**Directory Structure**

/hydra

/cm // version-control root

/tps // third-party software

/logs // log files

/workarea // workspace for the hydra webapp

/server

/repos // artifacts repository

/<app> // application, ie. wise, planck

/<version> // versions

/config // runtime configuration files

/tomcat // Tomcat installation home

/nodes

/<hostname1> // Tomcat instance 1

/<hostname2> // Tomcat instance 2

**…**

**Installing third-party software**

* **Tomcat**
  + - Download Tomcat version 6.x from <http://tomcat.apache.org/download-60.cgi>
      * Verify the download using md5 checksum.
    - Copy tar.gz over to /hydra/server/ directory
    - Extract gz file.
      * *gunzip -c apache-tomcat-6.0.18.tar.gz | tar xvf –*
    - Create a tomcat link pointing to the extracted directory.
      * *ln -s apache-tomcat-6.0.18 tomcat*
    - (Optional) remove the apache-tomcat-6.0.18.tar.gz installation file.
  + **Ant**
    - Download Ant version 1.7.x from <http://ant.apache.org/bindownload.cgi>
      * Verify the download using md5 checksum
    - Copy tar.gz over to /hydra/tps directory
    - Extract gz file
      * *gunzip -c apache-ant-1.7.1-bin.tar.gz | tar xvf –*
    - Create an ant link pointing to the extracted directory.
      * *ln -s apache-ant-1.7.1 ant*
    - Remove the apache-ant-1.7.1-bin.tar.gz installation file.
  + **Java**
    - Download Java jdk 1.6.x Linux x64 from <http://java.sun.com/javase/downloads/index.jsp>
    - Copy .bin file over to /hydra/tps directory
    - *chmod +x jdk-6u11-linux-x64.bin*
    - Run the self-extracting binary.
      * ./ jdk-6u11-linux-x64.bin
    - Create a java link pointing to the extracted directory.
      * *ln -s jdk1.6.0\_11 java*
    - Remove the jdk-6u11-linux-x64.bin installation file.

**Setup environment**

* Setup user environment. Add the following variable into your .cshrc

*setenv JAVA\_HOME /hydra/tps/java*

*setenv ANT\_HOME /hydra/tps/ant*

*set path = ( $JAVA\_HOME/bin $ANT\_HOME/bin $path )*

* Setup ant environment
  + - create an .antrc file in your home directory with these entries

*ANT\_ARGS="-logger org.apache.tools.ant.NoBannerLogger"*

*ANT\_OPTS="-Xmx256m"*

**Setup git**

* Clone the git repository:

*cd /hydra/cm*

*git clone ssh://shadev1.ipac.caltech.edu/depot/ife*

**Build from source**

* Get the source from git.

|  |  |  |  |
| --- | --- | --- | --- |
| Project | I&T branch name | OPS branch name | Build target |
| Heritage | rc-sha | rel-sha | heritage |
| WISE | rc-wise | rel-wise | wise |
| Planck | rc-planck | rel-planck | planck |
| Finder Chart | rc-fc | rel-fc | fc |

***From here on, I will use Heritage as an example. Replace the heritage’s specific parameters with the ones above when building other projects.***

* + - *cd /hydra/cm*
    - *git pull origin rel-sha*
    - *git checkout rel-sha*
* Ant build
  + - *cd ife/common/java; ant all; cd -*
    - *cd ife/firefly/java; ant heritage; cd –*

**Tomcat Servers setup**

* + Initial setup
    - *cd /hydra/cm/ife/firefly/conf*
    - *edit env.properties with properties for this environment*
    - *ant all*
  + Setup a Tomcat instance
    - Using the host name, setup a Tomcat directory structure under the /hydra/server/nodes directory.
      * Should have these sub-directories.

conf/ logs/ temp/ webapps/ work/

*cd /hydra/server/nodes*

*mkdir <hostname>*

*cd <hostname>*

*cp -r ../../tomcat/conf .*

*cp -r ../../tomcat/webapps .*

*mkdir /hydra/logs/<hostname>*

*ln -s /hydra/logs/<hostname> logs*

*mkdir temp work*

* + - Edit conf/server.xml
      * Replace line:

*<Engine name="Catalina" defaultHost="localhost">*

with line:

*<Engine name="Catalina" defaultHost="localhost"* ***jvmRoute="<hostname>"****>*

* + - Create startup script
      * Login to *<hostname>*
      * cp ife/firefly/config/tomcat\_init /etc/init.d
      * edit /etc/init.d/tomcat\_init to point CATALINA\_BASE to *<hostname>*
      * change TOMCAT\_OWNER if needed.
    - Add the service
      * */sbin/chkconfig --add tomcat\_init*
    - Confirm it is added
      * */sbin/chkconfig --list tomcat\_init*

*>> tomcat\_init 0:off 1:off 2:off 3:on 4:on 5:on 6:off*

* + - Confirm it has execute privilege
    - Test setup
      * */sbin/service tomcat\_init start*
      * point browser to http://*<hostname>*.ipac.caltech.edu:8080/
        + should see Tomcat’s welcome page.
* Repeat “Setup Tomcat instance” steps for each additional tomcat servers. Replace *<hostname>* with the server hostname.

**Apache Server setup**

* + Copy config file into /etc/httpd/conf.d/ directory
    - *cp /hydra/cm/ife/firefly/config/apache/ops\_proxy.conf /etc/httpd/conf.d/*
  + Edit ops\_proxy.conf
    - Make sure hostname is correct. Add/remove sections as needed.
  + Start up apache
    - /sbin/service httpd start
  + Verify that apache is setup to startup upon reboot. If not, do so.
    - /sbin/chkconfig --level 345 httpd on

**MySQL Server setup**

* + Start up MySQL
    - /sbin/service mysqld start
  + Verify that apache is setup to startup upon reboot. If not, do so.
    - /sbin/chkconfig --level 345 mysqld on
  + Setup User and Schema
    - Content from ife/firefly/db/heritage/MySQL\_setup.txt

# set a password for root.

$ mysqladmin -u root password <NEWPASSWORD>

# use the mysql program to connect to the server as the MySQL root user

$ mysql --user=root -p

# creating the user\_management schema

mysql> CREATE SCHEMA user\_management;

mysql> USE user\_management;

execute all of the CREATE statements in the ife/firefly/db/heritage/UserManagementDB.sql file.

this will create the tables needed.

# create a local user

mysql> CREATE USER '<user\_id>'@'localhost' IDENTIFIED BY '<user\_psswd>';

mysql> GRANT ALL PRIVILEGES ON `user\_management`.\* TO '<user\_id>'@'localhost';

# allow the user to connect from any host; you may narrow the hosts by, ie. %.ipac.caltech.edu.

mysql> CREATE USER '<user\_id>'@'%' IDENTIFIED BY '<user\_psswd>';

mysql> GRANT ALL PRIVILEGES ON `user\_management`.\* TO '<user\_id>'@'%';

**Build Hydra**

* Using luntbuild build-server
  + - [*http://shadev1:8888/luntbuild/*](http://shadev1:8888/luntbuild/)
    - *admin username, password – luntbuild, TB3QZluigiX^7*
    - *Lunt build artifact directory: /hydra/server/luntbuild/publish*
    - *Note- only when doing a pre-tagged build:*
      * *Go to project’s VCS adaptor*
      * *Login to admin (luntbuild)*
      * *Change label to be the tag for the git repository*
      * *Change back after build*
      * *When you click on the build, Change build condition in schedules to always*
    - *Anytime we doing ops or test:*
      * *Go to project page*
      * *Edit the variables*
      * *Modify the version variables, revPart for test, minorPart for ops*
      * *Change back after build*
      * *Note- lunt version variables are passed to and override the env\*.properties variables*
    - *Best to access via web, build history link*
    - *run the desire schedule*
    - *copy the build artifacts to /hydra/server/repos*
      * *the ops and test machines do not let you pull files*
      * *best to push from shadev1 to the machine*
      * *follow directory naming convention for consistency*

**Deploy**

* Run deploy script
  + - Log into server box 1.. etct shadev1
    - cd /hydra/server/repos
    - *verify that <project>/current is pointing to the version you wise to deploy*
    - *run the hydra\_deploy.sh script*
      * *./hydra\_deploy.sh <project>*
      * *This script does the following*
        + Stop tomcat on both servers shadev1 and shadev2
        + copy war file into the tomcat’s deploy directory
        + replace the configuration files under /hydra/server/config
        + clear cache
        + start up tomcat on both servers.