**Lab Assignment #1 Report**

**OMNeT++**

Student’s Name: Sidney Sanders

Lab#1: CS 4390

Due Date: Nov 22, 2021

**Introduction**

This is the lab assignment, meant to give a broad understanding to a new piece of software that I have not worked with yet. In this report I explore the inner workings of this software and walk through a few tutorials and record the results of examples given in the lab. This report gives you an overview of what OMNeT++ is and a few of the different important aspects the software contains. A big important part of this software is the user-friendly interface that makes the software easy to use and gives the user a few different ways to edit their code modules.

1. **Main observations**

OMNeT++ is an extensible, modular, component-based C++ reproduction library structure which incorporates a coordinated turn of events and a graphical runtime. Space explicit usefulness is given by model structures, created as autonomous tasks. There are expansions for ongoing recreation, network copying, support for elective programming dialects, information base coordination, SystemC mix, HLA, and a few different capacities.

OMNeT++ represents a framework approach. Instead of directly providing simulation components for computer networks, queuing networks or other domains, it provides the basic machinery and tools to write such simulations. Specific application areas are supported by various simulation models and frameworks such as the Mobility Framework or the INET Framework. These models are developed completely independently of OMNeT++ and follow their own release cycles.

1. **IDE and User Interface**

The IDE is the place where you make and assess your reenactments. There are a few different components when working with the IDE in OMNeT, the NED editor can alter NED extension files both graphically or in text mode, and the client can switch between the two modes whenever they would like, utilizing the tabs at the lower part of the editorial manager window you can switch between files and modes.

In graphical mode, you can make compound modules, channels, and other part types. Submodules can be made utilizing the range of accessible module types. Visual and non-visual properties can be changed in the Properties View, or by exchanges summoned from the setting menu.

1. **The INET Library Module**

OMNeT++ gives a rich article library to straightforward module implementers. There are a few distinctive elements between this library and other universally useful or reproduction libraries. The OMNeT++ module library gives many useful aspects which makes it easy to carry out significant level troubleshooting. Memory leaks, pointer associating, and other memory distribution issues are normal in C++ programs not composed by trained professionals; OMNeT++ reduces this issue by following article proprietorship and recognizing bugs brought about by associated pointers and abuse of shared items.

The prerequisites for usability, measured quality, open information interfaces and backing of installing likewise vigorously impacted the plan of the class library. The steady utilization of article arranged methods makes the reenactment part minimized and thin. This makes it somewhat straightforward, which is a valuable property for both troubleshooting and instructive use.

1. **Tutorial - Tic Toc**

The tictoc tutorial was interesting to go through and explore. It was hard to understand what was happening in the beginning but after some time I can to understand a little built about what was going on. Walking through each file of the tictoc tutorials helped as I would read the comments about what was going on and that way, I would learn more about what each simulation was responsible for doing and that helped me grasp the tutorial better. Reading the comments help tremendously and doing some research online and watching someone else walk through the tutorial helped make this part easier to complete.

I looked up the documentation on this tutorial and it was interesting to get a more in depth idea of what was going on and made helped explain it better than me just looking at the tutorial and not knowing where to begin. It also helped because I haven’t used this software before and it was weird trying to understand the controls and where all the tools needed to work in this environment were.

1. **Examples**

**Graphical user interface, table, Excel

Description automatically generated**

**Graphical user interface, text, application, chat or text message

Description automatically generatedDiagram

Description automatically generated**

1. **Bonus - Problems/Experience**

The initial downloading process went very smoothly from the start of the assignment. I was downloading this software to run on Mac OS so I was nervous before I began to see if I would run into any problems with the extra software, I would have to get to run it completely. For Mac I needed to have downloaded the most recent Java update and the homebrew kit to run the OMNeT++ in the terminal correctly. Lucky for me I had both the java and the homebrew downloaded so there was no extra issue with the beginning of the GDB download. Beginning the download, I had to make sure I selected the correct version to avoid the 6.0 issues that had been mentioned. After downloading I needed to unpack the files in the correct location which gave no trouble, an easy copy and paste into the correct directory. I then ran the target command in the terminal and opened the zip file containing the software. Changing the environment variables also went smoothly, entering the correct directory through the terminal and then setting the .bashrc file. Relaunching the terminal to set the variables, all that was left was to configure OMNeT and create the make files.

The configuration ran smooth and when I made it to run the make command, I hadn’t known how long it was going to take but was very surprised at the length of time it for it to complete the make process. But thankfully the make process completed and I was able to begin the tutorials and open the IDE. I was able to run through a few of the tictoc pages and got to about simulation 5 before stopping to take a break. When trying to continue through the tutorial I tried launching the OMNeT software and I kept getting an error that I was an unknown developer, and I could no longer run the simulations as I had run into a privacy error. Solving this issue was a bit confusing as I couldn’t find much documentation on running into this type of issue. I ended up reconfiguring the software, cleaning the make file, and re making it. This solved my issue, and I was able to run through the rest of the tutorial after that point.

**Conclusion**

Overall, I think this lab was very informative and working with the OMNeT++ software was interesting. It was interesting to see the simulations running and that was a nice change from still coding I usually experience. The software was easy to find documentation for which I think was very important and very useful. There is a lot of documentation on the components of the software and intensive documentation on the tutorials that they provide, there are many YouTube videos that can help you through the information as well which is very helpful. This lab was interesting to complete due to its complexity in the download process but was good when experiencing the different components that the OMNeT++ software provides.

**Reference**

“A Quick Overview of the OMNeT++ IDE.” *A Quick Overview of the Omnet++ Ide*, https://omnetpp.org/documentation/ide-overview/.

FOKUS, Fraunhofer. “Network Simulator Omnet++.” *Eclipse MOSAIC – A Multi-Domain and Multi-Scale Simulation Framework for Connected and Automated Mobility*, 4 May 2019, https://www.eclipse.org/mosaic/docs/simulators/network\_simulator\_omnetpp/.

Varga, András, and Rudolf Hornig. *An Overview of the Omnet++ Simulation ... - Doc.omnetpp.org*. https://doc.omnetpp.org/publications/1416290.pdf.

Varga, Andrá. *OMNeT⁠+⁠+ Ide Developers Guide - Doc.omnetpp.org*. https://doc.omnetpp.org/omnetpp/IDE-DevelopersGuide.pdf.