



Defining Coordinate Systems in GTViewer 14

Overview

With **GTViewer** version 14, there is a change to the way your data's Coordinate System is defined. This change will affect anyone who does one or more of the following:

- Uses the GPS to Track your location
- Uses the Latitude/Longitude mode for the Coordinate Readout (on the status bar)
- Uses the Locate X/Y command with the Latitude/Longitude option.
- Uses the FromLatLong or ToLatLong methods in your code.

Previously, the **GPSInfoFile** entry in the General Info section of the .GTM file was defined to point to a gpsinfo.ini file which was created when you selected a coordinate system on the GPS Dialog. This process has been simplified somewhat and more directly integrated into the .GTM file as part of the data's configuration.

New Method

Now you must specify the coordinate system with the following entries in the [Additional Properties] section of the .GTM file. There are new 3 entries that can be used:

- **CoordSys** – This entry specifies the data's Coordinate System. Currently, two different specifications are accepted:
 - **ProjectionParameters(name)** The Name value will appear on the GPS Dialog. This specification type allows a PROJ.4 coordinate system parameter string to be specified with the CoordSysParam1 entry. The CoordSysParam2 entry can be used to specify the Lat/long projection (it will default to NAD83).
 - **GDA94(zone)** – This specification type will the Geocentric Datum of Australia projection. No parameters other than the zone are required.
- **CoordSysParam1** – this entry is used to specify the first parameter for Coordinate System if required. Currently, this entry is only used if the CoordSys entry uses the ProjectionParameters type and is set to a PROJ.4 parameter string:
- **CoordSysParam2** – optional – this entry is used to specify the second parameter which specifies the Datum for the Latitude and Longitude projection to use. By default, it is set to NAD83 for the ProjectionParameters type, but it can also be set to NAD27 and WGS84.

Example 1:

```
CoordSys=ProjectionParameters(NAD83 - Alabama East)

CoordSysParam1=+proj=tmerc +lat_0=30.5 +lon_0=-85.83333333333333 +k=0.99996 +x_0=199999.9999999999
+y_0=0 +ellps=GRS80 +datum=NAD83 +units=us-ft +no_defs

CoordSysParam2=NAD83
```

Example 2:

```
CoordSys=ProjectionParameters(WGS84-UTM Zone 13N-Int Ft)

CoordSysParam1=+proj=utm +zone=13 +ellps=WGS84 +datum=WGS84 +units=ft +no_defs

CoordSysParam2=WGS84
```

When using the **ProjectionParameters** option, the **CoordSysParam1** value is defined as a Proj.4 parameter string (for more info on Proj.4: <https://trac.osgeo.org/proj/>). There is also a website, <http://spatialreference.org/>, that can be used to get your coordinate systems specification. Use its Search option to find your coordinate system and then get the Proj.4 parameter string:

The screenshot shows a Firefox browser window displaying the Spatial Reference website. The address bar shows the URL spatialreference.org/ref/epsg/26929/. The page title is "Spatial Reference epsg projection 26929 - nad83 / alabama east". The navigation menu includes "Home", "Upload Your Own", "List user-contributed references", and "List all references". The breadcrumb trail shows "Previous: EPSG:26923: NAD83 / UTM zone 23N" and "Next: EPSG:26930: NAD83 / Alabama West". The main heading is "EPSG:26929" followed by "NAD83 / Alabama East (Google it)". A list of metadata includes:

- **WGS84 Bounds:** -86.7900, 31.0000, -84.8900, 35.0000
- **Projected Bounds:** 108635.9019, 55812.3670, 290090.6760, 499449.4104
- **Scope:** Large and medium scale topographic mapping and engineering survey.
- **Last Revised:** June 2, 1995
- **Area:** USA - Alabama - SPCS - E

A list of available formats is shown in a box, with "Proj4js format" circled in red:

- [Well Known Text as HTML](#)
- [Human-Readable OGC WKT](#)
- [Proj4](#)
- [OGC WKT](#)
- [JSON](#)
- [GML](#)
- [ESRI WKT](#)
- [.PRJ File](#)
- [USGS](#)
- [MapServer Mapfile](#) | [Python](#)
- [Mapnik XML](#) | [Python](#)
- [GeoServer](#)
- [PostGIS spatial_ref_sys INSERT statement](#)
- [Proj4js format](#)

Since the units default to meters, you may have to add a parameter to convert your coordinate:

```
+units=us-ft    => for U.S. Survey Feet
_units=ft      =>for International Feet
```

You can also provide the conversion factor:

+to_meter=0.3048006096012192 => For US Feet
+to_meter=0.304801 => for International Feet

Alternative ways to Specify Coordinate System

Since this new method of specifying the coordinate system is a fairly significant change to **GTViewer**, you may find that you have .gtx files or older sets of data that don't have this information in it, but you still want to use it with the current version of **GTViewer**. There are several options you can use to add this new information into your previous datasets:

- 1) You can create a text file with the same name as the .gtm or .gtx file with ".coordsys" on the end of it (demo.gtm would be demo.gtm.coordsys). The file will just contain the same entries that you would normally put in the .gtm file. This file will only be used if there is no coordinate information defined in the .GTM or .GTX file.
- 2) You can put a default file called **coordsys.ini** in the

`%appData%\Graphic Technologies Inc\GTViewer\CoordSys`

This file should contain the coordinate system entries that you would normally put in the .GTM file. It will be used if no other coordinate system information is found (in the .gtm or in a .coordsys file).

- 3) You can also use the addProp.ini file to add or override the coordinate system entries to the additional properties section, but this approach will only work with .GTX files, and will apply to all datasets in the same directory.

Support

If you want assistance getting the new parameters for your coordinate system, please contact us:

support@gti-us.com