

Go

## GoClipse – Setup and use of GoClipse

In this optional lab we'll be installing the Eclipse IDE and the GoClipse Go development environment plugin. This is an optional lab and need only be completed if you would like to code the remaining labs in an IDE.

Eclipse is largish and your lab VM is configured for 2GB of ram. If you can increase your lab VM's memory to 4 GB or even 8GB it will improve the performance of Eclipse. You will need to shutdown the VM to change the memory settings and then restart it.

To install Eclipse and GoClipse we will have to add the following components to the lab system:

- Java 8 (Eclipse is a Java application)
- Firefox (we'll use the browser for various things but here, to download eclipse)
- Eclipse Platform Runtime (this is the minimal Eclipse IDE)
- GoClipse (the Go feature enabling Go builds in Eclipse)

### 1. Install Java 8

Eclipse is a Java application (We need to install Java to program in Go!?!). The versions of Eclipse supporting GoClipse Java 8. To install Java 8 we will need to enable the Oracle PPA:

```
user@ubuntu:~$ sudo add-apt-repository ppa:webupd8team/java
[sudo] password for user:
Oracle Java (JDK) Installer (automatically downloads and installs Oracle JDK7 / JDK8 / JDK9). There are no actual Java files in this PPA.

More info (and Ubuntu installation instructions):
- for Oracle Java 7: http://www.webupd8.org/2012/01/install-oracle-java-jdk-7-in-ubuntu-via.html
- for Oracle Java 8: http://www.webupd8.org/2012/09/install-oracle-java-8-in-ubuntu-via-ppa.html

Debian installation instructions:
- Oracle Java 7: http://www.webupd8.org/2012/06/how-to-install-oracle-java-7-in-debian.html
- Oracle Java 8: http://www.webupd8.org/2014/03/how-to-install-oracle-java-8-in-debian.html

Oracle Java 9 (for both Ubuntu and Debian): http://www.webupd8.org/2015/02/install-oracle-java-9-in-ubuntu-linux.html

For JDK9, the PPA uses standard builds from: https://jdk9.java.net/download/ (and not the Jigsaw builds!).

Important!!! For now, you should continue to use Java 8 because Oracle Java 9 is available as an early access release! You should only use Oracle Java 9 if you explicitly need it, because it may contain bugs and it might not include the latest security patches! Also, some Java options were removed in JDK9, so you may encounter issues with various Java apps. More information and installation instructions (Ubuntu / Linux Mint / Debian): http://www.webupd8.org/2015/02/install-oracle-java-9-in-ubuntu-linux.html
More info: https://launchpad.net/~webupd8team/+archive/ubuntu/java
Press [ENTER] to continue or ctrl-c to cancel adding it

gpg: keyring `/tmp/tmpunexfrxk/secring.gpg' created
gpg: keyring `/tmp/tmpunexfrxk/pubring.gpg' created
gpg: requesting key EEA14886 from hkp server keyserver.ubuntu.com
gpg: /tmp/tmpunexfrxk/trustdb.gpg: trustdb created
gpg: key EEA14886: public key "Launchpad VLC" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:      imported: 1 (RSA: 1)
OK
user@ubuntu:~$
```

Now that we have added the Oracle repository we need to update the package indexes:

```
user@ubuntu:~$ sudo apt-get update
Get:1 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Hit:2 http://ppa.launchpad.net/longsleep/golang-backports/ubuntu xenial InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu xenial InRelease
Get:4 http://ppa.launchpad.net/webupd8team/java/ubuntu xenial InRelease [17.6 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Get:6 http://ppa.launchpad.net/webupd8team/java/ubuntu xenial/main amd64 Packages [2,864 B]
Get:7 http://ppa.launchpad.net/webupd8team/java/ubuntu xenial/main i386 Packages [2,864 B]
```

```
Get:8 http://ppa.launchpad.net/webupd8team/java/ubuntu xenial/main Translation-en [1,260 B]
Get:9 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Fetched 331 kB in 1s (267 kB/s)
Reading package lists... Done
user@ubuntu:~$
```

Now we can install Java 8 (this is a 180MB download):

```
user@ubuntu:~$ sudo apt-get install oracle-java8-installer

...

Preparing to unpack .../oracle-java8-set-default_8u121-1~webupd8~2_all.deb ...
Unpacking oracle-java8-set-default (8u121-1~webupd8~2) ...
Selecting previously unselected package gsfonst-x11.
Preparing to unpack .../gsfonst-x11_0.24_all.deb ...
Unpacking gsfonst-x11 (0.24) ...
Processing triggers for fontconfig (2.11.94-0ubuntu1.1) ...
Setting up oracle-java8-set-default (8u121-1~webupd8~2) ...
Setting up gsfonst-x11 (0.24) ...

user@ubuntu:~$
```

You will need to accept Oracle's terms to complete the install. When the install completes check the Java version:

```
user@ubuntu:~$ java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)

user@ubuntu:~$
```

If you get Java 1.8 or higher, all is well.

## 2. Install Firefox

The GoClipse documentation is web based and it is easiest to install GoClipse using a browser. We'll install the Firefox web browser:

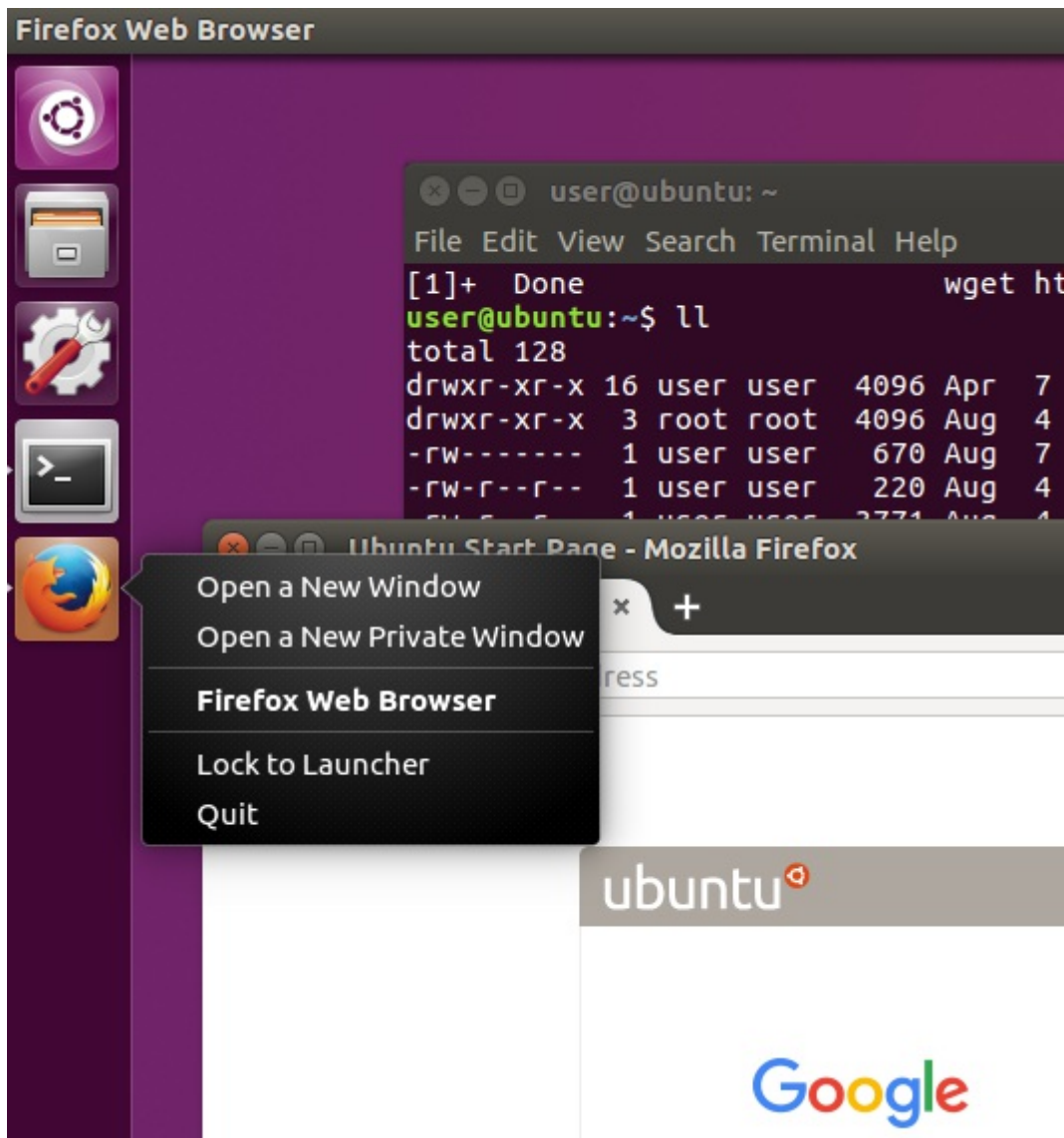
```
user@ubuntu:~$ sudo apt install firefox
[sudo] password for user:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libdbusmenu-gtk4 xul-ext-ubufox
Suggested packages:
  fonts-lyx
The following NEW packages will be installed:
  firefox libdbusmenu-gtk4 xul-ext-ubufox
0 upgraded, 3 newly installed, 0 to remove and 179 not upgraded.
Need to get 46.9 MB of archives.
After this operation, 114 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 firefox amd64 52.0.2+build1-0ubuntu0.16.04.1 [46.8 MB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libdbusmenu-gtk4 amd64 16.04.1+16.04.20160927-0ubuntu1 [26.5 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 xul-ext-ubufox all 3.2-0ubuntu1 [35.2 kB]
Fetched 46.9 MB in 1min 39s (471 kB/s)
Selecting previously unselected package firefox.
(Reading database ... 130183 files and directories currently installed.)
Preparing to unpack .../firefox_52.0.2+build1-0ubuntu0.16.04.1_amd64.deb ...
Unpacking firefox (52.0.2+build1-0ubuntu0.16.04.1) ...
Selecting previously unselected package libdbusmenu-gtk4:amd64.
Preparing to unpack .../libdbusmenu-gtk4_16.04.1+16.04.20160927-0ubuntu1_amd64.deb ...
Unpacking libdbusmenu-gtk4:amd64 (16.04.1+16.04.20160927-0ubuntu1) ...
Selecting previously unselected package xul-ext-ubufox.
Preparing to unpack .../xul-ext-ubufox_3.2-0ubuntu1_all.deb ...
Unpacking xul-ext-ubufox (3.2-0ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for gnome-menus (3.13.3-6ubuntu3.1) ...
Processing triggers for desktop-file-utils (0.22-1ubuntu5) ...
Processing triggers for bamfdaemon (0.5.3~b2r0+16.04.20160701-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for mime-support (3.59ubuntu1) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
Setting up firefox (52.0.2+build1-0ubuntu0.16.04.1) ...
update-alternatives: using /usr/bin/firefox to provide /usr/bin/gnome-www-browser (gnome-www-browser) in auto mode
```

```
update-alternatives: using /usr/bin/firefox to provide /usr/bin/x-www-browser (x-www-browser) in auto mode
Please restart all running instances of firefox, or you will experience problems.
Setting up libdbusmenu-gtk4:amd64 (16.04.1+16.04.20160927-0ubuntu1) ...
Setting up xul-ext-ubufox (3.2-0ubuntu1) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...

user@ubuntu:~$
```

Run firefox and lock it to the launcher.

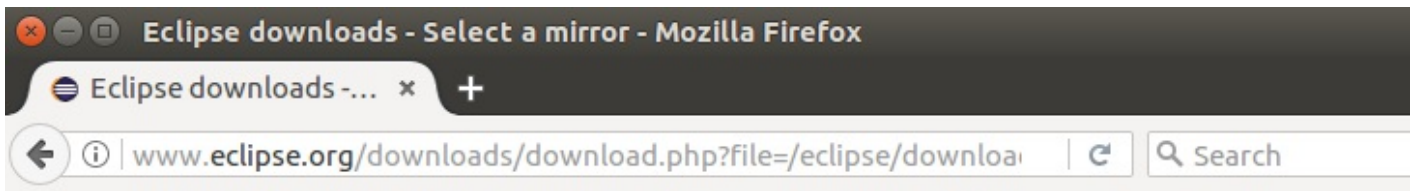
```
user@ubuntu:~$ firefox &
[1] 56467
user@ubuntu:~$
```



### 3. Download Eclipse

Now we're ready to download Eclipse. Since we will be using only Go we can download the Platform Runtime Binary. The Platform Runtime contains only the Eclipse Platform with user documentation and no source and no programmer documentation. The Java development tools and Plug-in Development Environment are not included, reducing the Eclipse foot print. We can use the Platform Runtime package as a minimal base for our Go development environment.

Point Firefox at the platform download URL for 64 bit Linux: [http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops4/R-4.6.3-201703010400/eclipse-platform-4.6.3-linux-gtk-x86\\_64.tar.gz](http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops4/R-4.6.3-201703010400/eclipse-platform-4.6.3-linux-gtk-x86_64.tar.gz)



[GETTING STARTED](#) [MEMBERS](#) [PROJECTS](#) [MORE ▾](#)

[HOME](#) / [DOWNLOADS](#) / [ECLIPSE DOWNLOADS - SELECT A MIRROR](#)

All downloads are provided under the terms and conditions of the **Eclipse Foundation Software User Agreement** unless otherwise specified.

 **DOWNLOAD**

**Download from:** Canada - University of Waterloo Computer Science Club ([http](http://www.uwaterloo.ca/~csclub/))

**File:** [eclipse-platform-4.6.3-linux-gtk-x86\\_64.tar.gz](#) **SHA-512**

**>> Select Another Mirror**

Click the download button to install the eclipse installer. Click "OK" to save the download file.

When the download completes uncompress the installer:

```
user@ubuntu:~$ ls -l Downloads/
total 69168
-rw-rw-r-- 1 user user 70824502 Apr  7 22:36 eclipse-platform-4.6.3-linux-gtk-x86_64.tar.gz

user@ubuntu:~$ tar -zxf Downloads/eclipse-platform-4.6.3-linux-gtk-x86_64.tar.gz

user@ubuntu:~$ ls -l
total 44
drwxr-xr-x 2 user user 4096 Aug  4 2016 Desktop
drwxr-xr-x 2 user user 4096 Aug  4 2016 Documents
drwxr-xr-x 2 user user 4096 Apr  7 22:36 Downloads
drwxrwxr-x 8 user user 4096 Mar  1 03:01 eclipse
-rw-rw-r-- 1 user user  877 Apr  7 22:19 eclipse.tar.gz
drwxrwxr-x 3 user user 4096 Apr  7 20:58 go
drwxr-xr-x 2 user user 4096 Aug  4 2016 Music
drwxr-xr-x 2 user user 4096 Aug  4 2016 Pictures
drwxr-xr-x 2 user user 4096 Aug  4 2016 Public
drwxr-xr-x 2 user user 4096 Aug  4 2016 Templates
drwxr-xr-x 2 user user 4096 Aug  4 2016 Videos

user@ubuntu:~$ ls -l eclipse
total 316
-rw-rw-r-- 1 user user  49706 Mar  1 03:01 artifacts.xml
drwxrwxr-x 4 user user  4096 Mar  1 03:01 configuration
drwxrwxr-x 2 user user  4096 Mar  1 03:01 dropins
-rwxr-xr-x 1 user user 80385 Mar  1 03:00 eclipse
-rw-rw-r-- 1 user user   316 Mar  1 03:01 eclipse.ini
drwxrwxr-x 18 user user  4096 Mar  1 03:01 features
-rwxr-xr-x 1 user user 140566 Mar  1 03:00 icon.xpm
drwxrwxr-x 4 user user  4096 Mar  1 03:01 p2
```

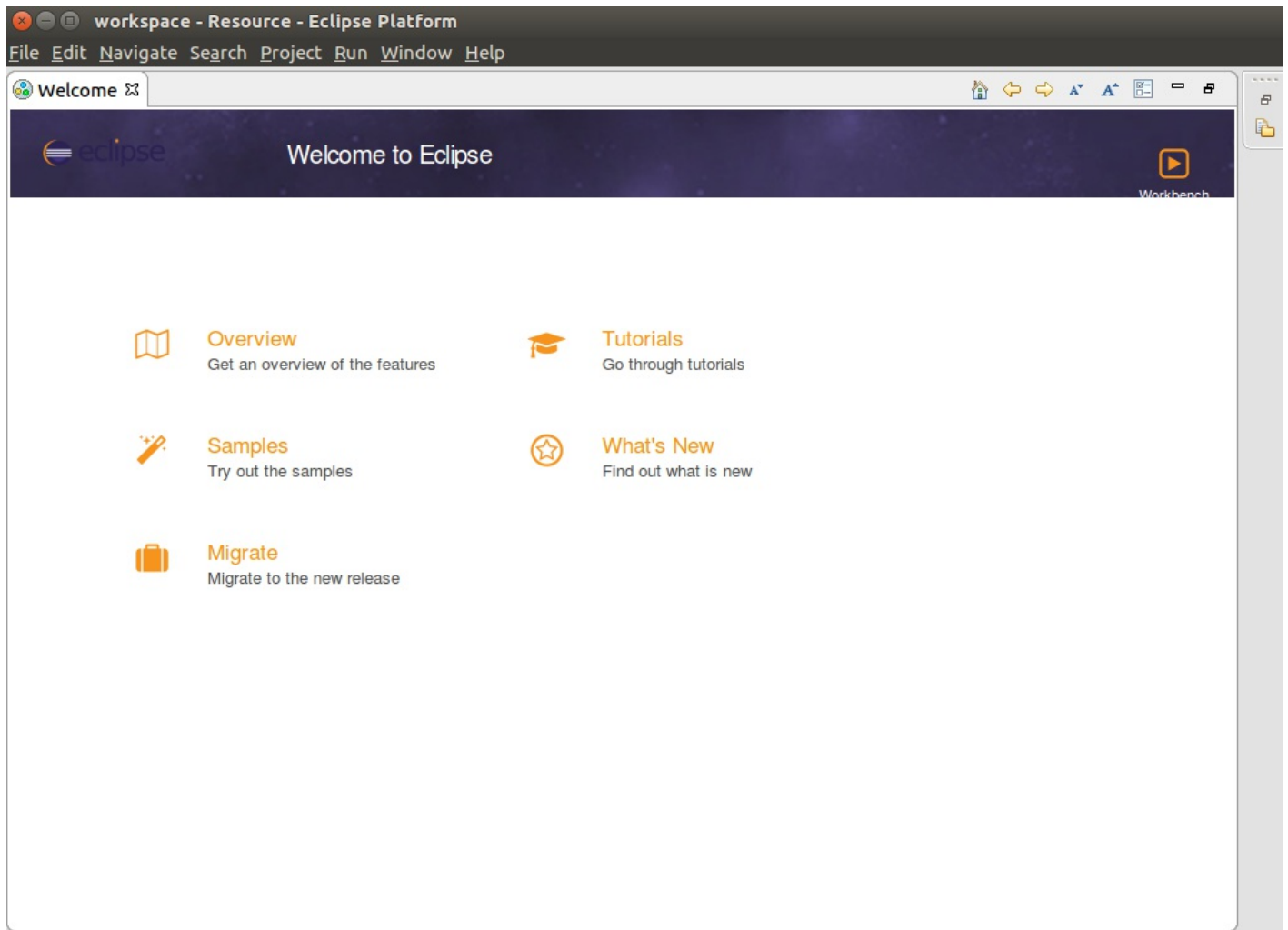
```
drwxrwxr-x  6 user user  20480 Mar  1 03:01 plugins
drwxrwxr-x  2 user user   4096 Mar  1 03:01 readme

user@ubuntu:~$
```

This places the Eclipse binaries in the ~/eclipse directory. Test the installation by running Eclipse:

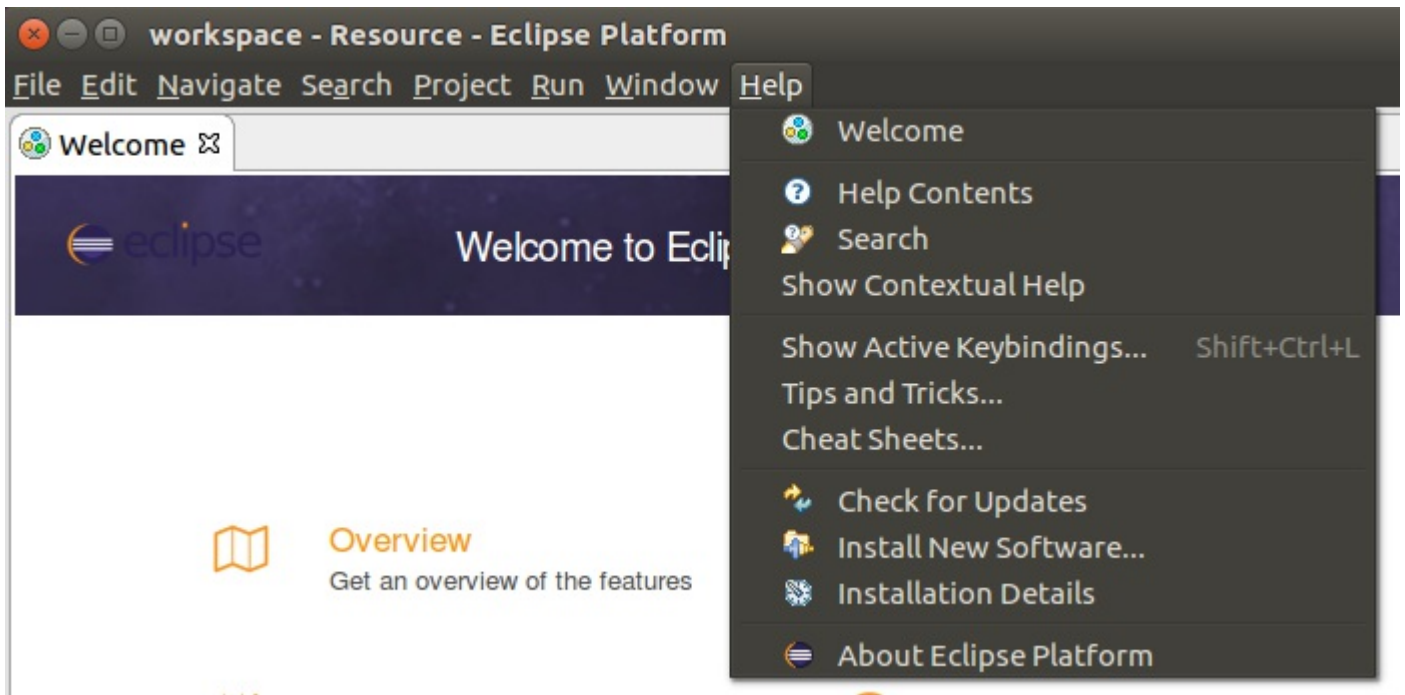
```
user@ubuntu:~$ eclipse/eclipse &
[1] 56467
user@ubuntu:~$
```

When Eclipse first launches it will ask you to confirm the default workspace directory. Set the workspace to "/home/user/go/src" and check "Use this as the default and do not ask again." and click "OK".

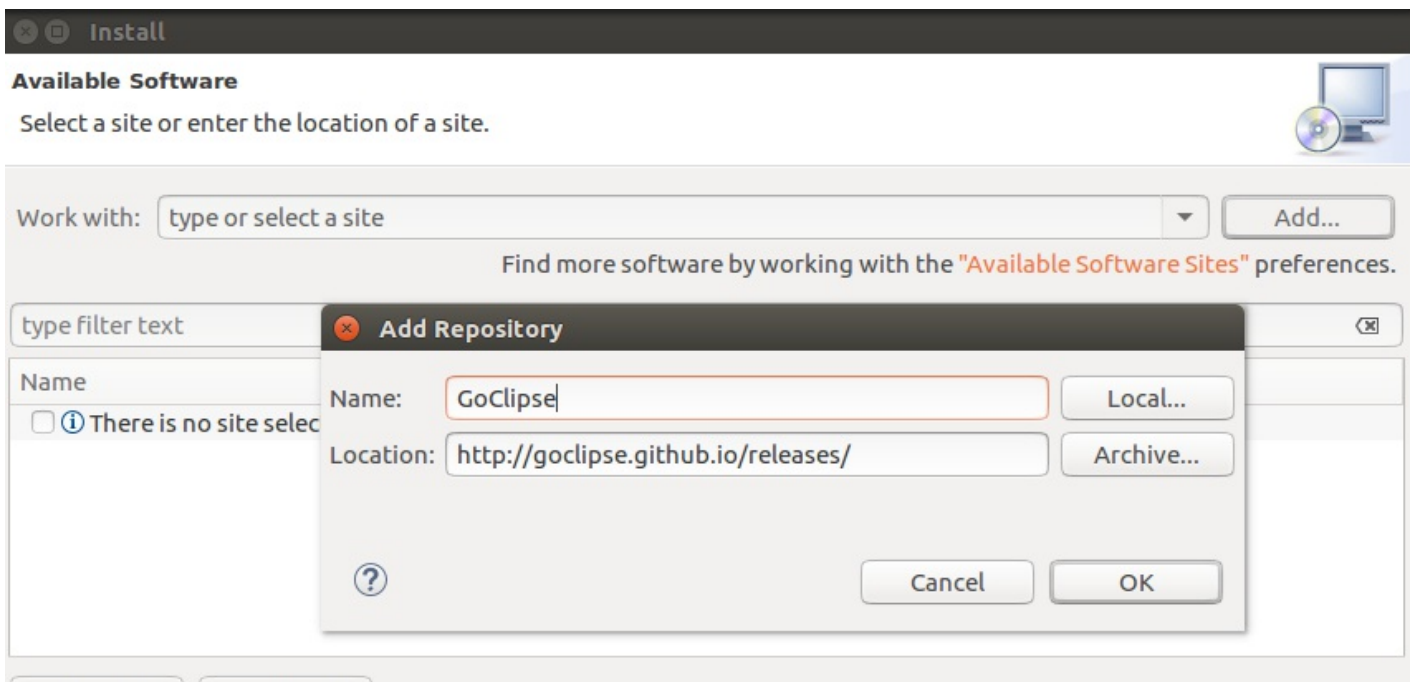


## 4. Configure GoClipse

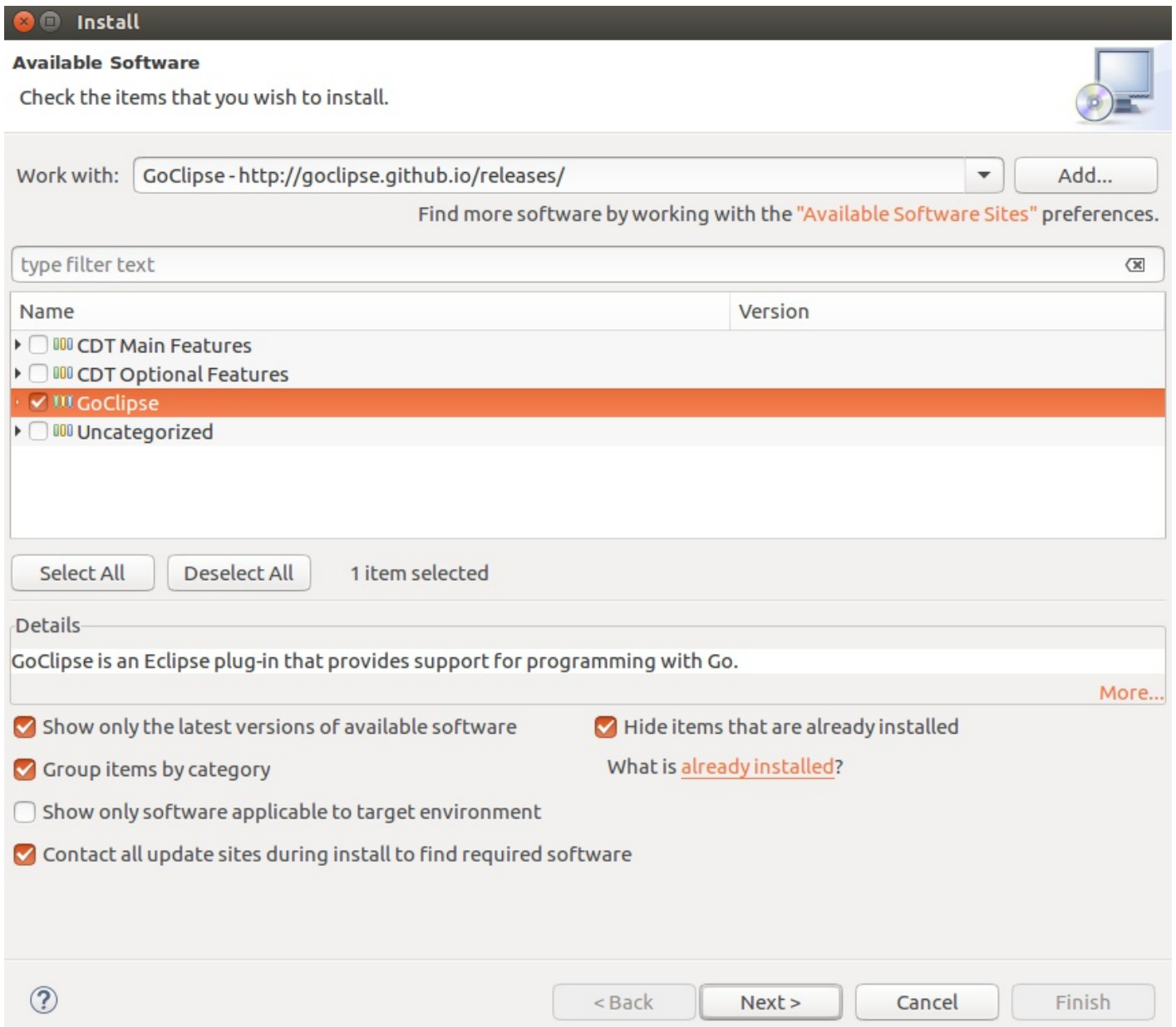
At last we can configure GoClipse for Go development. Start Eclipse, and choose: "[Help] -> Install New Software..."



Click the "Add..." button, then enter the Update Site URL: <http://goclipse.github.io/releases/> in the Location field, then click OK.



Select "recently added update site" in the "Work with:" dropdown. Type GoClipse in the filter box. The Goclipse feature should appear below. Select the GoClipse feature, make sure "Contact all update sites during install to find required software" is enabled.

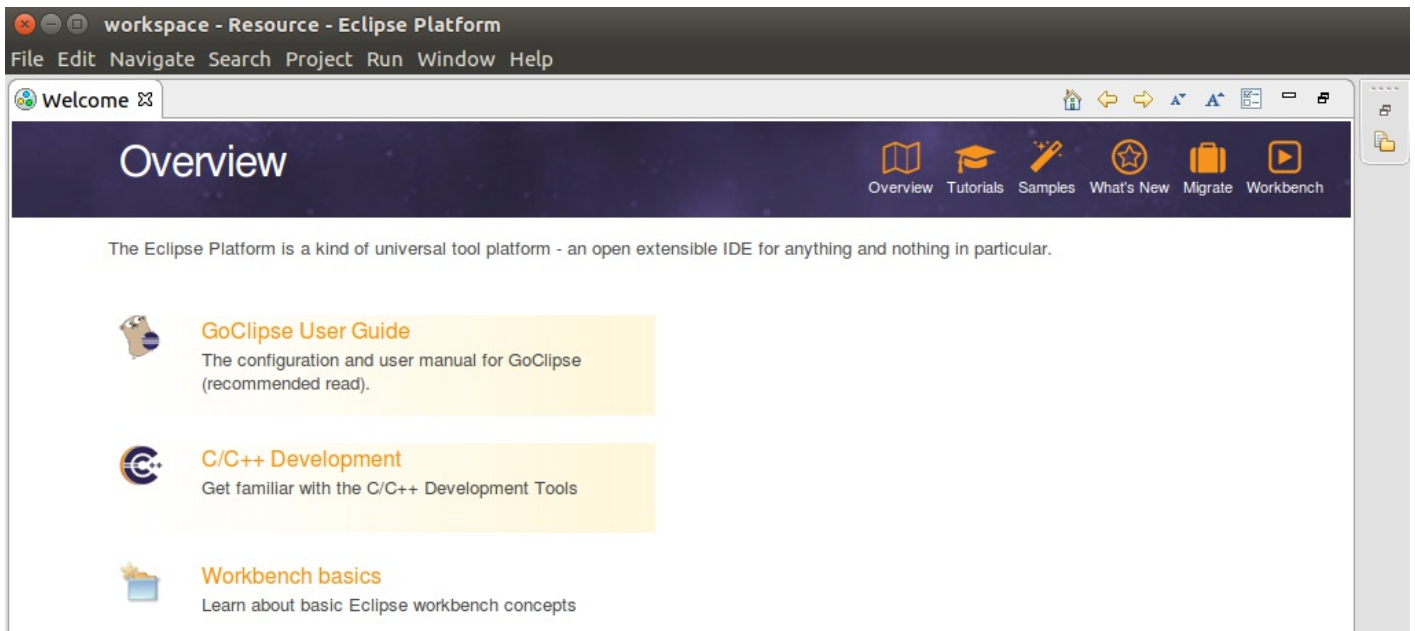


Click "Next" to move to the "Install Details" page, then click "Next" again to move to the "Review License" page. Accept the License and click "Finish". When the install finishes confirm that you trust the Bruno Medeiros certificate.

When asked if you'd like to restart Eclipse click "Yes".

When Eclipse restarts you should see the GoClipse User Guide item on the desktop.





A C/C++ development item has also been added to the desktop. GoClipse depends on Eclipse CDT (the C/C++ development environment) and automatically installed it.

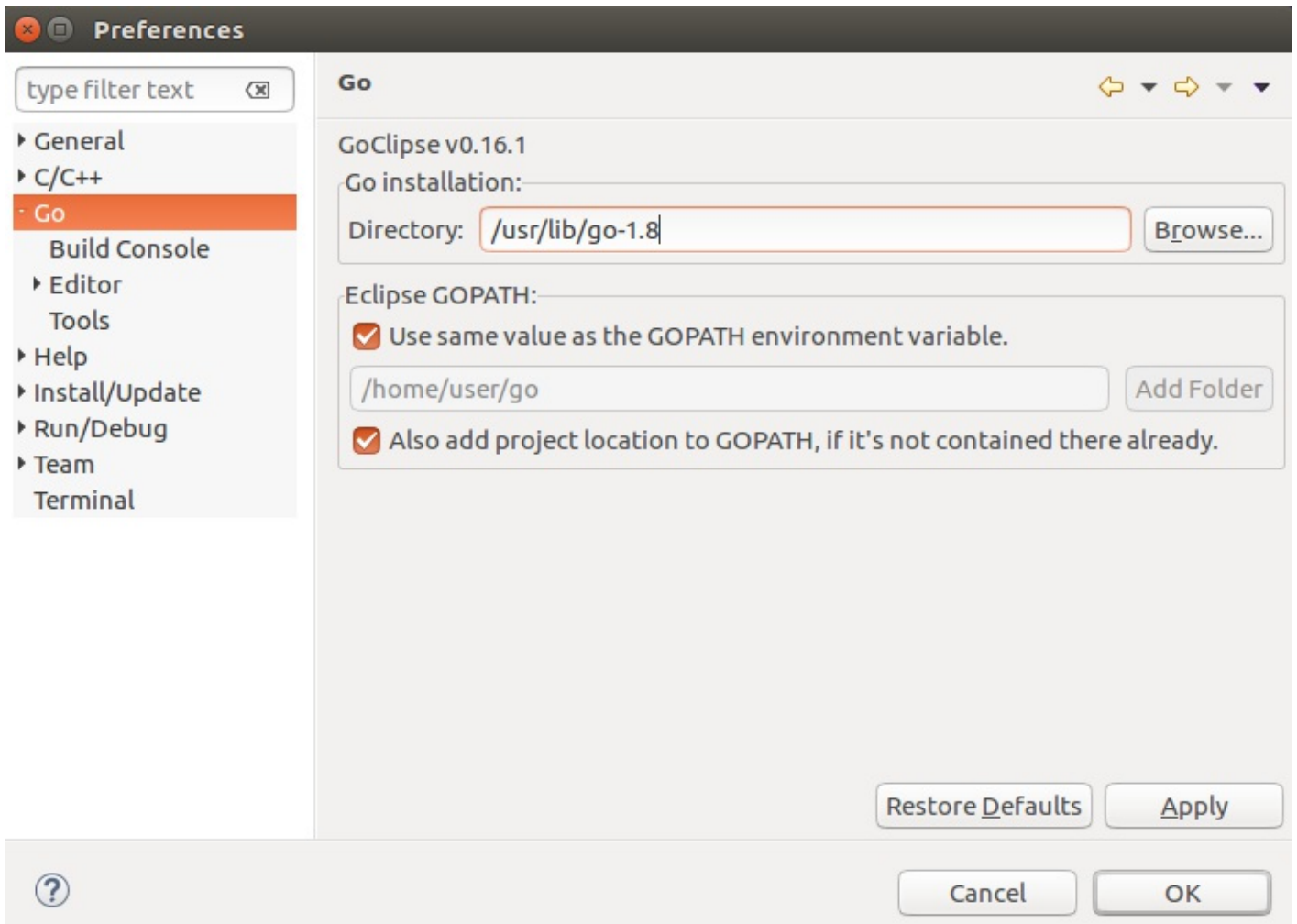
## 5. Configure GoClipse

Close any running instances of Eclipse and rerun Eclipse with the proper Go environment variables set:

```
user@ubuntu:~$ cd
user@ubuntu:~$ export GOPATH="$HOME/go"
user@ubuntu:~$ export GOROOT="/usr/lib/go-1.8"
user@ubuntu:~$ eclipse/eclipse &
[1] 61601
user@ubuntu:~$
```

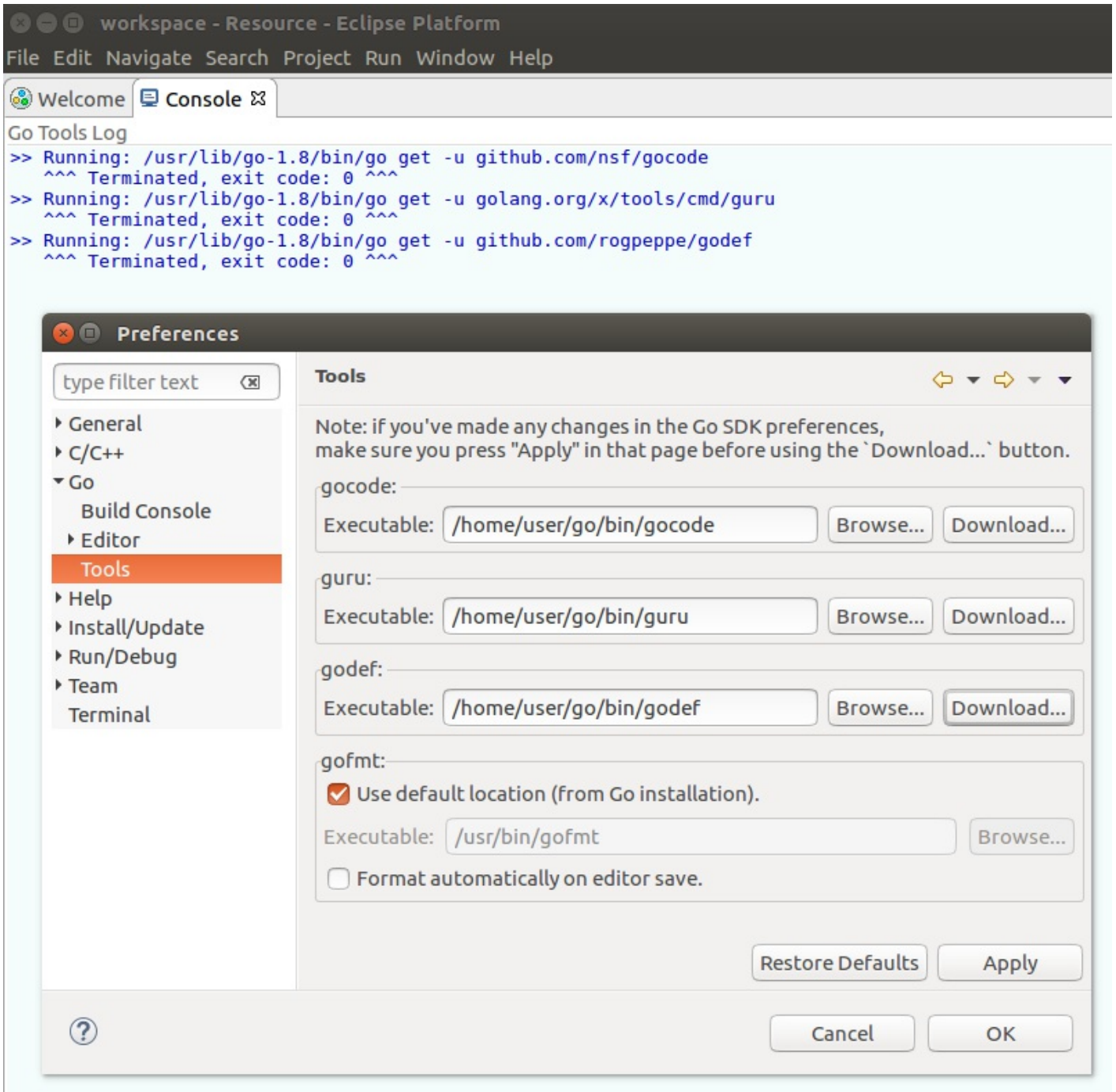
To configure GoClipse for use, open the [Windows]->Preferences dialog and select the Go item in the left hand side navigation pane.





Enter "/usr/lib/go-1.8" in the Go Installation Directory text box and click "Apply".

Now select the Go->Tools item in the left hand navigation and download each of the support tools for GoClipse. This will allow GoClipse to provide the maximum in editor support.



Click ok to complete the configuration.

## 6. Build a simple program with GoClipse

Choose [File]->New->Project to create a new project. From the New Project dialog "Go" folder select "Go Project". Click Next and from the "New Go Project" dialog enter a name for your project (like "example").

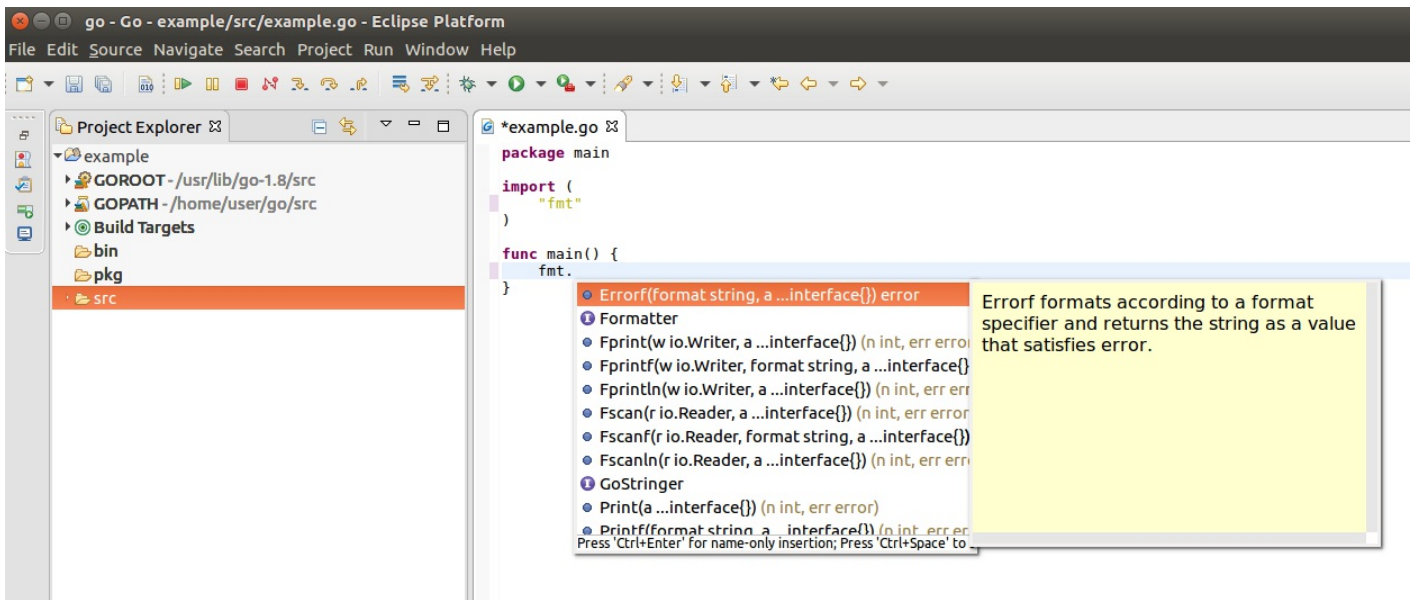
**IMPORTANT** Make sure your project "Location" is on the GOPATH under src. For example the Location in this example should be "/home/user/go/src".

Click "Finish" and click "Yes" when prompted to switch to the Go Perspective.

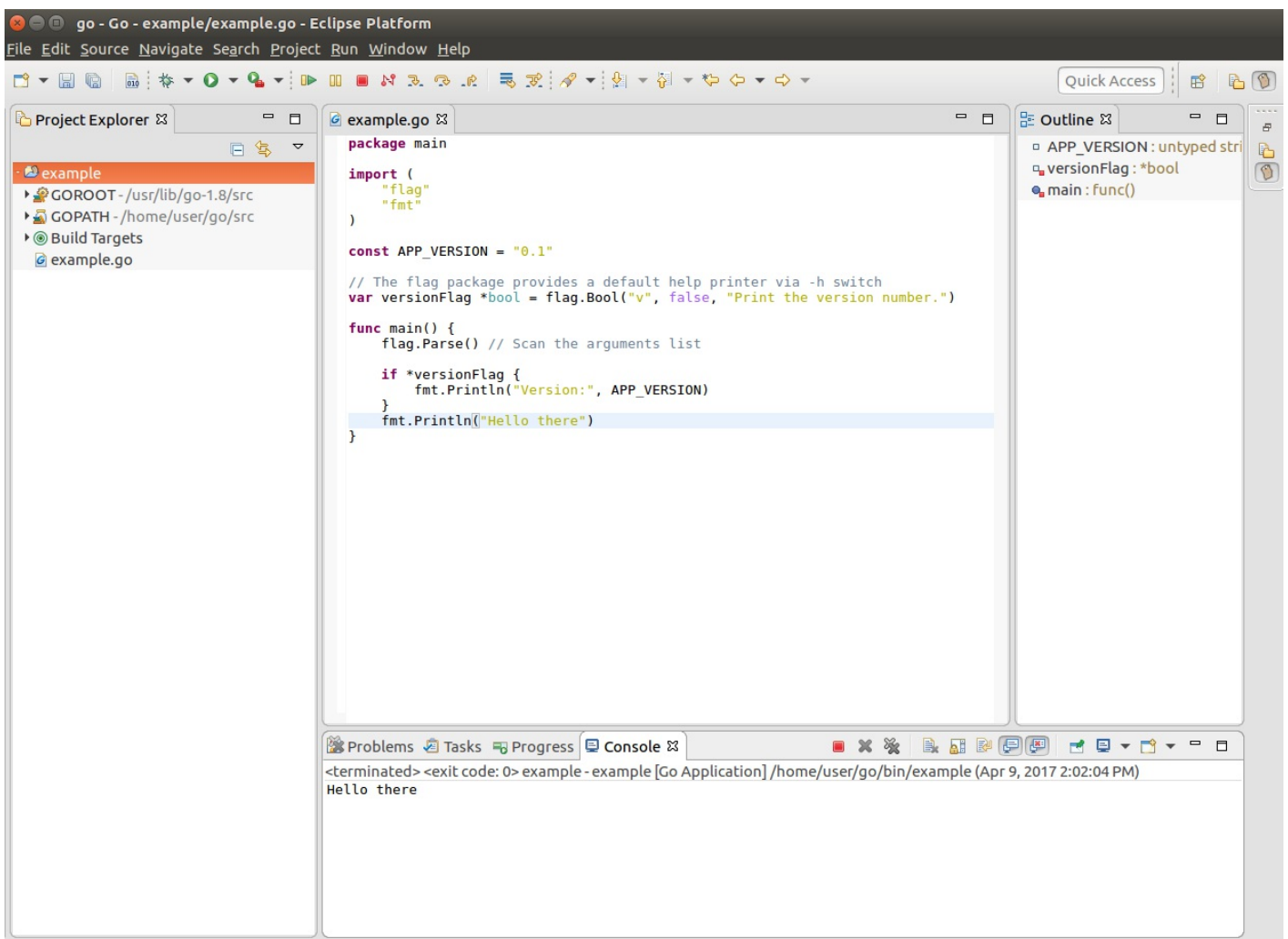
If you the Go project layout does not immediatly appear, click the little Gopher at the top right of the Eclipse window and resize as necessary.

Choose [File]->New->Go File. In the dialog enter the name example.go and choose to use the Command Source File type.

Create a hello world program. Eclipse should now be providing autocomplete, error/warning and tool tips.



Click the Green triangle Run button in the top tool bar. You should see your output in the Console tab of the bottom view.



## 7. Create a second project with GoClipse

Create a new project with GoClipse and enter the following program:

```
package main

import "fmt"

func main() {
    var a string = "initial"
    fmt.Println(a)
}
```

```
var b, c int = 1, 2
fmt.Println(b, c)

var d = true
fmt.Println(d)

var e int
fmt.Println(e)

f := "short"
fmt.Println(f)
}
```

Note the output you expect.

Run the program and compare the actual results to your expectations.

Congratulations you have completed the lab!!

*Copyright (c) 2013-2017 RX-M LLC, Cloud Native Consulting, all rights reserved*