Edit and Build Docker image by Eclipse Docker Plugin and run it on remote Docker Daemon hosted on Amazon AWS

[](http://blog.arungupta.me/wp-content/uploads/2014/03/docker-logo.png)[](http://blog.arungupta.me/wp-content/uploads/2014/03/eclipse-logo.png)

Suprakash Bhowmik

Abstract –

Docker tooling (Eclipse plugin for Docker) provides easy start/stop and deploy docker container from Eclipse. It also provides better overview and easier/faster access to common operations, from a visual perspective. The Docker tooling is aimed at providing at minimum the same basic level features as the command-line interface, but also provide some advantages by having access to a full fledged UI. This article describes to do the developemnt and run docker container by using Eclipse. – **Source: eclipse.org**

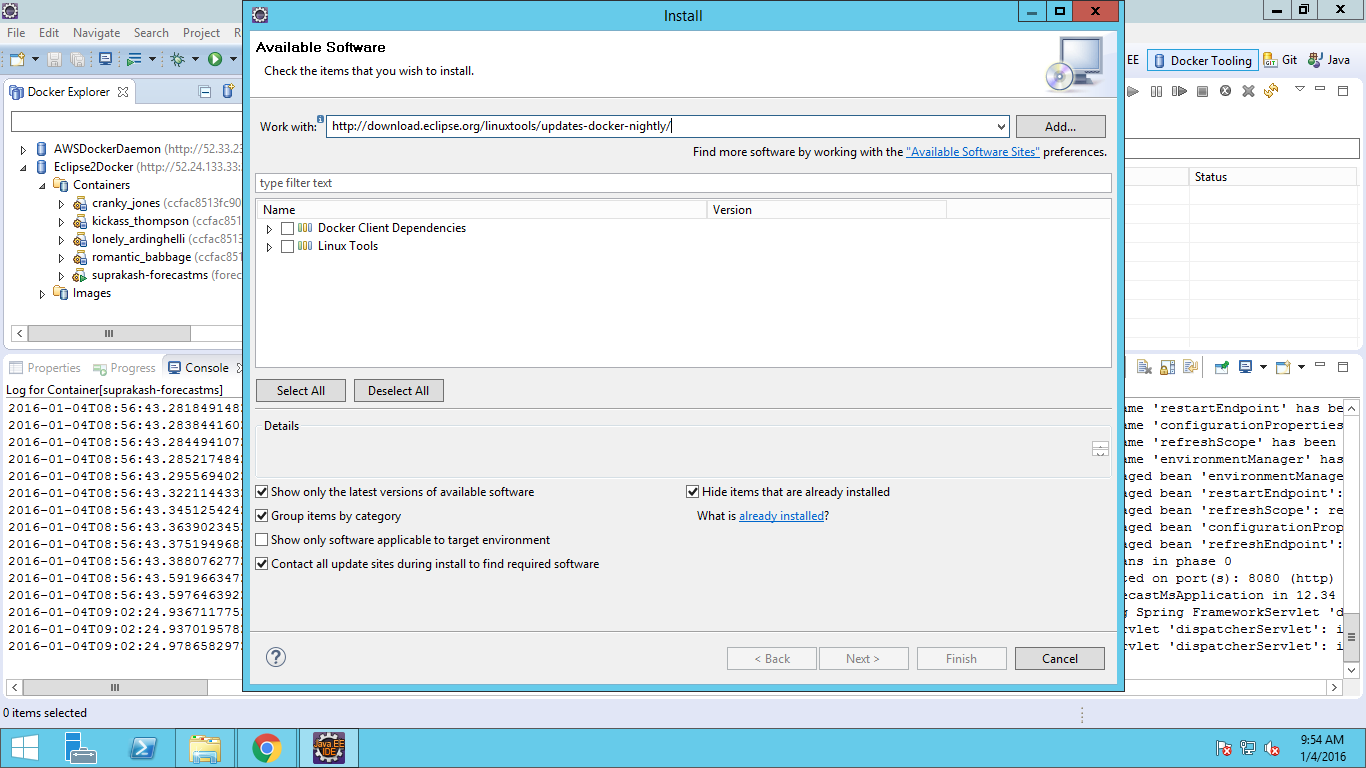
This Tech Note contains –

* How to Install Docker Tool Plugin in Eclipse
* Connect with Remote Docker Host
* Views and Perspective of Docker Plugin
* Build Docker Image from Eclipse
* Run Docker Container
* Test Application
* Benefits
* Limitation
* Conclusion

How to Install Docker Tool Plugin

You can get the latest and greatest build of this plugin by this below links, go to Eclipse (Mars)-> Help -> Install New Software and enter this below url to install:

<http://download.eclipse.org/linuxtools/updates-docker-nightly/>



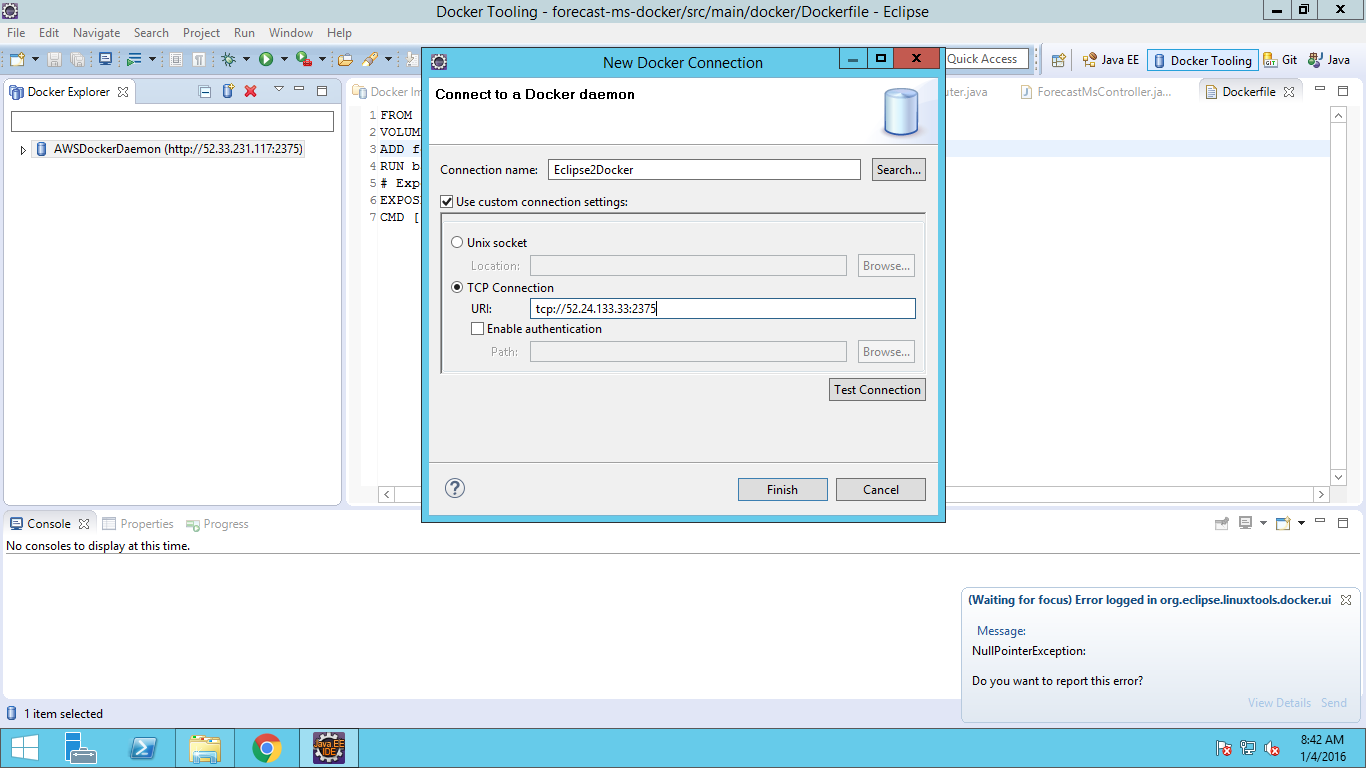
Connect with Remote Docker Host

Go to Docker Tooling Perspective and create a connection to connect with Docker Daemon. In this POC, Docker Host running on Cloud (Amazon ec2 Linux instance).

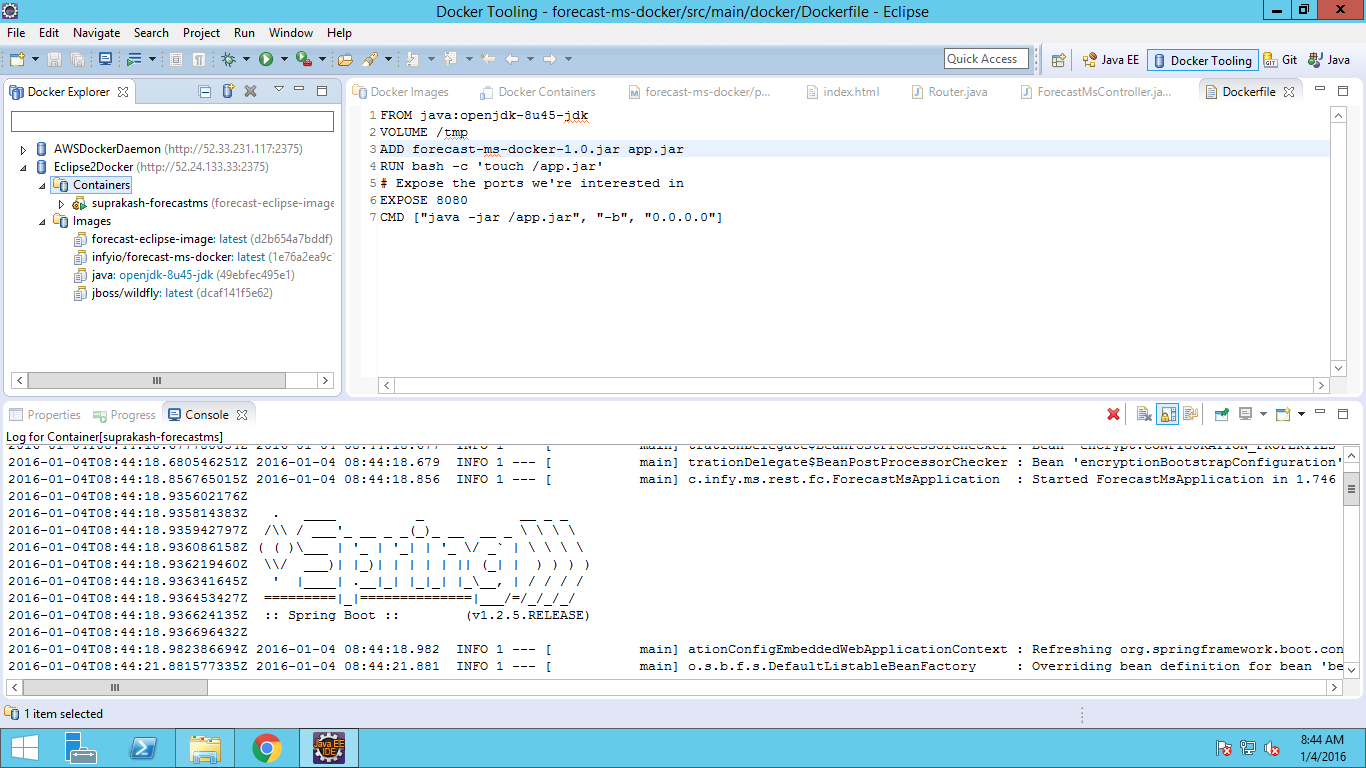
To make sure your docker host/daemon is listent to tcp port, execute this below command on Docker host.

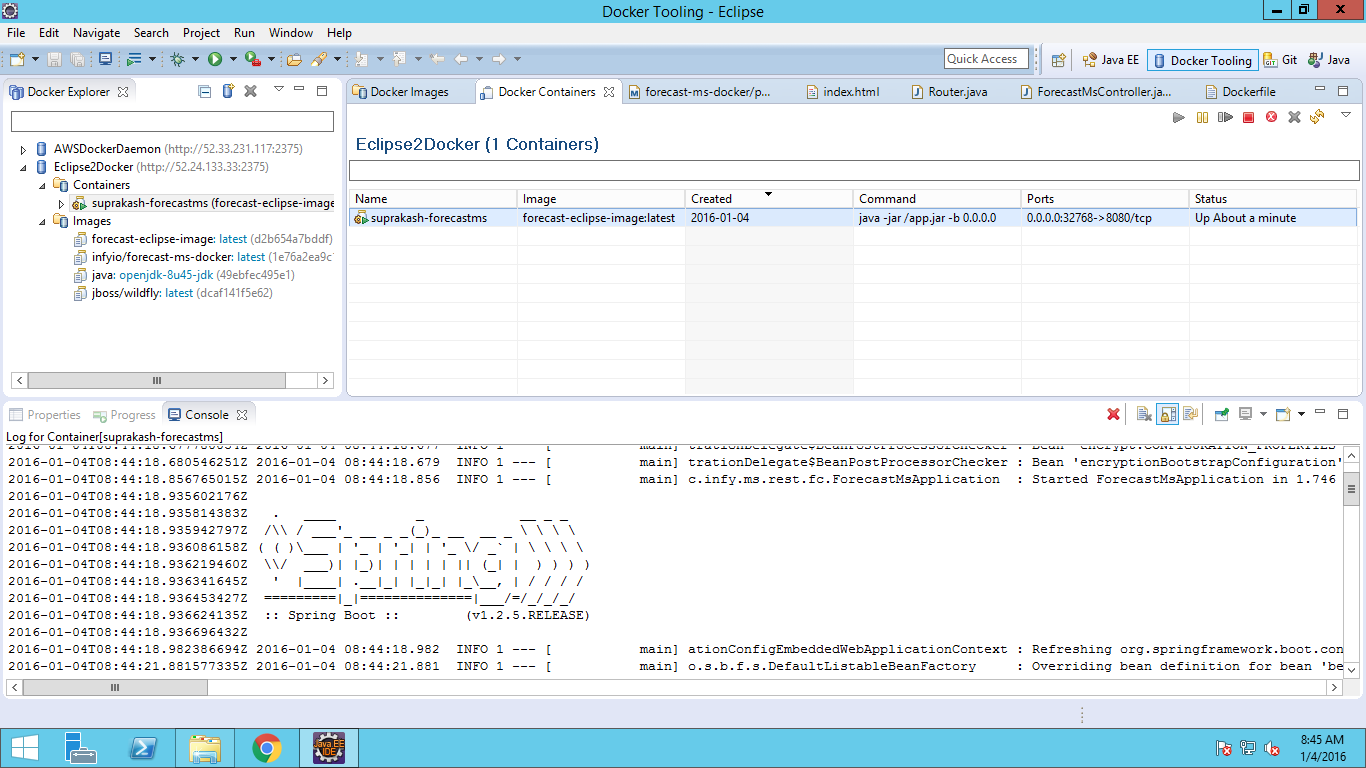
sudo docker -H 0.0.0.0:2375 -d &

In Eclipe, specify the setting shown below to create connection with docker host. If you enable authentication, create all the certificate from docker host machine and copy those certificate to Eclipse env and specify path in ‘Enable Authentication’.

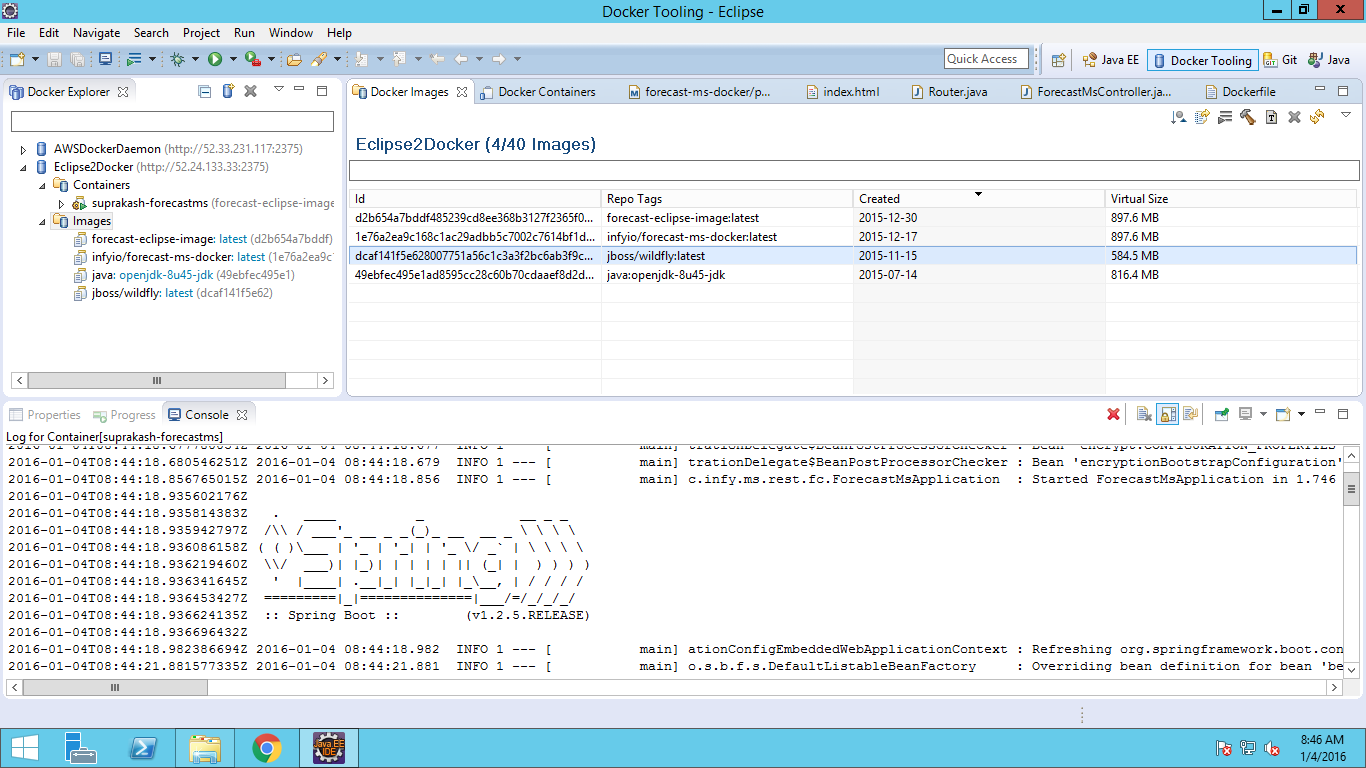


After successful connection, it shows overview of the existing images and container running on Docker Host machine.

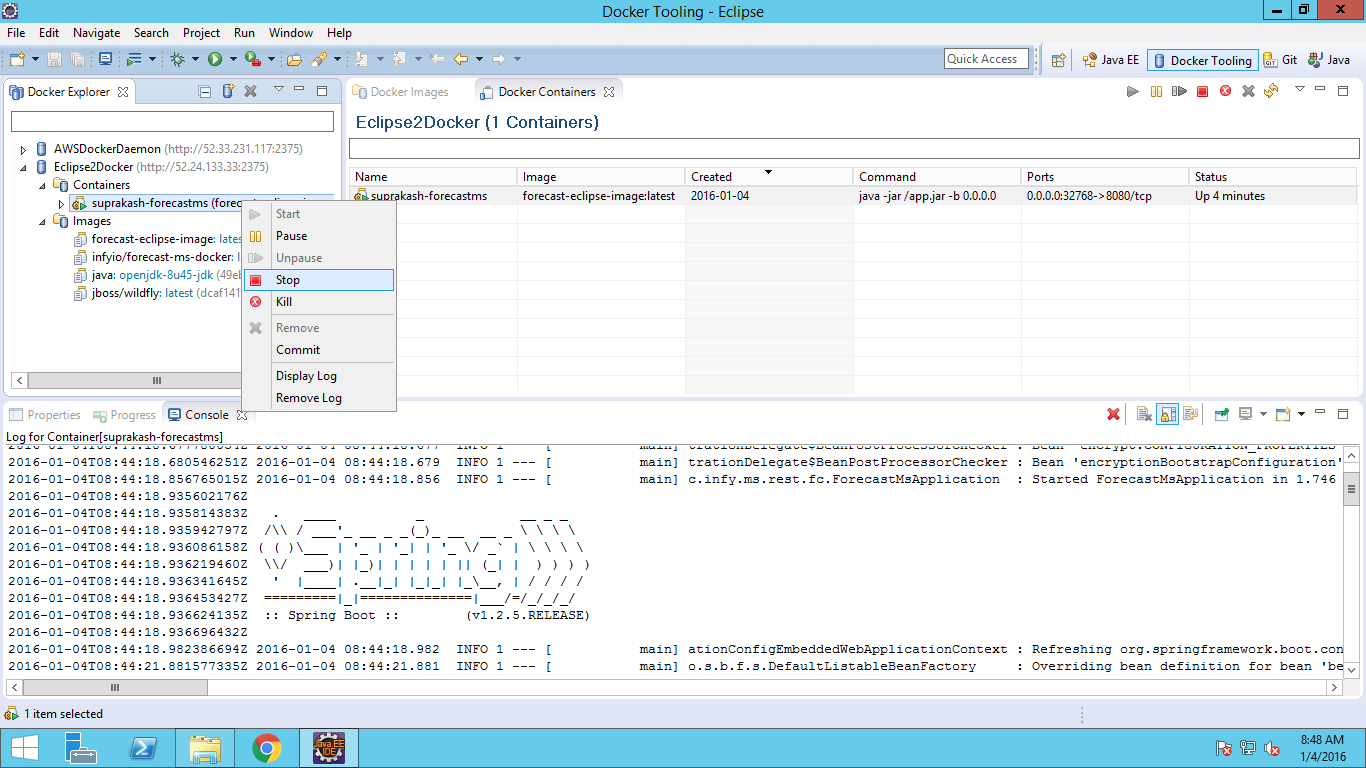


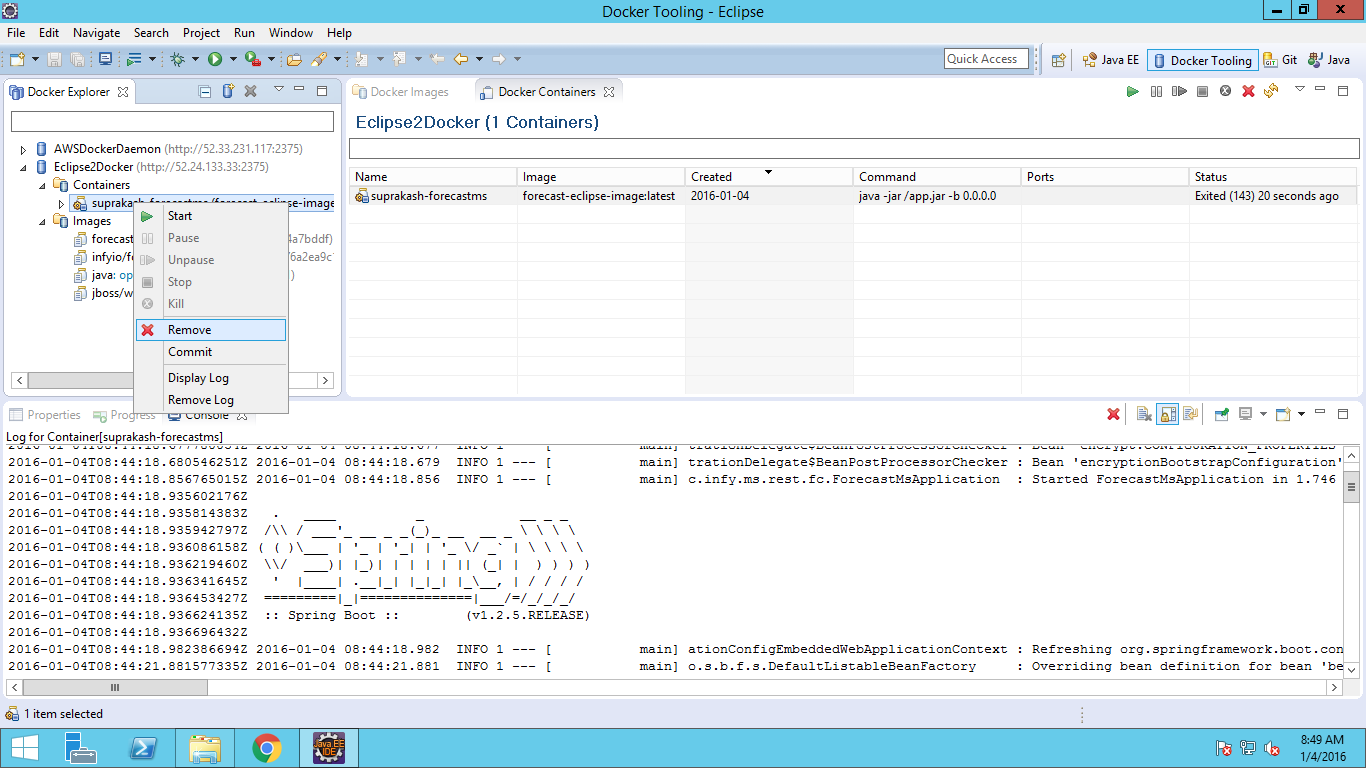


View and Perspective

The Docker Images view lists all images in the Docker host selected in the Docker Explorer view. This view allows user to manage images, including push/pull, build image from DockerProperty and also create container.

Docker Containers view lets the user manage the containers. The view toolbar provides commands to start, stop, pause, unpause, display the logs and also command to kill the containers.



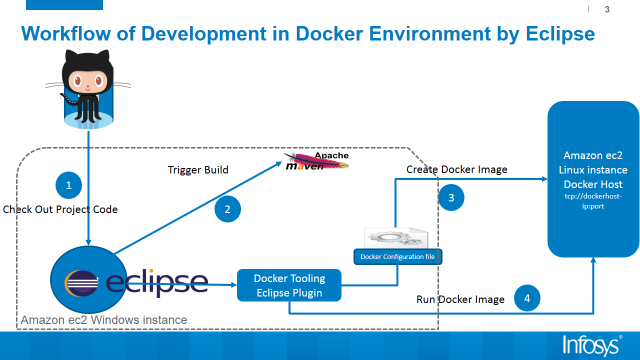


Build Docker Image from Eclipse

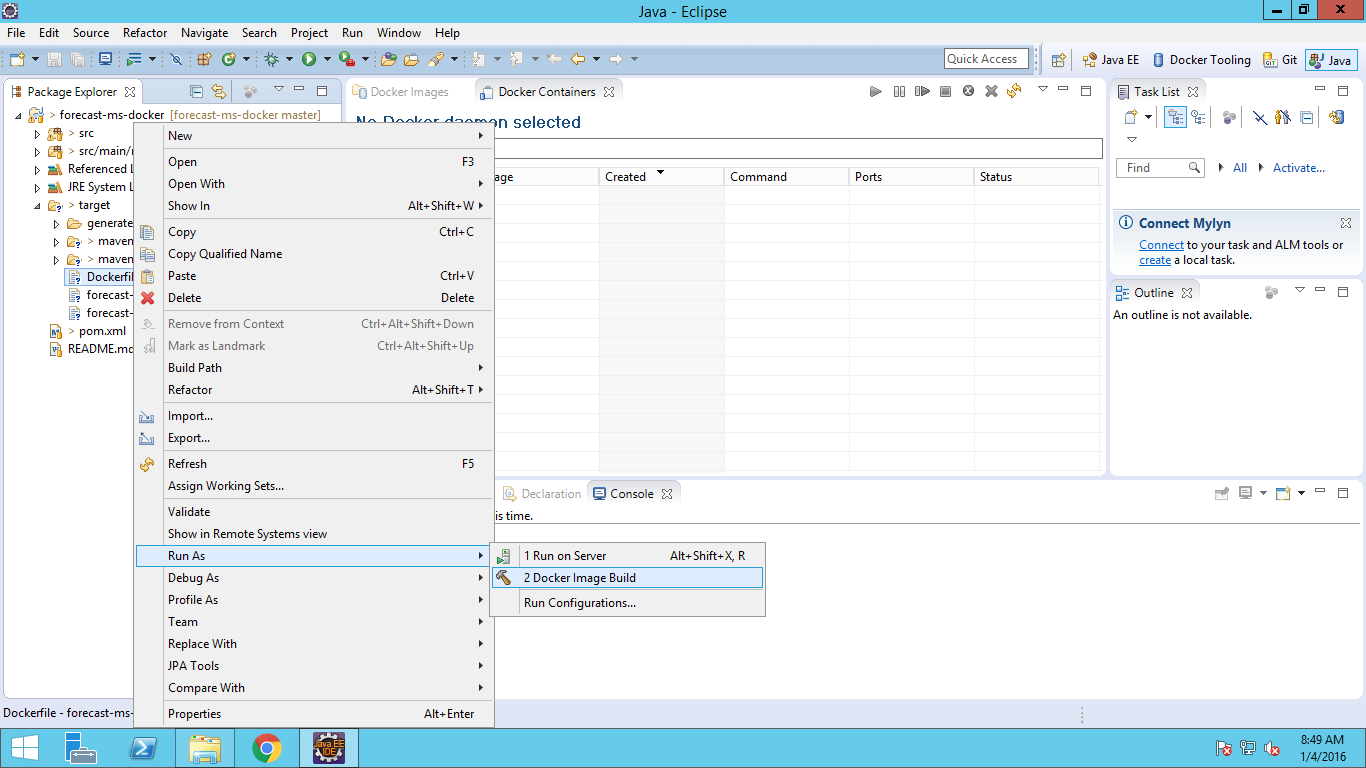
This below reference architecture can be follow to do the development in Docker Environment by using Eclipse:

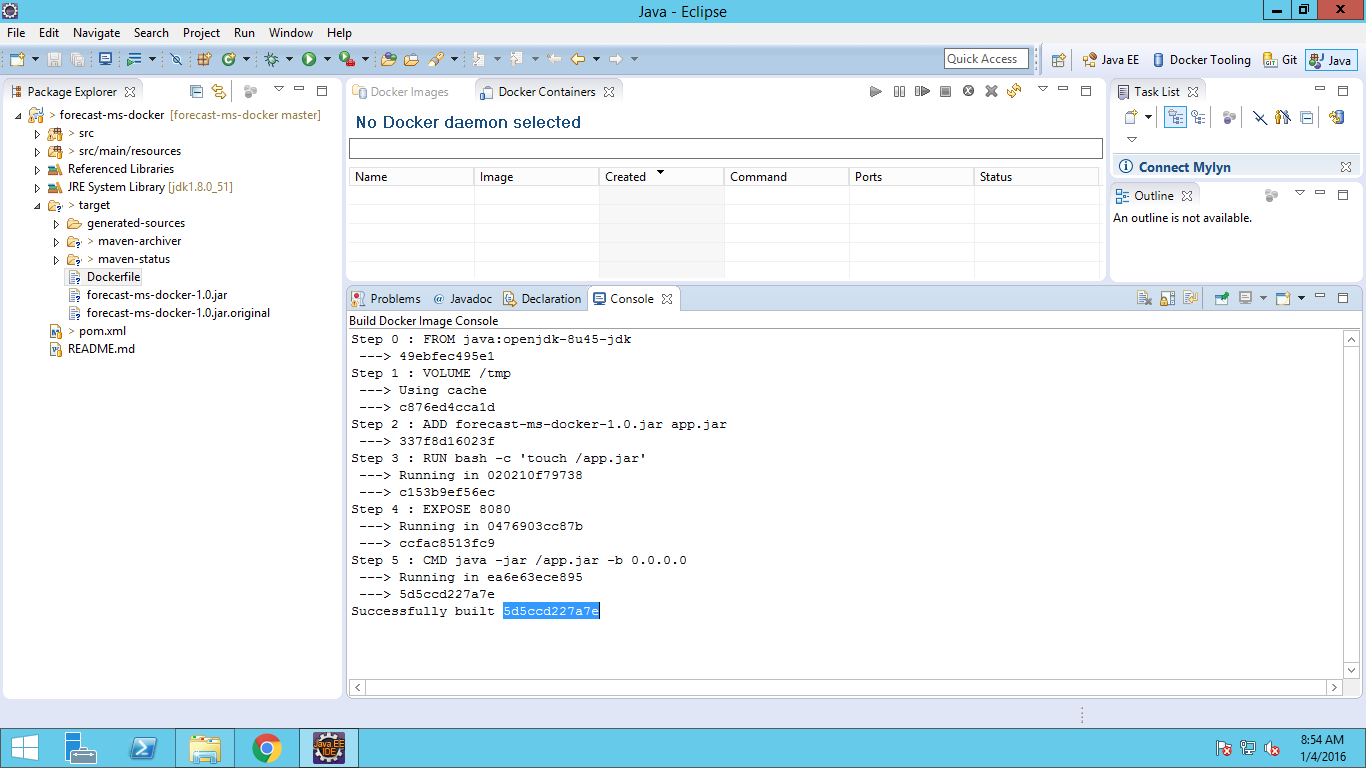
* In one machine/vm (windows) install eclipse (with Docker plugin), maven, java
* In other machine (preferable linux) install docker.
* Connect eclipse with Docker Daemon (installed on linux environment) by Docker plugin
* Check out the code from Github in Eclipse
* Build docker image from DockerProperty file.
* Run the docker image

The architecture workflow can be seen in below diagram



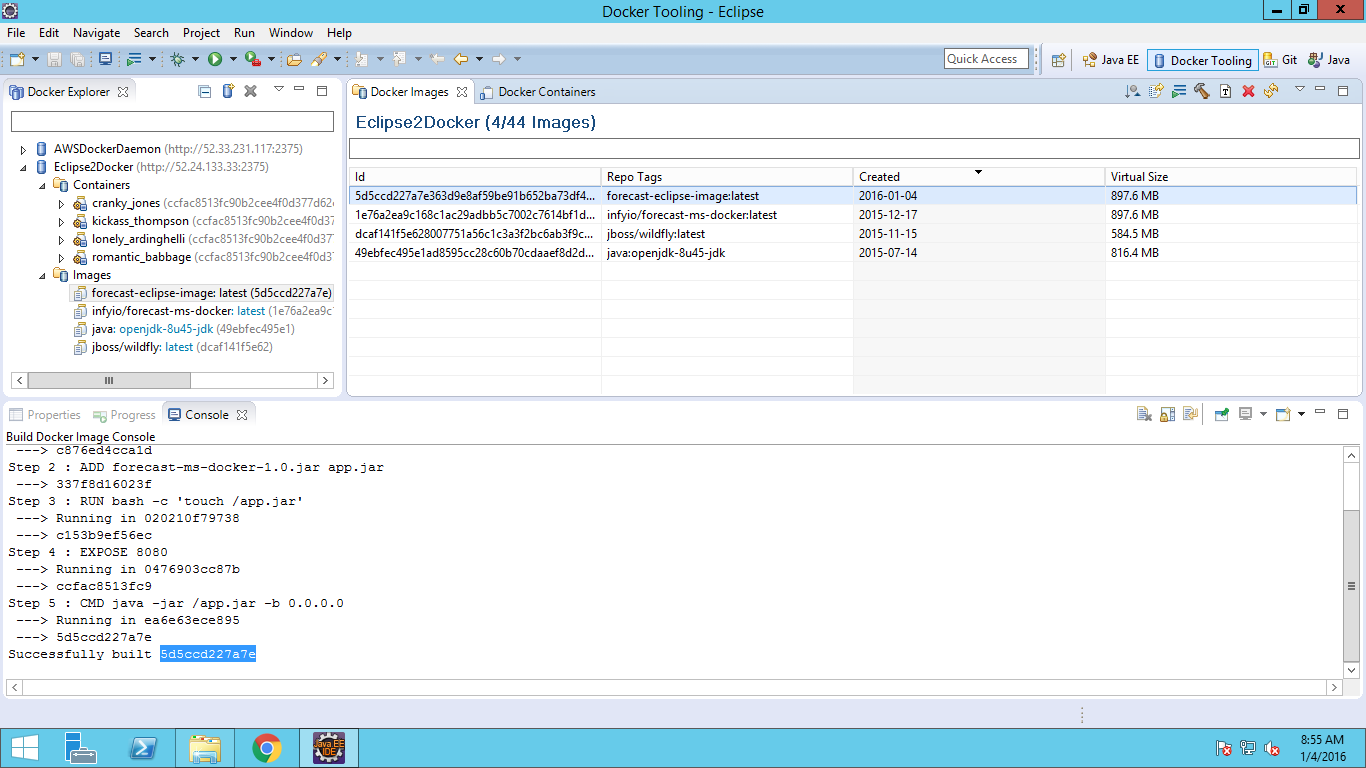
To create docker image by using Docker Plugin, right click on Dockerfile and Run As Docker Image build, as show in below image. You can also show log in console.



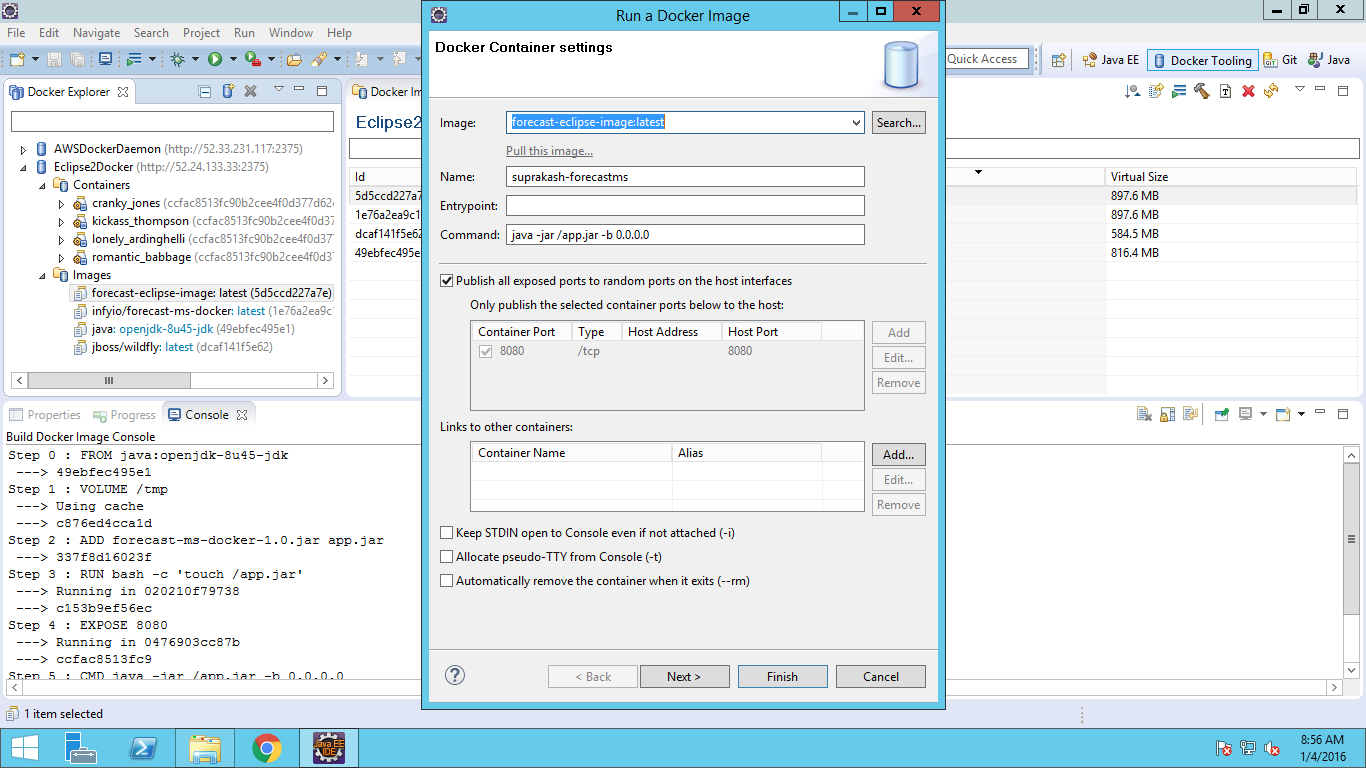


Run Docker Container from Eclipse

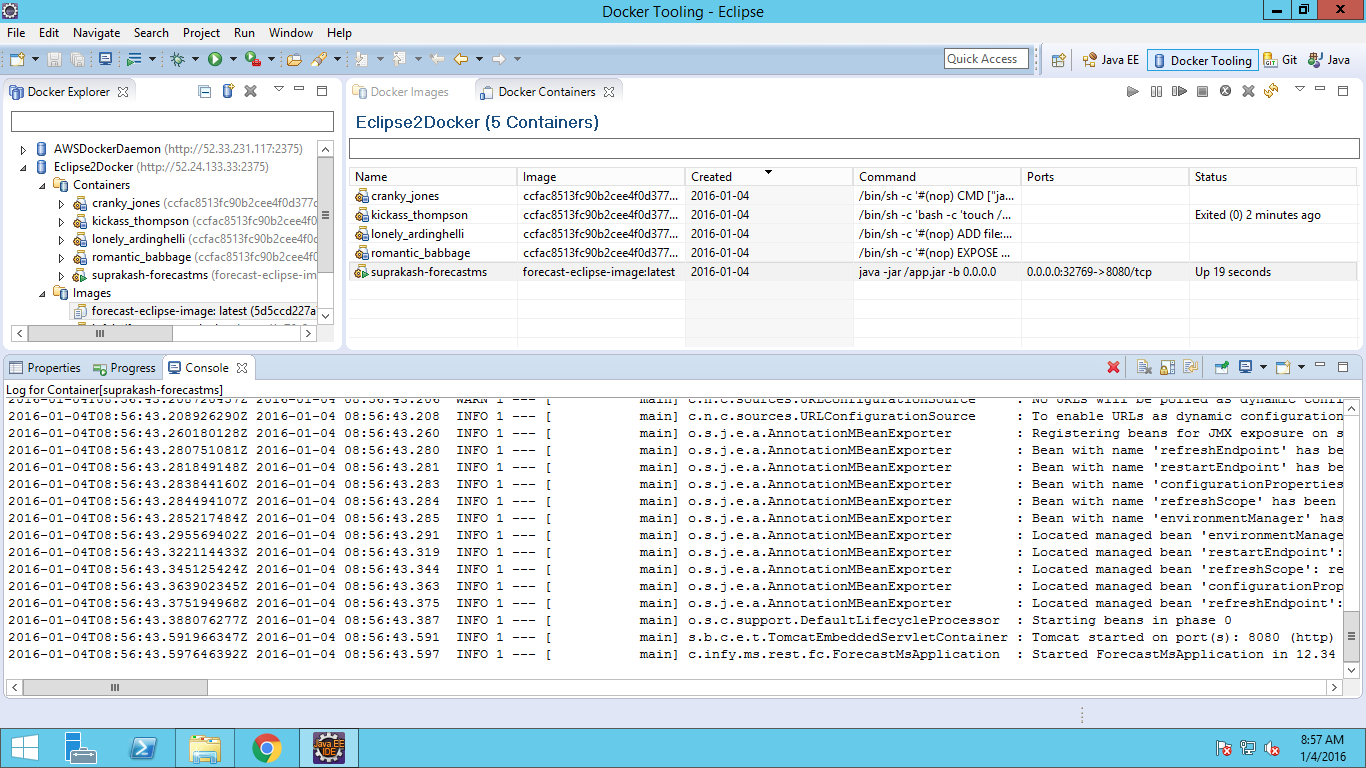
Once build is successful, the new image will show up and ready to use for running.



To run the image, right click on docker image and run. By default, all exports ports from the image are mapped to random ports on the host interface. This setting can be changed by unselecting the first checkbox and specify the port mapping. Click on Finish to start the container.



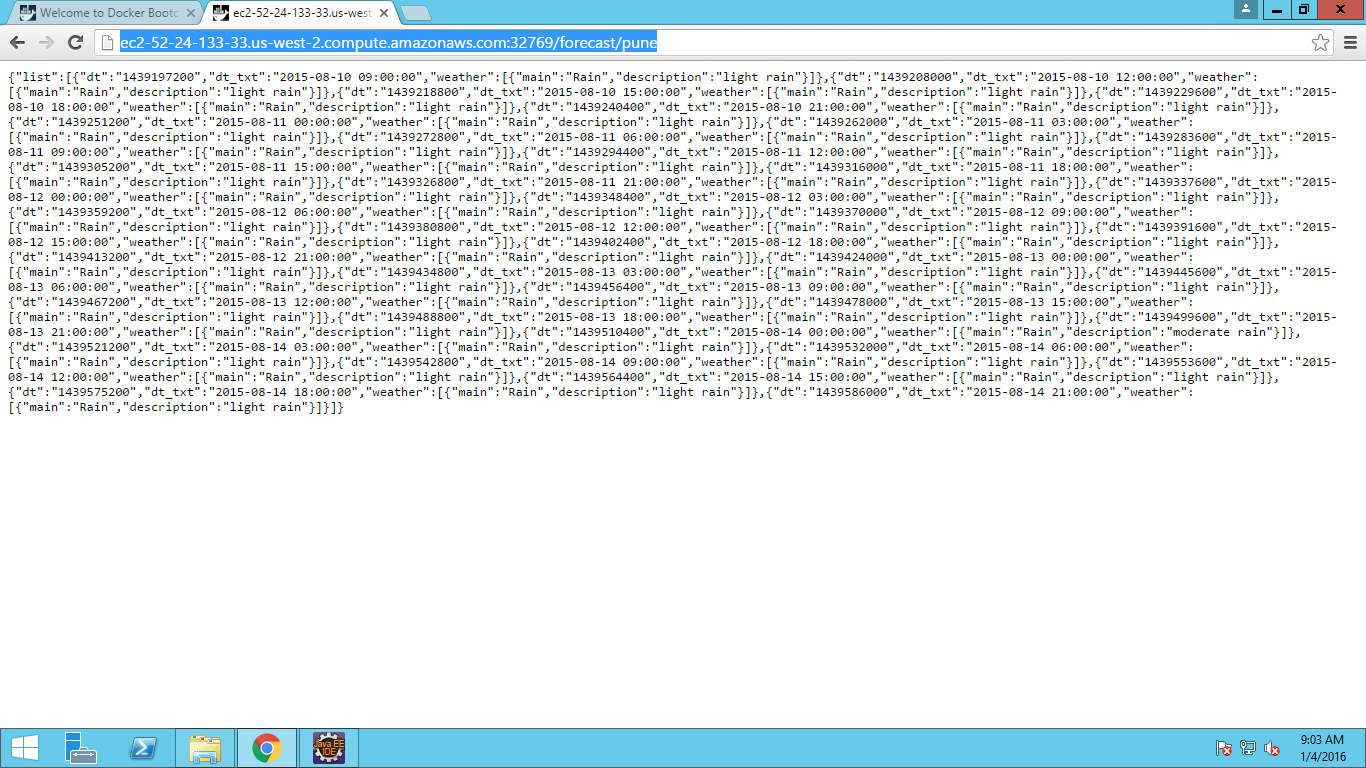
When the container is started, all logs are streamed into Eclipse Console. Here, the port at 8080 (the web server) is mapped to 32769 on the Docker daemon.



Test Application

To test the application goto http://dockerhost:32769 . dockerhost is the IP of the docker daemon.





Sample project you can find in this below github repository:

https://github.com/suprakashbh/forecast-ms-docker

Benefits

* Docker will improve the developer productivity and no need to setup runtime environment in every developer’s local machine.
* This Eclipse Docker Plugin will allow running Development Environment on Windows or Mac, but target native deployment on multiple various Linux Docker Host machine without setting up complex run time environment like install and configure application server, search engine, DB etc.
* Developers can share same Docker host machine without interfering other Developer work.
* Support management of Docker images and Container in Eclipse
* Support of search/push/pull operation to own and third party registries

Limitation

* This plugin is not stable yet, you might see some of the functionalities are not working.
* There is no integration with Docker and Eclipse server Adapter yet. This feature will ease the web application deployment to a Docker container.
* You have to manually delete old and untagged images from Docker Host.

Conclusion

Using Docker Eclipse plugin in docker envinment will be very useful in dealing with complex application.But you have to remember that this is still in early phase and should consider the limitation and compelxity before decide to go ahead to use this plugin.

There are other software like Vagrant and Fig also available to setup Docker developer environment. Docker has hired its main developer from Fig and is strongly supporting it as to tool to setup Docker based development environments.