Vertical Datums: An Introduction and Software Review
Areas to Cover

- Theoretical Introduction
- Representation in EPSG
- Representation in OGC WKT
- Incorporation in PROJ.4
- Incorporation in GDAL
- Future Work
Introduction to Vertical Datums

• Basis for measuring elevation values.
• Generally “0” is roughly sea level
• Often local and national datums exist.
• Three Classes of Vertical Datums
  > Ellipsoidal
  > Orthometric
  > Tidal
Ellipsoidal Vertical Datums

- Height is measured from the ellipsoid.
- Height is normal to the ellipsoid surface
- Ellipsoid positioning determined by horizontal datum.
- Traditional GIS height model.
- eg. NAD83, WGS84, ITRF90
Tidal Datums

- Based on observations of water levels.
- Means computed over a stated time period (often a 19 year epoch)
- Affected by various physical factors (wind, etc)
- eg.
  - LMSL (Local Mean Sea Level)
  - MLW (Mean Low Water)
Orthometric Vertical Datums

- Height is measured from the Geoid
- Measured along a plumb line
- Many national and regional implementations:
  - NGVD29 (National Geodetic Vertical Datum 1929)
  - NAVD88 (North American Vertical Datum 1988)
  - IGLD85 (International Great Lakes Datum of 1985)
Geoid

- Equipotential gravity surface
- Least squares best fit with global mean sea level
- Undulates significantly due to mountains, rock density, etc.
- Varies by up to 100m from geocentric ellipsoid
- Various approximations available based on physical readings and interpolation.
- eg USA Geoid2003, global EGM 96
WGS-84 Geoid Height

From DMA 10 by 10 Degree Geoid Height Grid

Peter H. Dana 11/05/95
Ellipsoid, Geoid, and Orthometric Heights

\[ h = H + N \]

- \( h \) (Ellipsoid Height) = Distance along ellipsoid normal (Q to P)
- \( N \) (Geoid Height) = Distance along ellipsoid normal (Q to \( P_0 \))
- \( H \) (Orthometric Height) = Distance along plumb line (\( P_0 \) to P)
Vertical Coordinates in EPSG

Compound Coordinate Reference System (crs table):
- Horizontal CRS (ie. UTM 11 WGS84)
- Vertical CS (ie. NAVD88)

Vertical Coordinate Reference System (crs table):
- Vertical Datum
- Linear Units
- Axis (direction)
Vertical Coordinates in EPSG

Vertical Datums (datum table):
- Area of use, comments, etc.

Transformations:
- Based on Vertical CRS
- Often many defined for one Vertical CRS.
- Many transformation methods defined.
- No obvious "pivot vertical datum"
EPSCG Transformation Methods

- Simple offset (ie. Baltic Sea vs. Black Sea)
- Offset+Slope (spatial slope)
- Grid Shift Files (VERTCON, geoid, many distinct transformation methods)
- Others I don't understand and haven't reviewed
OGC Well Known Text Representation

COMPD_CS["OSGB36 / British National Grid + ODN", PROJCS["OSGB 1936 / British National Grid", GEOGCS["OSGB 1936", DATUM["OSGB_1936", ...

AUTHORITY["EPSG","27700"], VERT_CS["Newlyn", VERT_DATUM["Ordnance Datum Newlyn",2005, AUTHORITY["EPSG","5101"], UNIT["metre",1,AUTHORITY["EPSG","9001"], AXIS["Up",UP], AUTHORITY["EPSG","5701"], AUTHORITY["EPSG","7405"]]}
OGC Notes

- No TOWGS84[] comparable method to represent transformations.
- VERT_DATUM “type” has no equivalent in EPSG
- Ellipsoidal vertical datums not COMPD_CS
Liblas Vertical Datum Support

- Liblas is a LAS (Lidar format) library (liblas.org)
- Funding from US ACE via Howard Butler for vertical datum support
- Liblas uses PROJ.4, libgeotiff, GDAL for coordinate system handling
- Goal:
  > Las2las -t_srs NAD83+NAVD88 in.las out.las
Libgeotiff Changes

- Modify the “build_pcs.py” script to produce:
  - vertcs.csv – vertical crs + transformation
  - compdcs.csv – compound crs list
  - primarily for use of GDAL EPSG lookups
- Clarify representation of vertical crs:
  - VerticalCSTypeGeoKey
  - VerticalDatumGeoKey
  - No equivalent to compound crs
GDAL Changes

Complete:
- ImportFromEPSG() support for VERT_CS
- ImportFromEPSG() support for COMPD_CS
- gtx (vertical grid shift format) read/write

To Do:
- to/from Proj.4 vertical grid shift parameters
- WKT EXTENSION for vertical shift parameters
- VERT_CS, COMPD_CS verification
PROJ.4 Changes

Done:
- +geoidgrid= parameter
- .gtx vertical grid shift reader
- Ellipsoidal vertical datum conversion

To Do:
- Chaining +geoidgrid= ?
- Compound crs in epsg init file?
- Vertical offsets?
Notes

- WGS84 ellipsoidal as pivot vertical datum
- .gtx picked as vertical datum format
- Translate to gtx with GDAL
- Convert 0-360 longitude files to -180 to 180
- NOAA VDatum used extensively for tutorial materials, grid shift files and validation
Software Review

- Not Comprehensive, or Accurate!

CSMap (Autodesk/Mentor):
- Includes support for geoid using gtx
- Many gtx files distributed with software

GeoToolbox (Java):
- Support COMPD_CS WKT
- Geoid transform can be requested specifically (using spherical harmonics model)
Conclusions

- Ellipsoidal datums ok for many GIS purposes
- Geoid needed for many formal vertical datums (such as NAVD88)
- Vertical datums particularly important on coast
- Word underway to incorporate support in liblas, libgeotiff, GDAL and PROJ.4.