Chapter 3

Telecom Environment

The telecom modeling facilities in the netWorks application were originally designed to investigate performance issues in connection-oriented networks, such as a telephone network. The focus was on simulating and measuring transmission impairments such as transmission delay, noise, and error rates [Bell Labs 1983] and the simulated traffic generation functions were fairly basic. The application has been enhanced to address issues of network capacity, reliability, and availability, and the traffic generators have been improved as well.

In a typical telecom modeling scenario you have various geographic locations with station equipment (telsets, faxes, and so forth) interconnected through a switched network. You create a model network in the drawing panel using telecom equipment models and then, using the Attribute palette, select the performance attributes you want to investigate in your simulation. You must also identify the paths through your model that are of interest; depending on the size and scope of your model network, a subset of paths may be more appropriate for your simulation than studying all possible paths. Once the path identification is completed, you can start generating simulated traffic over your model network and begin collecting data.

Telecom Equipment Models

Models and Systems

As discussed in Chapter **??**, "**??**," there are two types of equipment models in netWorks—simple and compound. To be consistent with telecom terminology, compound equipment models are referred to as *systems* in the telecom environment. A system configuration may be fairly simple like the Echo Canceler model or a bit more detailed like the Digital Tandem Switching System. See Figure 3.1.

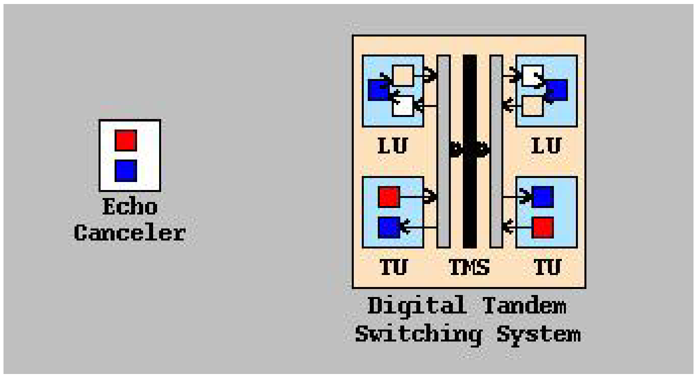


Figure 3.1. Telecom System Examples

As mentioned in Chapter **??**, “**??**,” only simple equipment models have attributes associated with them and only simple equipment models can be connected by arcs. The primary purpose of systems (compound models) is to speed the model network creation process and add visual cues to the network representation.

Both families of simple equipment models and systems employ the notions of hierarchy and inheritance. These hierarchies can be represented as trees, with the simpler models serving as building blocks for the more detailed models. Figure 3.2 shows the simple equipment model hierarchy.

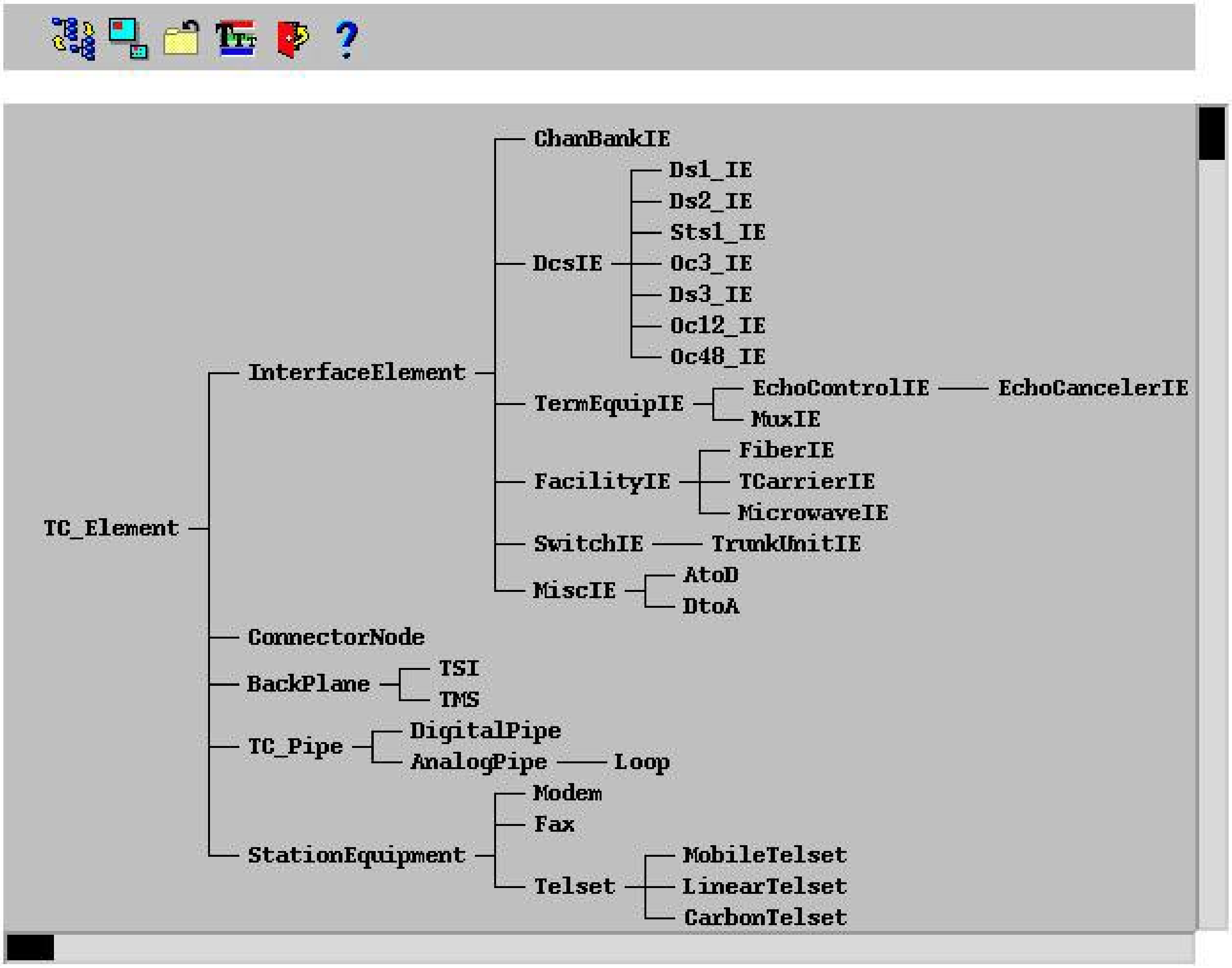


Figure 3.2. Telecom Simple Model Browser on the Simple Equipment Model Hierarchy

Each simple equipment model has a list of attributes associated with it, and the inheritance paradigm provides a convenient means of propagating attributes throughout the model tree. For example, by using inheritance, any attributes defined on the TC_Element model will be available in all subsequent models, like InterfaceElement and SwitchIE. Similarly, any attribute additions or modifications made to intermediate models in the hierarchy will be reflected in all other models derived from them. For example, when the attribute Length is added to the TC_Pipe model, it is automatically inherited by DigitalPipe, AnalogPipe, and Loop models as well.

In the systems hierarchy, the models inherit configurations instead of attributes. A configuration is a template defining how many components a system will have and how those components are connected with arcs. Some configurations supply no internal arcs while, with others, the internal arcs are critical to the system's implementation. Unlike with simple equipment models, you cannot change the configuration of a system—the location of a system in the systems hierarchy determines its configuration. The default systems hierarchy is shown in Figure 3.3.

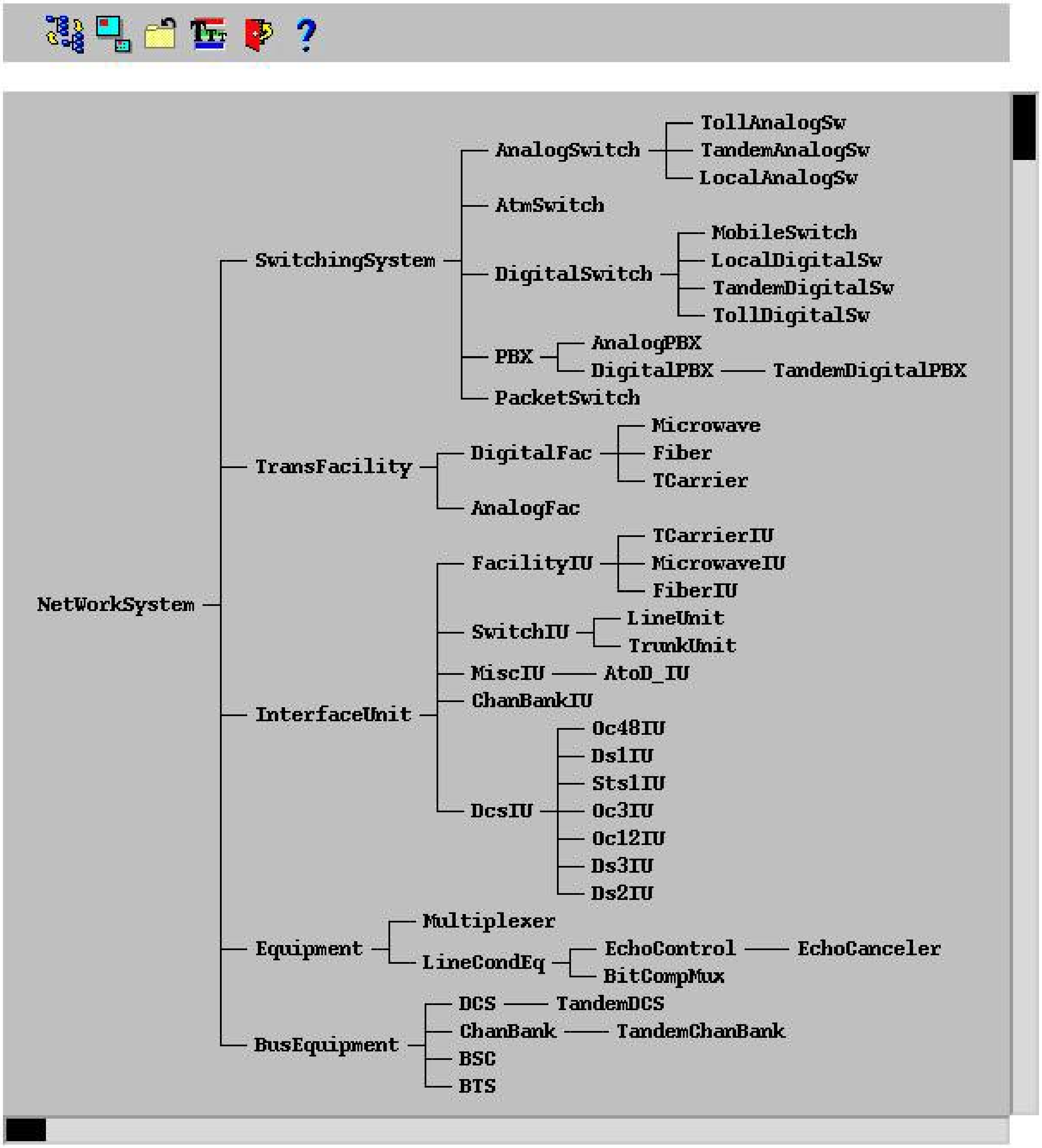


Figure 3.3. Telecom Systems Browser on the Systems Configurations Hierarchy

Two-Wire versus Four-Wire Equipment

In real telecommunication networks, some equipment uses the same wiring to transmit and receive information while other equipment might require separate wiring for transmitting and receiving data. This distinction is commonly referred to as two-wire versus four-wire equipment. An ordinary telephone set is an example of a two-wire piece of equipment whereas a lightwave carrier system is a four-wire facility.

These distinctions are carried through in the netWorks application. Two types of arcs are provided to accommodate bidirectional and unidirectional traffic between equipment models. And while it is the your responsibility to provide the right type of arc between models, in some cases netWorks will try to help you make the right choice. The wrong type of arc between equipment models will affect the path generation process discussed in the “Paths” section.

Browsers

Given the multitude and proliferation of telecommunications equipment in the marketplace, it is virtually impossible for any tool to enumerate all your networking options, let alone provide models for all available equipment. The netWorks application instead provides models for generic categories of equipment and systems models that you can modify and extend to meet your needs. You can create your own permanent equipment models or systems models, or you can edit existing models using the telecom equipment browsers. The model and system browsers are accessible through the drawing panel pop-up menu.

The telecom model browser presents a hierarchical view of all the default simple equipment models available in the application. (See Figure 3.2.) Using its pop-up menu, you can invoke routines for editing the default attributes associated with a model or create a new model derived from one of the existing models. You can use your new or modified equipment models in your model network as soon as you exit the browser, or you can save your changes in a SAS data set for use in subsequent netWorks sessions. Most often you will use the equipment model browser in conjunction with the systems browser.

The systems browser also displays its information using a hierarchical format (see Figure 3.3), and you create new systems using a routine similar to that in the simple equipment models browser. However, the editing process for a system model is very different from a simple equipment model. As mentioned earlier, a system model enforces a predefined component configuration; that is, the system has well-defined positions for holding specific types of models. To modify the default components in a system model, you first invoke the Edit routine on the system model in the systems browser through its pop-up menu. This generates a new window displaying the configuration of that system. (See Figure 3.4.)

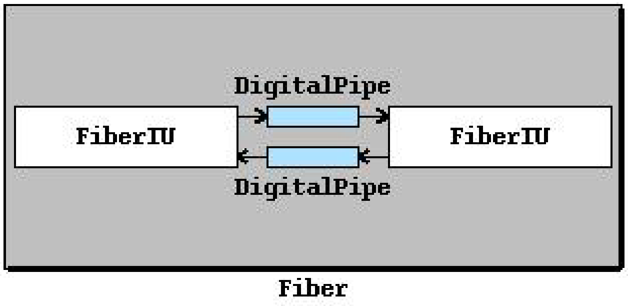


Figure 3.4. Sample Telecom Systems Editor

You can then drag and drop items from either the model or systems browsers (whichever is appropriate) into the component slots in the systems editor window. Some systems restrict what models can be dropped in their component slots, so you may first have to create some new equipment or system models before constructing a larger, new telecom system model. As with the model browser, you can save changes for use in later netWorks sessions.

If you create new models, or edit existing ones, you will probably want to save these models for use in future netWorks sessions. You can use **File**⇒**KnowledgeBase**⇒**Save** to save your changes in a SAS data set. Similarly, if you use new models in your network model, you will have to reload the base models into netWorks before you can load a saved network model that uses them.

Attributes

Each simple equipment model has a collection of attributes associated with it. There are seven types of attributes defined in the netWorks application:

- Descriptive
- Performance
- Distributional
- Scalar
- Scalar Distributional
- Ratio
- Capacity

A Descriptive attribute is used for informational purposes only, and it has no effect on simulation results. You might use a Descriptive attribute to store a name of a model or the author, creation date, and so forth. This type of attribute is very useful when using netWorks to create a repository for network equipment information. The remaining attribute types fall under the umbrella of Performance attributes, and all of these have some form of numeric value(s) associated with them. These attributes can have an impact on your network simulation results. The primary difference between the various types of Performance attributes is how their associated numeric values are calculated. The following sections present a brief synopsis of the family of Performance attributes.

Performance Attribute

The Performance attribute provides the base for the entire family of Performance attributes. In addition to its name, it stores four pieces of information about the attribute:

- *minimum value* (lowest possible numeric value allowed)
- *maximum value* (largest possible numeric value allowed)
- *current value*
- *concatenation rule* (algorithm to "add" two attributes of this type)

A Performance attribute returns the number stored in its current value slot when you retrieve the attribute's value. You have to manually edit a Performance attribute to change its current value. The minimum and maximum value numbers are used as constraints when you edit the current value slot.

The default Performance attributes in the netWorks application are

- Constant (default:1)
- Cost (default:1)
- Loss (default:0)
- C_NotchNoise (default:0, max:95)
- Length (default:0)
- ROLR (default:0)
- TOLR (default:0)
- Capacity (min:0, max:10000)
- CallsAttempted (default:0)
- CallsBlocked (default:0)

The definitions for the specific attributes can be found in [Bell Labs 1983], [Fennick 1988], and [Bell Labs 1982].

Distributional Attribute

A Distributional attribute stores a statistical distribution as well as all the Performance attribute information. When a model queries the value of a Distributional attribute, it samples from its statistical distribution and returns that value, subject to its minimum value and maximum value constraints. The statistical distributions available in netWorks are described in Appendix **??**, “**??**,”.

The default Distributional attributes in the netWorks application are

- C_MsgNoise (default:Exponential(1), max:90)
- BitErrorRate (default:Exponential(1))
- BlockErrorRate (default:Exponential(1))
- ErrorFreeSeconds (default:Exponential(1))
- TimeToFailure (default:Exponential(10000))
- TimeToRestore (default:Exponential(1))
- CallHoldingTime (default:Exponential(1))
- InterCallTime (default:Exponential(1))

Scalar Attribute

A Scalar attribute uses the value of another attribute, *factor*, as well as the value it stores in its current value slot to compute the value to return upon a query. That is, when a Scalar attribute gets a request for its value, it first retrieves the value from some other attribute (factor) and then returns the product of this number and the number stored in its current value slot. A common use of this type of attribute is when an equipment model has an attribute named Length associated with it. For example, the transmission delay associated with a lightwave facility could be calculated using a Scalar attribute that stores a delay per kilometer value and uses it along with the value of the Length attribute to calculate the total transmission delay for the facility. In this case, the value of factor would be Length.

Scalar Distributional Attribute

A Scalar Distributional attribute combines the functionality of a Scalar and Distributional attribute. When it is queried for its value, a Scalar Distributional attribute returns the product of some other attribute (factor) and a sample from its own statistical distribution.

The default Scalar Distributional attributes in the netWorks application are

- AbsoluteDelay (default:Exponential(1), factor:Constant)

Ratio Attribute

A Ratio attribute has two other attribute names associated with it. When you query the value of a Ratio attribute, it retrieves the values of these two attributes and returns the result of dividing the first attribute value by the second attribute value.

The default Ratio attribute in the netWorks application is

- BlockingRate (numerator:CallsBlocked, denominator:CallsAttempted)

Capacity Attribute

A Capacity attribute also uses another attribute to determine the value it returns upon query. This type of attribute retrieves the value of another attribute, along with the other attribute's maximum value, and returns the result of their division.

The default Capacity attribute in the netWorks application is

- Utilization (factor:Capacity)

Concatenation Rules

Simple addition is not always the appropriate technique to "add" two values from attributes of the same type together. Concatenation rules provide algorithms to add various types of attributes. The five attribute concatenation rules available in netWorks are as follows:

- Sum (+)
  $r = a_1 + a_2$
- PowerSum ($\oplus$)
  $r = 10 * log_{10}(10^{\frac{a_1}{10}} + 10^{\frac{a_2}{10}})$
- NegPowerSum ($\ominus$)
  $r = -10 * log_{10}(10^{\frac{-a_1}{10}} + 10^{\frac{-a_2}{10}})$
- LengthFactor
  $r = a_1 + (a_2 * Length)$
- NoiseLoss
  $r = (noise_1 \oplus noise_2) - loss_2$

Browser

As with the telecom equipment models, you may need attributes other than those already provided in the netWorks application, or the default values for a given attribute may not be appropriate for your simulation. You can modify existing attributes, or create new ones, using the Attribute registry browser. It is accessible with the drawing panel pop-up menu, and it is shown in Figure 3.5. Its usage is fairly straightforward.

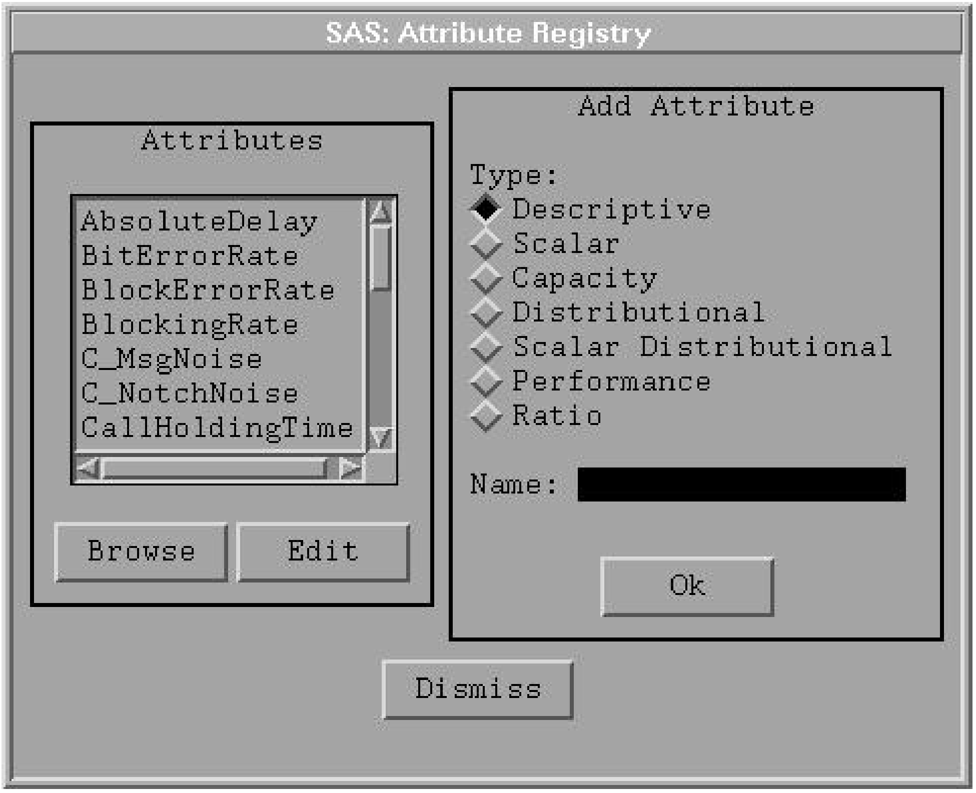


Figure 3.5. Attribute Registry Browser

Building a Telecom Network Model

You build a telecom network model as described in Chapter **??**, “**??**,” ensuring that four-wire and two-wire equipment models are connected with the appropriate types of arcs. The Telecom Equipment palette (Figure 3.6) provides quick access to some of the more commonly used telecom models.

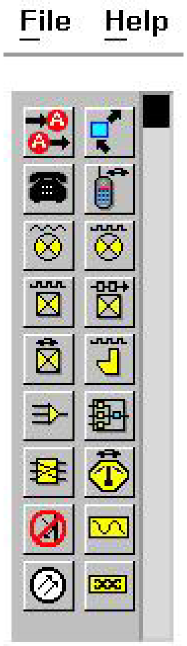


Figure 3.6. Telecom Palette

If your network model will be used for simulation purposes (as opposed to documentation purposes only), it is important to include station equipment models (for example, telsets) in your network model. These models provide the primary traffic generators for a simulation. (More on traffic generation in the "Traffic Generators" section.)

Running a Telecom Simulation

Before you can successfully run a simulation on your network model, the following criteria must be met:

- The equipment models must be connected properly.
- You must have identified which paths through the network you want to investigate.
- Your network model must have traffic generators.
- You must have selected attributes to be investigated during the simulation.

Paths

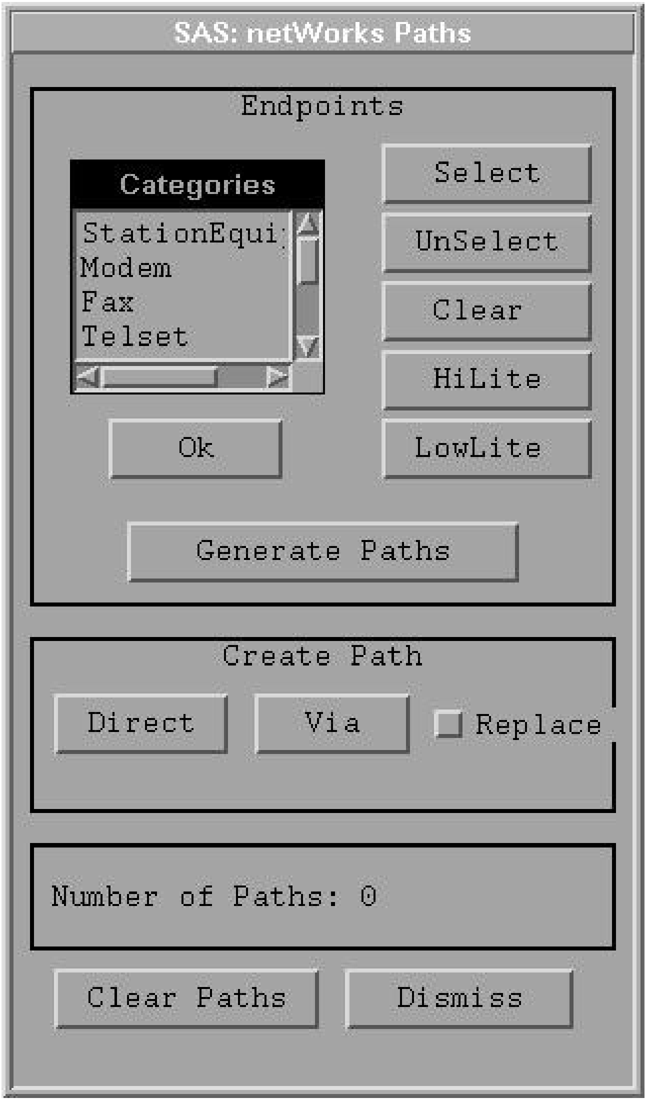
Path Terminology

A path in the telecom environment of netWorks is defined as an ordered series of properly connected equipment models with each model instance appearing only once.

Note that paths are one-directional. Any simple equipment model can be the head or source of the path and likewise for the tail or destination. Each equipment model has an attribute named Capacity associated with it, and any given path's Capacity will be the minimum Capacity value of any equipment model in the path. Equipment models also have the property of being *in-service* or *out-of-service*. A path is considered in-service when all of its equipment models are in-service. The netWorks application borrows the notions of seizing and releasing a path from telecom terminology. Before a path can be seized, it must be available; that is, the path must be below its maximum capacity and in-service. When a path is seized, the value of the capacity attribute for each equipment model in the path is incremented, and when a path is released, this value is decremented.

Path Generation

Before you can run a simulation in netWorks telecom environment, you must identify the paths of interest through your network model. You accomplish this using the Paths dialog box (Figure 3.7), which appears when you select the button in the command panel.



Number of Paths: 0

Figure 3.7. Paths Dialog Box

This dialog box provides two approaches for generating paths through your model network. One way is to select all of the endpoints (simple equipment models) between which you want to simulate traffic and then let the netWorks identify all possible pairwise paths between them. The other approach is to use the Create Path feature and select the source and destination of each path individually, then let netWorks determine the intermediate equipment models.

Different network models require different approaches to path generation, but the most common technique is to use endpoint selection through the `Categories` list box in the Paths dialog box. You select a class (or classes) of equipment models in the list box and then click the Generate Paths button. This creates all possible paths between any instances of the selected equipment model types. A note of caution—it is possible to select any simple equipment model as a source or destination of a path, but the interpretation of the simulation results may be a bit more difficult if you do not restrict the endpoints to similar classes of equipment (for example, all station equipment models).

It is possible for netWorks to identify paths through a model network that could not possibly occur in a real telecom network. For example, in Figure 3.8, there is nothing apparent that prevents the highlighted path from being generated even though it could never happen in a real telecom network.

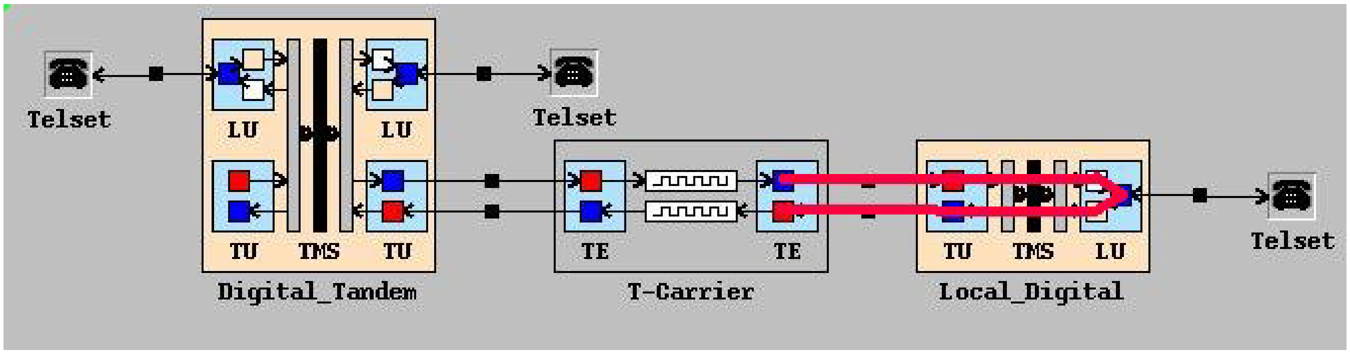


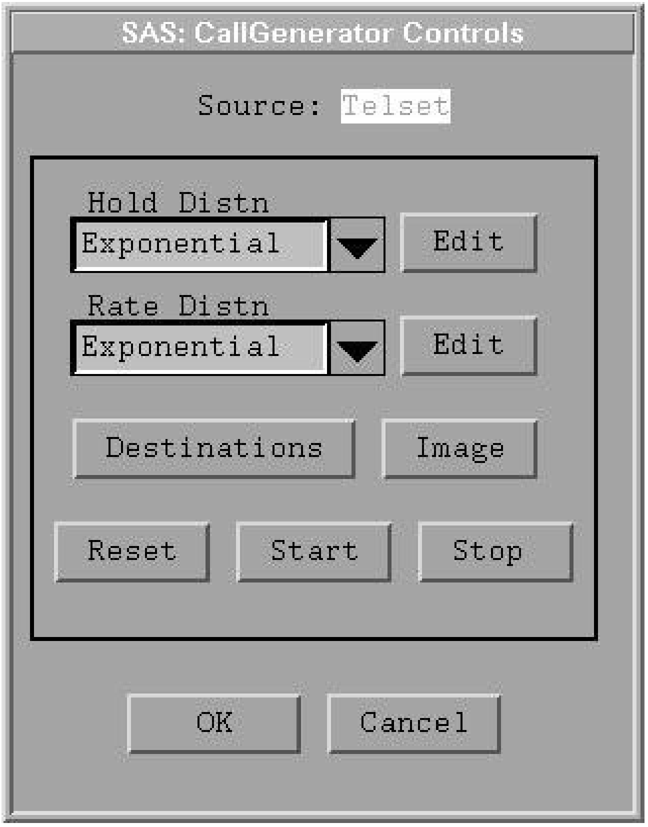
Figure 3.8. Invalid Path Example

The netWorks application uses a concept of *antidromes* to address this issue. The "**??**" section of Appendix **??**, "**??**," provides a detailed description of antidromes and their usage.

Traffic Generators

When a path has been identified by the application, it attaches a traffic generator to the source endpoint (simple equipment model) of the path. (A path would be of little value to the simulation if you could not cause traffic to traverse it.) A traffic generator gives you control over the rate at which simulated calls are generated from this source and also how long the calls last. If you can call multiple destinations from this source (that is, there are multiple paths from this source), you can also set the probability that the call will be placed to each possible destination through the traffic generator.

When an equipment model has a traffic generator attached to it, the Edit Attributes dialog box will have a push button for displaying the generator's control panel. A sample traffic generator control panel is shown in Figure 3.9.



Source: Telset

Hold Distn: Exponential

Rate Distn: Exponential

Figure 3.9. Sample Traffic Generator Controls

The controls labelled Hold Distn determine how long a simulated call will last (that is, the time between the path being seized and subsequently released), and the controls labelled Rate Distn determine the time between calls. Clicking the Destinations button creates a new window in which you can edit the probabilities associated with a call being placed to each destination.

Attribute Selection

Before you can run a network model simulation, you must select at least one Performance attribute on the Attributes palette. The Attributes palette (Figure 3.10) displays icons representing attributes in the performance attribute hierarchy.

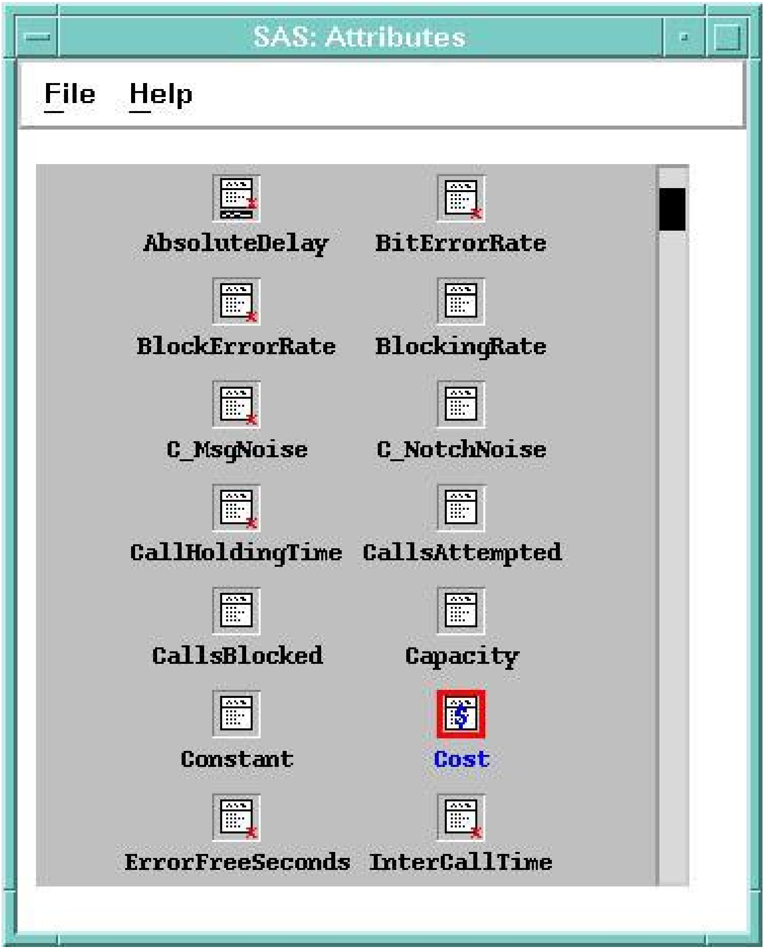


Figure 3.10. Attributes Palette

Different icons are used to identify different types of performance attributes. For example, ▦ is used to denote a Distributional attribute and ▦ indicates a Scalar attribute. Unique icons are not yet available for Ratio or Capacity attributes.

To include an attribute in your simulation, position the cursor over the attribute's icon and click the left mouse button. A red box appears around the icon to indicate its selection. (Note that, even though an attribute has been selected in the Attributes palette, it is of no value to your simulation if none of the equipment models in your paths contain that attribute.) Once an attribute is selected (sometimes referred to as *activated*), you can access the attribute's control panel through its pop-up menu. Its control panel provides options for saving path traversal values of this attribute calculated during a simulation in a SAS data set. There is also an option to change the size of the temporary buffer used to store the most recent values of this attribute so you can display more values in plots of this attribute.

One attribute icon in the Attributes palette always has a "$" drawn on it. This indicates that the attribute is being used as the *cost* measure when calculating paths through the network. The optimum (shortest) path between two points in a model network is the one exhibiting the lowest cost. You can change which attribute is used as a cost attribute through the pop-up menu on the attribute's icon.

Starting a Simulation

After the simulation paths and attributes are identified and the traffic generators are in place, you can start running the simulation; that is, you can begin generating simulated traffic on your model network. There are two ways to do this.

- Select the 🚦 button on the command panel.
- Select the simulation check box in the simulation control panel.

[Note: You can also start each traffic generator individually using a push button on its control dialog box, but this approach is a bit cumbersome when you want to start multiple traffic generators.]

When you start the simulation using either the command button or the control panel, all traffic generators are started simultaneously. When started, a traffic generator first calculates a *holding time* and selects a destination (as defined in the path selection process). It then finds the shortest path to that destination from its list of paths and tries to determine if the path is available for traversal. If the path is available, it seizes the path (decrementing the capacity attribute for each simple equipment model in the path) and *holds* the path for the calculated holding time. The attribute values (for the attributes selected in the Attributes palette) are calculated for the path traversal, and they are placed in their corresponding attribute buffers. After the holding time expires on the simulation clock, the path is released and the traffic generator calculates how long it must wait until it attempts its next call.

If the initial path is not available for traversal, the traffic generator checks the next shortest path for availability and continues this process until either it finds an available path or there are no paths left to try. If it does not find a path to seize, the traffic generator increments the Blocked Calls attribute for its associated model and waits until it is time to attempt another call.

Animation

The animation feature of the netWorks application provides visual feedback during path generation and also during a simulation. You can toggle animation on or off by selecting either the button on the command panel or the Animation check box in the netWorks control panel.

If animation is turned on during path generation, a small black square traverses the newly found path, and the associated arcs are highlighted in red as the square passes

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/OR Software: The netWorks Application, Version 8*, Cary, NC: SAS Institute Inc., 1999. 89 pp.

SAS/OR Software: The netWorks Application, Version 8

ISBN 1-58025-487-X

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

The Institute is a private company devoted to the support and further development of its software and related services.