**PHIX-HA DEPLOY SCRIPT FUNCTIONING**

Jenkins phix-ha deploy

Only DB deploy

Enter parameters

Get artifact name from parameters

Validation

Parameters,build & directory str.

Structure validation

Stop process

error

if correct

Transfer artifacts & scripts on

server

Backup of application. Stop services as per selection

Deploy the artifact on server.

Update serverversion file with application’s latest information

Apply DB patches if selected in build

Start services

(as per module)

Deployment

done

**Email for startup of job**

Send email to the user on startup of deploy job.

**If only db deploy**

**Stop tomcat**

undeploy existing application

**Db deploy**: take svn export of "$build\_db\_url" from revision number "129783" on $home\_dir/db/$code\_base/$stream\_name location.

then svn export $revision\_number\_core $dbdeploy\_patch\_url\_core

svn export -r $revision\_number $dbdeploy\_patch\_url\_ha

copy $apps\_user@$host\_name:/etc/tomcat7/Catalina/localhost/notification- system.xml to build-db

**DB context file location**

context\_path=$home\_dir/db/$code\_base/$stream\_name/build-db

context\_file=$context\_path/notification-system.xml

update "db.properties" file with db\_url, username , password, patch\_directory\_phix, outputfile\_phix, undoOutputfile\_phix, and changelog.

db\_property\_file=$home\_dir/db/$code\_base/$stream\_name/build-db/db.properties

Apply DB patches.

**Start tomcat**

**Pre deployment Steps**

1. Extract artifact in $artifact\_dir/$code\_base/$stream\_name/${moded\_dir\_name} .

Where moded\_dir\_name=$$ .

1. Go to moded\_dir\_name and Take svn export from $configuration\_url/$host\_name\_config\_folder.
2. For config: copy

$host\_name/batch-jobs.properties config/phix-batch-jobs/properties/ to $mod\_wd

Where $mod\_wd is pwd.

1. Create a tar file of $mod\_wd.

**Validation**

1. Check whether buil exists or not.
2. Validating Directory structure on target machine.

**Transferfiles**

1. Transfering artifacts at target server at location : $target\_upload\_dir/temp
2. Copy $artifact\_dir/$code\_base/$stream\_name/modules\_to\_be\_deploy\_${stream\_name}\_${module\_artifact\_name}\_${revision\_number} to $apps\_user@$host\_name:$target\_upload\_dir/temp/modules\_to\_be\_deploy
3. Copy $template\_file\_path to $apps\_user@$host\_name:$target\_upload\_dir/temp

**Variables:**

* **home\_dir**=/usr/local
* **upload\_dir**=$home\_dir/ha\_phix\_staging
* **upload\_current\_dir**=$upload\_dir/current
* **logs\_dir**=/home/ubuntu/logs
* **artifact\_dir**=$upload\_dir/temp
* **tomcat\_home**=/var/lib/tomcat7
* **build\_file\_path**=$HOME/ServerVersion/server\_version.txt
* **template\_file\_path**=$upload\_current\_dir/build\_info\_template\_generic
* **build\_info\_file**=$upload\_current\_dir/build\_info
* **context\_path**=/etc/tomcat7/Catalina/localhost
* **artifact\_name**=ha\_phix\_portal\_"$stream\_name\_ha"\_"$module\_artifact\_name"\_"$revision\_number\_ha".tar.gz
* **modules\_to\_be\_compile**=$home\_dir/src/HA-PHIX/$code\_base\_ha/$stream\_name\_ha/modules\_to\_be\_deploy\_${stream\_name\_ha}\_${module\_artifact\_name}\_${revision\_number\_ha}
* **installcommand**="mvn -DskipTests -Dmaven.repo.local=/mnt/maven\_m2\_repositories/ha\_phix/.m2/repository clean install -f "
* **installcommand\_batchjob**="mvn -Dmaven.repo.local=/mnt/maven\_m2\_repositories/ha\_phix/.m2/repository clean install -P prod"

**Check modules to be deploy.**

If modules selected are “all”, "batch-jobs", "individual-portal-build", "broker-portal-build", "employee-portal-build", and "employer-portal-build" then keep batch\_jobs\_flag as “yes”.

Check modules selected equal to the modules selected during buil process.

**Backup:** Taking backup of existing artifacts on $host\_name. For example:

* Go to $upload\_dir/backup, if stream name is "HA\_Ph1\_RC\_1.0" and modules selected are "all" and "lifestatus-management-service-build" then copy $apps\_dir/lifestatus-service to $upload\_dir/backup/.
* If module is “all” and "employee-portal-build" then copy $apps\_dir/employee to $upload\_dir/backup/.
* If module is “all” and "employer-portal-build" then copy $apps\_dir/employer to $upload\_dir/backup/.

And so on for rest of the services.

**Undeploy:** Undeploying application from tomcat and Stopping all instances of $host\_name. Removing cache logs repositories templates from employer, employee, broker and individual.

Delete all sevices’s folder from $apps\_dir as per selection of services.

**Deploy**

|  |  |  |
| --- | --- | --- |
| **Modules** | **Source** | **Destination** |
| “all” and “employee-portal-build” | ha-employee-\* | $apps\_dir/employee |
| “all” and “employer-portal-build” | ha-employer-\* | $apps\_dir/employer |
| “all” and “plan-service-rest” | rest-endpoint-\* | $apps\_dir/plan-service |
| “all” and "user-management-service-build" | user-management-rest-endpoint-\* | $apps\_dir/user-service |
| “all” and "small-business-application-build" | small-business-application-rest-\* | $apps\_dir/sb-service-app |
| “all” and "small-business-rest" | small-business-rest-endpoint-\* | $apps\_dir/sb-service |
| “all” and "enrollment-service-build" | enrollment-rest-endpoint-\* | $apps\_dir/enrollment-service |
| “all” and "rate-rest-build" | rate-service-rest-endpoint-\* | $apps\_dir/rate-service |
| “all” and "shopping-service-build" | shopping-service-rest-endpoint-\* | $apps\_dir/shop-service |
| “all” and "guided-shopping-build" | guided-shopping-rest-endpoint-\* | $apps\_dir/guided-shopping-service |
| “all” and "notification-service-build" | notification-system-\* | $apps\_dir/notification-system |
| “all” and "individual-rest" | indv-service-rest-endpoint\* | $apps\_dir/indv-service |
| “all” and "individual-portal-build" | ha-individual-view-\* | $apps\_dir/individual |
| “all” and "admin-build" | ha-admin-view-\* | $apps\_dir/admin |
| “all” and "broker-portal-build" | ha-broker-view-\* | $apps\_dir/broker |
| “all” and "indv-application-build" | indv-application-rest-\* | $apps\_dir/indv-service-app |
| “all” and "provider-search-rest-build" | provider-search-rest-endpoint-\* | $apps\_dir/provider-service-new |
| “all” and "broker-rest-build" | broker-rest-endpoint-\* | $apps\_dir/provider-service-new |
| “all” or "broker-application-build" | broker-application-rest-\* | $apps\_dir/broker-service-app |
| “all” or "individual-portal-build" or "broker-portal-build" or "employee-portal-build" or "employer-portal-build" or "batch-jobs" | config | $apps\_dir/config |
| “all” or "batch-jobs" | batch-jobs-\* | $apps\_dir/batch-jobs |
|  | /usr/share/tomcat7/lib/ojdbc6.jar | $apps\_dir/batch-jobs/lib |

**Post Deployment Steps**

1. **Create build\_info\_file** : update information in serverversion file with the following:

phix\_core\_svn\_url, revision\_number, phix\_ha\_svn\_url, revision\_number\_ha, date and jira ticket number.

If modules selected are “all” update information for all modules, else update as per selected modules.

1. **Update Context file:** Update following informations

context\_file\_url

context\_file\_username

context\_file\_password

batch\_jobs\_file

**Restart:** Tomcat restart required in deployment.

**Troubleshooting:** Mostly error occurs due to selection of wrong parameters. All checks are present in the script to check every condition and display of messages makes it easier to understand.

Further if any changes are coming like adding of any parameter will be mentioned with the usage and requirement.