

Model Coding Project

2020 NREIP Internship

Daniel D. Sartori

dsartori@uri.edu

Beginning November 16 2020

Sunday, 6 Dec 2020

First Research: WireShark and MSCGEN

Exploring the DahlgrenTest/model.code/readme file, interns are provided a high level overview of the project goals covered in the first meeting. Initial examples included are processing message data to generate message type and recipients over time using WireShark.

WireShark is a useful network analysis program which enables the user to capture the transfer of packets in a network. A Flow Chart listing the IP-addresses pinged and the direction of information exchanged between the addresses can be generated. One application of such a chart is ensuring proper configuration of a firewall where network segments that should not be communicating information directly can be properly cut off from one another. This is a good example of modeling the flow of data within a network and works well on a Windows system.

Another example program is Mscgen. Here we can specify the communication process between any sort of entity as a Message Sequence Chart (MSC) and produce an image that neatly outlines these interactions. The entities represented can be hardware on a system for instance, which can be accessed using an SDL library.

After downloading the mscgen executable and completing the setup, the mscgen application does not run. One of the main problems can be seen in the examples included in the application folder. These .msc files are recognized by windows as Microsoft Management Console files (.mmc). Directly changing the application used to open the files is possible, but produces the same result, that is, the mscgen application will not run.

Python is compatible with the desired data types (csv, json, xml, etc.),

and Sphinx is a documentation compatible with Python with an available extension to allow the inclusion of MSC generated images. So it was thought that Python could be used as a viable workaround to call mscgen and fix the problem. Doxygen is another viable documentation software, however Sphinx was chosen over Doxygen because it was originally written for Python and doesn't require extra packages. Coincidentally Sphinx can also work with PlantUML.

As a side note, the latest version of Python 3.9 isn't fully debugged. Python 3.8.6 is the most stable release and has a better package compatibility. Important packages such as pandas used for data analysis and visualization have not yet been updated to run with Python 3.9.