Gpsinfo Specification Version 1.0alpha

Server

This section specifies how gpsinfo stores data on a server. The main idea is to index geospatial raster data in a simple manner, that does not require any further functionality from the server apart from serving text-based files. It will be the job of the client libraries to make the data accessible in an easy manner.

General

The raster data is stored in ESRI's ASC file format (http://resources.esri.com/help/9.3/arcgisdesktop/com/gp_toolref/spatial_analyst_tools/esri_ascii_raster_format.htm). Gpsinfo's indexing and configuration syntax follows the ASC format. Configuration is done in ASCII text files, encoded in UTF-8, and comprises

```
KEY VALUE
```

entries, one per line. KEY is a single word (without any space), that is seperated from the VALUE by at least a single space. VALUE spans until the end of line and may be a multi-word, e.g. comprise spaces.

Configuration at Service Level (gpsinfo_index.conf)

A gpsinfo service is published and known under its BASEURL (e.g. http://gpsinfo.org/service). A server may host multiple services with distinct BASEURLs. At the BASEURL, the configuration at server level is stored in a file called **gpsinfo_index.conf**, that comprises the following entries:

KEY	VALUE Description	VALUE Format
BASEURL	The service's base URL. Shall not end with a slash.	TEXT
VERSION	Version of the specification this service instance adheres to.	TEXT
LAYERS	Space separated list of layer names, yielding the relative path to the layer specific configuration	TEXT

```
BASEURL https://gpsinfo.org/service
VERSION 1.0
LAYERS AT_OGD_DHM_LAMB_10M AT_OGD_DHM_LAMB_10M_COMPRESSED
```

Configuration at Layer Level (gpsinfo_layer.conf)

At BASEURL/LAYERNAME/gpsinfo_layer.conf, the layer-specific configuration file is stored in the same general format as the server-level configuration. The data covers a rectangular region, that is broken down into rectangular tiles, that are all of the same size and that are organized in rows and columns. Rows and columns are indexed, starting with (0,0) in the lower left corner. The value of a cell yields the value at the cell center.

KEY	VALUE Description	VALUE Format
LAYERNAME	The layer's name. Shall be the same as in LAYERS at server level.	TEXT
VERSION	Version of the specification this service instance adheres to.	TEXT
DESCRIPTION	The layer's description.	TEXT
YEAR	The year the layer's data was acquired.	NUMERIC
SOURCE	The source the layer originates from, e.g. where does the data come from? Example: link where the data was downloaded.	TEXT
LICENSE	The data's license. If original data's terms require attribution of the copyright holder, include that here. Example: *This data set is made available by <COPYRIGHT_HOLDER> under the terms of the Creative Commons Attribution 4.0 International license*.	TEXT
EPSG	EPSG code of this layer's coordinate system	NUMERIC
UNIT	The coordinate system's unit (meter, degrees, …)	TEXT
ORIGIN_X	The origin is located at the lower-left corner. This key yields the origin's x coordinate in the layer's coordinate system.	NUMERIC
ORIGIN_Y	This key yields the origin's y coordinate in the layer's coordinate system.	NUMERIC
NR_TILES_X	Number of tiles in x direction, e.g. the number of tile columns.	NUMERIC
NR_TILES_Y	Number of tiles in y direction, e.g. the number of tile rows.	NUMERIC
NCOLS	Number of columns in any of the ASC files	NUMERIC
NROWS	Number of rows in any of the ASC files.	NUMERIC
CELLSIZE	All pixels are squares of this edge length.	NUMERIC
ATTRIBUTE_MAP	(optional) Name of a file in the layer's base path, resolving numeric attributes to string representations	TEXT
COMPRESSION	If TRUE, the ASC file is compressed (ZIP file format). If FALSE, the ASC file is not compressed.	TRUE/FALSE

```
LAYERNAME AT_OGD_DHM_LAMB_10M
VERSION 1.0
DESCRIPTION Example implementation of the gpsinfo specification
YEAR 2015-2019
SOURCE https://www.data.gv.at/katalog/dataset/dgm
LICENSE CC BY 4.0, (c) geoland.at
EPSG 31287
UNIT meter
ORIGIN_X 500000
ORIGIN_Y 500000
NR_TILES_X 5
NR_TILES_Y 4
NCOLS 190
NROWS 75
CELLSIZE 10
```

```
COMPRESSION FALSE
```

Attribute Map

An attribute map maps integer raster data to string representation. The main purpose of an attribute map is to safe storage with the raster data. The optional attribute map is located at BASEURL/LAYERNAME/ATTRIBUTE_MAP.

```
0 Burgenland
1 Carinthia
2 Lower Austria
3 Upper Austria
4 Salzburg
5 Styria
6 Tyrol
7 Vorarlberg
8 Vienna
```

Directory Structure

To work around possible limitations of the maximal number of allowed files per subdirectory, the tiles of a layer are organized in subdirectories. Each column is stored in subdirectory, named by the column's index. The tiles of a column are named by their row index.

http://gpsinfo.org/service/gpsinfo_index.conf - Configuration at service level

http://gpsinfo.org/service/AT_OGD_DHM_LAMB_10M/gpsinfo_layer.conf - Configuration at layer level

http://gpsinfo.org/service/AT_OGD_DHM_LAMB_10M/3/2.asc - Uncompressed tile in ASC file format of tile (2,3).

http://gpsinfo.org/service/AT_OGD_DHM_LAMB_10M_COMPRESSED/0/3.asc.zip - ZIP-compressed tile in ASC file format of tile (3,0).

Clients

Clients are available in different programming languages and realize easy access to the server data. We provide a specification of the client API in pseudo code that follows an object-oriented naming scheme. Arguments to a method are by default input only arguments. Keyword *Out* defines an argument as output only argument (e.g. its value on input is ignored). Keyword *InOut* defines an argument as input and as output argument (e.g. its value on input does matter). All methods except the property getters do return a string that is SUCCESS in case of success and that yields a meaningful error message otherwise.

Server Level - ServerInfo

```
String ServerInfo::connect(String BASEURL)
```

Loads and parses gpsinfo_index.conf from BASEURL.

```
String ServerInfo::baseurl()
```

Returns the BASEURL property.

```
String ServerInfo::version()
```

Returns the VERSION property.

```
StringArray ServerInfo::layers()
```

Returns all LAYERS available from this server.

Layer Level - LayerInfo

```
String LayerInfo::connect(ServerInfo serverInfo, String LAYERNAME)
```

Loads and parses BASEURL/LAYERNAME/gpsinfo_layer.conf.

```
String LayerInfo::layername()
```

Returns the LAYERNAME property. All other properties defined in gpsinfo_layer.conf are accessed accordingly. For the sake of brevetiy, we do not give the full list of getters here.

```
String LayerInfo::value(Method method, Real x, Real y, out Real value)
```

Queries the layer's value at the given coordinates. Method = {CLOSEST, INTERPOLATION}. CLOSEST yields the value of the closest grid point. INTERPOLATION performs bilinear interpolation from the grid cell's corner values. The value is returned in the output argument "value".

```
String LayerInfo::values(
      InOut Real xLowerLeft,
      InOut Real yLowerLeft,
      InOut Real xUpperRight,
      InOut Real yUpperRight,
      Out RealArray values)
```

Queries the layer's values in the interior or on the boundary of the rectangular region defined the given coordinates. On output, the coordinates yield the regions real coordinate bounding box. The values are returned in a row-major array, starting with the lowest line (from left to right).

```
String attribute(Real numericAttribute, Out String stringAttribute)
```

Translates the given numeric attribute into its string representation.

Acknowledgements

In 2019, gpsinfo receives funding from the Internet Privatstiftung Austria (IPA) within the netidee 2018 call.

License Information

Document History

Date	Author	Changes
1 Feburary 2019	Simon Flöry	First version of server-side spec
26 February 2019	Simon Flöry	First version of API spec
29 March 2019	Simon Flöry	Releasing version 1.0alpha of the specification