

## System

# What Is a Shape Object?

CAD, topological vectors, and TINs are common structures for spatial data with associated attributes stored in a linked or internal database. Each of these spatial structures has advantages in specific geospatial applications requiring geometric data. All can be directly used in the TNT products as linked layers or internal spatial objects.

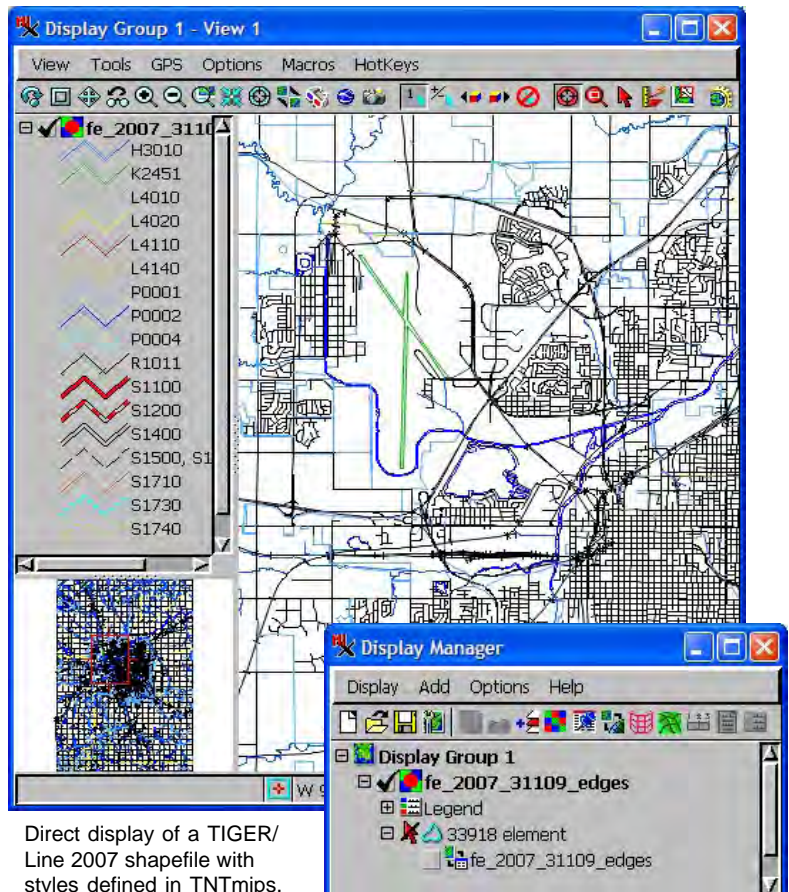
Over recent years database systems have been extended to ingest and integrate both geometric and raster components into their relational table structures. Each popular database system has its independent approach to storing geometric elements and rasters as specialized fields in tables. Oracle uses Oracle Spatial, PostgreSQL uses PostGIS, MySQL and DB2 use Spatial Extensions, SQL Server 2008 provides spatial support, and so on. There is enough similarity in how these structures store spatial components and their georeference that the geodata layers they can define can be treated as shape objects in TNTmips.

Database spatial structures can be imported by TNTmips, TNTedit, and TNTview into vector or CAD objects. However, since they are often being maintained and used as databases, they are subject to editing, analysis, and change by their creator, manager, or the public using other software products. To accommodate this, and the simple topology they maintain, the TNT products only link to these relational database structures. When a link is made to define a shape object, indexes are built in the link to provide for rapid access to geometric components and associated attributes. The shape objects defined by these links can be used in the TNT processes just like the other geometric object types. They can be viewed as layers and analyzed and edited to alter the linked relational tables. If the content of the linked tables in the external database system is changed by the database product or other software, the TNT process is aware of this and appropriate action is taken, such as rebuilding the link. For this reason, and because TNT links to these structures are rapidly built or rebuilt, there is no imported internal version of a shape object.

A spatial layer in a database is defined by relational database tables including a “spatial table” of geometric elements. Only minimum topology is maintained in this relational structure and, thus, in the linked TNT spatial object: polygons cannot cross themselves but polygon islands are kept track of (an island is a polygon completely inside of and not touching another polygon). There is only one element type per shape object: points, multipoints, lines/multiline, or polygons/multipolygons. Multipoints, multilines, and multipolygons are single elements made up of more than one point, line, or polygon, respectively, and these elements may be spatially separated (e.g., multiple spatially separated polygons). If your analysis or application requires a more complex level of topology, then import the shape object into a vector object. Import will create the selected network, planar, or polygonal (full) topology but disconnects this spatial data from any subsequent alterations of it in its original relation table structure and database system.

The following database, list, and table geometric structures are used as shape objects in TNTmips. Shapefiles and LIDAR LAS points are automatically linked and directly accessed as shape objects. Links to Oracle Spatial, PostGIS, ESRI Personal Geodatabase, and MySQL with Spatial Extender as shape objects require user input and are established in the Import process.

For more information on the use of shape objects in the TNT products see the following Technical Guides: *Direct Display of Shapefiles/Legends/Styles*, *Import: Linking to PostGIS Spatial Layers*, *Import: Linking to MySQL Spatial Layers*, and *Import: Linking to Personal Geodatabases*.



Direct display of a TIGER/Line 2007 shapefile with styles defined in TNTmips.