ArcGIS 9.3

ArcGIS Schematics Designer Tutorial II:
Working With the Custom Query Based Builder

Contents

Welcome to the ArcGIS Schematics Designer Tutorial II.
This tutorial has been designed to assist you in your ArcGIS® Schematics project conception. By specifying diagrams that will work with the Custom Query-Based Builder, it will help you understand more about how to customize your schematic project—the entry point to access your schematic definition environment. This schematic project is a set of definitions and configurations, which references all primary schematic components managed by your Schematics application.

Introducing the sample database content

A summary of the Tutorial II database

When the ArcGIS Schematics tutorials were installed on your computer, the C:\arcgis\ArcTutor\Schematics\Designer folder should have been created. This directory contains the Schematics_Designer_Tutorial database on which this tutorial is based.

This sample database contains an electrical network. The data describing this network is organized into four tables: Station, Feeder, LV_Line, and HV_Line. The LV_Line and HV_Line tables list all the feature edges (links) of the network. The Station and Feeder tables contain all the network feature junctions (nodes).

The complete network is composed of two parts; a high-voltage part and a low-voltage part, that connect to each other.

- High-voltage network: Feeder nodes comprise a high-voltage network. In the high-voltage network, feeder nodes are connected to each other through high-voltage links (HV_Line).

- Low-voltage network: Station nodes comprise a low-voltage network. All station nodes depend on one unique feeder node that could be considered a root node for part of the low-voltage network. Low-voltage links connect a station to another station or a station to a feeder. The link that connects the station to the feeder connects the low-voltage part of the network to the high-voltage part.

Specifications for the tutorial schematic project

The purpose of this tutorial is to graphically display the electrical network data contained in the Schematics_Designer_Tutorial database. The final schematic project will be organized into two main types of schematic diagrams that will both work with the Custom Query-Based Builder:

- The first will contain the complete network (low-voltage and high-voltage parts).

- The second will allow the display of network data related to a single feeder subnetwork (that is, related to a given low-voltage part of the network; each one is managed by one feeder node).

Tutorial database's table of contents

The data model can be presented as follows:
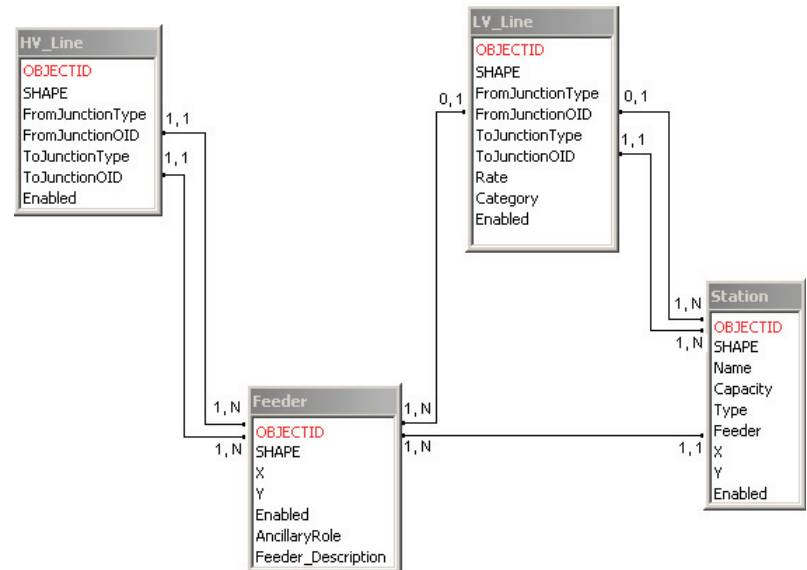
The Station table contains six fields:

- OBJECTID: Station identifier (number)
- SHAPE: ESRI Binary Large Object (BLOB) field
- Name: Station name (string)
- Capacity: Station capacity (number)
- Type: Station type (a variable coded as A, B, C, D, E, or F)
- Feeder: Feeder ObjectID to which the current station belongs (a variable coded as 1, 2, or 3)

The LV_Line table contains eight fields:

- OBJECTID: Low-voltage line identifier (number)
- SHAPE: ESRI BLOB field
- FromJunctionType: Low-voltage line's origin node type (a variable coded as station or feeder depending on the type of the origin node)
- FromJunctionOID: Low-voltage line's origin identifier (number)
- ToJunctionType: Low-voltage line's extremity node type (a variable coded as station or feeder depending on the type of the extremity node)
- ToJunctionOID: Low-voltage line's extremity identifier (number)
- Rate: Ratio value associated with the low-voltage line (number)
- Category: Low-voltage line category (a variable coded as S, M, or B)

The Feeder table contains two fields:

- OBJECTID: Feeder node identifier (number)
- SHAPE: ESRI BLOB field

The HV_Line table contains six fields:

- OBJECTID: High-voltage line identifier (number)
- SHAPE: ESRI BLOB field
- FromJunctionType: High-voltage line's origin node type (a variable coded as Station or Feeder depending on the type of the origin node)
- FromJunctionOID: High-voltage line's origin identifier (number)
- ToJunctionType: High-voltage line's extremity node type (a variable coded as station or feeder depending on the type of the extremity node)
- ToJunctionOID: High-voltage line's extremity identifier (number)

Introducing the tutorial main steps

In this tutorial, the schematic project conception is organized into eight exercises:

- In the first exercise, you will create a schematic dataset containing all the data related to the schematic project you will build, and then you will edit the related schematic project parameters within Schematics Designer.
- During the second exercise, you will become familiar with the schematic project's main parameters and specify default values for these parameters.
- In the third exercise, you will create all schematic components required to display all network data contained in the sample database in a unique schematic diagram managed by the Custom Query-Based Builder.
- During the fourth exercise, you will learn how to animate and document network elements displayed in schematic diagrams according to database field values.
- In the fifth exercise, you will examine the low-voltage part of the network. The purpose will be to create new schematic components to display several schematic diagrams, including one for the low-voltage subnetwork related to a given feeder.
- In the sixth exercise, you will learn how to specify parameters to manage association between features and schematic elements so the Propagate Schematic Selection To Map and Propagate Map Selection To Schematic commands and the Identify tool work in ArcMap™.
- During the seventh exercise, you will learn how to specify parameters to relate schematic elements contained in a schematic diagram to each other.
- The last exercise concerns application behaviors. You will create behaviors that impact schematic diagrams and elements.

Exercise 1: Creating a schematic dataset and editing the schematic project

In this exercise, you will learn how to create a schematic dataset that contains all the schematic data related to the schematic project you will build. Then you will learn how to edit the related schematic project parameters within Schematics Designer.

Creating a schematic dataset

A schematic dataset resides in a personal, file, or ArcSDE® geodatabase; it's the schematic equivalent of a geodatabase feature dataset. The schematic dataset acts as the entry point for your schematic application. A schematic dataset contains a collection of schematic diagram classes and schematic element classes that share the same application domain—for example, water and electrical, and so on.

ArcCatalog™ is the application used to create schematic datasets. To create a schematic dataset:

1. Start ArcCatalog.
2. Browse to the Schematics_Designer_Tutorial personal geodatabase stored in the C:\arcgis\ArcTutor\Schematics\Designer folder.
3. Right-click this personal geodatabase, point to New, then click Schematic Dataset.

   A new schematic dataset appears in the Catalog tree.
4. Type a new name for the schematic dataset ("TutorialII", for example).
5. Press Enter.

Creating a schematic dataset in ArcCatalog automatically creates the associated schematic project where the components that specify the graphic parameters of your schematic application will be stored. These components can be edited and customized using Schematics Designer.

Note: An ArcEditor™ or ArcInfo® license is required for schematic dataset creation.

Editing the schematic project within Designer

A schematic project is a set of configurations that provides access to your application's graphic components, including data sources, diagram types, element types, and behaviors.

To define the schematic diagram types and schematic element types related to a schematic dataset, you need to edit the schematic project associated with that schematic dataset within Schematics Designer:

1. Right-click the new schematic dataset in the Catalog tree and click Edit Project.

   The Schematics Designer application starts, loads the associated schematic project parameters into memory, and opens directly in Design mode. As no diagram type exists in this new schematic dataset, the Diagram Type Assistant automatically launches.
2. Cancel the Diagram Type Assistant for the moment and look at the Designer interface.

When you are working in design mode, all schematic project components already defined are displayed in the Designer Editor window. The Designer tree on the left side of the window shows all the schematic components that have already been defined for the opened schematic project. The right side of the Designer Editor window shows the current state and values for all parameters that have been defined for the currently selected component.

Working in design mode allows you to create and modify any project component's parameters (diagram type, element type, and so forth). When components are created, a new entry corresponding to the newly created component is automatically created in the Designer tree.

To enable the run mode when you are running in design mode, click Close on the bottom right corner of the Designer Editor window. Working in run mode allows you to test and visualize instantly the schematic project parameters you have just modified when running in design mode; any parameter change is automatically taken into account and can be graphically visualized. To activate the design mode when you are in run mode, click the Edit tool.

The default parameters for the graphic display of the schematic elements are defined at the Schematic Project level. These include default node symbol, flag symbol, text size and color, selection style appearance.

When the Schematic Project entry is selected in the Designer tree, the right side of the window lists all the default project parameters. These parameters can be modified according to your needs. They are regrouped into three tabs:

• The General tab displays where all general graphic project parameters are stored.

• The Effects tab includes line, fill, and text effects used for graphic representations.

• The Highlighting tab regroups parameters related to the display style and mode for elements when they are selected in the diagrams.

In this exercise, default values will be set for the following:

• Symbol for flags when no specific symbols are defined for their related flag model

• Symbol for nodes when no specific symbols are defined to represent them

• Symbol scaling factor

• Line effects (color, width) that will be used to represent the links

• Text color and font that will be used to display a label

• Selection style.

Defining the default flag symbol

The Default Flag Symbol parameter available from the General tab will be used by default to represent a flag's symbol when this parameter is defined.

1. Click anywhere in the Default Flag Symbol field.

2. Click the Edit Symbol button . The CGM Symbol Editor and Vector Drawing tool (NgUSymbolEditor) appears.

3. Use the NgUSymbolEditor to build your CGM symbol as desired.

4. Click File and point to Save As. Type the name that will reference your first symbol ("DefaultFlag", in this example).

5. Exit the Symbol Editor (Click File and click Exit).

The Default Flag Symbol field is automatically completed with the name of the symbol you have created.

Note: For more details about the CGM Symbol Editor and Vector Drawing tool, please see the "NgUSymbolEditor tool" topic in the Schematics Designer book of the ArcGIS Desktop Help System.
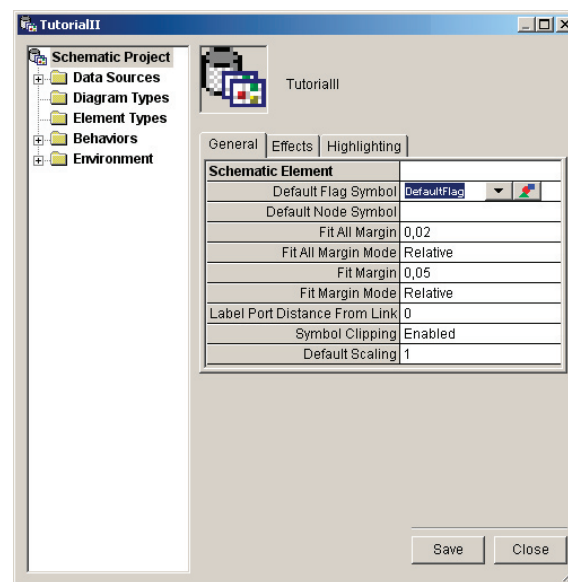
Defining the default node symbol

The Default Node Symbol parameter on the General tab is used by default to represent schematic nodes when their own symbol parameter (defined for their related node element type) is not set.

To define this default symbol, follow the steps below:

1. Click the Default Node Symbol field.

2. Click the Symbol Editor button to launch the CGM Symbol Editor and Vector Drawing tool (NgUSymbolEditor).

3. Use the NgUSymbolEditor drawing tools to build your CGM symbol as desired.

4. Click File and click Save As, then type a name for the current symbol ("DefaultNode", for example) .

5. Click File and click Exit to exit the NgUSymbolEditor.

The Default Node Symbol field is automatically populated with the name of the symbol you have just created.
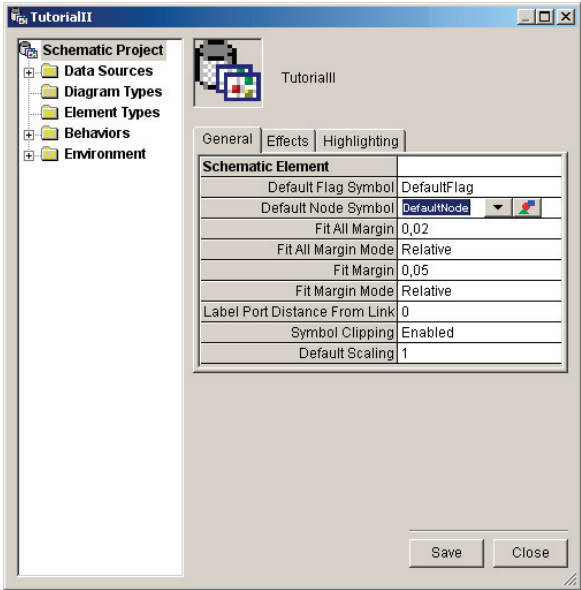
Defining the default scaling

The Default Scaling parameter on the General tab will be applied by default to the schematic diagram symbols or labels when their own scale parameters are not set.

1. Click the Default Scaling field.

2. Type a value.

Note: The correct value for your schematic project default scaling factor is not easy to determine, and a value that is correct for one schematic project could be inappropriate for another schematic project. In fact, this value depends on your network coordinate system and how close to, or far from, each other the nodes are in your schematic view. Generally, you will have to test several values before finding the best one.

5
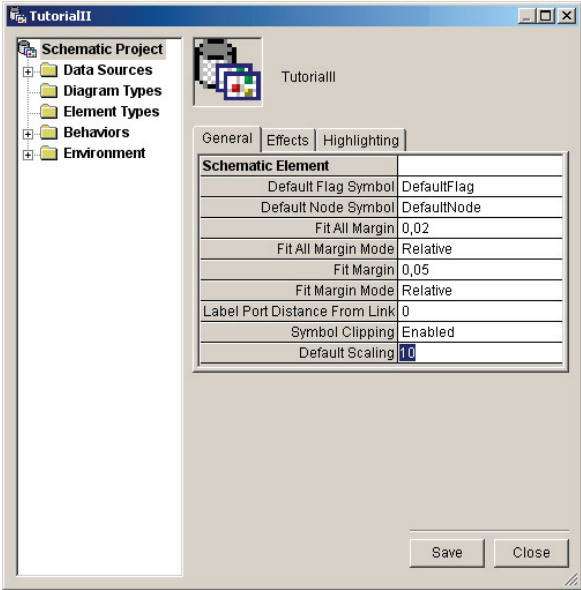
Defining the default line effects

Parameters defining line effects that will be used by default to represent a link line (when these effects are not defined at the link element type level) are available on the Schematic Project Effects tab. They are assembled into the Line Effects subset of parameters.

1. Click the Effects tab.
2. Click the Line Color field and click the resulting color box.The Table Color Editor appears.
3. Choose or define a color.
4. Click OK.
5. Click the Line Width field to specify the default width for the lines. Modify the value with the one you want.
6. Click the Line Style drop-down arrow and click the desired default style for lines.
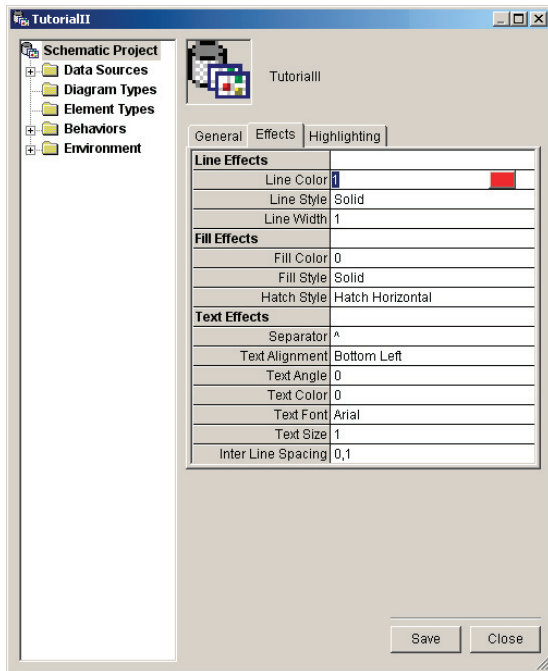
Defining the default text effects

Parameters defining text effects that will be used by default to display the labels when these effects are not defined are available on the Schematic Project Effects tab. They are assembled into the Text Effects subset of parameters.

1. Click the Effects tab.
2. Click the Text Color field and click the resulting color box. The Table Color Editor appears.
3. Choose or define a color and click OK.
4. Click the Text Font to define the default text font for the labels.
5. Click one of the default fonts already defined from the drop-down list or click the ellipsis button on the right to open the Font Editor. The Font Editor lists all fonts installed on your computer and allows you to choose the font you want.
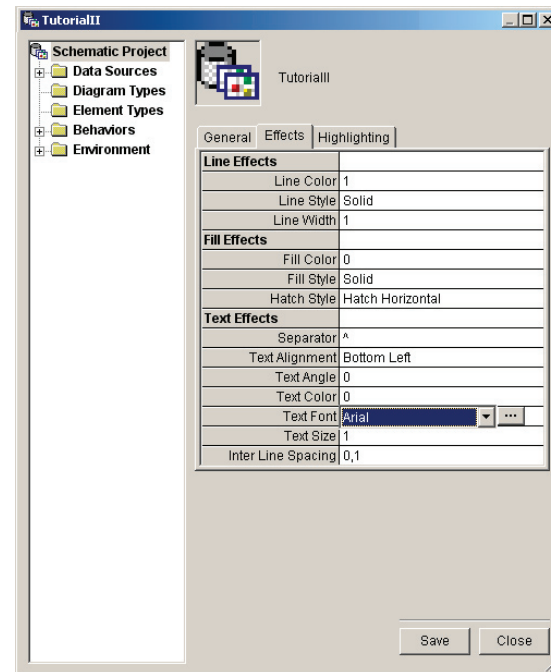
Defining the selection style

Parameters defining the selection style are assembled on the Highlighting tab.

Three selection styles are available:

- With the Handle style, a selected schematic element is singled out by the display of its bounding box.

- With the Highlight style, a selected element is singled out by specific color, width, style, and fill.

- With the Handle And Highlight style, the Handle and Highlight selection styles are combined.

When the Highlight or Handle And Highlight style is used, some line, fill, and text effects must be set to complete the selection style definition and the appearance of links, nodes, and labels when they are selected in a schematic view.

1. Choose Handle And Highlight from the Selection Style drop-down list.

2. Click the Line Color parameter and click the color box. The Table Color Editor appears.

3. Choose or define the desired color that will be used to highlight lines when they are selected.

4. Click the Line Width parameter and type the desired value.

5. Click the Fill Style drop-down arrow and click Solid, then define the Fill Color parameter. These fill attributes will be used to display the symbol fill of the nodes when they are selected.

6. Click the Text Color field and click the color box that appears. Choose or define the color you want in the Table Color Editor dialog box.

7. Click OK.
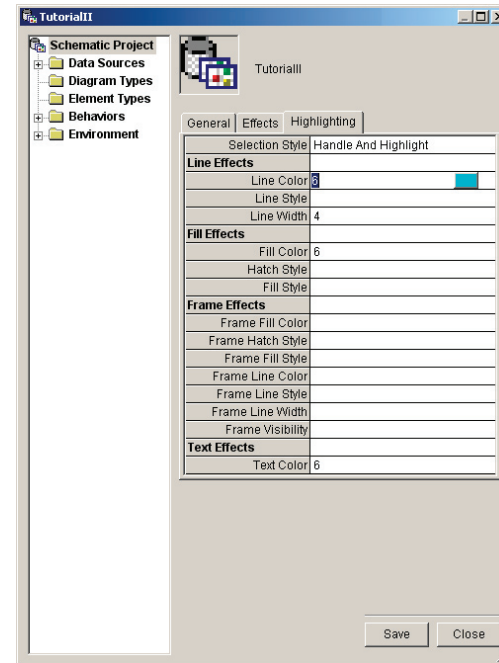
8. Click Save.

   By default, the schematic project definition is automatically checked. This causes the Please verify dialog box to open before saving. This dialog box sums up some points that may have been forgotten during your schematic project building. Note that the Please verify dialog box lists warnings with a critical level mark (red, orange, or green traffic light) without preventing saving; that is, each warning is only informative, and your project definition can be completely operational even if red warnings are shown on the dialog box.

9. Check Do not perform this verification when saving any more and click Yes. The verification phase is deactivated for the next save operation.

10. Click OK.

Note: To reactivate the verification phase, right-click the Schematic Project entry in the Designer tree and check Verify On Save.

7

Exercise 3: Displaying all network data in a unique schematic diagram

In this exercise, you will create the schematic components required to display all network data stored in the Schematics_Designer_Tutorial database in a unique diagram. The diagram type that will implement this schematic diagram will work with the Custom Query Based Builder. With this predefined builder, you need to specify the queries that return the elements you want to be displayed in your diagrams.

Three main steps are required to create such diagrams:

1. Create the schematic diagram type that will implement the desired schematic diagram.

2. Create the node element types that will implement the node elements contained in the schematic diagram.

3. Create the link element types that will implement the link elements contained in the schematic diagram.

Creating the first schematic diagram type

A network is made up of nodes and links. These nodes and links are displayed in different generic structures known as diagrams. A diagram is a collection of schematic views and is related to a single schematic project. A diagram type is a template that brings together a set of diagrams that have common characteristics. More than one diagram type can be created for a schematic project. Typically, one diagram type can be defined for a specific view of the network. For example, a diagram type is defined for a schematic of the electric network of primaries and switches, and a second diagram type is defined for a schematic of the internal schema (inside plant) of the devices. You can either create a diagram type that inherits another diagram type or create a diagram type that inherits no parent diagram type.

In the following steps, you will define a schematic diagram type to implement a unique schematic diagram; that is, the schematic diagram where all network data stored in the Schematics_Designer_Tutorial database will be displayed.

1. Right-click Diagram Types in the Designer tree and click Create. The Create Diagram Type dialog box opens.

2. Type the name that will be used to reference the new diagram type ("All", for example).

Since no diagram has already been created, this new diagram type cannot

inherit from another one (no parent diagram type can be set).
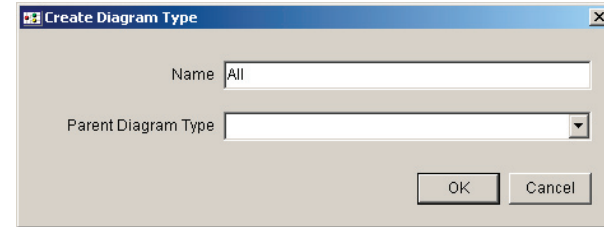
3. Click OK.

The new diagram type subentry is automatically created below the Diagram Types main entry in the Designer tree.
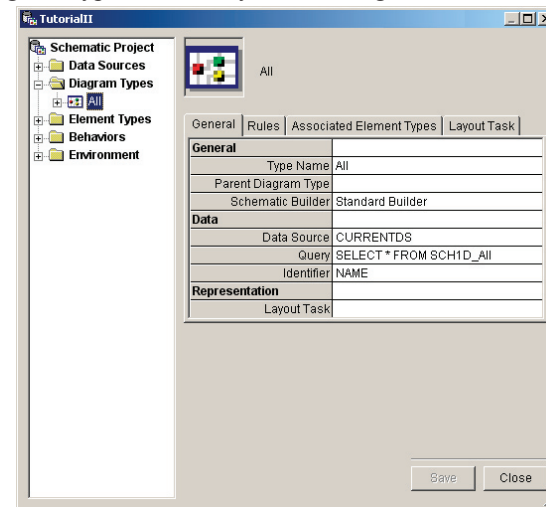
When a new diagram type is created, Schematics specifies the Standard Builder Builder by default for the Schematic Builder field; it automatically creates the two DIAGRAMCLASSID and DIAGRAMOBJECTID attributes and sets some parameters for the newly created diagram type.

In this exercise, you have to create a schematic diagram type that will work with the Custom Query Based Builder. Moreover, because this diagram type will implement a unique schematic diagram, no query, identifier, or data source is required.

When a new diagram type is created, Schematics specifies the Standard 1.

1. Change Standard Builder to Custom Query-Based Builder from the Schematic Builder drop-down list.

2. Click the Query field and click the Ellipsis button that appears. The Query Editor dialog box opens.

3. Select the SELECT * FROM SCH1D_All query specified in the Query area and remove it.

4. Click OK to close the Query Editor dialog box.
   The Query field on your diagram type's General tab is now empty.

5. Click the Identifier field and click the ellipsis button that appears to open the Identifier Editor dialog box.

. Click NAME from the Identifier area and click the up arrow to remove the NAME field from this area.

7. Click OK to close the Identifier Editor dialog box.
   The Identifier field on your diagram type's General tab is now empty.

8. Specify no data source for the Data Source field.

9. Expand the Attributes entry under the Diagram Types entry in the Designer tree, right-click DIAGRAMCLASSID, then click Delete. This predefined attribute is removed.

10. Repeat step 9 to remove the second DIAGRAMOBJECTID predefined attribute.

Your Diagram Types entry should appears as follows:
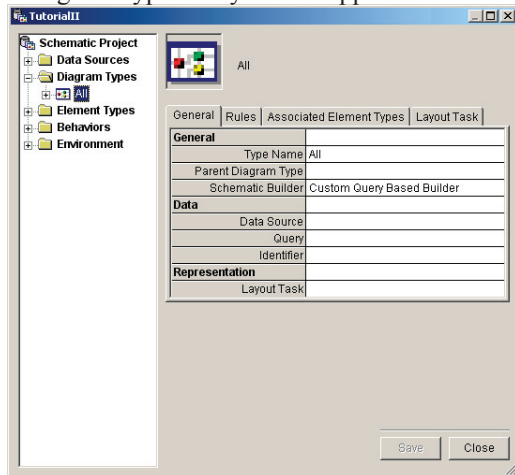
Creating the node element types

An ArcGIS junction is implemented as a Schematics node element. The node represents a device or other point feature of a network. It is displayed as a symbol and is assigned graphic (color, fill style, and so forth), geometric (scaling factor, rotation, and so forth), and visual (visibility, highlighted state, and so forth) parameters. Any node is implemented by a schematic node element type that holds the common property of a set of nodes.

In the following steps, you will create two schematic node element types: one to implement the display of all records stored in the Station table and another to implement the Feeder table's records.

Creating the StationNodes element type

1. Right-click Element Types in the Designer tree and click Create. The Create Element Type dialog box opens.

2. Type the name that will be used to reference your first schematic node element type ("StationNodes" in this example).

3. Click Node from the Type drop-down list.

Since no node element type has already been created for this schematic project, this new node element type cannot inherit from another one (no parent element type can be set).

4. Keep the Element Group option checked by default to ensure you can customize your element type display afterward.
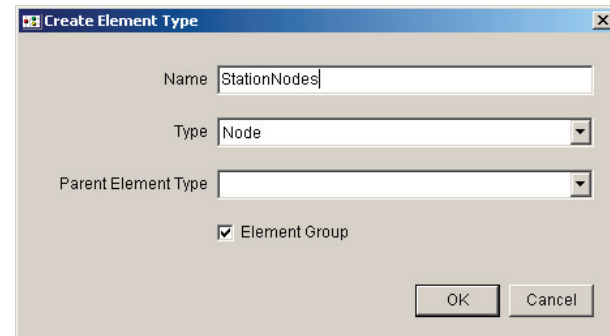
5. Click OK.

The new node element type subentry is automatically created below the Element Types main entry in the Designer tree.
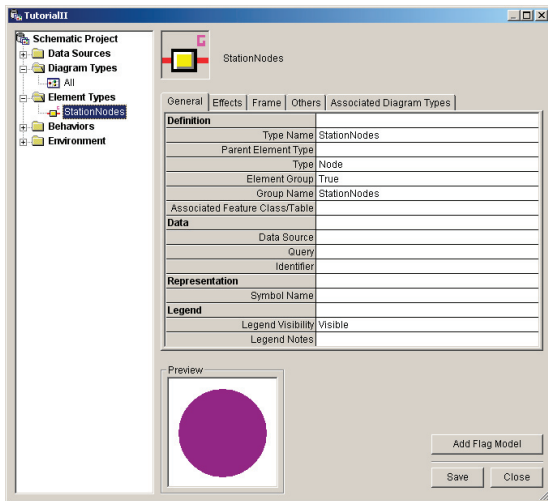
The Preview window shows the symbol that will be used by default to represent all nodes of this type. In this case, it is the schematic project's default node symbol you defined in exercise 2.

Set the parameters for this new node element type and, in particular, set parameters to query and identify each node this StationNodes element type will implement. These parameters are available on the General tab, and they are assembled into the Data subset of parameters.

1. Click the General tab and click the CURRENTDS data source from the Data Source drop-down list.

2. Click the Query field and click the ellipsis button that appears.
   The Query Editor dialog box opens.

3. Click the Station table from the Tables list where all records corresponding to the network stations are stored.

4. Click All Records.
   The SELECT * FROM Station SQL query automatically appears in the Query area.

5. Click OK.

The Query Editor dialog box closes. The specified query automatically appears on the General tab's Query field.

6. Click the Identifier field on the General tab and click the ellipsis button that appears.
   The Identifier Editor dialog box opens.

The Fields list on the Identifier Editor dialog box displays all fields returned by the query you have just specified. These fields can be used to identify the schematic elements of the element type.

The SCH_DiagramType and SCH_DiagramName items are also available at the end of this list. These predefined items are related to the diagram type name and the diagram name where schematic elements will be displayed. They can help you ensure that each element identifier contained in your schematic diagram is unique.

In this example, you will identify each StationNodes node by concatenating the diagram type name, the diagram name, the Station string, and the OBJECTID field values.

Note: Using the Station string in the node identifier building is useful in this example because the FromJunctionType and ToJunctionType fields available from the tables of links specify whether each link connects a station-type node or a feeder-type node. By using the Station string in your StationNodes nodes' identifiers, it will be easy to specify the origin and extremity node for your links afterward.

1. Double-click the SCH_DiagramType and SCH_DiagramName fields available from the Fields list so these two items appear in the Identifier list.

2. Type "Station" in the central text box and validate this string by clicking the down arrow.

3. Double-click the OBJECTID field from the Fields list to select it and click OK.

The specified items automatically appear in the General tab's Identifier field.

When specifying Identifier for an element type, remember these two rules:

• All elements related to an element type must be identifiable as a unique element.
• Two elements of the same type (node, link, and so forth) cannot have the same identifier when they are found in the same diagram.

Note: For more information about the Query Editor and Identifier Editor, open the Schematics Designer book in the ArcGIS Desktop Help

System, load the Setting the node element type General parameters topic, and click the Query Editor and Identifier Editor hyperlinks in the Data section.

Your first node element type entry should appear as follows:
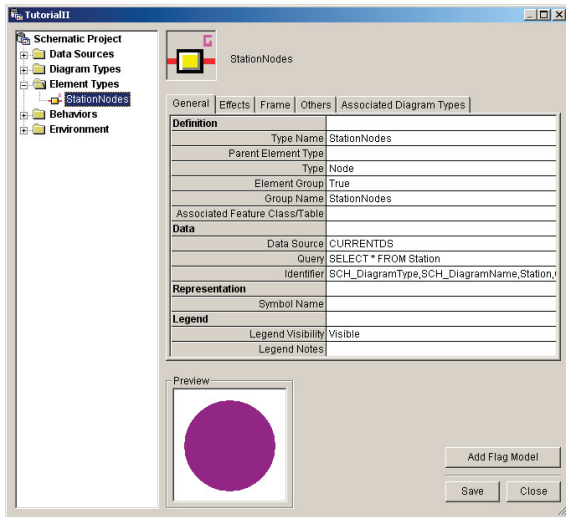
Previously, you created the All schematic diagram type that will implement one schematic diagram where all network data stored in the Schematics_Designer_Tutorial database will be displayed. Now specify that nodes implemented by the StationNodes node element type you have just defined must be displayed in this schematic diagram. To specify the schematic elements you want contained in a schematic diagram, you must associate the desired element types with this diagram type:

1. Right-click the All diagram type entry in the Designer tree and click Edit Associations. The Element Type Associations dialog box opens.
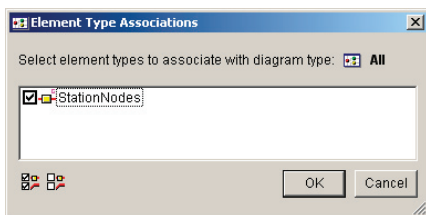
2. Check the StationNodes check box.

3. Click OK.

The Associated Element Types tab appears for your diagram type. This tab lists all the schematic element types that are currently associated with a diagram type. At the same time, the Associated Diagram Types tab has been created for your StationNodes element type; this tab lists all the diagram types with which that element type is associated.
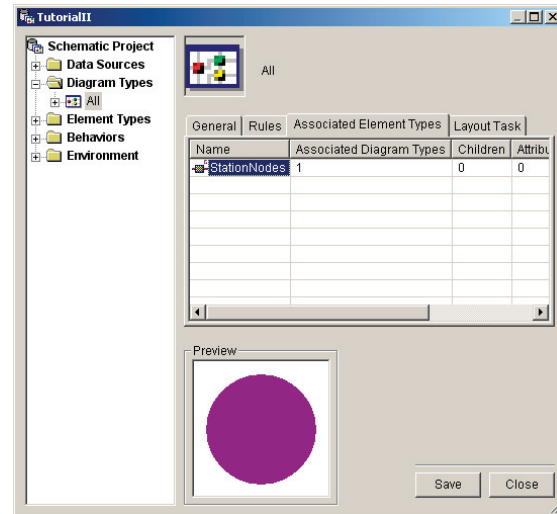
Test that your StationNodes node is displayed correctly in your schematic diagram.

1. Click Save at the bottom right corner of the Designer Editor window.

2. Click Close. You are now in run mode.

3. Click File and click Diagram Toolbar.
   The Diagram toolbar appears.

4. Click the Open Diagrams button 🗗.
   The Select Diagram To Open dialog box appears. The only diagram type, All, is available from the Diagram Type drop-down list, and the All diagram is available from the Diagram Name drop-down list.

5. Click OK.

Your first schematic diagram opens. Since no geometric attributes are defined for any node (that is, x,y coordinates are not specified for each node), Designer automatically places the nodes on a default grid.
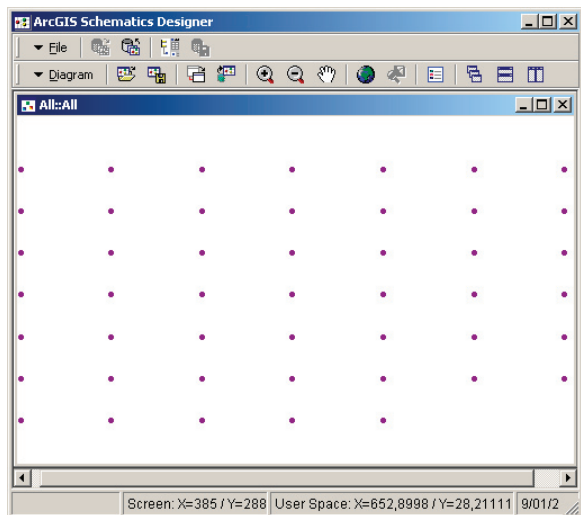
Creating the FeederNodes element type

Next, you will create the second node element type to implement the nodes related to the records stored in the Feeder table.
Before creating this new schematic component, you must switch to design mode:

1. Close the open diagram and click the Edit button [icon] to reedit your schematic project parameters.

Then, create your new node element type as follows:

2. Right-click Element Types in the Designer tree and click Create. The Create Element Type dialog box opens.

3. Type a name for the new node element type ("FeederNodes", for example).

4. Click Node from the Type drop-down list.

5. Ensure that the Element Group check box is checked and click OK.

Under the Element Types entry, the new FeederNodes node element type entry is automatically created.

Complete your FeederNodes node element type definition by specifying the data source and the query that will return all feeder nodes stored in the Schematics_Designer_Tutorial database and by identifying all nodes the FeederNodes node element type will implement.

6. Click FeederNodes in the Designer tree.

7. Click CURRENTDS from the Data Source drop-down list.

8. Click the Query parameter field and click the Ellipsis button. The Query Editor dialog box opens.

9. Double-click the Feeder table from the Tables list; the SELECT * FROM Feeder SQL query automatically appears in the Query field.

10. Click OK.
    The Query Editor dialog box closes.

11. Click the Identifier field and click the ellipsis button. The Identifer Editor dialog box opens.

As previously done for the StationNodes element type identifier, each FeederNodes node will be identified by concatenating the diagram type name and the diagram name where it will be displayed, the Feeder string, and the OBJECTID field values.

12. Double-click the SCH_DiagramType and SCH_DiagramName items available at the end of the Fields list. These two fields appear in the Identifier list.

13. Type "Feeder" in the central text box and validate by clicking the down arrow.

14. Double-click the OBJECTID field from the Fields list to add this field to the Identifier list.

15. Click OK.
    The Identifier Editor dialog box closes.

At the end of these steps, the General tab related to your new FeederNodes element type appears as follows:
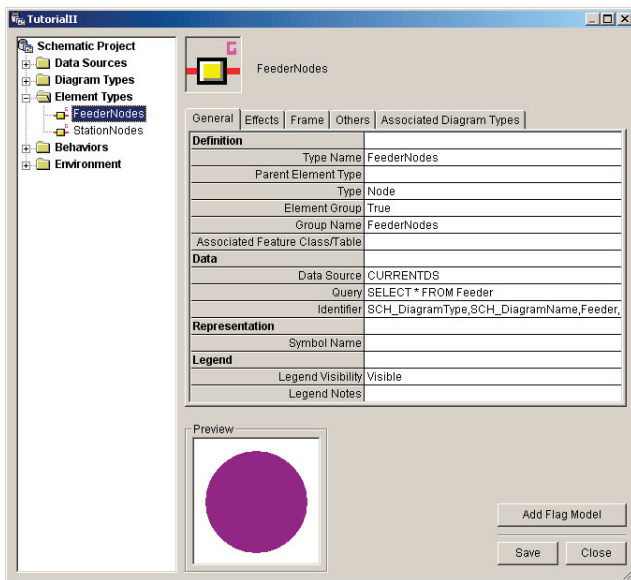
For the StationNodes element type, for instance, no specific CGM symbol has been defined to represent station nodes, so the default symbol defined at the Schematic Project level is used by default. For the FeederNodes element type, a specific CGM symbol will be created. This symbol is defined when the Symbol Name parameter on the General tab is set.

1. Click the Symbol Name field and click the Edit Symbol button . The CGM Symbol Editor and Vector Drawing tool opens.

2. Draw your symbol.

3. Click File and click Save.

4. Type a name for the newly created symbol.
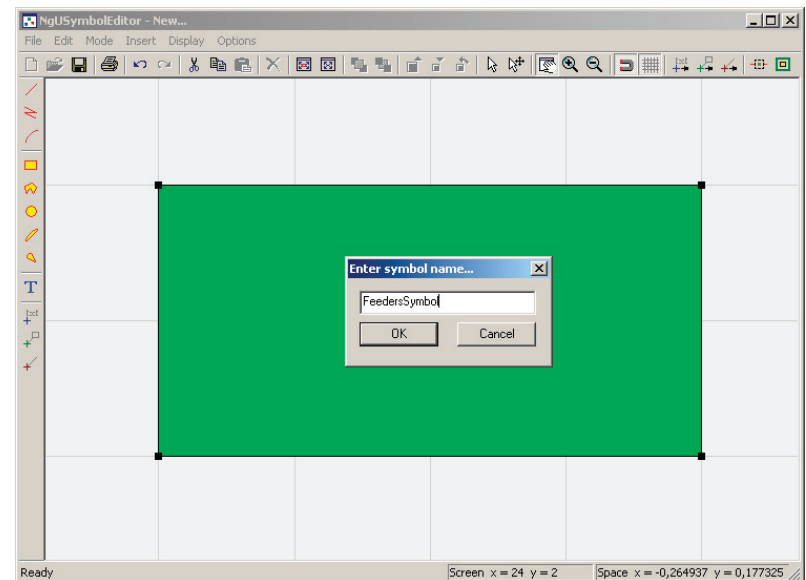
5. Click File and click Exit.
   CGM Symbol Editor and Vector Drawing tool closes.

The Symbol Name field on the FeederNodes General tab is automatically populated with the name of the symbol you have created. The Preview subwindow on the left corner of the Designer Editor window shows this new symbol. The General tab related to your FeederNodes type appears as follows:

13

Close the diagram and click the Edit button to switch to design mode.
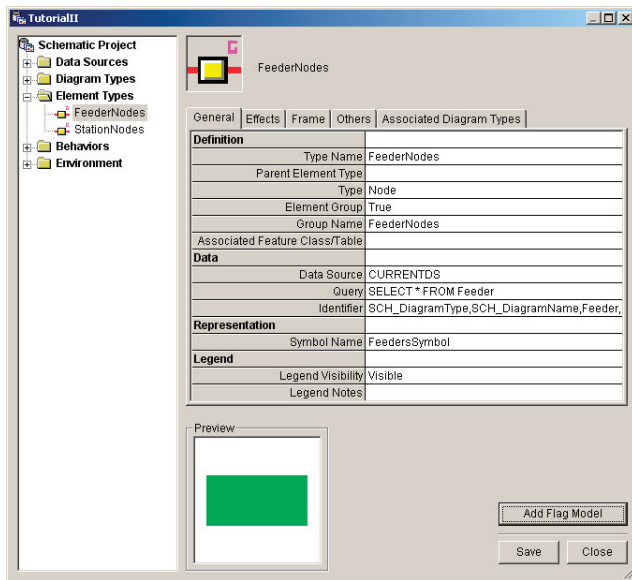
You must now associate the newly created FeederNodes element type with the All diagram type so that all nodes (station and feeder) stored in the Schematics_Designer_Tutorial database appear in the diagram that the All diagram type implements:

1. Right-click the All diagram type entry and click Edit Associations.

2. Check the FeederNodes item in the list and click OK.

The FeederNodes item automatically appears on the Associated Element Types tab related to the All diagram type.

3. Click Save.

To test that the nodes implemented by the new FeederNodes element type are correctly displayed in your schematic diagram, close the Designer Editor window and click the Open Diagrams button. Click the All diagram from the Select Diagram To Open dialog box and click OK. The All schematic diagram appears as follows:

14
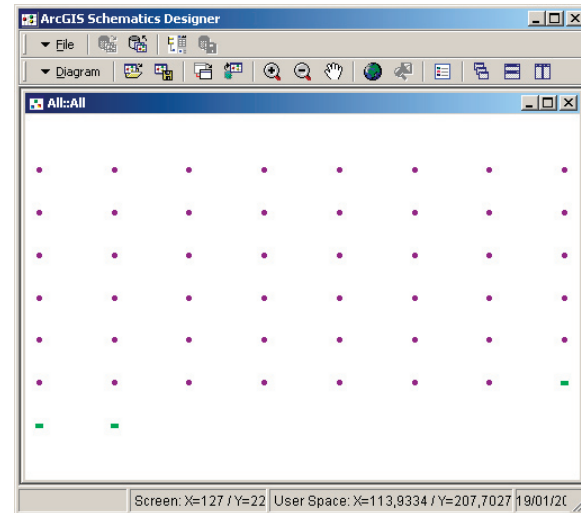
Creating the link element types

An ArcGIS edge is implemented as a Schematics link element. The link represents an edge of a network and is represented either by a direct single line segment or by several line segments passing through one or more link points or vertices. Line patterns and line styles (color, style, width) can be applied to links to refine their graphic display. Any link is implemented by a schematic link element type that holds the common property of a set of links.

In the following steps, you will create two schematic link element types: one to implement the display of all records stored in the LV_Line table and another to implement the HV_Line table's records.

Creating the LV_LineLinks element type

1. Right-click Element Types in the Designer tree and click Create. The Create Element Type dialog box opens.
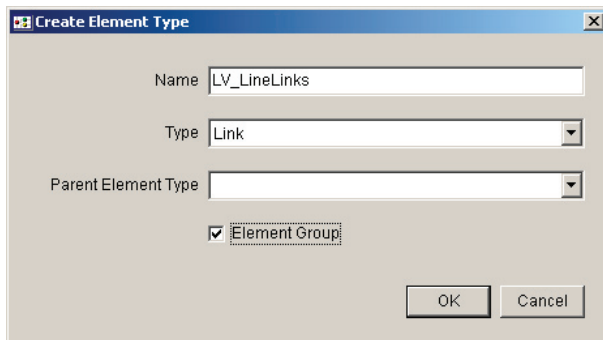
2. Type a name for the link element type ("LV_LineLinks", for example).

3. Click Link from the Type drop-down list.

4. Ensure the Element Group check box is checked and click OK.

Under the Element Types entry, the new LV_LineLinks element type entry has been automatically created. The Preview window shows how the links of this type will appear. Since no specific values are specified for the line display, the schematic project's default values are used.
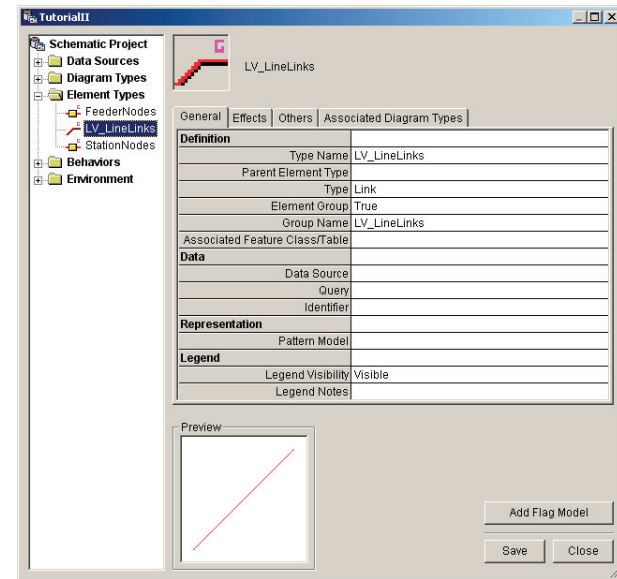
Set the parameters for this new link element type and, in particular, set parameters to query and identify each link this LV_LineLinks element type will implement. These parameters are available on the General tab, and they are assembled into the Data subset of parameters.

1. Click LV_LineLinks under Element Types in the Designer tree.

2. Click CURRENTDS from the Data Source drop-down list.

3. Click the Query field and click the ellipsis button. The Query Editor dialog box opens.

4. Click the LV_Line table From the Tables list. This is where all records corresponding to the network LV_LineLinks are stored.

5. Click All Records.
   The SELECT * FROM LV_Line SQL query automatically appears in the Query area.

6. Click OK.

The specified query automatically appears in the General tab's field.

7. Click the Identifier field and click the ellipsis button.
The Identifier Editor dialog box opens.

To be homogeneous with the node element identifying process, each LV_LineLinks element identifier will be built according to the same format: the diagram type name, the diagram name, the LV_Line string value, and the OBJECTID field values will be concatenated to build the identifiers.

8. Double-click the SCH_DiagramType and SCH_DiagramName items available at the end of the Fields list; these two items appear in the Identifier list.

9. Type "LV_Line" in the central text box and validate this string by clicking the down arrow.

10. Double-click the OBJECTID field from the Fields list to select it and click OK.

The Identifier Editor dialog box closes and the specified items automatically appears in the General tab's Identifier field.

The General tab related to the LV_LineLinks element type appears as follows:
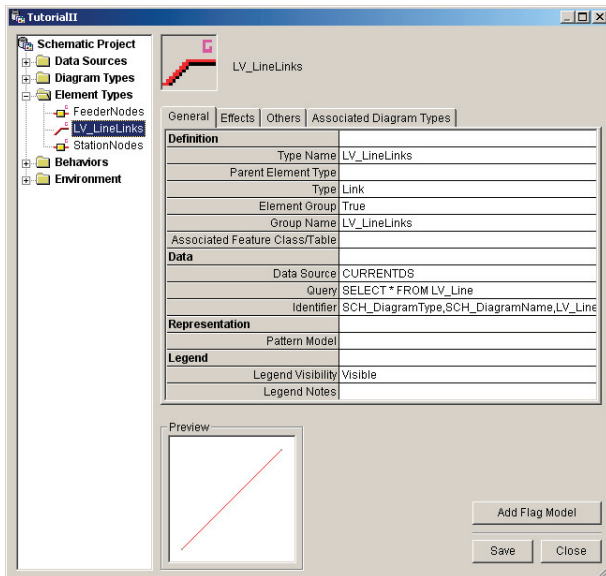
Suppose that you want to test the display of your network. Even if you associate your new link element type with your schematic diagram type (essential to displaying your new links), your network will appear as follows:
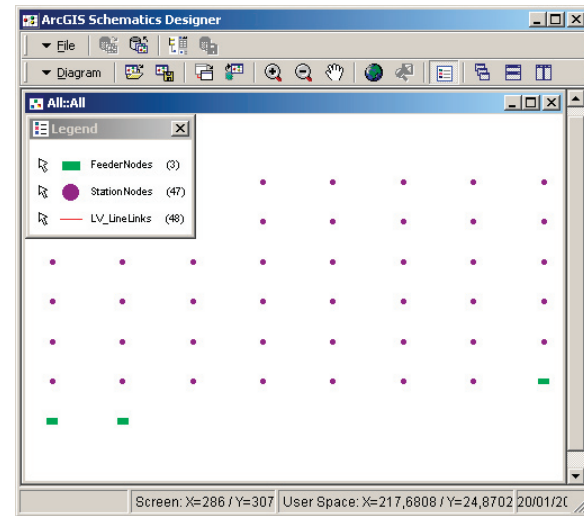
Note that even though the links appear in the Legend subwindow, they are not displayed. The schematic links' display is conditioned by their topological properties; that is, origin and extremity schematic nodes must be specified to display a schematic link. To specify this information, you will have to create two specific attributes for your link element type. These attributes must be created with the OriginNode and ExtremityNode predefined names. When such attributes are created for a link element type, their values should return the origin node identifier and the extremity node identifier that connect each link of this element type. The following steps define these two compulsory attributes for the LV_LineLinks element type:

1. Right-click LV_LineLinks in the Designer tree and click Create Attribute.
The Create Element Type Attribute dialog box opens.

2. Click OriginNode from the Name drop-down list.

16

3. Specify the type of attribute. In this case, because origin node identifiers can be retrieved from the fields stored in the LV_Line table and since these fields are returned by the query defined for the LV_LineLinks element type, you have to create a field attribute.

4. Click Field Attribute from the Type drop-down list.
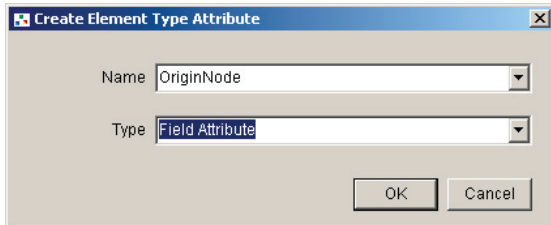
5. Click OK.

The Identifier Editor dialog box automatically opens. It allows you to complete the attribute definition by specifying the field or the set of fields returned by the LV_LineLinks element type's query that will be merged to compute each origin node identifier. The OriginNode attribute format should follow the node identifier format. In this example, each node identifier merges its diagram type name, its diagram name, a string value corresponding to the type of node (Station or Feeder), and its OBJECTID field value. The OriginNode attribute values must be merged with this information:

6. Double-click the SCH_DiagramType and SCH_DiagramName items at the end of the Fields list so these two items appear in the Identifier list.

7. Double-click FromJunctionType from the Fields list. This specific field takes part in the identifying process (this field, stored in the LV_Line table, identifies the link's origin node type, that is, station or feeder node).

8. Repeat this operation with the FromJunctionOID field (this field, stored in the LV_Line table, identifies the link's origin node ObjectID).

9. Click OK.

The OriginNode attribute's General tab appears as follows:
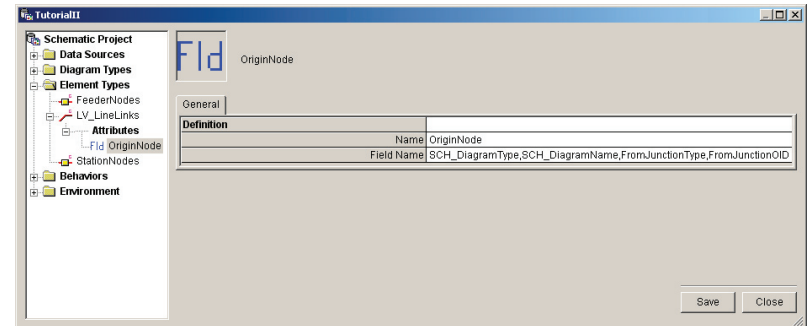
Next, you will create the ExtremityNode attribute.

1. Right-click LV_LineLinks and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Click ExtremityNode from the Name drop-down list.

3. Ensure the Field Attribute value is selected in the Type drop-down list and click OK. The Identifier Editor dialog box opens.

4. Double-click the SCH_DiagramType and SCH_DiagramName items available at the end of the Fields list so these two items appear in the Identifier list.

5. Double-click ToJunctionType from the Fields list; this specific field takes part in the identifying process (this field, stored in the LV_Line table, identifies the link's extremity node type, that is, station or feeder node).

6. Repeat this operation with the ToJunctionOID field (this field, stored in the LV_Line table, identifies the link's extremity node OBJECTID).

7. Click OK.

The ExtremityNode attribute definition is now complete.

The LV_LineLinks element type definition is now complete. You only need to associate this new link element type with the All diagram type so the links implemented by the LV_LineLinks element type are displayed in the diagram implemented by the All diagram type.

As previously done when you associated your element node types with your diagram type:

1. Right-click the All diagram type entry in the Designer tree and click Edit Associations.

2. Click LV_LineLinks from the Name drop-down list and click OK.

The LV_LineLinks item is automatically added to the All diagram type's Associated Element Types tab.

If you switch to run mode and open the All diagram, it appears as follows:
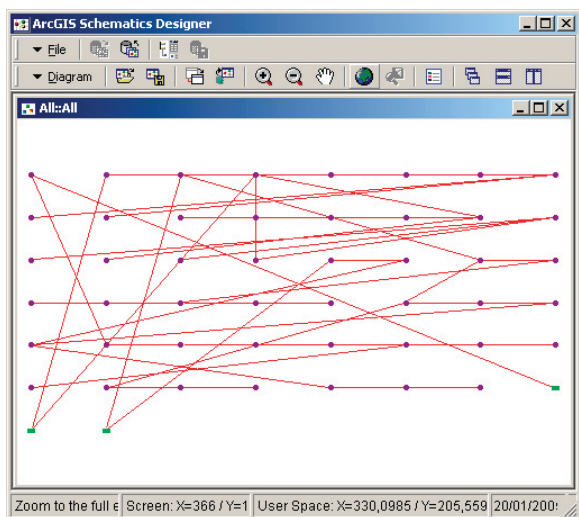
Creating the HV_LineLinks element type

Create the second link element type that will implement the links related to the records stored in the HV_Line table.
Because this new link element type definition will be close to the LV_LineLinks element type you have just created, there is a quick way to proceed, by copying the LV_LineLinks element type definition:

1. If you are in run mode, close the diagram and click the Edit button to switch to design mode.

2. Right-click LV_LineLinks in the Designer tree and click Copy. The Copy Element Type dialog box opens.

3. Keep the Parent Element Type text box empty.

4. Type a name for your new link element type ("HV_LineLinks", in this example).
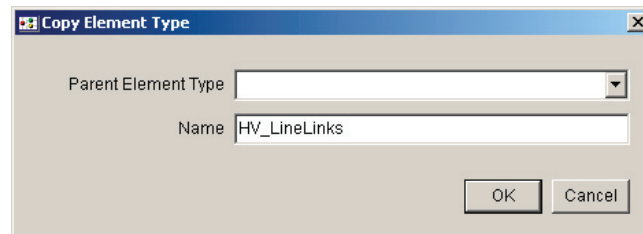
5. Click OK.

The Copy Element Type dialog box closes. The new HV_LineLinks element type entry appears under Element Types in the Designer tree.

All general parameters (Query, Identifier) and attributes defined for the LV_LineLinks element type from which the new HV_LineLinks element type has been copied are automatically reported. The HV_LineLinks element type entry appears as follows:
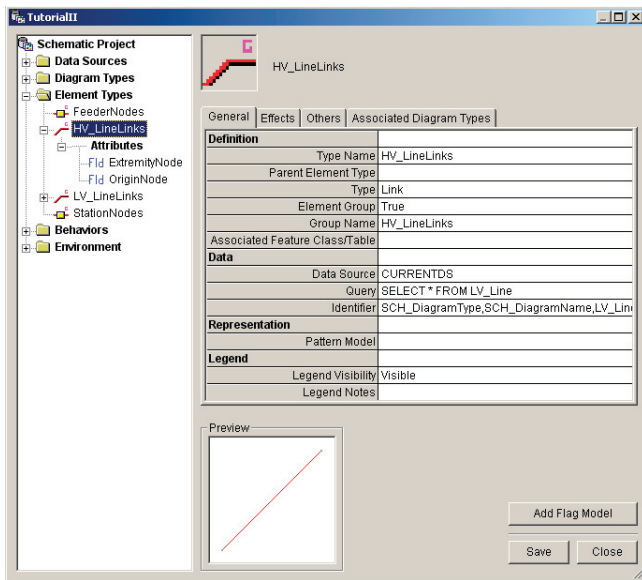
You must now modify the definition of the HV_LineLinks element type so it is related to the HV_Line table instead of the LV_Line table.

6. Click the HV_LineLinks element type General tab.
    • Verify that the copied data source is OK.
    • Then, notice the Query field. The copied query returns all the links stored in the LV_Line table. You must modify this query so it returns all the links stored in the HV_Line table.

7. Click the Query field and click the ellipsis button. The Query Editor dialog box opens.

8. Remove the query from the Query field.

9. Double-click the HV_Line table from the Tables list so the SELECT * FROM HV_Line SQL query automatically appears in the Query area.

10. Click OK.
    The Query Editor dialog box closes. The Query value on the HV_LineLinks General tab is updated with the newly specified query.

As you have done for all previously defined element types, each HV_LineLinks link will be identified by concatenating the diagram type name, the diagram name where it will be displayed, the HV_Line string, and the OBJECTID field values. The copied identifier must be modified:

1. Click the Identifier field and click the ellipsis button. The Identifier Editor dialog box opens.

2. Choose the OBJECTID item from the Identifier list and click the up arrow to remove the item from the list.

3. Proceed in the same way to remove the LV_Line item from the Identifier list.
    Then, keep both SCH_DiagramType and SCH_DiagramName items in the Identifier list and complete the definition as follows:

4. Type "HV_Line" in the central text box and validate by clicking the down arrow.

5. Double-click the OBJECTID field from the Fields list to add this field to the Identifier list.

6. Click OK.

The Identifier Editor dialog box closes and the specified items are automatically displayed in the General tab's Identifier field. The General tab related to the HV_LineLinks element type appears as follows:
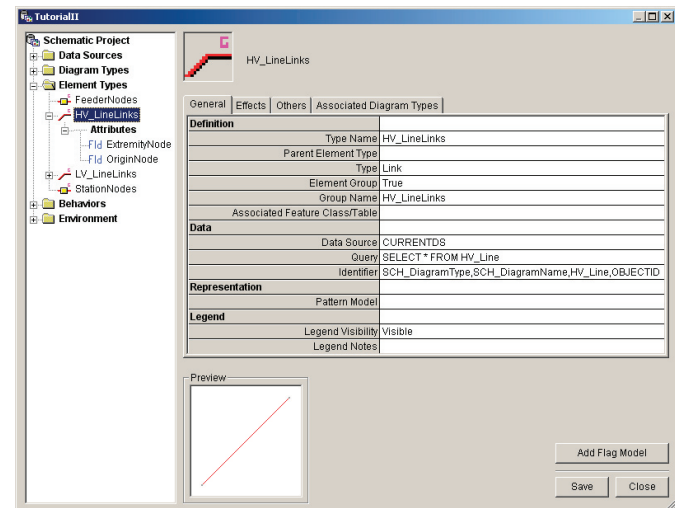
Now, notice the OriginNode and ExtremityNode attributes that have been copied for the HV_LineLinks element type during the copy operation. These OriginNode and ExtremityNode attribute formats must follow the node identifier format that, in this example, merges the diagram type name, the diagram name where these nodes display, the Station or Feeder string (depending on the node type), and the node OBJECTID. All this information is available from the HV_Line table, and all the HV_Line table's fields are returned by the query defined for the HV_LineLinks element type. You only need to check that the fields used to generate these attributes are correctly specified:

1. From the OriginNode attribute's General tab, the Field Name field indicates how the attribute values will be built. It is obtained by concatenating SCH_DiagramType, SCH_DiagramName, and the FromJunctionType and FromJunction OID fields. The information is specified as expected.

2. Then, click on the ExtremityNode attribute and check the Field Name value on its General tab. The concatenated items are also specified as expected.

For the LV_LineLinks element type definition, no graphic effects have been defined to customize the link display. In this case, since no specific values are specified for the line color, line width, and line style parameters on the Effects tab, the schematic project's default parameter values are used. For the HV_LineLinks element type, it could be interesting to customize its own line parameters so this link's display is different from the LV_LineLinks links.

1. Click HV_LineLinks in the Designer tree and click the Effects tab.

2. Click the Line Color field and click the color box. The Color Editor appears.

3. Choose the desired color and click OK.
   The Color Editor closes. The Line Color parameter on the Effects tab references the newly specified color.

4. Type a value in the Line Width field if you want to change the width of the lines for the HV_LineLinks element type.

The following screen shot shows the Effects tab content after these steps:
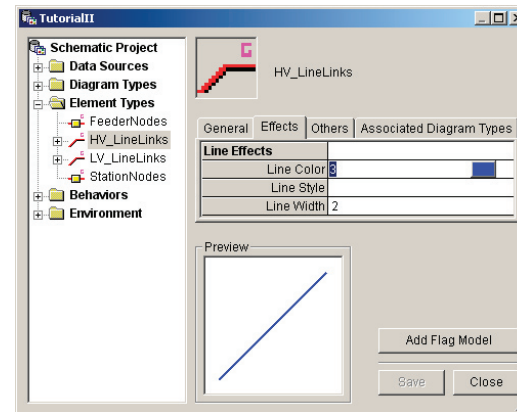
The HV_LineLinks element type definition should be correct.
You must now associate this new element type with the All diagram type before testing that your HV_LineLinks links are displayed correctly in the diagram.

5. Right-click the All diagram type entry in the Designer tree and click Edit Associations.

6. Check the HV_LineLinks item and click OK.

The HV_LineLinks item is added to the All diagram type's Associated Element Types tab.

If you switch to run mode and open the All diagram, it appears as follows:
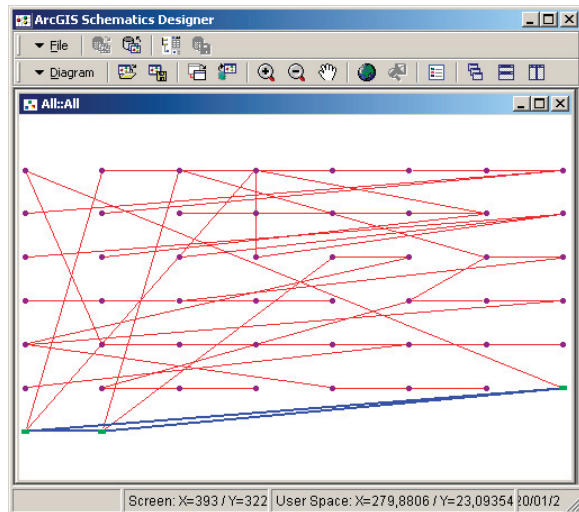
## Displaying schematic elements using their geographic coordinates

Since no geometric attribute has been defined for the node element types (that is, x- and y-coordinates are not specified for the schematic node elements), ArcGIS Schematics automatically places the nodes contained in the All diagram on a default grid. You must now create two specific attributes for the StationNodes and FeederNodes element types so the node elements implemented by these element types display at their geographic positions. These two attributes are InitialXPosition and InitialYPosition.

Generally, InitialXPosition and InitialYPosition attributes must be linked to specific fields stored in the database.

• If your nodes' x- and y-coordinates are clearly stored in two fields in your database, you can define these attributes, such as simple attribute fields.

• If no x and y fields are clearly available in your database tables and if the data source used to return the node elements from this database is an ESRI data source, ArcGIS Schematics can decode the information from the ESRI Shape field using a geometry attribute.

In this example, the StationNodes and FeederNodes element types' specified queries return the node elements from the Station and Feeder tables using the CURRENTDS data source. As this data source is an ESRI GDB data source, you can create geometry attributes for defining the InitialXPosition and InitialYPosition attributes.

1. Right-click the FeederNodes element type entry and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Click InitialXPosition in the Name drop-down list.

3. Click Geometry Attribute in the Type drop-down list.
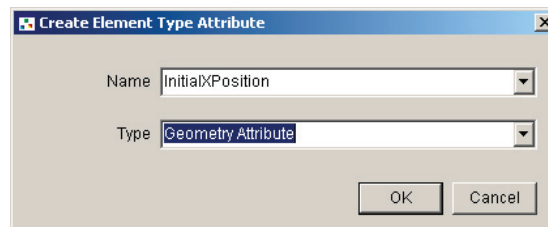
4. Click OK.

The Create Element Type Attribute dialog box closes. A new Attributes entry is automatically created below the FeederNodes element type entry, and the newly created InitialXPosition attribute is referenced below this new entry.

5. Keep the default value for the Type parameter (X-coordinate).

6. Click SHAPE in the Field Name drop-down list and click OK.

The InitialXPosition attribute's General tab content appears as follows:
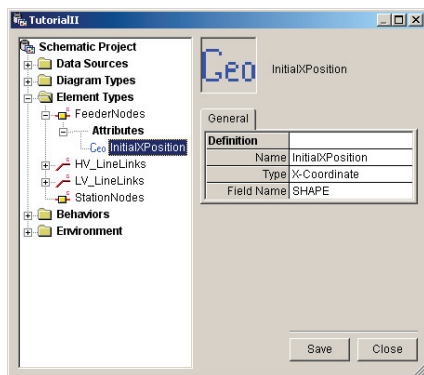
Repeat the process to create the InitialYPosition attribute required to display the feeder nodes according to their y-coordinates.

7. Right-click the FeederNodes element type entry and click Create Attribute. The Create Element Type Attribute dialog box opens.

8. Click InitialYPosition from the Name drop-down list and click Geometry Attribute from the Type drop-down list .

9. Click OK.

10. Click Y-coordinate from the Type drop-down list on the InitialYPosition attribute's General tab, .

11. Click SHAPE in the Field Name drop-down list and click OK.

The FeederNodes nodes are displayed at their geographic location.

12. Repeat these 1 through 11 to create the same InitialXPosition and InitialYPosition attributes for the StationNodes element type so these nodes also display at their geographic location.

13. Click Save to save the specified parameters.

You can now switch to run mode and test the All diagram display. It appears as follows:

22
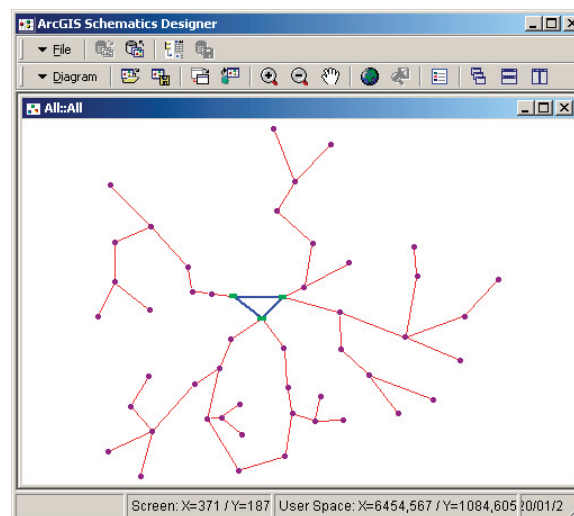
For the link element types, there is also a specific attribute InitialListPoints that can be set so ArcGIS Schematics automatically displays vertices along links.

To be interpreted correctly, the values returned by the InitialListPoints attribute must be strings formatted as follows:

N;X1;Y1; . . . ;Xi;Yi; . . . ;XN;YN where:

- N is the number of vertices.
- Xi and Yi are the coordinates of point i.
- The ";" character works as a separator.

In the Schematics_Designer_Tutorial database, the ListPoints field containing such string values is available from the LV_Line links table. To use this database field, the InitialListPoints specific attribute must be created as follows:

1. In the Designer tree, right-click the LV_LineLinks element type entry and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Click InitialListPoints from the Name drop-down list.

3. Ensure that Field Attribute is selected in the Type drop-down list.
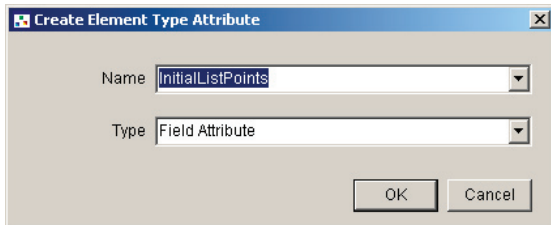
4. Click OK. The Identifier Editor dialog box opens

5. Click ListPoints in the Field list and click the down arrow.

6. Click OK.

The Identifier Editor dialog box closes.

The InitialListPoints attribute General tab appears as follows:
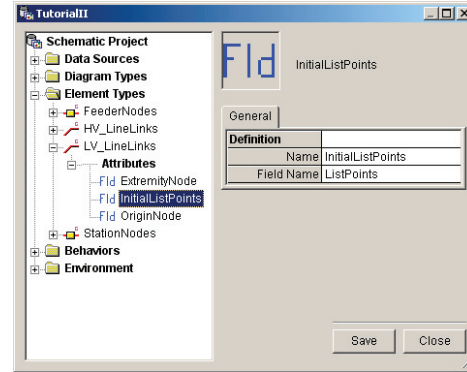
7. Click Save.

You can now switch to run mode and test the All diagram display. It should appear as follows:
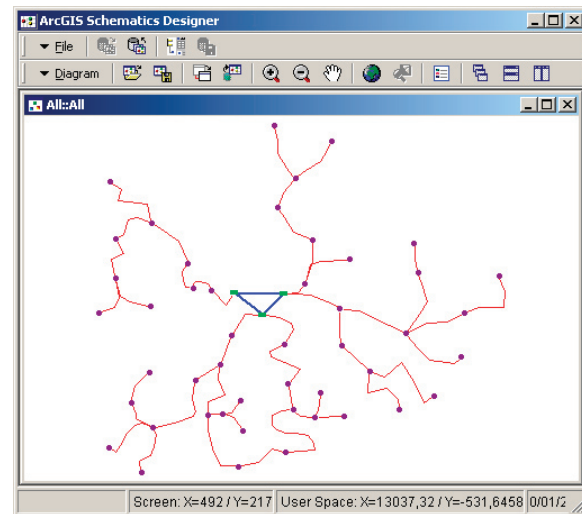
In this exercise, you will learn how to label and symbolize the schematic elements contained in your schematic diagram according to data stored in the database. You will become familiar with property creation and use.

Properties can be created and defined for all types of schematic elements on one condition—the element type must be an element group. Properties are used to highlight the characteristics of a group of schematic elements. When they are activated, they have a direct impact on the graphic representation of a network. The overall state of a network is determined by the values of properties at a given time. The values of properties often come from field values stored in the database table that the schematic element type query is based on. In this case, an attribute related to the desired database field must be defined and associated with the property. When this association is done, the schematic element's database field value must be interpreted by the property to determine each element's appearance.

This exercise is organized into three parts:

- In the first part, you will create textual properties for labeling your schematic nodes and links. You will also learn how to create and use flags to display these labels.
- In the second part, you will create discrete properties. You will learn how to create a composed CGM symbol to display a node element type and use this composed symbol to specify interesting subsymbols' effects to represent your discrete filters.
- Third, you will learn about bounded properties. You will also learn how to create pattern models that can be an interesting alternative to the basic effects usually used for displaying schematic links.

Displaying labels for StationNodes and LV_LineLinks

In this part, you will create two textual properties:

- The Label1 property will display the ObjectID of each link in blue, centered below each link.
- The Label2 property will display a label for each station node. This label will merge the values of two database fields, Name and Feeder, formatted as follows: Name (F-Feeder).

The property label will be centered at the top of the station.

In a second step, you will create a flag model and learn how flags can be attached to schematic nodes to display the Label2 property labels.

Displaying a label below the schematic links

The Label1 property will use the OBJECTID field returned by the LV_LineLinks query. Begin by creating an attribute corresponding to this OBJECTID field:
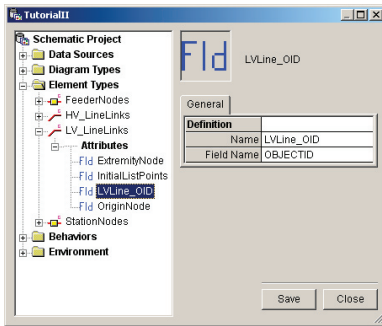
1. Right-click the LV_LineLinks entry in the Designer tree and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type the name of the new attribute in the Name field ("LVLine_ OID", in this example), click Field Attribute from the Type drop-down list, and click OK. The Identifier Editor dialog box opens.

3. Click the OBJECTID field from the Fields list, click the down arrow, then click OK.

The Identifier Editor and Create Element Type Attribute dialog boxes close. The General tab related to the newly created LVLine_OID attribute appears as follows:

Next, you must create the textual property that will manage this attribute display:

4. Right-click the LV_LineLinks entry and click Create Property.
   The Create Property dialog box opens.

5. Type a name for the new property in the Property Name field ("Label1", for example).
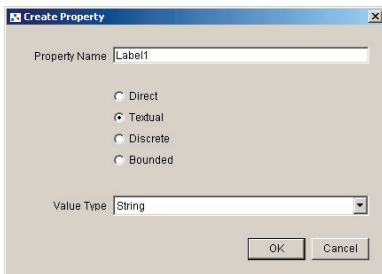
6. Since the default parameters set on this dialog box (Textual property type and String value type) are those you need, click OK.
   The Create Property dialog box closes.

Under the LV_LineLinks element type entry, a new Properties entry is automatically created. The newly created Label1 property appears under this main entry.
Relate the LVLine_OID attribute to the newly created property:

7. Click the General tab.

8. Click LVLine_OID from the Attribute Name drop-down list.

9. Click Invisible from the Legend Visibility drop-down list so this label doesn't display in the Legend subwindow and in the ArcMap TOC (when the diagram is open in the ArcMap environment).

The Label1 textual property's General tab appears as follows:
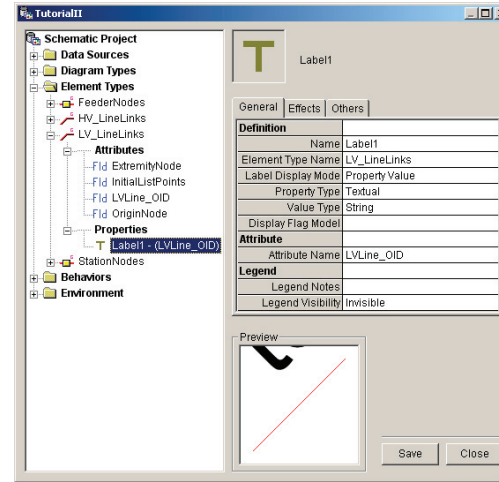
Specify that the label will display in blue, centered below each link.

10. Click the Effects tab.

11. Click Top Center in the Text Alignment drop-down list.

12. Click the Text Color field and click the color box that appears. The Color Editor opens. Choose the desired blue color and click OK.
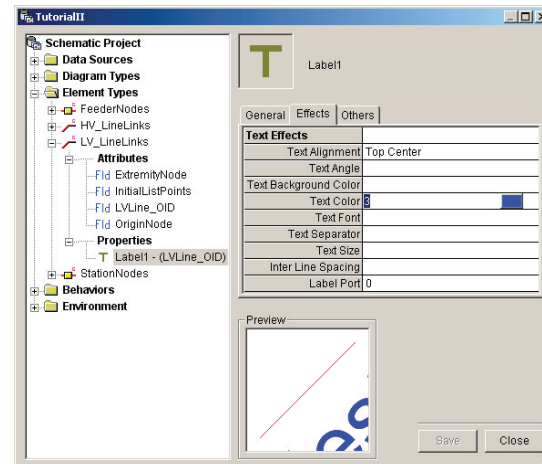
13. Click Save.

14. To test the Label1 property definition result, click Close and open the All diagram. The result should look similar to the following screen shot:
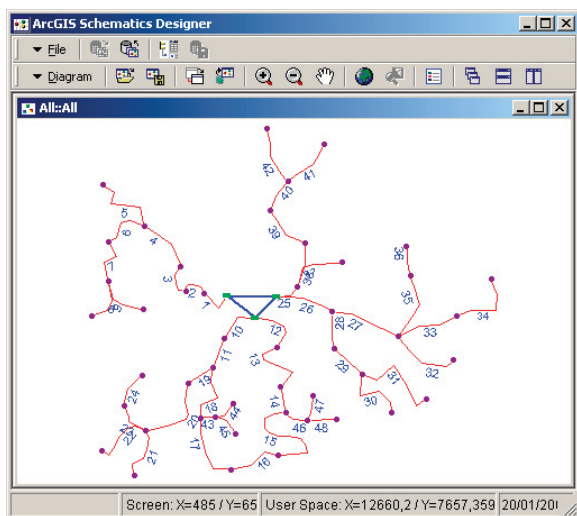

15. Switch to design mode.

**Displaying a label for the schematic nodes**

The Label2 property must use the Name and Feeder field returned by the query specified for the StationNodes element type as formatted parameters to display a label formatted as follows: Name (F-Feeder). Begin by creating the two attribute fields related to the station's Name and station's Feeder database fields before building the final formatted attribute that will be displayed.

1. Right-click the StationNodes element type and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type "StationName" in the Name field, click Field Attribute from the Type drop-down list, then click OK. The Identifier Editor dialog box opens.

3. Click the Name item from the Field's list, click the down arrow, then click OK.

The Identifier Editor and Create Element Type Attribute dialog boxes close.
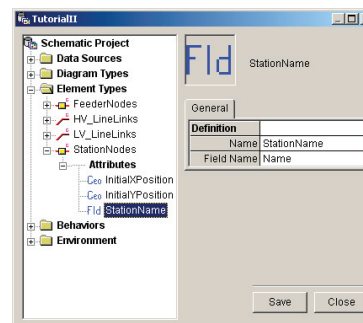
4. Proceed in the same way to create the StationFeeder attribute field related to the Station table's Feeder field.

26
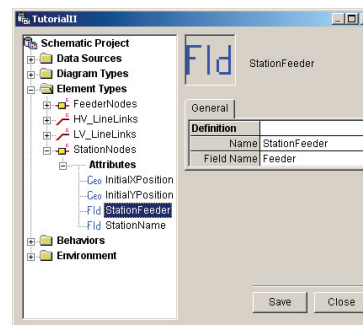
You must now create the formatted attribute that will concatenate the StationName and StationFeeder attributes to generate values formatted as follows: Name (F-Feeder).

5. Right-click the Attributes entry related to the StationNodes element type and click Create. The Create Element Type Attribute dialog box opens.

6. Type "StationLabel" in the Name field; click Formatted Attribute in the Type drop-down list, then click OK.

7. On the General tab related to the newly created attribute, you need to specify the format that will be used to build the new attribute value (Format parameter). The format is generally a string combining words or characters in C language-compliant format. For example, if your formatted attribute must generate values by concatenating two attribute values, each one being related to a string field, and if these two attribute values must be separated by the ">>" characters, the format will be: %s >> %s. The %s is used to indicate that the attribute values are string values. Use %e for double and %d for integer.

    In this example, type "%s (F-%s)" for the Format parameter.

8. Press Enter. The Format parameter is validated.

Designer automatically detects that two attributes are needed to build the formatted attribute. The new Attribute1 and Attribute2 fields are automatically displayed.

9. Click StationName from the Attribute1 drop-down list.

10. Click StationFeeder from the Attribute2 drop-down list.

The StationLabel attribute's General tab appears as follows:
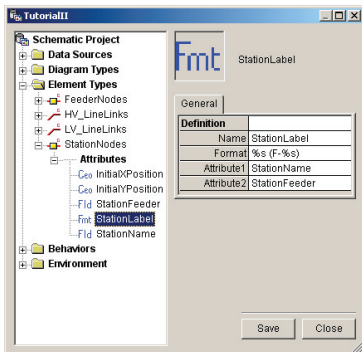
As the label values you want to display for your station nodes are defined, you can now create the textual property that will manage this information display.

1. Right-click the StationNodes element type entry and click Create Property. The Create Property dialog box opens.

2. Type "Label2" in the Name field and click OK.

Under the StationNodes element type entry, a new Properties entry is automatically created. The newly created Label2 property appears under this main entry. You need to relate the StationLabel attribute to the newly created property.

3. Click the General tab.

4. Click StationLabel from the Attribute Name drop-down list.

5. Click Invisible from the Legend Visibility drop-down list so this label doesn't display in the Legend subwindow and in the ArcMap TOC (when the diagram is open in the ArcMap environment).

27
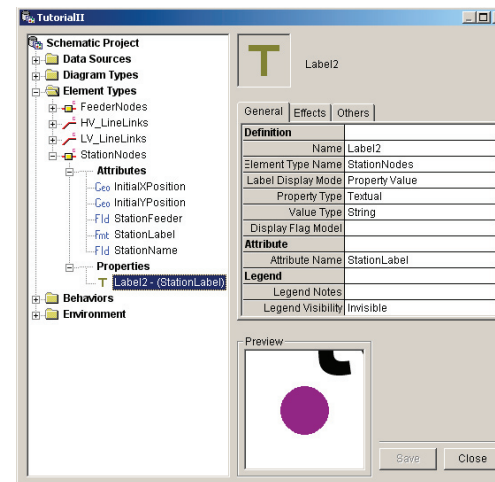
Define the property label effects. In this example, the label must be centered at the top of each node.

6. Click the Text Alignment drop-down list on the Label2 property's Effects tab and click Bottom Center.

Note that bottom center text alignment means that the element node will appear centered at the bottom of the property label; that is, the label will be centered at the top of each node.
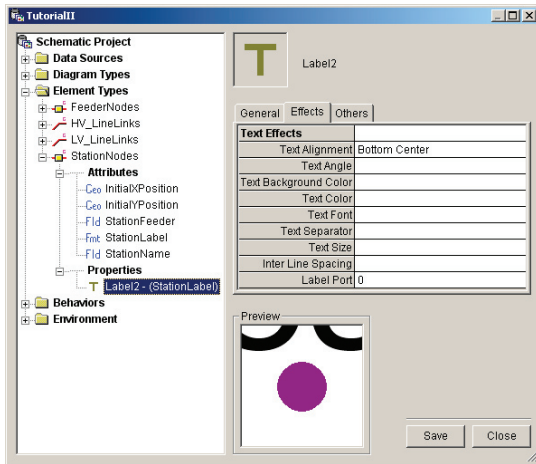
7. Click Save.

Close the Designer Editor window and test your network display. The result should be similar to the following screen shot:
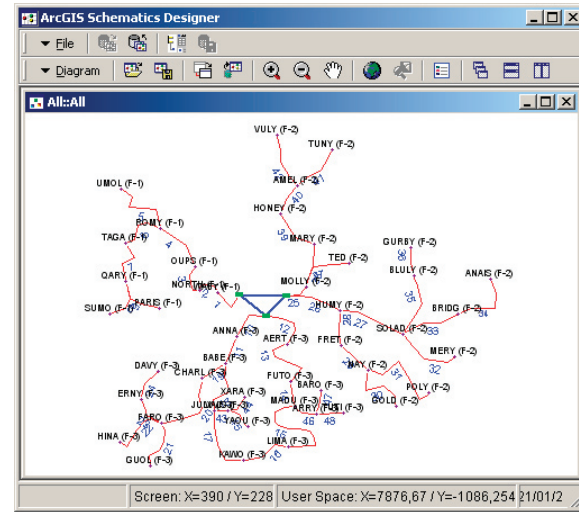
As the size of the displayed labels is too large, you can redefine the Text Size parameter value on the schematic project's Effects tab so all label sizes are decreased.

1. Switch to Design mode.

2. In the Designer tree, click Schematic Project and click the Effects tab.

3. Type "0.2" in the Text Size field.

4. Click Save, close the Designer Editor window, then test the network display again.

   When the diagram is at full extent, the labels are invisible, but when zooming in on a set of elements, the labels appear.

Displaying labels in flags

In this part, you will have to modify the Label2 property defined for the StationNodes element type so this property's labels are displayed on a flag attached to each station.

Begin by creating the flag model that will implement the flags into which the labels will display.

1. In the Designer tree, expand Environment, right-click Flag Models, then click Create. The Create Flag Model dialog box opens.

2. Type a name for the new flag model ("ClassicFlag" in this example) in the Name field.
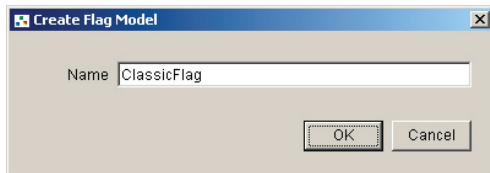
3. Click OK.

The new flag model entry is automatically created below the Flag Models entry. The Preview window shows the symbol that will be used by default to represent all flags of this type. The default flag symbol (DefaultFlag) you defined at the Schematic Project level in exercise 2 is shown in this window:
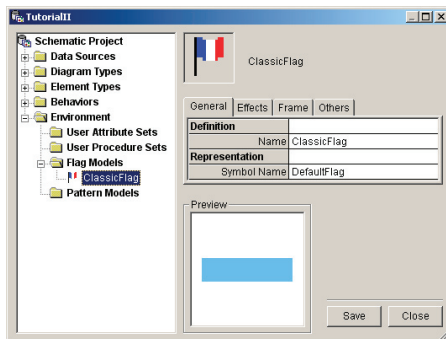

Specify the flag model parameters:

4. On the General tab, change the Symbol Name parameter by creating a new CGM symbol, if desired.

5. Click the Others tab where parameters related to the flag geometry and visualization are available.

6. Type "2" for the Scaling parameter.

7. Specify how the flags will be displayed regarding the elements they attach to: type "45" for Shift Angle and "500" for Shift Distance.

8. Click Save.

The flag model's Others tab appears as follows:
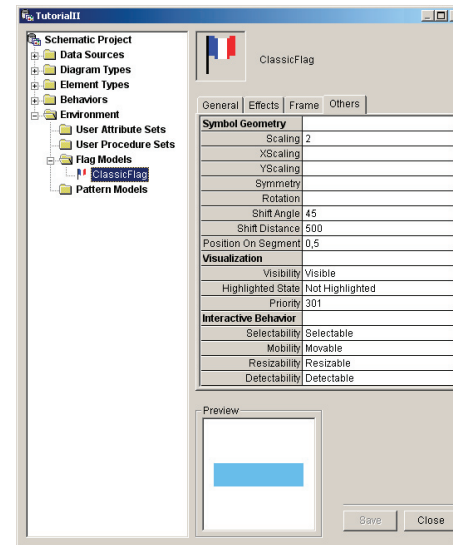
Now, modify the Label2 property so this new flag model is used to display the property label.

1. In the Designer tree, click the Label2 property entry and click the General tab.

2. Click ClassicFlag from the Display Flag Model drop-down list.

3. Click the Text Alignment parameter from the Effects tab and click Half Center so labels display centered on their attached flags.

Choosing ClassicFlag causes the ClassicFlag flag model to be associated with the StationNodes element type group; a new Flag Models tab is automatically created for this element type.

You can specify parameters on the new FlagModels tab for the pole that will connect each flag to its attached element:
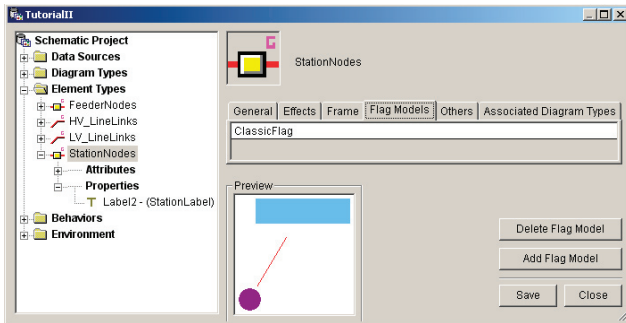
4. Click the StationNodes element type in the Designer tree, and click the Flag Models tab.

5. Click the ClassicFlag flag model line that has just been automatically added to the Flag Models tab and click the ellipsis button.
The Pole Effect dialog box opens.

6. Click the Effects tab, click the Line Color parameter, then specify the color you want using the Color Editor.
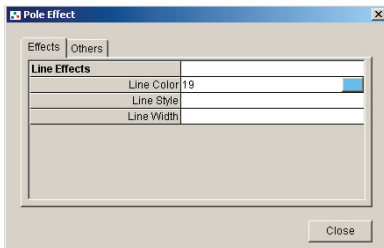
7. Click Close. The Pole Effect dialog box closes.

8. Click Save. Switch to run mode and test the flags display.

After zooming in on part of the All schematic diagram content, the result is similar to the following screen shot:
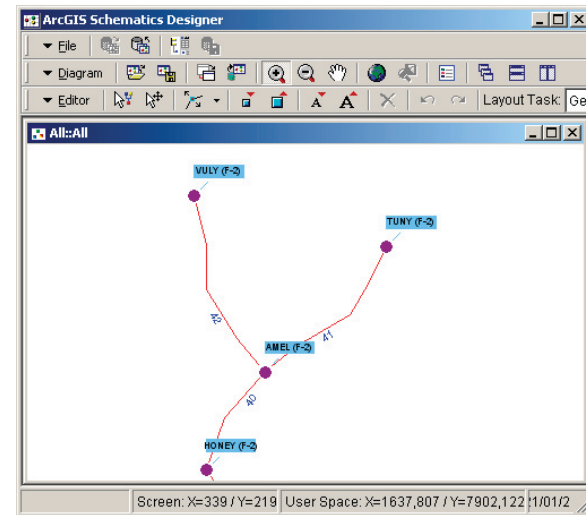
Creating discrete properties for StationNodes and LV_LineLinks

A discrete property has specified values that belong to a finite set of numbers or alphanumeric codes. The values for the property are generally returned by an attribute that must be associated with the property.

Two discrete properties will be created:

- First, symbolize the low-voltage links display according to the Category database field values. Each link has a Small, Medium, or Big (S, M or B value respectively). A discrete property will be created, and display rules will be defined for the discrete filters based on these categories.
- After defining a composed CGM symbol to represent the station nodes, you will learn how to define subsymbol effects to symbolize the station nodes display according to the Type database field values. Since this field stores discrete values (A, B, C, D, E, or F), a discrete property with six discrete filters (one for each value) will be created. Subsymbol display rules will then be defined for each discrete filter.

Symbolizing the schematic links according to their category

The Category field returned by the LV_LineLinks element type query stores the category of each link. Each link has S, M, or B category value. In this section, you need to represent this information graphically. You will create the DisplayByCategory discrete property composed of three discrete filters, with each filter corresponding to one category value. This property will display LV_LineLinks according to the Category field as follows:

| Category value	Filter graphic effect
S	Line Width: 1.0
M	Line Width: 2.0
B	Line Width: 4.0

Since the DisplayByCategory property will use the Category field values returned by the query, begin by creating an attribute field corresponding to this field.

1. Expand the Element Types entry, right-click LV_LineLinks, then click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type a name for the attribute ("CategoryLine" in this example) in the Name field, click Field Attribute from the Type drop-down list, then click OK.

3. Click Category from the Fields list on the Identifier Editor dialog box, click the down arrow to validate this field, then click OK.

The Identifier Editor and Create Element Type Attribute dialog boxes close.

The CategoryLine attribute's General tab appears as follows:
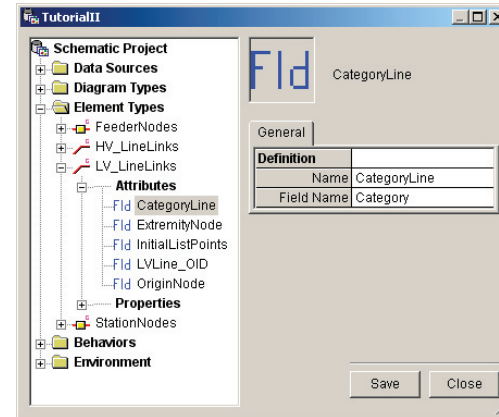
Create the DisplayByCategory discrete property that will display each LV_LineLinks link according to the CategoryLine attribute value.

4. Right-click the Properties entry displayed below the LV_LineLinks element type and click Create. The Create Property dialog box opens.

5. Type "DisplayByCategory" in the Property Name field, check the Discrete option, then keep String for the Value Type parameter.
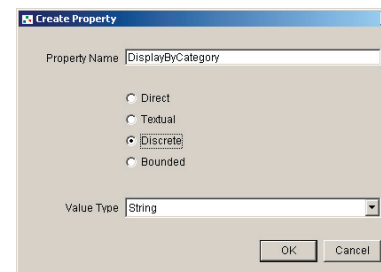
6. Click OK. The Create Property dialog box closes.

Now, you have to associate this new property with the CategoryLine attribute.

7. Click CategoryLine from the Attribute Name drop-down list on the DisplayByCategory property's General tab.

Now, create the discrete filters related to each Category field's value.

8. Click the Add Filter button displayed on the lower right corner of the Designer Editor window. The Add Discrete Filter dialog box opens.

9. Type a name for the discrete filter ("Small" for example), and specify the value for the discrete filter ("S" in this example). This is one of the values used by the Category field stored in the LV_Line table.
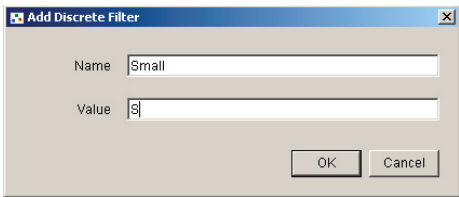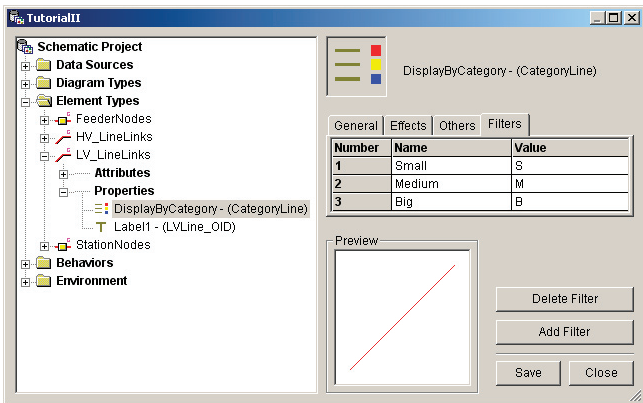

10. Click OK .

The Filters tab appears for the DisplayByCategory property. A line corresponding to the discrete filter you have just created is shown on this tab.

11. Repeat steps 9 and 10 to create two new discrete filters for the M and B values of the Category field database.

The Filters tab appears as follows:

Define the graphic effects related to each discrete filter.

12. Click the Filters tab, click the first discrete filter parameter line (Small filter), then click the ellipsis button.

The Filter Effects dialog box opens. The links that will verify this Small discrete filter must display with a line width of 1.0.

13. Click the Effects tab and type "1.0" in the Line Width field.
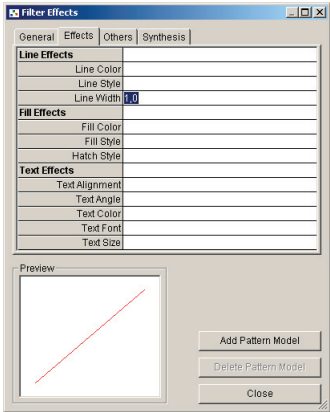
14. Click Close. The Filter Effects dialog box closes.

15. Repeat steps 12 through 14 to define the effects related to Medium and Big discrete filters as follows:

 • For the Line Width parameter related to the Medium filter, type "2.0".

 • For the Line Width parameter related to the Big filter, type "4.0".

16. Click Save.

When testing the All diagram content display, it appears as follows:
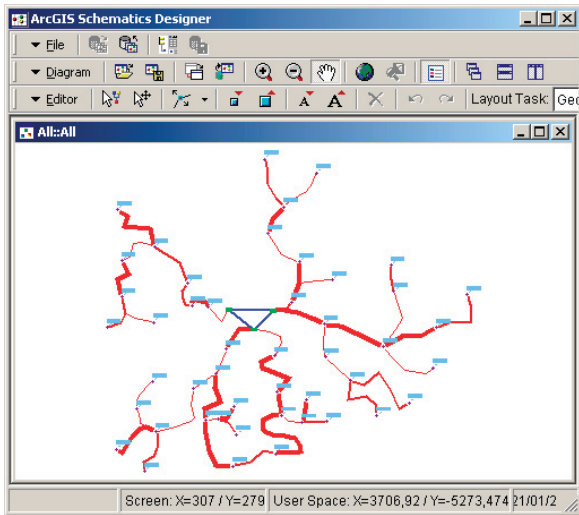
For a better understanding of the other exercises, you will temporarily deactivate the display of the ClassicFlag.

1. Click the Label2 entry and cancel the ClassicFlag selection from the Display Flag Model field.

2. Click the StationNodes element type in the Designer tree and click the Flag Models tab.

3. Click the line corresponding to the ClassicFlag flag model and click Delete Flag Model.

4. Click Save.

Symbolizing the schematic nodes according to their type

The Type field returned by the StationNodes element type query stores the type of each station. Each station node is either an A, B, C, D, E, or F type. The purpose of this section is to create the DisplayByType discrete property composed of six discrete filters, with each filter corresponding to one type value.

In this example, you will create a CGM symbol based on the default node symbol you created during the second exercise, and you will add two subsymbols. This composite symbol will be used by default to represent the StationNodes elements. Then, when defining the graphic effects related to the six discrete filters you create, you will have interesting alternatives for representing the filters by specifying subsymbol effects. The following table lists the graphic effects to specify for each discrete filter:

Type value	Filter graphic effects
A	The first subsymbol will be invisible.
B	The second subsymbol will be invisible.
C	The first subsymbol will be invisible; the second one will be rotated 90 degrees, and its fill color will be changed.
D	The two subsymbols will be invisible. The fill color of the basic main symbol will be changed.
E	The two subsymbols will both be rotated 45 degrees.
F	A new CGM symbol will be used to represent this type of station.

Begin by creating the composite CGM symbol. As the StationNodes element type is represented by the default node symbol, specify this new composed symbol for representing StationNodes.

1. Expand the Element Types entry in the Designer tree and click StationNodes.

2. Click the StationNodes' General tab, click the Symbol Name field, then click the Symbol Editor button ![icon]. The CGM Symbol Editor and Vector Drawing tool appears.

3. Click the Insert Rectangle button and draw a green rectangle as in the following screen shot.

Note: To change the rectangle color fill, double-click the newly created rectangle to open the Graphic Attributes dialog box; click the Fill Attributes tab and change the Color parameter.

4. Click File and click Save. Type a name for the first subsymbol ("SubSymbol1" for example), and click OK.
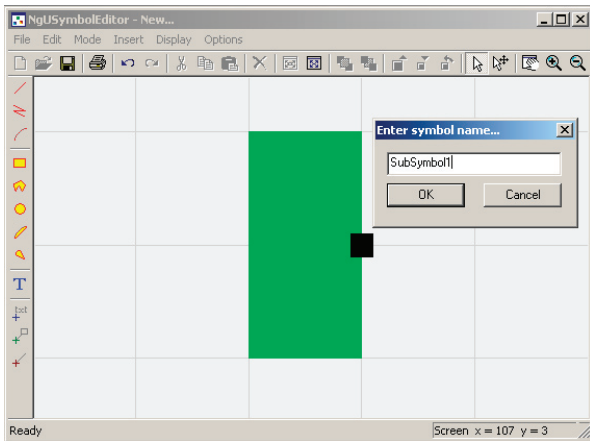
5. Click File and click New. Click the Insert Rectangle button and draw a pink rectangle as in the following screen shot. Save this new symbol (as SubSymbol2, for example).
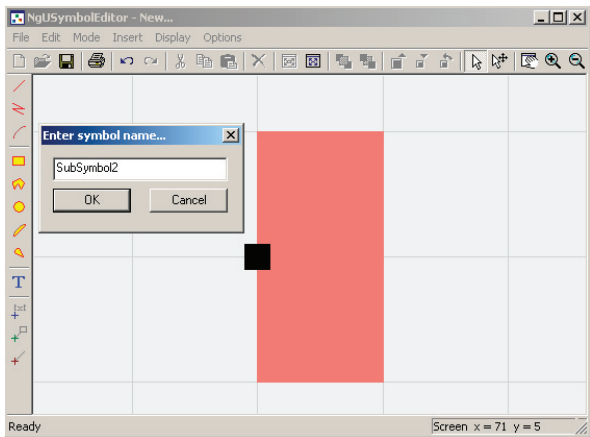
Now, you will create the composite CGM symbol. This symbol will be based on the DefaultNode CGM symbol built during the second exercise and will combine the two subsymbols you have just created.

6. Click File and click New.

7. Click File and click Import. The Symbol Browser dialog box opens.

8. Select the DefaultNode CGM symbol you created during the second exercise ("Defining the Default Node Symbol").
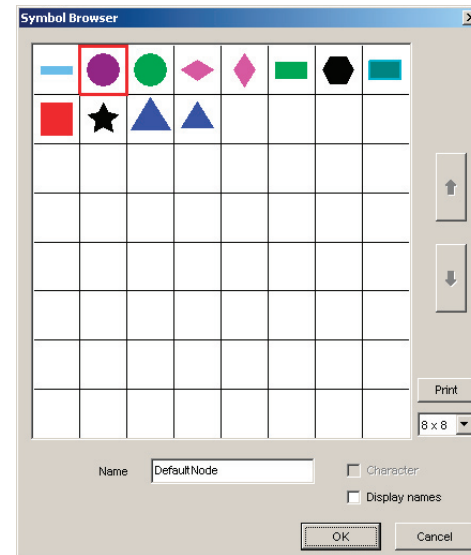
9. Click OK. The Symbol Browser dialog box closes.

10. Keep the Import graphics option checked on the Import dialog box and click OK. The symbol related to the DefaultNode CGM symbol is displayed.

Insert the first subsymbol you have created.

11. Click Insert, point to SubSymbol, then click Internal SubSymbol. The Symbol Browser dialog box that opens

12. Click SubSymbol1 and click OK.

SubSymbol1 is inserted.

13. Repeat steps 11 and 12 to insert the second subsymbol.

14. Click File and click Save.Type a name for this new symbol ("StationSymbol" for example).

34

15. Click OK

15. Exit the CGM Symbol Editor and Vector Drawing tool.

The newly composed symbol is automatically used as the StationNodes default symbol.

The Sub Symbols tab, which contains all subsymbols that comprise this symbol, appears among the other tabs.
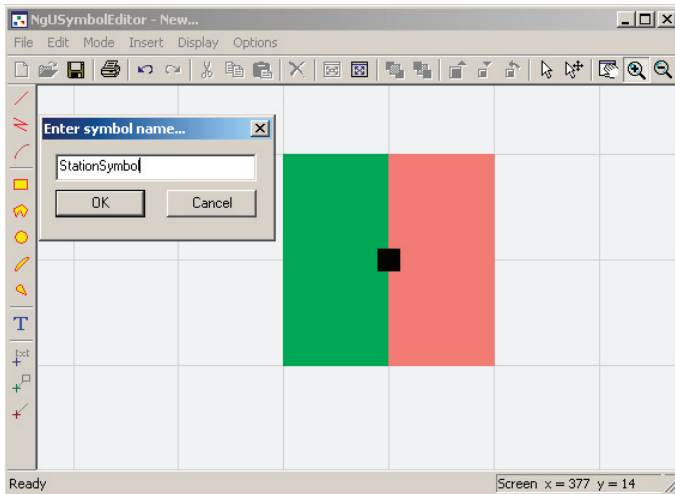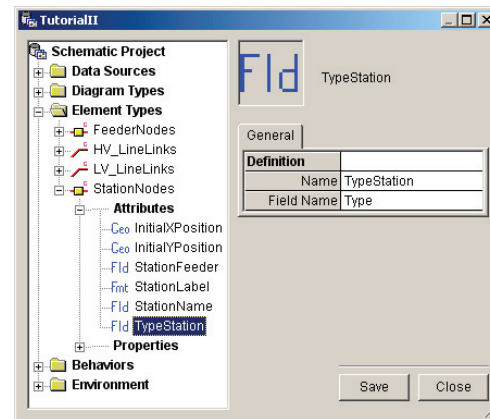

Create the DisplayByType discrete property.

As this discrete property will be based on the Type field values stored in the Station table, begin by creating the attribute field related to this database field.

1. Right-click the StationNodes element type entry and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type a name for the attribute ("TypeStation" in this example) in the Name field.

3. Click Field Attribute value from the Type drop-down list and click OK.

4. On the Identifier Editor dialog box that opens, click Type from the Fields list, click the Down arrow to validate this field, and click OK.

The Identifier Editor and the Create Element Type Attribute dialog boxes close.

The TypeStation General tab now appears as follows:

Next, create the DisplayByType discrete property that will display each StationNodes node according to the TypeStation attribute value.
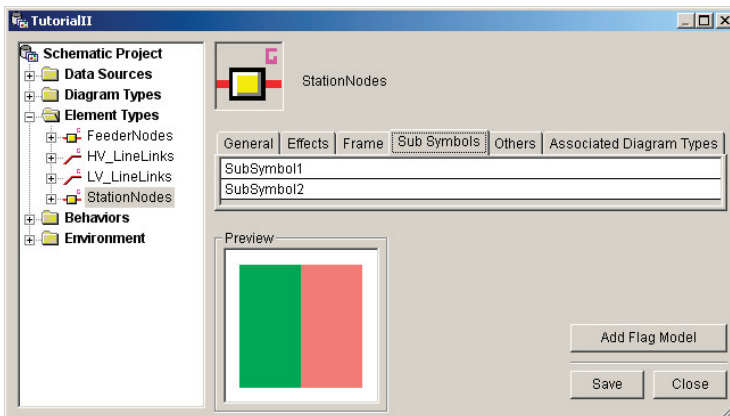
5. Below the StationNodes element type, right-click Properties and click Create. The Create Property dialog box opens.
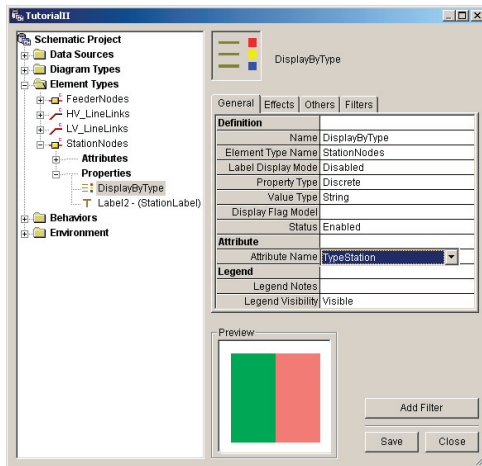
6. Type "DisplayByType" in the Property Name field, check the Discrete option, keep String for the Value Type parameter, and click OK.

The Create Property dialog box closes.

You must now associate the property with the TypeStation attribute.

7. Click the DisplayByType General tab and click TypeStation from the Attribute Name drop-down list.

Create the discrete filters related to each Type field's value.

8. Click Add Filter. The Add Discrete Filter dialog box opens.

9. Type a name for the discrete filter ("Type A" for example) and specify the first value for the discrete filter ("A" in this example). This is one of the values in the Type field stored in the Station table.

10. Click OK.

The Filters tab containing the first created discrete filter appears for the DisplayByType property.

11. Repeat steps 8 through 10 to create five new discrete filters for the B, C, D, E, and F values of the Type field database.

The Filters tab appears as follows:
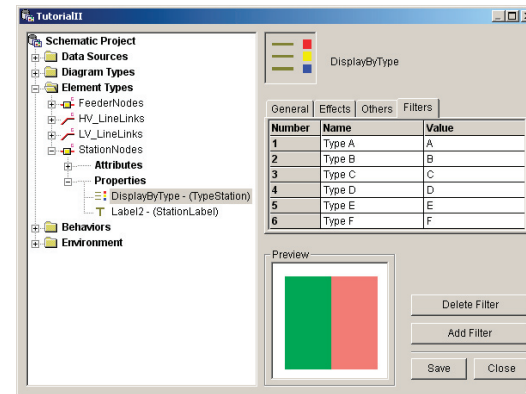
Next, define the graphic effects that will represent each discrete filter.

To define graphic effects related to a filter, click the desired discrete filter parameter on the Filters tab. Then, by clicking the ellipsis button at the end of this parameter line, the Filter Effects form automatically opens.

In this case, as the station symbol is composed of subsymbols, a Sub Symbols tab containing all subsymbols that comprise the StationNodes symbol will be available on the Filter Effects dialog box.

Since filter effects you need to specify will work on subsymbols, the Sub Symbols tab will be the central tab of this exercise.
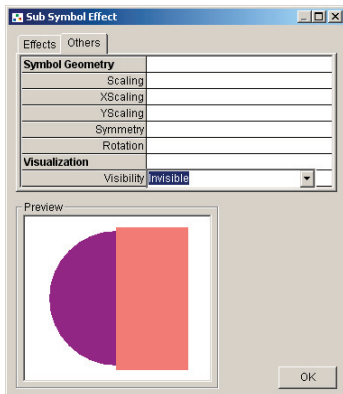
Start with the Type A filter to set the first subsymbol as invisible.

1. Click the Filters tab, click the Type A discrete filter, then click the ellipsis button. The Filter Effects dialog box opens.

2. Click the Sub Symbol tab.

3. Double-click the SubSymbol1 and field and click the ellipsis button.
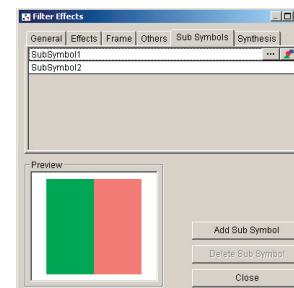
The Sub Symbol Effect dialog box related to SubSymbol1 opens.

4. Click the Others tab, and click Invisible from the Visibility drop-down list.


5. Click OK. The Sub Symbol Effect dialog box closes.

6. Close the Filter Effects dialog box.

Then, specify the Type B filter effects to set the second subsymbol as invisible.
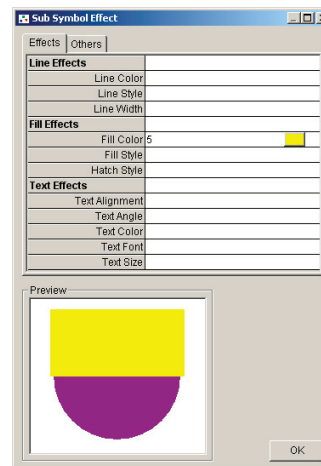
7. Click the Type B discrete filter on the Filters tab and click the ellipsis button. The Filter Effects dialog box opens.

8. Click the Sub Symbols tab.

9. Double-click the SubSymbol2 field and click the ellipsis button.

10. Click the Others tab and click Invisible from the Visibility drop-down list.
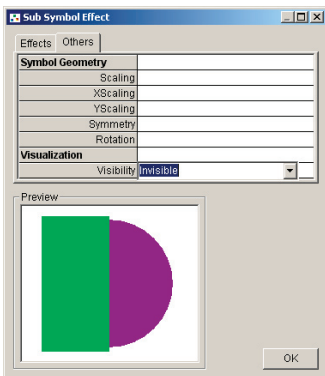
11. Click OK.

12. Close the Filter Effects dialog box.

For the Type C discrete filter graphic effects, the first subsymbol must be invisible, and the second subsymbol will be rotated 90 degrees with a different fill color.

13. Open the Filter Effects dialog box corresponding to the third discrete filter (Type C).

14. Click the Sub Symbols tab and open the Sub Symbol Effect dialog box related to SubSymbol1.

15. Click the Others tab and click invisible from the Visibility drop-down list, then click Ok.

16. Open the Sub Symbol Effect dialog box related to SubSymbol2.

17. Click the Others tab, type "90" in the Rotation field.

18. Click the Effects tab. Click the Fill Color field and set the color you want.


19. Click OK.

20. Close the Filter Effects dialog box.

Next, define the filter effects related to the Type D discrete filter. For this filter, you need to set the two subsymbols to be invisible and change the color for the basic symbol.

21. Open the Filter Effects dialog box corresponding to the fourth discrete filter (Type D).

22. Click the Sub Symbols tab and open the Sub Symbol Effect dialog box related to SubSymbol1.

23. Click the Others tab, click invisible from the Visibility drop-down list and click Ok.

24. Repeat steps 22 and 23 to deactivate the visibility for SubSymbol2.

25. Click the Effects tab on the Filter Effects dialog box. Click the Fill Color parameter and set the color you want.

The Preview subwindow on the Filter Effects dialog box appears similarly to the following screen shot:
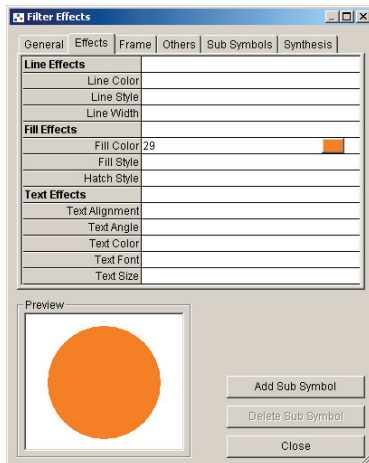
26. Click Close.

For the Type E discrete filter, the subsymbols must both be rotated 45 degrees:

27. Open the Filter Effects dialog box corresponding to the fifth discrete filter (Type E).

28. Click the Sub Symbols tab and open the Sub Symbol Effect dialog box related to SubSymbol1.

29. Click the Others tab. Type "45" for the Rotation parameter, then click OK.

30. Repeat steps 28 and 29 for SubSymbol2 to rotate this subsymbol in the same way.

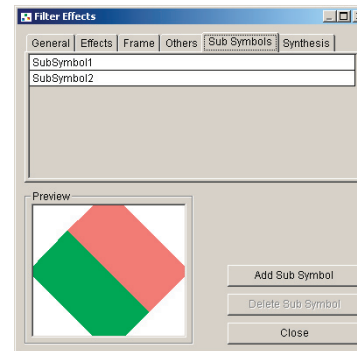The Preview subwindow on the Filter Effects dialog box appears similarly to the following screen shot:

31. Click Close.

For the Type F discrete filter, you must create a new CGM symbol.

32. Open the Filter Effects dialog box corresponding to the sixth discrete filter (Type F).

33. Click the General tab, click the Symbol Name field, then click the Symbol Editor button . The CGM Symbol Editor and Vector Drawing tool appears.

34. Draw a new CGM symbol and save it.
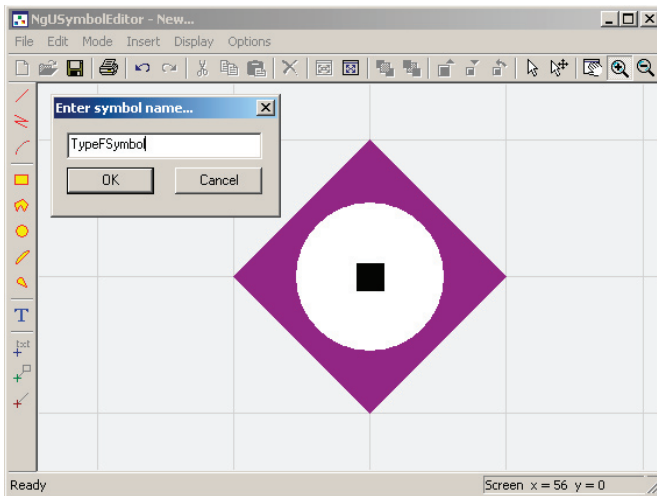
35. Exit the CGM Symbol Editor and Vector Drawing tool.

The Symbol Name field on the Filter Effects dialog box is automatically populated with the new CGM symbol name. The Preview subwindow appears similarly to the following screen shot:
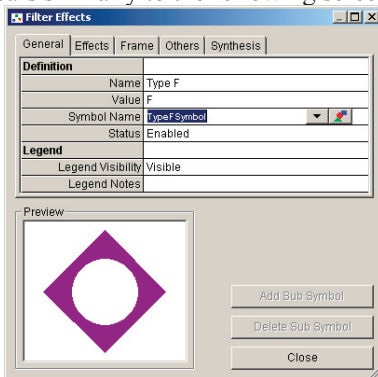
36. Click Close.

37. Click Save on the Designer main window.

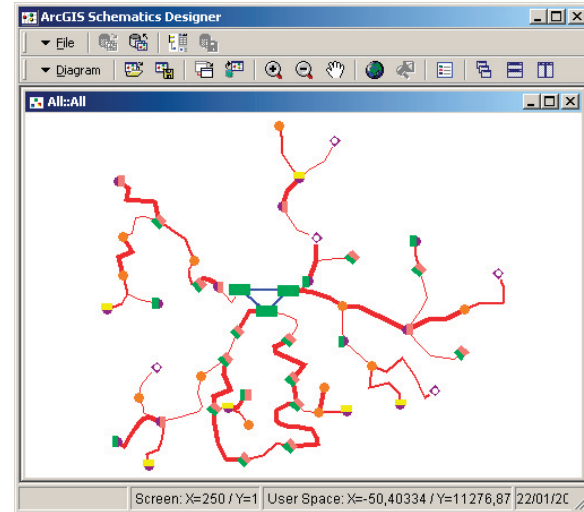To test the impact of the DisplayByType property on the schematic StationNodes elements, click Close, switch to run mode, and open the All diagram. It appears as follows:

Creating bounded properties for StationNodes and LV_LineLinks

In this section, you will learn about bounded properties. A bounded property allows you to specify the graphic effects matching different value ranges defined for this property. The value ranges defined for the property are generally based on an attribute that must be associated with the property.

Two bounded properties will be created:

- You will first animate the station display according to the Capacity database field values. Bounded property and display rules will be defined for each bounded filter, so the size of the symbols used to represent stations varies according to the Capacity field values.

- Then, you will create pattern models. A pattern is a symbol or a character that is repeated once or several times along a link path route. Displaying patterns is an interesting alternative to the basic line styles you can set for links. Using pattern models to represent bounded filters, you will animate the network links according to the Rate database field values.

Modifying the schematic station sizes according to their capacities

The Capacity field returned by the StationNodes query stores the capacity of each station as an integer value.

You will need to represent this information graphically. You will create the DisplayByCapacity bounded property composed of four bounded filters with each filter corresponding to one range of values. This property will display the nodes according to the Capacity field values as follows:

Capacity lower value	Capacity upper value	Filter graphic effects
0	2	Scaling factor: 1.0
3	5	Scaling factor: 2.0
6	8	Scaling factor: 3.0
9	10	Scaling factor: 4.0

Since the DisplayByCapacity property will use the Capacity field returned by the query, you must begin by creating an attribute corresponding to this field.

1. Right-click the Attributes entry displayed under the StationNodes element type and click Create. The Create Element Type Attribute dialog box opens.

2. Type the name that will be used to reference this attribute in the Name field ("StationCapacity", for example). Click Field Attribute from the Type drop-down list, then click OK.

3. Choose the Capacity field from the Fields list on the Identifier Editor dialog box and click OK.

At the end of these steps, the StationCapacity attribute appears as follows:
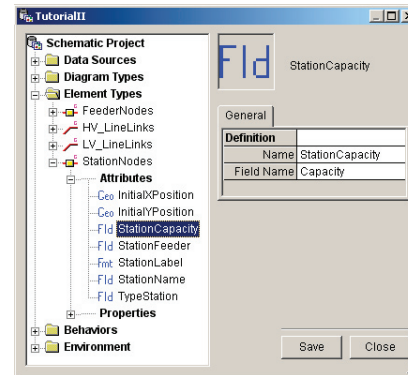
Create the DisplayByCapacity bounded property that will display each station according to the StationCapacity attribute values.

4. Expand StationNodes in the Designer tree, right-click Properties, then click Create. The Create Property dialog box opens.

5. Type "DisplayByCapacity" in the Name field.

6. Check the Bounded option.

7. Click Integer on the Value Type drop-down list.

8. Click OK.
   The new DisplayByCapacity property is automatically referenced below the Properties entry.

9. Click General tab and click StationCapacity from the Attribute Name drop-down list.
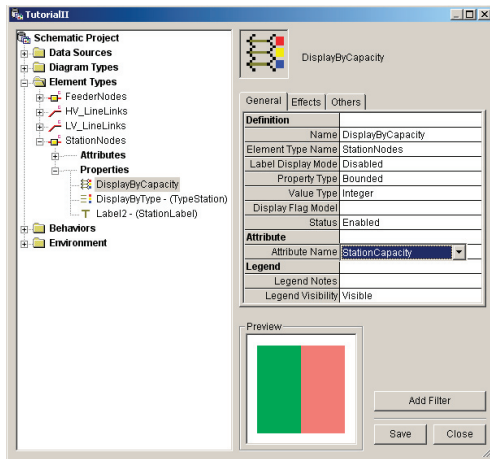
Next, you need to create the bounded filters.

10. Click Add Filter. The Add Bounded Filter dialog box opens.

11. Type a name for the first bounded filter in the Name field ("Capacity 1" for example) and set the lower and upper values (0 and 2 in this exercise) that will define this first bounded range.
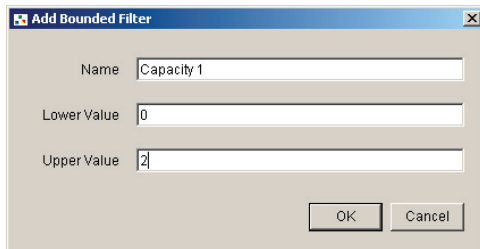
12. Click OK.
    The Add Bounded Filter dialog box closes.

13. Repeat steps 10 through 12 to create three new filters corresponding to the following ranges:

    • The 3 to 5 value range (Capacity 2)

    • The 6 to 8 value range (Capacity 3)
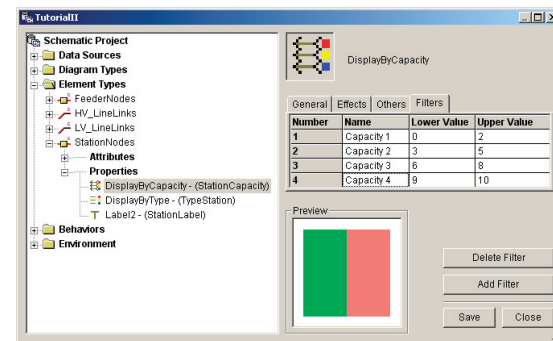
    • The 9 to 10 value range (Capacity 4)

The Filters tab should look like the following screen shot:

Define the graphic effects that will represent each bounded filter.

Start with the Capacity 1 bounded filter that must set the station scaling factor to 1.

14. Click the Filters tab, click the Capacity 1 bounded filter, then click the ellipsis button. The Filter Effects dialog box opens.

15. Click the Others tab and type "1" for the Scaling parameter.
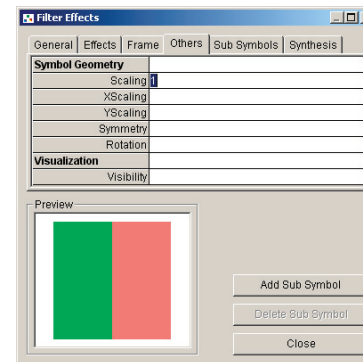
16. Click Close.

17. For defining the graphic effects related to Capacity 2, 3, and 4, repeat steps 14 through 16 to set the Scaling parameter with values of 2, 3, and 5, respectively.

18. Click Save.

To test the impact of the DisplayByCapacity property on the schematic diagram, click Close, switch to run mode, then open the diagram. It appears as follows:
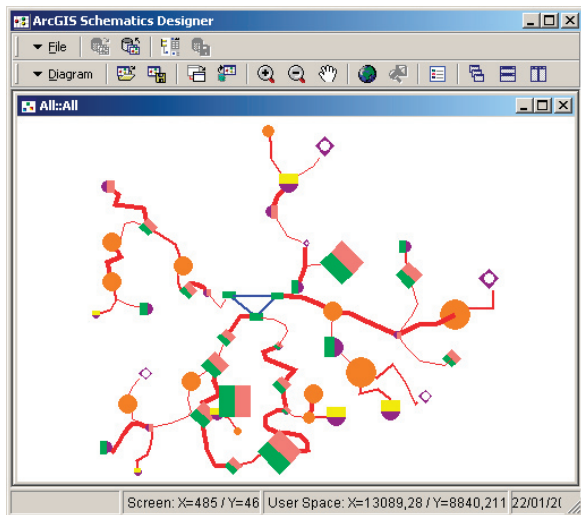
Using pattern models to animate the links according to the Rate field values

In the following steps, you will create two pattern models that you will use afterward to display the links according to a new bounded property with values that will be based on the Rate field stored in the LV_Line table.

The PatternSymbol1 model you will create is a pattern model symbol that will display an arrow placed on the middle of each link path route.

The PatternText1 model will be a pattern model text character that displays an X character along each link path route. The rules to repeat the pattern along the segment are as follows:

• Spacing between items is 5.0 units.

• The first item is 3.0 units away from link origin (Shift Distance).

The steps required to create the PatternSymbol1 model are detailed below:

1. Right-click the Pattern Models entry under the Environment entry in the Designer tree and click Create. The Create Pattern Model dialog box opens

2. Type the name that will be used to reference your first pattern model ("PatternSymbol1" in the example).

3. Click Pattern Symbol from the Type drop-down list.
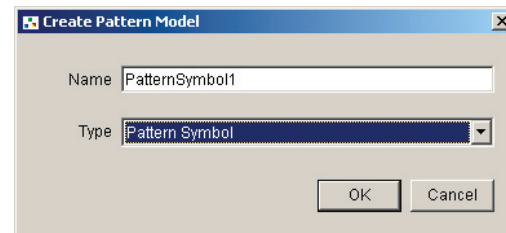
4. Click OK.

5. Click the General tab. Specify the CGM symbol that will represent each pattern. Click the Symbol Name field, and click the Symbol Editor button. The CGM Symbol Editor and Vector Drawing tool appears.

6. Draw the new CGM file representing the desired symbol and save it.
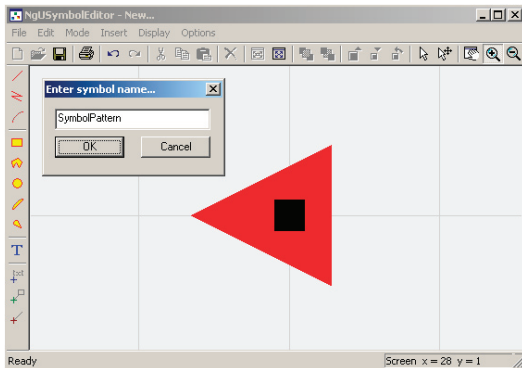
7. Close the CGM Symbol Editor and Vector Drawing tool.
   The parameters needed to define the rules that will be used to repeat the patterns along the segment are those available from the pattern model Others tab.

8. Click the Others tab.
   The default values set for the Positioning Mode, Shift Distance, Spacing, and Max Items On Segment parameters are those you need to display a pattern on the middle of each link path route.

   You have created your first pattern model definition.

The PatternText1 pattern will change automatically according to the color of the link in which it will be displayed. The rules to repeat the pattern along the segment are as follows:

 • Spacing between items is 5.0 units.
  • The first item is 3.0 units away from link origin (Shift Distance).
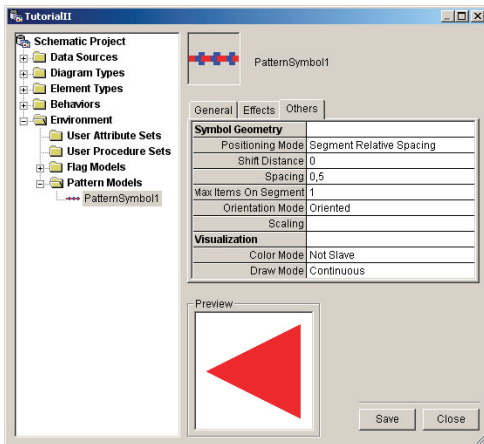
1. Right-click Pattern Models in the Designer tree and click Create. The Create Pattern Model dialog box opens.

2. Type the name that will be used to reference your second pattern model in the Name field ("PatternText1", for example), click Pattern Text value from the Type drop-down list, then click OK.

3. Click the new pattern model entry below the Pattern Models entry. Click the General tab and type "X" in the Pattern Text field, as in the following screen shot:
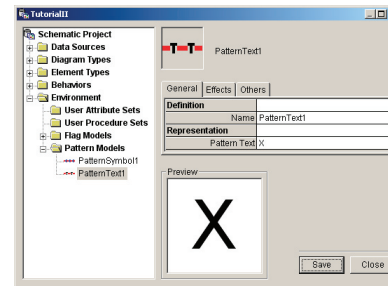
4. Click the Effects tab and click Half Left from the Text Alignment drop-down list.

Now, you must define the rules that will be used to repeat the patterns along the segment. The pattern model must implement five items along the link path route, with the first item 3 units away from the link origin:

5. Click the Others tab and click Absolute Spacing from the Positioning Mode drop-down list.

6. Type "3" in the Shift Distance field.

7. Type "5" in the Spacing field.

8. Click Slave from the Color Mode drop-down list so the pattern color automatically changes according to the link color.
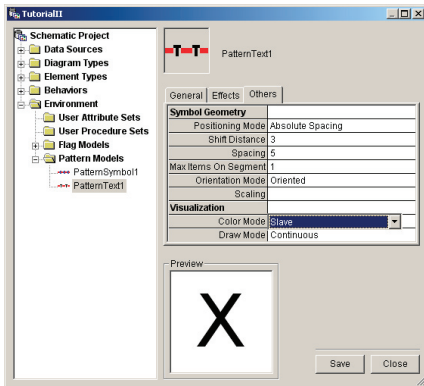
You have created your second pattern model definition.

9. Click Save.

Create the DisplayByRate bounded property that you will use to animate LV_LineLinks according to the Rate field returned by the LV_LineLinks query. This Rate field contains double values. The property will display the links according to the Rate field as follows:

Rate lower value	Rate upper value	Filter graphic effects
0	200,000	Use the PatternSymbol1 pattern model.
200,000	400,000	Display the link in green and use the PatternText1 pattern model.
400,000	600,000	Use the PatternSymbol1 and PatternText1 pattern models.
600,000	800,000	Display the link in pink.
800,000	1,000,000	Display the link in black with the dashed dotted line style.

Because the DisplayByRate bounded property will use the Rate field returned by the query to animate the link display, you must create an attribute corresponding to this field.

1. Right-click the Attributes entry under the LV_LineLinks element type in the Designer tree, and click Create. The Create Element Type Attribute dialog box opens.

2. Type "RateLine" in the Name field, click Field Attribute from the Type drop-down list, then click OK.

3. Click Rate from the Field's list on the Identifier Editor dialog box and click OK.
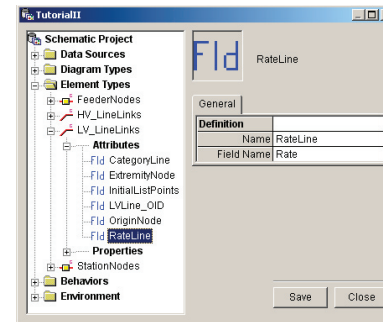
Create the DisplayByRate bounded property that will display each link according to the created RateLine attribute values.

4. Right-click the Properties entry under the LV_LineLinks element type in the Designer tree and click Create. The Create Property dialog box opens.

5. Type "DisplayByRate" in the Name field, check the Bounded option from the Value Type drop-down list, click Double, then click OK.

6. Click the property's General tab. Click RateLine from the Attribute Name drop-down list.

44
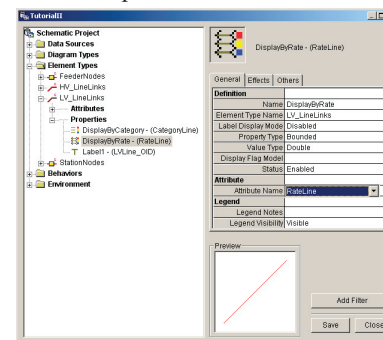
Next, you need to create the five bounded filters that will be used to represent the links according to the DisplayByRate property values.

1. Click Add Filter to create the property's first bounded filter. The Add Bounded Filter dialog box opens.

2. Type "Rate 1" in the Name field and set the lower and the upper values that will define this first bounded range ("0" and "200000", respectively).
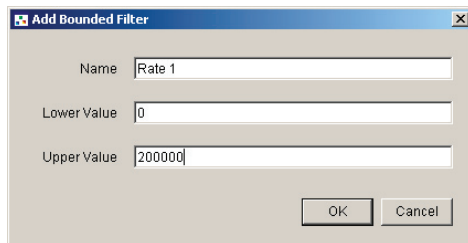
3. Click OK.

4. Repeat steps 2 and 3 to create four other filters corresponding to the value ranges specified on the dialog box below.

5. Click the Filters tab, click the Rate 1 bounded filter and click the ellipsis button. The Filter Effects dialog box opens.

6. Click Add Pattern Model. The Add Pattern Model dialog box opens.

7. Select PatternSymbol1 in the Name drop-down list and click OK.

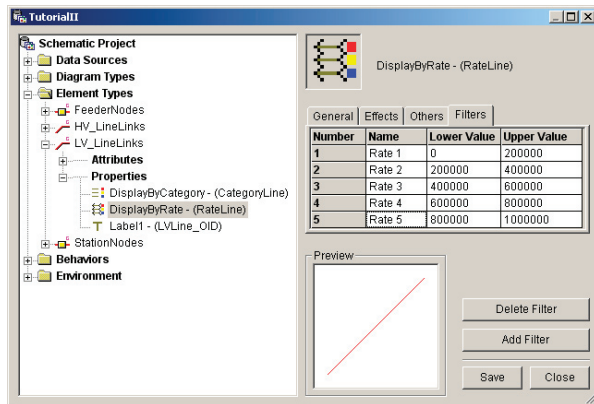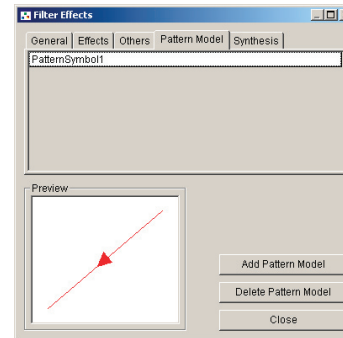The Pattern Model tab automatically appears on the Filter Effects dialog box related to the first bounded filter.
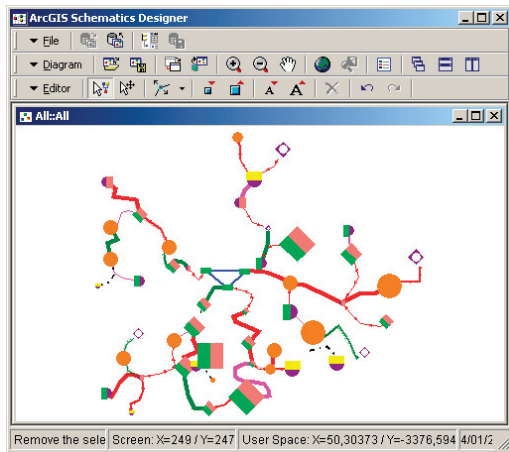

8. Click Close on the Filter Effects dialog box.

9. Click the Rate 2 bounded filter paramter line and open its associated Filter Effects dialog box.
   a) Click Add Pattern Model.
   b) Select PatternText1 from the Name drop-down list on the Add Pattern Model dialog box, then click OK.
   c) Click the Effects tab on the Filter Effects dialog box, specify the Line Color parameter as green, then click Close.

10. For the Rate 3 bounded filter, repeat steps 5 through 7 twice; the first time to add the PatternSymbol1 model and the second time to add the PatternText1 model.

For the Rate 4 and Rate 5 bounded filter effects, the graphic effects to specify are all on the Effects tab of the Filter Effects dialog box.

11. Click the Filters tab, click the Rate 4 bounded filter, then click the ellipsis button. The Filter Effects dialog box opens.
    a) Click the Effects tab.
    b) Choose pink as the Line Color parameter, then click Close.

13. Click the Filters tab, click the Rate 5 bounded filter, then click the ellipsis button. The Filter Effects dialog box opens.
    a) Click the Effects tab.
    b) Choose black as the Line Color parameter.
    c) Click Dashed-Dotted from the Line Style drop-down list, then click Close.

14. Click Save.

Note: If the Pattern Symbol adding is directly reflected in the preview window, the Pattern Text adding is not reflected because the pattern sizes are too small. The impact of the Pattern Text will be only visible in your diagram as in the following screenshot.

When you test the All diagram display, it appears as follows:

Three subnetworks comprise the network stored in the Schematics_Designer _Tutorial database, each one describing a feeder. This exercise will examine the low-voltage part of the network. You will create new schematic diagram and element types to generate three schematic diagrams, one for each low-voltage subnetwork related to a given feeder. In this way, the Feeder field stored in the Station table and the FromJunctionType and ToJunctionType fields stored in the LV_Line table can be used to filter out the records stored in the database, so you will display three diagrams, one per feeder.

The exercise is organized into three parts:

- You will begin by creating a new diagram type that will implement these three diagrams.

- New node and link element types will be created and associated with this new diagram type. You will become familiar with the inheritance relationship that can be defined between two schematic element types. The new element type that you create will inherit from the StationNodes and LV_LineLinks element type you created in exercise 3.

- In the last part, you will learn how properties inherited from a parent element type can be redefined at the child element type level.

Until now, you created your diagram type and element types without using the Diagram Type Assistant. The Diagram Type Assistant guides you in your new diagram types creation and parameters specification, that is, configuring the diagram type parameters and the element types you want to be associated with it. In this exercise, you will learn how to use this utility during your schematic project conception.

Creating and configuring a new diagram type to visualize feeder subnetworks separately

In this step, you will define a new diagram type that will implement the three feeder subnetworks stored in the database using the Diagram Type Assistant:

1. Right-click Diagram Types in the Designer tree and click Diagram Type Assistant. The Diagram Type Assistant dialog box opens.

2. Type a name for the new diagram type in the Name text box on the active Name/Builder tab ("Feeder", for example).

3. Even if the All diagram type created in exercise 3 has no particular characteristics, the new diagram type you are creating will inherit from this All diagram type. In this case, if you decide to create specific behaviors for the parent diagram type, these behaviors will be automatically inherited at the child level. Choose All in the Parent Diagram Type drop-down list.

4. Choose Custom Query Based Builder from the Schematic Builder drop-down list.
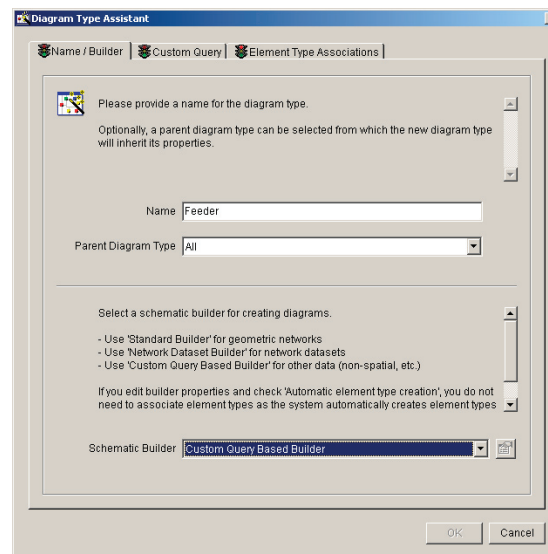
You must now specify the Custom Query tab parameters.

5. Click the Custom Query tab.

Because this new diagram type must filter out the schematic elements that will be associated with it according to the feeder subnetwork these elements belong to, the query that will return all the schematic diagrams of this type should return the distinct occurrences of the subnetworks stored in the database. In this case, the SQL query for the Feeder diagram type can be defined from the Feeder table that contains all network feeders.

6. Click the Filter data option and click CURRENTDS from the Data Source drop-down list.
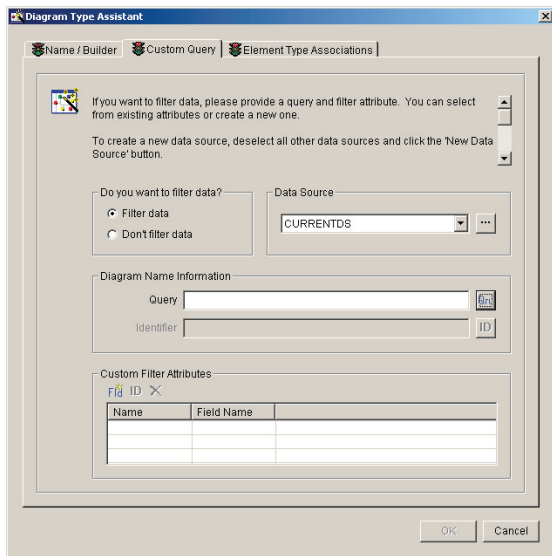
7. Click the Query Editor button at the right of the Query text box. The Query Editor dialog box opens.

8. Click the Feeder table in the Tables list and click All Records so the SELECT * FROM Feeder SQL query is automatically displayed in the Query area.

9. Click OK. The Identifier Editor automatically opens.

10. Type "Feeder" in the central text zone and click the down arrow to validate this parameter.

11. Click OBJECTID from the Fields list and click the down arrow to add this field.

12. Click OK.
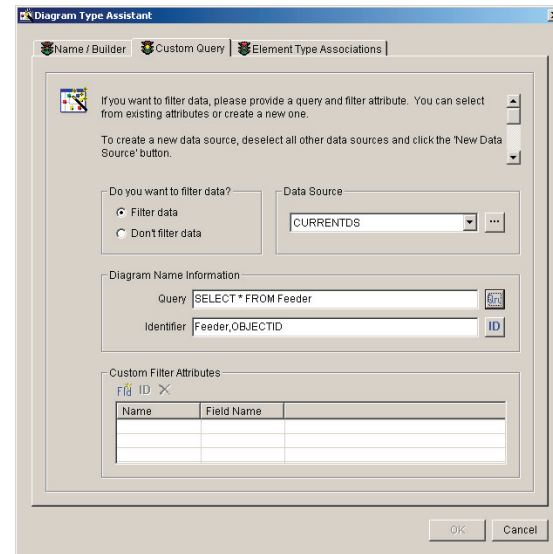
When such a diagram type is created to filter out elements stored in the database, you must define the diagram type attribute that will later be used to filter the schematic elements associated with a given diagram of this type. This way, these schematic elements will be automatically associated with each diagram of this type. In this case, it is the Feeder table's OBJECTID field that you will need.

13. Click the Create Field Attribute icon Fld in the Custom Filter Attributes area.

14. Type a name for the attribute by changing the default NewAttribute name to the desired one (FeederSubnetwork in this example), and click OK. The Identifier Editor automatically opens.

15. Click OBJECTID from the Fields list, click the down arrow, and click OK.

48

All parameters related to your new diagram type are now specified. The Custom Query tab now appears as follows:
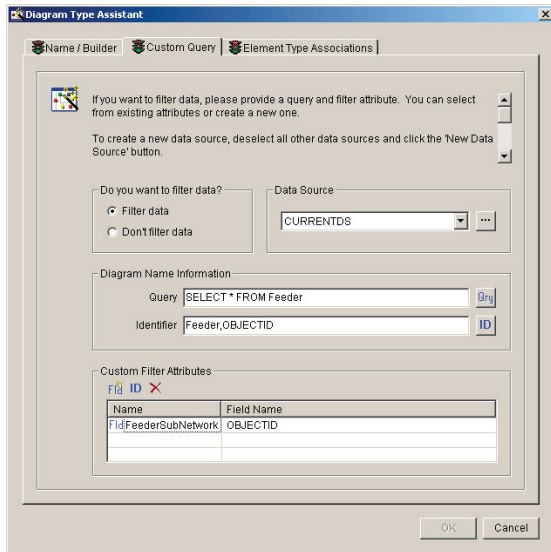
You must now specify the elements that will be contained in each feeder diagram implemented by this diagram type. Using the Diagram Type Assistant, these parameters must be set on the Element Type Associations tab.

## Creating new schematic element types to filter out nodes and links

In this section, you will create two new node element types and one new link element type that will be associated with the new Feeder diagram type. The queries that will return the elements of these types will be built according to the FeederSubNetwork attribute (defined for the Feeder diagram type) so they automatically filter out the schematic elements related to one feeder. Using the Diagram Type Assistant, those parameters can be specified on the Element Type Associations tab.
The Element Type Associations tab should contain the four element types you have created during the previous exercises. Since these element types are not associated with the current diagram type you are specifying, their related check boxes at the left is unchecked.
Now you are going to create the new node and link element types that will be associated with your diagram type.

Creating the StationNodes_F new node type

1. Click the Create Element Type button on the Element Type Associations tab. The Create Element Type dialog box opens.

2. Type "StationNodes_F" in the Name text box.

3. Click Node from the Type drop-down list .

4. Click StationNodes from the Parent Element Type drop-down list.

5. Uncheck the Element Group check box.
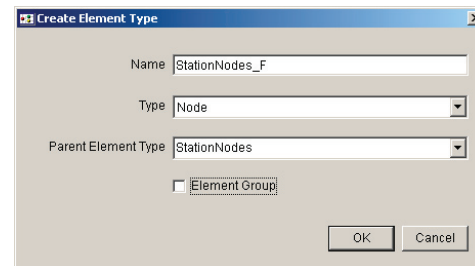
6. Click OK.

The Query Editor automatically opens to allow you to specify the query that will filter out the stations stored in the database according to the FeederSubnetwork attribute values defined for this Feeder diagram type.

7. Remove the default query from the Query text area, and type this one: "SELECT * FROM Station Where Feeder=?"

49

8. Click FeederSubnetwork (the attribute you defined for the Feeder diagram type StationNodes_F is associated with) from the Name drop-down list in the Parameters area.

9. Click N from the Text drop-down list; this attribute returns numerical values (the OBJECTID field related to this attribute stores numerical values).

10. To test your query, type "1"," 2", or "3" in the Test Value field and click Show Results in the Query area (the query result is automatically shown in the Results area).
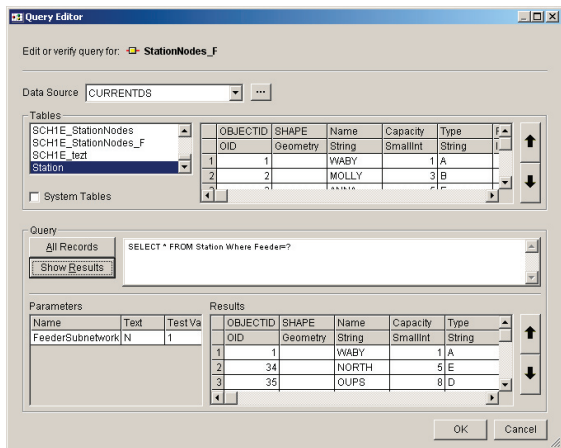
11. Click OK.

The Identifier Editor dialog box automatically opens. By default, as the StationNodes_F element type you are specifying inherits from the StationNodes element type, all the fields you specified to build the StationNodes node's identifier are automatically inherited.

12. Examine these default inherited fields to ensure the resulting identifiers are unique.

13. Click OK.

You have completed the first new StationNodes_F element type definition.

The Element Type Associations tab should now display as follows:
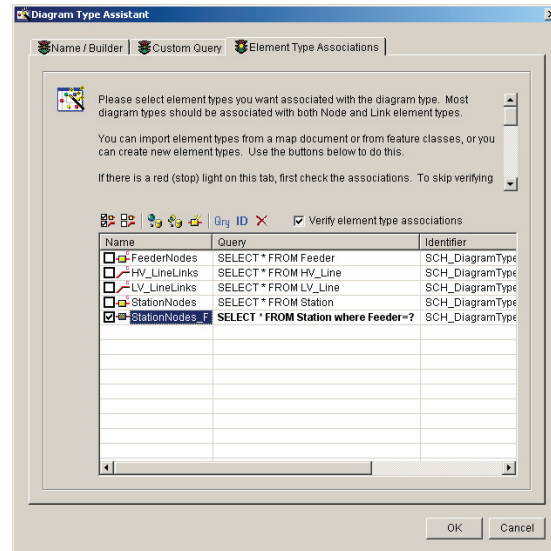
An item corresponding to StationNodes_F now displays on this tab.

Defining the FeederNodes_F new node type

Create the new FeederNodes_F element type that will return the unique feeder related to each feeder subnetwork diagram. This new node element type will inherit from the FeederNodes element type you created in exercise 3.

1. Click the Create Element Type button. The Create Element Type dialog box opens.

2. Type "FeederNodes_F" in the Name text box.

3. Click Node from the Type drop-down list.

4. Click FeederNodes from the Parent Element Type drop-down list.

5. Uncheck the Element Group check box.

6. Click OK.

The Query Editor dialog box automatically opens so you can specify the query that will return all the feeder nodes of this type.

7. Remove the default query from the Query text area, and type this one: SELECT * FROM Feeder Where OBJECTID=?

8. Click FeederSubnetwork (the attribute you defined for the Feeder diagram type the FeederNodes_F is associated with) from the Name drop-down list in the Parameters area.

9. Click N from the Text drop-down list, this attribute returns numerical values (the OBJECTID field related to this attribute stores numerical values).

10. To test your query, type "1","2", or "3" in the Test Value field and click Show Results i nthe Query area. The query result is automatically shown in the Results zone.
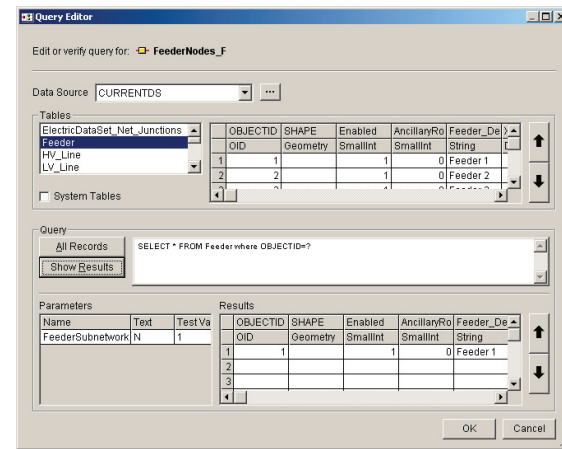

11. Click OK.

12. The Identifier Editor dialog box automatically opens. Examine the default inherited fields to ensure the resulting identifiers are unique and click OK.

You have completed the FeederNodes_F definition and the Element Type Associations tab appears as follows:
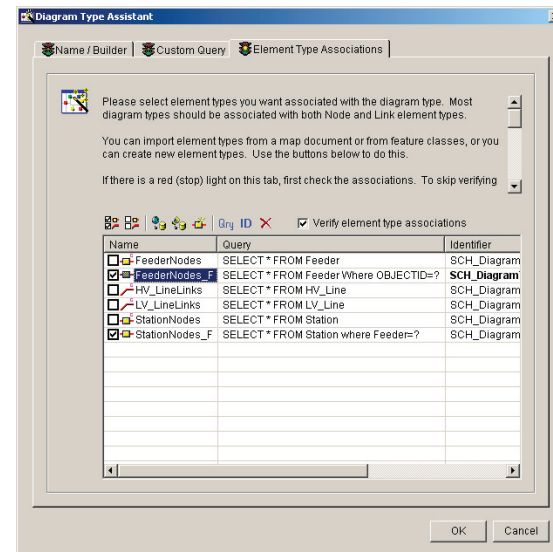
Defining the new LV_LineLinks_F element type

You will now create a new element type link that will filter out all the links stored in the LV_Line table, according to the subnetwork to which they belong. This new link type will inherit from the LV_LineLinks element type.

Note that since the HV_Line table contains links that connect Feeder nodes to other feeder nodes and the new Feeder diagram type will be used to display subnetworks related to a unique feeder, no new link type must be created from the HV_Line table.

1. Click the Create Element Type button. The Create Element Type dialog box opens.

2. Type "LV_LineLinks_F" in the Name text box.

3. Click Link from the Type drop-down list.

4. Click LV_LineLinks from the Parent Element Type drop-down list

5. Uncheck the Element Group check box.

6. Click OK.

The Query Editor dialog box automatically opens.

7. Remove the default query set in the Query area and type the following one:
   "SELECT * FROM LV_Line WHERE ToJunctionOID IN (SELECT OBJECTID FROM Station where Feeder = ?)"

8. In the Parameters area, from the Name drop-down list, click FeederSubNetwork and for the Text parameter select N.

9. Test the specified query by typing "1", "2", or "3" in the Test Value field and clicking Show Results in the Query area.
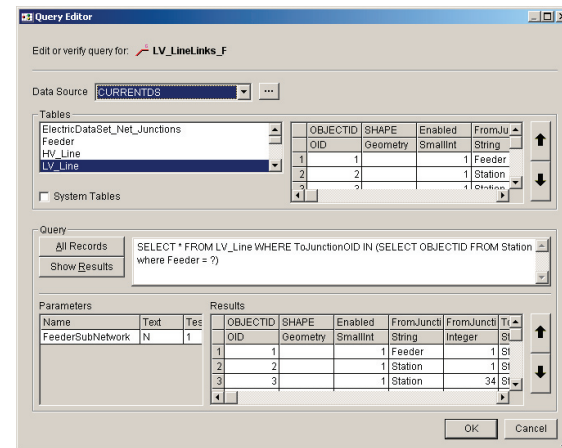
10. Click OK.
    The Identifier Editor dialog box automatically opens.

11. Examine the default inherited fields to ensure the resulting identifiers are unique and click OK.

You have completed the LV_LineLinks_F definition. The Element Type Associations tab appears as follows:
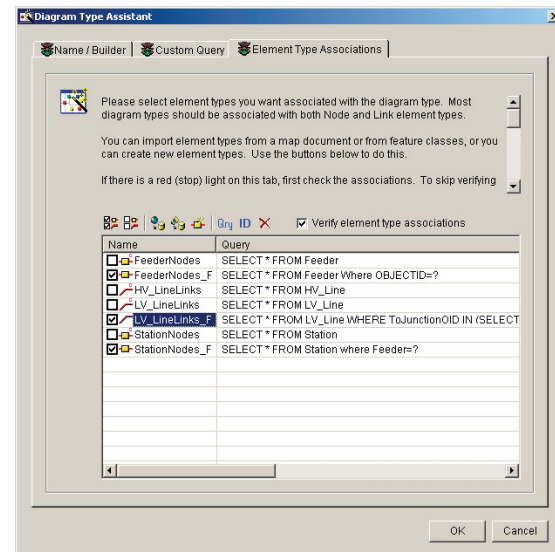
12. Click OK on the Diagram Type Assistant.

13. From the Designer tree, ensure that the StationNodes_F, FeederNodes_F, and LV_LineLinks_F element type entries are correctly displayed.
In particular, look at the LV_LineLinks element type entry. A link can be displayed only when Schematics knows to which origin and extremity nodes the link connects. In this example, because the query defined for this link element type returns all the fields required to compute the inherited OriginNode and ExtremityNode attribute values, each link should correctly connect to its origin and extremity nodes for FeederNodes_F and StationNodes_F.

14. Click Save.

You can now check that the new Feeder diagram type correctly generates three diagrams related to each Feeder subnetwork.

1. Switch to run mode by closing the Designer Editor window.

2. Click Open Diagrams. The Select Diagram To Open dialog box opens.

3. Click Feeder in the Diagram Type drop-down list.

4. The Diagram Name drop-down list contains three items (Feeder-1, Feeder-2, and Feeder-3), one for each feeder stored in the Schematics_Designer_Tutorial database. Click Feeder-1.

5. Click OK. The Feeder-1 schematic diagram opens as follows:
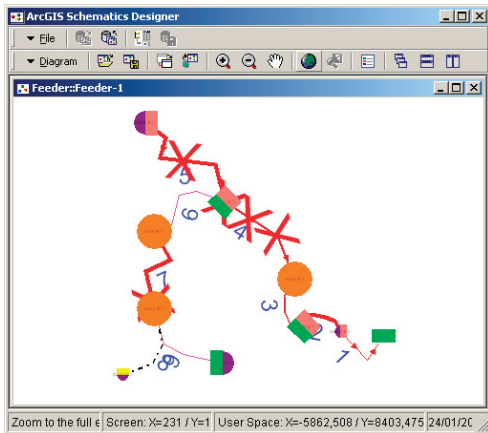
Note that the diagram names available from the Diagram Name drop-down list of the Select Diagram To Open dialog box are those returned by the Identifier diagram type parameter.

Note also that because the new StationNodes_F, FeederNodes_F, and LV_LineLinks_F element types inherit from the StationNodes, FeederNodes, and LV_LineLinks element types, all properties defined for the parent element types are automatically reported on the children element types. As the queries set for the new element types return the same parent fields needed by these inherited properties, the properties' representation is effective.

The following screen shots show how the Feeder-2 and Feeder-3 diagrams should appear:
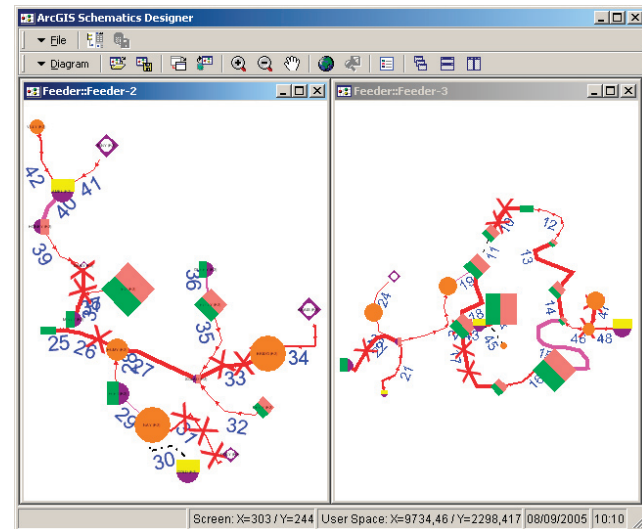
6. Close the Feeder-2 and Feeder-3 diagrams.

7. Click Edit to switch to design mode.

Redefining inherited properties for the schematic elements contained in the Feeder diagrams

Due to inheritance, the Label1 and Label2 properties defined for the StationNodes and LV_LineLinks element types are automatically reported on the StationNodes_F and LV_LineLinks_F element types. However, inherited properties can also be redefined at the child level if needed.

You will redefine these property labels as follows:

• In Feeder diagrams, the Label2 textual property will display each station NAME field value stored in the database (instead of displaying the inherited, formatted StationLabel attribute value).

• In Feeder diagrams, the Label1 textual property will display a new label that will concatenate its origin node OBJECTID and extremity node OBJECTID as follows: FromJunctionOID-ToJunctionOID.

**Associating an inherited property with an inherited attribute**

The StationName attribute field defined for the StationNodes element type corresponds to the NAME field stored in the database. As the StationNodes_F element type inherits from the StationNodes element type, this attribute is automatically inherited at the child level.
At the StationNodes element type level, the Label2 property is associated with the formatted StationLabel attribute. In the Feeder diagrams, you want to associate the Label2 inherited property with the StationName inherited attribute. For the StationNodes_F element type, the Label2 property redefining consists of reassociating this inherited property with the inherited StationName attribute.

1. Right-click the StationNodes_F entry in the Designer tree and click Associate Property. The Associate Property dialog box opens.

2. Click the inherited Label2 property from the Name drop-down list.

3. Click the StationName attribute from the Attribute drop-down list.
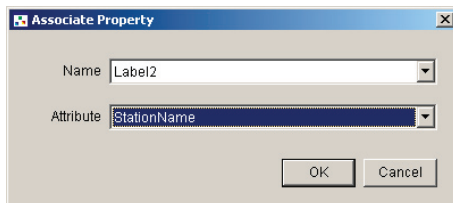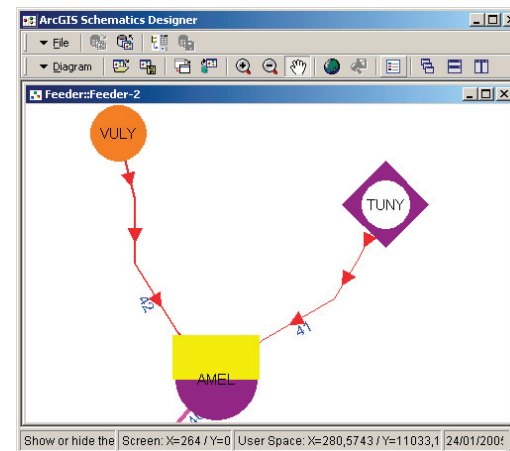

4. Click OK.

The property entry should display as follows:
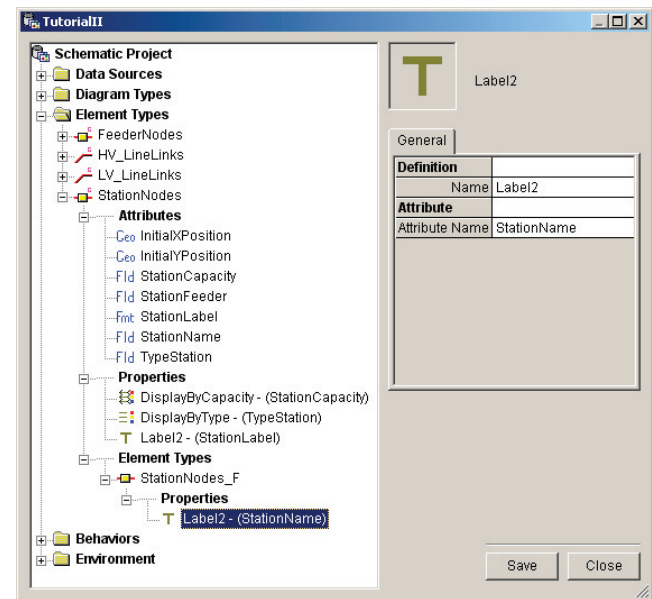
5. Click Save.

To test the inherited property label redefinition, switch to run mode and open the Feeder-1, Feeder-2, or Feeder-3 diagram. After zooming in on a set of stations, the label appears in a similarly to the following screen shot:

At the LV_LineLinks element type level, the Label1 property is associated with the LVLine_OID field attribute. For the Feeder diagrams, you will have to change the inherited property label so that this property displays a formatted label built as follows: FromJunctionOID-ToJunctionOID.

Start by creating a new attribute at the LVLineLinks_F level to build this new label. Since a field attribute can concatenate several fields returned by its related element type's query, with each concatenated field value being hyphenated, this label will be defined as a field attribute.

1. Right-click the LVLineLinks_F entry in the Designer tree and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type a name ("FromTo", for example) in the Name field. Click Field Attribute from the Type drop-down list, then click OK. The Identifier Editor dialog box automatically opens.

3. Double-click the FromJunctionOID field available from the Fields list so this field is added to the Identifier list,

4. Double-click the ToJunctionOID field so it is also added to the Identifier list.

5. Click OK.

The Identifier Editor and Create Element Type Attribute dialog boxes close. The FromTo attribute's General tab should display as follows:
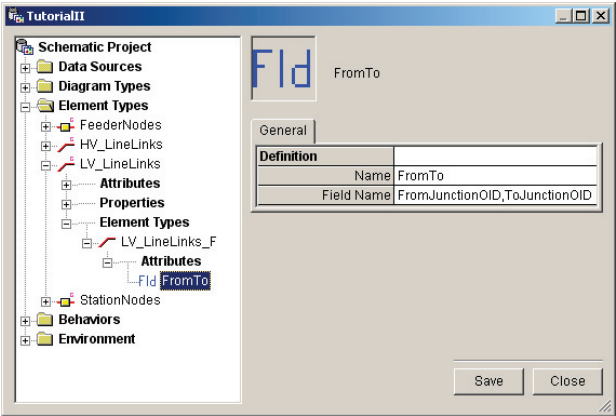
The Label1 property redefinition needs the reassociation of this inherited property with the new FromTo attribute defined at the LVLineLinks_F level.

6. Right-click the LVLineLinks_F entry in the Designer tree and click Associate Property. The Associate Property dialog box opens.

7. Click the inherited Label1 property from the Name drop-down list. Click the FromTo attribute from the Attribute drop-down list.
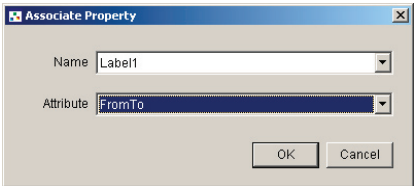
8. Click OK.

The newly associated property's General tab appears as follows:
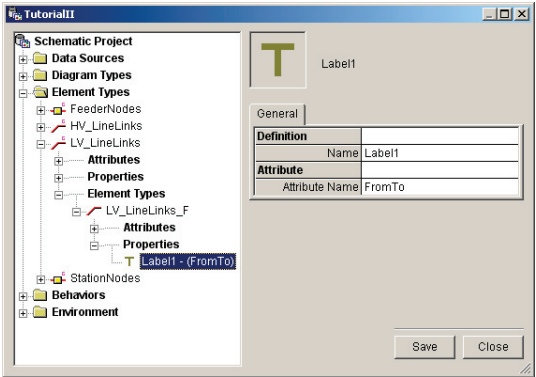
9. Click Save.

To test the inherited property label redefinition, switch to run mode and open the Feeder-1, Feeder-2, or Feeder-3 diagram. After zooming in on a set of links, the labels appear similarly way to the following screen shot:

Exercise 6: Linking features/objects and schematic elements

In this exercise, you will create and specify the parameters required to manage associations between features/objects and schematic elements so the Propagate Schematic Selection To Map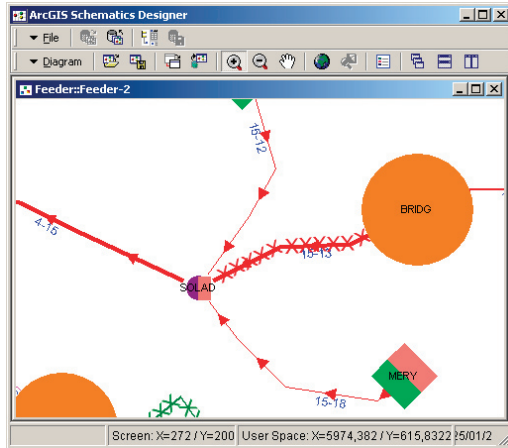 and Propagate Map Selection To Schematic commands work when you open the All, Feeder-1, Feeder-2, or Feeder-3 diagram in ArcMap.This ensure that the Identify Results dialog box displays properties related to the clicked schematic element and those related to its associated features/objects when you are using the Identify tool on any schematic element contained in these diagrams.

For diagrams that work with the Custom Query Based Builder, Schematics automatically tries to manage the associations between features/objects contained in a geodatabase and schematic elements contained in these diagrams by searching for three predefined user data values on the schematic elements—FOID, FCN, and FDSN:

- The FOID user data value must specify the ObjectID of the feature(s)/object(s) associated with the schematic element.

- The FCN user data value indicates the feature class names of the features and/or the object table names of the objects you want to be associated with the schematic element.

- The FDSN user data value specifies the names of the data sources used to connect the geodatabase that contains these associated features/objects (these data sources must use an ESRI connection—ESRI Access GDB, file GDB, SDE® GDB, etc).

Note that only the FOID user data is mandatory for Schematics to manage these associations. If the FCN user data is missing for the schematic element, Schematics searches for a feature class or table whose name corresponds to the name of the element type that implements the schematic element. If the FDSN user data is missing, Schematics uses the data source specified for the element type query.

The FOID, FCN, and FDSN user data value can return unique values or values list when a schematic element is associated with several features/objects that can be stored in a unique or several feature classes/object tables coming from a unique datasource or different datasources. To associate schematic elements with a set of features/objects, the user data is associated with formatted attributes with each feature/object value separated by the semicolon (;) character in a list.

Creating the user data for the StationNodes element type

Since user data is always related to an attribute which returns the user data values, any user data definition starts with an attribute creation.

The schematic elements implemented by the StationNodes element type must be associated with the features contained in the Station feature class stored in the Schematics_Designer_Tutorial database. Since the CURRENTDS data source used to query the schematic elements of this type connects to the Schematics_Designer_Tutorial database, the FDSN user data doesn't need to be specified. You will only specify the FOID and FCN user data values.

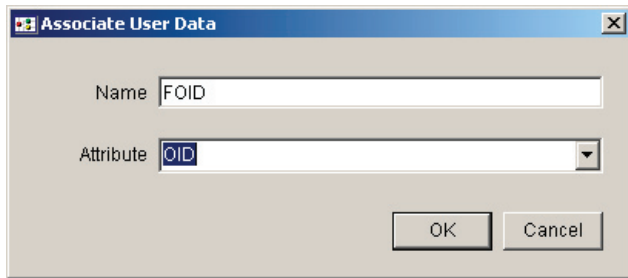1. Right-click StationNodes in the Designer tree and point to Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type a name for the new attribute ("OID "for example).

3. Click Field Attribute for the Type parameter and click OK. The Identifier Editor dialog box opens.

4. Double-click ObjectID from the Fields list and click OK.

The OID attribute entry appears as follows:
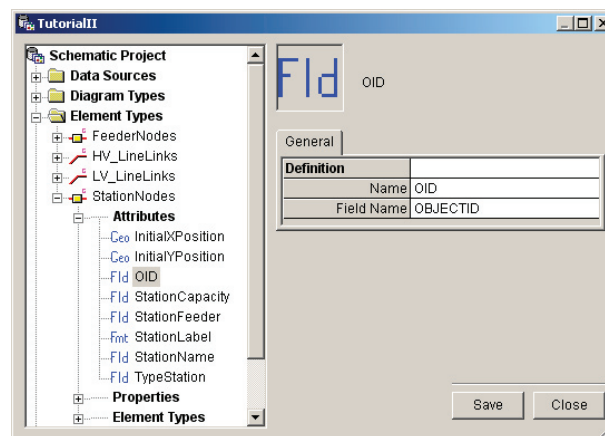
You must now create the FOID predefined user data related to this attribute.

5. Right-click StationNodes in the Designer tree and click Associate User Data. The Associate User Data dialog box opens.

6. Type "FOID" in the Name text box.

7. Click the attribute you have just created from the Attribute drop-down list (OID in this example).
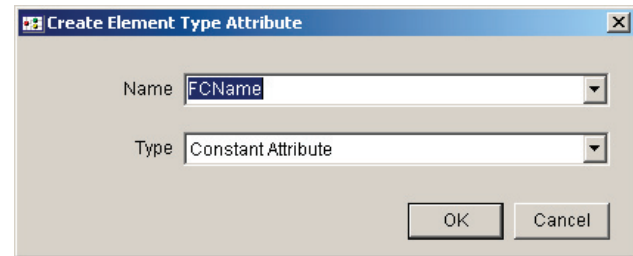

8. Click OK.
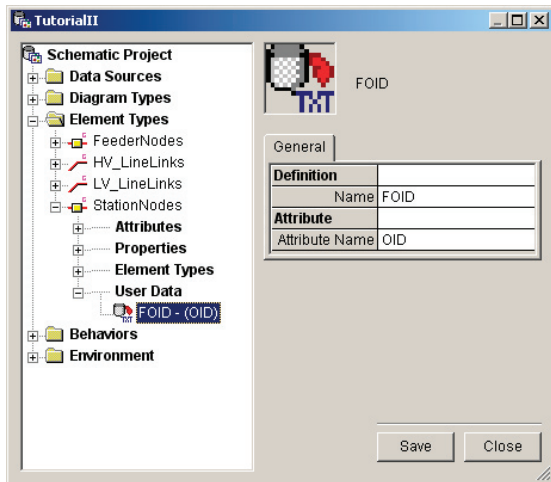
The FOID user data definition for the StationNodes elements is now finished.

Create the FCN user data that will specify the feature class name of the StationNodes elements. Since the StationNodes elements represent the features stored in the Station feature class table, creating a constant

attribute whose value is "Station" is the solution here.

9. Right-click StationNodes from the Designer tree and click Create Attribute. The Create Element Type Attribute dialog box opens.

10. Type a name for the new attribute ("FCName", for example).

11. Click Constant Attribute from the Type drop-down list.

12. Click OK.

A new entry corresponding to this attribute now appears in the Designer tree. You need to complete this attribute definition.

13. Type "Station" in the Value text box on the General tab and click OK.

Next, create the FCN user data you want to be associated with this attribute.

14. Right-click StationNodes from the Designer tree and point to Associate User Data. The Associate User Data dialog box opens.

15. Type "FCN" for the user data Name.

16. Choose the attribute name you have just created in the Attribute drop-down list (FCName, in this example) and click OK.
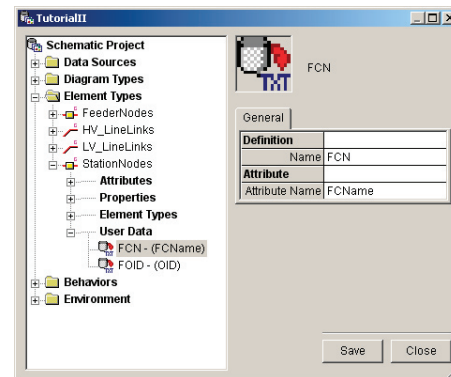
Test the impact of these definitions in ArcMap.

17. Click Save and exit ArcGIS Schematics Designer.

18. Start ArcMap with a new, empty map.

19. Click Add Data, browse to the Schematic_Designer_Tutorial database installed in C:\arcgis\ArcTutor\Schematics\Designer, click ElectricDataSet, then click Add. The Unknow Spatial Reference dialog box opens. Click OK to close it. Zoom into the set of features added in the map several times to correctly see them.

20. Click Open Schematic Diagrams. The Select schematic diagrams to open dialog box opens.

21. Browse to the TutorialII schematic dataset contained in the Schematic_Designer_Tutorial database and choose All.

22. Click Add.

23. Click View, point to Layout View, then arrange the geographic and schematic data frames.

24. Click the Select Schematic Elements tool and select any part of the network in the All diagram that contains StationNodes nodes.

25. Click Propagate Schematic Selection To Map.

The Station features related to the StationNodes schematic nodes currently selected in the All diagram are now selected in your map.
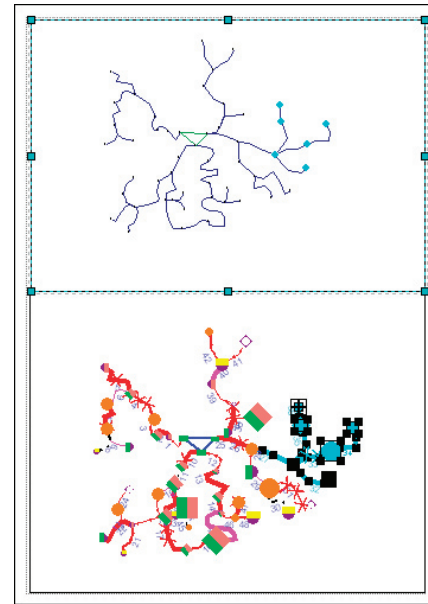
26. Clear the selection features set from your map.

27. Use the Select Features tool to create a new selection set of features in your map.

28. Click Propagate Map Selection To Schematic.

The StationNodes nodes associated with the Station features selected in your map are automatically selected in the All schematic diagram.

Note: The Propagate Schematic Selection To Map and Propagate Map Selection To Schematic commands work only for the schematic nodes and feature points contained in the Station feature class. For the other elements and features to also be automatically selected during the commands execution, you need to create the appropriate user data on the FeederNodes, LV_LineLinks, and HV_LineLinks element types.

29. Save your .mxd file and exit ArcMap.

30. Edit the TutorialII schematic project in ArcGIS Schematics Designer.

Creating the user data for the FeederNodes, HV_LineLinks, and LV_LineLinks element types

1. Repeat the previous steps 1 through 8 for each FeederNodes, HV_LineLinks, and LV_LineLinks element type to create the FOID user data. This user data must be associated with an attribute field link to the OBJECTID field returned by each element type query.

Please note that for the LV_LineLinks element type, an attribute that returns the OBJECTID field values already exist (In this example, it is LV_LineOID). In this case, to create the predefined FOID user data for LV_LineLinks, you only need to associate the FOID user data with the LV_LineOID attribute.

2. Repeat the previous steps 9 through 16 for the FeederNodes, HV_LineLinks, and LV_LineLinks element types to create the FCN user data. This user data will be associated with an attribute constant whose value is Feeder, HV_Line, or LV_Line, depending on the element type.

3. Click Save and exit ArcGIS Schematics Designer.

4. Load your .mxd file in ArcMap and test the Propagate Schematic Selection To Map and Propagate Map Selection To Schematic commands again.

For any schematic element selected in the All diagram, Propagate Schematic Selection To Map now automatically selects the associated feature in your map.

Conversely, for any feature selected in your map, using the Propagate Map Selection To Schematic selects the associated feature in the All diagram.

Since any element type associated with the diagrams implemented by the Feeder diagram type inherits from the StationNodes, FeederNodes, or LV_LineLinks element types, the Propagate Map Selection To Schematic and Propagate Schematic Selection To Map commands should also work on the Feeder diagrams.

Note: In this example, as each schematic element is associated with a unique feature, the FOID, FCN, and FDSN user data returns a unique value. For an example of schematic elements associated with several features/objects, you can edit the ElecDemo schematic dataset in the ElecDemo personal geodatabase within ArcGIS Schematics Designer (the ElecDemo database should have been installed in the ArcTutor\Schematics\Schematics folder). Look at the InsidePlant_RootNodes and InsidePlant_RootLinks parent element type's definitions. In this case, each schematic element that inherits from these element types is associated with both a feature stored in the ElectricNetwork feature dataset and with an object stored in the Inside_Nodes or Inside_Links object tables. Formatted attributes associated with the predefined FOID and FCN user data have been created to specify these multiple associations.

Note: Instead of creating the FCN user data on each parent element type, it is easier to specify the Associated Feature Class/Table field value for each parent and child element type on the General tab, that is, for StationNodes, FeederNodes, LV_LineLinks, StationNodes_F, FeederNodes_F, and LV_LineLinks_F element types.
Oneach General tab related to these element types, you need to do the following:
1. Click the Associated Feature Class/Table field and click the ellipsis button that appears. The Associated Feature Class/Table dialog box opens.

2. Specify the data source (Data Source field) and feature class or table name (Name field) you want to be associated with this element type.

In this case, only the FOID user data definition will be required.

In this exercise, you will learn about the parameters that can be specified to relate schematic elements contained in a schematic diagram to each other. Schematic elements can be related to each other by specifying rules or executing custom code during the diagram loading. For example, schematic nodes can be displayed as containers around schematic elements by specifying a Spatial Query rule or Relationship rule. Schematics can also automatically display nodes or drawings as containers around schematic element groups when two predefined user data values are specified for the elements of those groups—PEN and PTN:

- The PEN user data value must specify the name of the container that will be automatically related to the element.

- The PTN user data value indicates the name of the schematic element type that implements this container.

Note that only the PEN user data is mandatory for Schematics to manage the container relations. The PTN user data is required only when several schematic elements contained in the schematic diagram have the same identifier.

In this exercise, you will customize the All diagram display so each FeederNodes node and its related StationNodes nodes display in a specific container, one container per feeder. The OBJECTID field available from the Feeder feature class and the Feeder field stored in the Station feature class will be used to identify the container that will be related to each of them. Start by creating a new element type to implement the containers you want to be displayed in your All diagram.

## Creating the new FeederContainers drawing element type

1. Edit your schematic project within Designer.

2. Right-click Element Types in the Designer tree and click Create. The Create Element Type dialog box opens.

3. Type "FeederContainers" in the Name text box.

4. Click Drawing from the Type drop-down list.

5. Keep the Element Group option checked by default and click OK. The new drawing element type subentry is automatically created below the Element Types main entry in the Designer tree.

6. Click the General tab and click CURRENTDS from the Data Source drop-down list.

7. Click the Query field and click the ellipsis button. The Query Editor dialog box opens.

8. Double-click the Feeder table where all records corresponding to the containers you want to implement are stored.

9. Click OK.

10. Click the General tab's Identifier field and click the ellipsis button. The Identifier Editor dialog box opens.

As you have previously done for your element type identifier, each FeederContainers drawing will be identified by concatenating the diagram type name, the diagram name where it will be displayed, the FeederContainer string, and the OBJECTID field values.

11. Double-click the SCH_DiagramType and SCH_DiagramName items at the end of the Fields list; these two fields appear in the Identifier list.

12. Type "FeederContainer" in the central text box and validate by clicking the down arrow.

13. Double-click the OBJECTID field from the Fields list; this field is added to the Identifier list.

14. Click OK. The Identifier Editor dialog box closes.

15. Click the Symbol Name field and click the Select Symbol button. The Symbol Browser opens.

16. Click the ContainerLine symbol and click OK.

The General tab related to your new FeederContainers element type now appears as follows:
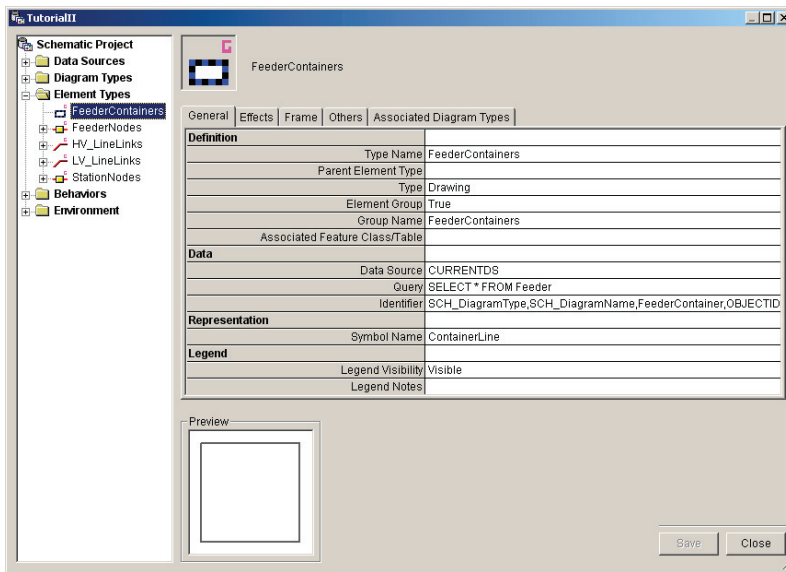
You must now associate this new element type with the All diagram type:

17. Right-click the FeederContainers element type entry in the Designer tree and click Edit Associations. The Diagram Type Associations dialog box opens.

18. Check All and click OK.

19. Click Save.

Creating the user data for the FeederNodes element type

Create the PEN user data that will specify the FeederContainer identifier for each FeederNodes node you want to be related to a container. You must build an attribute that will be associated with the PEN user data.

1. Right-click the FeederNodes element type entry in the Designer tree and click Create Attribute. The Create Element Type Attribute dialog box opens.

2. Type "ContainerName" in the Name text box.

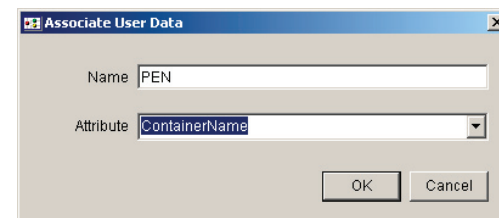3. Keep Field Attribute for the Type parameter and click OK.

4. Double-click the SCH_DiagramType and SCH_DiagramName items at the end of the Fields list on the Identifier Editor dialog box; these two items appear in the Identifier list.

5. Type "FeederContainer" in the central text box and validate this string by clicking the down arrow.

6. Double-click the OBJECTID field from the Fields list to select it.
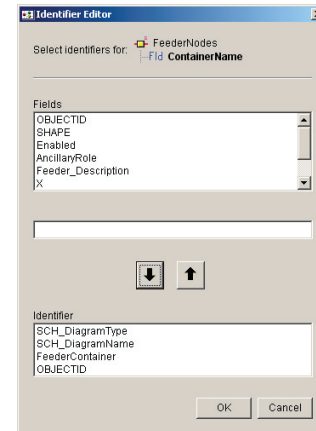
7. Click OK.

You must now associate this newly created attribute with the PEN user data.

8. Right-click the User Data entry under the FeederNodes entry in the Designer tree and click Associate.

9. Type "PEN" in the Name text box on the Associate User Data dialog box that opens.

10. Click ContainerName in the Attribute drop-down list.
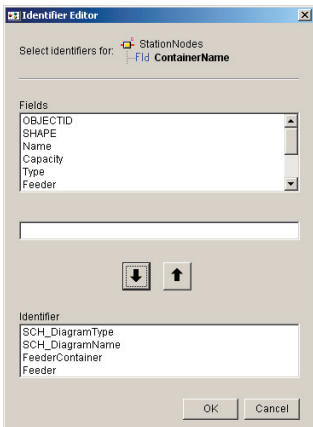

11. Click OK.

Creating the user data for the StationNodes element type

In the same way, create the same PEN user data for the StationNodes nodes.

1. Right-click the StationNodes element type entry and click Create Attribute. The Create Element Type Attribute dialog box appears.

2. Type "ContainerName" in the Name text box, keep Field Attribute for the Type parameter and click OK.

In the Station table, this is the Feeder field that provides the ObjectID of the Feeder related to each station.

3. Double-click SCH_DiagramType and SCH_DiagramName from the Identifier Editor dialog box, type "FeederContainer" in the central text box, then validate this string by clicking the down arrow.

4. Double-click the Feeder field from the Fields list.

5. Click OK.

Now you need to associate this ContainerName attribute with the PEN user data as you did previously for FeederNodes.

6. Right-click the User Data entry under the StationNodes entry in the Designer tree and click Associate.

7. Type "PEN" for the Name text box on the Associate User Data dialog box that opens.

8. Click ContainerName in the Attribute drop-down list and click OK.

9. Click Save.

The PEN user data definition for the StationNodes and FeederNodes element types is now complete. The definition of the second predefined PTN user data, required when the same identifer is used for several schematic elements contained in a single diagram, is not necessary in this case because each schematic element identifier is unique. If you switch to run mode and open the All diagram, it appears as follows:
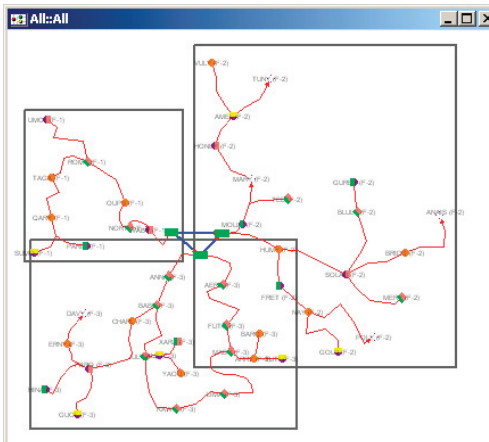
Exercise 8: Defining schematic behaviors

In this exercise, you will learn how to define schematic behaviors. A behavior defines how an application responds to a user's actions. Behaviors can be applied to schematic diagrams, elements, and views. Note that when schematic diagrams open in the Schematics Designer environment, behaviors can also be applied to legend entries. But when these diagrams display in the ArcMap environment as schematic layers, behaviors defined for legend entries cannot be intercepted.

Defining behaviors that will impact schematic diagrams

In this section, you will define a behavior impacting any schematic diagram (or view) when it is right-clicked. Generally, the main commands needed from a diagram are as follows:

- Fitting all the elements in the diagram
- Repositioning the elements displayed in the diagram according to their initial positions

You are going to create a behavior that will display a shortsutmenu for launching these three commands:

To create behaviors that impact diagrams, follow these steps:

1. Right-click the Behaviors entry in the Designer tree and click Create.

2. Choose RightClick from the Event Name drop-down list on the Create Behavior dialog box that opens and click Schematic Diagram from the Meta Type drop-down list.
   In this example, because the behavior you are defining will impact any diagram defined for the schematic project, keep the Object Type parameter empty.
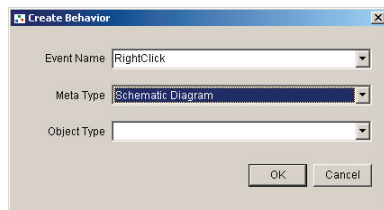

3. Click OK.

The new RightClick behavior entry is automatically created under the Diagram Types Behaviors main entry in the Designer tree.

You must now specify the command that will be automatically triggered each time a schematic diagram is right-clicked. The predefined command used to display a shortcut menu is CommandPopupMenu:

4. Click CommandPopupMenu from the Command drop-down list on the new behavior's General tab.
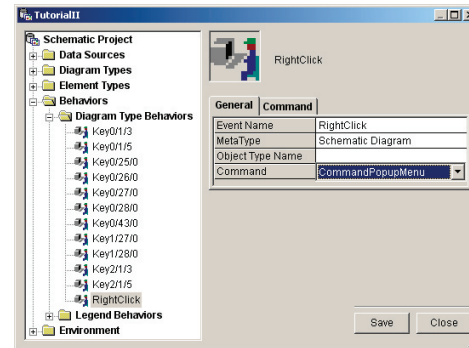

Define the CommandPopupMenu parameters. The Command tab related to a behavior definition always lists the parameters for the main command set in the General tab's Command parameter.

5. Click the Command tab.

In this example, the first menu command will be used to fit all the elements contained in the diagram.

6. Type a label for the first item of the shortcut menu in the Item1 field ("Full Extent", in this example) and press Enter.

7. Click CommandFitAll from the Command1 drop-down list.
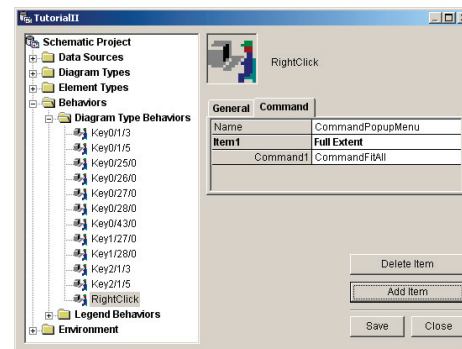
The first item definition is finished.

You must now define the second menu command that will activate the pan mode:

8. Click Add Item to create the second shortcut menu command.

9. Type the item label ("Initial Positions", in this example).

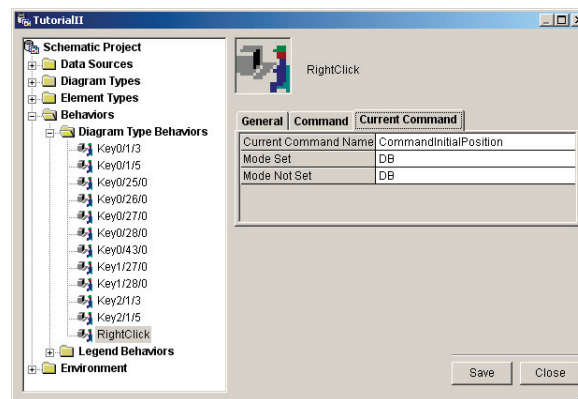10. Click CommandInitialPosition from the Command2 drop-down list and press Enter.

The first predefined command you have used for the previous menu command definition was a simple command without parameters. In this case, the CommandInitialPosition is a parameterized command.

11. Click the Command2 field and click the ellipsis button at the end of the command line.
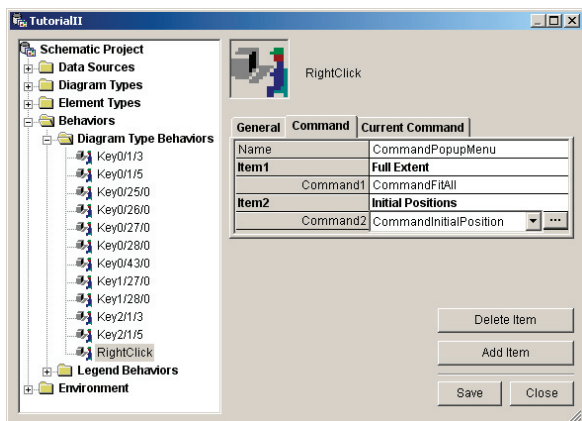

The Current Command tab appears where the parameters related to the CommandInitialPositions are displayed.

12. To reposition the elements in the diagram at their initial coordinates, choose DB from the Mode Set and Mode Not Set drop-down lists. The Current Command tab content appears as follows:

13. Click Save.

If you want to test this shortcut menu display, switch to run mode, open a diagram, and right-click this diagram background. The shortcut menu appears as follows:
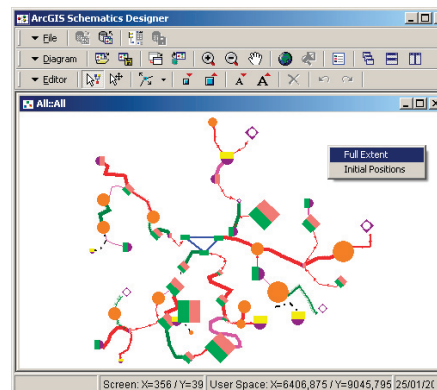
Note: For more details on the CommandInitialPosition command parameters, open the Schematics Designer Help system.

Defining behaviors that will impact schematic elements

In this section, you will create behaviors that will impact the StationNodes and LV_LineLinks element types when an element of this type is right-clicked.

For example, the shortcut menu displayed from any right-clicked StationNodes station will allow you to do the following:

- Rotate all stations by 45 degrees.
- Center the clicked node in the view.
- Activate or deactivate the Label2 property label display.
- Activate or deactivate the DisplayByCapacity property graphic effects.
- Activate or deactivate the DisplayByType property graphic effects.

From any right-clicked LV_LineLinks link, the shortcut menu displayed will allow you to do the following:

- Activate or deactivate the Label1 property label display.
- Activate or deactivate the DisplayByCategory property graphic effects.
- Activate or deactivate the DisplayByRate property graphic effects.

Note: Since StationNodes_F and LV_LineLinks_F element types inherit from StationNodes and LV_LineLinks element types, behaviors defined for StationNodes and LV_LineLinks element types will also impact StationNodes_F and LV_LineLinks_F element types.

Defining the sample behavior related to the StationNodes type

If the first, third, fourth, and fifth shortcut menu commands you have to define for the right-clicked StationNodes element type node will correspond to a single parameterized command, the second one that will automatically center the clicked node in the view must chain several commands. In fact, the predefined command used to center an element in a diagram (CommandCenterObject) works on a selected element only. Since right-clicking a node doesn't select it, you will have to select the node (using the CommandSelectObject predefined command) before centering it. But the predefined CommandContainer command usually used to chain several commands cannot be called directly from a shortcut menu command. In this case, the solution consists of defining a user event to trigger the command chaining that you will call from the shortcut menu command through a CommandDelegate command.

The following steps describe this user event behavior:

1. Right-click the Behaviors entry in the Designer tree and click Create. The Create Behavior dialog box opens.

2. Type a name ("Centering", for example) in the Event Name parameter that will be used to reference this behavior.

3. Click Schematic Element from the Meta Type drop-down list.
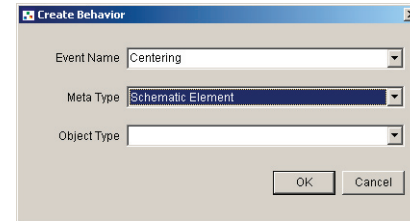

4. Click OK.

A new Element Type Behaviors entry appears under the Behaviors main entry in the Designer tree. The newly created Centering behavior appears under Element Type Behaviors.

5. Click CommandContainer from the Command drop-down list on the new behavior General tab.
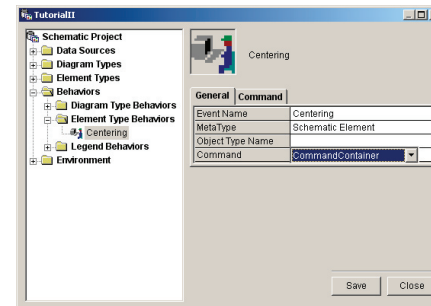
6. Click the Command tab, then the CommandContainer parameters must be specified.

7. Click CommandSelectElement from the Command1 drop-down list. This predefined command will be used to select the clicked element.

8. Click Add Command to create the second command that will chain with the first.

9. Click CommandCenterElement from the Command2 drop-down list. This predefined command will be used to center the clicked element.
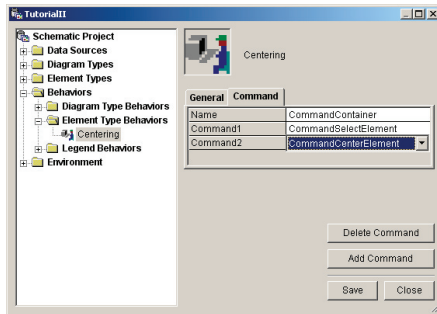

10. Click Save.

Note: This user event definition can end by calling the CommandUnselectAll command so the selection is automatically cancelled after it is centered in the diagram.

Because this user event is now defined, each command menu you want to be displayed when a StationNodes node is right-clicked can be easily defined and customized as follows:

1. Right-click the Behaviors entry in the Designer tree and click Create. The Create Behavior dialog box opens.

2. Choose RightClick from the Event Name drop-down list, click Schematic Element from the Meta Type drop-down list, then click StationNodes from the Object Type drop-down list.
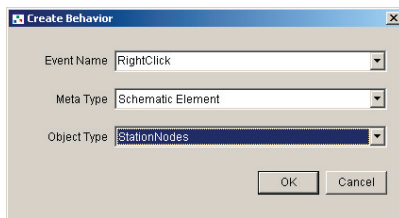

3. Click OK.

4. Click CommandPopupMenu from the Command drop-down list on the new behavior General tab.

5. Click the Command tab where parameters related to the CommandPopupMenu must be specified.

Start defining the command that will correspond to the first shortcut menu command. Clicking the first menu command must rotate all stations by 45 degrees:

6. Type the first item that will appear on the menu ("Rotate 45°", in this example).

7. Click CommandRotate from the Command1 drop-down list.
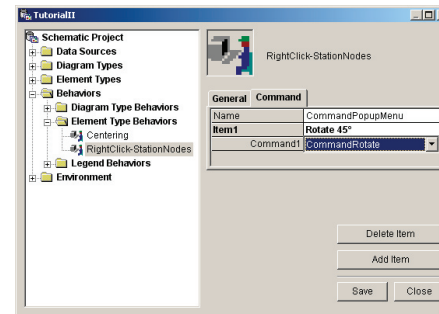
8. Click the ellipsis button on the right of the Command1 field. Its associated Current Command tab appears where the CommandRotate command parameters are available.

9. Click the Current Command tab and type the Parameter value that corresponds to the angle that will be used to rotate the nodes ("45", in this example).
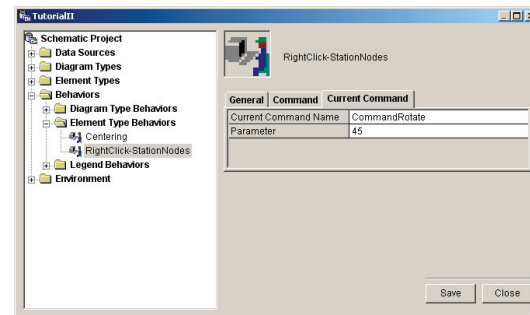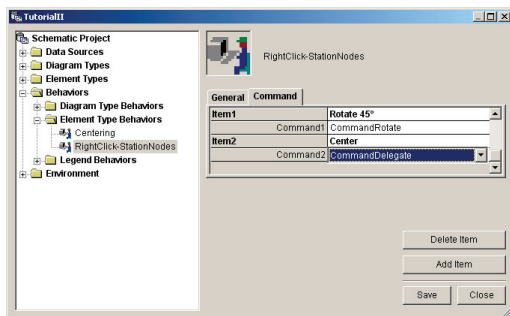

10. Click Save.

Define the second shortcut menu command. This new menu command will be used to center the clicked node in the diagram:
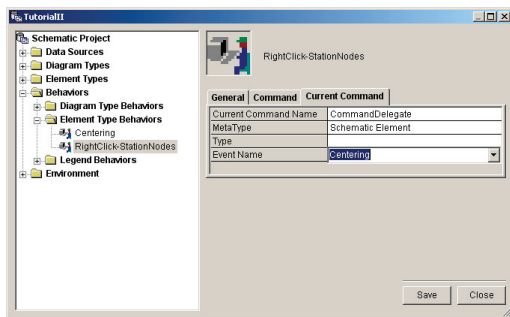
11. Click the Command tab and click Add Item.

12. Type the second item that will appear in the shortcut menu ("Center", in this example).

The command you specify here is not a simple, predefined command. You must launch the execution of the Centering behavior you previously defined. CommandDelegate is the predefined command you must use to trigger the execution of an existing behavior:

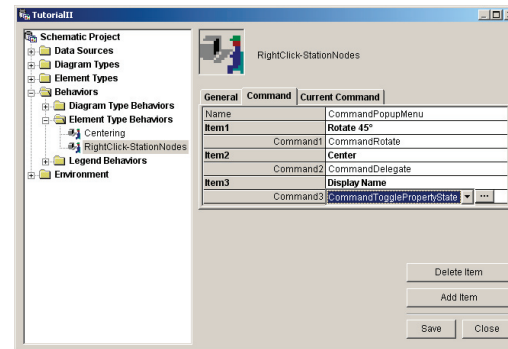13. Click CommandDelegate from the Command2 drop-down list.

14. Click the ellipsis button on the right of the Command2 field to display its associated Current Command tab. Click this tab.

15. The Meta Type default value is the correct one. Click Centering from the Event Name drop-down list.
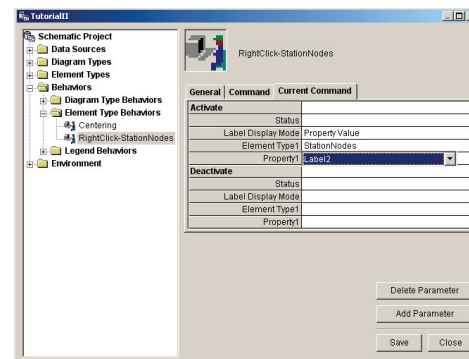

16. Click Save.

Define the third shortcut menu command. Clicking this new menu command must activate or deactivate the Label2 property label display.

17. Click the Command tab and click Add Item.

18. Type the third item that will appear on the shortcut menu ("Display Name", in this example).

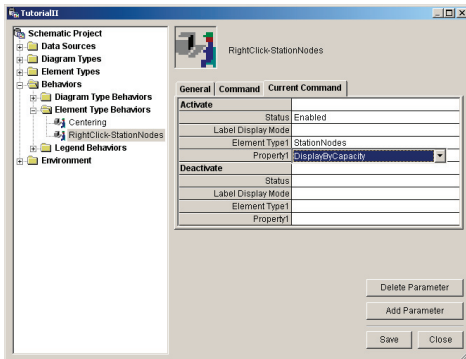19. Choose CommandTogglePropertyState from the Command3 drop-down list.

20. Click the ellipsis button at the right of the Command3 field to display its associated Current Command tab and click this tab.

21. Choose Property Value from the Label Display Mode drop-down list so that the property label display is turned on/off. Choose StationNodes from the Element Type1 drop-down list, and choose Label2 from the Property1 drop-down list.

22. Click Save.

Define the fourth shortcut menu command that will be used to activate or deactivate the DisplayByCapacity property's graphic effects.

23. Click the Command tab and click Add Item.

24. Type the fourth item label that will appear on the shortcut menu ("Display by Capacity", in this example).

25. Choose CommandTogglePropertyState from the Command4 drop-down list.

26. Click the ellipsis button at the right of the Command4 field to display its associated Current Command tab and click this tab.

27. Click Enabled from the Status drop-down list so the property graphic effects are turned on or off according to the current property status. Click StationNodes in the Element Type1 field and click the related property, DisplayByCapacity, from the Property1 drop-down list.

28. Click Save.

Define the last shortcut menu command that will be used to activate or deactivate the DisplayByType property's graphic effects.

29. Click the Command tab and click Add Item.

30. Type the item label that will appear on the shortcut menu ("Display by Type", in this example).

31. Click CommandTogglePropertyState from the Command5 drop-down list.

32. Click the ellipsis button at the right of the Command5 field to display its associated Current Command tab and click this tab.

33. Click Enabled from the Status drop-down list so the property graphic effects are turned on or off according to the current property status. Click StationNodes in the Element Type1 field and click the related property, DisplayByType, from the Property1 field.
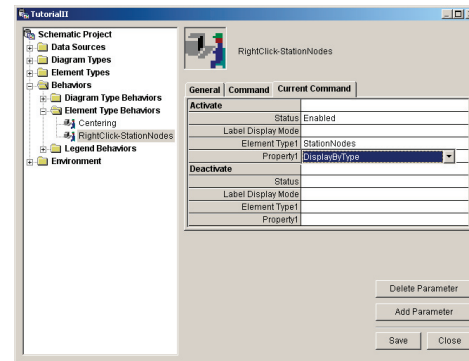
34. Click Save.

To test this new shortcut menu display, switch to run mode, open a diagram, then right-click any StationNodes schematic node contained in it. The context menu appears as in the following screen shot.

Note: Because this behavior is defined for the StationNodes element type, it is also available from the inherited StationNodes_F element type; that is, this shortcut menu also displays when right-clicking a StationNodes_F node in any diagram implemented by the Feeder diagram type.
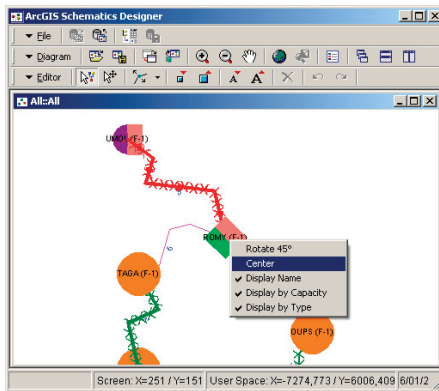
Note: For the last three menu commands, the item is checked when the related property's effects are activated and unchecked when they are deactivated. Clicking one of these three shortcut menus when they are checked deactivates the related property, while clicking one of these three shortcut menu commands when they are unchecked activates the related property.

Defining the sample behavior related to the LV_LineLinks type

As you did for the StationNodes element type, you must now create a new behavior to display a shortcut menu from any right-clicked LV_Lines link.

1. Right-click the Element Type Behaviors entry in the Designer tree and click Create.

2. Choose RightClick from the Event Name drop-down list on the Create Behavior dialog box.
   Because the Create Behavior dialog box has been opened from the Element Type Behaviors entry, Schematic Element should already be selected for Meta Type.

3. Click LV_LineLinks from the Object Type drop-down list.

4. Click OK.

5. Click CommandPopupMenu from the Command drop-down list on the newly created behavior's General tab.
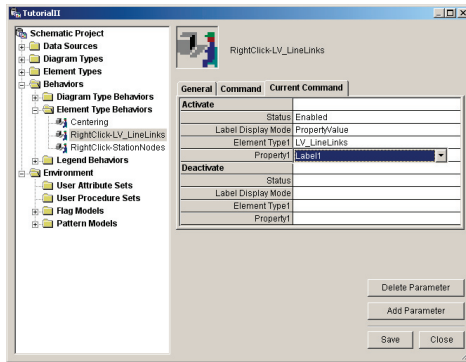
6. Click the Command tab.

The first menu command must activate or deactivate the Label1 property label display:

7. Type the Item1 value ("Display Information", in this example).

8. Click CommandTogglePropertyState from the Command1 drop-down list.
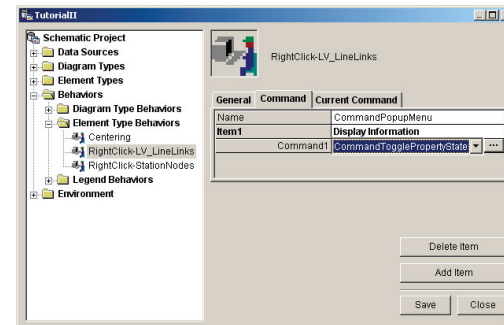
9. Click the ellipsis button at the right of the Command1 field to display its associated Current Command tab and click this tab.

10. Click PropertyValue from the Label Display Mode drop-down list so the property label display is turned on or off.

11. Click LV_LineLinks from the Element Type1 drop-down list.

12. Click Label1 in the Property1 drop-down list.

13. Click Save.

You must now define the second shortcut menu command that will be used to activate or deactivate the DisplayByCategory property's graphic effects.

14. Click the Command tab and click Add Item.

15. Type the second item that will appear on the shortcut menu ("Display by Category" in this example).

16. Click CommandTogglePropertyState from the Command2 drop-down list.

17. Click the ellipsis button at the right of the Command2 field to display its associated Current Command tab and click this tab.

18. Click Enabled from the Status drop-down list; click LV_LineLinks in the Element Type1 drop-down list; then click the related property, DisplayByCategory, for the Property1 drop-down list.
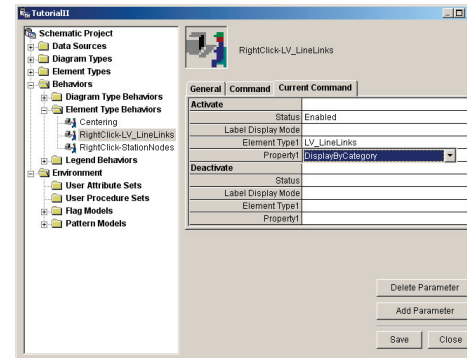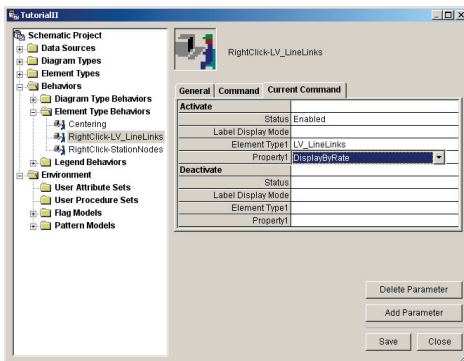

19. Click Save the specified parameters.

Create the last menu item command that must activate or deactivate the DisplayByRate property's graphic effects.

20. Repeat steps 14 through 16 so the Command tab content appears as follows:
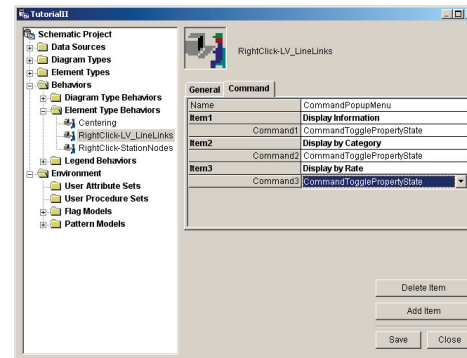
21. Repeat steps 17 and 18 for the DisplayByRate property. The Current Command tab appears as follows:

22. Click Save.

If you switch to run mode and test this new shortcut menu display, it should appear as follows: