VMWare Virtual Network Visualizer

User Manual

Version 2.0

November 14, 2012

Team
–Intentionally Left Blank–

Pierce Trey, Everett Bloch,
Jonathan Lamb, Daniel Palmersheim

Prepared For
UI CS 383—Software Engineering I
Instructor: Paul Oman, Ph.D.
Fall 2012

Table of Contents

List of Figures

1. Introduction

The VMWare Virtual Network Visualizer is a software designed to assist in the analysis and maintenance of virtual networks—specifically those operating under VMWare, though it can be used for any network, virtual or real, that can output an appropriately structured xml file containing all nodes on the network and all neighbors of each node.

2. System Requirements

The VMWare Virtual Network Visualizer can be run on any system with Java 7 or higher installed. However, the visualization of large graphs can take up a lot of system resources, so it is recommended that, for better performance, this product be run on a system with at least 4 Gigabytes of RAM and a 2 GigaHertz processor. Performance may vary from system to system.

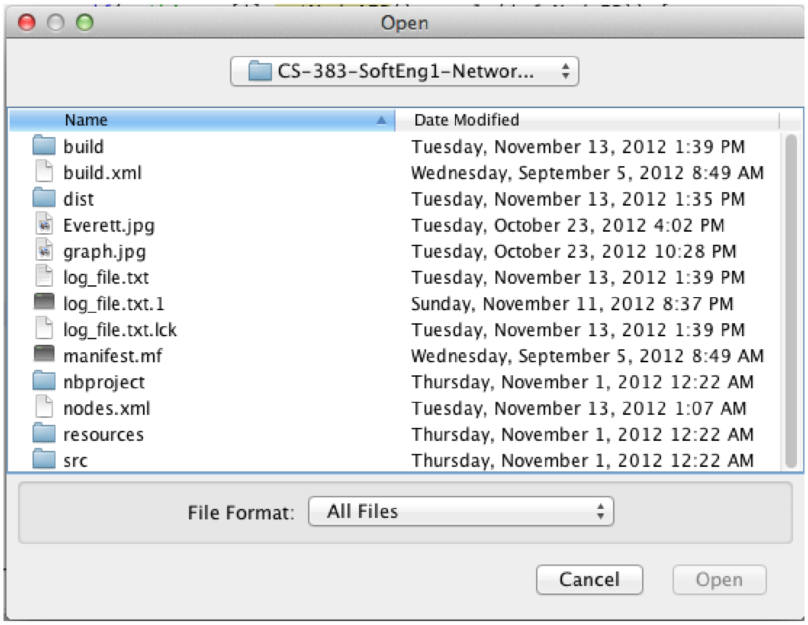
3. Features

3.1 Installation

To Install the VMWare Virtual Network Visualizer, the user needs to simply unzip the packaged folder in a desired directory, open this directory and double-click the application's jar file (VMWare-Network-Visualizer.jar).

3.2 Running the Program

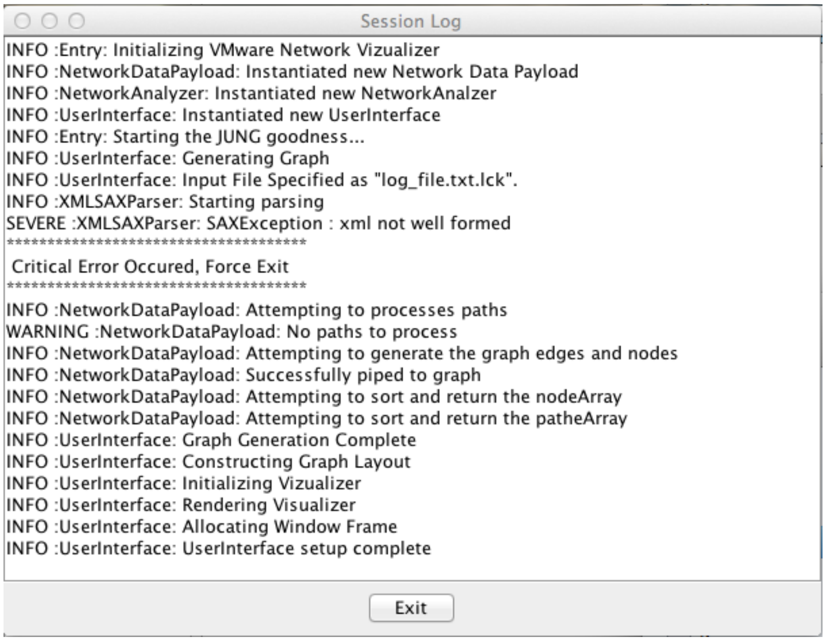
Upon opening the program, the user should be prompted to select an input xml file from a file dialogue. If the user presses the cancel button, the program attempts to open a default file (named nodes.xml) in the directory that the program resides in. If the user chooses an invalid file (or the default file is invalid/ not present), the program will show the logger window with an indication of the file error and the user will be prompted to exit the program.



Name	Date Modified
build	Tuesday, November 13, 2012 1:39 PM
build.xml	Wednesday, September 5, 2012 8:49 AM
dist	Tuesday, November 13, 2012 1:35 PM
Everett.jpg	Tuesday, October 23, 2012 4:02 PM
graph.jpg	Tuesday, October 23, 2012 10:28 PM
log_file.txt	Tuesday, November 13, 2012 1:39 PM
log_file.txt.1	Sunday, November 11, 2012 8:37 PM
log_file.txt.lck	Tuesday, November 13, 2012 1:39 PM
manifest.mf	Wednesday, September 5, 2012 8:49 AM
nbproject	Thursday, November 1, 2012 12:22 AM
nodes.xml	Tuesday, November 13, 2012 1:07 AM
resources	Thursday, November 1, 2012 12:22 AM
src	Thursday, November 1, 2012 12:22 AM

File Format: All Files

Figure 1 - File Open Dialogue at Startup



```
INFO :Entry: Initializing VMware Network Vizualizer
INFO :NetworkDataPayload: Instantiated new Network Data Payload
INFO :NetworkAnalyzer: Instantiated new NetworkAnalzer
INFO :UserInterface: Instantiated new UserInterface
INFO :Entry: Starting the JUNG goodness...
INFO :UserInterface: Generating Graph
INFO :UserInterface: Input File Specified as "log_file.txt.lck".
INFO :XMLSAXParser: Starting parsing
SEVERE :XMLSAXParser: SAXException : xml not well formed
****************************************
 Critical Error Occured, Force Exit
****************************************
INFO :NetworkDataPayload: Attempting to processes paths
WARNING :NetworkDataPayload: No paths to process
INFO :NetworkDataPayload: Attempting to generate the graph edges and nodes
INFO :NetworkDataPayload: Successfully piped to graph
INFO :NetworkDataPayload: Attempting to sort and return the nodeArray
INFO :NetworkDataPayload: Attempting to sort and return the patheArray
INFO :UserInterface: Graph Generation Complete
INFO :UserInterface: Constructing Graph Layout
INFO :UserInterface: Initializing Vizualizer
INFO :UserInterface: Rendering Visualizer
INFO :UserInterface: Allocating Window Frame
INFO :UserInterface: UserInterface setup complete
```

Figure 2 - Log Window Showing File Error and Prompting for Exit

Once an input file has been chosen, a window displaying the log file of the current program should be displayed to indicate that the program is processing the file and getting ready to display the graph.

The logger window should close after the file is done processing, and a window containing the graph, as well as a menu, various options for the graph display, and a sidebar on the right containing information about the graph, and an option for gathering information about a specific node. This window should be sized to the screen.

3.3 Operating Program Features

3.3.1 Graph Display Options Toolbar

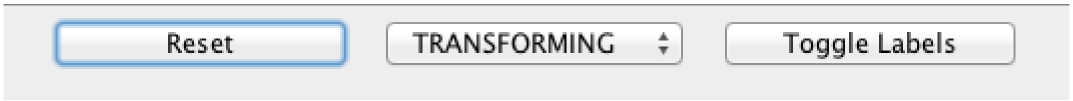


Figure 3 - Graph Display Options Toolbar

The Graph Display Options Toolbar (pictured in Figure 3) contains options for resetting the graph (after nodes have been moved from their original positions), changing between graph manipulation modes (Transforming and Picking, Figure 4), and toggling the node labels on and off (Figures 5 and 6).

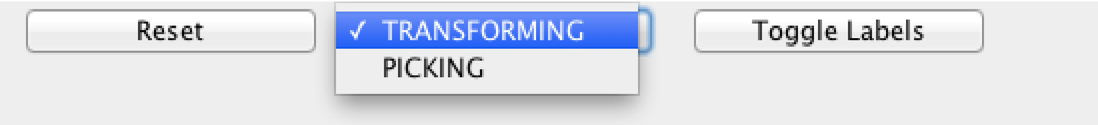


Figure 4 - Transforming/Picking Mode Selection

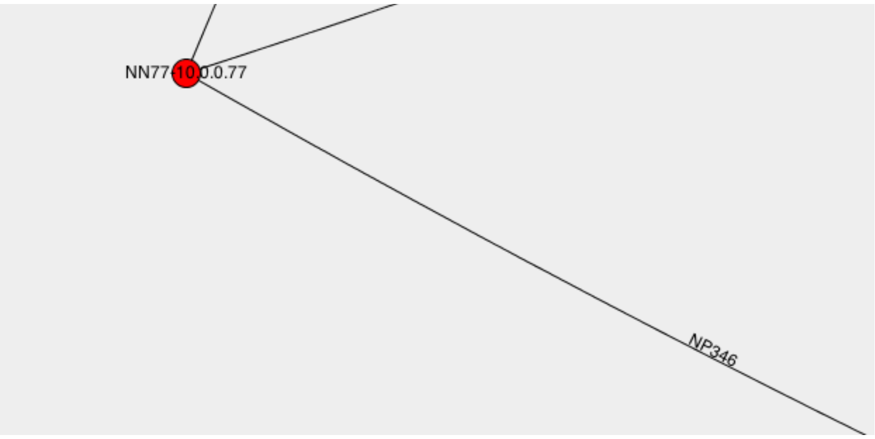


Figure 5 - Node Labels On (Initial State)

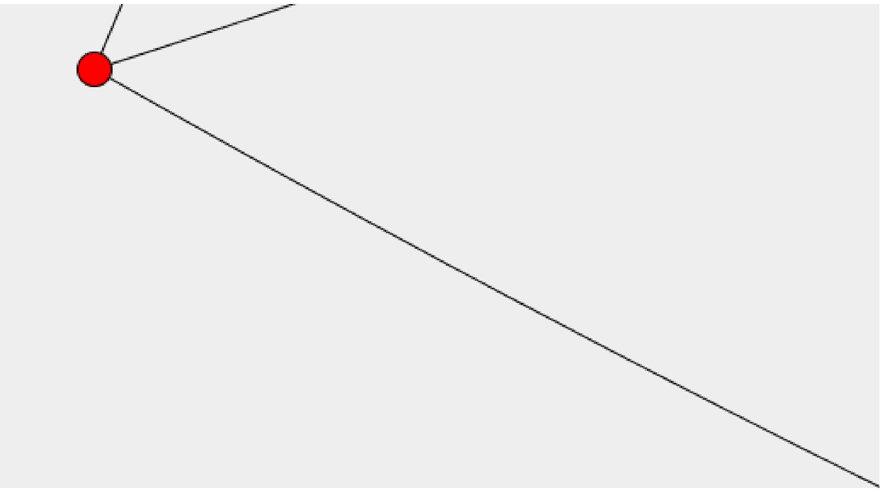


Figure 6 - Node Labels Off (After Toggling)

Transforming Mode allows the user to:

- Move the whole graph around on the screen by clicking in the graph window and dragging
- Zoom in and out of the graph by centering mouse on an area and scrolling in (zoom out) and out (zoom in)
- Rotate the graph by shift-clicking and moving the mouse in a circular fashion
- Skew the graph by control-clicking (or command-clicking on Mac) the graph and moving the mouse side-to-side

Note: Transforming Mode changes to the graph are not reset via the reset button

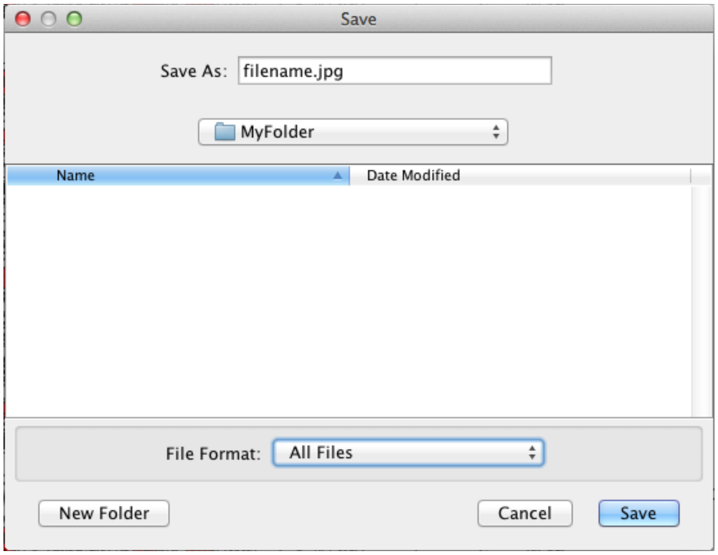
Picking Mode allows the user to:

- Click on a single node and drag it around the screen
- Select multiple nodes (click and drag the mouse to create a box around multiple nodes, or shift-click multiple nodes) and drag them as a unit
- Select paths (for no purpose other than highlighting them)

Note: Picking Mode changes made to the graph are reset via the reset button

3.3.2 File Menu

The File Menu (located in the upper left corner of the application screen) contains the option for saving the graph in its current configuration as a JPG image. This image includes any highlighted nodes, as well as any changes made to the graph in either transforming or picking modes. This image may be used for future reference to the visualized network. Upon selecting the Save as JPG option, a file system dialogue should pop up (Figure 7). The user must enter a filename for the program to output the image to (including the .jpg extension) to and navigate to the directory they wish the file to be placed in.



Save As: filename.jpg

MyFolder

File Format: All Files

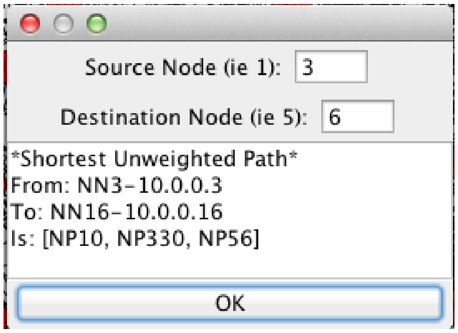
Figure 7 - Save as JPG File Dialogue

The File Menu also contains an option for viewing the current session log. This contains various information that is output by the program to the log file in the program directory, and can be used to diagnose unexpected program behavior. The user may hide this session log by clicking the “hide” button at the bottom of the pop-up window.

Finally, the File Menu contains an option for quitting the program.

3.3.3 Algorithms Menu

The Algorithms Menu (located in the upper left corner next to the File Menu) contains options for calculating the shortest unweighted or weighted path between two nodes. Clicking on either of these options in the menu should cause a window to pop up containing the interface for the selected algorithm. From here the user may enter two node ID’s (within the range of nodes) and calculate the path between them (Figure 8).


Source Node (ie 1): 3

Destination Node (ie 5): 6

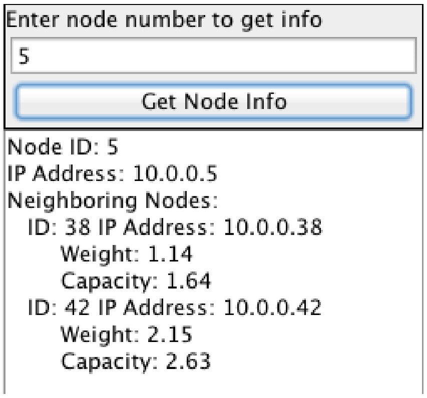
```
*Shortest Unweighted Path*
From: NN3-10.0.0.3
To: NN16-10.0.0.16
Is: [NP10, NP330, NP56]
```

Figure 8 - Algorithm Dialogue Sample

3.3.4 Node Information Sidebar

There is a sidebar on the right side of the screen that provides the ability to gather information about a specific node. At the top of this sidebar, there is a text field for entering a node ID (i.e. 1, 2, 3, etc.). Once the user enters the node ID and presses "Get Node Info," the text area below should display a variety of information about the desired node, such as its ID, IP Address, and a list of its neighbors and their respective ID's, IP's and the weight and capacity of the edge that connects them (Figure 9).

The bottom of this sidebar also contains the total number of nodes in the graph, as well as the total number of edges in the graph (Figure 10).


Enter node number to get info

5

```
Node ID: 5
IP Address: 10.0.0.5
Neighboring Nodes:
  ID: 38 IP Address: 10.0.0.38
      Weight: 1.14
      Capacity: 1.64
  ID: 42 IP Address: 10.0.0.42
      Weight: 2.15
      Capacity: 2.63
```

Figure 9 - Top of Node Information Sidebar

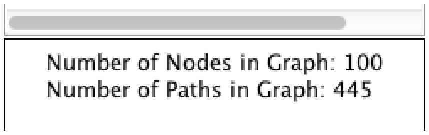

```
Number of Nodes in Graph: 100
Number of Paths in Graph: 445
```

Figure 10 - Bottom of Node Information Sidebar

3.3.5 Exiting the Program

To exit the program, the user may simply click on the close icon in the upper right-hand corner of the window (Windows) or on the upper left-hand corner (Mac/Linux) as they would any other program. Alternately, there is a quit option under the file menu.

3.3.6 Help System

This user manual is accessible from the help menu (next to the algorithm menu). If the user manual pdf is not present in the program's directory, a logger window pops up informing the user of this fact and telling them that the user manual is available from the project website. Note: this requires a pdf reader program to be installed on the user's system.

4. Troubleshooting

4.1 Problem opening a file?

- First identify where the problem occurred in the logger (directly above the "Critical Error Occured" message with the asterisks) and what the logged problem is. The error should occur under XMLSAXParser.
- If the forced exit happens after a "**SAXException: xml not well formed**" message then the selected XML file has an error in it. Check the faulty XML file for proper syntax and restart the program.
- If the forced exit happens after a "**IO error**" message then the file was unable to be found or properly opened. Be sure that you have read permissions on the selected file, then restart the program (Note: In unlikely circumstances files can fail to open due to an operating system error, in this case another attempt will most likely succeed).
- If the forced exit happens after a "**ParserConfig error**" message then there is a problem with the program's internal XML parser. This is a critical error and will require immediate maintenance on the XML parser before the program will work properly.

4.2 Problem creating your own XML file or your default XML file is missing?

- This program expects a particular format for it's XML input file. See the default XML file for an example (nodes.xml). If the default XML file is missing it can be found at our team GitHub repository: https://github.com/celloman/ILB-Network-Visualizer

4.3 Have a lot of lag when picking and transforming the graph?

- One simple solution is to toggle labels off. The labels are set to the middle of their respective objects, and when the graph is moved they need to adjust their positions, using system resources and slowing down the movement process.
- Some operating systems guess at the amount of resources a process will require so to keep resources open for other processes. Since this program depends on a dynamic input file it does not know the resources required on startup, this could result in slow performance for large networks. If this is the case, have the operating system grant more resources to this program.

4.4 Graph jumping arround on startup?

- Due to the nature of the JUNG layout features the graph may jump around on startup in and attempt to find the best configuration for the data. To stop this click the reset button at the top of the screen.

4.5 Reset button not resetting correctly?

- The reset button relaxes the graph back to its original data values, which the visualizer interprets and places nodes on the "canvas" to best represent the data. So a node may not be placed in the same place every time. In addition this does not reset the canvas, only what is going on the canvas. So if the graph is skewed (using the Ctrl + Click and Drag feature with Transforming mode enabled) the reset button will not un-skew the graph, and the camera will not be recentered on the graph. To fix these problems the program will need to be restarted.

4.6 Don't like the current graph layout, want to try a different one?

- This program is set to the JUNG FRLayout by defalut, to change this you can simply download the source code from our GitHub repository (https://github.com/celloman/ILB-Network-Visualizer) and change it in the UserInterface class at the top of the ConstructLayout method. Two other layouts CircleLayout and KKLayout are already set up, just commented out. Be sure to comment out the layout that you are not using before compiling the new layout. Other layouts can be found at the JUNG2 Java doc layout site: http://jung.sourceforge.net/site/apidocs/edu/uci/ics/jung/algorithms/layout/package-summary.html. Be sure to construct other layouts similar to the ones in our code, and import their respective libraries.

5. XML Generator Utility

This software comes bundled with a utility that can be used for generating input xml files that are properly formatted for this application. There is no documentation on this utility, as it was created for testing purposes, but it can be modified to produce a number of different sizes of random input files with varying numbers of nodes and neighbors per node. It is assumed that anyone using this tool is familiar with simple Java source code manipulation. This utility is provided as-is in a zipped netbeans project format, and is located in a folder titled “Utilities” that resides on the project GitHub repository.

This Page Intentionally Left Blank