

# Export Geometric Tilesets

The Export Geometric Tileset process in TNTmips Pro (Tileset / Geometry / Export) converts a styled vector object into a geometric tileset with your choice of the SVG, SVGz, or KML geometric tileset structure. These tiled structures allow you to present styled geometric data on the web for any size area and at different resolutions over a range of zoom levels. (See the *Tileset Technical Guides* entitled *SVG Geometric Structure* and *Geometric KML Structure* for descriptions and illustrations of these products.) Modern web browsers support the display of the geometric data in these tilesets and automatically apply enhancements such as anti-aliasing to provide a high-quality rendering of your map data.

The Export Geometric Tileset process renders styled vector elements (polygons, lines, points, and labels) into the output tileset. The styles created in the tileset depend upon the complexity of the input vector element styles and the capabilities of the output SVG or KML graphics format. SVG geometric tilesets allow a more faithful representation of complex point and line styles, including CartoScript symbology, than KML tilesets. See the *Tileset TechGuide* entitled *Styling Map Elements for Geometric Structures* for more information.

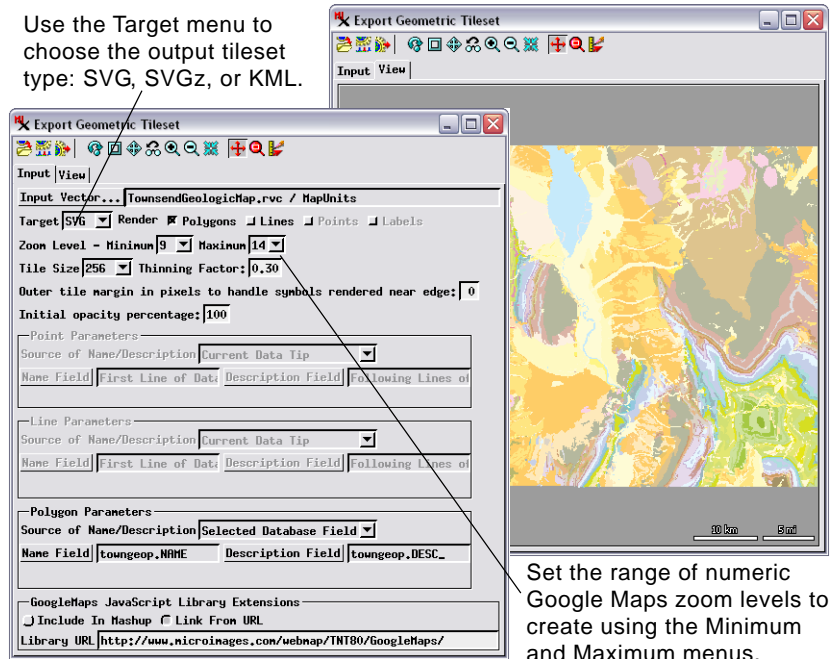
You can choose which element type or types to convert and choose the range of Google Maps zoom levels to be created in the tileset. The process independently renders the tiles at each zoom level from the source vector object starting with the maximum zoom level. Line elements in the source vector object are progressively simplified (thinned) prior to creating each successively lower zoom level to provide a level of map detail that is appropriate for viewing each level. You can also specify the source of the name/description information that will be shown in Info Windows when the user clicks on the map elements while viewing the tileset in a web browser.

## General Process Settings

The Export Geometric Tileset process uses the display style and element selection parameters you have previously set for the input vector object as the starting point to control rendering the vector elements to a tileset. You can further constrain the tileset rendering and set other process parameters using the controls on the Input tabbed panel.

The Target menu offers the choice of SVG, SVGZ (compressed SVG), or KML format for the output tileset. Render checkboxes are provided for Polygons, Lines, Points, and Labels; an individual checkbox is active if that element type is present in the input vector and selected for display. Use the active Render checkboxes to indicate which available vector element types to transfer to the tileset. Label elements are rendered as text to both SVG and KML tilesets. Dynamic labels set in the source vector object for polygon, line, or point elements are rendered

Use the Target menu to choose the output tileset type: SVG, SVGz, or KML.



Set the range of numeric Google Maps zoom levels to create using the Minimum and Maximum menus.

as text to SVG tilesets but are ignored when creating a KML geometric tileset.

The Minimum and Maximum zoom level menus let you set the range of Google Maps zoom levels to create in the tileset. To help determine an appropriate range of zoom levels, you can use the Show Zoom pane in any TNT View window sidebar (including the View panel in the Export Geometric Tileset window) to preview your input vector. In this pane you can set the zoom method to Google Maps and pick any zoom level from the Level menu (see the TechGuide entitled *Spatial Display: Show Zoom Pane*). In most cases it is good practice to set a maximum zoom level such that the corresponding map scale is close to that at which the original map data were created. Choose a minimum zoom level that avoids overcrowding the map elements, which may produce an uninformative and unattractive map and which can also produce very large tile files for the lower zoom levels, resulting in slow display of the tileset in the browser.

You can run the Export Geometric Tilesets process as a single process using the Run icon button or as a group of jobs under the TNTmips Job Processing system by pressing the Save Jobs or Queue Jobs icon buttons. The latter option can provide significantly faster processing, as you can set job processing to run several jobs simultaneously to take advantage of your computer's multiple cores. When you name the output TileSet Definition (TSD) file for an SVG tileset or the output master KML file for a KML tileset, an enclosing directory with the same name is automatically created to contain all of the contents of the output tileset.

Each new geometric tileset is automatically provided with an HTML/JavaScript file that references the tileset and enables it

(over)

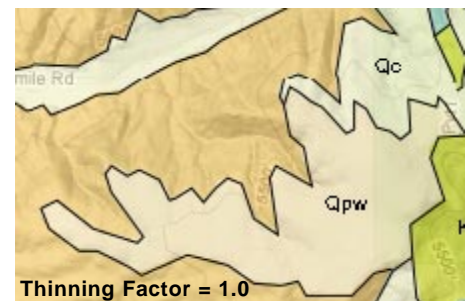
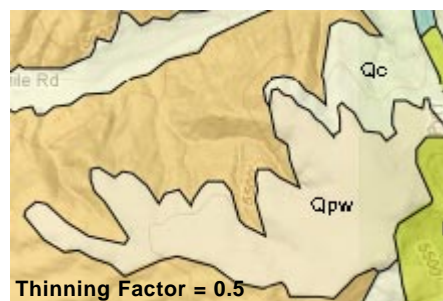
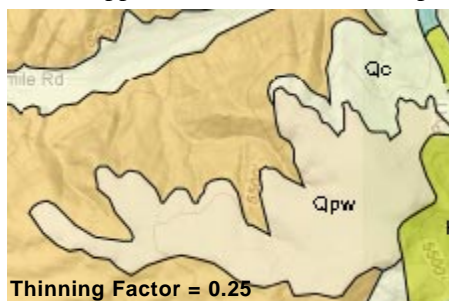
to be immediately viewed in Google Maps. The *Google Maps JavaScript Library Extensions* settings specify the location of the JavaScript library to be used by this file. You can choose to save a copy of the library with the tileset, or choose to link to the library posted on your website, in which case you must enter the full web address of the library directory in the Library URL field.

### Choice of Tile Sizes

The standard size for tiles to be used in web geomashups is 256 by 256 pixels. But the Export Geometric Tileset process allows you to create geometric tilesets with larger tile sizes, such as 512 by 512 or 1024 by 1024 pixels. For tilesets containing relatively simple geometric data, using a larger tile size reduces the number of tiles that the browser has to manage in a particular view and thus may result in faster and more efficient display of the tileset. The option for larger tile sizes is most useful for SVG geometric tilesets, in which the geometric data is stored compactly and is interpreted directly by the web browser. MicroImages does not recommend using a larger tile size for KML geometric tilesets or for SVG tilesets with extremely complex geometric data, as the larger-sized tile files may require excessive download and loading times.

### Thinning Factor

To produce tileset zoom levels with differing amounts of detail, the input vector object is automatically simplified prior to rendering the tiles at each level (except for the maximum zoom). Lines are simplified (thinned) by selectively removing vertices that cause the line to deviate from a locally straight trend. A nominal thinning distance (line deviation threshold) is set automatically for each output zoom level based on the ground dimensions of a screen pixel at that Google Maps level. You can modify the thinning distance and the amount of line thinning actually applied by varying the value of the Thinning Factor setting. This value can range from 0.1 (minimum thinning) to 1.0 (maximum thinning). The value to set for the Thinning Factor depends on the complexity, curvature, and vertex density of the input vector lines and the type of output tileset you are creating. Lines that have been overly-thinned for a particular zoom level will exhibit obvious straight line-segments separated by sharp angles. Reducing the Thinning Factor can retain smoothness in curving lines at all zoom levels. (Examples of different settings for the Thinning Factor are illustrated below.) You may need to run several trials with a particular vector object to produce the desired appearance of lines in the output geometric tileset.



Automatic line thinning example illustrating the visual effect of using different values for the Thinning Factor. Each illustration shows an SVG tileset with filled polygons and an overlying SVG tileset with black lines that form the polygon boundaries; the same Thinning Factor was applied to both tilesets in each example. Tilesets are shown at one less than the maximum zoom level, so one step of line thinning has been applied to the source vector lines. The amount of line thinning increases from left to right in these examples. A smaller value for the Thinning Factor results in less line thinning and smoother lines, but larger tile files. (continued)

Keep in mind that reducing line thinning increases the size of the output geometric tile files, especially at the lower zoom levels where each tile covers a larger map area and thus includes many more graphic elements than at higher zoom levels. Because the tiles in KML geometric tilesets must be interpreted and converted to a different graphic form prior to being displayed in the browser, insufficient line thinning can result in very slow load times for these tilesets. Tiles in SVG tilesets, on the other hand, are stored in a compact format and are read directly by the web browser, so tile size and graphic complexity have much less impact on browser performance. You can therefore use less line thinning and retain greater line complexity without compromising browser performance when creating an SVG tileset; a KML tileset produced from the same input will likely require a higher degree of thinning to produce a tileset with acceptable browser performance.

In some cases the line thinning procedure may encounter a line for which the automated deletion of vertices would produce line segments that cross each other or which intersect another line. Such lines are silently skipped and not thinned, in which case a relatively complex line is rendered unchanged into each tileset zoom level. If you notice such effects, you may wish to use the TNT Spatial Editor to edit the input vector object and manually simplify the problem lines before remaking the geometric tileset.

### Point, Line, and Polygon Parameters

The Point Parameters, Line Parameters, and Polygon Parameters sections of the Input tabbed panel provide controls for setting the Info Window content for the output tileset. Each of these control sections is active if its element type is selected for rendering to the tileset, allowing you to specify the source for text to assign to the *Name* and *Description* attributes of the corresponding elements in the tileset. When the tileset is viewed in a geomashup in a web browser, the Name attribute text for the map element at the mouse cursor location is automatically shown as a floating label next to the cursor. The Name attribute is also used as a bold heading in the Info Window that opens when the user clicks on a map element. The Description attribute text is shown in nonbold text beneath the Name in the Info Window.

The Source of Name/Description menu provides two choices: *Current DataTip* and *Selected Database Field*. Using the *Current DataTip* option, the first line of text of the vector element DataTip is set as the Name of the element, and any subsequent

text lines (if any) are set as the Description. This setting is useful if you have designed a multiline DataTip for the input vector object (see the TechGuide entitled *Geomedia Publishing: DataTip Design for Google Maps/Earth Info Windows* for more information). Alternatively, you can choose Selected Database Field from the Name/Description menu. This choice activates Name Field and Description Field pushbuttons, which you can use to choose separate database table and field sources for the name and description of the elements.

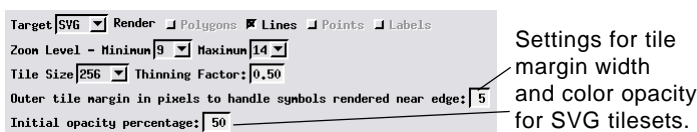
### Additional Settings for SVG Geometric Tilesets

In an SVG geometric tileset, point and line symbols are rendered as vector graphics into the SVG file for each tile. Because each tile area is independently rendered, large point symbols near tile boundaries may be partially cut off at the edge of a tile. In addition, polygons that cross tile boundaries must be stored as closed shapes with boundary segments along tile edges. If the polygons are rendered with borders, spurious tile-edge polygon borders can appear.

To accommodate such style issues, you can choose to create “oversize” SVG tiles in which the area for storing styled elements in each tile is expanded by adding an outer margin, specified in screen pixels. The *Outer tile margin...* field in the Input panel for SVG lets you set the width of this margin, which is not actually displayed in the browser. If the tileset you are making includes polygons with borders, you should set the width of the tile margins to be equal to the width (in screen pixels) you have set for the polygon borders in the parent vector object. The spurious tile-edge borders thus lie entirely within the tile margins. The JavaScript code for displaying the tiles in a geomashup clips the browser rendering of the elements along the nominal tile boundaries, so the tile-edge polygon borders do not appear in the browser view.

If the tileset you are making includes large point symbols, you should set the tile margin width to be as large as the point symbol size. Near-edge point elements are then included in both adjacent oversize tiles. When these tiles are viewed in the browser, the symbols for near-edge points are rendered in both tiles but are clipped at the nominal tile boundary so the visible symbols originate partially in one tile and partially in the adjacent tile, but appear seamless in the browser rendering.

You can also set a common opacity percentage for all colors in the tileset using the *Initial opacity percentage* field. This option is most useful when you are making a tileset from polygons with opaque fill colors, letting you set all fills to be partially transparent in the tileset to reveal underlying basemap features when viewed in a geomashup.



### Additional Settings for KML Geometric Tilesets

When vector points are rendered to a KML geometric tileset, the point symbols are automatically rendered to very small PNG

image files (“icons”) that are automatically linked to the relevant individual KML tile files. The Marker Image Size field in the Point Parameters section of the Input panel for KML lets you set the size (in pixels) of the PNG icon files produced from the source vector point symbols. The KML file format also incorporates an icon scale factor that determines the dimensions on screen of the point icons when the KML tileset is viewed in Google Maps or Google Earth. You can use the Icon Scale numeric field to set a scale factor to adjust the screen size of the KML point symbols, or turn on the *Assign icon scale based on view* toggle to have the scale set automatically based on the vector display settings. One strategy to produce attractive symbols on-screen is to render the TNT symbols to a larger size (24 or 32 pixels, for example) but use an icon scale factor less than 1 to reduce the KML symbols to a smaller screen size.

NOTE: When you create a KML geometric tileset consisting only of points, it is not necessary to create multiple levels. Simply determine the lowest zoom level at which the points can be viewed without crowding and set this level on both the Minimum and Maximum menus. In a geomashup you can view the KML point layer at any higher zoom level without adversely affecting viewing performance.

Lines are stored in KML files with style information that specifies the line color and width in screen pixels. When lines are rendered into KML tiles from the source vector object, the nominal line width is determined automatically from the TNT line style settings. However, you can adjust this width determination in two ways. The Minimum Line Width setting for KML lets you set a minimum screen width in pixels for all lines. You can also set a global Line Scale Factor that is applied when determining the KML line width from the TNT line style. If all lines are too thin or too thick in the output KML geometric tileset, you can adjust the Line Scale Factor and remake the tileset without having to adjust all of the line styles in the source vector object.

Elements in a KML tileset can be set to float at a set height above the terrain in Google Earth, with lines and polygons connected to the terrain by a colored curtain. The Point Parameters, Line Parameters, and Polygon Parameters sections on the Input panel for KML each include an Extrusion Field pushbutton. This pushbutton lets you select an associated database table and numeric field containing a height value to transfer to the KML tileset elements to extrude them above the Google Earth terrain.

