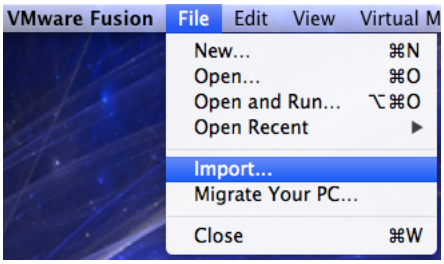
download the Fast Track VM Image

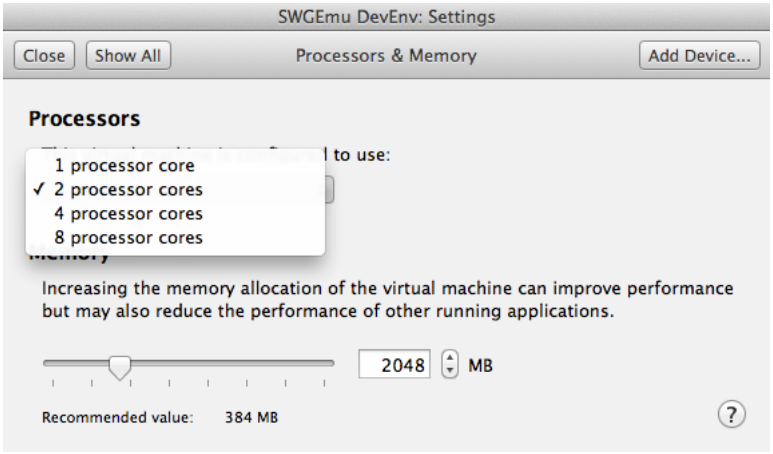
http://www.swgemu.com/forums/showthread.php?t=85624

import to vmware

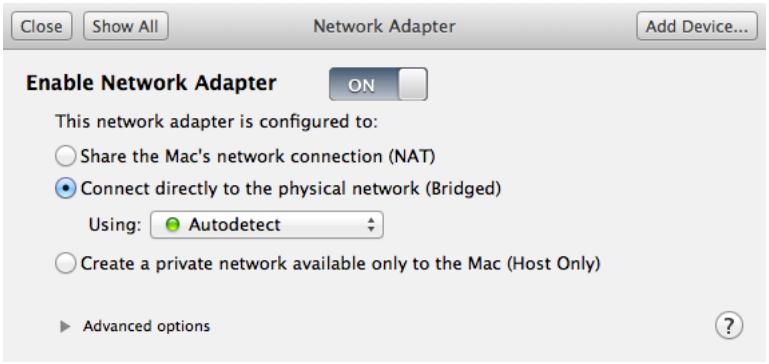


Find the *.vmx file from the expanded zip file.

Edit settings (if you have more memory and cores you might want to bump these up).

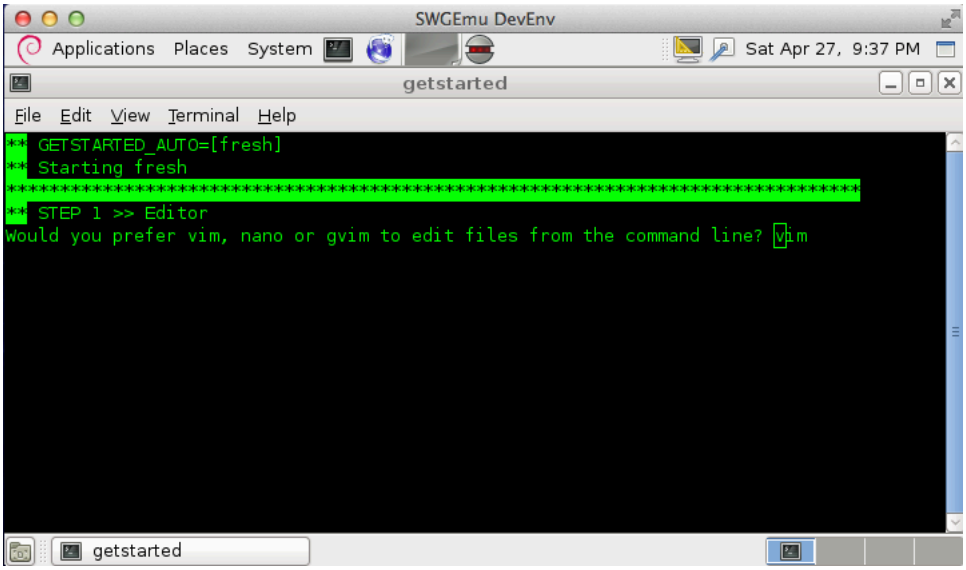


You will most likely want to setup directly connected to your network or else you won't be able to connect to the server from the windows client.

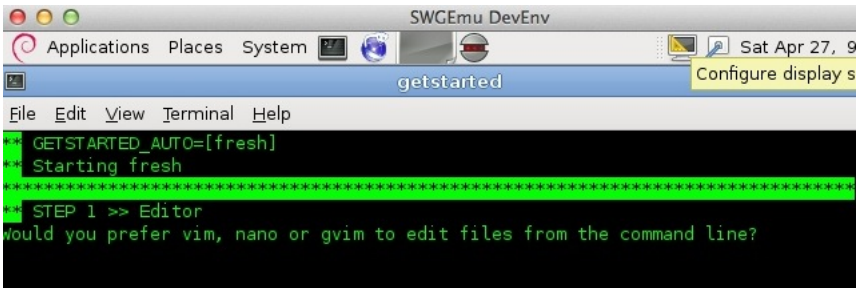


Fire it up...

It should auto-login to something like this:



Choose your favorite editor and you might want to resize your screen to make the rest of the process a bit easier..



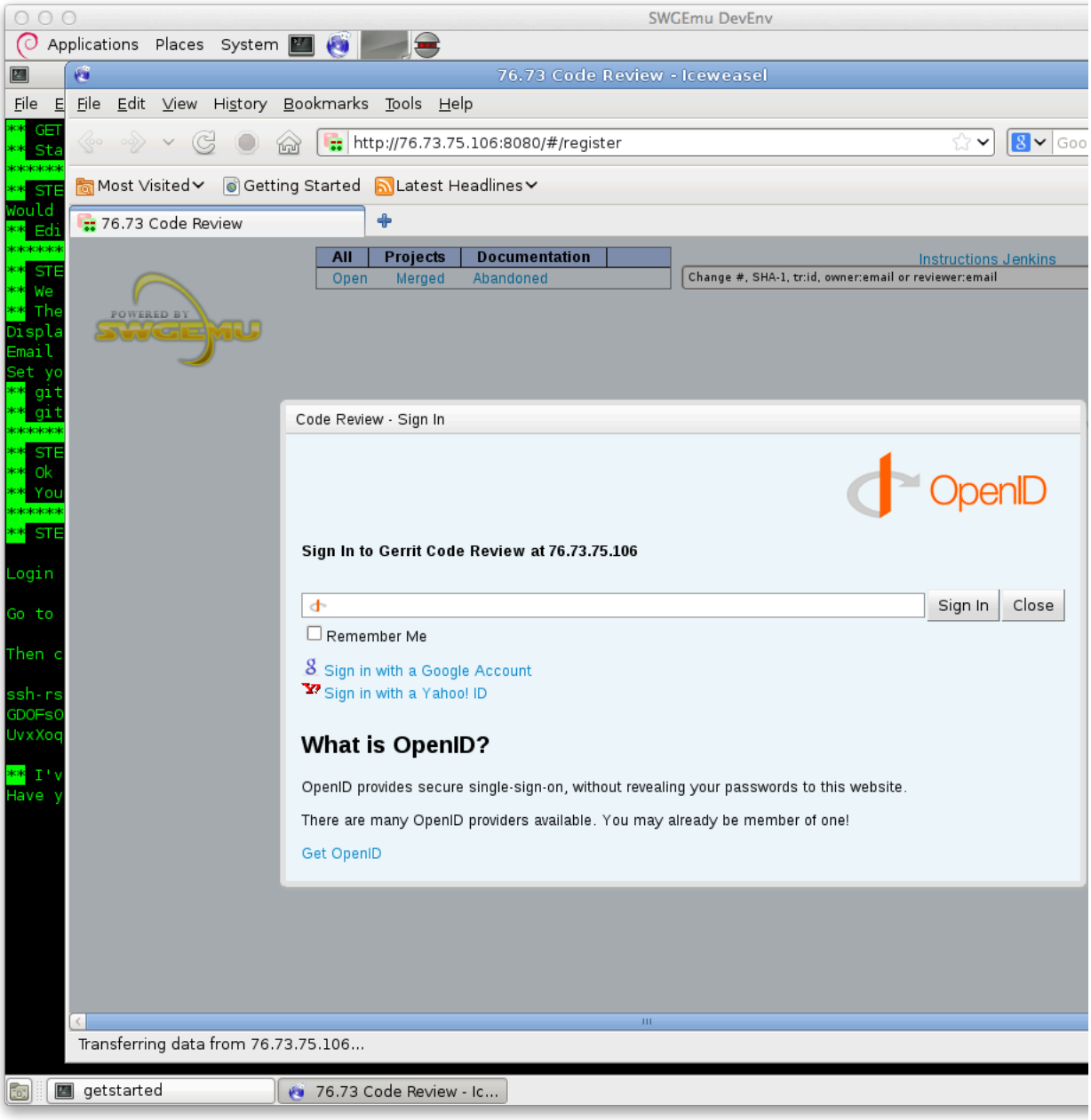
Enter your name and email for git commit messages:

```
********************************************************************************************************
** STEP 2 >> Git config
** We need to setup your git info for your commits.
** These will show up when you commit to Gerrit so you want them to be useful to the real world.
Display Name for git (i.e. Elmer Fudd): Darth Vadev
Email for git (i.e. elmerfudd@example.com): darthvadev@gmail.com
Set you git user info to: name=[Darth Vadev] email=[darthvadev@gmail.com]?
```

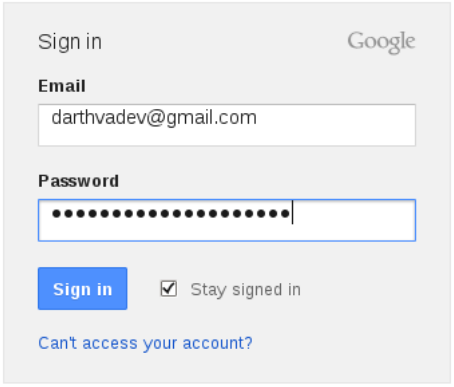
You ssh key will be generated, pasted to your clipboard...

```
********************************************************************************************************************
** STEP 3 >> SSH Key
** Ok looks like you need ssh keys, let's make some for you!
** Your new key is in ~/.ssh/id_rsa and the public key is in ~/.ssh/id_rsa.pub
********************************************************************************************************************
** STEP 4 >> Gerrit registration

Login to http://76.73.75.106:8080/ and register an account using an openID provider

Go to your Gerrit account and hit Settings -> SSH Public Keys -> [Add Key...]

Then cut-paste the following into the Add SSH Public Key box:

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCfUuLsJ7Zc7zO3s0MlUjXipbhaHIXH32lLb4MIFq7MN8+IqO/S64Grjvr7XjjROUwruwkr+VIUNTLK
GDOFsOyad/FZ5Sr8YpwYN77TT2gO7IKFW4i2jgWj4koT5g39FuAi/fBRbOCtQR2mxxqsNXQsgIP/Q4UTT+OMVYQwYHRSFjN+g+BMhHITUjNAZSM6uA86
UvxXoqG6r8OZ/cz8c2bXY4lFxofYZBM3pEjVRjUuohxkDgMEexFSEdw2NOJZ36IEBNn2h31ThPe4BeDqnLpazgXt emudev-20130427-darthvadev@

** I've also sent this to your clipboard via xclip to make it easier if you're on a graphical console to just paste
Have you done this yet?
```

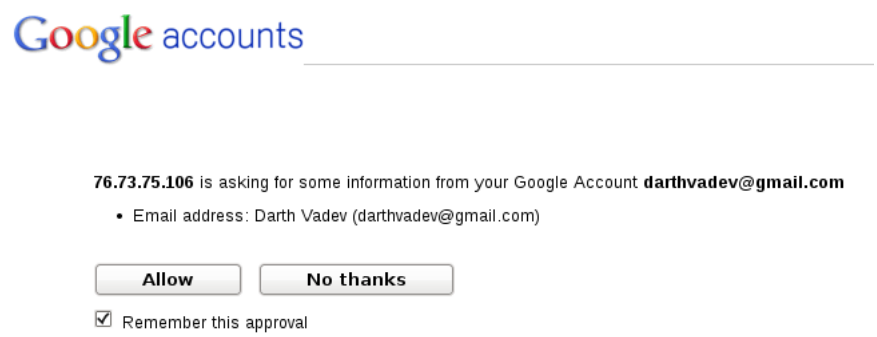
The firefox will be launched so you can register (or login) to gerrit..



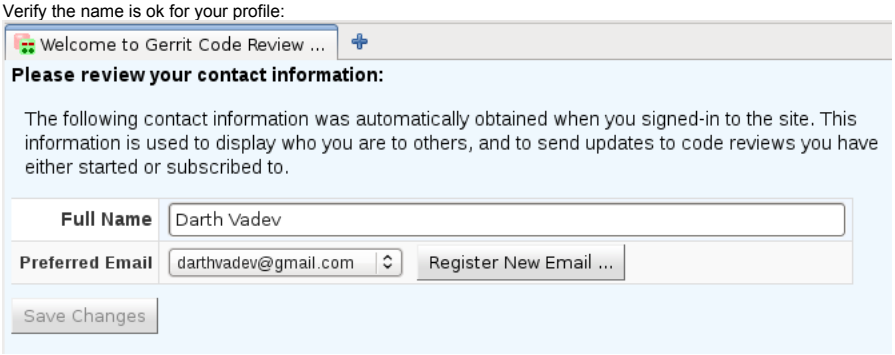
Click sign in with a "Google Account" or Yahoo whatever you have..



Approve the authorization (your password is NOT shared with gerrit that's why we use openID):

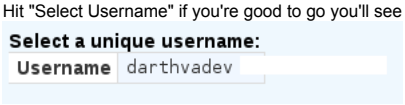


Verify the name is ok for your profile:

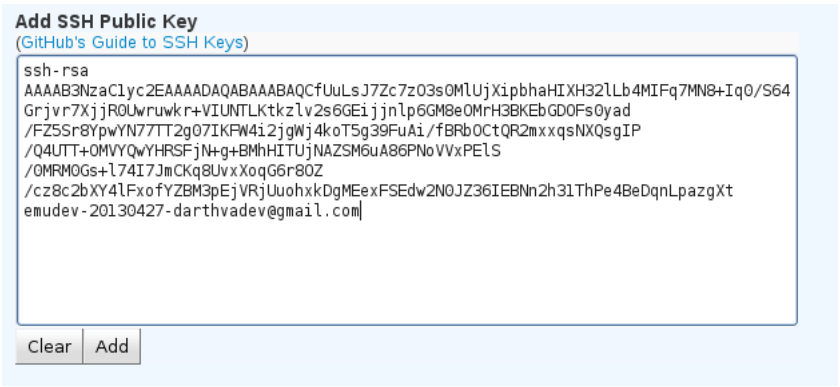


Choose a username (remember this you'll need it soon!)

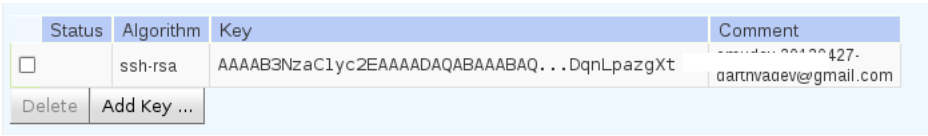
Hit "Select Username" if you're good to go you'll see something like this:



Paste your new SSH key into the dialog box:



Hit Add button and you should see something like:

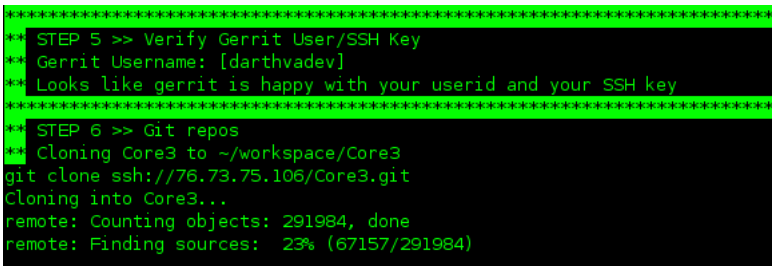


Now you can minimize the firefox window and return to the "getstarted" window…

Answer Yes to the "Have you done this yet?" prompt

Enter the username you registered with gerrit above..

If all is well you should see the SSH worked and it will start cloning the Core3 repo:



At this point go get some coffee, read hacker news whatever you do to kill time, the script will run for a while getting all the right repos down and setting stuff up for you.

IGNORE THIS:

```
warning: remote HEAD refers to nonexistent ref, unable to checkout
```

It's harmless as the script will check out a copy of origin/unstable to your local repo after the clone is complete.

```
Resolving deltas: 100% (251726/251726), done.
warning: remote HEAD refers to nonexistent ref, unable to checkout.
```

```
Install commit-msg hook
commit-msg                                                                                          100% 4270     4.2KB/s
00:00
Checking out files: 100% (27364/27364), done.
Branch unstable set up to track remote branch unstable from origin.
Switched to a new branch 'unstable'
** Cloning PublicEngine to ~/workspace/PublicEngine
git clone ssh://76.73.75.106/PublicEngine.git
Cloning into PublicEngine...
remote: Counting objects: 5911, done
remote: Finding sources: 100% (5911/5911)
remote: Total 5911 (delta 3668), reused 5911 (delta 3668)
Receiving objects: 100% (5911/5911), 129.13 MiB | 4.82 MiB/s, done.
Resolving deltas: 100% (3668/3668), done.
Install commit-msg hook
commit-msg                                                                                          100% 4270     4.2KB/s
00:00
** Cloning Tools to ~/workspace/Tools
git clone ssh://76.73.75.106/Tools.git
Cloning into Tools...
remote: Counting objects: 1731, done
remote: Finding sources: 100% (1731/1731)
remote: Total 1731 (delta 980), reused 1731 (delta 980)
Receiving objects: 100% (1731/1731), 55.23 MiB | 2.77 MiB/s, done.
Resolving deltas: 100% (980/980), done.
Install commit-msg hook
commit-msg                                                                                          100% 4270     4.2KB/s
00:00
**************************************************************************************************************************************************
*******
** STEP 7 >> Symlinks
** lrwxrwxrwx 1 swgemu swgemu 25 Apr 27 22:05 Core3/MMOEngine -> ../PublicEngine/MMOEngine
** lrwxrwxrwx 1 swgemu swgemu 16 Apr 27 22:05 MMOCoreORB -> Core3/MMOCoreORB
** lrwxrwxrwx 1 swgemu swgemu 22 Apr 27 22:05 MMOEngine -> PublicEngine/MMOEngine
**************************************************************************************************************************************************
*******
** STEP 8 >> Engine Library
Setup unix library link
** Using engine library: ../linux64/libengine3.a
**************************************************************************************************************************************************
*******
** STEP 9 >> MySQL database
** Restore blank swgemu database to mysql
** Runing mysql update: /home/swgemu/workspace/Core3/MMOCoreORB/sql/updates/account_ips.sql
** Runing mysql update: /home/swgemu/workspace/Core3/MMOCoreORB/sql/updates/deletedcharacters_add_dbdeleted.sql
**************************************************************************************************************************************************
*******
** STEP 10 >> Server configuration
** By default only tatooine and tutorial zones are enabled
```

After a while you should see something like this:

```
************************************************************
** STEP 10 >> Server configuration
** By default only tatooine and tutorial zones are enabled
Would you like to edit the default configuration? Y
```

If you want to mess with the config now is the best time, you can edit the config, it'll check the syntax for you and make sure you didn't make an obvious syntax error etc..

Your config will be checked and then you'll be prompted to copy the game TRE files to your server:

```
** Checking config syntax...
** SUCCESS - /home/swgemu/run/conf/config.lua passed lua parser
** Zones enabled: naboo tatooine tutorial
********************************************************************************
** STEP 11 >> TRE files
NOTE: You must make sure to copy all the *.tre files from your game disk to: /home/swgemu/workspace/tre

EXAMPLE (on OSX box):

  cd /Volumes/BOOTCAMP/SWGEmu
  scp *.tre swgemu@172.16.10.80:/home/swgemu/workspace/tre

Did you do this yet? Y
```

On OSX this is pretty easy as you can see above, Windows you'll need to figure out a smart way to copy the files to the ~/workspace/tre directory.

The default password is '123456' until you change it, you'll need that to copy to the swgemu user's home via ssh..

```
$ scp *.tre swgemu@172.16.10.80:/home/swgemu/workspace/tre
swgemu@172.16.10.80's password:
bottom.tre                                                                    100%   15M
data_animation_00.tre                                                         100%   56M
data_music_00.tre                                                             100%   74M
data_other_00.tre                                                             100%   44M
data_sample_00.tre                                                            100%  100M
data_sample_01.tre                                                            100%  100M
data_sample_02.tre                                                            100%  100M
data_sample_03.tre                                                            100%  100M
data_sample_04.tre                                                            100%   32M
data_skeletal_mesh_00.tre                                                     100%   99M
data_skeletal_mesh_01.tre                                                     100%   41M
data_sku1_00.tre                                                              100%  100M
data_sku1_01.tre                                                              100%  100M
data_sku1_02.tre                                                              100%  100M
```

when it's done.. you can tell script you're ready for the next step..

next the eclipse environment will be setup and the projects imported to eclipse.

This will take a bit as eclipse refreshes the files in the project against the filesystem.

```
Did you do this yet? Y
********************************************************************************
** STEP 12 >> Setup Eclipse
** Restoring basic eclipse install from /home/swgemu/setup/eclipse-clean.tar.gz
100% [=============================================================]
** Import eclipse project settings (might take a few minutes)
** Importing Core3/MMOCoreORB
Create.
Opening 'MMOCoreORB-Unstable'.
Refreshing '/MMOCoreORB-Unstable'.
```

Next steps are up to you, you can build the server right now or exit the script and build in eclipse..

```
** Setup Egit properties for MMOCoreORB-Unstable
** Setup Egit properties for MMOEngine
** Eclipse setup and import complete
************************************************************
** STEP 13 >> Go for it
**
** Now you should be able to build the server with:
**
** build config
**
** And if that works you can run the server with:
**
** run_dev
**
** Have fun!
Would you like to build and run the server now? Y
```

If you said yes to build and run the server will be built and ran using the 'run_dev' script.

```
(97 s) [CityManager] Loaded 0 player city regions.
(98 s) [DatabaseManager] trying to create database auctionitems
(98 s) [AuctionManager] Checking 99 bazaar terminals
(98 s) [AuctionManager] Checking 0 vendor terminals
(98 s) [AuctionManager] loaded auctionsMap of size: 0
[New Thread 0x7fff699d5700 (LWP 14775)]
(98 s) [StatusServer] initialized
[New Thread 0x7fff691d4700 (LWP 14776)]
[New Thread 0x7fff689d3700 (LWP 14777)]
[New Thread 0x7fff5bfff700 (LWP 14778)]
(98 s) [Core] initialized
>
```

If you hit Control-C you'll be in gdb:

```
> ^C
Program received signal SIGINT, Interrupt.
0x00007ffff5dc364d in read () from /lib/libc.so.6
(gdb)
```

And can do normal gdb things there..

Some cool tips, there are a number of command line tools to make life much easier when logged in as swgemu user:

CDPATH is setup for a bit of directory changing fun (and confusion if you don't understand it).

cd src -> takes you to ~/workspace/MMOCoreORB/src
cd bin -> takes you to ~/workspace/MMOCoreORB/bin

openfile {filename} - open file in eclipse
run_dev - Build and run the development server and launch it under gdb on a 'screen'
freeze - Save your development server state so you can repeat the same tests over and over
thaw - allow server to continue from previous state each time you run it
latest - do a quick git-stash and pull so you can get to the latest code w/o loosing local work
ack - Nice source grep tool (try: cd ~/workspace/MMOCoreORB/src; ack PlanetManager)
sudo - run stuff as root, try and avoid this as permissions can get ruined quite quickly
myip - display the ip of the VM and login port for quick configuration of the windows client

NOTE: run_dev uses gdb in batch mode and starts with the commands in ~/run/run_gdb which you can change to your pleasing (breakpoints, dumps, settings etc.)

You can detach from the console by typing: Control-A followed by 'd' ({Control-A}{d}) and rejoin by typing screen -r (for more fun see: man screen)