Working with GeoMedia® Transportation Manager

INTERGRAPH

Working with GeoMedia Transportation Manager

DJA084580

(6.0.1)

Intergraph Corporation
Huntsville, Alabama 35894-0001

What do you think about *Working with GeoMedia Transportation Manager*?

DJA084580

We would like your opinion of this document. Please help us improve our documentation by completing this questionnaire. Write your comments or circle your responses to the questions. Circle all responses that apply, and include additional pages of comments if you wish. When you have completed the questionnaire, fold and mail it to Intergraph. Postage is prepaid. Thank you for taking the time to let us know what you think.

1. How much of this document have you worked through? — 1. 1 - 2 - 3 - 4 - 5
(1=entire document, 2=most of it, 3=about half, 4=about one-third, 5=one-fourth or less)

2. How easy was this document to read and understand? — 2. 1 - 2 - 3 - 4 - 5
(1=difficult . . . 5=easy)

3. How easy was it to find specific information in this document? — 3. 1 - 2 - 3 - 4 - 5
(1=difficult . . . 5=easy)

4. What was your primary intent in using this document? — 4. 1 - 2 - 3 - 4 - 5
(1=orientation . . . 5=thorough understanding)

5. How well did this document satisfy this intent? — 5. 1 - 2 - 3 - 4 - 5
(1=poorly . . . 5=very well)

6. Have you used Intergraph software before? (1=yes, 2=no) — 6. 1 - 2

7. What percentage of your work time is spent using Intergraph software? — 7. 1 - 2 - 3 - 4 - 5
(1=10% or less, 2=25%, 3=50%, 4=75%, 5=90% or more)

8. How would you rate your level of software use? — 8. 1 - 2 - 3 - 4 - 5
(1=demanding . . . 5=casual)

9. Did this document help you learn to use the software in a satisfactory way? — 9. 1 - 2
(1=yes, 2=no)

10. How would you rate this document overall? (1=poor . . . 5=excellent) — 10. 1 - 2 - 3 - 4 - 5

11. Have you used Intergraph documentation before? (1=yes, 2=no) — 11. 1 - 2

12. Please describe and note page numbers of any errors or inconsistencies in this document.

____

____

13. How would you improve this document?

____

____

Name ____________________ Date __________

Organization ____________________ Phone __________

Address ______________________________

City ________________ State ______ Zip Code ________

Thanks again for your time and effort.

INTERGRAPH

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 9079 HUNTSVILLE, AL

POSTAGE WILL BE PAID BY ADDRESSEE

Security, Government & Infrastructure (SG&I) Division
Manager, Technical Documentation
Mail Stop IW17B2
P.O. Box 6695
Huntsville, AL 35824-6695

Table of Contents

1 – Start Here

What's New in Transportation Manager 6.0.1

GeoMedia Transportation Manager 6.0.1 is a re-release of GeoMedia Transportation Manager 5.2.1 (5.2a) on the GeoMedia Professional 6.0.1 platform.

The **Dynamic Segmentation** and the **LRS Precision Location** commands, two commands that are very useful when running GeoMedia Transportation Manager, are now located on the GeoMedia Professional menu bar under **Analysis** and **Tools**, respectively:

To access Dynamic Segmentation: **Analysis > Dynamic Segmentation**

To access LRS Precision Location: **Tools > LRS Precision Location**

Overview of the Document

This document contains information on using GeoMedia Transportation Manager for linear data analysis and routing analysis. The structure of this document is as follows:

- There are two major divisions: Chapters 2-19 pertain to Linear Referencing Systems (LRS), and Chapters 20-37 pertain to Network Routing.
- Chapter 2 provides an introduction to the basic concepts behind linear referencing.
- Chapters 3-6 provide step-by-step instructions on how to use each of the LRS (Linear Referencing System) Maintenance commands.
- Chapters 7-13 provide step-by-step instructions on how to use each of the LRS Analysis commands.
- Chapters 14-16 provide step-by-step instructions on how to use four of the elements common to many of the LRS commands included in this product: the **LRS Properties** dialog box, the **Markers Properties** dialog box, the **Event Properties** dialog box, and the **Coordinate System** dialog box.
- Chapters 17-19 provide instruction on common workflows used in the analysis of linear data.

- Chapter 20 provides an introduction to the basic concepts behind network routing.
- Chapters 21-28 provide step-by-step instructions on how to use each of the Routing Maintenance commands.
- Chapters 29-33 provide step-by-step instructions on how to use each of the Routing Analysis commands.
- Chapters 34-35 provide step-by-step instructions on how to use two of the elements common to many of the Routing commands included in this product: the **Configure Network** dialog box and the **Stop Properties** control.
- Chapters 36-37 provide instruction on common workflows used in Routing analyses.
- Appendix B provides details of the LRS data structures supported by the GeoMedia Transportation product line.

Each section of this document takes you through a systematic process to use the software commands.

Documents Shipped

The following online documents are shipped with GeoMedia Transportation Manager:

Document	Number	Description
Working with GeoMedia Transportation Manager	DJA0845	An overview of the workflows and commands for performing most software tasks. Available online in *.pdf* format at **Start > All Programs > GeoMedia Transportation Manager > User Documentation > Working with GeoMedia Transportation Manager.**
Installing GeoMedia Transportation Manager	DJA0853	Instructions for installing the product. Available online in *.pdf* format at **Start > All Programs > GeoMedia Transportation Manager > User Documentation > Installing GeoMedia Transportation Manager.**
GeoTrans Transportation Data Model		A suggested transportation data model for use with the GeoMedia Transportation product line. Available online in *.pdf* format at **Start > All Programs > GeoMedia Transportation Manager > User Documentation > GeoTrans Data Model.**

GeoMedia Transportation Manager Object Reference		Programmer's guide for objects, methods, and properties. Available from the *\Program Files\GeoMedia\Help* folder or at **Start > All Programs > GeoMedia Transportation Manager > Developer Documentation > GeoMedia Transportation Manager Object Reference.**
GeoMedia Transportation Object Diagrams		Object diagrams for programmers. Available online in *.pdf* format at **Start > All Programs > GeoMedia Transportation Manager > Developer Documentation > GeoMedia Transportation Object Diagrams.**
GeoMedia Transportation Workflow Diagrams		Workflow diagrams for programmers. Available online in *.pdf* format at **Start > All Programs > GeoMedia Transportation Manager > Developer Documentation > GeoMedia Transportation Workflow Diagrams.**

Visit our web site at *http://support.intergraph.com/Documentation.asp* for the latest version of these documents.

Document Conventions

Typeface Conventions Used in the Documents

ALL CAPS — Keyboard keys.

If keys are separated by a comma, press them in sequence. For example: ALT, F5. If they are joined by a plus sign, press them at the same time.

For example: CTRL+`z.`

Bold unserifed type — An item in the graphical interface, such as the title of a dialog box or a tool. Paths through menus use right angle brackets between items you select.

For example: Select **File > Open** to load a new file.

`Courier type` — Information you type.

For example: Type `original.dat` to load the ASCII file.

Italic type — A document title, the first occurrence of a new or special term, folder and filenames, or information about what the software is doing.

2 – Introduction to Linear Referencing

The purpose of this chapter is to outline the basic concepts behind the linear referencing capabilities of GeoMedia Transportation Manager. Each of the major components of a Linear Referencing System (LRS) is described, and table descriptions are provided. Lastly, the various LRS analysis and maintenance tools are described.

What is Linear Referencing?

Linear Referencing is simply the tracking and analysis of data that is associated with locations along a linear network. Some road transportation examples include tracking the location of and condition of signage, the condition of pavement, and the location and severity of accident occurrences. The biggest uses of linear referencing are Asset Tracking and Asset Analysis.

Asset Tracking primarily encompasses the following four items:

- What, where, and when of the assets (for example, a pothole at kilometer post 41.7 along Route 66, reported June 6th, 2002)
- Asset conditions (for example, a stretch of pavement with rutting and cracking)
- Incidents along the network (for example, a traffic accident)
- Activities along the network (for example, construction projects)

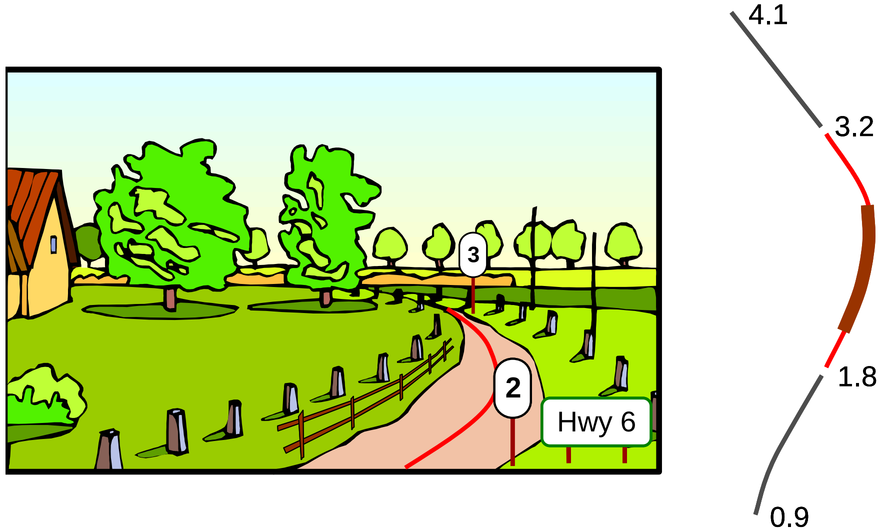
Asset Analysis includes such activities as “hot-spotting” (finding areas with an unusual density of a given type of problem) and cross-discipline analysis (for example, cross analyzing areas that have both run-off-the-road accidents and no guardrails). This type of analysis can be important for any number of areas, including the following:

- Protecting the public (for example, finding areas of high-density accidents and finding the common contributing factor)
- Optimizing usage of the assets (for example, locating areas with both high volume and low number of lanes in order to identify areas of congestion)
- Optimizing budget usage (for example, locating the areas with the most traffic and the worst pavement conditions to be first priority for resurfacing)

Linear Referencing and GeoSpatial Technology

The main impetus to merge linear referencing with geospatial technology can be summed up simply: it's often desirable to view location data on a map. It also opens up a lot of other analysis capabilities, such as summing up data within an area feature (for example, the kilometers of rail track that require maintenance within a given jurisdiction) or finding data within a proximity of linearly referenced data (for example, finding residences within a buffer zone of a construction project).

Using GeoMedia Transportation Manager is not the only way to merge linear referencing and geospatial technology, but it is certainly the easiest. GeoMedia Transportation Manager enables you to create map features, including pavement conditions, accident data, and average daily traffic. This kind of information will help you plan improvements for deteriorating assets, will identify where your organization is spending its money, and will provide critical information clearly and accurately to all participants involved in your projects. GeoMedia Transportation Manager increases the value of your data by turning it into business-critical, decision-support information.



The preceding diagram shows a portion of road on the left and its geospatial representation on the right. The road has kilometer posts that indicate cumulative linear measures along the road. It also has a road name, Highway 6 in this example. A section of fencing along the road is also shown in both the left and right views. Based on the kilometer posts, it can be determined in the field that this stretch of fence runs along Highway 6 from kilometer measure 2.0 to 2.6.

On the geospatial side we have three linear features, known as LRS Linear Features, that will all have a road name and begin and end measure attribution. These LRS Linear Features are the backbone of the LRS and are used in automating the mapping of linearly referenced data, such as this stretch of fencing, onto our map view.

Of course, this mapping of linearly referenced data does not have to be automated. Without GeoMedia Transportation Manager, you can estimate where kilometer measures 2.0 and 2.6 are along the road, and then you can digitize a linear feature between these two points and along the road. This is not too hard for a few features, but what if you have a tabular report for hundreds or thousands of linearly referenced items that you want to map? With GeoMedia Transportation Manager, all of these items can be mapped with a single command.

The methodology used by GeoMedia Transportation Manager to do this bulk mapping of linearly referenced tabular data is called *Dynamic Segmentation* (or linear geocoding). This methodology interpolates the location of linearly referenced data along the LRS Linear Features by making use of the road (or rail, ferry line, and so on) name and measurement attributes stored on those features.

LRS Linear Features and Event Data

As mentioned before, *LRS Linear Features* are the backbone of the LRS. But working with the linearly referenced tabular data, known as *Event Data*, is the whole reason for building the LRS in the first place. Although the data structures of these two components are described in detail in the GeoMedia Transportation LRS Data Structures Appendix, we will provide a brief summary here.

The LRS Linear Features represent the network itself. Each LRS Linear Feature table is a linear feature class that has the following fields:

- ID – This is a long integer value that uniquely identifies each feature within the table.
- LRSKeys1-4 – This is one to four fields that together define the "route" that this feature belongs to.
- StartMeasure – This is a numeric value that contains the measurement value for the *beginning* of this feature.
- EndMeasure or Duration – This is a numeric value that contains either the measurement value for the *end* of this feature or the *length* measurement for this feature.
- BeginMarker (optional) – This field stores a name for the *beginning* position of this feature. This is referred to as an *internal marker*.
- EndMarker (optional) – This field stores a name for the *end* position of this feature. This is referred to as an *internal marker*.
- ReversedGeometry (optional) – This Boolean (True/False) field declares whether the GeoMedia Transportation Manager software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end, and vice-versa (True).

The Event Data represents the linearly referenced data. Note that Event Data can either be *point* data (occurring at just one spot on the linear network) or *linear* data (occurring at a span of distance along the linear network). Each Event Data table is (usually) a non-graphic table that has the following fields:

- ID – This is a long integer value that uniquely identifies each record within the table.
- LRSKeys1-4 – This is one to four fields that together define the "route" that this record lies along.
- Measurement data (pick one of the following options)

- Measure Option – For point event data, this consists of one numeric Measure field that indicates the relative location of the point event record on the "route" defined by the LRS Key fields. For linear event data, this consists of two numeric fields: a StartMeasure field and an EndMeasure field. These define the relative location of the start and end points of the linear event record on the "route" defined by the LRS Key fields.
- Marker Offset Option – For point event data, this consists of a Marker name field and a numeric Offset distance field. The point event data is located by first locating the marker and then by adding the offset distance to that location. For linear event data, there are two Marker fields and two Offset fields defining the start and end of the linear event record.
- Coordinate Option – For point event data, this consists of two fields which, depending on the referenced Coordinate System File, may by either projected coordinates (for example, Northing & Easting) or geographic coordinates (Latitude & Longitude). For linear event data, there are four fields defining the coordinates for both the start and the end of the linear event record.
- Duration Option – This is a slight variation on the Measurement Option and only applies to linear event data. It consists of a StartMeasure field and a Duration (or Length) field that together define the relative location of the record along its "route."

• Other Attributes (optional) – These are optional, but they are also the whole reason for doing linear referencing. For bridge events, these will store bridge data; for accident events, they will store accident data; and for pavement events, they will store pavement data.

Optional components of the LRS that we have not discussed as yet are the *External Markers*. External Markers mark points along the network just like the Internal Markers discussed earlier, but these are not bound to just the beginning and end of LRS Linear Features. External Markers can occur anywhere along the LRS network and are functionally equivalent to point-event data using the Measure option. They are useful for modeling milestones and monuments that are commonly used to measure locations from. They can be used, along with Internal Markers, to locate event data using the Marker Offset option.

Each External Marker table is (usually) a non-graphic table that has the following fields:

• ID – This is a long integer value that uniquely identifies each record within the table.

- LRSKeys1-4 – This is one to four fields that together define the “route” that this record lies along.
- Measure – This is a numeric field that indicates the relative location of the External Marker on the “route” defined by the LRS Key fields.
- MarkerName – This field stores a name for this Marker.

Detailed information on how to create a LRS Linear Feature class is provided in the “LRS Data Preparation Workflows” chapter. Information on at least one way to populate Event Data and External Marker tables is provided in the “Working with the Insert LRS Event Command” chapter.

Linear Referencing Analysis

This section provides brief descriptions of the major LRS Analysis tools provided by GeoMedia Transportation Manager. Detailed instructions on how to use each of these commands is provided elsewhere.

Dynamic Segmentation – This command, which has already been referred to above, takes linearly referenced tabular data and creates a graphic query class from it that can be viewed in the map view. This lets you visualize your organization's inventory of assets more clearly than by simply reviewing tabular data. For more information see the “Linear Referencing” chapter in *Working with GeoMedia* or *Working with GeoMedia Professional*.

Note: The **Dynamic Segmentation** command is now located on the GeoMedia Professional menu bar under **Analysis.**

LRS Event Overlay – This command compares two different Event Data tables and calculates intersections, differences, or unions of the two tables and presents the results as a graphic query class that can be viewed in the map window. An example of this is to map all the locations with run-off-the road accidents and no guard rails. For more information see the “Working with the LRS Event Overlay Command” chapter.

Resolve LRS Event Overlaps – This command takes a single linear Event Data table and provides a very user-definable method for removing overlaps in the data. For example, in a table of pavement conditions you may have many overlapping records, but you may want to only see the most current pavement condition records. For more information see the “Working with the Resolve LRS Event Overlaps Command” chapter.

LRS Precision Location – This command allows you to get real-time LRS locations of your cursor location in the map view. It also allows you to use key-ins of LRS locations to place points in the map view. These points may just be used for orientation, but they also can be used for placing vertices of new geometry. For more information see the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with GeoMedia Professional*.

Note: The **LRS Precision Location** command is now located on the GeoMedia Professional menu bar under **Tools.**

LRS Event Conversion – This command allows you to convert the measurement portion of your event data from one measurement option to another (these options were described earlier). This can allow you, among other things, to convert event data collected in a number of ways to a single measurement method for more consistent reporting. For more information see the "Working with the LRS Event Conversion Command" chapter.

LRS Event Generation – This command provides you with a number of ways to generate event data at various defined points along the LRS network. This comes in handy for both annotation and aggregation workflows. For more information see the "Working with the LRS Event Generation Command," "LRS Annotation Workflows," and "LRS Analysis Workflows" chapters.

LRS Keys for Coordinate Events – This command will take Event data that has coordinate data but no LRS Key data and add the LRS Key data to it. This will make this kind of Event Data, commonly collected using GPS instruments, usable with the other analysis commands, such as Dynamic Segmentation and LRS Event Overlay. For more information see the "Working with the LRS Keys for Coordinate Events Command" chapter.

Routes and Sections to LRS – This command takes the two component classes from an ArcRoute system (Routes and Sections) and creates a single query of an LRS class that is usable with the other commands within GeoMedia Transportation Manager. For more information see the "Working with the Routes and Sections to LRS Command" chapter.

Insert LRS Event – This command allows you to interactively create event data by clicking on the map view to define LRS key, begin measures, and (for linear events) end measures. This will allow you to create event data via "heads-up" digitizing. For more information see the "Working with the Insert LRS Event Command" chapter.

Linear Referencing System Maintenance

This section provides brief descriptions of the major LRS Maintenance tools provided by GeoMedia Transportation Manager. Detailed instructions on how to use each of these commands is provided elsewhere. A good overview of LRS maintenance is provided in the "LRS Data Preparation Workflows" chapter.

Interactive LRS Calibration – This command can be used to name and calibrate (that is, populate measure values in) LRS features one route at a time. For more information see the "Working with the Interactive LRS Calibration Command" chapter.

LRS Calibration – This command can be used to calibrate (that is, populate measure values in) LRS features an entire feature class at a time. For more information see the "Working with the LRS Calibration Command" chapter.

LRS Validation – This command can be used to validate that a linear feature class is also a working LRS feature class that will provide accurate results. It queues up errors for easy correction. For more information see the "Working with the LRS Validation Command" chapter.

Reformat Linear Collections – This command can be used to repair some of the problems common to data sets containing geometry collections. It assures that components of geometry collections are in order and have correct and consistent digitizing directions. This helps assure an LRS that provides correct analysis results. For more information see the "Working with the Reformatting Linear Collections Command" chapter.

3 – Working with the Interactive LRS Calibration Command

The **Interactive LRS Calibration** command can be used to write key and measure values to a group of linear features making up a route. This is what is known as *calibrating* the LRS. The **Interactive LRS Calibration** command calibrates the LRS one route at a time. The **LRS Calibration** command calibrates an entire feature class at a time but assumes that the feature class already has valid LRS Key values (see the "Working with the LRS Calibration Command" chapter).

The **Interactive LRS Calibration** command is designed to handle several scenarios. If you have, for example, digitized raw linework representing a network of roads, then you can use the command to define routes through the network and to write key and measure values to the features making up the routes. By adding the key and measure values to the raw linework, the feature class becomes an LRS feature class. A second example is to use this command to re-calculate measures for a portion of a route. For example, construction eliminates a sharp bend in a portion of a route, so the features from the construction zone to the end of the route need to have their measures adjusted.

Defining a route to be calibrated is done by selecting linear features at the beginning and end of the route (or a portion thereof) in the map window. The software makes use of a shortest path algorithm to automatically select, in the proper sequence, the features in between the two selected features. This allows an entire route to be selected, often with just two mouse clicks. This is particularly handy for routes that have some segments that are so small that they are hard to select manually and are easy to accidentally overlook. This method requires that the linear feature class to be calibrated must be displayed in a map window. If the feature class is not displayed, the command will add the feature class to the legend and display it.

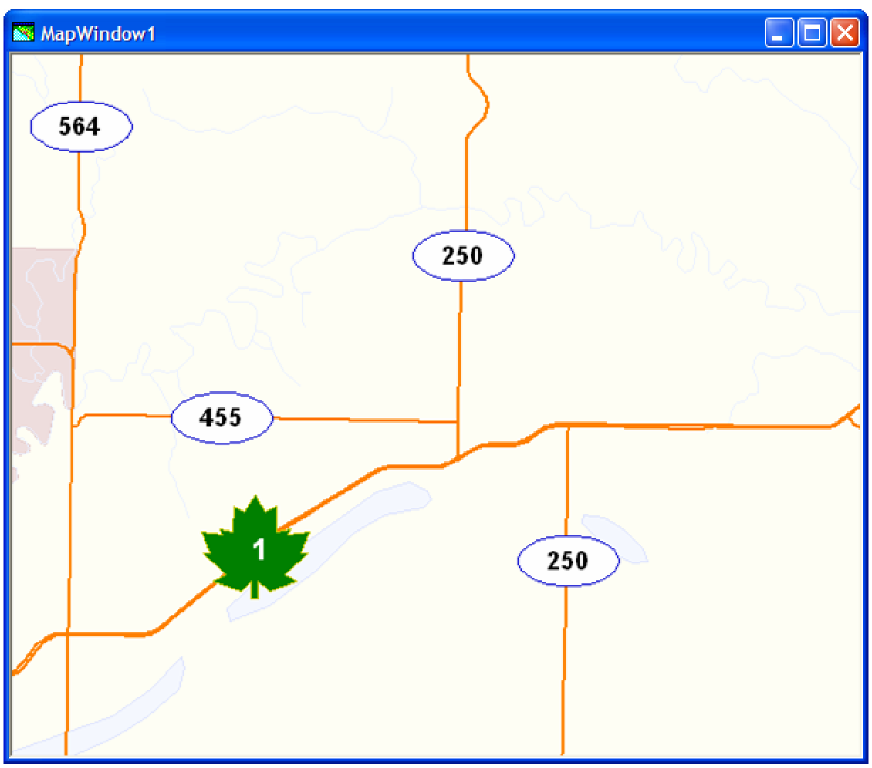
The command can fill in NULL values or overwrite existing values. The fields for the keys and measures must exist in the feature class before running the command, that is, the command will not create new fields. The command cannot be used to create new features or split existing features.

It also automatically sets the direction of the features within a route to match the overall direction of the route. Depending on the setting in the **LRS Properties** dialog box, this will either be done by setting the digitizing direction of each feature or by setting the value of a Geometry

Reversed field to **True** or **False** as needed. This second method is often useful when working with queries that might have geometry that is not editable.

The measurement calculations can be performed in three ways. The first is to use the geometry lengths as the basis for measurement calculations. The second is to use length field values on the segments themselves (if the value is NULL for a given segment, the geometry length will be used). The third way is to override any calculated measures and to key in measure values manually.

An important capability of this command is that it allows some features to be treated as gaps within the route, and it allows you to either use or not use the length of the gap features when calculating measures for the features making up the route. The gap segments themselves are not edited. A common usage of this is where a minor road meets a major road and then continues on some distance down the major road, like Route 250 in the image below. If desired, the length of road along the major route can be added to the measure of the minor route when it starts up again on the other side of the major route.



The Interactive LRS Calibration Command Workflow

This section presents the procedural steps for using the **Interactive LRS Calibration** command.

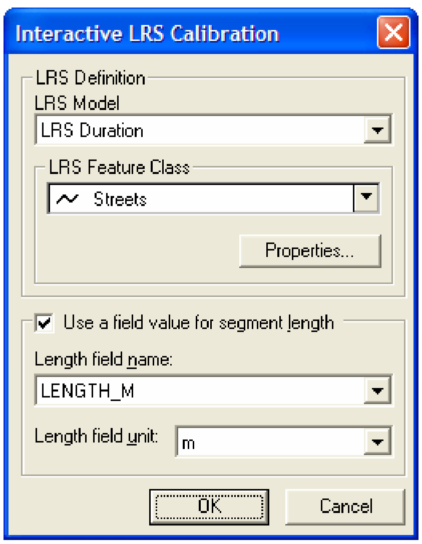
1. Open a GeoWorkspace that contains one or more read/write connections.

Note: Due to the dynamic nature of queries in GeoMedia, it is recommended that this command be run in a GeoWorkspace with a minimal number of active queries that are based on your LRS network. That is because you may pay a substantial performance penalty when running these commands while waiting for these queries to update.

---

2. Select **Transportation > LRS Maintenance > Interactive LRS Calibration**.

The **Interactive LRS Calibration** *dialog box is displayed.*



3. Select an LRS model from the **LRS Model** drop-down list. For more information on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.

Note: The measure values calculated and written to each feature making up a route will be **Begin** and **End** measures for the **LRS Measure** model type, and **Begin** and **Duration** for the **LRS Duration** model type.

---

4. Select an LRS feature class or query from the **LRS Feature Class** drop-down list.

Note: If a query is used, it must have writable LRS key and measure fields. It also must have either editable geometry or have a writable **Geometry Reversed** field. This last part is so that the command can correct the digitizing direction if necessary.

---

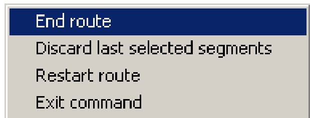
5. Click **Properties** to define metadata for the LRS feature class, including LRS key field names, begin and end measure field names, and measure units.

   The **LRS Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for more information on this dialog box.

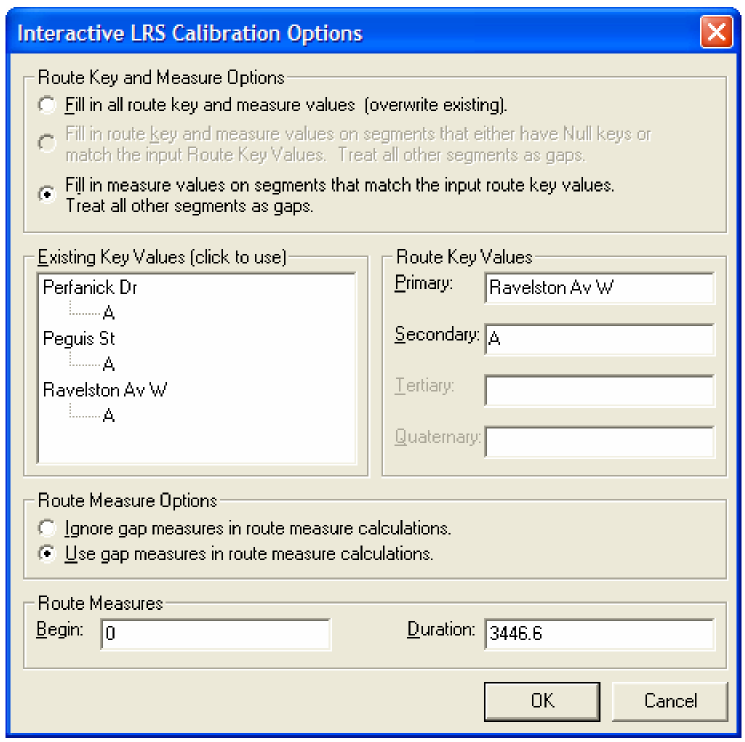
---

6. Optional: Click **Use a field value for segment length** to designate a field containing a length value, which will be used instead of geometry lengths in the calibration calculations.
7. If you checked **Use a field value for segment length**, select a **Length field name** and a **Length field unit** from the two drop-down lists.
8. Click **OK**.
9. Follow the on-screen prompts to define a route by selecting features in the map window. First you will be prompted to **"locate the first segment of the route, right click to exit."** After you select the first segment, you will be prompted to **"locate the next segment of the route, right click to exit."** Repeat this until you have selected all the segments necessary to outline the desired route. You need not select every segment because the shortest path algorithm will select the intermediate segments. As you select each subsequent segment, the intermediate segments will be added to the highlighted selection.
10. Select **End route** from the pop-up menu (right mouse click) to display the **Interactive LRS Calibration Options** dialog box.



This pop-up menu allows you to (1) select **End route** to indicate that the route definition is complete and to display the **LRS Calibration Options** dialog box, (2) to select **Discard last selected segments** to discard the most recently selected segments and to continue defining the route, (3) to select **Restart route** to discard the entire route and to start over, or (4) to **Exit command**.

When you select **End route**, *the* **Interactive LRS Calibration Options** *dialog box is displayed.*



11. Select the appropriate **Route Key and Measure Options.**

    Fill in all route key and measure values (overwrite existing) is the default option, and it is the only option available if the key values for all features in the route are Null. Selecting this option will update the key and measure fields for all features in the route.

    Fill in route key and measure values only on segments that either have Null keys or match the Input Route Key Values treats segments with keys that both do not match the selected route and are not Null as gaps. This option is available when some features making up the route have existing key values and some features have Null key values. When you select this option, only the features that match the selected route or that have Null key values will be updated with new key values and measures.

Fill in measure values on segments that match the input route key values treats all other segments as gaps. This option is available when at least some of the features in the route have existing key values and there is more than one set of existing key values. Select this option only when updating the measures for one of the existing sets of key values.

12. Define the **Route Key Values** either by typing in or by clicking on one of the existing keyset values in the tree view control. The number of key fields defined in the **LRS Properties** dialog box determines the number of boxes enabled.

    Existing key values are displayed in a tree view, with the primary key as the parent, the secondary key as the first child, and so on. The tree view is disabled if all key values for all features in the route are Null. Clicking on a parent, or any children of a parent, writes the key values into the **Primary, Secondary**, and so on, text boxes in the **Route Key Values** portion of the dialog box.

13. Select the appropriate **Route Measure Options**.

    Ignore gap measures in route measure calculations is the default option. In this case only segments that are not considered to be gaps are used when calculating measures for the features making up the route.

    Use gap measures in route measure calculations uses the length of the all segments, including gaps, when calculating the measures. If measurement attributes are available on the gap segments, the lengths defined by these attributes will be used; otherwise, the geometric length of gap segments will be used.

14. Define the **Route Measures** for both the **Begin** and **End** of the selected portion of the route, or the **Begin** measure and **Duration** length in the case of a Duration model. The default value for the **Begin** field will be the begin measure attribute of the first segment selected. If that value is Null, the default **Begin** field value will be 0. The default **End** field value will equal the default **Begin** field value plus the sum of the geometric lengths of all segments to be calibrated. If the **Use gap measures in route measure calculations** option is used, the default End field value will equal that value plus the lengths of the gaps. The gap lengths will be defined by measurement attributes if available; otherwise, it will be based on geometric lengths.

15. Click **OK** to write Key and Measure values to the appropriate fields and records of the feature class. This will also set the direction of all segments to match the direction of the route. If a **Geometry Reversed** field was selected in the **LRS Properties** dialog box, then this will be accomplished by setting the value of this field rather than reversing the geometry.
16. Follow the prompts to define another route, or click **Exit command** from the pop-up menu to exit the command.

4 – Working with the LRS Calibration Command

The **LRS Calibration** command can be used to write measure values to a group of linear features making up a route. This is what is known as *calibrating* the LRS. Linear feature classes that contain LRS Key and measure values can be defined as a Linear Referencing System, and can be used in analysis processes such as dynamic segmentation.

The **LRS Calibration** command calibrates an entire feature class at a time and makes use of the assumption that the feature class already has valid LRS Key values that define which features are grouped together as *routes*. On the other hand, the **Interactive LRS Calibration** command calibrates the LRS one route at a time and does not require that the LRS Key values be filled in ahead of time (see the "Working with the Interactive LRS Calibration Command" chapter).

The **LRS Calibration** command is designed to handle the scenario where you have a network of streets that are named (via key values), but which lack measures. You can then use this command to calculate and write measure values and correctly set feature directions to all of the features making up the street network. Because this command works on both linear feature classes and queries, you can also use it to calibrate or recalibrate portions of a network defined by a query.

The measurement calculations can be performed in two ways. The first is to use the geometry lengths as the basis for measurement calculations. The second is to use length field values on the segments themselves (if the value is Null for a given segment, the geometry length will be used).

Another influence on the way measures are calculated is the way gaps are handled. These gaps occur between portions of a route that are geographically disconnected. This usually occurs when one route intersects another route, follows along the other route for some distance, and then separates again. You are given three choices as to how these gaps influence measure values. The first method is to use the map distance across the gap. The begin measure of the first segment after the gap is equal to the end measure of the last segment before the gap plus the straight line distance across the gap. The second method uses a constant, typed in, distance for all gaps. The third method is to find the shortest path along the network across the gap. This method assumes that other members of the selected feature class, but with different key values, define a continuous path across the gap. In this case, the shortest path distance across the gap is determined and is used to calculate the measures.

An important capability of this command is that it automatically sets the direction of the individual features within a route to match the overall direction of the route. Depending on the setting in the **LRS Properties** dialog box, this will either be done by setting the digitizing direction of each feature or by setting the value of a **Geometry Reversed** field to **True** or **False** as needed. This second method is often useful when working with queries that might have geometry that is not editable.

The overall direction of the route can be determined by either of two methods: the digitized direction of the segments making up the route or by the compass direction of the route. To determine the route direction by the digitized direction, the lengths of all segments digitized in one direction are added, the lengths of all segments digitized in the opposite direction are added, and the route direction will be determined to be in the direction of greatest length. To determine route direction by compass direction, an imaginary line is drawn between the two end points of the route. The trend of this line, in the local coordinate system, will be either east-west or north-south. On rare occasions, when the trend is exactly between north-south and east-west, then northeast and southwest are grouped as north-south trends, and northwest and southeast will be grouped as east-west trends. You can select a **South to North** or **North to South** option, and a **West to East** or **East to West** option. Based on the user selection and the trend of the line, route direction and the begin and end of the route can be determined.

The LRS Calibration Command Workflow

This section presents the procedural steps for using the **LRS Calibration** command.

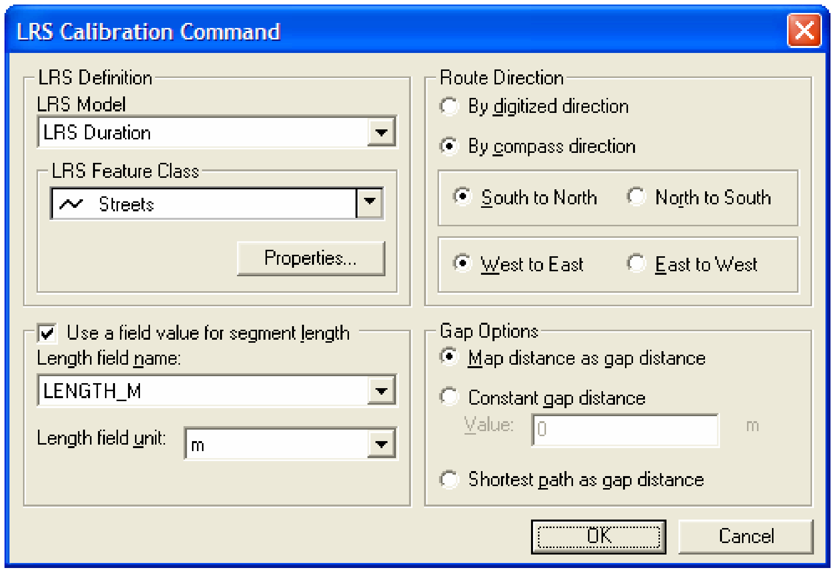
1. Open a GeoWorkspace that contains one or more read/write connections.

Note: Due to the dynamic nature of queries in GeoMedia, it is recommended that this command be run in a GeoWorkspace with a minimal number of active queries that are based on your LRS network. That is because you may pay a substantial performance penalty when running these commands while waiting for these queries to update.

---

2. Select **Transportation > LRS Maintenance > LRS Calibration**.

The **LRS Calibration Command** *dialog box is displayed.*



3. Select an LRS model from the **LRS Model** drop-down list. For more information on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.

Note: The measure values calculated and written to each feature making up a route will be **Begin** and **End** measures for the **LRS Measure** model type, and **Begin** and **Duration** for the **LRS Duration** model type.

---

4. Select an LRS feature class or query from the **LRS Feature Class** drop-down list.

Note: If a query is used it must have writable measure fields. It also must have either editable geometry or have a writable **Geometry Reversed** field. This last part is so that the command can correct the digitizing direction if necessary.

---

5. Click **Properties** to define metadata for the LRS feature class, including LRS key field names, begin and end measure field names, and measure units.

   The **LRS Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for more information on this dialog box.

---

6. Click **OK**.
7. If desired, click **Use a field value for segment length** to designate a field on the input feature class or query containing a length value, which will be used instead of geometry lengths in the calibration calculations.
8. If you checked **Use a field value for segment length**, select a **Length field name** and a **Length field unit** from the two drop-down lists.
9. Select one of the **Route Direction** options. Select **By digitized direction** if the digitized directions of the segments making up the route are used as the criteria for determining the overall direction of the route. In the case of a route that has conflicting digitizing directions, the direction that constitutes the majority of the length of that route will be used. Select **By Compass Direction** to use general compass directions as the criteria for determining the overall direction of the route. The direction of the route will determine the direction in which measures will increase.
10. If you chose the **By Compass Direction** option, select **South to North** or **North to South** and select **West to East** or **East to West** to determine how the command will orient the routes and in which direction the measures of the routes will increase.
11. Select one of the options for processing measures across gaps. If you select **Map Distance as Gap Distance**, the calibration continues across the gap, with the begin measure equal to the end measure of the previous segment plus the straight-line distance across the gap. Select **Constant Gap Distance** to specify a constant distance to be used in all gap calculations (key in a value in the **Value** field). Select **Shortest path across gap** if the gap is spanned by features with different key values (different routes). The command will use a shortest path algorithm to calculate the length of the gap and will add that distance to the end measure of the last feature before the gap to calculate the begin measure of the first feature after the gap.

Note: The **Shortest path across gap** option gives the most accurate results, but at a cost of extra processing time.

---

12. Click **OK** to kick off the calibration process, or **Cancel** to exit the command. The calibration process can take a fair amount of time to process so a status bar is provided to keep you informed if its progress. All measure fields within the input feature class or query will be updated, and the direction of all individual features will be set to match the direction of the route it belongs to. If a **Geometry Reversed** field was selected in the **LRS Properties** dialog box, then this will be accomplished by setting the value of this field rather than reversing the geometry.

5 – Working with the LRS Validation Command

This section provides an overview of the **LRS Validation** command, which identifies anomalies in the feature classes to be used as the linear referenced network. The LRS may be made up of a single linear feature class that contains all the information necessary for defining the LRS, or it may also contain a point feature class that is used as Markers or as Nodes. Anomalies that are found by the **LRS Validation** command can be corrected using standard GeoMedia Professional placement and editing commands. Later in this chapter we present a general step-by-step workflow that you can apply to your data.

The **LRS Validation** command is a preparation step that should be used before linear analysis is performed with an LRS analysis command such as the **Dynamic Segmentation** command or the **Event Overlay** command. This ensures that the proper results are returned by these analysis functions. The tasks outlined in this chapter are part of an overall data preparation workflow, which is described in the "LRS Data Preparation Workflows" chapter.

The following checks are performed by the **LRS Validation** command, depending on the LRS model type you select:

- *Missing LRS keys*—Looks for records in the input *LRS feature class* that are missing the necessary LRS keys. An LRS feature class is a feature class that contains linear geometry; fields that define the routes (LRS Keys); and fields with either begin and end measures, or begin measure and duration. It can optionally contain begin and end marker fields.
- *Missing Begin or End measures*—Looks for records in the input LRS feature class that are missing begin or end (duration) values. For duration models, zero or negative duration values are also flagged.
- *Missing node IDs*—For nodes models, looks for records in the input LRS feature class that are missing the begin or end node ID values.
- *Begin distance >= End distance*—Looks for records in the input LRS feature class that have a begin distance greater than or equal to the end distance.
- *Overlapping distances*—Looks for records that have the same LRS key values and that contain overlapping distances between the end

measure of one record and the begin measure of the next record in the input LRS feature class.

- *Linear Measure/Geometry ratio*—Calculates a ratio of the user-defined linear measure (end distance minus begin distance) to the measured map distance (length) for each record of the LRS feature class. If the calculated ratio is outside the range of the minimum and maximum values entered on the dialog box, an anomaly will be created for that record.
- *Geometry sequence/direction problems*—Determines if the digitized direction is consistent for the records in the input LRS feature class and checks for sequencing problems and gaps between geometries in a collection. If the LRS Measure or Duration with Node model is selected, the correct digitized direction will be determined through the direction defined by the Begin Node to End Node as this relates to the actual location of the referenced nodes. In addition, nodes that are not at the ends of the LRS feature are flagged as anomalies.
- *Gaps between geometries*—Looks for gaps between geometries in the input feature classes based on a user-specified tolerance and for gaps between geometries in records that have the same LRS key values. While gaps between geometries of a route are valid, gaps less than the minimum tolerance are flagged as possible digitizing errors.

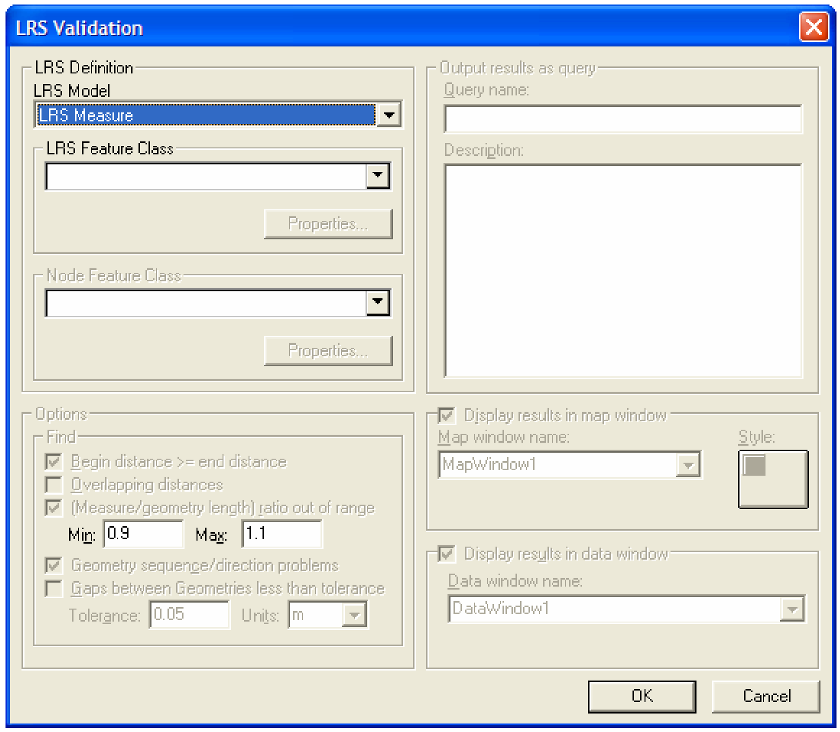
You can output the anomalies as a query to a map window and/or a data window for evaluation and correction.

The LRS Validation Command Workflow

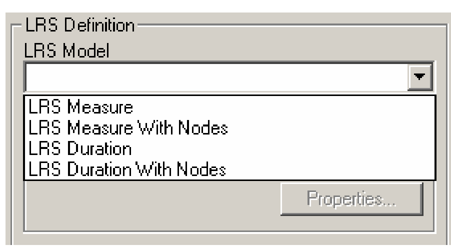
This section presents the procedural steps for using the **LRS Validation command.**

1. Open the GeoWorkspace connected to the data source containing the linear referenced network.
2. Select **Transportation > LRS Maintenance > LRS Validation**.

The **LRS Validation** *dialog box is displayed.*



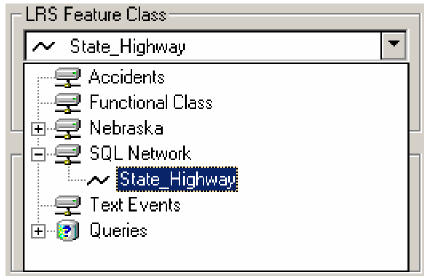
3. Click the **LRS Model** drop-down list; then select the model to be used. This is the one command that is part of GeoMedia Transportation Manager that explicitly supports a link-node structure. If you want to have the command analyze a link-node structure, pick one of the two node models. For more on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.



You are returned to the **LRS Validation** *dialog box.*

4. Click the **LRS Feature Class** drop-down list; then select the connection the LRS feature class resides in.

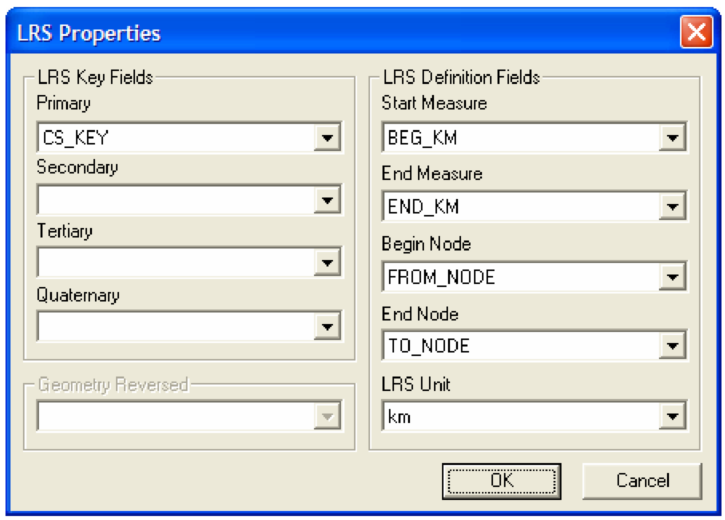
5. Select the feature to be used as the LRS feature.



You are returned to the **LRS Validation** *dialog box.*

6. In the **LRS Feature Class** portion of the dialog box, click **Properties**.

   The **LRS Properties** *dialog box is displayed.*



7. In the **LRS Key Fields** portion, select **Primary**, **Secondary**, **Tertiary**, and **Quaternary** keys of the LRS feature class for as many keys as you use.

8. In the **LRS Definition Fields** box, select the field names of the **Start Measure** and **End Measure** from the drop-down lists.

9. If you selected one of the node model types, select the field names of the **Begin Node** and **End Node** from the drop-down lists.

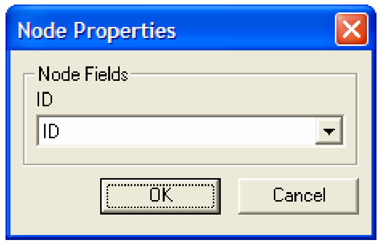
10. In the **LRS Unit** box, select the field name of the unit for the LRS feature class.

11. Click **OK** after setting the appropriate values.

    The **LRS Validation** *dialog box is returned.*

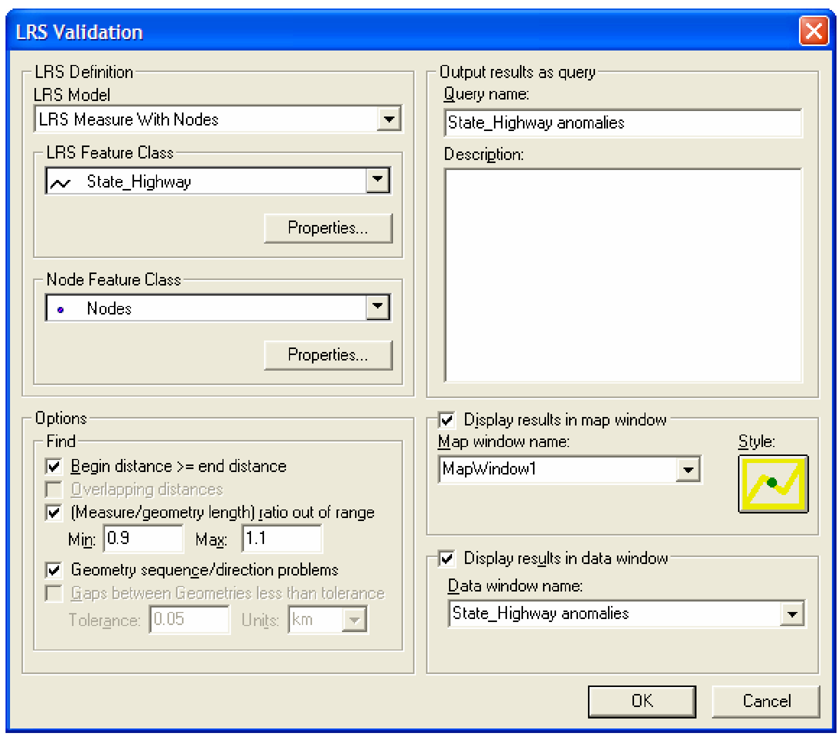
12. In the **Node Feature Class** portion of the dialog box, click **Properties**.

The **Node Properties** *dialog box is displayed.*



13. In the **ID** field, select the **ID** for the Node feature class from the drop-down list.
14. Click **OK** after setting the appropriate values.

    The **LRS Validation** *dialog box is returned.*



15. From the **Options** portion of the dialog box, select and/or deselect the appropriate validation checks to be performed.
16. In the **Output results as query** portion, type the **Query name** or accept the default.

    The results are saved as a query that can be edited and redisplayed in the workspace.

17. If you want to see the results in a map window, make sure that the check box to the left of the **Display results in map window** field is checked on and that the appropriate map window name is selected.

    The **Map window name** *field is enabled. The anomalies will be added to the legend and map window of your choice. You can choose an existing map window or create a new one. It is typically best to select an existing map window. In addition, you edit the line styles in order that the anomalies will be visible in the map window.*

    Note: The query is output to the map window as not locatable. The original LRS model feature class should be edited to correct each anomaly.

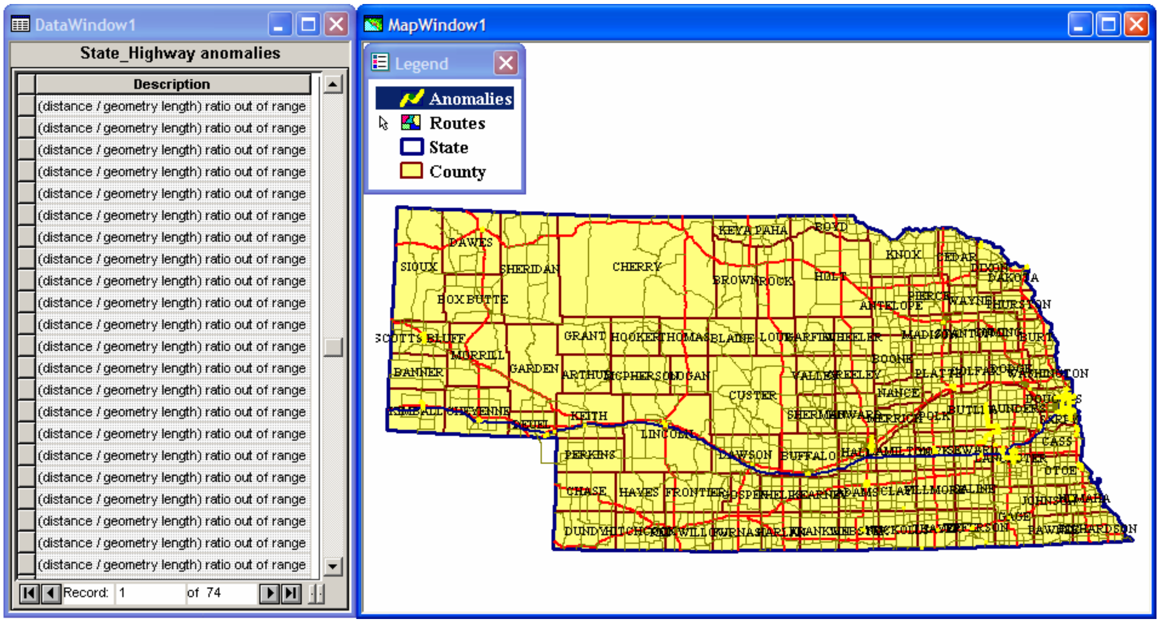
    ---

18. *Optional*: Click **Style** to define the display settings for the results in the map window.
19. If you want to see the results in a data window, make sure that the check box to the left of the **Display results in data window** field is checked on and that the appropriate **Data window name** is selected.

    The **Data window name** *field is enabled. The anomalies will be added to a data window of your choice. You can choose an existing data window or create a new one. It is typically best to create a new one.*

20. When you have set the appropriate values, click **OK**.
21. When the results are returned, you can tile the windows vertically.

    The GeoWorkspace would then resemble the following:



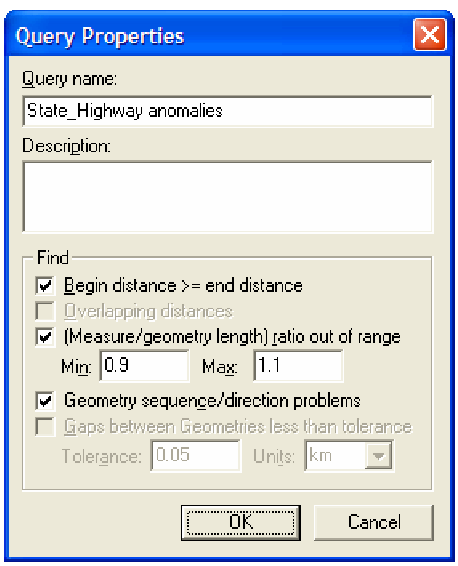
22. Correct any anomalies by using the GeoMedia Professional placement and editing commands on the LRS feature class and the Node/Marker feature class, if used.

    As mistakes in the input feature classes are corrected, the associated anomalies will disappear from the map and data windows. Once all of the anomalies are corrected, the data window will be empty, and the legend entry in the map window for the anomaly query will empty.

    If you perform any new feature additions or edits on the input feature classes while the query is still active, the new and/or edited features will be checked for anomalies. If anomalies are found, entries will appear in the anomaly query and in any data window or map window that contains the anomaly query.

Note: Selecting a record in the data window will highlight the corresponding entry in the map window so that you can see where the problem occurred. Use the **Fit Select Set** command, and **Zoom In** to the area containing the anomaly, or use the **Map Window Properties** command to set automatic view manipulation options as features are selected. The primary key value of the LRS feature that has the error is included in the data window for quick reference. Double click on the LRS features in the map window to determine which one is in error, or you may prefer to display a data window for the LRS feature class. The geometry and attributes of the LRS feature, along with the textual description of the problem, should allow you to quickly determine the appropriate fix to make. Remember that the anomaly is there just to show you what and where the error is. You do not edit the anomaly. You must edit the input LRS and/or Marker feature classes to correct the problem.

Remember, not all anomalies need to be edited. Some things are flagged as errors based on parameter settings and may indicate possible anomalies. The (distance/geometry length) ratio and gaps between geometries less than the tolerance are examples of anomalies flagged that you may determine to be valid conditions. Use the **Query Properties** dialog box in GeoMedia to change the validation parameters.



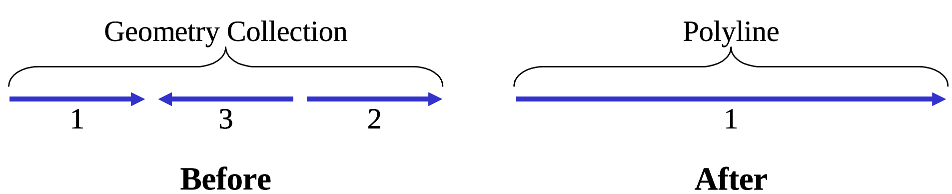
Note (continued): There is no single workflow for correcting anomalies. Try different options, settings, and workflows until you determine which is most productive for you and your data. However, it is recommended that you begin the corrections at the beginning measure of the LRS and work toward the end of the LRS. It is also recommended that you correct any gaps in the LRS before validating geometry sequences.

6 – Working with the Reformat Linear Collections Command

The **Reformat Linear Collections** command allows you to repair some of the problems common to data sets containing *geometry collections*. A geometry collection is a grouping of multiple geometries that are associated with a single record in the dataset. If left unrepaired, these problems can cause GeoMedia Transportation Manager to provide incorrect results.

It is common for CAD or MGE/MGSM data as served to GeoMedia via data servers to contain linear collections that are out of order or have incorrect/inconsistent digitizing directions. The **Reformat Linear Collections** command processes one linear collection at a time from the selected feature class or query in order to correct the above mentioned problems.

The processing of linear collections rearranges the vertices contained in the geometry. Linear collections that are connected end to end will be converted to polylines; whereas linear collections that contain gaps will remain as collections. The updated geometry is then written back to the database.



As the process continues, the progress bar and status bar are updated to indicate the number of records processed. At the end of command execution, a message box is displayed to show information that consists of the number of records processed, the total number of records, the number of linear collections processed, the total number of linear collections, and the name and location of the log file, if generated.

On the command dialog box, you can either specify the location and name of the log file or accept the defaults. A log file is generated only if errors were encountered. Each error message reported to the log file is appended by the primary key name and its value.

You can interrupt processing of the command at any time by pressing ESC. Also, you can undo the command results with the GeoMedia Professional **Undo** command.

Since the **Reformat Linear Collections** command processes one feature at a time, it does not keep track of the direction of routes that consist of multiple features. The command will rearrange the vertices of a feature to be in the correct sequence and to be going in the right direction, but that direction could be wrong with regard to other features making up the route. Such features will be flagged as anomalies by the **LRS Validation** command as it compares direction within a route. You can use the **Reverse Direction** command under the **Edit > Geometry** menu from GeoMedia Professional to fix direction problems on the flagged features.

If the input geometry has kickbacks or duplicate points, the **Reformat Linear Collections** command may not be successful in correcting the out-of-order or digitizing problems. You can use the **Validate Geometry** and **Fix Geometry** commands under the **Tools** menu from GeoMedia Professional to fix the kickbacks and duplicate points before using **Reformat Linear Collections**. For more on a workflow for preparing LRS data, see the "LRS Data Preparation Workflows" chapter.

Note: This command is provided primarily for LRS data that is imported from MicroStation design files. It may convert some linear collections to polylines.

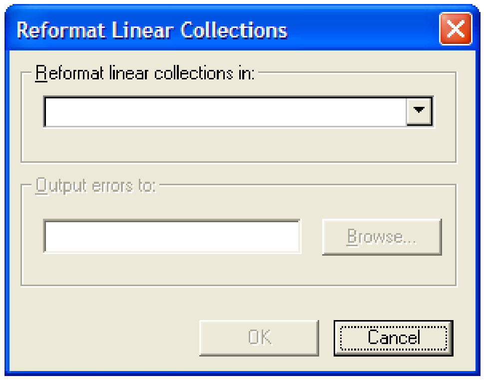
The Reformat Linear Collections Command Workflow

This section presents the procedural steps for using the **Reformat Linear Collections** command.

1. Open a GeoWorkspace that contains one or more read/write connections.
2. Select **Transportation > LRS Maintenance > Reformat Linear Collections**.

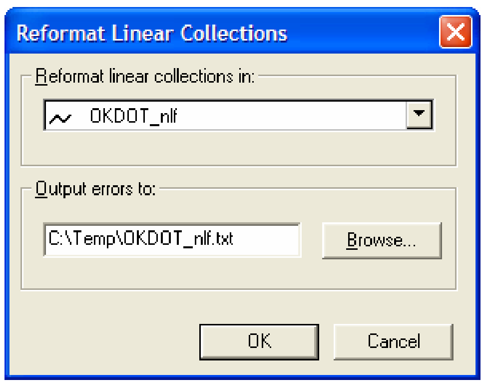
   Note: You must have an open read/write connection to use this command.

   The **Reformat Linear Collections** *dialog box is displayed.*



3. From the **Reformat linear collections in** drop-down list, select a linear feature class or query to use as input to the command.

   The **Output Errors to** *box is enabled.*



4. In the **Output errors to** box, accept the default, or type in or browse to change the default name or location of the log file generated to report the processing errors.
5. Click **OK** to update the linear geometries in the selected feature class.

7 – Working with the LRS Event Overlay Command

This section provides an overview of the **LRS Event Overlay** command. *Event overlay* is the process of overlaying two event sets, which could be tables or subsets of tables or queries, in order to find the intersection, union, or difference of the event records. The **LRS Event Overlay** command provides the capability to overlay two *event feature classes*.

This command is very useful for performing cross-discipline analysis to locate areas that may require attention. For example, you can locate areas where there is both heavy traffic and few lanes to identify possible congestions areas, or you could identify areas with run-off-the-road accidents and no guardrails.

There are three basic linear overlay operators supported: **Intersection**, **Union**, and **Difference**. There are also two versions of each of these three operators: normal and **Without Overlaps**. Lastly, this command works on both point and linear events in any combination (a total of four combinations). The three operators, with two versions each, working on four different event type combinations brings the total number of different behaviors you can get out of this one command to 24. Each one of these behaviors is explained in the next section.

Note: Versions 5.1a (05.01.01.06) and earlier did not have normal and **Without Overlaps** options. The **Intersection**, **Difference**, and **Union** options in those versions functioned in the same way as the **Intersection Without Overlaps**, **Difference Without Overlaps**, and the **Union Without Overlaps** options function in the current version.

You should perform the validation and correction of anomalies in the linear feature class before using the **Event Overlay** command. This ensures that the proper analysis results are returned

Behavior and Results

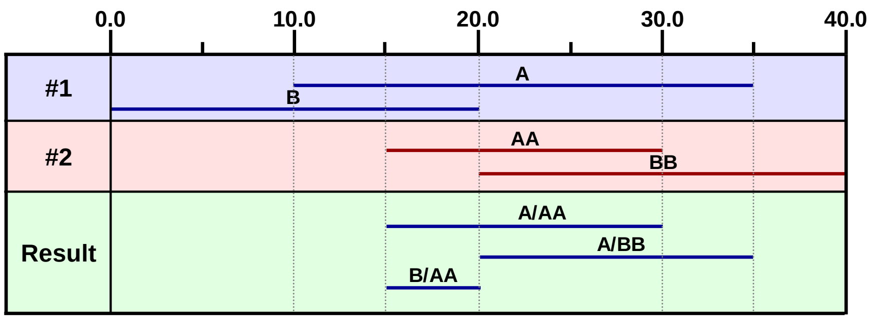
The following is a description of the behavior and results from each of the **LRS Event Overlay** operators.

Event Intersection

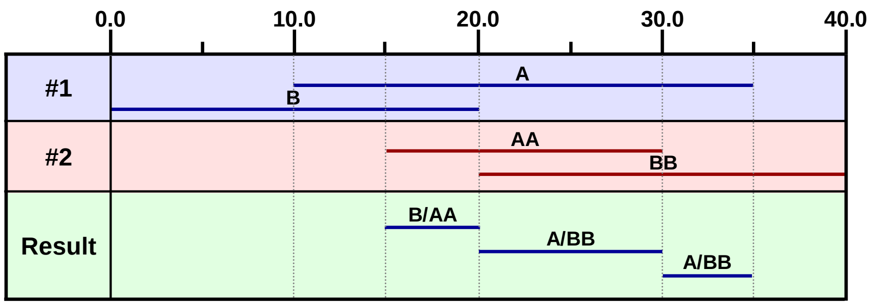
The Event Intersection overlay operators are designed to take two input event features and generate an output query that represents the intersection overlay of them. The output query will contain all the fields from the first input event feature and all the fields from the second input event feature. New fields will be added that contain the measure fields of the output query.

The input events can be linear or point. If both the input events are linear, the result is linear; if both input events are points, the result is point. But, if one input event is linear and the other is point, the output event type is determined by the event type of the first event feature. If the first is linear, the output is linear; if the first is point, the output is point.

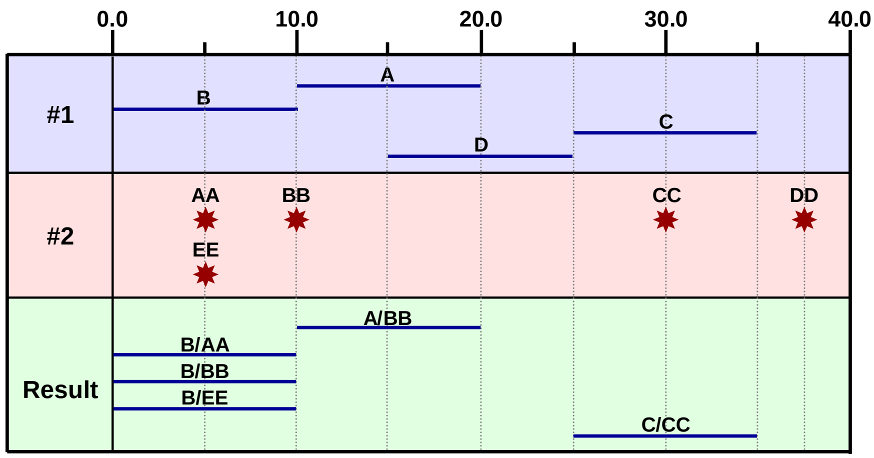
Event Intersection overlay operation when both events are linear is as follows:



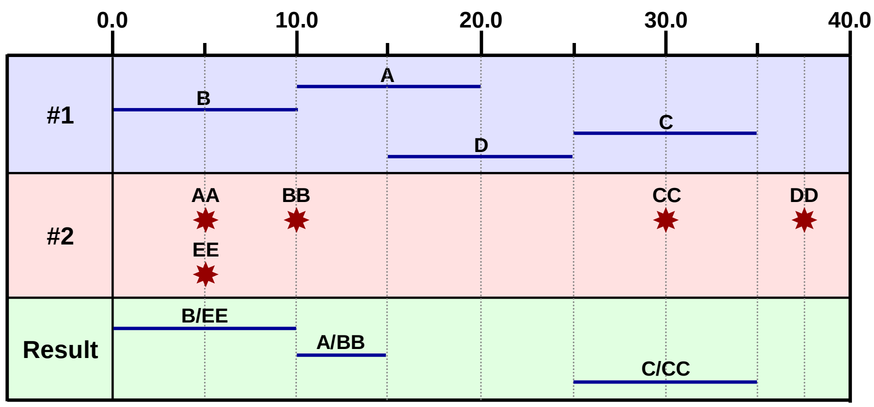
Event Intersection overlay operation (without overlaps) when both events are linear is as follows:



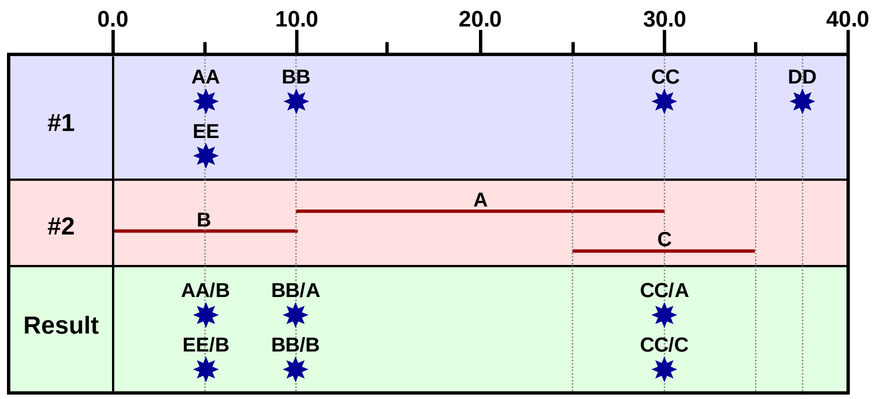
Event Intersection overlay operation when the first event is linear and the second is point is as follows:



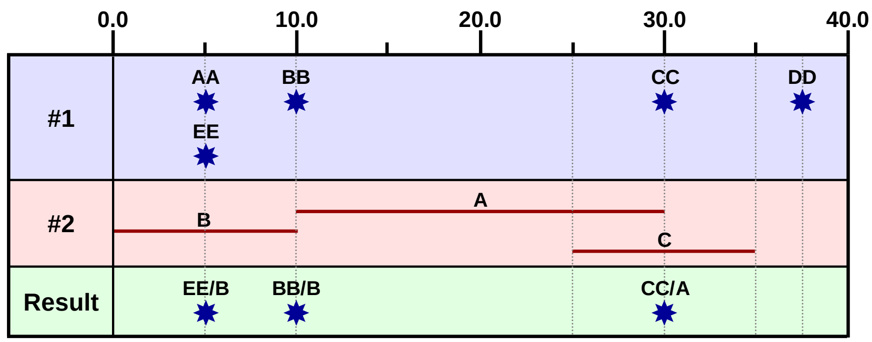
Event Intersection overlay operation (without overlaps) when the first event is linear and the second is point is as follows:



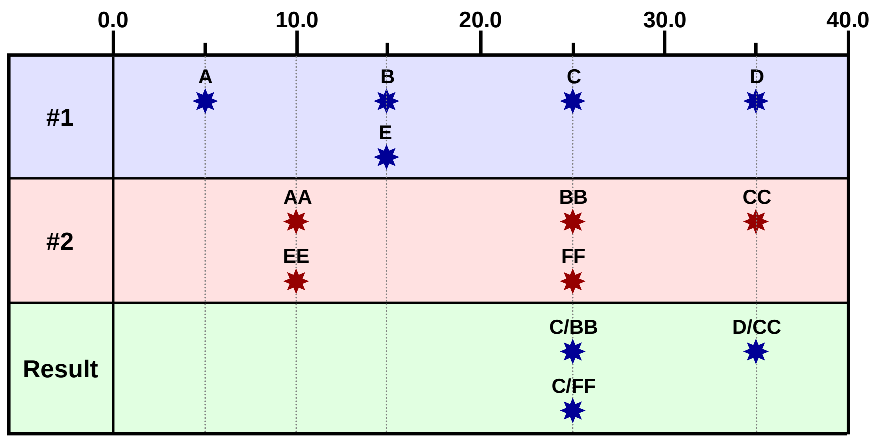
Event Intersection overlay operation when the first event is point and the second is linear is as follows:



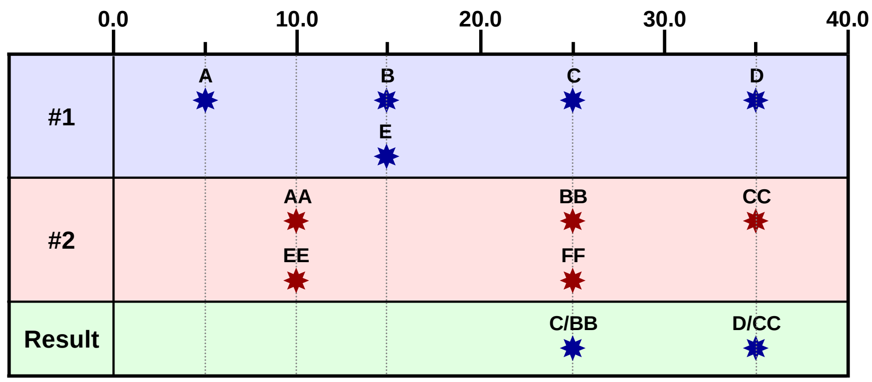
Event Intersection overlay operation (without overlaps) when the first event is point and the second is linear is as follows:



Event Intersection overlay operation when both events are points is as follows:



Event Intersection overlay operation (without overlaps) when both events are points is as follows:

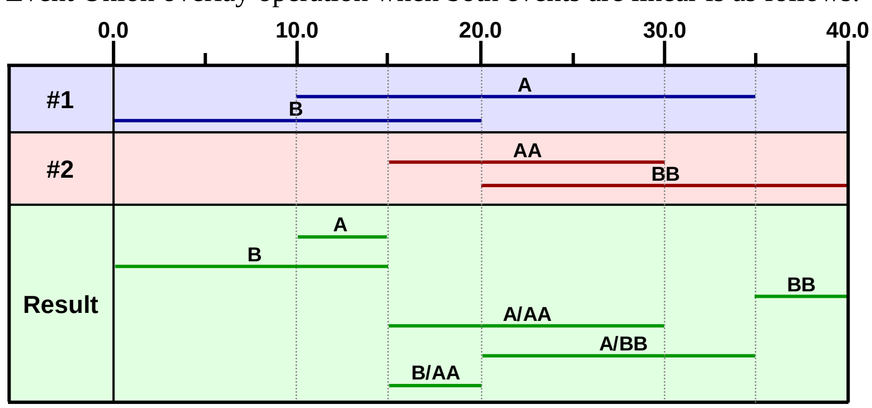


Event Union

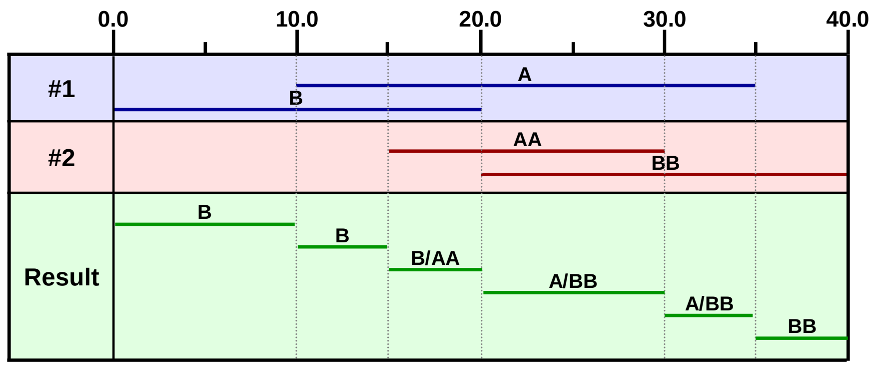
The Event **Union** overlay operators are designed to take two input event features and generate an output query that represents the union overlay of the two. The output query will contain all the fields from the first input event feature and all the fields from the second input event feature. New fields will be added that contain the measure fields of the output query.

The input events can be linear or point. If both the input events are linear, the result is linear; if both input events are points, the result is point. But, if one input event is linear and the other is point, the output event type is determined by the event type of the first event feature. If the first is linear, the output is linear; if the first is point, the output is point.

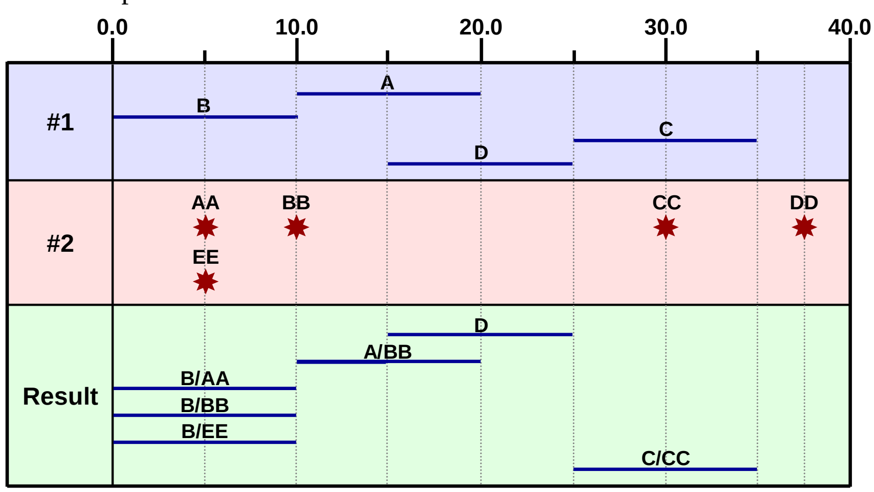
Event Union overlay operation when both events are linear is as follows:



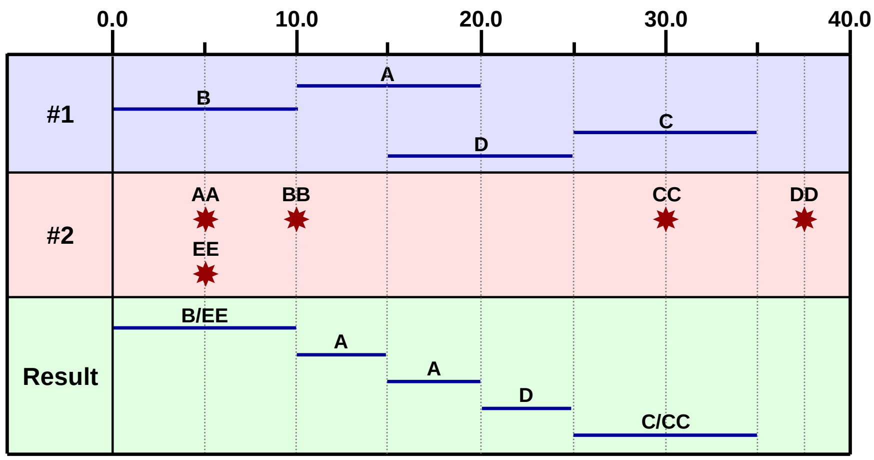
Event Union overlay operation (without overlaps) when both events are linear is as follows:



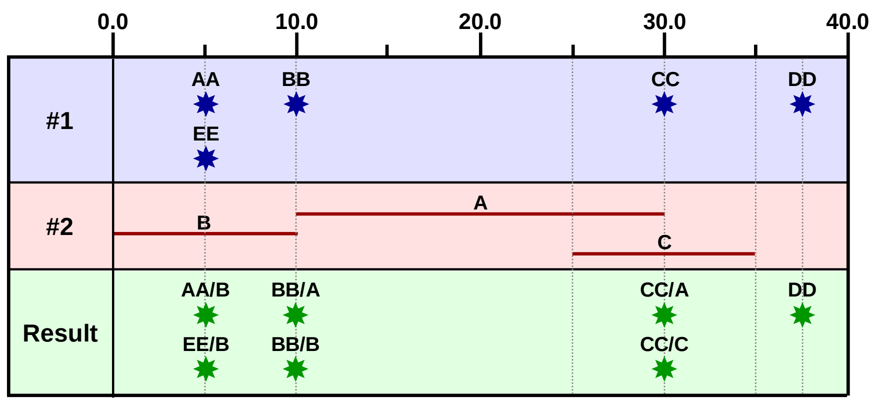
Event Union overlay operation when the first event is linear and the second is point is as follows:



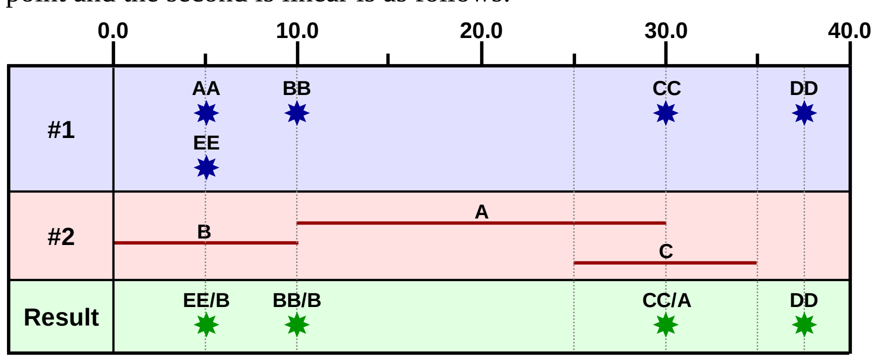
Event Union overlay operation (without overlaps) when the first event is linear and the second is point is as follows:



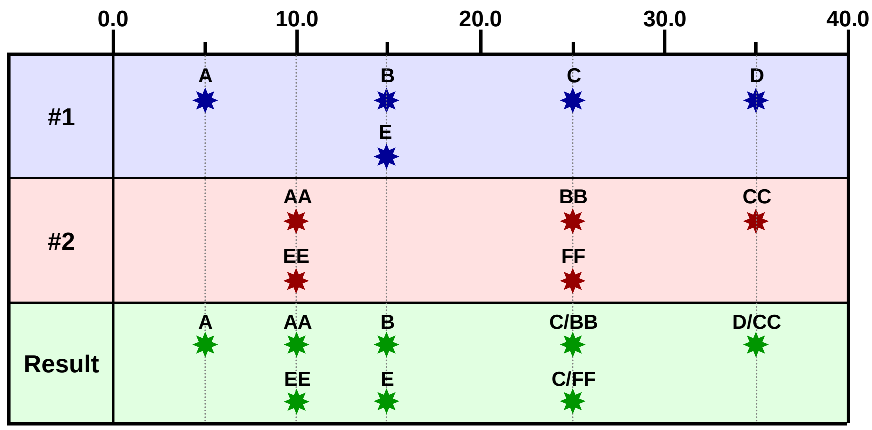
Event Union overlay operation when the first event is point and the second is linear is as follows:



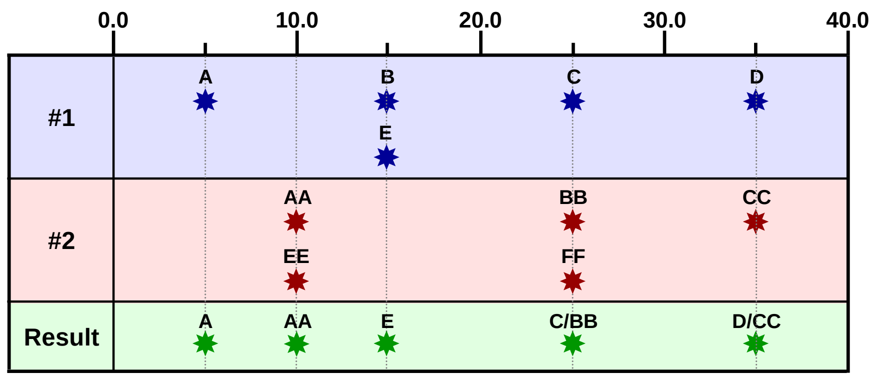
Event Union overlay operation (without overlaps) when the first event is point and the second is linear is as follows:



Event Union overlay operation when both events are points is as follows:



Event Union overlay operation (without overlaps) when both events are points is as follows:

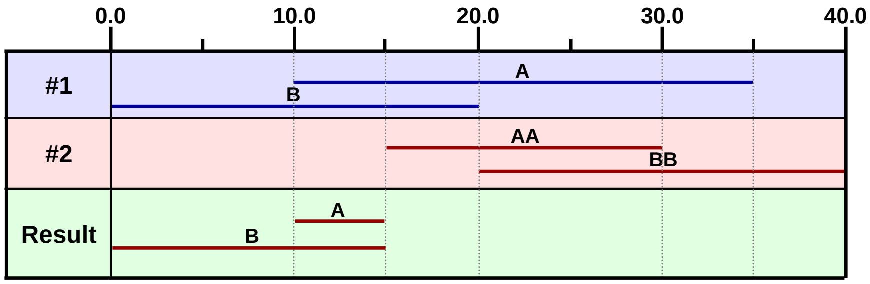


Event Difference

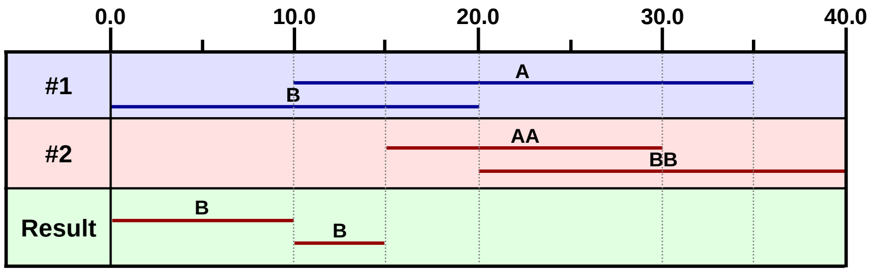
The Event **Difference** overlay operators take two input event features and generate an output query, which represents the difference of the overlay. The output is a query that contains all the fields from the first input event feature and the computed output **Start Measure** and the output **End Measure** fields. The values in the measure fields are the results of the difference overlay

The input event features can be linear or point. If both input features are linear, the result is linear; if both are points, the result is point. But, if one is linear and the other is point, the output event type is determined by the event type of the first input feature class. If the first is linear, the output is linear; if the first is point, the output is point.

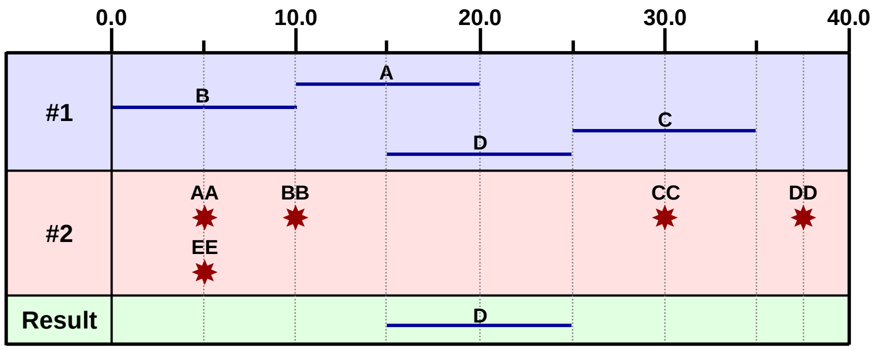
Event Difference overlay operation when both events are linear is as follows:



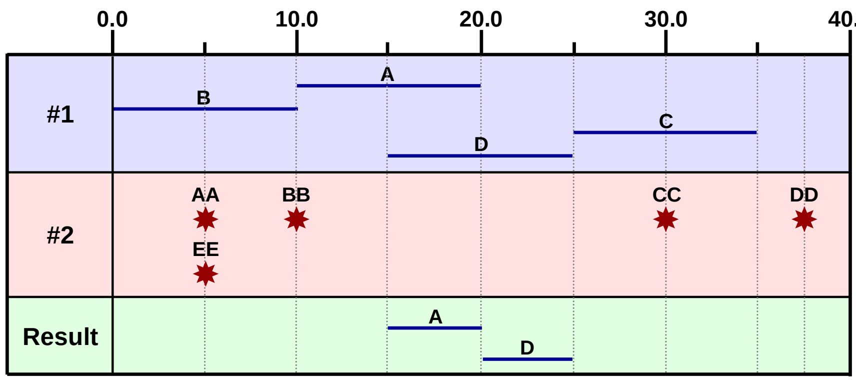
Event Difference overlay operation (without overlaps) when both events are linear is as follows:



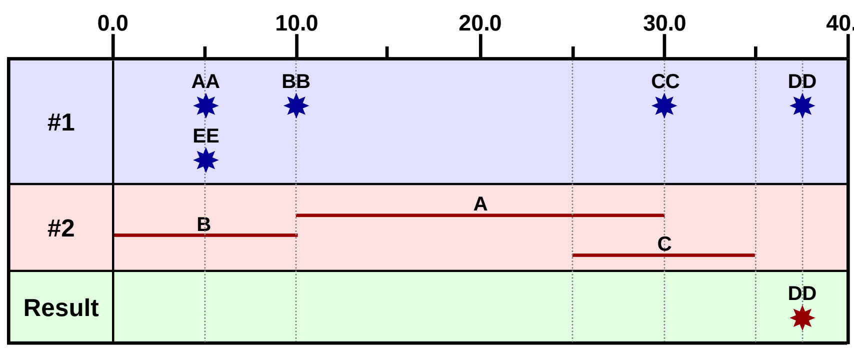
Event Difference overlay operation when the first event is linear and the second is point is as follows:



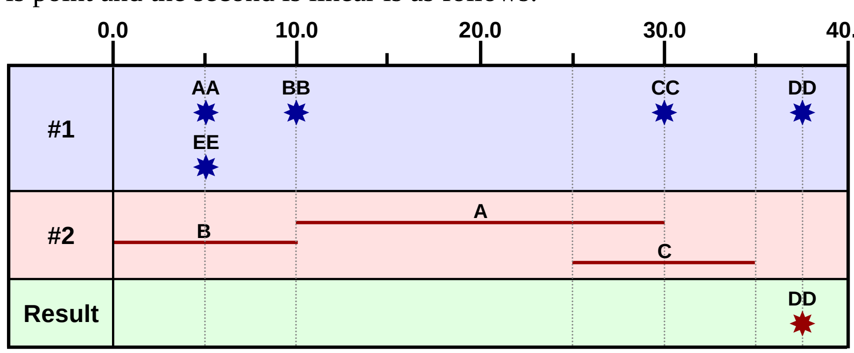
Event Difference overlay operation (without overlaps) when the first event is linear and the second is point is as follows:



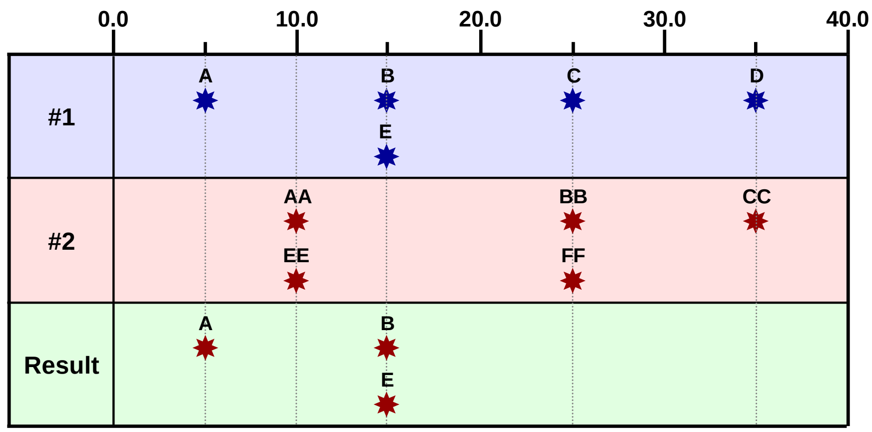
Event Difference overlay operation when the first event is point and the second is linear is as follows:



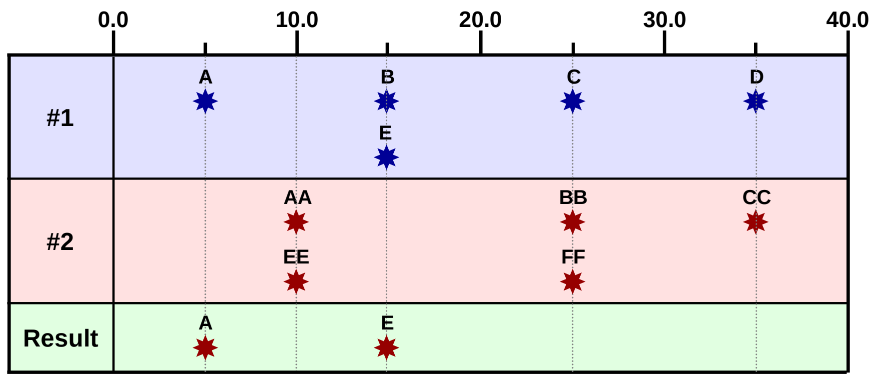
Event Difference overlay operation (without overlaps) when the first event is point and the second is linear is as follows:



Event Difference overlay operation when both events are points is as follows:



Event Difference overlay operation (without overlaps) when both events are points is as follows:

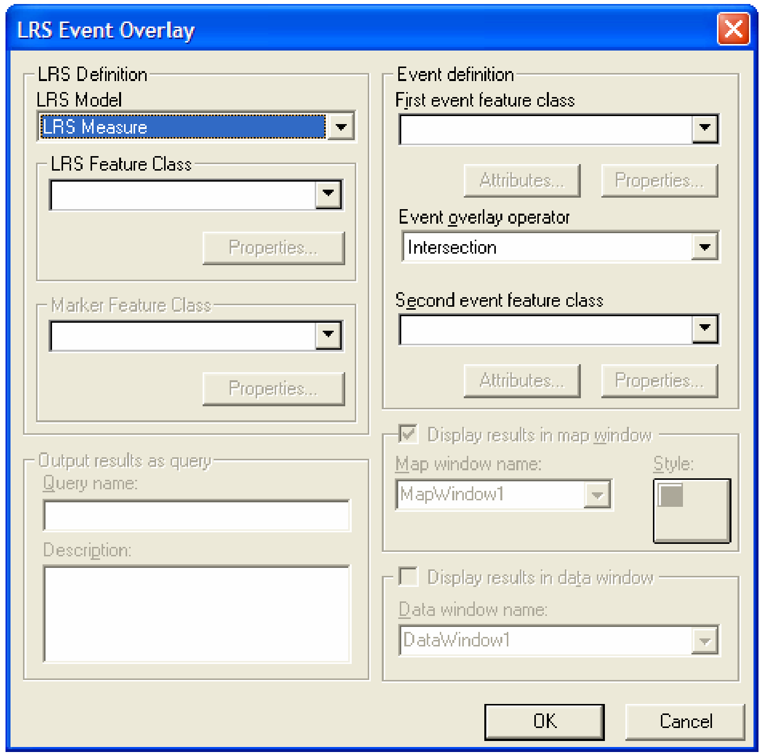


The LRS Event Overlay Command Workflow

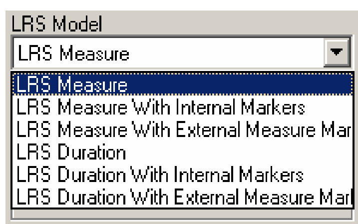
This section presents the procedural steps for using **Event Overlay.**

1. Open a GeoWorkspace; then connect to the warehouse containing the LRS feature class. If the event feature classes are in a different warehouse, connect to that warehouse also. It is possible that each of the event feature classes could be from different warehouses; if so, you need to make the appropriate connections.
2. Select **Transportation > LRS Analysis > LRS Event Overlay**.

   The **LRS Event Overlay** *dialog box is displayed.*



3. Click the **LRS Model** drop-down list; then select the correct LRS model. For more on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.



4. Click the **LRS Feature Class** drop-down list; then select the appropriate linear network feature to be used for **Event Overlay** analysis.

5. If the LRS Model selected was **LRS Measure with External Measure Markers** or **LRS Duration with External Measure Markers**, select the **Marker Feature Class** from the drop-down list.

6. Click the **LRS Feature Class – Properties** button.

   The **LRS Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for information on using this dialog box. If this is the first time this **Properties** dialog box is used, there will be no entries in any of the **LRS Key Fields** or **LRS Definition Fields**.

---

7. Click **OK** after setting the appropriate values.

   You are returned to the **LRS Event Overlay** *dialog box.*

8. Click the **First event feature class** drop-down list; then select the connection and the event table/query to be used for event intersection analysis.

9. Click the **First event feature class – Properties** button.

   The **Event Properties** *dialog box is displayed.*

Note: See the "Working with the Event Properties Dialog Box" chapter for information on using this dialog box.

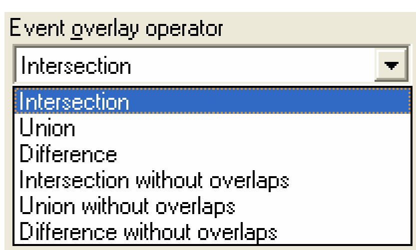
---

10. Click **OK** after setting the appropriate values.

    Y*ou are returned to the* **LRS Event Overlay** *dialog box.*

11. Click the **First event feature class – Attributes** button, and select the attributes to be included in the output recordset.

12. Click the **Event Overlay Operator** drop-down list; then select the appropriate overlay operator.



13. Click the Second event feature class drop-down list; then select the connection and the event table/query to be used for event intersection analysis.

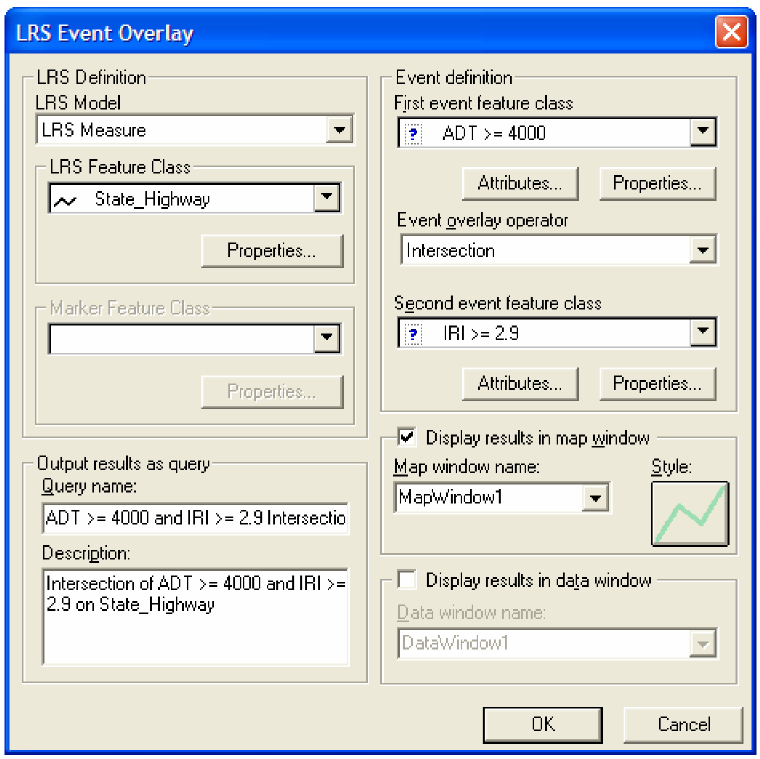
14. Click the **Second event feature class – Properties** button.

15. On the **Event Properties** dialog box, select the appropriate columns for the **Key Fields** and **Measure Fields**.

16. Set the **Event Type** and **Event Reference Type** option buttons.

17. Select the appropriate **Unit – Measure** for the event data; then click **OK**.

You are returned to the **LRS Event Overlay** *dialog box.*



18. Click the **Second event feature class – Attributes** button, and select the attributes to be included in the output recordset.

19. If you want to see the results in a map window, make sure that the check box to the left of the **Display results in map window** field is checked on and that the appropriate **Map window name** is selected.

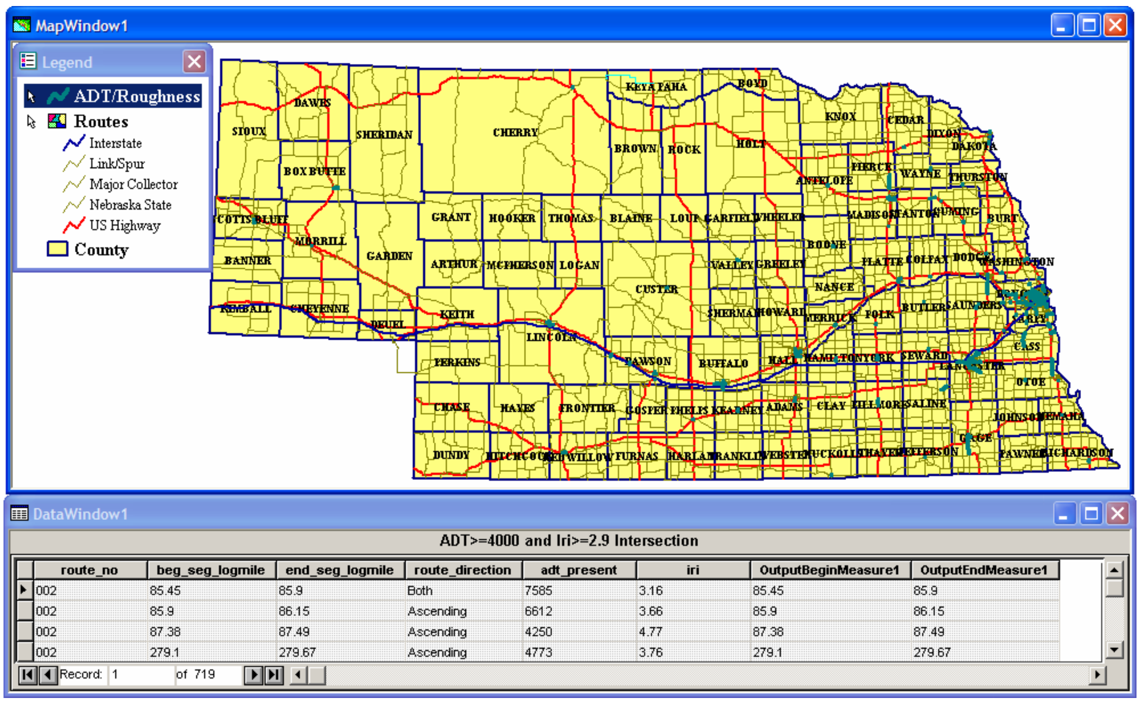
20. *Optional*: Click **Style** to define the display settings for the results in the map window.

21. If you want to see the results in a data window, make sure that the check box to the left of the **Display results in data window** field is checked on and that the appropriate **Data window name** is selected.

22. When you have made the appropriate settings, click **OK**.

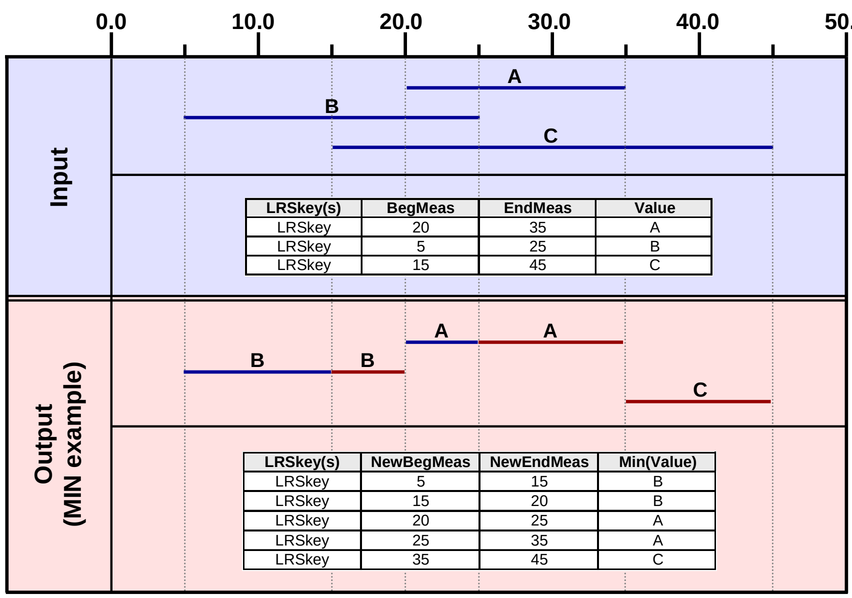
23. When the results are returned, tile the windows.

    The GeoWorkspace should resemble the following:



8 – Working with the Resolve LRS Event Overlaps Command

The **Resolve LRS Event Overlaps** command provides the ability to resolve the overlaps that may exist within a Linear Event feature class. This ensures that there are no two linear events that cover the same portion of the network. This command gives you control over what criteria are used to define which of the overlapping events are chosen for output. What this command does is illustrated in the example below.



In the Input portion of this illustration you will see three different linear event records that overlap each other. For this example we used the MIN function (minimum) function to select those linear events that have the lowest Value attribute. The command makes use of the Functional Attributes capability of GeoMedia, which provides great flexibility to define what criteria are used in choosing which linear events are output.

This command is for linear events only. Identical functionality is available for point events using the **Analytical Merge** command. The result of this command is tabular data. To see the results in the map view you can use the **Dynamic Segmentation** command to add geometry to each recordset.

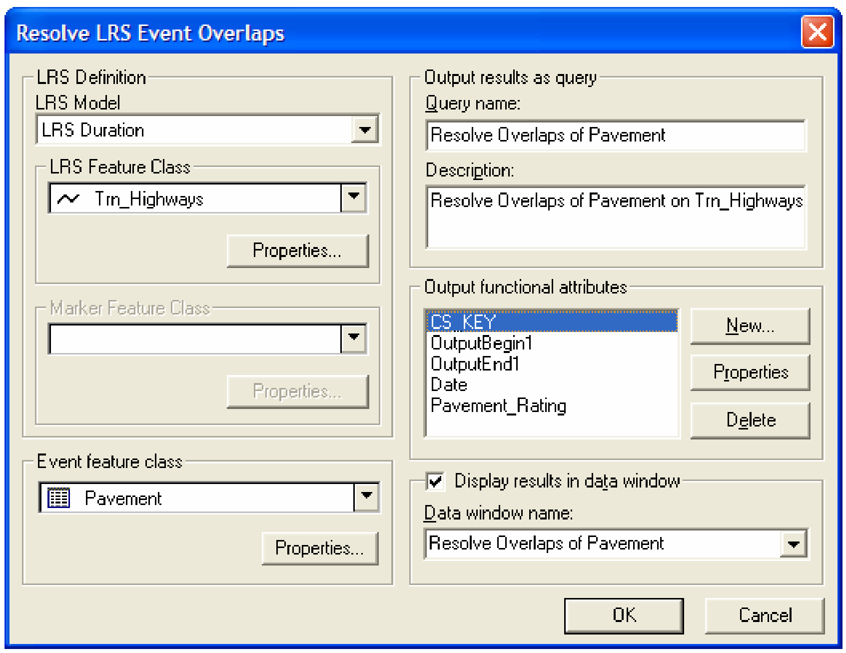
The Resolve LRS Event Overlaps Command Workflow

This section presents the procedural steps for using **Resolve LRS Event Overlaps.**

1. Open a GeoWorkspace that contains one or more readable connections.

2. Select **Transportation > LRS Analysis > Resolve LRS Event Overlaps.**

   The **Resolve LRS Event Overlaps** *dialog box is displayed.*



3. Select the **LRS Model**, the **Marker Feature Class,** and the **Event feature class** from the drop-down lists; then select the **Properties** button for each. Note this command is for linear events only.

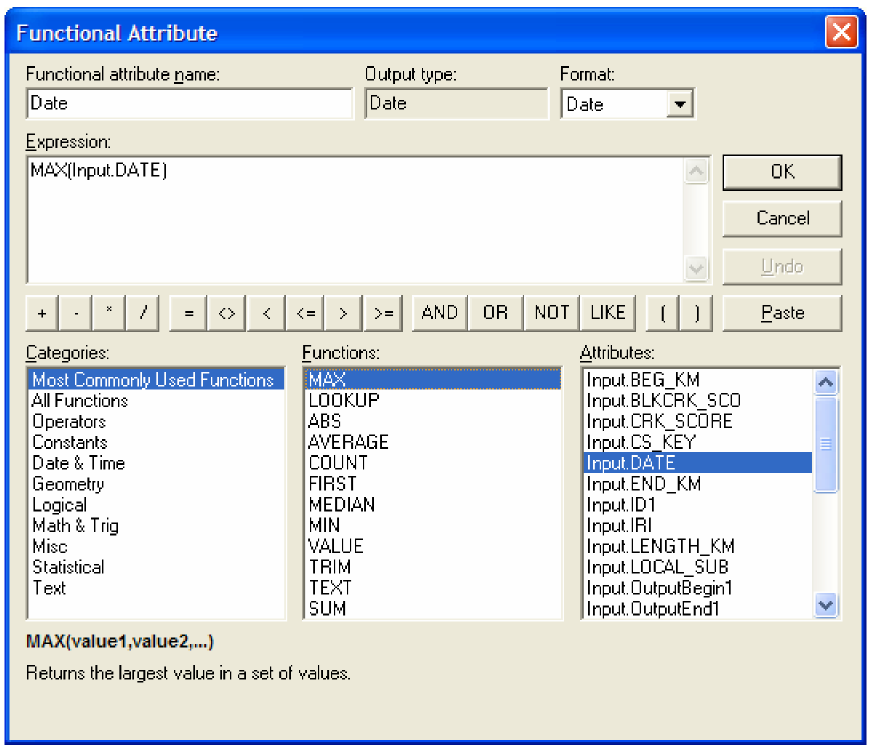
Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for information on the workflow for the LRS feature class and the LRS marker feature class; and see the "Working with the Event Properties Dialog Box" chapter for information on the workflow for the Event feature class.

---

4. Type in a query name and an optional description in the **Output results as query** fields.

5. In the **Output functional attributes** box, click **New** to specify any functional attributes, or click **Properties** to edit an exiting functional attribute. This will define how the overlaps in your linear feature class are resolved. In the example shown we are finding the most recent records by finding the maximum date. If there are two or more pavement condition readings for the same portion of the network, this will return only the most current record.

   If you select **New** *or* **Properties**, *the* **Functional Attribute** *dialog box will appear.*



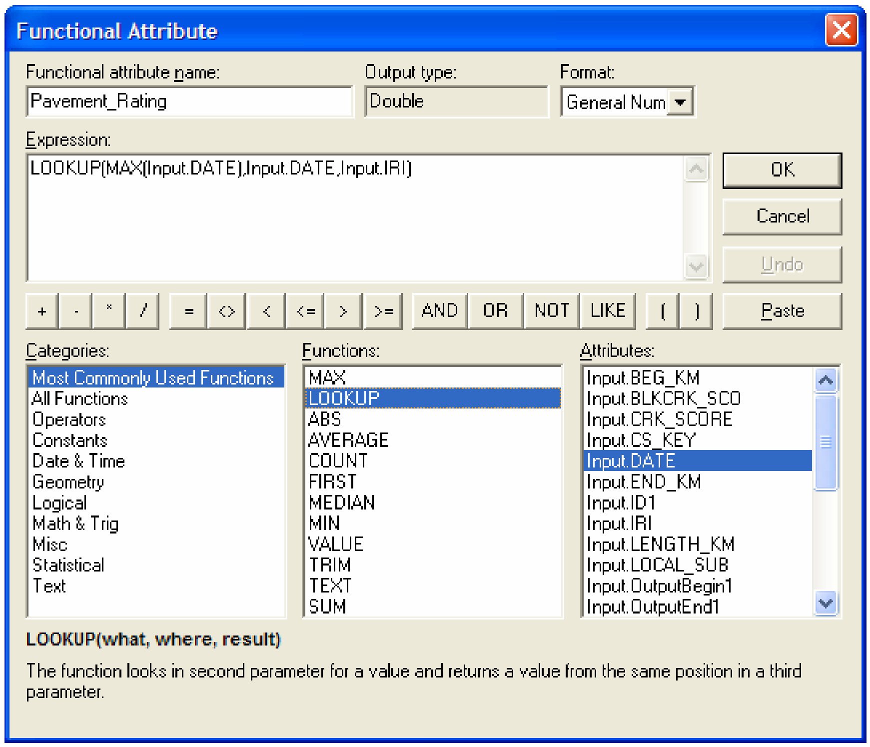
6. Click **OK** on the **Functional Attribute** dialog box to save your input or **Cancel** if you do not wish to save your work.

   The **Functional Attribute** *dialog box will be dismissed, and control will return to the* **Resolve LRS Event Overlaps** *dialog box.*

7. In the **Output functional attributes** box, click **New** again to specify another functional attribute. This second functional attribute will bring along additional attributes that belong to the records selected via the functional attribute defined in Step 5. In the example shown, we are

finding the most pavement condition readings that are on the most recent records.

If you select **New** *or* **Properties**, *the* **Functional Attribute** *dialog box will appear.*



Note: The LOOKUP functional attribute is very valuable to users of this command. In the example shown here the first parameter of the LOOKUP function is the same function as defined in Step 5, the second parameter is the field name that is narrowed by that functional attribute, and the third parameter is the field name we want to return.

8. Click **OK** on the **Functional Attribute** dialog box to save your input or **Cancel** if you do not wish to save your work.

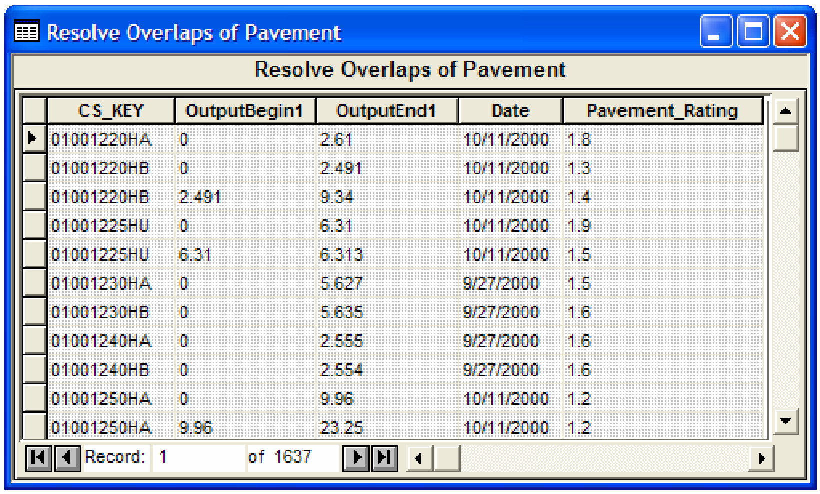
   The **Functional Attribute** *dialog box will be dismissed, and control will return to the* **Resolve LRS Event Overlaps** *dialog box.*

9. Repeat Steps 7 and 8 for each other attribute you wish to bring along from the records selected via the functional attribute defined in Step 5.

10. *Optional*: Select **Display results in data window**; then accept the default or select or type in a data window name.

11. Click **OK** to generate and display the results; or click **Cancel** to dismiss the dialog box.

    The Data Window below shows the output of this command.



Resolve Overlaps of Pavement

CS_KEY	OutputBegin1	OutputEnd1	Date	Pavement_Rating
01001220HA	0	2.61	10/11/2000	1.8
01001220HB	0	2.491	10/11/2000	1.3
01001220HB	2.491	9.34	10/11/2000	1.4
01001225HU	0	6.31	10/11/2000	1.9
01001225HU	6.31	6.313	10/11/2000	1.5
01001230HA	0	5.627	9/27/2000	1.5
01001230HB	0	5.635	9/27/2000	1.6
01001240HA	0	2.555	9/27/2000	1.6
01001240HB	0	2.554	9/27/2000	1.6
01001250HA	0	9.96	10/11/2000	1.2
01001250HA	9.96	23.25	10/11/2000	1.2

9 – Working with the LRS Event Conversion Command

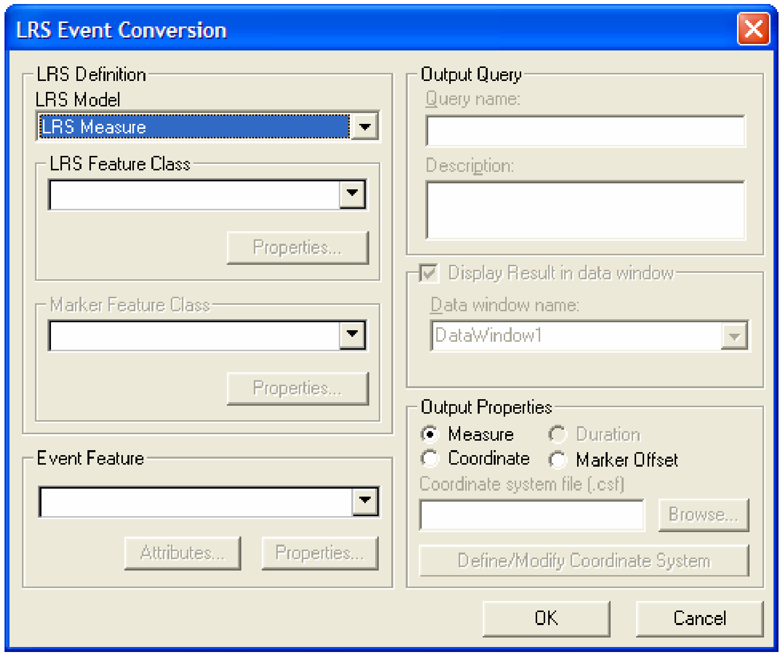
This section provides an overview of the **LRS Event Conversion** command, which provides the ability to convert events from one LRS Event Reference Type to another. For example, you can convert Coordinate events to Measure events, or vice versa. **LRS Event Conversion** allows the choice of the input recordset for the Linear Referencing System (LRS), the event, the marker features, and the properties of these recordsets; and it will convert the Event data format from the source to the Event data format specified by the **Output Properties**.

The LRS Event Conversion Command Workflow

This section presents the procedural steps for using **LRS Event Conversion**.

1. Open a GeoWorkspace; then connect to the warehouse containing the LRS feature class. If the event feature classes are in a different warehouse, connect to that warehouse also. It is possible that each of the event feature classes could be from different warehouses; if so, you need to make the appropriate connections.
2. Select **Transportation > LRS Analysis > LRS Event Conversion**.

   The **Event Conversion** *dialog box is displayed.*



3. Click the **LRS Model** drop-down list; then select the correct LRS model. For more on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.

4. Click the **LRS Feature Class** drop-down list; then select the appropriate feature class to be used for **LRS Event Conversion** analysis.

5. Click the **LRS Feature Class** – **Properties** button.

   The **LRS Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for information on using this dialog box as well as the **Marker Properties** dialog box. If this is the first time this dialog box is used, there will be no entries in any of the **LRS Key Fields** or **LRS Definition Fields**.

---

6. Click **OK** after setting the appropriate values.

   You are returned to the **Event Conversion** *dialog box.*

7. If the LRS model is **LRS Measure with External Measure Markers** or **LRS Duration with External Measure Markers**, select the input marker feature class from the drop-down list in the **Marker Feature Class** box.

Note: The **Marker Feature Class** drop-down list and the **Properties** button will not be enabled unless the LRS model is one of the two listed in Step 7.

8. Click **Properties** in the **Marker Feature Class** box.

   The **Marker Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for information on using this dialog box.

9. Click **OK** after setting the appropriate values.

   You are returned to the **LRS Event Conversion** *dialog box.*

10. On the **LRS Event Conversion** dialog box, select the input event feature class from the drop-down list in the **Event Feature** box.

11. Click the **Event Feature – Properties** button.

    The **Event Properties** *dialog box is displayed.*

Note: See the "Working with the Event Properties Dialog Box" chapter for information on using this dialog box.

12. If applicable, define the event coordinate system by clicking **Browse** or by clicking **Define/Modify Coordinate System**.

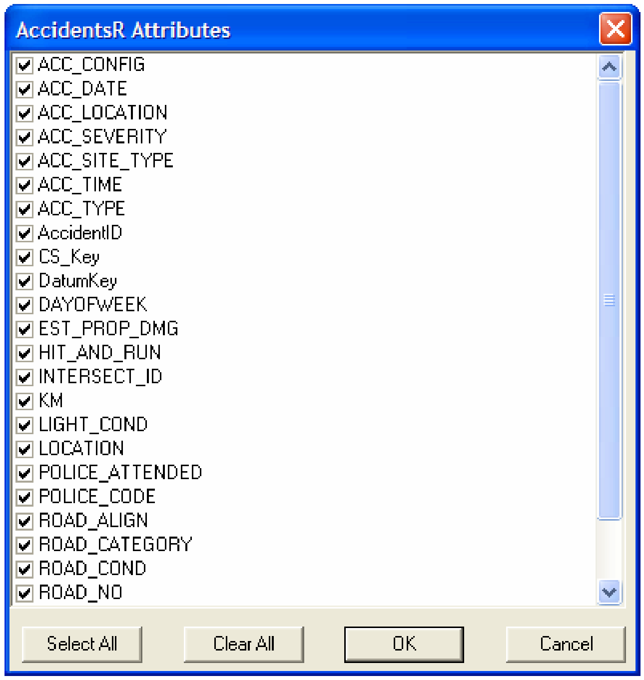
Note: See the "Working with the Coordinate System Dialog Box" chapter for information on the dialog boxes that appear.

13. Click **OK** after setting the appropriate values.

    You are returned to the **LRS Event Conversion** *dialog box.*

14. Click the **Event Feature – Attributes** button to select attributes from the input event feature that you want to have available in the output query.

    The **Event Attributes** *dialog box is displayed.*



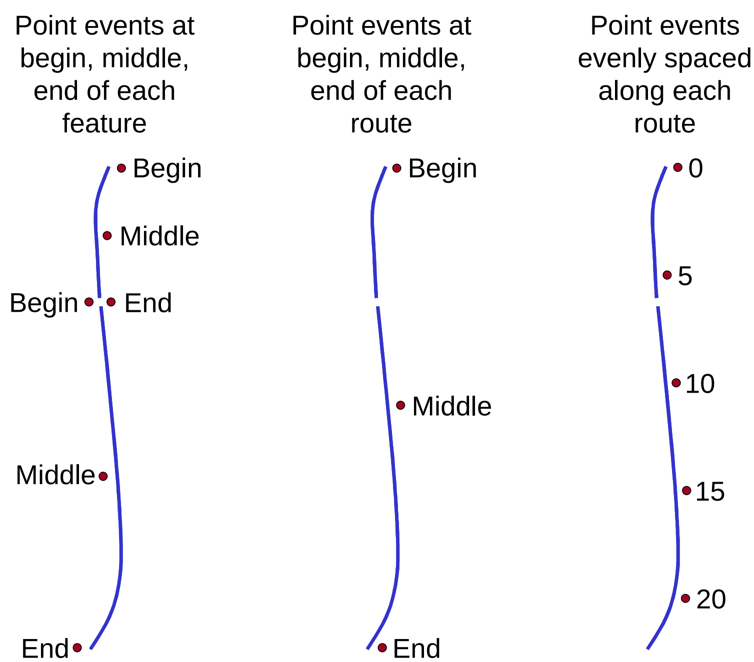
15. Select those attributes that you want reflected in the output recordset. **Select All** and **Clear All** buttons are provided to speed this process.
16. Click **OK** after setting the appropriate values.

    You are returned to the **Event Conversion** *dialog box.*
17. Select the output Event Reference Type from the choices in the **Output Properties** group box.
18. If the **Coordinate** type output property is selected, the coordinate system for the output must be defined by selecting a coordinate system file, by clicking **Browse,** or by clicking **Define/Modify Coordinate System**.
19. Accept the default, or choose another **Query name** field.
20. *Optional*: Assign a description in the **Description** field.
21. Check the **Display Result in data window** check box.
22. Accept the default, or type in a data window name.
23. Click **OK** to generate and display the Event Conversion results in the specified data window.

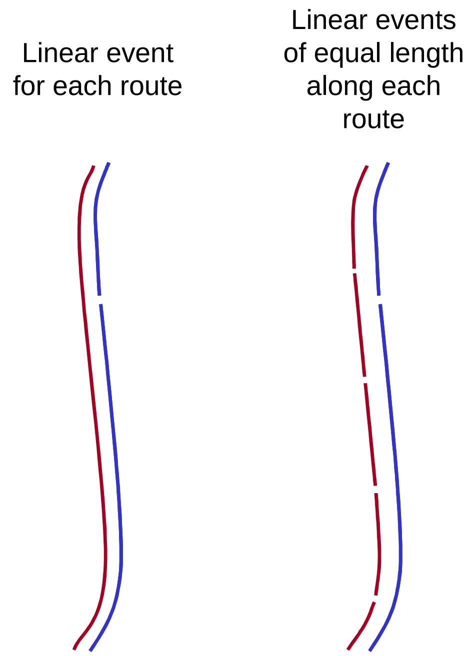
10 – Working with the LRS Event Generation Command

The **LRS Event Generation** command provides the ability to create either point or linear events for an existing LRS feature class. A variety of methods are available to control where the new events are located along the LRS. The output events from the command can be used in numerous workflows, including labeling and aggregation analysis. For more detail on some of these workflows, see the "LRS Annotation Workflows" and "LRS Analysis Workflows" chapters.

The command takes an input LRS recordset and generates point or linear events. Point events can be generated at various points along the LRS network, depending on the options selected. The placement options include placing the points at the begin, middle, and/or end of each LRS feature instance; at the begin, middle, and/or end of each LRS Keyset group (features that share the same LRS keys); or at a user-defined interval along the LRS Keyset group.



Linear events can be generated at various points along the LRS network, depending on the options selected. The placement options include linear events generated for each LRS Keyset group, or equal-length segments which are generated at a user-defined length along the LRS Keyset group.



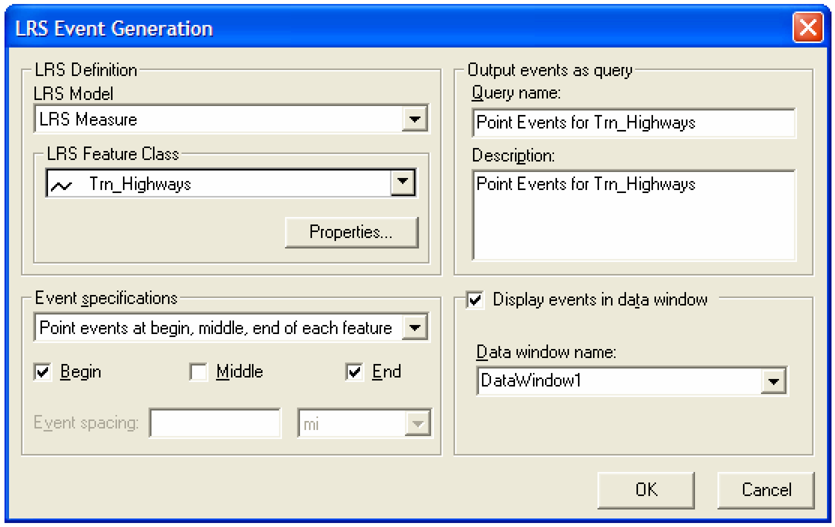
The command uses the LRSEventGenerationPipe to perform the functionality. The result of this pipe is tabular data. To see the results in the map view, you can use the **Dynamic Segmentation** command to add geometry to each recordset.

The LRS Event Generation Command Workflow

This section presents the procedural steps for using the **LRS Event Generation** command.

1. Open a GeoWorkspace with readable connections.
2. Select **Transportation >LRS Analysis > LRS Event Generation**.

   The **LRS Event Generation** *dialog box is displayed.*



3. Select the **LRS Model** and the **LRS Feature Class** from the drop-down lists. For more on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.

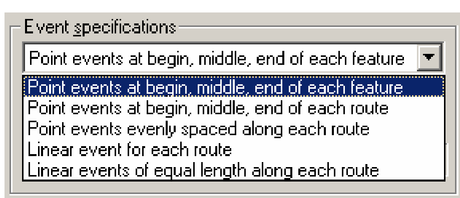
4. Click the **LRS Feature Class – Properties** button.

   The **LRS Properties** *dialog box is displayed.*

Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for information on using this dialog box. If this is the first time the dialog box is used, there will be no entries in any of the **LRS Key Fields** or **LRS Definition Fields**. Subsequent uses of the command will have the fields automatically set to what you selected the last time you ran the command.

---

5. Choose an option from the **Event specifications** box.

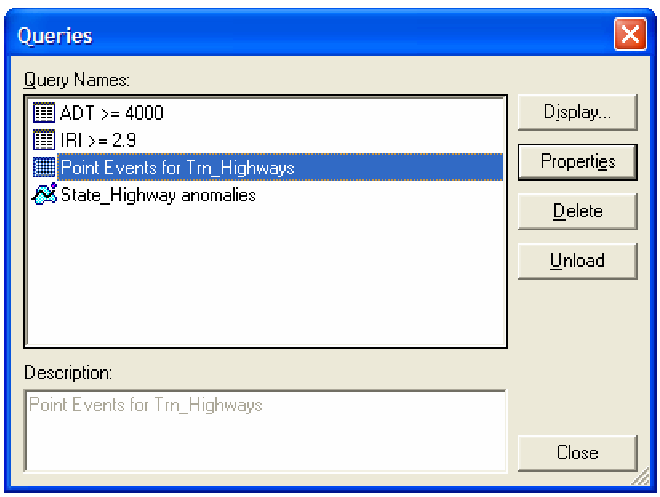


6. If the **Point events at begin, middle, end of each feature** or the **Point events at begin, middle, end of each route** option is selected, Choose one or all of the **Begin**, **Middle**, and **End** check boxes.

7. If the **Point events evenly spaced along each route** option is selected, type the measure interval between the point events and the unit.

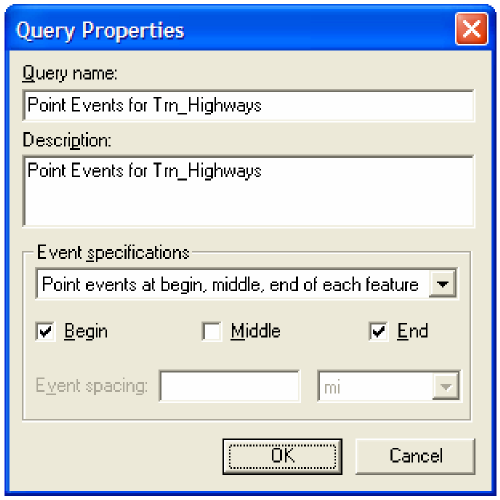
8. If the **Linear events of equal length along each route** option is selected, key in the duration of equal length linear event and the unit.
9. Accept or override the default query name assigned in the **Query name** field, and assign an optional description using the **Description** field.
10. To display the output in a data window, check the **Display events in data window** check box; then accept the default **Data window name**, or select or key in a data window name.
11. Click **OK** to generate and to display the results in the specified data window.
12. After reviewing the result, you can change the event specifications settings by editing the query. Do this by first selecting **Analysis > Queries** from the GeoMedia Professional menu bar.

The **Queries** *dialog box is displayed.*



13. Select the query generated by the **LRS Event Generation** command, and then click **Properties.**

The **Query Properties** *dialog box is displayed.*



14. Edit the **Query** name, the **Description**, and the **Event specifications** settings as desired. When finished click **OK**. This will alter the results of the query according to your new input.

11 – Working with the Insert LRS Event Command

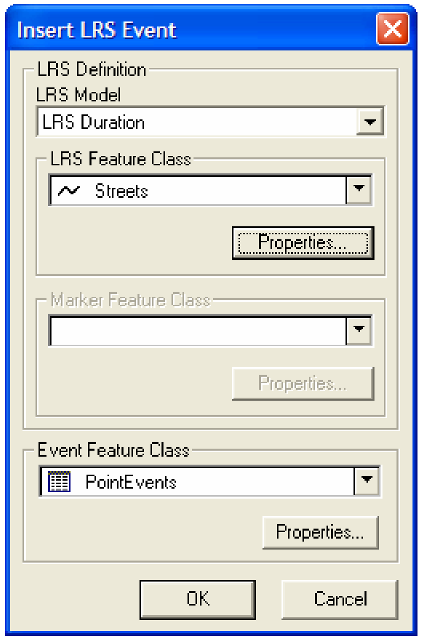
The **Insert LRS Event** command allows you to interactively create event data by clicking on the map view to define LRS key, begin measures, and (for linear events) end measures. You will also be able to fill in non-location-related attribution. This is particularly useful in situations where map data exists (such as aerial photography) that can be interpreted to create event data (for example, the number of lanes or bridge locations).

The Insert LRS Event Command Workflow

This section presents the procedural steps for using **Insert LRS Event.**

1. Open a GeoWorkspace that contains one or more read/write connections.
2. *Optional:* Display the LRS linear feature class along which you want to create event data. This step makes it easier to select points along the LRS.
3. *Optional:* If you have any map data that will help you discern where event data is located, display that data. This may come in many forms, such as aerial photography.
4. *Optional:* Create and display a dynamic segmentation query of the Event class you want to add records to. This step is helpful in that it provides visual feedback of the event data you will be creating. For more information on this, see the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with GeoMedia Professional*.
5. Select **Transportation > LRS Analysis > Insert LRS Event.**

   The **Insert LRS Event** *dialog box is displayed.*



6. Click the **LRS Model** drop-down list; then select the correct LRS model. For more on the different LRS Models supported, see the "GeoMedia Transportation LRS Data Structures" appendix.
7. Click the **LRS Feature Class** drop-down list; then select the appropriate feature class along which event data is to be created.
8. Click the **LRS Feature Class – Properties** button to set the Key field and measure information.

   The **LRS Properties** *dialog box is displayed.*

   Note: See the "Working with the LRS and Marker Properties Dialog Boxes" chapter for a detailed discussion of this dialog box.

9. Click **OK** after setting the appropriate values.

   You are returned to the **Insert LRS Event** *dialog box.*

10. If the LRS model is **LRS Measure with External Measure Markers** or **LRS Duration with External Measure Markers**, select the input marker feature class from the drop-down list in the **Marker Feature Class** box.

Note: The **Marker Feature Class** drop-down list and the **Properties** button will not be enabled unless the LRS model is one of the two listed in Step 10.

---

11. Click **Properties** in the **Marker Feature Class** box to set the Key field, marker name, and measure information.

    The **Marker Properties** *dialog box is displayed.*

Note: See the “Working with the LRS and Marker Properties Dialog Boxes” chapter for a detailed discussion of this dialog box.

---

12. Click **OK** after setting the appropriate values.

    You are returned to the **Insert LRS Event** *dialog box.*

13. On the **Insert LRS Event** dialog box, select the input event feature class from the drop-down list in the **Event Feature** box.
14. Click the **Event Feature – Properties** button.

    The **Event Properties** *dialog box is displayed.*

Note: See the “Working with the Event Properties Dialog Box” chapter for a detailed discussion of this dialog box.

---

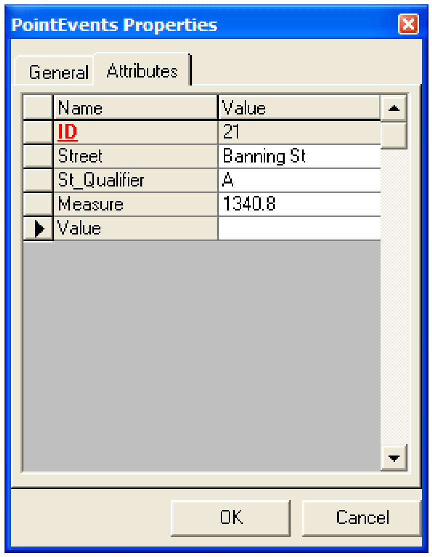
15. Click **OK** after setting the appropriate values.

    You are returned to the **Insert LRS Event** *dialog box.*

16. Click **OK** on the **Insert LRS Event** dialog box to dismiss this dialog box and to start creating event data.
17. Follow the prompts to collect event data locations by clicking in the map window. The first prompt is to **“Click on a feature for begin distance, press ESC to exit.”** If it is a point event, only one click is necessary. If it is a linear event, you will be prompted to give a second click with this prompt: **“Click on the feature for end distance, press Backspace to undo previous click or ESC to exit.”** Note that the second click will be ignored unless it is on an LRS linear feature that shares the same LRS Keys as the LRS linear feature identified by the first click. When there are multiple LRS features located within the tolerance zone of the cursor, you can select the required feature through the **PickQuick** dialog box. More information on the

PickQuick dialog box is available in either the *Working with GeoMedia* or the *Working with GeoMedia Professional* documents.

The **Event Properties** *dialog box is displayed with the key and measure field values that reflect the position you clicked on the LRS on the map window.*



18. Fill in other event data attributes, and click **OK** to dismiss the **Select Set Properties** dialog box.
19. Repeat the two previous steps as often as desired to collect event data. To exit, press ESC.

12 – Working with the LRS Keys for Coordinate Events Command

The **LRS Keys For Coordinate Events** command provides a way to linearly reference data whose only geographic reference is coordinates or latitude and longitude. The command searches in the vicinity of these coordinate locations and locates the nearest LRS linear feature within a given tolerance. The output of the command is the input data plus the LRS keys of the located LRS linear feature for each record. Since the other commands in the product can handle events data located by LRS keys and coordinates, this data is now ready to use with commands such as **Dynamic Segmentation** or **Event Overlay**. Coordinate events that are not within the given tolerance of any LRS feature are not given key values.

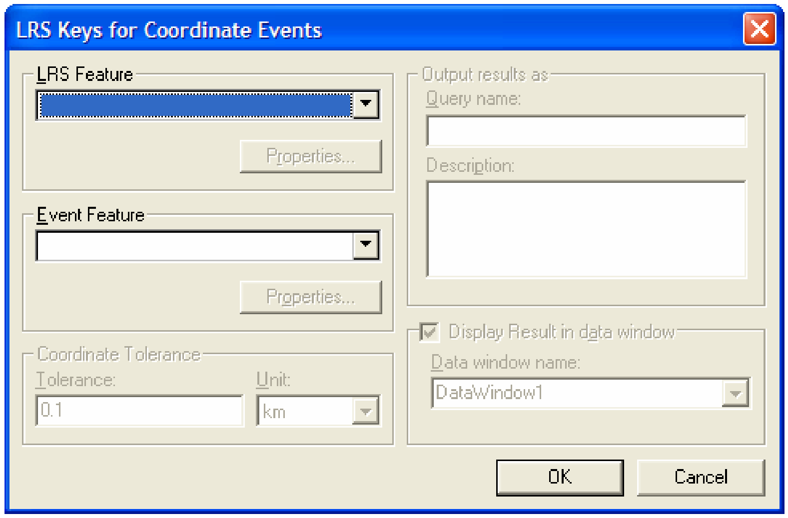
This command is particularly useful for event data collected with GPS because this data often does not identify the LRS keys of the road that the data is associated with. It is also useful for GIS point features that you want to linearly reference. In this case, use the GeoMedia Professional **Analyze Geometry** command to add coordinate attributes to the recordset. From there the **LRS Keys For Coordinate Events** command can add LRS keys, and the data is ready to use in a linear referencing environment.

The LRS Keys for Coordinate Events Command Workflow

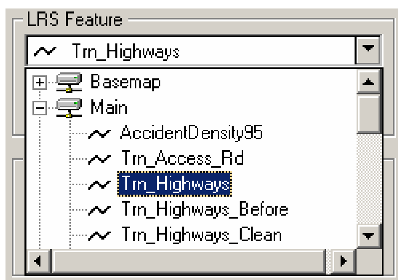
This section presents the procedural steps for using the **LRS Keys For Coordinate Events** command.

1. Open a GeoWorkspace with readable connections.
2. Select the **Transportation > LRS Analysis > LRS Keys For Coordinate Events**.

   The **LRS Keys For Coordinate Events** *dialog box is displayed.*

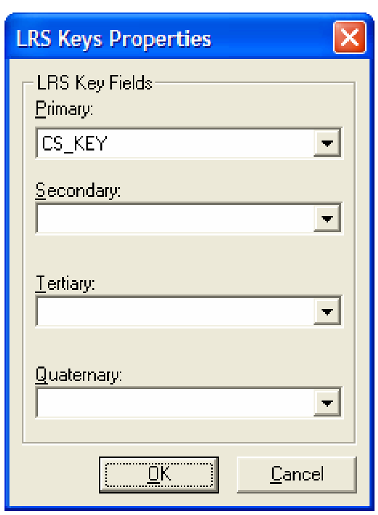


3. Select a feature or query from the **LRS Feature** drop-down list.

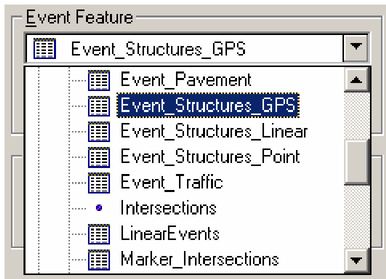


4. Click **Properties** in the **LRS Feature** box**.**

   The **LRS Keys Properties** *dialog box is displayed.*

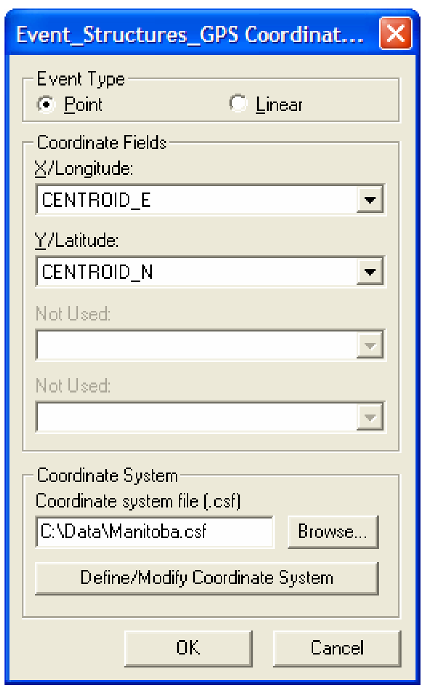


5. On the **LRS Keys Properties** dialog box, accept the defaults, or choose other **Primary, Secondary, Tertiary**, and **Quaternary** Keys for the LRS recordset from the appropriate drop-down lists.
6. Click **OK** to exit this dialog box.
7. On the **LRS Keys For Coordinate Events** dialog box, select an event feature from the **Event Feature** drop-down list.



8. Click **Properties** in the **Event Feature** box**.**

   The **Coordinate Event Properties** *dialog box is displayed.*



9. On the **Coordinate Event Properties** dialog box, select either **Point** or **Linear** as the **Event Type**.
10. In the **Coordinate Fields** box, accept the defaults, or choose other properties for each drop-down list.

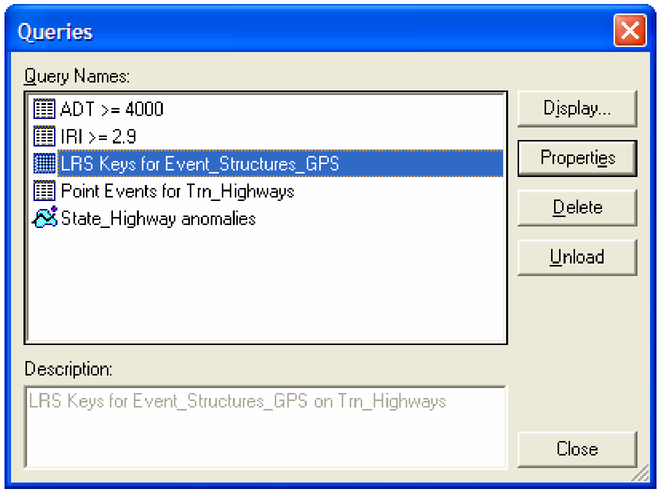
11. Accept the default, or define another event coordinate system by clicking **Browse** or by clicking **Define/Modify Coordinate System**. For more on this see the "Working with the Coordinate System Dialog Box" chapter.

Note: If you want to use the coordinate system from the current GeoWorkspace, you can do this by first saving it to a *.csf* file by selecting **View > GeoWorkspace Coordinate System** from the GeoMedia Professional menu bar. This brings up the **GeoWorkspace Coordinate System** dialog box. From there select the **Save As** button and assign it a filename and location.

---

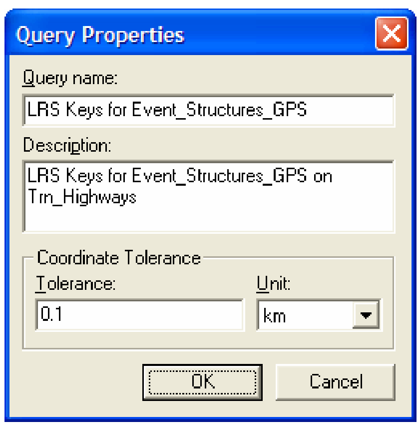
12. Click **OK** to exit the **Coordinate Event Properties** dialog box.
13. In the **Coordinate Tolerance** box on the **LRS Keys For Coordinate Events** dialog box, accept the default, or key in another number for the coordinate tolerance in the **Tolerance** textbox.
14. Accept the default, or choose another unit for the coordinate tolerance from the **Unit** drop-down list.
15. In the **Output Query** box, accept the default, or choose another query name in the **Query Name** field; then assign an optional description in the **Description** field.
16. Check the **Display results in data window** check box to display the new results in a data window. Then accept the default, or select or key in a **Data window name.**
17. Click **OK** to generate and to display the results in the specified data window.
18. After reviewing the result, you can change the tolerance by editing the query. Do this by first selecting **Analysis > Queries** from the GeoMedia Professional menu bar.

The **Queries** *dialog box is displayed.*



19. Select the query generated by the **LRS Keys for Coordinate Events** command, and then click **Properties**.

The **Query Properties** *dialog box is displayed.*



20. Edit the **Query name**, the **Description**, the **Tolerance**, and the **Unit** as desired. When finished click **OK**. This will alter the results of the query according to your new input.

13 – Working with the Routes and Sections to LRS Command

This section provides an overview of the **Routes and Sections To LRS** command. This command is used to create an LRS query recordset that can be used by other transportation processes. It combines elements from ARC/INFO® Routes and Sections feature classes because neither class has all of the fields necessary for an LRS model. The output of this command is a query containing an LRS Linear feature class. This query can be used as input to any of the GeoMedia Transportation Manager commands that need an LRS linear feature class (for example, **Dynamic Segmentation** or **LRS Event Overlay**). If desired, the **Warehouse > Output to Feature Class** command can be used to permanently store the resultant LRS Linear feature class. This will facilitate validation and editing of the LRS feature class.

An ARC/INFO Route System, as served up by the ARC/INFO data server, is essentially two tables: the Routes table and the Section table. The Routes table is derived from the Section table by the following:

1. Finding all sections that make a route,
2. Finding the vertices that make up each section of the route,
3. Creating polyline geometries from the vertices, and
4. Creating a geometry collection of the section polylines to form the geometry for one row in the Routes table.

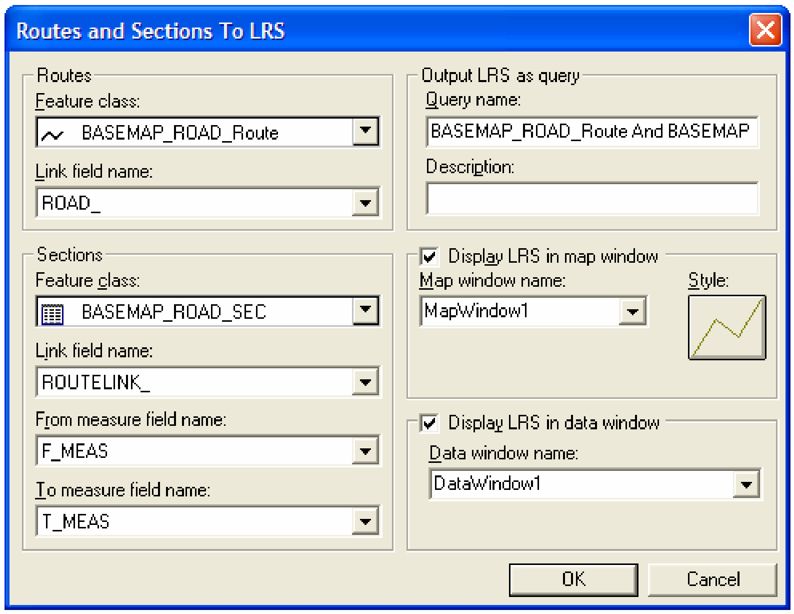
The **Routes and Sections To LRS** command builds a recordset (query) that contains all fields from the Routes table except for geometry, adds the Begin and End measure fields from the Section table, and adds a new geometry field where the geometry collection is unbundled back to individual polylines. This recordset, a combination of both the Routes and Section tables, can then be used by other GeoMedia Transportation Manager commands.

The Routes and Sections to LRS Command Workflow

This section presents the procedural steps for using the **Routes and Sections To LRS** command.

1. Open a GeoWorkspace with readable connections.
2. Select **Transportation > LRA Analysis > Routes and Sections To LRS.**

   The **Routes and Sections To LRS** *dialog box appears.*



3. From the **Feature class** drop-down list in the **Routes** box, select a Routes feature class to represent the Routes table.
4. From the **Link field name** drop-down list in the **Routes** box, select the Routes link field name, whose values link the Routes table to the Sections table.
5. From the **Feature class** drop-down list in the **Sections** box, select a Section feature class to represent the Sections table.
6. From the **Link field name** drop-down list in the **Sections** box, select a Sections link field name, whose values link the Sections table to the routes table.
7. Select a name from the **From measure field name** drop-down list. The values of this field name represent the begin distance for the section.
8. Select a name from the **To measure field name** drop-down list. The values of this field name represent the end distance for the section.
9. In the **Query name** field, accept the default query name, or type in a new name.

10. Check the **Display LRS in map window** check box if the resulting query is to be displayed in a map window, or click the **Display LRS in data window box** if the resulting query is to be displayed in a data window.

Note: You can select an existing map or data window or enter a new map or data window name, and you can set the style of the output linework.

---

11. Click **OK**.

14 – Working with the LRS and Marker Properties Dialog Boxes

This section provides an overview of the **LRS Properties** and **Marker Properties** dialog boxes**,** which provide a graphic user interface for entering LRS properties in GeoMedia Transportation Manager. For more on the subject of LRS properties and models, see the "GeoMedia Transportation LRS Data Structures" appendix

Use these two dialog boxes in the **Dynamic Segmentation**, **Insert LRS Event, Interactive LRS Calibration, LRS Calibration, LRS Event Conversion, LRS Event Generation, LRS Event Overlay**, **LRS Keys for Coordinate Events, LRS Precision Location,** and **Resolve LRS Event Overlaps** commands to enter LRS properties. These dialog boxes provide a standard method for entering this information.

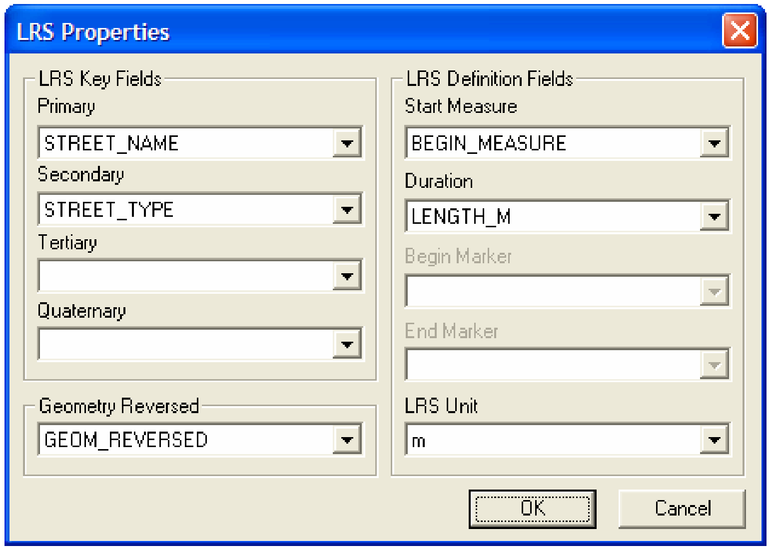
The following are two workflows, one for the LRS feature class and one for the Marker feature class.

The LRS Properties Dialog Box Workflow for an LRS Feature Class

This section presents the procedural steps for using the **LRS Properties** dialog box for an LRS feature class.

1. In the **LRS Feature Class** portion of the dialog box, click **Properties**.

   The **LRS Properties** *dialog box is displayed.*



2. In the **LRS Key Fields** box, select **Primary, Secondary, Tertiary**, and **Quaternary** keys of the LRS feature class for as many keys as you use.

3. In the **Geometry Reversed** field, select the Boolean (true or false) field name that defines whether to use each linear feature's digitizing direction as its direction (Geometry Reversed is False) or to assume that the direction of the linear feature is the opposite of its digitizing direction (Geometry Reversed is True). Use of this field is optional and, if not used, it is assumed that each linear feature's digitizing direction is its direction of increasing measures.

4. In the **LRS Definition Fields** box, select the field names of the **Start Measure** and **End Measure** from the drop-down lists if you selected a Measure model type. If you selected a Duration model type, select the field names of the **Start Measure** and **Duration** (length) from the drop-down lists.

4. If you selected an Internal Marker model type, select the field names of the **Begin Marker** and **End Marker** (End Marker is optional) from the drop-down lists.

5. In the **LRS Unit** field, select the field name of the unit for the measures of this LRS feature class.

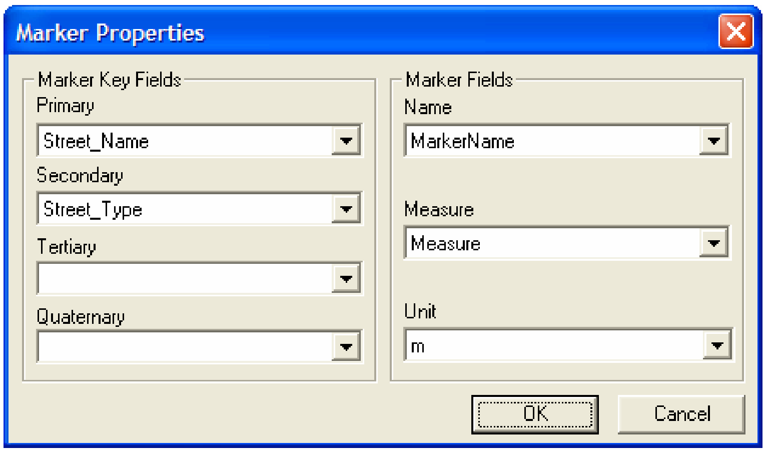
6. Click **OK** after setting the appropriate values or **Cancel** to discard your changes.

The Marker Properties Dialog Box Workflow for a Marker Feature Class

This section presents the procedural steps for using the **Marker Properties** dialog box for a Marker feature class.

1. In the **Marker Feature Class** portion of the dialog box, click **Properties**.

   The **Marker Properties** *dialog box is displayed.*



2. In the **Key Fields** box, select **Primary, Secondary, Tertiary**, and **Quaternary** keys of the Marker feature class for as many keys as you use.
3. In the **Name** field, select the field name of the marker name for the Marker Feature class.
4. In the **Measure** field, select the field name of the measure for the Marker feature class.
5. In the **Unit** field, select the field name of the unit of measure for the Marker feature class.
6. Click **OK** after setting the appropriate values or **Cancel** to discard your changes.

15 – Working with the Event Properties Dialog Box

This section provides an overview of the **Event Properties** dialog box, which provides support for setting and retrieving Event properties. For more on the subject of Event properties (and LRS models), see the "GeoMedia Transportation LRS Data Structures" appendix.

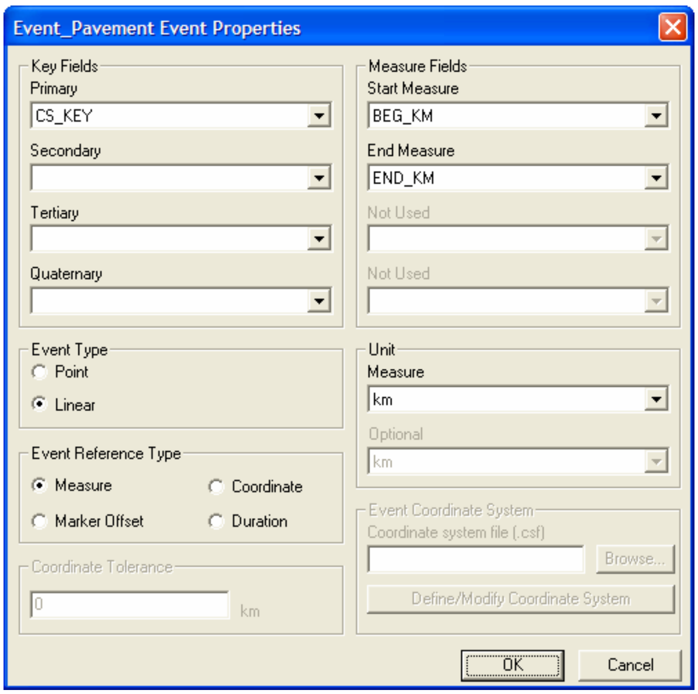
Use this dialog box in the **Dynamic Segmentation**, **LRS Event Overlay**, **Resolve LRS Event Overlaps**, **LRS Event Conversion**, and **Insert LRS Event** commands to input Event properties. This dialog box provides a standard method for entering this information.

The Event Properties Dialog Box Workflow

This section presents the procedural steps for using the **Event Properties** dialog box.

1. Click **Properties.**

   The **Event Properties** *dialog box is displayed.*



2. In the **Key Fields** box, select the **Primary, Secondary, Tertiary**, and **Quaternary** keys of the Event feature class for as many keys as you use.
3. Select the **Event Type** (**Point** is the default).
4. Select the **Event Reference Type** (**Measure** is the default)**.**
5. In the **Measure Fields** box, select from the drop-down lists the available field names that are appropriate for your selection of the **Event Type** and the **Event Reference Type**.
6. In the **Unit** box, select the field name of the unit for the event feature class (the default is the unit you set for the **Distance** on the **Units and Formats** tab of the **GeoWorkspace Coordinate System** dialog box).
7. If your **Event Reference Type** is **Coordinate**, set the coordinate system information for the selected events in the **Event Coordinate System** box. The **Define/Modify Coordinates System** button brings up the **Coordinate System** dialog box. For more on this see the "Working with the Coordinate System Dialog Box" chapter.

Note: If you want to use the coordinate system from the current GeoWorkspace, you can do this by first saving it to a *.csf* file by selecting **View > GeoWorkspace Coordinate System** from the GeoMedia Professional menu bar. This brings up the **GeoWorkspace Coordinate System** dialog box. From there select the **Save As** button and assign it a filename and location.

---

8. If your **Event Reference Type** is **Coordinate**, type in a numeric value for the coordinate tolerance in the **Coordinate Tolerance** box (the unit for this value is the one you set for the **Distance** on the **Units and Formats** tab of the **GeoWorkspace Coordinate System** dialog box).

16 – Working with the Coordinate System Dialog Box

This section provides an overview of the **Coordinate System** dialog box, which provides support for setting and retrieving coordinate system properties.

Use this dialog box in the **Dynamic Segmentation**, **LRS Event Overlay**, **Resolve LRS Event Overlaps**, **LRS Event Conversion**, **Insert LRS Event**, and **LRS Keys for Coordinate Events** commands to input coordinate system properties. This dialog box provides a standard method for entering this information.

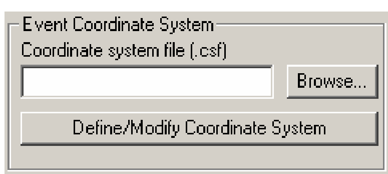
This control provides a graphic user interface for the following:

- Keying in or selecting *.csf* files
- Defining or modifying the coordinate system
- Setting or retrieving coordinate system properties

The Coordinate System Dialog Box Workflow

This section presents the procedural steps for using the **Coordinate System** dialog box.

1. Click **Browse** in the **Coordinate System** box to locate an existing coordinate system file.



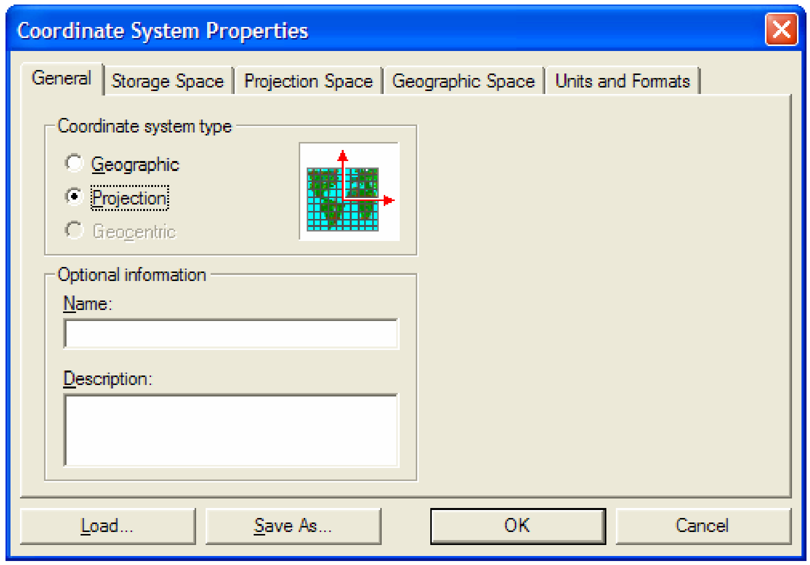
2. Select a coordinate system from the dialog box.

Note: The file type supported by this control is *.csf*.

OR

3. Click **Define/Modify Coordinate System** to define a new coordinate system file.

The **Coordinate System Properties** *dialog box is displayed.*



4. Select the **Coordinate system type**, and set or modify the coordinate system using this dialog box.

Note: See the *GeoWorkspace Coordinate System Dialog Box* topic in GeoMedia and/or GeoMedia Professional Help for more information.

On the **Coordinate System** dialog box, select the **Projection Unit** and the **Projection Quadrant** if the selected coordinate system type is Projection, or select the **Geographic Unit** and the **Geographic Quadrant** if the selected coordinate system type is Geographic.

17 – The LRS Data Preparation Workflows

This section provides recommendations for preparing an LRS feature class for use. Although the steps do not necessarily need to be taken in the exact order shown, this is the recommended order. These procedures make use of an assortment of commands from both the GeoMedia Transportation Manager product and from the core product, GeoMedia Professional.

The following data preparation steps are covered in this section:

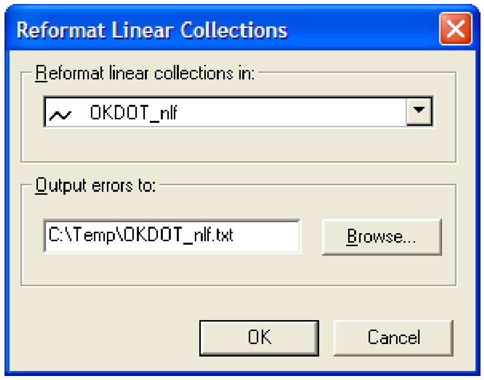
- Reformat Linear Collections
- Validate and Fix Geometry
- Validate and Fix Connectivity
- Calibrate the LRS
- Validate the LRS
- Fix the LRS
- Index the LRS

Reformat Linear Collections

In this section we will use the **Reformat Linear Collections** command to repair some of the problems common to datasets that make use of geometry collections. A geometry collection is a grouping of multiple geometries that are associated with just one record in the dataset. It sometimes happens that components of these collections can be out of order, or some of them may have incorrect digitizing directions. These errors can produce incorrect and erratic results in analysis. The **Reformat Linear Collections** command corrects these problems. For more information on the **Reformat Linear Collections** command, see the "Working with the Reformat Linear Collections Command" chapter. This workflow assumes that an LRS feature class exists.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. Select the **Reformat Linear Collections** command.

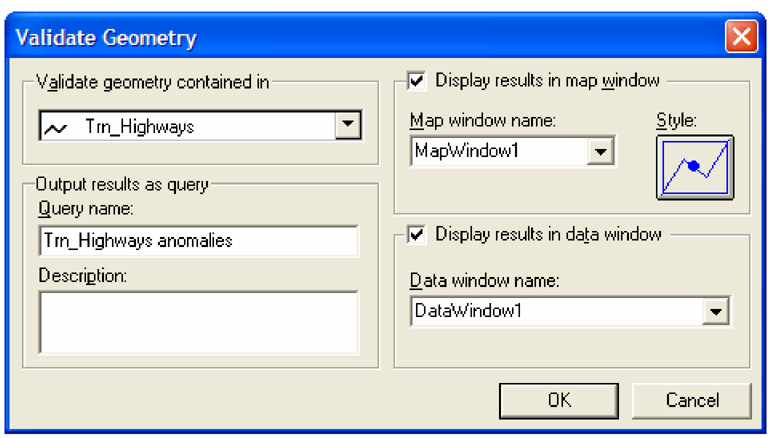
3. Under the **Reformat linear collections in** drop-down, pick your LRS feature class, and click **OK**.



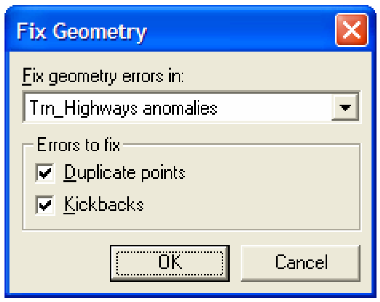
Validate and Fix Geometry

In this section you will identify geometry anomalies and correct them. These can produce incorrect and erratic results in analysis. The types of anomalies that can be found and corrected using this step are duplicate points, empty geometry collections, invalid geometries, and so on. This workflow assumes that there exists an LRS network.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. From the GeoMedia Professional menu bar, select **Tools > Validate Geometry**.
3. Under the **Validate geometry contained in** drop-down, pick your LRS feature class, and click **OK**.



4. From the GeoMedia Professional menu bar, select **Tools > Fix Geometry**.
5. Set the check boxes to fix both **Duplicate points** and **Kickbacks**, and click **OK.**

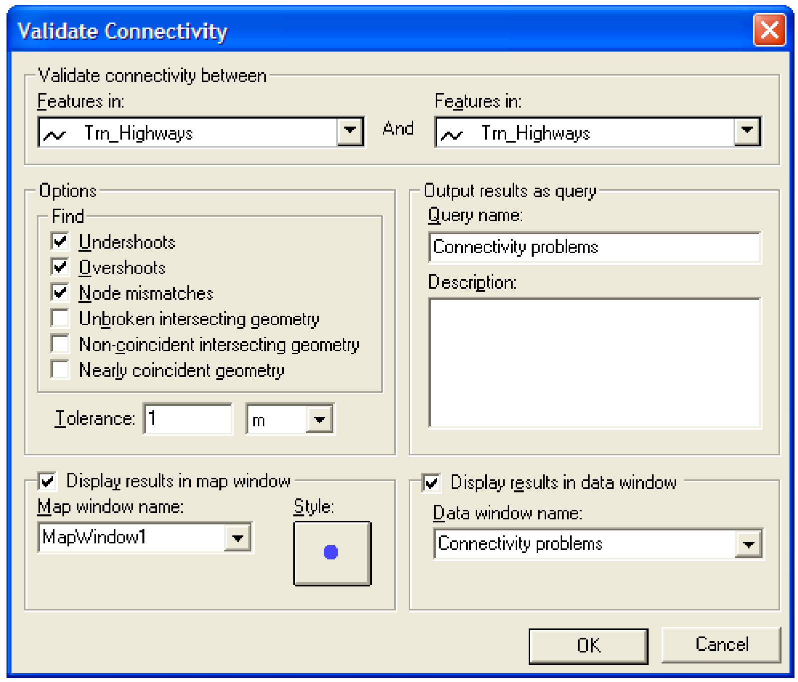


Note: The step above will remove problems from the anomalies query set as they are fixed. Any remaining problems may need to be fixed manually, usually using the **Edit Geometry** command in GeoMedia Professional.

Validate and Fix Connectivity

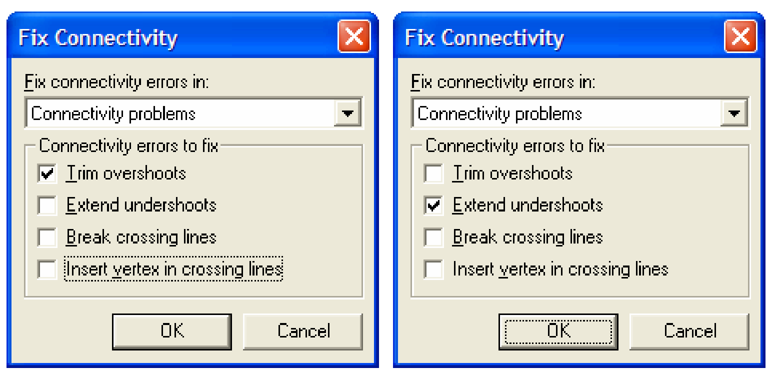
In this section you will identify connectivity problems and correct them. These problems can produce incorrect and erratic results in analysis. The types of problems that can be found and corrected using this step are undershoots, overshoots, and so on. This workflow assumes that there exists an LRS feature class.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. From the GeoMedia Professional menu bar, select **Tools > Validate Connectivity**.
3. Under both of the **Features in** drip-down lists, pick your LRS feature class, and click **OK**. Select the **Undershoots**, **Overshoots**, and **Node Mismatches** check boxes, and set the tolerance as desired. Then click **OK.**



4. From the GeoMedia Professional menu bar, select **Tools > Fix Connectivity**.

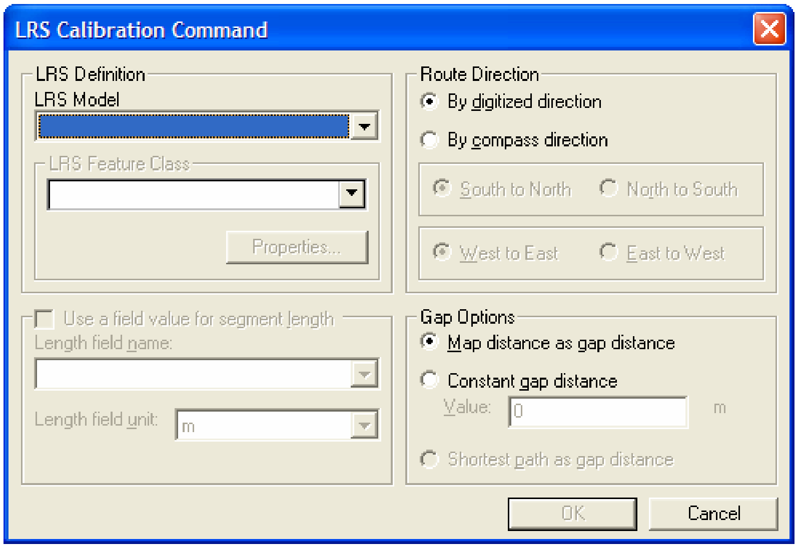
5. First, select the **Trim overshoots** check box, and click **OK.** Next, run it again, but select the **Extend undershoots** check box, and click **OK.**



Note: The step above will remove problems from the connectivity problems query set as they are fixed. Any remaining problems may need to be fixed manually, usually using the **Edit Geometry** command.

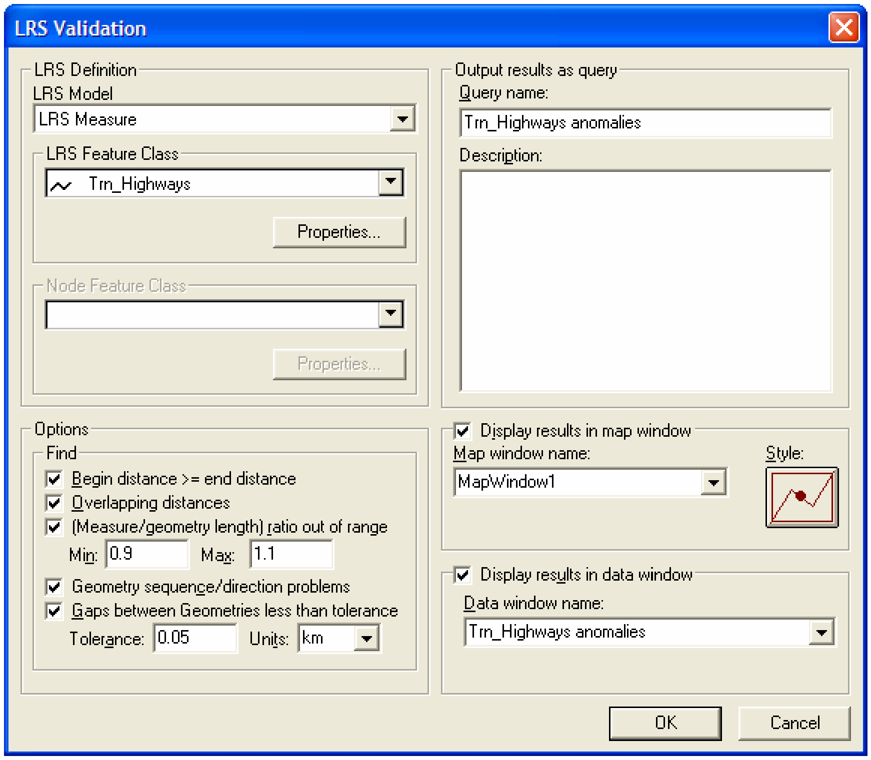
Calibrate the LRS

The next step is to add both LRS Key and measurement attribution to your LRS. This can be facilitated using either the **Interactive LRS Calibration** command or the **LRS Calibration** command. Instructions on how to use these command are shown in the "Working with the Interactive LRS Calibration Command" and "Working with the LRS Calibration Command" chapters.



Validate the LRS

The next step is to run the **LRS Validation** command, which can identify many of the LRS-specific errors that can occur. These errors can include mistakes in measurement attribution, incorrect sequencing of elements that make up a given route, errors in digitizing direction, and so on. Instructions on how to use this command are shown in the "Working with the LRS Validation Command" chapter.



Fix the LRS

There are a variety of errors that can be identified by the **LRS Validation** command, and there is no one command that corrects them all. Many of the errors that are identified by **LRS Validation** are errors in attribution, which of course can be corrected by using the **Edit > Select Set Properties** command in GeoMedia Professional to enter corrected attribution.

One of the other common errors is reversed digitizing direction (although the Calibration commands should prevent these). There is a command in GeoMedia Professional that can be used to correct these errors: the **Edit > Geometry > Reverse Direction** command. Simply select the LRS features you want to reverse, and then pick this command. This command works well in conjunction with the digitizing direction annotation workflow shown in the "LRS Annotation Workflows" chapter.

Index the LRS

For optimum performance, it is recommended that certain columns of your LRS be indexed. The LRS Key fields should be indexed as well as the

Begin Measure attribute. If Internal Marker fields are used, these should be indexed as well. If an External Marker table is used, the Marker Name and Measure columns of that table should be indexed.

18 – The LRS Annotation Workflows

This section provides step-by-step instructions on how to do a variety of LRS annotations. These procedures make use of an assortment of commands from both the GeoMedia Transportation Manager product and from the core product, GeoMedia Professional. The purpose of this section is to provide useful ways to add annotation that is useful in creating and maintaining a functional LRS.

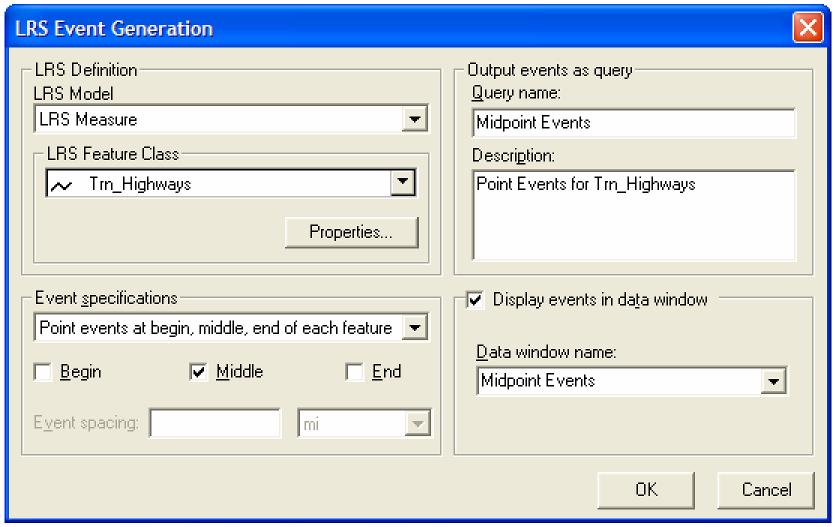
The following workflows are covered in this section:

- Annotate digitizing direction
- Annotate begin and end measurements
- Annotate even measures along the LRS

Annotate Digitizing Direction

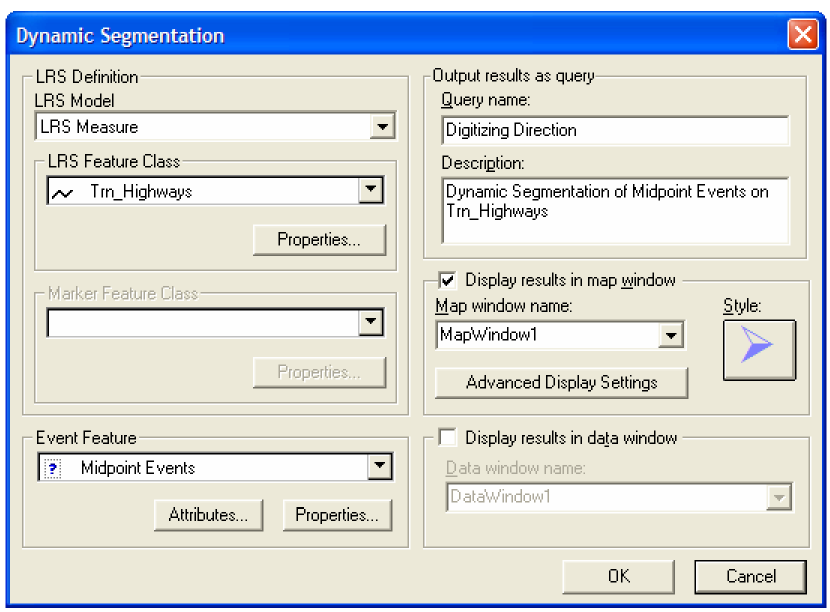
In a functioning LRS, all of the segments that make up a particular road (or pipe, and so on) need to all have the same digitizing direction. However, one cannot discern this digitizing direction just by looking at the features. The workflow in this subsection will annotate the features with an arrow that points in the digitizing direction. It is based on dynamic pipes so that if the digitizing direction is reversed, the arrow will automatically reverse as well. This workflow assumes that an LRS feature class exists.

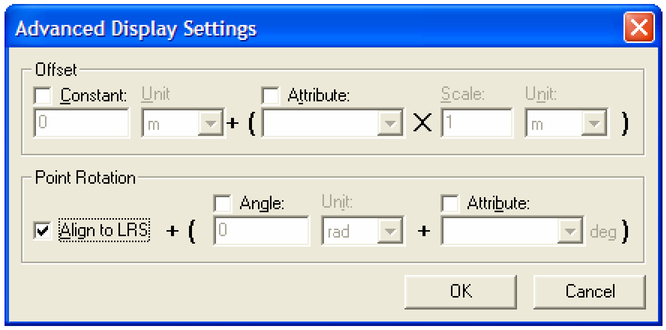
1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.
3. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create point events at the midpoint of each LRS feature.



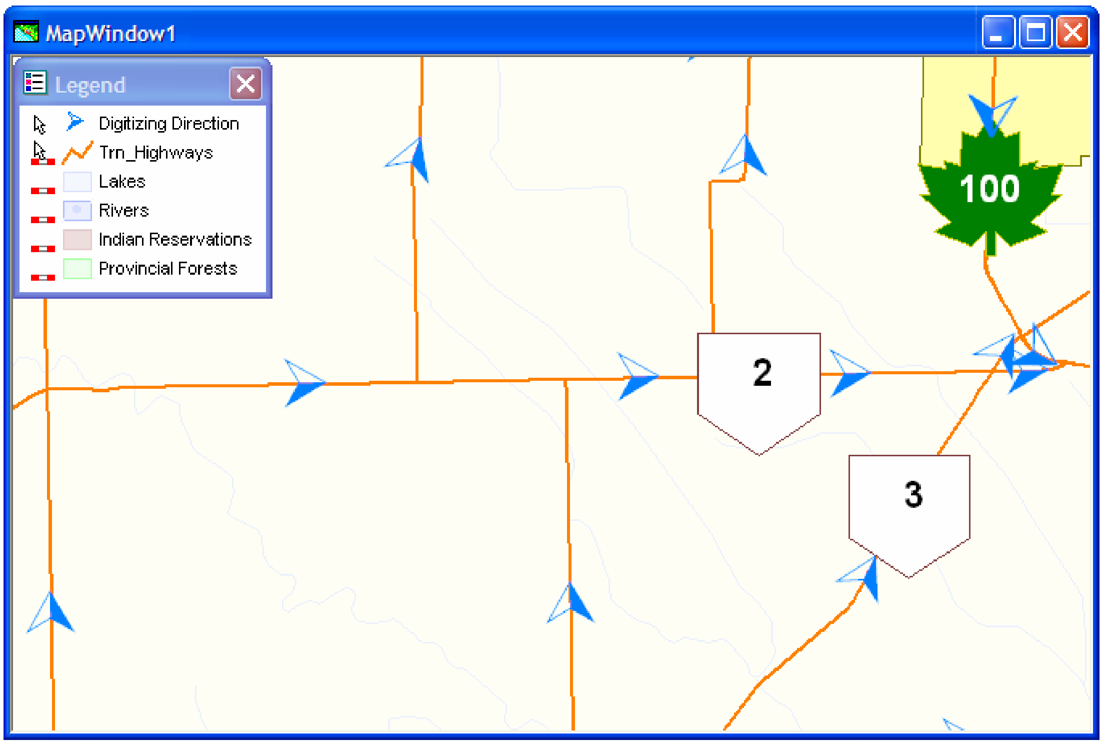
4. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

5. Use the **Dynamic Segmentation** command as described in the “Linear Referencing” chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the accident point events. Pick point symbology that will show directionality. In this example, we picked an arrow from the Wingdings font. Also make sure, under the **Advanced Display Settings**, to set the rotation to be aligned to the LRS.





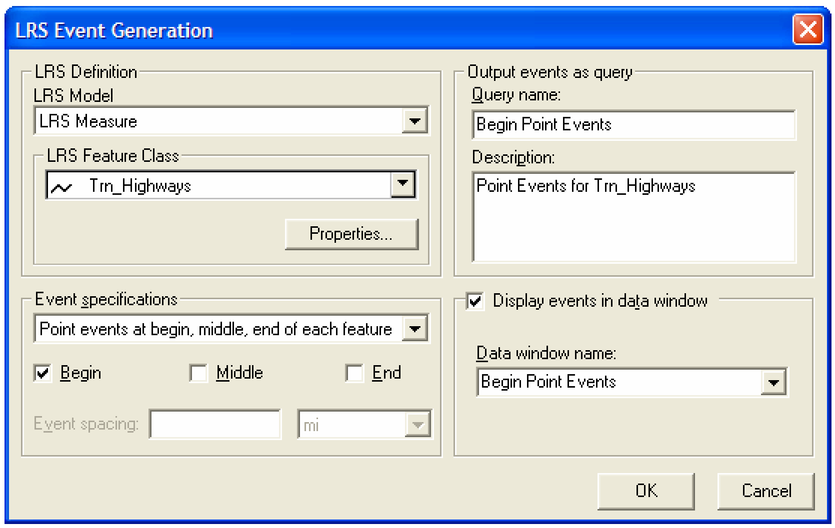
6. When the results are returned, the GeoWorkspace will resemble the following:



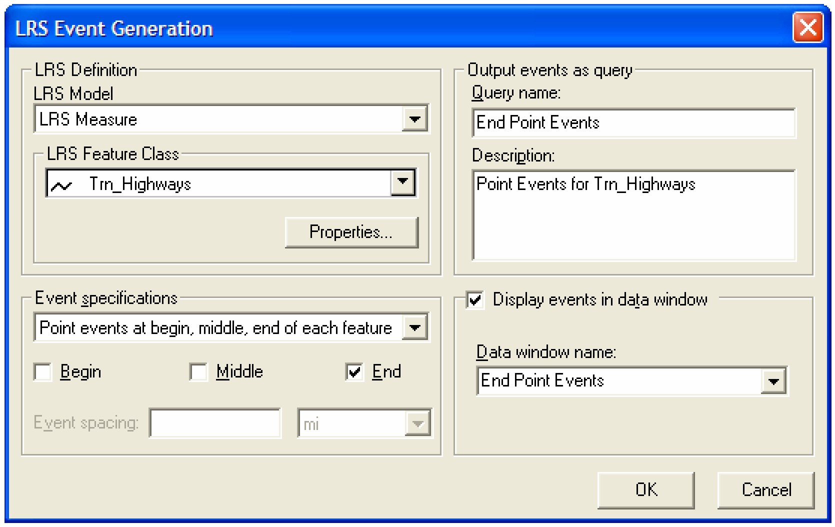
Annotate Begin and End Measurements

In building and maintaining an LRS, it is often convenient to be able to see the measurement attributes at the beginning and end of each LRS feature. In the workflow in this subsection we will annotate the begin and end measures of each these features with distinct symbology. It is based on dynamic pipes, so if the measurement attribution is changed, the annotation will change as well. This workflow assumes that there exists an LRS feature class.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.
3. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create point events at the beginning of each LRS feature.

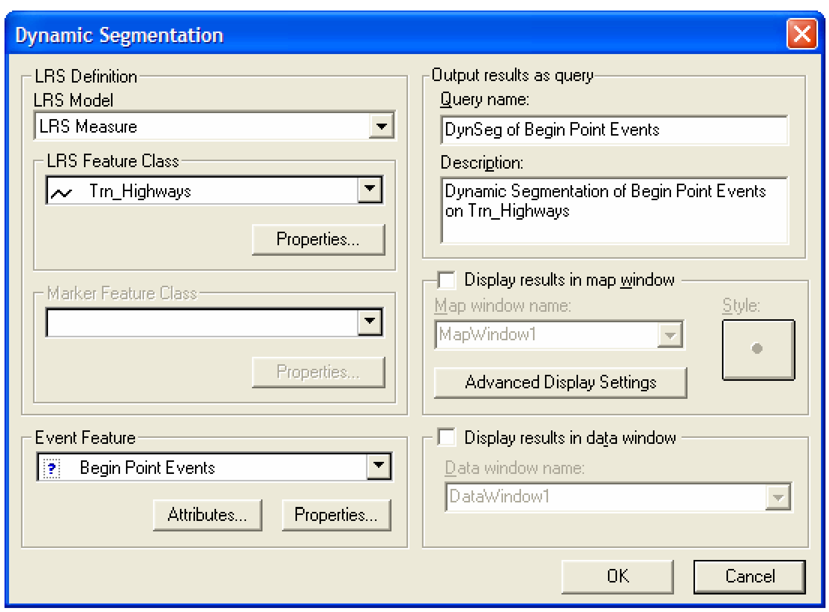


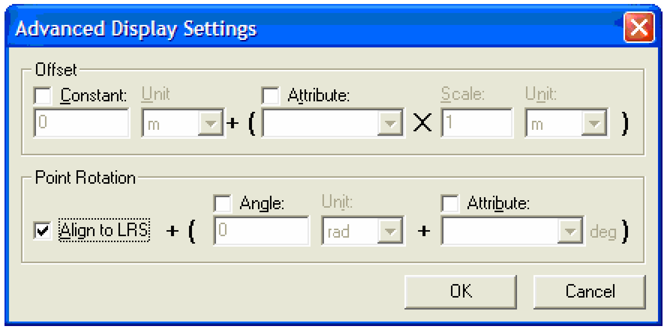
4. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.
5. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create point events at the end of each LRS feature.



6. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

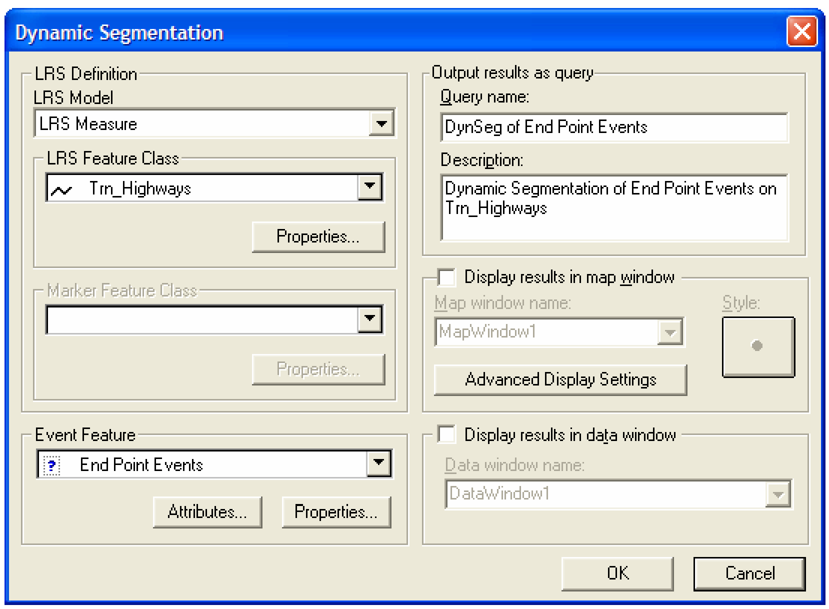
7. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the begin measure point events. Make sure, under the **Advanced Display Settings**, to set the rotation to be aligned to the LRS.

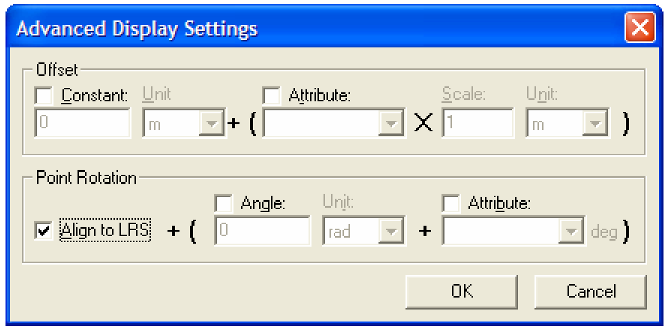




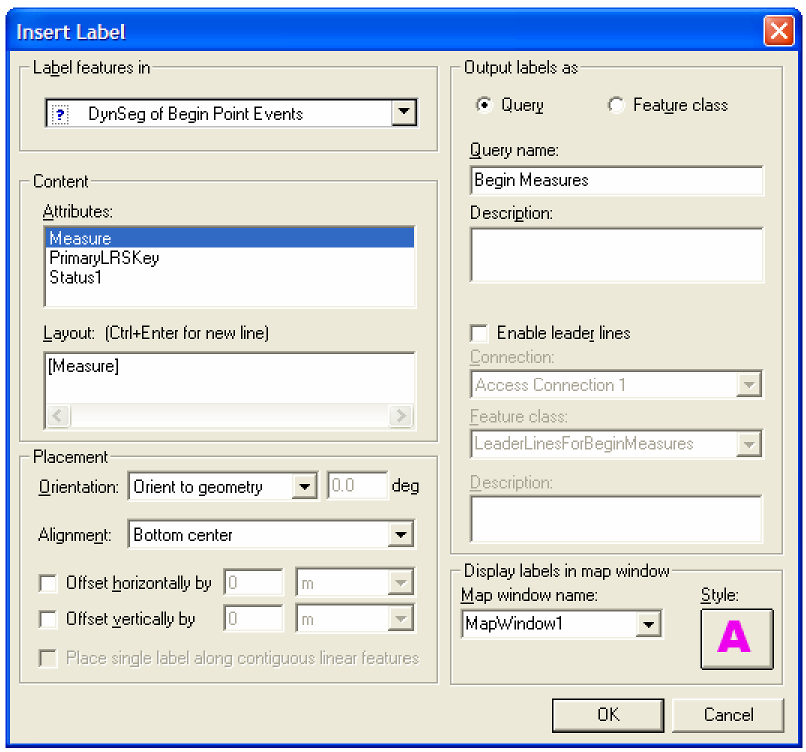
8. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

9. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the end measure point events. Make sure, under the **Advanced Display Settings**, to set the rotation to be aligned to the LRS.

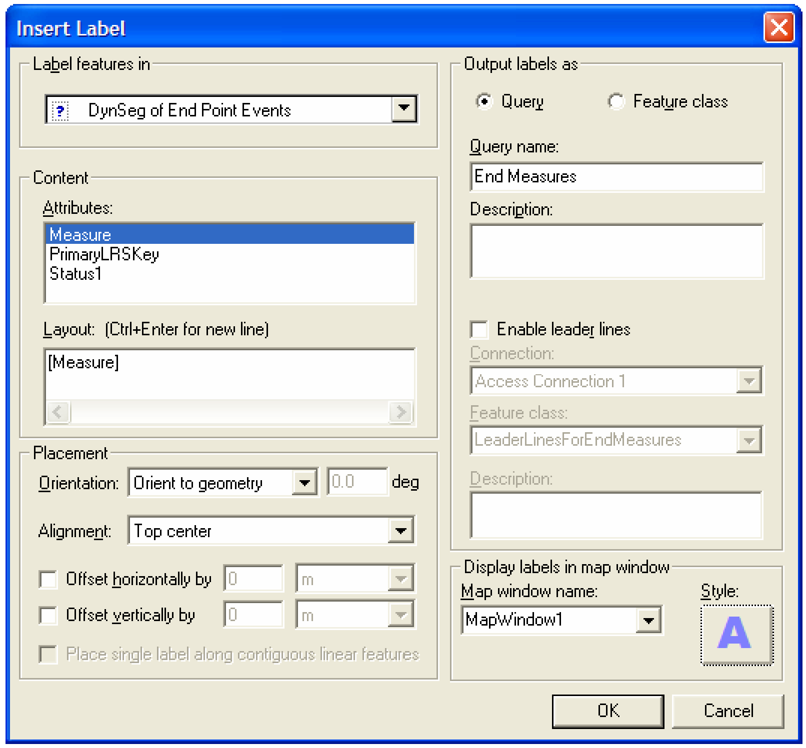




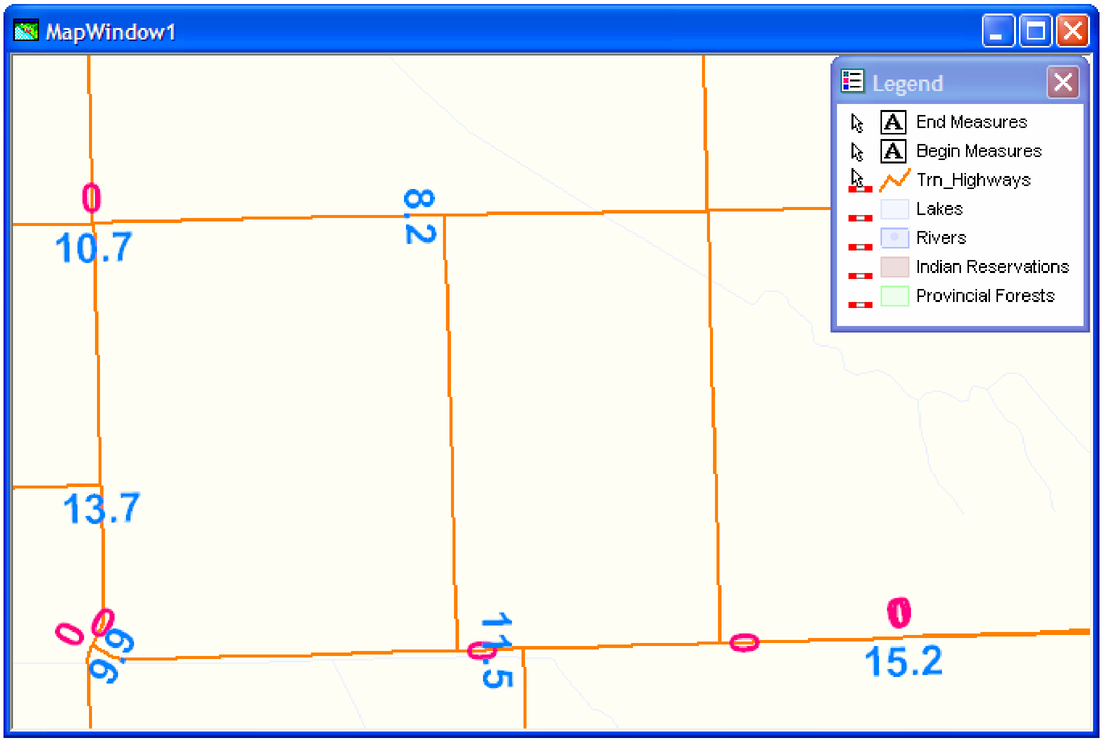
10. From the GeoMedia Professional menu bar, select **Insert > Label** and choose settings similar to those shown here.



11. From the GeoMedia Professional menu bar, select **Insert > Label** and choose settings similar to those shown here.



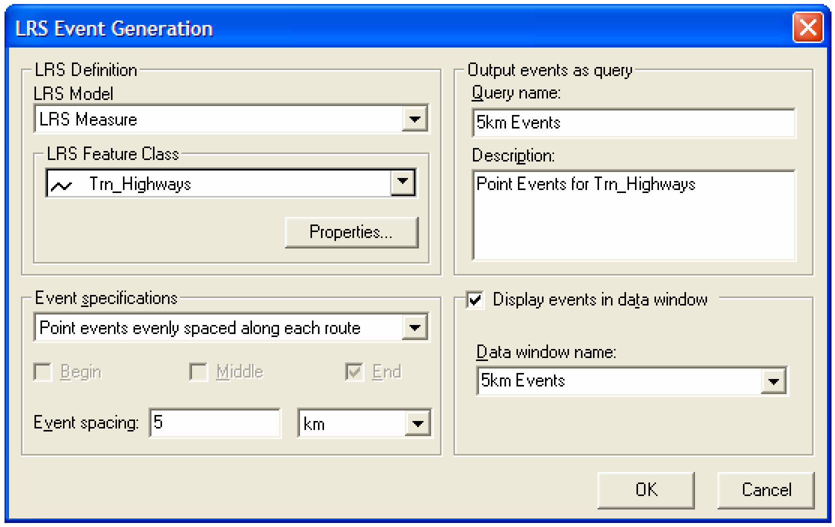
12. When the results are returned, the GeoWorkspace will resemble the following:



Annotate Even Measures Along the LRS

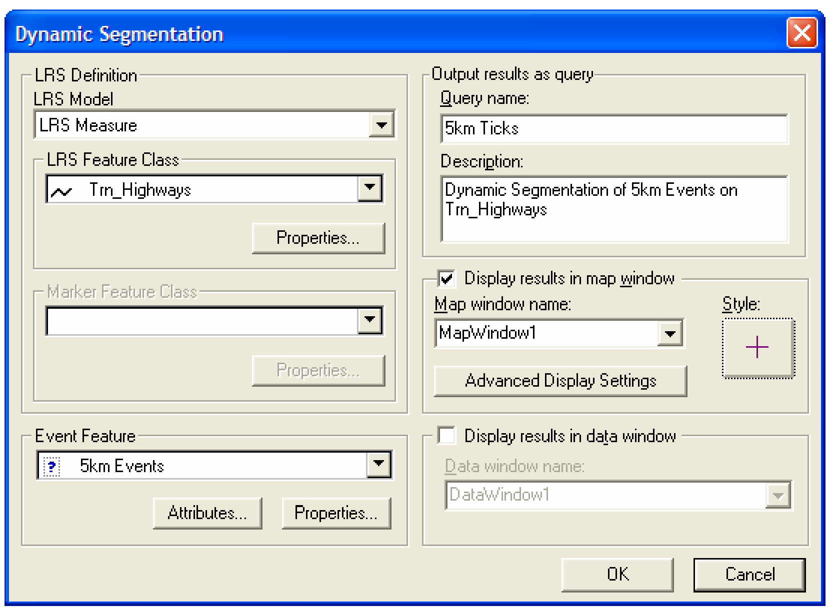
Another useful type of annotation used in LRS maintenance is marking evenly spaced measures along the LRS. In the workflow in this subsection, we will annotate these evenly spaced measures with both a symbol and a label. These symbols and labels are based on dynamic pipes, so if the measurement attribution is changed, the annotation will change as well. This workflow assumes that an LRS feature class exists.

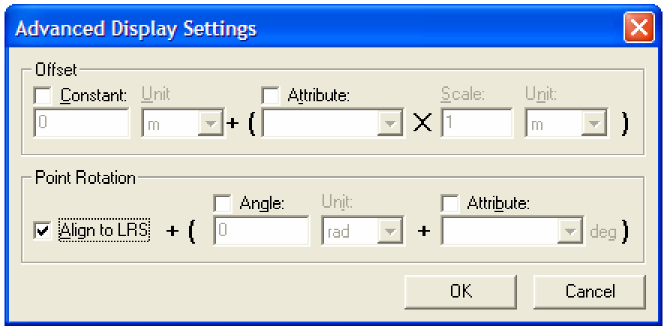
1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network.
2. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.
3. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create point events at evenly spaced measures along the LRS.



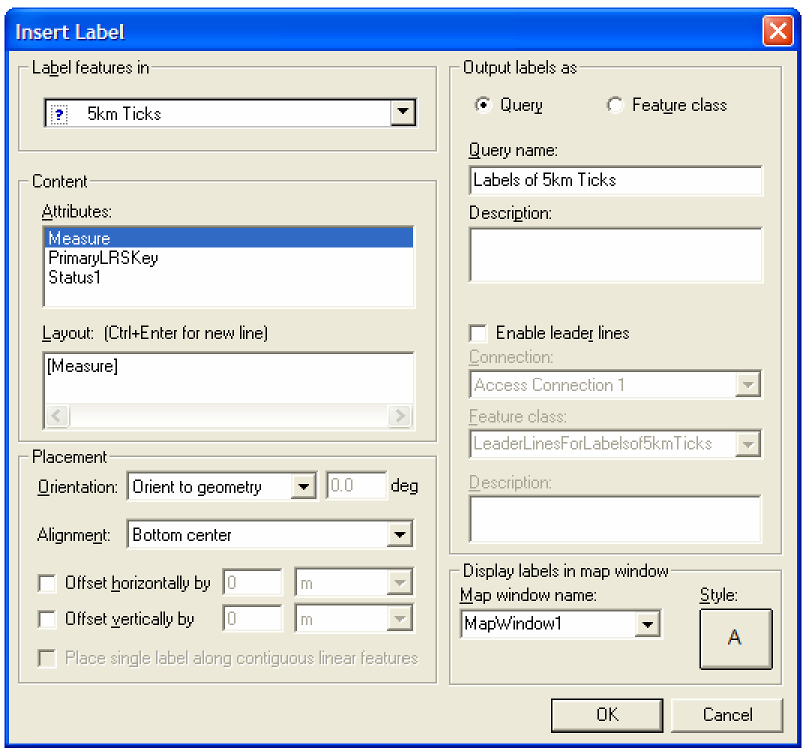
4. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

5. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the evenly spaced point events. Make sure, under the **Advanced Display Settings**, to set the rotation to be aligned to the LRS.

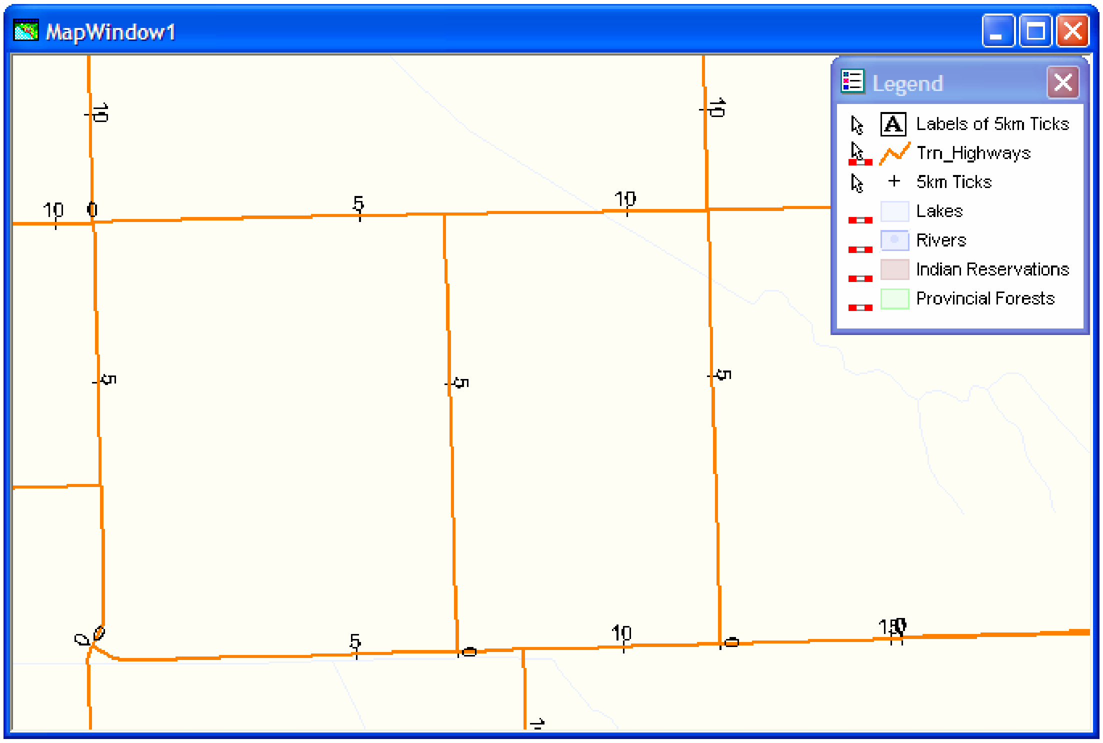




6. From the GeoMedia Professional menu bar, select **Insert > Label** and choose settings similar to those shown here.



7. When the results are returned, the GeoWorkspace will resemble the following:


MapWindow1
Legend
Labels of 5km Ticks
Trn_Highways
5km Ticks
Lakes
Rivers
Indian Reservations
Provincial Forests

19 – The LRS Analysis Workflows

This section provides step-by-step instructions on how to do a variety of LRS analyses. These procedures make use of an assortment of commands from both the GeoMedia Transportation Manager product and from the core product, GeoMedia Professional. The purpose of this section is to provide useful ways to turn an often overwhelming amount of linear referenced data into meaningful, readily usable information.

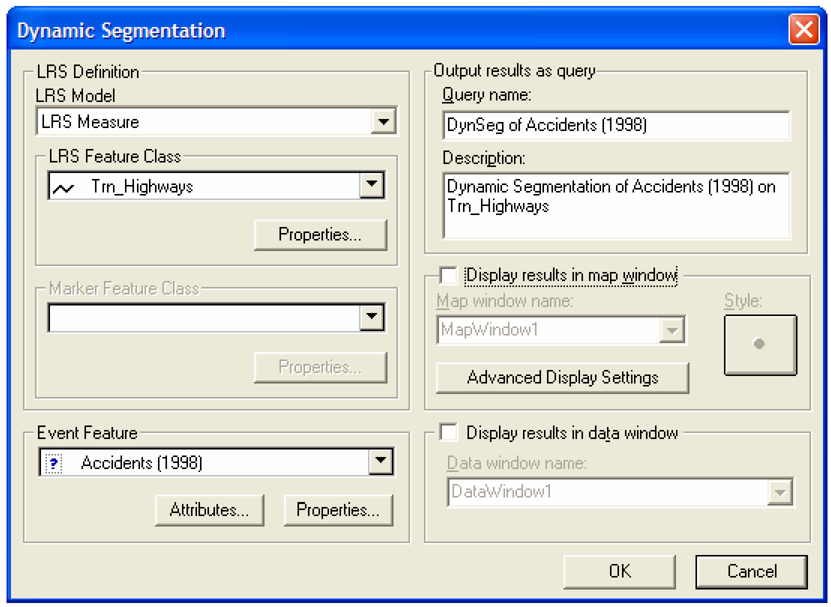
The following workflows are covered in this section:

- Aggregate point event data onto linear segments
- Aggregate point event data onto points by proximity
- Aggregate linear event data within boundaries
- Aggregate segment event data onto linear segments by proportion

Aggregate Point Event Data onto Linear Segments

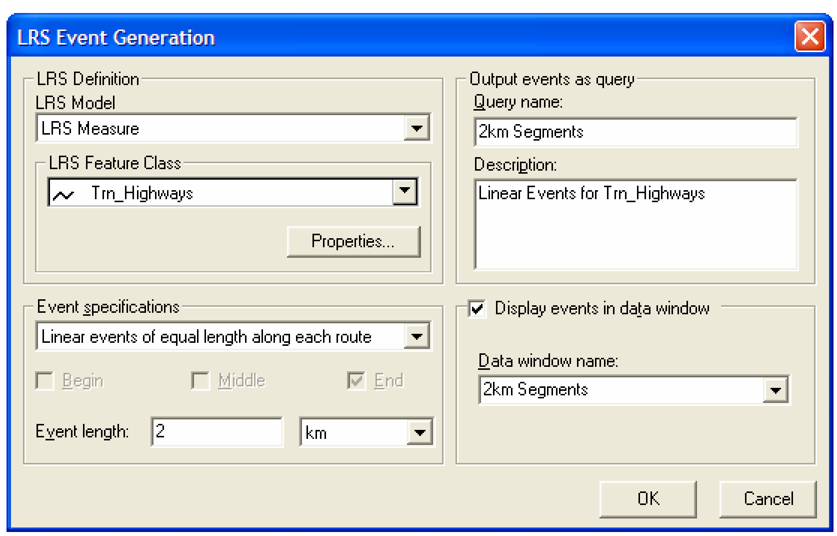
This workflow makes use of source point event data to identify *hot spots* that may represent areas requiring special attention. In the example used, we are locating areas along a transportation network with an unusually high density of accidents. It is assumed that there exists both an LRS feature class and accident point-event data.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network. If the event feature class is in a different warehouse, make a connection to that warehouse also.
2. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.
3. Use the **Dynamic Segmentation** command as described in the “Linear Referencing” chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the accident point events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.



4. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.

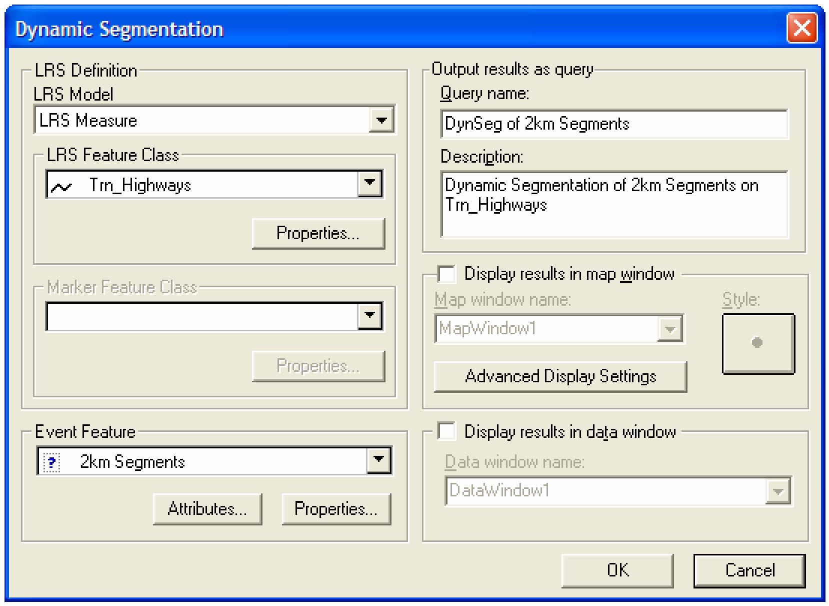
5. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create equal length linear events along each route.



6. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

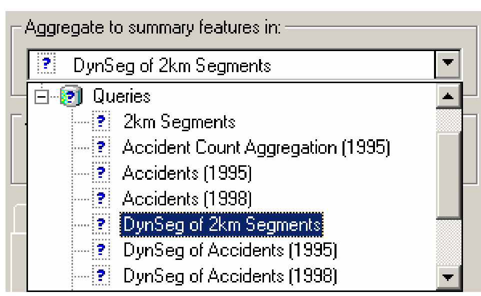
7. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with*

Geomedia Professional to create a graphic feature class for the equal length linear events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.

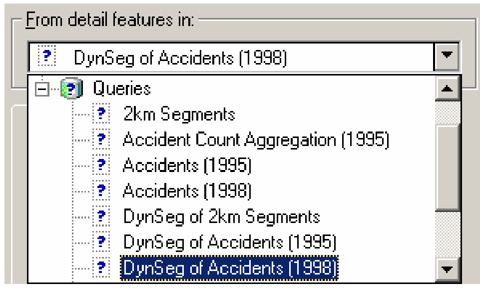


8. From the GeoMedia Professional menu bar, select **Analysis > Aggregation**.

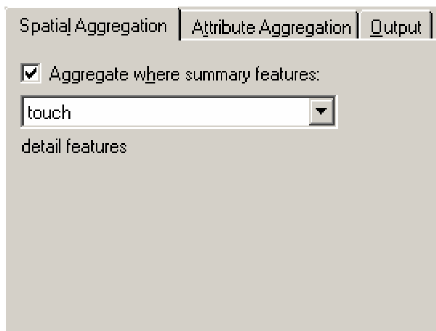
9. For the **Aggregate to summary features in** drop-down list, select the dynamically segmented equal length linear events. This feature class is referred to as the *input* class.



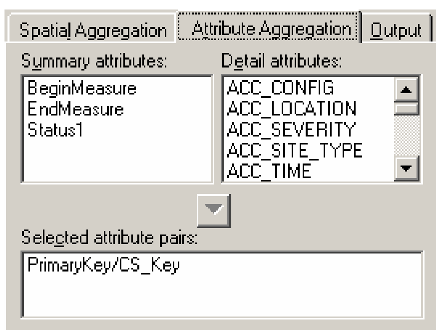
10. From the **From detail features in** drop-down list, select the dynamically segmented Accident point events. This feature class is referred to as the *detail* class.



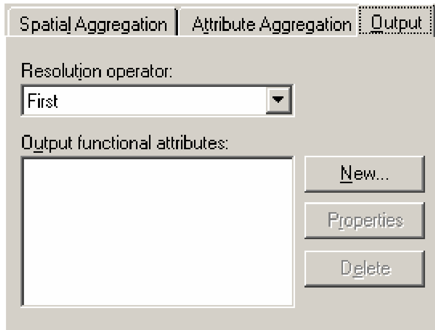
11. Under the **Spatial Aggregation** tab, pick the **touch** operator.



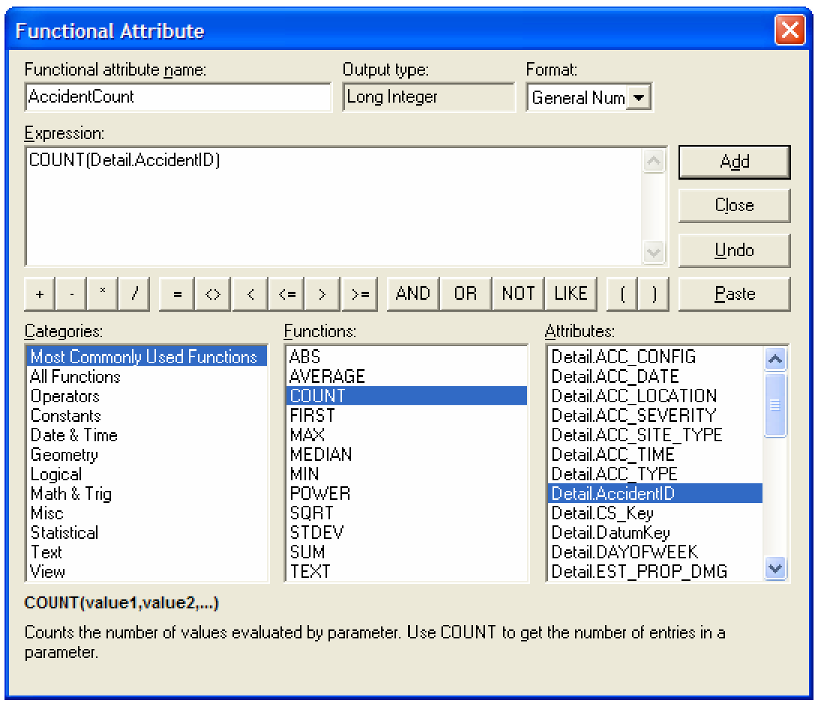
12. Under the **Attribute Aggregation** tab, pick the LRS Keys from each list. This will ensure, at intersections, that the accidents are counted towards the correct road.



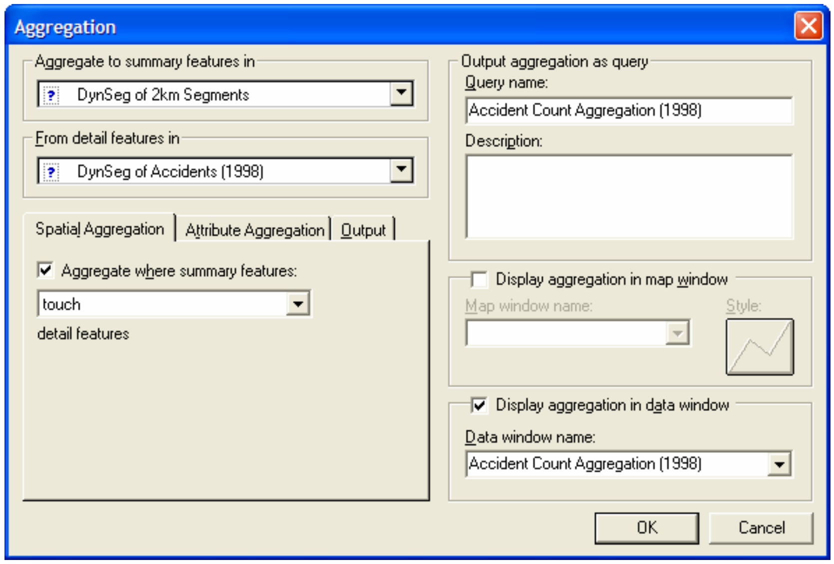
13. Under the **Output** tab, set the **Resolution Operator** to **First**. This means, in our example, that if an accident is located right at the split between two linear features, it will only be counted once and will be counted towards the first of these two linear features that is found in the warehouse.



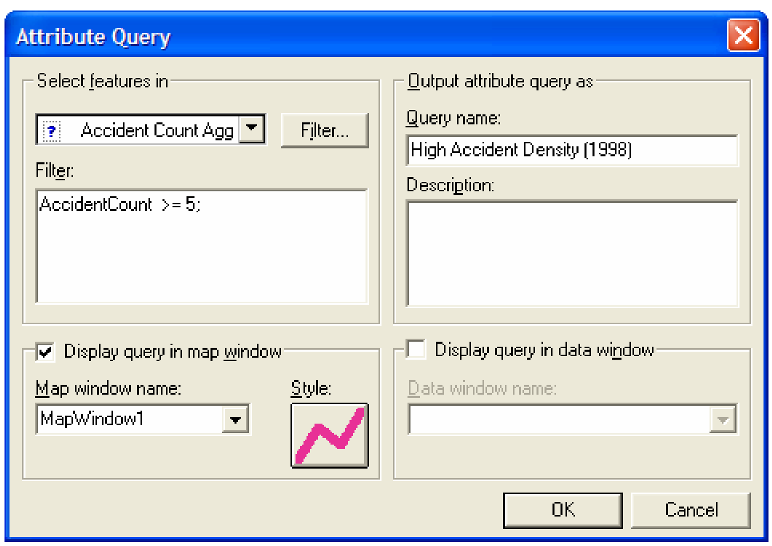
14. Click the **New** button under the **Output** tab to bring up the **Functional Attribute** dialog box. In the following picture we have created a count of the accidents.



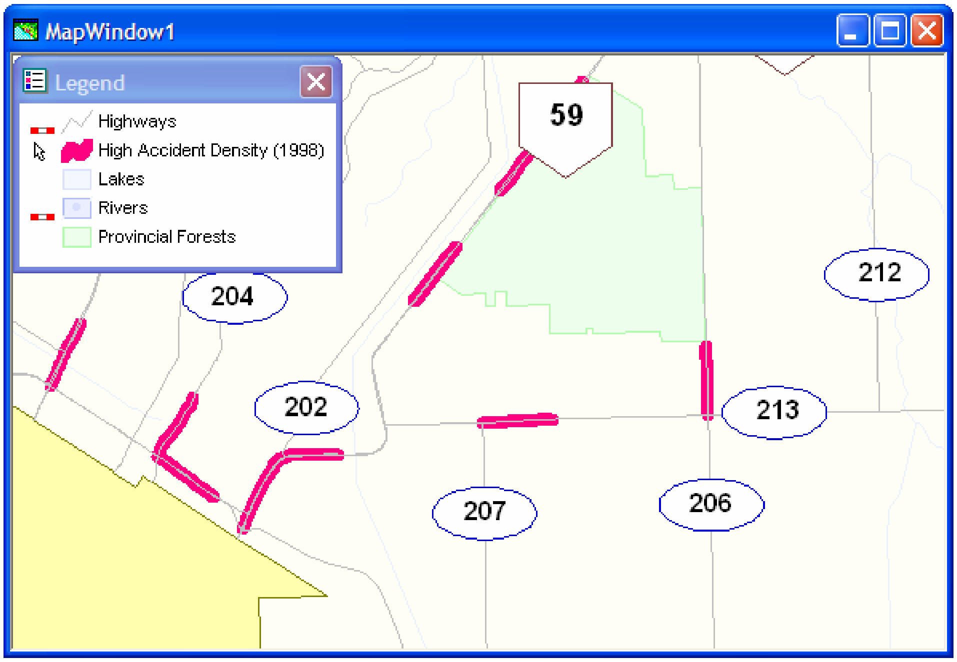
15. After filling in the **Functional Attribute** dialog box and dismissing requests to create additional functional attributes, we are returned to the **Aggregation** dialog box. In this case we will not display the results in the map window. Click **OK** to process the aggregation.



16. From the GeoMedia Professional menu bar, select **Analysis > Attribute Query**. For the feature class, choose the results of the last step. For the **Filter**, choose something that locates the most egregious situations. In the example below we are picking 2-km segments that had five or more accidents in the year in question.



17. When the results are returned, the GeoWorkspace will resemble the following:



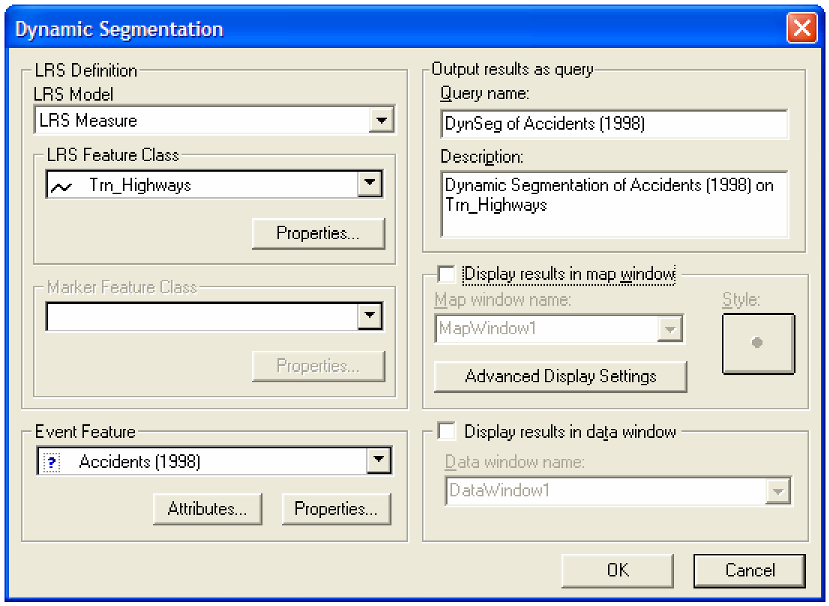
Aggregate Point Event Data onto Points by Proximity

This workflow makes use of source point event data and aggregates the data onto a separate point feature class. This technique is especially useful to identify *hot spots* that may represent areas requiring special attention. The example we are using here is accidents near intersections. We are summing up the number of accidents that occur near each intersection. In this way we can identify intersections with particularly high accident rates. It is assumed that an LRS network, an Intersection point feature class, and Accident point event data exist. For hints on how to create an Intersection point feature class, see Section 5.12 of the *GeoTrans Transportation Data Model* document.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network. If the Intersection feature class or Accident event feature classes are in different warehouses, make a connection to those warehouses also.

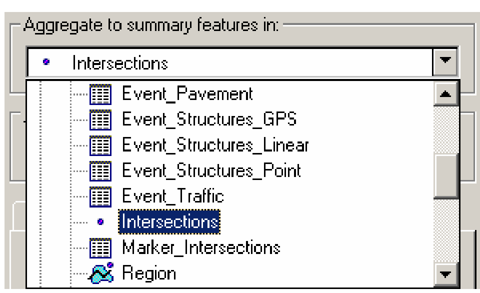
2. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

3. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the accident point events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.

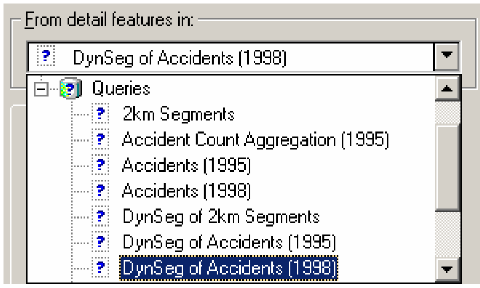


4. From the GeoMedia Professional menu bar, select **Analysis > Aggregation**.

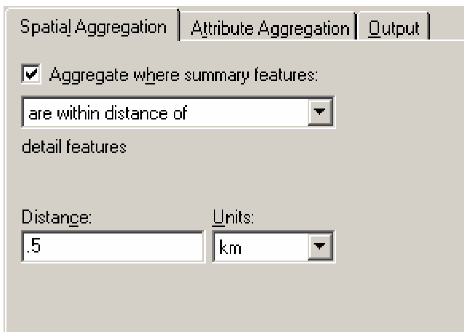
5. For the **Aggregate to summary features in** drop-down list, select the Intersection feature class. This feature class is referred to as the *input* class.



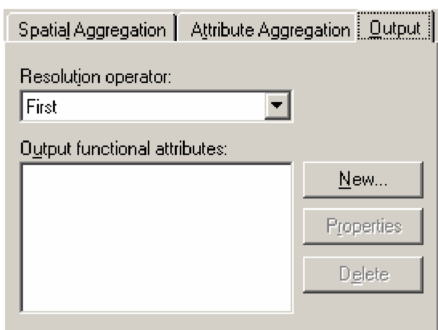
6. From the **From detail features in** drop-down list, select the dynamically segmented Accident point events. This feature class is referred to as the *detail* class.



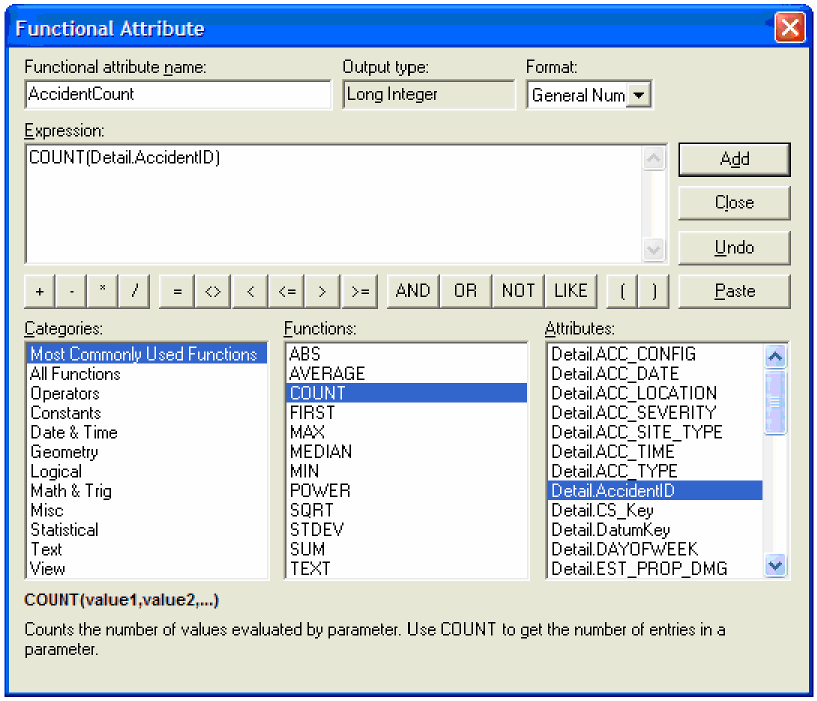
7. Under the **Spatial Aggregation** tab, pick the **are within distance of** operator.



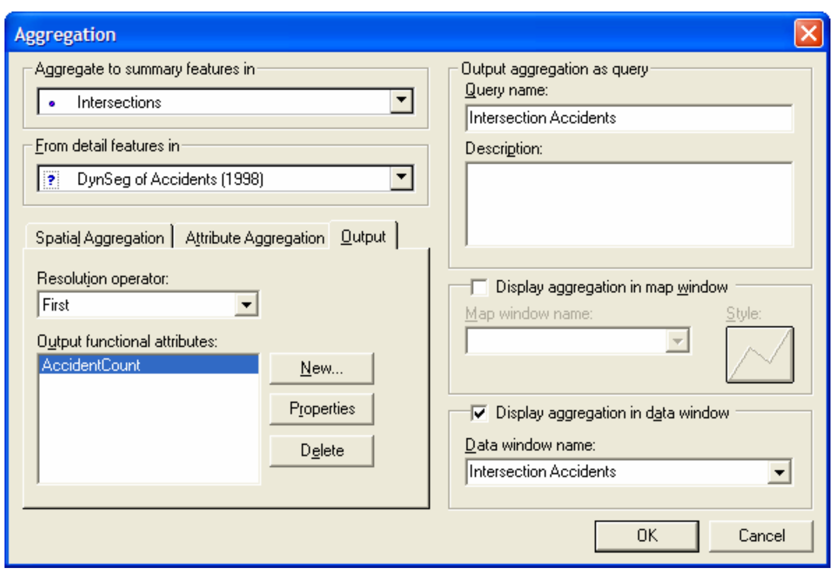
8. Under the **Output** tab, set the **Resolution operator** to **First**. This means, in our example, that if an accident falls within the distance we set of more than one intersection, that it will only be counted once. For some analyses it might also make sense to use the **All** resolution operator, in which case an accident of this type will be counted against all intersections it is close to.



9. Click the **New** button under the **Output** tab to bring up the **Functional Attribute** dialog box. In the following picture we have created a count of the accidents:

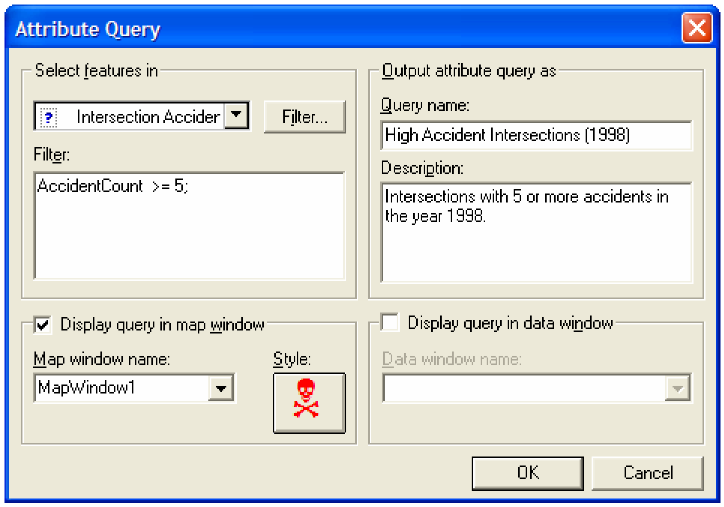


10. After filling in the **Functional Attribute** dialog box and dismissing requests to create additional functional attributes, we are returned to the **Aggregation** dialog box. In this case we will not display the results in the map window. Click **OK** to process the aggregation.

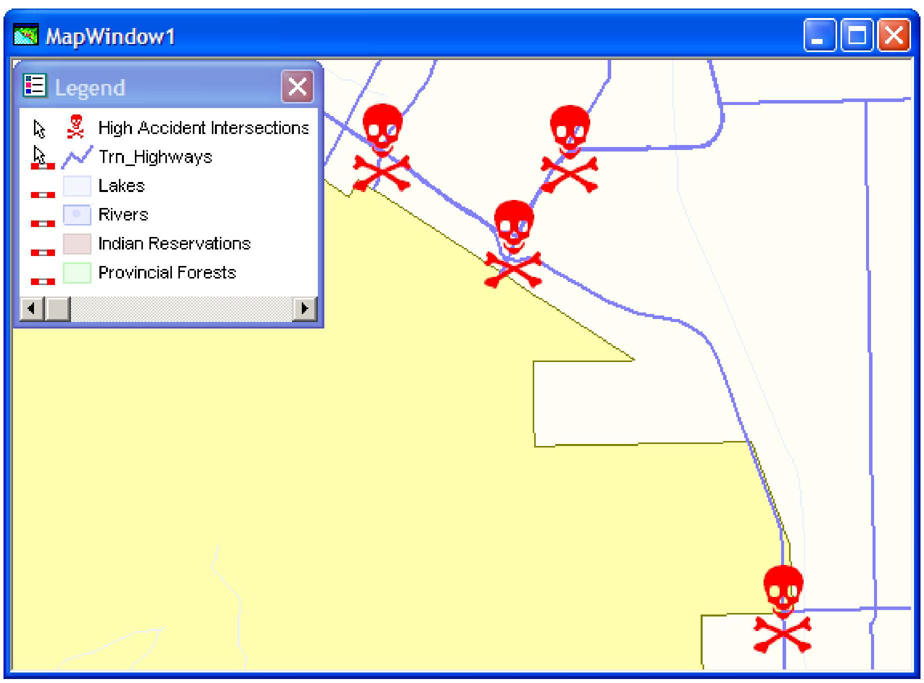


11. From the GeoMedia Professional menu bar, select **Analysis > Attribute Query**. For the feature class, choose the results of the last

step. For the **Filter**, choose something that locates the most egregious situations. In the example below, we are picking intersections that had five or more accidents in the year in question.



12. When the results are returned, the GeoWorkspace will resemble the following:



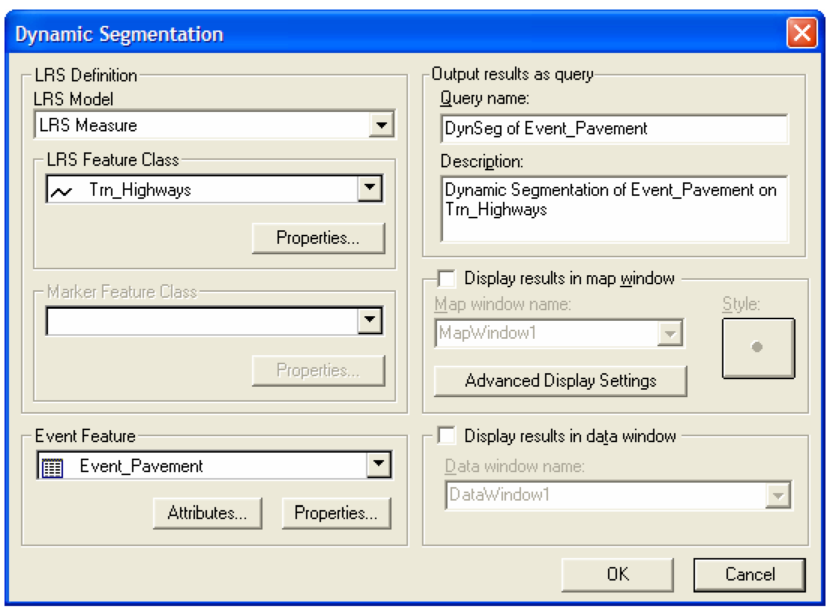
Aggregate Linear Event Data within Boundaries

This workflow makes use of source linear event data and aggregates the data onto a separate area feature class. This technique is especially useful for creating summary reports. The example we are using here is Roadway Classification event data, and we are summing up the mileage of each classification within regional boundaries. It is assumed that there exists an LRS network, a Boundary area feature class, and Road Classification linear event data.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network. If the Boundary feature class or Road Classification event feature classes are in different warehouses, make a connection to those warehouses also.

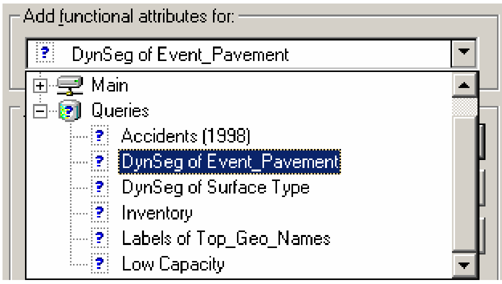
2. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

3. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the Road Classification linear events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.



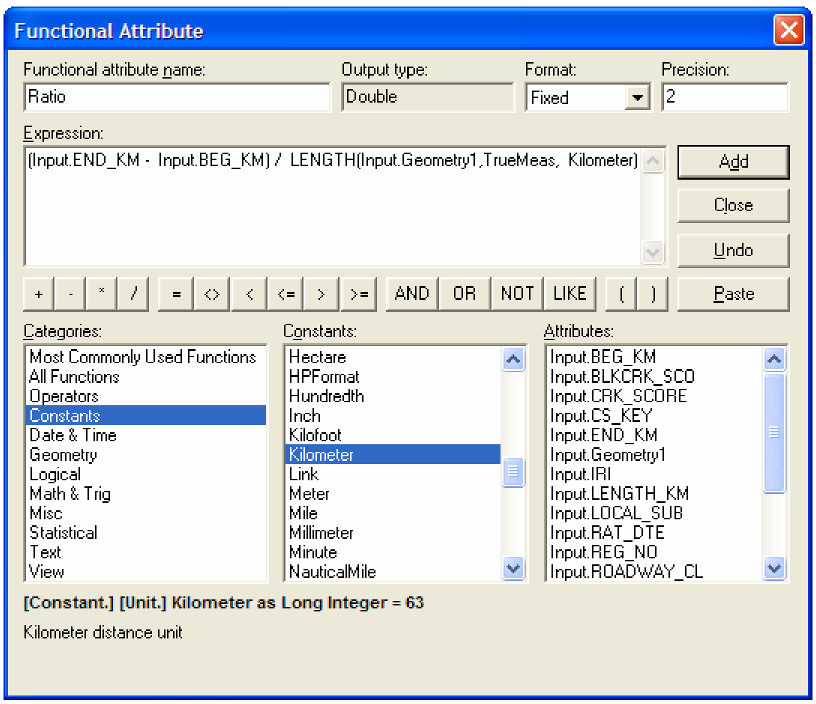
4. From the GeoMedia Professional menu bar, select **Analysis > Functional Attributes**.

5. For the **Add functional attributes for** drop-down list, select the dynamic segmentation of the Road Classification events. This feature class is referred to as the *input* class.



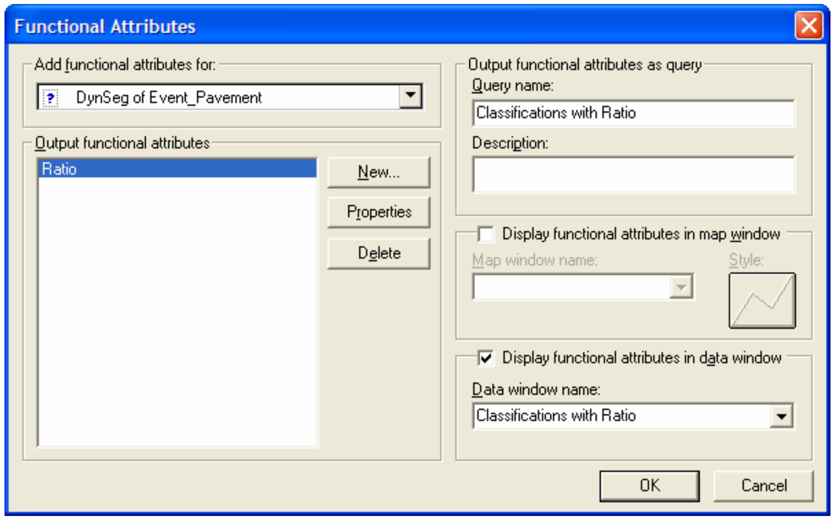
6. Click the **New** button under the **Output** tab to bring up the **Functional Attribute** dialog box. In the following picture, we have created a Ratio attribute that stores the ratio between the event length based on measurement attribution and the length based on the geometry. The formula shown below is:

```
(Input.END_KM - Input.BEG_KM) /
LENGTH(Input.Geometry1, TrueMeas, Kilometer)
```

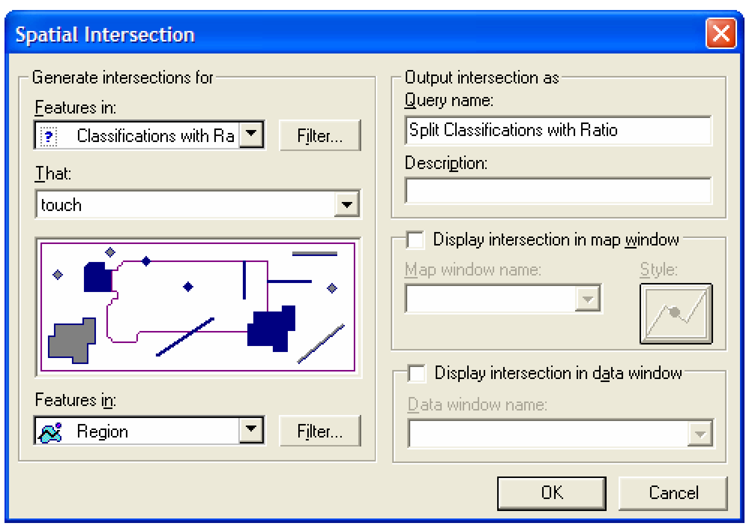


7. After filling in the **Functional Attribute** dialog box and dismissing requests to create additional functional attributes, we are returned to the main **Functional Attributes** dialog box. In this case we will not

display the results in the map window. Click **OK** to process the command.

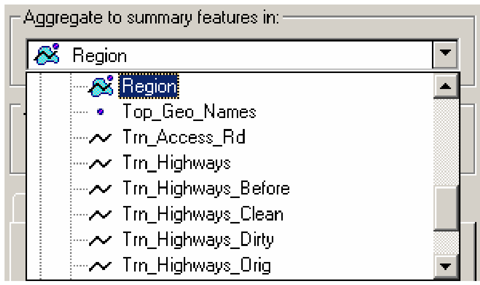


8. From the GeoMedia Professional menu bar, select **Analysis > Spatial Intersection**. For the first feature class, pick the results of the previous step. For the second feature class, pick the area boundary feature class. Select the **touch** operator. Click **OK** to process this command, which will split our roads where they cross area boundaries. This will make it easier to sum up the lengths within each boundary.

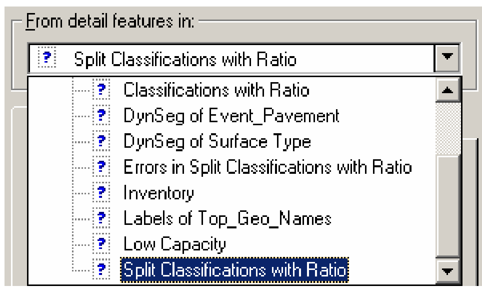


9. From the GeoMedia Professional menu bar, select **Analysis > Aggregation**.

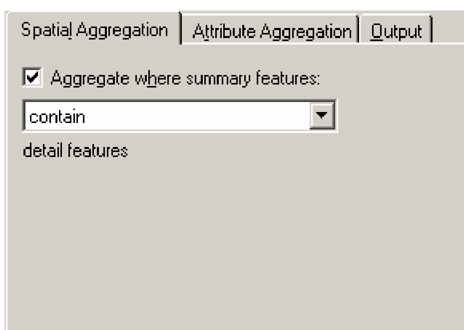
10. For the **Aggregate to summary features in** drop-down list, select the Boundary area feature class. This feature class is referred to as the *input* class.



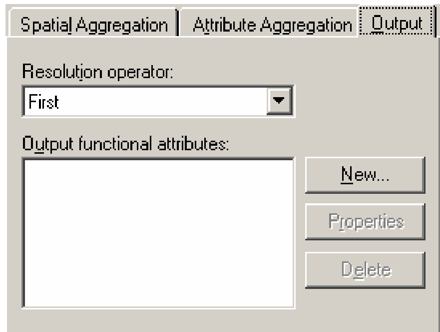
11. From the **From detail features in** drop-down list, select the results of the Spatial Intersection command. This feature class is referred to as the *detail* class.



12. Under the **Spatial Aggregation** tab, pick the **contain** operator because we are trying to aggregate the roads that are contained within each area.

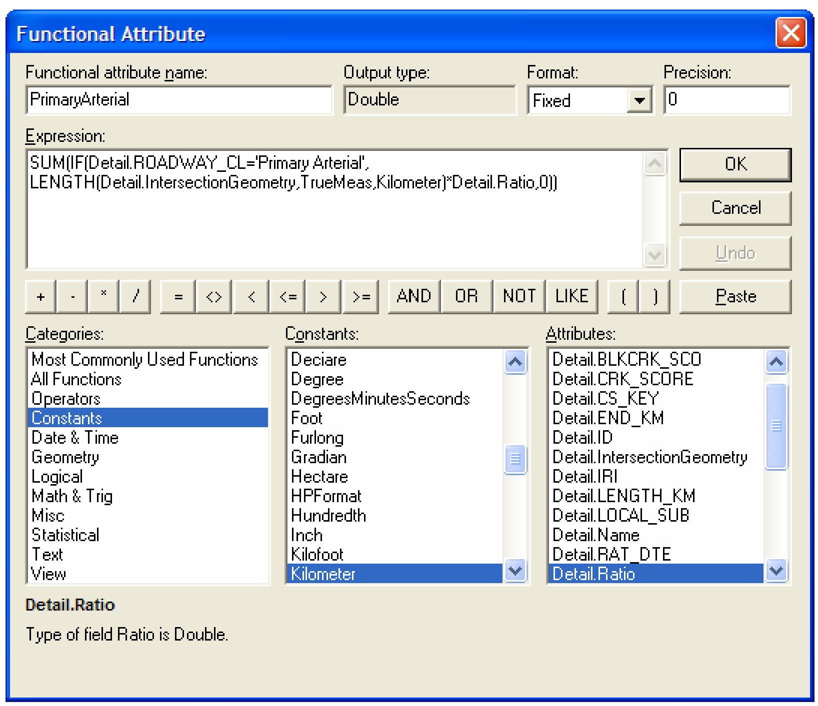


13. Under the **Output** tab, set the **Resolution Operator** to **First**. This is not terribly critical since the **Spatial Intersection** step has already ensured that each road segment can be easily assigned to one and only one area.

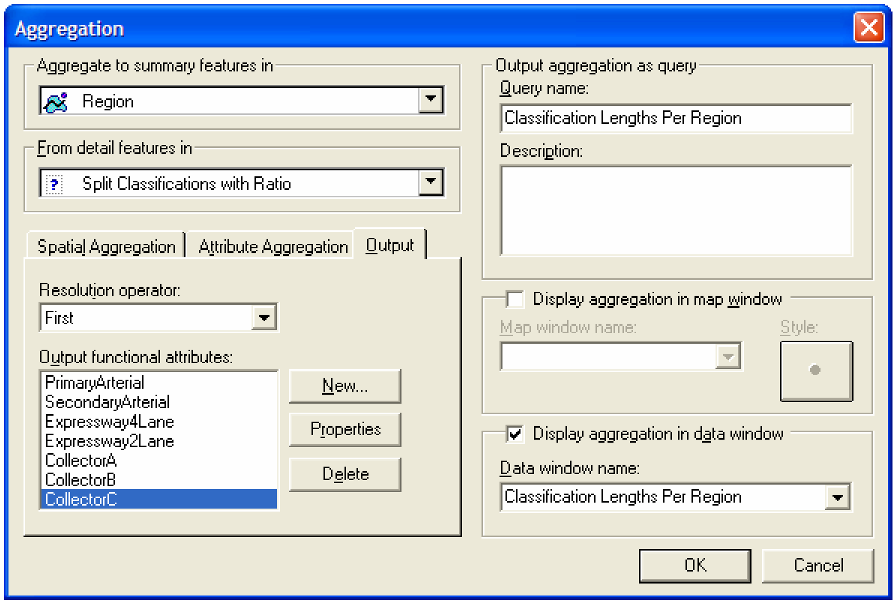


14. Click the **New** button under the **Output** tab to bring up the **Functional Attribute** dialog box. In the following picture, we have calculated the total length of roads in the Primary Arterial class. The formula shown below is:

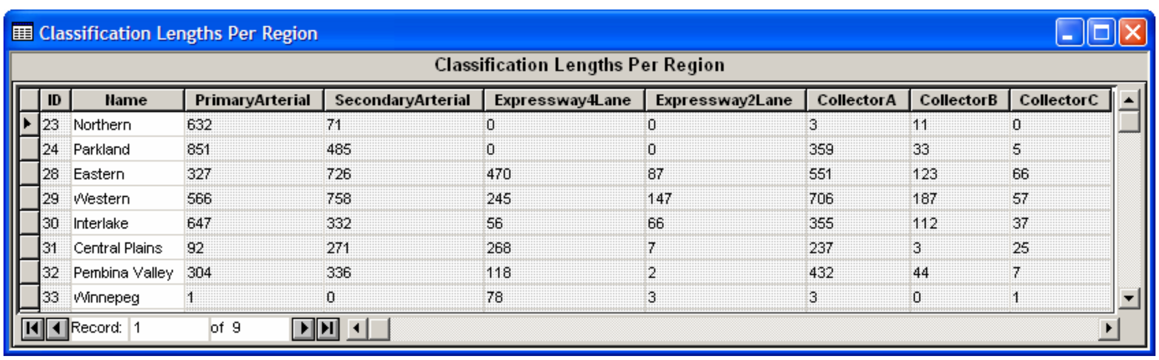
```
SUM(IF(Detail.ROADWAY_CL='Primary Arterial',
LENGTH(Detail.IntersectionGeometry, TrueMeas,
Kilometer) * Detail.Ratio,0))
```



15. After filling out the **Functional Attribute** dialog box, fill out other similar functional attributes for each roadway classification. Afterwards, dismiss requests to create additional functional attributes, and we are returned to the **Aggregation** dialog box. In this case we will not display the results in the map window. Click **OK** to process the aggregation.



16. The resultant data window will summarize the length of roads in each classification in each region.



Classification Lengths Per Region

ID	Name	PrimaryArterial	SecondaryArterial	Expressway4Lane	Expressway2Lane	CollectorA	CollectorB	CollectorC
23	Northern	632	71	0	0	3	11	0
24	Parkland	851	485	0	0	359	33	5
28	Eastern	327	726	470	87	551	123	66
29	Western	566	758	245	147	706	187	57
30	Interlake	647	332	56	66	355	112	37
31	Central Plains	92	271	268	7	237	3	25
32	Pembina Valley	304	336	118	2	432	44	7
33	Winnepeg	1	0	78	3	3	0	1

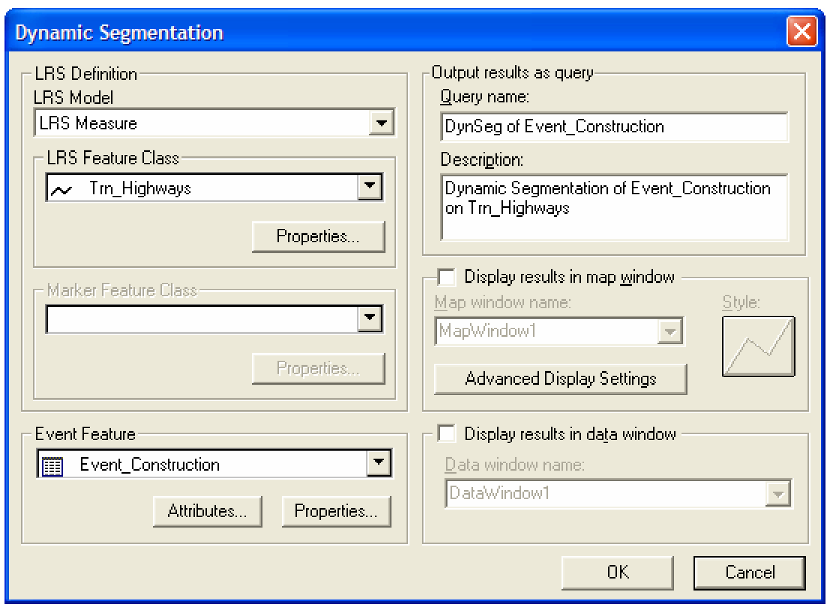
Record: 1 of 9

Aggregate Segment Event Data onto Linear Segments by Proportion

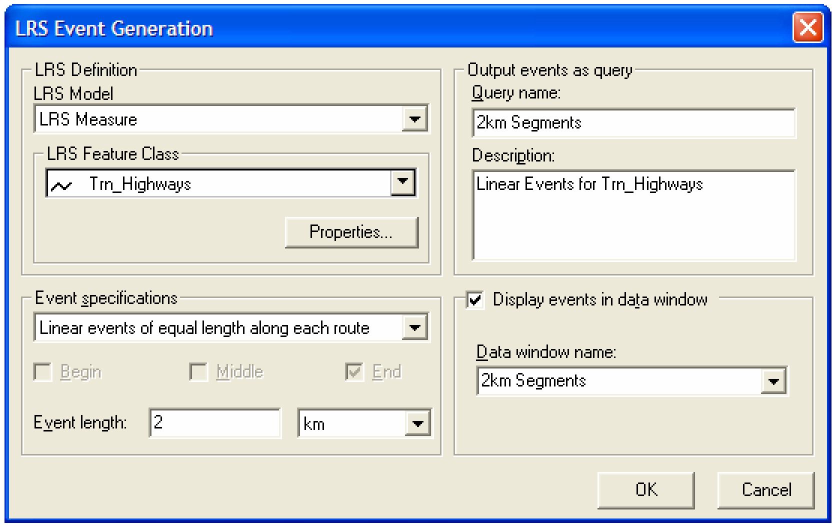
This workflow makes use of source linear event data and aggregates the data onto equal length segments. This technique is especially useful to identify *hot spots* that may represent areas requiring special attention. The example we are using here is construction project events. We are summing up the construction costs of all projects onto the underlying equal length segments. In this way we can identify areas in which construction costs have been particularly high.

It is understood that the source construction project event data might overlap several of the equal length segments and that the construction project events themselves may overlap each other. It is assumed that there exists both an LRS network and Construction Project linear event data.

1. Open a GeoWorkspace; then connect to the warehouse containing the linear feature class representing the transportation network. If the event feature classes are in a different warehouse, make a connection to that warehouse also.
2. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.
3. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the Construction Project linear events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.

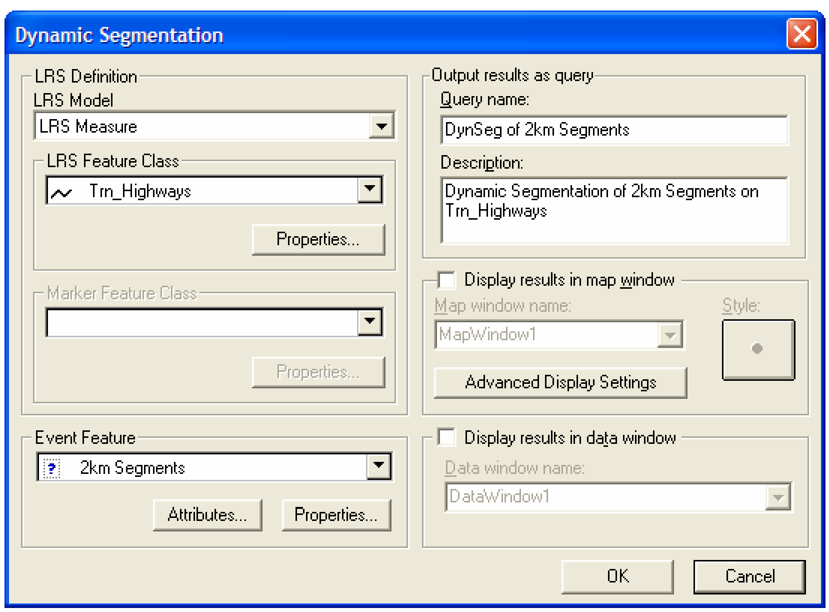


4. From the GeoMedia Professional menu bar, select **Transportation > LRS Event Generation**.
5. Use the **LRS Event Generation** command as described in the "Working with the LRS Event Generation Command" chapter to create equal length linear events along each route.



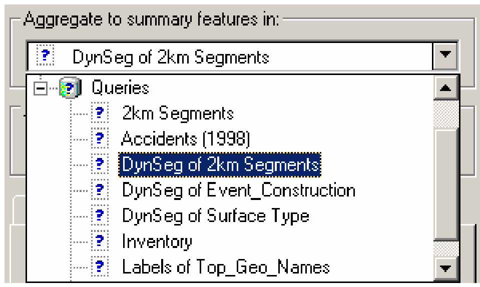
6. From the GeoMedia Professional menu bar, select **Analysis > Dynamic Segmentation**.

7. Use the **Dynamic Segmentation** command as described in the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional* to create a graphic feature class for the equal length linear events. For this analysis it is not necessary to display the results of the dynamic segmentation to the map window.

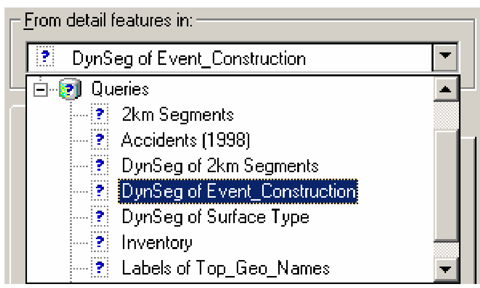


8. From the GeoMedia Professional menu bar, select **Analysis > Aggregation**.

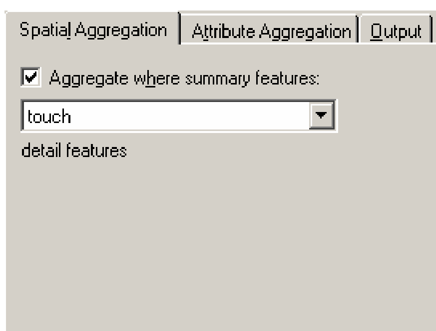
9. For the **Aggregate to summary features in** drop-down list, select the dynamically segmented equal length linear events. This feature class is referred to as the *input* class.



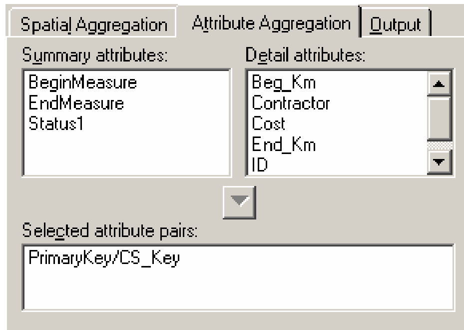
10. From the **From detail features in** drop-down list, select the dynamically segmented Pavement Type linear events. This feature class is referred to as the *detail* class.



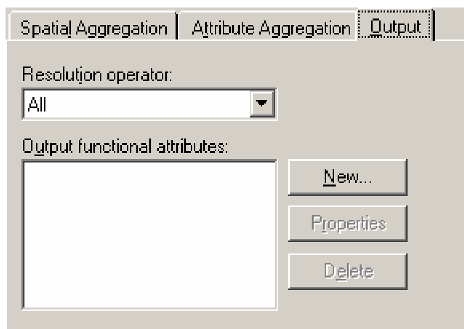
11. Under the **Spatial Aggregation** tab, pick the **touch** operator.



12. Under the **Attribute Aggregation** tab, pick the LRS Keys from each list. This will ensure, at intersections, that the accidents are counted towards the correct road.

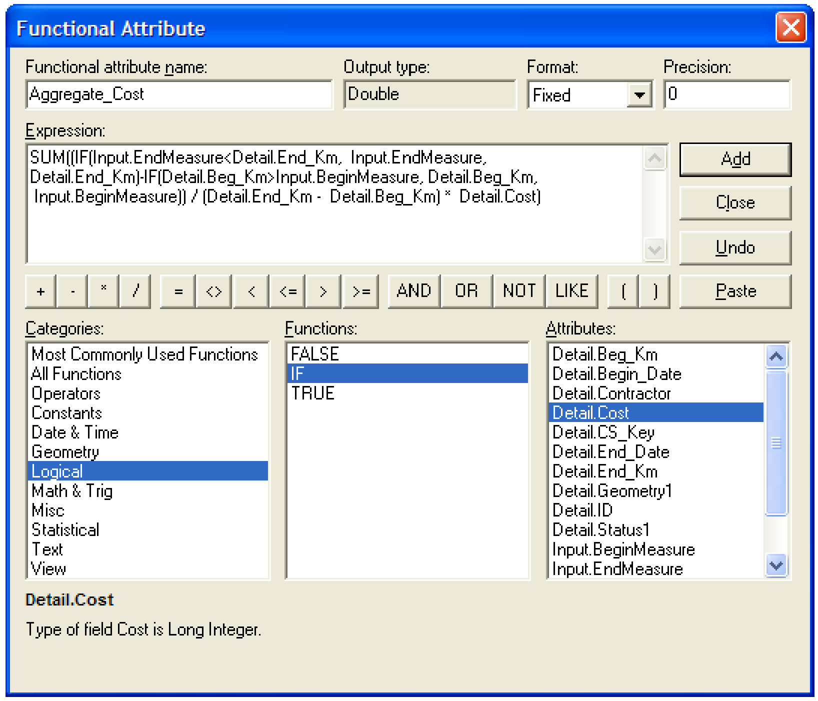


13. Under the **Output** tab, set the **Resolution Operator** to **All**. This means, in our example, that all construction project events will be counted to any equal length segments that they lie upon.

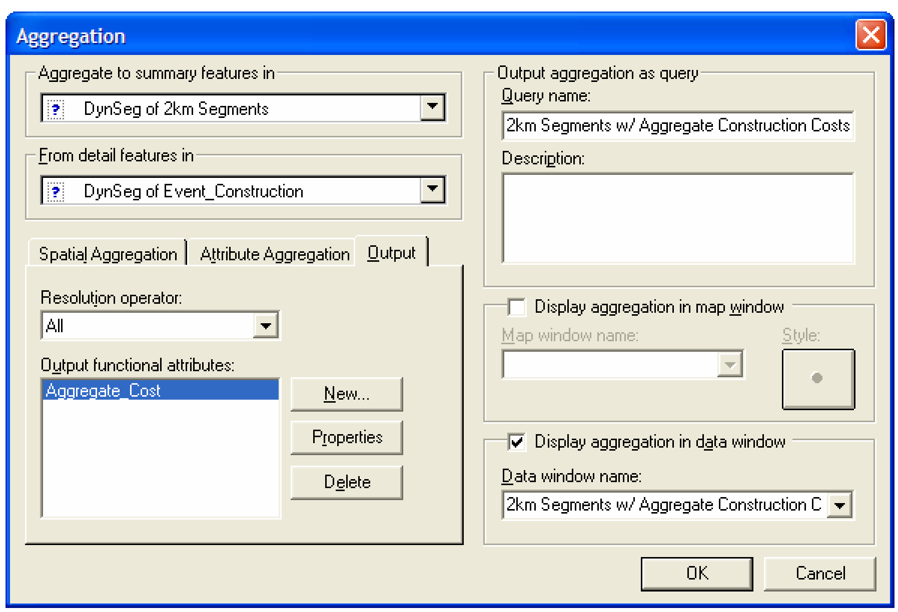


14. Click the **New** button under the **Output** tab to bring up the **Functional Attribute** dialog box. In the following picture we have created an Aggregate Cost attribute. The formula shown does a number of things. The costs for each construction project event that lies along an equal length segment are multiplied by the length of the portion of the construction project event that coincides with the equal length segment and then are divided by the full length of the construction project event. This proportionally allocates the construction costs to each equal length segment that the construction project event crosses. Then these proportionally allocated costs are summed up for each equal length segment and rounded off. The formula from this functional attribute is:

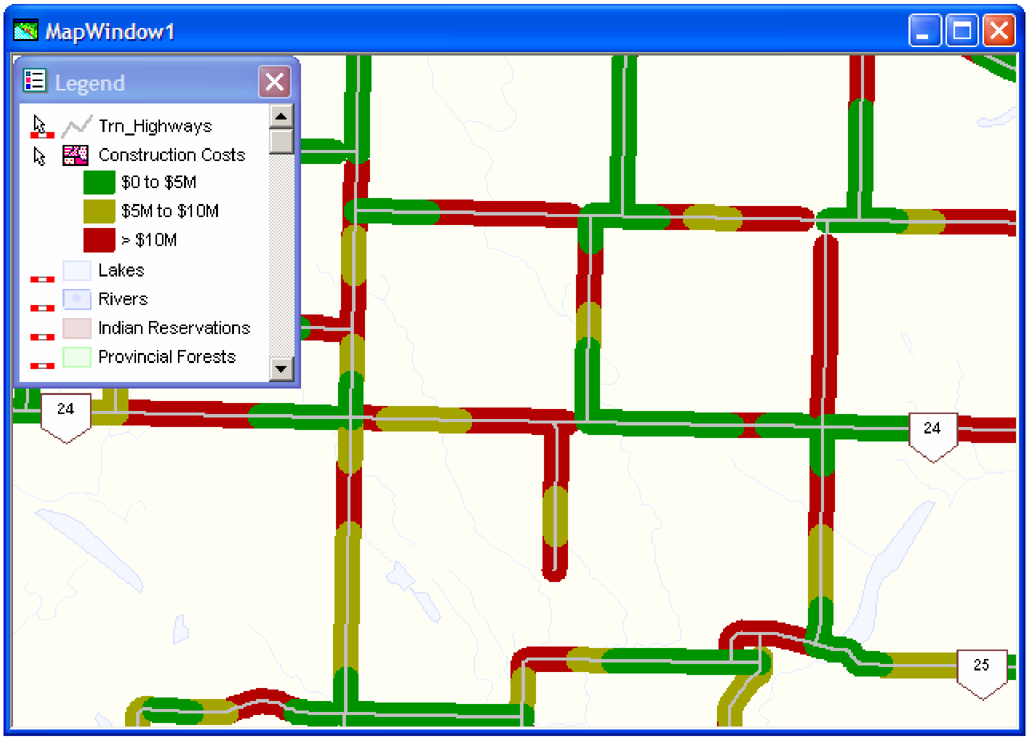
```
SUM((IF(Input.EndMeasure < Detail.End_Km,
Input.EndMeasure, Detail.End_Km)-
IF(Detail.Beg_Km > Input.BeginMeasure,
Detail.Beg_Km, Input.BeginMeasure)) /
(Detail.End_Km - Detail.Beg_Km) * Detail.Cost)
```



15. After filling in the **Functional Attribute** dialog box and dismissing requests to create additional functional attributes, we are returned to the **Aggregation** dialog box. In this case we will not display the results in the map window. Click **OK** to process the aggregation.



16. There are a number of ways to view the results of this aggregation. The picture below shows a thematic view of the aggregate construction costs we have calculated.



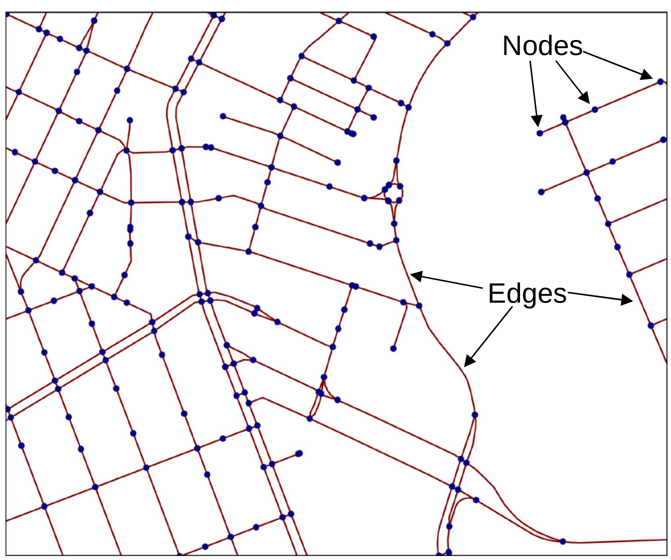
20 – Introduction to Routing

The purpose of this chapter is to outline the basic concepts behind the routing capabilities of GeoMedia Transportation Manager. Each of the major components of a routing network is described, and table descriptions are provided. Lastly, the various routing analysis and maintenance tools are described.

What is a Routing Network?

A routing network is a system of connected linear features that can be used to support the simulated transportation of goods, services, or communications between locations on the network. A network can be thought of as an abstract model that is derived from a set of linear features and their relationships. Each network model is primarily composed of a set of geographic features called *Edges* and a set of implied features called *Nodes*.

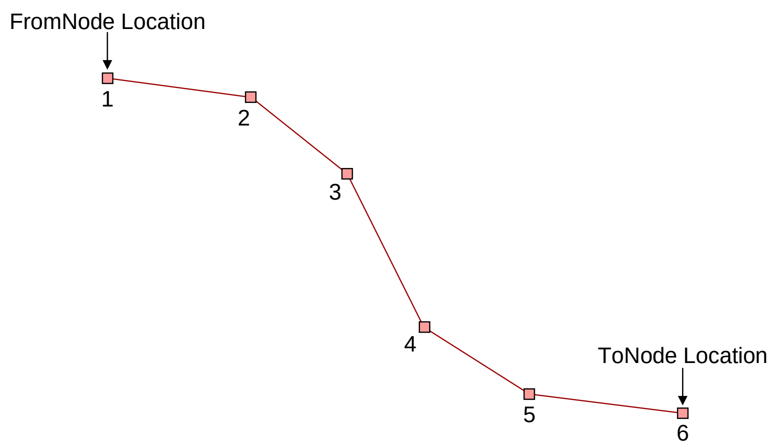
Each Edge in a network represents one component of the transportation system that is being modeled. Streets, highways, shipping lanes, railway tracks, and even Ethernet cables between computer systems are all examples of Edges in a network. Nodes, on the other hand, are implied features that connect the end points of Edges together. A Node might have a real-world analog, such as a street intersection, a seaport, or a computer name, but often there is no real-world equivalent for a Node.



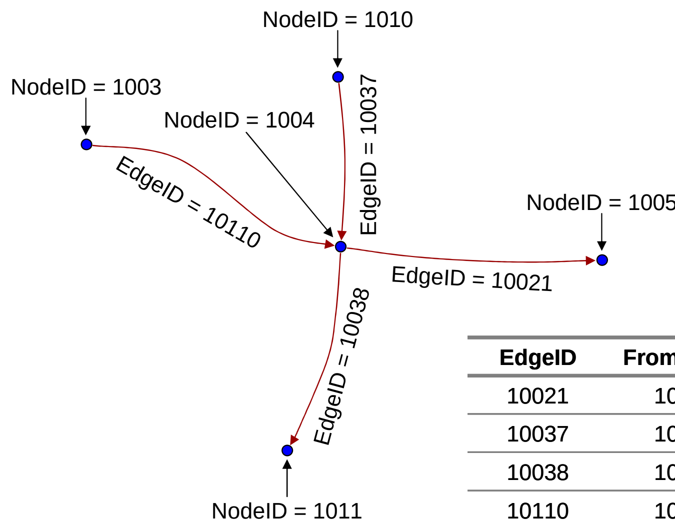
This preceding figure of a street network illustrates the relationship between Edges and Nodes. Each Edge in the Streets feature class is connected to exactly two Nodes, and each Node is connected to one or more Edges. In this example, the Nodes are stored and displayed as a point feature class. In your application, however, the Node locations might only be implied — and never created as a feature class that can be displayed in a map.

Edge-Node Connectivity

The spatial relationship between Edges and Nodes defines the implicit connectivity of the network. Each Node in the network is given a unique identifier termed a *NodeID*. Similarly, each Edge in the network is given a unique identifier termed an *EdgeID*. Network connectivity is maintained by storing the relationship of each EdgeID and the two NodeIDs that connect to it. This relationship is often termed *Edge-Node Connectivity* and is the basis for all the routing analysis tools included in GeoMedia Transportation Manager.



Within GeoMedia, each Edge is defined as a series of points or vertices that define its shape. There is an implicit ordering of the vertices from the first vertex to the last vertex in the chain that gives each Edge a *direction*. The Edge in the preceding diagram is composed of six vertices. The Node that occurs at the first vertex in the chain is referred to as the *FromNode,* and the Node that occurs at the last vertex in the chain is referred to as the *ToNode*. The location of the FromNode is always coincident with the first vertex in the chain. Similarly, the location of the ToNode is always coincident with the last vertex in the chain. In this model, Edges always and only connect at a Node. When two Edges connect, they are said to be *adjacent* to one another and to have at least one Node in common and, therefore, at least one coincident vertex location.



EdgeID	FromNode	ToNode
10021	1004	1005
10037	1010	1004
10038	1004	1011
10110	1003	1004

Each Edge table is a linear feature class that has the following fields:

- EdgeID – This is a long integer value that uniquely identifies each Edge within the routing network.
- FromNodeID – This is a long integer value that identifies the Node at the *beginning* of this Edge.
- ToNodeID – This is a long integer value that identifies the Node at the *end* of this Edge.
- Length (recommended) – This is a numeric value that defines the length of this Edge. Alternately, Begin Measure and End Measure fields can be used, and the length will be calculated as the difference between the two values. As a last alternate, no length fields may be used at all, and the various routing analysis tools can calculate the length from the Edge's geometry. This is considerably slower than the other two options.
- Cost(s) (optional) – This is a numeric value showing the impedance of navigating this Edge (more on impedance later). Some users use different cost fields to model different situations, such as rush hour vs. weekend travel. Also costs can be defined either symmetrically (the same for both travel directions) or asymmetrically (different for each travel direction). Asymmetrical Edge costs require two fields.
- OneWay (optional) – This is a long integer value that holds the code value that defines any one-way navigation restrictions for this Edge. For more on this, see the "Working with the OneWay Editor Command" and the "Working with Display OneWays Command" chapters.

- Blocked (optional) – This is a long integer value that holds the code value that defines any blockages for this Edge. For example, a street might be temporarily blocked due to construction, an accident, or some other reason. It may be blocked in one travel direction or both. For more on this, see the “Working with the Blockage Editor Command” and the “Working with Display Blockages Command” chapters.
- Restrictor source field(s) (optional) – These are fields used in constructing Edge Restrictors. Restrictors are formulaic expressions that define whether a network component is navigable or not (more on Restrictors later).

Each Node table is a point feature class that has the following fields:

- NodeID – This is a long integer value that uniquely identifies each Node within the routing network.
- Cost (optional) – This is a numeric value showing the impedance of crossing this Node (more on impedance later).
- Blocked (optional) – This is a Boolean value (true or false) showing whether navigation across this Node is blocked or not.
- Restrictor source field(s) (optional) – These are fields used in constructing Node Restrictors. Restrictors are formulaic expressions that define whether a network component is navigable or not (more on Restrictors later).

The process of creating a routing network from a linear feature class, along with the required Edge-Node connectivity (also known as explicit topology), is done with the Build Network command (see the “Working with the Build Network Command” chapter).

Impedance

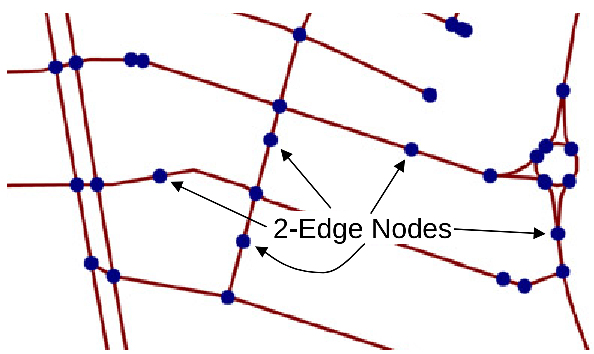
Impedance measures the separation (time, cost, distance, etc.) of different places on a network. Impedances can be based on distance, cost, time, or any other numeric measure assigned to your network. Impedances can be stored on the various components of the routing network: Edges, Nodes, Turns, and Stops (more on Turns and Stops later). Finding Paths with minimum impedance is what routing analysis is all about. The routing analysis tools give two different choices for minimizing impedances. One is to **Minimize Length,** which simply means that it finds Paths with the minimum sum of Edge lengths. The other is to **Minimize Cost,** which finds Paths with the minimum sum of Cost values for Edges and, optionally, Nodes, Turns, and/or Stops. It is assumed that all Cost fields use the same units.

Turn Tables

The routing network, defined by Edges and Nodes and the connectivity between them, can be further refined in a number of ways to better model reality. The first of these ways is with a Turn table. Each record in a Turn table is linked to a specific Edge-to-Edge movement at a Node in the network. Turn tables are commonly used to add delay to turning movements at street intersections or to prohibit certain turn options at a given intersection. For instance, it might take longer to make left turns than right turns at intersections. These asymmetrical delays and prohibited turns are easily managed using Turn tables.

Turn tables are not required. Neither is it required to have a record in a Turn table for every turn option possible in your routing network. If a turn option is not included, the routing analysis tools assume that the turn is permissible and that no cost (e.g., a delay) is associated with that turn. In general it is best to omit records for all turn options that are not prohibited and for which there is no known delay because this will improve performance during routing analyses. On the other hand, if the Path results of your analysis seem unreasonably “jaggy,” it is likely because you are modeling no delay for turns.

One special case for which it is customary not to include turn records is for “two-edge nodes”. That is, a Node to which only two Edges are connected, making it impossible to make real turns (see the following illustration). The Build Network command explicitly lets you omit these turn options from the Turn table it generates.

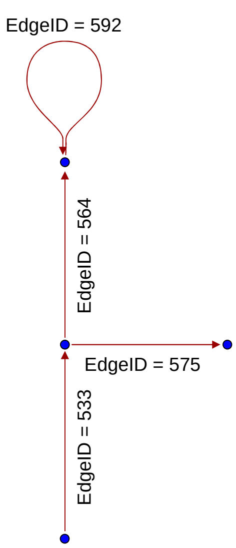


U-turns are another special case. Even in cases where U-turns are permitted, you may want to exclude them from your analysis, especially if you are routing large vehicles. For this case you do not need to create prohibited U-turn records in your turn table. You can simply use the **Disable U-turns** option while doing your analysis.

Each Turn table is a non-graphic table that has the following fields:

- Signed FromEdgeID – This identifies the *in-bound* Edge. The sign says whether the direction of travel is with the digitizing direction (+) or in the reverse of the digitizing direction (-).
- Signed ToNodeEdgeID– This identifies the *out-bound* Edge. The sign says whether the direction of travel is with the digitizing direction (+) or in the reverse of the digitizing direction (-).
- Cost (optional) – This is a numeric value showing the impedance of this turn option (more on impedance later).
- Blocked (optional) – This is a Boolean value (true or false) showing whether this turn option is blocked to navigation or not.
- Restrictor source field(s) (optional) – These are fields used in constructing Turn Restrictors. Restrictors are formulaic expressions that define whether a network component is navigable or not (more on Restrictors later).

The following diagram shows a portion of a routing network and a sample Turn table that shows all of the possible turn options. You can calculate the number of possible turn options at each Node by squaring the number of Edges attached to the Node. So for a three-edge intersection there will be nine possible turn options, and for a four-edge intersection, there will be 16. It should be noted that this turn table could be simplified a bit just by making Edge 592 a one-way street (more on one-way streets later).



FromEdgeID	ToEdgeID	Cost	Blocked	Description
-533	533	4	True	U-turn
533	-533	4	False	U-turn
533	564	1	False	Straight
533	575	2	False	Right
-564	-533	1	False	Straight
-564	564	4	False	U-turn
-564	575	3	False	Left
564	-564	4	True	U-turn
564	592	1	False	Veer right
564	-592	1	True	Veer left
-592	-564	1	True	Straight
-592	592	1	False	Circle Back
-592	-592	4	True	U-turn
592	-564	1	False	Straight
592	592	4	True	U-turn
592	-592	4	True	Circle Back
-575	-533	3	False	Left
-575	564	2	False	Right
-575	575	4	False	U-turn
575	-575	4	True	U-turn

There are two tools provided for working with Turn tables. One is the **Build Network** command, which allows you to create all the Turn table records that you want with a single command with a great amount of flexibility (see the "Working with the Build Network Command" chapter). The second is the **Turn Editor** command, which allows you to interactively edit turns one intersection at a time (see the "Working with the Turn Editor Command" chapter).

Restrictors

Restrictors are expressions defined for network components (Edges, Nodes, and Turns) that are used to constrain the network optimization algorithms. For example, routing a heavy truck over a road network requires each Edge in the path to have a maximum operating weight greater than the operating weight of the vehicle. Restrictors are necessary in logistics applications that involve managing specialized vehicles or in cases where physical or institutional constraints are imposed on the vehicle's Path. Height, width, and weight limits are commonly used Restrictors for vehicles on a network. The routing analysis tools in GeoMedia Transportation Manager can create Paths through the network where these Restrictors are used to constrain the resultant Paths.

Restrictors are modeled as a series of expressions that create Boolean (true/false) results. Each Restrictor models a particular constraint over the routing network. Examples of Edge Restrictors are as follows:

- *Maximum Operating Weight > 80,000lbs*
- *Maximum Vertical Clearance > 13'6''*

The first Restrictor would restrict Paths in the network to those Edges that have a maximum operating weight greater than 80,000 lbs. The second Restrictor would restrict the Path to those Edges that have a maximum vertical clearance greater than 13 feet 6 inches. If both Restrictors were enabled, the Path would be restricted to those Edges that have both a maximum operating weight greater than 80,000 lbs. **and** a maximum vertical clearance greater than 13 feet 6 inches.

- *Truck Route = True*

The third example illustrates the case in which a single Restrictor could replace one or more Restrictors used in combination. If feasible truck Edges were known in advance and stored in the feature table as a Boolean variable, then the Paths created could be restricted to those Edges with a Truck Route value of True.

Note: GeoMedia Transportation Manager assumes that Edge Restrictors are symmetrically applied. This means that any Restrictor applied to an Edge feature in the network feature table will apply in both directions.

Although Edge Restrictors are the most common type of Restrictors, they can also be used on Nodes and/or Turns. Examples of Node and Turn Restrictors are as follows:

- *4-Way Stop = False (Node)*
- *Disallowed during rush hour = False (Turn)*
- *Turn name <> Left (Turn)*

The first Restrictor would restrict Paths over the network from navigating through 4-way stops, which may be useful to avoid slow intersections. The second Restrictor is useful for creating Paths that eliminate turns that are usually legal but which are prohibited during rush hour. The third Restrictor would restrict the Path from taking turns designated as Left. This may be desirable for some vehicles such as snow plows.

Each Restrictor table is a non-graphic table that has the following fields:

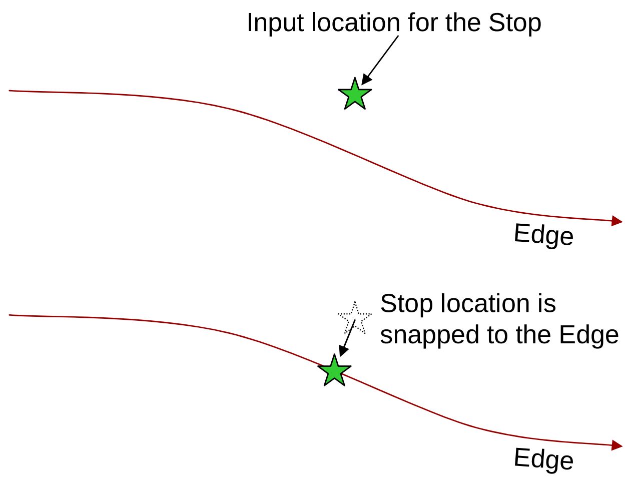
- Name – This text field uniquely identifies this Restrictor within this table. It is recommended that it be as descriptive as possible as to what this Restrictor is for.
- Enabled – This Boolean (true/false) field is used to check off whether a Restrictor is to be used or not used during routing analysis.
- Expression – This text field holds a Functional Attribute formula that equates to a Boolean (true/false) value. Network components will only be considered for routing analysis resultant Paths if their expression equates to True. This expression usually has references to fields within the network component (Edge, Node, or Turn) for which it is being used to constrain.

Unlike the other methods shown so far to fine tune the routing network, Restrictors are not so much descriptors of the network as they are descriptors of how certain vehicles navigate that network. As they are usually vehicle-specific, they might be useful for one routing analysis and not for another. To accommodate this, GeoMedia Transportation Manager has made it easy to check off which Restrictors you wish to use for each analysis. For more on how to create, edit, enable, and disable Restrictor see the “Working with the Restrictor Editor Command” chapter.

Stops

Network Stops are important locations on a network that will be used as part of an analysis. A Stop can represent any type of geographic phenomena; common examples of Stops are Customers, Distribution Centers, Fire Stations, and Crime scenes. The exact nature of what constitutes a Stop is defined in the context of your application and might be a point that is digitized from a map, a feature in a point feature class, or an input by a GPS or some other external mechanism.

Each point of interest must be turned into a Stop by snapping the input location to an Edge in the network and storing some information about where the snap point occurred. This process is illustrated in the following figure:



Each Stop table is a point feature class that has the following fields:

- StopName – This text field is used to provide a descriptive name for each Stop and can be used in labeling and directions.
- StopSetName – This text field contains a group name for a set of Stops within a Stop table. This allows you to have a single Stop table that serves a number of purposes.
- Enabled – This is a Boolean (true/false) field that indicates whether this Stop is to be used in a routing analysis or not.
- StopEdgeID – This long integer field identifies which Edge within the routing network that this Stop lies along.
- StopEdgePosition – This numeric field indicates the relative position of the Stop along its Edge. A value of 0 means it as at the beginning of

the Edge, and a value of 1 means it as at the end. Intermediate values between 0 and 1 describe intermediate locations along the Edge.

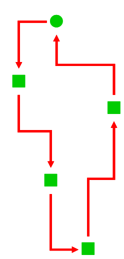
- ToleranceValue – This numeric field holds the snap distance used to convert the point geometry location of each record into an Edge location.
- ToleranceUnitID – This long integer field contains the code value that defines the units of the ToleranceValue field.
- StopOrder – This long integer field contains the sequential order of Stops within a Stop Set. This is used by the Best Path command if the **Best Order** option is not picked.
- Cost – This is a numeric value showing the impedance of this Stop. For example, a maintenance Stop to change a light bulb might have a cost of two minutes, while a maintenance Stop to fix an air conditioner might have a cost of 60 minutes.

GeoMedia Transportation Manager contains a specialized tool called the Stop Manager for taking locations you provide and assigning them to the network to create Stops. For more on how to create and manage Stops, see the “Working with the Stop Manager Command” chapter.

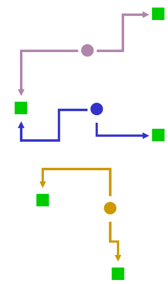
Routing Analysis

There are five different routing analysis tools provided with GeoMedia Transportation Manager. Although their operation is described in detail elsewhere, a brief summary of all of them is provided here.

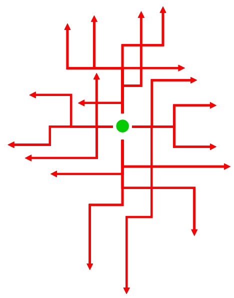
Best Path – This command gives the ability to find the best path for a vehicle that needs to make one or more stops. You may select whether Paths will be optimized so as to minimize distance traveled or to have the command minimize other “user costs,” such as time, money, or even safety. It also has an option to optimize the order of stops to further optimize the path. The time savings for maintenance, delivery, or other vehicles can result in significant cost savings, or in the case of emergency vehicles, even saved lives. For more information see the “Working with the Best Path Command” chapter.



Easy Path – This is a “quick and dirty” version of the **Best Path** command. The difference is that instead of using stored Stops in the analysis, you simply click on the map view, and the Paths are calculated as you click. For more information see the “Working with the Easy Path Command” chapter.



Find Closest Stops – This command finds the closest *n* destination Stops to a set of origin Stops. This is particularly useful for finding, for example, the two closest hospitals to each of a set of sports facilities. It is also useful in travel demand modeling workflows. Path optimization may be by distance or by user cost. For more information see the "Working with the Find Closest Stops Command" chapter.



Network Coverage – This command traces all paths that can be reached from a set of origin Stops within a given user cost or distance. This is useful, for example, in finding all the locations that can be reached from a fire station in a given amount of time and identifying areas where coverage is insufficient. You can define the number of bands of solutions using multiples of the given user cost or distance (e.g., three bands at 5 minutes would generate 5, 10, and 15 minute solutions). For more information see the "Working with the Network Coverage Command" chapter.

Generate Path Directions – This command will take the result of either the **Best Path** or the **Find Closest Analysis** command and will create directions to navigate between Stops. The format of the directions is user configurable. For more information see the "Working with the Generate Path Directions Command" chapter.

Routing Maintenance

There are eight different routing maintenance tools provided with GeoMedia Transportation Manager. Although their operation is described in detail elsewhere, a brief summary of all of them is provided here. A good overview of Routing maintenance is provided in the "Routing Data Maintenance Workflows" chapter.

Build Network – This command gives the ability to create routing-ready Edge, Node, and Turn feature classes. The results of this command can be directly used with the routing analysis commands. For more information see the "Working with the Build Network Command" chapter.

Stop Manager – This command gives the ability to create and manage Stops. Stops are a special kind of point feature used by the routing analysis commands to define key points along their calculated paths. For more information see the "Working with the Stop Manager Command" chapter.

Turn Editor – This command gives the ability to interactively edit turn options. You can set costs associated with various turn options as well as being able to disallow turn options as desired. For more information see the “Working with the Turn Editor Command” chapter.

Restrictor Editor – This command gives the ability to define and edit restrictions for edges, nodes, and turns. Restrictions are very useful tools that work in conjunction with the various routing analysis commands to allow you to restrict which edges, nodes, and turns will be considered when calculating paths. For more information see the “Working with the Restrictor Editor Command” chapter.

OneWay Editor – This command gives the ability to interactively edit the one-way attributes of selected edges. This allows you to more accurately model the allowable navigation routes through a network. For more information see the “Working with the OneWay Editor Command” chapter.

Blockage Editor – This command gives the ability to interactively edit the blockage attributes of selected edges. This allows you to more accurately model the allowable navigation routes through a network. For more information see the “Working with the Blockage Editor Command” chapter.

Display OneWays – This command gives the ability to display the direction of permitted movement in the map view for all those features of the selected edge feature class that have a one-way attribute set. For more information see the “Working with the Display OneWays Command” chapter.

Display Blockages – This command gives the ability to display in the map view the direction of blockages for those edges of the selected edge feature class that have the Blockages attribute set. For more information see the “Working with the Display Blockages Command” chapter.

21 – Working with the Build Network Command

The **Build Network** command allows you to create Edge, Node, and Turn queries for routing analysis. The results of this command can be directly used with the following routing analysis commands: **Easy Path**, **Best Path**, **Find Closest Stops**, and **Network Coverage**. For more information on building and maintaining a Routing network, see the "Routing Data Maintenance Workflows" chapter.

The minimum input is a linear feature class or query. Optionally, you can have a Point feature class or query as input for the Node output. This is useful when cost/blocked data already exists for these nodes. The output of Node and Turn queries is optional.

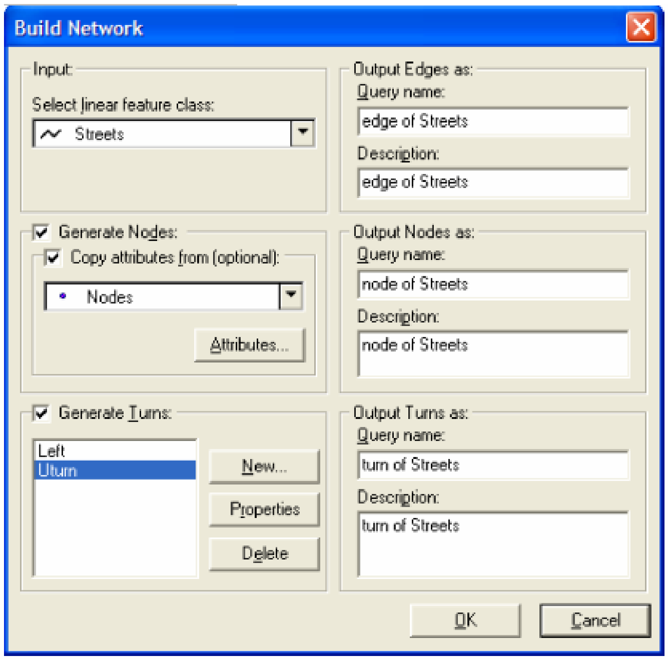
The output Edge queries will be identical to the input linear feature class except for the addition of network topology fields: an EdgeID field, a FromNode field, and a toNode field. The output Node query has point geometry and will have a NodeID field plus any selected fields from the input Node table if that option is chosen. The output Turn class will have a signed FromEdgeID field, a signed ToEdgeID field, a TurnName field, a TurnAngle field, a Cost Field, and a boolean Blocked field. A positive edge ID means the turn's direction is same as the edge's digitizing direction; a negative edge ID means the turn's direction is reverse to the edge's digitizing direction.

The Build Network Command Workflow

This section presents the procedural steps for using the Build Network command.

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select **Transportation > Routing Maintenance > Build Network**.

   The **Build Network** *dialog box is displayed.*

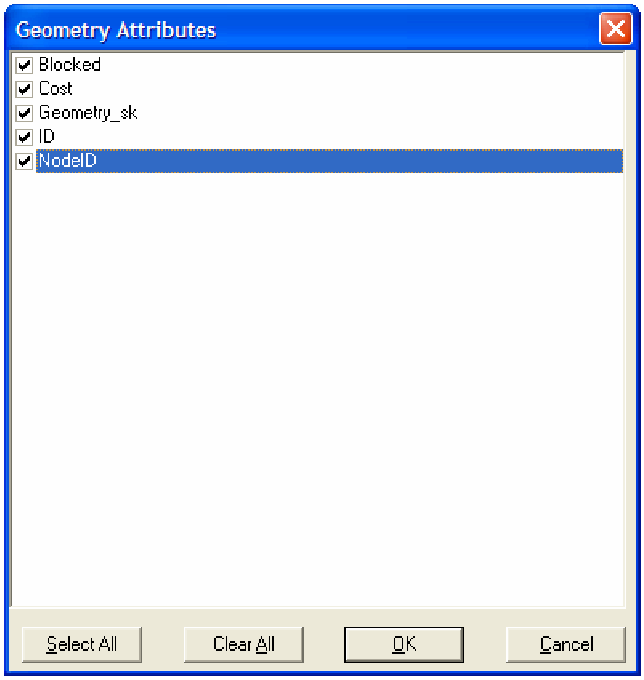


3. On the **Select linear feature class** pull-down list, select a linear feature class or query from the list of all connections and queries. All fields from this feature class will be copied to the output query and will have extra network topology fields added to it so as to make it ready to use with the various routing analysis commands.
4. To generate nodes, click the **Generate Nodes** check box to enable the contents of the group box.

Note: Nodes are not required for routing analysis, but they are necessary if you want to assign costs to passing through nodes or if you want to block passage past selected nodes.

---

5. If you have an existing Node feature class or query that you want to copy attribution from, check the **Copy attributes from** check box. Then select the point feature class or query that you want to copy from in the pull-down list.
6. You can then click on the **Attributes** button to pick which Node attributes that you want to copy.



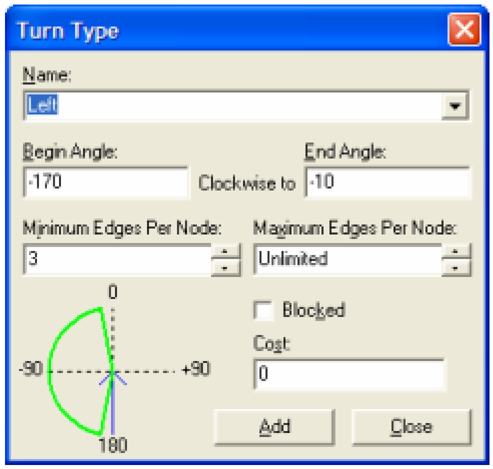
7. To generate turns, check the **Generate Turns** check box to enable the contents of the group box.

Note: Turns are not required for routing analysis, but are necessary if you want to assign costs to or block certain turn options. Generating turns like this is useful for making global turn assignments to represent standard conditions. To learn how to edit individual turns, see the "Working with the Turn Editor Command" chapter.

---

8. Click **New** to begin adding new turn types.

   The **Turn Type** dialog box is displayed**.**



9. Select from the **Name** pull-down list one of the pre-defined turn types, or key in a name of your own choosing.

Note: When you pick one of the pre-defined names, the other fields of this dialog box are automatically filled in.

---

10. Key in a **Begin Angle** and an **End Angle**. The diagram in this dialog box will update to graphically show the range of turn angles that these two angles will encompass.
11. Set the **Minimum Edges Per Node** and the **Maximum Edges Per Node**. This allows you to exclude instances such as when only two edge segments are connected that represent just straight segments of road and not a turn opportunity.
12. Check or uncheck **Blocked** to give a default value for this turn type. If you check **Blocked**, then none of the generated turns will be allowed.
13. Key in a default **Cost** value (for example, the time to navigate a certain turn option).
14. Click **Add**.
15. Click **Properties** to modify a turn's properties (except for the name) on the **Turn Type** dialog box.
16. Accept the default **Output Edges as** query name, or key in a new name (a description is optional). This field is enabled when a linear feature class is selected as input. It allows for storing resultant edges as a query in the GeoWorkspace. A default query name is assigned.
17. Accept the default **Output Nodes as** query name, or enter a new name (a description is optional). This field is enabled if you checked

Generate Nodes. It allows for the storage of the resultant nodes as a query in the GeoWorkspace.

18. Accept the default **Output Turns as** query name, or enter a new name (a description is optional). This field is enabled if you checked **Generate Turns**. It allows for storing resultant turns as a query in the GeoWorkspace.
19. Click **OK** to process the inputs, or click **Cancel** to exit the command without processing.

22 – Working with the Stop Manager Command

The **Stop Manager** command allows you to create and manage Stops. Stops are a special kind of point feature and are one of the fundamental components of network analysis. They are used by the **Best Path, Find Closest Stops**, and **Network Coverage** commands. Each of these commands uses Stops to define origin and/or destination points for the paths that they calculate along a linear network.

The **Stop Manager** command allows you to create or edit Stop feature classes. It also allows you to group Stops into *Stop sets*. This allows the logical organization of Stops and enables one Stop feature class to contain Stops used for many different purposes. New Stops can be added to the Stop Set from a feature class or a saved query, or by clicking on the map window.

The command also allows you to edit information about individual Stops. For instance, you can enable or disable individual Stops, you can reorder them, or you can edit their names or costs (for example, time spent at each Stop).

The Stop Manager Workflows

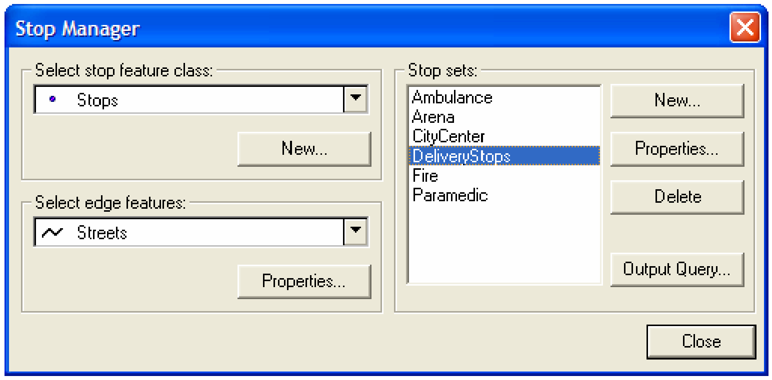
There are two workflows for this command:

- Workflow for creating new stop feature classes
- Workflow for managing Stop sets

Workflow for creating new stop feature classes

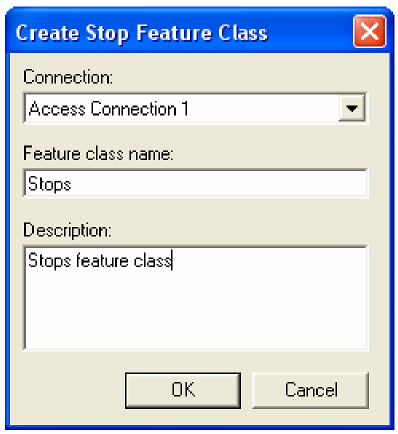
1. Open a GeoWorkspace that contains one or more read/write connections.
2. Select **Transportation > Routing Maintenance > Stop Manager.**

   The **Stop Manager** *dialog box is displayed.*



3. Click the **New** button in the **Select stop feature class** box to create a new Stop feature class.

   The **Create Stop Feature Class** *dialog box is displayed.*



4. Select a read/write connection from the drop-down list, and specify a name (and optional description) for the new feature class.
5. Click **OK** to dismiss the **Create Stop Feature Class** dialog box.

Workflow for managing stop sets

1. Open a GeoWorkspace that contains one or more read/write connections.
2. Select **Transportation > Routing Maintenance > Stop Manager**.

   The **Stop Manager** *dialog box is displayed.*

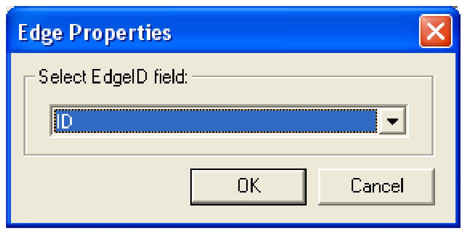
3. On the **Stop Manager** dialog box, select an existing Stop feature class from the drop-down list, which will show all point feature classes. However, not all point feature classes are Stop feature classes because they may not have all of the required fields. If you select a point

feature class that is not a Stop feature class, you will get an error message instructing you to select a valid Stop feature class.

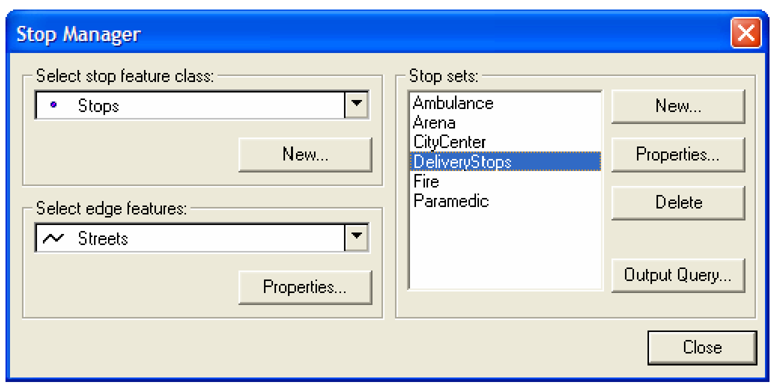
4. Select an edge feature class from the **Select edge features** drop-down list.

5. Click **Properties** to select the **EdgeID** field on the **Edge Properties** dialog box.

   The **Edge Properties** *dialog box is displayed.*



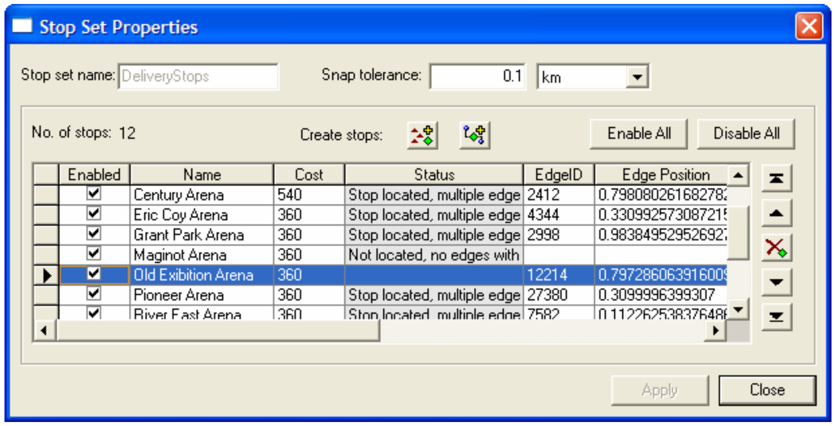
6. Select an **EdgeID** field, and click **OK** to dismiss this dialog box and to return to the **Stop Manager** dialog box. If there are existing Stop sets in the Stop feature class, they will be presented in the **Stop sets** list box, as shown below.



7. From here you can select a Stop set and delete it by using the **Delete** button or edit it using the Stops sets **Properties** button. If you want to create a new Stop set, use the Stop sets **New** button.

   Using either the Stop sets **New** *or* **Properties** *button will cause the* **Stop Set Properties** *dialog box to be displayed.*

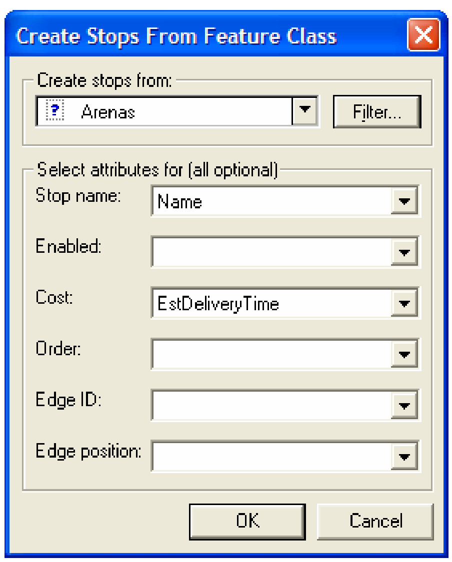
Note: To view more stops in the grid, resize the height of the **Stop Set Properties** dialog box by dragging its border.



8. If the **Stop Set Properties** dialog box was reached by using the **New** Stop sets button on the **Stop Manager** dialog box, then the grid portion will be empty and the **Stop set name** field will be editable. In this case, fill in the desired name for the new Stop set. If you reached the **Stop Set Properties** dialog box by using the Stop set **Properties** button on the **Stop Manager** dialog box, then skip to the next step.

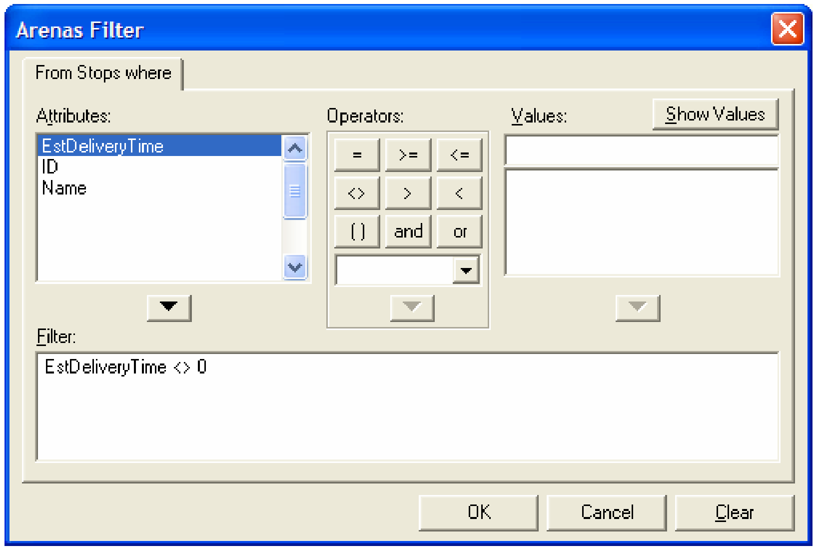
9. Click the **Create Stops from Feature Class** button  to create Stops from an existing point feature class or query. If you do not want to do this, skip to Step 13.

   The **Create Stops From Feature Class** *dialog box will be displayed.*



10. Select a point feature class or query from the **Create stops from** drop-down list. If you want to only use certain records from that class, you

can use the **Filter** button to narrow the records used by using the standard GeoMedia query dialog box.



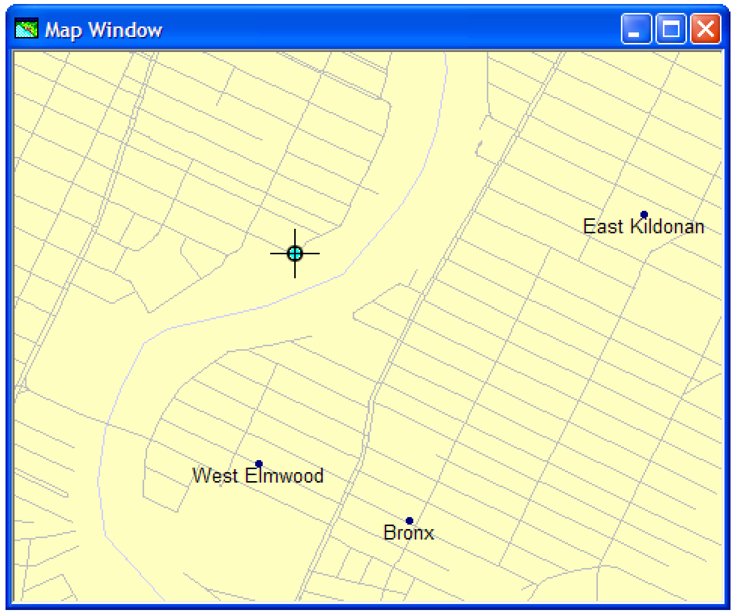
11. If you have any attribute data from the source feature class or query that you want to use to populate the Stop feature class fields, you can select them from the **Select Attributes for** drop-down lists. Note that all of these attribute field names are optional. Those fields that are not used will have Null or default values in the resultant Stop records.

12. Click **OK** to dismiss the **Create Stops From Feature Class** dialog box.

    This will return you to the **Stop Set Properties** *dialog box, where you will see your newly added Stops at the bottom of the list in the grid.*

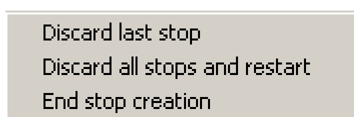
13. Click the **Create Stops from Map** button  to add new Stops directly to the map. If you do not want to do this, skip to Step 19.

    The **Stop Set Properties** *dialog box will temporarily disappear, and a point feature will appear on your cursor when in the* **Map View**.



14. Click in the **Map View** to place as many Stops as desired.

15. At any point in the process, you can bring up the following pop-up menu by right-clicking in the map window.



16. Select **Discard last stop** if you want to undo your last Stop click in the map window.

17. Select **Discard all stops and restart** if you want to clear all the Stops in the Stop set and restart.

18. When finished, you can either press ESC on your keyboard or select **End stop creation** from the right-click menu. The following dialog box will be displayed when ESC is pressed. Click **Yes** to return to the **Stop Set Properties** dialog box, or click **No** to place more Stops. If you click **Yes**, your newly added Stops can be seen at the bottom of the list in the grid. All attribute values for these new Stops will be Null or zero.

19. To edit an individual Stop, select it by clicking on the gray button to the left side of the row depicting that Stop. That row will be highlighted, and the buttons to the right of the grid will become enabled. Also, that feature will be highlighted in the **Map View**.

20. Once selected, there are a variety of edits that can be done to the Stop. The **Name** and **Cost** values can be edited by clicking and typing in a new value; the Stop can be deleted by clicking the **Delete** button ; the Stop can be moved up or down in the sequence by using the , , , and buttons; and the Stop can be enabled or disabled by clicking on the **Enabled** check box or by using the **Enable All** or **Disable All** buttons.

Note: The **Delete** button on this dialog will become disabled when there is only one stop left in the list box. If you choose to do so, an entire Stop set can be deleted using the **Delete** button on the **Stop Manager** dialog box.

The Cost value is used by the various routing analysis command if the **Minimize Cost** option is selected. The sequence of the stops within the grid is used by the **Best Path** command when the **Best Order** option is not used or if either the **Start Fixed** or the **End Fixed** option is used in that command. Stops that do not have their **Enabled** boxes checked will be ignored by the various routing analysis commands.

21. If you want to exit the **Stop Set Properties** dialog box without saving any of the changes you have made, click the **Close** button. Otherwise, first click **Apply**.

    The **Apply** button makes use of the **Snap Tolerance** value to find the closest edge to each of the Stops in the list. This may take some time to calculate based on the size of the dataset. When completed, the **Status**, **EdgeID**, and **Edge Position** fields will be filled out in the grid.

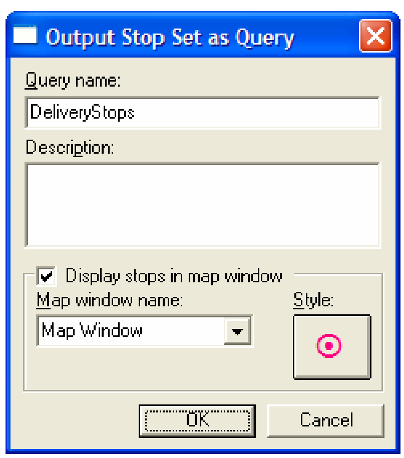
    The **Status** field will indicate the relative success of finding an Edge within the snap tolerance of each Stop. A blank field indicates success. The message "*Stop located, multiple edges may be within tolerance*" indicates success, but that more than one Edge was found within tolerance. The closest Edge is used in this case. The message "*Not located, no edges within tolerance*" indicates no Edge was located. Stops with this status will be ignored by any of the routing analysis commands.

The **EdgeID** and **EdgePosition** fields indicate which Edge a Stop was snapped to and the relative position (0 to 1) of that Stop along that Edge.

Note: The **EdgeID** and **EdgePosition** values calculated here are static values. That means that if your network changes, you may need to update these values to reflect those changes. You can update these values by revisiting the **Stop Set Properties** dialog box and by using the **Apply** button. It is possible to have the **EdgeID** and **EdgePosition** values calculated dynamically using the **Output Stop Set as Query** option (see Step 23 below). For example, if a network edge query is being used for network analysis, and stops have been assigned to edges prior to applying a spatial filter to the workspace or network edge query, the **Stop Manager** command needs to be run again to reassign the stops in a stop set.

---

22. Click **Close** to exit the **Stop Set Properties** dialog box and to return to the **Stop Manager** dialog box. If you have not first selected the **Apply** button, an informational message will appear, letting you know that if you exit the dialog box now, none of your changes will be saved.
23. To view your Stops, select the **Output Query** button. This brings up a standard output to query output dialog box. Fill out as desired, and click **OK** to execute the output or **Cancel** to cancel.



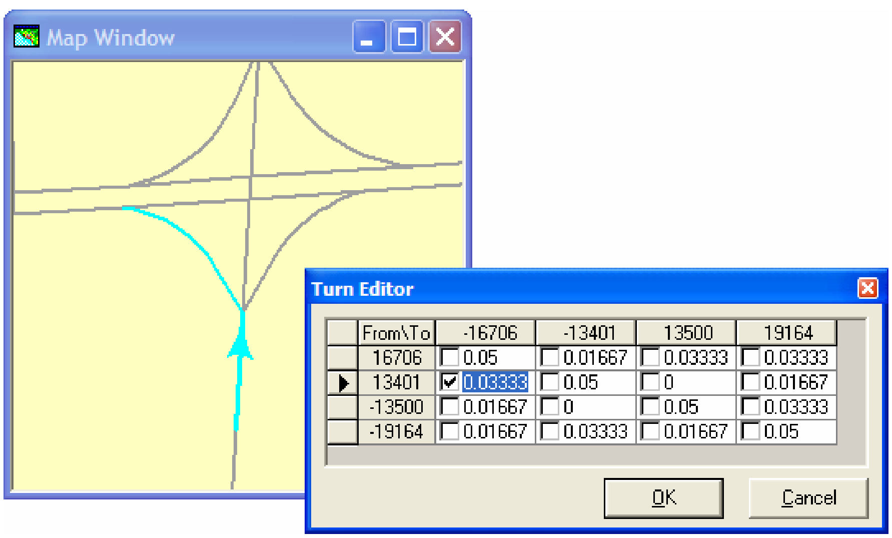
Note: The **Output Stop Set as Query** has another important feature. The **EdgeID** and **EdgePosition** values within this query are calculated dynamically. That means that if your network changes, these values will update automatically. This is especially useful when working with Spatial Filters. However, this dynamic behavior does inflict a performance penalty. Both static and dynamic methods for maintaining your routing network are discussed in the "Routing Data Maintenance Workflows" chapter.

24. Click **Close** to exit the **Stop Manager** dialog box and the **Stop Manager** command.

23 – Working with the Turn Editor Command

The **Turn Editor** command allows you to interactively edit turn options. You define the intersection of interest by clicking in the map window; then you can set costs associated with various turn options as well as being able to disallow turn options as desired. As each intersection may have many turn options, graphical feedback is given in the map window to clearly indicate which turn option is being edited.

For a turn action not previously defined in the turn table, a new turn record will be created if you enter data for that turn. If a turn is not checked as prohibited and its cost is set to 0, the command will delete that turn record. When you click **OK**, the changes are written to the database. Clicking **Cancel** discards the changes.



The Turn Editor Workflow

1. Open a GeoWorkspace that contains one or more read/write connections.

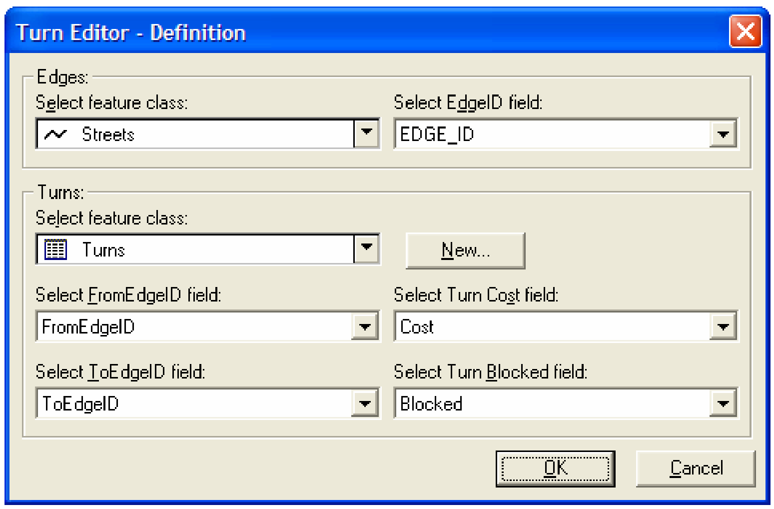
Note: Due to the dynamic nature of queries in GeoMedia, it is recommended that this command be run in a workspace with a minimal number of active queries that are based on your routing network components. That is because you may pay a substantial performance penalty when running these commands while waiting for these queries to update.

---

2. Display the LRS linear feature class for which you wish to edit turns. Use the Legend to make sure it is locatable.

3. Select **Transportation > Routing Maintenance > Turn Editor**.

   The **Turn Editor - Definition** *dialog box is displayed.*

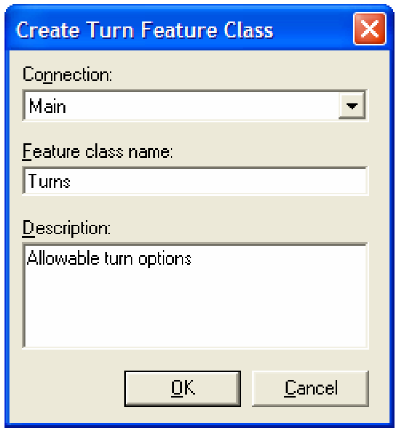


4. Select a linear feature class for **Edges** from the **Select feature class** drop-down list.

5. Select an **EdgeID** from the **Select EdgeID field** drop-down list.

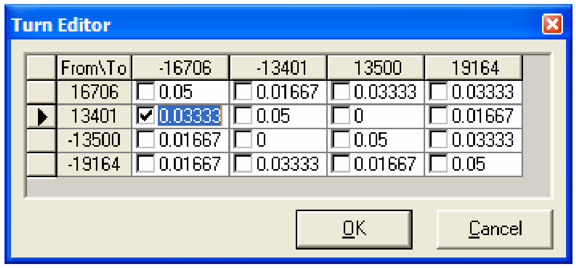
6. Click **New** in the **Turns** group box if you want to create a new Stop feature class. Otherwise skip to Step 9.

   The **Create Turn Feature Class** *dialog box is displayed.*



7. Select a read-write connection from the drop-down list, and specify a name (and optional description) for the new feature class.
8. Click **OK** to dismiss the **Create Turn Feature Class** dialog box and return to the **Turn Editor - Definition** dialog box.
9. In the **Turns** box, select a Turn Table feature class from the **Select feature class** drop-down list. For more information on how to create this table see the “Working with the Build Network Command” chapter.
10. Select the FromEdgeID field from the **Select FromEdgeID field** drop-down list.
11. Select the ToEdgeID field from the **Select ToEdgeID field** drop-down list.
12. Select the Turn Cost field from the **Select Turn Cost field** drop-down list.
13. Select the Turn Blocked field from **Select Turn Blocked field** drop-down list.
14. Click **OK** to accept your Turn definition settings, or click **Cancel** to exit the command. In either case the **Turn Editor – Definition** dialog box is dismissed.
15. If you clicked **OK**, you will be prompted to **“Click on an intersection where turns are to be edited. Press ESC to exit this command.”** Click inside the map window on the intersection of edge features where there are turn options you want to edit.

    If one or more edges are located, they are processed to display the **Turn Editor** *dialog box.*

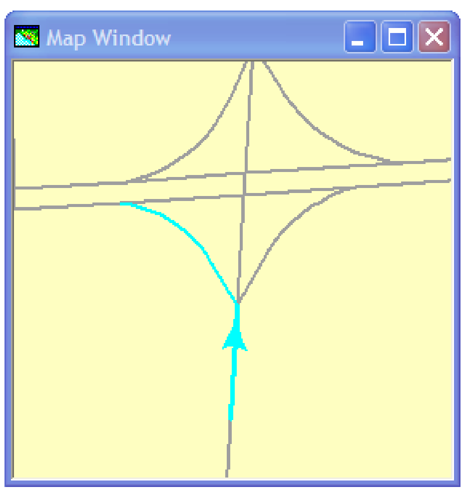


If no edges are located, a message box is displayed prompting for another click.

Note: Located EdgeIDs are displayed along the row and the column of the grid. Turn cost and blocked values are displayed in the grid cells.

---

16. Click in a grid cell in this dialog box. Each cell represents a turn option. When you click in a cell, the software highlights the From Edge and the To Edge associated with that turn movement. The From Edge will be further identified by an arrow displayed on it. Once the turn option that you want to edit is identified, you can edit the cost associated with this turn option by typing it into the text box in the cell. You can also disallow the turn option by checking the **Blocked** check box.



Note: For one-way streets it is more efficient to mark these streets as one-way streets (see the “Working with the OneWay Editor Command” chapter) than to disallow turns that would be traveling in the wrong direction. It is not necessary to both mark the street as one-way and to disallow the turn.

17. Click **OK** to save the edits, or click **Cancel** to dismiss the dialog box.

    The **Turn Editor** *dialog box disappears, and you are once again prompted to identify another intersection.*

18. Repeat Steps 15 through 17 to edit as many intersections as you need. Press ESC when the **Turn Editor** dialog box is not showing to exit the command.

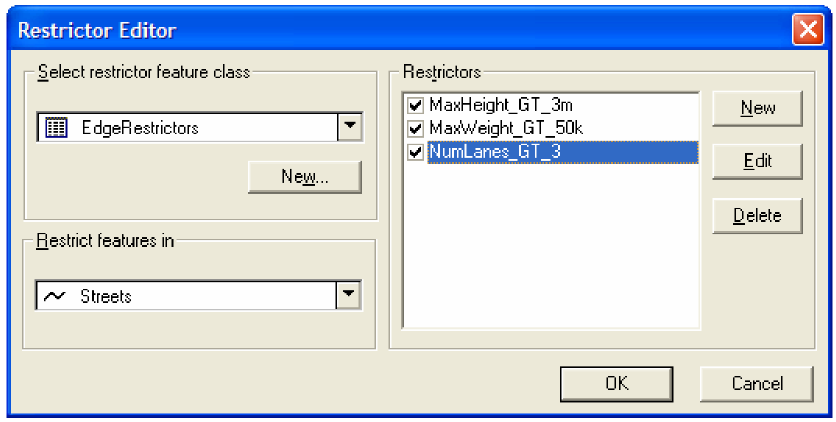
24 – Working with the Restrictor Editor Command

The **Restrictor Editor** command allows you to define and edit restrictions for edges, nodes, and turns. Restrictions are very useful tools that work in conjunction with the various routing analysis commands to allow you to solve specific routing problems. For example, if you are routing a very large vehicle, you may want to limit your routing analysis to consider only edges with at least a 30,000 pound weight rating and a height rating of at least 13 feet. If you are routing a vehicle during rush hour, you may want to avoid turns that are illegal during those hours. If you are doing emergency planning for a holiday, you may want to avoid certain intersections that are closed due to street fairs.

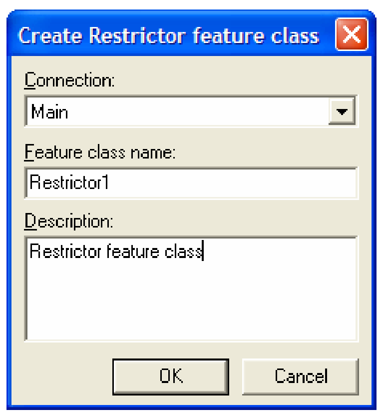
These restrictions are defined using the **Functional Attribute** capabilities of GeoMedia. The **Functional Attribute** interface allows you to easily define simple restrictions, but you also have the power to model very complex restrictions based on multiple fields with intricate equations. These Functional Attribute expressions need to resolve to a Boolean (True or False) value. Only edges, nodes, or turns whose expression evaluates as True will be considered to be in the Path results of any of the routing analysis commands. The various restrictions created with this command can be individually enabled or disabled and can be used in combination with each other. This gives you a great deal of flexibility to do analyses such as routing large vehicles during rush hour. For more information, see the discussion of the **Functional Attributes** command in *Working with GeoMedia* or *Working with GeoMedia Professional*.

The Restrictor Editor Workflow

1. Open a GeoWorkspace with at least one read/write connection to a warehouse.
2. Select **Transportation > Routing Maintenance > Restrictor Editor.**



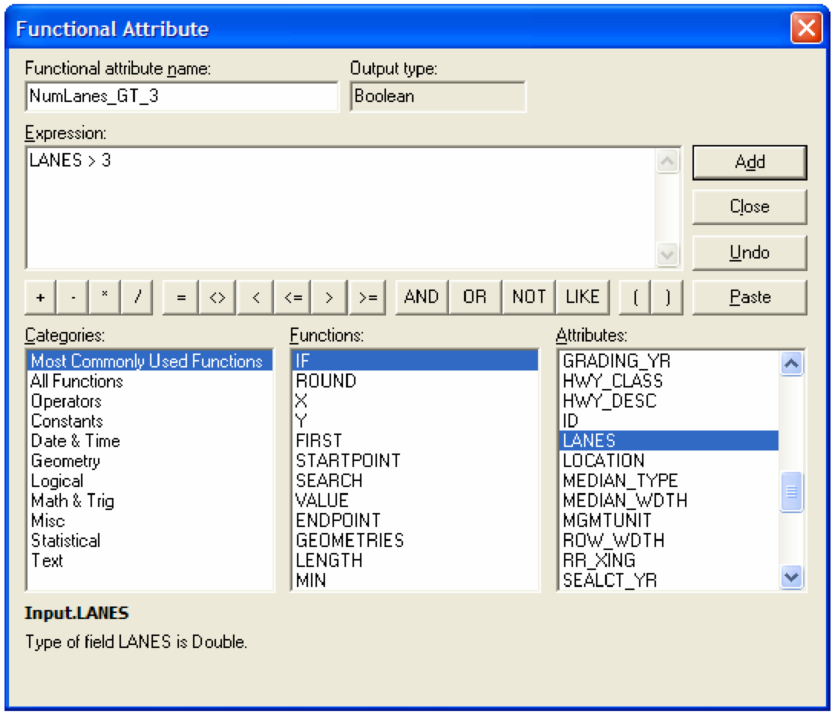
3. Click **New** in the **Select Restrictor feature class** box to create a new Restrictor feature class on the **Create Restrictor feature class** dialog box. To use an existing Restrictor feature class, skip to Step 5.



4. Select an existing connection name from the **Connection** drop-down list, key in name of the restrictor feature class in the **Feature class name** field and, optionally, key in a **Description**. Then click **OK** to create the new feature class and to add columns with predefined names.

5. Select a feature class from the **Select Restrictor feature class** pull-down list.

6. On the **Restrictor Editor** dialog box, click **New** in the **Restrictors** box to define a new restrictor on the **Functional Attribute** dialog box.



7. Accept or override the default name for the restrictor in the **Functional attribute name** field, and then build an expression that results in a Boolean value (True or False). If you create an expression that does not result in a Boolean value, the software will issue this error message: **"The expression evaluates to an ineligible data type."** Only edges, nodes, or turns whose expression evaluates as True will be considered to be in the Path results of any of the routing analysis commands.

Note: For more information, see the discussion of the **Functional Attributes** command in *Working with GeoMedia* or *Working with GeoMedia Professional*.

---

8. Click **Add** to save the restrictor.
9. Select a restrictor in the list, and click **Edit** in order to edit the restrictor on the **Functional Attribute** dialog box.
10. Edit the restrictor name or expression, and click **OK** to save the changes.
11. On the **Restrictor Editor** dialog box, select a restrictor in the list, and click **Delete** to delete the restrictor.

12. Click on the check box next to each item in the list to enable or disable the restrictors.
13. Click **OK** to save the edits and to dismiss the dialog box, or click **Cancel** to dismiss the dialog box without saving the edits.

Note: Edits made with the **Restrictor Editor** command have no effect on pre-existing results from the **Best Path**, **Find Closest Stops**, or **Network Coverage** commands. To see the effects of changes to restrictions, you need to re-run any routing analyses.

---

25 – Working with the OneWay Editor Command

The **OneWay Editor** command allows you to interactively edit the one-way attributes of selected edges. This allows you to more accurately model the allowable navigation routes through a network. By default, every edge in the network model is assumed to be bi-directional. A bi-directional edge is one that can be traversed in either the forward direction (that is, in the direction of digitization or FromNode to ToNode) or the reverse direction (that is, in the reverse direction of digitization or ToNode to FromNode). However, using this command, individual edges can also be set to one-way in either the forward direction or the reversed direction.

The command uses visual feedback in the map window to facilitate the editing of one-way restrictions. The selected edges will be symbolized in the map window based on their one-way attribution. As you edit the one-way restrictions, the symbology of the selected edges will change to match the current setting.

Underlying this visual feedback is the editing of an integer value of the One-way attribute of the Edge feature class. The supported integer values for this attribute, and their meanings, are listed below:

Value	Meaning
0 or Null	No one-way restrictions to navigation
1	One-way navigation allowed in reverse digitizing direction
2	One-way navigation allowed in digitizing direction

The OneWay Editor Workflow

1. Open a GeoWorkspace that contains one or more read/write connections.

Note: Due to the dynamic nature of queries in GeoMedia, it is recommended that this command be run in a workspace with a minimal number of active queries that are based on your routing network components. That is because you may pay a substantial performance penalty when running these commands while waiting for these queries to update.

2. Select one or more edges in the map window that you wish to edit.

   This will cause the **OneWay Editor** *command to become enabled.*

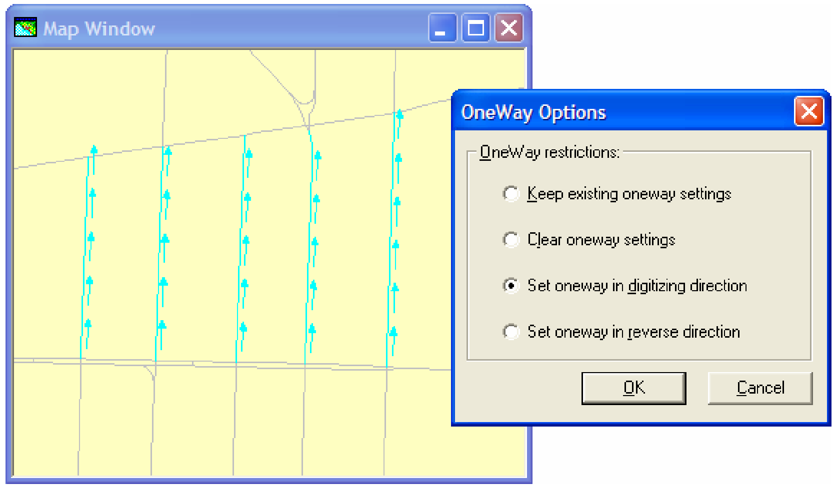
3. Select **Transportation > Routing Maintenance > OneWay Editor.**

   The **OneWay Editor** *dialog box will be displayed.*

4. Select an Edge feature class from the **Edit oneway restrictions in** pull-down list. The list will be limited to only those linear feature classes in your select set.

5. Select a Oneway field from the **Select oneway field** drop-down list.

6. Click **OK** to dismiss the **OneWay Editor** dialog box and bring up the **OneWay Options** dialog box.



7. Choose one of the four **OneWay** options. Their effect will be indicated in the map window by arrows showing allowed travel flow. If there are no arrows, that indicates that travel is allowed in both directions.

8. Once you have set the **OneWay** restrictions, click **OK** to save these changes to the database, or click **Cancel** to discard these changes.

Note: Due to the interactive nature of the routing analysis tools, any active routing analysis queries will be updated if the **Use OneWays** option was chosen.

---

26 – Working with the Blockage Editor Command

The **Blockage Editor** command allows you to interactively edit the blockage attributes of selected edges. This allows you to more accurately model the allowable navigation routes through a network. By default, every edge in the network model is assumed to be unblocked. An unblocked edge is one that can be traversed in either the forward direction (that is, in the direction of digitization or FromNode to ToNode) or the reverse direction (that is, in the reverse direction of digitization or ToNode to FromNode). However, using this command, individual edges can also be set to blocked in the forward direction, blocked in the reversed direction, or blocked in both directions.

The command uses visual feedback in the map window to facilitate the editing of blockage restrictions. The selected edges will be symbolized in the map window based on their blockage attribution. As you edit the blockage restrictions, the symbology of the selected edges will change to match the current setting.

Underlying this visual feedback is the editing of an integer value of the Blockage attribute of the Edge feature class. The supported integer values for this attribute, and their meanings, are listed below:

Value	Meaning
0 or Null	No blockage to navigation
1	Navigation blocked in reverse digitizing direction
2	Navigation blocked in digitizing direction
3	Navigation blocked in both directions

The Blockage Editor Workflow

1. Open a GeoWorkspace that contains one or more read/write connections.

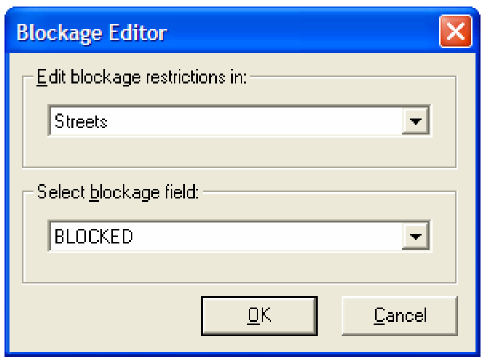
Note: Due to the dynamic nature of queries in GeoMedia, it is recommended that this command be run in a workspace with a minimal number of active queries that are based on your routing network components. That is because you may pay a substantial performance penalty when running these commands while waiting for these queries to update.

2. Select one or more edges in the map window that you wish to edit.

   This will cause the **Blockage Editor** *command to become enabled.*

3. Select **Transportation > Routing Maintenance > Blockage Editor.**

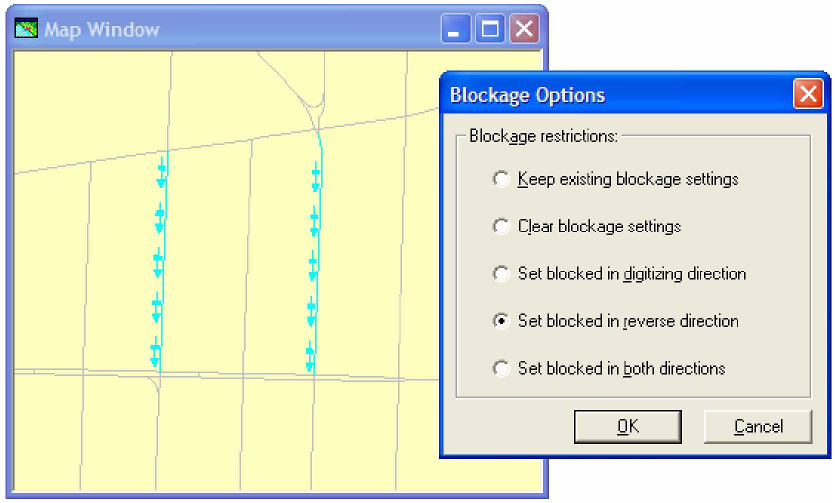
   The **Blockage Editor** *dialog box will be displayed.*



4. Select an Edge feature class from the **Edit blockage restrictions in** pull-down list. The list will be limited to only those linear feature classes in your select set.

5. Select a blockage field from the **Select blockage field** pull-down list.

6. Click **OK** to dismiss the **Blockage Editor** dialog box and bring up the **Blockage Options** dialog box.



7. Choose one of the five **Blockage** options. Their effect will be indicated in the map window by *blocked arrows* showing blocked travel flow. If there are no blocked arrows, that indicates that travel is unblocked in both directions.

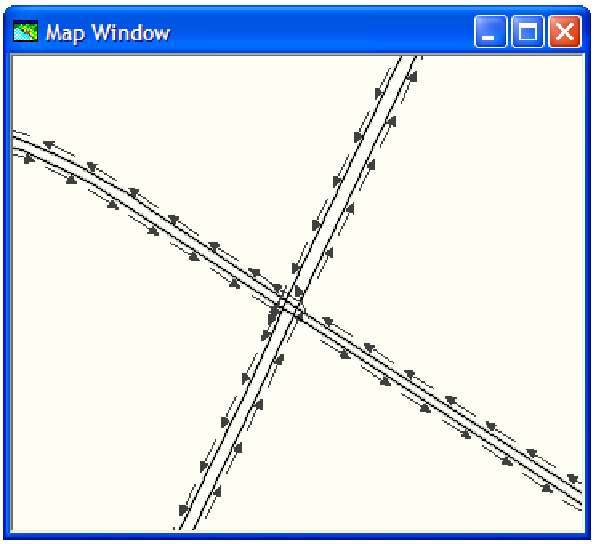
8. Once you have set the **Blockage** restrictions as desired, click **OK** to save these changes to the database, or click **Cancel** to discard these changes.

Note: Due to the interactive nature of the routing analysis tools, any active routing analysis queries will be updated if the **Use Blockages** option was chosen.

---

27 – Working with the Display OneWays Command

The **Display OneWays** command displays the direction of permitted movement in the map window for all those features of the selected edge feature class that have a one-way attribute set. The permitted direction of flow is indicated by arrows pointing in that direction. Edges without one-way restrictions have no arrows.



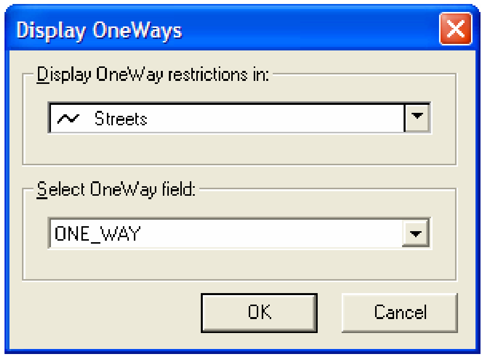
The command generates unique legend entry names that correspond to the arrow pattern for displaying the one-way in forward (that is, in the direction of digitization) and the reverse (that is, opposite to the direction of digitization) direction. The display of one-way can be switched off from the map view by deleting or making the display off for the corresponding legend entry(s).

The Display OneWays Workflow

This section presents the procedural steps for using **Display OneWays**.

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select **Transportation > Routing Maintenance > Display OneWays**.

   The **Display OneWays** *dialog box is displayed.*



3. Select the edge feature class from the **Display OneWay restrictions in** drop-down list.
4. Select the one-way field for the edge feature class in the **Select OneWay field** drop-down list.
5. Click **OK** to display the one-ways for the selected edge feature class in the map view.

   The one way edges in the network will be added to your map window and legend that display the one-way conditions of those edges that have a one-way restriction set. If there are no one-way restrictions set in the entire selected edge feature class, an error message to this effect will be displayed.

28 – Working with the Display Blockages Command

The **Display Blockages** command displays in the map window the direction of blockages for those edges of the selected edge feature class that have the Blockages attribute set. Blockages are flags used to restrict movement over the network. They are assumed to be more dynamic than one-way restrictions and can change quickly over time. If a blockage flag is not set, it is assumed that the edge feature is not blocked. An edge can be temporarily blocked due to construction, an accident, or some other reason. Blockages can limit travel in either or both directions. The blocked direction of flow is indicated by *blocked arrows* pointing in that direction. Edges without blockages have no blocked arrows.

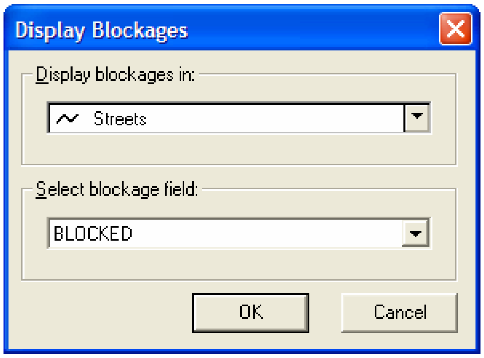
The command generates unique legend entry names that correspond to the arrow pattern for displaying the blockages in forward (that is, in the direction of digitization), the reverse (that is, opposite to the direction of digitization), and both direction(s). The display of blockages can be switched off from the map view by deleting or making the display off for the corresponding legend entry(s).

The Display Blockages Workflow

This section presents the procedural steps for using **Display Blockages**.

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select **Transportation > Routing Maintenance > Display Blockages**.

   The **Display Blockages** *dialog box is displayed.*

3. Select the edge feature class from the **Display blockages in** drop-down list.
4. Select the blockage field for the edge feature class in the **Select blockage field** drop down list.
5. Click **OK** to display the blockages for the selected edge feature class in the map view.

   The Blocked edges in the network will be added to your map window and legend that display the blockage conditions of those edges that have a blockage set. If there are no blockages set in the entire selected edge feature class, an error message to this effect will be displayed.

29 – Working with the Easy Path Command

The **Easy Path** command provides a simple method to find the best path for a vehicle that needs to make one or more stops (one vehicle, many stops). Route optimization may be by distance or by user cost. This command allows you to define a best path by simply clicking on **Network edges** displayed in a map window.

The **Configure Network** dialog box allows you to specify the inputs that define the network for computing the best path. Properties can be set on the tabs of this dialog box for the Edge, Node, and Turn feature classes. If a resultant path is to be optimized by cost, a tab is available to set the **Cost** fields. Specifying restrictor features on the **Restrictors** tab for Edges, Nodes, and Turns can restrict the results.

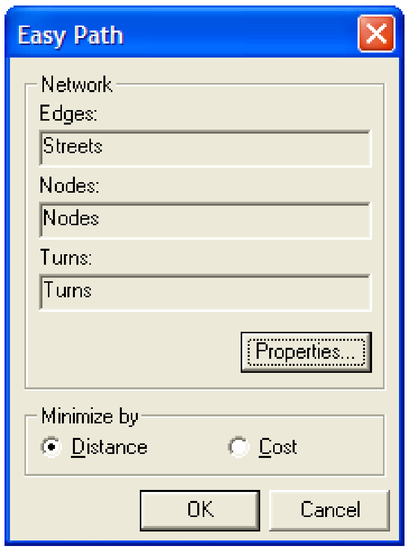
The order in which stops are visited in the path result is determined by the order in which you create stops by clicking on the map. A pop-up menu is provided to allow various levels of undo and also to allow you to output the Stops and Path as queries and display them on the map.

The Easy Path Workflow

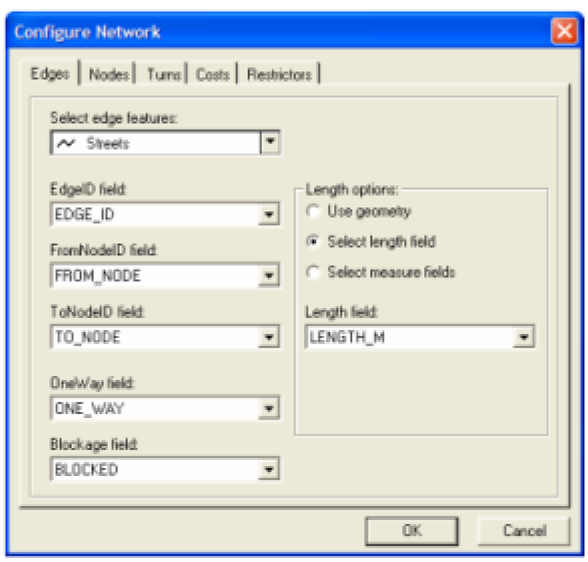
This section presents the procedural steps for using **Easy Path.**

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select the **Transportation > Routing Analysis > Easy Path.**

   The **Easy Path** *dialog box appears.*



3. Click **Properties** to select the inputs to define the network on the **Configure Network** dialog box.



4. Set the properties for the **Edges**, **Nodes**, and **Turns** feature classes on the dialog box tabs. If a resultant path is to be optimized by cost, a tab is available to set the **Costs** inputs. In order to restrict the results, specify restrictor features on the **Restrictors** tab.

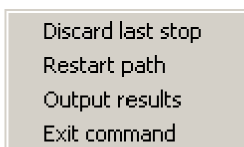
Note: See the "Working with the Configure Network Dialog Box" chapter for a detailed discussion of this dialog box.

---

5. Click **OK**.

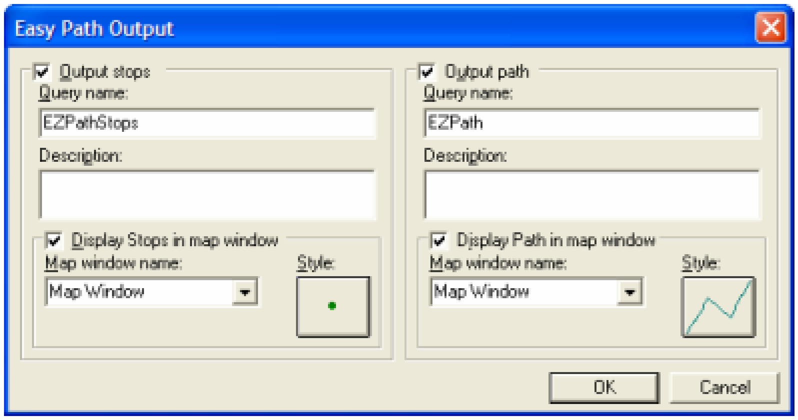
   The selected edge, node, and turn features are displayed on the **Easy Path** *dialog box in the* **Edges, Nodes***, and* **Turns** *text boxes.*

6. On the **Easy Path** dialog box, select **Distance** or **Cost** in the **Minimize by** box to optimize the path.
7. Click **OK**.
8. Follow the **First Stop** and **Next Stop** prompts to define a path by placing stops on the network in the map window.
9. At any point in this process you can bring up the Easy Path pop-up menu by right-clicking in the map window.



10. Select **Discard last stop** if you want to undo your last Stop click in the map window. The path will revert to what it was before this last Stop.
11. Select **Restart path** if you want to undo all of your Stop clicks in the map window. The path will be erased altogether.
12. Select **Output results** from the pop-up menu to save the completed path and stops to the query folder, and, optionally, to add them to the legend for display in the map window.

    The **Easy Path Output** *dialog box appears.*



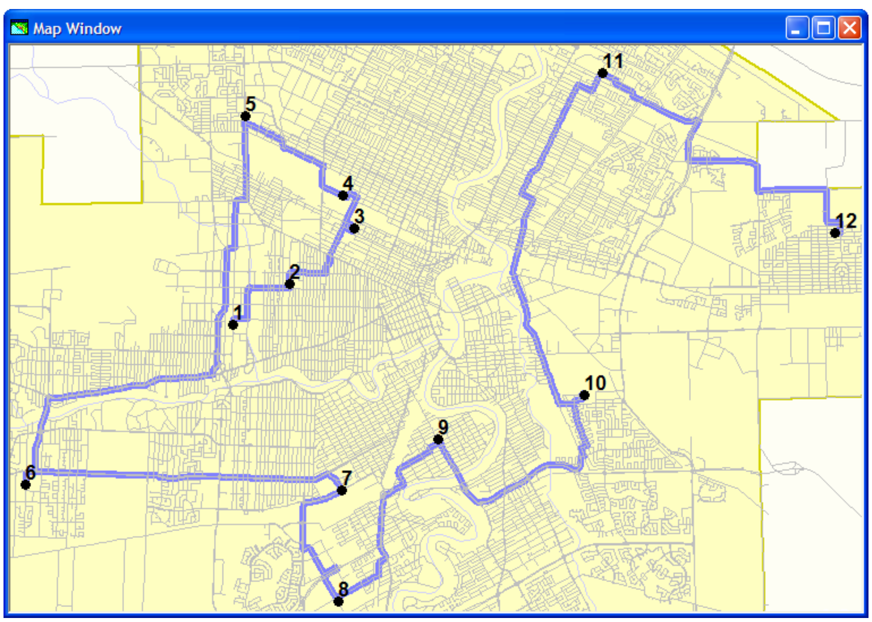
13. Check **Output Stops** to define a name and description for the stops query.
14. Check **Display Stops in map window** to set the style and to display the stops in the map window.
15. Check **Output path** to define a name and description for the path query.
16. Check **Display Path in map window** set the style and to display the path in the map window.
17. Select **Exit command** from the pop-up menu to exit the command.

30 – Working with the Best Path Command

The **Best Path** command will allow you to find an optimal Path through a given set of Stops on a network; for example, finding the best path for a vehicle that needs to make one or more stops. Path results can be optimized either by distance or by cost (for example, travel time). **Route options** and **Order options** specify the optimization criteria and the order in which stops are to be visited.

The **Configure Network** dialog box allows you to specify the inputs that define the network for computing the best path. Properties can be set on the tabs of this dialog box for the Edge, Node, and Turn feature classes. If a resultant path is to be optimized by cost, a tab is available to set the **Cost** fields. Specifying restrictor features on the **Restrictors** tab for Edges, Nodes, and Turns can restrict the results.

The output of the best path analysis will be a linear feature class showing the Path solution and, optionally, a point and text feature class used for Stop symbology. The results of the solution will be dynamic so that changes to stops, restrictions, blockages, and so on, are reflected in a change to the calculated paths.

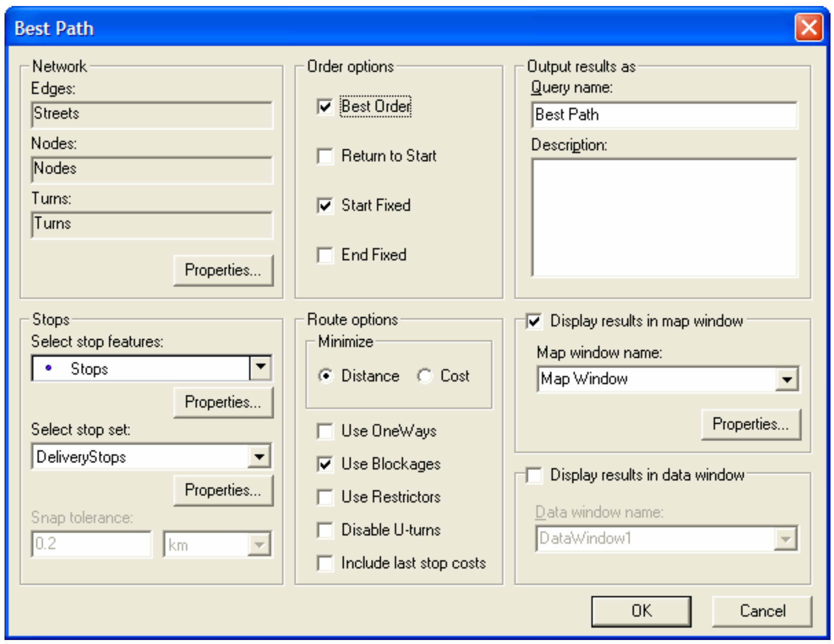


The Best Path Workflow

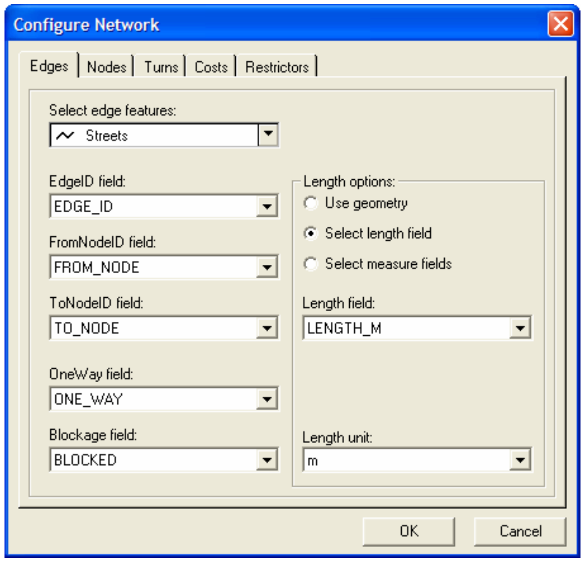
This section presents the procedural steps for using the **Best Path** command.

1. Open a GeoWorkspace that contains one or more readable connections.

2. Select **Transportation > Routing Analysis > Best Path.**



3. In the **Network** box, click **Properties** to select the inputs to define the network on the **Configure Network** dialog box.



4. Set the properties for the **Edges**, **Nodes**, and **Turns** feature classes on the dialog box tabs. A tab is available to set the **Costs** inputs. In order to restrict the results of the Path pipe, specify restrictor features on the **Restrictors** tab.

Note: See the "Working with the Configure Network Dialog Box" chapter for a detailed discussion of this dialog box.

---

5. Click **OK**.
6. Select a Stops feature class and Stop set or a Stop set query from the **Stops** portion of the dialog box.

Note: See the "Working with the Stop Properties Control" chapter for a detailed discussion on how to use this dialog box. For a discussion on the differences between Stop feature classes and Stop Set queries, see the "Working with the Stop Manager Command" chapter.

---

7. On the **Best Path** dialog box, optionally set the **Order options** to override the defaults. The four order options are **Best Order**, **Return to Start**, **Start Fixed**, and **End Fixed**.
   - The **Best Order** option tells the software to disregard the assigned order of the Stops and to optimize the order of the Stops so as to minimize either cost or distance.

- The **Return to Start** option means that the path created will both start and stop at the same Stop.
- The **Start Fixed** option slightly overrides the **Best Order** option to ensure that the first Stop in the path is the first Stop in the Stop Set.
- The **End Fixed** option slightly overrides the **Best Order** option to ensure that the last Stop in the path is the last Stop in the Stop Set.

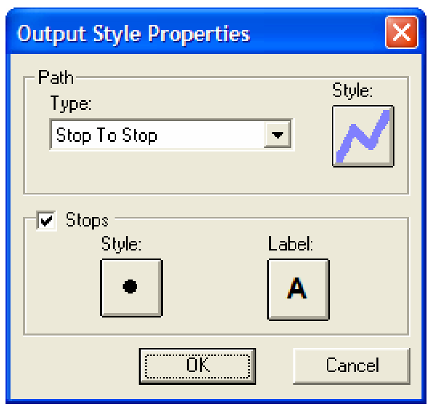
8. Optionally set the **Route options** to override the defaults. The six route options are **Minimize Distance/Cost**, **Use OneWays**, **Use Blockages**, **Use Restrictors**, **Disable U-turns**, and **Include last-stop costs**.

   - The **Minimize Distance/Cost** option allows you to select whether the path created minimizes the length of the path or the “cost” of the path. The cost is the sum of the values in the fields designated in the **Configure Network** dialog box as cost fields for Edges and, optionally, Nodes and/or Turns. These cost fields can be used to store any numeric value, such as time, monetary expense, or safety indices.
   - The **Use OneWays**, **Use Blockages**, and **Use Restrictors** options allow you to select whether to use or ignore the definitions set by you in the **Configure Network** dialog box for OneWays, Blockages, and Restrictors. When routing an emergency vehicle, for example, it might be useful to not **Use OneWays**; or when routing a regular-sized vehicle, it might be useful not to **Use Restrictors** developed to handle the limitations of 18-wheel vehicles.
   - The **Disable U-turns** option allows you to globally disable U-turns when calculating paths. This is particularly useful when routing large vehicles that cannot easily make U-Turns.
   - The **Include last-stop costs** option tells the software whether to include the cost associated with the last stop in a path in the accumulated cost for that path. If this option is not picked, the accumulated cost will indicate the cost to get to the last path, but not any cost associated with that last stop.

9. Provide a **Query name** if you do not like the default, and, optionally provide a **Description**.
10. Choose whether to display the results in a map window, and choose the specific map window if different from the current one.

11. Click **Properties** in the **Display results in map window** group box to specify the **Path Type** and its symbology on the **Output Style Properties** dialog box. The two **Type** options are **Individual Segments** and **Stop To Stop**. An Individual Segments path reflects each of the underlying edges. The Stop To Stop paths aggregate the individual segments into path segments that run from one stop to another.

    To set the Path symbology, pick the **Style** button in the **Path** group. Optionally, you can display Stops by picking the **Stops** option and can set the Stops display options by using the **Style** and **Label** buttons in the **Stops** group.

12. Click **OK**.

13. On the **Best Path** dialog box, choose whether to **Display the results in data window**, and choose the specific data window from the drop-down list.

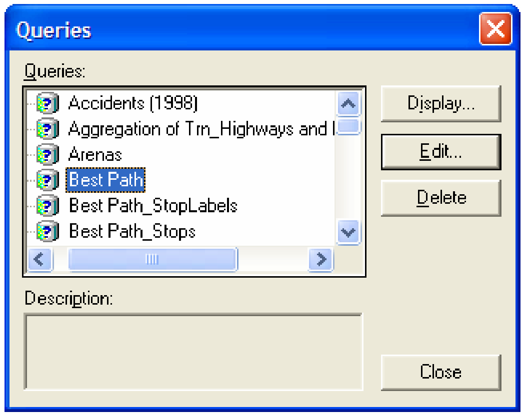
Note: The Query name, Description, Display results in map window and Display results in data window boxes are enabled only when valid input is given in the **Configure Network** dialog box and when a valid stop feature class has been selected.

---

14. Click **OK** to initiate the Best Path calculation

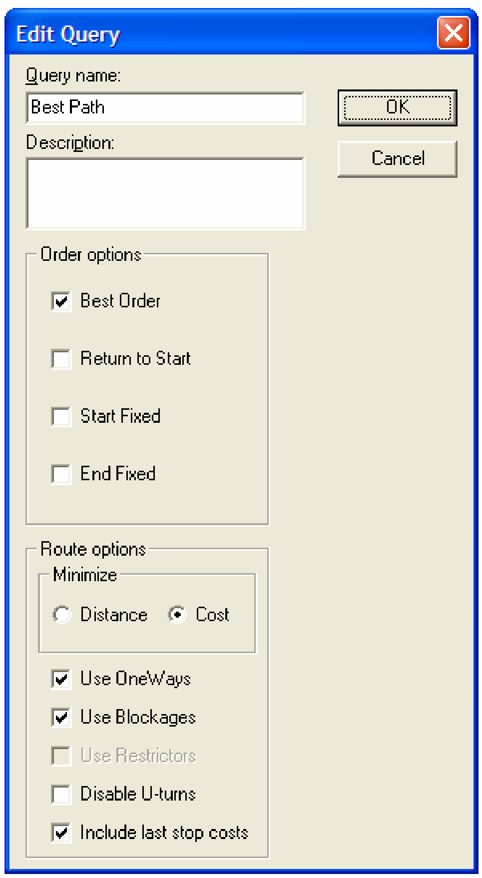
15. After reviewing the result, you can change the order and route options by editing the query. Do this by first selecting **Analysis > Queries** from the GeoMedia Professional menu bar.

The **Queries** *dialog box is displayed.*



16. Select the query generated by the **Best Path** command, and then click **Edit.**

The **Edit Query** *dialog box is displayed.*



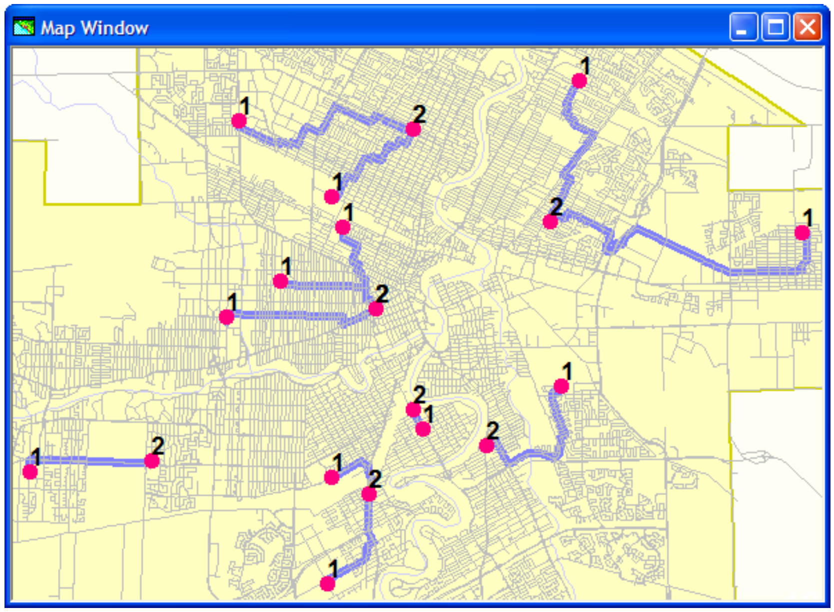
17. Edit the **Query** name, the **Description**, the **Order options**, and the **Route options**. When finished click **OK**. This will alter the results of the query according to your new input.

31 – Working with the Find Closest Stops Command

The **Find Closest Stops** command allows you to find the closest *N* destination stops to a set of origin stops. This is useful for a number of tasks, including making assignments or for emergency pre-planning, such as finding the two closest hospitals to each sporting arena in a particular jurisdiction. Path results can be optimized either by distance or by cost (for example, travel time). **Path solution options** and **Route options** specify the optimization criteria and the number of paths to be created.

The **Configure Network** dialog box allows you to specify the inputs that define the network for computing the find closest stops solutions. Properties can be set on the tabs of this dialog box for the Edge, Node, and Turn feature classes. If a resultant path is to be optimized by cost, a tab is available to set the **Cost** fields. Specifying restrictor features on the **Restrictors** tab for Edges, Nodes, and Turns can restrict the results.

The output of the find-closest-stops analysis (also known as proximity analysis) will be a linear feature class showing the Path solution and, optionally, a point and text feature class used for Stop symbology. The results of the solution will be dynamic so that changes to stops, restrictions, blockages, and so on, are reflected in a change to the calculated paths, as shown in the following example:

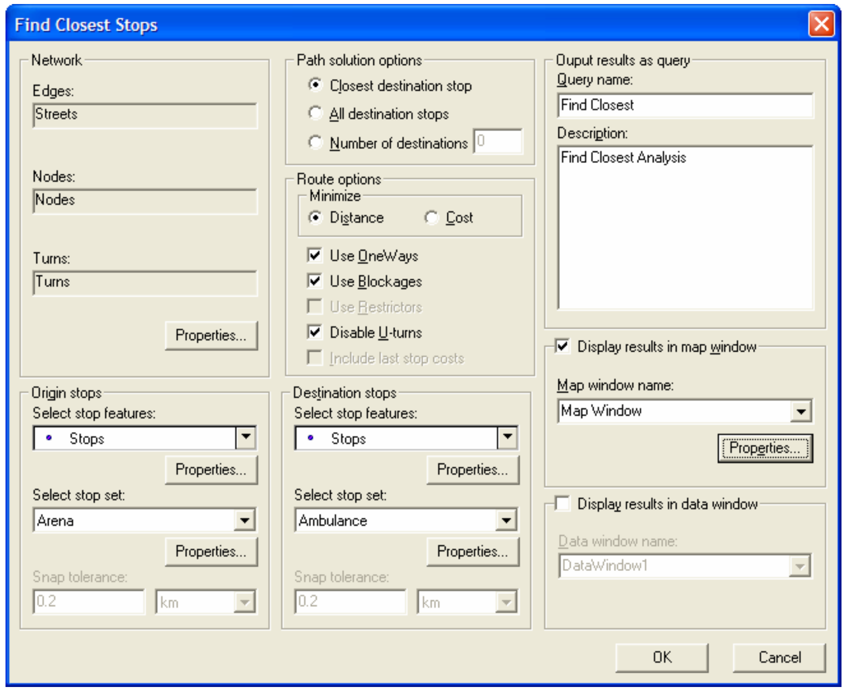


The Find Closest Stops Workflow

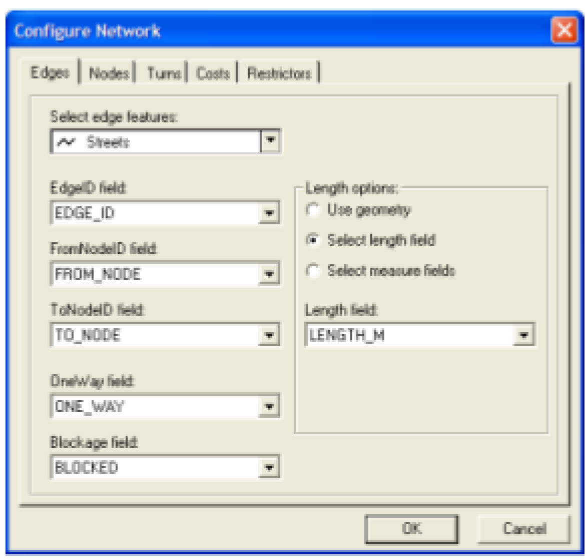
This section presents the procedural steps for using **Find Closest Stops.**

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select **Transportation > Routing Analysis > Find Closest Stops.**

   The **Find Closest Stops** *dialog box is displayed.*



3. Click **Properties** in the **Network** box to select the inputs to define the network on the **Configure Network** dialog box.



4. Set the properties for the **Edges**, **Nodes**, and **Turns** feature classes on the dialog box tabs. If a resultant path is to be optimized by cost, a tab is available to set the **Costs** inputs. In order to restrict the results, specify restrictor features on the **Restrictors** tab.

Note: See the "Working with the Configure Network Dialog Box" chapter for a detailed discussion of this dialog box.

---

5. Click **OK**.

   The selected Edge, Node, and Turn features are displayed on the **Find Closest Stops** *dialog box in the* **Edges, Nodes**, *and* **Turn***s fields.*

6. Select a Stops feature class and Stop set or a Stop set query from the **Origin stops** portion of the dialog box.

Note: See the "Working with the Stop Properties Control" chapter for a detailed discussion on how to use this dialog box. For a discussion on the differences between Stop feature classes and Stop Set queries, see the "Working with the Stop Manager Command" chapter.

---

7. Repeat Step 6 for the **Destination stops**. These stops define the set of end points that will be selected from in this analysis.

8. Select **Closest destination stop** (the default), **All destination stops**, or **Number of destinations** (enter the number of destination paths in the associated text box if this option is chosen) from the **Path solution options** group.

   - The **Closest destination stop** option generates a path solution from each stop in the origin stop set to the single closest destination to each origin stop.

   - The **All destination stops** option generates a path solution from each stop in the origin stop set to each and every stop in the destination stop set.

   - The **Number of Destinations** option allows you to specify a number of destinations to generate a path solution from each stop in the origin stop set to the specified number of closest stops in the destination stop set. For example if you use a value of 2, it will find the Paths between each Origin Stop and the closest Destination Stop to each, as well as the Paths between each Origin Stop and the second closest Destination Stop.

9. Optionally set the **Route options** to override the defaults. The six route options are **Minimize Distance/Cost**, **Use OneWays**, **Use Blockages**, **Use Restrictors**, **Disable U-turns**, and **Include last-stop costs**.

The **Minimize Distance/Cost** option allows you to select whether the path created minimizes the length of the path or the “cost” of the path. The cost is the sum of the values in fields designated in the **Configure Network** dialog box as cost fields for Edges and, optionally, Nodes and/or Turns. These cost fields can be used to store any numeric value, such as time, monetary expense, or safety indices.

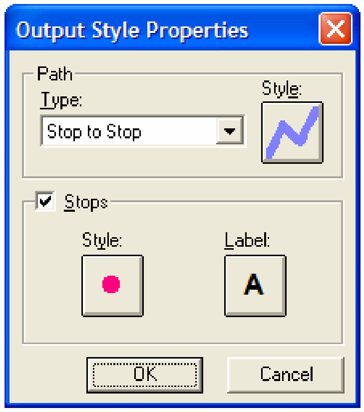
The **Use OneWays**, **Use Blockages**, and **Use Restrictors** options allow you to select whether to use or ignore the definitions set by the user in the **Configure Network** dialog box for OneWays, Blockages, and Restrictors. When routing an emergency vehicle, for example, it might be useful to not **Use OneWays**, or when routing a regular-sized vehicle, it might be useful not to **Use Restrictors** developed to handle the limitations of 18-wheel vehicles.

The **Disable U-turns** option allows you to globally disable U-turns when calculating paths. This is particularly useful when routing large vehicles that cannot easily make U-Turns. This option does not require that a turn table be defined.

The **Include last-stop costs** option tells the software whether to include the cost associated with the last stop in a path in the accumulated cost for that path. If this option is not picked, the accumulated cost will indicate the cost to get to the last path, but not any cost associated with that last stop.

10. Accept or override the default query name specified in the **Query name** field, and assign an optional description in the **Description** field.
11. To display the results in a map window, verify that the **Display results in map window** is checked. Then accept the default, or select or key in a map window name.
12. Click **Properties** in the **Display results in map window** group box to specify the **Path Type** and its symbology on the **Output Style Properties** dialog box. The two **Type** options are **Individual Segments** and **Stop To Stop**. An Individual Segments path reflects each of the underlying edges. The Stop To Stop paths aggregate the individual segments into path segments that run from one stop to another.

    To set the Path symbology, click **Style** in the **Path** group. Optionally, you can display and label Stops by clicking the **Stops** check box and can set the Stops display options by checking the **Style** and **Label** buttons in the **Stops** group.



13. To display the results in a data window, verify that the **Display results in data window** is checked. Then accept the default, or select or key in a data window name.

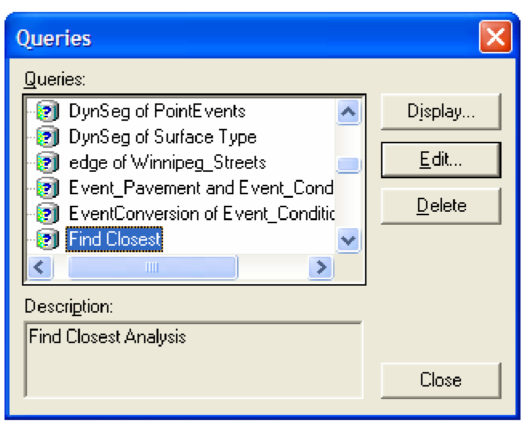
Note: The **Query name**, **Description**, **Display results in map window** and **Display results in data window** fields are enabled only when valid input is given in the **Configure Network** dialog box and when valid stop feature classes have been selected.

---

14. Click **OK** to initiate the Find Closest Stops calculation.

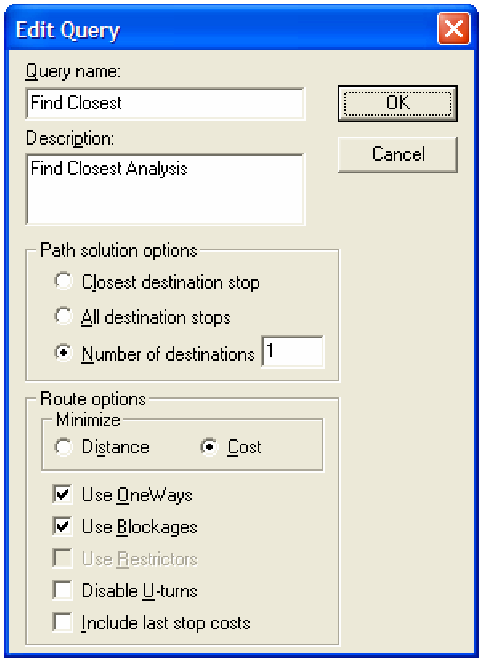
15. After reviewing the result, you can change the path solution and route options by editing the query. Do this by first selecting **Analysis > Queries** from the GeoMedia Professional menu bar.

    The **Queries** *dialog box is displayed.*



16. Select the query generated by the **Find Closest Stops** command, and then click **Edit.**

The **Edit Query** *dialog box is displayed.*



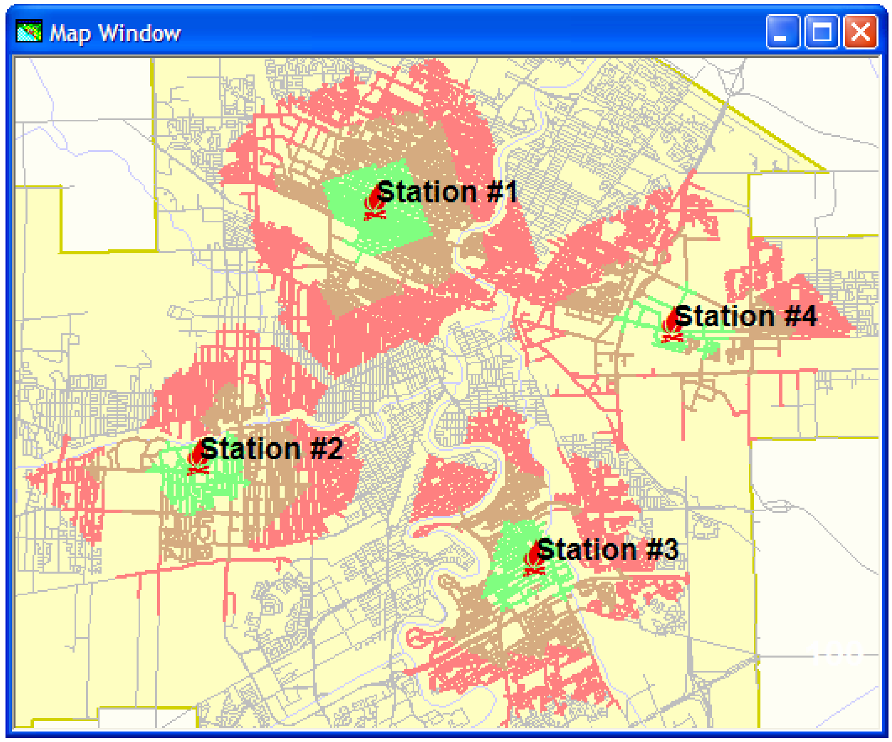
17. Edit the **Query name**, the **Description**, the **Path solution options**, and the **Route options.** When finished click **OK**. This will alter the results of the query according to your new input.

32 – Working with the Network Coverage Command

The **Network Coverage** command allows you to find all paths that can be traversed from a set of origin Stops within a given user cost or distance. This is useful, for example, for planning what locations can be reached by emergency response teams in a given time from a given location. Path results can be optimized either by distance or by cost (for example, travel time). Coverage options and Route options specify the optimization criteria and the number of bands of solutions to be created (for example, 3 bands at 5 minutes would generate 5, 10, and 15 minute solutions).

The **Configure Network** dialog box allows you to specify the inputs that define the network for computing the network coverage solutions. Properties can be set on the tabs of this dialog box for the Edge, Node, and Turn feature classes. If a resultant path is to be optimized by cost, a tab is available to set the **Cost** fields. Specifying restrictor features on the **Restrictors** tab for Edges, Nodes, and Turns can restrict the results.

The output of the network coverage analysis will be a thematic linear feature class showing the Path solution and, optionally, a point and text feature class used for Stop symbology. The results of the solution will be dynamic so that changes to stops, restrictions, blockages, and so on, are reflected in a change to the calculated paths.

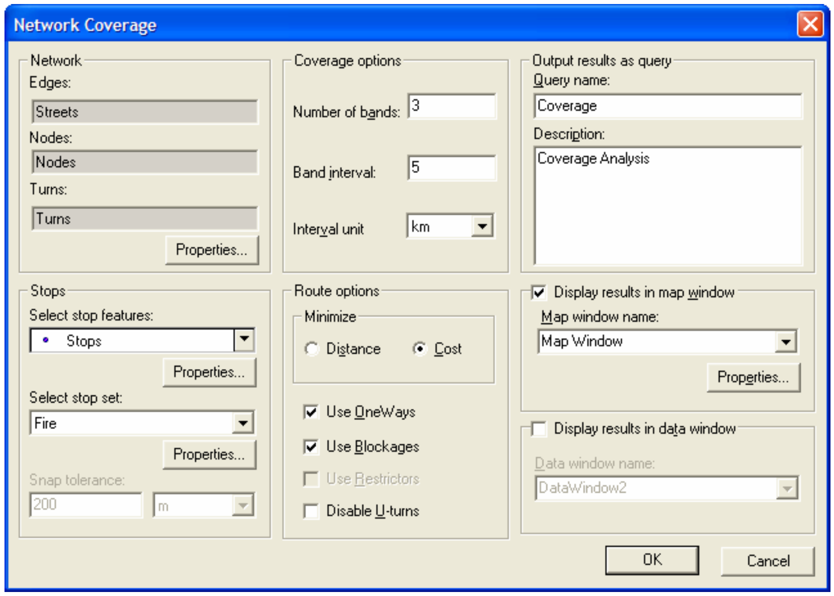


The Network Coverage Command Workflow

This section presents the procedural steps for using the **Network Coverage** command.

1. Open a GeoWorkspace that contains one or more readable connections.
2. Select **Transportation > Routing Analysis > Network Coverage.**

   The **Network Coverage** *dialog box is displayed.*



3. Click **Properties** in the **Network** box to set the properties for the network on the **Configure Network** dialog box.
4. Set the properties for the **Edges**, **Nodes**, and **Turns** feature classes on the dialog box tabs. If a resultant path is to be optimized by cost, a tab is available to set the **Costs** inputs. In order to restrict the results, specify restrictor features on the **Restrictors** tab.

Note: See the "Working with the Configure Network Dialog Box" chapter for a detailed discussion of this dialog box.

---

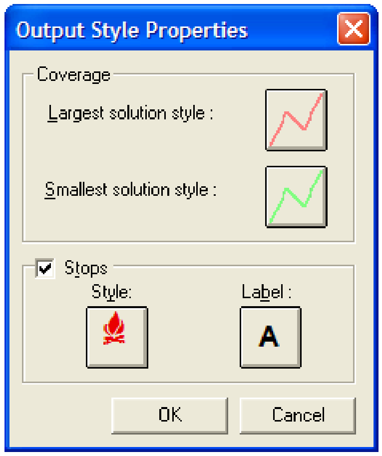
5. Click **OK**.
6. Select a Stops feature class and Stop set or a Stop set query from the **Stops** portion of the dialog box.

Note: See the "Working with the Stop Properties Control" chapter for a detailed discussion on how to use this dialog box. For a discussion on the differences between Stop feature classes and Stop Set queries, see the "Working with the Stop Manager Command" chapter.

---

7. Enter values for the **Number of bands** and **Band interval**. For instance if you select three bands with an interval of 500, a thematic feature class will be created showing paths up to 500 units from the origin, 1000 units from the origin, and 1500 units from the origin.

8. Select the unit of measure for your interval key-in from the **Interval unit** drop-down list if you are optimizing distance (see next step). If you are optimizing by cost, then you will be shown the cost units set in the **Configure Network** dialog box. All costs (Edge, Node, Turn, and Stop) are assumed to share these same units.
9. Optionally set the **Route options** to override the defaults. The five route options are **Minimize Distance/Cost**, **Use OneWays**, **Use Blockages**, **Use Restrictors**, and **Disable U-turns**.
   - The **Minimize Distance/Cost** option allows you to select whether the path created minimizes the length of the path or the "cost" of the path. The cost is the sum of the values in fields designated in the **Configure Network** dialog box as cost fields for Edges and, optionally, Nodes, and/or Turns. These cost fields can be used to store any numeric value, such as time, monetary expense, safety indices, etc.
   - The **Use OneWays**, **Use Blockages**, and **Use Restrictors** options allow the user to select whether to use or ignore the definitions set by the user in the **Configure Network** dialog box for OneWays, Blockages and Restrictors. When routing an emergency vehicle, for example, it might be useful to not **Use OneWays**, or when routing a regular-sized vehicle it might be useful not to **Use Restrictors** developed to handle the limitations of 18-wheel vehicles.
   - The **Disable U-turns** option allows the user to globally disable U-turns when calculating paths. This is particularly useful when routing large vehicles that cannot easily make U-Turns. This option does not require a turn table to be defined.
10. Accept or override the default query name specified in the **Query name** field, and assign an optional description.
11. To display the results in a map window, check **Display results in map window**. Then, accept the default, or select or key-in a map window name. Click **Properties** to change the coverage and/or the stops style on the **Output Style Properties** dialog box.



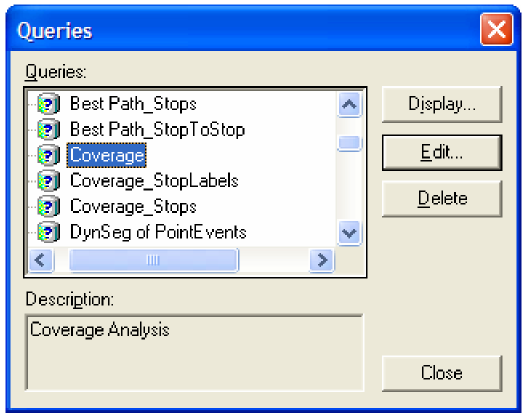
12. To change the coverage styles, click **Largest solution style** and **Smallest solution style**. These set the styles for the first and last coverage band. Intermediate bands have a style that is interpolated between these two styles.
13. Check **Stops** if you want the origin Stops to be displayed and labeled. Click the **Style** and **Label** buttons to change the symbology options.
14. To display the results in a data window, check **Display results data window**. Then, accept the default, or select or key-in a map window name.

Note: The **Query name**, **Description**, **Display results in map window** and **Display results in data window** fields are enabled only when valid input is given in the **Configure Network** dialog box and when a valid stop feature class has been selected.

---

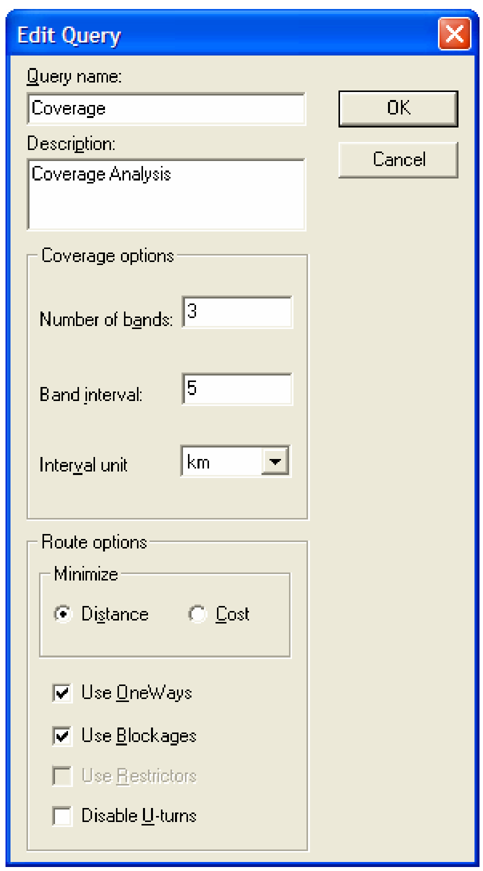
15. Click **OK** to initiate the Network Coverage calculation.
16. After reviewing the result, you can change the coverage and route options by editing the query. Do this by first selecting **Analysis > Queries** from the GeoMedia Professional menu bar.

The **Queries** *dialog box is displayed.*



17. Select the query generated by the **Network Coverage** command, and then click **Edit.**

    The **Edit Query** *dialog box is displayed.*



18. Edit the **Query** name, the **Description**, the **Coverage options**, and the **Route options**. When finished click **OK**. This will alter the results of the query according to your new input.

33 – Working with the Generate Path Directions Command

The **Generate Path Directions** command allows you to generate and display descriptive direction information for Best Path or Find Closest Stops result. You specify a feature class or query of Best Path or Find Closest Stops results and a directions configuration file like the one that is delivered with GeoMedia Transportation.

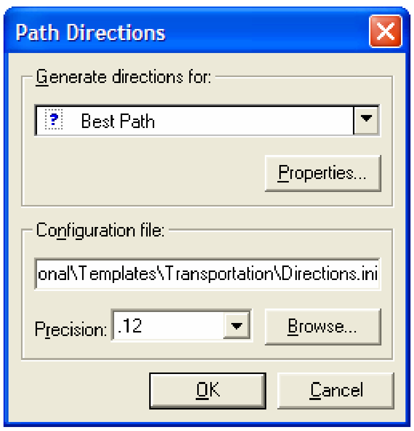
The command also allows you to specify the path's descriptive fields and the travel lengths and costs. You can also specify the decimal precision that is used to display travel lengths and costs, and you can set the units for length displays.

The Generate Path Directions Command Workflow

This section presents the procedural steps for using the **Network Directions** command.

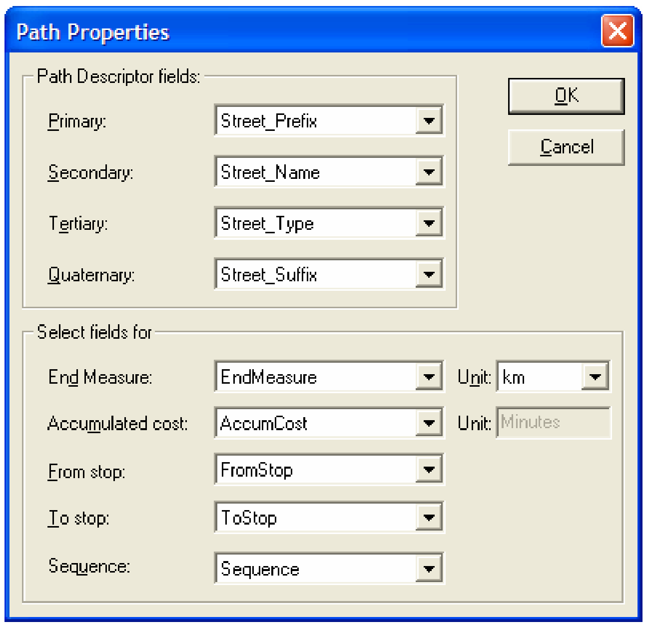
1. Open a GeoWorkspace that contains one or more connections.
2. Run a **Best Path** or **Find Closest Stops** analysis. For more information, see the "Working with the Best Path Command" or the "Working with the Find Closest Stops Command" chapter.
3. Select **Transportation > Routing Analysis > Generate Path Directions.**

   The **Path Directions** *dialog box is displayed.*



4. Select a Path feature class in the **Generate directions for** drop-down list. These are the outputs of either the **Best Path** or **Find Closest Stops** command.
5. Click **Properties**.

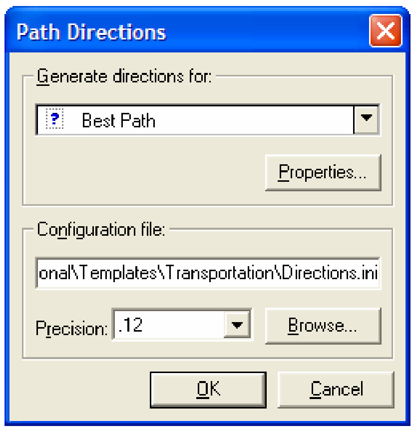
   The **Path Properties** *dialog box is displayed.*



6. Select the **Primary**, **Secondary**, **Tertiary**, and **Quaternary** descriptor fields in the appropriate drop-down lists. You can use as many of these fields as you need to fully describe the road names (or other types of linear network) being traveled. If more than one field is used here, the road name shown in the directions will be a concatenation of these fields with a space between each field value.
7. Select the **End Measure** field from the drop-down list. This field defines the accumulated length traveled to the end of each path segment. Then select the **Unit** that you want to use for display of travel distances in the **Unit** drop-down list.
8. Select the **Accumulated cost** field from the drop-down list. This field defines the accumulated cost of travel to the end of each path segment. Then **Unit** of this cost is displayed in the read-only Unit field.
9. Select a **From Stop** field from the drop-down list. This is a field created by either the **Best Path** or **Find Closest Stops** command to define the stop at the beginning of a path segment.

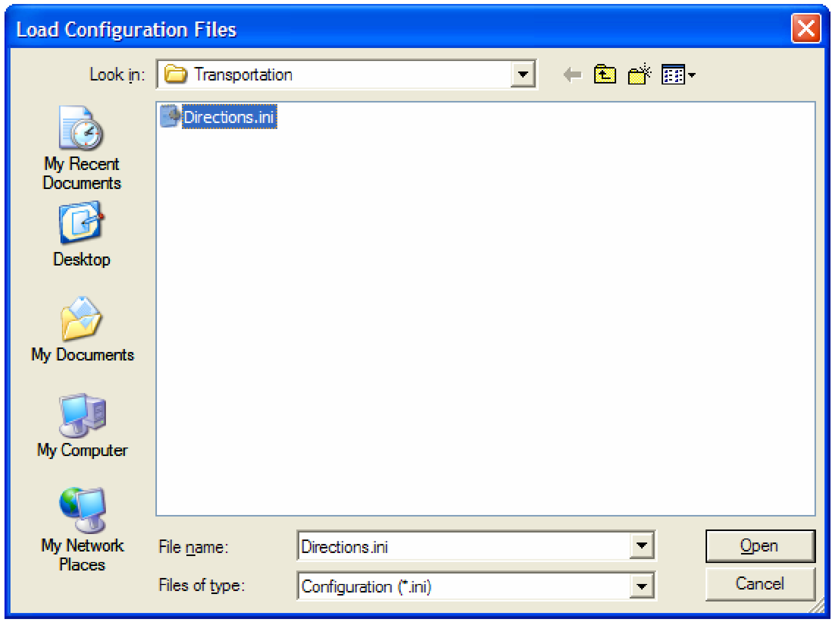
10. Select a **To Stop** field from the drop-down list. This is a field created by either the **Best Path** or **Find Closest Stops** command to define the stop at the end of a path segment.
11. Select a **Sequence** field from the drop-down list. This is a field created by either the **Best Path** or **Find Closest Stops** command to define the place of each path segment within the entire path result.
12. Click **OK** to save these inputs, or click **Cancel** to exit the dialog without saving.

    You are returned to the **Path Directions** *dialog box.*



13. On the **Path Directions** dialog box, click **Browse** to select a directions configuration file from an open dialog box. A default configuration file is delivered with the product at *<Product Directory>\Templates\Transportation\Directions.ini.* This is a self-documented file that defines the format of the directions to be created. You can edit this file.

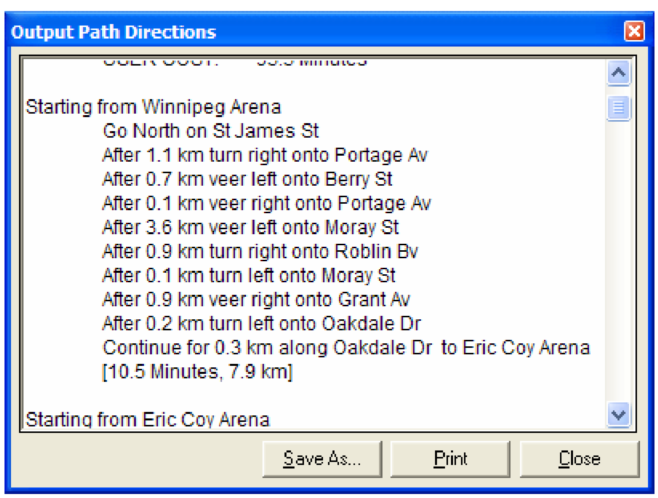
    The **Load Configuration Files** *dialog box is displayed.*



14. Select a display precision in the **Precision** combo box. This sets the number of decimal places used in the display of cost and length values.

15. Click **OK** to process the inputs, or click **Cancel** to exit without processing.

    The **Output Path Directions** *dialog box is displayed if you clicked* **OK**.



16. Click **Save As** to save the directions as a text file, and click **Print** to print the directions; or click **Close** to exit the command without saving or printing.

34 – Working with the Configure Network Dialog Box

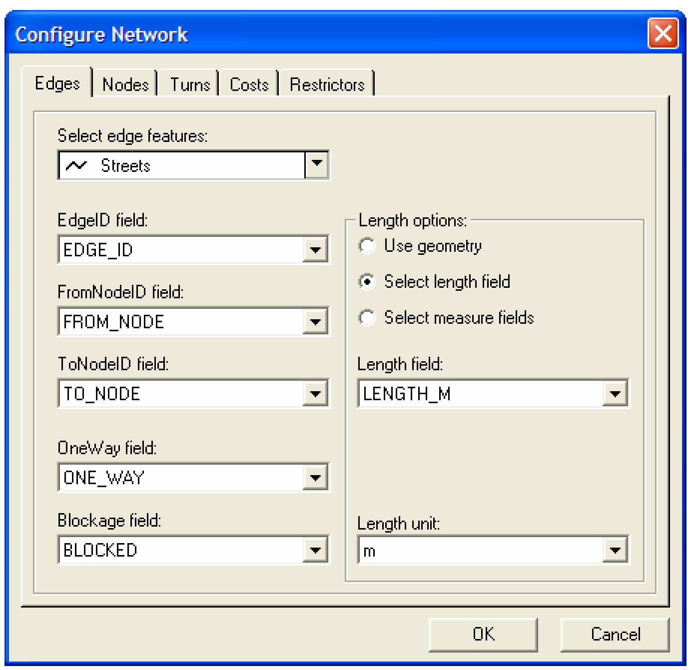
This section provides an overview of the **Configure Network** dialog box, which provides a graphic user interface for entering Network properties in GeoMedia Transportation Manager. Use this dialog box in the **Easy Path**, **Best Path**, **Find Closest Stops**, and **Network Coverage** commands to input the Network properties. This dialog box provides a standard way for entering this information.

The Configure Network Dialog Box Workflow

This section presents the procedural steps for using **Configure Network** dialog box.

1. Click the **Edge** tab.

   The **Edge Properties** *group is displayed.*



2. In the **Select edge features** field, select an Edge feature class. This is the linear feature class along which the networking analysis will be conducted.

3. In the **EdgeID**, **FromNodeID**, and **ToNodeID field** fields, select the names of the topology fields for this Edge class. For information on how to create and populate these fields, see the "Working with the Build Network Command" chapter.

4. *Optional:* In the **OneWay field** box, select the field name of the **OneWay** field. For information on how to populate this field see the "Working with the OneWay Editor Command" chapter.

5. *Optional:* In the **Blockage field** box, select the field name of the **Blockage** field. For information on how to populate this field, see the "Working with the Blockage Editor Command" chapter.

6. In the **Length** options group you have three choices:

   - The first is to select **Use Geometry** and have the system calculate lengths based on geometry lengths of the Edges. This method has the slowest processing time due to the extra calculations involved.

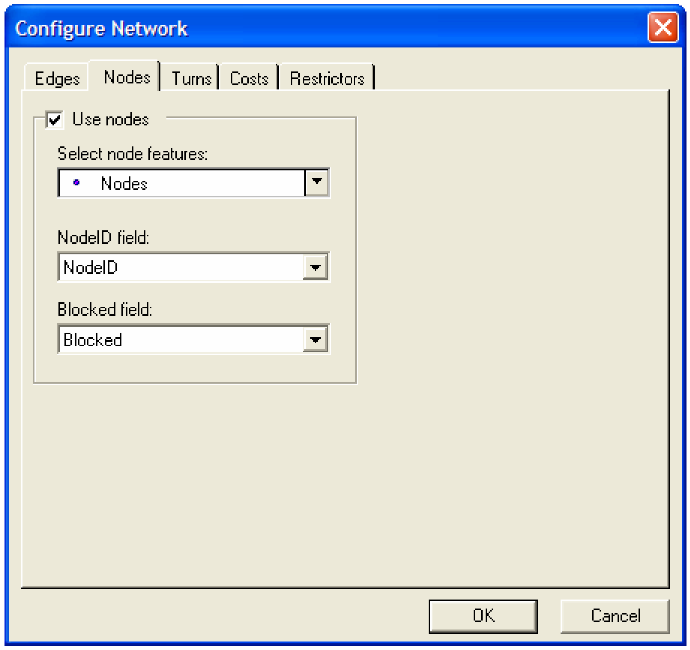
   - The second option is to choose **Select length field** and pick a single field containing length values.

   - The third option is to choose **Select measure fields** and pick both a **Start measure** and an **End measure** field. The difference between the values in these two fields will define the length of the Edge.

7. If you picked either the **Select length field** or the **Select measure fields** option, then also pick the length unit of measure from the **Length unit** drop-down list. If you picked the **Use Geometry** option, then the distance units set for this workspace are displayed in a read-only field.

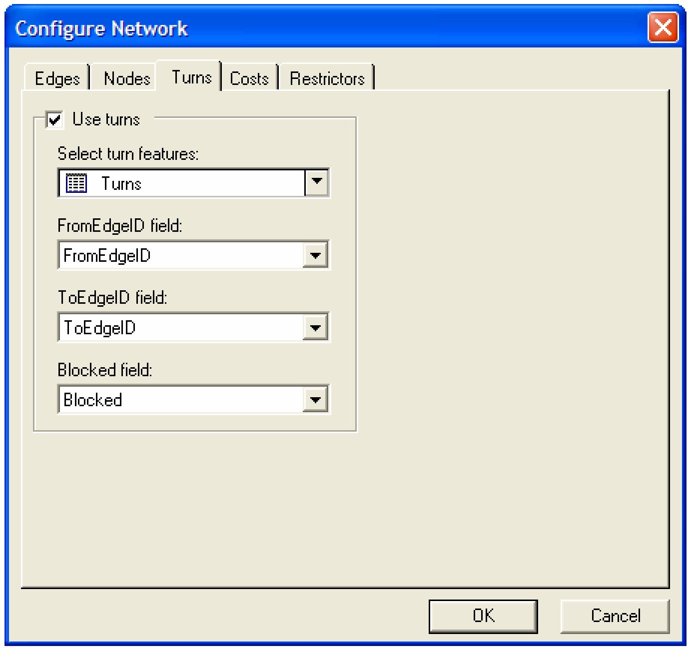
8. Click the **Node** tab.

   The **Node Properties** *group is displayed.*



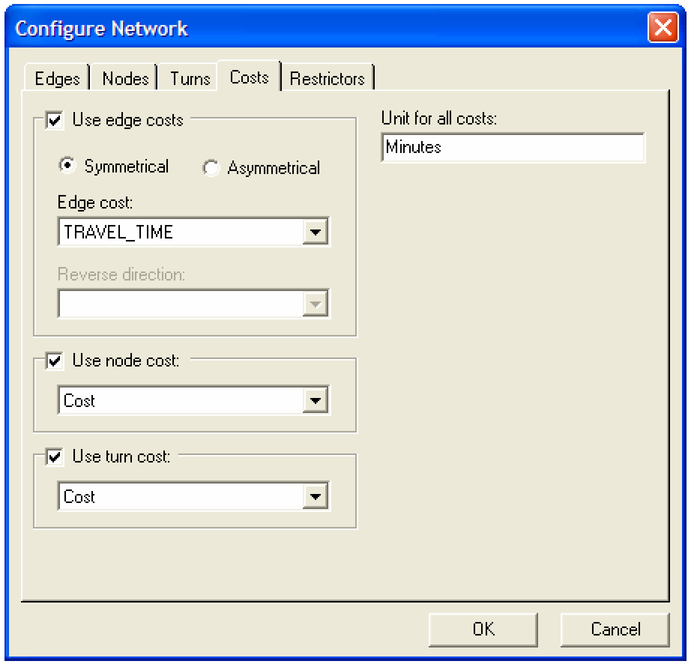
9. *Optional:* Select the **Use nodes** option. If this step is not chosen as an option, skip to Step 13.
10. In the **Select node features** field, select a Node feature class. This is the point feature class that defines the intersections in our network.
11. In the **NodeID field** field, select the name of the **NodeID** field for this Node class. For information on how to create and populate this field see the “Working with the Build Network Command” chapter.
12. *Optional:* In the **Blocked field** box, select the name of the **Blocked** field. This contains true or false values as to whether each Node is blocked or not.
13. Click the **Turns** tab.

    The **Turn Properties** *group is displayed.*



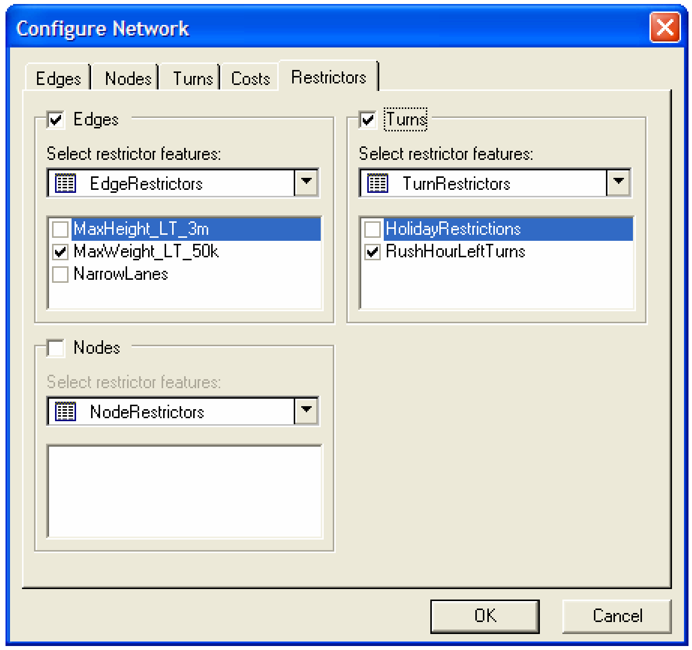
14. *Optional:* Select the **Use turns** option. If this step is not chosen as an option, skip to Step 18.
15. In the **Select turn features** field, select a Turn feature class. This is the non-graphic feature class that defines the turn costs and restrictions within our network.
16. In the **FromNodeID** and **ToNodeID field** fields, select the names of the Edge fields for this Turn class. For information on how to create and populate these fields see the "Working with the Build Network Command" chapter.
17. *Optional:* In the **Blocked field** field, select the name of the **Blocked** field. This contains true or false values as to whether each Turn option is blocked or not. For information on how to edit these records, see the "Working with the Turn Editor Command" chapter.
18. Click the **Costs** tab.

    The **Cost Properties** *group is displayed.*



19. *Optional:* Select the **Use edge costs** option. If this step is not chosen as an option, skip to Step 22.
20. Pick **Symmetrical** if the edge costs are the same in both travel directions, and **Asymmetrical** if they are not.
21. If you chose **Symmetrical**, select the **Edge Cost** field name from the **Edge Cost** box. If you chose **Asymmetrical**, select both the **Forward direction** (digitizing direction) and **Reverse direction** field names from the **Forward direction** and **Reverse direction** fields.
22. *Optional:* Select the **Use node cost** option. If this step is not chosen as an option, skip to Step 24.
23. Select the **Node cost** field name from the **Use node cost** field.
24. *Optional:* Select the **Use turn cost** option. If this step is not chosen as an option, skip to Step 26.
25. Select the **Turn cost** field name from the **Use turn cost** field.
26. If you chose any of the costs options, then also fill out the units of measure for those costs in **Unit for all costs** field. All costs (Edge, Node, Turn, and Stop) are required to share these same units.
27. Click the **Restrictors** tab.

    The **Restrictor Properties** *group is displayed.*



28. *Optional:* Select the **Edges** option. If this step is not chosen as an option, skip to Step 31.

29. In the **Select restrictor features** field, select an Edge Restrictor feature class. This is the non-graphic feature class that defines the conditions under which travel along certain edges may be restricted. For information on how to create and populate this table see the "Working with the Restrictor Editor Command" chapter.

30. The list box under the **Select restrictor features** field will show all of the restrictor records from this Edge Restrictor feature class. Check off those you wish to use for this analysis.

31. *Optional:* Select the **Nodes** option. If this step is not chosen as an option, skip to Step 34.

32. In the **Select restrictor features** field, select a Node Restrictor feature class. This is the non-graphic feature class that defines the conditions under which travel past certain nodes may be restricted. For information on how to create and populate this table see the "Working with the Restrictor Editor Command" chapter.

33. The list box under the **Select restrictor features** field will show all of the restrictor records from this Node Restrictor feature class. Check off those you wish to use for this analysis.

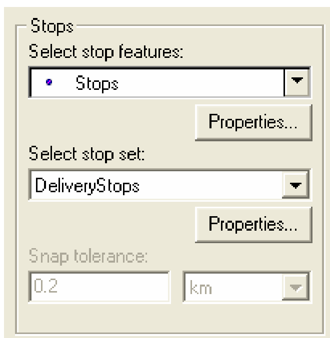
34. *Optional:* Select the **Turns** option. If this step is not chosen as an option, skip to Step 37.

35. In the **Select restrictor features** field, select a Turn Restrictor feature class. This is the non-graphic feature class that defines the conditions under which travel along certain turn options may be restricted. For information on how to create and populate this table see the "Working with the Restrictor Editor Command" chapter.
36. The list box under the **Select restrictor features** field will show all of the restrictor records from this Turn Restrictor feature class. Check off those you wish to use for this analysis.
37. Click **OK** after setting the appropriate values or **Cancel** to discard your changes.

35 – Working with the Stop Properties Control

This section provides an overview of the **Stop Properties** control**,** which provides a graphic user interface for entering Stop and Stop set properties in GeoMedia Transportation Manager. For more on the subject of Stops and Stop sets, see the "Working with the Stop Manager Command" chapter.

Use this control in the **Best Path**, **Find Closest Stops**, and **Network Coverage** commands to enter Stop and stop Set properties. This control provides a standard method for entering this information.



The Stop Properties Control Workflow

This section presents the procedural steps for selecting Stops and Stop sets and setting their properties.

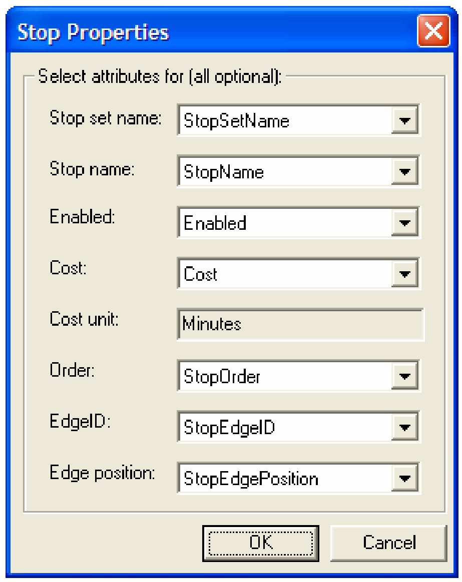
1. Select a Stops feature class or a Stop set query from the **Select stop features** drop-down list. This is the source of Stops to be used in this analysis.

   If a Stop feature class was selected, the **Select stop set** *drop-down menu will be populated with the Stops sets stored in the selected class. If a Stop set query was selected, the* **Select stop features Properties** *button, the* **Select stop set** *drop-down list, and the* **Select stop set Properties** *button will be disabled. That is because all of these options are preset with a Stop set query, and you can skip the rest of the steps in this chapter.*

Note: For a discussion on the differences between Stop feature classes and Stop set queries, see the "Working with the Stop Manager Command" chapter.

2. Click the **Select stop features Properties** button to set the properties for the Stop feature classes.

The **Stop Properties** *dialog box is displayed.*

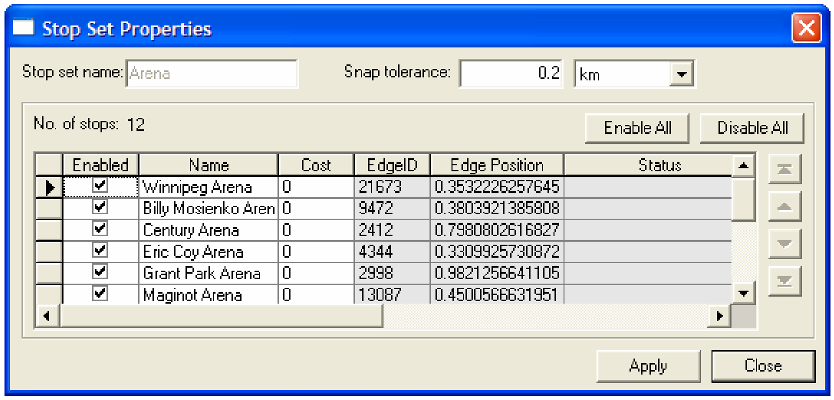


Note: Depending on which command you are using, some of the items in this dialog box may be disabled because they do not apply to that command.

3. Click **OK** to dismiss the **Stop Properties** dialog box.
4. Select a Stop set from the **Select stop set** drop-down list. This is the Stop set to be used in this analysis. Leave the **Select stop set** drop-down list blank if you want all Stops in the feature class to be used in the analysis.
5. In the **Select Stop Set** portion of the control, the **Properties** button will be enabled if a **Stop set** was selected in the previous step; click it to review the selected Stop set properties.

The ***Stop Set Properties*** *dialog box is displayed.*

Note: To view more Stops in the grid, resize the height of the **Stop Set Properties** dialog box by dragging its border.



Note: The **Enabled, Name,** and **Cost** columns are shown on the **Stop Set Properties** dialog only if corresponding Stop feature class properties were defined in Step 2. For example, if the **Enabled** field is not specified, the **Enabled** column and the **Enable All** and **Disable All** buttons will not be shown. Similarly, if the **Order** field is not specified, the buttons used to move the Stops up or down will not be available. If the Stop feature class is non-editable, then all the columns are displayed with gray background.

6. To edit an individual Stop, select it by clicking on the gray button to the left side of the row depicting that Stop. That row will be highlighted, and the buttons to the right of the grid will become enabled. Also, that feature will be highlighted in the **Map View**.

7. Once selected, there are a variety of edits that can be done to the Stop. The **Name** and **Cost** values can be edited by clicking and typing in a new value; the Stop can be moved up or down in the sequence by using the , , , and buttons; and the Stop can be enabled or disabled by clicking on the **Enabled** check box ☑ or by using the **Enable All** or **Disable All** buttons.

   The Cost value is used by the various routing analysis command if the **Minimize Cost** option is selected. The sequence of the Stops within the grid is used by the **Best Path** command when the **Best Order** option is not used or if either the **Start Fixed** or the **End Fixed** option is used in that command. Stops that do not have their **Enabled** box checked will be ignored by the various routing analysis commands.

8. If you want to exit the **Stop Set Properties** dialog box without saving any of the changes you have made, click the **Close** button. Otherwise, first click **Apply**.

   The **Apply** button makes use of the **Snap Tolerance** value to find the closest edge to each of the Stops in the list. This may take some time to calculate based on the size of the dataset. When completed, the **Status**, **EdgeID**, and **Edge Position** fields will be filled out in the grid.

   The **Status** field will indicate the relative success of finding an Edge within the snap tolerance of each Stop. A blank field indicates success. The message **"Stop located, multiple edges may be within tolerance"** indicates success, but that more than one Edge was found within tolerance. The closest Edge is used in this case. The message **"Not located, no edges within tolerance"** indicates no Edge was located. Stops with this status will be ignored by any of the routing analysis commands.

   The **EdgeID** and **EdgePosition** fields indicate which Edge a Stop was snapped to and the relative position (0 to 1) of that Stop along that Edge.

Note: The **EdgeID** and **EdgePosition** values calculated here are static values. That means that if your network changes, you may need to update these values to reflect those changes. You can update these values by revisiting the **Stop Set Properties** dialog box and by using the **Apply** button. It is possible to have the **EdgeID** and **EdgePosition** values calculated dynamically using the **Output Stop Set as Query** option within the Stop Manager command (see the "Working with the Stop Manager Command" chapter.

---

9. Click **Close** to dismiss the **Stop Set Properties** dialog box. If you have not first selected the **Apply** button, an informational message will appear, letting you know that if you exit the dialog box now, none of your changes will be saved.

   At the bottom of the control there are read-only fields that display the tolerance value and units set in the **Stop Set Properties** *dialog box.*

Note: You can edit these values only in the following two cases: a Stop set was not selected or EdgeID and EdgePos fields were not specified for the selected feature class

---

36 – The Routing Data Maintenance Workflows

The purpose of this chapter is to outline the basic steps needed to build and maintain a routing network. Two approaches are presented. The first is a dynamic routing network approach based on queries that are automatically updated anytime the underlying linear geometry is changed. The second is a table-based approach that assumes the network components are stored as tables. The dynamic approach makes data maintenance very simple but has some limits on how much you can fine tune the network characteristics and is more memory intensive. The table-based approach allows you complete flexibility in fine tuning network characteristics but places a heavier burden on you to maintain the network when there are changes to network geometry. Lastly, some steps for validating your routing network are presented so as to ensure accurate routing analysis results.

The Dynamic Routing Data Maintenance Workflow

This workflow uses the **Build Network** command to create dynamic queries that represent the three basic network components: Edges, Nodes, and Turns. Because these queries are dynamic, the network topology is automatically updated anytime the underlying linear feature class is changed. This is very helpful for dynamic data sets with geometry that changes fairly often. However, this method does not give you the flexibility to set individualized settings for turns or nodes after the initial network build (as the Node and Turn queries are not editable). So this methodology is excellent for a network that may have frequent geometry changes but whose turn characteristics are predictable.

1. Locate the source linear feature class for your network. This is the linear feature class that geometrically defines your network.
2. Add columns as desired to your source linear feature class for OneWays (long integer), Blockages (long integer), or Restrictor source fields using the **Feature Class Definition** command.
3. Locate, if available, the source Node attribute feature class for your network.
4. Use the **Build Network** command to create Edge, Node, and Turn queries. Set the options to create default turn costs and/or blockages. If you have a source Node attribute feature class, select the option to

copy over any fields that you would like in your output Node query. For more information on this see the "Working with the Build Network Command" chapter.

5. Set OneWays and Blockage values as desired on individual Edges. This can be done using either the Edge query or the source linear feature class. For more information on this see the "Working with the OneWay Editor Command" and "Working with the Blockage Editor Command" chapters.
6. Develop Restrictors for Edges, Nodes, and Turns as desired. For more information on this see the "Working with the Restrictor Editor Command" chapter.
7. Create Stops as needed for any routing analyses. Use the results of the **Output Stop Set as Query** option within the **Stop Manager** command when selecting Stops within any routing analysis command. It is important to use this query as opposed to the Stop table itself because the **EdgeID** and **EdgePosition** values for the stops are calculated dynamically within this query; whereas the **EdgeID** and **EdgePosition** values in the Stop table are static. For more information see the "Working with the Stop Manager Command" chapter.

The best part of this workflow is that now you are done. If you make changes to the geometry in your input linear feature class, your output network components will be automatically updated to reflect those changes with no interference needed on your part.

Manual Routing Data Maintenance – The Initial Network Creation Workflow

This is the first part of a two-part workflow for creating and maintaining a table-based routing network. This section is the one-time task of initial network creation. The next section covers the tasks needed to update the network to reflect changes in the network geometry that occur over time. Together these two workflows create an alternative to the Dynamic Routing Data Maintenance Workflow described previously. These two workflows require a little more work but offer more flexibility in defining your network.

1. Locate the source linear feature class for your network. This is the linear feature class that geometrically defines your network.
2. Locate, if available, the source Node attribute feature class for your network.

3. Use the **Build Network** command to create Edge, Node, and Turn queries. Set the options to create default turn costs and/or blockages. If you have a source Node attribute feature class, select the option to copy over any fields that you would like in your output Node query. For more information on this see the "Working with the Build Network Command" chapter.
4. Use the **Output to Feature Class** command to save your Edge, Node, and Turn query classes to tables in a read/write connection.
5. Delete the Edge, Node, and Turn queries that were created using the **Build Network** command.
6. Add columns as desired to your Edge class for Length (double), OneWays (long integer), Blockages (long integer), or Restrictor source fields using the **Feature Class Definition** command.
7. Set OneWays and Blockage values as desired on individual Edges. For more information on this see the "Working with the OneWay Editor Command" and "Working with the Blockage Editor Command" chapter.
8. Use the **Update Attributes** command to populate the Length field of the Edge feature class.
9. Add columns as desired to the Node and Turn feature classes for source Restrictor fields, and so on, using the **Feature Class Definition** command.
10. Set individual Node costs/blockages to suit using GeoMedia commands such as **Select Set Properties** or **Data Window**.
11. Use the **Turn Editor** to set individual Turn characteristics to suit. For more on this see the "Working with the Turn Editor Command" chapter.
12. Develop Restrictors for Edges, Nodes, and Turns as desired. For more information on this see the "Working with the Restrictor Editor Command" chapter.
13. Create Stops as needed for your routing analyses. For more information on this see the "Working with the Stop Manager Command" chapter.

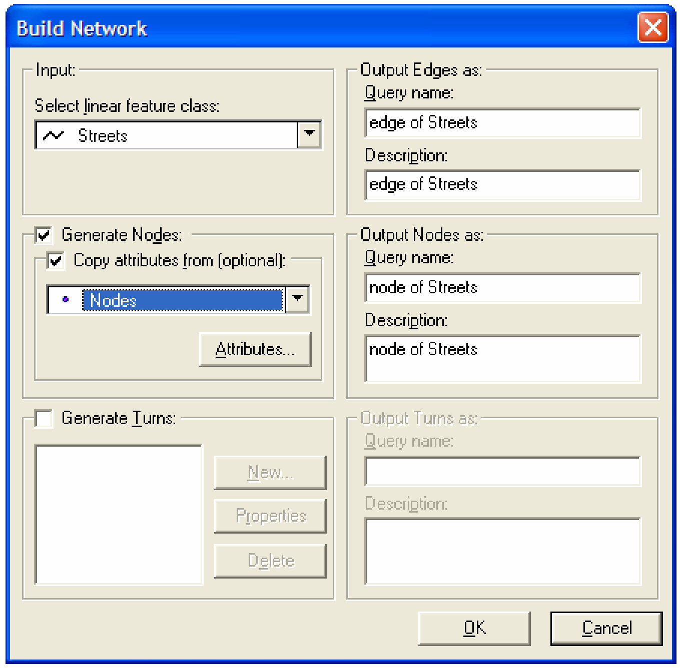
Manual Routing Data Maintenance – The Network Update Workflow

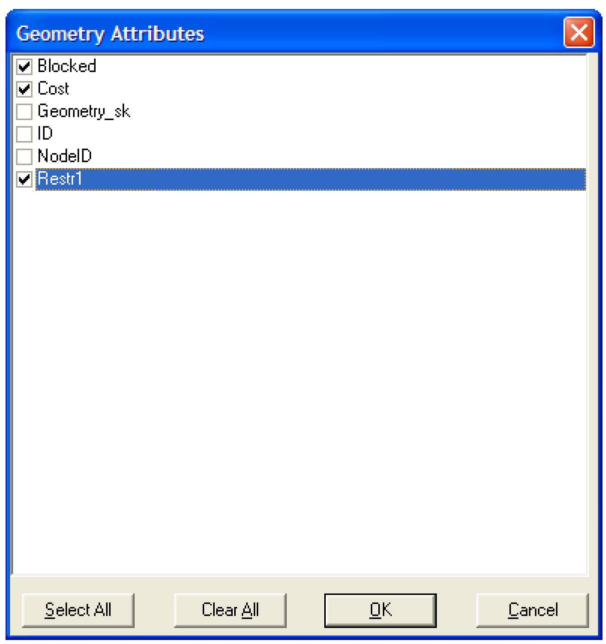
This is the second part of the table-based routing network workflow. The first part created the initial network and made it ready for analysis. This part is what needs to be done anytime there are changes to the network geometry. This workflow updates the fields that define the network topology to ensure correct routing analysis results. There are several steps involved, but the basic approach can be summarized as follows:

- Use the **Build Network** command to calculate new EdgeID, FromNodeID, and ToNodeID values for the Edges and new NodeID values for the Nodes.
- Replace the old Node feature class with the newly calculated Node query.
- Use the Edge query created by the **Build Network** command, which contains both new and old EdgeID values, to update the old Turn table. Use the queries FromNodeID and ToNodeID values to ensure the correct directionality of the Edges in the Turn table.
- Update the EdgeID, FromNodeID, and ToNodeID values in your Edge feature class.
- Update any Stops you have using the **Stop Manager** command.

Because of the possibility of confusion in interpreting these steps some key words have been italicized.

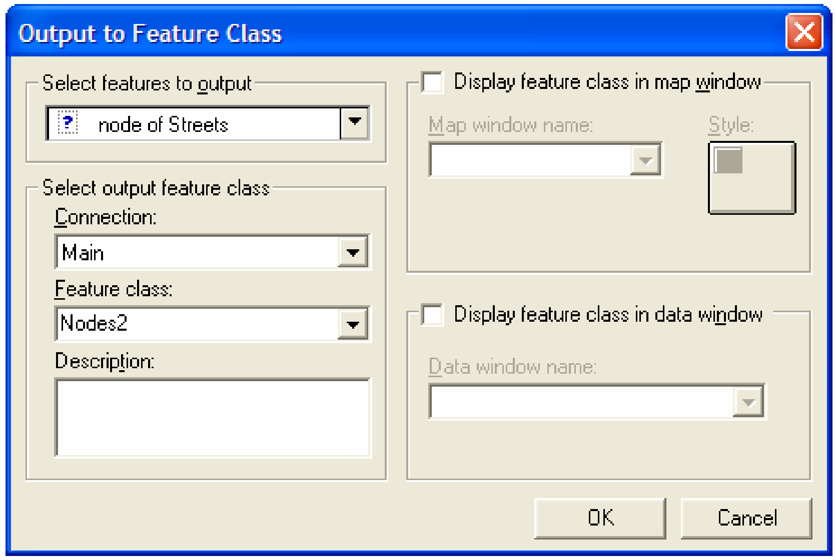
1. Use the **Build Network** command, with your existing Edge and Node *feature* classes as input, to create new Edge and Node *query* classes. This will create an Edge *query* class with both new and old network topology fields and will copy from your existing Node *feature* class any data you might want to preserve such as costs, blockages, and restrictor source fields. Although your field names may vary, for purposes of this workflow we will say that the *old* Edge topology fields are named EdgeID, FromNodeID, and ToNodeID and that the *new* Edge topology fields are named EdgeID1, FromNodeID1, and ToNodeID1. For more information on this see the “Working with the Build Network Command” chapter.



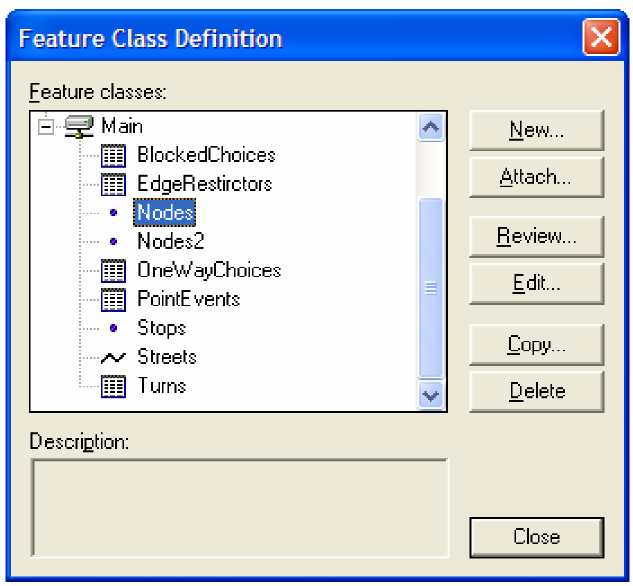


First, let's replace the Node table. . .

2. Use the **Output to Feature Class** command to save your Node *query* class to a *new* Node *feature* class. This is a replacement for the *old* Node *feature* class.

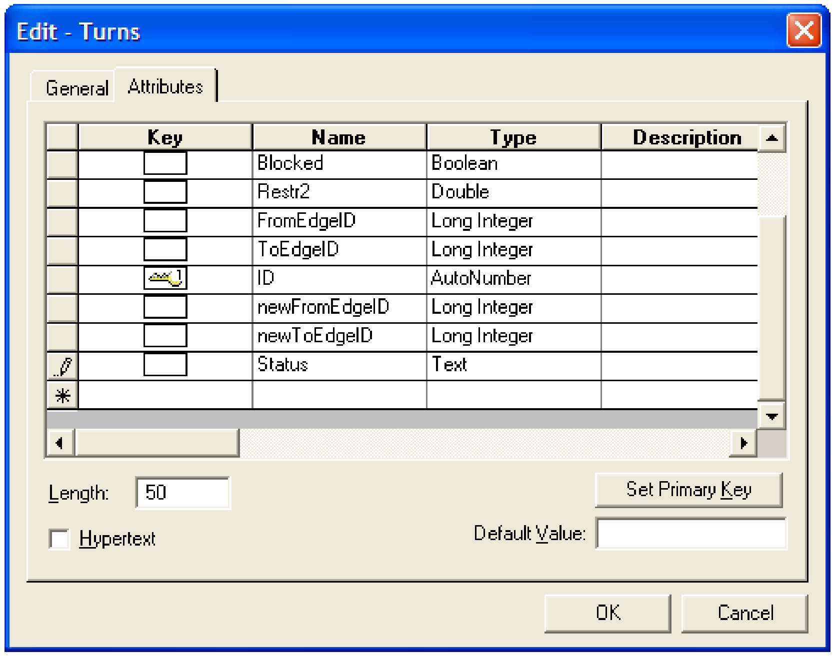


3. Delete the *old* Node *feature* class using the **Feature Class Definition** command.

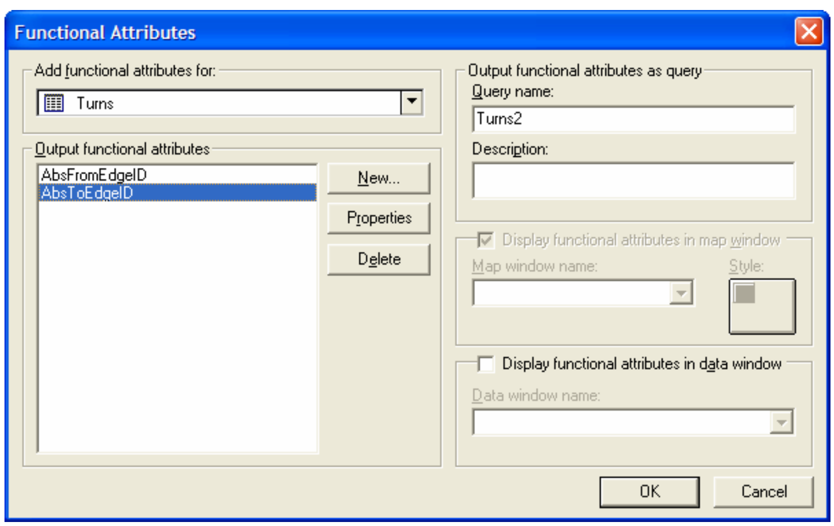


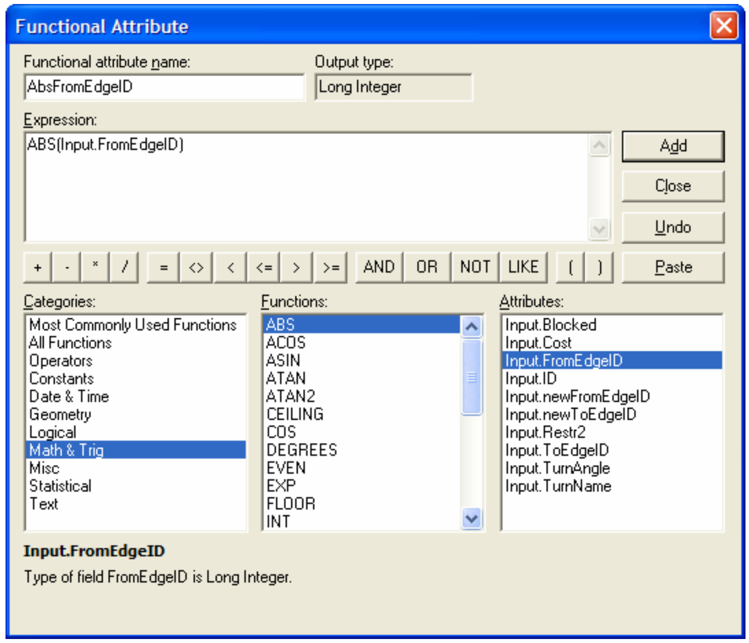
Now, let's update the Turn table…

4. Use the **Feature Class Definition** command to add three new fields to the existing Turn *feature* class: newFromEdgeID (long integer), newToEdgeID (long integer), and Status (text).



5. Use the **ABS** function of the **Functional Attributes** command on the existing Turn *feature* class to create two new calculated fields: AbsFromEdgeID and AbsToEdgeID. These will be equal to the absolute values of the FromEdgeID and ToEdgeID fields, respectively. Call the output query from this command **Turns2**.

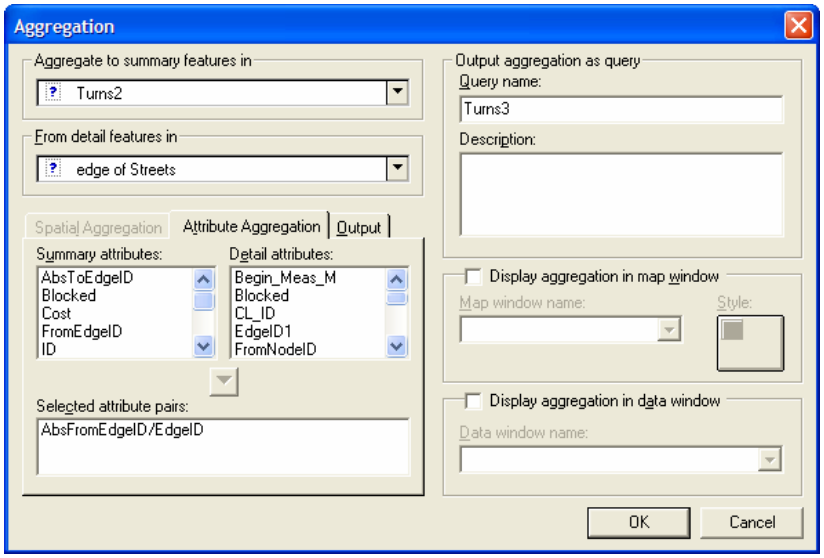


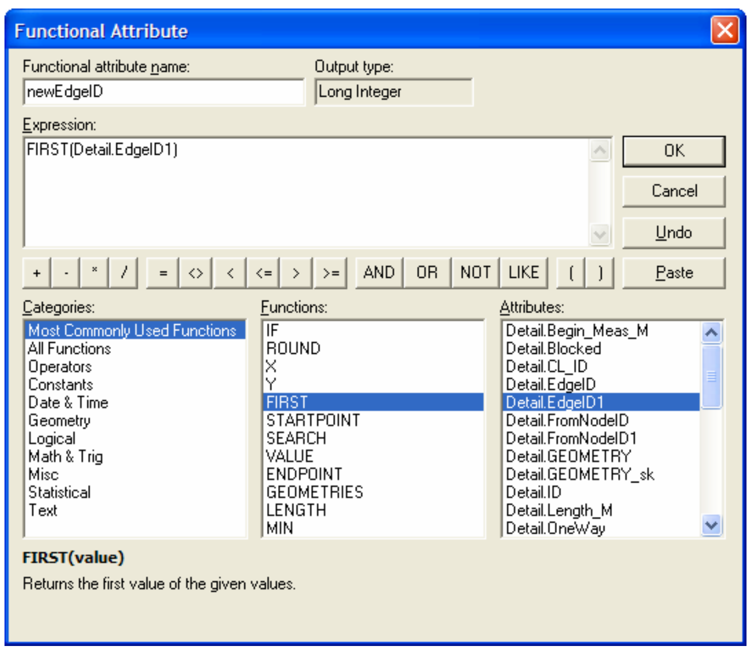


6. Use the **Aggregation** command using the **Turns2** query class from the last step as the *summary* class and the Edge *query* class as the *detail* class. Pick the **Attribute Aggregation** option, and select *AbsFromEdgeID* from **Turns2** and the *original* EdgeID field from the Edge *query* class for the join fields. Set the Resolution operator to **All**. Create the following three functional attributes:

   - **FE_EdgeID** = First(Detail.EdgeID1); this is the new Edge ID value for the From Edge.
   - **FE_FromNodeID** = First(Detail.FromNodeID1); this is the new From Node ID value for the From Edge.
   - **FE_ToNodeID** = First(Detail.ToNodeID1); this is the new To Node ID value for the From Edge.

   Call the output query from this command **Turns3**.





7. Use the **Aggregation** command using the **Turns3** query class from the last step as the *summary* class and the Edge *query* class as the *detail* class. Pick the **Attribute Aggregation** option, and select *AbsToEdgeID* from **Turns3** and the *original* EdgeID field from the Edge *query* class for the join fields. Set the Resolution operator to **All**. Create the following three functional attributes:

   - **TE_EdgeID** = First(Detail.EdgeID1); this is the new Edge ID value for the To Edge.

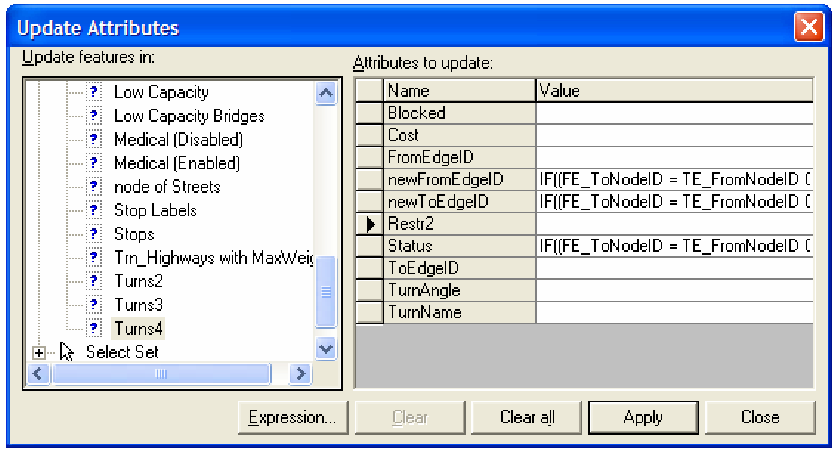
- **TE_FromNodeID** = First(Detail.FromNodeID1); this is the new From Node ID value for the To Edge.
- **TE_ToNodeID** = First(Detail.ToNodeID1); this is the new To Node ID value for the To Edge.

Call the output query from this command **Turns4**.

8. Use the **Update Attributes** command on **Turns4** to set the following attribute values. This process may take a while on a large network.

   - **newFromEdgeID** = IF((FE_ToNodeID = TE_FromNodeID OR FE_ToNodeID = TE_ToNodeID) AND (FE_FromNodeID = TE_FromNodeID OR FE_FromNodeID = TE_ToNodeID), (FromEdgeID/AbsFromEdgeID) * FE_EdgeID, IF(FE_ToNodeID = TE_FromNodeID OR FE_ToNodeID = TE_ToNodeID, FE_EdgeID, -1 * FE_EdgeID))
   - **newToEdgeID** = IF((FE_ToNodeID = TE_FromNodeID OR FE_ToNodeID = TE_ToNodeID) AND (FE_FromNodeID = TE_FromNodeID OR FE_FromNodeID = TE_ToNodeID), (ToEdgeID/AbsToEdgeID) * TE_EdgeID, IF(TE_FromNodeID = FE_FromNodeID OR TE_FromNodeID = FE_ToNodeID, TE_EdgeID, -1 * TE_EdgeID))
   - **Status** = IF((FE_ToNodeID = TE_FromNodeID OR FE_ToNodeID = TE_ToNodeID) AND (FE_FromNodeID = TE_FromNodeID OR FE_FromNodeID = TE_ToNodeID), “Ambiguous direction”, “”)

   This process derives the directionality of the Edges from the FromNodeID and ToNodeID attribution where possible. In ambiguous cases (such as Edges that begin and end at the same Node), it merely copies the directionality from the existing Turn table and marks these with a message in the Status field. This can be reviewed and edited as needed using the **Turn Editor** command. For more information on this see the “Working with the Turn Editor Command” chapter.

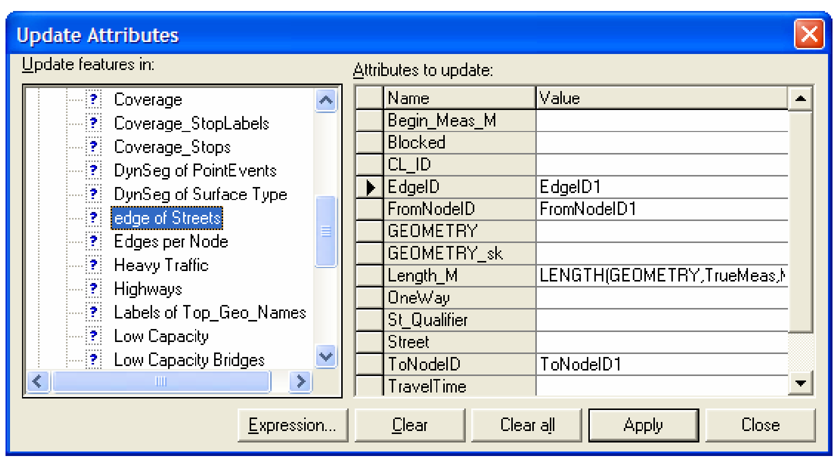


9. Delete queries **Turns2**, **Turns3**, and **Turns4**.

10. Use the **Feature Class Definition** command on the Turn *feature* class to delete original FromEdgeID and ToEdgeID fields and to rename the newFromEdgeID field to FromEdgeID and rename the newToEdgeID field to ToEdgeID.

Now let's update the Edge table…

11. Use the **Update Attributes** command on the new Edge *query* class to set the old EdgeID, FromNodeID, and ToNodeID values equal to the new EdgeID1, FromNodeID1, and toNodeID1 values that were created by the **Build Network** command. Also you can update the Length field at the same time. This process may take a while on a large network.



12. Delete the Edge and Node query classes created by the **Build Network** command.

Finally, let's update the Stop table…

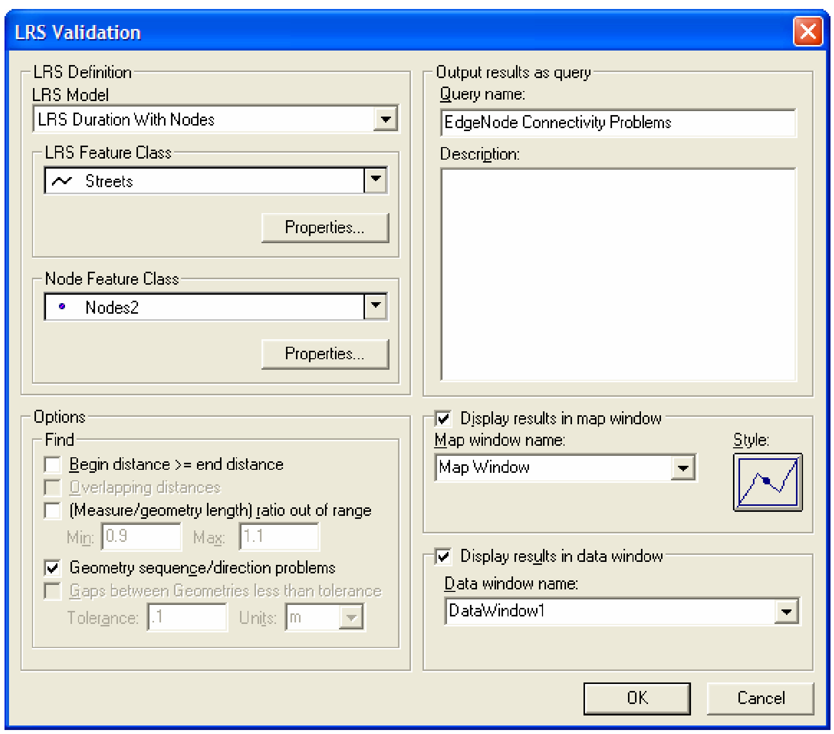
13. Use the **Stop Manager** command to recalculate Stop locations. For more information on this see the "Working with the Stop Manager Command" chapter.

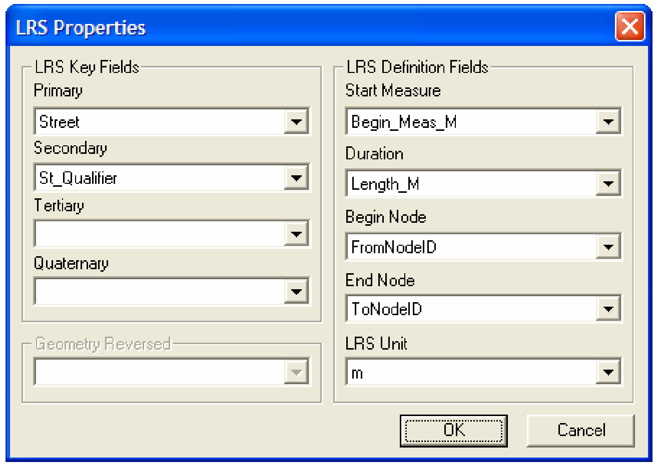
It is worth noting that automating the workflow just presented is currently planned for a future version of GeoMedia Transportation Manager.

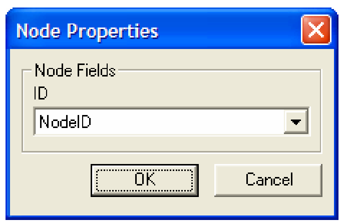
The Routing Network Validation Workflow

This last workflow in this chapter addresses checks that can be made to find common problems with routing networks. We will do three separate checks: Edge-Node connectivity, unwanted subnetworks, and unwanted disconnected Edges.

1. Check the Edge-Node connectivity using the **LRS Validation** command. Choose one of the two "**With Nodes**" options for the LRS Model, and only choose the **Geometry sequence/direction problems** check. For more information on this see the "Working with the LRS Validation command" chapter.



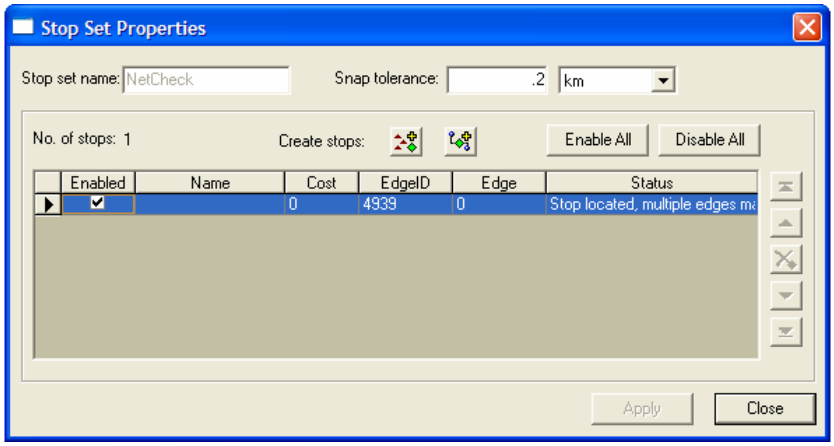




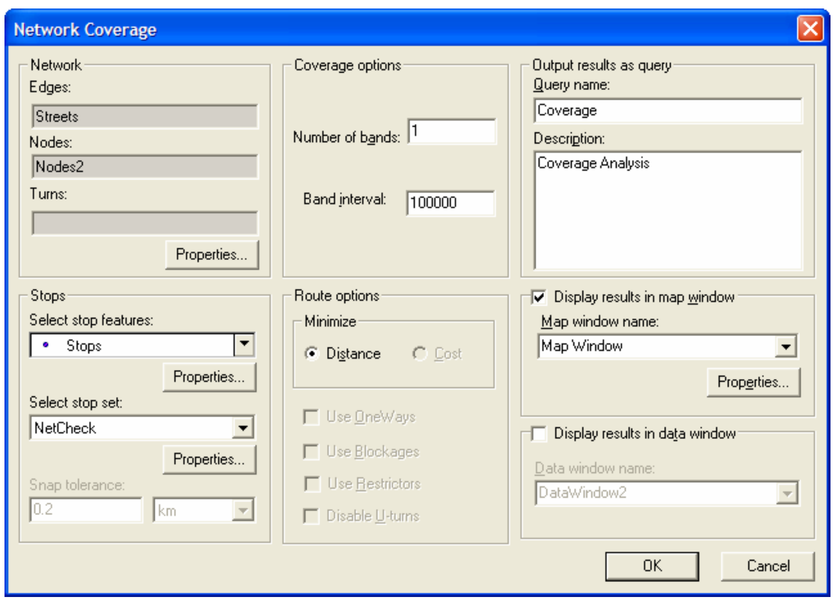
2. Review the results of the **LRS Validation** command. Note that some perfectly valid, but questionable, situations (for example, cul-de-sacs that tie back in on themselves) are located by this command so that you can use your own good judgment to decide if you want to correct them or not.

Next, let's use the **Network Coverage** command to see if there are any unwanted subnetworks. Subnetworks are portions of the network that are, perhaps unintentionally, disconnected from the rest of the network. If this happens, the results of your routing analyses may not be accurate.

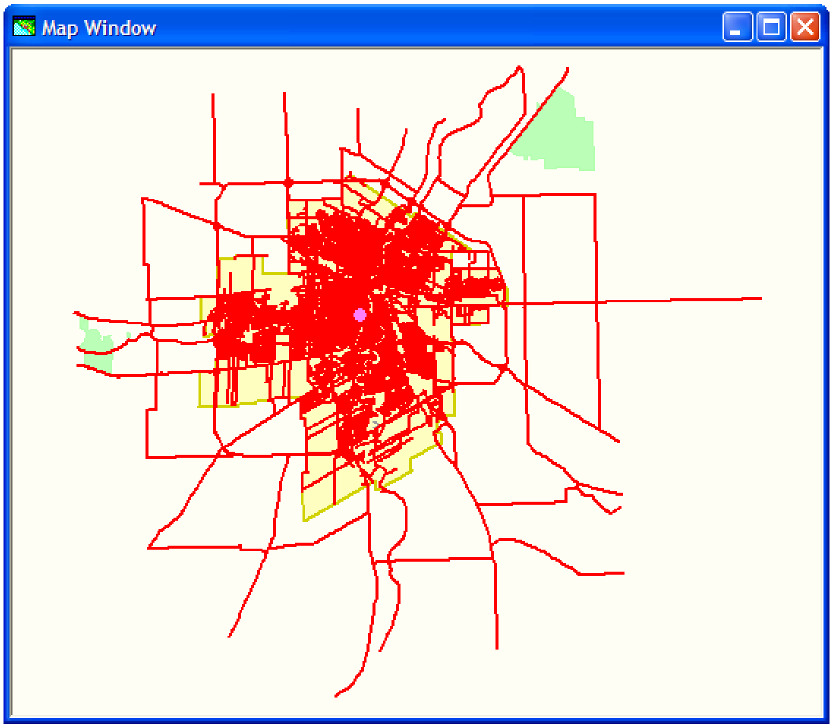
3. Use the **Stop Manager** command to create a one-Stop *Stop Set*. The one Stop in this Set can be anywhere in the network, but it probably makes good sense to pick a stop somewhere towards the middle of the network. For more on this see the "Working with the Stop Manager Command" chapter.



4. Use the **Network Coverage** command using your routing network and the new Stop Set you just created. Use one band with a distance that you know to be sufficiently long to easily stretch from your Stop to any point on the network. Do not bother with Turns, OneWays, Blockages, or Restrictors. For more on this see the "Working with the Network Coverage Command" chapter.

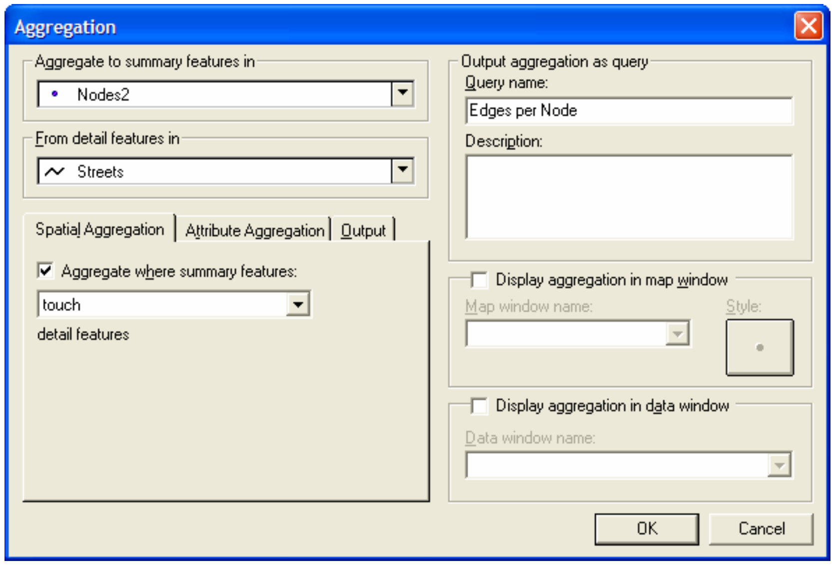


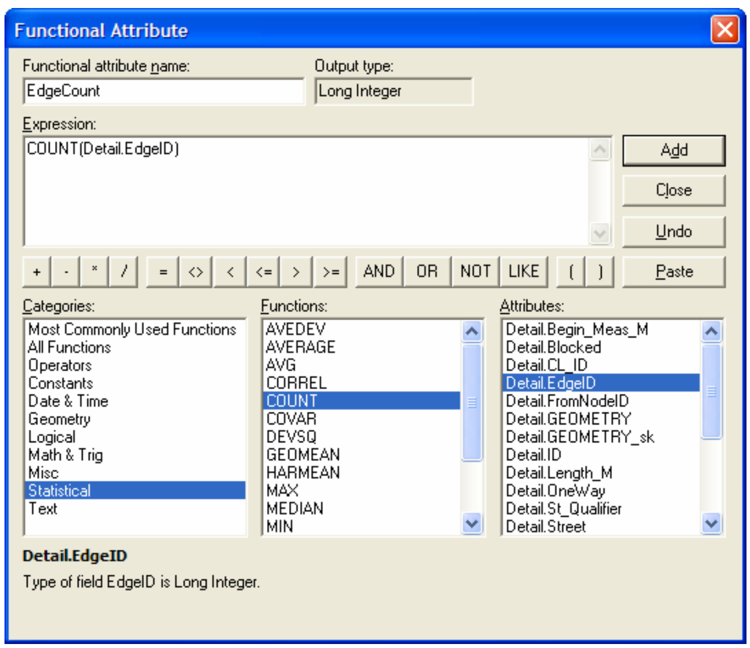
5. Review the results visually to see if the results of the **Network Coverage** command completely cover the entire routing network. For a more rigorous check, you can use the **Spatial Difference** command in GeoMedia to compare the results of the **Network Coverage** command with your routing network.



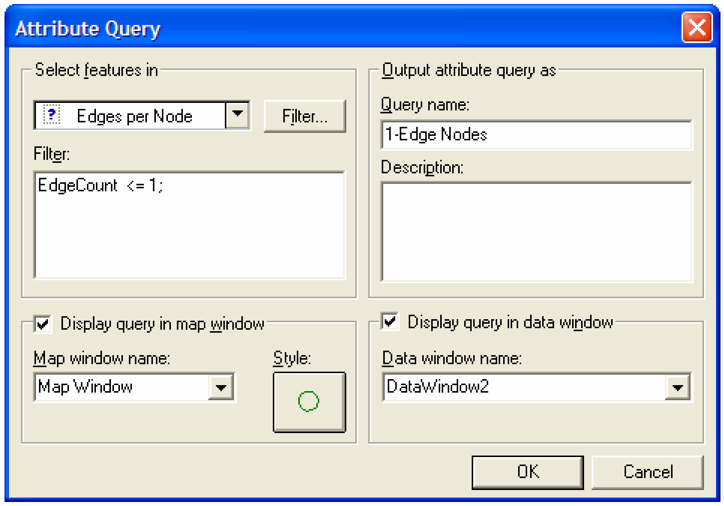
Lastly, let's use the **Aggregation** and **Attribute Query** commands to see if there are any unwanted disconnected Edges. This can happen when there is an intersection to which, for example, four Edges are supposed to be connected, but one of them falls short of the intersection by a small amount. These situations are hard to detect visually, but they can cause errors in routing analysis results. The steps below locate all Nodes to which only one Edge is connected (sometimes called *Terminal Nodes*). These can be perfectly valid and are common around the perimeter of a routing network. However, by reviewing these one-Edge Nodes, you can catch the most common cases of unwanted disconnected Edges.

6. Use the **Aggregation** command to count how many Edges touch each Node.

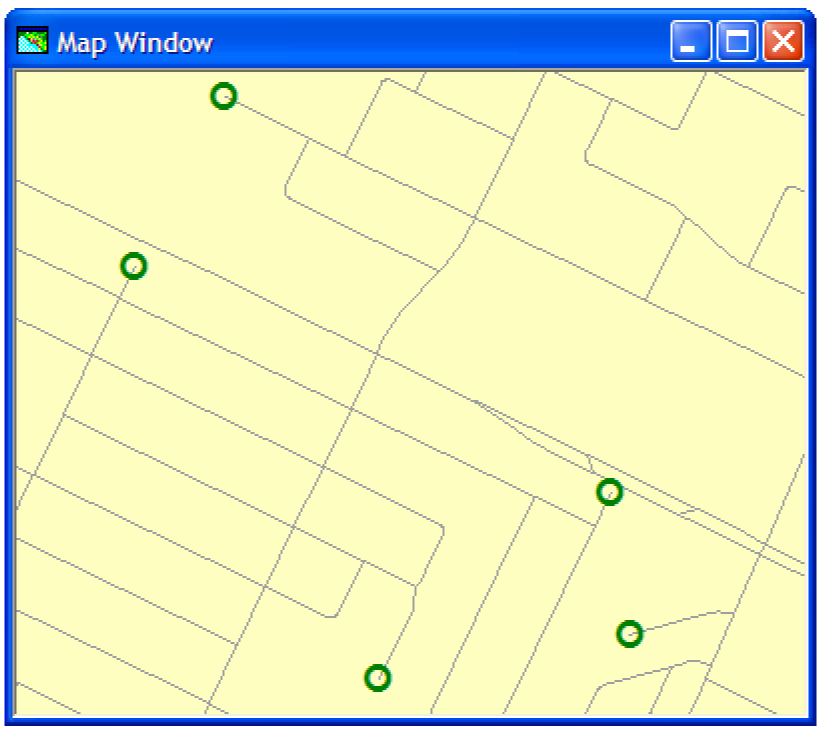




7. Use the **Attribute Query** command to find all Nodes with one or fewer Edges touching them.



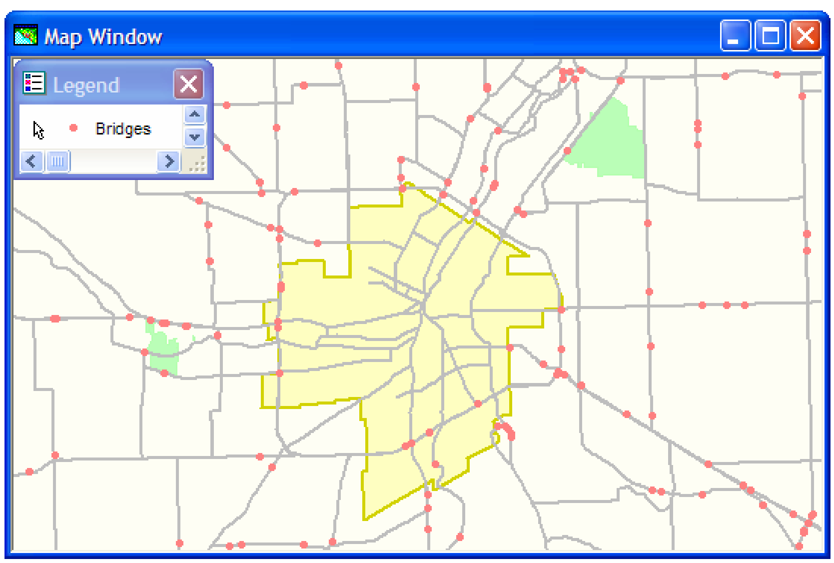
8. Visually inspect these one-Edge Nodes to make sure no Edges are disconnected unintentionally.



37 – The LRS-based Routing Restrictions Workflow

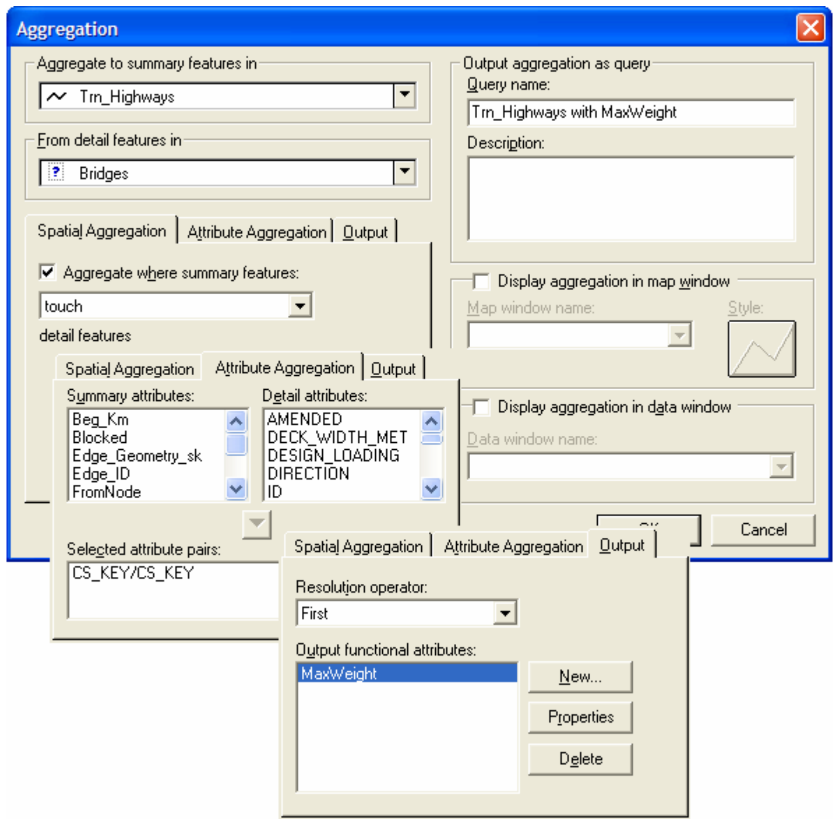
The purpose of this chapter is to outline the basic steps needed to develop routing restrictions from LRS-based data. The example we are going to use to illustrate this makes use of attribution on LRS bridge points to restrict routing analyses. In this case we will avoid any edge that contains a bridge with less than the desired maximum weight rating, as one might want to do when routing a heavily loaded vehicle. This example can be extended to any number of cases where the data that you want to base your routing restriction on exists on LRS-based Event data and not on the components of your routing network.

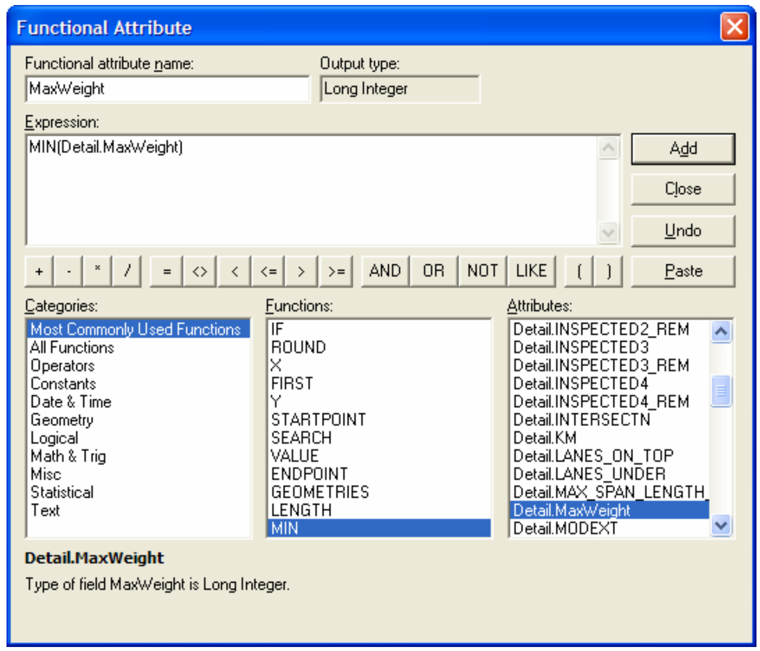
1. Use the **Dynamic Segmentation** command to place the Event data that contains the fields you want to base your routing restriction on. In this example, we have dynamically segmented bridges onto our network. For more information on this see the "Linear Referencing" chapter in *Working with GeoMedia* or *Working with Geomedia Professional*.



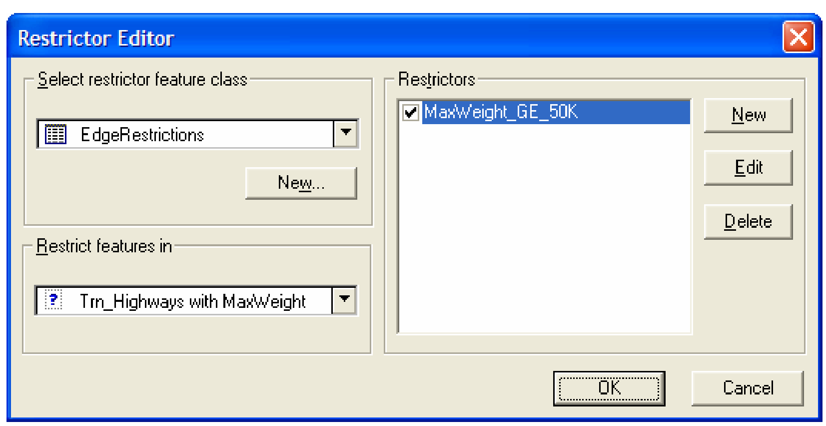
2. Use the **Aggregation** command to transfer Event data attribution to our routing Edges. In this case we are going to transfer the lowest Maximum Weight rating of any bridge that lies along an Edge to that Edge. For the **Spatial Aggregation** operator we have picked **touch**,

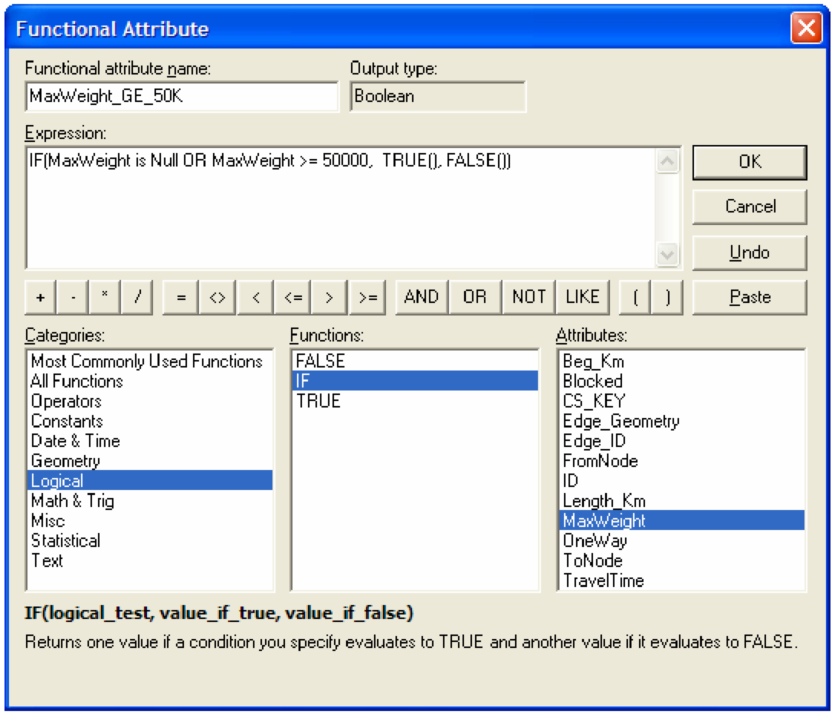
for the **Attribute Aggregation** operator we have picked our LRS Key, and for our **Resolution** operator we have picked **First**.



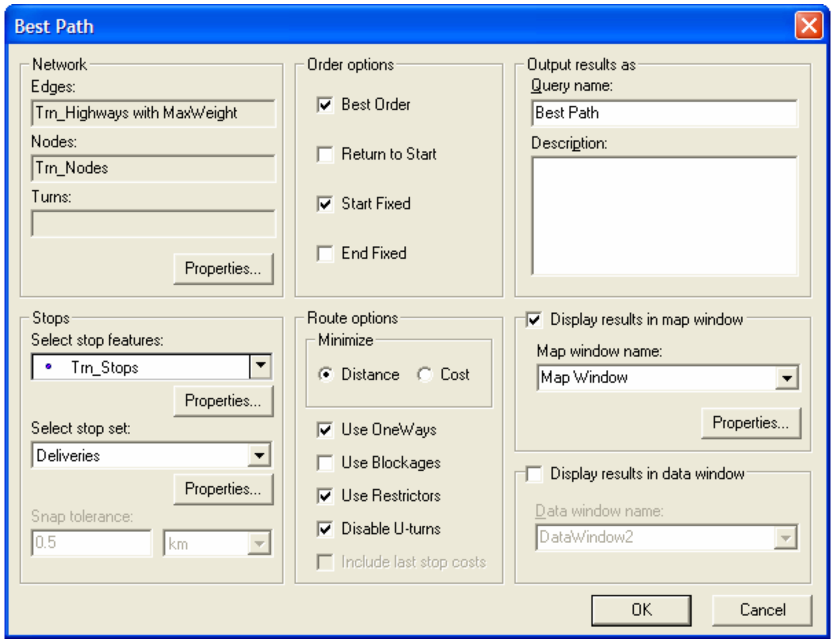


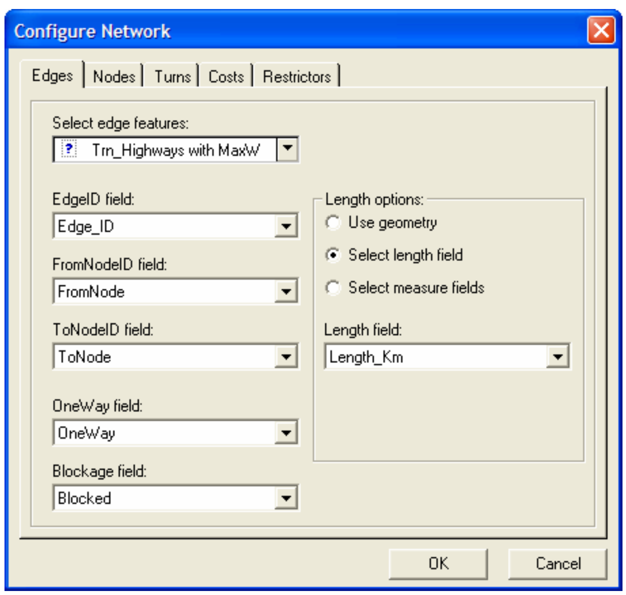
3. Use the **Restrictor Editor** to develop an Edge restriction that evaluates to True for Edges that you want to be included in your analysis and False for Edges you want to exclude. In this case we have included Edges with either no bridges or with bridges that have a 50,000 pound or greater Maximum Weight rating. Make sure to use the results of the **Aggregation** step as the query class to restrict. For more information on this see the "Working with the Restrictor Editor Command" chapter.

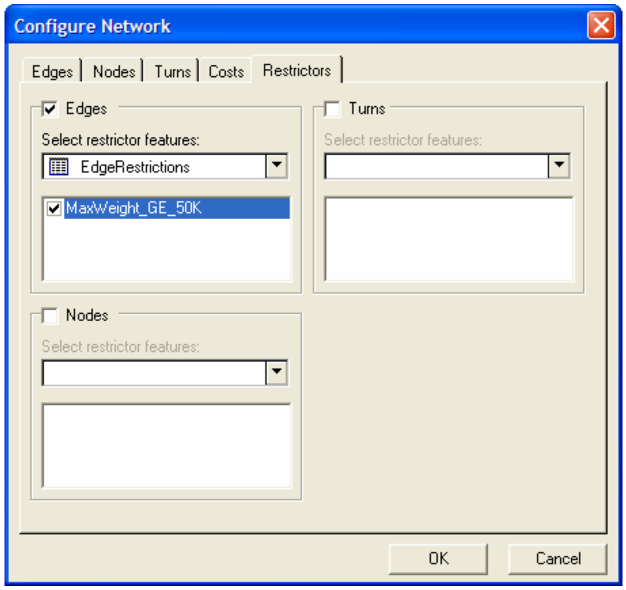




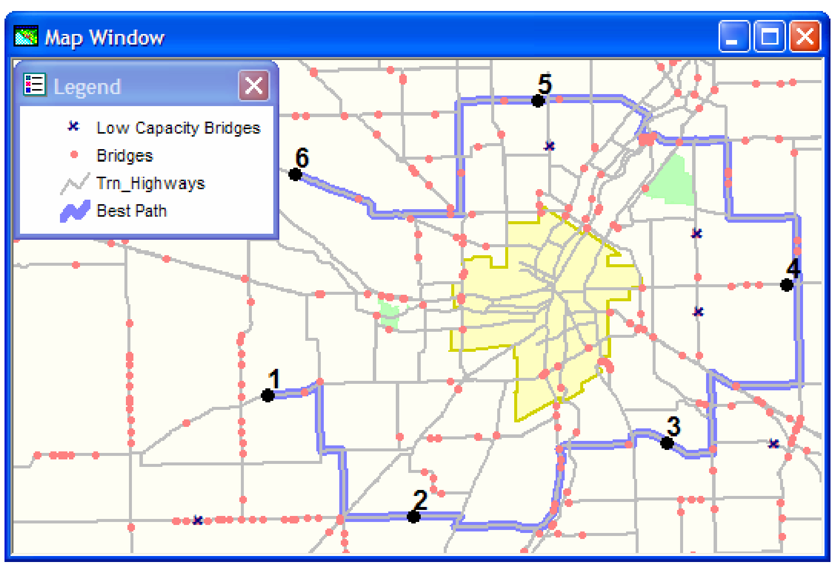
4. At this point you are ready to begin your routing analysis. For our example we will run an analysis using **Best Path**. It is important to pick the results of the **Aggregation** step for the Edge class, to choose the Restrictor created previously, and to pick the **Use Restrictors** option. For more information on this see the "Working with the Best Path Command" chapter.


Best Path
Network
Edges:
Trn_Highways with MaxWeight
Nodes:
Trn_Nodes
Turns:
Properties...
Order options
Best Order
Return to Start
Start Fixed
End Fixed
Output results as
Query name:
Best Path
Description:
Stops
Select stop features:
Trn_Stops
Properties...
Select stop set:
Deliveries
Properties...
Snap tolerance:
0.5
km
Route options
Minimize
Distance
Cost
Use OneWays
Use Blockages
Use Restrictors
Disable U-turns
Include last stop costs
Display results in map window
Map window name:
Map Window
Properties...
Display results in data window
Data window name:
DataWindow2
OK
Cancel


Configure Network
Edges
Nodes
Turns
Costs
Restrictors
Select edge features:
Trn_Highways with MaxW
EdgeID field:
Edge_ID
FromNodeID field:
FromNode
ToNodeID field:
ToNode
OneWay field:
OneWay
Blockage field:
Blocked
Length options:
Use geometry
Select length field
Select measure fields
Length field:
Length_Km
OK
Cancel



5. The following picture shows the results of our analysis. For clarity, bridges with less than 50,000 pound capacity are shown with an ×. Note that the **Best Path** result avoids Edges with these low capacity bridges. One of the more interesting things about this workflow is that if the MaxWeight attribution of one of these bridges is updated, the **Best Path** solution will automatically be updated to reflect that change.



A. How to Reach Intergraph

Electronic Self-Help Support

Intergraph provides several electronic self-help support tools to answer your support questions 24 hours a day, seven days a week. Using any web browser, you can access Security, Government & Infrastructure User Support on the World Wide Web at http://support.intergraph.com.

The *Knowledge Base* and *Technical Notes* are available at http://support.intergraph.com/kb.asp.

For additional product information, please visit our product web site at http://www.intergraph.com/sgi/products/.

B - GeoMedia Transportation LRS Data Structures

This appendix describes the Linear Referencing System (LRS) data structures that are directly supported by the GeoMedia Transportation Suite of products as delivered (GeoMedia Transportation Manager, GeoMedia Transportation Analyst, GeoMedia WebMap Professional, and IntelliWhere Location Server).

Overview

Data for Transportation Asset systems (using dynamic segmentation) generally falls into two categories: LRS data and Event data. LRS data describes the naming, measurement system, and geometry of the linear network. Event data describes attributes of the linear network, such as pavement conditions, roadway inventory data (for example., guardrails and signage), and accident occurrences. GeoMedia Transportation Manager provides great flexibility in the structuring of both of these data types, as shown below:

LRS Data Structure Options

- Measure
- Measure with Internal Markers
- Measure with External Markers
- Duration
- Duration with Internal Markers
- Duration with External Markers

Event Data Structure Options

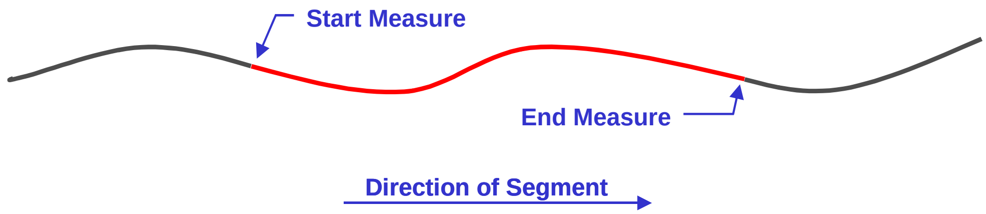
- Measure
- Marker Offset
- Coordinate
- Duration

LRS Data Structures

An LRS data structure is typically made up of a number of geometric segments that together define the entire LRS. All but the two External Marker options, described in the following subsections, consist only of a table describing those segments. The External Marker options have an additional table that describes marker points. Note that all of the field names used in the subsections to follow are representative only. You are free to name the fields as you please.

Option 1 – Measure

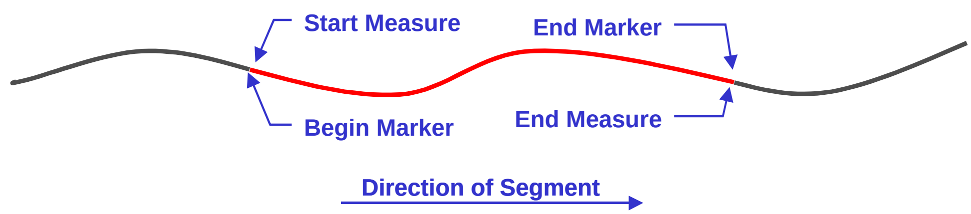
The segments in this option are defined by a set of up to four LRS keys to name the route, a start and an end measure value, and geometry.



LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
End Measure	18.2	This is the measurement value for the ending position of this feature.
Geometry	blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

Option 2 – Measure with Internal Markers

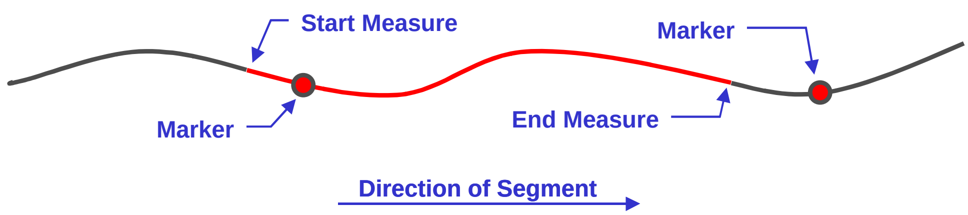
The segments in this option are defined by a set of up to four LRS keys to name the route, a start and an end measure value, a Begin Marker name, an optional End Marker Name, and geometry. The markers allow this structure to be used with event data using the Marker Offset option.



LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
End Measure	18.2	This is the measurement value for the ending position of this feature.
Begin Marker	M45	This field stores a name for the beginning position of this feature.
End Marker	M46	This field stores a name for the ending position of this feature. Its use is optional.
Geometry	blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

Option 3 – Measure with External Markers

The segments in this option are defined by a set of up to four LRS keys to name the route, a start and an end measure value, and geometry. This option also requires markers that are defined by up to four LRS keys, a marker name, and a measure value. The markers allow this structure to be used with event data using the Marker Offset option.

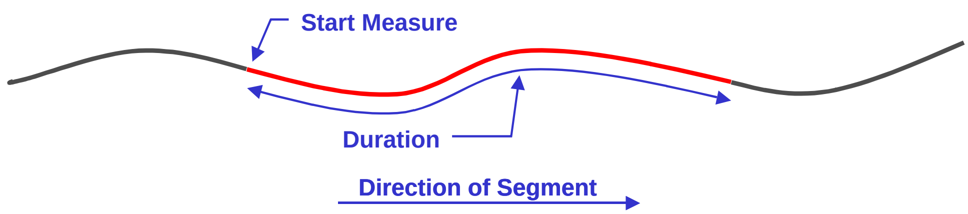


LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
End Measure	18.2	This is the measurement value for the ending position of this feature.
Geometry	blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

Marker Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Name	M34	This is the identifying name of the marker.
Measure	17.5	This is the measurement value at the marker.

Option 4 – Duration

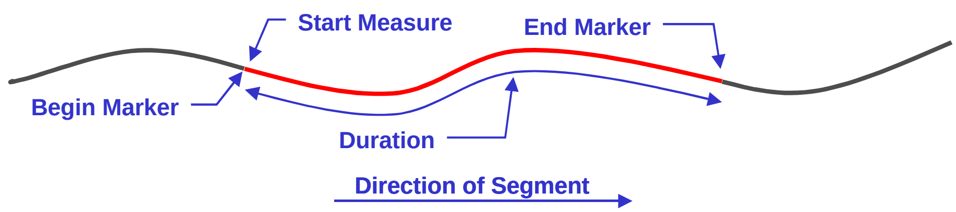
The segments in this option are defined by a set of up to four LRS keys to name the route, a start measure value, a duration (length) value, and geometry.



LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
Duration	5.9	This is distance value from the beginning position to the ending position of this feature.
Geometry	blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

Option 5 – Duration with Internal Markers

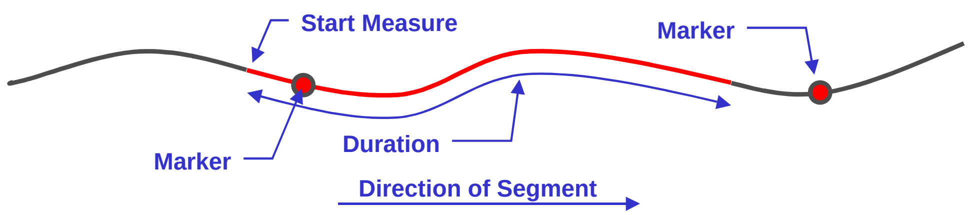
The segments in this option are defined by a set of up to four LRS keys to name the route, a start measure value, a duration (length) value, a Begin Marker name, an optional End Marker Name, and geometry. The markers allow this structure to be used with event data using the Marker Offset option.



LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
Duration	5.9	This is distance value from the beginning position to the ending position of this feature.
Begin Marker	M45	This field stores a name for the beginning position of this feature.
End Marker	M46	This field stores a name for the ending position of this feature. Its use is optional.
Geometry	blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

Option 6 – Duration with External Markers

The segments in this option are defined by a set of up to four LRS keys to name the route, a start measure value, a duration (length) value, and geometry. This option also requires markers that are defined by up to four LRS keys, a marker name, and a measure value. The markers allow this structure to be used with event data using the Marker Offset option.



LRS Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	12.3	This is the measurement value for the beginning position of this feature.
Duration	5.9	This is distance value from the beginning position to the ending position of this feature.
Geometry	Blob	This field contains the linear geometry that describes the linear segment geometrically.
Reversed_ Geometry	True	This Boolean (True/False) field declares whether the software should treat this linear feature as is (False) or as if its digitizing direction were reversed and its beginning were its end and vice-versa (True).

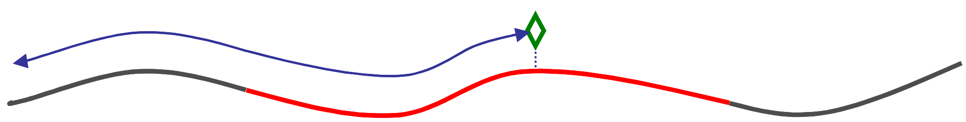
Marker Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Name	M34	This is the identifying name of the marker.
Measure	17.5	This is the measurement value at the marker.

Event Data Structures

Note that these structures have no geometry. The whole purpose of the dynamic segmentation process is to combine the LRS data structure and the Event data structure to create a new geographic feature. Note that all of the field names used in the subsections to follow are representative only. You are free to name the fields as you please.

Option 1 – Measure

The events in this option are defined by a set of up to four LRS keys to name the route, a start measurement, and for linear events only, an end measure value.

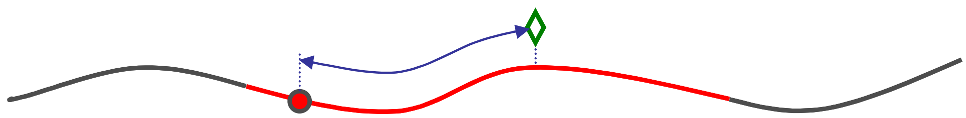


Event Feature Class

Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	14.4	This is the measurement value for the beginning position of this event. This field is required for both point and linear events.
End Measure	16.2	This is the measurement value for the ending position of this event. This field is used for linear events only.

Option 2 – Marker Offset

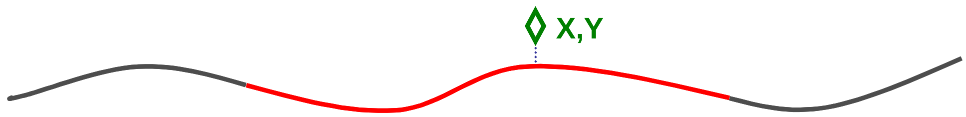
The events in this option are defined by a set of up to four LRS keys to name the route, a begin marker name and an offset value, and for linear events only, an end marker name and offset value.



Event Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Begin Marker	M34	This is the name of the marker from which the beginning point of the event is measured. This field is required for both point and linear events.
Begin Offset	0.6	This is distance value from the Begin Marker to the beginning position of this event. This field is required for both point and linear events.
End Marker	M35	This is the name of the marker from which the ending point of the event is measured. This field is used for linear events only.
End Offset	0.4	This is distance value from the End Marker to the ending position of this event. This field is used for linear events only.

Option 3 – Coordinates

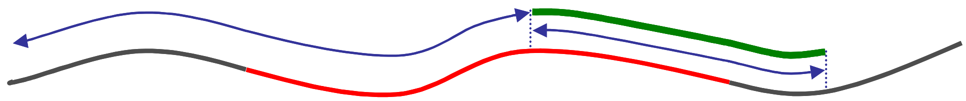
The events in this option are defined by a set of up to four LRS keys to name the route, a begin X/Y or Longitude/Latitude coordinate, and for linear events only, an end X/Y or Longitude/Latitude coordinate.



Event Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Begin X	2546234.2	This is the X or Longitude coordinate for the beginning position of this event. This field is required for both point and linear events.
Begin Y	753124.4	This is the Y or Latitude coordinate for the beginning position of this event. This field is required for both point and linear events.
End X	2584123	This is the X or Longitude coordinate for the ending position of this event. This field is used for linear events only.
End Y	745654.6	This is the Y or Latitude coordinate for the ending position of this event. This field is used for linear events only.

Option 4 – Duration

The events in this option are defined by a set of up to four LRS keys to name the route, a begin measure, and a duration (length) value. The Duration option applies to linear events only.



Event Feature Class		
Field	**Sample**	**Description**
LRSkey1	US	The LRS key fields identify the "route" that this segment of the LRS belongs to. This identification can be done with anywhere from 1 to 4 fields within the LRS feature class. These fields are the same fields the event features will use to identify the "route". In the sample shown LRSkey1 identifies the roadway system, LRSkey2 contains the route number, LRSkey3 identifies whether the segment is part of a spur, and LRSkey4 identifies whether this segment is part of an alternative route. All together the route name is "US6SA".
LRSkey2	6	This is the 2nd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey3	S	This is the 3rd key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
LRSkey4	A	This is the 4th key field that can be used to identify the "route" that this segment of the LRS belongs to. Its use is optional.
Start Measure	14.4	This is the measurement value for the beginning position of this event.
Duration	1.8	This is distance value from the beginning position to the ending position of this event. Note: the Duration option is for linear events only.

Index

E

F

G

H

I

K

L

Additional information on Intergraph Support and Services is available on the Internet. Use a Web browser to connect to Intergraph Online at *http://intergraph.com*.

For general Intergraph information, call 1-800-791-3357 (U.S. and Canada) or 1-256-730-2000 (international).

Working with GeoMedia Transportation Manager

10/05

DJA084580