

# ***Introduction***

---

## **Overview**

Welcome to Modeler!

This tutorial teaches you the basics of using Modeler. If you are new to Modeler, this short introduction will help get you started.

If you are doing this tutorial for the first time, read the following pages for important information about using this tutorial and Modeler.

## About the Models

This tutorial was written for use with release 10.0 of the Modeler software and the August 2003 models. If you are using a different release, you might get different results or be unable to complete some lessons.

To complete the tutorials, you must install the Modeler standard models, which include the required tutorial models. These models are normally installed when you install the Standard Model Library.

Modeler standard models cover common protocols and vendor devices. The standard models are in the subdirectories under the OPNET release directory (<reldir>):

**<reldir>\models\std\<protocol\_name>**

**<reldir>** describes the directory that contains the current Modeler software.

You can find your **<reldir>** by doing the following:

- 1 Select **Help > About This Application** in the main menu.
- 2 In the About OPNET Modeler dialog box, click on the **Environment** tab, then expand the **System Information** section.
- 3 Under System Information, find the **OPNET release directory**.

For example, the **<reldir>** for a default installation of this release of Modeler on Windows is

**C:\PROGRA~1\OPNET\10.0.A**

---

The tutorials use the Windows convention of the backslash character (\) as the separator in directory pathnames. If you are using Solaris, replace the backslash with a forward slash (/).

---

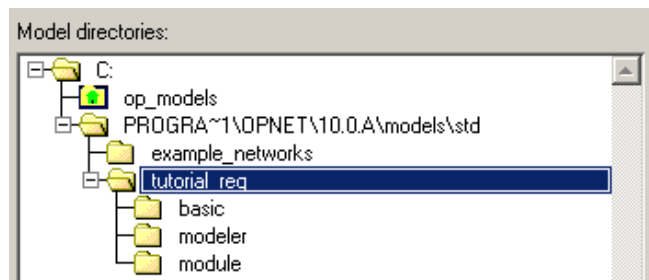
## Opening Tutorial Files

In several tutorials, you will be asked to open an OPNET model file and save it with a unique prefix, such as your initials (<your\_initials>). This way, several users can create and complete their own copy of the same tutorial without interfering with each others' work.

When you do a tutorial, you are asked to open special tutorial model files. These model files are required to do the tutorial; they are located in the **<reldir>\models\std\tutorial\_req** directory or its subdirectories.

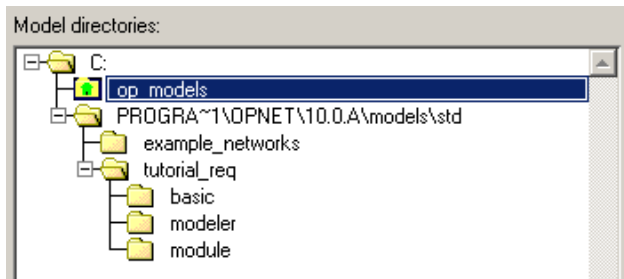
When you are asked to open model files that are NOT prefixed with your initials, make sure you navigate the directory structure in OPNET's open file browser (in the left pane) to the **tutorial\_req** directory.

### Navigating to the tutorial\_req Directory



When you save a new model file, or use the Save As... command to save an existing model file with your initials, remember to navigate to the OPNET working directory (normally **<home>\op\_models**) to save your files.

### Navigating to the op\_models Directory



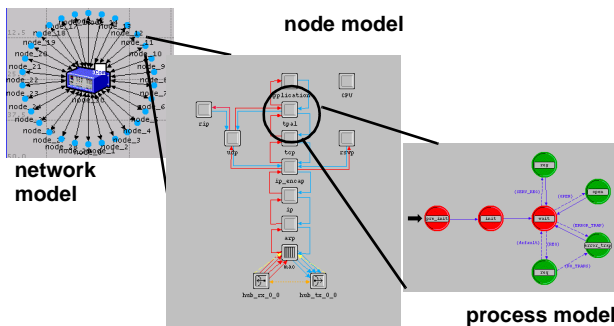
## About the Lessons

In this tutorial, you learn how to use the Modeler interface and some basic modeling theory. You will also learn how to use the Modeler features to build and analyze network models.

You can download the latest version of this tutorial from the OPNET web site. Visit [www.opnet.com/support/home.html](http://www.opnet.com/support/home.html) and select the Product Updates link.

Each lesson presents a modeling problem for you to solve by building a network, node, or process model, gathering statistical information about the network, and analyzing those results.

### Network, Node, and Process Models



Each lesson helps you become familiar with Modeler and demonstrates the range of problems Modeler can solve.

We suggest that you do the lessons and topics in sequence.

---

Most lessons have Key Concept paragraphs like this one that contain new information about Modeler or describe important aspects of modeling theory.

---

Position this tutorial window next to Modeler on your screen so that you can view both windows at the same time.

By default, the tutorial window always stays below the OPNET window. If you want to change this, set the **modeler.tutorial\_top** preference to **TRUE** (for information about changing preferences, see the Productivity Features lesson, Preferences topic).

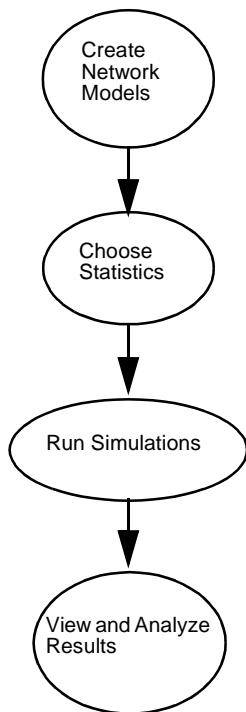
## About Modeler

Before you start, you should understand the Modeler workflow, the workspace, and the editors. Become familiar with these essential features, then begin the tutorial lessons.

The workflow for Modeler (that is, the steps you use to build a network model and run simulations) centers around the Project Editor.

In this editor, you can create a network model, choose statistics to collect from each network object or from the whole network, execute a simulation, and view results.

Your first look at Modeler in Lesson 1 demonstrates how to use the Project Editor to build a small internetwork. You can also use advanced editors for specialized needs.



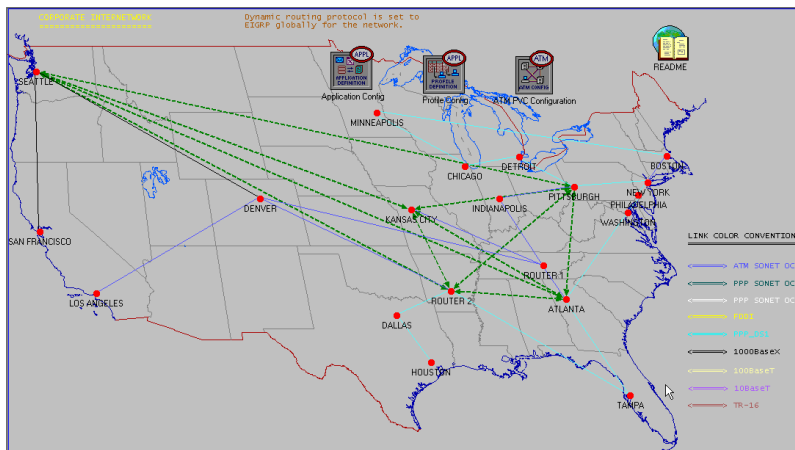


For the nuts-and-bolts aspects of modeling, such as programming the underlying process of a particular network object or defining a new packet format, you will need to use additional editors. Each editor is described in detail on the following pages.

## The Project Editor

The Project Editor is the main staging area for creating a network simulation. From this editor, you can build a network model using models from the standard library, choose statistics about the network, run a simulation, and view the results.

### A Network Model in the Project Editor

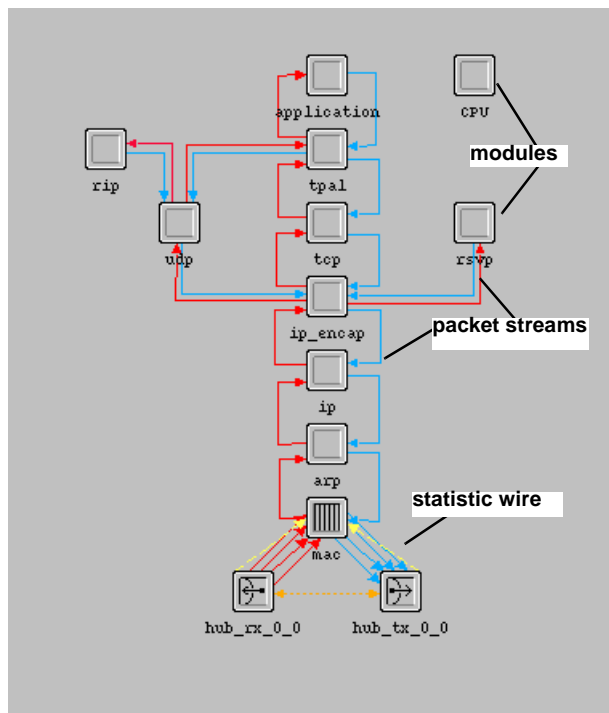


You can also create node and process models, build packet formats, and create filters and parameters, using specialized editors that you can access from the Project Editor.

## The Node Editor

The Node Editor lets you define the behavior of each network object. Behavior is defined using different **modules**, each of which models some internal aspect of node behavior such as data creation, data storage, etc. Modules are connected through **packet streams** or **statistic wires**. A network object is typically made up of multiple modules that define its behavior.

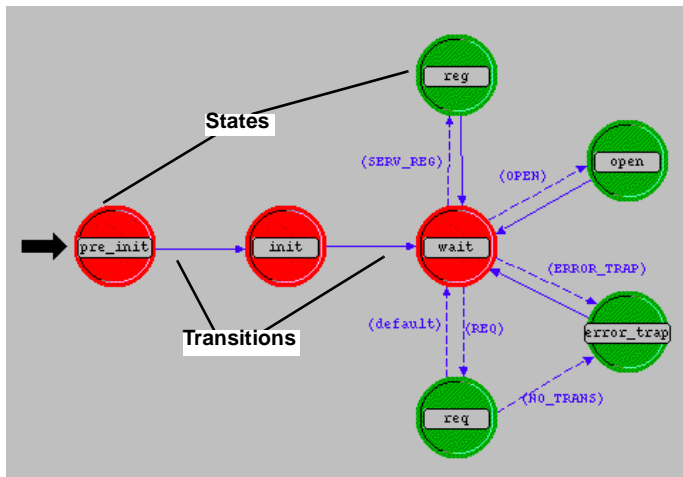
## Node Model



## The Process Model Editor

The Process Editor lets you create process models, which control the underlying functionality of the node models created in the Node Editor. Process models are represented by finite state machines (FSMs), and are created with icons that represent **states** and lines that represent **transitions** between states. Operations performed in each state or for a transition are described in embedded C or C++ code blocks.

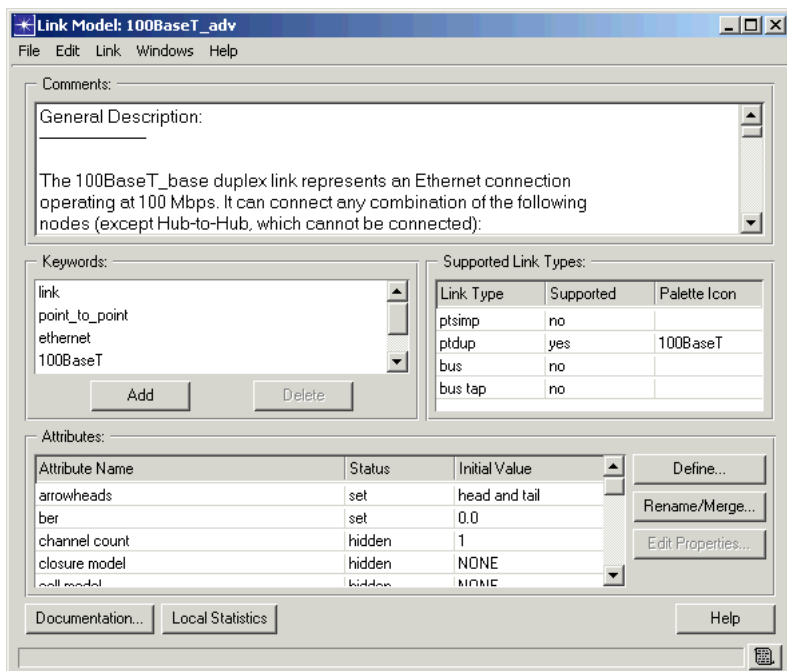
### Process Model



## The Link Model Editor

The Link Model Editor lets you create new types of link objects. Each new type of link can have different attribute interfaces and representation. Comments and keywords can also be specified for easy recognition.

### Link Model Editor



## The Demand Editor

The Demand Editor lets you define demand models; each demand object's underlying model determines its attribute interfaces, presentation, and behavior.

You create or modify the demand model specifications in the Demand Editor dialog box.

### Demand Editor

**Demand Model: ip\_traffic\_flow**

File Edit Demand Windows Help

Comments:

IP Traffic Flow:


Represents the IP layer traffic flow between the specified "source" and "destination". The following attributes can be configured for this flow:

Keywords:

Add Delete

Demand Properties:

Type: Flow

Icon:  IP\_demand

☒ Source site required

☒ Destination site required

☐ Path required

Acceptable Path Models:

Edit...

Attributes:

Attribute Name	Status	Initial Value
arrowheads	hidden	tail
color	set	#0000FF
financial cost	set	0.0
line style	set	dotted
symbol	hidden	none
technology	hidden	IP
thickness	set	2

Define...

Rename/Merge...

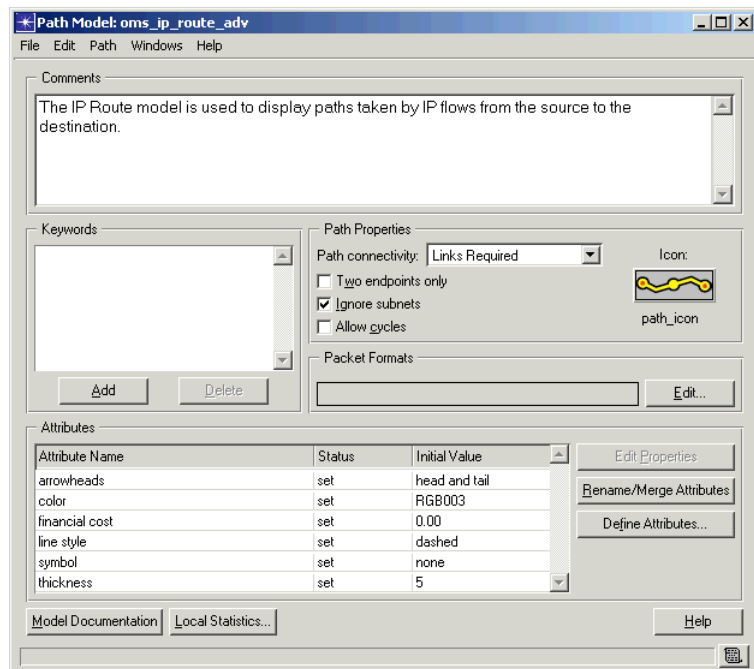
Edit Properties...

Documentation... Local Statistics... Help

## The Path Editor

Use the Path Editor to create new path objects that define a traffic route. Any protocol model that uses logical connections or virtual circuits (MPLS, ATM, Frame Relay, etc.) can use paths to route traffic.

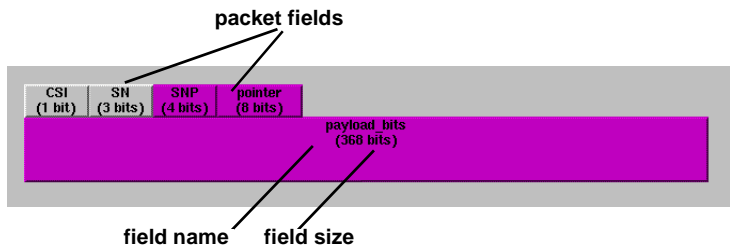
### Path Editor



## The Packet Format Editor

The Packet Format Editor lets you define the internal structure of a packet as a set of fields. A packet format contains one or more **fields**, represented in the editor as colored rectangular boxes. The size of the box is proportional to the number of bits specified as the field's size.

### Packet Format Editor

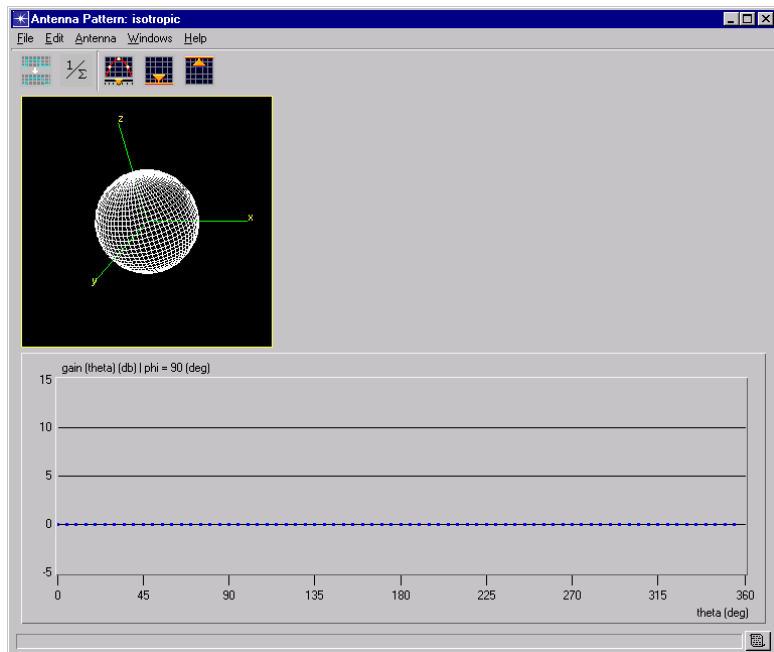




## The Antenna Pattern Editor (with Wireless Module)

In Modeler/Wireless, the Antenna Pattern Editor lets you model the direction-dependent gain properties of antennas. Modeler can use these gain patterns to determine gain values, given knowledge of the relative positions of nodes.

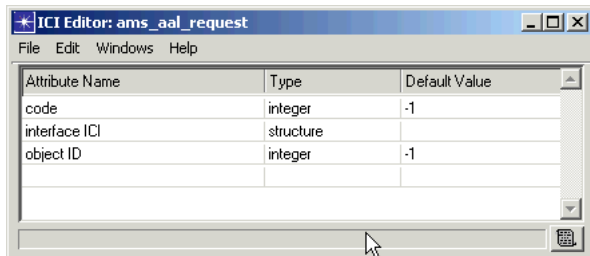
### Antenna Pattern Editor



## The ICI Editor

The ICI (Interface Control Information) Editor lets you define the internal structure of ICIs. ICIs are used to formalize interrupt-based interprocess communication.

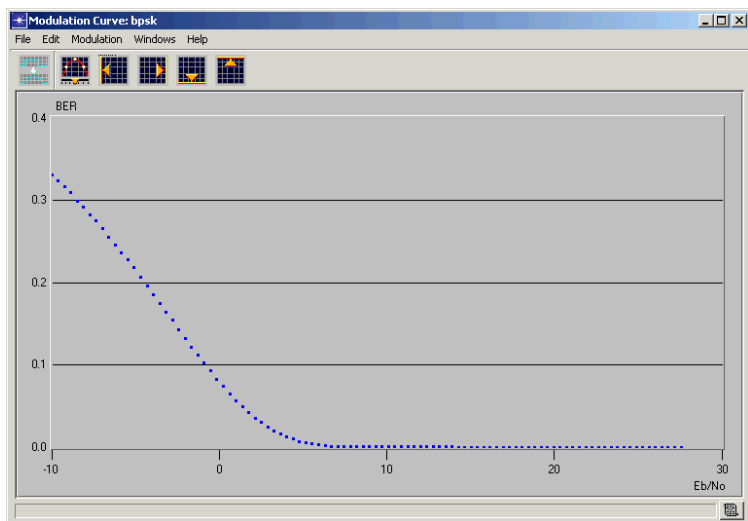
### Internal Formats of an ICI in the ICI Editor



## The Modulation Curve Editor (with Wireless Module)

In Modeler/Wireless Module, the Modulation Curve Editor lets you create modulation functions to characterize the vulnerability of an information coding and modulation scheme to noise. These modulation functions plot the bit error rate (BER) of an information signal as a function of the effective signal-to-noise ratio ( $E_b/N_0$ ).

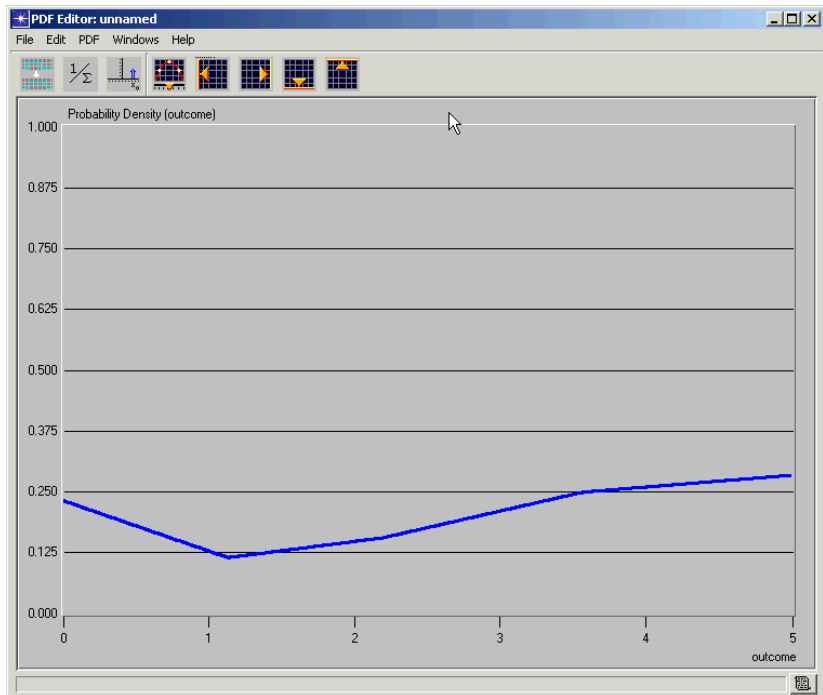
### A Modulation Curve



## The PDF Editor

The PDF (Probability Density Function) Editor lets you describe the spread of probability over a range of possible outcomes. A PDF can model the likelihoods associated with packet interarrival times, or it can model the probability of transmission errors.

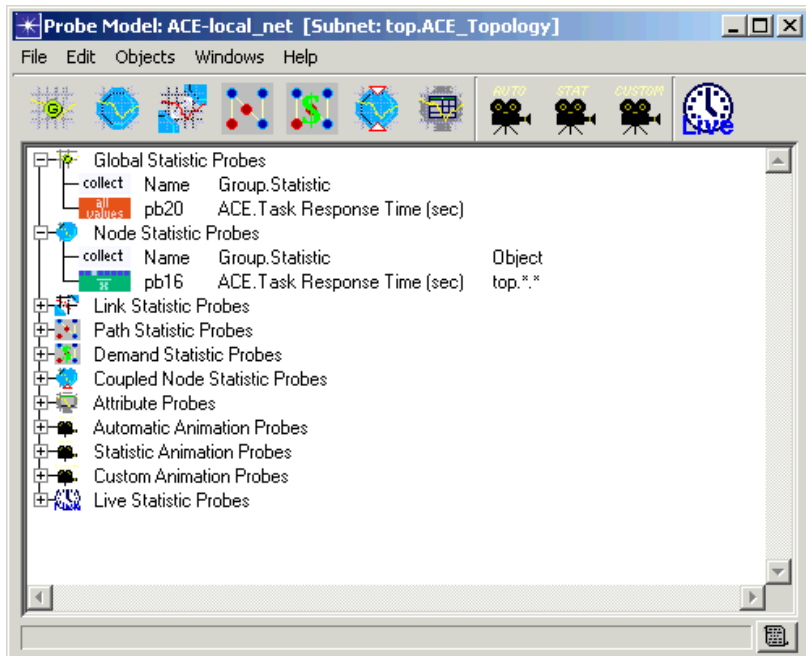
### A PDF Modeling Interarrival Times



## The Probe Editor

The Probe Editor lets you specify the statistics to be collected during simulation. While you can do this in the Project Editor, you can also set additional characteristics of each probe with the Probe Editor. There are several different types of statistics that can be collected using different probes, including global statistics, link statistics, node statistics, attribute statistics, and several types of animation statistics.

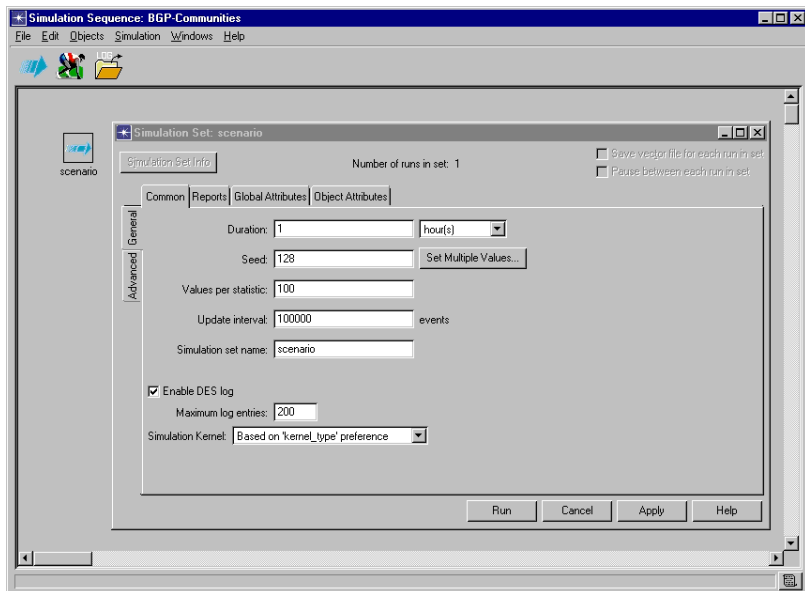
### Probe Editor Dialog Box with Global Statistic and Node Statistic Probes



## The Simulation Sequence Editor

Although you can run simulations from within the Project Editor, you may want to specify additional simulation constraints in the Simulation Sequence Editor. Simulation sequences are represented by **simulation icons**, which contain a set of **attributes** that control the simulation's run-time characteristics.

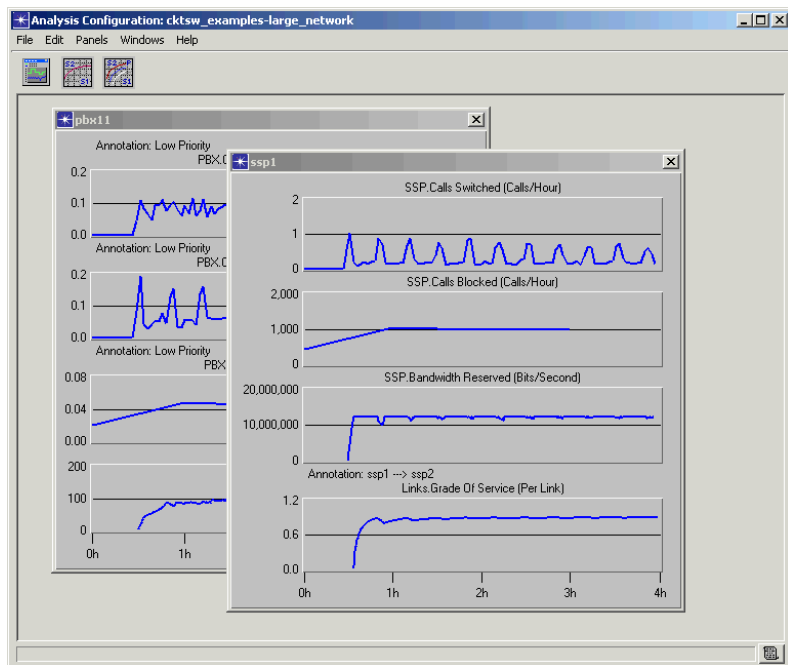
### Simulation Sequence Editor with icon and Simulation Set Dialog Box



## The Analysis Tool

Although you can view simulation results in the Project Editor, the Analysis Tool has several useful additional features. You can, for example, create scalar graphs for parametric studies, define templates to which you apply statistical data, and create analysis configurations that you can save and view later.

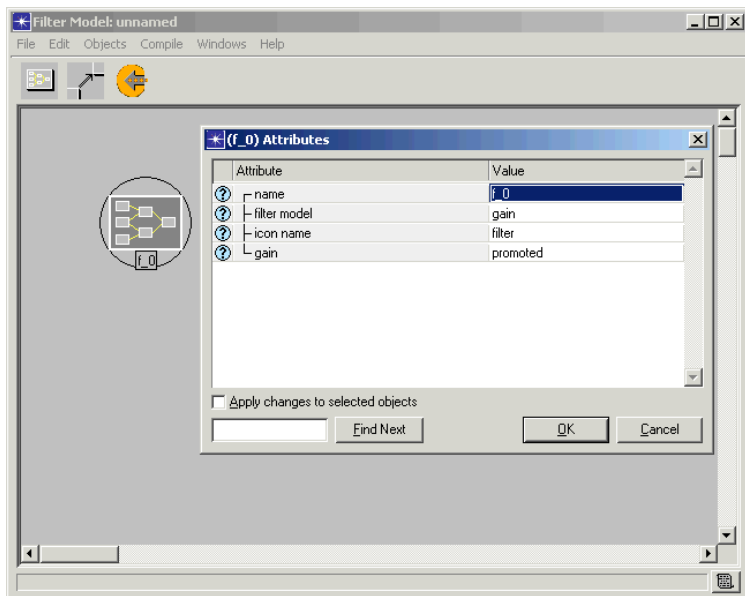
### Analysis Tool Showing Four Graphs



## The Filter Editor

Although Modeler comes with built-in data filters, the Filter Editor enables you to create additional filters. You can build new filter models by combining existing models with each other.

### Filter Editor with Filter Icon and Attributes Dialog Box



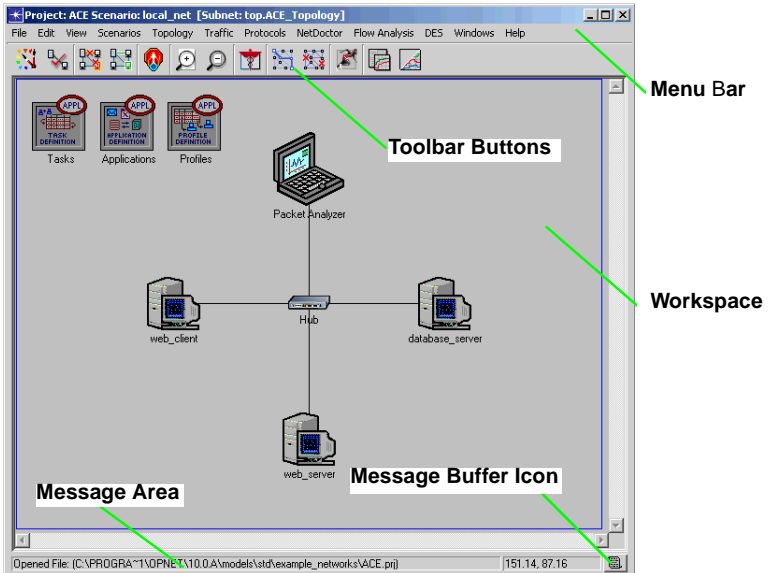


## The Project Editor Window

There are several areas in the Project Editor window that are important for building and executing a model. These are shown on the following pages.

When you open an existing project, your screen should look similar to the following figure.

### Project Editor Window



## ***The Menu Bar***

The menu bar is at the top of the editor window. It organizes all the non-context-sensitive editor operations into a set of topical menus.

The exact set of menus and menu operations available change according to the product modules that are enabled.

Context-sensitive editor operations are available by right-clicking on an object or on the background of the workspace.

## ***Toolbar Buttons***

Several of the more commonly used menu bar selections can also be activated through toolbar buttons. The buttons shown in the following figure appear in the Project Editor:

### **Toolbar Buttons in the Project Editor**



You will use many of these toolbar buttons in the tutorial.

<b>1</b> Open Object Palette	<b>8</b> Configure/Run NetDoctor
<b>2</b> Verify Links	<b>9</b> Configure/Run Flow Analysis
<b>3</b> Fail Selected Objects	<b>10</b> Configure/Run Failure Analysis
<b>4</b> Recover Selected Objects	<b>11</b> Configure/Run Discrete Event Simulation (DES)
<b>5</b> Go to Parent Subnet	<b>12</b> View Results
<b>6</b> Zoom to Rectangle	<b>13</b> Hide/Show Graph Panels
<b>7</b> Zoom to Previous	

### ***The Workspace***

The central, scrollable region of the editor window is the workspace. The network model appears in the workspace, where you can select and drag network objects, and choose context-sensitive menu operations by right-clicking on the background.

### ***The Message Area***

The message area is located at the bottom of the editor window. It provides information about the status of the tool.

#### **Message Area**

No reports have been generated for the project (Frame\_Relay) scenario (attr\_based\_pvc).

You can left-click on the icon next to the message area to open the message buffer window.



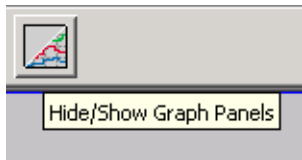
The message buffer window shows a list of messages that have appeared in the message area. You can open the message buffer window if part of the message line is truncated in the message area or if you think a later message has replaced an important alert or notification.

### **Tooltips**

If you rest your cursor over a tool button or a network object in the workspace, a help balloon appears. The tooltip describes one of the following:

- the action that occurs if the button is pressed
- information about the network object

### **Tooltip**



## Documentation

From time to time you may have questions about certain aspects of the tutorial, the tool, or the models. You can refer to these sources:

- **Built-in documentation** is available throughout the tool. For example, you can get model descriptions, attribute definitions, and statistic descriptions by selecting an item and clicking on the Details button. Some dialog boxes also have Help buttons that bring up additional descriptive information.
- **OPNET product documentation.** Select **Help > Product Documentation**.
- **FAQ** (Frequently Asked Questions). Point your browser to **[www.opnet.com/support](http://www.opnet.com/support)**.

You need your user name and password to access the web FAQ. OPNET Technical Support normally sends your user name and password by email after you register your OPNET software for purchase or evaluation.

- **Recent updates** to the tutorial lessons and models can be downloaded from the web site: **[www.opnet.com/support](http://www.opnet.com/support)**.

- **Model User Guides** provide the latest information on the protocol models and how to use them. Select **Protocols > <protocol name> > Model User Guide**.

Now you are ready to begin the tutorial. Return to the main tutorial menu and choose **Small Internetworks** from the list of available lessons. Have fun learning Modeler!