GeoTrans

Transportation Data Model

INTERGRAPH

Table of Contents

GeoTrans Transportation Data Model

1. Preface

There is a wealth of research that has been done on the subject of data models for transportation. The purpose of this document is to distill some of this research into a format that can be directly implemented by our customers. We at Intergraph Mapping and GIS Solutions sincerely hope that this data model is a helpful tool in meeting your transportation information needs. We welcome any input you may have on this subject (contact phil.hardy@intergraph.com).

1.1 Audience

This White Paper is intended for users with an interest in developing data models for transportation information systems. It provides a suggested transportation data model for use with the GeoMedia Transportation product line. The document assumes that the reader has some understanding of the GeoMedia Transportation product line, transportation information systems, and data modeling. It may be useful to the readers of this document to first read Appendix B – Glossary to acquaint themselves with terminology used in this document.

1.2 Organization

This document contains the following information:

- A review of the design goals for this data model
- A discussion of the approach taken by the designers of this data model to address the stated design goals
- A description of the structure of the data model
- A discussion of some of the many implementation issues involved
- A listing of some of the areas to be addressed in future versions of this document
- Appendices that cover references, terminology used, etc.

2. Design Goals

This current version of the GeoTrans data model is limited to linear referencing systems. Thus the design goals pertain to some of the most important aspects of modeling linear referencing systems.

- Event Location Stability – This refers to the requirement for events to maintain their locations regardless of changes to the linear network.

- Temporal Persistence – This is closely related to event location stability in that changes to the linear network will happen over time. This requirement is for the ability to view and query the linear network and event data for any time period for which data exists.

- Multiple Linear Referencing Method (LRM) Support – Almost all transportation agencies have developed, over time, more than one way to locate events on their network. Ten or more LRMs in an agency is not uncommon. This requirement is for the ability to support any number of LRMs without having to duplicate network geometry.

- Implementable – This data model is designed to be immediately implementable with software that is available "off-the-shelf," namely Intergraph's GeoMedia Transportation suite of products.

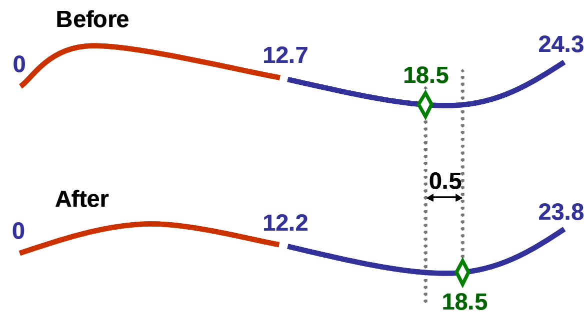
Our overriding goal is to provide a service to our customers. It is important to note that this is only a suggested data model and by no means the required data model. One of the strengths of the GeoMedia Transportation products is the flexibility they have to use an assortment of data structures with no required field names. This allows for any number of data models to meet a user's particular needs. However, it is hoped that this paper will be a useful tool to many users who are looking for some guidance in how to structure a data model that both addresses their needs and which is designed to work with the GeoMedia Transportation products.

3. Data Model Approach

This section outlines the approach and reasoning that went into creating the portion of the data model pertinent to each of the major design goals for the GeoTrans data model. The actual structure of the data model is discussed later in Section 4, *Data Model Structure*.

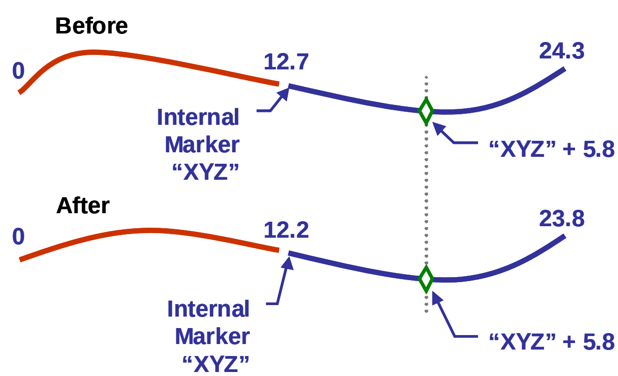
3.1 Event Location Stability

One of the weaknesses of linear referencing is that the spatial description of the event data is based on a dynamic linear network. Suppose that there is a realignment on a road that reduces the length of the road by a half mile. If the measurement attributes of the road are updated to reflect this new length, all of the 'measure-referenced' event data that occurs on segments "downstream" from the realignment will mysteriously move "downstream" by a half mile. This is not the desired result! If an event, such as a traffic accident, happened at a certain spot, you want it to stay there!



One common approach to this problem is to build and maintain a complicated series of database triggers that cascade any network changes throughout the network and to any event data (that the system knows about). The designers of this model think that this method is overly complicated and expensive and is really just a work around to what is really needed: a data model designed from the ground up with event location stability in mind.

Our approach to preventing this unwanted wandering of event data is to relate the location of event data to individual geometric segments. This localizes any changes to just those segments that were realigned. This act of tying event data to geometric segments is done via a one-time step known as Event Registration (see Section 5.1, *Event Registration*). The Event Registration step makes a revised copy of your event data that uses the internal marker offset method of location. After an event table has been "registered," no further action is necessary to accommodate changes in the linear network. Your event data will stay put!



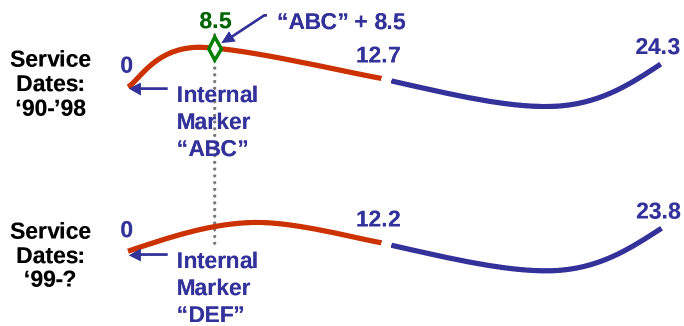
To support this method of maintaining event location stability, the GeoTrans data model has a field on both the geometric segments and the LRMs that can be used as the internal marker for each segment, thus giving each segment a unique name. This solves the problem for the "downstream" segments, but there is still the problem of event data tied to the realigned segment. This problem is addressed in the next section on temporal persistence.

3.2 Temporal Persistence

As mentioned above, most transportation networks are dynamic. New roads are added, existing roads are extended and realigned, and old roads are sometimes abandoned or demolished. All of these changes happen in time. A view of a transportation network today is often quite different than it was 20 years ago or what it will be 20 years into the future. Because of this, it is often desirable to be able to show what the network looked like at any given point of time and to be able to show the event data associated with the network at that time. For a variety of reasons, including legal, it is also often important to be able to show what the database looked like at a given point in time. These are two separate but related temporal issues.

To address the first issue, we have added begin_service and end_service date attributes to the base linear segments and to the LRM segments. These dates represent the item's valid lifespan. To address the second issue, we have added a DBtimestamp to these items that stores the time that the item was created in the database.

Any time a feature is edited to represent a change in the physical network, the end_service is filled out for the existing segment, and a new copy of the segment is created with a begin_service value equal to the end_service value of the original segment. Equally important, the new feature receives a new unique internal marker name. This last task is essential for event location stability. Because of this step, Events that occurred along the original segment will remain linked to that segment and not to the new edited segment, thus preserving the location of the event.



Event data may also have begin_service and end_service date attributes, or it may only have a single date attribute. Events, such as construction projects, that exist over a period of time will usually use the two date method; whereas incidents that are just points in time, such as traffic accidents, will normally use the one date method.

Queries can be made on the LRS linear features for any time frame for which data exists. Dynamic segmentation and other analyses can be performed on the results of these queries. Similarly, temporal queries can be performed on event data, and then the results of those queries can be used for dynamic segmentation.

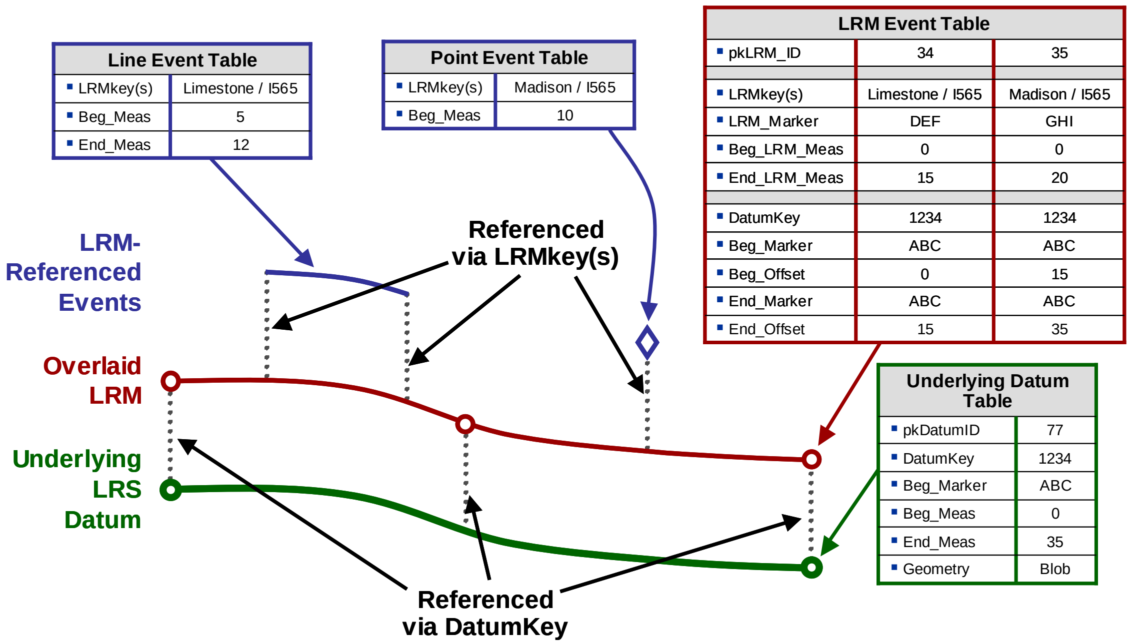
As you can tell from the above, the actual effect of supporting temporal persistence on the data model is simple – just a few data fields. It's the data maintenance that can be difficult. The processes involved in this data maintenance are discussed in Section 5.2, *Temporal Data Maintenance*.

3.3 Multiple LRM Support

Most every organization that captures and analyzes transportation information uses more than one Linear Referencing Method (LRM). Examples of LRMs commonly used are County Milepost, Engineering Stationing, and Street Addresses. In a County Milepost LRM, the LRS keyset may consist of County and Route number attributes, and the measurements are in miles. In an Engineering Stationing LRM, the LRS keyset may consist of Project number and Route number attributes, and the measurements are in feet. In a Street Address LRM, the LRS keyset may consist of Street Prefix, Street Name, Street Type, and Street Suffix attributes, and the measurements are unitless.

There are a number of approaches that are commonly used for handling a multiple LRM system, most of which have major disadvantages. One is to use one set of geometry, but to carry as attributes on that geometry all of the keyset and measurement attributes needed for all of the supported LRMs. This necessitates that the geometry be segmented at every change in attribution for every LRM. This makes maintenance of the LRS quite a chore. Another way is to have separate geometry for each LRM. This increases the storage needs, but, more importantly, it necessitates that any geometry change be made multiple times, once for each LRM.

The approach we have used in the GeoTrans data model is to model each LRM as linear events that are dynamically segmented onto the base geometry. This way each LRM can be maintained independently of each other, but they still share the same geometry. As shown in the following diagram, the bottom half of the LRM Event table contains references that allow it to be geocoded onto the underlying geometry. The top half exposes LRM-specific naming and measurement conventions that allow events to be geocoded onto it.



The example in this diagram shows a County Milepost LRM being overlaid onto underlying geometry. The LRM breaks at a county boundary even though the underlying geometry does not. Furthermore, the measurements on the LRM restart at 0 at the county boundary even though the measurements on the underlying geometry do not. The event data shown is referenced to the County Milepost LRM and not to the underlying geometry. This is made possible because the LRM inherits the underlying geometry via dynamic segmentation. It is possible to create as many of these LRM Event tables as desired, and they all are independent of each other — yet they all share the geometry located in the LRS datum.

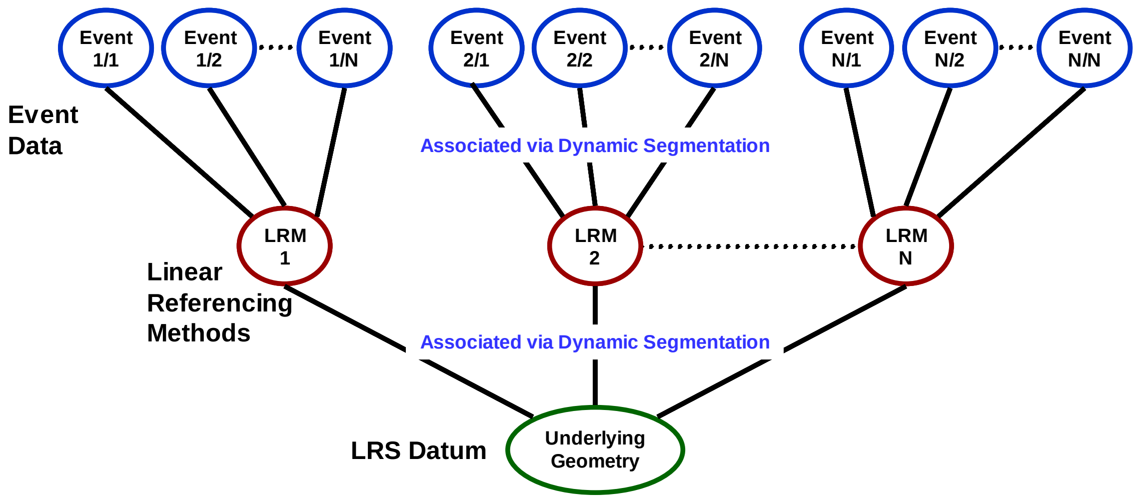
There are two noteworthy aspects to implementing LRMs via event tables. First, please note that all location references in the LRM Event tables use internal marker offsets. This provides location stability for the LRMs. Event data tied to the LRM using internal marker offsets will share this same stability. Secondly, LRM Event tables should be dated as described in Section 3.2, *Temporal Persistence* to allow for temporal queries.

4. Data Model Structure

In the preceding section, the challenges to transportation data modeling were discussed piecemeal. In this section, we bring all of those components together to create the overall transportation data model.

4.1 Overall Data Model Structure

The diagram below provides a schematic of the overall transportation data model. It can be thought of as consisting of three layers, each linked to the next via dynamic segmentation. There is only one LRS Datum table, which defines the underlying geometry. There can be any number of Linear Referencing Method tables linked to the LRS Datum. In the top layer, there can also be any numbers of Event tables, which, depending on their attribution, can be linked to any of the LRM tables. The sections to follow define each of these layers.



4.2 LRS Datum Layer

The LRS Datum layer is where the underlying network geometry resides. It is also where the base road naming and measurement system resides. Since the naming convention and the measurement system of this LRS Datum layer is not directly used by the average end-user, the main criterion is that it should be made simple and easy to maintain. However, for maximum flexibility, it is recommended that the traversals (groups of linear segments that share the same key values) be as long as possible. This is because linear events (such as the LRM Events discussed in the next section) must start and stop on the same traversal. However, the length of the individual segments that make up the traversal is another matter and is discussed in Section 5.4, *Segment Length*. The LRS Datum layer consists of just one table, which is described below.

LRS Datum		
Field		Description
ID	pkDatumID	This is a non-meaningful ID that uniquely identifies linear features within the LRS Datum table.
Datum Keyset	DatumKey	It is recommended that this be a non-meaningful ID (i.e., something like 5346, not SR66) that uniquely identifies traversals within the LRS Datum table. This field correlates to the DatumKey field of the LRM Event tables.
Measurement	Marker_Name	This is a unique ID for a segment within a traversal. It only needs to be unique among segments that share the same DatumKey value. Some users may want to use pkDatumID as the Marker_Name, and that is perfectly acceptable. This field is used to tie LRMs to a given LRS Datum feature in a non-volatile fashion.
	Beg _Meas	These are the measures for the beginning and ending points of an LRS linear feature. These define the relative length of a given segment. The units of measure can be pretty much anything as long as they are consistent within the entire LRS Datum. There are 28 choices of units supported by the GeoMedia Transportation products.
	End_Meas	
	Geometry	This is the cartographic representation of the segment.
Timestamp	Beg_Service	These two fields define the time period for which a particular LRS Linear feature is valid. If the End_Service date is not known, just leave the value as Null.
	End_Service	
	DBtimestamp	This field defines the time that this record was first entered into the database.

4.3 LRM Layer

The LRM layer is the glue that links event data with the geometry on the LRS Datum layer. These tables have both references to the underlying geometry layer, and they exposee a naming and measurement convention of their own. The generic structure of these tables is depicted below.

<table>
<tr><th colspan="3">Linear Referencing Methods</th></tr>
<tr><th colspan="2">Field</th><th>Description</th></tr>
<tr><td>ID</td><td>pkLRM_ID</td><td>This is a unique, non-meaningful ID for a given segment of this LRM.</td></tr>
<tr><td rowspan="4">LRM Reference</td><td>LRMkey(s)</td><td>These are one to four ID fields that together uniquely identify the LRM traversal that this segment is a part of.</td></tr>
<tr><td>LRM_Marker</td><td>This is a unique ID for this segment within this LRM traversal. It only needs to be unique among LRM segments that share the same LRMkey(s) values. Some users may want to use pkLRM_ID as the LRM_Marker, and that is perfectly acceptable. This field is used to tie Events to a given LRM segment in a non-volatile fashion.</td></tr>
<tr><td>Beg_LRM_Meas</td><td rowspan="2">These are the measures for the beginning and ending points of an LRM segment. These define the relative length of a given LRM segment. The units of measure can be pretty much anything as long as they are consistent within the entire LRM table. There are 28 choices of units supported by the GeoMedia Transportation products. For unitless measures, such asstreet addresses, you can use any of these units, but make sure to pick the same units when specifying your event data.</td></tr>
<tr><td>End_LRM_Meas</td></tr>
<tr><td rowspan="5">Datum Reference</td><td>DatumKey</td><td>This is a non-meaningful ID that uniquely identifies the Datum traversal that a segment of the LRM is mapped to. This field correlates to the DatumKey field of the LRS Datum table.</td></tr>
<tr><td>Beg_Marker</td><td>This correlates to the Marker_Name of the LRS linear feature in the LRS Datum table that the beginning of this LRM segment is mapped to.</td></tr>
<tr><td>Beg_Offset</td><td>This is the offset from the marker identified in the Beg_Marker field where this LRM segment begins.</td></tr>
<tr><td>End_Marker</td><td>This correlates to the Marker_Name of the LRS linear feature in the LRS Datum table that the ending of this LRM segment is mapped to.</td></tr>
<tr><td>End_Offset</td><td>This is the offset from the marker identified in the End_Marker field where this LRM segment ends.</td></tr>
<tr><td rowspan="3">Timestamp</td><td>Beg_Service</td><td rowspan="2">These two fields define the time period for which a particular LRM event record is valid. If the End_Service date is not known, just leave the value as Null.</td></tr>
<tr><td>End_Service</td></tr>
<tr><td>DBtimestamp</td><td>This field defines the time that this record was first entered into the database.</td></tr>
</table>

You can have as many of these tables as you want, and most organizations will have several. The LRM key fields and the units of measure chosen inside of the GeoMedia Transportation products are what will vary from one LRM table to the next. The following table shows some samples of this variation.

Sample Linear Referencing Methods			
Linear Referencing Method	**County Milepost LRM**	**Street Addresses LRM**	**Engineering Stationing LRM**
LRMkey(s)	County	Street_Prefix	Project_Num
	Route	Street_Name	Route
	Carriageway	Street_Type	StationEquationID
		Street_Suffix	
Units	Miles	Anything that matches event data units	Feet

An optional table that can be used is called an External Marker table. This table type is needed to support events of the external marker offset type. There are a number of cases in which events of this type are useful. One instance is for events referenced to mileposts whose locations do not match what is printed on the sign (see Section 5.11, *Reference Mileposts*) and another is for intersection-referenced events (see Section 5.12, *Intersection-referenced Events*).

This table is populated with point events that can lie anywhere along the LRM. Events of the external marker offset type can then be located by an offset from these points. These tables are linked to individual LRM Event tables and derive their locations from these tables. The basic format of an External Marker table is shown in the table below.

External Markers	
Field	**Description**
LRM Keyset	These are the field(s) that together uniquely identify which LRM traversal a marker is being mapped to. These fields should correlate with the LRMkey field(s) in the LRM Event table to which the specific External Marker table is associated.
Measure	This defines the location of the external marker along the LRM traversal identified by the LRM Keyset fields.
Name	This field defines the name given to the external marker. It should be unique with this LRM traversal.

4.4 Event Data Layer

The Event Data layer is where the greatest amount of variability occurs. Part of this is because the GeoMedia Transportation products provide support for event data in a variety of formats. Even so, there are four standard components that every event table should have. The table below shows those standard components.

Event Tables	
Component	**Description**
LRM Keyset	These are one to four ID fields that together uniquely identify which LRM traversal an event is being mapped to. These fields should correlate with the LRMkey field(s) in the LRM Event table to which the specific event table is associated.
Location	These fields define where an event should be mapped to along an LRM traversal. There are a number of different formats that this location can take, and these formats are shown in a table later in this section. Regardless of the format chosen, these fields should correlate with the LRM Reference data in the LRM Event table to which the specific event table is associated.
Timestamp	These field(s) identify the valid timeframe for an event. For some events, such as construction projects, this might require two fields: a begin and an end date. For other events, such as traffic accidents, the events may be considered as occurring at points in time and therefore only require one date field.
Attributes	Attributes for an event are totally specific to the event table and can be anything that is needed. The only guideline is that it is good practice to store these attributes in a separate table if multiple cartographic representations of the event are desired (see Section 5.10, *Multiple Event Representations*). In that case a foreign key ID will be stored in the event table to link back to the table containing the event attribution.

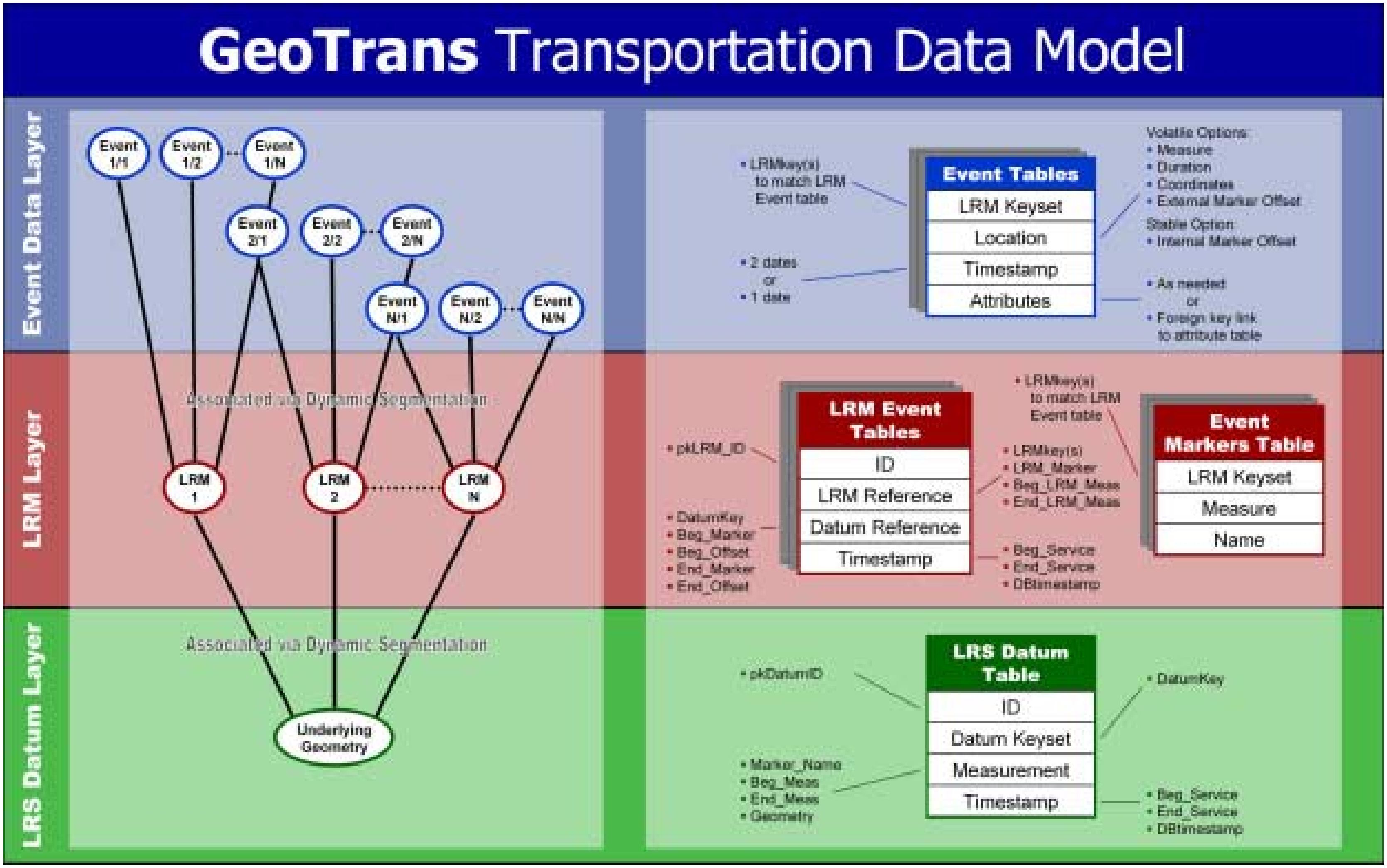
There are 5 different formats that can be used to structure the fields that define the location information for the event table. The table below depicts these formats. All, but the last format are listed as volatile as it is possible for the event data to shift if there are changes to the measurement attribution in the underlying LRM. The last option is designed to eliminate this shifting as is discussed in Section 3.1, *Event Location Stability*. In practice many or all of these formats will be used, but it is recommended that event data be converted to the internal marker offset type using the Event Registration step referred to in Section 5.1, *Event Registration*.

Location Options		
Option		**Description**
Volatile Events	Measure	This method requires a start measurement field and (for linear events only) an end measurement field. These fields hold the measurement values along the LRM for the beginning and ending positions of the event.
	Duration	This method requires two fields, a start measurement and a duration (length) value. The start measurement is the measurement value for the beginning position of this event. The duration is the distance value from the beginning position to the ending position of this event. Note: Because of its format, this method only applies to linear events.
	Coordinates	This method requires begin X and Y coordinate (or longitude and latitude) fields and (for linear events only) end X and Y coordinate (or longitude and latitude) fields.
	External Marker Offset	This method requires a begin marker name field and an offset field and (for linear events only) an end marker name field and an end offset field. These marker names are references to an external marker table.

Location Options		
Option		Description
Stable Events	Internal Marker Offset	This method requires a begin marker name field and an offset field and (for linear events only) an end marker name field and an end offset field. These marker names are references to internal marker names within a specific LRM Event table.

These options for event data are discussed in more detail in the "GeoMedia Transportation Data Structures" appendix of the *Working with GeoMedia Transportation Manager* document. Some sample event tables are presented in this document in Appendix C – Sample Event Tables.

4.5 Conceptual Model and Physical Model Template



5. Implementation Approaches

This section gets into the details of how to implement the GeoTrans data model. It addresses some of the trickier situations to address in a data model as well as some of the workflows dictated by the GeoTrans data model.

5.1 Event Registration

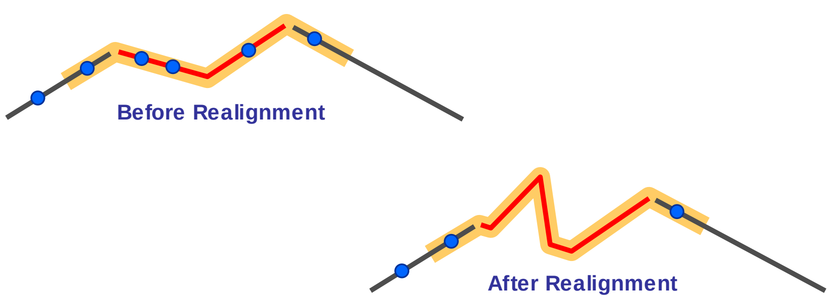
The Event Registration step simply takes event data and converts the measurement data to an internal marker offset. For point events, this is done quite simply by using the Event Conversion command in GeoMedia Transportation Manager. The steps for doing this are as follows:

1. Use GeoMedia Transportation Manager to perform an Event Conversion on the input Event table to the internal marker- offset type. The important new fields to come out of this step are Marker1 and Offset1.

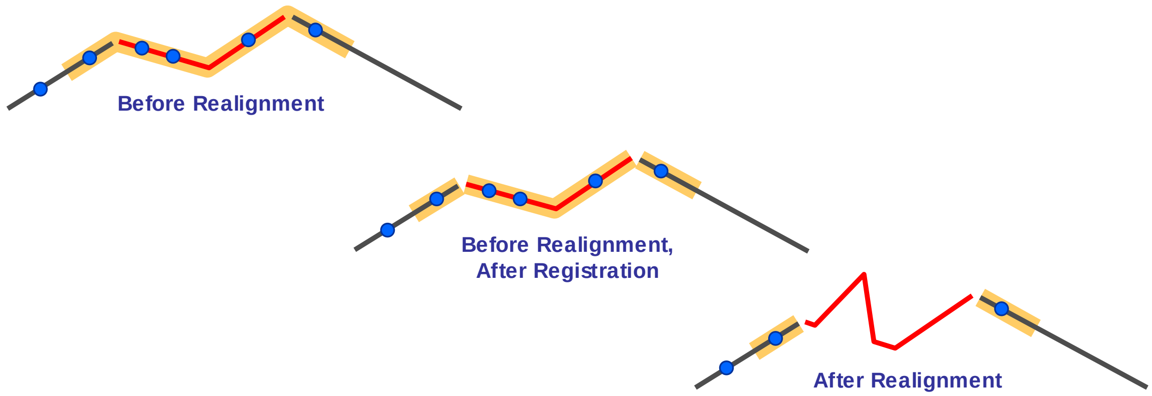
2. Use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.

3. Use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields, such as the original measurement fields. Make sure to keep the Marker1 and Offset1 fields.

For most linear events, the steps outlined above will also work well. However, there is one case where this one step is insufficient. That one case is illustrated in the figure below.



The figure above shows three LRM segments with several point events and one linear event mapped to them. Note that the point events that were mapped to the middle segment on the top part of the figure do not show up in the bottom part. This is as desired. These point events were “registered” to the original middle segment. When the middle segment was edited, the new middle segment was given a new internal marker name. Because of that, these “registered” point events do not show up in what would have been the wrong location. Unfortunately, this is not the case for the linear event. Because the segments in which this event begins and ends are unchanged, this linear event maps even when it should not. The answer is to break the linear events into pieces that are no longer than any of the underlying LRM segments. See the following diagram to see how this should work:



The steps for splitting up linear events vary depending on what type of attribute data is carried on the event. Some attributes, for example the number of lanes, would stay the same for each of the new pieces of the event. The steps for processing this kind of linear event are as follows:

1. Use GeoMedia Transportation Manager to perform an Event Overlay between the results of the last step and the LRM event table it is mapped to. Choose the Intersection option. The LRS Feature class to use is the DynSeg of the LRM Event table. Choose the LRM reference fields in the Properties dialog. The First Event Feature Class to use is the Input Linear Event table. The Second Event Feature Class to use is the LRM Event table. Choose the LRM reference fields in the Properties dialog. The important new fields to come out of this step are OutputBeginMeasure1 and OutputEndMeasure1.

2. Use GeoMedia Transportation Manager to perform an Event Conversion on the results of the previous step to the internal marker offset type. The important new fields to come out of this step are BeginMarker1, BeginOffset1, EndMarker1, and EndOffset1.

3. Use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.

4. Use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields, such as the original and intermediate measure fields. Make sure to keep the BeginMarker1, BeginOffset1, EndMarker1, and EndOffset1 fields.

For some event tables, you will want to proportionally allocate the attributes to the new pieces of the event. An example of this might be the cost of construction projects in which you would want each piece of the event to have a fraction of the total cost proportional to the length of the piece. The steps for processing this kind of linear event are as follows:

1. Use the GeoMedia Transportation Manager Dynamic Segmentation command to add geometry to your event table.

2. Add a length field to the results of the last step using the Analyze Geometry command in GeoMedia Professional. Usually this field will be named Length.

3. Use GeoMedia Transportation Manager to perform an Event Overlay between the results of the last step and the LRM event table it is mapped to. Choose the Intersection option. The LRS Feature class to use is the DynSeg of the LRM Event table. Choose the LRM reference fields in the Properties dialog. The First Event Feature Class to use is the results of the last step. The Second Event Feature Class to use is the LRM Event table. Choose the LRM reference fields in

the Properties dialog. The important new fields to come out of this step are OutputBeginMeasure1 and OutputEndMeasure1.

4. Add another length field to the results of the last step using the Analyze Geometry command in GeoMedia Professional. Usually this field will be named Length01.

5. For those attribute fields that need to be proportionally divided, create a new attribute field using the Functional Attributes command in GeoMedia Professional by talking the original value and multiplying it by the Length01 field and by dividing it by the Length field.

6. Use GeoMedia Transportation Manager to perform an Event Conversion on the results of the previous step to the internal marker offset type. The important new fields to come out of this step are BeginMarker1, BeginOffset1, EndMarker1, and EndOffset1.

7. Use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.

8. Use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields, such as the original and intermediate measure fields. Make sure to keep the BeginMarker1, BeginOffset1, EndMarker1, and EndOffset1 fields.

The results of these registration processes can be safely stored and depended on as the organization's "stable" data. There are plans to include tools in future versions of the GeoMedia Transportation products to automate these steps, but until then these steps can be used to ensure the stability of your data.

5.2 Temporal Data Maintenance

As discussed in Section 3.2, *Temporal Persistence*, there are four fields that are critical to temporal persistence and event location stability: begin_service, end_service, DBtimestamp, and the marker name (either Marker_Name or LRM_Marker). Whenever a Datum or LRM segment is edited, care must be taken to ensure that these fields have the proper values. The steps involved in this are as follows:

1. Make a copy of the feature to be edited.

2. Enter an end_service date for the original feature.

3. Edit the copied feature as desired.

4. Enter the present date and time into the DBtimestamp field of the copied feature.

5. Enter a begin_service date on the copied feature equal to the end_service date of the original feature. For current features, the end_service date can be left blank unless an out of service date is already known.

6. Lastly, assign a new unique internal marker name to the copied feature. This is one of the keys to maintaining event location stability. This way, events located via reference to an internal marker will be linked to only one segment, which is itself linked to a specific timespan.

It is probably quite apparent that these steps would be rather arduous and error-prone if much editing is required. That is why we recommend an automated approach. This is where GeoMedia Transaction Manager (GTM) comes in. GTM works in coordination with GeoMedia Professional and Oracle's Workspace Manager to automate this process as you edit your data as you normally would. GTM is not a “shrinkwrapped” product, but rather a customizable product that can be configured to perform the workflow outlined above. It is recommended because it both saves time and helps ensure data integrity. The database manager can pick and choose which tables are to be managed by the GTM product. It is likely that most implementations will only need GTM to manage tables in the LRS Datum layer and the LRM Event layer (see Section 4.1, *Overall Data Model Structure*).

5.3 LRM Conversion

It is often desirable to convert event data that is referenced to one LRM so that it is referenced to another LRM. This can be useful for a number of reasons. One is to allow event data collected using a variety of LRMs to be reported against a single LRM. The steps to perform this conversion are as follows:

1. Use the Event Conversion command in GeoMedia Transportation Manager to convert the original event data to a coordinate measure.

2. Use the LRS Keys For Coordinate Events command in GeoMedia Transportation Manager to assign keys from the target LRM to the results of the previous step.

3. If desired, use the Event Conversion command in GeoMedia Transportation Manager to convert the results of the last step to another measure type. Make sure to use the keys from the last step and the coordinates from the first step as input.

4. If desired, use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.

5. If desired, use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields from this table, making sure to keep the new keys and measures as well as any important descriptive attributes about the event.

5.4 Segment Length

LRS segments store length in two distinct ways. One is the geometric length, which is 2D, and the other is the measure attribution length, which in most cases is 3D. There are a number of reasons why these two lengths will differ, including accuracy of digitization and slope on the ground. The bottom line is that the ratio between these two lengths will vary over most datasets, and so it is wise to use linear segments that are not overly long. This allows each segment to reflect the ratio that is local to that area and not just reflect an average ratio over a much larger area.

Another consideration is that events are located by interpolation from the end points of the LRS segments. The longer the segments, the more room for inaccuracy in interpolation. This is another reason that LRS segments should be kept relatively short.

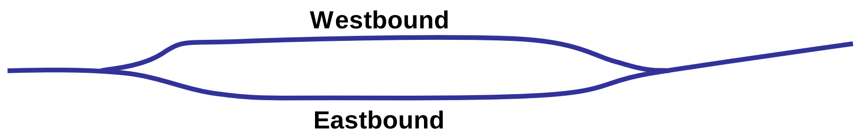
5.5 Performance Considerations

Normalized databases, which usually have a lot of tables with few fields and many joins between them, are ideal for data maintenance because they minimize the chances for the introduction of errors during updates and deletions. Likewise, the pipes that GeoMedia and GeoMedia-based products use are very useful in that changes to the source data are automatically cascaded throughout the downstream processes, such as labeling and dynamic segmentation. Conversely, large denormalized tables with no joins and geoworkspaces with few pipes are optimal for performance. Our recommendation is that you use normalized tables and workspaces with as many pipes as you need to build and maintain your data, but that you present your end users with denormalized tables with a minimum number of pipes to use for analysis. These denormalized tables should be read-only copies of the source data. Procedures, either manual or automated, should be set up to create these read-only tables on a periodic basis from the source data kept in normalized tables.

An example of where this applies to the GeoTrans data model is with LRMs. For an LRM event table to be of use, it must be dynamically segmented against the Datum geometry. The GeoMedia Transportation Manager Dynamic Segmentation command utilizes pipe technology to generate a copy of the LRM Event table with an additional geometry field. For performance considerations, the results of this dynamic segmentation can be written to a table using either the Output to Feature Class command or the Update Attributes command.

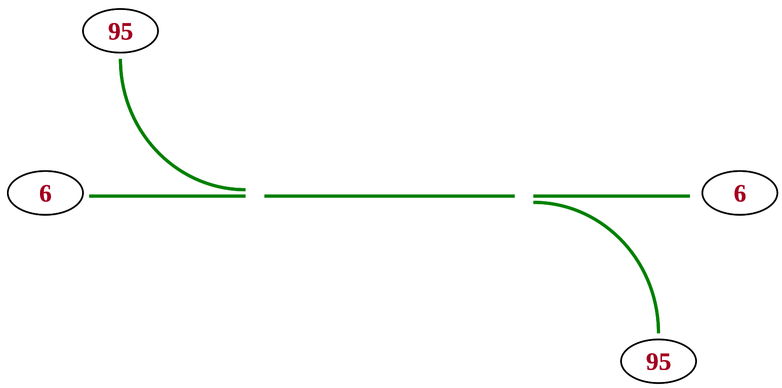
5.6 Divided Highways

Many datasets use a single centerline for lower classification roads, but use two centerlines for divided highways. This double centerline creates ambiguity as to which centerline events should be geocoded to. The bottom line is that there needs to be an unambiguous name and measurement for each point on the network. One approach is to add an extra LRS key field, perhaps named Carriageway, to distinguish between the two centerlines. The values for this field would typically be N, E, S, or W indicating the traffic direction. Note that LRS key fields cannot be Null, so for single centerline segments the value for Carriageway might be something like B for both. In order for this to work, event data also needs to be collected with Carriageway attributes.



5.7 Overlaps and Gaps

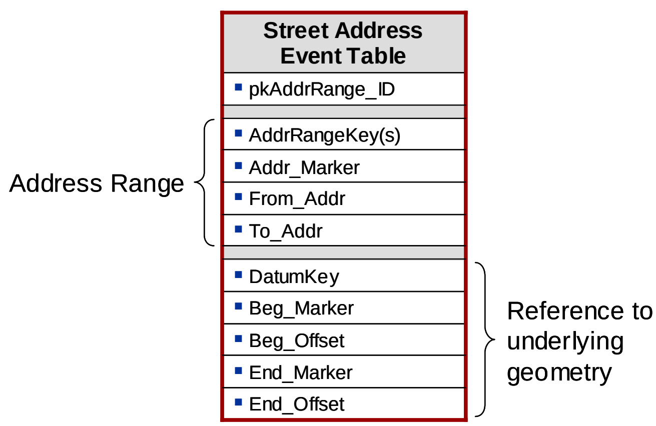
Overlaps occur when two roads come together for a while and then split again. This portion of the road where the two become one may be referred to as either one route number or both. Depending on the application, each of these possibilities may be what is needed. This is part of the beauty of supporting multiple LRMs. You can have an LRM for each of these possibilities. For the case where you want the route number to be both, simply have two entries in the LRM Event table. Both entries would have identical data referring to the underlying geometry, but they would have different LRM keyset and LRM measurement attributes.



There are a variety of situations where gaps may occur in the transportation network. One is shown in the following picture, where a state route changes name as it passes through a city. In this case, this creates a gap in State Route 1. Although these situations are flagged by GeoMedia Transportation Manager's Validate LRS command, they are perfectly acceptable and will not affect the performance of commands like Dynamic Segmentation or Event Overlay.

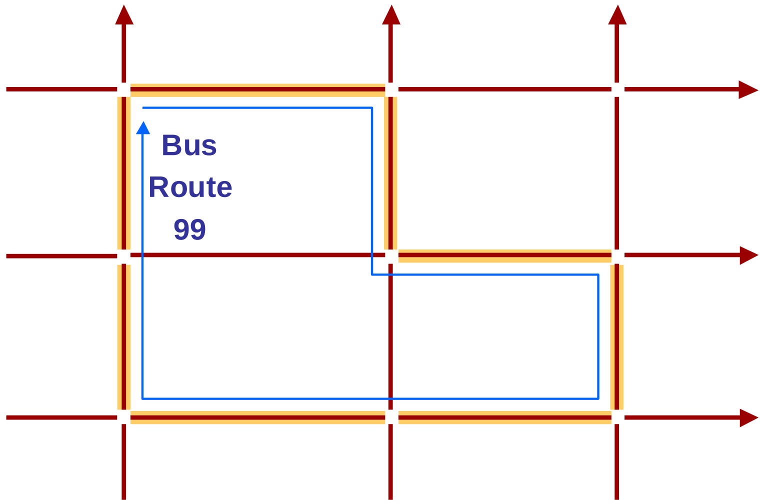
5.8 Address Matching

As mentioned in the first paragraph of Section 3.3, *Multiple LRM Support*, Street Addresses are simply another LRM. As such they are modeled with an LRM Event table, as shown in the following diagram. The main difference is that the LRM measurements in this case are unitless. To accommodate this, simply use the same measurement units in both the LRS Options and Event Options dialog boxes in GeoMedia Transportation Manager. It doesn't matter which units you choose as long as they are the same in both dialog boxes. Please make note that this affects the use of marker offset events. Since the length attribute of the linear segments is unitless, it will not correctly interpret real world units in your offsets. This makes this type of LRM unusable with marker offset events (except for event registration).



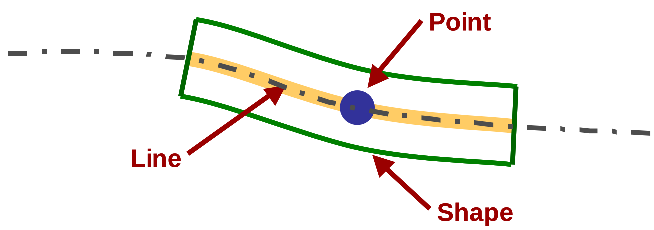
5.9 Bus Routes

Transit Agencies often want to be able to track events according towhere they occur along bus routes. For these purposes, these bus routes can be modeled like any other LRM, but with one difference. It is inevitable that some legs of the bus route will be in the opposite direction of the increasing measures and digitizing of the underlying datum. Because of this, in the datum reference portion of the LRM Event table, it will sometimes be necessary to go from a higher offset to a lower offset. When dynamically segmented, these segments of the LRM will have an opposite digitizing direction than the underlying datum segments, as desired.

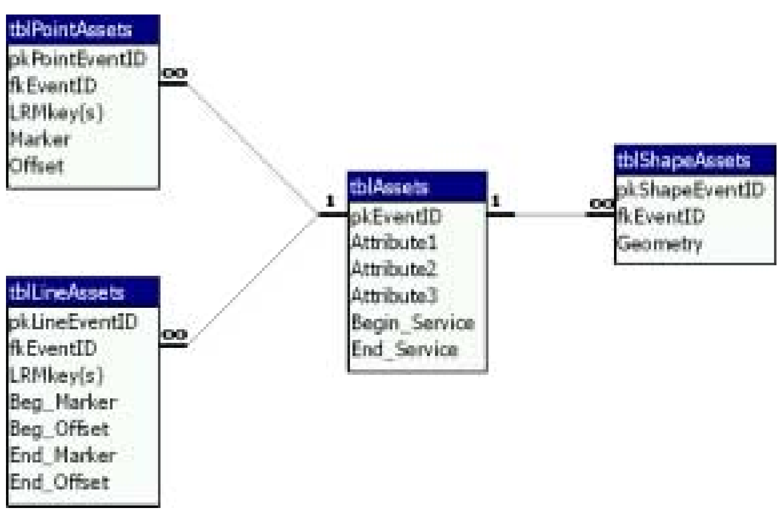


5.10 Multiple Event Representations

Often there is a need to depict a single asset in different ways depending on a particular need. An example is a bridge. For some depictions, such as a large-scale drawing, a point may be the appropriate way to show a bridge; for other depictions, a line or even a shape may be appropriate. One of the possible results of having multiple representations is the duplication of data and the data maintenance problems that this causes.



Our recommendation for modeling multiple asset representations is to separate the asset attributes from its cartographic depiction, as shown below. Point and line assets will derive their geometry from dynamic segmentation; whereas the shape assets are GIS features in their own right.



All the asset attributes are stored in one central table for easy data maintenance. Database views can be used to allow the viewing of both the geometry and the asset attributes in one virtual table if desired. If your database supports it, it is recommended that a cascade delete constraint be used to ensure that all geometric depictions of an asset are automatically deleted if an asset is deleted.

5.11 Reference Mileposts

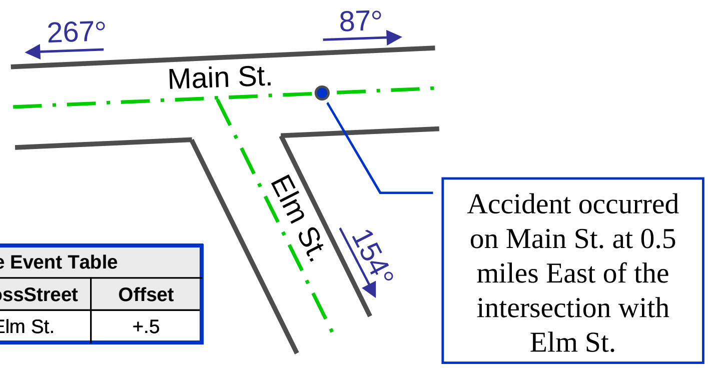
For roads that have mileposts placed alongside them, one of the most convenient and common methods for collecting event data locations is by measuring the distance from the closest milepost to the event. The trouble with this method is that the milepost is often not accurately located where the text of the sign says it is. Many transportation agencies have taken to calling these "reference" mileposts and using the marker offset method to measure the distance from these known locations, as opposed to taking the measure on the milepost at face value. This requires that the location of the mileposts be measured, often using GPS.

So the first step towards implementing a reference milepost system is turning these GPS latitude/longitude measurements into external markers. More than likely, the collected data on the mileposts will include latitude/longitude, the route the milepost lies along, and the mileage printed on the sign. The only step needed to turn this into an external marker table is to use the Event Conversion command in GeoMedia Transportation Manager to convert the latitude/longitude into a measure. Note that this measure is unlikely to match what is printed on the sign. The printed mileage will be used as the marker name.

Events collected using the reference milepost system should include LRMkey(s), Marker Name (the mileage printed on the sign), and an offset distance (positive if the event's measure is greater than the milepost and negative if less). The problem lies with the legacy data that may have been collected by reference milepost but which was stored using only LRMkey(s) and Measure. For data of this sort, one approach is to parse out the Marker Name and Offset from the Measure. This can be done with the Functional Attributes command in GeoMedia Professional. You can create the Marker Name field by truncating the Measure field, and you can create the Offset field by subtracting the new Marker Name field from the Measure field. The results of the Functional Attributes command can then be used as a marker offset type event.

5.12 Intersection-referenced Events

The intersection reference is perhaps the most utilized linear referencing method in urban areas. It defines a location by Occurred On Street, Cross Street, Distance from Intersection, and Direction from Intersection. The proposed method of dynamically segmenting intersection-referenced event data is to make use of the External Marker offset method supported by GeoMedia Transportation Manager. A total of three tables are required for this: an LRS feature class that contains the linear network, an Intersection Marker table, and the Event table. The LRS feature class is no different than is required for any other type of event. The Intersection Marker table is relatively static and can be created periodically via an automatable process (more on this later). The Event table has to be modified slightly from the data collected in the field, and this can either be done via an automatable process or via a workflow that takes advantage of one of the new features in GeoMedia Professional version (more on this later).



Sample Event Table		
LRMkey	**CrossStreet**	**Offset**
Main St.	Elm St.	+.5

Sample Intersection Marker Table					
ID	**LRMkey**	**Measure**	**CrossStreet**	**AzimuthPlus**	**AzimuthMinus**
112	Main St.	12.3	Elm St.	87	267
113	Elm St.	45.6	Main St.	-1	154

The picture above represents an example of a three-legged intersection followed by the portion of the Event table and the Intersection Marker table that represents the intersection and this one event. The table below describes the fields in these two tables.

Sample Event Table	
Field	**Description**
LRMkey	This is the "Occurred On" street mentioned earlier. This consists of one to four fields and needs to match the LRMkey fields in the LRM Event table.
CrossStreet	This is the name of the "Cross Street" of the intersection. It also is the "marker name." This can only be one field, so it may have to be a concatenation if more than one LRMkey field is used. This needs to match the CrossStreet field in the Intersection Marker table.
Offset	This is the distance that the event occurred from the intersection. It is positive if the offset is in the direction of increasing measure, and negative if the offset is in the direction of decreasing measure.

Sample Intersection Marker Table	
Field	**Description**
ID	The ID field is not required, but it is good database design practice. It uniquely defines each Marker. There is one marker for each road involved in an intersection (two in the example picture).
LRMkey	This is the "Occurred On" street. This consists of one to four fields and needs to match the LRMkey fields in the LRM Event table.
Measure	This is the measure of the Occurred On street at the point of intersection.
CrossStreet	This is the name of the "Cross Street" of the intersection. It also is the "marker name.". This can only be one field, so it may have to be a concatenation if more than one LRMkey field is used.
AzimuthPlus	This is the azimuth of the "Occurred On" street going from the intersection in the direction of increasing measure. If the "Occurred On" street *terminates* at this

Sample Intersection Marker Table	
Field	**Description**
	intersection, so that there are no measures on the LRS feature *higher* than the measure at the intersection, then this value will be -1 (as is the case with Elm St. in the example above).
AzimuthMinus	This is the azimuth of the "Occurred On" street going from the intersection in the direction of decreasing measure. If the "Occurred On" street *originates* at this intersection, so that there are no measures on the LRS feature *lower* than the measure at the intersection, then this value will be -1.

Populating the Intersection Marker Table

The population of the Intersection Marker table can be done manually, but it is arduous enough that it should be automated. The basic steps are as follows:

1. Perform a GeoMedia Professional Spatial Intersection of the LRS feature class against itself (using the Touch operator) in order to locate all of the intersections within the dataset.
2. Use a query to exclude those found records where the first set of LRMkeys is the same as the second set of LRMkeys.
3. Use the Functional Attributes command in GeoMedia Professional to create a field that is a concatenation of the second set of LRMkeys to make the CrossStreet field.
4. Add a length field to the recordset using the Analyze Geometry command in GeoMedia Professional.
5. Use a query to exclude any records that have a Non-Null Length field.
6. Add X and Y fields to the recordset using the Analyze Geometry command in GeoMedia Professional.
7. Use the Event Conversion command in GeoMedia Transportation Manager to use the LRMkey(s) and the X,Y coordinates to add a Measure field to the recordset. This is the Measure at Intersection field.
8. Use the Functional Attributes command in GeoMedia Professional to create two new fields, one that has a value that is .1 greater than the measure at intersection and another that is .1 less than the measure at intersection.
9. Use the Event Conversion command in GeoMedia Transportation Manager twice to use the LRMkey(s) and the two new measures to add X,Y values for each of these two measures to the recordset.
10. Use the Functional Attributes command in GeoMedia Professional to create two new fields, one that is the azimuth from the intersection to the point with a .1 smaller measure (AzimuthMinus) and one that is the azimuth from the intersection to the point with a .1 larger measure (AzimuthPlus). If there is no intersection leg in one of these directions, use a value of -1 for that field. The azimuth is calculated as shown below:

    Azimuth = $90° - \mathrm{Tan}^{-1}(\Delta Y/\Delta X)$ + (180°, if the X value of the "to" point is less than the X value of the intersection)

    Here are some sample Functional Attribute equations for these 2 fields:

    <u>AzimuthMinus</u>
    if (Input.X1 is null, -1, (90 - Degrees(ATAN((value(Input.Y1) - value(Input.SecondProjectionCoordinate)) / (value(Input.X1) -

value(Input.FirstProjectionCoordinate)))) + IF((value(Input.X1) - value(Input.FirstProjectionCoordinate)) < 0,180,0)))

<u>AzimuthPlus</u>
if (Input.X2 is null, -1, (90 - Degrees(ATAN((value(Input.Y2) - value(Input.SecondProjectionCoordinate)) / (value(Input.X2) - value(Input.FirstProjectionCoordinate)))) + IF((value(Input.X2) - value(Input.FirstProjectionCoordinate)) < 0,180,0)))

11. Use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.
12. Use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields, and rename the ones you wish to keep as desired. The fields that need to be kept are those shown in the Sample Intersection Markers table above.

Transforming the Event Table

The source data for an intersection-referenced event will usually have the following location fields: Occurred On Street, Cross Street, Distance from intersection, and Direction from intersection. The direction is usually a text field containing a cardinal direction represented as a single character: N, E, S, or W. The event table we need has the first three fields mentioned, but the Direction field is represented only by the sign (+ or -) of the Offset distance. The steps below show how to make the transition between these formats:

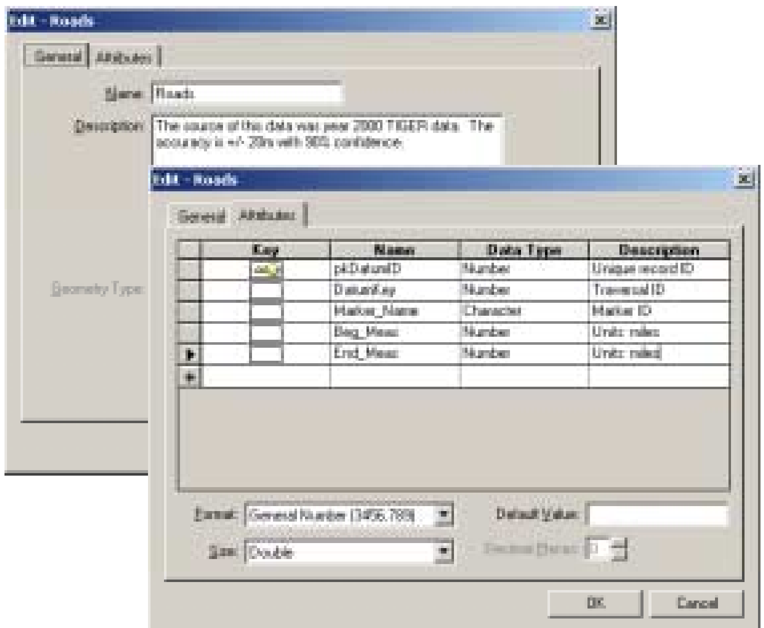
1. Use the Join command in GeoMedia Professional to create a recordset that combines the input Event table (the one with a Direction field) with the Intersection Marker table. The join should be on the LRMkey(s) and the CrossStreet fields.
2. Use the Functional Attributes command in GeoMedia Professional to add an Azimuth field. If the Direction is N, the Azimuth is 0, for E it's 90, for S it's 180, and for W it's 270.
3. Use the Functional Attributes command in GeoMedia Professional to create an Offset field from the Distance field. If the absolute value of the Azimuth field minus the AzimuthMinus field is less than the absolute value of the Azimuth field minus the AzimuthPlus field, then multiply the Distance field by –1 to get the Offset field. Otherwise, the Offset field equals the Distance field. If the Azimuth field's value is 0, also check using an Azimuth value of 360. Do not consider AzimuthPlus or AzimuthMinus values of –1.
4. Use the GeoMedia Professional Output to Feature Class command to save the results of the last step to a table.
5. Use the Feature Class Definition command in GeoMedia Professional to eliminate unneeded fields, and rename the ones you wish to keep as desired. The fields that need to be kept are those shown in the Sample Event table above plus any attributes used to describe the event. The resultant Event table can be used directly with the GeoMedia Transportation products as a marker offset type event.

Handling Duplicate Intersections

Lastly, we need to address how to handle duplicate intersections (e.g., Main St. loops around and crosses First St. twice). Our recommendation is to add one more step to the population of the Intersection marker table. Search for duplicate CrossStreet fields that have the same LRMkey(s), and sequence them by adding a 1, 2, and so on to the end of the CrossStreet field. In operation, all events for these intersections will fail in dynamic segmentation or event overlay because the marker name will not match exactly. The user can quickly locate the events with the corresponding status message and manually pick which Intersection Marker they want to use. Because the Dynamic Segmentation process makes use of a pipe, as soon as the marker name is corrected, the event geometry will be properly created.

5.13 Metadata

An important part of any database system is the metadata, the data about the data. This type of data may include the source of the data, the units of any numeric fields, the accuracy of the data, and the geographic projection. It is also good practice to enter into the table description the names of any other tables that the table in question is dependent upon. For instance, the description of LRM Event tables should tell what Datum table it is dependent upon. The description of Event and External Marker tables should tell what LRM Event table they are dependent upon.



The simplest approach (pictured above) for entering metadata is to use the Feature Class Definition command in GeoMedia Professional to edit in the metadata as needed. For a more rigorous approach to metadata, and for compliance with the Federal Geographic Data Committee (FGDC) metadata standard, there is SMMS for GeoMedia. This product is an add-on to the GeoMedia family that not only provides for metadata editing, but has metadata search and analysis tools as well.

6. Future Development

This document is the second version of the GeoTrans transportation data model. We have plans to expand it further to provide support for even more of the needs of the transportation information community. Among the capabilities currently planned are as follows:

- Support for a unified model that handles both linear referencing and routing
- Support for multi-modal navigable networks
- Support for multiple cartographic representations of roads

Appendix A – References

1. Adams, T.M. et al, 2001, "NCHRP Report 460 – Guidelines for the Implementation of Multimodal Transportation Location Referencing Systems"

2. CEN Technical Committee 278, 1995, "Comite Europeen de Normalisation (CEN) Geographic Data Files (GDF), European Standard"

3. Curtin, K. et al, 2001, "ArcGIS Transportation Data Model (Draft)"

4. Dueker, K.J. and Butler, J.A., 1997, "GIS-T enterprise data model with suggested implementation choices"

5. Ground Transportation Subcommittee – Federal Geographic Data Committee, 2000, "NSDI Framework Transportation Identification Standard – Public Review Draft"

6. GIS/Trans, Ltd., 1998, "Federal Highways Administration Linear Referencing Practitioners Handbook"

7. Hardy, P.H., 2001, "GeoMedia Transportation Data Structures"

8. Miller, H.J. and Shaw, S.L., 2001, "Geographic Information Systems for Transportation: Principles and Applications"

9. Vanderohe, A.P. et al, 1997, "A Generic Data Model for Linear Referencing Systems"

Appendix B – Glossary

There are a variety of terms commonly used when writing about transportation information systems. To ensure clear communication throughout this document, we have listed below the most common of these terms and the meaning intended for these terms as they are used in this document. Many of these definitions are derived from the sources listed in Appendix A – References.

- Database Views – This is the capability provided for in many database management systems to create a virtual table from two or more joined tables. These are very handy to use when good database design practice suggests the use of multiple tables, but the application software being used is looking for just one. For joining across multiple data sources that are not contained within the same database, the GeoMedia Join capability will provide the same functionality.

- Dynamic Segmentation – This is the process of geocoding Event data.

- Event Data – This is a term commonly used to describe linearly referenced tabular data. The linear references may refer to either points or linear segments. Linear segments require two linear references, one for the beginning of the segment and one for the ending.

- Event Registration – This is the process, described in Section 5.1, *Event Registration*, for linking events to the individual segments of the LRM that they belong to in order to promote location stability.

- Geocoding – This is the process of locating tabular data onto a map based on the attributes in that tabular data. An example is taking a table of longitude/latitude pairs and placing points on a map for each pair.

- Geometry – This is the term used to describe the cartographic representation of data within a GIS system. The geometry can be in various forms, such as points, linestrings, and shapes.

- Keyset – These are the set of one to four fields that, together, are used to store transportation network descriptors, such as road names or bus route names. Both the LRS Datum and the LRM Event tables have keysets.

- Linear Reference – This is textual description by which information can be related to a specific position along a linear network. An example would be State Route 66 at milepost 44.4.

- Linear Referencing Method – This is the technique used to identify a specific point or segment of a roadway (or other linear network). This is accomplished using a combination of a linear naming convention and a linear measurement convention. An example of a linear naming convention is identifying a road by a combination of County name and Route number. An example of a linear measurement convention is to measure linear elements in miles starting at 0 at each county boundary. Another example is street addresses where each segment of road is assigned a range of addresses. It is very common for one organization to use multiple LRMs, such as County Milepost, Street Addresses, and Engineering Stationing.

- Linear Referencing System – This is the term for a system used to determine and retain a record of specific points or segments along a roadway (or other linear network). The system consists of one or more LRMs and the procedures for storing, maintaining, and retrieving location information about points or segments along roadways (or other linear networks).

- LRM – Linear Referencing Method.

- LRM Event Table – This is the method used in the GeoTrans data model to model an LRM. It is an event table that holds both the naming convention and the measurement convention of an LRM. It has no geometry itself, but rather is referenced, via dynamic segmentation, to geometry that resides in the LRS Datum (see Section 3.3, *Multiple LRM Support*).

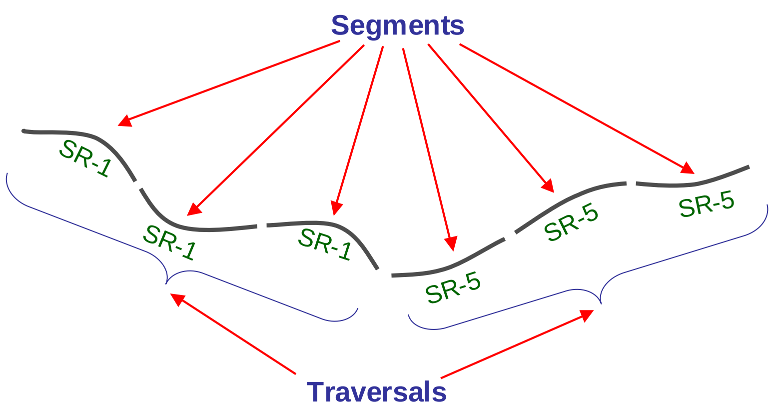
- LRS – Linear Referencing System.

- LRS Datum – This is the portion of the transportation information system that contains the underlying roadway geometry and the base naming and measurement conventions. This underlying geometry is referenced to a geographic datum that defines its location on the globe.

- Markers – These are known points with known measures that are each associated with a particular traversal. These points can be referred to in order to create a linear reference (e.g., this event occurred 200 feet past Marker XYZ). The GeoMedia Transportation products support two types of markers: internal and external. Internal markers are names given to the beginning points (and optionally, the end points) of segments. External markers are independent points that can appear anywhere along a traversal.

- Measures – These are the attributes given to the beginning and ending point of segments that allow the dynamic segmentation process to geocode linearly referenced data via interpolation.

- Segment – A segment is one record in either an LRS Datum or an LRM Event table. It represents a single linear piece of geometry in the GIS.



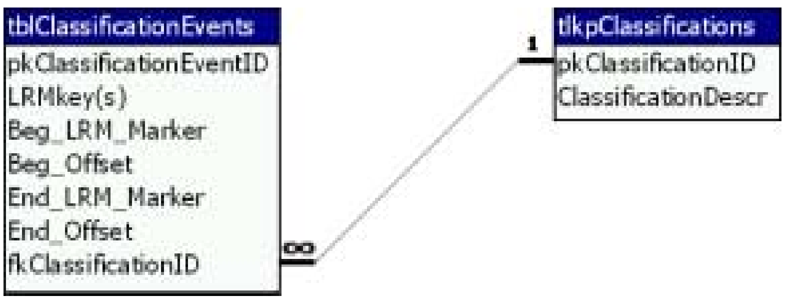
- Traversal – These are a series of segments that are usually connected, usually have sequential measurement spans, and always share the same Keyset values. Both the LRS Datum and LRMs can have traversals.

Appendix C – Sample Event Tables

As an aid to the reader, this section displays a suggested structure for two of the more common event tables used in transportation information systems.

Roadway Classification

Almost every transportation information system has data concerning roadway classification. Many systems carry this information on the datum layer. We, however, recommend that this data be carried in an event table because classification is just another attribute of the roadway system. For analyses in which one would like to see the classification data juxtaposed against other event data, the user can employ the Event Overlay commands of the GeoMedia Transportation products. More than likely, the classification types will be in a lookup table. Either a database view or a GeoMedia Join can be utilized to see these two tables as if they were one virtual table.



Videologs

Videologging has become an extremely important and cost-saving tool for transportation professionals. The table below provides a suggested way of modeling videolog frames.

Videologs	
Field	**Description**
pkVideoFrameID	This is a unique, non-meaningful ID for a given videolog frame.
LRMkey(s)	These are the field(s) that together uniquely identify which LRM traversal a videolog frame is being mapped to. These fields should correlate with the LRMkey field(s) in the LRM Event table to which a specific videolog event table is associated.
LRM_Marker	This correlates to the Marker_Name of the LRM segment that a videolog frame is being mapped to.
Offset	This is the offset from the marker identified in the LRM_Marker field where a videolog frame was captured.
Date	This is the date on which a videolog frame was captured.
FrameDir	This is the directory that contains the graphic file for a videolog frame. The directory is stored separately from the filename to ease database updates if videologs are moved to another disk location.
FrameFileName	This is the filename of the graphic file for a videolog frame.

Videologs	
Field	**Description**
Direction	This field will have a value of +1 if the frame was shot while driving in the direction of increasing LRM measures or –1 if shot in the direction of decreasing measures.
PrevFrameID	This is the record ID of the previous videolog frame in the sequence. This facilitates the ability to backup through the traversal.
NextFrameID	This is the record ID of the next videolog frame in the sequence. This facilitates the display of a "drive through" of the traversal.

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).

GeoTrans Transportation Data Model
10/05