Deploy Status post-processing script

properties.put("Status", "Success");

if (properties.get("exitCode") != 0) {

properties.put("Status", "Failure");

scanner.register("(?i)System.Exception: Could not connect to server", function(lineNumber, line) {

var errors = properties.get("SSRSDeployStatus");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("SSRSDeployStatus", errors);

}

errors.add(line+"\n");

properties.put("Status", "Failure");

});

scanner.scan();

var value = properties.get("SSRSDeployStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("SSRSDeployStatus", value);

}

properties.put("SSRSDeployStatus", value.toString());

}

else {

scanner.register("(?i)Report:", function(lineNumber, line) {

var value = properties.get("SSRSDeployStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("SSRSDeployStatus", value);

}

value.add(line+"\n");

properties.put("Status", "Success");

});

scanner.scan();

var value = properties.get("SSRSDeployStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("SSRSDeployStatus", value);

}

properties.put("SSRSDeployStatus", value.toString());

}

++++++++++++++++++++++

DB Deploy post-processing scripts

properties.put("Status", "Success");

if (properties.get("exitCode") != 0) {

properties.put("Status", "Failure");

}

else {

scanner.register("(?i)FAILED", function(lineNumber, line) {

var errors = properties.get("DBDeployStatus");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("DBDeployStatus", errors);

}

errors.add(line+"\n");

properties.put("Status", "Success");

});

scanner.register("(?i)The module will still be created; however, it cannot run successfully until the object exists", function(lineNumber, line) {

var errors = properties.get("DBDeployStatus");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("DBDeployStatus", errors);

}

errors.add(line+"\n");

properties.put("Status", "Success");

});

scanner.register("(?i)already exists, or you do not have permission to create it.", function(lineNumber, line) {

var errors = properties.get("DBDeployStatus");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("DBDeployStatus", errors);

}

errors.add(line+"\n");

properties.put("Status", "Failure");

});

scanner.scan();

var errors = properties.get("DBDeployStatus");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("DBDeployStatus", errors);

}

properties.put("DBDeployStatus", errors.toString());

}

++++++++++++++++++++

Post-processing script

properties.put("Status", "Success");

if (properties.get("exitCode") != 0) {

properties.put("Status", "Failure");

scanner.register("(?i)no such host", function(lineNumber, line) {

var value = properties.get("AWS-PushCFStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("AWS-PushCFStatus", "Rerun");

}

properties.put("Status", "Failure");

});

scanner.scan();

var value = properties.get("AWS-PushCFStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("AWS-PushCFStatus", value);

}

properties.put("AWS-PushCFStatus", "Rerun");

}

else {

scanner.register("(?i)no such host", function(lineNumber, line) {

var value = properties.get("AWS-PushCFStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("AWS-PushCFStatus", "Rerun");

}

properties.put("Status", "Failure");

});

scanner.scan();

var value = properties.get("AWS-PushCFStatus");

if (value == null) {

value = new java.util.ArrayList();

properties.put("AWS-PushCFStatus", "Rerun");

}

properties.put("AWS-PushCFStatus", "Rerun");

}

+++++++++++++++++

Service status

properties.put("Status", "Success");

if (properties.get("exitCode") != 0) {

scanner.register("(?i)No such file or directory", function(lineNumber, line) {

var errors = properties.get("Error");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("Error", errors);

}

errors.add(line+"\n");

properties.put("Status", "Success");

});

scanner.register("(?i)Couldn't stop", function(lineNumber, line) {

var errors = properties.get("Error");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("Error", errors);

}

errors.add(line+"\n");

properties.put("Status", "Failure");

});

scanner.register("(?i)already installed", function(lineNumber, line) {

var errors = properties.get("Error");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("Error", errors);

}

errors.add(line+"\n");

properties.put("Status", "Success");

});

scanner.register("(?i)already stopped", function(lineNumber, line) {

var errors = properties.get("Error");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("Error", errors);

}

errors.add(line+"\n");

properties.put("Status", "Success");

});

scanner.scan();

var errors = properties.get("Error");

if (errors == null) {

errors = new java.util.ArrayList();

properties.put("Error", errors);

}

properties.put("Error", errors.toString());

}

else {

properties.put("Status", "Success");

}

+++++++++++++++++++

Cleanup

IF EXIST ${p:environment/IIS\_WKDIR}\Web.config (

for /f "delims=" %%a in (' dir /b ') do (

IF %%a == Product (

echo " The folder is %%a"

for /f "delims=" %%b in ('dir /b %%a') do (

IF %%b == images (

echo "The sub-folder of Product is %%b & Don't delete"

) else IF EXIST %%b\NUL (

echo " Deleting the directory %%b"

rmdir /S /Q "%%a\%%b"

) else (

echo " Deleting file %%b"

DEL /Q "%%a\%%b"

)

)

) else IF %%a == TitleImages (

echo " The folder is %%a don't delete"

) else IF EXIST %%a\NUL (

echo " Deleting directory %%a"

rmdir /S /Q "%%a"

) else IF %%a == sslcheck.htm (

echo "Don't delete %%a"

) else (

echo " Deleting the file %%a"

DEL /Q "%%a"

)

)

) ELSE (

echo "No Cleanup required"

)

++++++++++++++++

Windows backup script

IF NOT EXIST "${p:environment/Backup\_Path}" (

MD ${p:environment/Backup\_Path}

)

D:

D:

cd ${p:environment/Backup\_Path}

echo %time%

set t=%time:~-11,2%%time:~-8,2%%time:~-5,2%

for /f "tokens=\* delims= " %%a in ("%t%") do set t=%%a

MD ${p:environment/AppName}\_%Date:~-7,2%%Date:~-10,2%%Date:~-4,4%\_%t%

cd ${p:AppName}\_%Date:~-7,2%%Date:~-10,2%%Date:~-4,4%\_%t%

IF EXIST ${p:environment/IIS\_WKDIR}\Web.config (

XCOPY "${p:environment/IIS\_WKDIR}" /E /Y /R

) ELSE (

echo 'No Backup taken as Web.config does not exist already'

)

+++++++++++++++++++

Shell script to deploy

#!/bin/bash

## Choose appropriate endpoints based on the region provided

if [[ ${p:component/repository} = "contracts" || ${p:component/repository} = "titles" ||${p:component/repository} = "schedules" ]]

then

echo "setting the memory 4GB for ${p:component/repository}"

PCF\_MEMORY=4G

else

echo "setting the memory 2GB for ${p:component/repository}"

PCF\_MEMORY=2G

fi

echo "Execute push and start application on ${p:environment/region} region for ${p:component/repository} using $PCF\_MEMORY"

if [ ${p:environment/EnvName} = "prod" ]

then

cf push ${p:component/repository} -d new.compass.${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target/${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m $PCF\_MEMORY -s cflinuxfs3 -u process

cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ PCF\_MEMORY\_calculator: { stack\_threads: 200 } }"

cf set-env ${p:component/repository} TZ "America/New\_York"

cf start ${p:component/repository}

else

cf push ${p:component/repository} -d compass-${p:environment/EnvName}.${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target/${p:component/repository}.jar -m $PCF\_MEMORY -s cflinuxfs3 -u process

cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ PCF\_MEMORY\_calculator: { stack\_threads: 200 } }"

cf set-env ${p:component/repository} TZ "America/New\_York"

cf start ${p:component/repository}

fi

+++++++++++++++++

Windows batch script

@REM =====================================================================================

@REM This script can deploy specified Microservice/Application to the specified COMPASS environment.

@REM Execute this script without any parameters to see usage

@REM Last update - Vasu - 11302020 -- new script provided by Ananthi

@REM Last update - Vasu - 06252020 -- Requested by KR

@REM Last update - Vasu - 11132019 -- new script provided by Ananthi

@REM =====================================================================================

@echo off

echo "executing deploy for ${p:environment/region},${p:component/repository},${p:environment/EnvName} "

@REM push the application according to environment and start application according to region.

if ${p:environment/region} EQU asz (

echo "The selected reqion is asz"

set HTTP\_PROXY=http://10.40.102.22:80

set HTTPS\_PROXY=http://10.40.102.22:80

if "${p:environment/EnvName}"=="staging" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS “-Dhttps.proxyHost=10.40.102.22 -Dhttps.proxyPort=80 -Dhttp.proxyHost=10.40.102.22 -Dhttp.proxyPort=80 -Dhttp.nonProxyHosts=uaa.sysasz.dmz.tfayd.com|p-spring-cloud-services.uaa.sysasz.dmz.tfayd.com|\*.uaa.sysasz.dmz.tfayd.com|\*.login.sysasz.dmz.tfayd.com -Dhttps.nonProxyHosts=uaa.sysasz.dmz.tfayd.com|p-spring-cloud-services.uaa.sysasz.dmz.tfayd.com|\*.uaa.sysasz.dmz.tfayd.com|\*.login.sysasz.dmz.tfayd.com”

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://10.40.102.22:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://10.40.102.22:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:environment/EnvName}"=="prod" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://10.40.102.22:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://10.40.102.22:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

) else if ${p:environment/region} EQU usz (

echo "The selected region is usz"

set HTTP\_PROXY=http://216.178.96.20:80

set HTTPS\_PROXY=http://216.178.96.20:80

if "${p:environment/EnvName}"=="staging" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS Ash Stg\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://216.178.96.20:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://216.178.96.20:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:environment/EnvName}"=="prod" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://216.178.96.20:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://216.178.96.20:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

) else if ${p:environment/region} EQU ush (

echo "Execute push and start application on other regions is ush"

if "${p:environment/EnvName}"=="staging" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS Ash Stg\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:environment/EnvName}"=="prod" (

echo "The selected reqion is ush prod"

if "${p:component/repository}"=="contracts" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m 4G -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:component/repository}"=="schedules" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m 4G -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:component/repository}"=="titles" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i 4 -k ${p:environment/harddisk} -m 8G -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

@REM call cf start ${p:component/repository}

) else if "${p:environment/EnvName}"=="staging1" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS Ash Stg1\" -Dappdynamics.agent.tierName=${p:component/repository} -Dspring.cloud.config.label=release4and5"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

) else (

echo "Execute push and start application on other regions except asz, usz, ush"

if "${p:environment/EnvName}"=="staging" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS Ash Stg\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:environment/EnvName}"=="prod" (

echo "Executing prod"

if "${p:component/repository}"=="contracts" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m 4G -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:component/repository}"=="schedules" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m 4G -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else if "${p:component/repository}"=="titles" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i 4 -k ${p:environment/harddisk} -m 8G -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS RePlatform Prod\" -Dappdynamics.agent.tierName=${p:component/repository}"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

) else if "${p:environment/EnvName}"=="staging1" (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -i ${p:environment/numberofinstances} -k ${p:environment/harddisk} -m ${p:environment/memory} -s cflinuxfs3

call cf set-env ${p:component/repository} JAVA\_OPTS "-Dappdynamics.http.proxyHost=proxy.inbcu.com -Dappdynamics.http.proxyPort=80 -Dappdynamics.agent.applicationName=\"COMPASS Ash Stg1\" -Dappdynamics.agent.tierName=${p:component/repository} -Dspring.cloud.config.label=performance"

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf bind-service ${p:component/repository} appdynamics

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

) else (

call cf push ${p:component/repository} -d ${p:environment/Domain} -b https://github.com/cloudfoundry/java-buildpack#v4.19 --no-hostname --no-start --route-path ${p:component/repository} -p target\${p:component/repository}.jar -s cflinuxfs3

call cf set-env ${p:component/repository} JBP\_CONFIG\_OPEN\_JDK\_JRE "{ memory\_calculator: { stack\_threads: 200 } }"

call cf set-env ${p:component/repository} HTTPS\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} HTTP\_PROXY "http://proxy.inbcu.com:80"

call cf set-env ${p:component/repository} TZ "America/New\_York"

call cf start ${p:component/repository}

)

)

+++++++++++++++++++

Backup process to cleanup old 10 backups in windows

cd ${p:application/Backup\_Path}

for /f "skip=10 eol=: delims=" %%F in ('dir /b /o-d') do rd /S /Q %%F

MD ${p:application/AppName}\_${p:version.name}

cd ${p:application/AppName}\_${p:version.name}

XCOPY "${p:application/IIS\_WKDIR}" /E /Y /R

+++++++++++++++++++

Pre-condition

(properties.get("environment.name") != "Local Deploy")

(properties.get("Deploy CITRIX") == "true") || (properties.get("Clean Citrix Folder") == "true")

++++++++++++++++++++

echo "Adding Agent ${p:agent.name} to Team ${p:TEAM}"

udclient addAgentToTeam -agent ${p:agent.name} -team ${p:TEAM}

+++++++++++++++