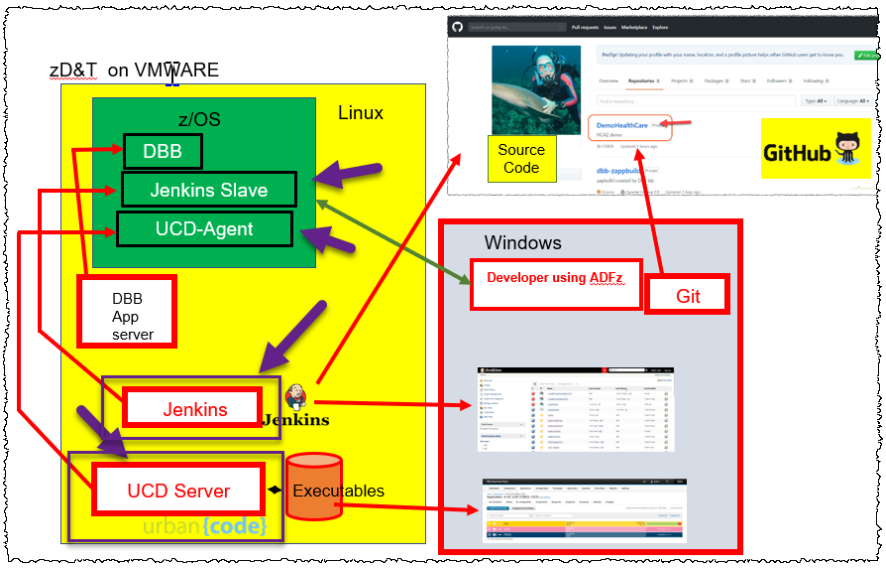
Jenkinsfile_example_using_DBB_and_UCD

Created by rbarosa@us.ibm.com – Aug 11, 2020
Thanks to Suman Gopinath for tips and guidance.

Introduction

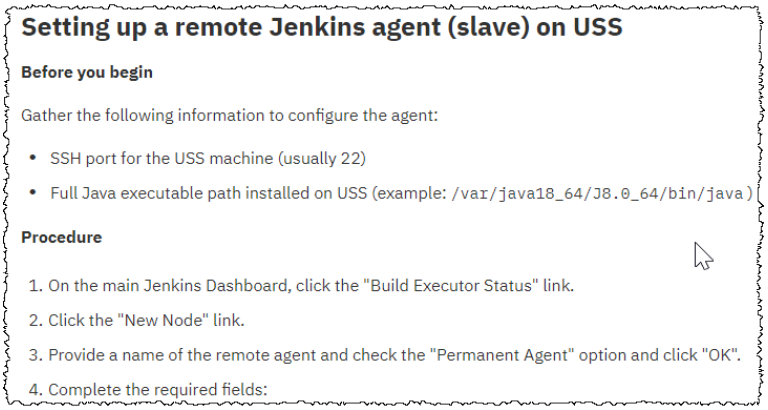
This paper may help you using Jenksfile using the DBB sample provided at the dbb-zappbuild framework.
For details on how to install and configure DBB you can also refer to the IBM docs.
I will show one example where Jenkins and UCD server are running on Linux with their respective agents on z/OS.
Below the architecture that I have tested:



Jenkins z/OS agent configuration

To setup a remote Jenkins agent (slave) on z/OS refer to the IBM docs.
at: https://www.ibm.com/support/knowledgecenter/en/SS6T76_1.0.9/jenkinsintegration.html

Follow the instructions on that link.



Setting up a remote Jenkins agent (slave) on USS

Before you begin

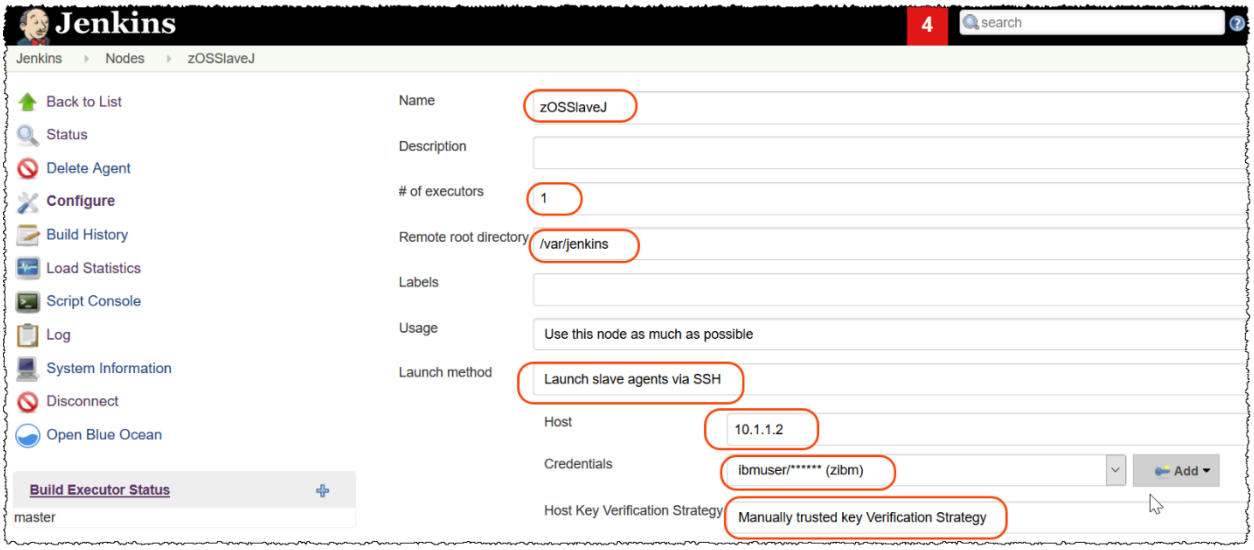
Gather the following information to configure the agent:

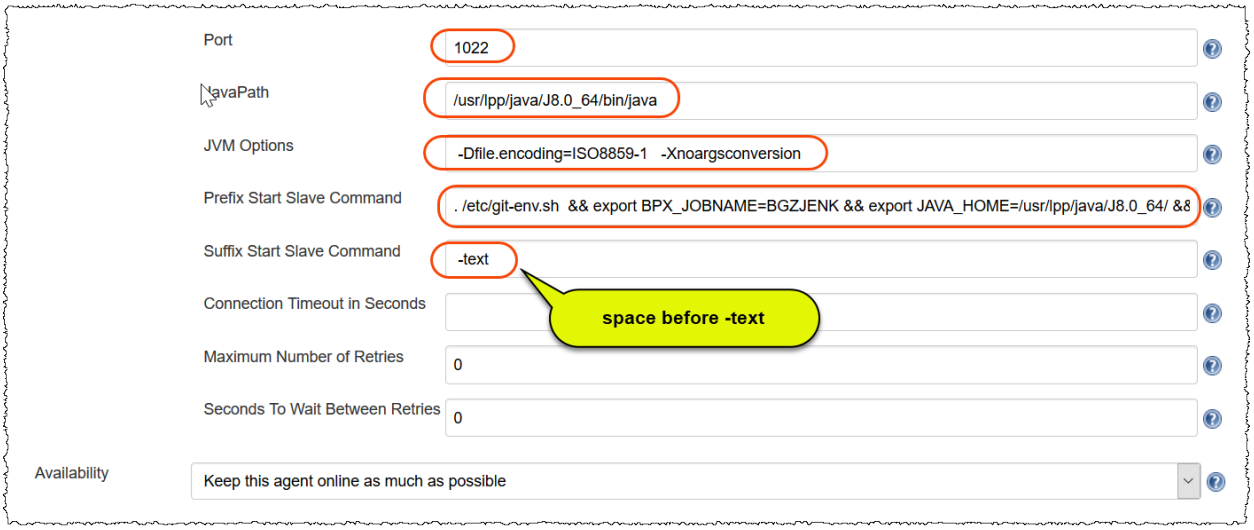
- SSH port for the USS machine (usually 22)
- Full Java executable path installed on USS (example: `/var/java18_64/J8.0_64/bin/java`)

Procedure

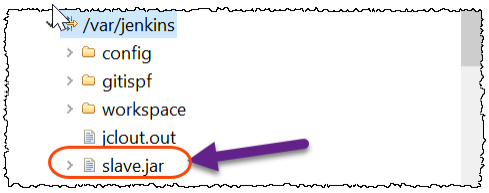
1. On the main Jenkins Dashboard, click the "Build Executor Status" link.
2. Click the "New Node" link.
3. Provide a name of the remote agent and check the "Permanent Agent" option and click "OK".
4. Complete the required fields:

Below an example that I have configured on my environment:
Prefix Start Slave Command is:
`. /etc/git-env.sh  && export BPX_JOBNAME=BGZJENK && export JAVA_HOME=/usr/lpp/java/J8.0_64/ && export IBM_JAVA_ENABLE_ASCII_FILETAG=ON  &&  env  &&`





The z/OS agent component will be “automatically” installed at the first execution.
Below the Jenkins “jar” file installed on z/OS

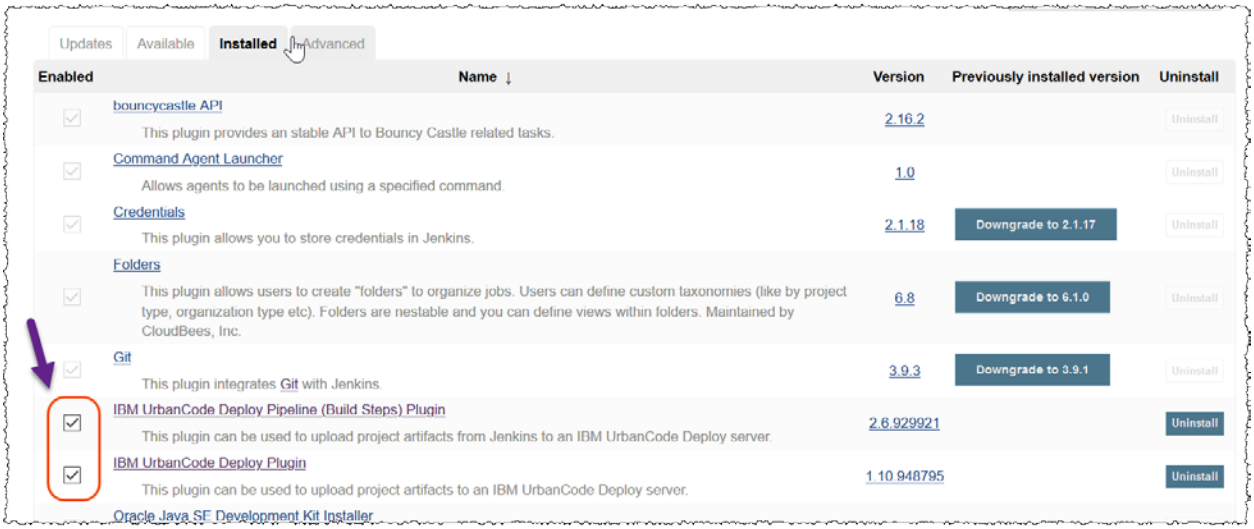


Installing UCD Plugin on Jenkins

Since I am using [UrbanCode Deploy] on this example I need the UCD plugin..
You can verify that there the Groovy plugin is installed on Jenkins:

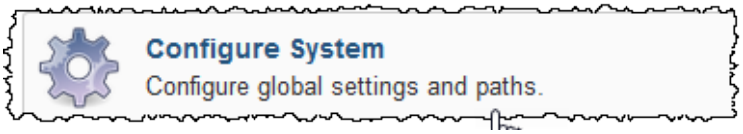
- On the main Jenkins dashboard click the "**Manage Jenkins**" link
- Click the "**Go to plugin manager**" link
- Click the "**Installed**" tab
- Verify that there are two IBM UrbanCode yplugin installed
- If not installed click on the "**Available**" tab and install those plugins, like you have it done for Git.

See on my example:

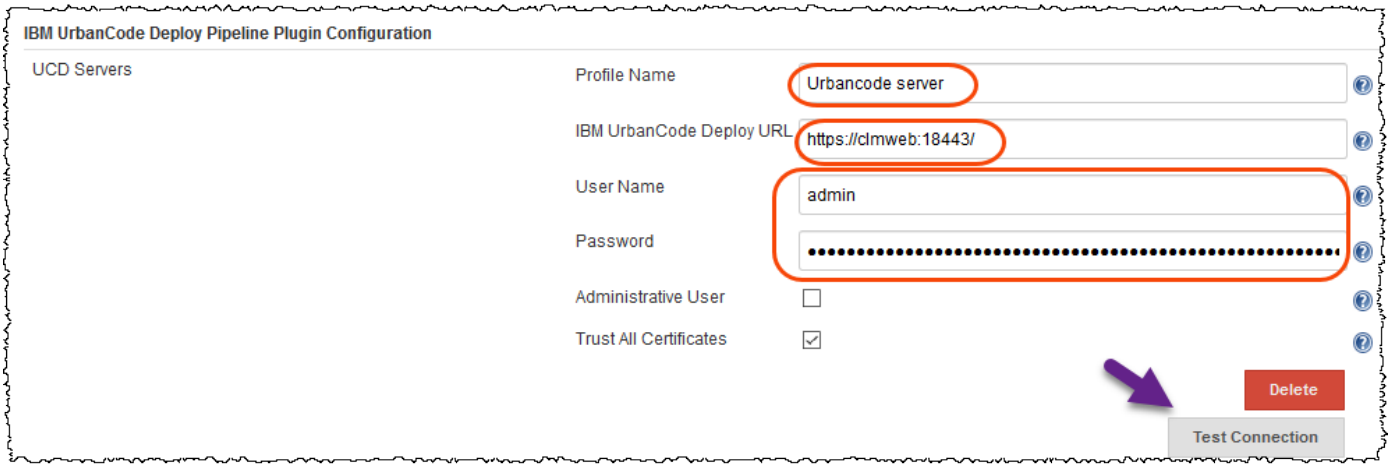


You also need to configure the UCD and point to where is your UCD server..

Go to **Jenkins > Manage Jenkins** > scroll down and select **Configure System**



Scroll down until you see the **IBM UrbanCode Deploy** and specify the **Profile Name**, **the URL** and the **UCD credentials** . You will need the Profile name at the Jenksfile.
It is a good idea to test this connection before saving it.
See my example:



Installing other Jenkins plugins

Install the plugin **Pipeline Utility Steps** its is required for my jenkinsfile example

Note: A nice plugin to have it installed is the Blue Ocean (see below)

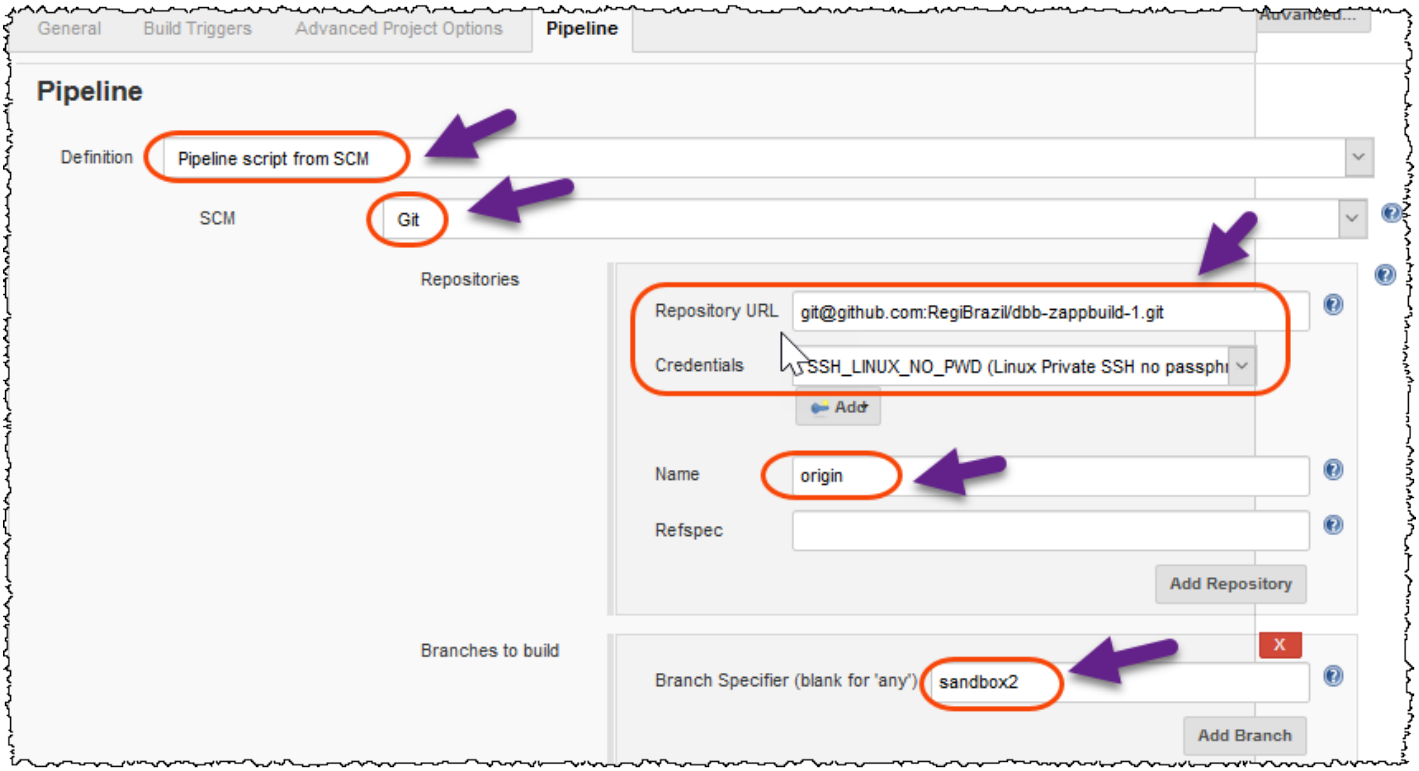
Setting up a Jenkins build project for MortgageApplication

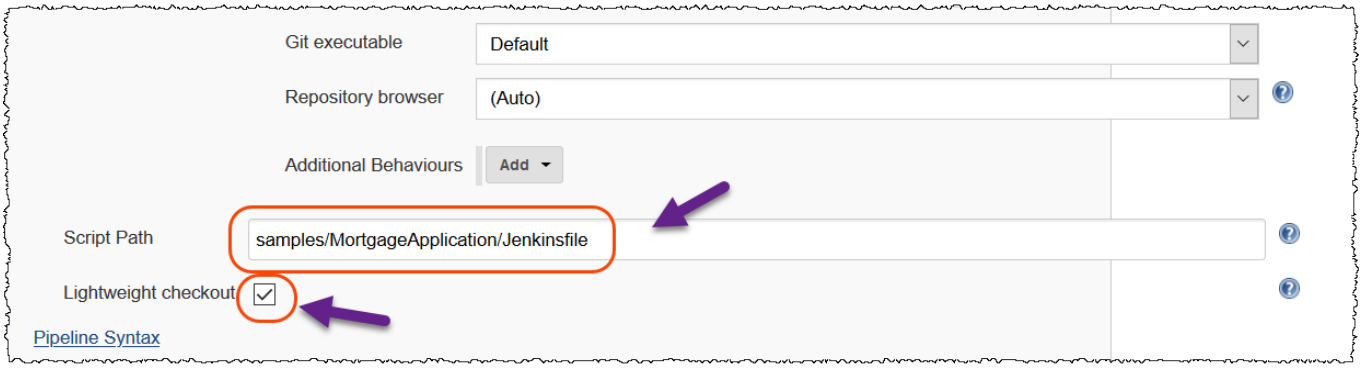
For details see: https://www.ibm.com/support/knowledgecenter/en/SS6T76_1.0.9/jenkinsintegration.html

To setup a remote Jenkins agent (slave) on z/OS refer to the IBM docs.
On My example to create the Jenkins pipeline I did:

- On the main Jenkins dashboard click the "**New Item**" link
- Enter a name (example: **dbb-zappbuild-1-sandbox2**)
- Choose **Pipeline script from SCM**
- SCM : **Git**
- Repository URL – Can use the copy/paste from GitHub
- Credentials better use SSH - See here for GitHub.
- Name **: origin**
- Branches to build : I used a branch named **sandbox2.**
- Script Path : The directory where the *Jenkinsfile* is stored.
  Example **samples/MortgageApplication/Jenkinsfile**

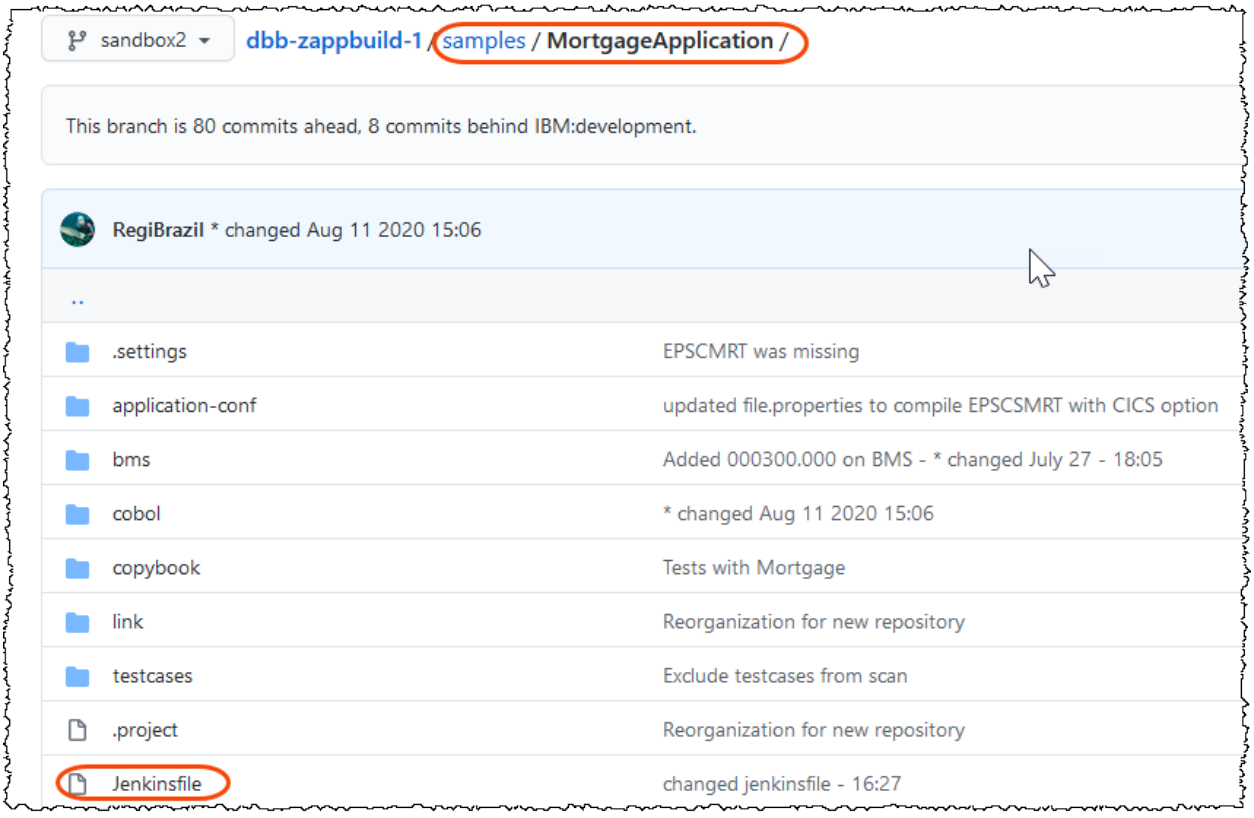
Below my example:"





Note that the Jenkinsfile location is related to the GitHub repository, NOT the local windows client projects..

In my example the “Script Path” for the Jenkinsfile must be **`samples/Mortgage/Application/Jenkinsfile`** .
S
ee the screen capture of my GitHub below:

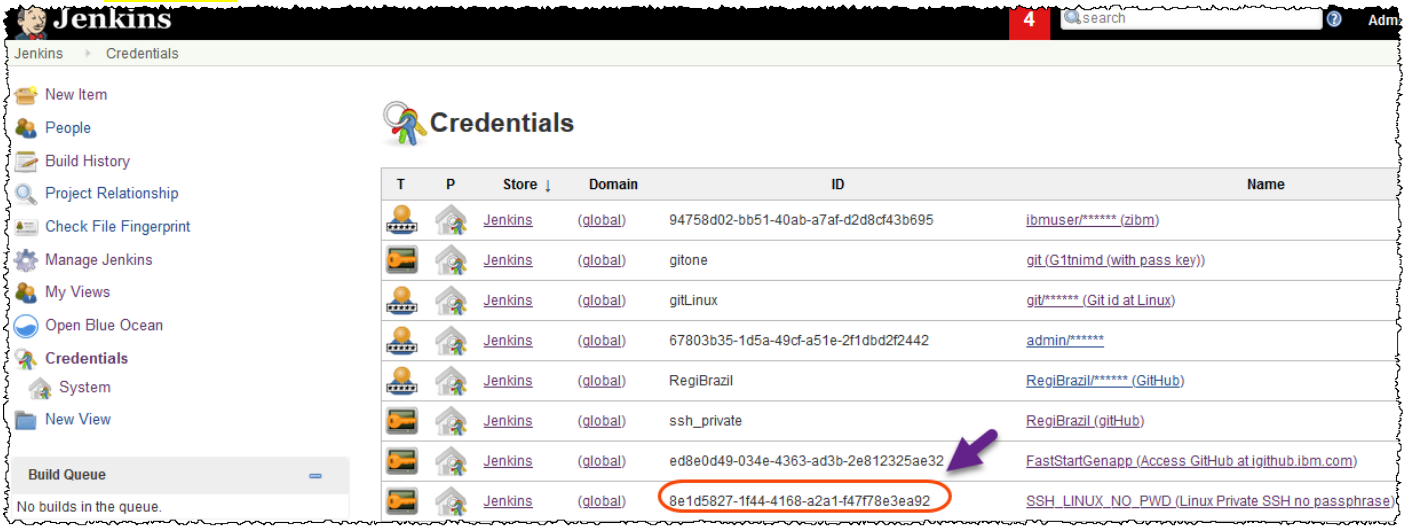


Creating the Jenkinsfile

The easiest way is to create the Jenkinsfile at the project root.

In my example using IDz I created as below:

You might need the credentials id .. In my example this value is below.. I pasted at jenkisfile in the variable `gitCredId`



Here the Jenkinsfile content:
(**master** is the agent on Linux and **zOSSlaveJ** is the agent on z/OS)

```
// Updated by Regi to work with Regi Repo at github.com
// --> Simplified jenkinsfile using sandbox2 branch
// * July 24, 2020  - for Mortgage sample using RegiBrazil
// * without --verbose - see line 16)
// * build type = -i (incremental - see line 41)
// * First time do a "full build" to populate the dbb server
// Global variables
// This is modified to use GitHub and dbb-zappbuild-1/samples/MortgageApplication
    println "**Regi IF First build change to full build"
    println "**Regi IF using VPN the gitHost must be physical IP"
```

```
// --------------------------Agents labels
def linuxAgent = 'master'
def agentLabel = 'zOSSlaveJ'
// --------------------------Verbose
//def verbose = false
def buildVerbose = ''
//def buildVerbose = '-v'
//println "**Regi - Verbose in effect"
// --------------------------Hosts and ports
def linuxHost = '10.1.1.1'
def zosHost = '10.1.1.2'
def zosPort = '22'
// --------------------------DBB
def dbbUrl = 'https://'+linuxHost+':11043/dbb'
def dbbHlq = 'JENKINS'
def dbbDaemonPort = '8080'
def dbbGroovyzOpts= ''
// --------------------------- Git (GitHub)
def gitCredId = '8e1d5827-1f44-4168-a2a1-f47f78e3ea92'
def gitCred = '8e1d5827-1f44-4168-a2a1-f47f78e3ea92'
def gitOrg = 'RegiBrazil'
def srcGitBranch = 'sandbox2'
def gitHost = 'github.com'
def srcGitRepo =     'git@'+gitHost+':'+gitOrg+'/dbb-zappbuild-1.git'
// git@github.com:RegiBrazil/dbb-zappbuild-1.git
// def gitHost = '140.82.114.3'
// --------------------------- Build type
//  -i: incremental
//  -f: full
//  -c: only changed source
def buildType='-i'
// def buildType='-f'
// ----------------------Build properties related to a specific jenkins agent
// def buildConf=''
// UCD definitions - Using V6
def ucdApplication = 'MortgageApplication'
def ucdProcess = 'InstallMortgage'
def ucdComponent = 'JKEMortgageCICS'
def ucdEnv = 'QA'
def ucdBuztool = '/etc/ibm-ucd/v6.2.6/dtsc-agent/bin/buztool.sh'
// --------------------------  Build extra args
//  -d: COBOL debug options
def buildExtraParams='-d'
// --------------------------- Deploy only in case of source code modifications
def needDeploy = true
// ============================================================
pipeline { agent { label linuxAgent }
      environment { WORK_DIR = "${WORKSPACE}/BUILD-${BUILD_NUMBER}" }
      options { skipDefaultCheckout(true) }
```

Remove **//** if wants verbose

See Jenkins Credentials on previous picture

Git Repo

First time you do the build use option **-f**

UCD Definitions

```
// ---------------------------------------------------------------------------
      stages { stage('Init') {steps { script {env.DBB_HOME = '/var/dbb/v.1.0.6'
             echo "Repository: ${srcGitRepo} - branch: ${srcGitBranch} "
       if ( env.ZOS_HOST ) {zosHost = env.ZOS_HOST}
       else {env.ZOS_HOST = zosHost}
       if ( env.ZOS_PORT ) {zosPort = env.ZOS_PORT}
       else {env.ZOS_PORT = zosPort}
       if ( env.BRANCH_NAME != null ) {srcGitBranch = env.BRANCH_NAME; }
         if ( env.DEBUG_PIPELINE && env.DEBUG_PIPELINE == 'true' )
         {verbose = true  buildVerbose = '-v'
       echo sh(script: 'env|sort', returnStdout: true)}
             }}}
// ---------------------------------------------------------------------------
        stage('Git Clone/Refresh') {
            agent { label agentLabel }
            steps {
                script {
                    println "**Regi >branch: ${srcGitBranch}"
                    println "**Regi >WORKSPACE  is ${WORKSPACE}"
                    println "**REGI > Simplified clone **"
 // ---------------------------------------------------------------------
                    dir('dbb-zappbuild-1') {
   checkout([$class: 'GitSCM', branches: [[name: srcGitBranch]],
doGenerateSubmoduleConfigurations: false,
            submoduleCfg: [], userRemoteConfigs: [[url: srcGitRepo]]])
                                  } } } }
// ---------------------------------------------------------------------------
             stage('DBB Build') {
                   steps {
                          script{
                                 node( agentLabel ) {
             if ( dbbDaemonPort != null ) {
             def r = sh script: "netstat | grep ${dbbDaemonPort}", returnStatus: true
             if ( r == 0 ) {
       println "DBB Daemon is running?.."
       bbGroovyzOpts = "-DBB_DAEMON_PORT ${dbbDaemonPort} -DBB_DAEMON_HOST 127.0.0.1"
                         }
             else {
       println "WARNING: DBB Daemon not running build will be longer.."
                                              }
                                       }

 sh "$DBB_HOME/bin/groovyz ${WORKSPACE}/dbb-zappbuild-1/build.groovy --logEncoding
UTF-8 -w ${WORKSPACE} --application MortgageApplication --sourceDir ${WORKSPACE}/dbb-
zappbuild-1/samples  --workDir ${WORKSPACE}/BUILD-${BUILD_NUMBER}  --hlq
${dbbHlq}.MORTGAGE --url $dbbUrl -pw ADMIN $buildType  $buildVerbose
$buildExtraParams "

def files = findFiles(glob: "**BUILD-${BUILD_NUMBER}/buildList.txt")
```

If DBB Daemon is configured

This requires a Pipeline Utility Steps plugin installed

```
 //  ------------------------
// Do not deploy if nothing in the build list
              needDeploy = files.length > 0 && files[0].length > 0
       if (needDeploy) {
          sh "iconv -f ISO8859-1 -t IBM-1047 ${WORKSPACE}/BUILD-
${BUILD_NUMBER}/buildList.txt > ${WORKSPACE}/BUILD-${BUILD_NUMBER}/buildList-
1047.txt"
                                        }
       def files1 = findFiles(glob: "**BUILD-${BUILD_NUMBER}/buildList-1047.txt")
       needTest = files1.length > 0 && files1[0].length > 0

                                    }
                              }
                        }
                        post {
                              always {
                              node( agentLabel ) {
                              dir("${WORKSPACE}/BUILD-${BUILD_NUMBER}") {
                              archiveArtifacts allowEmptyArchive: true,
                              artifacts: '*.log,*.json,*.html',
                              excludes: '*clist',onlyIfSuccessful: false
                                          }
                                    }
                              }
                        }
              }
// ---------------------------------------------------------------------------
        stage('Push to UCD Code station') {
            steps {
                script{
                    if ( needDeploy ) {
                        node( agentLabel ) {
   println "Push to UCD Code station"
   println "**Regi > buztool:   ${ucdBuztool}"
   println "**Regi > Component: ${ucdComponent}"
   sh "$DBB_HOME/bin/groovyz ${WORKSPACE}/dbb-zappbuild-1/utilities/deploy.groovy --
buztool  ${ucdBuztool} --workDir ${WORKSPACE}/BUILD-${BUILD_NUMBER}  --component
${ucdComponent}   "
                                          }
                                    }
                        }
                    }
                                          }
```

If not using UCD
remove all below..

```
// ---------------------------------------------------------------------
        stage('Deploy to CICS using UCD') {
            steps {
                script{
                      echo "Invoke UCD plugin to deploy to CICSTS5.3 - see Appl:
${ucdApplication} "
                    if ( needDeploy ) {
                        node( linuxAgent ) {
                            script{
                                step(
                                  [$class: 'UCDeployPublisher',
                                    deploy: [
                                        deployApp: ucdApplication,
                                        deployDesc: 'Requested from Jenkins',
                                        deployEnv: ucdEnv,
                                        deployOnlyChanged: false,
                                        deployProc: ucdProcess,
                                        deployVersions: ucdComponent + ':latest'],
                                    siteName: 'Urbancode server'])
//  siteName  is the UCD Profile name that is defined at Jenkins configuration
                                }
                                    }
                                }
                    }
                }
}}}
```

This requires the UCD plugin installed

Latest deployed version

The siteName is configured at Jenkins. SeeJenkins Global Tool Configuration

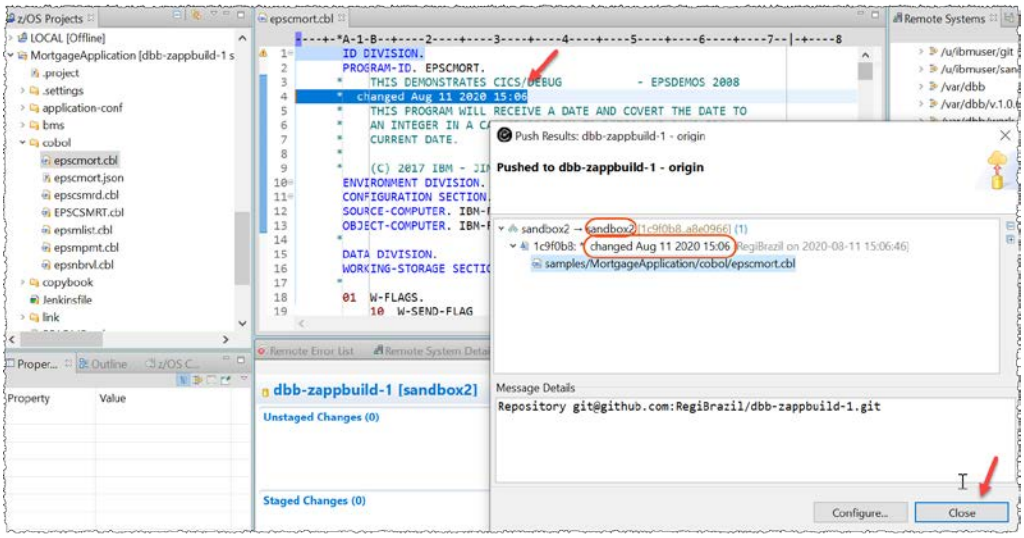
Where to get this sample Jenkins file?

You can get from here:

https://github.com/RegiBrazil/dbb-zappbuild-1/blob/sandbox2/samples/MortgageApplication/Jenkinsfile

Pipeline run example

I changed the program **EPSCMORT** and commit to *G*:



Using jenkins click **Build Now..**

And see the log:

```
tarted by user Admin User
Obtained samples/MortgageApplication/Jenkinsfile from git
git@github.com:RegiBrazil/dbb-zappbuild-1.git
Running in Durability level: MAX_SURVIVABILITY
[Pipeline] Start of Pipeline
[Pipeline] echo
**Regi IF First build change to full build
[Pipeline] echo
**Regi IF using VPN the gitHost must be physical IP
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Init)
[Pipeline] script
[Pipeline] {
[Pipeline] echo
Repository: git@github.com:RegiBrazil/dbb-zappbuild-1.git - branch: sandbox2
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Git Clone/Refresh)
[Pipeline] node
Running on zOSSlaveJ in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] {
[Pipeline] script
[Pipeline] {
```

```
[Pipeline] echo
**Regi >branch: sandbox2
[Pipeline] echo
**Regi >WORKSPACE  is /var/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] echo
**REGI > Simplified clone **
[Pipeline] dir
Running in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/dbb-zappbuild-1
[Pipeline] {
[Pipeline] checkout
No credentials specified
 > /var/dbb/bin/git-jenkins.sh rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
 > /var/dbb/bin/git-jenkins.sh config remote.origin.url
git@github.com:RegiBrazil/dbb-zappbuild-1.git # timeout=10
Fetching upstream changes from git@github.com:RegiBrazil/dbb-zappbuild-1.git
 > /var/dbb/bin/git-jenkins.sh --version # timeout=10
 > /var/dbb/bin/git-jenkins.sh fetch --tags --progress
git@github.com:RegiBrazil/dbb-zappbuild-1.git
+refs/heads/*:refs/remotes/origin/*
Checking out Revision 1c9f0b8823b7d584be99b65d905bf32f94cf2da1
(origin/sandbox2)
 > /var/dbb/bin/git-jenkins.sh rev-parse origin/sandbox2^{commit} #
timeout=10
 > /var/dbb/bin/git-jenkins.sh config core.sparsecheckout # timeout=10
 > /var/dbb/bin/git-jenkins.sh checkout -f
1c9f0b8823b7d584be99b65d905bf32f94cf2da1
Commit message: "*  changed Aug 11 2020 15:06"
 > /var/dbb/bin/git-jenkins.sh rev-list --no-walk
a8e096690544f2b3239f8a2e62299601ce48d4e4 # timeout=10
[Pipeline] }
[Pipeline] // dir
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (DBB Build)
[Pipeline] script
[Pipeline] {
[Pipeline] node
Running on zOSSlaveJ in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] {
[Pipeline] sh
+ netstat
+ grep 8080
DBBS1    000000AF 127.0.0.1..8080        0.0.0.0..0              Listen
[Pipeline] echo
DBB Daemon is running..
```

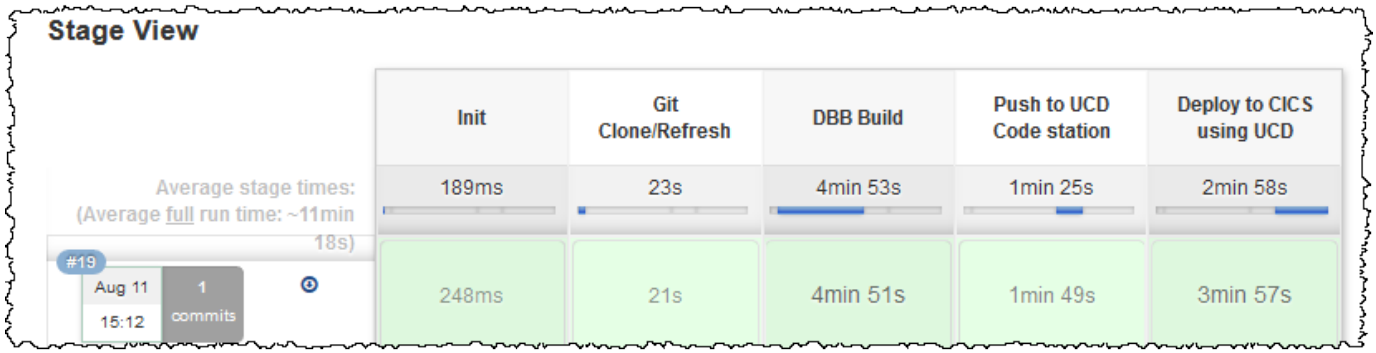
```
+ /var/dbb/v.1.0.6/bin/groovyz /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/dbb-zappbuild-1/build.groovy --logEncoding UTF-8 -w /var/jenkins/workspace/dbb-zappbuild-1-sandbox2 --application MortgageApplication --sourceDir /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/dbb-zappbuild-1/samples --workDir /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19 --hlq JENKINS.MORTGAGE --url https://10.1.1.1:11043/dbb -pw ADMIN -i -d

Cannot contact zOSSlaveJ: java.lang.InterruptedException
** Build start at 20200811.071545.015
** Repository client created for https://10.1.1.1:11043/dbb
** Build output located at /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19
** Build result created for BuildGroup:MortgageApplication-sandbox2 BuildLabel:build.20200811.071545.015 at https://10.1.1.1:11043/dbb/rest/buildResult/8999
** --impactBuild option selected. Building impacted programs for application MortgageApplication
** Writing build list file to /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19/buildList.txt
** Invoking build scripts according to build order: BMS.groovy,Cobol.groovy,LinkEdit.groovy
** Building files mapped to Cobol.groovy script
*** Building file MortgageApplication/cobol/epscmort.cbl
** Writing build report data to /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19/BuildReport.json
** Writing build report to /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19/BuildReport.html
** Build ended at Tue Aug 11 19:17:57 GMT 2020
** Build State : CLEAN
** Total files processed : 1
** Total build time  : 2 minutes, 12.749 seconds


** Build finished
[Pipeline] findFiles
[Pipeline] sh
+ iconv -f ISO8859-1 -t IBM-1047 /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19/buildList.txt
+ 1> /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19/buildList-1047.txt
[Pipeline] findFiles
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // script
Post stage
[Pipeline] node
Running on zOSSlaveJ in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] {
[Pipeline] dir
Running in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19
[Pipeline] {
[Pipeline] archiveArtifacts
```

```
Archiving artifacts
[Pipeline] }
[Pipeline] // dir
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Push to UCD Code station)
[Pipeline] script
[Pipeline] {
[Pipeline] node
Running on zOSSlaveJ in /var/jenkins/workspace/dbb-zappbuild-1-sandbox2
[Pipeline] {
[Pipeline] echo

Push to UCD Code station
[Pipeline] echo
**Regi > buztool:   /etc/ibm-ucd/v6.2.6/dtsc-agent/bin/buztool.sh
[Pipeline] echo
**Regi > Component: JKEMortgageCICS
[Pipeline] sh
+ /var/dbb/v.1.0.6/bin/groovyz /var/jenkins/workspace/dbb-zappbuild-1-
sandbox2/dbb-zappbuild-1/utilities/deploy.groovy --buztool /etc/ibm-
ucd/v6.2.6/dtsc-agent/bin/buztool.sh --workDir /var/jenkins/workspace/dbb-
zappbuild-1-sandbox2/BUILD-19 --component JKEMortgageCICS

** Create version start at 20200811.071909.019
** Properties at startup:
   component -> JKEMortgageCICS
   startTime -> 20200811.071909.019
   workDir -> /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-19
   buztoolPath -> /etc/ibm-ucd/v6.2.6/dtsc-agent/bin/buztool.sh
** Read build report data from /var/jenkins/workspace/dbb-zappbuild-1-
sandbox2/BUILD-19/BuildReport.json
** Find deployable outputs in the build report
   JENKINS.MORTGAGE.OBJ(EPSCMORT), null
   JENKINS.MORTGAGE.DBRM(EPSCMORT), DBRM
   JENKINS.MORTGAGE.LOAD(EPSCMORT), LOAD
** Generate UCD ship list file
** Write ship list file to  /var/jenkins/workspace/dbb-zappbuild-1-
sandbox2/BUILD-19/shiplist.xml

** Create version by running UCD buztool
/etc/ibm-ucd/v6.2.6/dtsc-agent/bin/buztool.sh createzosversion -c
JKEMortgageCICS -s /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-
19/shiplist.xml -o /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-
19/buztool.output
zOS toolkit config   : /etc/ibm-ucd/v6.2.6/dtsc-agent/ (6.2.6,20170906-2200)
zOS toolkit binary   : /etc/ibm-ucd/v6.2.6/dtsc-agent/ (6.2.6,20170906-2200)
zOS toolkit data set : BUZ626 (6.2.6,20170907-0249)
```

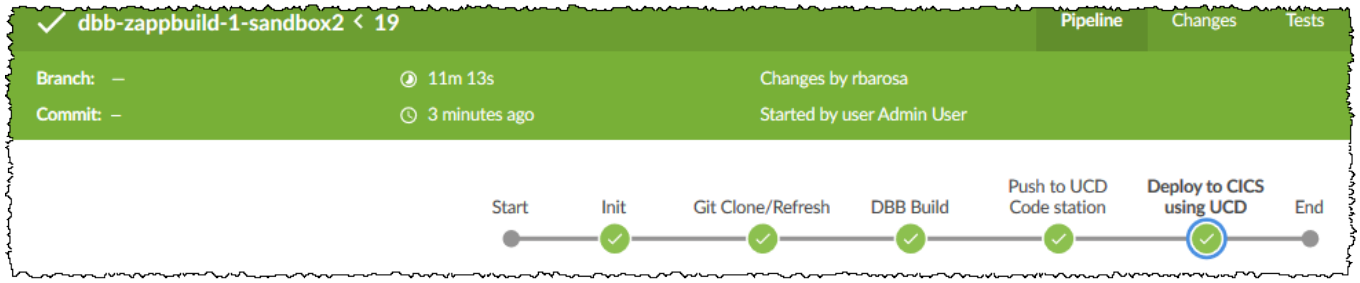
```
Reading parameters:
....Command : createzosversion
....Component : JKEMortgageCICS
....Generate version name : 20200811-191941
....Shiplist file : /var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-
19/shiplist.xml
....Output File:/var/jenkins/workspace/dbb-zappbuild-1-sandbox2/BUILD-
19/buztool.output
Verifying version
....Repository location : /etc/ibm-ucd/v6.2.6/dtsc-
agent/var/repository/JKEMortgageCICS/20200811-191941
Pre-processing shiplist:
....Shiplist after processing :/etc/ibm-ucd/v6.2.6/dtsc-
agent/var/repository/JKEMortgageCICS/20200811-191941/shiplist.xml
Packaging data sets:
....Location to store zip : /etc/ibm-ucd/v6.2.6/dtsc-
agent/var/repository/JKEMortgageCICS/20200811-191941
....Zip name : package.zip
....JENKINS.MORTGAGE.OBJ.bin
....JENKINS.MORTGAGE.DBRM.bin
....JENKINS.MORTGAGE.LOAD.bin
....Elapsed time for data set package or deploy operation : 7.257710
Post-processing package:
PackageManifest file post-processing completed.
Create version and store package:
....Error adding properties to version in UrbanCode Deploy server

** Build finished
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to CICS using UCD)
[Pipeline] script
[Pipeline] {
[Pipeline] echo
Invoke UCD plugin to deploy to CICSTS5.3 - see Appl: MortgageApplication
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/dbb-zappbuild-1-sandbox2@2
[Pipeline] {
[Pipeline] script
[Pipeline] {
[Pipeline] step
Deploying component versions '{JKEMortgageCICS=[latest]}'
Starting deployment process 'InstallMortgage' of application
'MortgageApplication' in environment 'QA'
Deployment request id is: '173def6f-b12e-687b-1705-6c3562cff3e0'
Deployment is running. Waiting for UCD Server feedback.
Finished the deployment in 235 seconds
The deployment result is SUCCEEDED. See the UrbanCode Deploy deployment logs
for details.
[Pipeline] }
[Pipeline] // script
[Pipeline] }
```

```
[Pipeline] // node
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

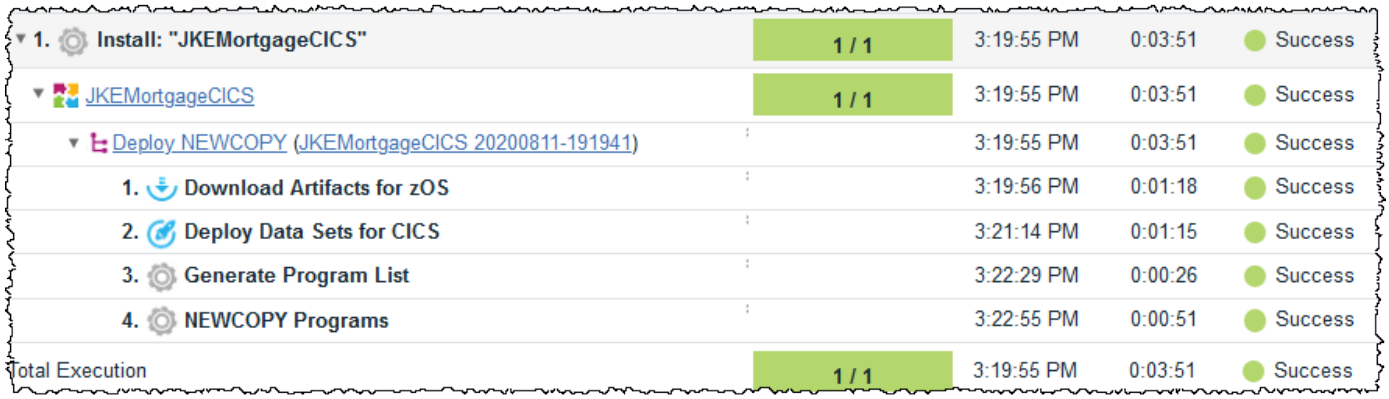
Jenkins format traditional:



Jenkins Ocean format:



UCD deploy to QA output on browser:



Running the application deployed under CICS:

Use transaction EPSP: