

Managing Linear Reference Model Traversals with Network Analyst

ESRI User Conference 2006

Kevin Hunt
Brad Hibner

Table of Contents

Abstract	3
Overview	3
Case Study: New York State Department of Transportation	4
Illustrations	6
Technical Implementation	9
Input Datasets.....	9
Network Dataset Generation.....	9
Traversal Point Development	10
Traversal Point Editing	10
Final Route Association	11
Apply RouteID(s) to Parent Geometry	12
Create PolylineM with Multiple Route Identification Attributes	12
Calibrate Routes.....	12
Conclusion	13
Acknowledgements.....	13
Author Information	13

Abstract

Managing linear reference model traversal with Network Analyst describes methods for utilizing existing mile point locations to generate specific routes on a state wide scale. The paper describes source data manipulation, network dataset configuration and tuning, transfer of existing mile point locations to updated versions of source data, and developing scripts to automatically and semi-automatically generating routes while providing direct ID linkage to source data.

Overview

The data and functional requirements necessary for state department of transportation organizations (DOTs) are steadily increasing. In addition to current duties, there is increasing need for accident location, real time mapping, and more integration with other organizations to map business information to a common accurate base map.

Adopting a commercial base map may allow increased data accuracy, and functionality, but historically it has been very difficult to initially “seed” the database with linear referencing or “asset” attribution (SectionID, RouteID, FuncClass, etc). To complicate this further, the underlying street basemap is changing with each data release, making it problematic to keep linear reference models in sync with the commercial basemap.

Traditionally, linear referencing models (LRMs) have been constructed from beginning, intermediate, and ending point locations that define a “field locatable” location and specific measurement defining the travel distance between points. These locations are generally stored/maintained in proprietary database(s) for use with other business information and are usually not directly mapped to a “real world” basemap.

In order to take advantage of a commercial basemap and utilize business information and logic, some early adopting DOTs have begun to map linear attributes to the basemap and/or reposition the location points on the commercial basemap.* While this has been partially successful utilizing highway attribution, conflation, and manual editing - the ability to maintain the routes had been problematic and cumbersome requiring additional conflation and manual editing to transfer information to the latest version of the commercial basemap.

Tele Atlas addresses this problem by utilizing LRM points, Dynamap Transportation data, and ESRI Network Analyst routing software to generate the routes. Once a DOT has initially defined the route beginning, intermediate, and ending locations these points are utilized with network routing software to automatically generate the route while also determining specific segments that participate in the route. These LRM locations and resulting routes can be easily and accurately persisted between versions of Dynamap Transportation data.

**Before embarking on mapping LRM attribution to the basemap, it is necessary to ensure that the basemap is spatially accurate. This may require that the commercial basemap provider and DOT work together to realign features to an agreed upon spatial accuracy.*

Case Study: New York State Department of Transportation

In 2002 New York State (NYS) entered into a major data build project with Tele Atlas (formerly Geographic Data Technology) to support an Accident Location and Information System (ALIS). Tele Atlas repositioned segments to recent orthorectified imagery and performed attribute updates supporting addressing and naming enhancements to the data. The final database was delivered in December 2004 providing NYS with a spatially accurate statewide base map.

New York State Department of Transportation (NYDOT) utilized the initial December 2004 data delivery as the baseline database to apply their linear referencing models (LRM). Through conflation and extensive manual editing NYDOT built a polylineM route system and associated milepoint and reference marker point datasets. This route system has been in use supporting DOT business activities for the last two years. However, because of on-going database maintenance and updates to street geometry from data maintenance initiatives the current LRM has become out of sync with the baseline database.

Throughout 2005 Tele Atlas and NYS have explored numerous street conflation based methods to transfer the current LRM to the updated database. While these have been somewhat successful, transferring information for over 140,000 segments still required extensive manual editing to account for subtle differences in the baseline database and the LRM.

Example LRM Reference Points

Early in 2006 Tele Atlas began exploring the use of LRM reference points to transfer the LRM. Since the reference points are generally located on a “field locatable” object such as a street intersection, locating the corresponding position in the updated baseline database was much more accurate and only had to account for only 6500 reference points instead of the 140,000 street segments, making any manual fallout much easier to deal with.



Now that the reference points could be transfer **Routing Through Bi-Directional Segments** on utilizing network route technology provided in Network Analyst 9.1 to automatically generate the routes. Initial testing proved very successful and an application was developed to automatically generate a route for each set of reference markers for a giving route identifier. Creating the routes via Network Analyst eliminated the problems with conflation – by correctly routing through double digitized segmentation, complex ramping structures, and providing primary and reverse route directionality



Routing Through Ramp Structures



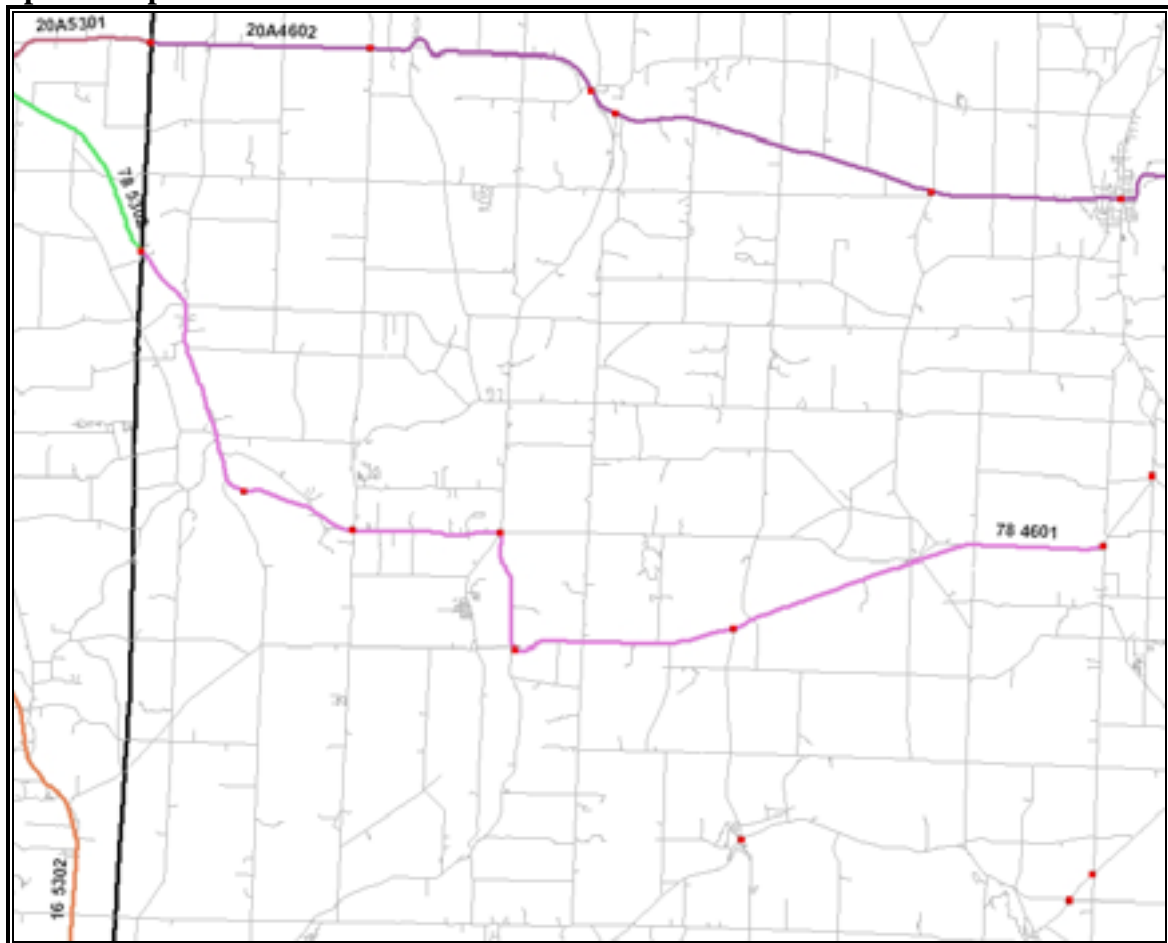
The output from the route generation application; Dynamap ID and associated route identifier could then be joined to the baseline street layer enabling generation of a polylineM route feature layer. The resultant polylineM route layer can then be calibrated from the reference points that were initially used to create the routes.

Illustrations

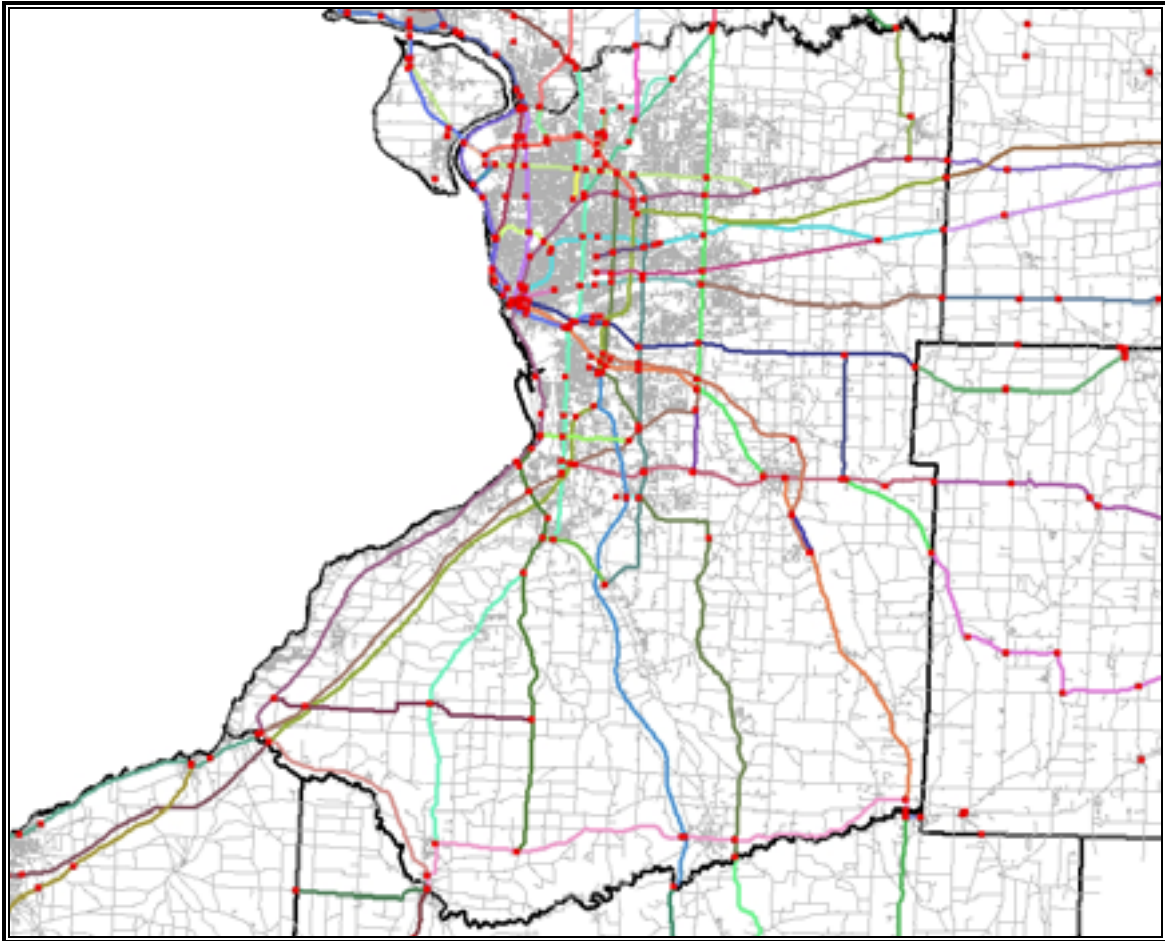
Traversal points for Route "78 4601"

FEI	Shape	ROUTE	MP MEAS	DESCRIPTION	PC	width	width	width	width	width	width	width	width	width	width	width
8079 Point M	78 4601	0	START RTE		40	0	0	0	0	0	0	0	0	0	0	0
2040 Point M	78 4601	0.87	RT 362 W OF 326TH LCP		40	0	0	0	0	0	0	0	0	0	0	0
8480 Point M	78 4601	1			40	0	0	0	0	0	0	0	0	0	0	0
1757 Point M	78 4601	12.67	STRT 36 OLAF W OF 3RVA		40	0	0	0	0	0	0	0	0	0	0	0
1756 Point M	78 4601	15.18	END 36 OLAF JAVA CENTER		40	0	0	0	0	0	0	0	0	0	0	0
8480 Point M	78 4601	15.8			40	0	0	0	0	0	0	0	0	0	0	0
8079 Point M	78 4601	22.25	END RTE		40	0	0	0	0	0	0	0	0	0	0	0

Spatial Representation of Route "78 4601"



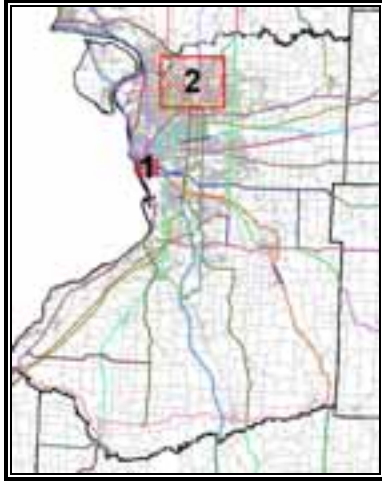
Overview of Routes for Erie and Surrounding Counties in New York



Red points indicate point traversal whereas the colored lines indicate the specific routes.

Selected Detailed Views

Overview



Zoom Area 1



Zoom Area 2



Technical Implementation

Input Datasets

Dynamap Transportation Street layer

Dynamap Transportation Maneuver layer

Specific Route Information

- Textual description of route beginning/ending with specific measurements
- Spatial representation of routes

Network Dataset Generation

- Configure street layer to support basic (non-hierarchical) routing
 - NYS Custom *StreetSegment* layer exported from database to shapefile
 - Note – NYS *StreetSegment* layer contains no alternate names or alternate segments, if using standard Dynamap Transportation shapefile, remove alternate name segments ($f_zlev = -9$)
 - Re-calculate FT_COST / TF_COST in order to generate shortest path non-hierarchy routing
 - Note – The NYS *StreetSegment* layer is projected calculate $FT/TF_Cost = shape_len$
 - Check to make certain F_ZLEV/T_ZLEV are integers (short)
 - Create two new fields fzlev, tzlev
 - Calculate from $fzlev = f_zlev / tzlev = t_zlev$
- Add spatial index to both street/maneuver, and attribute index on DYNAMAP_ID on street layer
- Generate turn featureclass from point maneuvers, via custom code supplied by ESRI
 - Export subset of maneuvers
 - Select $MAN_TYPE = 'I'$ or $MAN_TYPE = 'P'$
 - na_mv_to_turn.mxd
 - Modify the input paths in module *CreateTurnFeatureClassFromMultiEdgeTurnTable*
 - sFolderPath = "C:\hibner\projects\lrs_traversals\data\ny\"
 - sTurnFCName = "nyxxxxtn"
 - sStreetsFCName = "nyxxxxst"
 - sTurnTableName = "nyxxxxmn"
 - Run: *CreateTurnFeatureClassFromMultiEdgeTurnTable*
- Build network dataset via ArcCatalog
 - Connectivity (F_ZLEV / T_ZLEV)
 - Turns (Global / Turns)
 - Cost (FT_COST / TF_COST / Miles)
 - NO Directions
 - NOTE – Network will build with some errors (internal turns)

- *Source ID: 2, Object ID: 578, The edges of the turn element conflict with existing interior/exterior edges.*

Traversal Point Development

The traversal points are used to determine the Dynamap segments that participate in the route as well as calibrate the resulting polylineM route features. The routes are determined from the traversal points by a custom application that utilizes the network dataset created from the baseline street network. Traversal points of a common route identifier are input and sorted by the accumulation mileage, the route is generated and the results are output a single text file of Dynamap_IDs and associated route identifier. The code supports individual and batch route generation.

Re-Positioning Existing Milepoints

Existing milepoint information can be formatted/enhanced to generate routes. In the case of NYS – the existing milepoint layer will have points adjusted to the associated node location in the street segment and add additional points at some beginning/ending segment midpoints to facilitate correct routing.

- Update position of existing milepoints with custom conflation program
 - C:\Hibner\projects\newyork\code\ny_routes\nyroute_conflation.mxd
- Added traversal point fields: (position, direction, supflag)

POSITION	I	10	N	Point position indicator 0= Node 1= Segment midpoint 2= True position on node, but not utilized for route generation
DIRECTION	I	10	N	Direction of route 0=Primary 1=Reverse
SUPFLAG	I	10	N	Supplemental traversal points 0= False 1= True

Traversal Point Editing

MXD Setup

- Enable Editor, Network Analyst Toolbar
- Load the following layers, with the following order in the table of contents
 - Milepoint calibration points
 - Streetsegment
 - Milepoint routes
 - Network Dataset
- Set snapping environment – snap points to nodes/segments of street layer

Point Editing

1. Select route
2. Run code (*SolveRoutes*)
3. Verify route
4. Modify stops as necessary
 - Stop in incorrect position – move to correct position
 - Section of route missed - add stop to guide the route
 - Add point, set routeid, set relative MP_MEAS, set position, and supflag = 1
 - Incorrect route at intersections (usually if route is doing something weird it is because the stop is on a node with an elevated intersection)
 - Remove stop from processing
 - Set position = 2
 - Remove stop from processing and add supplemental stop on segment.
5. Go to step 2
6. Save edits

Note –Traversal points placed at segment midpoint

Place traversal points at nodes wherever possible. Because of an existing limitation of Network Analyst*, points may have to be placed at the midpoint of the segment. These points should be flagged with position = 1. If the point is moved to a midpoint and that point includes a measure, a second point will be added at the true location and flagged with position = 2.

* When a stop is located on a junction, the stop is associated with the nearest segment which may be in the opposite direction causing an incorrect route to be produced. If the solver is changed to associate the stop with junction, and the junction has multiple edges, one edge will be randomly chosen by network analyst potentially generating an incorrect route.

Final Route Association

- Select all traversal points
- Generate Routes for all traversal points
 - Primary Direction (*SolveRoutes*)
 - Reverse Direction (*SolveRoutes*)

Apply RouteID(s) to Parent Geometry

In order to support overlapping routes (A segment is utilized in more than one route). A specific program has been developed to load the RouteID/DynamapID association file to specific route attributes.

- 8 fields are added to the parent feature class (<>_st.shp)
 - Run (*AddRouteIDFields*)
 - RouteID1, RouteID2, RouteID3, RouteID4
 - RouteID1r, RouteID2r, RouteID3r, RouteID4r
- A program reads the route association text file and populates the appropriate RouteID(s) field
 - Create index on DynamapID with ArcCatalog
 - Run (*LoadRouteIDFile*)

Create PolylineM with Multiple Route Identification Attributes

Standard ESRI tools do not support the creation of polylineM features from more than one route identification fields. In order to support multiple routes, a program was written to create a polylineM feature for each unique RouteID that is potentially contained in one of the 4 primary RouteID and 4 reverse RouteID attribute fields.

- Create PolylineM shapefile (*CreatePolylineMShapefile*)
 - → <>_PLM.shp
- Create a polylineM feature for each unique Route ID (*AddPolylineMFeatures*)

Calibrate Routes

- Export calibrated traversal points
 - Select stops where $MP_MEAS > -1$ and $direction = 0$ and $supflag = 0$
 - Export → <>cal_p.shp
- Calibrate PLM via Toolbox, make certain to **uncheck** the extrapolate checkboxes.

Conclusion

Utilizing Network Analyst to associate route identifiers to a commercial street database was very successful. Transferring identifiers via routing software eliminated potential miscoding while also enhanced the efficiency of transferring the data. While there are a few nuances generated by the Network Analysis solver, these could be overcome by simple traversal point modifications. In conclusion, combining routing technology, a commercial basemap, and traversal points allowed the transfer of route identifiers supporting the development of a specific polylineM route layer for NYS DOT.

Acknowledgements

Alan Hatakeyama, ESRI

Author Information

Kevin Hunt

GIS Section
NYS Department of Transportation
(518) 485-7152
Fax: (518) 485-5766
khunt@dot.state.ny.us

Brad Hibner

Software Engineer
Tele Atlas North America
11 Lafayette Street
Lebanon, NH 03766-1445
800-331-7881 x3079
brad.hibner@teatlas.com