Building a TomEE cluster using Docker

1. Setup Docker
   1. Go to <https://www.docker.com/> and follow the steps to install Docker at your OS
   2. The Docker download is at:

<https://www.docker.com/products/docker-toolbox>

* 1. If you are not used to Docker, I recommend you to follow its first tutorial. You will spend half an hour to understand its general features
  2. Once you are ready, its time to build our TomEE container.

1. Build a TomEE container with Docker
   1. Go to <https://hub.docker.com/>
   2. On “Search” textbox, write “tomee” and press Enter
   3. Click in the result called “tomee official”
   4. In the next page you will find some informations about this image
   5. Open de “Docker Quickstart Terminal” (in Mac OS X, or its equivalents in others OS’s)
   6. Once the terminal is opened, check the message where says:

“docker is configured to use the default machine with IP...”

* 1. Take a note of this IP, you will need it to access the TomEE running in your container
  2. Execute this command:

docker run -it --rm -p 8080:8080 tomee:8-jre-1.7.2-webprofile

* 1. Hey! Don’t forget the “-p 8080:8080” above. You are telling the Docker to expose this port, otherwise you won’t be able to access it from your host.

1. Deploy your app at your container
   1. Stop your running TomEE (CTRL + C), choose a folder and run this command:

mkdir mytomeedocker

* 1. Go to your this new folder and create a Dockerfile

touch Dockerfile

* 1. Open the file

open -e Dockerfile

* 1. Write those lines

FROM tomee:8-jre-1.7.2-webprofile

ADD <app>.war /usr/local/tomee/webapps/<app>.war

* 1. Change the <app> to the very name of you .war file
  2. IMPORTANT: Put your .war file at the same folder as Dockerfile
  3. Save your Dockerfile and close it
  4. Check if your lines were saved:

cat Dockerfile

* 1. Now lets build your image! Run this command:

docker build -t tomee-war .

* 1. IMPORTANT: do not forget the dot (“.”) in the end of command. It won’t work if you do
  2. Wow! Now you have your own Docker image, with your own application. Let’s run it:

docker run -it --rm -p 8080:8080 tomee-war

* 1. Now let’s configure the TomEE to run in cluster. Edit the file:

/usr/local/tomee/conf/server.xml

Add those lines (at Engine node):

<Cluster className=*"org.apache.catalina.ha.tcp.SimpleTcpCluster"*

channelSendOptions=*"6"*>

<Manager className=*"org.apache.catalina.ha.session.BackupManager"*

expireSessionsOnShutdown=*"false"* notifyListenersOnReplication=*"true"*

mapSendOptions=*"6"* />

<!-- <Manager className="org.apache.catalina.ha.session.DeltaManager" expireSessionsOnShutdown="false"

notifyListenersOnReplication="true"/> -->

<Channel className=*"org.apache.catalina.tribes.group.GroupChannel"*>

<Membership className=*"org.apache.catalina.tribes.membership.McastService"*

address=*"228.0.0.4"* port=*"45564"* frequency=*"500"* dropTime=*"3000"* />

<Receiver className=*"org.apache.catalina.tribes.transport.nio.NioReceiver"*

address=*"auto"* port=*"5000"* selectorTimeout=*"100"* maxThreads=*"6"* />

<Sender className=*"org.apache.catalina.tribes.transport.ReplicationTransmitter"*>

<Transport

className=*"org.apache.catalina.tribes.transport.nio.PooledParallelSender"* />

</Sender>

<Interceptor

className=*"org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"* />

<Interceptor

className=*"org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor"* />

<Interceptor

className=*"org.apache.catalina.tribes.group.interceptors.ThroughputInterceptor"* />

</Channel>

<Valve className=*"org.apache.catalina.ha.tcp.ReplicationValve"*

filter=*".\*\.gif|.\*\.js|.\*\.jpeg|.\*\.jpg|.\*\.png|.\*\.htm|.\*\.html|.\*\.css|.\*\.txt"* />

<Deployer className=*"org.apache.catalina.ha.deploy.FarmWarDeployer"*

tempDir=*"/tmp/war-temp/"* deployDir=*"/tmp/war-deploy/"* watchDir=*"/tmp/war-listen/"*

watchEnabled=*"false"* />

<ClusterListener

className=*"org.apache.catalina.ha.session.ClusterSessionListener"* />

</Cluster>

* 1. Restart TomEE and there you are!
  2. (To be continued...)