Parallel Programming

Installing Eclipse Parallel Tools Platform (PTP) (Linux Distributions)

Author B. Wilkinson - Modification date May 29, 2015

Preliminaries - Install Java

Java is needed for the Eclipse IDE. Check that Java is not already installed by issuing the command:

java -version

If not installed, install it. On Ubuntu it can be done using **apt-get** as follows:

(a) Check package repository is up-to-date

Issue the command:

sudo apt-get update

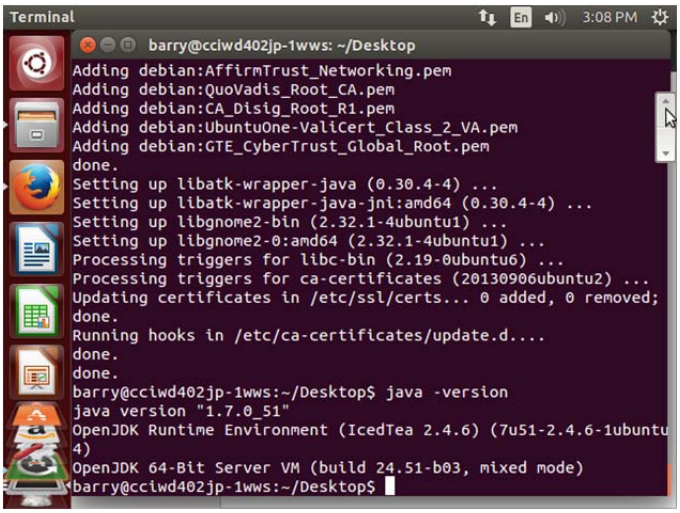
(b) Installing Java JDK

Issue the command:

sudo apt-get install default-jdk

(which will install OpenJDK not JavaJDK[1]). After the installation completes, check with:

java -version



[1] Sun Java JDK can be installed with **apt-get install sun-java6-jdk**.

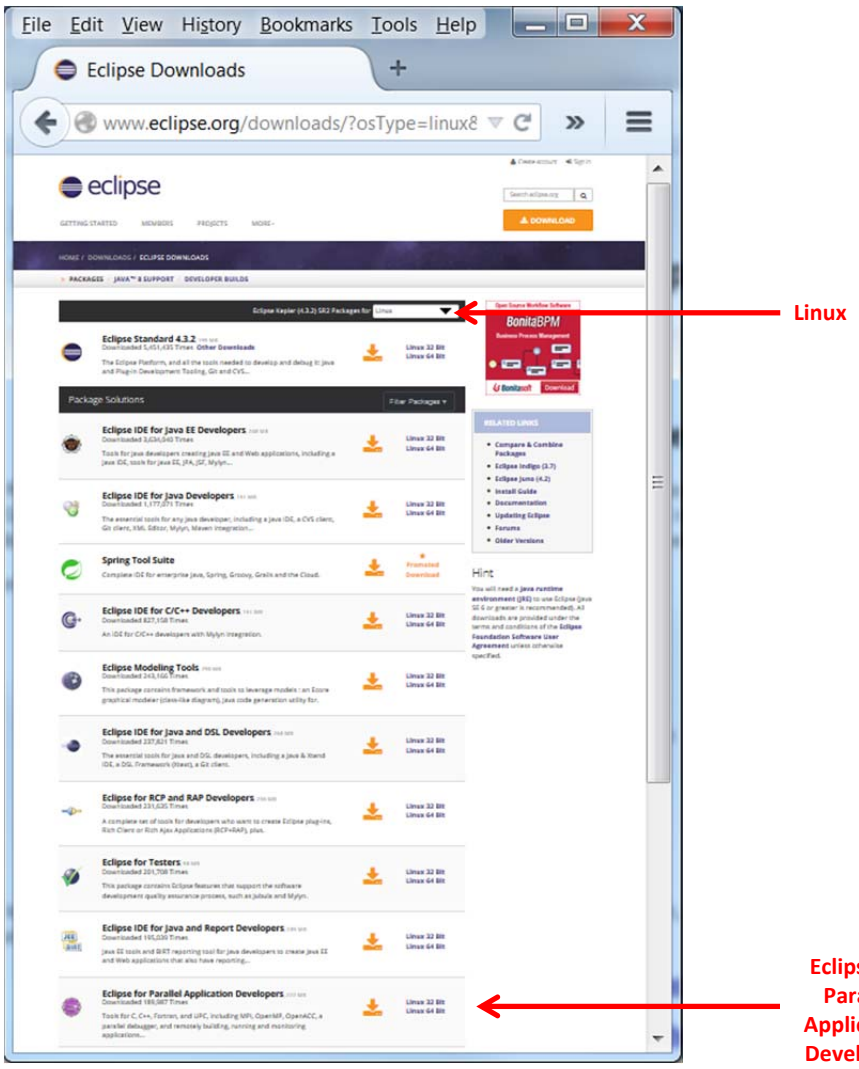
Installing Eclipse-PTP

The best way to install Eclipse-PTP is from the source, *NOT from the Ubuntu Package Repository*.[2]

(a) Download Eclipse

Go to: http://www.eclipse.org/downloads/

Choose "Eclipse for Parallel Applications Developers Linux 64 bit" and download file (**eclipse-parallel-kepler-SR2-linux-gtk-x86_64.tar.gz** or the most recent)[3]



[2] Eclipse can be installed using the Ubuntu Package Repository, e.g. **sudo apt-get install eclipse-platform**, but you will get an older unsupported version of Eclipse without PTP (Juno version 3.8). PTP then has to be installed through the Eclipse software update feature. *It is not recommended to start with an unsupported version of Eclipse.*

[3] Note "Eclipse for Parallel Applications Developers" does not include the standard Java environment. An alternative is to download and install the standard Eclipse for Java and then use the software update feature **Help > Install New Software**, select the software site, search on "parallel" to find the Parallel Tools Platform, select and update Eclipse.

(b) Uncompress Eclipse tar file

Open a terminal and type:

cd ~/Downloads

tar zxvf eclipse-parallel-kepler-SR2-linux-gtk-x86_64.tar.gz

Move the uncompressed **eclipse** folder to a suitable location, e.g. /**usr/local**:

sudo mv eclipse /usr/local

(c) Set up path to eclipse executable

Add the lines:

export PATH=/usr/local/eclipse:$PATH

to the end of the **~./profile** file. The **~/.profile** script is executed when the machine is started. To execute it now, type

source ~/.profile

Check the path with **echo**.

Using Eclipse

Start Eclipse on the command line by typing:

eclipse

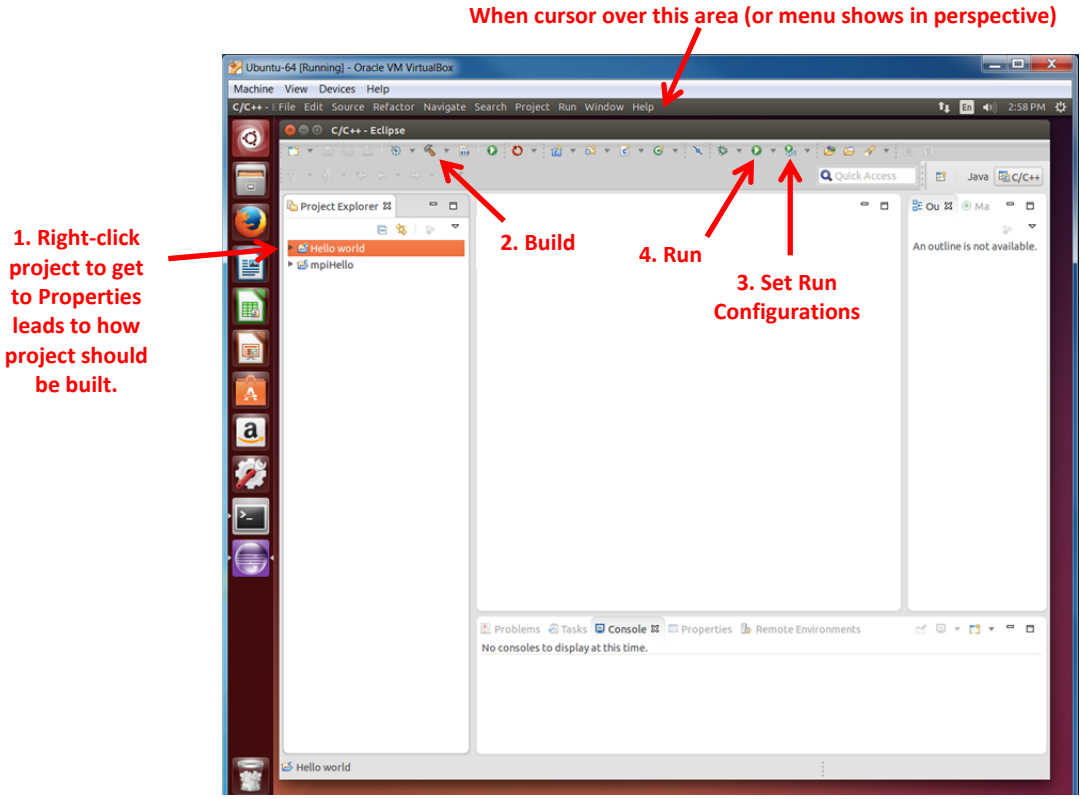
Select the default workbench location **/home/<user>/workspace** and go to the workbench. The following screenshots are from Eclipse Kepler (4.3.2) SR 2.

(a) General

There are different project types for different environments. MPI programs are done as C/C++ projects. Programs here will be C projects.

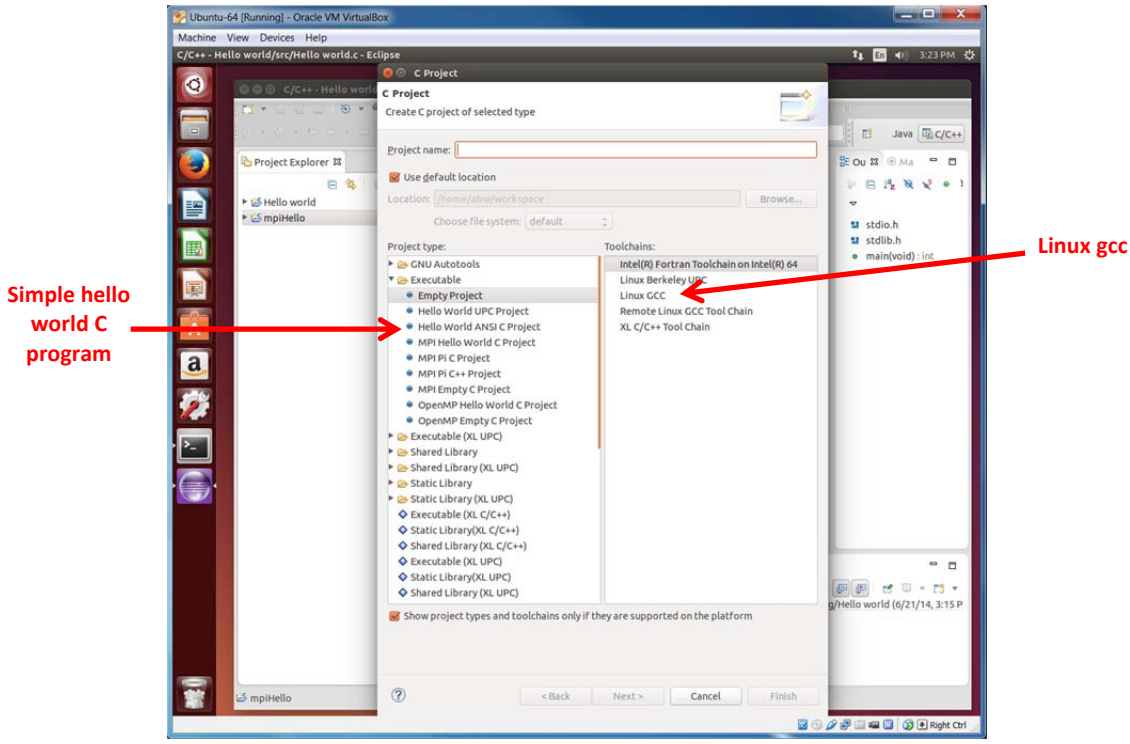
Basic steps:

1 Set how to build (compile) project in **Properties > ... Build**
2. Build project (compile to create executable)
3. Set how to execute compiled program in **Run Configurations**
4. **Run** (execute) using the specified run configurations

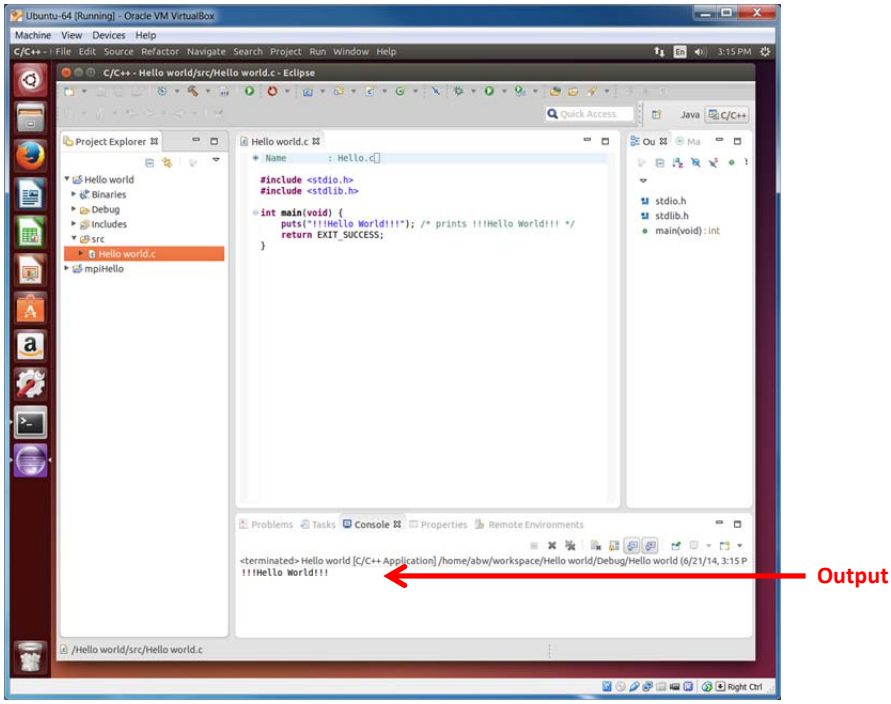


(b) C program

At this point, it is recommended to make sure you can use Eclipse with a regular C program (Hello World). Create a new C project (**File > New > C Project**), and select the "**Hello World ANSI C** Project" type and **gcc** compiler:

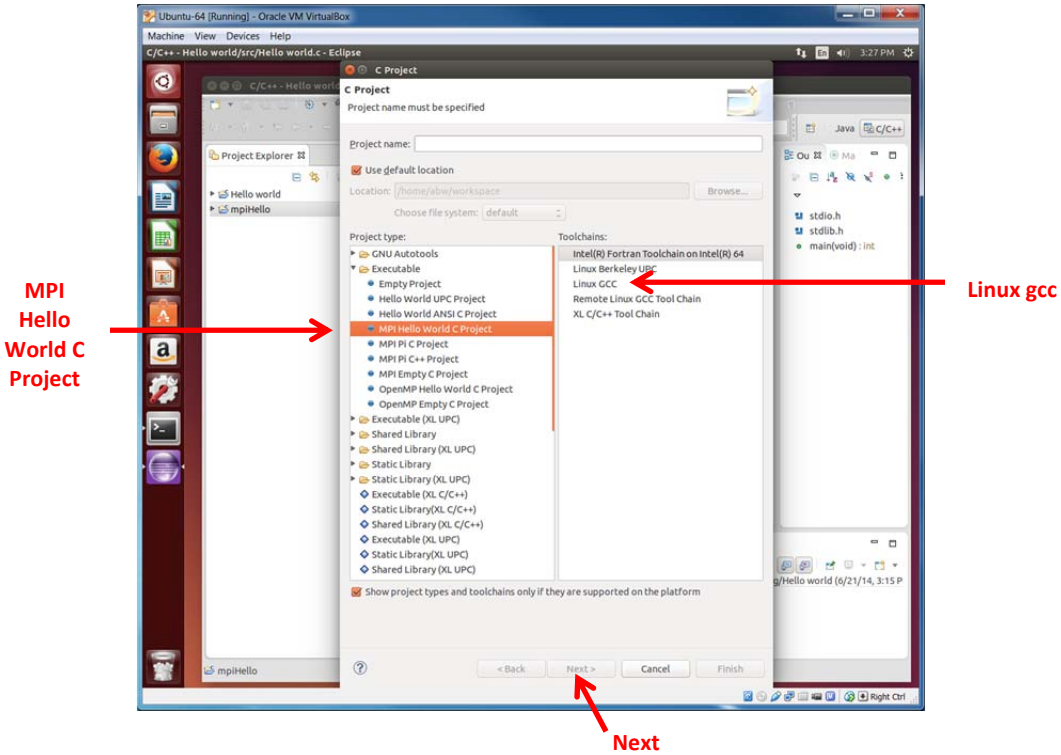


Create the “Hello world” C project, built it, and run:

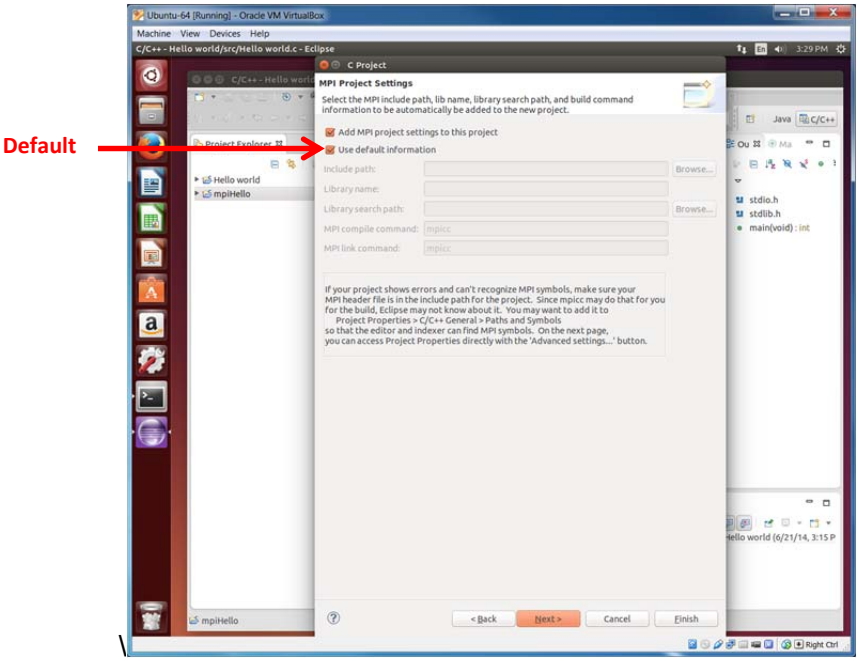


(c) MPI program

Next try a simple MPI hello program. Select the **“MPI Hello World ANSI C Project”** type and **gcc** compiler:

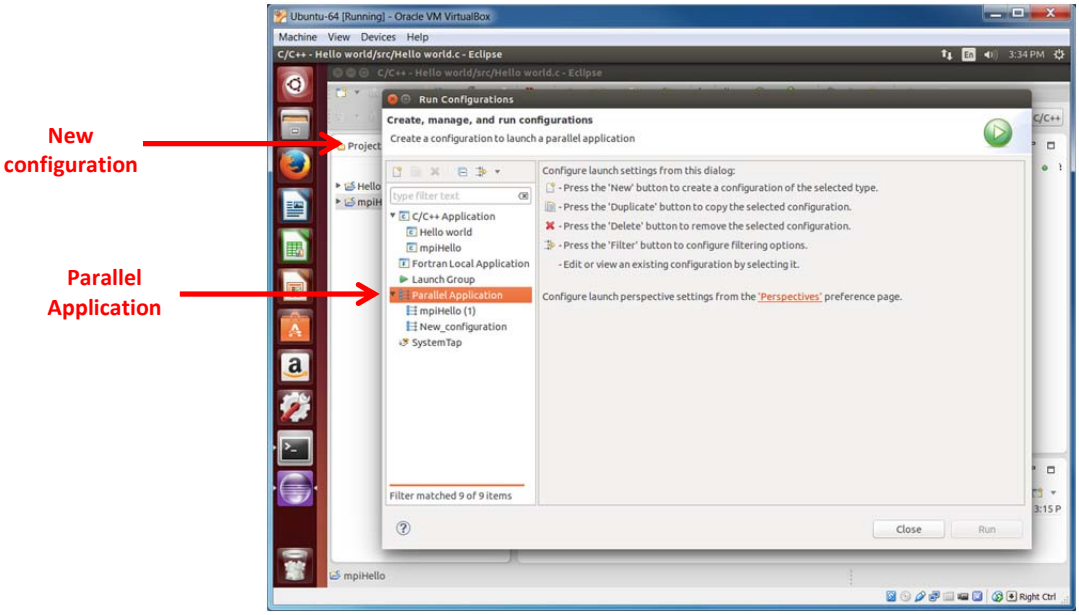


Default paths to libraries and includes should work. (In fact these are with the **mpicc** script and do not need to be changed). Note the compile/link command is **mpicc**. Eclipse will put in the appropriate command line options and file names. Compile and link are separate, which enables object files to be compiled separately and then linked, see using Suzaku later.

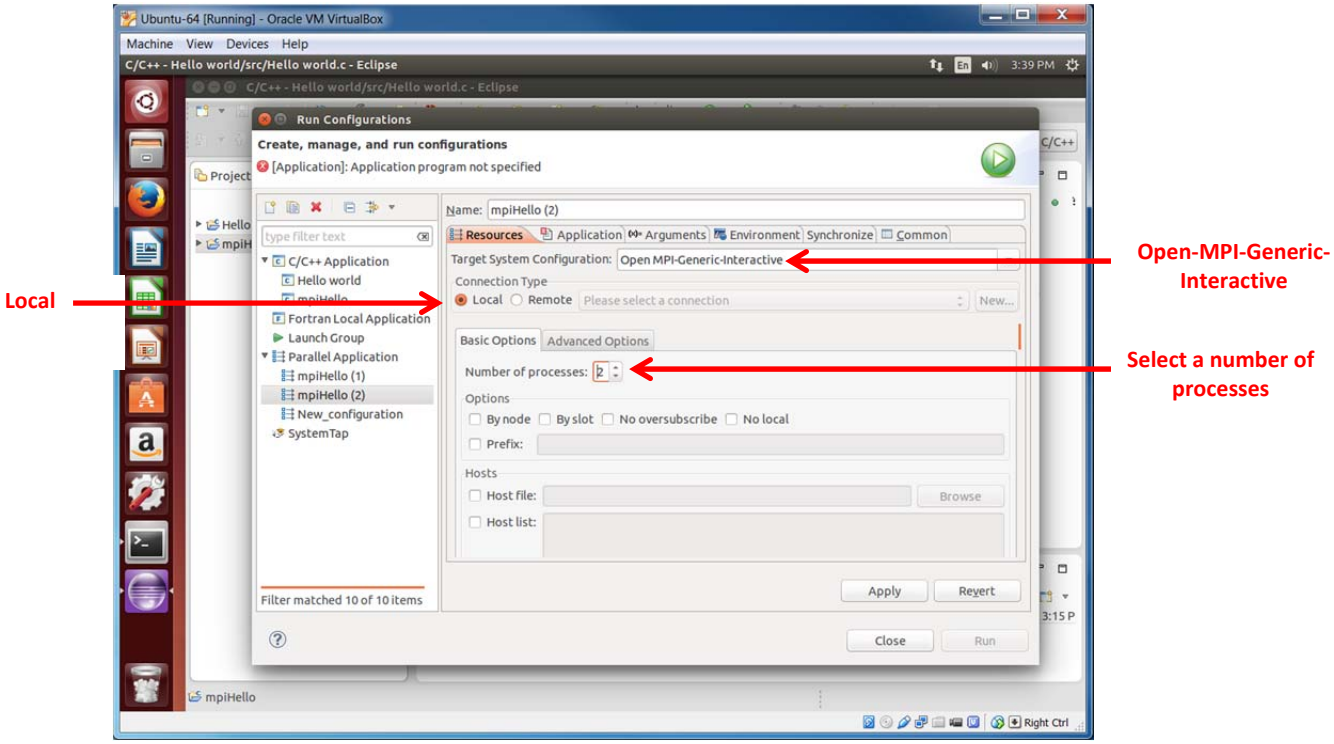


Test build with **Properties > Build Project**.

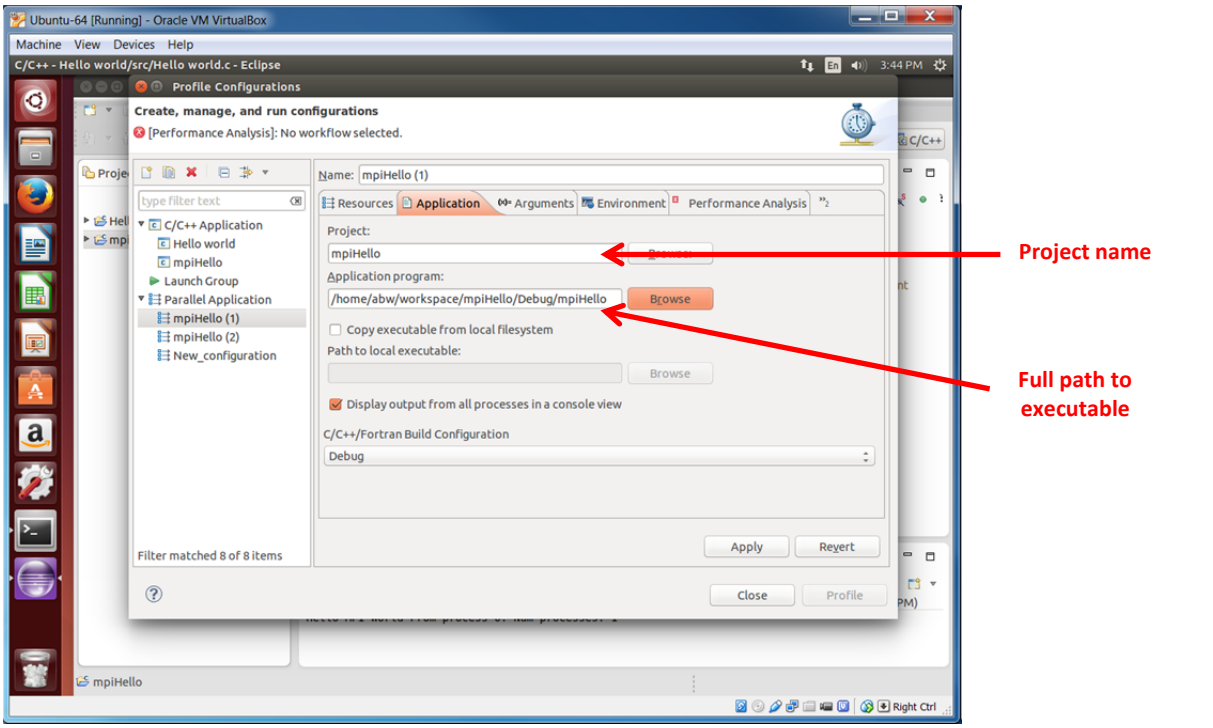
Execution. In the Run Configurations, choose a new **Parallel Application**:



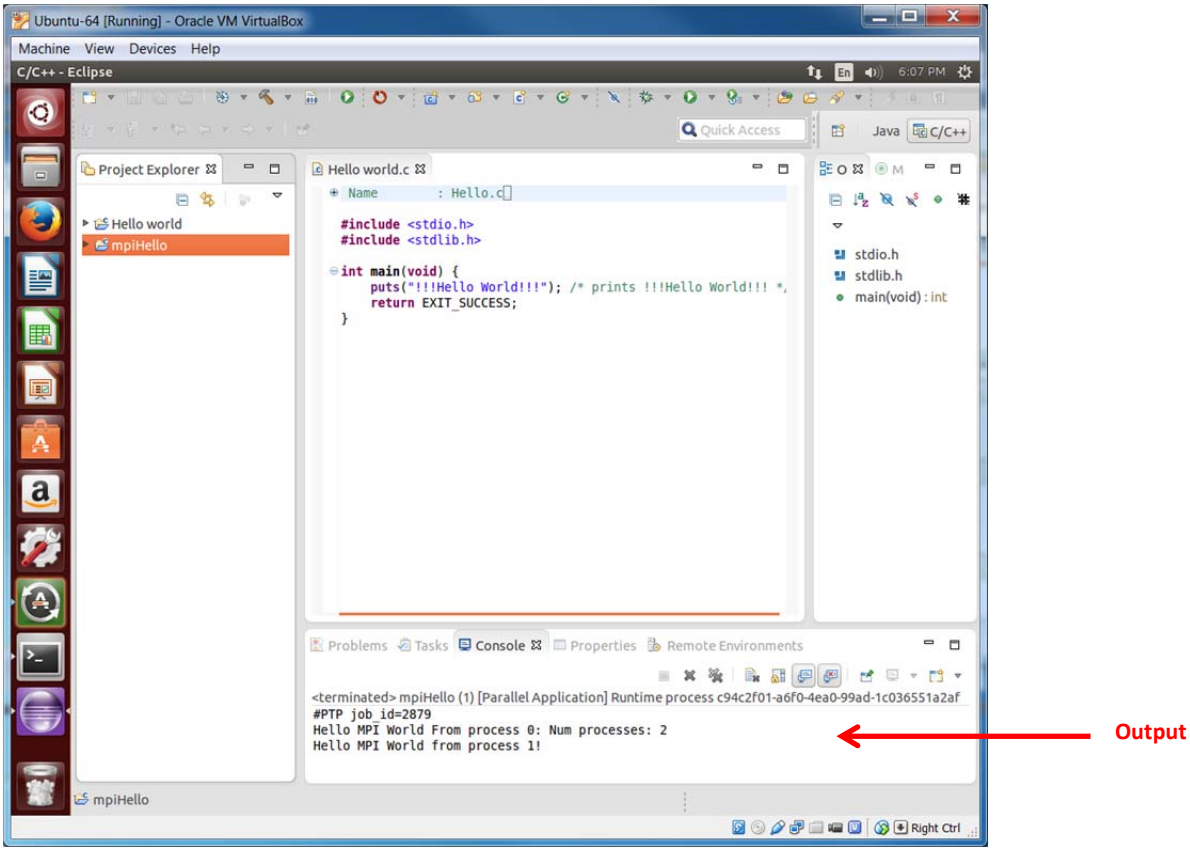
In the **Resource** tab, select Target System Configuration as **"Open-MPI-Generic-Interactive"** and connection type as "**local"**, and number of processes:



In the **Application** tab, enter project name and the full path to the executable (browse for it):



The **Run** button should now be available.[4] Run:



Updating Eclipse for a major release

To update Eclipse for a major release, for example from Kepler (version 4.3) to Luna (version 4.4), it is now not necessary to re-install Eclipse. Go to **Preferences** > **Install** > **Update** > **Available Software Sites** and update the repository URL. e.g. http://download.eclipse.org/releases/luna and install updates.

More Information

Ubuntu Documentation Eclipse IDE: https://help.ubuntu.com/community/EclipseIDE

[4] Possible reason for the RUN button to be still grayed out is the path to the executable does not exist.