grep "EFT"|wc -l`
if [[ $fileNo -eq 0 ]]
then
  echo "There is no file to grab ...."
  exit
fi

DATE=`date +%Y%m%d`

if [[ ! -d "/codafiles/ePay/EFT_arch/$DATE" ]]
then
   mkdir /codafiles/ePay/EFT_arch/$DATE
fi

touch /coda/codaapp1/status/ftp_EFT.status

cd /codafiles/ePay/output
for file in `ls|grep "EFT"`
do
  mv $file /codafiles/ePay/EFT
done

cd /codafiles/ePay/EFT
for file in `ls|sed 's/.dat$//'`
do
timestamp=`date +%Y%m%d%H%M%S`
```

```
#newFile=`echo $file|awk -F "-" '{print $1"-"$2"-"$3"-"$4"-"$6"-"$7"-"$8"-
"$9}'`
newFile=EFT$timestamp
ftp -v -n 10.253.129.19<<!
    user estatement Coda2012
    bin
    cd /bankout/rbc
    #cd /Bank/BMO/EFT_In
    put $file.dat $newFile.temp
    rename $newFile.temp $newFile.dat
    quit
!
mv $file.dat /codafiles/ePay/EFT_arch/$DATE/$file.dat.$timestamp
sleep 5
done

#mv /codafiles/ePay/EFT/* /codafiles/ePay/EFT_arch/$DATE
#Move the files;
#cd /codafiles/ePay/EFT
#timestamp=`date +%Y%m%d%H%M%S`
#for file in `ls`
#do
#  mv $file /codafiles/ePay/EFT_arch/$DATE/$file.$timestamp
#done

cd /codafiles/ePay/EFT_arch

find . -mtime +35 -type d -exec rm -rf {} \;

rm -rf /coda/codaapp1/status/ftp_EFT.status
```

codaapp1:lchen@/coda/codaapp1/script>cat ftp_eStatement.ksh

```
#!/usr/bin/ksh

echo "\n ====== Start eStatement FTP process `date` on `hostname` =====\n"

set -A AlertList ""

#set -A AlertList "" \
#               computerops@"".com

lsvg -o|grep codaapp1_codavg

if [[ $? == 1 ]]
 then
   echo "*** Alert: Resource is not available on node `hostname` ..."
   exit 1
fi

if [[ -f /coda/codaapp1/status/ftp_eStatment.status ]]
then
 echo "\n `date`  ** ALERT : Previous job is still running on `hostname` **\n"
 mail -s "`hostname`: Previous eStatement FTP is still running" ""</dev/null
 exit 0
fi
```

```
touch /coda/codaapp1/status/ftp_eStatement.status

umask 111

cd /codafiles/eStatement/temp
#Get the files;
echo "\n\n `date` -- Get files from BLACK BOX ...\n"
ftp -v -n 10.253.129.19<<!
user estatement Coda2012
prompt
asc
cd /Bank
mget *.txt
quit
!

fileNo=`ls *.txt|wc -l`
if [[ $fileNo -gt 0 ]]
then
echo "We have files to handle ...\n"
timestamp=`date +%Y%m%d%H%M%S`
for file in `ls *.txt`
do
##Now delete it on target site;
ftp -v -n 10.253.129.19<<!
user estatement Coda2012
cd /Bank
delete $file
quit
!

#Rename the input file...
mv $file $file.$timestamp
done
else
echo "There is no input file to handle...\n"
fi

#fileList="MT9854709202.txt MT9854709194.txt RBC1115971.txt RBC1116227.txt
RBC1116813.txt RBC1116920.txt RBC1116987.txt RBC4046223.t
xt RBC4046462.txt RBC4046637.txt RBC4056651.txt SB28658617.txt"

fileList="MT9854709202.txt MT9854709194.txt RBC1115971.txt RBC1116227.txt
RBC1116813.txt RBC1116920.txt RBC1116987.txt RBC4046223.tx
t RBC4046462.txt RBC4046637.txt RBC4056651.txt SB28658617.txt BMO1895886.txt
BMO4678384.txt BMO1891033.txt BMO1894031.txt BMO4676928
.txt BMO1890997.txt BMO4676047.txt BMO4675626.txt BMO1895851.txt BMO4678376.txt
BAM483043664936.txt BAM483043592244.txt"

#Move processed files to archive directory;
DATE=`date +%Y%m%d`
if [[ ! -d "/codafiles/eStatement/archive/$DATE" ]]
then
```

```
    mkdir /codafiles/eStatement/archive/$DATE
fi

#archive processed
cd /codafiles/eStatement/processed
timestamp=`date +%Y%m%d%H%M%S`
for done in `ls /codafiles/eStatement/processed`
do
  mv $done /codafiles/eStatement/archive/${DATE}/${done}.${timestamp}
done

find . -mtime +35 -type d -exec rm -rf {} \;

#Handle the new bank file;
cd /codafiles/eStatement/temp
fileNo=`ls|wc -l`
if [[ $fileNo -eq 0 ]]
then
  echo "There is no file to handle ...."
  exit
fi

for file in $fileList
do
  if [[ ! -a /codafiles/eStatement/input/$file ]]
    then
      #fileNo=`ls $file.*|wc -l`
      #if [[ $fileNo -gt 0 ]]
      ls $file.* 2>/dev/null
      if [[ $? -eq 0 ]]
        then
          newFile=`ls $file.*|head -1`
          mv $newFile /codafiles/eStatement/input/$file
      fi
  fi
done

rm -rf /coda/codaapp1/status/ftp_eStatement.status
```

codaapp1:1chen@/coda/hacmp>cat app_start.ksh

```
#!/usr/bin/ksh

LogFile=/coda/hacmp/app_start.out

date >$LogFile

echo "===== Starting CODA Application Server @ `date` on `hostname` ... \n"
|tee -a $LogFile

su - codauser >>$LogFile 2>&1 <<EOF
  cd /home/codauser/bin
  ./startcodaapp1.ksh
  if [ $? -ne 0 ]; then
    echo "***** ALert : problem to start CODA Application Server @ `date` on
`hostname` ...\n"|tee -a $LogFile
```

```
      exit 1
    fi
EOF

echo "===== CODA Application Server is started @ `date` on `hostname` ... \n"
|tee -a $LogFile

exit 0
```

codaapp1:lchen@/coda/hacmp>cat app_stop.ksh
```
#!/usr/bin/ksh

LogFile=/coda/hacmp/app_stop.out

date >$LogFile

echo "===== Stopping CODA Application Server @ `date` on `hostname` ... \n"
|tee -a $LogFile

su - codauser >>$LogFile 2>&1 <<EOF
  cd /home/codauser/bin
  ./stopcodaapp1.ksh
  if [ $? -ne 0 ]; then
    echo " *****  Alert : problem to stop CODA Application Server @ `date` on
`hostname` ...\n"|tee -a $LogFile
      exit 1
    fi
EOF

echo "===== CODA Application Server is stopped @ `date` on `hostname` ... \n"
|tee -a $LogFile

exit 0
```

codaapp1:lchen@/coda/codaapp1/bin>cat startcodaapp1.ksh
```
#!/usr/bin/ksh

LogFile=/coda/codaapp1/bin/startcodaapp1.out

date >$LogFile
HostName=`hostname`

su - codauser >>$LogFile 2>&1 <<EOF
  echo "Start CODA Application Server on $HostName ... \n"
  cd /home/codauser/bin
  ./startcodaapp1.ksh
  if [ $? -ne 0 ]; then
    echo "problem to start CODA Application Server on $HostName ...\n"
     exit 1
    fi
EOF

exit 0
```

codaapp1:lchen@/coda/codaapp1/bin>cat stopcodaapp1.ksh

```
#!/usr/bin/ksh


LogFile=/coda/codaapp1/bin/stopcodaapp1.out

date >$LogFile
HostName=`hostname`

su - codauser >>$LogFile 2>&1 <<EOF
  echo "Stop CODA Application Server on $HostName ... \n"
  cd /home/codauser/bin
  ./stopcodaapp1.ksh
  if [ $? -ne 0 ]; then
    echo "problem to stop CODA Application Server on $HostName ...\n"
     exit 1
  fi
EOF

exit 0
```

codaapp1:lchen@/home/codauser/bin>cat stopcodaapp1.ksh
```
#!/usr/bin/ksh

Log=/home/codauser/bin/stopcodaapp1.out

echo "\n=======================================================\n">>$Log
date >>$Log

echo "\n Stop the CODA applications now .....\n"  >>$Log

clear
list="codaprod_icm_lsv codaprod_epay_lsv codaprod_fin_lsv codaprod_fwk_lsv
codaprod_efa_lsv codaprod_cnf_lsv oas_nsv"
for lsv in $list
  do
    pid=`ps -ef|grep "$lsv"|grep -v "grep"|awk '{print $2}'`
    if [[ $pid -gt 0 ]]
      then
        kill -9 $pid|tee -a $Log
    fi
  done

for pid in `ps -ef|grep "/usr/coda/oas/bin/oas_asv"|awk '{print $2}'`
do
  kill -9 $pid
done

echo "Complete the stop procedure ....\n" >>$Log
echo "\n =================================================\n" >>$Log

exit 0
```

codaapp1:lchen@/home/codauser/bin>cat startcodaapp1.ksh
```
#!/usr/bin/ksh
```

```
CODA_USR_JAVA=/usr/java6_64/jre/bin/java
export CODA_USR_JAVA

JAVA_HOME=/usr/java6_64/jre
export JAVA_HOME

ORACLE_HOME=/home/codauser/oracle/product/10.2.0/db_1
export ORACLE_HOME

CODA_USR_CLASSPATH=$ORACLE_HOME/jdbc/lib/ojdbc14.jar
export CODA_USR_CLASSPATH

ORACLE_SID=codaprod
export ORACLE_SID

TWO_TASK=codaprod
export TWO_TASK

Log=/home/codauser/bin/startcodaapp1.out
app1Log=/home/codauser/bin/coda-nameServer.out
app2Log=/home/codauser/bin/coda-adminConsole.out
app3Log=/home/codauser/bin/coda-financial.out
app4Log=/home/codauser/bin/coda-assets.out
app5Log=/home/codauser/bin/coda-ePay.out
app6Log=/home/codauser/bin/coda-eConfigurator.out
app7Log=/home/codauser/bin/coda-icm.out

status=`tnsping codaprod|tail -1|awk '{print $1}'`

if [[ $status != 'OK' ]]
  then
    echo "Alert: Listener is not up ....\n"
    echo "CODA application is not started due to DB environment is not ready
...\n"
    exit 1
fi

echo "\n===========================================================\n">>$Log
date >> $Log

echo "\nStart the CODA application now .....\n" >>$Log

echo "\n========================================================\n">>$app1Log
date >> $app1Log
echo "\n --- starting CODA Name Server .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/OAS/oasstrnsv >>$app1Log 2>&1 &
if [ $? -eq 1 ]
 then
   echo "\n *** ALARM *** : Name Server is not up ... \n" >>$app1Log
   exit 1;
fi

echo "\n====================================================\n">>$app3Log
date >> $app3Log
echo "\n --- starting CODA Financials .... \n" >>$Log
```

```
nohup /usr/coda/instances/CODAPROD/OAS/oasstrlsv >>$app3Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : Financial Server is not up ... \n" >>$app3Log
    exit 1;
fi

echo "\n==========================================================\n">>$app2Log
date >> $app2Log
echo "\n --- starting CODA Administration Console .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/FRS/frsstrlsv >>$app2Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : Administration Console Server is not up ... \n"
>>$app2Log
    exit 1;
fi

echo "\n==========================================================\n">>$app4Log
date >> $app4Log
echo "\n --- starting CODA Assets .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/EFA/efastrlsv >>$app4Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : ASSETS is not up ... \n" >>$app4Log
    exit 1;
fi

echo "\n==========================================================\n">>$app5Log
date >> $app5Log
echo "\n --- starting CODA Addons (E-Pay) Server .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/ADD/addstrlsv >>$app5Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : ePay is not up ... \n" >>$app5Log
    exit 1;
fi

echo "\n==========================================================\n">>$app6Log
date >> $app6Log
echo "\n --- starting CODA eConfigurator .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/CNF/cnfstrlsv >>$app6Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : eConfigurator is not up ... \n" >>$app6Log
    exit 1;
fi

echo "\n==========================================================\n">>$app7Log
date >> $app7Log
echo "\n --- starting CODA ICM .... \n" >>$Log
nohup /usr/coda/instances/CODAPROD/ICM/icmstrlsv >>$app7Log 2>&1 &
if [ $? -eq 1 ]
 then
    echo "\n *** ALARM *** : ICM is not up ... \n" >>$app7Log
```

```
        exit 1;
    fi

    sleep 30

    echo "Complete the start procedure ....\n" >>$Log
```

Codadb1/codadb2

Job	Frequency	Location	email
NIM MKSYSB	Weekly	/usr/local/scripts/client.mksysb.ksh	""
CFG2HTML report	Weekly	/usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1	""
Rsync to DR	Daily	/usr/local/scripts/codadb1_rsync_dr	""
system error monitor	Daily	/coda/codadb1/monitor/alertDog.ksh	""
DB Space	Daily	/home/oracle/script/db_space_rpt.ksh /home/oracle/script/db_userRpt.ksh	""
DB Backup	Daily	/home/oracle/script/db_full_backup.ksh /home/oracle/script/db_rcp.ksh /home/oracle/script/codarpt_job.ksh /home/oracle/script/reset_flag.ksh	""
Db monitor	Daily	/coda/codadb1/monitor/monProcess.ksh	""
DB stat	Daily	/home/oracle/script/db_stat.ksh /home/oracle/script/statpack.ksh	""
Finance Report	Daily	/ora_fra/dataextract/aging/aging.ksh /ora_fra/dataextract/hyperion/extract_spe.ksh	""
??		/ora_fra/dataextract/rakesh/test.ksh	

codadb1:oracle@/home/oracle>crontab -l
```
#Db space report;
1 6 * * * /home/oracle/script/db_space_rpt.ksh >>/home/oracle/report/db_space_rpt.log
2>&1
1 2 * * * /home/oracle/script/db_userRpt.ksh >/home/oracle/report/db_userRpt.log 2>&1

#Db daily backup;
1 23 * * * /home/oracle/script/db_full_backup.ksh
>/home/oracle/report/db_full_backup.log 2>&1

1 2 * * * /home/oracle/script/db_rcp.ksh >/home/oracle/report/db_rcp.out 2>&1
```

```
31 4 * * * /home/oracle/script/codarpt_job.ksh >/home/oracle/report/codarpt_job.out
2>&1
1 22 * * * /home/oracle/script/reset_flag.ksh >/home/oracle/report/reset_flag.out 2>&1


#Db monitor;
2,32 * * * * /coda/codadb1/monitor/monProcess.ksh
>/coda/codadb1/monitor/monProcess.out 2>&1


#DB stat;
1 2 * * * /home/oracle/script/db_stat.ksh >>/home/oracle/report/db_stat.out 2>&1
1,16,31,46 * * * * /home/oracle/script/statpack.ksh >/dev/null 2>&1


#Finance Report
1 6 * * * /ora_fra/dataextract/aging/aging.ksh >/ora_fra/dataextract/aging/aging.out
2>&1
## below 1 extract are disabled as per request from Finance;
##1 18 * * 0 /ora_fra/dataextract/hyperion/extract_spe.ksh
>/ora_fra/dataextract/hyperion/extract_spe.out 2>&1
1 8 * * 2-5 /ora_fra/dataextract/hyperion/extract_spe.ksh
>/ora_fra/dataextract/hyperion/extract_spe.out 2>&1


#Rakesh
30 4 * * * /ora_fra/dataextract/rakesh/test.ksh >/ora_fra/dataextract/rakesh/test.out
2>&1
```

codadb1:oracle@/home/oracle>cat /home/oracle/script/db_space_rpt.ksh

```
#!/usr/bin/ksh

Rpt=$HOME/report/tbl_usage.out

. $HOME/.profile

cd $HOME/sql

sqlplus /nolog @tbl_usage.sql > $Rpt

mail -s "Oracle Database Space Usage Report @ `date`" "" <$Rpt
```

codadb1:oracle@/home/oracle/script>cat db_userRpt.ksh

```
#!/usr/bin/ksh

Rpt=$HOME/report/mon_user.out

. $HOME/.profile

cd $HOME/sql

VG_status=`lsvg -o|grep -E "dbvg|fravg"`
if [[ $? -ne 0 ]]
then
   echo "Database VG is not found on this NODE `hostname` ...\n"
   echo "User Audit will be skipped @ `date` ...\n"
exit 1
fi
```

```
        sqlplus / as sysdba @audit_purge.sql > $Rpt

        mail -s "`hostname`: Oracle Database User Access Report @ `date`" "" <$Rpt
```

codadb1:oracle@/home/oracle/sql>cat audit_purge.sql

```
        delete from aud$ where ntimestamp# < systimestamp - 7;

        quit;
```

codadb1:oracle@/home/oracle/script>cat db_full_backup.ksh

```
        #!/usr/bin/ksh

        Rpt=$HOME/report/backup_rpt.out

        . $HOME/.profile

        VG_status=`lsvg -o|grep -E "dbvg|fravg"`
        if [[ $? -ne 0 ]]
        then
           echo "Database VG is not found on this NODE `hostname` ...\n"
           echo "Oracle db backup will be skipped @ `date` ...\n"
        exit 1
        fi

        echo " ========================\n"
        date

        cd /ora_fra/CODAPROD/dbbackup
        rm -fe /ora_fra/CODAPROD/dbbackup/*

        cd $HOME/sql

        DB_status=`sqlplus / as sysdba @db_status.sql|grep 'ACTIVE'`

        if [[ $? -ne 0 ]]
        then
           echo "Oracel DB is not UP, backup will quit ...."
           mail -s "CODAPROD ALARM : DB backup failed due to DB is not up @ `date`" ""
        </dev/null
        exit 1
        fi

        rman TARGET / < db_full_backup.rman > $Rpt

        #Also clean the trace log;
        find /u01/app/oracle/diag/rdbms/codaprod/codaprod/trace -name "*.trc" -mtime +3
        -exec rm -f {} \;
        find /u01/app/oracle/diag/rdbms/codaprod/codaprod/trace -name "*.trm" -mtime +3
        -exec rm -f {} \;
        find /u01/app/oracle/diag/tnslsnr/codadb1/listener/alert -name "*.xml" -mtime
        +7 -exec rm -f {} \;
        find /u01/app/oracle/admin/codaprod/adump -name "*.aud" -mtime +30 -exec rm -f
        {} \;
```

```ksh
        mail -s "CODAPROD : Oracle Daily Backup Report @ `date`" "" <$Rpt
```

codadb1:oracle@/home/oracle/script>cat db_rcp.ksh

```ksh
        #!/usr/bin/ksh

        #copy DB backup to reporting server;
        echo "=== Start RCP process on `hostname` @ `date` ..."

        rcp -p /ora_fra/CODAPROD/autobackup/* codarpt:/ora_fra/CODAPROD/autobackup
        rcp -p /ora_fra/CODAPROD/dbbackup/* codarpt:/ora_fra/CODAPROD/dbbackup

        echo "=== RCP process is completed on `hostname` @ `date` ..."
```

codadb1:oracle@/home/oracle/script>cat codarpt_job.ksh

```ksh
        #!/usr/bin/ksh

        . $HOME/.profile

        typeset count=0

        while [[ $count -lt 18 ]]
        do
         cd $HOME/sql
         flag=`sqlplus codauser/codaliv@codaprod @verify_flag.sql|grep "NIGHTLY"|awk
        '{print $2}'`
         if [[ $flag == 'Y' ]]
           then
               #echo "*** `date` : Nightly job is completed ...., going to create
        special backup ...\n"
               sqlplus codauser/codaliv@codaprod @verify_flag.sql
               echo "*** `date` : Nightly job is completed ...., going to move archive
        logs ...\n"

        sqlplus / as sysdba<<EOF
        alter system archive log all;
        alter system archive log current;
        quit;
        EOF

               rcp -p /ora_fra/oradata/codaprod/archivelog/*
        codarpt:/ora_fra/oradata/codaprod/archivelog

               touch /ora_fra/CODAPROD/dbbackup/rcp.done
               rcp /ora_fra/CODAPROD/dbbackup/rcp.done
        codarpt:/ora_fra/CODAPROD/dbbackup
               break
         fi

         echo "=== `date` : Nightly job is not completed, continue to wait ...\n"
         sleep 300
         count=$(($count+1))
        done

        sleep 300
```

```
      #reset job flag;
      cd $HOME/sql
      sqlplus codauser/codaliv@codaprod @reset_flag.sql

      rm -rf /ora_fra/CODAPROD/dbbackup/rcp.done

      if [[ $count -eq 18 ]]
         then
             echo "Backup windows missed, exits program ...\n"
             mail -s "ALERT: CODAPROD BACKUP windows missed ..." ""</dev/null
      fi
```

codadb1:oracle@/home/oracle/script>cat reset_flag.ksh
```
      #!/usr/bin/ksh

      . $HOME/.profile

      cd $HOME/sql
      sqlplus codauser/codaliv@codaprod @reset_flag.sql
```

codadb1:oracle@/home/oracle/sql>cat tbl_usage.sql
```
      CONN / as sysdba;

      select b.tablespace_name, tbs_size SizeMb, a.free_space FreeMb
      from
      (select tablespace_name, round(sum(bytes)/1024/1024 ,2) as free_space
      from dba_free_space group by tablespace_name) a,
      (select tablespace_name, sum(bytes)/1024/1024 as tbs_size
      from dba_data_files group by tablespace_name
      UNION
      select tablespace_name, sum(bytes)/1024/1024 tbs_size
      from dba_temp_files
      group by tablespace_name ) b
      where a.tablespace_name(+)=b.tablespace_name;

      exit;
```

codadb1:oracle@/home/oracle/sql>cat verify_flag.sql
```
      select * from job_status where job_name='NIGHTLY';
      quit;
      codadb1:oracle@/home/oracle/sql>
```

codadb1:oracle@/home/oracle/script>cat /home/oracle/script/db_stat.ksh
```
      #!/usr/bin/ksh

      date

      . $HOME/.profile

      cd $HOME/sql

      echo "Start the Schema Stat ....\n"

      time sqlplus / as sysdba @schema_stat.sql
```

```
        echo "@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@\n\n"
        echo "Start the System Stat ....\n"

        time sqlplus / as sysdba @sys_stat.sql

        echo "System Stat is completed .....\n"

        echo "=================================================\n\n"
```

codadb1:oracle@/home/oracle>cat /home/oracle/script/statpack.ksh

```
        #!/usr/bin/ksh

        . $HOME/.profile

        sqlplus perfstat/perfstat@codaprod<<EOF
        exec statspack.snap(i_snap_level=>5);
        quit;
        EOF
```

codadb1:oracle@/home/oracle/script>cat /ora_fra/dataextract/aging/aging.ksh

```
        #!/usr/bin/ksh

        . $HOME/.profile

        DATE=`date +%Y%m%d`
        cd /ora_fra/dataextract/aging

        echo "*****  Aging Report is started @ `date`  *****\n"

        time sqlplus -S codauser/codaliv @aging.sql
        time sqlplus -S codauser/codaliv @unload.sql

        cat header >aging_$DATE.txt
        cat aging.temp>>aging_$DATE.txt

        echo "*****  Aging Report is completed @ `date`  *****\n"

        ftp -v -n 10.253.129.20<<!
            user FTPExtract GotoHyp_130
            bin
            cd aging
            put aging_$DATE.txt aging_$DATE.txt.tmp
            rename aging_$DATE.txt.tmp aging_$DATE.txt
            quit
        !

        echo "***** FTP Aging Report is completed @ `date`  *****\n"

        find ./ -name "*.txt" -mtime +7 -exec rm {} \;
```

codadb1:oracle@/home/oracle/sql>cat tbl_usage.sql

```
        CONN / as sysdba;

        select b.tablespace_name, tbs_size SizeMb, a.free_space FreeMb
```

```
        from
        (select tablespace_name, round(sum(bytes)/1024/1024 ,2) as free_space
        from dba_free_space group by tablespace_name) a,
        (select tablespace_name, sum(bytes)/1024/1024 as tbs_size
        from dba_data_files group by tablespace_name
        UNION
        select tablespace_name, sum(bytes)/1024/1024 tbs_size
        from dba_temp_files
        group by tablespace_name ) b
        where a.tablespace_name(+)=b.tablespace_name;

        exit;
```

codadb1:oracle@/home/oracle/sql>cat startdb.sql

```
        startup;
        exit;
```

codadb1:oracle@/home/oracle/sql>cat stopdb.sql

```
        shutdown immediate;
        exit;
```

codadb1:oracle@/home/oracle/sql>cat sys_stat.sql
```
        exec dbms_stats.gather_system_stats();

        exit;
```

codadb1:oracle@/home/oracle/sql>cat db_full_backup.rman
```
        set echo off;
        backup database;
        backup archivelog all delete all input;
        allocate channel for maintenance type disk;
        delete noprompt obsolete device type disk;
        release channel;
        list backup completed between 'SYSDATE-1' and 'SYSDATE';
        exit;
```

codadb1:oracle@/home/oracle/sql>cat db_inc_backup.rman
```
        set echo off;
        backup archivelog all delete all input;
        allocate channel for maintenance type disk;
        delete noprompt obsolete device type disk;
        release channel;
        list backup completed between 'SYSDATE-1' and 'SYSDATE';
        exit;
```

codadb1:oracle@/home/oracle/sql>cat clean_session.sql
```
        conn / as sysdba;

        TRUNCATE TABLE codauser.com_session_snap;
        quit;
```

```
codadb1:oracle@/home/oracle/sql>cat codarpt.rman
        connect target /
        backup database;
        backup archivelog all;
        list backup;
        exit;

codadb1:oracle@/home/oracle/sql>cat ctrl_backup.rman
        connect target /
        run {
        host 'rm /home/oracle/ctrl_bkup/BACKUP_control.ctl';
        sql "alter database backup controlfile to
        ''/home/oracle/ctrl_bkup/BACKUP_control.ctl''";
        }
        exit;

codadb1:oracle@/home/oracle/sql>cat db_status.sql
        COLUMN STATUS heading INSTANCE_STATUS
        COLUMN INSTANCE_NAME format A15
        COLUMN HOST_NAME format A10
        COLUMN STARTUP_TIME format A15
        COLUMN STATUS format A15
        COLUMN DATABASE_STATUS format A15

        select INSTANCE_NAME,HOST_NAME,STARTUP_TIME,STATUS,DATABASE_STATUS
        from v$instance;
        exit;

codadb1:oracle@/home/oracle/sql>cat reset_flag.sql
        update job_status set flag='N' where job_name='NIGHTLY';
        quit;

codadb1:oracle@/home/oracle/sql>cat verify_flag.sql
        select * from job_status where job_name='NIGHTLY';
        quit;

root@codadb1:/>#  cat /usr/local/scripts/codadb1_rsync_dr
        #!/usr/bin/ksh93

        #    Script written by Dave Benvie CGI
        #    This script rsync's file systems/directories identified by
        #    Livingston Internation to be kept in syncronization with their current
        #    DR partitions.
        #    codaapp1

        RSYNC_LIST='/coda'

        PDC_IP='10.153.255.212'

        LOG=/usr/local/logs/rsync/codadb1_rsync_log`date +%Y"-"%m"-"%d`

        for i in $RSYNC_LIST
        do
           echo "Starting rsync for file system $i `date`" >> $LOG
           echo "Statistics of syncronization below" >> $LOG
```

```
        echo >> $LOG
        rsync -acvps --del $i/ root@$PDC_IP:$i >> $LOG
        RESULT=${?}
        echo "${RESULT}" >/tmp/rsyncStatus.$(uname -n)
        echo >> $LOG
        echo "Finished syncronizarions for file system $i." >> $LOG
        echo "#####################################################" >> $LOG
        echo  >> $LOG
    done
```

codadb1:""@/home/""/>more /home/""/perl/mon_alert.pl

```perl
    #!/usr/bin/perl

    #Verify Active Node?
    if ( !`lsvg -o|grep fravg` )
    {
      print "Not an active node ...!\n";
      exit 1;
    }

    #Check to see how to use localtime function;
    $Source =
    "/u01/app/oracle/diag/rdbms/codaprod/codaprod/trace/alert_codaprod.log";
    $TodayLog = "/home/""/temp/today_alert.log";
    $ErrLog = "/home/""/temp/ora_error.log";

    #Get time stamp;
    ($sec,$min,$hour,$mday,$mon,$year,$wday,$yday,$isdst) = localtime(time);

    %weekday=(
            0=>'Sun',
            1=>'Mon',
            2=>'Tue',
            3=>'Wed',
            4=>'Thu',
            5=>'Fri',
            6=>'Sat',
            );
    %month=(
            0=>'Jan',
            1=>'Feb',
            2=>'Mar',
            3=>'Apr',
            4=>'May',
            5=>'Jun',
            6=>'Jul',
            7=>'Aug',
            8=>'Sep',
            9=>'Oct',
           10=>'Nov',
           11=>'Dec',
            );
    $year += 1900;
    print "Year of : $year \n";
    print "Month of the year : $month{$mon} $mon     \n";
```

```perl
print "Day of the Week : $weekday{$wday}\n";

if ( $mday < 10 )
 {
    $mday=sprintf ("0%d",$mday);
#    print "Today is : $mday \n";
 }

$time=sprintf("%3s %3s %2s",$weekday{$wday},$month{$mon},$mday);
#print "Time Stamp is : $time \n";

#Main ;
open (OUTFILE,">$TodayLog");
open (LOGFILE,"$Source")||die " Open Files Failed ... ";

#Set today flag;
$flag=0;

#Begin Processing the online log file;
while (<LOGFILE>)
  {
    if ( /$time/ )
        {
         $flag = 1;
          }
    if ( $flag == 1 )
        {
         print OUTFILE $_;
         };
      }
close LOGFILE;
close OUTFILE;

open (OUTFILE,">$ErrLog");
open (LOGFILE,"$TodayLog")||die " Open Files Failed ... ";

while(<LOGFILE>)
  {
    if ( /ORA-/ or /[e|E]rror/ )
        {
          print OUTFILE $temp;
          print OUTFILE $_;
          while(<LOGFILE>)
            {
             if ( /$time/ )
               { $temp = $_;
                 last;
                 }
             else
               { print OUTFILE $_;}
             }
        }
    else
      { $temp = $_;}
    }
```

```
        close LOGFILE;
        close OUTFILE;

        if ( -s $ErrLog )
          {
             `mail -s "Oracle Alarm on CODAPROD $year-$month{$mon}-$mday !!!" ""\@"".com
        < $ErrLog`
          }
```

codadb1:""`@/home/""`/>cat /home/""`/script/mon_dbSpace.ksh

```
        #!/usr/bin/ksh

        for tbl in CODATBL CODAIDX
          do
             size=`grep $tbl /home/oracle/report/tbl_usage.out|awk '{print $3}'`
             if [[ $size -lt 30720 ]]
               then
                    mail -s "!!! DB ALERT : Free Space on $tbl tablesoace is less than
        30G !!!" ""</home/oracle/report/tbl_usage.out
             fi
        done

        size=`grep SYSTEM /home/oracle/report/tbl_usage.out|awk '{print $3}'`
        if [[ $size -lt 512 ]]
          then
             mail -s "!!! DB ALERT : Free Space on SYSTEM tablesoace is less than 512M
        !!!" ""</home/oracle/report/tbl_usage.out
             fi
```

codadb1:oracle@/home/oracle/script>act /hyperion/extract_spe.ksh

```
        #!/usr/bin/ksh

        . $HOME/.profile

        set -A MailList "" \
                        jbaptist@"".com \
                        bghuman@"".com \
                        druddock@"".com \
                        jstewart@"".com \
                        asunga@"".com \
                        pwoods@"".com \
                        rsullivan@"".com

        cd /ora_fra/dataextract/hyperion

        YEAR=`date +%Y`
        MON=`date +%m`
        PERIOD=`date +%m|sed 's/0*//'`
        OUTPUT=$YEAR$MON.txt

        rm -rf *.dat

        #Non-spe
        cd /ora_fra/dataextract/hyperion/non-spe
        sqlplus -S codauser/codaliv@codaprod @part1.mod.sql $YEAR $PERIOD
```

```
sqlplus -S codauser/codaliv@codaprod @part2.mod.sql $YEAR $PERIOD
sqlplus -S codauser/codaliv@codaprod @part3.mod.sql $YEAR $PERIOD
sqlplus -S codauser/codaliv@codaprod @part4.mod.sql $YEAR $PERIOD

#Spe
cd /ora_fra/dataextract/hyperion/spe
sqlplus -S codauser/codaliv@codaprod @part1.mod.sql $YEAR $PERIOD
sqlplus -S codauser/codaliv@codaprod @part2.mod.sql $YEAR $PERIOD
sqlplus -S codauser/codaliv@codaprod @part3.mod.sql $YEAR $PERIOD
sqlplus -S codauser/codaliv@codaprod @part4.mod.sql $YEAR $PERIOD

#Final
cd /ora_fra/dataextract/hyperion
echo $YEAR >$OUTPUT
echo $PERIOD >>$OUTPUT
echo $PERIOD >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part1.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part2.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part3.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part4.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/spe/part1.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/spe/part2.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/spe/part3.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/spe/part4.out >>$OUTPUT

echo "*****  Data extraction is completed @ `date`  *****\n"

#ftp the output file;
ftp -v -n 10.253.129.20<<!
    user FTPExtract GotoHyp_130
    bin
    put $OUTPUT $OUTPUT.temp
    rename $OUTPUT.temp $OUTPUT
    quit
!

rm -rf /ora_fra/dataextract/hyperion/spe/part*.out
rm -rf /ora_fra/dataextract/hyperion/non-spe/part*.out

echo "*****  Ftp process is completed @ `date`  *****\n"
echo "================================================\n"

mail -s "Daily Hyperion Extract is completed." ${MailList[*]} </dev/null
```

codarpt:

Job	Frequency	Location	email
NIM MKSYSB	Weekly	/usr/local/scripts/client.mksysb.ksh	""
CFG2HTML report	Weekly	/usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1	""
Rsync to DR	Daily	/usr/local/scripts/codadb1_rsync_dr	""

system error monitor	Daily	/coda/codarpt/monitor/alertDog.ksh	""
Reporting Database	Daily	/home/oracle/script/db_clean.ksh /home/oracle/script/restore.ksh	""
Finance Report	Monthly	/ora_fra/dataextract/pwc_monthly/extract_pwc_monthly.ksh	""
	Monthly	/ora_fra/dataextract/gst/extract_gst.ksh /ora_fra/dataextract/revenue/monthly/revenue_monthly.ksh	

root@codarpt:/var/spool/cron/crontabs># cat oracle
```
##Db space report;
##5 1 * * * /home/oracle/script/db_space_rpt.ksh
>>/home/oracle/report/db_space_rpt.log 2>&1

#Db daily backup;
##1 23 * * * /home/oracle/script/db_backup.ksh >/home/oracle/report/db_backup_rpt.log
2>&1

#Reporting Database;
1 1 * * * /home/oracle/script/db_clean.ksh >> /home/oracle/report/db_clean.out 2>&1
1 5 * * * /home/oracle/script/restore.ksh >/home/oracle/report/restore.log 2>&1

#
1 10 7 * * /ora_fra/dataextract/pwc_monthly/extract_pwc_monthly.ksh >
/ora_fra/dataextract/pwc_monthly/extract_pwc_monthly.out 2>&1
1 8 15 * * /ora_fra/dataextract/gst/extract_gst.ksh >
/ora_fra/dataextract/gst/extract_gst.out 2>&1
1 9 15 * * /ora_fra/dataextract/revenue/monthly/revenue_monthly.ksh > /dev/null 2>&1
```

codarpt:oracle@/home/oracle$cat daily.env
```
     PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.

     export PATH


     if [ -s "$MAIL" ]         # This is at Shell startup.  In normal
     then echo "$MAILMSG"      # operation, the Shell checks
     fi                        # periodically.

     set -o vi

     export PS1=`hostname`:'$USER'@'$PWD'\$

     alias rm='rm -i'

     ORACLE_BASE=/u01/app/oracle
```

```
        export ORACLE_BASE

        ORACLE_SID=codaprod
        export ORACLE_SID

        ORACLE_OWNER=oracle
        export ORACLE_OWNER

        ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1
        export ORACLE_HOME

        AIXTHREAD_SCOPE=S; export AIXTHREAD_SCOPE

        ORACLE_HOSTNAME=codarptapp
        export ORACLE_HOSTNAME

        LD_LIBRARY_PATH=$ORACLE_HOME/lib
        export LD_LIBRARY_PATH

        export ORACLE_UNQNAME=codarpt

        NLS_LANG=American_America.WE8MSWIN1252
        export NLS_LANG

        #PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/u01/local/
        bin:/u01/app/oracle/OraHome_2/bin:/u01/app/oracle/Ora
        Home_2/jdk/jre/bin:.

        PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/u01/local/b
        in:$ORACLE_HOME/bin:$ORACLE_HOME/jdk/jre/bin:.

        export PATH
```

codarpt:oracle@/home/oracle$cat monthly.env

```
        PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.

        export PATH


        if [ -s "$MAIL" ]           # This is at Shell startup.  In normal
        then echo "$MAILMSG"        # operation, the Shell checks
        fi                          # periodically.

        set -o vi

        export PS1=`hostname`:'$USER'@'$PWD'\$

        alias rm='rm -i'

        ORACLE_BASE=/u02/app/oracle
        export ORACLE_BASE

        ORACLE_SID=monthrpt
        export ORACLE_SID
```

```
        ORACLE_OWNER=oracle
        export ORACLE_OWNER

        ORACLE_HOME=/u02/app/oracle/product/11.2.0/dbhome_1
        export ORACLE_HOME

        AIXTHREAD_SCOPE=S; export AIXTHREAD_SCOPE

        ORACLE_HOSTNAME=codarptapp
        export ORACLE_HOSTNAME

        LD_LIBRARY_PATH=$ORACLE_HOME/lib
        export LD_LIBRARY_PATH

        export ORACLE_UNQNAME=codarpt

        NLS_LANG=American_America.WE8MSWIN1252
        export NLS_LANG

        PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/u02/local/b
        in:$ORACLE_HOME/bin:$ORACLE_HOME/jdk/jre/bin:.

        export PATH
```

root@codarpt:/var/spool/cron/crontabs># cat /home/oracle/script/db_clean.ksh

```
        #!/usr/bin/ksh

        #Clean up the environment;

        echo " ===== `date` =====\n"

        df -g |grep "/ora_fra"

        echo " --- Working on /ora_fra/CODAPROD/autobackup ...\n "
        cd /ora_fra/CODAPROD/autobackup
        rm -fe *

        echo " --- Working on /ora_fra/CODAPROD/dbbackup ...\n "
        cd /ora_fra/CODAPROD/dbbackup
        ls -l
        rm -fe *

        echo " --- Working on /ora_fra/oradata/codaprod ...\n "
        cd /ora_fra/oradata/codaprod/archivelog
        rm -fe *

        df -g |grep "/ora_fra"

        echo " ===== Clean done ... =====\n"
```

root@codarpt:/var/spool/cron/crontabs># cat /home/oracle/script/restore.ksh

```
        #!/usr/bin/ksh

        Rpt=$HOME/report/restore.out
```

```
echo " ===== `date` ====="

. $HOME/daily.env

export NLS_DATE_FORMAT="Mon DD YYYY HH24:MI:SS"

typeset count=0

while [[ $count -lt 36 ]]
do
 if [[ -a /ora_fra/CODAPROD/dbbackup/rcp.done ]]
   then
       echo "Found TOKEN file, going to run the restore ...\n"
       cd $HOME/sql
       rman @restore.rman > $Rpt
       break
 fi
 echo "`date`: No TOKEN file found , continue to wait ...\n"
 sleep 300
 count=$(($count+1))
 if [[ $count -eq 36 ]]
    then
       echo "Restore windows missed, exits program ...\n"
       mail -s "WARNING: Reporting Database is available but the data is not up
to date due to no backup copy to restore...@ `date`"
 codaadmin@"".com</dev/null
#       mail -s "There is no DB copy available to restore ... Restore skipped
... @ `date`" ""</dev/null
       exit 1
 fi
done

sqlplus / as sysdba<<EOF
alter database open resetlogs;
shutdown immediate;
EOF

rm -rf /ora_temp/*.dbf

sqlplus / as sysdba<<EOF
startup;
EOF

sqlplus / as sysdba @$HOME/sql/conn_test.sql|grep "OPEN"
if [[ $? -ne 0 ]]
 then
  mail -s " ** WARNING: Reporting DB is not availble, please check.. @`date`
**" ""</dev/null
  exit 1
fi

#set flag for unmatch job
#touch /ora_fra/dataextract/unmatch/restore.done
```

```
find /u01/app/oracle/diag/rdbms/codaprod/codaprod/trace  -name "*.trc" -mtime
+3 -exec rm -f {} \;
find /u01/app/oracle/diag/rdbms/codaprod/codaprod/trace  -name "*.trm" -mtime
+3 -exec rm -f {} \;
find /u01/app/oracle/admin/codaprod/adump -name "*.aud" -mtime +30 -exec rm -f
{} \;

#mail -s "Reporting Database is available now ... @ `date`" codaadmin@"".com
</dev/null
mail -s "CODARPT : Daily Restore Report @ `date` on `hostname`" "" <$Rpt
```

codarpt:oracle@/ora_fra/dataextract/hyperion$cat extract.ksh

```
#!/usr/bin/ksh

. $HOME/daily.env

cd /ora_fra/dataextract/hyperion

echo "==================================================\n"
echo "Date : `date`"
echo "Report Year : $1"
echo "Report Period : $2"
echo "Email Address : $3\n"
echo "==================================================\n"

rm -rf *.dat

if [[ $2 -lt 10 ]]
then
  OUTPUT=${1}0${2}.dat
else
  OUTPUT=${1}${2}.dat
fi

#Non-spe
cd /ora_fra/dataextract/hyperion/non-spe
sqlplus -S codauser/codaliv@codaprod @part1.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part2.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part3.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part4.mod.sql $1 $2

#Spe
cd /ora_fra/dataextract/hyperion/spe
sqlplus -S codauser/codaliv@codaprod @part1.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part2.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part3.mod.sql $1 $2
sqlplus -S codauser/codaliv@codaprod @part4.mod.sql $1 $2

#Final
cd /ora_fra/dataextract/hyperion
echo $1 >$OUTPUT
echo $2 >>$OUTPUT
echo $2 >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part1.out >>$OUTPUT
cat /ora_fra/dataextract/hyperion/non-spe/part2.out >>$OUTPUT
```

```
        cat /ora_fra/dataextract/hyperion/non-spe/part3.out >>$OUTPUT
        cat /ora_fra/dataextract/hyperion/non-spe/part4.out >>$OUTPUT
        cat /ora_fra/dataextract/hyperion/spe/part1.out >>$OUTPUT
        cat /ora_fra/dataextract/hyperion/spe/part2.out >>$OUTPUT
        cat /ora_fra/dataextract/hyperion/spe/part3.out >>$OUTPUT
        cat /ora_fra/dataextract/hyperion/spe/part4.out >>$OUTPUT

        echo "*****  Data extraction is completed @ `date`  *****\n"

        #ftp the output file;

        #ftp -v -n 10.253.129.20<<!
        #    user FTPExtract GotoHyp_130
        #    bin
        #    put $OUTPUT $OUTPUT.temp
        #    rename $OUTPUT.temp $OUTPUT
        #    quit
        #!
        #
        rm -rf /ora_fra/dataextract/hyperion/non-spe/part*.out
        rm -rf /ora_fra/dataextract/hyperion/spe/part*.out

        #echo "*****  Ftp process is completed @ `date`  *****\n"
        #echo "=================================================\n"

        mail -s "Hyperion data is ready for YEAR: $1, Period: $2." $3 </dev/null
```

codarpt:oracle@/home/oracle$cat /ora_fra/dataextract/pwc_monthly/extract_pwc_monthly.ksh

```
        #!/usr/bin/ksh

        . $HOME/daily.env

        DATE=`date +%Y%m`

        cd /ora_fra/dataextract/pwc_monthly

        echo "***** PWC Monthly Data extraction is started @ `date`  *****\n"

        DATE=`date +%Y%m`

        mkdir $DATE

        #cmplist="C325 U125 H125 U135 U165 H145 E135 C310 E120 U140 F425 U250 E250 U255
        U275 C165 C180 U260 E265 E260 U265 E255 U175 E175 G1
        30 U235 U190 C320 C335 E185 U130 E190 E310 E335 E345 P325 P140 P315 H135 P330
        P130 F340 P135 B310 E245 U245 U240 E340 E360 C360 U270
         C170 F335 E365 C155 E355 U155 C175 C100 F325 E195 E145 C135 U170 E270 E275
        E125 E375 C110 C115 C120 C130 C140 C185 F000 F100 F125 F
        135 F150 F315 F320 F350 F415 F435 P100 P110 P310 C145 C160 F330 G350 H100 U150
        U160 U230 E230  E240 E110 E115 E150 E155 E160 E215 E1
        00 E105 E130 E140 E170 E330 E350 E385 E450 E485 E475 E165 E180"

        #cmplist="C165"
```

```
#for cmp in $cmplist

sqlplus codauser/codaliv @get_cmp.sql

for cmp in `cat cmp.lst`
  do
    echo "\n doing job for company : $cmp"
#    time sqlplus -S codauser/codaliv @pwc_monthly.sql $cmp 11
    time sqlplus -S codauser/codaliv @pwc_monthly.sql $cmp
  done

mv *.txt $DATE
cd $DATE

echo "***** PWC Monthly Data extraction is completed on `hostname` @ `date`
*****\n"
mail -s "***** PWC Monthly Data extraction is completed on `hostname` @ `date`
*****\n" ""</dev/null

#
ftp -v -n 192.168.250.11<<!
    user AUDCODA CODA_1209
    bin
    cd datacoda
    mkdir $DATE
    prompt
    cd $DATE
    mput *.txt
    quit
!


cd /ora_fra/dataextract/pwc_monthly
rm -rf $DATE
```

codarpt:oracle@/ora_fra/dataextract/pwc_monthly$cat get_cmp.sql

```
set pagesize 0
set echo off;
set term off;
set feedback off;
set verify off;

spool cmp.lst
select unique cmpcode from oas_dochead order by cmpcode;
spool off;
quit;
```

codarpt:oracle@/ora_fra/dataextract/pwc_monthly$cat pwc_monthly.sql

```
set pagesize 0;
set trimout on;
set trimspool on;
set linesize 600;
set echo off;
set term off;
set feedback off;
```

```
set verify off;

spool pwc_&&1\.txt;

select
'COMPANY|compname|DOCCODE|DOCNUM|DOCLINENUM|USERNAME|AUTHUSER|ORIGUSER|INPDATE|
INPTIME|YEAR|PERIOD|DOCDATE|GL|BU_DIV_REG|BRCH
_TEAM_DEPT|LOC|VENDOR_CLIENT|CER|EXTREF1|EXTREF2|EXTREF3|EXTREF4|EXTREF5|EXTREF
6|DOCVALUE|DOCCURR|HOMEVALUE|HOMECUR|DUALVALUE|DUALCU
R|EL1_CURCODE|EL1_VALUE|EL2_curcode|EL2_value|EL3_curcode|EL3_value|EL4_curcode
|EL4_value|EL5_CURCODE|EL5_VALUE|EL6_CURCODE|EL6_VALU
E|DEB_CRED_IND|MATCHREF|STATPAYINT|PARTPAYMENT|MODDATE' from dual;

SELECT /*+OPT_PARAM('_optimizer_sortmerge_join_enabled','false')*/
  DH.CMPCODE||'|'|| C.name||'|'|| DH.DOCCODE||'|'|| DH.DOCNUM||'|'||
DL.DOCLINENUM||'|'||
  DH.USRNAME||'|'||DH.AUTHUSER||'|'||DH.ORIGUSER||'|'||
  DH.INPDATE||'|'|| DH.INPTIME||'|'||
  DH.YR||'|'|| DH.PERIOD||'|'||
  DH.DOCDATE||'|'||
  DL.EL1||'|'||
  DL.EL2||'|'||
  DL.EL3||'|'||
  DL.EL4||'|'||
  DL.EL5||'|'||
  DL.EL6||'|'||
  DL.REF1||'|'||
  DL.REF2||'|'||
  DL.REF3||'|'||
  DL.REF4||'|'||
  DL.REF5||'|'||
  DL.REF6||'|'||
  DL.VALUEDOC||'|'||
  DH.CURDOC||'|'||
  DL.VALUEHOME||'|'||
  C.HOMECUR||'|'||
  DL.VALUEDUAL||'|'||
  C.DUALCUR||'|'||
  CL.EL1_CURCODE||'|'|| EL1_VALUE||'|'||
  CL.EL2_curcode||'|'|| EL2_value||'|'||
  CL.EL3_curcode||'|'|| EL3_value||'|'||
  CL.EL4_curcode||'|'|| EL4_value||'|'||
  CL.EL5_CURCODE||'|'|| EL5_VALUE||'|'||
  CL.EL6_CURCODE||'|'|| EL6_VALUE||'|'||
  DL.DEB_CRED_IND||'|'||
  PL.MATCHREF||'|'||
  DL.STATPAYINT||'|'||
  PL.PARTPAYMENT||'|'||
  PL.MODDATE
 FROM OAS_DOCHEAD DH
  INNER JOIN OAS_DOCLINE DL
    ON DL.CMPCODE = DH.CMPCODE AND DL.DOCCODE = DH.DOCCODE AND DL.DOCNUM =
DH.DOCNUM
  LEFT OUTER JOIN OAS_CURLINE CL
```

```
      ON DL.cmpcode = CL.cmpcode and DL.doccode = CL.doccode and DL.docnum =
CL.docnum and DL.doclinenum = CL.doclinenum
   LEFT OUTER JOIN OAS_PAYLINE PL
      ON PL.CMPCODE = DL.CMPCODE AND PL.DOCCODE = DL.DOCCODE AND PL.DOCNUM =
DL.DOCNUM AND PL.DOCLINENUM = DL.DOCLINENUM
    INNER JOIN OAS_COMPANY C
      ON C.CODE = DH.CMPCODE
   WHERE DH.CMPCODE = '&&1'
     AND DH.STATUS = 78
     AND DH.YR = TO_NUMBER(TO_CHAR(SYSDATE - 10,'YYYY'))
     AND DH.PERIOD = TO_NUMBER(TO_CHAR(SYSDATE - 10,'MM'))
     AND to_char(PL.MODDATE,'YYYY') = TO_CHAR(SYSDATE - 10,'YYYY')
     AND to_char(PL.MODDATE,'MM') = TO_CHAR(SYSDATE - 10,'MM');


     spool off;

     quit;
```

codarpt:oracle@/home/oracle$cat /ora_fra/dataextract/gst/extract_gst.ksh

```
     #!/usr/bin/ksh

     . $HOME/daily.env
     #. $HOME/monthly.env

     cd /ora_fra/dataextract/gst

     echo "*****  GST Data extraction is started @ `date`  *****\n"

     typeset -Z2 period
     month=`date +%m`
     year=`date +%Y`

     period=$((month-1))

     if (( month == 1));then
     period=12
     year=$((year-1))
     fi

     echo $year $period

     time sqlplus -S codauser/codaliv @gst.sql $year $period

     echo "*****  GST Data extraction is completed @ `date`  *****\n"

     ftp -v -n 10.253.129.20<<EOF
         user FTPExtract GotoHyp_130
         cd gst
         put GST.out GST_${year}_${period}.txt.tmp
         rename GST_${year}_${period}.txt.tmp GST_${year}_${period}.txt
         quit
     EOF
```

```
        mail -s "*****  GST Data extraction is completed @ `date` on `hostname` [CGI]
        *****" ""</dev/null
```

codarpt:oracle@/home/oracle$cat
/ora_fra/dataextract/revenue/monthly/revenue_monthly.ksh
```
        #!/usr/bin/ksh

        cd /ora_fra/dataextract/revenue/monthly

        nohup ./revenue-C310.ksh>./revenue-C310.out 2>&1 &
        nohup ./revenue-C165.ksh>./revenue-C165.out 2>&1 &
```

codarpt:oracle@/ora_fra/dataextract/revenue/monthly$cat revenue-C165.ksh
```
        #!/usr/bin/ksh

        . $HOME/daily.env

        DATE=`date +%Y%m%d`

        typeset -Z2 period
        month=`date +%m`
        year=`date +%Y`

        period=$((month-1))

        if (( month == 1));then
        period=12
        year=$((year-1))
        fi

        echo $year $period

        cd /ora_fra/dataextract/revenue/monthly

        echo "*****  Data extraction is started @ `date`  *****\n"

        time sqlplus -S codauser/codaliv @revenue-C165.sql $year $period

        echo "*****  Data extraction is completed @ `date`  *****\n"

        #mv revenue_C165.lst revenue_C165-$year-$period.txt

        echo "\n ==== Start FTP process `date` ....\n"

        ftp -v -n 10.253.129.20<<EOF
            user FTPExtract GotoHyp_130
            cd gst
            put revenue_C165.lst revenue_C165-${year}-${period}.txt.tmp
            rename revenue_C165-${year}-${period}.txt.tmp revenue_C165-${year}-
        ${period}.txt
            quit
        EOF
```

codarpt:oracle@/ora_fra/dataextract/revenue/monthly$cat revenue-C310.ksh
```
        #!/usr/bin/ksh
```

```
        . $HOME/daily.env

        DATE=`date +%Y%m%d`

        typeset -Z2 period
        month=`date +%m`
        year=`date +%Y`

        period=$((month-1))

        if (( month == 1));then
        period=12
        year=$((year-1))
        fi

        echo $year $period

        cd /ora_fra/dataextract/revenue/monthly

        echo "*****  Data extraction is started @ `date`  *****\n"

        time sqlplus -S codauser/codaliv @revenue-C310.sql $year $period

        echo "*****  Data extraction is completed @ `date`  *****\n"

        #mv revenue_C310.lst revenue_C310-$year-$period.txt

        echo "\n ==== Start FTP process `date` ....\n"

        ftp -v -n 10.253.129.20<<EOF
            user FTPExtract GotoHyp_130
            cd gst
            put revenue_C310.lst revenue_C310-${year}-${period}.txt.tmp
            rename revenue_C310-${year}-${period}.txt.tmp revenue_C310-${year}-
        ${period}.txt
            quit
        EOF
```

codarpt:oracle@/ora_fra/dataextract/revenue/monthly$cat revenue-C165.sql

```
        set pagesize 0;
        set trimout on;
        set trimspool on;
        set linesize 600;
        set echo off;
        set term off;
        set feedback off;
        set verify off;

        spool revenue_C165.lst

        SELECT /*+OPT_PARAM('_optimizer_sortmerge_join_enabled','false')*/
                DL.cmpcode|| ','||
                DL.el1|| ','||
                DL.el2|| ','||
```

```
             DL.el3|| ','||
             DL.el4|| ','||
             DL.el5|| ','||
             E.name|| ','||
             DL.doccode|| ','||
             DL.docnum|| ','||
             DL.doclinenum|| ','||
             DH.docdate|| ','||
             DH.yr|| ','||
             DH.period|| ','||
             DH.curdoc|| ','||
             DL.valuehome|| ','||
             DL.valuedual|| ','||
             DL.ref3|| ','||
             DL.ref4
      FROM OAS_DOCHEAD DH
      INNER JOIN OAS_DOCLINE DL
         ON DL.CMPCODE = DH.CMPCODE
        AND DL.DOCCODE = DH.DOCCODE
        AND DL.DOCNUM = DH.DOCNUM
      LEFT OUTER JOIN OAS_ELEMENT E
        ON E.cmpcode = DL.cmpcode and E.elmlevel = 5 and E.code = DL.el5
      WHERE DL.CMPCODE = 'C165'
        AND DH.STATUS = 78
        AND DH.YR = &&1
        AND DH.PERIOD = &&2
        AND DL.EL1 LIKE '4%'
        AND DL.EL4 LIKE '1011%';

      spool off;

      quit;
```

codarpt:oracle@/ora_fra/dataextract/revenue/monthly$cat revenue-C310.sql

```
      set pagesize 0;
      set trimout on;
      set trimspool on;
      set linesize 600;
      set echo off;
      set term off;
      set feedback off;
      set verify off;

      spool revenue_C310.lst

      SELECT /*+OPT_PARAM('_optimizer_sortmerge_join_enabled','false')*/
             DL.cmpcode|| ','||
             DL.el1|| ','||
             DL.el2|| ','||
             DL.el3|| ','||
             DL.el4|| ','||
             DL.el5|| ','||
             E.name|| ','||
             DL.doccode|| ','||
             DL.docnum|| ','||
```

```
                DL.doclinenum|| ','||
                DH.docdate|| ','||
                DH.yr|| ','||
                DH.period|| ','||
                DH.curdoc|| ','||
                DL.valuehome|| ','||
                DL.valuedual|| ','||
                DL.ref3|| ','||
                DL.ref4
        FROM OAS_DOCHEAD DH
        INNER JOIN OAS_DOCLINE DL
           ON DL.CMPCODE = DH.CMPCODE
          AND DL.DOCCODE = DH.DOCCODE
          AND DL.DOCNUM = DH.DOCNUM
        LEFT OUTER JOIN OAS_ELEMENT E
          ON E.cmpcode = DL.cmpcode and E.elmlevel = 5 and E.code = DL.el5
        WHERE DL.CMPCODE = 'C310'
          AND DH.STATUS = 78
          AND DH.YR = &&1
          AND DH.PERIOD = &&2
          AND DL.EL1 LIKE '4%'
          AND DL.EL4 LIKE '1011%';

        spool off;

        quit;

codarpt:oracle@/ora_fra/dataextract/hyperion/non-spe$cat part1.mod.sql
        set pagesize 0;
        set echo off;
        set term off;
        set feedback off;
        set verify off;

        spool part1.out;
        select a.el3 || a.cmpcode || a.ccode || ',' || a.el1 || ',' ||
        to_char(a.full_value)
        from (select el3, cmpcode,
              CASE
              when curcode = 'CAD' then 'C'
              when curcode = 'USD' then 'U'
              when curcode = 'MXN' then 'M'
              when curcode = 'GBP' then 'P'
              when curcode = 'EUR' then 'E'
              when curcode = 'HKD' then 'H'
              when curcode = 'PLN' then 'Z'
              when curcode = 'CNY' then 'Y'
              when curcode = 'VIN' then 'R'
              ELSE substr(curcode,1,1)
              END as ccode,
              el1,
              CASE
              when substr(el1,1,1) in ('2','3','4')
                  then sum(full_value) * -1
              ELSE sum(full_value)
```

```
            END as full_value
            from CODAUSER.OAS_BALANCE
            where (cmpcode not like 'C%' and cmpcode not like 'U%'
                    and cmpcode != 'VCA' and cmpcode != 'VMX' and cmpcode != 'VUP' and
cmpcode != 'VLUX' )
                and repbasis = 0
                and yr = &1
                and (period >= 0 and period <= &2 )
                and balcode = 'ACTUAL'
                and (el1 like '1%'
                    or el1 like '2%'
                    or el1 like '3%')
            group by el3,cmpcode,
                CASE
                when curcode = 'CAD' then 'C'
                when curcode = 'USD' then 'U'
                when curcode = 'MXN' then 'M'
                when curcode = 'GBP' then 'P'
                when curcode = 'EUR' then 'E'
                when curcode = 'HKD' then 'H'
                when curcode = 'PLN' then 'Z'
                when curcode = 'CNY' then 'Y'
                when curcode = 'VIN' then 'R'
                ELSE substr(curcode,1,1)
                END,
                el1) a
        /

        spool off;
        quit;
```

codarpt:oracle@/ora_fra/dataextract/hyperion/non-spe$cat part2.mod.sql

```
        set pagesize 0;
        set echo off;
        set term off;
        set verfiy off;
        set feedback off;

        spool part2.out;

        select a.el3 || a.cmpcode || a.ccode ||','|| a.el1 || ',' ||
        to_char(a.full_value)
        from ( select el3, cmpcode,
                CASE
                when curcode = 'CAD' then 'C'
                when curcode = 'USD' then 'U'
                when curcode = 'MXN' then 'M'
                when curcode = 'GBP' then 'P'
                when curcode = 'EUR' then 'E'
                when curcode = 'HKD' then 'H'
                when curcode = 'PLN' then 'Z'
                when curcode = 'CNY' then 'Y'
                when curcode = 'VIN' then 'R'
                ELSE substr(curcode,1,1)
                END as ccode,
```

```
            el1,
            CASE
            when substr(el1,1,1) in ('2','3','4') then sum(full_value) * -1
            ELSE sum(full_value)
            END as full_value
            from CODAUSER.OAS_BALANCE
            where (cmpcode not like 'C%' and cmpcode not like 'U%'
                    and cmpcode != 'VCA' and cmpcode != 'VMX' and cmpcode != 'VUP'
    and cmpcode != 'VLUX' )
                and repbasis = 0
                and yr = &1
                and (period >= 1 and period <= &2)
                and balcode = 'ACTUAL'
                and (el1 like '4%'
                    or el1 like '5%'
                    or el1 like '6%')
            group by el3,cmpcode,
              CASE
              when curcode = 'CAD' then 'C'
              when curcode = 'USD' then 'U'
              when curcode = 'MXN' then 'M'
              when curcode = 'GBP' then 'P'
              when curcode = 'EUR' then 'E'
              when curcode = 'HKD' then 'H'
              when curcode = 'PLN' then 'Z'
              when curcode = 'CNY' then 'Y'
              when curcode = 'VIN' then 'R'
              ELSE substr(curcode,1,1)
              END,
              el1
            ) a
    /

    spool off;
    quit;
```

codarpt:oracle@/ora_fra/dataextract/hyperion/non-spe$cat part3.mod.sql

```
    set pagesize 0;
    set echo off;
    set feedback off;
    set term off;
    set verify off;

    spool part3.out;

    select a.el3 || a.cmpcode || a.ccode ||','|| '967010' || ',' ||
    to_char(a.full_value)
    from ( select el3, cmpcode,
            CASE
            when curcode = 'CAD' then 'C'
            when curcode = 'USD' then 'U'
            when curcode = 'MXN' then 'M'
            when curcode = 'GBP' then 'P'
            when curcode = 'EUR' then 'E'
            when curcode = 'HKD' then 'H'
```

```
                    when curcode = 'PLN' then 'Z'
                    when curcode = 'CNY' then 'Y'
                    when curcode = 'VIN' then 'R'
                    ELSE substr(curcode,1,1)
                    END as ccode,
                    sum(full_value) as full_value
                    from CODAUSER.OAS_BALANCE
                    where (cmpcode not like 'C%' and cmpcode not like 'U%'
                            and cmpcode != 'VCA' and cmpcode != 'VMX' and cmpcode != 'VUP'
        and cmpcode != 'VLUX' )
                        and repbasis = 0
                        and yr = &1
                        and (period >= 1 and period <= &2)
                        and balcode = 'TRANSACTIONS'
                        and el1 like '967%'
                    group by el3,cmpcode,
                        CASE
                        when curcode = 'CAD' then 'C'
                        when curcode = 'USD' then 'U'
                        when curcode = 'MXN' then 'M'
                        when curcode = 'GBP' then 'P'
                        when curcode = 'EUR' then 'E'
                        when curcode = 'HKD' then 'H'
                        when curcode = 'PLN' then 'Z'
                        when curcode = 'CNY' then 'Y'
                        when curcode = 'VIN' then 'R'
                        ELSE substr(curcode,1,1)
                        END
                    ) a
        /

        spool off;
        quit;

codarpt:oracle@/ora_fra/dataextract/hyperion/non-spe$cat part4.mod.sql
        set pagesize 0;
        set echo off;
        set feedback off;
        set term off;
        set verify off;

        spool part4.out;

        select a.el3 || a.cmpcode || a.ccode ||','|| '901400' || ',' ||
        to_char(a.full_value)
        from ( select el3, cmpcode,
                    CASE
                    when curcode = 'CAD' then 'C'
                    when curcode = 'USD' then 'U'
                    when curcode = 'MXN' then 'M'
                    when curcode = 'GBP' then 'P'
                    when curcode = 'EUR' then 'E'
                    when curcode = 'HKD' then 'H'
                    when curcode = 'PLN' then 'Z'
                    when curcode = 'CNY' then 'Y'
```

```
           when curcode = 'VIN' then 'R'
           ELSE substr(curcode,1,1)
           END as ccode,
           sum(full_value) as full_value
           from CODAUSER.OAS_BALANCE
           where (cmpcode not like 'C%' and cmpcode not like 'U%'
                   and cmpcode != 'VCA' and cmpcode != 'VMX' and cmpcode != 'VUP'
    and cmpcode != 'VLUX' )
               and repbasis = 0
               and yr = &1
               and (period >= 1 and period <= &2)
               and balcode = 'HEADCOUNT'
               and el1 like '901%'
           group by el3,cmpcode,
             CASE
             when curcode = 'CAD' then 'C'
             when curcode = 'USD' then 'U'
             when curcode = 'MXN' then 'M'
             when curcode = 'GBP' then 'P'
             when curcode = 'EUR' then 'E'
             when curcode = 'HKD' then 'H'
             when curcode = 'PLN' then 'Z'
             when curcode = 'CNY' then 'Y'
             when curcode = 'VIN' then 'R'
             ELSE substr(curcode,1,1)
             END
           ) a
    /

    spool off;
    quit;


codarpt:oracle@/ora_fra/dataextract/monthend$cat monthend-1.ksh
    #!/usr/bin/ksh

    #. $HOME/daily.env
    . $HOME/monthly.env

    echo "==== Monthend job started @ `date`"

    cd /ora_fra/dataextract/monthend

    CODES="C165-DFA-UNBILLCD C165-DFA-UNBILLUS C165-FR-K84-DUTY C165-FR-UNPAID
    C310-DFA-UNBILLCD C310-DFA-UNBILLUS C310-FR-K84-DUTY C310
    -FR-UNPAID U125-DFA-UNPAID-DUTY U125-DFA-UNPAID-FREIGHT U125-DFA-UNBILLED"

    #CODES="U125-DFA-UNPAID-FREIGHT"

    for code in $CODES
    do
    echo "\n === Running job for : $code ===="
    time sqlplus codauser/codaliv @$code.sql
    done

    #echo " === Start the ftp process on `hostname` @ `date` ..."
```

```
        #ftp -v -n 10.253.129.20<<!
        #    user FTPExtract GotoHyp_130
        #    bin
        #    prompt
        #    cd cathy
        #    mput *.txt
        #    quit
        #!
        #echo " === Ftp process is completed on `hostname` @ `date` ..."

        echo "==== Monthend job completed @ `date`"

        mail -s "Monthend Job is now completed @ `date`, please stop CRON job !!!"
        ""</dev/null
```

codarpt:oracle@/ora_fra/dataextract/monthend$cat monthend.ksh

```
        #!/usr/bin/ksh

        . $HOME/daily.env
        #. $HOME/monthly.env

        echo "==== Monthend job started @ `date`"

        cd /ora_fra/dataextract/monthend

        CODES="C165-DFA-UNBILLCD C165-DFA-UNBILLUS C165-FR-K84-DUTY C165-FR-UNPAID
        C310-DFA-UNBILLCD C310-DFA-UNBILLUS C310-FR-K84-DUTY C310
        -FR-UNPAID U125-DFA-UNPAID-DUTY U125-DFA-UNPAID-FREIGHT U125-DFA-UNBILLED C310-
        DFA-212659 C165-DFA-212659"

        for code in $CODES
        do
        echo "\n === Running job for : $code ===="
        time sqlplus codauser/codaliv @$code.sql
        done

        echo " === Start the ftp process on `hostname` @ `date` ..."
        ftp -v -n 10.253.129.20<<!
            user FTPExtract GotoHyp_130
            bin
            prompt
            cd cathy
            mput *.txt
            quit
        !
        echo " === Ftp process is completed on `hostname` @ `date` ..."

        echo "==== Monthend job completed @ `date`"

        mail -s "Monthend Job is now completed @ `date` !!!" ""</dev/null
        mail -s "Monthend Data is now ready @ `date`..." cgriffi@"".com</dev/null
        mail -s "Monthend Data is now ready @ `date`..." sreese@"".com</dev/null
```

codarpt:oracle@/ora_fra/dataextract/monthend$cat U125-DFA-UNPAID-FREIGHT.sql

```
        set pagesize 0;
```

```
set trimout on;
set trimspool on;
set linesize 600;
set echo off;
set term off;
set feedback off;

spool U125-DFA-UNPAID-FREIGHT.txt;

SELECT
'CMPCODE|EL5|NAME|CURDOC|REF3|VALUEDOC|DOCDATE|REF2|REF1|EL6|REF5|REF4|REF6|EL1
|EL2|EL3|SNAME|EL4|DOCCODE|DOCNUM|STATPAY|COMM
ENTS' from dual;

SELECT OAS_DOCHEAD.CMPCODE||'|'||
       OAS_DOCLINE.EL5||'|'||
       OAS_ELEMENT.NAME||'|'||
       OAS_DOCHEAD.CURDOC||'|'||
       OAS_DOCLINE.REF3||'|'||
       OAS_DOCLINE.VALUEDOC||'|'||
       OAS_DOCHEAD.DOCDATE||'|'||
       OAS_DOCLINE.REF2||'|'||
       OAS_DOCLINE.REF1||'|'||
       OAS_DOCLINE.EL6||'|'||
       OAS_DOCLINE.REF5||'|'||
       OAS_DOCLINE.REF4||'|'||
       OAS_DOCLINE.REF6||'|'||
       OAS_DOCLINE.EL1||'|'||
       OAS_DOCLINE.EL2||'|'||
       OAS_DOCLINE.EL3||'|'||
       OAS_ELEMENT_1.SNAME||'|'||
       OAS_DOCLINE.EL4||'|'||
       OAS_DOCHEAD.DOCCODE||'|'||
       OAS_DOCHEAD.DOCNUM||'|'||
       OAS_DOCLINE.STATPAY||'|'||
       OAS_DOCCLIST.COMM
FROM OAS_DOCLINE
LEFT JOIN OAS_DOCCLIST
  ON OAS_DOCLINE.CMPCODE = OAS_DOCCLIST.CMPCODE
 AND OAS_DOCLINE.DOCCODE = OAS_DOCCLIST.DOCCODE
 AND OAS_DOCLINE.DOCNUM = OAS_DOCCLIST.DOCNUM
 AND OAS_DOCLINE.DOCLINENUM = OAS_DOCCLIST.DOCLINENUM
LEFT JOIN OAS_ELEMENT
  ON OAS_DOCLINE.EL5 = OAS_ELEMENT.CODE
 AND OAS_DOCLINE.CMPCODE = OAS_ELEMENT.CMPCODE
LEFT JOIN OAS_DOCHEAD
  ON OAS_DOCLINE.DOCNUM = OAS_DOCHEAD.DOCNUM
 AND OAS_DOCLINE.DOCCODE = OAS_DOCHEAD.DOCCODE
 AND OAS_DOCLINE.CMPCODE = OAS_DOCHEAD.CMPCODE
LEFT JOIN OAS_ELEMENT OAS_ELEMENT_1
  ON OAS_DOCLINE.EL3 = OAS_ELEMENT_1.CODE
 AND OAS_DOCLINE.CMPCODE = OAS_ELEMENT_1.CMPCODE
WHERE OAS_DOCHEAD.CMPCODE = 'U125'
  AND (OAS_DOCLINE.EL1 = '212615'
        OR OAS_DOCLINE.EL1 = '212616'
```

```
          OR OAS_DOCLINE.EL1 = '212620'
          OR OAS_DOCLINE.EL1 = '212621'
          OR OAS_DOCLINE.EL1 = '212634'
          OR OAS_DOCLINE.EL1 = '212635'
          OR OAS_DOCLINE.EL1 = '212636'
          OR OAS_DOCLINE.EL1 = '212640'
          OR OAS_DOCLINE.EL1 = '212660')
     AND (OAS_DOCHEAD.DOCCODE = 'MIGRATEAPOPU'
          OR OAS_DOCHEAD.DOCCODE = 'SALE-PAY-USD'
          OR OAS_DOCHEAD.DOCCODE = 'JRNLFRT'
          OR OAS_DOCHEAD.DOCCODE = 'JRNLFRTAP'
          OR OAS_DOCHEAD.DOCCODE = 'APACQUIRED'
          OR OAS_DOCHEAD.DOCCODE = 'PAY-MAN-USF'
          OR OAS_DOCHEAD.DOCCODE = 'PAY-MAN-USC')
     AND (OAS_DOCLINE.STATPAY = 84
          OR OAS_DOCLINE.STATPAY = 93)
     AND OAS_ELEMENT.ELMLEVEL = 5
     AND OAS_ELEMENT_1.ELMLEVEL = 3
     AND OAS_DOCHEAD.STATUS = 78;

     spool off;

     quit;
```

codatest:

Job	Frequency	Location	email
NIM MKSYSB	Weekly	/usr/local/scripts/client.mksysb.ksh	""
CFG2HTML report	Weekly	/usr/local/scripts/cfg2html -1 -o /tmp -e 1 > /tmp/cfg2html_aix_log.txt 2>&1	""
Rsync to DR	Daily	/usr/local/scripts/codadb1_rsync_dr	""
system error monitor	Daily	/coda/codarpt/monitor/alertDog.ksh	""
Db monitor		/coda/codadb1/monitor/monProcess.ksh	
Db space report	Daily	/home/oracle/script/db_space_rpt.ksh	
Db backup	daily	/home/oracle/script/db_backup.ksh	
DB stat	Daily	/home/oracle/script/db_stat.ksh	

root@codatest:/var/spool/cron/crontabs># cat oracle
```
#Db space report;
5 6 * * * /home/oracle/script/db_space_rpt.ksh >>/home/oracle/report/db_space_rpt.log
2>&1

#Db daily backup;
1 23 * * * /home/oracle/script/db_backup.ksh >/home/oracle/report/db_backup_rpt.log
2>&1
```

```
#Db monitor;
#10 * * * * /coda/codadb1/monitor/monProcess.ksh
>/admin/codadb1/monitor/monProcess.out 2>&1

#DB stat;
1 5 * * * /home/oracle/script/db_stat.ksh >> /home/oracle/report/db_stat.out 2>&1
```

root@codatest:/$ cat starttestdb.ksh
```
        #!/usr/bin/ksh

        #Created by ""

        export JAVA_HOME=/usr/java6_64/jre

        PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr/vac/bin
        :$JAVA_HOME/bin:.


        LogFile=/coda/codatest/bin/starttestdb.log

        date > $LogFile
        su - oracle >> $LogFile 2>&1 <<EOF
           echo "\n===  Start Oracle DB Server ...."
           sqlplus / as sysdba @/home/oracle/sql/startdb.sql
           if [ $? -ne 0 ]; then
              echo "problem to startup Oracle DB ..."
              exit 1
           fi
           sleep 5

           echo "\n===  Start Oracle Listener ...."
           lsnrctl start
           if [ $? -ne 0 ]; then
              echo "Problem to Start Oracle Listener ..."
              exit 1
           fi
           sleep 5

           echo "\n===  Start EM Console ...."
           emctl start dbconsole
           if [ $? -ne 0 ]; then
              echo "Problem to Start DB Console ..."
              exit 1
           fi
        EOF

        echo "\n===   Oracle Environment is Ready   ===\n"|tee -a $LogFile
```

root@codatest:/$ cat starttestweb.ksh
```
        #!/usr/bin/ksh

        Log=/coda/codatest/bin/starttestweb.out
```

```
      date >> $Log

      #Start apache HTTP Server;

      cd /usr/local/apache2/bin
      ./apachectl start
      if [ $? -eq 1 ]
       then
         echo "Problem to start HTTP Server ...quit\n" >>$Log
         exit 1
       else
         echo "HTTP Server has been started\n" >>$Log
      fi

      #Start apache TomCat Server;
      cd /usr/apache-tomcat-6.0.28/bin
      ./startup.sh
      if [ $? -eq 1 ]
       then
         echo "Problem to start TomCat ...quit\n" >> $Log
         exit 1
       else
         echo "TomCat has been started\n" >>$Log
      fi

      #Start CODA web application;
      cd /usr/coda/instances/CODADEV/DSS
      nohup dssstrlsv >>$Log 2>&1 &
      if [ $? -eq 1 ]
       then
         echo "Problem to start CODA web application ...\n" >>$Log
         echo "quit.....\n"
         exit 1
       else
         echo "Coda web application has been started\n" >>$Log
      fi
$


root@codatest# cat /home/oracle/script/db_backup.ksh
      #!/usr/bin/ksh

      Rpt=$HOME/report/backup_rpt.out

      . $HOME/.profile

      cd $HOME/sql

      DB_status=`sqlplus / as sysdba @db_status.sql|grep 'ACTIVE'`

      if [[ $? -ne 0 ]]
      then
         echo "Oracel DB is not UP, backup will quit ...."
         mail -s "CODATEST ALARM : DB backup failed due to DB is not up @ `date`" ""
      </dev/null
```

```
        exit 1
        fi

        cd /ora_fra/CODADEV/dbbackup
        rm -fe /ora_fra/CODADEV/dbbackup/*

        cd $HOME/sql

        rman TARGET / < db_backup.rman > $Rpt

        #Also clean log;
        find /u01/app/oracle/diag/rdbms/codadev/codadev/trace -name "*.trc" -mtime +3 -
        exec rm -f {} \;
        find /u01/app/oracle/diag/rdbms/codadev/codadev/trace -name "*.trm" -mtime +3 -
        exec rm -f {} \;
        find /u01/app/oracle/admin/codadev/adump -name "*.aud" -mtime +30 -exec rm -f
        {} \;

        find /u01/app/oracle/diag/tnslsnr/codatest/listener/alert -name "*.xml" -mtime
        +3 -exec rm -f {} \;

        mail -s "CODATEST : Oracle Daily Backup Report @ `date`" "" <$Rpt
```

codatest:oracle@/home/oracle/sql $cat ctrl_backup.rman
```
        connect target /
        run{
        host 'rm /home/oracle/ctrl_bkup/BACKUP_control.ctl';
        sql "alter database backup controlfile to
        ''/home/oracle/ctrl_bkup/BACKUP_control.ctl''";
        }
        exit;
```

codatest:oracle@/home/oracle/sql $cat db_backup.rman
```
        set echo off;
        backup database;
        backup archivelog all not backed up delete all input;
        allocate channel for maintenance type disk;
        delete noprompt obsolete device type disk;
        release channel;
        list backup completed between 'SYSDATE-1' and 'SYSDATE';
        exit;
```

$ cat startdb.sql
```
        startup;
        exit;
```
$ cat stopdb.sql
```
        shutdown immediate;
        exit;
```

codatest:oracle@/home/oracle$emctl start dbconsole
```
        Oracle Enterprise Manager 11g Database Control Release 11.2.0.2.0
        Copyright (c) 1996, 2010 Oracle Corporation.  All rights reserved.
```

```
https://codadevdb:1158/em/console/aboutApplication
Starting Oracle Enterprise Manager 11g Database Control ..... started.
---------------------------------------------------------------
Logs are generated in directory
/u01/app/oracle/product/11.2.0/dbhome_1/codadevdb_codadev/sysman/log
```

codatest:oracle@/home/oracle$emctl status agent

```
Oracle Enterprise Manager 11g Database Control Release 11.2.0.2.0
Copyright (c) 1996, 2010 Oracle Corporation.  All rights reserved.
---------------------------------------------------------------
Agent Version      : 10.2.0.4.3
OMS Version        : 10.2.0.4.3
Protocol Version   : 10.2.0.4.3
Agent Home         : /u01/app/oracle/product/11.2.0/dbhome_1/codadevdb_codadev
Agent binaries     : /u01/app/oracle/product/11.2.0/dbhome_1
Agent Process ID   : 18022504
Parent Process ID  : 2425158
Agent URL          : https://codadevdb:3938/emd/main
Repository URL      : https://codadevdb:1158/em/upload/
Started at         : 2015-11-20 13:50:53
Started by user    : oracle
Last Reload        : 2015-11-20 13:50:53
Last successful upload                    : (none)
Last attempted upload                     : (none)
Total Megabytes of XML files uploaded so far :    0.00
Number of XML files pending upload        :      30
Size of XML files pending upload(MB)      :    26.73
Available disk space on upload filesystem :    40.48%
Data channel upload directory             :
/u01/app/oracle/product/11.2.0/dbhome_1/codadevdb_codadev/sysman/recv
Collection Status                         : Disabled by Upload Manager
Last successful heartbeat to OMS          : 2015-11-20 13:51:25
---------------------------------------------------------------
```