**Program/Project/Group Deploy/Promotion Form - PRD**

|  |
| --- |
| **Program/Project/Group Requirement(s)** |

This form is to be filled each time a new object or changes to an object is to be deployed on UAT Environment:

**Note: All the below mentioned scripts should be tested in the lower environments.**

|  |  |
| --- | --- |
| Program: |  |
| Group/Project Name: | COGX API |
| Revision/Release version: | Version <> |
| APM ID: |  |
| Code Reviewed Date: | 06/20/2019 |
| Code Approved BY: | Kotari, Rajesh |
| What kinds of objects are being deployed/promoted? | Jobs schedules loading data  Parameter files/tables  UNIX Files  Deployed ETL Jobs  Unix Directories  JAR  Python  Other |
| Service ID: | srcccpcogxapipr |
| Edge Node: | 1.bdpr3r12e3pr.wellpoint.com  2.bdpr3r11e1pr.wellpoint.com  3.bdpr3r11e2pr.wellpoint.com |
| Bit Bucket: | On PRD Server (bdpr3r12e3pr.wellpoint.com/bdpr3r11e1pr.wellpoint.com/bdpr3r11e2pr.wellpoint.com)  **Check Gunicorn status the Gunicorn process ( run the below command):**  sudo /opt/cloudera/parcels/Anaconda/bin/supervisorctl status all  conda install -c conda-forge fuzzywuzzy==0.17.0  conda install -c conda-forge apscheduler==3.6.0  conda install -c conda-forge sqlalchemy==1.3.3  **Edit /etc/supervisor/gunicorn\_supervisor\_cogx\_ini**  Add an entry  **COGX\_ENV=pr**  for environment variable.  **Edit /etc/supervisor/gunicorn\_cogx.conf.py**  **Update imports**  from apscheduler.schedulers.background import BackgroundScheduler  from apscheduler.jobstores.sqlalchemy import SQLAlchemyJobStore  import os  import sys  import glob  sys.path.append("/ua/app/ve2/ccp/cogx/phi/gbd/r000/bin/ds-cogx-api/src-lib/")  from scheduler import cleanup  **Increase the number of workers to 20**  workers = 20  **Configure hooks**  def when\_ready(server):    print('Server started')  sqlite = 'sqlite:///{}.sqlite'.format(os.getpid())  rerun\_monitor = BackgroundScheduler(jobstores={'default': SQLAlchemyJobStore(url=sqlite)})  job = rerun\_monitor.add\_job(cleanup, args=('5'), trigger='cron', hour='00', minute='00', second='1', id='cleanup', replace\_existing=True)  rerun\_monitor.start()  pass  def on\_exit(server):  print('exiting server')  files = glob.glob('\*.sqlite')  for file in files:  os.remove(file)  pass  def post\_worker\_init(worker):  pass  **/opt/cloudera/parcels/Anaconda/bin/supervisorctl update**  **Ensure /opt/cloudera/security/x509/cert.pem exists**  cd /pr/app/ve2/ccp/cogx/phi/gbd/r000/bin/ds-cogx-api  git checkout master  git pull  git checkout cogx\_new\_tag (tag will be provided at a later point and will be documented with the change task) |
| Custom UNIX Directories: | ~~Action (Create/Delete):~~  ~~Script Location:~~  ~~Create directories in the location~~ |
| Custom Unix Installation | ~~export PATH=/opt/cloudera/parcels/Anaconda:/opt/cloudera/parcels/Anaconda/bin:$PATH~~    ~~pip install supervisor~~  ~~Update supervisord.service and replace reference for /pr/app/ve2/pdp/spfi/phi/gbd/r000/bin/api/venv/spfi\_env/bin  with /opt/cloudera/parcels/Anaconda/bin~~ |
| Custom HDFS Directories: | ~~Action (Create/Delete):~~  ~~Script Location:~~  ~~Example: /home/$ServiceID/$REPODIR/createhdfsdir.sh~~ |
| Parameter Files: (bash profile, config) | ~~Action (Add/Replace/Delete):~~  ~~Comments:~~ |
| DDL: | ~~Action (Create/Drop):~~  ~~Script Location:~~  ~~Example: beeline -u 'jdbc:hive2://dwbdprod1r1m.wellpoint.com:2181,dwbdprod1r2m.wellpoint.com:2181,dwbdprod1r3m.wellpoint.com:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2'~~  ~~-f /home/$ServiceID/$REPODIR/ddl/DDL.hql~~  ~~Comments~~**~~:~~** |
| Environment setup: | ~~Script Location:~~   1. ~~.condarc need to be update to point /pr/app/ve2/ccp/cogx/phi/gbd/r000/bin/venv~~ 2. ~~cd /pr/app/ve2/ccp/cogx/phi/gbd/r000/bin~~ 3. ~~mkdir venv~~ 4. ~~git clone path should be executed in bin (as described above in bit bucket block)~~ 5. ~~cd ds-cogx-api~~ 6. ~~git checkout feature/sprint-01~~ 7. ~~cd env~~ 8. ~~conda env create -f environment.yml~~ 9. ~~mkdir -p /pr/app/ve2/ccp/cogx/phi/no\_gbd/r000/logs~~ 10. ~~mkdir -p /pr/app/ve2/ccp/cogx/phi/gbd/r000/logs/gunicorn~~ 11. ~~create suprviror.ini files for cogx project~~ 12. ~~create environment variable COGX\_LOGS~~ 13. ~~Optional test gunicorn through command~~ 14. ~~Configure nginx for reverse proxy~~ |
| Additional Tasks: | **Destroy Kerberos ticket**  kdestroy  **Start the Gunicorn process :**  sudo /opt/cloudera/parcels/Anaconda/bin/supervisorctl restart gunicorn\_cogx |