

# Export Geometric Tilesets

The Export Geometric Tileset process in TNTmips Pro (Web GIS / Geometry / Export) converts a styled vector, CAD, or shape object into a geometric web 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 at different levels of detail 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.) Geometric tilesets use Google Maps zoom level numbering and can be used in either Google Maps or Open Layers geomashups. 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 geometric object elements (polygons, lines, points, and/or labels) into the output tileset. The styles created in the tileset depend upon the complexity of the input 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 renders the tiles at each zoom level directly from the source object starting with the maximum (most detailed) zoom level. Line elements in the source 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.

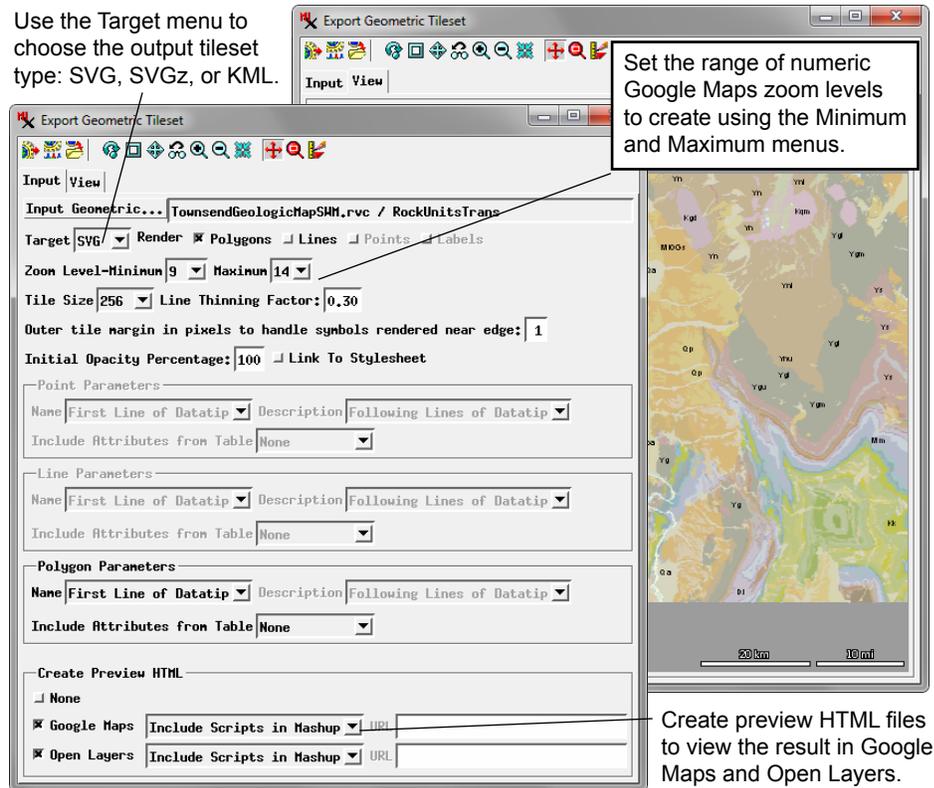
## General Process Settings

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

## Target and Render Options

Use the *Target* menu to choose the format for the output tileset:

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



SVG, SVGZ (compressed SVG), or KML. *Render* toggle buttons are provided to choose the element type to render: Polygons, Lines, Points, and Labels. An individual toggle button is active if that element type is present in the input object and is selected for display. 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 as text to SVG tilesets but are ignored when creating a KML geometric tileset.

## Zoom Levels

The *Minimum* and *Maximum* zoom level menus let you set the range of zoom levels to create in the tileset. Default minimum and maximum levels are set automatically when you select the input object. These default values are based on the map extents of the object, but you can modify these settings. To help determine an appropriate range of zoom levels, you can use the Zoom panel in the sidebar of any TNTgis View window (including the View panel in the Export Geometric Tileset window) to preview your input vector. In this panel you can set the zoom method to Google Maps and pick any zoom level from the *Level* menu (see the TechGuide entitled *Sidebar Zoom Panel*). 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.

(continued)

## 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 choose larger tile sizes, such as 512 by 512 or 1024 by 1024 pixels, from the *Tile Size* menu. 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 tile size larger than 256 by 256 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.

## Line 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 after the maximum zoom level. (Line thinning is not available for CAD and shape input). 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 level. You can modify the thinning distance and the amount of line thinning actually applied at each level by varying the value in the *Line Thinning Factor* field. This value can range from 0.1 (minimum thinning) to 1.0 (maximum thinning). The value to set for the *Line 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.

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

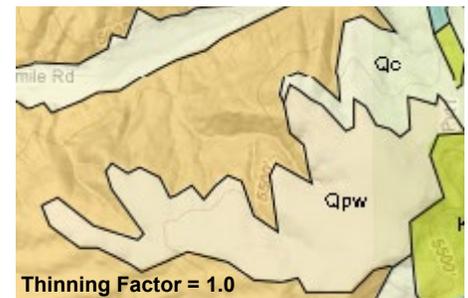
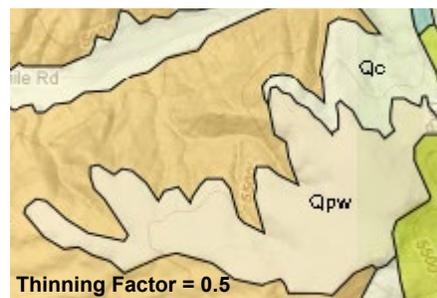
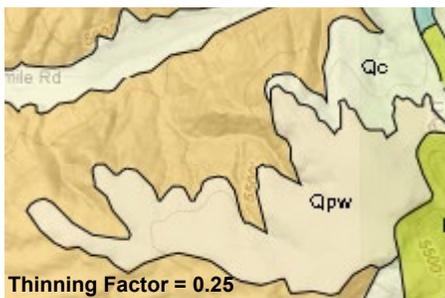
the tiles in KML geometric tilesets must be interpreted and converted to a different graphic form prior to being displayed in the browser, insufficient line thinning can result in very slow load times for these tilesets. Tiles in SVG tilesets, on the other hand, are stored in a compact format and are 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 TNTmips Editor to edit the input vector object and manually simplify the problem lines before remaking the geometric tileset.

## Point, Line, and Polygon Parameters

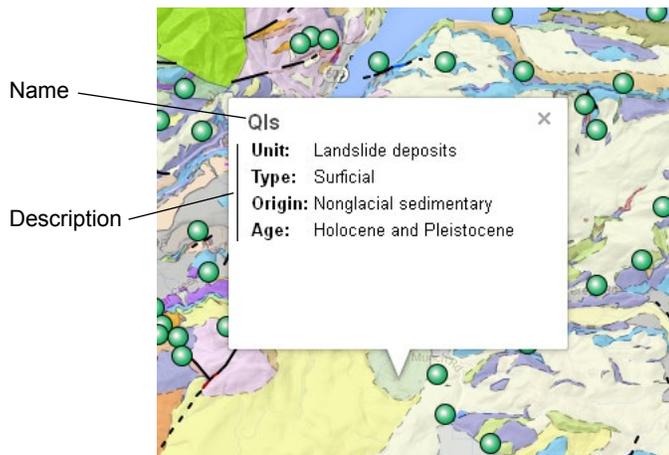
The Point Parameters, Line Parameters, and Polygon Parameters sections of the Input tabbed panel provide controls for setting the Info Window content to be shown in the web browser for the output tileset. Each of these control sections is active if its element type is selected for rendering to the tileset, 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 can be 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.

Separate menus are provided to let you set the source for the Name and Description text. The *Name* menu provides the options *None*, *Select Field*, and *First Line of DataTip*. If you choose *Select Field* you are automatically prompted to choose an associated database table and field to provide the text for the element names. The *First Line of DataTip* option is useful when you



Automatic line thinning example illustrating the visual effect of using different values for the Line 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 Line 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 Line Thinning Factor results in less line thinning and smoother lines, but larger tile files.

(continued)



Browser view of a geometric tileset with polygons, with an Info Window opened by left-clicking on one of the polygons. The source vector object has a multiline DataTip set from a computed text field in the polygon database that concatenates information from various fields in the attribute table. In the Export Geometric Tileset process the Name menu for Polygons was set to *First Line of DataTip* (which sets the Description option to *Following Lines of DataTip*), so the Info Window replicates the information in the TNTgis multiline DataTip.

have set up a multiline DataTip for the geometric elements using a computed text field in the element database. With this option the first line of text of the element DataTip is set as the Name; the *Description* menu is automatically set to *Following Lines of DataTip*, so any subsequent lines in a multiline DataTip are set as the Description (see the illustration above for an example and the TechGuide entitled *Geomedia Publishing: DataTip Design for Google Maps/Earth Info Windows* for more information).

When you have set the *Name* menu to *Select Field* or *None*, the *Description* menu is active and presents the options *None*, *Select Field*, and *Select Table*. The *Select Field* option prompts you choose a single database field to provide the description text. The *Select Table* option prompts you to select a table, and automatically creates a multiline description using all of the fields in the table, showing the field name followed by the field attribute value on each line.

### Preview HTML

Controls in the *Create Preview HTML* box at the bottom of the Input tabbed panel are provided to optionally create an HTML file (web page) to display the completed geometric web tileset. Turn on the toggle button(s) to create a Google Maps and/or Open Layers preview, or None. Each preview web page must reference a corresponding JavaScript library (Google Maps or Open Layers) in order to display the tileset; the menu next to each preview option lets you specify the library location. The default option *Include Scripts in Mashup* makes a copy of the required library in the same directory as the tileset. If you already have an up-to-date version of the library on your website, you can choose the *Link to JavaScript from* option and enter the web address of the library directory in the URL field to the right of the menu.

The preview files are created to let you immediately view the output tileset. To present the tileset on your website, you should use the Assemble Geomashup process to design the appropriate presentation for the tileset.

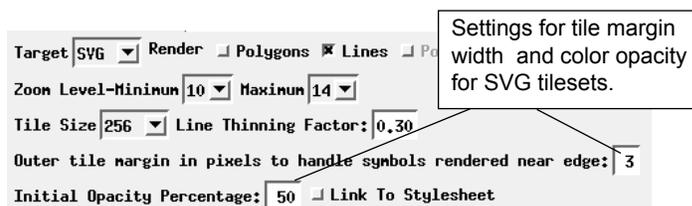
## Settings for SVG Geometric Tilesets

In an SVG (or SVGZ) 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 area, 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. (The Assemble Geomashup process also lets you include a Transparency control for a layer in the geomashup so that the user can interactively vary the transparency of layers.)



Style information (color definitions, line widths, etc.) for rendering geometric elements is stored by default in the individual SVG tile files in the tileset. The same style information may be repeated in many tiles, increasing the stored size of the tiles and possibly affecting loading times in the web browser. If you turn on the *Link to Stylesheet* toggle, element style information is stored separately in a single style file (Cascading Style Sheet

(continued)

or CSS file), and the individual SVG tile files then are created with links to this style sheet file, reducing their stored size and improving tileset performance. Note that for polygons with hatch patterns (not recommended for conversion to tilesets), the hatch patterns may not render properly in some web browsers if the tileset has linked CSS styles rather than embedded styles.

## Settings for KML Geometric Tilesets

### Altitude Settings

KML geometric tilesets can be displayed in 3D in Google Earth, where elevation information included with the elements can be used to control the rendering of the elements. The Point, Line, and Polygon Parameters controls for KML provide an *Altitude* menu with choices *None* and *Select Field*. If you choose *None*, no altitude information is provided for the KML elements, and they are rendered directly on the terrain surface in Google Earth (default Clamp to Ground setting in KML). Choosing the *Select Field* option automatically prompts you to choose a table and field containing altitude values. Selecting a field activates the Mode menu, which contains choices that tell Google Earth how to interpret the altitude values. The choices are *Absolute*, which interprets the values as absolute elevation above sea level, and *Relative to Ground*, which interprets them as heights above the terrain surface. If you turn on the *Extrude* toggle, Google Earth renders the elements with vertical curtains connecting them to the terrain, making the elements appear as if they were extruded upward from the surface. These settings have no effect when a KML tileset is displayed in a web browser with Google Maps or Open Layers.

### Line Settings

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 object, the nominal line width is determined automatically from the TNTgis 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 TNTgis 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.

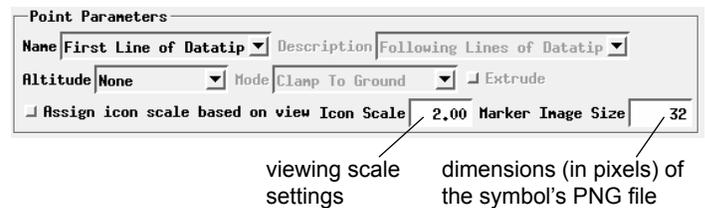


KML line style settings

### Point Settings

When point elements 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 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 source object display settings. One strategy to produce attractive symbols on-screen is to render the TNTgis 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.

KML point symbol settings:



viewing scale settings      dimensions (in pixels) of the symbol's PNG file

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.

## Running the Process

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 (see the TechGuide entitled *TNTmips Job Processing System*). 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.

