C3PO Build/Deploy Instructions for c3po webapp

Contents

[Software Installations (required) 1](#_Toc424826680)

[Maven 1](#_Toc424826681)

[Tomcat 2](#_Toc424826682)

[Database 2](#_Toc424826683)

[Web Application (clinical3PO) 2](#_Toc424826684)

[Hadoop Services 2](#_Toc424826685)

[Hadoop related (HDFS) 3](#_Toc424826686)

[Accumulo Installation 4](#_Toc424826687)

[Ganglia Installation 5](#_Toc424826688)

[STEPS TO START ENVIRONMENT ON CLUSTER 7](#_Toc424826689)

**Before the actual steps are listed, it is assumed**

1. Hortonworks hadoop is installed (relevant version).
2. The login used to copy data to hadoop (hdfs) and the login used to start the apache tomcat is same.

Software Installations (required)**.**

1. Install java/jdk, set appropriate variables (JAVA\_HOME, PATH), 1.6 is the least version that is required.
2. Set PATH to have hadoop binaries (so that hadoop related commands can be executed).
3. Install apache maven, for installation instructions look at the MAVEN section.
4. Install apache tomcat, for installation instructions look at the Tomcat section.
5. Requires Firefox browser.
6. CLASSPATH variable should NOT be set (at least anything related to c3po project).

Maven

1. Download apache maven (http://maven.apache.org/download.cgi) , 3.0 is the least version that is required.
2. Extract the binary to a folder, update the PATH variable to include bin directory from the extracted folder. Create an environment variable M2\_HOME which points to the extracted folder.

Tomcat

1. Download the tomcat installer (<http://www.bizdirusa.com/mirrors/apache/tomcat/tomcat-7/v7.0.53/bin/apache-tomcat-7.0.53.tar.gz>)
2. Extract to a folder, update the PATH variable to include bin directory from the extracted folder.
3. Create an environment variable CATALINA\_HOME which points to the extracted folder.
4. catalina.sh start will start the server, catalina.sh stop will stop the server.

Database

1. Create the database.
2. Execute the scripts located in utilities\db-objects. Read Readme.txt to know the order of execution of the \*.sql scripts.
3. The same database entries (username,password…should be entered in settings.xml).

Web Application (clinical3PO)

1. Download the source from git hub(after setting up the relevant ssh keys), git clone [git@github.com:Clinical3PO/Display.git](mailto:git@github.com:Clinical3PO/Display.git)
2. Copy the settings.xml from conf directory (which is under apache maven installation) to /home/<user>/.m2 directory (if .m2 is not there, please create it).
3. Copy the profile segment from README which is under utilities->build->maven (project source) to settings.xml which is present under /home/<user>/.m2
4. Modify the values appropriately to suite the target environment.
5. In project where app, hadoop-services, batch....etc are present, at the root folder, say mvn clean install –DskipTests –P{your profile}
6. Change directory to app, say mvn clean package –P{your profile} (this should be executed under app). The clinical3PO war will be created.
7. Copy the clinical3PO.war (under target) to webapps (under apache tomcat).
8. Start the catalina, catalina.sh start (it is assumed that bin directory of apache tomcat is in PATH).
9. Check logs (under apache tomcat) for any issues.
10. App can be accessed through http://<ec2-ip-address>:8080/c3po/**,** generally it is 8080 port.
11. Now click existing searches and see if page is loading properly (without data it will load)
12. Please do additional steps if hadoop jobs have to be created, (below tasks).

Hadoop Services

1. When any patient id is given for search, application in turn calls a shell script that executes a hadoop job, this project deals with the source code pertaining to this activity.
2. In project where app, hadoop-services, batch....etc are present, at the root folder, say mvn clean install –DskipTests –P{your profile}.
3. Change directory to batch project, cd batch, execute mvn clean install –P{your profile} (this will build batch project which in turn is used by hadoop services)
4. Execute mvn clean assembly:assembly -Dbinary=true –DskipTests –P{your profile} under hadoop-services directory, a binary with a name c3po-hadoop-services-1.0.0-SNAPSHOT-bin.tar is created under target directory.
5. Extract this tar archive under a specified directory, whatever the directory that was given against < clinical3PO.hadoop.shellscripts.dir >in settings.xml, for example below we are saying the relevant scripts (hadoop related) and hadoop services are stored under /home/c3po/c3po-hadoop-scripts.

<!-- Directory where shell scripts related to hadoop search are stored -->

< clinical3PO.hadoop.shellscripts.dir >/home/ c3po / clinical3PO -hadoop-scripts</ clinical3PO.hadoop.shellscripts.dir >

1. So we would say tar –xvf c3po-hadoop-services-1.0.0-SNAPSHOT-bin.tar -C c3po-hadoop-scripts with the assumption that /home/ c3po/ clinical3PO -hadoop-scripts is already created (and this is what is configured in settings.xml)
2. Application when it invokes hadoop scripts, it calls the scripts under this directory.
3. Hadoop logs (application, shell redirection) are under logs directory under whatever is configured against < clinical3PO.logging.file> and < clinical3PO.perfLogging.file> in settings.xml

***–P{your profile} is the name of the profile that should be applied for the current build. If multiple profiles are in settings.xml this should be used, otherwise by default (env-properties) is active.***

***Please follow the README.txt from the folder Display\clinical3PO\utilities\build\maven***

Hadoop related (HDFS)

1. In HDFS whatever the location and filename pertaining to category and observation should be appropriately configured in settings.xml, for example the below says.

They are under /cmctest/concept/concept.txt and /cmctest/observation/observation\_2.txt respectively.

<hadoop.file.conceptFile>/cmctest/concept/concept.txt</hadoop.file.conceptFile> <hadoop.file.observationFile>/cmctest/observation/observation\_2.txt</hadoop.file.observationFile>

1. For hadoop cluster, generate ssh keys and share the same with all the participating nodes. In hadoop configuration file core-site.xml, give the property “fs.default.name” with hdfs://<name node host>:9000. In yarn-site.xml, give the property “yarn.resourcemanager.hostname “ with <resource manager host>, “yarn.resourcemanager.address” with <resource manager host>:8032, “yarn.resourcemanager.scheduler.address” with <resource manager host>:8030, “yarn.resourcemanager.admin.address” with <resource manager host>:8033, “yarn.resourcemanager.resource-tracker.address” with <resource manager host>:8031, “yarn.nodemanager.address” with 0.0.0.0:8041. Specify the hadoop cluster participating nodes in slaves file where datanode, nodemanager daemons will run.

Accumulo Installation

1. Download accumulo-1.5.1-bin.tar.gz from <http://accumulo.apache.org/downloads/> and untar the same.

Create environment variable ACCUMULO\_HOME pointing to the extracted directory.

Download zookeeper-3.4.5.tar.gz from <http://www.us.apache.org/dist/zookeeper/> and untar the same.

Create environment variable ZOOKEEPER\_HOME pointing to the extracted directory.

Edit the environment variable PATH by adding $ACCUMULO\_HOME/bin and $ZOOKEEPER\_HOME/bin.

1. Edit $ ZOOKEEPER\_HOME/conf/zoo.cfg and add parameter maxSessionTimeout=600000 at the end.

Start Zookeeper by the command zkServer.sh start

1. Edit $ACCUMULO\_HOME/bin/bootstrap\_config.sh and go to line 62.

Change the line from if [ "${TYPENAME}" = "native" ]; then to if [ "${TYPENAME}" = "Native" ]; then and run bootstrap\_config.sh. Chose heap as 512MB or 1GB and memory-map type as Native.

1. Edit $ ACCUMULO\_HOME/conf/accumulo-site.xml and give proper values for instance.zookeeper.host with the Zookeeper server and port, instance.zookeeper.timeout to 300s, tserver.client.timeout to 60s, instance.secret with the root user password, trace.token.property.password with the root user password.
2. Edit $ ACCUMULO\_HOME/conf/generic\_logger.xml and comment lines from 22 to 30. Edit $ACCUMULO\_HOME/conf/monitor\_logger.xml and comment lines from 22 to 30. Edit $ACCUMULO\_HOME/conf/monitor and replace localhost by the IP address.

Edit $ACCUMULO\_HOME/conf/accumulo\_env.sh and go to line 55. Change the line from export ACCUMULO\_KILL\_CMD='kill -9 %p' to export ACCUMULO\_KILL\_CMD='sleep 10'.

1. Edit /etc/sysconfig/iptables to add an entry to allow port 50095 and run the command /etc/rc.d/init.d/iptables restart.

Verify port 50095 is accessible for outside traffic by the command telnet <ec2-ip-address> 50095.

Edit /etc/sysctl.conf to add parameter vm.swappiness = 10 at the end and run the command sysctl –p.

Verify the added parameter by the command cat /proc/sys/vm/swappiness.

1. Initialize accumulo by the command accumulo init and start the accumulo daemons by the command $ACCUMULO\_HOME/bin/start-all.sh
2. Start accumulo shell and run the command setauths -s Doc1,Doc2,Doc3,Doc4,Doc5 -u root. Add splits to observation, personid, personobservationid.
3. For accumulo cluster, generate ssh keys and share the same with all the participating nodes. Specify the nodes in master file where master daemon will run. Specify the accumulo cluster participating nodes in slaves, tracers file where tserver, tracer daemons will run.

Ganglia Installation

1. Verify the following dependent packages are available by the command rpm –qa and install the missing packages by the command yum install.

rpm -qa | grep php

rpm -qa | grep httpd

rpm -qa | grep apr

rpm -qa | grep libconfuse

rpm -qa | grep expat

rpm -qa | grep pcre

rpm -qa | grep rrdtool

yum install gcc

yum install apr-devel

yum install rrdtool

1. Ganglia installation is by rpm not from ganglia-3.6.0.tar.gz.

Download the package libconfuse-2.7-4.el6.x86\_64.rpm from <http://pkgs.org/centos-6/epel-x86_64/libconfuse-2.7-4.el6.x86_64.rpm.html> and install the same by the command

rpm –ivh libconfuse-2.7-4.el6.x86\_64.rpm

Download the package ganglia-gmond-modules-python-3.6.0-1.x86\_64.rpm, libganglia-3.6.0-1.x86\_64.rpm, ganglia-gmond-3.6.0-1.x86\_64.rpm, ganglia-gmetad-3.6.0-1.x86\_64.rpm from <http://vuksan.com/centos/RPMS-6/x86_64/> and install the same by the command

rpm -ivh ganglia-gmond-modules-python-3.6.0-1.x86\_64.rpm libganglia-3.6.0-1.x86\_64.rpm ganglia-gmond-3.6.0-1.x86\_64.rpm

rpm -ivh ganglia-gmetad-3.6.0-1.x86\_64.rpm

Verify /etc/ganglia directory is created and the following executables are present

/usr/sbin/gmetad

/usr/sbin/gmond

/usr/bin/gmetric

/usr/bin/gstat

1. Edit /etc/ganglia/gmetad.conf and modify the data source with the cluster name and IP address of gmond servers.

data\_source "c3po cluster" 10.168.49.94

Edit /etc/ganglia/gmond.conf and modify the cluster name.

cluster {

name = "c3po cluster"

1. Start the gmond and gmetad daemons by the command

/etc/rc.d/init.d/gmond start

/etc/rc.d/init.d/gmetad start

1. Download ganglia-web-3.5.12.tar.gz from <http://en.sourceforge.jp/projects/sfnet_ganglia/downloads/ganglia-web/3.5.12/ganglia-web-3.5.12.tar.gz/>.

Login as root user and untar ganglia-web-3.5.12.tar.gz.

Change to directory ganglia-web-3.5.12 and edit Makefile to change macro GDESTDIR = /var/www/html/ganglia and APACHE\_USER = apache. Give APACHE\_USER value as the login under which httpd server is running.

Next run the command make install.

1. Install the following packages if semanage is not installed.

rpm -ivh audit-libs-python-2.2-2.el6.x86\_64.rpm

rpm -ivh libcgroup-0.40.rc1-5.el6\_5.1.x86\_64.rpm

rpm -ivh libselinux-python-2.0.94-5.3.el6\_4.1.x86\_64.rpm

rpm -ivh libsemanage-python-2.0.43-4.2.el6.x86\_64.rpm

rpm -ivh setools-libs-3.3.7-4.el6.x86\_64.rpm

rpm -ivh setools-libs-python-3.3.7-4.el6.x86\_64.rpm

rpm -Uvh policycoreutils-2.0.83-19.39.el6.x86\_64.rpm

rpm -ivh policycoreutils-python-2.0.83-19.39.el6.x86\_64.rpm

Edit iptables to allow 8649 (tcp & udp), 8652.

Run the following commands to allow access for ports 8649, 8651, 8652.

semanage port -a -t http\_port\_t -p tcp 8652

semanage port -a -t http\_port\_t -p tcp 8651

semanage port -a -t http\_port\_t -p tcp 8649

semanage port -a -t http\_port\_t -p udp 8649

setsebool -P httpd\_can\_network\_connect 1

Verify by the command

semanage port -l | grep 80

semanage port -l | grep 8649

semanage port -l | grep 8080

semodule -l | egrep -i "ganglia|gmond|gmetad|gweb"

Run the following commands

/etc/rc.d/init.d/iptables restart

/etc/rc.d/init.d/gmond restart

/etc/rc.d/init.d/gmetad restart

1. Now access the Ganglia web by the <ec2-ip-address>/ganglia and verify the system stats.
2. To monitor Java Virtual Machine statistics run the command

git clone https://github.com/ganglia/jmxetric

mvn clean install

mvn assembly:assembly

cp target/jmxetric-1.0.7-bin.zip /tmp and unzip.

Edit etc/jmxetric.xml and change <!ATTLIST sample dmax CDATA "" > to <!ATTLIST sample dmax CDATA "0" > and finally run the command ./run

1. Now access the Ganglia web by the <ec2-ip-address>/ganglia and verify the JVM stats.

STEPS TO START ENVIRONMENT ON CLUSTER

1. IpAddresses of the machines configured in cluster:
2. In **MASTER:** Stop all the processes starting from Accumulo, Zookeeper, Hadoop.
3. In **SLAVES:** Stop Zookeeper.
4. In **MASTER:** Start Hadoop. Now, '**jps**' on all the 3-machines. You would notice **namenode, datanode, nodemanger, resourcemanager** on **master** and only **datanodes, nodemanager** on both slaves.
5. If that is not the expected output, refer the logs of susheel.
6. If working as per the expectation, start Zookeeper in all the 3-machines.
7. Now, Start Accumulo only on **MASTER.**
8. Now**, 'jps'** on all the 3-machines.
9. To test the cluster status, check hadoop shell commands and accumulo shell commands.
10. Done.