Building IEP services on Ubuntu 20

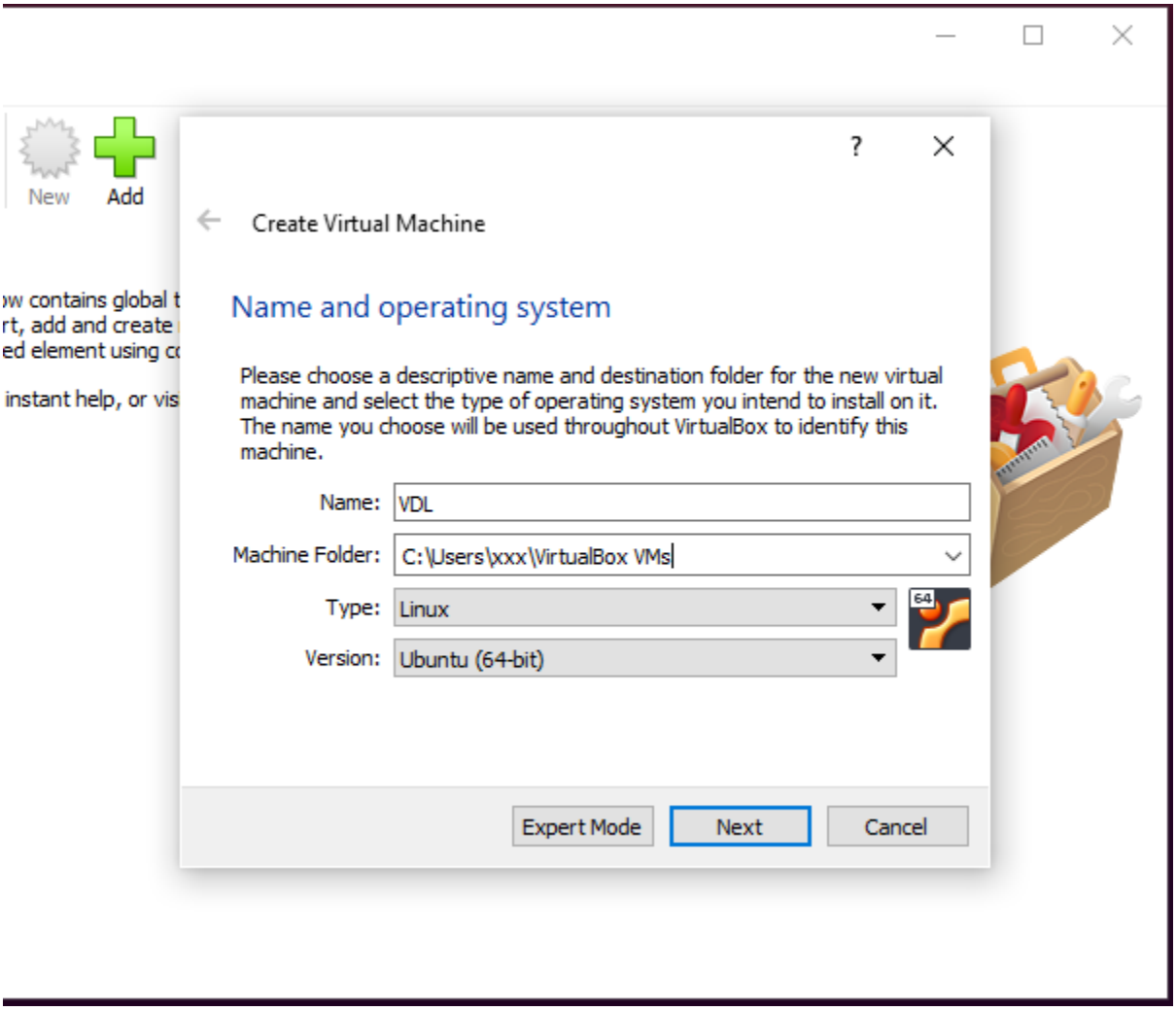
WORK IN PROGRESS - MAY NOT WORK YET

Part 1 Install Required Software

Install Oracle VirtualBox

https://www.virtualbox.org/wiki/Downloads

Create a new virtual machine



Click the New button
Give it a name such as VDL
Select Linux/Ubuntu (64-bit)
Select 8GB (8192MB)
Create a virtual hard disk
Give it 40GB of space or more - IEP can use a lot of space

When you have created your VM right click on it and select Settings...
Select System
Click on processor tab
Change Processor(s) to 2/4/6 depending on your laptop - keep it in the green, but the more the better for better performance

Useful keyboard shortcuts for VirtualBox

Right Ctrl + F : Change to full screen and back
Right Ctrl : release keyboard from VirtualBox so you can alt-tab, etc in Windows

Install Ubuntu on virtual machine

This guide was written for Ubuntu 20.04LTS - you may decide to use a different version, but some parts of this guide may not work correctly

https://www.ubuntu.com/download/desktop
Run your new VM (double click it in VirtualBox or click it and press Start button)
Insert Ubuntu into VirtualBox start up disk
Follow all the prompts to install Ubuntu
Select install updates if you wish
Then Erase disk and install Ubuntu (this will only erase the virtual hard disk not your laptop's hard disk)
When it has finished installing press Restart Now button
Then press Enter to restart (The disk is automatically ejected so ignore that part)

Install Guest Additions on virtual machine

```
sudo apt-get update
sudo apt-get install build-essential
```

VirtualBox Devices menu -> Insert Guest Additions CD image...
A window should open. Click Run then type your password.
When it finished press Return to close the window
Enable clipboard sharing so you can copy commands between device and the VM: Devices -> Shared Clipboard -> Bidirectional

Reboot

Power/cog icon in top right -> Shutdown -> Restart
Or you can open a terminal (ctrl+alt+t) and type

```
sudo reboot
```

Open a terminal window

ctrl+alt+t or through Ubuntu menu
ctrl+alt+5 will make the window fullscreen

Update apt-get

```
sudo apt-get update
```

(Optional) Install vim (nice colours and syntax hilighting in vi)

```
sudo apt-get install vim
```

(Optional) Install Chromium web browser (or stick to built in Firefox if you prefer)

```
sudo apt-get install chromium-browser
```

Run Chromium in terminal
Open terminal ctrl+alt+t

```
chromium-browser
```

Pin it to side bar so you can load it again (rlight click Chromium on left panel and select "Add to favourites" and close Chromium

(Optional) Install Intellij (only required to make any code changes - or you could use vi for that if you really want)

https://www.jetbrains.com/idea/download/

(Optional) Install Robo3T for interacting with Mongo database

https://robomongo.org/

Download Robo3T and save it somewhere
Extract tar.gz file somewhere
Locate the bin folder in the folder you just extracted
Double click robo3t
Agree to the terms (obviously read them first)
You don't need to give a name, just click Finish
Right click Robo3T icon on left panel and select Lock to launcher

Install curl

```
sudo apt-get install curl
```

Install SDKMAN!

Some software is difficult to find recently as repositories that were previously used to install software is no longer available. SDKMAN! can be used to install these

```
curl -s "https://get.sdkman.io" | bash
```

Install Java JDKs

Since the migration to Mongo Atlas and TES, some applications require Java 8 - most still run on Java 7, so we require to versions of Java. Use SDKMAN! to install these

```
sdk install java 7.0.332-zulu
sdk install java 8.0.332-zulu
```

Install Mongo

This will install version 4.2 of MongoDB.

```
wget -qO - https://www.mongodb.org/static/pgp/server-4.2.asc | sudo apt-
key add -
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu
bionic/mongodb-org/4.2 multiverse" | sudo tee /etc/apt/sources.list.d
/mongodb-org-4.2.list
sudo apt-get update
sudo apt-get install -y mongodb-org
sudo systemctl enable mongod
sudo service mongod start
```

Install RabbitMQ Server

```
sudo apt-get install rabbitmq-server
```

Install Maven

```
sudo apt-get install maven
```

Install Ruby to get RugyGems

```
sudo apt-get install ruby2.7-dev
```

Install SASS

```
sudo gem install sass
```

Install git

```
sudo apt-get install git
```

Install Python 2.7

```
sudo apt-get install python2.7
```

Create a soft link to point “python” to python 2.7. This will make sure some IEP scripts work correctly

```
sudo ln -s /usr/bin/python2.7 /usr/bin/python
```

Install python-pip

```
curl https://bootstrap.pypa.io/pip/2.7/get-pip.py --output get-pip.py
sudo python get-pip.py
```

Install python requests package

```
sudo pip install requests
```

Install SBT

```
sdk install sbt 0.13.17
```

Part 2 Build VDL from source code

Setup Ivy Cache

Some of the files required are no longer available for SBT to download so you will need to manually copy them to the correct place.

Download this file ivy2-cache.tar.gz

You will need to copy this to the home folder of your VM. You may be able to access the link from the VM. If not you may need to create a shared folder that your VM can access files from outside the VM.

```
cd ~
tar -xf ivy2-cache.tar.gz
```

Clone VDR repo in BitBucket

Make sure you are in your home directory (~ is a shortcut for home)

```
cd ~
git clone https://bitbucket.tooling.dvla.gov.uk/scm/iep/vdr.git
cd vdr
```

Edit script to download all repos

```
sed -i 's/cd ..//g' download-vdr-projects.sh
sed -i 's/\.git /\.git -b master /g' download-vdr-projects.sh
sed -i 's/git-il2-internal.iep.dvla.gov.uk\/dvla-dev/collaboration.dvla.
gov.uk\/bitbucket\/scm\/iep/g' download-vdr-projects.sh
sed -i 's/git-il2-internal.iep.dvla.gov.uk\/testers/collaboration.dvla.
gov.uk\/bitbucket\/scm\/iep/g' download-vdr-projects.sh
```

Download all repos

```
./download-vdr-projects.sh
```

Edit main pom file to remove .git from folder names

```
sed -i 's/.git//g' pom.xml
```

Copy Maven settings file into maven settings folder

```
mkdir ~/.m2
cp settings.xml ~/.m2
```

Change some settings to make services access locally instead of docker containers

Note: It's possible that changes are made to these files and these commands don't work as they should

You can make these changes manually if you prefer

Indentation is important in these files
Best to verify each file and make sure they are using "localhost" and not "dev-???-server", etc
You can copy the lot and paste them in one go if you want

```
sed -i 's/dev-.*-server/localhost/g' vdr-customer-portal/conf/params.
json
```

```
sed -i 's/dev-driver-enquiry-server/localhost/g' vdr-services
/authentication-server/src/main/resources/config.yaml
sed -i 's/dev-ida-linkstore-reader/localhost/g' vdr-services
/authentication-server/src/main/resources/config.yaml
sed -i 's/^licenceAccessTokenService/##licenceAccessTokenService/g' vdr-
services/authentication-server/src/main/resources/config.yaml
sed -i 's/#licenceAccessTokenService/licenceAccessTokenService/g' vdr-
services/authentication-server/src/main/resources/config.yaml
sed -i 's/^dwpServiceEndpoint/##dwpServiceEndpoint/g' vdr-services
/authentication-server/src/main/resources/config.yaml
sed -i 's/#dwpServiceEndpoint/dwpServiceEndpoint/g' vdr-services
/authentication-server/src/main/resources/config.yaml
```

```
sed -i 's/#  - localhost/  - localhost/g' vdr-services/driver-enquiry-
server/src/main/resources/config.yaml
sed -i 's/ - token-service-mongodb/# - token-service-mongodb/g' vdr-
services/driver-enquiry-server/src/main/resources/config.yaml
```

```
sed -i 's/^baseUrl/##baseUrl/g' vdr-services/driver-filter-server/src
/main/resources/config.yaml
sed -i 's/#baseUrl/baseUrl/g' vdr-services/driver-filter-server/src/main
/resources/config.yaml
```

```
sed -i 's/#- localhost/- localhost/g' vdr-services/driver-write-server
/src/main/resources/config.yaml
sed -i 's/- token-service-mongodb/#- token-service-mongodb/g' vdr-
services/driver-write-server/src/main/resources/config.yaml
sed -i 's/^ fetchUrl/## fetchUrl/g' vdr-services/driver-write-server/src
/main/resources/config.yaml
sed -i 's/# fetchUrl/ fetchUrl/g' vdr-services/driver-write-server/src
/main/resources/config.yaml
```

Build the source (This will only work if VPN is connected and if the Nexus server is running)

```
```

```
mvn clean install -DskipTests -Dmaven.test.skip=true -Djavax.xml.
accessExternalSchema=all
```

When it gets to the customer-portal, SBT will look like it's not doing anything for a while while it loads stuff. This will only happen the first time. Don't stop it - it is working.

Have lots of fun trying to fix pom versions

You'll no doubt receive errors when building this. Most likely to do with the version of some packages in pom files.
First error I had was

> *Non-resolvable parent POM for uk.gov.dvla.iep:vdr-bridge-validators:1.1.6-SNAPSHOT: Failure to find uk.gov.dvla.iep:vdr-parent:pom:1.6-SNAPSHOT*

This says it can't find vdr-parent version 1.6-SNAPSHOT on nexus which the vdr-bridge-validators is trying to use.
Looking at vdr-parent/pom.xml the current one is 1.11-SNAPSHOT

```
less vdr-parent/pom.xml
```

and look at the version number (you might have to change 1.11 to something else if it's different there)
edit vdr-bridge-validators/pom.xml to change vdr-parent to 1.11-SNAPSHOT:

```
vi vdr-bridge-validators/pom.xml
```

press i to change vi to insert mode

edit vdr-parent version from 1.6-SNAPSHOT to 1.11-SNAPSHOT
ESC :wq ENTER
try running

```
mvn clean install -DskipTests -Dmaven.test.skip=true -Djavax.xml.
accessExternalSchema=all
```

again. When it gets to the customer-portal it will look like it has stopped while installing SBT. Don't stop this, it takes a while.

If you manage to build some packages and get a failure later on you can continue from where you left off.

```
mvn clean install -DskipTests -Dmaven.test.skip=true -rf :<package-
name> e.g.
mvn clean install -DskipTests -Dmaven.test.skip=true -rf :vdr-customer-
portal
```

Hopefully at the end of that **SUCCESS**

If you manage that you have built VDL and can get some stuff running on it

Part 3 Run VDL software and play with some test data

Time to insert some test data into the database and get the services running

Setup Batch folder Path

```
echo 'export PATH=$PATH:~/vdr/batch' >> ~/.bashrc
Close terminal window (exit, ctrl+d or cross at top of window)
```

re-open terminal (ctrl+alt+d)

```
cd vdr
```

Run driver-write-server

```
iep_services start driver-write-server
```

Install test data

```
python batch/loadtestdata.py vdr-test-utils
```

Hopefully that will work and insert the test data into the database

Stop driver-write-server

```
iep_services kill driver-write-server
```

Start essential services required to load customer portal

```
iep_services start essentials
```

Customer portal takes a little while to load. You can check this in the logs/customer-portal.log

```
less logs/customer-portal.log
```

press shift-f to wait for new data in the log
wait til you see something like:

> *2018-08-24 16:49:56,305 [ESC[37minfoESC[0m] [play] Listening for HTTP on /0:0:0:0:0:0:0:0:9000*

press ctrl+c to stop waiting for log data
press q to quit less

Load customer portal

Open web browser of choice and vist either
http://localhost:9000/index - this is what customers see
or
http://localhost:9000/ops/authHelper - this is like the ops portal contact center or we use
username and password is test/test

If you run the customer portal but do not shutdown the VM properly or kill the customer portal service, the portal will fail to start when you try to run if the next time. If you look in the customer-portal.log file in the logs directory, you will see something like:

> *Play server process ID is 2359*
> *This application is already running (Or delete /home/<user>/vdr/vdr-customer-portal/target/universal/stage/RUNNING_PID file).*

To fix this you need to delete the file /home/<user>/vdr/vdr-customer-portal/target/universal/stage/RUNNING_PID

```
rm /home/<user>/vdr/vdr-customer-portal/target/universal/stage
/RUNNING_PID
```

Load a record from test data

Open Robo3T
Create a new connection
Give it a name if you wish, leave all other options the same
Connect to server
You should see the database open on the left panel
Double click on DVLA
Double click on Collections
Double click on drivers
You should see a list of all the test records
Double click one of the records to see its data
Right click on currentDriverNumber and click copy value
If you are using the ops portal version you can just paste that into the DLN box at the bottom, select an option on the left for why you are viewing the record (doesn't matter on here which one)
If you are using the customer portal you will also need the post code from the DB and a NINO. NINO uses dwp-mock to mock the dwp link which authenticates NINOs. Doesn't matter what you put here so long as it follows the proper NINO format. AA123456A will work.

Hopefully you will see the record in VDL.

Insert Enquirer Reference and IP into database for ADD to work

Load up Robo3T and connect to the database
Double click DVLA
Right click Collections and Create Collection... and call it "organisations"
Double click Collections
Right click on organisations and select Insert Document...
paste the following starting and ending at the curly brackets and click save

```
{
    "_id" : "DVLA",
    "enquirerReference" : "DVLA",
    "ipAddresses" : [
        "123.123.123.123"
    ]
}
```

Access driver using ADD API

For some reason the config file for add-api-service does not go into the target directory
from the vdr folder:

```
cp add-api-service/src/main/resources/config.yaml add-api-service/target
/classes/
```

You can either start all the services with:

```
iep_services start all
```

but the mib-batch-2-file-loader will start filling up your log dirctory quickly
so:

```
iep_services kill mib-batch-2-file-loader
```

or just start the ADD services (assuming you have already run start essentials)

```
iep_services start add-api-service
iep_services start organisation-service
```

Then grab a DLN from the database and run this, replacing DLN with the DLN:

```
curl "http://localhost:11000/licence-enquiry/api/v1/driver?
guid=e9890bc3-4e92-431d-8670-3745bef0d5fa&enquirerRef=DVLA&sourceIp=123.
123.123.123&dln=DLN"
```

You may get errors (connection refused or system error) until all the services are running correctly.

ADD can also show CPC and Tacho information by appending queries to the URL.

```
For CPC add: &cpc=true
For Tacho add: &tacho=true
```

This should return some XML that third parties use for viewing driver records.

ADD can also return the results in JSON format by adding the following to the curl request:
-H "Accept: application/json"

Access a record using MIB

MIB can be accessed just by running

```
iep_services start essentials
```

then (replacing DLN with the DLN you want):

```
curl "http://localhost:9200/iiadd/api/v1/driver?guid=c2a20841-4c37-49b4-
b8f8-22288da4f9d6&dln=DLN"
```