

~ 1 ~

# VNEC User Guide

Tomas Dej  
&  
François Gagnon  
[fgagnon@sce.carleton.ca](mailto:fgagnon@sce.carleton.ca)

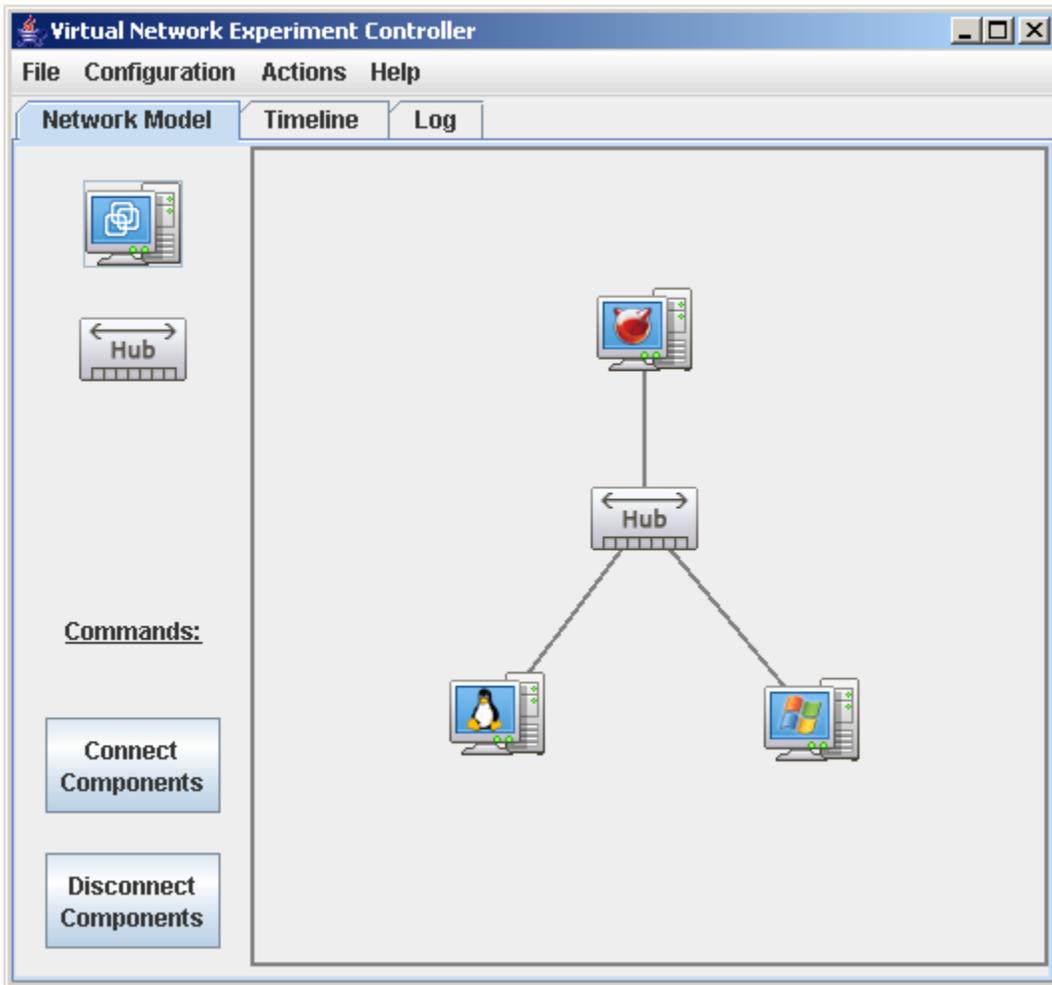
NMAI Research Group  
[www.nmai.ca](http://www.nmai.ca)  
Carleton University

Version: 0.1  
Last Modified: October 13 2008



## Table of Contents

- Introducing VNEC ..... 3
  - What is VNEC ..... 3
  - Where is VNEC developed ..... 3
  - What is VNEC used for ..... 3
- Starting Up ..... 4
- Tabs ..... 5
  - Network Model Tab ..... 5
  - Timeline Tab ..... 6
  - Log Tab ..... 10



## INTRODUCING VNEC

---

### What is VNEC?

VNEC stands for Virtual Network Experiment Controller. VNEC allows the specification and execution of networking experiments in a virtual environment. Since several network experiments are either too sensitive (from a security point of view) or too expensive (due to required hardware) to be executed in a physical network, a virtual is often a wise alternative. VNEC allows the user to specify a network experiment and then automatically execute it in a VMWare virtual environment. An experiment is a network configuration (i.e., network topology and computer configuration) together with a set of tasks to be executed and their execution order.

### Where is VNEC developed?

VNEC is currently being developed in the Network Management and Artificial Intelligence (NMAI) research lab in the department of systems and computer engineering at Carleton University in Canada. The project is supervised by a Ph.D. student and two undergraduate students are actively developing VNEC. VNEC is designed in collaboration with the network security research group at Canada's Communication Research Center (CRC) and is supported by the Talent First program of Carleton University.

### What is VNEC used for?

VNEC is developed mainly for three applications:

- Gathering operating system (OS) signature for OS discovery tools.
- Analyzing viruses spreading behavior.
- Studying reaction of different OS against specific attacks.

## STARTING UP

---

Welcome to the VNEC tool!

When you start up the program you should see the following (See Figure 1)

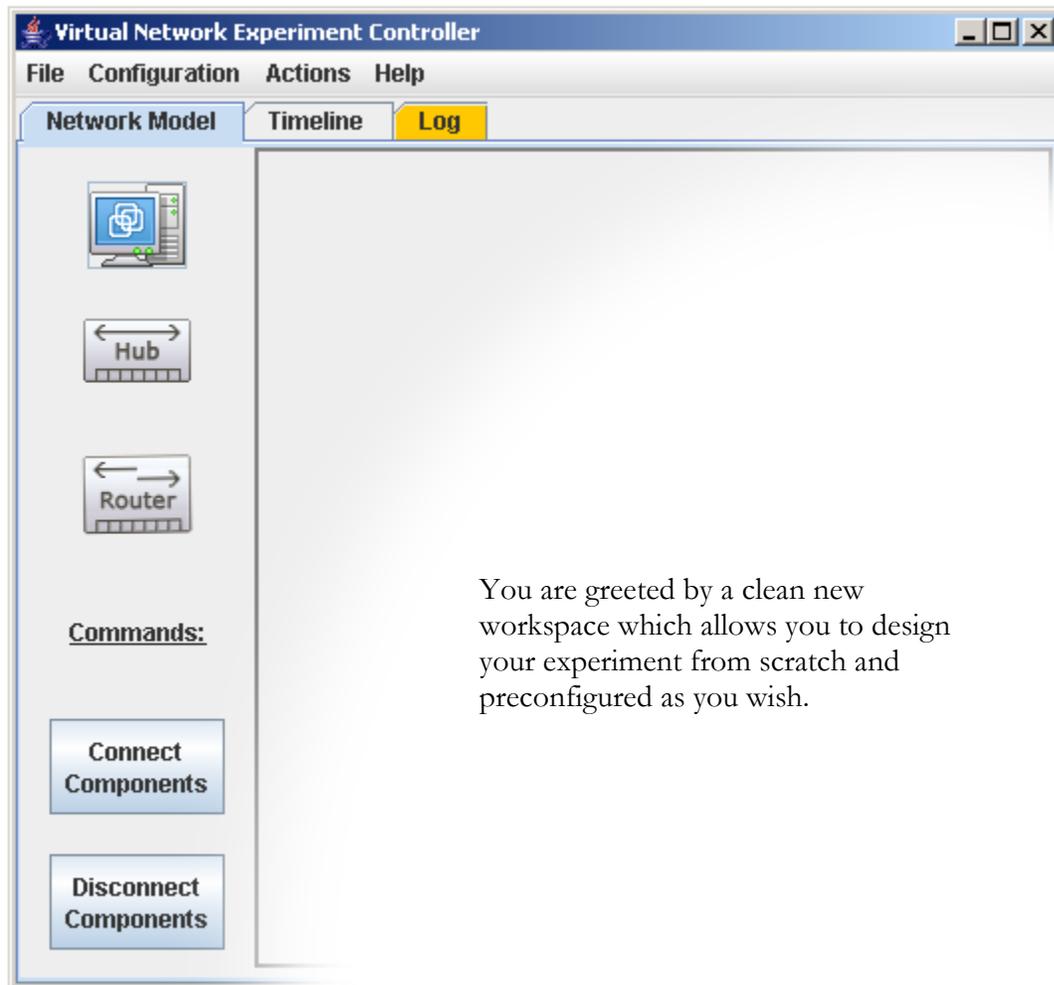
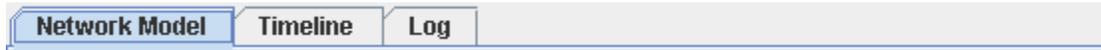


Figure 1 : The first thing you see when starting up - You are in the Network Model view

## TABS

---



There are 3 different tabs provided to keep the work area simple and uncluttered.  
The default view is the Network Model tab

### The Network Model Tab

This is the tab where you set up the topology of your network that you wish to run. On the left hand side you see two objects that are of types Virtual Machine, and Hub. These physical objects can be drag and dropped onto the work area on the right.

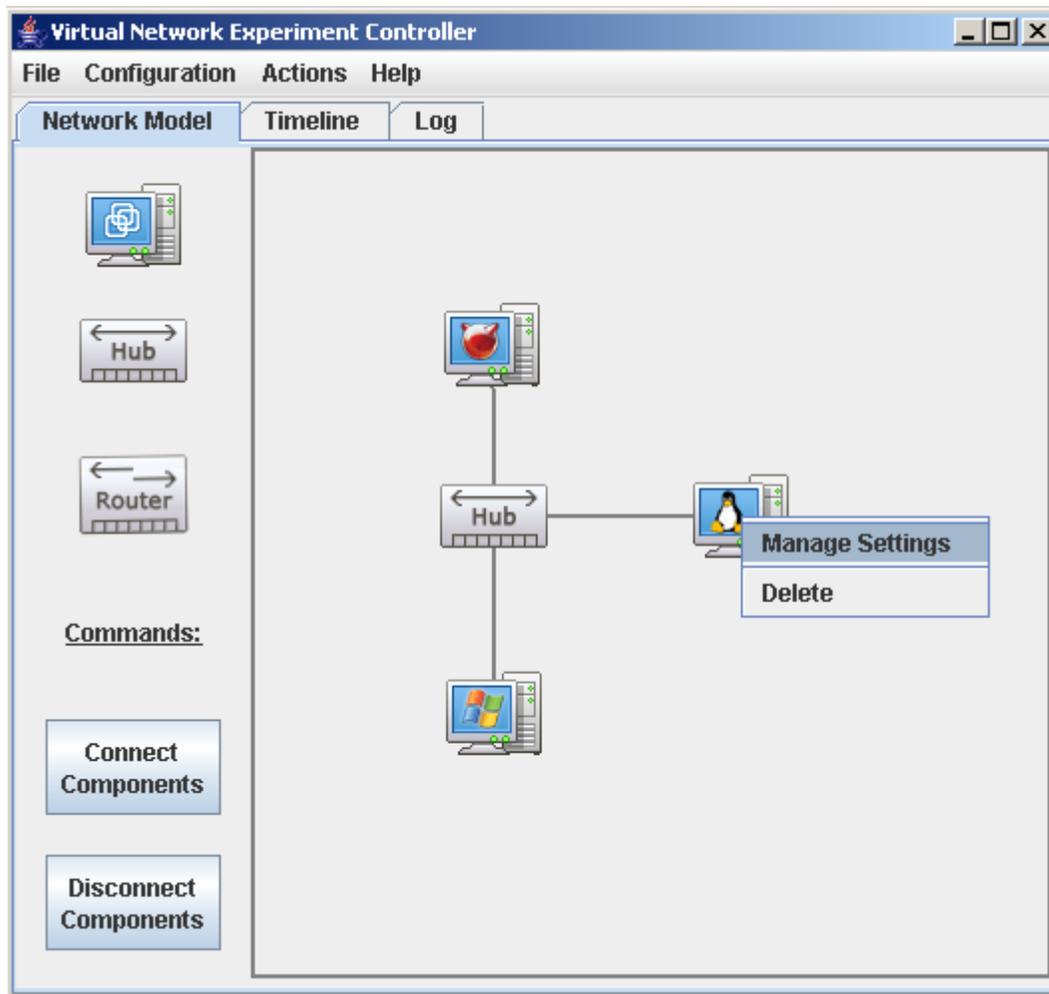


Figure 2 – The Network Model tab. Here a VM object has been right-clicked

On the left side you also see 2 command buttons. These are for connecting/disconnecting the physical elements together as you would physically to set up the network. Note that the elements of same type cannot be connected together, so you will always have a hub with numerous Virtual Machines connected to it. To connect/disconnect simply press the corresponding command button. You are to click on one of the objects you wish to connect, and then click on the next object. After having done this you should see a line representing the connection. Not doing this properly (or clicking elsewhere) will automatically take you out of this mode.

In the work area we can move around and configure the objects. To configure an object simply right click on it and a drop down menu corresponding to that object will appear. In Figure 2 we see a Virtual Machine has been right clicked and a menu appearing with “Manage Settings” and “Delete”.

The Delete selection simply deletes the object from the work area. Any current connections to the deleted object are also removed in the process. The Manage Settings selection only appears when we right click on a Virtual Machine object. Clicking on it opens up a Virtual Machine Settings pane (See Figure 3). This pane is responsible for setting up the specific

details of the VM, such as the actual VM itself and the corresponding snapshot you wish to load by default when you run it. Note that the corresponding snapshot is not currently fully functional, as it will only load the VM to its last state. When we press OK you will notice that the VM object image has now changed to reflect the OS type we have just selected. This is simply to make differentiating the Virtual Machines in the workspace a little easier.

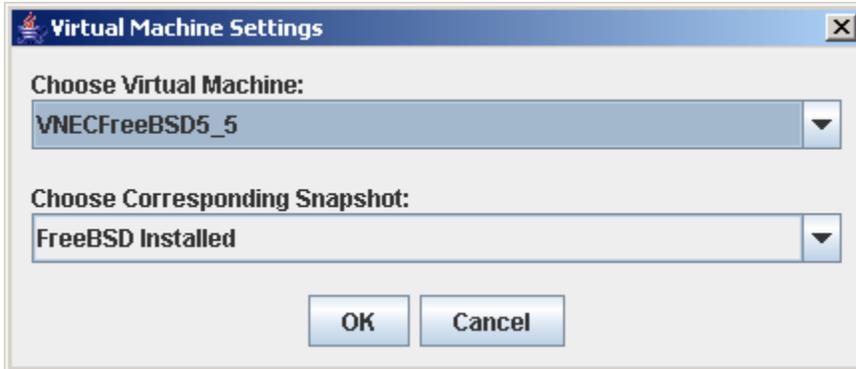


Figure 2

### The Timeline Tab

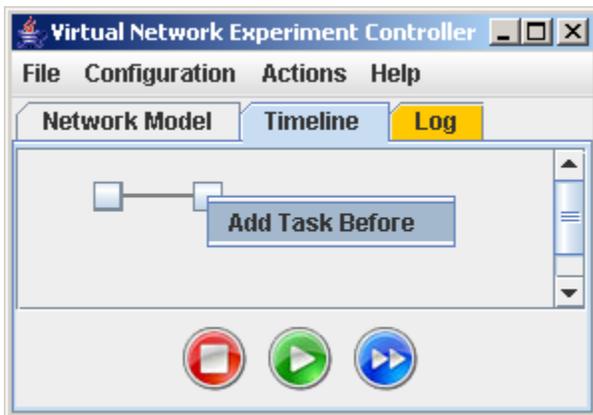


Figure 3

The Timeline Tab is where you preset the execution flow of your experiment. This is simply a sequence of all the events or commands that are to run when you execute the experiment.

### Two Types of Nodes:

The timeline workspace consists of two types of buttons. The first and last buttons of the sequence are always the “dummy” nodes and can be identified as the smaller buttons. The purposes of these are simply to provide a clean layout so the beginning and end of the timeline is clearly drawn and so that we can work around it from scratch. They are also responsible for running behind-the-scenes tasks that take care of the execution flow of the experiment such as starting and closing the dispatcher.

### Drop Down Menu

As we can see if Figure 4, right clicking on a Node brings about a drop-down menu. Here is a list of what each element does:

Edit Task	Brings about the edit task pane for specifying the command
Add Task Before	Creates a new node in the workflow before the current node
Add Task After	Creates a new node in the workflow following the current node
Split Task	This adds another task under the parent of the selected Node, or splits the current Node into two. This means that the parent of this node must execute these split nodes concurrently after having finished. Note that concurrency is not currently fully implemented.
Delete Task	This removes the Node from the execution workflow

Note: Right-clicking on a dummy node (first and last) brings about a drop-down-menu consisting only of either “Add Task Before” or “Add Task After” since they are purely for layout purposes. We can see an example of this in Figure 3.

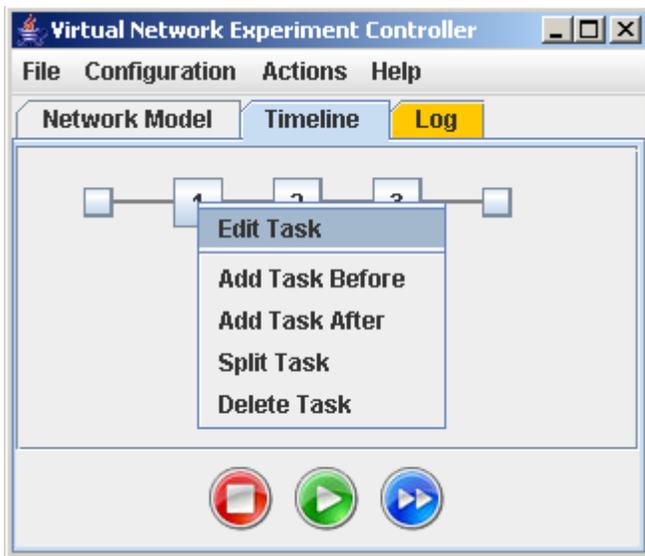


Figure 4

Edit Task Pane:

Selecting “Edit Task” from the drop-down menu after right-clicking a Node brings about the Edit Task pane. This is where you will be setting up the details for the instruction that is to be executed to this corresponding node.

There are generally 3 areas that are to be filled in that mutate as you specify the details you need. This involves choosing the VM type, choosing the Task along with its corresponding VM (if applicable) and then filling in the Parameters for the input that are involved with the corresponding selected task.

Choosing the VM type is based on whether or not the task you need to execute has to have a VM associated with it or not. Single VM is chosen for tasks that are to be carried out only by a specific VM. See Figure 6 for an example of Single VM being selected. Here we see the Ping Task command is chosen by default. The All VM radio button is for commands that apply to all the VMs in the Network Model such as Power On. You can see this is Figure 7. The last radio button option is No VM. This applies for administrative tasks that do not

have anything to do with Virtual Machines. An example of this is the Wait command which simply delays the execution workflow for a specified period of time. See Figure 8.

### Execution of workflow

On the bottom right of the Timeline tab we see three media buttons. These are for controlling the execution of the experiment. Play and Stop executes/halts the entire experiment correspondingly. Pause leaves the experiment in its current stage, ready to be continued later. Forward simply runs only the next task and pauses. Note that stop and pause finish executing the current Node before halting.

The nodes in the execution workflow are colour oriented. Green nodes mean they have finished executing. Yellow means they are currently executing, and non-coloured ones are yet to be executed in the future, see Figure 5.

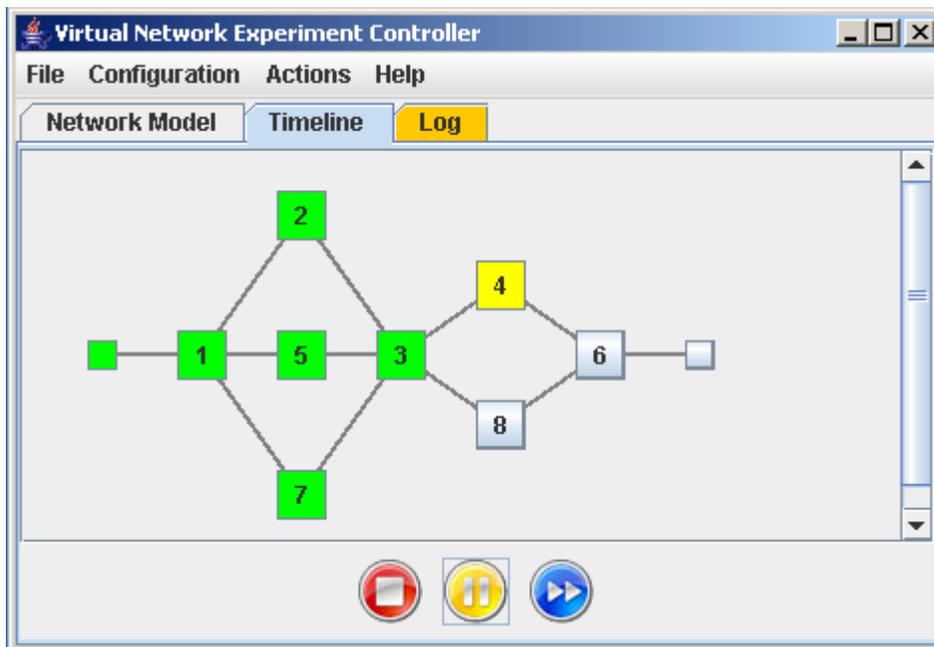


Figure 5: Execution workflow of a running experiment. Node 4 is currently executing.

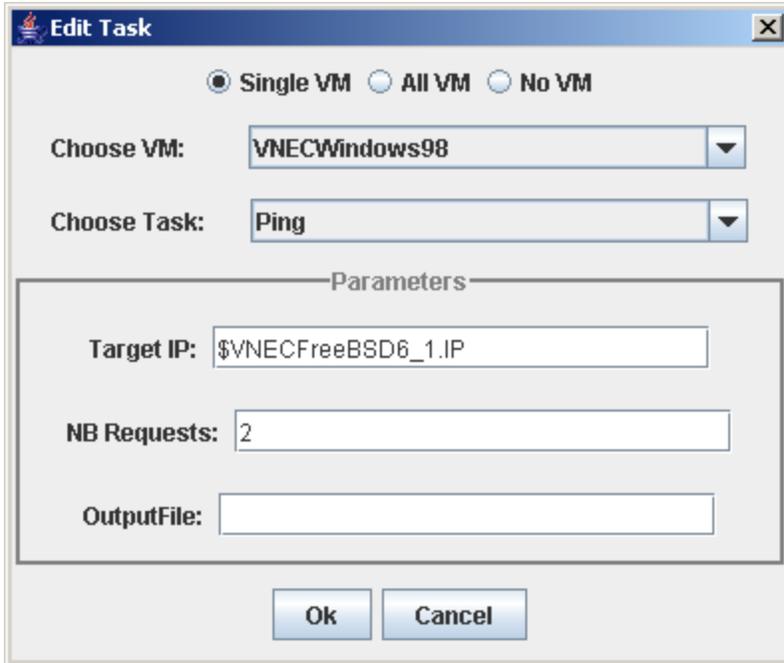


Figure 6

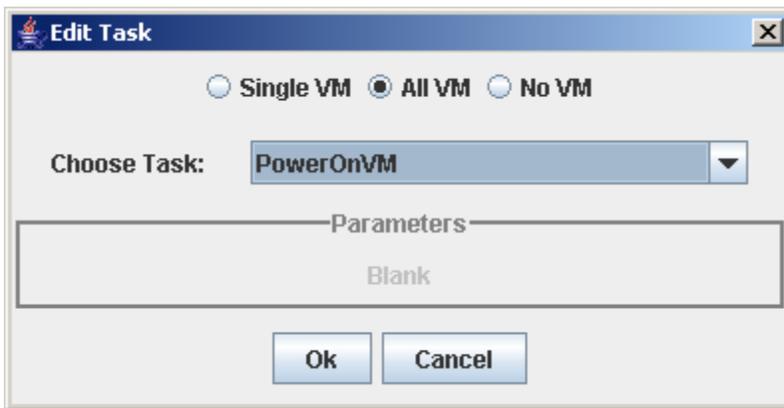


Figure 7

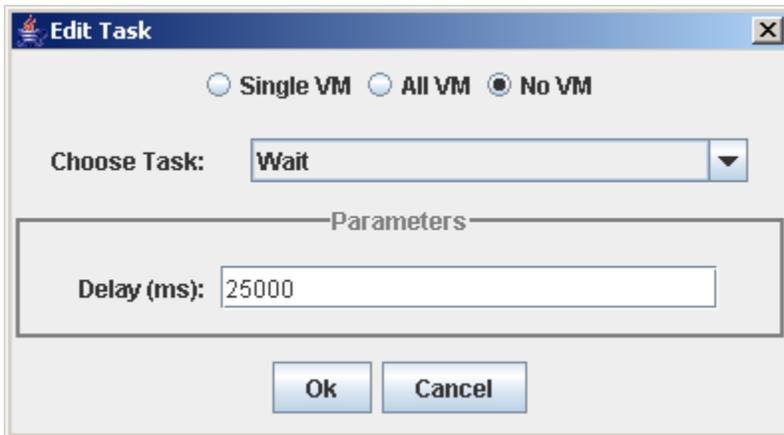


Figure 8

## The Log Tab

The log tab, see Figure 9, is where you will find a list of all the major notifications you will receive throughout your work. This includes notifications such as major changes you have made to the experiment as well as any error messages you may receive. Note that when you are not currently in the Log tab, and you receive a notification, the log tab title itself will turn orange to signify this. Going into the log tab view will clear the colored tab.

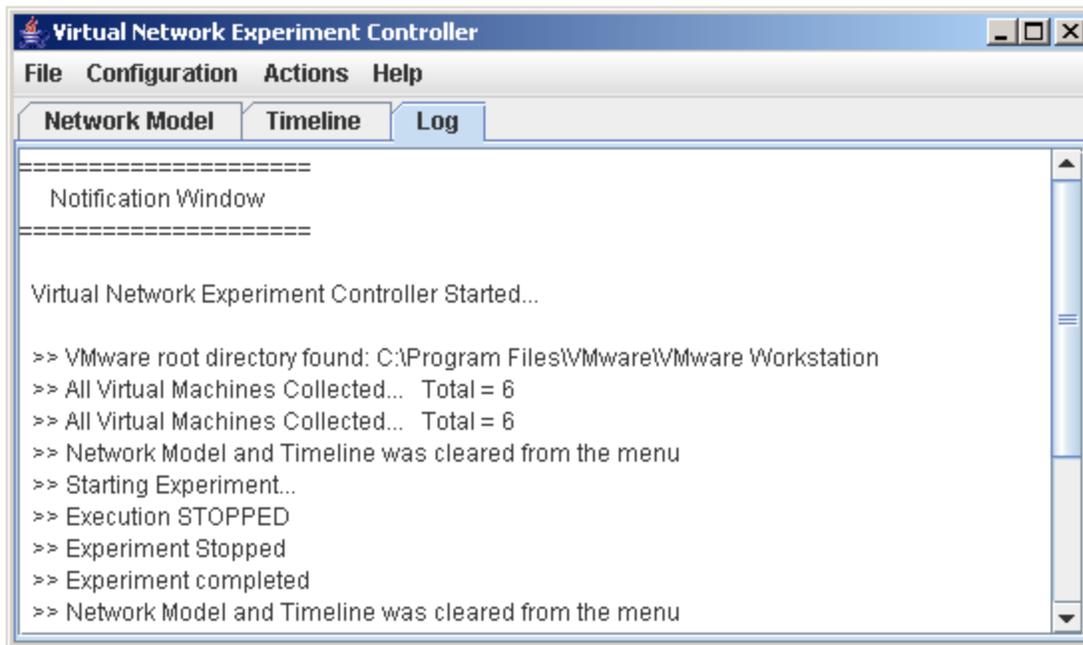


Figure 9 – A view of the log tab