PROJ.4
Handling Coordinate Systems

Frank Warmerdam
OSGIS
June 2004

Objectives

- PROJ.4 background
- Coordinate systems background
- PROJ.4 parameters
- Defining common projections for PROJ.4
- Handling datums
- Using PROJ.4 dictionaries
- PROJ.4 gotchas and tips

PROJ.4 Origins

- Developed by Gerald Evenden at the USGS
- Supported dozens and dozens of projections
- Support for NAD27/83 datum shifting
- Originally placed in the public domain
- Fell out of maintenance circa 1995

PROJ.4 at RemoteSensing.org

- I renew maintenance circa 1999
- MIT/X licensing
- Hosted at RemoteSensing.org: web pages, CVS, Bugzilla, ftp, mailing list
- Ported to automake/libtool
- 4.4.x series of releases
- Added 3/7 parameter datum shifting
- New pj_transform() API for unified coordinate system to coordinate system conversion

PROJ.4 Directions

- Gerald Evenden now actively developing libproj4
- Libproj4 is stripped down to just projections
- I hope to produce a layered PROJ.4 library supplying datum/dictionary/etc services on top of libproj4 at some point
- No fixed timeline

PROJ.4 Based Packages

- GRASS
- MapServer
- Thuban
- PostGIS
- GDAL/OGR (OGRCoordinateTransformation)
- OGDI
- libgeotiff
- Various comercial packages.

Command Usage Example

Command:

```
cs2cs +proj=latlong +datum=WGS84
   +to +proj=utm +zone=11 +datum=WGS84
```

Input:

```
-118.0 33.0
```

Output:

```
406582.22      3651730.97 0.00
```

Geographic Coordinate Systems

eg. “ lat long, WGS84”

- Position as latitude (degrees north of equator) and longitude (degrees east of prime meridian)
- Ellipsoid (eg. Clark 1880, or WGS84)
  - Semi-major axis (center to equator, in meters)
  - Semi-minor axis (center to pole, in meters)
- Prime Meridian (normally Greenwich)
- Units (degrees, radians, gradians)
- Datum ....

Datum

- Based on an ellipsoid
- Roughly, a name for a survey network
- Surveys accumulate error
- MAGIC!
- Conversions done with:
  - Grid shift files (ie. NAD27/83)
  - 3/7 parameter transformations
  - Polynomials (not supported by PROJ)
- Conversions often expressed relative to WGS84
- Is WGS84 the universal datum?

Projected Coordinate System

eg. “ UTM zone 11 north, WGS84”
Location expressed in meters east/north of some reference location.

Needs:

- Projection method (ie. Transverse Mercator)
- Parameters (ie. Central Meridian, False Easting)
- Geographic Coordinate Systems (ie. WGS84)
- Linear units (ie. Meters, or feet)

PROJ.4: Linear Units

Defined as:

- +units=<unit_name>
- +to_meter=<multiple>

Examples:

- “ +units=m”
- “ +to_meter=1000.0” (kilometers)

“ cs2cs -lu” for listing of named units.

PROJ.4: Ellipsoid (Spheroid)

Defined as:

- +ellps=<name>
- +a=<semi_major_axis> +b=<semi_minor_axis>
- +a=<semi_major_axis> +rf=<inverse_flattening>

Axis defined in meters.

Examples:

- “ +ellps=WGS84”
- “ +a=6378137.0 +rf=298.257223563”

Use “ cs2cs -le” to get a list of known ellipsoids.

PROJ.4: Datums

Defined as:

- +datum=<datum_name>
- +towgs84=<x_shift>,<y_shift>,<z_shift>
- +towgs84=<xs>,<ys>,<zs>,<xr>,<yr>,<zr>,<s>
- +nadgrids=<list of grid shift files>

Examples:

- “ +datum=WGS84”
- “ +towgs84=-263.0,6.0,431.0 +ellps=clark80”
- “ +nadgrids=ntv1_can.dat +ellps=clrk66”

Use “ cs2cs -ld” to get a list of known datums.

PROJ.4: Projection Parameters

- +lon_0=<angle>
  - Central Meridian, Longitude of Origin, Center Long
- +lat_0=<angle>
  - Latitude of Origin, Center Latitude
- +k=<scale_factor>
- +x_0=<false_easting>
- +y_0=<false_northing>

Almost all projections have +lon_0, +x_0, +y_0.

PROJ.4: Projection Parameters

- +lat_1=<first_standard_parallel_lat>
- +lat_2=<second_standard_parallel_lat>
- +zone=<zone>

Transverse Mercator

Aka Gauss-Kruger

+proj=tmerc +lon_0=<central meridian>
+lat_0=<latitude of origin> +k=<scale factor>
+x_0=<false easting> +y_0=<false northing>

Example (UTM 11 North):

+proj=tmerc +lon_0=-117 +lat_0=0 +k=0.9996
+x_0=500000 +y_0=0 +datum=WGS84

Universal Transverse Mercator

Aka UTM
+proj=utm +zone=<zone>

Example (UTM zone in which Ottawa falls) :
+proj=utm +zone=17 +datum=WGS84

Just an alias for:
+proj=tmerc +lon_0=-81 +k=0.9996
+x_0=500000 +datum=WGS84

Lambert Conic Conformal (2SP)

+proj=lcc +lat_1=<1st std. Parallel>
+lat_2=<2nd std. Parallel>
+lat_0=<origin lat> +lon_0=<origin long>
+x_0=<false easting> +y_0=<false northing>

Example (Tennessee State Plane):

+proj=lcc +lat_1=35.25 +lat_2=36d25
+lat_0=34d40 +lon_0=-86
+x_0=609601.2192024384
+y_0=30480.06096012192
+datum=NAD27 +units=ft

Stereographic

Aka Oblique Stereographic
+proj=stere +lat_0=<origin lat>
+lon_0=<origin long> +k=<scale factor>
+x_0=<false easting> +y_0=<false northing>

Example (Poland Zone 1):
+proj=stere +lat_0=50d37.5 +lon_0=21d5
+k=0.9998 +x_0=4637000 +y_0=5647000
+ellps=krass +units=m
+towgs84=33.4,-146.6,-76.3,-0.359,-0.053,0.844,-0.84

Mercator

+proj=merc +lat_ts=<latitude of true scale>
  +lon_0=<central meridian>
  +k=<scale factor>
  +x_0=<false easting> +y_0=<false northing>
NOTE: Use of lat_ts instead of lat_0!

Example (Segara – Indonesia):
+proj=merc +lat_ts=0 +lon_0=110 +k=0.997
  +x_0=3900000 +y_0=900000
  +ellps=bessel +units=m

Geographic

Aka lat/long
+proj=latlong

- Not really a projection!
- Still need datum or at least ellipsoid.
- Can include prime meridian.
- Units is implicitly degrees.

More and more and more...

- PROJ.4 supports supports approximate 120 projection methods.
- Details for common ones are at: www.remotesensing.org/geotiff/proj_list/
- Additional information in the PROJ.4 PDFs.

PROJ.4 Dictionaries

- Common coordinate systems defined in dictionaries.
- Format: +init=<dictionary>:<name>
- Example: +init=epsg:4326
- Dictionaries are text files in /usr/local/share/proj
- Search them with a text editor!
- Declarations look like:

```
# WGS 84
<4326> +proj=longlat +datum=WGS84 +no_defs  <>
```

PROJ.4 Dictionaries Cont.

Distributed Dictionaries:

- epsg: Definitions for EPSG GCS and PCS.
- nad27: State plane zones keyed on USGS zone#
- nad83: State plane zones keyed on USGS zone#
- esri: ESRI extended “EPSG” database
- other.extra: OGC WMS “EPSG” extensions
- world: assorted additional common projections

Gotchas

- PROJ.4 *may* default to WGS84 ellpsoid if not given, be explicit!
- Aea and lcc projections have default standard parallels for USA ... use +no_def.
- Longitude signs matter, Ottawa is *west* of greenwich which is a negative longitude.
- Alternate axis orientation not supported.
- Did you download grid shift files?
- False easting/northing *always* in meters.
- Europeans do +towgs84 signs backwards.

Tips

- Test a known point with command line tools.
- Use -v flag with cs2cs to see actual values used.
- Verify datum shift is doing something.
- Are grid shift files being found?
- Set PROJ_DEBUG environment variable to see files accessed.
- Don't trust the “ epsg” dictionary, especially with regard to datum shifting and uncommon projections.

Conclusion

- Visit remotesensing.org/proj for more info.

Questions?