Castalia-3.2 Port to OMNeT++ on Windows and Linux

INSTALLATION GUIDE

Authors: Alex Lacerda, Marcella Pinheiro

Table of Contents

1. General Information

1.1. Introduction

This document describes how to apply the `updateCastalia.sh` file. After executing this bash script, you will be able to:

1. <u>Run Castalia on Windows from the command line</u> (using the `mingwenv.cmd` command prompt available at OMNeT++ installation directory).

2. Import Castalia to the OMNeT++ IDE (on both Windows and Linux).

In addition, this document describes how to run Castalia simulations from the OMNeT++ IDE. Instructions are given for both Windows and Linux.

> **NOTE**
>
> When we run a Castalia simulation from the command prompt, two resulting files are generated. The first file is named `Castalia-Trace.txt`, it contains a trace of all events recorded by the nodes. The second file is named after the current date, in the form `YYMMDD-HHMMSS.txt`. This file is processed by the `CastaliaResults` script to generate summary results about the simulation, such as total number of messages sent, total energy spent, etc.
>
> Since the `YYMMDD-HHMMSS.txt` file is generated by the `Castalia` script, when you run a simulation from the IDE, that file will not be generated. The IDE only generates the trace of the simulation, which is redirected to the console of the IDE, instead of the `Castalia-Trace.txt` file.
>
> For that reason, the IDE should be used in conjunction with the command prompt. You can use the IDE to implement your simulations and benefit from all of the features of the IDE, such as text highlight, code completion, etc. You can also run simulations from the IDE and watch the simulation trace being written in the console as the events occur. This is very useful when you want to check if the events of your simulation are happening in the right sequence. When you have finished implementing your simulation and you want to generate summary results, you should use the command prompt, so that you can benefit from the features of `Castalia`, `CastaliaResults` and `CastaliaPlot` scripts.

1.2. Supported Versions

We have tested the modified version of *Castalia-3.2* on *OMNeT++ 4.4.1, 4.5 and 4.6,* running on the following operating systems:

- Windows 7 and 8 (32/64-bit)
- Ubuntu 14 (32/64-bit)

2. Windows

First, we need to install OMNeT++ and Castalia. Then we can apply the `updateCastalia.sh` script. Finally, we can import Castalia to the IDE or run it from the command prompt.

2.1. Installing OMNeT++

To install OMNeT++, follow its installation guide, available at https://omnetpp.org/.

2.2. **Installing Castalia-3.2**

Download the file `Castalia-3.2.tar.gz` from https://forge.nicta.com.au/frs/?group_id=301.

Copy the downloaded archive to the directory where you want to install it. Choose a directory whose full path **does not contain any space**.

Extract the tar file using a specific program such as *WinRAR*. This will create a directory named `Castalia-3.2` containing directories named `bin`, `Simulations`, `src`, and files named `CHANGES.txt`, `LICENSE`, `makemake`, etc.

Next, you must add Castalia's `bin` directory to the Path environment variable. To do so, follow the steps below:

1. Click the *start* button and begin typing *environment variables* in the search bar.
2. Select the option *Edit the system environment variables* when it appears in the search results list. This will open the *advanced* tab of the *system properties* window.
3. In that tab, click the *environment variables* button. This will open the *environment variables* window.
4. In that window, select the *Path* variable (under *System variables*) and click the *Edit…* button (see Figure 1).
5. Now, at the end of the *variable value* field, add the path to Castalia's bin directory, using a semicolon to separate it from the other paths there. For example, assuming you have extracted Castalia to `C:\Castalia-3.2`, you should add `;C:\Castalia-3.2\bin` to the *variable value* field (see Figure 1).
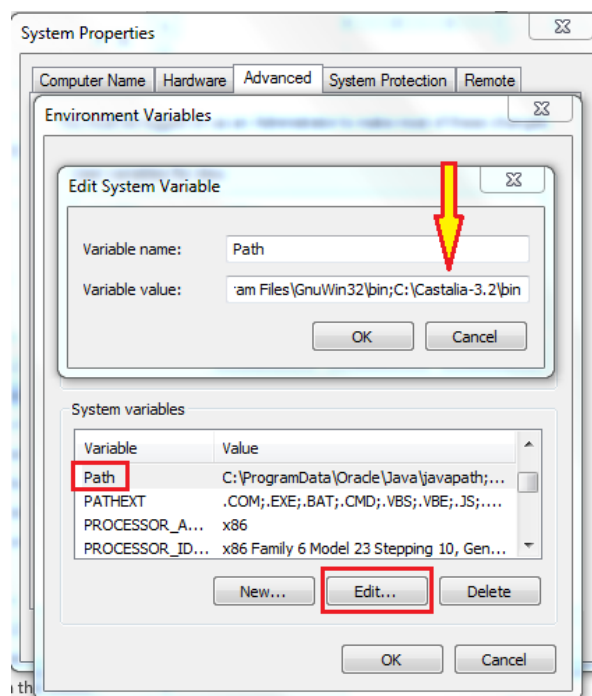6. Press **OK** button until all windows have been closed.

Figure 1. Adding Castalia to the Path

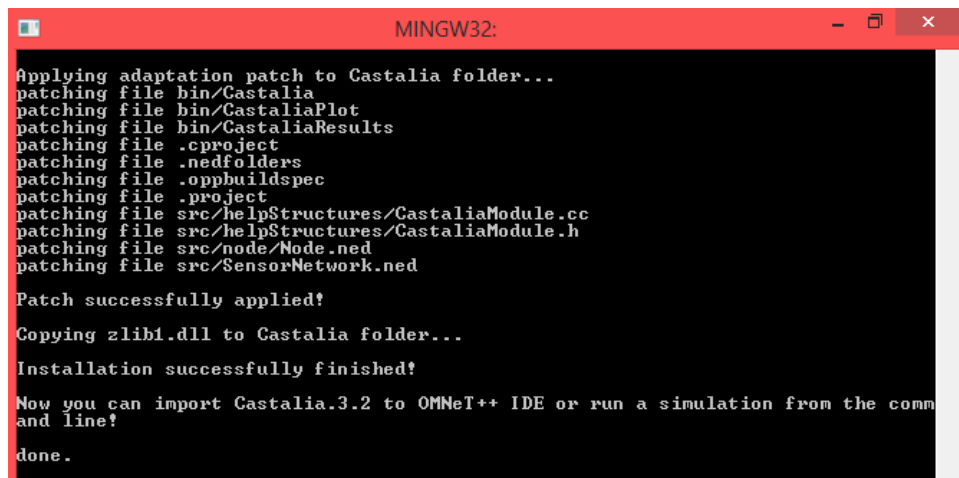Now, you are ready to apply the `updateCastalia.sh` script.

2.3. Executing the updateCastalia.sh script

Start `mingwenv.cmd` in your `omnetpp-4.x` directory by double-clicking it in Windows explorer.

Assuming you have downloaded the file `Castalia-3.2_OMNeT-IDE_Windows_Linux-master.zip` from https://github.com/alexlacerda/Castalia-3.2_OMNeT-IDE_Windows_Linux/archive/master.zip and extracted it to `C:\Castalia-3.2_OMNeT-IDE_Windows_Linux-master\`, execute the following commands to change to that directory and run the `updateCastalia.sh` script:

```
$ cd C:/Castalia-3.2_OMNeT-IDE_Windows_Linux-master
$ ./updateCastalia.sh
```

If the commands have been successfully executed, you should see a screen like the one in the figure below.

Figure 2. Successfully executing updateCastalia.sh

At this moment, you should be able to build and run Castalia simulations from the command line (`mingwenv.cmd`) just like Linux users do. To see how to do so, refer to Section 2.5.

In addition, you can import Castalia to OMNeT++ IDE and execute simulations from there. In the next section, we give you instructions on how to do this.

2.4. Running Castalia on the OMNeT++ IDE

In this section, we describe how to import Castalia to OMNeT++, build it, run simulations and view the simulation results (trace).

2.4.1. Importing Castalia-3.2 to OMNeT++ IDE

Open OMNeT++ IDE and click on *File > Import*. In the import window, expand the folder *General*, select the option *Existing Projects into Workspace* and click *Next*. Then, click the *Browse* button (see Figure 3) and select the *Castalia-3.2* installation directory. Click *Finish*.
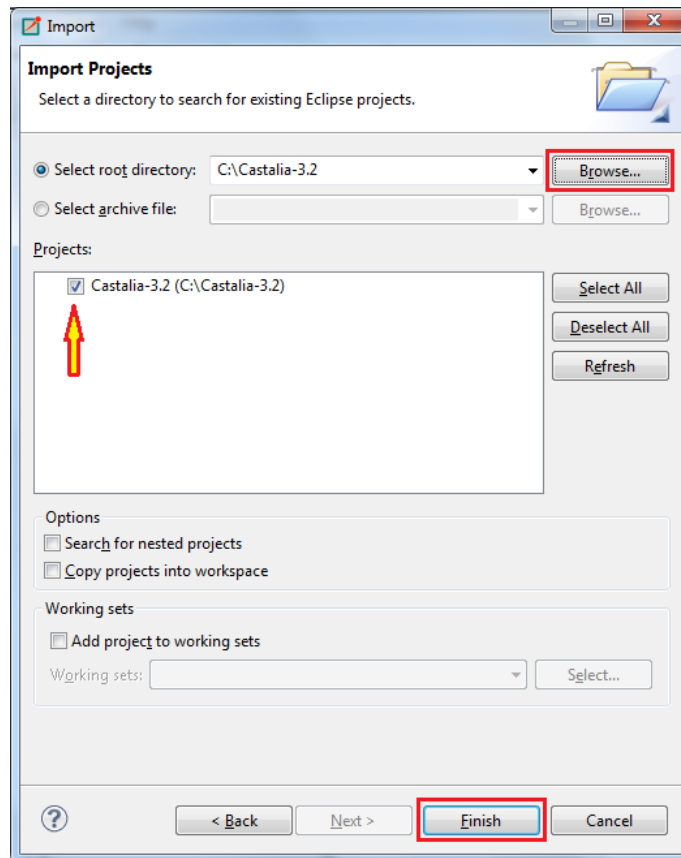
Figure 3. Importing Castalia-3.2 to OMNeT++ IDE

After that, *Castalia-3.2* project should appear in the *Project Explorer* tab (see Figure 4 below).
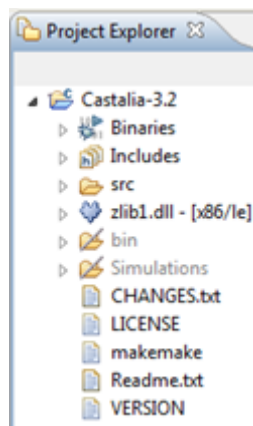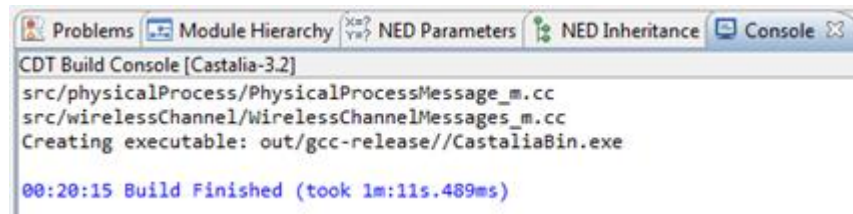

Figure 4. Castalia-3.2 in the Project Explorer

2.4.2. Building Castalia-3.2 from OMNeT++ IDE

To build Castalia, right-click *Castalia-3.2* project in the *Project Explorer* and select the option *Build Project*.

If the build process has successfully finished, your console should look like the one in the figure below.
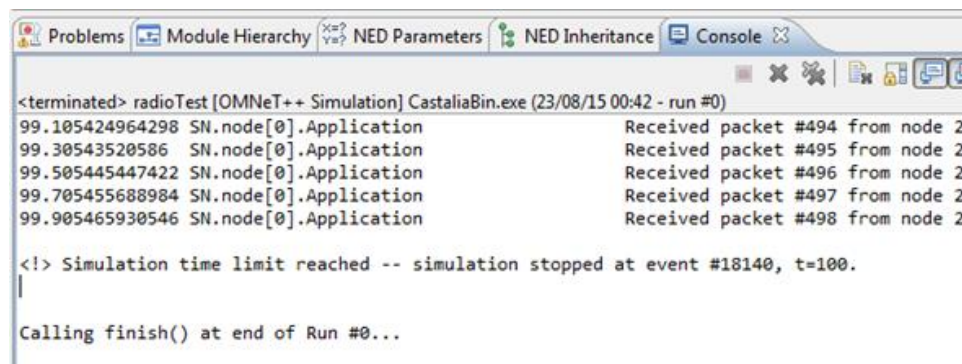
Figure 5. Castalia Build result in the console

2.4.3. **Running a Castalia Simulation from OMNeT++ IDE**

To execute the *RadioTest* example simulation, expand the `Simulations/radioTest` directory, right-click the file `omnetpp.ini` and select the option *Run As > 1 OMNeT++ Simulation*.

If everything goes well, you should be able to see the simulation trace in the console, just like in the figure below.

Figure 6. Simulation trace in the console

Now that you know how to use the IDE, you should refer to the *Castalia User's Manual* to learn how to implement your own simulations. If you need to run Castalia from the command prompt, read the next section.

2.5. **Running Castalia from the command prompt**

In Windows, to run Castalia from the command prompt, just like Linux users do, you just need to open the `mingwenv.cmd` prompt and type the commands there.

Following, we briefly show you how to use the main Castalia commands, so that you can check whether your Castalia installation have been successful.

Before running a simulation, you first need to build Castalia. Assuming you have installed Castalia in `C:\Castalia-3.2\`, open `mingwenv.cmd` and type:

```
$ cd C:/Castalia-3.2
$ make clean
$ ./makemake
$ make
```

Now that you have built Castalia, you can change to a simulation directory and run it. For example, to run the *radioTest* simulation, type the commands below (refer to *Castalia User's Manual* for more details).

```
$ cd Simulation/radioTest
$ Castalia -c General
```

This will run the simulation and generate the trace file (`Castalia-Trace.txt`) and the results file (`150823-012227.txt`, in our system). You can check whether those files have been generated by typing:

```
$ ls
```

To process the results file, you can use the `CastaliaResults` script. For example, to see the *total number of packets received by the sink node*, type the following command (refer to *Castalia User's Manual* for more details):

```
$ CastaliaResults -i 150823-012227.txt -s packets --sum
```

If you want to see the results graphically, you can use the `CastaliaPlot` script. However, you first need to download and install *Gnuplot*, then you have to add its `bin` directory to the *Path* (just like you did with Castalia's `bin` directory in Section 2.2). Do not forget to close and reopen the `mingwenv.cmd` prompt, so that the *Gnuplot* installation takes effect.

Assuming you have correctly installed *Gnuplot*, you can get back to the *radioTest* simulation directory and type the following command to generate a graph (refer to *Castalia User's Manual* for more details):

```
$ CastaliaResults -i 150823-012227.txt -s packets -n |
CastaliaPlot -o figureName.png -s stacked
```

This will generate an image named `figureName.png`, in the *radioTest* simulation directory, showing the number of packets the sink received from nodes 1 and 2. The graph generated should look like the figure below.
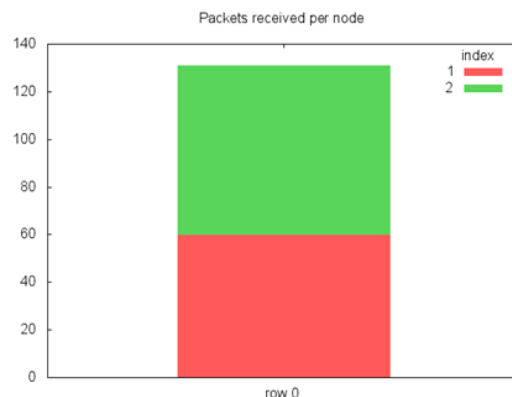

Figure 7. Graph generated by CastaliaPlot script

3. Linux

First, we need to install OMNeT++ and Castalia. Then we can apply the `updateCastalia.sh` script. Finally, we can import Castalia to the IDE or run it from the command prompt.

3.1. Installing OMNeT++

To install OMNeT++, follow its installation guide, available at https://omnetpp.org/.

3.2. Installing Castalia-3.2

Download the file `Castalia-3.2.tar.gz` from https://forge.nicta.com.au/frs/?group_id=301.

Copy the downloaded archive to the directory where you want to install it. Choose a directory whose full path **does not contain any space**.

Untar and unzip the archive.

```
$ tar -xvzf Castalia-3.2.tar.gz
```

This will create a directory named `Castalia-3.2` containing directories named `bin`, `Simulations`, `src`, and files named `CHANGES.txt`, `LICENSE`, `makemake`, etc.

Next, you must add Castalia's `bin` directory to the Path environment variable. To do so, you first have to open your `.bashrc` file:

```
$ sudo gedit $HOME/.bashrc
```

Now, assuming you have extracted Castalia to your home directory, add the following line to the end of your `.bashrc` file:

```
export PATH=$PATH:$HOME/Castalia-3.2/bin
```

Now, you are ready to apply the `updateCastalia.sh` script. Do not forget close all terminal windows so that the changes take affect.

3.3. Executing the updateCastalia.sh script

Assuming you have downloaded the file `Castalia-3.2_OMNeT-IDE_Windows_Linux-master.zip` from https://github.com/alexlacerda/Castalia-3.2_OMNeT-IDE_Windows_Linux and extracted it to `$HOME\Castalia-3.2_OMNeT-IDE_Windows_Linux-master\`, execute the following commands to change to that directory and run the `updateCastalia.sh` script:

```
$ cd $HOME/Castalia-3.2_OMNeT-IDE_Windows_Linux-master
$ chmod 755 updateCastalia.sh
$ ./updateCastalia.sh
```

If the commands have been successfully executed, you should see a screen like the one in the figure below.

Figure 8. Successfully executing updateCastalia.sh

At this moment, you should be able to build and run Castalia simulations from the command prompt. To see how to do so, refer to *Castalia User's Manual*.

In addition, you can import Castalia to OMNeT++ IDE and execute simulations from there. In the next section, we give you instructions on how to do this.

3.4. Running Castalia on the OMNeT++ IDE

To learn how to run simulations from the IDE, see Section 2.4.