NIST
National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

# GMPLS Lightwave Agile Switching Simulator-Topology and Simulation Creator (GLASS-TSC)

**Version: Draft 1.0**

## TABLE OF CONTENTS

## TABLE OF FIGURES

# 1 INTRODUCTION

The GLASS_TSC is one of two ways to start the GLASS tool. It provides a visual environment from which one can configure and run the simulator. Specifically, it gives the user the ability to:

- Design and create topologies from scratch, as well as load topologies from previously stored files.
- Develop simulation scenarios by specifying the characteristics of the topology. This includes selecting algorithms for light path assignment, selecting protocol stacks for the nodes, and scripting events that will occur at specified times within the simulation.
- Suspend, resume, slow down, and speed up the simulation.
- Visualize message flow, message contents, message exchange between nodes, and lightpaths.
- Debug protocols and algorithms as they are implemented and added to the simulator.

The following is a typical sequence of steps to perform a simulation run:

1. Create the physical network topology within which the simulation will run. The network will consist of a set of network nodes (optical switches (OXC), routers, and hosts) interconnected by optical network interfaces (ONIC) or the abstract network interfaces (NIC).
2. Assign the protocol stack to be run on each of the network nodes.
3. If predefined connections (referred to as static routes) are to be set up before the beginning of simulation, define the connections and select the routing and wavelength assignment algorithms to be used.
4. Schedule events that will occur at a precise time within the simulation.
5. Compute the static routes with the selected algorithm and run the simulation.
6. View the output produced by the simulation. This includes performance statistics, as well as any messages sent to the standard output device.

This manual provides an introduction to the GUI and its components; It provides the reader with enough information to get started using the GUI immediately. All menus, toolbars, and tabs are explained, as well as navigating each section of the GUI.

# 2 STARTING THE GLASS-TSC

The GLASS tool is programmed in Java (java 1.4.0 is required). It is assumed that the software package will be installed in directory *x*/glass/lib, and *x*/glass/cfg where *x* represents the directory path of the host system. They are multiple ways to start the GUI depending on the environment:
If the directory *x*/glass/lib contains the jar files needed for simulation, including the SSFNet jar files, then start the self-executable jar file designer.jar to star the GLASS-TSC.

With the same configuration, using a command line:
set GVlib= *x*\glass\lib
javaw.exe -jar %GVlib%\designer.jar

It is also possible to run the GLASS-TSC by running the class gov.nist.antd.merlin.gui.Main. In this case, it is necessary to specify the whole classpath.

As the last configuration, if the location where the GLASS-TSC is started is not under *x/glass* then The user may have to give the system property GLASS_HOME as follow:
java classpaths "-DGLASS_HOME=c:\my\directory\path\glass" gov.nist.antd.merlin.gui.Main

The software has been running under Windows and Unix computers.

# 3 A QUICK TOUR OF THE GLASS-TSC

When the GLASS GUI is started, the initial screen appears as shown in Figure 1. The top portion consists of a menu, and toolbar icons; the middle panel is the workspace view, which can be changed by the workspace view selector; the middle part contains the simulation panel for control of simulation runs.

**Figure 1: The GLASS-TSC**

The workspace view selector is composed of six tabs, each of which select different information to be displayed in the workspace. These tabs can also be selected from the View Menu. A brief description of each tab follows.

## 3.1 DESIGN VIEW (THE "TOPOLOGY" TAB)

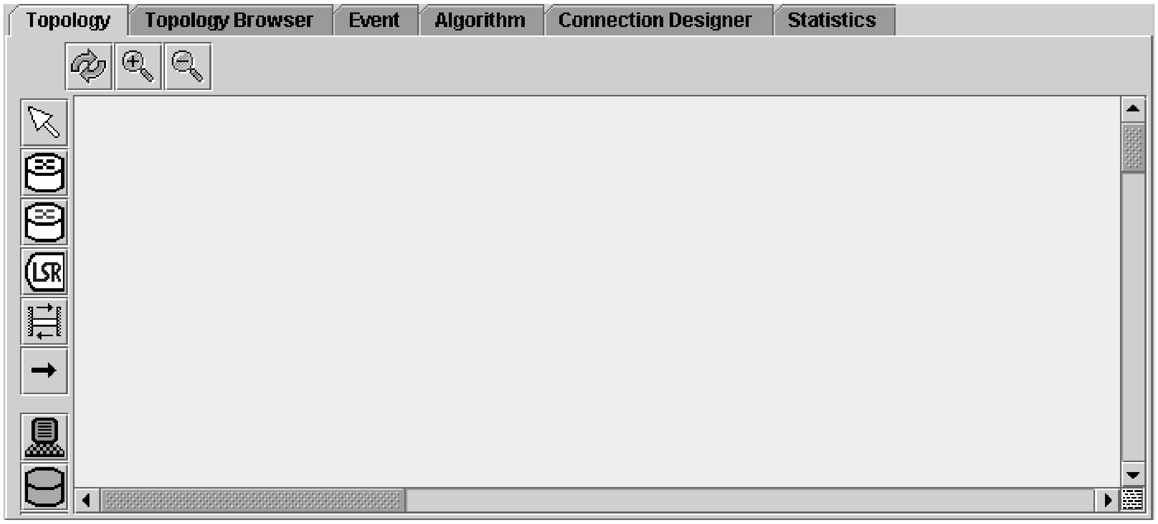



**Figure 2: Topology panel**

This is where you see a graphical representation of the network topology. From here you can create nodes, links, and light path requests with a few mouse clicks. You can start a simulation, and observe the light path being set up and torn down, as well as messages going back and forth between nodes.

## 3.2 BROWSER VIEW

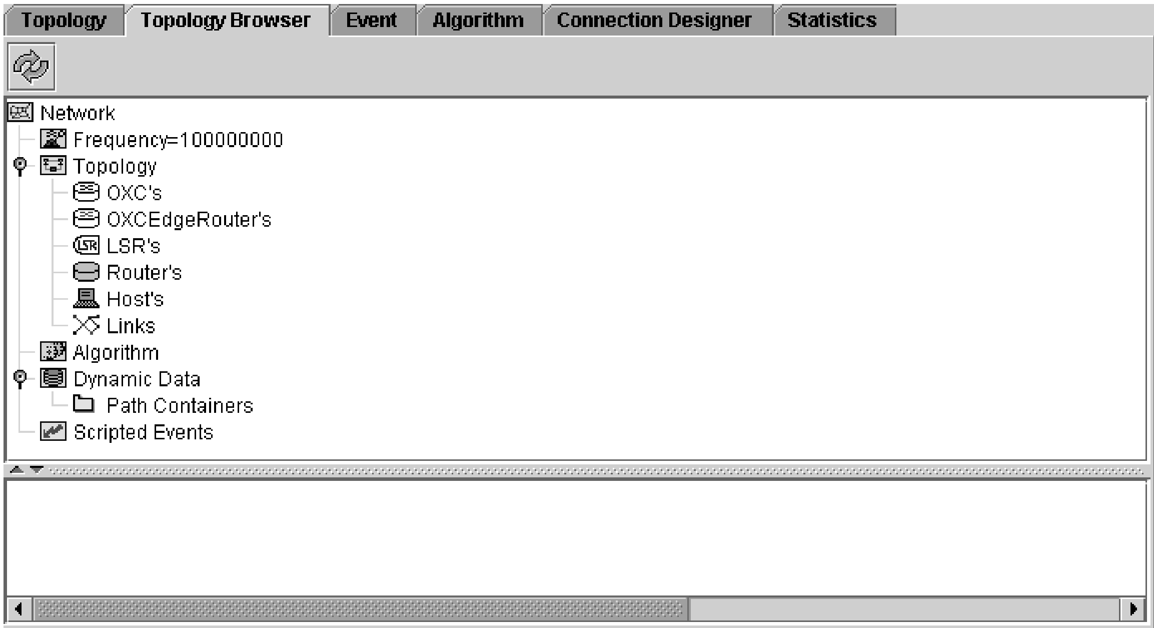



**Figure 3: Topology Browser**

The browser tab shows you the textual view of the network topology in the Java tree structure, which gives you the detailed characteristics of each component of the topology. You can also view specific information about algorithms, dynamic data, and scripted events. The bottom half of the workspace shows the print out of the standard output device generated by the currently running algorithms and protocol stack.

## 3.3 EVENT VIEW

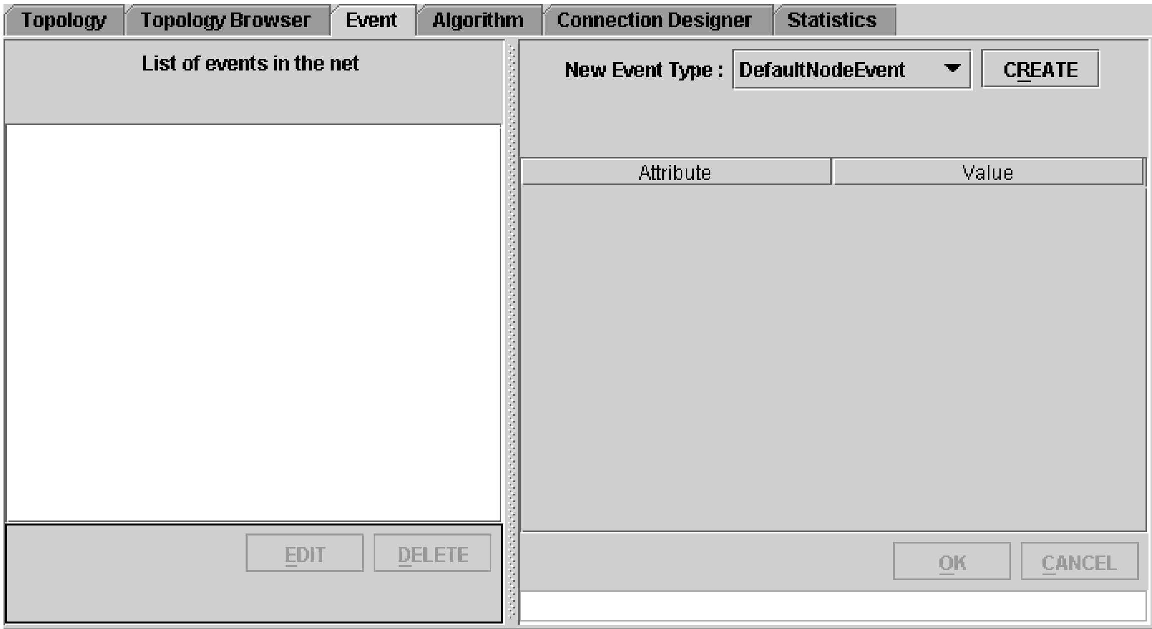



**Figure 4: Event panel**

The event view shows events scheduled to occur at a predetermined future time. The events could be loaded from a previously saved file, or created using this view. They are associated with network fault conditions at the node, interface, link, fiber, or lambda level.

## 3.4 ALGORITHM VIEW

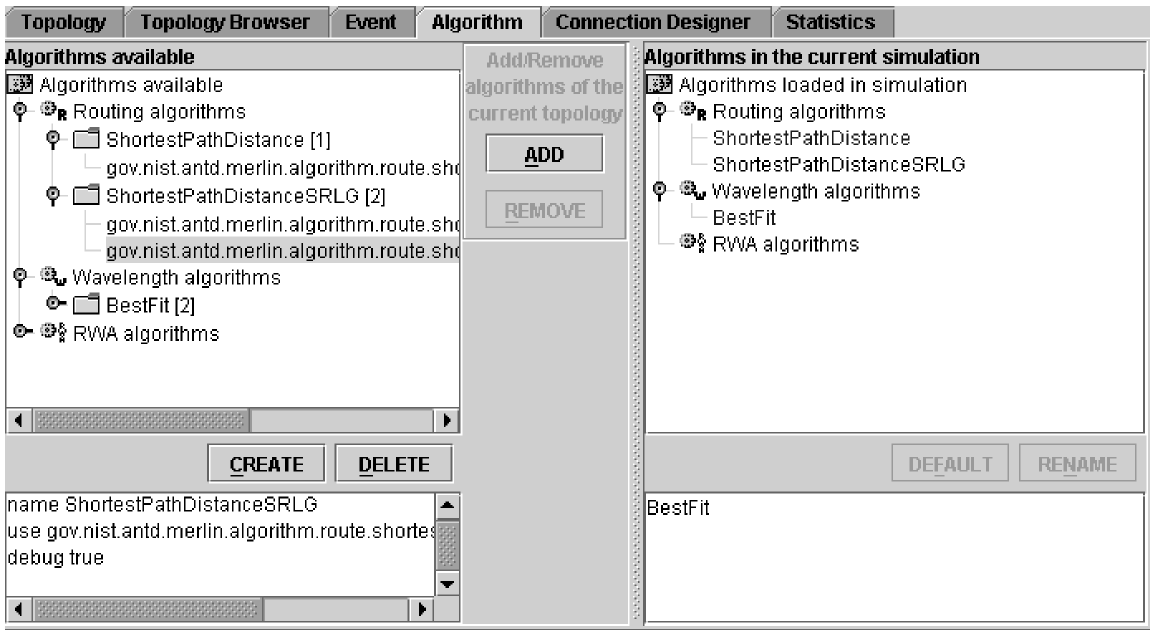



**Figure 5: Algorithm panel**

The algorithm view displays the algorithms that the current topology is using for path creation and wavelength assignment. Algorithms can be used to establish light paths and their protection or restoration paths. This view is used to select, set as default, and add customized algorithms to be used in the simulation.

## 3.5 CONNECTIONS VIEW

The connections view displays connections or light paths between nodes along with the properties of these connections, which might include things like bandwidth, delay, and the routing algorithm associated with a given connection. GLASS-TSC provides 2 views of the connections: A graphical view and a textual view. They both provide the capability of creating, removing, and editing connections.

Next is the graphical connection designer, followed by the scripted designer.

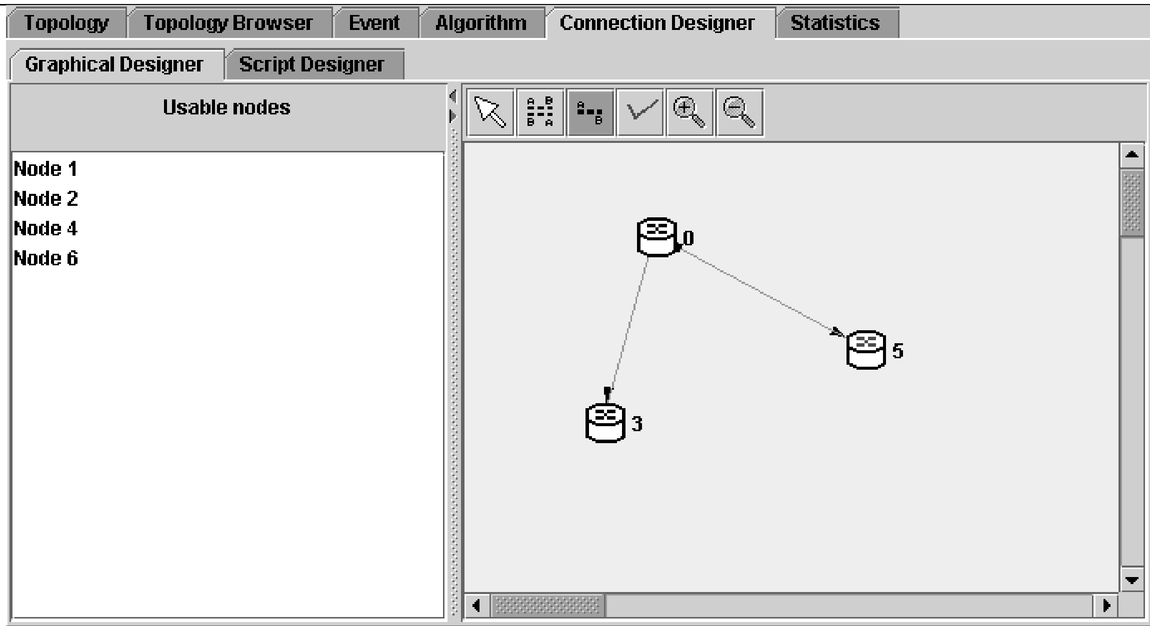



**Figure 6: Graphical connection designer**

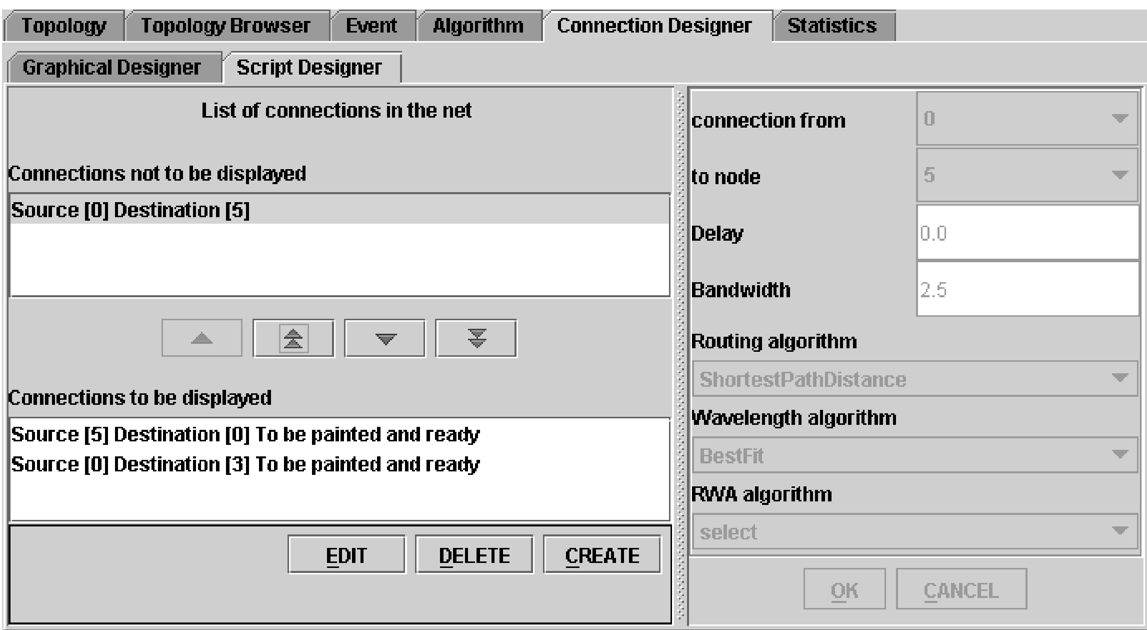
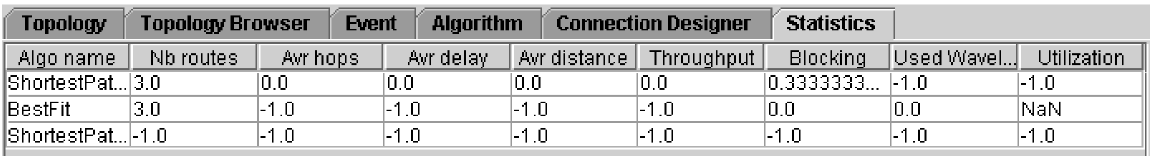



**Figure 7: Scripted connection designer**

## 3.6 STATISTICS

The statistics view displays performance information for each algorithm currently in use within the loaded topology. Information displayed includes throughput, utilization, average delay, and average hops.


Topology | Topology Browser | Event | Algorithm | Connection Designer | Statistics

| Algo name | Nb routes | Avr hops | Avr delay | Avr distance | Throughput | Blocking | Used Wavel... | Utilization |
|---|---|---|---|---|---|---|---|---|
| ShortestPat... | 3.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.3333333... | -1.0 | -1.0 |
| BestFit | 3.0 | -1.0 | -1.0 | -1.0 | -1.0 | 0.0 | 0.0 | NaN |
| ShortestPat... | -1.0 | -1.0 | -1.0 | -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |

**Figure 8: The statistics panel**

## 3.7 TOOLBARS AND MENUS

### 3.7.1 FILE MENU

The file menu and some icons in the shortcut toolbar manage network topology files.

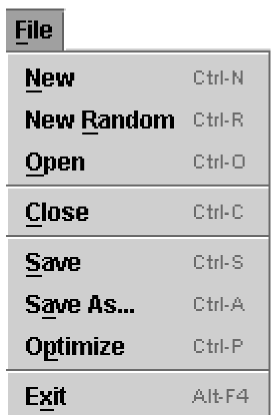



**Figure 9: File menu**

And the associated topology toolbar.

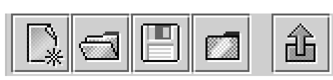
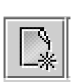


**Figure 10: Shortcut toolbar**

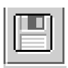


Create a new topology. This will remove the current topology from the workspace, and leave a blank workspace. This is equivalent to **File→New** from the Menu. A new physical network

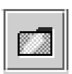
topology can then be created by selecting the appropriate icons for network components and placing them in the workspace. See the section entitled “Topology Designer” for more coverage and other ways to generate a topology.

To create a random network, use **File→New Random**

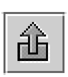
 Open a DML file and load the topology specified by that DML file. This is equivalent to the **File→Open** command from the Menu. A DML file could be created by a previous save operation, or by using a text editor.

 Save a topology. This is equivalent to **File→Save** from the Menu. The current topology and simulation information will be saved as an SSFnet DML file. The saved file can later be reloaded using the open menu.

To change the name of the file, use **File→SaveAs**.

 Close the topology. This is equivalent to **File→close** from the Menu. Any changes made to the topology since it was last saved will be lost.

The **File→Optimize** option allows the user to compact the DML file by generating dictionary entries of redundant information.

 Close the GLASS-TSC. This is equivalent to **File→exit**. All unsaved work is lost.

### 3.7.2 The View menu

The View menu allows you to change the current workspace. This menu is equivalent to clicking on one of the tabs to select the workspace (shown in Figure 1).

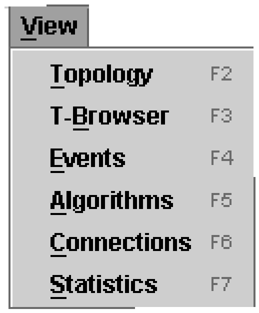



**Figure 11: The view menu**

### 3.7.3 The Layout menu

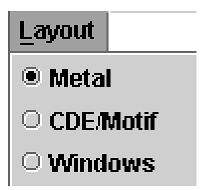



**Figure 12: The layout menu**

This menu provides the user with different look and feel for the interface.

### 3.7.4 The Simulator menu.

The Simulator menu and some icons in the simulation control panel (see section 5) provide ways to run the simulation, as well as controlling the selection and execution of the static routing and wavelength assignment algorithms. This also where the display of the static routes can be toggled on or off.

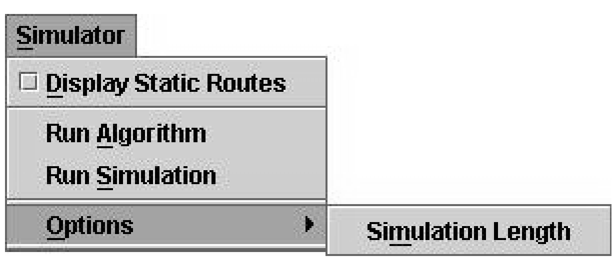



**Figure 13: The simulation menu**

**Run Algorithm**: Calculate all requested connections with the Algorithm that was chosen to compute the connections.

**Display Static Routes**: Display all pre-calculated connections within the current network topology. The static routes are displayed in different colors on the Topology Designer workspace.

**Run Simulation**: This runs the currently loaded topology. Before starting a simulation run, all simulation parameters must be selected properly: simulation length.

**Simulation Length**: This allows you to modify the length of the simulation by bringing up a dialog box:

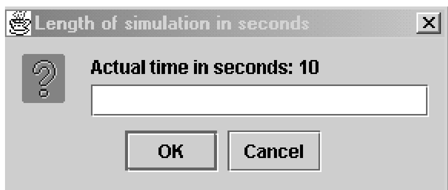
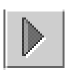



**Figure 14: Simulation length configuration**

 This button located in the simulation control panel is equivalent to **Simulator→Run Simulation**.

## 3.7.5 THE T-BROWSER MENU

This menu is associated with the Browser panel (see section 3.2) and configure the output capture.

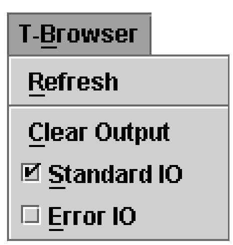



**Figure 15: T-Browser menu**

It allows the user to specify which output to capture (Standard and Error).

### 3.7.6 THE PLUG-IN MENU

This plug-in shows the current available plug-in that is installed in the GLASS-TSC.

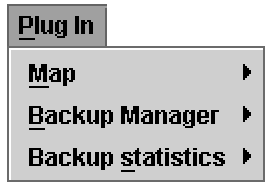



**Figure 16: The Plug-in menu**

The configuration of the Plug-in menu is dynamic and users can add or remove plug-in. To do so, the class must extend the interface gov.nist.antd.merlin.gui.MenuPlugIn and modify the file menu.properties located under glass/cfg.

Note: this file also configures the menus available in the GLASS-TSC.

### 3.7.7 THE HELP MENU

The help menu provides the user with the on-line help and also the credits for the GLASS projects.


**Figure 17: The Help menu**

The user will also find the contact to report bugs and questions about the GLASS tool.

Depending on the current panel displayed in the GLASS-TSC, pressing F1 will show help related to it.

## 4 CREATING AND CONFIGURING A TOPOLOGY WITH THE GLASS-TSC

Under the GLASS, a network would consist of a set of nodes: hosts, routers, and optical cross connects (OXC), interconnected with links through Network Interface Cards (NICS). The *host* and *router* nodes are the same as defined in the SSFNet [1]: a host is a node with one NIC while a router is a node with multiple NICs. GLASS adds multiple new components. The Label Switching Router (LSR), OXCEdgeRouter and OptcialCrossConnect (OXC) for the nodes [2]. An optical link is also available and an optical network interface card (ONIC) to attach them [3].

## 4.1 THE FREQUENCY

The frequency specifies the resolution of the clock ticks to be used in the simulation run. It is specified in unit of fraction of one second, and must be the first item specified when creating a new topology. This is because it is used to configure the elements of the topology, for example to convert the link speed and queuing delays into appropriate clock ticks. The default value is 10,000,000 or 10 microseconds.

 This button will open an input dialog that shows the current frequency and as a default in the input text, the next frequency that will be used when new topologies will be created.
This button is available in the TopologyPanel (Figure 2).

After changing the frequency, the user needs to open a new topology in order to validate the change. The frequency specified in a dml file will always overwrite the frequency specified in the GLASS-TSC.

## 4.2 THE NODES

Figure 18 shows a network topology under construction. Four nodes (node 0 to 3) are linked by bi-directional optical links and regular links. The properties of node 1 are shown on the right side of the figure.

Currently host or router nodes can be displayed but are not configurable yet. However, to create a full topology using hosts and router, one can use the GLASS-TSC to create and save a template for the dml file, and then the user should modify the dml file to include protocols and configure the nodes.

**Figure 18:A simple network topology and an editing node**

### 4.2.1 CREATION

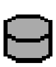
The nodes can be created from the TopologyPanel (Figure 2).

 *Add an OXC*. Click on this button and then click on a location in the workspace to insert an OXC at that location.

 *Add an OXC Edge Router*. Click on this button and then click on a location in the workspace to insert an OXC edge router at that location.

*Add an LSR*. Click on this button and then click on a location in the workspace to insert an LSR at that location.

 *Add a host*. Click on this button and then click on a location in the workspace to insert a host at that location.

 *Add a Router*. Click on this button and then click on a location in the workspace to insert a Router at that location.

For information about the configuration of each node and their specifications, see the documentation about the nodes in GLASS [2].

Right clicking on a node will bring up a menu with two options:

**Delete**: this will delete the selected node from the topology.

**Properties**: this will bring up a property editor box. For more about node properties, see section 3.5, Viewing and Changing Properties for a Node

### 4.2.2 Configuration of nodes

The properties associated with nodes (Except Host and Routers) are modified via the Property Editor box, but can be done only before the starting of the simulation. Right clicking on a node and selecting the **Properties** menu option will display the Property Editor box:

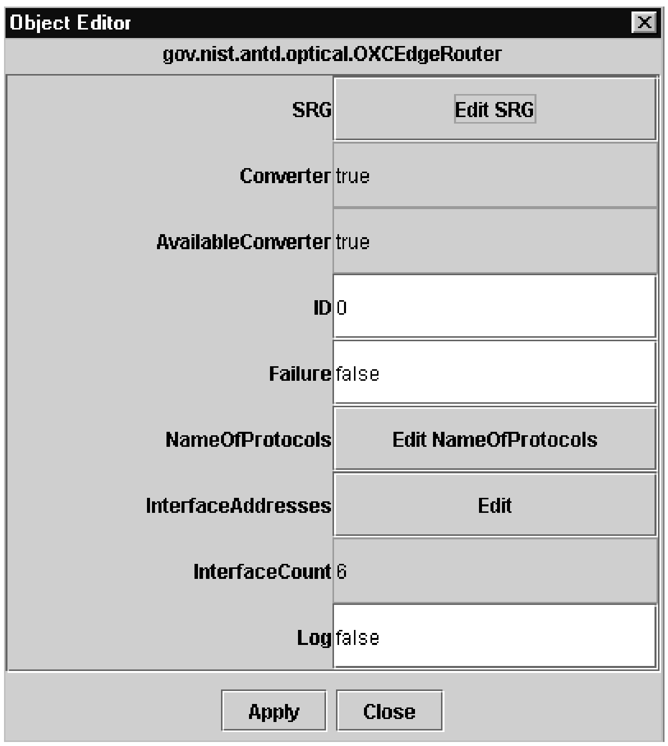



**Figure 19: Property Editor for an OXCEdgeRouter**

**ID**: the unique ID of the node within the network topology. This is automatically assigned to be the order in which the node was created, starting with 0.

**Interface count**: the number of network interface cards present in this node.

**Interface addresses**: the IP addresses of the ONICs in this node. Clicking on Edit brings up a list of the ONICs, along with the bit error rate, delay, and jitter of each ONIC.

Highlighting an ONIC and selecting Edit will open the ONIC editor, in which you can edit an ONIC's properties.

**Failure**: A Boolean field indicating whether or not the node has a failure in one of its components.

**SRG**: Shared Risk Group of this ONIC. Clicking on Edit displays the SRG editor.

**Converter**: A Boolean value indicating weather or not this node has wavelength conversion capabilities.

**Available Converter**: A Boolean value indicating weather or not this node has any available converters, i.e., this is false if all converters are being used.

**Name of Protocols**: clicking on Edit Name of Protocols will bring up the NameOfProtocols editor, which is discussed shortly in section 3.7, Setting Up Protocols for a Node".

Once the simulation is started (by clicking the play button on the simulation control panel), a different property view will be shown (Figure 20). During the simulation the characteristics of a node can only be displayed but not modified.


**Figure 20: Property viewer for a node**

Note: As the property panel is generic, some attributes may be accessible and modifiable but their modification would result on an incorrect configuration of the node.

#### 4.2.2.1 ADDING PROTOCOL SESSIONS

In order to simulate the operations of a network, one must define the protocol stack or processes associated with each node in the network; this is defined in SSFNet as "Protocol Sessions." For example, one may assign a program to each OXC Edge Router to simulate the arrival of wavelength requests with some statistical properties. Similarly if one wants to study the behavior of a TCP/IP based signaling protocol to do wavelength assignments and restorations, then he must install TCP/IP as well as the signaling protocol on each node. For OXCs, the default protocols are **oxcswitch** and IP. "oxcswitch" is the process that simulates the switching function of an OXC.

Additional protocols can be added to any group of existing nodes from the topology designer. First, select a node and right-click on it. Select **Properties.** Then select "Edit NameofProtocols", which will bring up the following editor:

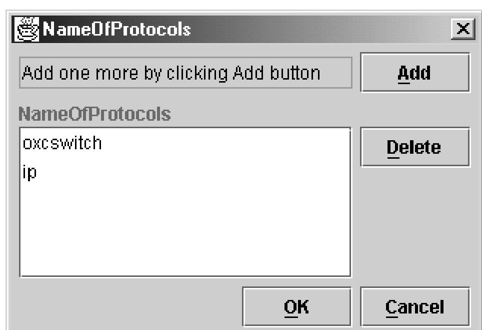



**Figure 21: ProtocolSessions in a node**

This displays the names of all of the protocols currently running on this node.

**Delete**: Removes the selected protocol from this node.

**Add**: Adds a protocol to this node. This will bring up the Protocol Session Editor, which is described in section 4.2.3.

### 4.2.3 THE PROTOCOLSESSION EDITOR

The Protocol Session Editor is used to add a protocol to this node. You can also load any accompanying parameters to the protocol from within a DML file. This file can be created from within the editor, or loaded from elsewhere.

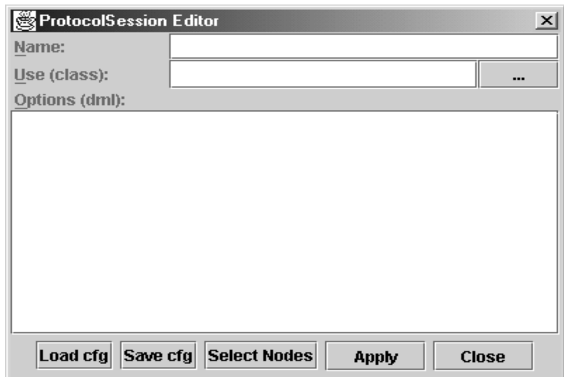



**Figure 22: Adding ProtocolSessions, enter configuration**

**Name**: the name of the new protocol, which the topology will use to refer to the protocol

**Use (class)**: the Java class file, which contains the protocol. Clicking on 

will bring up a directory menu from which you can select the class file.

**Options (dml)**: the options to be used for the new protocol in the dml format.

**Load cfg**: load a DML file, which contains options to be used with the new protocol. The dml file must contain only one entry of ProtocolSession. If more than one or no ProtocolSession entry is found an error message will be displayed. The dml file does not contain a full topology.

Examples of these configurations can be found under examples/optical/protocols. This can be useful to store templates of protocol configurations.

**Save cfg**: save the contents of the Options text area into a DML file.

**Select Nodes**: this brings up the node selector dialog box:

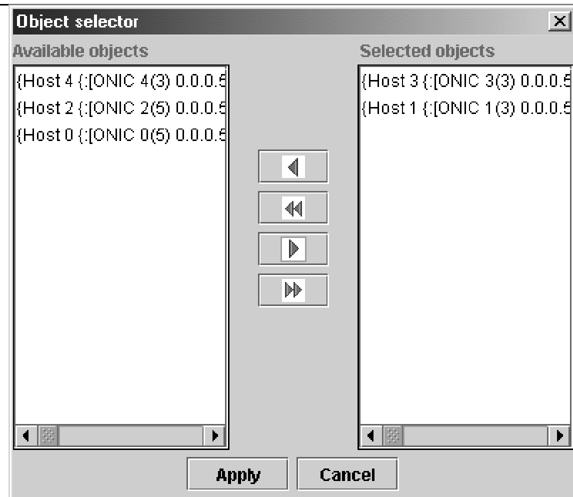



**Figure 23: Adding ProtocolSession, selecting nodes**

Available objects are nodes to which the protocol could be applied, and selected objects are nodes to which the protocol will apply. Click on **Apply** when all desired nodes are selected.

## 4.3 THE LINKS

There are 2 different types of link available under GLASS-TSC: The optical links and non-optical links. The optical links are also divided in two categories: the unidirectional and the bidirectional.
The bidirectional capability of an optical link is defined by the direction of the fibers that compose it and also the capability of the fibers to be bidirectional.

### 4.3.1 CREATION OF LINKS

*Create a bi-directional link.* After clicking this button, click on a node in the workspace. This will be one end of the link. Click on a second node to complete the link. Host and Router do not accept this kind of link.
The default configuration is: 1 bidirectional fiber with 4 lambdas. 2 of the lambdas are for control (for bidirectional control messages) and 2 are for data.

*Create a uni-directional link.* Use this the same way as the bi-directional link button. Host and Router do not accept this kind of link.
The default configuration provides 1 unidirectional fiber with 3 lambdas (1control and 2 data lambdas).

 *Create a non-optical link* between 2 nodes (OXC does not accept this type of link).

To delete a link, right click on the link and select delete from the menu.

### 4.3.2 EDITING LINKS

First click on the **Select** button in the Topology workspace (the button with the arrow icon). Then Right-click on a link, and select **Properties**. The following editor will appear:

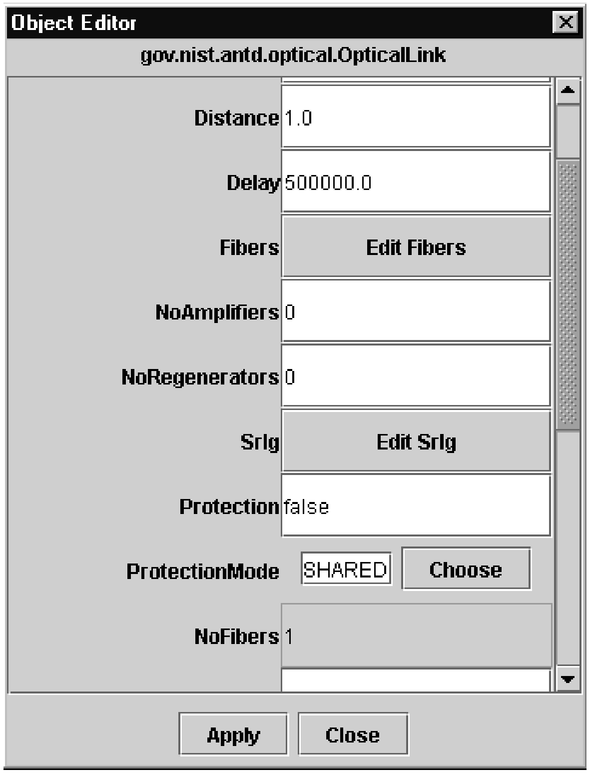



**Figure 24: Optical link editor**

**Distance:** The distance of this link in kilometers.

**Delay:** The delay of this link in clock ticks.

**Fibers:** Clicking on Edit Fibers displays the Fiber editor, which enables you to edit and view the properties of the fibers within this link.

**NoAmplifiers:** The number of amplifiers on this link.

**NoRegenerators:** The number of regenerators on this link.

**Srlg:** The Shared Risk Link group to which this link belongs. Clicking on **Edit Srlg** brings up an editor, which allows you to view and modify the Srlg of this link.

**Protection:** Indicates whether or not this is a protection link.

**ProtectionMode:** The protection mode of this link. Clicking on **Choose** allows you to change the protection mode.

**NoFibers:** The number of fibers in this link.

Many other attributes are available (Bit Error Rate, Failure). See [3] for more details.

Links in a failure state are displayed as red, while functional links are black. The state of a link may change during the simulation, and the color will change accordingly.

As for the nodes, it is not possible to modify the attributes of a link once the simulation is running, and a viewer will be called instead of the editor.

## 4.4 CONFIGURING THE ALGORITHMS

GLASS provides a set of algorithms that can be used for routing and wavelength assignment. The user can easily create and add new algorithms to the GLASS-TSC using the Algorithm panel.

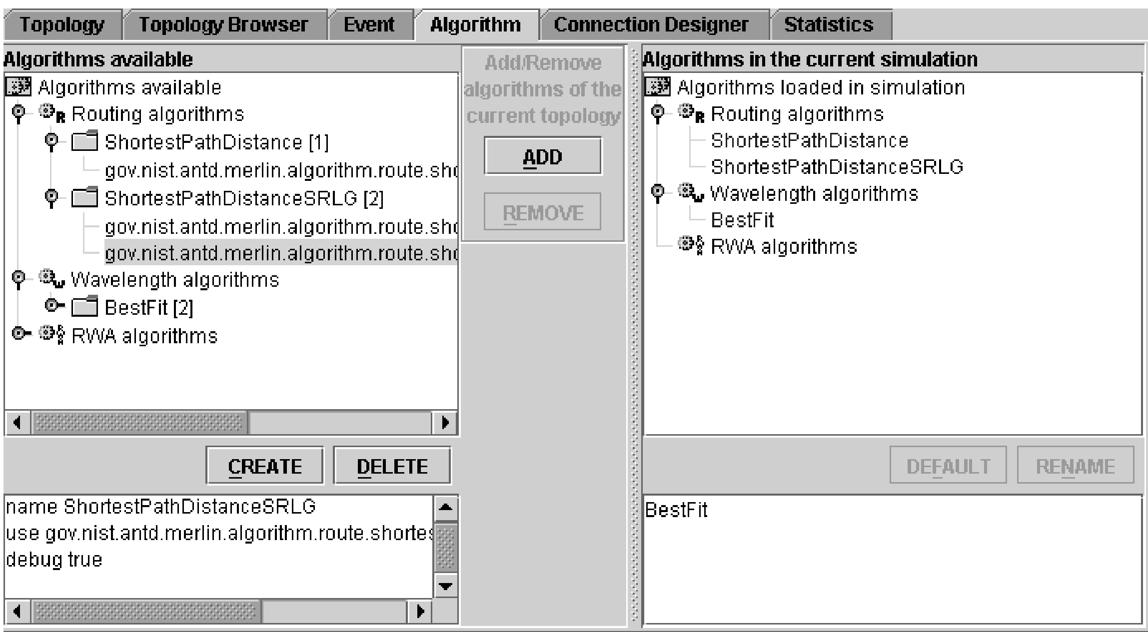



**Figure 25: Algorithm panel example**

The left part of the panel contains the list of algorithms that are available for all the topologies. The information about the algorithm (name, class, and configuration) is stored in the file cfg/AlgoConfig.dml.

The left panel also allows adding or removing algorithm out of the file. New algorithms are automatically saved in the file so that they can be used any time.

It is possible to have multiple times the same algorithm name with different class name or different configuration but you cannot have two algorithms with the same name loaded in the current topology.

The middle part of the panel contains 2 buttons to add or remove algorithm to the current topology. When added an algorithm in the topology, the GLASS-TSC makes sure that an algorithm name appears only once. If the user tries to add another algorithm with the same name, he will be asked to make the choice between either canceling his action or overwriting the algorithm by the new one.

The right part represents the list of algorithms currently loaded in the topology. The default algorithms are displayed in blue. There are two configurations of default algorithms: The first one is to select a routing algorithm and a wavelength algorithm; the second one is to select a Routing and Wavelength Assignment (RWA) algorithm. They can be changed anytime during the configuration of the topology but they are not save between two topologies.

## 4.5 CONFIGURING THE CONNECTIONS

There are multiple ways to configure and create the connections in the GLASS-TSC. We will explain each of them and their limitation.

First of all, in order to create connections, the user must have loaded some algorithms in the current topology (see 4.4).

### 4.5.1 CREATING CONNECTIONS FROM THE TOPOLOGYPANEL

The first way to create the topology is to use the button  located in the tool bar of the topology panel. This button allows the creation of unidirectional connection only.

To do so, the user needs to click on 2 nodes (source and destination of the connection). As the algorithms work only with optical network, the source and destination can only be OXC, OXCEdgeRouter or LSR.

Once the selection of the nodes is done, a connection configuration panel will appear as follow:

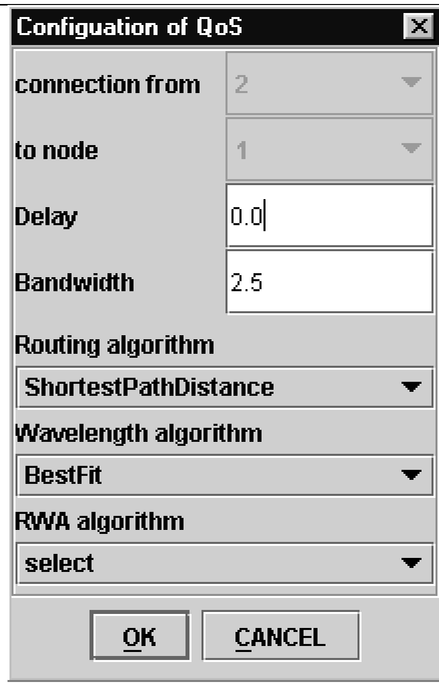



**Figure 26: Connection configuration**

The connection configuration is used to specify the quality of service for the connection. The attributes used are:

- **Delay**: this is the maximum delay of the path required for this connection.
- **Bandwidth**: the required bandwidth for the connection.
- **Routing algorithm**: The user can select one the routing algorithm that have been loaded into the topology.
- **Wavelength algorithm**: The selection of the wavelength algorithm that must be used after the routing algorithm
- **RWA algorithm**: the Routing and Wavelength Assignment algorithm to be used for the connection (the routing algorithm and wavelength algorithm can not be selected if using a RWA algorithm).

To valid the creation of the connection, click OK.

In the topology panel, it is only possible to create unidirectional connections. To create bidirectional connections, to edit or remove connections, use the ConnectionPanel as described next.

### 4.5.2 CONFIGURING CONNECTIONS USING THE GRAPHICAL CONNECTION PANEL

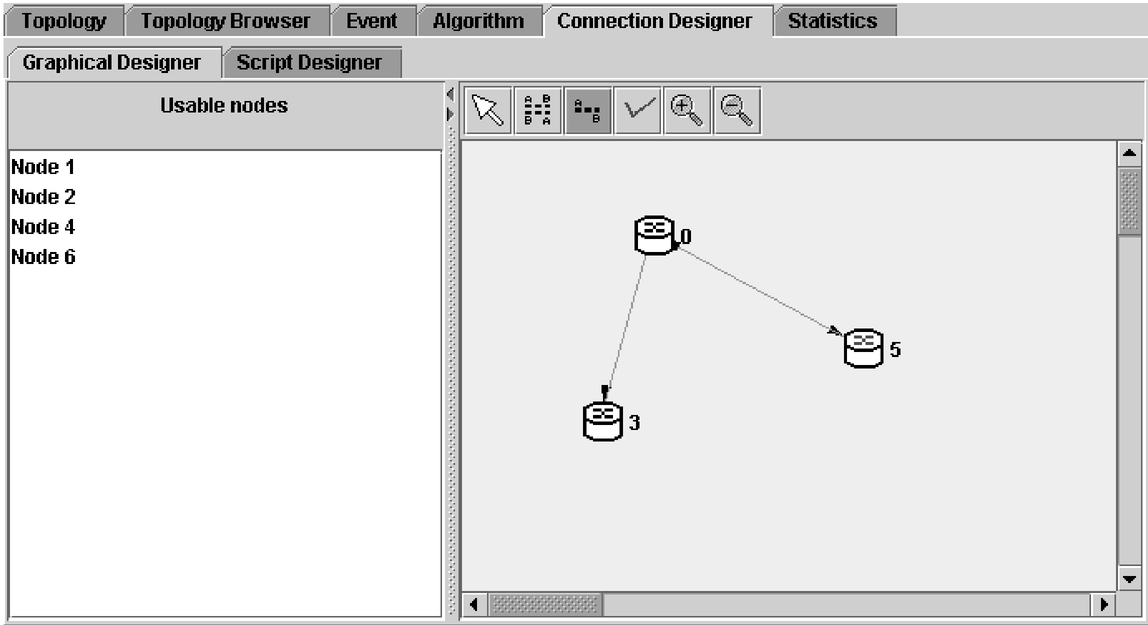



**Figure 27: Graphical connection designer**

As the algorithms are designed to create lightpath only, the Host and Router cannot be added to the connection designer. Only the Label Switching Router (LSR), OXCEdgeRouter, and OXC can be used for a connection.

To add a node select it from the Usable nodes list and click if the connection zone.

Two types of connection are possible, unidirectional or bidirectional. In fact, the bidirectional will only create 2 unidirectional connection is both directions. To do so, select the type of connection in the toolbar and then select the 2 nodes that are the source and the destination of the connection.

To view a static connection, simply right-click on it, and select properties or go in the scripted connection designer (see Figure 28).

### 4.5.3 Configuring connection in the Scripted view

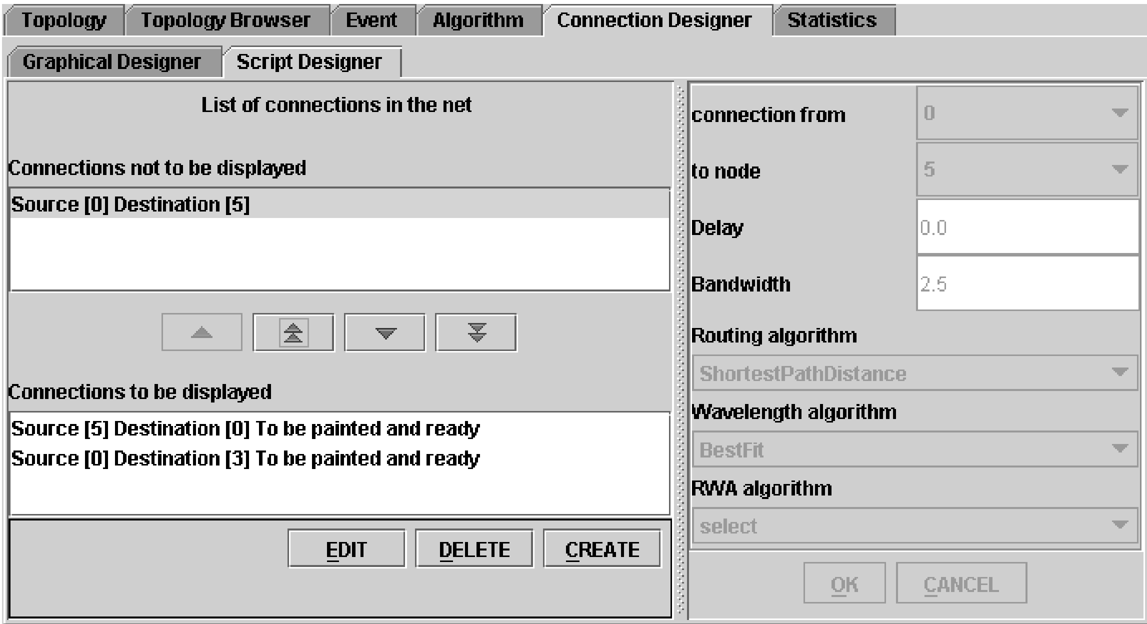



**Figure 28: Scripted connection designer**

To edit a connection, first click on it. Then click **EDIT**. You will now be able to change the values of the various attributes of the selected connection. Click **OK** when you are done.

To delete a connection, click on it then press **DELETE**.

The section "Connections not to be displayed" lists the connections that will not be displayed in the Topology workspace. Connections by default will be listed under Connections to be displayed when they are created. Use the arrows to move connections into the area that you want: either displayed, or not displayed.

To create a connection, click on **CREATE**, and fill in all of the attributes displayed on the right hand side of the workspace. Click **OK** to add the connection to the topology. Any errors for values of attributes are displayed at the bottom of the right-hand side of the screen.

## 4.6 Adding/Removing scripted events

When a simulation run is started, the dynamic events are generated as part of behaviour of the protocol sessions associated with network nodes. Pre-scheduled static events can be added from the Event workspace. Currently the available events are designed to simulate network failures: failure of

node (OXC, OXCEdgeRouter and LSR), ONIC, link, fiber, or lambda. The processes to handle these events must be configured in the protocol stack on the nodes where such events will occur [2]. To add an event, click on the **Event** tab to get to the Event work space:

**Figure 29: Event configuration**

To create an event:

1. Select the type of event that you want to create from the drop-down menu **New Event Type.** Events can occur at the Node, ONIC, Link, Fiber, or Lambda level.
2. Click on the **CREATE** button. This will display the event on the right hand portion of the screen under the categories **Attribute** and **Value**.
3. Add values to each attribute of the event. If the failure box is checked, this event will cause the given component to fail at the specified time. Time is in seconds. Click **OK** when you are done.

To delete or modify an event, select the event on the left-hand portion of the screen, and click **EDIT** or **DELETE**.

## 4.7 CREATING A TOPOLOGY USING THE RANDOM GRAPH GENERATOR

Topologies can also be created by using the Random Graph Generator, which randomly generates a topology according to a selected algorithm and the requisite input parameters.

Selecting **New Random→Generate** in the menu **File** creates a randomly generated topology of OXCEdgeRouters. The user is first prompted for the number of vertices (nodes):

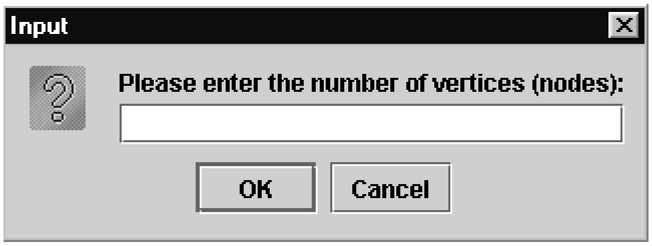



**Figure 30: Random graph input**

The user is then prompted to choose an algorithm for generating the vertices and edges (links):

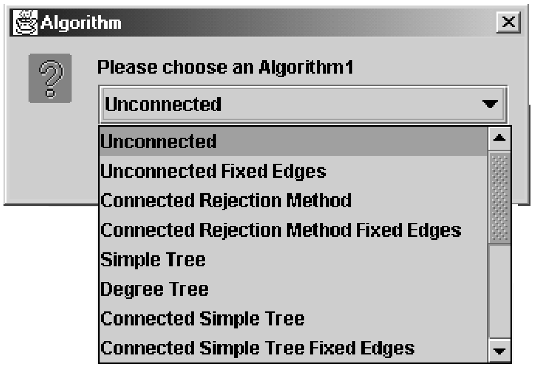



**Figure 31: Random  graph algorithms**

Depending on the algorithm chosen, the user then must define some other parameters. A dialog box will pop up asking the user for these additional parameters. For example, if Unconnected is chosen as the algorithm, the following dialog box will appear:

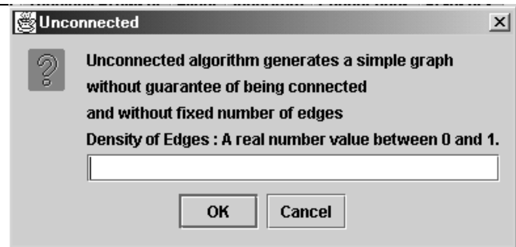



**Figure 32: Unconnected graph input**

If the input is valid, the generated Topology will be displayed in the workspace.

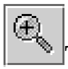
Figure 33 shows a network topology generated with the following parameters: 4 vertices, unconnected, and edge density of 1. The generated nodes are randomly placed on the topology workspace. You can rearrange the layout by dragging the nodes to appropriate place on the screen.

**Figure 33: A network topology created randomly**

The RandomGenerator only creates topology with OXCEdgeRouters connected by bidirectional optical links.

## 4.8 CHANGING THE VIEW OF THE TOPOLOGY

From within the topology workspace, you can zoom in or zoom out from the current topology.

This will zoom out from the topology from the point where the mouse was clicked.

This will zoom in upon the topology from the point where the mouse was clicked.

# 5 RUNNING THE SIMULATION

Once the configuration of the topology is done, the user can run a simulation. The simulation control panel allows the user to interact during the simulation.

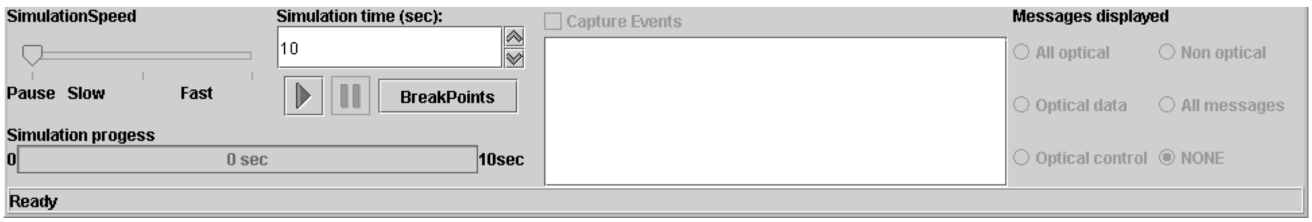



**Figure 34: The simulation control**

## 5.1 THE SIMULATION LENGTH

Before running the simulation, the user must specify the simulation length (in seconds).
It can be done from the menu **Simulator** or directly from the simulation control panel located on the bottom of the GLASS-TSC.
The simulation length must be larger than or equal to 1 second and must be an integer value.

## 5.2 BREAKPOINTS

The BreakPoint button allows the user to specify when the simulation must stop at certain time. This is very useful to stop the simulation just before or after an event occurs. The breakpoints can be modified before and during the simulation if the user clicks on the BreakPoints, the simulation will pause automatically and start when the configuration panel is closed.

The breakpoints are attached to a specific topology and are saved in the dml file. Even so it is possible to add and remove breakpoints during the simulation, it is useful to create them before running the simulation.

## 5.3 STARTING THE SIMULATION

To start the simulation, use the menu **Simulator** or press the button Play on the simulation control panel. The user will have the possibility to save before the simulation starts. Once the user starts the simulation, the control panel will be enabled for more control. The user can enable/disable displaying internal events, select the type of messages (being transmitted between nodes) to be displayed, and control the speed of the simulation.

## 5.4 CONTROLLING THE SIMULATION SPEED

The slider on the left side of the control panel allows the user to control the speed of the simulation.
The simulation progress shows the simulation length and the current simulation time.
Once the simulation starts, just move the slider to the right to start running the simulation.

The user can pause the simulation by pressing the pause button, and start it again by pressing the play button.

## 5.5 SELECTING THE MESSAGES DISPLAYED

On the right part, the user can select the type of messages that should be displayed during the simulation (Optical data, optical control, all optical, non-optical, all messages or no messages).
The simulation speed must be adjusted to see the messages going through the network.

Data messages in transit are large blue dots that travel along the links, control messages are shown in red, messages through non-optical link in green and the unknown type of messages in purple. Pausing the simulation and clicking on a message will result in a pop-up box that displays the contents and headers of that message:


**Figure 35: Example of message content**

## 5.6 Captures the Events

The GLASS-TSC is programmed to be notified when certain events occurred in the SSFNet simulation engine, for example, link failures, and creation, modification and deletion of objects. These events can be displayed in the middle of the control panel if the check box is selected. This can be a good debug tool to see the failure events and also all the messages sent from NICs.

# 6 Limitations/Bugs of the GLASS-TSC

The GLASS-TSC is under development, which means that some features may not be completely done and also that bugs may still exist in the coding. We would be happy if bugs are reported so that we can fix them.

## 6.1 Starting simulation multiple times

The GLASS framework extends SSFNet. The SSFNet has not been developed to handle multiple starts using the same java session. One of the biggest problem are the static attributes, that can not be reset (for example the package SSF.OS.sOSPF) and that interfere with the possibility to run twice a simulation.

## 6.2 Complex DML files

Even though we are able to see all type of nodes in the GLASS-TSC, the current version of GLASS-TSC does not provide the capability to open DML files that contains complex configuration with multiple level of net.

# 7 REFERENCES

[1] Scalable Simulation Framework (SSF)

URL: http://www.ssfnet.org

[2] Nodes in GMPLS Lightwave Switching Simulator (GLASS)

By NIST/ANTD

[3] Links in GMPLS Lightwave Switching Simulator (GLASS)

By NIST/ANTD