

GA World Wind Suite

Layer Definition File Specification

Geospatial Applications and Visualisation (GAV)

GEOSCIENCE AUSTRALIA
TRIM REFERENCE D2010-154358

by

James Navin



© Commonwealth of Australia (Geoscience Australia) 2011

This material is released under the Creative Commons Attribution 3.0 Australia Licence.
<http://creativecommons.org/licenses/by/3.0/au/>

Geoscience Australia has tried to make the information in this product as accurate as possible. However, it does not guarantee that the information is totally accurate or complete. Therefore, you should not solely rely on this information when making a commercial decision.

Contents

Contents	2
Introduction	6
Layer definition files	6
Conventions	6
File names	6
Standard layer definition elements	6
Common elements	6
<Layer>	6
<DisplayName>	7
<LatLon>	7
<Color>	7
<DataCacheName>	8
<Dimension>	8
Tiled Image Layers	8
Example	8
<Legend>	9
<Service>	9
Service types	9
Offline	10
WWTileService	10
OGC:WMS	10
DelegatorTileService	10
<URL>	10
<Delegates>	10
<Delegate>	10
MaskReader	10
ColorLimitTransformer	10
ColorToAlphaTransformer	11
TransparentColorTransformer	11
ResizeTransformer	12
StripingFilterTransformer	12
HttpRetriever	13
PassThroughZipRetriever	13
TextureTile	13
NearestNeighborTile	13
ColorMapReader	13
NormalMapReader	14
ShadedElevationReader	14
LocalRequester	15
URLRequester	15
<LastUpdate>	15
<DatasetName>	15
<ImageFormat>	16
<AvailableImageFormats>	16
<ImageFormat>	16
<FormatSuffix>	16
<NumLevels>	16
<TileOrigin>	17
<LevelZeroTileDelta>	17
<TileSize>	17

Layer Definition File Specification

<Sector>	17
<SouthWest>	18
<NorthEast>	18
<SectorResolutionLimit>	18
<UseTransparentTextures>	18
<ForceLevelZeroLoads>	18
<RetainLevelZeroTiles>	19
<TextureFormat>	19
<UseMipMaps>	19
<SplitScale>	19
<DetailHint>	19
<TransparencyColors>	20
<TimedExpiry>	20
<ColorMap>	20
<Entry>	20
Surface Shapefile Layers	21
Example	21
<URL>	21
<DrawInterior>	21
<DrawOutline>	21
<OutlineWidth>	22
<Antialiasing>	22
<InteriorColor>	22
<OutlineColor>	22
<InteriorOpacity>	23
<OutlineOpacity>	23
<StippleFactor>	23
<StipplePattern>	23
Point Layers	23
Example	24
<PointType>	24
<DataFormat>	25
<URL>	25
<Styles>	25
<Style>	25
<Property>	26
<Attributes>	32
<Attribute>	32
<Text>	32
<Link>	33
<Case>	33
<Regex>	34
<Range>	34
<Points>	34
<Point>	34
KML Layers	35
Example	35
<URL>	35
Curtain Image Layers	35
Example	35
<Service>	36
<Delegates>	36
<LastUpdate>	36
<DatasetName>	36
<DataCacheName>	36
<ImageFormat>	36
<AvailableImageFormats>	36

Layer Definition File Specification

<FormatSuffix>.....	37
<NumLevels>	37
<TileSize>.....	37
<FullSize>.....	37
<Path>.....	37
<CurtainTop>.....	37
<CurtainBottom>	37
<FollowTerrain>.....	37
<Subsegments>	38
<UseTransparentTextures>.....	38
<ForceLevelZeroLoads>.....	38
<RetainLevelZeroTiles>	38
<TextureFormat>	38
<UseMipMaps>	38
<SplitScale>.....	38
Historic Earthquake Layers.....	38
Example	38
<URL>	39
<Coloring>.....	39
<ColoringMinDate>.....	39
<ColoringMaxDate>	39
<PointSize>.....	40
Crust Layers	40
Example	40
<URL>	40
<Size>.....	40
<Scale>	41
<Sector>.....	41
<Wrap>	41
Skysphere Layers	41
Example	42
<TextureLocation>	42
<Rotation>	42
<Segments>.....	42
<Slices>	42
Fog Layers	43
Example	43
<NearFactor>	43
<FarFactor>.....	43
<FogColor>.....	43
Geometry Layers.....	44
Example	44
<URL>	44
<DataFormat>	44
<ShapeType>	45
<RenderType>	45
<Styles>	45
<Attributes>.....	48
References	56

Introduction

This document defines the various layer definition file formats available in the GA World Wind Suite of software, along with a description of what each element in the file is.

Layer definition files

CONVENTIONS

The following are conventions that should be followed when authoring layer definition files for use in the GA World Wind Suite.

File names

- File names should be descriptive and match closely with the `<DisplayName/>` element contained within the definition.
- File names should end with the extension `.xml`

Standard layer definition elements

- All layer definitions must contain the standard xml pre-amble `<?xml version="1.0" encoding="UTF-8"?>`
- All layer definitions must contain the `<Layer>` element as the root element
- All layer definitions must contain only a single `<Layer>` element
- All layer definitions must have the `version` attribute of the root `<Layer>` element set to '1'
- All layer definitions should contain a `<DisplayName/>` name element that provides a concise human-readable name for the layer

COMMON ELEMENTS

<Layer>

The `<Layer>` element is the common root element of all layer definition files.

ATTRIBUTE	DESCRIPTION
version	The version of the layer definition file format being used. Currently, the only valid value is '1'.
layerType	The type of layer defined in the file. Valid values are: <ul style="list-style-type: none"> • TiledImageLayer • SurfaceShapeShapefileLayer • PointLayer • KMLLayer • CurtainImageLayer • HistoricEarthquakesLayer • CrustLayer • SkysphereLayer* • FogLayer* • GeometryLayer

	<p>Each of these types are described in more detail in later sections of this document.</p> <p><i>*Currently only available for use in the Animator tool</i></p>
--	--

<DisplayName>

```
<DisplayName>Magnetics</DisplayName>
```

The <DisplayName> element is used to define a human-readable name for the layer. This name is used within the GA World Wind Suite to allow users to identify layers.

The display name should be concise and meaningful. All layer definition files should define a <DisplayName> element.

<LatLon>

```
<LatLon units="degrees" latitude="-90" longitude="-180" />
```

<LatLon> elements are used within other elements to define a location in latitude and longitude.

These elements should never appear as top-level elements within a layer definition file.

ATTRIBUTE	DESCRIPTION
units	<p>The units that the latitude and longitude are expressed in.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • degrees • radians
latitude	The latitude, expressed in the specified units
longitude	The longitude, expressed in the specified units

<Color>

```
<Color red="255" green="127" blue="55" alpha="240" />
```

<Color> elements are used within other elements to define a colour in the RGBA colour space.

ATTRIBUTE	DESCRIPTION
red	<p>The red channel. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
green	<p>The green channel. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
blue	<p>The blue channel. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
alpha	<p>The alpha channel. Integer in the range [0, 255].</p> <p>A value of 0 will give 100% transparent, while a value of 255 will give 100% opaque.</p> <p>Optional. Defaults to 255.</p>

Layer Definition File Specification

Some example of colours defined using the RGBA space are included below. RGBA colours can easily be generated using a variety of colour picker tools, including the Online Colour Picker ^[4].

RGBA	DESCRIPTION
(255,255,255,255)	White, 100% opaque.
(0,0,0,255)	Black, 100% opaque.
(255,255,0,128)	Yellow, 50% opaque.
(128,0,128,10)	A dull magenta, almost fully transparent.

<DataCacheName>

```
<DataCacheName>GA/Magnetics/Edition5</DataCacheName>
```

Identifies the location to store cached layer information within the WorldWindData folder. The location of this folder is platform-dependent, but in Windows is usually found under <Windows Install Location>\Profiles\All Users\Application Data

Layers for GA data should be stored within the GA/ subdirectory.

<Dimension>

```
<Dimension width="512" height="256" />
```

Specifies a rectangular dimension of width x height (in pixels).

ATTRIBUTE	DESCRIPTION
width	The width in pixels. Integer in the range (0, ∞).
height	The height in pixels. Integer in the range (0, ∞).

TILED IMAGE LAYERS

```
<Layer version="1" layerType = "TiledImageLayer">
```

Tiled Image Layers are the most common form of layer. These layers use tiled images stored on a server, loading the most appropriate tile for the user as they move around in the world window.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="TiledImageLayer">
  <DisplayName>Magnetics</DisplayName>
  <Legend>legend/legend.png</Legend>
  <Service serviceName="DelegatorTileService">
    <URL>http://www.ga.gov.au/apps/world-wind/tiles.jsp</URL>
  </Service>
  <Delegates>
    <Delegate>MaskReader</Delegate>
    <Delegate>PassThroughZipRetriever</Delegate>
  </Delegates>
  <LastUpdate>28 05 2010 00:00:00 GMT</LastUpdate>
  <DatasetName>magnetics/edition5/magnetics</DatasetName>
  <DataCacheName>GA/Magnetics/Edition5</DataCacheName>
  <ImageFormat>application/zip</ImageFormat>
```


Layer Definition File Specification

```
<AvailableImageFormats>
  <ImageFormat>application/zip</ImageFormat>
</AvailableImageFormats>
<FormatSuffix>.zip</FormatSuffix>
<NumLevels count="7" numEmpty="0" />
<TileOrigin>
  <LatLon units="degrees" latitude="-90" longitude="-180" />
</TileOrigin>
<LevelZeroTileDelta>
  <LatLon units="degrees" latitude="36" longitude="36" />
</LevelZeroTileDelta>
<TileSize>
  <Dimension width="512" height="512" />
</TileSize>
<Sector>
  <SouthWest>
    <LatLon units="degrees" latitude="-43.923147541499944" longitude="112.50336544294441" />
  </SouthWest>
  <NorthEast>
    <LatLon units="degrees" latitude="-9.026494833499997" longitude="154.66334857894435" />
  </NorthEast>
</Sector>
<UseTransparentTextures>true</UseTransparentTextures>
<ForceLevelZeroLoads>true</ForceLevelZeroLoads>
<RetainLevelZeroTiles>true</RetainLevelZeroTiles>
<TextureFormat>image/dds</TextureFormat>
<UseMipMaps>true</UseMipMaps>
<SplitScale>0.9</SplitScale>
</Layer>
```

<Legend>

```
<Legend>legend/legend.png</Legend>
```

Optional. Defines the URL of the legend image to use for the layer.

<Service>

```
<Service serviceName="DelegatorTileService">
  <URL>http://www.ga.gov.au/apps/world-wind/tiles.jsp</URL>
</Service>
```

Required. Defines the service type to use for retrieving image tiles, along with the URL to connect to to retrieve the tiles.

ATTRIBUTE	DESCRIPTION
serviceName	The name of the tile service to use when retrieving image tiles for the layer. Valid values are: <ul style="list-style-type: none">• Offline• WWTileService• OGC:WMS• DelegatorTileService More detailed descriptions of these service types are provided below.

Service types

The following is a description of each of the available service types that can be used within a Tiled Image Layer.

Offline

Interprets the provided service <URL> as a local URL when retrieving image tiles.

WWTileService

The standard NASA World Wind tile service. Is able to retrieve tiles from remote locations specified in the provided service <URL>.

OGC:WMS

A tile service that retrieves image tiles from a defined Web Map Service^[2] (WMS service).

DelegatorTileService

This is the most commonly used (and most powerful) tile service in the GA World Wind Suite. It extends on the functionality of the *WWTileService*, but provides access to a growing number of delegates that can be used to perform tile-retrieval control and tile pre-processing tasks.

Unless there is a good reason not to, this should be the default tile service used when authoring new layer definitions.

Note: When the *DelegatorTileService* is used, a <Delegates> element *must* be provided.

<URL>

Required. Defines the URL to be used by the tile service to retrieve image tiles for the layer.

<Delegates>

```
<Delegates>
  <Delegate>MaskReader</Delegate>
  <Delegate>PassThroughZipRetriever</Delegate>
</Delegates>
```

Required if the *DelegatorTileService* is used. Ignored if any other tile service is specified.

Specifies a list of <Delegate> elements that are to be applied to the tile retrieval process. These delegates may control tile retrieval, or perform pre-processing on tile images as they are retrieved from the server.

<Delegate>

Specifies a single delegate for use with the *DelegatorTileService*. Delegates may have parameters passed to them as a comma-separated list of parameter values surrounded by braces.

Currently, valid delegates are:

MaskReader

```
<Delegate>MaskReader</Delegate>
```

Supports loading of image tiles that have a corresponding mask tile. The mask reader creates a composite tile consisting of the colour channels of the input image tile and the alpha channel of the mask tile to output an RGBA tile that is ultimately rendered.

This delegate can find mask tiles from within a zip file, or from a *masks* directory relative to the image tiles location.

ColorLimitTransformer

Layer Definition File Specification

```
<Delegate>ColorLimitTransformer(r,g,b)</Delegate>
```

Limits each colour channel of the retrieved image tiles to be less than the provided limiting colour.

PARAMETER	DESCRIPTION
r	The red channel of the limiting colour. Integer in the range [0, 255].
g	The green channel of the limiting colour. Integer in the range [0, 255].
b	The blue channel of the limiting colour. Integer in the range [0, 255].

ColorToAlphaTransformer

```
<Delegate>ColorToAlphaTransformer(r,g,b)</Delegate>
```

Applies a colour-to-alpha transformation on the incoming images.

The colour-to-alpha transformation is similar in nature to that provided in the GIMP^{[5][6]}: The alpha level of each processed pixel is proportional to how close that pixel is to the target colour.

PARAMETER	DESCRIPTION
r	The red channel of the target colour. Integer in the range [0, 255].
g	The green channel of the target colour. Integer in the range [0, 255].
b	The blue channel of the target colour. Integer in the range [0, 255].

TransparentColorTransformer

```
<Delegate>TransparentColorTransformer(r,g,b,fuzz)</Delegate>
```

Converts a targeted colour to transparent with fuzziness. The fuzziness controls how close the colour of a pixel has to be to the target colour to be made transparent.

Similar to the `ColorToAlphaTransformerDelegate`, but sets any colour within fuzz percent of the target colour to fully transparent.

Useful for masking images that do not have an alpha channel but provide a 'mask' colour.

PARAMETER	DESCRIPTION
r	The red channel of the target colour. Integer in the range [0, 255].
g	The green channel of the target colour. Integer in the range [0, 255].
b	The blue channel of the target colour. Integer in the range [0, 255].

Layer Definition File Specification

fuzz	How close to the target colour a pixel must be to be made transparent. Expressed as a double percentage in the range [0, 1], where <ul style="list-style-type: none">• 0 = only the target colour,• 1 = all colours.
------	--

ResizeTransformer

```
<Delegate>ResizeTransformer(w,h)</Delegate>
```

Applies a resize transformation to each retrieved tile such that the resulting image dimensions match those specified in the delegate definition. Where the target dimensions do not match those of the retrieved image tile, bilinear interpolation is used for resizing the tile.

Useful for resizing tiles on-the-fly to the standard 512x512 size where server-side pre-processing is not an option. This might occur, for example, when retrieving image tiles from an external server.

PARAMETER	DESCRIPTION
w	The target width, in pixels. Integer in the range (0, ∞).
h	The target image height, in pixels. Integer in the range (0, ∞).

StripingFilterTransformer

```
<Delegate>  
  StripingFilterTransformer(loCols,loRows,hiCols,hiRows)  
</Delegate>
```

Applies a filter to the retrieved image tiles to remove striping noise using a combination of low-pass and high-pass filtering.

The parameters are used to specify (in pixels) the size of the low- and high-pass filters to apply to the rows and columns of the retrieved tile image.

A good rule of thumb when choosing the size of filter to apply is:

1. Make the size of the low-pass filter large enough to encompass the striping (e.g. twice the size of the striping)
2. Make the size of the high-pass filter small enough to fit within the striping (e.g. half the size of the striping)

PARAMETER	DESCRIPTION
loCols	The size (in pixels) of the low-pass filter to apply to the columns of the image. Integer in the range [0, <code>imageHeight</code>].
loRows	The size (in pixels) of the low-pass filter to apply to the rows of the image. Integer in the range [0, <code>imageWidth</code>].
hiCols	The size (in pixels) of the high-pass filter to apply to the columns of the image. Integer in the range [0, <code>imageHeight</code>].

Layer Definition File Specification

hiRows	The size (in pixels) of the high-pass filter to apply to the rows of the image. Integer in the range [0, imageWidth].
--------	--

Note: If the low-pass and high-pass filters are the same size for columns or rows, no effect will be seen in that direction of the image.

HttpRetriever

```
<Delegate>HttpRetriever</Delegate>
```

Used to retrieve image tiles with a HTTP connection. This is the normal behaviour to use when retrieving image tiles from a remote (non-GA) tile server.

PassThroughZipRetriever

```
<Delegate>PassThroughZipRetriever</Delegate>
```

An extension of the `HttpRetriever` that defers decompression of zipped image tiles to load time (rather than retrieve time). This can help reduce space requirements when caching downloaded tiles that have been compressed in a zip file.

TextureTile

```
<Delegate>TextureTile</Delegate>
```

Allows retrieved image tiles to be used within the OpenGL environment by turning them into textures, along with appropriate texture coordinates.

NearestNeighborTile

```
<Delegate>NearestNeighborTile</Delegate>
```

An extension of the `TextureTile` delegate that additionally uses the Nearest Neighbour scaling technique when viewing the highest resolution level of the layer tile set.

This may produce less pixelated imagery when viewing a layer up close, but is not appropriate for datasets that have a distinct value-per-pixel encoding (e.g. Landsat, where each pixel represents a discrete landcover value and ‘blurring’ these boundaries is undesirable).

ColorMapReader

```
<Delegate>
  ColorMapReader(pixelType,byteOrder,missingData)
</Delegate>
```

Treats the retrieved image tiles as an elevation model, and applies a colour map based on the elevations recorded in the retrieved tiles.

PARAMETER	DESCRIPTION
pixelType	The pixel format of the elevation tiles. Valid values are: <ul style="list-style-type: none"> • Float32 • Int32 • Int16 • Int8
byteOrder	The byte order of the elevation tiles ^[8] Valid values are:

Layer Definition File Specification

	<ul style="list-style-type: none"> • little • big
missingData	<p>The value used in the elevation tiles to represent missing data.</p> <p>Float in the range $(-\infty, \infty)$.</p>

When this delegate is used, a `<ColorMap>` must also be provided within the XML layer definition. This is described in more detail later in this document.

NormalMapReader

```
<Delegate>
  NormalMapReader(pixelType,byteOrder,missingData)
</Delegate>
```

Similar to the `ColorMapReader`. Treats the retrieved image tiles as an elevation model and calculates and displays the normal map^[7] for the elevation model mesh that would be generated from the elevation data.

This delegate is normally used for debugging purposes, as it displays the normal maps that are used internally in other delegates such as the `ShadedElevationReader`.

PARAMETER	DESCRIPTION
pixelType	<p>The pixel format of the elevation tiles.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • Float32 • Int32 • Int16 • Int8
byteOrder	<p>The byte order of the elevation tiles^[8]</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • little • big
missingData	<p>The value used in the elevation tiles to represent missing data.</p> <p>Float in the range $(-\infty, \infty)$.</p>

ShadedElevationReader

```
<Delegate>
  ShadedElevationReader(pixelType,byteOrder,
                        missingData,(sunX,sunY,sunZ),
                        exaggeration[(min,max)])
</Delegate>
```

Treats retrieved image tiles as elevation data, and generates a shading based on the elevation data combined with a provided virtual sun position. Shading is calculated as a simple dot product between the calculated normals of the elevation model and the sun vector.

PARAMETER	DESCRIPTION
pixelType	<p>The pixel format of the elevation tiles.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • Float32 • Int32 • Int16

Layer Definition File Specification

	<ul style="list-style-type: none"> • Int8
byteOrder	<p>The byte order of the elevation tiles ^[8]</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • little • big
missingData	<p>The value used in the elevation tiles to represent missing data.</p> <p>Float in the range $(-\infty, \infty)$.</p>
(sunX,sunY,sunZ)	<p>A vector that represents the direction of the virtual sun to use when calculating the shading effect. Note that this is expressed in arbitrary Cartesian coordinates, not geographic coordinates.</p> <p>Floats in the range $(-\infty, \infty)$.</p>
exaggeration	<p>The vertical exaggeration to bake into the shading.</p> <p>This is essentially a scaling value that will be applied to all elevation pixels before the normal map is calculated. Useful for exaggerating the normal colouring effect for flatter elevation models.</p> <p>Float in the range $(0, \infty)$.</p>
(min,max)	<p>The minimum and maximum elevation values to use when calculating shading.</p> <p>Optional.</p> <p>Expressed as doubles in metres.</p>

LocalRequester

```
<Delegate>LocalRequester</Delegate>
```

Retrieves image tiles directly from a local filesystem. Tiles will not be cached in the usual locations but will instead be read directly from the tileset as required.

URLRequester

```
<Delegate>URLRequester</Delegate>
```

The default requester delegate. Retrieves image tiles from the provided service <URL> and caches downloaded tiles in the usual location.

<LastUpdate>

```
<LastUpdate>01 01 2010 12:34:56 GMT+10:00</LastUpdate>
```

Records the timestamp of the last time the layer data was updated. This is compared to the timestamps of image tiles residing in the cache and used to trigger re-downloading of the tiles.

This field should be updated whenever the tileset for a layer is re-generated.

Expects timestamps in the format “dd MM yyyy HH:mm:ss Z”^[3].

<DatasetName>

```
<DatasetName>magnetics/edition5/magnetics</DatasetName>
```

Identifies the layer dataset on the tile server identified by the service <URL>.

Layer Definition File Specification

In the case of a remote retrieval, this field gets passed to the tile server as query parameter 'T'.

In the case of tiles retrieved using the `LocalRequester` delegate, the `<DataSetName>` is used in conjunction with the service `<URL>` to locate the tileset on the local filesystem. The choice of what part of the path to include in the `<URL>` and what to include in the `<DataSetName>` is somewhat arbitrary, but often local tilesets are stored in a common location so it makes sense to set the `<URL>` element to this common path and use the `<DataSetName>` to uniquely identify the target layer tileset inside that common location.

<ImageFormat>

```
<ImageFormat>application/zip</ImageFormat>
```

Specifies the default (preferred) image format to use for retrieving image tiles.

In the case of a remote retrieval, this field gets passed to the tile server as query parameter 'F'.

Values should be expressed as MIME media types^[9]. Valid values are:

- application/zip
- image/dds
- image/gif
- image/jpg
- image/png
- image/tga
- image/tif

If the image tiles are not available in the specified format, the retriever may fall back to those listed in the `<AvailableImageFormats>` element.

Note: Currently neither the Viewer nor the Animator support this fallback feature. This may be implemented at a later date.

<AvailableImageFormats>

```
<AvailableImageFormats>  
  <ImageFormat>application/zip</ImageFormat>  
  <ImageFormat>image/jpg</ImageFormat>  
</AvailableImageFormats>
```

Lists the available image formats for this layer.

These may be used in the event that the default image format is unavailable for some reason.

<ImageFormat>

Each `<ImageFormat>` element specifies an image format that this layer's image tiles are available in. They follow the same restrictions as the top-level `<ImageFormat>` element.

<FormatSuffix>

```
<FormatSuffix>.zip</FormatSuffix>
```

Specifies the extension to add to downloaded image tiles when storing them in the tile cache. This should correspond to the image format specified in the `<ImageFormat>` element.

<NumLevels>

```
<NumLevels count="7" numEmpty="0" />
```


Layer Definition File Specification

Specifies the number of levels available in the dataset, along with the number of 'empty' levels (i.e. levels that contain no data).

ATTRIBUTE	DESCRIPTION
count	The number of tile levels in the dataset.
numEmpty	<p>The number of top-level levels that contain no data.</p> <p>For example, if the Level Zero Tile Size (LZTS) is 36° and numEmpty=2, the first level of data will have tiles of size 9° (LZTS / 2^numEmpty) and will 'turn on' at level 2.</p> <p>Most often numEmpty should be set to 0.</p> <p>This attribute can be used to create layers that activate more specific imagery as the user approaches the surface of the globe.</p>

<TileOrigin>

```
<TileOrigin>
  <LatLon units="degrees" latitude="-90" longitude="-180" />
</TileOrigin>
```

Specifies the Lat/Lon location of the tileset's South-West corner.

For most layers this will be (-90, -180). This value will affect how tiles are referenced. You might want to change this value if your image tiles are not indexed from a (-90, -180) origin.

<LevelZeroTileDelta>

```
<LevelZeroTileDelta>
  <LatLon units="degrees" latitude="36" longitude="36" />
</LevelZeroTileDelta>
```

Specifies the size (in latitude/longitude) of the level zero tiles, and is commonly called the Level Zero Tile Size (LZTS).

This size is used to calculate the geographical extent of each tile in the layer at all levels.

In general, the geographical extent of a tile from level N can be calculated as (LZTS / 2^N).

<TileSize>

```
<TileSize>
  <Dimension width="512" height="512" />
</TileSize>
```

Specifies the dimensions (in pixels) of each tile in the layer.

<Sector>

```
<Sector>
  <SouthWest>
    <LatLon units="degrees" latitude="-43" longitude="112.5" />
  </SouthWest>
  <NorthEast>
    <LatLon units="degrees" latitude="-9" longitude="154.66" />
  </NorthEast>
</Sector>
```

Layer Definition File Specification

Specifies the geographical extents of the layer. Extents are specified using a South-West and North-East corner point (bottom-left and top-right).

This element essentially controls where on the globe the layer is mapped to.

<SouthWest>

The south-west (bottom-left) corner of the sector, expressed as a Lat/Lon. See <LatLon>.

<NorthEast>

The north-east (top-right) corner of the sector, expressed as a Lat/Lon. See <LatLon>.

<SectorResolutionLimit>

```
<SectorResolutionLimit maxLevelNum="6">
  <Sector>
    <SouthWest>
      <LatLon units="degrees" latitude="-22" longitude="117"/>
    </SouthWest>
    <NorthEast>
      <LatLon units="degrees" latitude="-19" longitude="121"/>
    </NorthEast>
  </Sector>
</SectorResolutionLimit>
```

Used to specify the highest level of detail available for specific sub-sectors of the data. This is useful when the data contains varying levels of detail where the generation of high-resolution images for low-resolution sections of the data doesn't make sense.

A layer definition file may contain 0-n <SectorResolutionLimit/> elements for 0 or more sub-sectors. If multiple <SectorResolutionLimit/> elements are defined and sub-sectors overlap, the highest resolution limit will be used

ATTRIBUTE	DESCRIPTION
maxLevelNum	The highest level to use for the provided sub-sector. Must be a positive integer.

<UseTransparentTextures>

```
<UseTransparentTextures>true</UseTransparentTextures>
```

Whether or not to enable transparency for this layer.

Valid values are:

- true
- false

<ForceLevelZeroLoads>

```
<ForceLevelZeroLoads>true</ForceLevelZeroLoads>
```

Whether or not to load level 0 tiles from the cache on the render thread so that they are immediately visible. Setting this to true will slow down the render thread on the first drawing of the layer, but ensures that at least a low-resolution version of the layer is available immediately.

Valid values are:

- true
- false

<RetainLevelZeroTiles>

```
<RetainLevelZeroTiles>>true</RetainLevelZeroTiles>
```

If true, will keep the layer's level 0 tiles in memory and prevent them from being cleared out of the cache when it gets full. This is used to ensure that at least a low-resolution version of the layer is available immediately.

Valid values are:

- true
- false

<TextureFormat>

```
<TextureFormat>image/dds</TextureFormat>
```

Specifies the texture format to use when loading image tiles into memory. If not specified, the image tiles will be loaded in their native format. If specified, images will be converted to the specified format on load.

Currently only supports the DDS (Direct Draw Surface) format. This is a format designed for use in realtime graphics, and is the recommended format to use where possible.

Values must be specified as MIME media formats^[9]. Valid values are:

- image/dds
- <blank>

<UseMipMaps>

```
<UseMipMaps>true</UseMipMaps>
```

Enable automatic texture mipmapping^[10]. This is an optimisation technique that may increase rendering speed of the layer.

Valid values are:

- true
- false

<SplitScale>

```
<SplitScale>0.9</SplitScale>
```

Deprecated. Use <DetailHint> instead.

This is the old way of controlling at what point tiles are split and the next level is loaded. The split scale is related to the resolution of the tile as viewed in the world window compared to the resolution of the image tile.

A <SplitScale> of 0.9 is equivalent to a <DetailHint> of 0.

Valid values are floats in the range [0, ∞).

<DetailHint>

```
<DetailHint>0.5</DetailHint>
```

This is the new (and preferred) way of controlling at what point tiles are split and the next level is loaded. The hint value is somewhat abstract, but the following is a guideline:

- 0 is default detail,

Layer Definition File Specification

- >0 is more detailed, and
- <0 is less detailed.

For example, setting a large value detail hint (say, 5) will result in higher resolution image tiles being loaded sooner, which may slow render times but increase the visual detail of the layer. It may also help prevent tile flickering that occurs when tiles change resolution within the user's field of view.

Valid values are floats in the range $(-\infty, \infty)$. In most cases the default of 0 will be satisfactory.

<TransparencyColors>

```
<TransparencyColors>
  <Color red="255" green="0" blue="255" alpha="0" />
  ...
</TransparencyColors>
```

Used to specify a colour, or list of colours, that should be made transparent in the layer. Useful for layers that use a colour mask to represent transparency rather than an alpha channel.

For more powerful control over transparency, consider using the delegates mechanism described in the <Delegates> section.

See the <Color> element for a description of its attributes.

<TimedExpiry>

TODO

<ColorMap>

```
<ColorMap interpolateHue="true">
  <Entry elevation="100" red="0" green="0" blue="0"
alpha="50" />
  ...
</ColorMap>
```

Used in conjunction with the `ColorMapReader` delegate to define an elevation \rightarrow colour mapping. If defined when the `ColorMapReader` is not used, this element will be ignored.

ATTRIBUTE	DESCRIPTION
interpolateHue	<p>Whether interpolation should be based on the Hue channel of the HSB colour space.</p> <p>If true, the Hue/Saturation/Brightness will be interpolated between the colours. If false, interpolation will occur on each channel in RGBA space.</p> <p>Valid values are:</p> <ul style="list-style-type: none">• true• false

<Entry>

```
<Entry elevation="100" red="0" green="0" blue="10" alpha="50" />
```

Defines a single entry in the colour map. Each entry maps a specific elevation to a colour in RGBA colour space.

Layer Definition File Specification

ATTRIBUTE	DESCRIPTION
elevation	The elevation this entry is for. Float value in metres (m). Required.
red	The red channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
green	The green channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
blue	The blue channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
alpha	The alpha channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 255.

SURFACE SHAPEFILE LAYERS

```
<Layer version="1" layerType="SurfaceShapeShapefileLayer">
```

Surface shapefile layers are able to read a single shapefile^[11] containing point, line or polygon data, and display that data as points/lines drawn onto the globe.

This layer type differs from the more general point and geometry layers in that the data is rendered as a texture onto the globe, and does not exist as polygons.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="SurfaceShapeShapefileLayer">

  <DisplayName>Marine Boundaries EEZ Limits</DisplayName>
  <URL>eez_limit.shp</URL>

  <OutlineColor red="128" green="128" blue="128" />
  <OutlineWidth>5</OutlineWidth>

</Layer>
```

<URL>

```
<URL>eez_limit.shp</URL>
```

The URL of the shapefile to load data from.

<DrawInterior>

```
<DrawInterior>>true</DrawInterior>
```

For polygonal shapefiles, specifies whether or not to draw the interior of the shape.

For point and line data, this element is ignored.

Valid values are:

- true
- false

<DrawOutline>

```
<DrawOutline>>true</DrawOutline>
```

Specifies whether or not to draw the outline of the shapefile data.

Valid values are:

- true
- false

<OutlineWidth>

```
<OutlineWidth>5</OutlineWidth>
```

Specifies the width (in pixels) of the outline to draw for the shapefile.

If <DrawOutline> is false, this element is ignored.

Accepts floats in the range [0, ∞).

<Antialiasing>

```
<Antialiasing>>true</Antialiasing>
```

Specifies whether antialiasing should be applied to the drawing of the shape.

Valid values are:

- true
- false

<InteriorColor>

```
<InteriorColor red="128" green="128" blue="128"/>
```

Specifies the colour to use when drawing the interior of the shape.

If <DrawInterior> is false, this element is ignored.

ATTRIBUTE	DESCRIPTION
red	The red channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
green	The green channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
blue	The blue channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.

<OutlineColor>

```
<OutlineColor red="128" green="128" blue="128"/>
```

Specifies the colour to use when drawing the outline of the shape.

If <DrawOutline> is false, this element is ignored.

ATTRIBUTE	DESCRIPTION
red	The red channel of the colour. Integer in the range [0, 255]. Optional. Defaults to 0.
green	The green channel of the colour. Integer in the range [0, 255].

	Optional. Defaults to 0.
blue	The blue channel of the colour. Integer in the range [0, 255].
	Optional. Defaults to 0.

<InteriorOpacity>

```
<InteriorOpacity>0.85</InteriorOpacity>
```

Specifies the opacity to use when drawing the interior of the shape.

If <DrawInterior> is false, this element is ignored.

Valid values are floats in the range [0, 1].

<OutlineOpacity>

```
<OutlineOpacity>0.85</OutlineOpacity>
```

Specifies the opacity to use when drawing the outline of the shape.

If <DrawOutline> is false, this element is ignored.

Valid values are floats in the range [0, 1].

<StippleFactor>

```
<StippleFactor>3</StippleFactor>
```

Specifies the stipple factor to use to draw the shape outline. The stipple factor is a repeater multiplier for each bit in the stipple pattern, and is used in the same way as it is defined in the OpenGL specification^[12].

If no stipple pattern is defined, or <DrawOutline> is false, this element is ignored.

Valid values are integers in the range [1, 256]. Defaults to 1.

<StipplePattern>

```
<StipplePattern>0xFF00</StipplePattern>
```

Specifies the stipple pattern to use to draw the shape outline (if applicable). The stipple pattern is a 16-bit pattern that defines how the shape outline will be rendered. It is combined with the <StippleFactor> to control how outlines are drawn.

If <DrawOutline> is false, this element is ignored.

Example patterns:

- 0xFFFF = Normal solid line (default)
- 0xF0F0 = Dashed line
- 0xFF00 = Long dashes
- 0x0101 = Dotted line

Valid values are expressions that resolve to an integer. Defaults to 0xFFFF.

POINT LAYERS

```
<Layer version="1" layerType="PointLayer">
```

Layer Definition File Specification

Point layers are used to display point information on the globe. Points can be provided via a shapefile (recommended), or hard-coded within the layer definition file. The point layer supports the notion of styles, and has powerful mechanisms for mapping point attributes to colours and text labels etc.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="PointLayer">
  <DisplayName>Volcanoes of the World (Smithsonian)</DisplayName>
  <PointType>Marker</PointType>
  <URL>smithsonian_volcanoes.zip</URL>
  <DataFormat>Shapefile</DataFormat>
  <DataCacheName>GA/Volcanoes/smithsonian_volcanoes.zip</DataCacheName>
  <Styles>
    <Style name="color1" default="true">
      <Property name="Material" value="128,128,128" />
      <Property name="ShapeType" value="gov.nasa.worldwind.render.markers.Cone" />
    </Style>
    <Style name="color2">
      <Property name="Material" value="0,0,255" />
      <Property name="ShapeType" value="gov.nasa.worldwind.render.markers.Cone" />
    </Style>
    <Style name="color3">
      <Property name="Material" value="0,85,255" />
      <Property name="ShapeType" value="gov.nasa.worldwind.render.markers.Cone" />
    </Style>
  </Styles>
  <Attributes>
    <Attribute name="NAME">
      <Text value="&lt;p&gt; &lt;b&gt;% value% &lt;/b&gt; &lt;/p&gt;" placeholder="% value%" />
    </Attribute>
    <Attribute name="STATUS">
      <Text value="Status: &lt;b&gt;% value% &lt;/b&gt; &lt;br&gt;" placeholder="% value%" />
    </Attribute>
    <Attribute name="TYPE">
      <Text value="Type: &lt;b&gt;% value% &lt;/b&gt; &lt;br&gt;" placeholder="% value%" />
    </Attribute>
    <Attribute name="TIMEFRAME">
      <Case value="?" style="color1" />
      <Case value="U" style="color2" />
      <Case value="Q" style="color3" />
      <Text value="Timeframe: &lt;b&gt;% value% &lt;/b&gt; &lt;br&gt;" placeholder="% value%" />
    </Attribute>
    <Attribute name="NUMBER">
      <Link url="http://www.volcano.edu/world/volcano.cfm?vnum=% value%"
placeholder="% value%" />
    </Attribute>
  </Attributes>
</Layer>
```

<PointType>

```
<PointType>Marker</PointType>
```

Specifies what to display at each point in the dataset.

Valid values are:

- **Marker** – Renders points as markers that can be specified in the `ShapeType` property of the `<Style>` element. Supports mouse-over display of additional information.
- **Annotation** – Renders points as annotation boxes that contain textual additional information. Annotations can be styled to appear as pop-up boxes or text labels etc.
- **Icon** – Renders points as user-facing icons. Supports mouse-over display of additional information.

<DataFormat>

```
<DataFormat>Shapefile</DataFormat>
```

Defines the format the point data is to be supplied in.

Currently, valid values are:

- **Shapefile** – Point data is supplied in a shapefile referenced in the <URL> element. This shapefile may be zipped. This is the recommended data format in most cases.
- **XML** – Point data is supplied within the layer definition file in a <Points> element. This can be useful when defining a limited, unchanging dataset (such as capital cities etc.)

<URL>

```
<URL>capital_cities.zip</URL>
```

When <DataFormat> is Shapefile, this element provides the location of the shapefile to use. The URL can be provided in absolute or relative format, and can refer to an uncompressed .shp file, or a compressed .zip archive containing the shapefile (recommended in most cases).

When the XML data format is used, this element will be ignored.

<Styles>

```
<Styles>
  <Style name="color1" default="true">
    <Property name="Material" value="128,128,128"/>
    <Property name="ShapeType" value="..." />
    ...
  </Style>
  ...
</Styles>
```

Styles are used to control the look-and-feel of points as they are displayed on the globe.

The <Styles> element may contain 0-many <Style> elements. Styles may be selected by selector elements listed in an <Attributes> section, or based on the ‘default’ property of child <Style> elements.

<Style>

```
<Style name="color1" default="true">
  <Property name="Material" value="128,128,128"/>
  <Property name="ShapeType" value="..." />
  ...
</Style>
```

Defines a single style. A style contains multiple <Property> elements that can be used to control particular properties of the point’s look-and-feel.

ATTRIBUTE	DESCRIPTION
name	The name used to identify this style. Style names can be any string value, but must be unique within a layer definition.
default	Whether or not this is the default style to use for this layer. Valid values are: <ul style="list-style-type: none"> • true

	<ul style="list-style-type: none"> • false <p>Optional. Defaults to false.</p>
--	---

<Property>

```
<Property name="Surface"
value="128,0,128|0.5"
type="Color|Double"/>
```

Sets a value for a particular property for the parent style. The properties are applied to a target object(s), controlled by the <PointType> element (e.g. applied to the attributes of a Marker, Icon or Annotation).

For example, if a <Property> element is specified with name="Opacity", the style mechanism will look for a method setOpacity(opacity) on the target object(s), and will set it to the value provided in the value attribute.

ATTRIBUTE	DESCRIPTION
name	<p>The name of the property this element is controlling.</p> <p>The property name should match a JavaBean^[13] property name on the target object (target objects are listed below).</p> <p>A list of commonly used properties is given below.</p>
value	<p>A pipe-separated list of values that correspond to arguments of the setter method for the names property.</p> <p>Each value may be a compound type with components separated by commas. For example, colour values are expressed as 1,3 or 4 comma-separated integers (e.g. "255,0,45").</p>
type	<p>A pipe-separated list of types that correspond to the types of the arguments provided in the value attribute.</p> <p>Optional. If not provided, the style mechanism will attempt to derive the correct type and coerce the value into that type.</p> <p>If provided, there must be a type provided for each element specified in value.</p> <p>Valid types are:</p> <ul style="list-style-type: none"> • String • Integer • Float • Long • Double • Character • Byte • URL • File • Color ('grey', 'r,g,b' or 'r,g,b,a', integers in range [0,255]) • Insets ('t,l,b,r', integers) • Dimension ('w' or 'w,h', integers) • Point ('xy' or 'x,y', integers) • Font (eg. 'Arial-bold-12pt')^[14]

Layer Definition File Specification

	<ul style="list-style-type: none"> • Material (As for Color) • Boolean
--	--

The most commonly used properties available for styling are given below. Note that this is not an exhaustive list. For a complete list, look at the Javadoc documentation for the classes that correspond to the point types listed in the table.

PROPERTY	DESCRIPTION
Marker	
<ul style="list-style-type: none"> • <code>gov.nasa.worldwind.render.markers.MarkerAttributes</code> 	
Material	<p>Controls the diffuse colour to apply to the marker.</p> <p>Type: Material</p>
ShapeType	<p>Specifies the fully qualified class name of the marker shape to use for markers controlled by the style.</p> <p>Classes must implement the <code>MarkerShape</code> interface.</p> <p>Some example shape types are:</p> <ul style="list-style-type: none"> • <code>gov.nasa.worldwind.render.markers.Sphere</code> • <code>gov.nasa.worldwind.render.markers.Cube</code> • <code>gov.nasa.worldwind.render.markers.Cone</code> • <code>gov.nasa.worldwind.render.markers.Cylinder</code> <p>Type: String</p>
Opacity	<p>Controls the opacity of the marker.</p> <p>Values must be in the range [0, 1].</p> <p>Type: Double</p>
MinMarkerSize	<p>Markers may grow and shrink based on the distance from the camera to the marker. This property controls the minimum size the marker may be.</p> <p>Type: Double</p>
MaxMarkerSize	<p>Markers may grow and shrink based on the distance from the camera to the marker. This property controls the maximum size the marker may be.</p> <p>Type: Double</p>
Icon	
<ul style="list-style-type: none"> • <code>gov.nasa.worldwind.render.UserFacingIcon</code> 	
ImageSource	<p>The absolute file path to the image for the icon.</p> <p>Type: String</p>
Highlighted	<p>Whether or not to highlight the icon.</p> <p>Type: Boolean</p>
HighlightScale	<p>A multiplier used to determine the size of the icon highlight as compared to the icon itself.</p> <p>Usually this should be > 1. Defaults to 1.2.</p> <p>Type: Double</p>
ToolTipFont	<p>The font to use when displaying the icon tooltip (if applicable).</p> <p>Type: Font</p>

Layer Definition File Specification

ToolTipTextColor	<p>The colour to use when displaying the tooltip text (if applicable).</p> <p>Defaults to the system look and feel colour.</p> <p>Type: Color</p>
ShowToolTip	<p>Whether or not to show a tooltip for the icon.</p> <p>Type: Boolean</p>
AlwaysOnTop	<p>Whether the icon should be drawn with depth-testing or not.</p> <p>Type: Boolean</p>
BackgroundImage	<p>The file path to the image for the background of the marker.</p> <p>Type: String</p>
BackgroundScale	<p>A multiplier used to determine the size of the icon background image as compared to the icon itself.</p> <p>Type: Double</p>
<p>Annotation</p> <ul style="list-style-type: none"> • au.gov.ga.worldwind.common.layers.point.annotation.EnhancedAnnotation • au.gov.ga.worldwind.common.layers.point.annotation.EnhancedAnnotationAttributes 	
AlwaysOnTop	<p>Whether the annotation should be drawn with depth testing or not.</p> <p>Type: Boolean</p>
MinActiveAltitude	<p>The minimum altitude of the camera at which the annotation is 'activated'.</p> <p>Values are doubles expressed in metres.</p> <p>Type: Double</p>
MaxActiveAltitude	<p>The minimum altitude of the camera at which the annotation is 'activated'.</p> <p>Values are doubles expressed in metres.</p> <p>Type: Double</p>
AltitudeMode	<p>Specifies the altitude position behaviour of the annotation.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • Empty - If the annotation Position elevation is lower than the highest elevation on the globe, the annotation will be drawn above the ground using its elevation as an offset, scaled by the current vertical exaggeration. Otherwise, the original elevation will be used. • 0 – Absolute positioning. The annotation is drawn at the elevation specified in its position. • 1 – Clamp to ground. The annotation is drawn at the surface of the globe. • 2 – Relative to ground. The annotation's position elevation is used as an offset above the globe, scaled by vertical exaggeration. <p>Type: Integer</p>
MinEyeDistance	<p>The minimum distance between the annotation and the camera</p>

Layer Definition File Specification

	<p>at which the annotation is 'activated'.</p> <p>Expressed in metres.</p> <p>Type: Double</p>
MaxEyeDistance	<p>The maximum distance between the annotation and the camera at which the annotation is 'activated'.</p> <p>Expressed in metres.</p> <p>Type: Double</p>
FadeDistance	<p>The camera-travel distance over which the annotation fades in/out between 'active' and 'inactive' states. A larger distance will result in a slower fade effect.</p> <p>Expressed in metres.</p> <p>Type: Double</p>
FrameShape	<p>The shape of the frame to draw around the annotation, if applicable.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • Render.FrameFactory.ShapeRectangle • Render.FrameFactory.ShapeEllipse • Render.FrameFactory.ShapeNone <p>Type: String</p>
Highlighted	<p>Whether or not to display the annotation as highlighted.</p> <p>Type: Boolean</p>
HighlightScale	<p>A multiplier used to control the size of the highlight compared to the size of the annotation.</p> <p>Usually this value should be > 1. Defaults to 1.2.</p> <p>Type: Double</p>
Scale	<p>A multiplier used to control the size of the annotation.</p> <p>Default is 1. Values < 1 will result in a smaller annotation. Values > 1 will result in a larger annotation.</p> <p>Values must be in the range [0, ∞).</p> <p>Type: Double</p>
Opacity	<p>Controls the opacity of the annotation.</p> <p>Values must be in the range [0, 1]. Default is 1.</p> <p>Type: Double</p>
Leader	<p>Specifies the shape to use for the leader (the connector from the annotation to the specific position on the globe).</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • Render.FrameFactory.LeaderTriangle • Render.FrameFactory.LeaderNone <p>Type: String</p>

<p>LeaderGapWidth</p>	<p>Set the gap (in pixels) that the leader shape spans.</p> <p>Defaults to 40. Values must be in the range [0, ∞).</p> <p>Type: Integer</p>
<p>CornerRadius</p>	<p>Sets the corner radius (in pixels) to use when drawing a rectangular frame.</p> <p>A value of 0 will result in square corners. Larger numbers will result in more rounded corners. Values must be in the range [0, ∞). The default value is 20.</p> <p>Type: Integer</p>
<p>BorderWidth</p>	<p>The width (in pixels) of the border to use when drawing the frame.</p> <p>A value of 0 will result in no border. Values must be in the range [0, ∞). Default value is 1.</p> <p>Type: Double</p>
<p>BorderStipplePattern</p>	<p>Specifies the stipple pattern to use to draw the frame border (if applicable). The stipple pattern is a 16-bit pattern that defines how the border will be rendered. It is combined with the <code>BorderStippleFactor</code> to control how outlines are drawn.</p> <p>Example patterns:</p> <ul style="list-style-type: none"> • 0xFFFF = Normal solid line • 0xF0F0 = Dashed line • 0xFF00 = Long dashes • 0x0101 = Dotted line <p>Valid values are expressions that resolve to an integer. Defaults to 0xAAAA (dotted).</p> <p>Type: Integer</p>
<p>BorderStippleFactor</p>	<p>Specifies the stipple factor to use to draw the shape outline (if applicable). The stipple factor is a repeater multiplier for each bit in the stipple pattern, and is used in the same way as it is defined in the OpenGL specification^[12].</p> <p>Valid values are in the range [1, 256]. Default is 1.</p> <p>Type: Integer</p>
<p>Font</p>	<p>Specifies the font to use to render the annotation text.</p> <p>Font string formats are defined in the Javadoc for the Java Font class^[14].</p> <p>Some example font strings:</p> <ul style="list-style-type: none"> • “Arial-BOLD-12” • “Verdana-PLAIN-8” • “Times New Roman-ITALIC-15” • “Courier New-BOLDITALIC-12” <p>Defaults to “Arial-Plain-12”.</p>

Layer Definition File Specification

	Type: Font
TextAlign	<p>Sets the text alignment for the annotation.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • gov.nasa.worldwind.avkey.Left • gov.nasa.worldwind.avkey.Right • gov.nasa.worldwind.avkey.Center <p>Defaults to “gov.nasa.worldwind.avkey.Left”.</p> <p>Type: String</p>
TextColor	<p>Specifies the colour to use for the annotation text.</p> <p>Defaults to black (“0,0,0”).</p> <p>Type: Color</p>
BackgroundColor	<p>Specifies the colour to use for the background of the annotation frame shape (if applicable).</p> <p>Defaults to white (“255,255,255”).</p> <p>Type: Color</p>
BorderColor	<p>Specifies the colour to use for the border of the annotation frame shape (if applicable).</p> <p>Defaults to grey (“171,171,171”).</p> <p>Type: Color</p>
ImageSource	<p>Specifies the path of an image to display with the annotation.</p> <p>If not specified, no image will be displayed.</p> <p>Type: String</p>
ImageScale	<p>The scale multiplier to apply to the annotation image (if applicable).</p> <p>Values must be in the range (0, ∞). Default value is 1. Values <1 will shrink the image. Values >1 will expand the image.</p> <p>Type: Double</p>
ImageOpacity	<p>Controls the opacity of the annotation image (if applicable).</p> <p>Values must be in the range [0, 1]. Default value is 1.</p> <p>Type: Double</p>
ImageRepeat	<p>Specifies the repeat style to use for the annotation image.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • gov.nasa.worldwind.avkey.RepeatNone • gov.nasa.worldwind.avkey.RepeatX • gov.nasa.worldwind.avkey.RepeatY • gov.nasa.worldwind.avkey.RepeatXY <p>Default is “gov.nasa.worldwind.avkey.RepeatXY”.</p> <p>Type: String</p>

Layer Definition File Specification

ImageOffset	<p>Specifies the location of the bottom-left corner of the image in relation to the annotation.</p> <p>Default is (0, 0).</p> <p>Type: Point</p>
Effects	<p>Specifies special effects to apply to the annotation text.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> gov.nasa.worldwind.avkey.TextEffectNone gov.nasa.worldwind.avkey.TextEffectOutline gov.nasa.worldwind.avkey.TextEffectShadow <p>Type: String</p>
Insets	<p>Specifies the insets/padding (in pixels) to apply to the annotation frame (if applicable).</p> <p>Default is (20,15,15,15).</p> <p>Type: Insets</p>

<Attributes>

```
<Attributes>
  <Attribute name="NAME">
    <Text value="&lt;b&gt;%v&lt;/b&gt;" placeholder="%v"/>
  </Attribute>
  ...
</Attributes>
```

Defines style selectors and text generators based on the attributes of points in the dataset. Attributes are defined in Shapefiles as attribute columns, and in XML as named attributes.

<Attribute>

```
<Attribute name="STATUS">
  <Text value="Status: %v&lt;br&gt;" placeholder="%v"/>
  ...
</Attribute>
```

Defines style selectors and text generators for a single named attribute of a point.

If a point does not contain a value for the named attribute, this element will be ignored for that point.

ATTRIBUTE	DESCRIPTION
name	The name of the attribute this element refers to.

<Text>

```
<Text value="Timeframe: &lt;b&gt;%value&lt;/b&gt;"
  placeholder="%value"/>
```

<Text> elements specify a piece of text that is to be added to the rendered point based on an attribute of the dataset. All text elements from all enabled attributes will be concatenated together in order and rendered according to the selected style in a manner appropriate for the point type specified.

Layer Definition File Specification

Text is rendered as follows:

- Marker: As a popup that occurs on mouse rollover
- Icon: As a tooltip that occurs on mouse rollover
- Annotation: As the annotation text itself.

ATTRIBUTE	DESCRIPTION
value	<p>The escaped^[15] html-encoded^[16] text to render for points that have a value for the attribute.</p> <p>The actual value of the attribute can be substituted using the placeholder value defined in the <code>placeholder</code> attribute.</p>
placeholder	<p>The placeholder string to replace with the actual value of the parameter.</p> <p>This can be any string value, but by convention the placeholder string is usually surrounded with ‘%’ characters.</p> <p>e.g. “%value%”, “%v%”, “%theName%”</p>

<Link>

```
<Link url="http://someurl.com/items?n=%v%" placeholder="%v%" />
```

Specifies the link to use for points that have a value for the attribute. Links are triggered when the user clicks on the point in the globe. If multiple <Link> elements are found, the last link element is used.

Hyperlinks can be embedded into the text of a point by using (escaped) HTML encoded <a> tags^[16]. See the <Text> element description for more details.

ATTRIBUTE	DESCRIPTION
url	<p>The URL to render for the link.</p> <p>The actual value of the attribute can be substituted using the placeholder value defined in the <code>placeholder</code> attribute.</p>
placeholder	<p>The placeholder string to replace with the actual value of the parameter.</p> <p>This can be any string value, but by convention the placeholder string is usually surrounded with ‘%’ characters.</p> <p>e.g. “%value%”, “%v%”, “%theName%”</p>

<Case>

```
<Case value="D7" style="RecentEruptionStyle" />
```

A style-selecting element used to describe one branch of a switch that selects a style based on the value of an attribute.

If multiple style-selecting elements from different <Attribute> elements match a style, the last match wins.

<Case> elements are useful for discrete-valued attributes (such as a key or classification ID).

ATTRIBUTE	DESCRIPTION
value	The attribute value that will trigger a match on this case element.

Layer Definition File Specification

style	The name of the style to select on a match.
-------	---

<Regex>

```
<Regex pattern="[a-zA-Z]*[0-9]" style="DefaultStyle" />
```

A style-selecting element used to match an attribute value based on a regular expression^[17].

If multiple style-selecting elements from different <Attribute> elements match a style, the last match wins.

<Regex> elements are useful for textual attributes (such as names or descriptions).

ATTRIBUTE	DESCRIPTION
pattern	The regular expression to apply to the attribute value to test for a match.
style	The name of the style to select on a match.

<Range>

A style-selecting element that matches an attribute value that falls within a specified closed range.

If multiple style-selecting elements from different <Attribute> elements match a style, the last match wins.

<Range> elements can only be used for numeric attributes (such as temperature, population etc.).

ATTRIBUTE	DESCRIPTION
min	The bottom of the closed range
max	The top of the closed range
style	The name of the style to select on a match.

<Points>

```
<Points>
<Point units="degrees" latitude="34.4" longitude="123.4"
[attr1="vall" ...]/>
...
</Points>
```

Used to define the point data for the layer when an external shapefile is not used.

This is designed primarily for small datasets that do not change often (for example capital cities). If the layer is to contain a large number of points, shapefiles are more appropriate as they can be compressed to reduce bandwidth requirements, and will not affect parsing performance of the layer definition file.

The <Points> element can contain multiple <Point> elements.

<Point>

```
<Point units="degrees" latitude="34.4" longitude="123.4"
[attr1="vall" ...]/>
```

Specifies a single point in a point layer. Each <Point> element must provide a lat/lon location, and can specify any number of additional, arbitrary attributes of the form `attributeName="value"`.

Layer Definition File Specification

Additional attributes are used by the Style and Attribute mechanisms to control the appearance of the point on the globe.

ATTRIBUTE	DESCRIPTION
units	As for the <LatLon> element. Optional. Defaults to “degrees”.
latitude	As for the <LatLon> element.
longitude	As for the <LatLon> element.
Custom attributes	Arbitrary attributes can be specified on a per-point basis. These should take form <code>attributeName="value"</code> . Example attributes: <ul style="list-style-type: none">• <code>Name="Canberra"</code>• <code>Population="330000"</code>• <code>Abbreviation="CBR"</code> Values can be any datatype that can be handled by the Style mechanism. If they cannot be coerced into any other type, values will be stored as Strings.

KML LAYERS

```
<Layer version="1" layerType="KMLLayer" >
```

KML layers provide a thin wrapper that allows Keyhole Markup Language (KML) data layers to be viewed inside the GA World Wind suite. It does nothing more than reference a KML or KMZ (compressed KML) file. All metadata (display name etc.) is extracted from the referenced KML file.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="KMLLayer">
  <URL>http://www.someurl.com/myKMLFile.kml</URL>
</Layer>
```

<URL>

```
<URL>http://www.someurl.com/myKMLFile.kml</URL>
```

Specifies the URL of the KML or KMZ file to reference.

CURTAIN IMAGE LAYERS

```
<Layer version="1" layerType="CurtainImageLayer" >
```

Curtain image layers are used to display high-resolution image data on vertical ribbon curtains that can be positioned above or below the surface of the earth. These layers support tiling, and can make more efficient use of textures than Tiled Image Layers for long, thin images.

Curtain image layers are useful for displaying seismic and radar datasets etc.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="CurtainImageLayer">
  <DisplayName>Curtain</DisplayName>
```

Layer Definition File Specification

```
<Service serviceName="DelegatorTileService">
  <URL>http://www.ga.gov.au/apps/world-wind/tiles.jsp</URL>
</Service>
<Delegates>
  <Delegate>MaskReader</Delegate>
  <Delegate>PassThroughZipRetriever</Delegate>
</Delegates>
<LastUpdate>19 09 2009 04:10:00 GMT</LastUpdate>
<DatasetName>s310_23_blured_wb_convert</DatasetName>
<DataCacheName>GA/Seismic Lines/s310_23_blured_wb_convert</DataCacheName>
<ImageFormat>application/zip</ImageFormat>
<AvailableImageFormats>
  <ImageFormat>application/zip</ImageFormat>
</AvailableImageFormats>
<FormatSuffix>.zip</FormatSuffix>
<NumLevels count="8" numEmpty="0" />
<TileSize>
  <Dimension width="512" height="512" />
</TileSize>
<FullSize>
  <Dimension width="52798" height="21423" />
</FullSize>
<Path>
  <LatLon units="degrees" latitude="-28.88778" longitude="112.51076" />
  <LatLon units="degrees" latitude="-29.8456" longitude="113.55251" />
</Path>
<CurtainTop>0</CurtainTop>
<CurtainBottom>-10000</CurtainBottom>
<FollowTerrain>>false</FollowTerrain>
<Subsegments>10</Subsegments>
<UseTransparentTextures>>true</UseTransparentTextures>
<ForceLevelZeroLoads>>true</ForceLevelZeroLoads>
<RetainLevelZeroTiles>>true</RetainLevelZeroTiles>
<UseMipMaps>>true</UseMipMaps>
<SplitScale>0.9</SplitScale>
</Layer>
```

<Service>

See Tiled Image Layer.

<Delegates>

See Tiled Image Layer.

<LastUpdate>

See Tiled Image Layer.

<DatasetName>

See Tiled Image Layer.

<DataCacheName>

See Tiled Image Layer.

<ImageFormat>

See Tiled Image Layer.

<AvailableImageFormats>

See Tiled Image Layer.

<FormatSuffix>

See Tiled Image Layer.

<NumLevels>

See Tiled Image Layer.

<TileSize>

See Tiled Image Layer.

<FullSize>

```
<FullSize>  
  <Dimension width="52798" height="21423" />  
</FullSize>
```

Specifies the total resolution (in pixels) of the curtain image. Often this will be a dimension with the 'width' significantly larger than the 'height'.

<Path>

```
<Path>  
  <LatLon latitude="-28.88" longitude="112.51" />  
  <LatLon latitude="-29.8" longitude="113.55" />  
</Path>
```

Specifies the path that the curtain is to follow.

A path is defined by 2 or more ordered <LatLon> elements. The curtain will be rendered as a vertical strip that passes through the points specified on the path. Between any two points on the path, the curtain section will follow a straight line.

<CurtainTop>

```
<CurtainTop>0</CurtainTop>
```

Specifies the elevation (in metres) of the top of the curtain.

If <FollowTerrain> is true, this value will be used as an offset from sea level (0m).

Valid values are doubles in the range $(-\infty, \infty)$ and must be greater (higher) than the value for <CurtainBottom>.

<CurtainBottom>

```
<CurtainBottom>-1000</CurtainBottom>
```

Specifies the elevation (in metres) of the bottom of the curtain.

If <FollowTerrain> is true, this value will be used as an offset from sea level (0m).

Valid values are doubles in the range $(-\infty, \infty)$ and must be lower than the value for <CurtainTop>.

<FollowTerrain>

```
<FollowTerrain>>true</FollowTerrain>
```

Whether or not the curtain should be deformed to follow the terrain of the elevation model.

If false, the curtain will be rendered at a constant elevation defined with `<CurtainTop>` and `<CurtainBottom>`. This will follow the underlying globe shape (e.g. spherical, ellipsoidal or flat depending on the globe used).

If true, the curtain will be deformed such that it follows the terrain of any elevation model present.

Valid values are:

- true
- false

Defaults to false.

<Subsegments>

```
<Subsegments>10</Subsegments>
```

Specifies the number of subsegments to divide the curtain into between points on the path.

A higher value will increase the terrain- and curvature-following resolution of the curtain, but may affect performance.

Valid values are integers in the range $[0, \infty)$. Default value is 1.

<UseTransparentTextures>

See Tiled Image Layer.

<ForceLevelZeroLoads>

See Tiled Image Layer.

<RetainLevelZeroTiles>

See Tiled Image Layer.

<TextureFormat>

See Tiled Image Layer.

<UseMipMaps>

See Tiled Image Layer.

<SplitScale>

See Tiled Image Layer.

HISTORIC EARTHQUAKE LAYERS

```
<Layer version="1" layerType="HistoricEarthquakesLayer">
```

The historic earthquakes layer is a specialised sub-surface layer type that reads earthquake data from a (optionally compressed) .dat file. Each data point is rendered as a point plotted at the recorded earthquake depth and coloured based on date of occurrence, magnitude or depth.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="HistoricEarthquakesLayer">
  <DisplayName>Earthquakes (by date)</DisplayName>
  <URL>historic_earthquakes.zip</URL>
  <Coloring>Date</Coloring>
```

Layer Definition File Specification

```
<ColoringMinDate value="1950-01-01" format="yyyy-MM-dd" />
<ColoringMaxDate value="2007-01-01" format="yyyy-MM-dd" />
<PointSize>2</PointSize>
</Layer>
```

<URL>

```
<URL>historic_earthquakes.zip</URL>
```

References the data file (compressed or uncompressed) that contains the earthquake data.

Each record in the data file should contain the following fields without any additional characters:

1. [Double] Latitude (in degrees)
2. [Double] Longitude (in degrees)
3. [Double] Elevation (in metres – negative indicates subsurface)
4. [Double] Magnitude
5. [Long] Timestamp (in milliseconds since epoch 01/01/1970 00:00:00 UTC)

The URL can be an absolute path, or can specify a path relative to the layer definition file.

<Coloring>

Specifies the colouring to apply to the data points.

Valid values are:

- Date
- Magnitude
- Depth

Default is “Magnitude”.

<ColoringMinDate>

```
<ColoringMinDate value="1950-01-01" format="yyyy-MM-dd" />
```

Only used when <Coloring> is “Date”. Specifies the start date to use when colouring the data points. Useful for restricting the colour map to a specific date range to enhance visualisation of data.

Optional. If not provided, will use the earliest occurrence date in the dataset.

ATTRIBUTE	DESCRIPTION
value	The date value, specified in the format described with the <code>format</code> attribute.
format	Specifies what format the date value is provided in. See the Standard Unicode Date Elements ^[3] for more details on date format symbols.

<ColoringMaxDate>

```
<ColoringMaxDate value="1970-01-01" format="yyyy-MM-dd" />
```

Only used when <Coloring> is “Date”. Specifies the end date to use when colouring the data points. Useful for restricting the colour map to a specific date range to enhance visualisation of data.

Optional. If not provided, will use the latest occurrence date in the dataset.

Layer Definition File Specification

ATTRIBUTE	DESCRIPTION
value	The date value, specified in the format described with the <code>format</code> attribute.
format	Specifies what format the date value is provided in. See the Standard Unicode Date Elements ^[3] for more details on date format symbols.

<PointSize>

```
<PointSize>2</PointSize>
```

Specifies the size (in pixels) of the points used to render earthquake occurrences on the globe.

Values must be doubles in the range [0, ∞).

Default value is 1.

CRUST LAYERS

```
<Layer version="1" layerType="CrustLayer">
```

Crust layers are a specialised sub-surface layers that display a surface deformed and coloured by elevation data read from a data file. These surfaces are used to represent crustal features that exist below the surface of the earth.

Crustal data files are comma- or whitespace-separated lists of elevation values expressed as doubles (in metres). Normally these values are all negative (indicating sub-surface values). The values are stored in a row-major ordering^[18] of dimensions width x height.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="CrustLayer">
  <DisplayName>Crustal Ice</DisplayName>
  <URL>map2.t2.zip</URL>
  <Size width="180" height="90" />
  <Scale>1000</Scale>
  <Sector>
    <SouthWest>
      <LatLon units="degrees" latitude="-89" longitude="1" />
    </SouthWest>
    <NorthEast>
      <LatLon units="degrees" latitude="89" longitude="359" />
    </NorthEast>
  </Sector>
  <Wrap>true</Wrap>
</Layer>
```

<URL>

```
<URL>map2.t2.zip</URL>
```

References the data file that contains the elevation data for the layer. The data file may be compressed within a zip archive to reduce bandwidth requirements.

The URL may be an absolute path, or may specify a path relative to the layer definition file.

<Size>

```
<Size width="180" height="90" />
```


Layer Definition File Specification

Defines the number of columns (width) and rows (height) in the dataset.

The data file should contain data in a row-major ordering of dimensions width x height.

ATTRIBUTE	DESCRIPTION
width	The number of columns in the dataset. Integer in the range [1, ∞).
height	The number of columns in the dataset. Integer in the range [1, ∞).

<Scale>

```
<Scale>1000</Scale>
```

Specifies an exaggeration to apply to the elevation data. This can be useful to exaggerate features in the data. Note that the crust surfaces are also affected by vertical exaggeration adjustments that can be made inside various World Wind tools.

Valid values are doubles in the range (0, ∞).

<Sector>

```
<Sector>
  <SouthWest>
    <LatLon units="degrees" latitude="-89" longitude="1" />
  </SouthWest>
  <NorthEast>
    <LatLon units="degrees" latitude="89" longitude="359" />
  </NorthEast>
</Sector>
```

Defines the geographical extents of the crustal layer.

For more information on defining a <Sector>, see the <Sector> element definition for the Tiled Image Layer type.

<Wrap>

```
<Wrap>true</Wrap>
```

Whether or not the crustal layer should be wrapped around the globe.

Only sensible when the crustal geographical extents encompass close to the full 360° longitude range.

Valid values are:

- true
- false

Default value is "false".

SKYSPHERE LAYERS

```
<Layer version="1" layerType="SkysphereLayer">
```

The Skysphere layer adds a hemispherical skysphere to the globe which is mapped with a specified image.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="SkysphereLayer">
  <DisplayName>Skysphere</DisplayName>
  <TextureLocation>sky10.png</TextureLocation>
  <Rotation>83</Rotation>
  <Segments>30</Segments>
  <Slices>30</Slices>
</Layer>
```

<TextureLocation>

```
<TextureLocation>sky10.png</TextureLocation>
```

Specifies the URL of the image to use as the skysphere. This image should be provided in a dome projection to ensure correct display of the image within the world window.

Any valid image format may be used. Currently, supported formats are:

- image/dds
- image/gif
- image/jpg
- image/png
- image/tga
- image/tif

The provided URL may be an absolute path, or may specify a path relative to the layer definition file.

<Rotation>

```
<Rotation>83</Rotation>
```

Specifies the rotation (in degrees) of the image texture as it is applied to the skysphere dome. This rotation can be used to fine-tune what part of the sky is visible in what direction on the globe (for example, to position a sunset sky image such that the sun is in the west).

Valid values are doubles in the range $(-\infty, \infty)$. Default value is 0. Rotation will be applied modulo 360°.

<Segments>

```
<Segments>30</Segments>
```

Specifies the number of segments the skysphere should be rendered with. Segments are the divisions along the circle profiles used to generate the hemisphere.

A higher number will result in a smoother sphere, but may affect render times.

Valid values are integers in the range $(0, \infty)$. Default value is 20.

<Slices>

```
<Slices>30</Slices>
```

Specifies the number of slices the skysphere should be rendered with. Slices are the number of circle profiles used to generate the hemisphere.

A higher number will result in a smoother sphere, but may affect render times.

Valid values are integers in the range $(0, \infty)$. Default value is 20.

FOG LAYERS

```
<Layer version="1" layerType="FogLayer">
```

Fog layers add a coloured fog that changes in density as it gets further away from the viewing camera. Fog can be useful to hide defects in data, or to focus the viewer’s attention on objects in the foreground. By using coloured fogs it can also be used to simulate snow, haze, water and dust etc.

Fog density is related to the altitude of the camera and the distance of the camera to the horizon and can be adjusted using the <NearFactor> and <FarFactor> values.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="FogLayer">
  <DisplayName>testName</DisplayName>
  <NearFactor>1.2</NearFactor>
  <FarFactor>1.3</FarFactor>
  <FogColor red="123" green="23" blue="222" />
</Layer>
```

<NearFactor>

```
<NearFactor>1.2</NearFactor>
```

Adjusts the starting point of the fog as a multiplier of the camera altitude.

The inclusion of camera altitude in the calculation of fog density ensures that fog density decreases as the camera elevation is increased.

Valid values are doubles in the range [0, ∞). Default value is 1.1

<FarFactor>

```
<FarFactor>1.3</FarFactor>
```

Adjusts the ending point of the fog as a multiplier of the camera-horizon distance.

Valid values are doubles in the range [0, ∞). Default value is 1.1

<FogColor>

```
<FogColor red="123" green="23" blue="222" />
```

Specifies the colour of the fog.

ATTRIBUTE	DESCRIPTION
red	The red channel. Integer in the range [0, 255]. Optional. Defaults to 0.
green	The green channel. Integer in the range [0, 255]. Optional. Defaults to 0.
blue	The blue channel. Integer in the range [0, 255]. Optional. Defaults to 0.

Note that the <FogColor> element is very similar to the common <Color> element, but does not include an attribute for an alpha channel.

GEOMETRY LAYERS

```
<Layer version="1" layerType="GeometryLayer">
```

Geometry layers are a general-use layer for presenting vector data that extends the functionality of the Surface Shape and Point layers.

Geometry layers can be used to display point, line and polygon data from shapefiles, and can be configured to display this data as

- Terrain-hugging vector graphics (as in the Surface Shape layer),
- Points, lines or polygons drawn at a fixed offset above the terrain,
- Curtains that follow the vector shapes; or
- 3D airspace volumes defined by polygonal data.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="GeometryLayer">
  <DisplayName>Marine Boundaries</DisplayName>
  <RenderType>Airspace</RenderType>
  <URL>eez_limit.zip</URL>
  <DataFormat>Shapefile</DataFormat>
  <ShapeType>Polygon</ShapeType>
  <Styles>
    <Style name="defaultStyle" default="true">
      <Property name="TerrainConforming" value="true|true"/>
      <Property name="Altitudes" value="30000|100000"/>

      <Property name="EnableAntialiasing" value="true"/>
      <Property name="EnableBlending" value="true"/>
      <Property name="EnableLighting" value="false"/>
      <Property name="EnableDepthOffset" value="false"/>
      <Property name="EnableLevelOfDetail" value="false"/>

      <Property name="DrawCurtain" value="true"/>

      <Property name="DrawInterior" value="true"/>
      <Property name="Opacity" value="0.5"/>
      <Property name="Material" value="0,255,0"/>

      <Property name="DrawOutline" value="false"/>
      <Property name="OutlineWidth" value="3"/>
      <Property name="OutlineMaterial" value="255,0,0"/>

      <Property name="DrawUpperShapeOutline" value="true"/>
      <Property name="DrawLowerShapeOutline" value="true"/>
      <Property name="DrawExtents" value="false"/>
      <Property name="DrawWireframe" value="false"/>
    </Style>
  </Styles>
</Layer>
```

<URL>

```
<URL>eez_limit.zip</URL>
```

References the data source for the shape data. Only required when <DataFormat> is not "XML".

The URL can be an absolute path, or can specify a path relative to the layer definition.

<DataFormat>

```
<DataFormat>Shapefile</DataFormat>
```

Layer Definition File Specification

Specifies the format the shape data is to be provided in.

Currently, valid values are:

- `Shapefile` – Shape data is sourced from an (optionally compressed) ESRI shapefile^[11]. Attributes will be read from attribute tables within the shapefile. The shape type will be deduced from the shapefile.
- `XML` – Shape data is provided within the layer definition itself. This is only recommended for small, non-changing datasets (e.g. capital cities etc.).

<ShapeType>

```
<ShapeType>Polygon</ShapeType>
```

Specifies the shape type when `<DataFormat>` is “XML”, and overrides the shape type when “Shapefile” is used.

This is an optional element when the `<DataFormat>` is “Shapefile”. If not specified, the shape type will be deduced from the contents of the shapefile.

Valid values are:

- `Line`
- `Polygon`

<RenderType>

```
<RenderType>Airspace</RenderType>
```

Specifies the object type to use when rendering the vector shape data. Currently, the only supported render type is “Airspace”. However, this render type is very flexible and can be configured using the styles and attribute mechanisms to draw lines, filled polygons, curtains and airspace volumes.

Valid values are:

- `Airspace`

<Styles>

```
<Styles>
  <Style name="defaultStyle" default="true">
    <Property name="TerrainConforming" value="true|true"/>
    <Property name="Altitudes" value="30000|100000"/>
    ...
  </Style>
</Styles>
```

Geometry layers use the same style and attribute mechanism that the Point layer does. For more information on the `<Styles>` element, refer to the Point Layer specification.

Available style properties are determined by the `<RenderType>` used. Below is a list of some of the style properties available for customisation within the Geometry layer. For a more detailed list, consult the Javadoc for the classes listed in the table for a specific render type.

PROPERTY	DESCRIPTION
Airspace	
• <code>gov.nasa.worldwind.render.airspaces.AirspaceAttributes</code>	
• <code>gov.nasa.worldwind.render.airspaces.AirspaceRenderer</code>	
• <code>au.gov.ga.worldwind.common.layers.geometry.types.airspace.ShapeOutlineCurtain</code>	

<ul style="list-style-type: none"> gov.nasa.worldwind.render.airspaces.Polygon 	
EnableAntialiasing	<p>Enable or disable antialiasing for the drawn shapes.</p> <p>Antialiasing will produce smoother looking lines and polygons, but may decrease performance if not supported by the user's hardware. Default value is false.</p> <p>Type: Boolean</p>
EnableBlending	<p>Enable blending for the shape. Blending is used to enhance antialiasing and for drawing non-opaque objects.</p> <p>Default value is true.</p> <p>Type: Boolean</p>
EnableLighting	<p>Whether or not to apply lighting effects to the drawn shapes.</p> <p>If disabled, a flat constant ambient shading will be applied to the shapes. If enabled, a point light source will be used to apply lighting effects as a simulated sun. This may improve depth perception etc. of the shape.</p> <p>Default value is false.</p> <p>Type: Boolean</p>
DrawWireframe	<p>Whether to draw the shape as a wireframe.</p> <p>Default value is false.</p> <p>Type: Boolean</p>
DrawInterior	<p>Whether to draw the filled interior of the shape.</p> <p>Applicable only for <code>Polygons</code>. This controls (a) whether the side curtains are drawn, and (b) whether the top cap is drawn (based on the value of the <code>EnableCaps</code> property).</p> <p>Default value is true.</p> <p>Type: Boolean</p>
DrawOutline	<p>Whether to draw the shape outline.</p> <p>Default value is false.</p> <p>Type: Boolean</p>
Material	<p>Specifies the colour to use when drawing the interior of the shape (if applicable).</p> <p>Type: Material</p>
OutlineMaterial	<p>Specifies the colour to use when drawing the outline of the shape (if applicable).</p> <p>Type: Material</p>
Opacity	<p>Specifies the opacity to use when drawing the interior of the shape (if applicable).</p> <p>Valid values are in the range [0, 1]. Default is 1.</p>

Layer Definition File Specification

	Type: Double
OutlineOpacity	<p>Specifies the opacity to use when drawing the outline of the shape (if applicable)</p> <p>Valid values are in the range [0, 1]. Default is 1.</p> <p>Type: Double</p>
OutlineWidth	<p>Specifies the width (in pixels) of the shape outline (if applicable).</p> <p>Valid values are in the range [0, ∞). Default is 1.</p> <p>Type: Double</p>
DrawUpperShapeOutline	<p>Whether to draw the upper shape outline.</p> <p>Defaults to false.</p> <p>Type: Boolean</p>
DrawLowerShapeOutline	<p>Whether to draw the upper shape outline.</p> <p>Type: Boolean</p>
DrawCurtain	<p>Whether or not to draw the curtain strips for the Line shapes. Analogous to the “DrawInterior” property for polygons.</p> <p>Defaults to true.</p> <p>Type: Boolean</p>
Altitudes	<p>Specifies the altitudes at for the top and bottom shapes. This controls where the outline shapes are drawn, and the height of curtains (if applicable).</p> <p>Altitudes are expressed in metres.</p> <p>Type: Double Double (bottom top)</p>
TerrainConforming	<p>Specifies whether the top and/or bottom shapes should be terrain conforming.</p> <p>If true, the shape will be adjusted according to the elevation model beneath it, maintaining a fixed offset specified by the <code>Altitudes</code> property.</p> <p>If false, the shape will be drawn at a fixed offset from a datum and will not be affected by the underlying elevation model.</p> <p>Defaults to false false.</p> <p>Type: Boolean Boolean (bottom top)</p>
EnableLevelOfDetail	<p>Whether to enable level-of-detailing for the shape. This may help to speed up render times if the shape contains a lot of points.</p> <p>Defaults to true.</p> <p>Type: Boolean</p>

Layer Definition File Specification

EnableCaps	Whether to draw the caps (top and bottom) for Polygon shapes. Defaults to false. Type: Boolean
------------	--

<Attributes>

The attributes mechanism for Geometry layers is the same as for Point layers. For more information, refer to the Point layer definition.

BOREHOLE LAYERS

```
<Layer version="1" layerType="BoreholeLayer">
```

Borehole layers are used to display boreholes/wells on the globe. Currently points can be provided via a shapefile. The borehole layer supports the notion of styles, and has powerful mechanisms for mapping borehole attributes to colours and text labels etc.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="BoreholeLayer">

  <DisplayName>BH_STRATIGRAPHY</DisplayName>
  <Legend></Legend>

  <URL>BH_STRATIGRAPHY_wgs84.zip</URL>
  <DataFormat>Shapefile</DataFormat>

  <LastUpdate></LastUpdate>
  <DataCacheName>BH_STRATIGRAPHY_wgs84.zip</DataCacheName>

  <UniqueBoreholeIdentifier>HOLE_ID</UniqueBoreholeIdentifier>
  <SampleDepthAttributes from="FROM_" to="TO" />
  <LineWidth>10</LineWidth>
  <MinimumDistance>1e4</MinimumDistance>

  <BoreholeStyles>
    <Style name="borehole" default="true">
      <Property name="Material" value="128,128,128" />
      <Property name="ShapeType" value="gov.nasa.worldwind.render.markers.Sphere" />
      <Property name="MinMarkerSize" value="0.5" />
    </Style>
  </BoreholeStyles>

  <BoreholeAttributes>
    <Attribute name="HOLE_ID">
      <Text value="&lt;p&gt; &lt;b&gt;% value% &lt;/b&gt; &lt;/p&gt;" placeholder="% value%" />
    </Attribute>
  </BoreholeAttributes>

  <SampleStyles>
    <Style name="default" default="true">
      <Property name="Color" value="145,0,131" />
    </Style>
    <Style name="Menindee">
      <Property name="Color" value="0,255,197" />
    </Style>
    <Style name="Coonambidgal">
      <Property name="Color" value="0,168,132" />
    </Style>
    <Style name="Lunette">
      <Property name="Color" value="255,85,0" />
    </Style>
  </SampleStyles>
</Layer>
```


Layer Definition File Specification

```
<Style name="Woorinen">
  <Property name="Color" value="255,255,115" />
</Style>
<Style name="Shepparton">
  <Property name="Color" value="255,170,0" />
</Style>
<Style name="Redbank">
  <Property name="Color" value="168,112,0" />
</Style>
<Style name="Blanchetown">
  <Property name="Color" value="223,115,255" />
</Style>
<Style name="Chowilla">
  <Property name="Color" value="232,190,255" />
</Style>
<Style name="Calivil">
  <Property name="Color" value="211,255,190" />
</Style>
<Style name="Loxton-Parilla">
  <Property name="Color" value="170,255,0" />
</Style>
<Style name="Renmark Group">
  <Property name="Color" value="0,112,255" />
</Style>
<Style name="Upper Renmark">
  <Property name="Color" value="190,210,255" />
</Style>
<Style name="Middle Renmark">
  <Property name="Color" value="122,142,245" />
</Style>
<Style name="Lower Renmark">
  <Property name="Color" value="158,170,215" />
</Style>
<Style name="Palaeozoic">
  <Property name="Color" value="245,122,182" />
</Style>
<Style name="aquifer">
  <Property name="Color" value="0,77,168" />
</Style>
</SampleStyles>

<SampleAttributes>
  <Attribute name="STRAT">
    <Case value="Menindee" style="Menindee" />
    <Case value="Coonambidgal" style="Coonambidgal" />
    <Case value="Lunette" style="Lunette" />
    <Case value="Woorinen" style="Woorinen" />
    <Case value="Shepparton" style="Shepparton" />
    <Case value="Redbank" style="Redbank" />
    <Case value="Blanchetown" style="Blanchetown" />
    <Case value="Chowilla" style="Chowilla" />
    <Case value="Calivil" style="Calivil" />
    <Case value="Loxton-Parilla" style="Loxton-Parilla" />
    <Case value="Renmark Group" style="Renmark Group" />
    <Case value="Upper Renmark" style="Upper Renmark" />
    <Case value="Middle Renmark" style="Middle Renmark" />
    <Case value="Lower Renmark" style="Lower Renmark" />
    <Case value="Palaeozoic" style="Palaeozoic" />
    <Case value="aquifer" style="aquifer" />

    <Text value="Stratigraphy: &lt;b&gt;% value%&lt;/b&gt; &lt;br&gt;" placeholder="% value%" />
  </Attribute>

  <Attribute name="SECTION">
    <Text value="Section: &lt;b&gt;% value%&lt;/b&gt; &lt;br&gt;" placeholder="% value%" />
  </Attribute>
```

Layer Definition File Specification

```
<Attribute name="THICKNESS">
  <Text value="Thickness: &lt;b&gt;% value%&lt;/b&gt; &lt;br/&gt;" placeholder="% value%" />
</Attribute>
</SampleAttributes>
</Layer>
```

<DataFormat>

```
<DataFormat>Shapefile</DataFormat>
```

Defines the format the borehole data is to be supplied in.

Currently, valid values are:

- **Shapefile** – Borehole data is supplied in a shapefile referenced in the <URL> element. This shapefile may be zipped.

<URL>

```
<URL>BH_STRATIGRAPHY_wgs84.zip</URL>
```

When <DataFormat> is **Shapefile**, this element provides the location of the shapefile to use. The URL can be provided in absolute or relative format, and can refer to an uncompressed .shp file, or a compressed .zip archive containing the shapefile (recommended in most cases).

<UniqueBoreholeIdentifier>

```
<UniqueBoreholeIdentifier>HOLE_ID</UniqueBoreholeIdentifier>
```

Defines the attribute/column which uniquely identifies a single borehole. Borehole shapefiles can contain multiple samples for a single borehole, each sample representing a different sample type. The shapefile should contain an attribute which identifies which borehole each sample is from. Generally all samples from the same borehole should also have the same lat/long point.

This is used by the layer to mark individual boreholes with markers.

<SampleDepthAttributes>

```
<SampleDepthAttributes from="FROM_" to="TO" positive="true" />
```

Defines the attributes/columns which provide sample depth information. The ‘from’ attribute should contain the shapefile attribute which provides the upper depth for each sample; the ‘to’ attribute should contain the shapefile attribute which provides the lower depth. The ‘positive’ attribute defines whether the depth samples are defined in positive depth or are negative (ie height values).

<LineWidth>

```
<LineWidth>10</LineWidth>
```

Borehole samples are rendered as lines. This property allows control over the line width.

<MinimumDistance>

```
<MinimumDistance>1e4</MinimumDistance>
```

This property defines the minimum distance from the borehole that will cause the borehole to be rendered. If the camera distance to the borehole is larger than this value, the borehole samples will not be rendered (however, the borehole surface marker will always be rendered).

<BoreholeStyles>

```
<BoreholeStyles>
  <Style name="color1" default="true">
    <Property name="Material" value="128,128,128"/>
    <Property name="ShapeType" value="..." />
    ...
  </Style>
  ...
</BoreholeStyles>
```

Styles are used to control the look-and-feel of the individual borehole markers as they are displayed on the globe.

The <Styles> element may contain 0-many <Style> elements. Styles may be selected by selector elements listed in an <Attributes> section, or based on the 'default' property of child <Style> elements.

<Style>

```
<Style name="color1" default="true">
  <Property name="Material" value="128,128,128"/>
  <Property name="ShapeType" value="..." />
  ...
</Style>
```

Defines a single style. A style contains multiple <Property> elements that can be used to control particular properties of the borehole marker's look-and-feel.

ATTRIBUTE	DESCRIPTION
name	The name used to identify this style. Style names can be any string value, but must be unique within a layer definition.
default	Whether or not this is the default style to use for this layer. Valid values are: <ul style="list-style-type: none"> • true • false Optional. Defaults to false.

<Property>

```
<Property name="Surface"
  value="128,0,128|0.5"
  type="Color|Double"/>
```

Sets a value for a particular property for the parent style. The properties are applied to the borehole marker object.

For example, if a <Property> element is specified with name="Opacity", the style mechanism will look for a method setOpacity(opacity) on the target object(s), and will set it to the value provided in the value attribute.

ATTRIBUTE	DESCRIPTION
name	The name of the property this element is controlling.

	<p>The property name should match a JavaBean^[13] property name on the target object (target objects are listed below).</p> <p>A list of commonly used properties is given below.</p>
value	<p>A pipe-separated list of values that correspond to arguments of the setter method for the names property.</p> <p>Each value may be a compound type with components separated by commas. For example, colour values are expressed as 1,3 or 4 comma-separated integers (e.g. “255,0,45”).</p>
type	<p>A pipe-separated list of types that correspond to the types of the arguments provided in the value attribute.</p> <p>Optional. If not provided, the style mechanism will attempt to derive the correct type and coerce the value into that type.</p> <p>If provided, there must be a type provided for each element specified in value.</p> <p>Valid types are:</p> <ul style="list-style-type: none"> • String • Integer • Float • Long • Double • Character • Byte • URL • File • Color (‘grey’, ‘r,g,b’ or ‘r,g,b,a’, integers in range [0,255]) • Insets (‘t,l,b,r’, integers) • Dimension (‘w’ or ‘w,h’, integers) • Point (‘xy’ or ‘x,y’, integers) • Font (eg. ‘Arial-bold-12pt’)^[14] • Material (As for Color) • Boolean

The most commonly used properties available for styling are given below. Note that this is not an exhaustive list. For a complete list, look at the Javadoc documentation for the classes that correspond to the point types listed in the table.

PROPERTY	DESCRIPTION
Marker	<ul style="list-style-type: none"> • <code>gov.nasa.worldwind.render.markers.MarkerAttributes</code>
Material	<p>Controls the diffuse colour to apply to the marker.</p> <p>Type: Material</p>
ShapeType	<p>Specifies the fully qualified class name of the marker shape to use for markers controlled by the style.</p> <p>Classes must implement the <code>MarkerShape</code> interface.</p> <p>Some example shape types are:</p> <ul style="list-style-type: none"> • <code>gov.nasa.worldwind.render.markers.Sphere</code> • <code>gov.nasa.worldwind.render.markers.Cube</code> • <code>gov.nasa.worldwind.render.markers.Cone</code>

	<ul style="list-style-type: none"> gov.nasa.worldwind.render.markers.Cylinder
	Type: String
Opacity	Controls the opacity of the marker. Values must be in the range [0, 1]. Type: Double
MinMarkerSize	Markers may grow and shrink based on the distance from the camera to the marker. This property controls the minimum size the marker may be. Type: Double
MaxMarkerSize	Markers may grow and shrink based on the distance from the camera to the marker. This property controls the maximum size the marker may be. Type: Double

<BoreholeAttributes>

The attributes mechanism for Borehole layers is the same as for Point layers. For more information, refer to the Point layer definition.

<SampleStyles>

The <SampleStyles> element defines the styling for the borehole samples (the sample lines which are drawn underneath the borehole marker). They use the same mechanism as the <BoreholeStyles>, but act on a BoreholeSample instance.

PROPERTY	DESCRIPTION
BoreholeSample	
<ul style="list-style-type: none"> au.gov.ga.worldwind.common.layers.borehole.BoreholeSample 	
Color	Controls the colour to apply to the borehole sample line. Type: Color

<SampleAttributes>

The attributes mechanism for Borehole layers is the same as for Point layers. For more information, refer to the Point layer definition.

MODEL LAYERS

```
<Layer version="1" layerType="ModelLayer">
```

Model layers are used to display georeferenced models (surfaces, lines, volumes) on the globe. Currently GOCAD TSurf, PLine and Voxet files are supported.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer version="1" layerType="ModelLayer">

  <DisplayName>GWMAR_NBC_INVERSION_40_40_01m.vo</DisplayName>
  <Legend></Legend>

  <URL>GWMAR_NBC_INVERSION_40_40_01m.zip</URL>
  <DataFormat>GOCAD</DataFormat>
```

Layer Definition File Specification

```
<LastUpdate></LastUpdate>
<DataCacheName>GWMAR_NBC_INVERSION_40_40_01m.zip</DataCacheName>

<MinimumDistance>1e5</MinimumDistance>

<ByteOrder>littleEndian</ByteOrder>
<BilinearMinification>true</BilinearMinification>
<Subsampling u="8" v="4" w="8" />
<DynamicSubsampling enabled="true" samples="50" />

<PointSprite>true</PointSprite>
<PointMinSize>2</PointMinSize>
<PointMaxSize>1000</PointMaxSize>
<PointSizeAttenuation constant="0" linear="0" quadratic="1E-12d" />

<CoordinateSystem>EPSG:28354</CoordinateSystem>

<ColorMap interpolateHue="true">
  <Entry value="0.01831667" red="255" green="0" blue="255" alpha="10" />
  <Entry value="0.745760085" red="0" green="255" blue="128" alpha="10" />
  <Entry value="1.4732035" red="255" green="0" blue="0" alpha="255" />
</ColorMap>

</Layer>
```

<DataFormat>

```
<DataFormat>GOCAD</DataFormat>
```

Defines the format the model data is to be supplied in.

Currently, valid values are:

- GOCAD – Model data is supplied in a GOCAD file referenced in the <URL> element. This file may be zipped.

<URL>

```
<URL>GWMAR_NBC_INVERSION_40_40_01m.zip</URL>
```

When <DataFormat> is GOCAD, this element provides the location of the GOCAD file to use. The URL can be provided in absolute or relative format, and can refer to an uncompressed file, or a compressed .zip archive containing the file (recommended in most cases).

<MinimumDistance>

```
<MinimumDistance>1e5</MinimumDistance>
```

This property defines the minimum distance from the model that will cause the model to be rendered. If the distance from the camera to the model is larger than this value, the model will not be rendered.

<ByteOrder>

```
<ByteOrder>littleEndian</ByteOrder>
```

This property defines the byte order of the input data. Currently this only applies to Voxet data.

<BilinearMinification>

```
<BilinearMinification>true</BilinearMinification>
```

This property controls whether bilinear minification is used when down-sampling Voxet data.

<Subsampling>

```
<Subsampling u="8" v="4" w="8" />
```

This property defines the subsampling to use when reading Voxet data. The u,v,w attributes control the subsampling in the u,v,w Voxet axes. For example, a value of 8 means that every 8th sample is read in that axis. This property has no effect if DynamicSubsampling is enabled.

<DynamicSubsampling>

```
<DynamicSubsampling enabled="true" samples="50" />
```

Specifies if dynamic subsampling should be used when reading Voxet data. When dynamic subsampling is enabled, the Voxet reader will skip samples to attempt to read 'samples' samples per axis (or as close as possible).

<PointSprite>

```
<PointSprite>true</PointSprite>
```

Specifies whether the layer should use OpenGL point sprites when rendering Voxet data.

<PointMinSize>

```
<PointMinSize>2</PointMinSize>
```

Specifies the minimum size to render OpenGL points at.

<PointMaxSize>

```
<PointMaxSize>1000</PointMaxSize>
```

Specifies the maximum size to render OpenGL points at.

<PointSizeAttenuation>

```
<PointSizeAttenuation constant="0" linear="0" quadratic="1E-12d" />
```

Allows control over the OpenGL point size attenuation values.

<CoordinateSystem>

```
<CoordinateSystem>EPSG:28354</CoordinateSystem>
```

The GOCAD reader supports reprojection into WGS84 if the source projection is not WGS84. This property specifies the source projection of the data. The format could be a EPSG code, a Proj4 string, or a WKT string.

<ColorMap>

```
<ColorMap interpolateHue="true">
  <Entry value="0.01831667" red="255" green="0" blue="255"
    alpha="10" />
  <Entry value="0.745760085" red="0" green="255" blue="128"
    alpha="10" />
  <Entry value="1.4732035" red="255" green="0" blue="0"
    alpha="255" />
</ColorMap>
```

The ColorMap specifies how the model colours are rendered. The colours are interpolated between the provided colour entries depending of the data value at the vertex.

ATTRIBUTE	DESCRIPTION
interpolateHue	<p>Whether interpolation should be based on the Hue channel of the HSB colour space.</p> <p>If true, the Hue/Saturation/Brightness will be interpolated between the colours. If false, interpolation will occur on each channel in RGBA space.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • true • false

<Entry>

<Entry value="100" red="0" green="0" blue="10" alpha="50"/>

Defines a single entry in the colour map. Each entry maps a specific elevation to a colour in RGBA colour space.

ATTRIBUTE	DESCRIPTION
value/elevation	<p>The value or elevation this entry is for. Float value.</p> <p>Required.</p>
red	<p>The red channel of the colour. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
green	<p>The green channel of the colour. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
blue	<p>The blue channel of the colour. Integer in the range [0, 255].</p> <p>Optional. Defaults to 0.</p>
alpha	<p>The alpha channel of the colour. Integer in the range [0, 255].</p> <p>Optional. Defaults to 255.</p>

References

1. The NASA World Wind Java SDK, <http://worldwind.arc.nasa.gov/java/>
2. The OGC WMS specification, <http://www.opengeospatial.org/standards/wms>
3. Standard Unicode date elements, http://unicode.org/reports/tr35/tr35-6.html#Date_Format_Patterns
4. Online colour picker, <http://www.colorpicker.com/>
5. GIMP – The GNU Image Manipulation Program, <http://www.gimp.org/>
6. GIMP Color-to-alpha process, <http://manual.gimp.org/en/plugin-coloralpha.html>
7. Normal Mapping, http://en.wikipedia.org/wiki/Normal_mapping
8. Endianness (a.k.a Byte Ordering), <http://en.wikipedia.org/wiki/Endianness>
9. MIME media types, <http://www.iana.org/assignments/media-types/>
10. Mipmapping, <http://en.wikipedia.org/wiki/Mipmap>
11. Shapefiles, <http://en.wikipedia.org/wiki/Shapefile>
12. OpenGL stipple reference, <http://www.opengl.org/sdk/docs/man/xhtml/gLLineStipple.xml>
13. JavaBean conventions, <http://en.wikipedia.org/wiki/JavaBean>
14. Java Font reference, <http://download.oracle.com/javase/1.4.2/docs/api/java/awt/Font.html#decode%28java.lang.String%29>

Layer Definition File Specification

15. HTML escape characters, http://www.htmlescape.net/escape_common.html
16. HTML 4.0 Reference, <http://www.w3schools.com/tags/default.asp>
17. Regular expressions, <http://www.regular-expressions.info/reference.html>
18. Row-major ordering, http://en.wikipedia.org/wiki/Row-major_order
- 19.