# UK HAPI FHIR Server Instructions

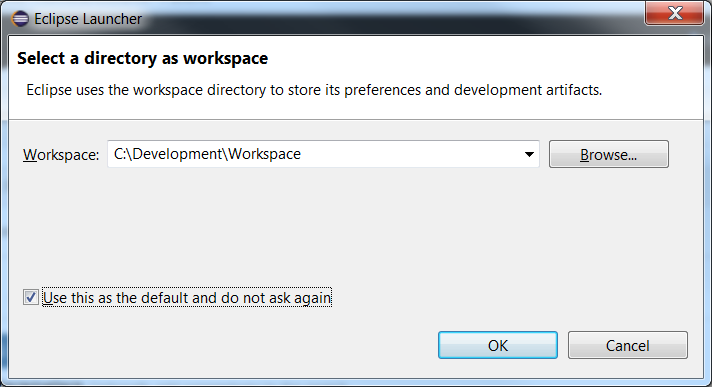
Version 1.0 beta

## Eclipse IDE

Download Eclipse IDE for Java Developers from <https://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/neonr>

(This includes git and maven which is required for development) Extract the zip and run eclipse.exe to start the IDE.

Enter in a workspace location.



## GIT

Install a git client e.g. <https://git-scm.com/>

## HAPI GitHub

Open up a GIT command shell (**Git CMD**). Create a directory to store the copy of the source code e.g.

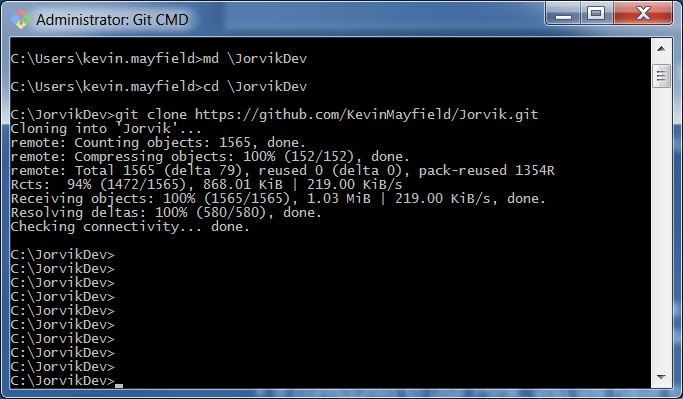
**md \JorvikDev**

Change to that directory and

cd \**JorvikDev**

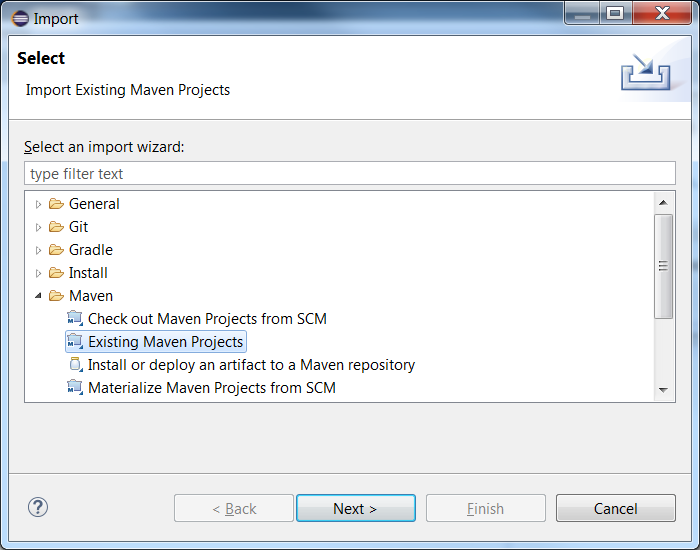
Issue the git clone command

**git clone** [**https://github.com/KevinMayfield/Jorvik.git**](https://github.com/KevinMayfield/Jorvik.git)

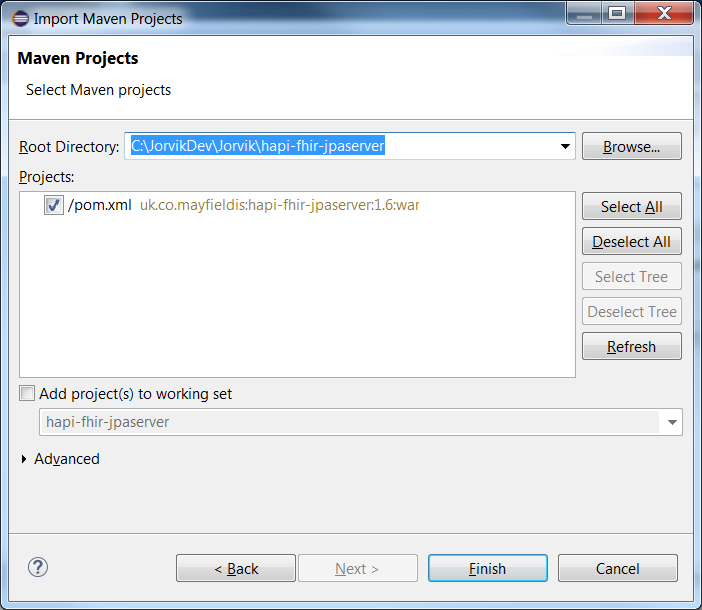


## Eclipse IDE

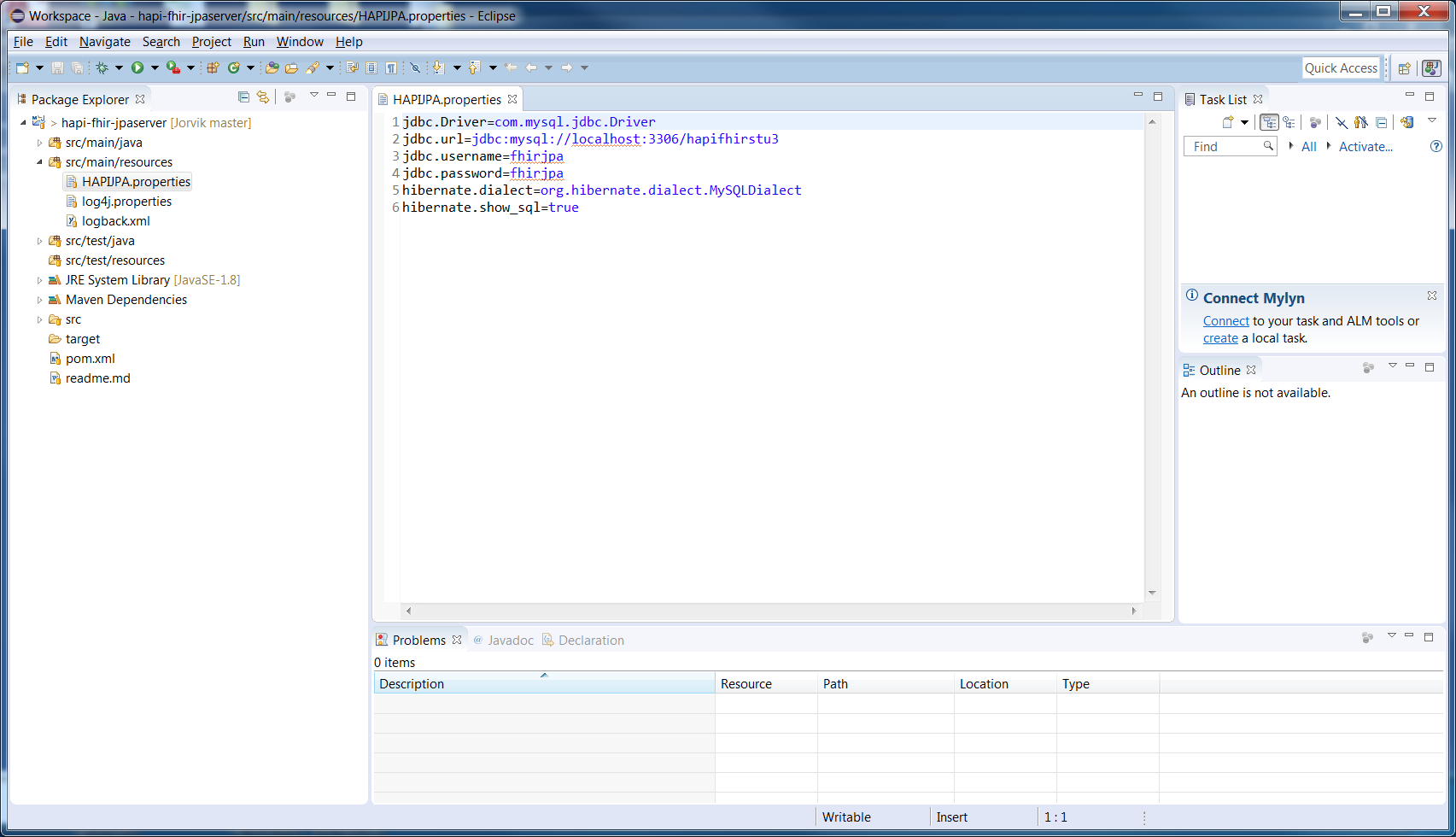
From the File menu of eclipse select import. Then locate ‘Existing Maven Projects’ wizard



Browse to the hapi-fhir-jpaserver project, if you followed the instruction above this would be in C:\JorvikDev\Jorvik\hapi-fhir-jpaserver



After a short while the project should have been imported into the IDE and it should look like:



I’ve opened up the HAPIJPA.properties file this is where the database configuration is stored. This configuration expects MYSql to be installed on your local machine running on port **3306**. The MYSql server also has a database called **hapifhirstu3** with a user called **fhirjpa**

HAPI uses hibernate to communicate with database. List of supported databases can be found here <https://developer.jboss.org/wiki/SupportedDatabases2>

## Java Development Kit 1.8 (JDK 1.8)

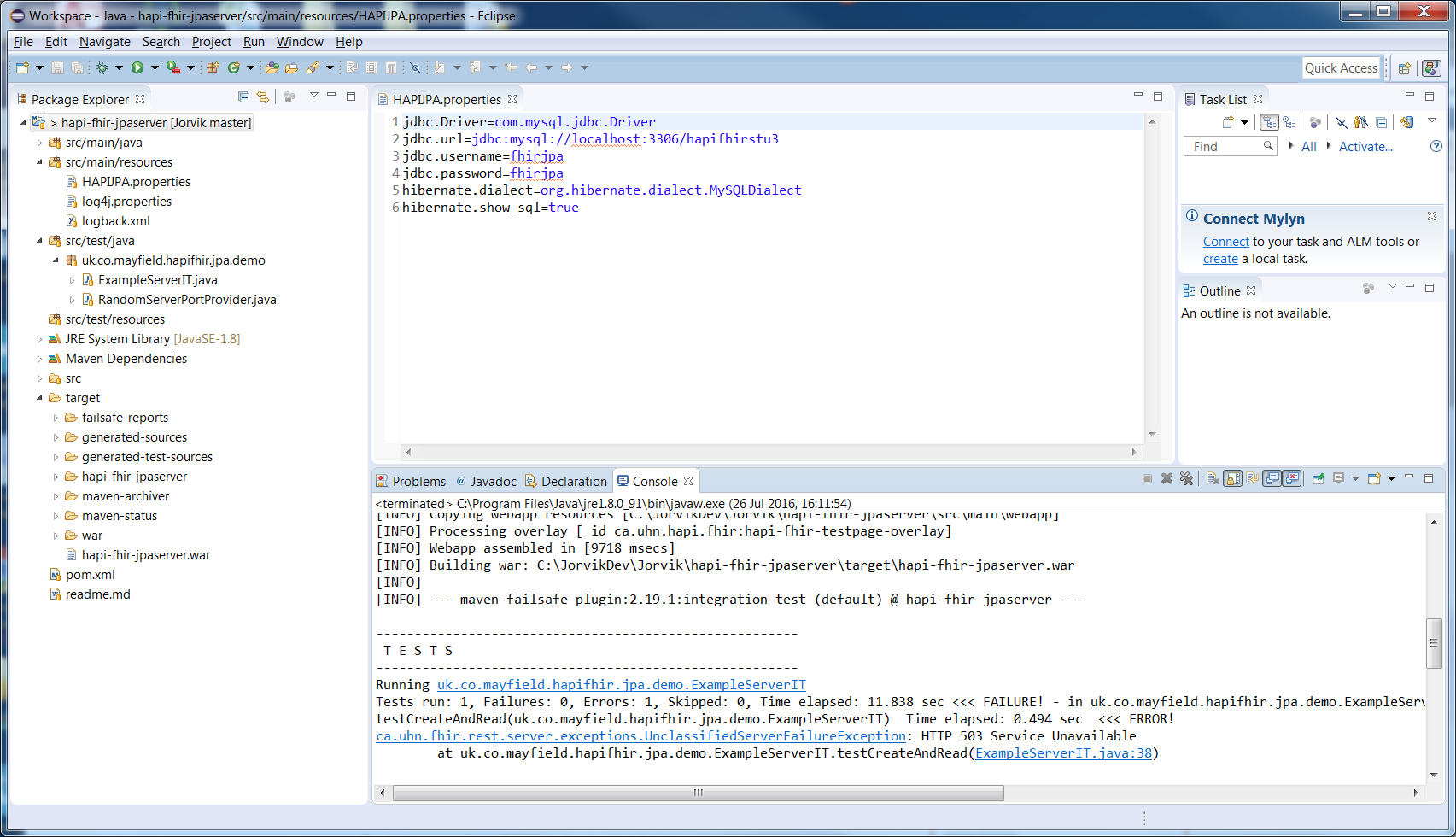
Before we can build the HAPI war we need to install a 1.8 JDK. This can be downloaded from <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Eclipse setup can be found here <http://stackoverflow.com/questions/13635563/setting-jdk-in-eclipse>

## Build HAPI War

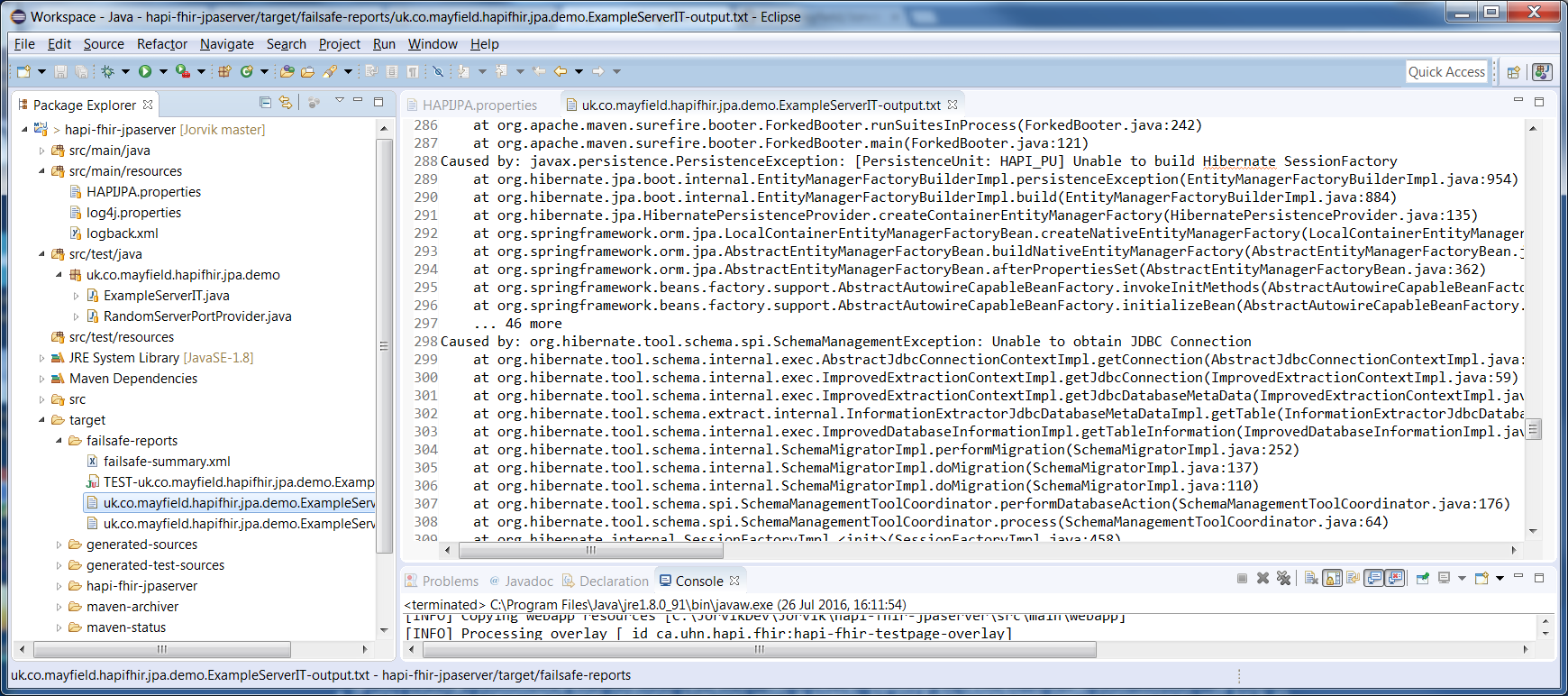
Right click on the project (hapi-fhir-jpaserver). Select **Run As->Maven Install**

This will compile the project and run tests against the build.



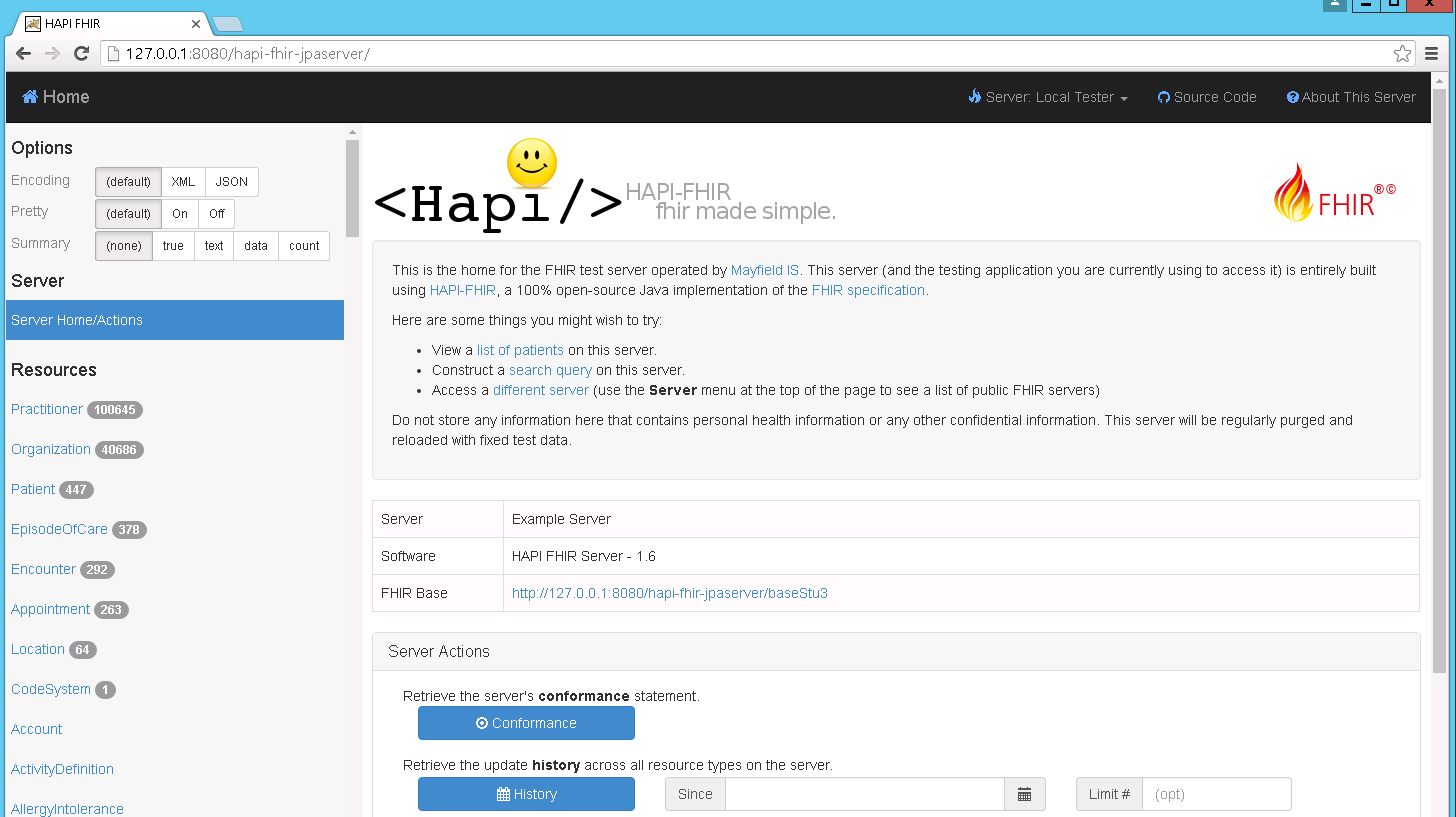
The tests didn’t work in the example above (note the 503 error) but it did produce the war file (look under target folder on the left).

To view the error reports look under the target->failsafe-reports folder. The output can be a little daunting but hunting through shows the tests failed to get a database connection (Unable to obtain JDBC connection), this was because the database server wasn’t started.



(Assuming the database server has been started)

The war file can now be deployed to a tomcat server (see tomcat tutorial) and the HAPI server will be operational.



## Other Apps – Apache ActiveMQ

To load other apps is similar but as we’ve done the complex work, a lot easier.

From the eclipse File menu, select Import again but this time choose the ActiveMQ-Jorvik folder (C:\JorvikDev\Jorvik\ActiveMQ-Jorvik)

Right click on the project and select **Run As->Maven Install**

This installs an embedded version of Apache ActiveMQ, which is used by the other projects. Deploy the compiled war to Apache Tomcat app.

# NHS SDS UPLOAD

Next compile SDS-HAPI webapp. When deployed to the tomcat server it will automatically load the latest gp and practice amendments into the hapi server (this runs every 24hours).

To process other files, browse to <http://systems.hscic.gov.uk/data/ods/datadownloads/index> , download and extract

|  |  |
| --- | --- |
| NHS Trusts | Etr.csv |
| Clinical Commission Groups | Eccg.csv |
| NHS Trust Sites | Ets.csv |
| GP Practices | Egppraccur.csv |
| English Hospital Consultants | Econcur.csv |
| GP Practitioners | Egcur.csv |

The extracted CSV files need to be placed into the C:\NHSSDS\extract folder on your tomcat machine. Load in the files in pairs starting with

* NHS Trusts and CCG’s
* Practices and NHS Trust Sites
* Consultants and Practitioners

The reason for the load order is each pair of files refers to the previous set. *This may take a while to complete.*

## HL7v2 ITK

If you have an existing HL7 v2 feed from your TIE (or PAS) you can use this to feed your HAPI server. This is a easy to use HAPI to provide a FHIR server for your organisation. The feed must be in NHS ITK HL7 format (pipe and hat not xml) <https://isd.hscic.gov.uk/trud3/user/guest/group/41/pack/34/subpack/200/releases>

The app can be found in the HL7v2-HAPI folder, compile as previously described. The webapp expects a TCP feed on port 8888 – it also uses the original version of HAPI which is used to process the HL7v2 messages.

## SNOMED CT / TERMINOLOGY

The HAPI install is configured as a terminology server. Please see the Upload Terminology section of the HAPI website <http://hapifhir.io/doc_cli.html>

I’ve only tested this with the core CT – would be interested to know if the UK supplement also loads.

<https://isd.hscic.gov.uk/trud3/user/guest/group/0/pack/1/subpack/102/releases>

|  |
| --- |
| Example command to run SNOMED import into HAPI from TRUD  ***hapi-fhir-cli upload-terminology -d c:\NHSSDS\SNOMED/SnomedCT\_RF2Release\_INT\_20150731.zip -f dstu3 -t http://localhost:8080/hapi-fhir-jpaserver/baseStu3 -u http://snomed.info/sct*** |

Test the load has worked by executing the following query

<http://127.0.0.1:8080/hapi-fhir-jpaserver/baseStu3/CodeSystem/$lookup?system=http://snomed.info/sct&code=209629006>

This should return

{

"resourceType":"Parameters",

"parameter":[

{

"name":"name",

"valueString":"Unknown"

},

{

"name":"display",

"valueString":"Complete tear, knee, anterior cruciate ligament (disorder)"

},

{

"name":"abstract",

"valueBoolean":false

}

]

}