

유용한 Shell script

파일을 찾아서 변경하는 shell

•

```
if [ "${2}" = "" ]
then
    echo Input 2 Arguments !!!
    exit
fi

DOMAIN_HOME=${PWD}
DOMAIN_HOME=/weblogic/wls64/user_projects/domains/dcnpwas
echo "DOMAIN_HOME : ${DOMAIN_HOME}"
echo "AS-IS : ${1}"
echo "TO-BE : ${2}"
echo "Start Step 1"
find ./ -type f -name "*.sh" -exec perl -pi -e "s/${1}/${2}/g" {} \;
echo "Start Step 2"
find ./ -type f -name "*.xml" -exec perl -pi -e "s/${1}/${2}/g" {} \;
echo "Start Step 3"
find ./ -type f -name "*.properties" -exec perl -pi -e "s/${1}/${2}/g" {} \;
echo "Start Step 4"
find ./ -type f -name "*.py" -exec perl -pi -e "s/${1}/${2}/g" {} \;
echo "Done"
```

혹시 61번 서버에서 테스트한 소스를 4.2나 4.3dp 옮기시려면

[2014-02-17 오후 5:49:42] 김태우님의 말:

```
scp -P 38 -rp * weblogic@172.19.4.2:/weblogic/wls64/user_projects/domains/bidwas1/p
```

예) server1 의 /data를 s01wasp2 의 /data로 복사

양쪽 서버 간에 rsh 로그인 되도록 설정후 실행

s01wasp2서버는 rsh 설정이 되어있음

gtar 명령어예제

server1 에서 다음을 수행

```
server1:[/] cd /data
```

```
server1:[/data] gtar -cvf - * | (rsh s01wasp2 "cd /data ; gtar -xvf -" )
```

rsync 명령어예제

server1 에서 다음을 수행

```
rsync -artv --rsh=rsh /data s01wasp2:/
```

조회한 파일리스트를 잘라내서 저장 - 전체 class 와 java 파일을 비교(diff)하기 위해서 사용함

```
find ./temp -name "*.java" | awk -F ".java" '{printf $1"\n"}' | awk -F/temp/src/ '{;
find ./webapps -name "*.class" | awk -F/webapps/WEB-INF/classes/ '{print $2}' | awk
```

jeus WAS 배포 - 전체 배포

```
#!/bin/sh

echo
echo '##### START DEPLOY #####'
echo

if [[ $# -eq 0 ]];then
    echo "ERROR 255: Must provide the following 1 system name parameters:"
    exit 255
fi

echo '#Actually system name = ' $1
echo

#### JAVA ENV ####
export JAVA_HOME=/usr/java7_64
export PATH=$JAVA_HOME/bin:$PATH:.

#### JEUS ENV ####
export JEUS_HOME=/sw/jeusadm/jeus7
export PATH="$JEUS_HOME/bin:$JEUS_HOME/lib/system:$JEUS_HOME/webserver/bin:${PATH}"
export JEUS_LOG_HOME=/swlog/jeus

echo
echo '1. Backup & Stop'
export NOW=`date +%Y%m%d`
export OLDDAY=$((NOW -2))
export NOWTIME=`date +%H%m`
export NUMOFSEVER=`uname -n | cut -f2 -d"0"`
export SERVER=$1"_server"$NUMOFSEVER
export DOMAIN=lp$1"_domain"
export PORT

if [ $1 = "sis" ]
then
    PORT=10100
else
    if [ $1 = "pis" ]
    then
        PORT=10200
    else
        if [ $1 = "scm" ]
        then
            PORT=10500
        else
```

```

        echo " ### ERROR SERVIER NOT EXIST only sis, pis, scm ###"
        exit 1

    fi

fi

echo ' > Deploy Server['$SERVER'], PORT['$PORT'], Domain['$DOMAIN'], Number of ser

echo ' > Today = ' $NOW$NOWTIME
cd /app/ldsp/$1
find /app/ldsp/$1 -xdev | grep -v 'nas' > /app/ldsp/$1/flist
tar -cf /app/ldsp/backup/$1-$NOW$NOWTIME.tar -L /app/ldsp/$1/flist
echo ' > Backup file : ' $1-$NOW$NOWTIME.tar
rm /app/ldsp/$1/flist

echo
echo '2. Shutdown ' $DOMAIN
jeusadmin -domain $DOMAIN -host 10.7.17.29:$PORT -f /sw/jeusadm/jeus7/bin/scripts/j

echo '3. Delete 20 day ago '
find /app/ldsp/backup -mtime +20 -print | xargs rm -f

echo
echo '4. Delete web root '
echo ' > Delete ' $1
cd /app/ldsp/$1
ls /app/ldsp/$1 | grep -v nas | grep -v export | grep -v import | xargs rm -rf
echo ' > Clear check ' $1
ls /app/ldsp/$1

echo
echo '5. Copy WAR to ' $1
echo ' > Copy War to ' $1
cp /app/ldsp/publishing/emdv3.war /app/ldsp/$1

echo
echo '6. Decompress WAR'
echo ' > Decompress ' $1
cd /app/ldsp/$1
jar -xf /app/ldsp/$1/emdv3.war

echo
echo '7. Deploy Report'
echo ' > Deploy report ' $1 ' --> /app/ldsp/rpt/ReportingServer/mrd'
cd /app/ldsp/$1
cp -fR /app/ldsp/$1/report /app/ldsp/rpt/ReportingServer/mrd/

echo
echo '8. Deploy Configurations'
echo ' > Deploy ldsp.xml'
cp /app/ldsp/publishing/config/context-ldsp.xml /app/ldsp/$1/WEB-INF/classes/spring
echo ' > Deploy logback.xml'
cp /app/ldsp/publishing/config/logback.xml /app/ldsp/$1/WEB-INF/classes
echo ' > Deploy ldsp.properties'
cp /app/ldsp/publishing/config/resources/properties/ldsp.properties /app/ldsp/$1/WE

echo
echo '9. Startup ' $1
startDomainAdminServerNM -host 10.7.17.29 -port 7730 -domain $DOMAIN -server $SERVE

echo
echo '##### FINISH DEPLOY #####'
echo
echo '10. Finish deploy'
echo 'Bye'

```

Jeus WAS 개별 배포

```
#!/bin/sh

echo
echo '##### START DEPLOY #####'
echo

if [[ $# -eq 0 ]];then
    echo "ERROR 255: Must provide the following 1 system name parameters:"
    exit 255
fi

echo '#Actually system name = ' $1
echo

#### JAVA ENV ####
export JAVA_HOME=/usr/java7_64
export PATH=$JAVA_HOME/bin:$PATH:.

#### JEUS ENV ####
export JEUS_HOME=/sw/jeusadm/jeus7
export PATH="$JEUS_HOME/bin:$JEUS_HOME/lib/system:$JEUS_HOME/webserver/bin:${PATH}"
export JEUS_LOG_HOME=/swlog/jeus

echo
echo '1. Backup & Stop'
export NOW=`date +%Y%m%d`
export OLDDAY=$((NOW -2))
export NOWTIME=`date +%H%m`
export NUMOFSEVER=`uname -n | cut -f2 -d"0"`
export SERVER=$1"_server"$NUMOFSEVER
export DOMAIN=lp$1"_domain"
export PORT

if [ $1 = "sis" ]
then
    PORT=10100
else
    if [ $1 = "pis" ]
    then
        PORT=10200
    else
        if [ $1 = "scm" ]
        then
            PORT=10500
        else
            echo " ### ERROR SERVIEW NOT EXIST only sis, pis, scm ###"
            exit 1
        fi
    fi
fi

echo ' > Deploy Server['$SERVER'], PORT['$PORT'], Domain['$DOMAIN'], Number of ser

echo ' > Today = ' $NOW$NOWTIME
cd /app/ldsp/$1
find /app/ldsp/$1 -xdev | grep -v 'nas' > /app/ldsp/$1/flist
```

```

tar -cf /app/ldsp/backup/$1-IND-$NOW$NOWTIME.tar -L /app/ldsp/$1/flist
echo ' > Backup file : ' $1-IND-$NOW$NOWTIME.tar
rm /app/ldsp/$1/flist

echo '3. Delete 20 day ago '
find /app/ldsp/backup -mtime +20 -print | xargs rm -f

echo
echo '3. Make temp folder for decompress : ' $1
mkdir /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
ls -l /app/ldsp/publishing

echo
echo '4. Copy JAR to tmp for ' $1
echo ' > Copy JAR to tmp for ' $1
cp /app/ldsp/publishing/emdv3-ind-was.jar /app/ldsp/publishing/temp-$1-$NOW$NOWTIME

echo
echo '5. Decompress JAR'
echo ' > Decompress ' $1
cd /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
jar -xf /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/emdv3-ind-was.jar
ls -lR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
export TOTALCLASSFILES=`find /app/ldsp/publishing/temp-$1-$NOW$NOWTIME -name '*.cla
export TOTALXMLFILES=`find /app/ldsp/publishing/temp-$1-$NOW$NOWTIME -name '*.xml'
export TOTALMRDFILES=`find /app/ldsp/publishing/temp-$1-$NOW$NOWTIME -name '*.mrd'
export TOTALFILES=`expr $TOTALCLASSFILES + $TOTALXMLFILES + $TOTALMRDFILES`
echo ' Total classes=' $TOTALCLASSFILES ' , xml=' $TOTALXMLFILES ' , mrd=' $TOTALMR
echo ' Total files = ' $TOTALFILES

if [ $TOTALFILES = 0 ]
then
    echo ' !!!!! ### No file ' $1 ' ### !!!!!'
else
    echo
    echo '2. Shutdown ' $DOMAIN
    jeusadmin -domain $DOMAIN -host 10.7.17.29:$PORT -f /sw/jeusadm/jeus7/bin/s

    echo
    echo '7. Copy class, mrd, xml to ' $1
    cd /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
    cp -fR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/java/WEB-INF/classes/com /
    cp -fR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/mrd/report /app/ldsp/rpt/R
    cp -fR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xml/WEB-INF/classes/sqlmap

    echo ' > Deploy ldsp.properties'
    cp /app/ldsp/publishing/config/resources/properties/ldsp.properties /app/ld

    echo
    echo '8. Startup ' $1
    startDomainAdminServerNM -host 10.7.17.29 -port 7730 -domain $DOMAIN -serve

fi

echo
echo '9 Clear delete files war & jar '
cd /app/ldsp/publishing/
rm -rf /app/ldsp/publishing/temp-$1-$NOW$NOWTIME

echo
echo '##### FINISH DEPLOY #####'
echo
echo '10. Finish deploy'
echo 'Bye'

```

Webtob 전체 배포

```
#!/bin/sh

echo
echo '##### START DEPLOY #####'
echo

if [[ $# -eq 0 ]];then
    echo "ERROR 255: Must provide the following 1 system name parameters:"
    exit 255
fi

if [ $1 = "sis" ]
then
    echo '#Actually system name = ' $1
else
    if [ $1 = "pis" ]
    then
        echo '#Actually system name = ' $1
    else
        if [ $1 = "scm" ]
        then
            echo '#Actually system name = ' $1
        else
            echo " ### ERROR SERVIEW NOT EXIST only sis, pis, scm ###"
            exit 1
        fi
    fi
fi

echo '#Actually system name = ' $1
echo
echo
echo '1. Backup & Stop'
export NOW=`date +%Y%m%d`
export OLDDAY=$((NOW -2))
export NOWTIME=`date +%H%m`

echo ' > Today = ' $NOW$NOWTIME
cd /app/ldsp/$1
find /app/ldsp/$1 -xdev | grep -v 'nas' > /app/ldsp/$1/flist
tar -cf /app/ldsp/backup/$1-$NOW$NOWTIME.tar -L /app/ldsp/$1/flist
echo ' > Backup file : ' $1-$NOW$NOWTIME.tar
rm /app/ldsp/$1/flist

echo '2. Delete two day ago = ' $1-$OLDDAY*
rm -rf /app/ldsp/backup/$1-$OLDDAY*.tar

echo
echo '3. Delete web root '
echo ' > Delete ' $1
cd /app/ldsp/$1
ls /app/ldsp/$1 | grep -v nas | grep -v export | grep -v import | xargs rm -rf
echo ' > Clear check ' $1
ls /app/ldsp/$1

echo
echo '4. Copy JAR to ' $1
echo ' > Copy JAR to ' $1
```

```

cp /app/ldsp/publishing/emdv3-web.jar /app/ldsp/$1

echo
echo '5. Decompress JAR'
echo ' > Decompress ' $1
cd /app/ldsp/$1
jar -xf /app/ldsp/$1/emdv3-web.jar

echo
echo '6. Copy config files'
echo ' > Deploy editor hp_SE2M_AttachQuickPhoto.js'
cp /app/ldsp/publishing/config/webapp/EMDV3/editor/photo_uploader/plugin/hp_SE2M_At
echo ' > Deploy editor attach_photo.js'
cp /app/ldsp/publishing/config/webapp/EMDV3/editor/photo_uploader/popup/attach_phot
echo ' > Deploy editor_portal hp_SE2M_AttachQuickPhoto.js'
cp /app/ldsp/publishing/config/webapp/EMDV3/editor_portal/photo_uploader/plugin/hp_
echo ' > Deploy editor_portal attach_photo.js'
cp /app/ldsp/publishing/config/webapp/EMDV3/editor_portal/photo_uploader/popup/atta
echo ' > Deploy TopFrame.xfdl.js'
cp /app/ldsp/publishing/config/webapp/EMDV3/Frame/TopFrame.xfdl.js /app/ldsp/$1/EMD

echo
echo '7. Clear delete files war & jar '
rm /app/ldsp/$1/emdv3-web.jar

echo
echo '##### FINISH DEPLOY #####'
echo
echo '8. Finish deploy'
echo 'Bye'

```

Webtob 개별 배포

```

#!/bin/sh

echo
echo '##### START DEPLOY #####'
echo

if [[ $# -eq 0 ]];then
    echo "ERROR 255: Must provide the following 1 system name parameters:"
    exit 255
fi

if [ $1 = "sis" ]
then
    echo " ### Ind EMDVSIS ###"
else
    if [ $1 = "pis" ]
    then
        echo " ### Ind EMDVPIS ###"
    else
        if [ $1 = "scm" ]
        then
            echo " ### Ind EMDVSCM ###"
        else
            echo " ### ERROR SERVIER NOT EXIST only sis, pis, scm ###"
        fi
    fi
fi

```

```

                                exit 1
                                fi
                                fi
                                fi

echo '#Actually system name = ' $1 ' Attention : JS file Only !!!!!'
echo
echo
echo '1. Backup & Stop'
export NOW=`date +%Y%m%d`
export OLDDAY=$((NOW -2))
export NOWTIME=`date +%H%m`

echo ' > Today = ' $NOW$NOWTIME
cd /app/ldsp/$1
find /app/ldsp/$1 -xdev | grep -v 'nas' > /app/ldsp/$1/flist
tar -cf /app/ldsp/backup/$1-IND-$NOW$NOWTIME.tar -L /app/ldsp/$1/flist
echo ' > Backup file : ' $1-IND-$NOW$NOWTIME.tar
rm /app/ldsp/$1/flist

echo '2. Delete two day ago = ' $1-IND-$OLDDAY
rm -rf /app/ldsp/backup/$1-IND-$OLDDAY*.tar

echo
echo '3. Make temp folder for decompress : ' $1
mkdir /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
ls -l /app/ldsp/publishing

echo
echo '4. Copy JAR to tmp for ' $1
echo ' > Copy JAR to tmp for ' $1
cp /app/ldsp/publishing/emdv3-ind-web.jar /app/ldsp/publishing/temp-$1-$NOW$NOWTIME

echo
echo '5. Decompress JAR'
echo ' > Decompress ' $1
cd /app/ldsp/publishing/temp-$1-$NOW$NOWTIME
jar -xf /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/emdv3-ind-web.jar
ls -lR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xfdl/EMDV3
find /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xfdl/EMDV3 -name '*.js' | wc -l | xa
export TOTALFILES=`find /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xfdl/EMDV3 -name
if [ $TOTALFILES = 0 ]
then
    echo ' !!!!! ### No file ' $1 ' ### !!!!!'
else
    echo
    echo '6. Copy xfdl to ' $1
    cd /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xfdl
    cp -fR /app/ldsp/publishing/temp-$1-$NOW$NOWTIME/xfdl/EMDV3 /app/ldsp/$1

    echo
    echo '7. Copy config files'
    echo ' > Deploy editor hp_SE2M_AttachQuickPhoto.js'
    cp /app/ldsp/publishing/config/webapp/EMDV3/editor/photo_uploader/plugin/hp
    echo ' > Deploy editor attach_photo.js'
    cp /app/ldsp/publishing/config/webapp/EMDV3/editor/photo_uploader/popup/att
    echo ' > Deploy editor_portal hp_SE2M_AttachQuickPhoto.js'
    cp /app/ldsp/publishing/config/webapp/EMDV3/editor_portal/photo_uploader/pl
    echo ' > Deploy editor_portal attach_photo.js'
    cp /app/ldsp/publishing/config/webapp/EMDV3/editor_portal/photo_uploader/po
    echo ' > Deploy TopFrame.xfdl.js'
    cp /app/ldsp/publishing/config/webapp/EMDV3/Frame/TopFrame.xfdl.js /app/lds;
fi

```

```

echo
echo '8 Clear delete files  war & jar '
cd /app/ldsp/publishing
rm -rf /app/ldsp/publishing/temp-$1-$NOW$NOWTIME

echo
echo '##### FINISH DEPLOY #####'
echo
echo '9. Finish deploy'
echo 'Bye'

```

port check

```

#!/bin/bash

CHECKREMOTESERVERSLIST=checkServerPortList.txt
TIMEOUT=2

OLDIFS=$CHECKREMOTESERVERSLIST
IFS=,

while read -a line;
do
    r=$(timeout 2 bash -c 'exec 3<> /dev/tcp/${line[0]}/${line[1]};echo $?' 2>/dev/null)
    #l_TELNET=`echo "quit" | telnet ${line[0]} ${line[1]} | grep "Escape character" && echo`
    if [ "$r" = "0" ]
    then
        echo "Success !"
    else
        if [ "$r" = "1" ]
        then
            echo "Checked"
        else
            echo "Failed"
        fi
    fi
done <$CHECKREMOTESERVERSLIST
IFS=$OLDIFS

```

HomePlus

```

#!/bin/sh

# /data/scm/homeplus-club-admin/homeplus-club-admin-web
echo
echo '##### START DEPLOY #####'
echo

echo '##### Set Env #####'

```

```

##### JAVA #####
export JAVA_HOME=/app/java/java8
export PATH=$PATH:$JAVA_HOME/bin/
export CLASSPATH=$CLASSPATH:$JAVA_HOME/lib/tools.jar
export Finalbackupfiles=5
export DeleteDate=3
PATH=$PATH:$HOME/bin
export PATH
echo " > JAVA_HOME=" $JAVA_HOME
echo " > Final Backup File count=" $Finalbackupfiles
echo " > DeleteDate=" $DeleteDate "day before"
echo
echo '1. Backup & Stop'
export NOW=`date +%Y%m%d`
export OLDDAY=$((NOW - $DeleteDate))
export NOWTIME=`date +%H%M`
echo ' > Today = ' $NOW$NOWTIME

cd /data/tmpsadm
find /data/tmpsadm -xdev | grep -v 'nas' > /data/tmpsadm/achiving.list
tar -cf /data/homeplus/backup/homeplus-club-admin/homeplus-club-admin-web/$NOW$NOWT
echo ' > Backup file : ' /data/homeplus/backup/homeplus-club-admin/homeplus-club-ad
echo '2. Delete ' $DeleteDate ' day ago = ' $OLDDAY ', but must remain ' $Finalback
Totalbackupfiles=`find /data/homeplus/backup/homeplus-club-admin/homeplus-club-admi
echo ' > Total backupfiles(all files) = ' $Totalbackupfiles
echo ' > Final backup files count = ' $Finalbackupfiles
Targetdeletebackupfiles=`expr $Totalbackupfiles - $Finalbackupfiles`
echo ' > Total backup files - Final backup files = ' $Targetdeletebackupfiles
BackupAllFiles=( `find /data/homeplus/backup/homeplus-club-admin/homeplus-club-admin
cnt=1
echo ' > ' $DeleteDate ' day before delete target file list'
for i in "${BackupAllFiles[@]}"
do
    if [ $Targetdeletebackupfiles -ge $cnt ]
    then
        echo " " $i " , " $cnt " DELETEed !"
    else
        echo " " $i " , " $cnt " SKIPed !"
    fi
    let cnt=cnt+1
done

echo '3. Delete Web ROOT'
cd /data/tmpsadm
ls /data/tmpsadm | grep -v nas | grep -v export | grep -v somethingspecial | xargs .
echo ' > Check Web ROOT = /data/tmpsadm'
echo ' > list ' `ls /data/tmpsadm`

echo '4. Copy tar to /data/tmpsadm'
echo ' > cp OOT-jar.tar.gz to /data/tmpsadm'
cp /data/scm/homeplus-club-admin/homeplus-club-admin-web/target/ROOT-jar.tar.gz /da

echo '5. Decompress TAR.GZ'
echo ' > tar -zxvf /data/tmpsadm/ROOT-jar.tar.gz'
cd /data/tmpsadm
tar -zxvf /data/tmpsadm/ROOT-jar.tar.gz -C /data/tmpsadm/ --strip-components=1

echo '6. Clear '
echo ' > Delete ROOT-jar.tar.gz and achiving.list in Web ROOT'
cd /data/tmpsadm
rm -rf /data/tmpsadm/ROOT-jar.tar.gz
rm -rf /data/tmpsadm/achiving.list
echo

echo '##### FINISH DEPLOY #####'
echo 'Bye'

```

Linux Core 수 확인

grep -c processor /proc/cpuinfo 코어 수 확인

Linux Memory 확인

cat /proc/meminfo | grep [MemTotal](#) 메모리 확인

Tomcat instance 운용 shell

•

```
#!/bin/bash

#####

##### TOMCAT #####
INSTID=$2
ARGV=$3
INSTNAME=$1
JAVA_HOME=${JAVA_HOME}
CATALINA_HOME=/app/tomcat8/tomcat-engine
CATALINA_BASE=/app/tomcat8/instance/$INSTNAME
TOMCAT_USER=alan
TPATH=/app/tomcat8/tomcat-engine/bin/bootstrap.jar
TLOG=${CATALINA_BASE}/logs
#LOGNAME=catalina.out

##### JAVA #####
### jvm option
JAVA_OPTS="$JAVA_OPTS -Dinstance_name=$INSTID"
JAVA_OPTS="$JAVA_OPTS -Dserver.type=prd"
JAVA_OPTS="$JAVA_OPTS -Dconnection.type=jndi"
JAVA_OPTS="$JAVA_OPTS -server"
JAVA_OPTS="$JAVA_OPTS -Dcomponent.name=$INSTNAME"
JAVA_OPTS="$JAVA_OPTS -Dlogs.dir=$TLOG"
#JAVA_OPTS="$JAVA_OPTS -Dfile.encoding=UTF-8"
JAVA_OPTS="$JAVA_OPTS -Dfile.encoding=EUC-KR -Dfile.client.encoding=EUC-KR -Dclient
JAVA_OPTS="$JAVA_OPTS -Dinstance.no=$INSTNAME"
JAVA_OPTS="$JAVA_OPTS -Djava.security.egd=file:/dev/urandom"

### performance option
```

```

JAVA_OPTS="$JAVA_OPTS -Xmx4096m -Xms4096m"
#JAVA_OPTS="$JAVA_OPTS -XX:PermSize=256m -XX:MaxPermSize=512m"
#JAVA_OPTS="$JAVA_OPTS -XX:NewRatio=3 -XX:SurvivorRatio=2"

### behavior option
JAVA_OPTS="$JAVA_OPTS -XX:+UseConcMarkSweepGC -XX:+UseParNewGC"

### debugging option
JAVA_OPTS="$JAVA_OPTS -verbose:gc -XX:+PrintGC -XX:+PrintGCDetails -XX:+PrintGCTime
JAVA_OPTS="$JAVA_OPTS -XX:+TraceClassUnloading -XX:+TraceClassLoading"
JAVA_OPTS="$JAVA_OPTS -Xloggc:$TLOG/gc.log_$(date +%Y%m%d-%H%M%S)"
JAVA_OPTS="$JAVA_OPTS -XX:+HeapDumpOnOutOfMemoryError"
JAVA_OPTS="$JAVA_OPTS -XX:HeapDumpPath=$TLOG/java_pid_$(date +%Y%m%d-%H%M%S).hprof"

### external CLASSPATH ###
### APM properties
#JAVA_OPTS="$JAVA_OPTS -agentpath:/app/dynatrace/dynatrace-7.1/agent/lib64/libdtage
#JAVA_OPTS="$JAVA_OPTS -javaagent:/app/scouter/agent.java/scouter.agent.jar"
#JAVA_OPTS="$JAVA_OPTS -Dscouter.config=/app/scouter/agent.java/conf/scouter_${INSTN

### RKM OPTION
#CLASSPATH="$CLASSPATH:/app/rkm/src/RKM_APP"
#CLASSPATH="$CLASSPATH:/app/rkm/src/RKM_APP/lib/commons-codec-1.10.jar"
#CLASSPATH="$CLASSPATH:/app/rkm/src/RKM_APP/lib/commons-lang-2.4.jar"
#CLASSPATH="$CLASSPATH:/app/rkm/src/RKM_APP/lib/commons-logging-1.1.jar"

export JAVA_HOME JAVA_OPTS CATALINA_HOME CATALINA_BASE CLASSPATH
export CATALINA_OPTS=${JMX_OPTS}
#####

case $ARGV in
start)
  runinst=$(ps -ef | grep java | grep -v grep | grep $INSTID | awk {'print $2 $16'})

  UNAME=`id -u -n`
  if [ e$UNAME != "e$TOMCAT_USER" ]
  then
    echo "[ Use by only user Account [ $TOMCAT_USER ] Start Fail Tomcat I
    exit;
  fi

  if [ -z "$runinst" ]; then
    echo " Starting Tomcat Instance [ $1 ] "
    echo $runinst
    mv $TLOG/catalina.$(date +%Y-%m-%d).out $TLOG/catalina.$(date +%Y-%m-%d_%H%M%
    $CATALINA_HOME/bin/catalina.sh $3
  else
    echo " Already Running Tomcat Instance [ $1 ][ $INSTID ] !!! "
    ERROR=$?
  fi
  ;;

stop)
  $CATALINA_HOME/bin/catalina.sh $3
  ;;

status)
  instcount=$(ps -ef | grep java | grep -v grep | grep $TPATH | wc -l)
  echo " instance Running Count = "$instcount
  ps -ef | grep java | grep -v grep | grep $TPATH | awk '{printf(" Tomcat Instanc
  echo -e $inststatus
  ;;

kill)

```

```

    ps -ef | grep $TPATH | grep -v grep | grep $INSTID | awk '{ printf(" Tomcat Instance ID: %s\n", $3); }'
    ps -ef | grep $TPATH | grep -v grep | grep $INSTID | awk '{ printf("kill -9 %s\n", $3); }'
    sh tmp.$$
    rm -f tmp.$$
    ;;

thread)
    threadPID=$(ps -ef | grep $TPATH | grep -v grep | grep $INSTID | awk '{print $3}')
    threadCount=$(ps uH $threadPID | wc -l )
    echo " Tomcat Instance ID \"$INSTID\" [ PID : \"$threadPID\" / Sub THREAD : \"$threadCount\" ]"
    ;;

log)
    tail -100f $TLOG/$INSTNAME/catalina.out
    ;;

*)
    echo "Invalid parameter [$ARGV]"
    echo "Usage) luncher.sh [instance name] [instance id] [ start|status|stop|kill|restart|start ]"
    echo "ex) luncher.sh inst1 inst11 start"
    echo "===== "
    echo " Tomcat Instance List "
    echo "===== "
    INSTLISTNUM=$(ls $CATALINA_BASE | wc -l )
    echo " Instance Count[ $INSTLISTNUM ] : " $(ls $CATALINA_BASE)
    echo "===== "

    ;;

esac

exit $ERROR

```

Backup Shell

- 02 *** /home/hyunsu/BackUpCenter/backUpPicture.sh >> /home/hyunsu/BackUpCenter/backupPictures.log

```

#!/bin/sh

BackUpFiles=5
OrgDirectory=/home/hyunsu/Pictures
BackUpDirectory=/data/ext/WD4TB/backup/Pictures

FilesFound=$(find /home/hyunsu/Pictures/ -mtime -$BackUpFiles )

checkFileExist()
{
    filename=`echo $1 | cut -d'/' -f5-`
    echo ">> " $BackUpDirectory/$filename
    if [ -f $BackUpDirectory/$filename ]; then
        echo "EXIST"
    else
        echo "NOT EXIST --> Copy " $1 " --> " $BackUpDirectory/$filename
        cp $1 $BackUpDirectory/$filename
    fi
}

```

```

checkDirectoryExist()
{
    filename=`echo $1 | cut -d'/' -f5-`
    echo ">> " $BackUpDirectory/$filename
    if [ -d $BackUpDirectory/$filename ]; then
        echo "EXIST"
        echo
    else
        echo "NOT EXIST --> Makedirectory " $BackUpDirectory/$filename
        mkdir $BackUpDirectory/$filename
    fi
}

for file in $FilesFound
do
    #echo "${counter}: ${file}"
    if [ -f "${file}" ]; then
        echo "${file} file."
        checkFileExist ${file}
    fi
    if [ -d "${file}" ]; then
        echo "${file} directory."
        checkDirectoryExist ${file}
    fi
done

```

Crontab에 걸어서 실행 여부를 체크하고 실행하지 않았으면 실행 - ELK 적용 사례

• */20 * * * * /app/checkELK.sh > /app/checkELK.log

```

#!/bin/sh

echo "\n\n"
echo "#####"
echo "##### CHECK ELK START ! #####"
echo "#####"

cd /app/elasticsearch-7.4.1/
elasticsearchcheck=`ps -ef | grep elasticsearch | awk '{print ($2)}' | wc -l`
if [ $elasticsearchcheck -gt 1 ]
then
    echo "Already Elasticsearch started !"
else
    echo "Start ElasticSearch !"
    /app/elasticsearch-7.4.1/startup.sh
fi

cd /app/kibana-7.4.1-linux-x86_64/
kibanacheck=`ps -ef | grep kibana | awk '{print ($2)}' | wc -l`
if [ $kibanacheck -gt 1 ]
then
    echo "Already Kibana started !"
else
    echo "Start Kibana !"
    /app/kibana-7.4.1-linux-x86_64/startup.sh
fi

```

```
cd /app/logstash-7.4.1/
logstashcheck=`ps -ef | grep logstash | awk '{print ($2)}' | wc -l`
if [ $logstashcheck -gt 1 ]
then
    echo "Already Logstash started !"
else
    echo "Start Logstash !"
    /app/logstash-7.4.1/startup.sh
fi

cd /app/
echo "#####"
echo "##### CHECK ELK END ! #####"
echo "#####"
echo "\n\n"
```

🔄Revision #2

★Created 2023-05-31 01:11:47 UTC by Hyeon Su Ryu

✎Updated 2023-05-31 14:52:12 UTC by Hyeon Su Ryu