

Kleine Dokumentation der geotools

von frank334,
2.3.2007

In den TXT-Hilfetexten oder Skripts (mit Texteditor ansehen) finden sich evtl. weitere Erläuterungen und Beispiele. Weiterhin zeigen die meisten Programme ihre Optionen an, wenn sie falsch oder ohne Parameter aufgerufen werden. Scheint ein Programm nicht zu reagieren, wartet es sicherlich auf eine Eingabe (dann muss die Eingabedatei per Redirect eingespielt werden, z.B. *kommando < eingabe.txt > ausgabe.txt*). Der Vorteil dieser Filterprogramme ist, dass sie mit sog. Pipes verkettet werden können, z.B. *ovl-zu-nmea* als "*ovl2csv < route.ovl | csv2nmea > route-log.txt*".

Generelle Annahme: es liegen die 4 Kalibrierpunkte auf den 4 Ecken der Kachel. Einige Tools funktionieren nur unter dieser Voraussetzung. Wenn die 4 Punkte nicht genau auf der Ecke liegen, so werden sie u.U. automatisch dorthin verschoben. Mit mehr oder weniger als 4 Punkten funktioniert das nicht. Kacheln aus den automatischen Konvertern (GMM, NH-Toptrans) erfüllen üblicherweise diese Anforderung.

Karten-Kacheln selektieren

coord2selection kal-directory degree_latitude degree_longitude > mapfilenames

Karten-Dateinamen aus Verzeichnis heraussuchen, die eine Koordinate enthalten

kachelwahl ausgangsverzeichnis zielverzeichnis auswahlbereich.ovl bitmap-extension

Kacheln selektiv aus einem Verzeichnis in ein anderes kopieren.
Die geografischen Bereiche werden durch Flächen bzw. eingrenzende Polygone (auch mehrere)
im ASCII-OVL-Format gekennzeichnet (Geogrid-Overlay-Format, evtl. mit GPSBabel und csv2ovl
konvertieren)

gmfkachelwahl ausgangsverzeichnis karte.gmf auswahlbereich.ovl bitmap-extension
wie kachelwahl, jedoch Ausgabe in GMF-Kartendatei

gmf-selection area.ovl map-source.gmf map-destination.gmf
wie zuvor, jedoch Kacheln von einer GMF-Datei in eine andere GMF-Datei aussortieren

gmf-selection-remove area.ovl map-source.gmf map-destination.gmf
wie zuvor, jedoch werden die Kacheln innerhalb der Auswahlfläche entfernt

kachelwahl-route ausgangsverzeichnis zielverzeichnis radius routefile bitmap-extension

Kacheln selektiv aus einem Verzeichnis in ein anderes kopieren.
Alle Kacheln entlang einer Route wählen, die mit maximal radius Abstand zur Route liegen.
route: komma-separierte ASCII-Koordinatenliste ("latitude,longitude", wie GPSBabel CSV export,o.Kompass)
radius: Abstand/Radius von Kachel-Eckpunkten zur Routenlinie in Kilometern (km)
(d.h. 2*radius ist etwa die minimale Schneisenbreite, die entlang der Route mit Kacheln abgedeckt wird)
Bitmap-extension der Grafikdateien z.B. png, jpg oder gif

gmf-selection-route map-source.gmf map-destination.gmf routefile radius
wie zuvor, jedoch Kacheln von einer GMF-Datei in eine andere GMF-Datei aussortieren

zipkachelwahl karte.zip auswahlname auswahlbereich.ovl bitmap-extension
kacheln selektiv aus einem ZIP-Kartenarchiv in ein verzeichnis "auswahlname" kopieren
das anschliessend zu einem ZIP diesen Namens komprimiert wird
die geografischen Bereiche werden durch Flächen bzw. eingrenzende Polygone (auch mehrere)
im ASCII-OVL-Format gekennzeichnet (Geogrid-Overlay-Format, evtl. mit GPSBabel und csv2ovl
konvertieren)

zipgmfkachelwahl karte.zip karte.gmf auswahlbereich.ovl bitmap-extension
wie zipkachelwahl, jedoch wird eine GMF-Kartendatei erzeugt, mit KMZ-Übersicht

zipkachelwahl-route karte.zip auswahlname radius routefile bitmap-extension

Kacheln selektiv aus einem ZIP in ein anderes "auwahlname.zip" kopieren.
Alle Kacheln entlang einer route wählen, die mit maximal radius abstand zur route liegen.
route: komma-separierte ASCII-Koordinatenliste ("latitude,longitude", wie im GPSBabel CSV export)
radius: Abstand/Radius von Kachel-Eckpunkten zur Routenlinie in Kilometern (km)
(d.h. 2*radius ist etwa die minimale Schneisenbreite,
die entlang der Route mit Kacheln abgedeckt wird)
bitmap-extension z.B. bmp, jpg oder gif

kal-nobitmap filename.kal

Output: Bitmap-Name der Grafikdatei, falls sie nicht existiert
(dient dem Löschen von KAL-Dateien, deren Grafikdateien bereits gelöscht wurden,
z.B. weil sie bereits in der Größensortierung als leere Kacheln entfernt wurden.

cat *.kal | **kal2mapname** > mapnames

Generiert eine Liste von Bitmap-Namen, die zu den KAL-Dateien gehören

kal2selection-range long_min long_max lat_min lat_max < allmapfiles-stream
kal2selection-range2 long_min long_max lat_min lat_max < allmapfiles-stream

Listet Kacheln auf, die innerhalb eines geografischen Rechtecks liegen (2: vollständig drin).
Wurde mittlerweile durch die Polygonauswahl in kal2selection verbessert.

kal2selection areas.ovl < allmapfiles-stream > selected-bitmap-names
kal2selection2 areas.ovl < allmapfiles-stream > selected-bitmap-names

Listet alle Kachelnamen auf, die innerhalb der durch areas.ovl definierten Polygonflächen (Fläche oder umgebender Polygonzug) liegen. 2.Variante: die Kachel muss komplett in der Fläche liegen, ansonsten auch teilweise.

kal2selectionpt degree_lat degree_long < allmapfiles-stream > mapfilenames

Listet alle Kachelnamen auf, die die angegebene Koordinate enthalten

kals-deletenobitmap "*.kal"

Löscht alle KAL-Dateien, zu denen die Bitmap fehlt (wenn zuvor die leeren Bildkacheln entfernt wurden)

kals-nobitmap "*.kal"

Listet alle KAL-Dateien auf, zu denen die Bitmap fehlt

ovl2borders < ovlfile

Output: long_min long_max lat_min lat_max
Gibt ein begrenzendes Rechteck aus, dass um die Punkte aus dem OVL-File gelegt wird

ovl2selection kal-directory areas.ovl > mapfilenames
ovl2selection2 kal-directory areas.ovl > mapfilenames

Karten-Kacheln auswählen, die in den OVL-Polygonflächen liegen (2: komplett drin)

route2selection radius routefile < allmapfiles-stream

Radius: min. distance route points to map-tile corners (measure: [km])
Route: ASCII lat/log-coordinate-list (comma-separated, one coordinate per line)
collect map tiles along route, approx. width 2*radius

cat *.kal | gmfgenerate directory map.gmf

kals2gmf directory map.gmf

gmfmake map.gmf

GMF-Karte map.gmf mit allen Kacheln aus Verzeichnis erzeugen, bzw. aus aktuellem Verzeichnis

gmf-extract map.gmf directory

Extrahiert Kacheln aus einer GMF-Kartendatei in ein Verzeichnis ("." = aktuelles Verzeichnis)

gmf-add map1.gmf map2.gmf map-destination.gmf

Zwei GMF-Karten map1 und map2 in eine einzige GMF-Datei zusammenfassen

gmf-merge map1.gmf map2.gmf map-destination.gmf

Zwei GMF-Karten zusammenfassen. Dabei werden nur die Kacheln aus map2 übernommen, deren Gebiet noch nicht durch map1 abgedeckt ist

Karten-Kacheln bzw. deren Lage darstellen

coord2kml < route.txt > route.kml

coord2kml-points < route.txt > route.kml

Koordinaten-Liste (longitude,latitude) nach KML track/points konvertieren

coord2kmz, coord2kmz-points entsprechend für KMZ

kal2coord-list *.kal > points.csv

alle Kalibrierpunkte als CSV auflisten (longitude,latitude,"mapfilename")

kal2coord < inputfile.kal > outputfile.txt

Glopus Kalibrierpunkte nach Wegepunkte exportieren zur Visualisierung in Karte (longitude,latitude,"mapfilename")

kal2index

Alle Kalibrierpunkte nach 00index.kmz (Google KMZ) übertragen

kal2index2

Alle Glopus-Kacheln in ein GoogleEarth GroundOverlay 00index-overlay.kml übertragen (ca. max. 10 stück)

kal2index3

Alle Glopus-Kacheln in ein GoogleEarth SuperOverlay mit Hauptdokument 00index-overlay.kml übertragen

kal2index4

Alle Glopus-Kacheln in ein GoogleEarth SuperOverlay 00index-overlay.kmz einpacken (incl. Bitmaps)

kal2index5

Ein SuperOverlay KMZ mit Referenz auf die Grafikdateien erzeugen (absoluter Pfad)

gmf-extract-names map.gmf

Kachel-Namen ausgeben

gmf-stat map.gmf > stat.txt

Statistik über das GMF-File anzeigen

gmf2index map.gmf

Alle Kalibrierpunkte nach map.gmf-index.kmz (Google KMZ) übertragen

gmf2index2 map.gmf

GoogleEarth Superoverlay map.gmf-overlay.kmz aus den Kacheln erzeugen

kal2kml-list *.kal > points.kml
Alle Kalibrierpunkte extrahieren

kalzip2index mapfile.zip
mapfile.zip-index.kmz Übersicht der Kartenkacheln generieren

cat *.kal | kal2kmloverlay opacity > overlay.kml
Glopus-Kartenkacheln in ein GoogleEarth-KML-Kartenoverlay übersetzen

kal2kmzoverlay "*.kal" opacity overlay.kmz
Glopus-Kartenkacheln in ein GoogleEarth-KMZ-Kartenoverlay packen
mit Transparenzwert opacity [0..255]

cat *.kal | kal2kmlsuperoverlay opacity minLodPixels maxLodPixels prefix_kml prefix_image directory overlay.kml
gmf-kmlsuperoverlay map.gmf opacity minLodPixels maxLodPixels prefix_kml prefix_image directory overlay.kml
Kacheln als GoogleEarth Super-Overlay formatieren
Für jede Kachel wird ein html-file erzeugt (abgeleitet aus dem Bilddatei-Namen) und in directory gespeichert
Die KML-Kacheln werden mit prefix_kml und die Bilddateien relativ hierzu mit prefix_image referenziert
Auch absolute Pfadangaben und Netzwerk-Links sind mit prefix_* möglich.
Kalibrierpunkte sollten auf den 4 Ecken liegen.
Die Kacheln werden bei einer Pixelgröße (Wurzel aus Fläche) zwischen
minLodPixels maxLodPixels dargestellt. Z.b. minLodPixels=500 (halbe Kachelgröße), maxLodPixels=-1
Nicht zu viele Kacheln gleichzeitig darstellen (>100), sonst stürzt Google ab.

kal2kmlsuperoverlay-all "*.kal" opacity minLodPixels maxLodPixels prefix_kml prefix_image directory overlay.kml
kann auch sehr viele Dateien per "*.kal"-Wildcard bearbeiten (cat *.kal hat gewisse Limits)

kal2kmzsuperoverlay "*.kal" opacity minLodPixels maxLodPixels superoverlay.kmz
wie kal2kmlsuperoverlay, jedoch werden die Kacheln zusammen mit den Bildern in ein KMZ Overlay gepackt

gmf2kmzsuperoverlay map.gmf superoverlay.kmz opacity minLodPixels maxLodPixels
wie vorhin, jedoch aus GMF Kartendatei

Konvertieren der Kalibrier-Dateien (map2kal, WorldFiles etc.)

gmf-correct map-source.gmf map-destination.gmf
Fehlerhafte longitude/latitude-Werte korrigieren, Kalibrierpunkte auf Kachelecken verschieben
Zeichensatz der Kachelnamen vereinfachen (kein "=")

kal-correct-bmp2gif *.kal
Dateinamen in KAL korrigieren

kalscale scalefactor < input.kal > output.kal

Kalibrierdatei skalieren, z.B. scalefactor=0.5 für eine Halbierung,
nach 50% Resize der Grafik (bzw. 1/4 der Fläche)

kalscale-all scalefactor sourcedirectory targetdirectory

Alle Glopus-KAL-Dateien um Faktor skalieren und in targetdirectory speichern.
Das ist die Korrektur, nachdem die Bildkacheln entsprechend verkleinert wurden.

map-gif2png
map-tif2png
Alle Dateien des aktuellen Verzeichnisses von GIF/TIF nach PNG konvertieren und
Kalibrierdateien entsprechend anpassen.

map2kal-all *.map

map2kal-all.bat

Alle OziExplorer-Kalibrierdateien nach Glopus KAL konvertieren

map2kal < inputfile.map > outputfile.kal

Eine Ozi-MAP-Datei nach Glopus-KAL konvertieren

geotiff2kal map.tif > map.kal

geotiff2kal-all *.tif

geotiff2kal-utm map.tif > map.kal

geotiff2kal-utm-all *.tif

GeoTIFF-Kalibrierinformation nach Glopus KAL übertragen (nur geografisch, 4 Kartenecken)
separate Behandlung für Geografische- und UTM-Koordinaten

Hinweis:

bei GeoTiff gibt es sehr viele Kalibrier-, Projektions- und Datum-Parameter.

die Skripte werten vorerst nur die UTM/Zone sowie die eingebetteten Karteneckpunkte aus

sollten diese fehlen, ist eine Konvertierung nicht möglich, d.h. entweder eine Komplettierung
der GeoTiff-Information oder eine Anpassung der Skripte ist nötig

geotiffgcp2kal map.tif > map.kal

geotiffgcp2kal-all *.tif

wie geotiff2kal, jedoch werden die GeoTIFF GCP Stützpunkte ausgewertet

bsb2kal map.kap > map.kal

bsb2kal-all *.kap

BSB/KAP-Kalibrierdaten nach Glopus KAL übertragen

asc2kal mapfilename sizex sizey < calpoints.asc > map.kal

Glopus KAL aus ASCII-File generieren (Zeilen: longitude latitude pixel_x pixel_y)

world2kal "mapfilename" sizex sizey < map.??w > map.kal

World File Kalibrierdaten nach Glopus KAL konvertieren

world2pt sizex sizey < map.??w > map.points

World File Kalibrierdaten als Eckpunktkoordinaten ausgeben (für UTM-Umwandlung benötigt)

kal2world < map.kal > map.world

World File Georeferenz aus KAL file berechnen

kal2geotiff map.kal neu.tif "+proj=utm +zone=32 +datum=WGS84"

kal2geotiffw map.kal neu.tif "+proj=utm +zone=32 +datum=WGS84"

Karte mittels KAL-Datei in ein GeoTIFF umwandeln, zusätzliche Projektionsangaben in proj-Syntax

Die 1. Variante nutzt Kalibrierpunkte aus dem KAL als GCP Stützpunkte (genauer, aber ohne corner points)

und die w-Variante nutzt affine Transformationsparameter via WorldFile (Umrechnung)

kals2tfw *.kal

mehrere KAL-Dateien in tfw World Files umwandeln

tfw2kal mapname.tif

tfw2kal-all *.tif

tfw2kal-utm mapname.tif utmzone

tfw2kal-utm-all utmzone

TIFF WorldFile calibration nach Glopus KAL (einzeln/alle), mit UTM-Variante (manuell Zone ermitteln)

jgw2kal, jwg2kal-all, jwg2kal-utm

wie tfw2kal*, aber für JPG-Dateien

wld2kal mapname.png

wld2kal-all *.png

wie tfw2kal*, aber für verschiedene Grafiktypen (png,tif,jpg) und *.wld World Files

POI - Points of Interest

kml2poi < poi.kml > poi.asc
kml2poi-all *.kml
kml2poi-all1 *.kml > allpoi.asc
kmz2poi poi.kmz poi.asc
kmz2poi-all *.kmz

Google Placemarks KML/KMZ konvertieren zu ASCII-CSV POI File (einzeln/alle)
ASCII-CSV POI Format: longitude,latitude,"symbol"

poi2kml < poi.asc > poi.kml
poi2kml-all *.asc
poi2kml-all1 *.asc > allpoi.kml
poi2kml-single < poi.asc
poi2kmz poi.asc poi.kmz

ASCII-CSV POI File zu Google Placemarks KML/KMZ konvertieren (einzeln/alle)

ovl2poi < textpoints.ovl > poi.asc
ovl2kmlpoi < textpoints.ovl > poi.kml
OVL Textpunkte zu Glopus oder KML POI konvertieren

poi2selection areas.ovl < poi.asc > selected-poi.asc
poi2selection-all "files" targetdirectory area.ovl

POI anhand einer per OVL definierten Polygonfläche regional begrenzen/auswählen.
Entweder einzelne POI-Files oder alle im Verzeichnis selektiert in ein Zielverzeichnis kopieren.

gmaps2poi "label" < url.asc > poi.asc
poi2gmapsurl type zoomlevel < poi.asc > url.asc
GoogleMaps Links von/zur POI
type h(hybrid) k(keyhole sat) m(Karte), zoom=17 nah

Tracks und Routen

gpsbabel-batch type "files" input-type output-extension output-type options

Mehrere Dateien automatisch per GPSTools konvertieren
type: t,w,r (track/waypoint/route), I/O-Typ von GPSTools und die zugehörige Dateinamenserweiterung
Beispiel:
gpsbabel-batch "-t" "*.ovl" overlay kml "kml,points=0,line_color=ff0000ff" "-x transform,trk=wpt,del"
Alle OVL-Dateien im Verzeichnis nach GoogleEarth KML konvertieren

csv2ovl < track.csv > track.ovl
ovl2csv < track.ovl > track.csv

OVL-Dateien von/zur CSV-Dateien konvertieren.
Die ASCII-CSV-Dateien bestehen aus "latitude, longitude"-Zeilen.
Gelesen werden Textpunkte, Linien und Flächenecken. Geschrieben werden nur Polygonzüge.
Übrigens ist dieses CSV-Format mit dem TK-Format identisch, das Glopus einen Track-Import ermöglicht.

kml2csv < polygon-line.kml > coord.csv
csv2kml < coord.csv > polygon-line.kml

KML-Track/Waypoints von/nach CSV (lat,long) konvertieren

pth2csv < line.pth > line.csv

pth2kml < line.pth > line.kml
MagicMaps PTH-Pfade nach CSV oder KML konvertieren

kml2kmz file.kml file.kmz
kml2kmz-all *.kml

kml2nmea track.kml nmea-log.txt

KML Track zu NMEA GPS-Logdatei konvertieren

kml2ovl < track.kml > track.ovl
ovl2kml < track.ovl > track.kml

KML Track zu OVL track konvertieren (nur ein Track, der Erste im KML), und umgekehrt

kmz2kml file.kmz file.kml
kmz2kml-all *.kmz

nmea-clean < log.nmea > cleanlog.nmea

NMEA tracks säubern (nur GPRMC,GPGGA,GPVTG,GPGSA,GPGSV, fehlende Zeilenvorschübe korr.)

nmea2csv < nmea-log.txt > track.csv
csv2nmea < track.csv > nmea-log.txt

NMEA Track in CSV (lat,long) konvertieren, und umgekehrt

nmea2kml < nmea-log.txt > track.kml
nmea2kml-all *.txt
nmea2kml-date < nmea-log.txt > track.kml date
nmea2kmz nmea-log.txt track.kmz
nmea2kmz-all *.txt
nmea2kmz-date nmea-log.txt track.kmz date

NMEA GPS-Logdatei zu KML/KMZ Track/Linie konvertieren
Datum in yyyyymmdd-Format wird benötigt, wenn das NMEA-Log kein Datum aufgezeichnet hat.

nmea2ovl < nmea-log.txt > track.ovl
NMEA track zu Geogrid OVL Track

nmea-speedplot nmea-log.txt
zeichnet ein Geschwindigkeitsprofil des NMEA-Tracks mit Hilfe des freien Matlab-Clones Octave
diese Information ist Teil der GPRMC-Sequenz, die daher mit aufgezeichnet werden muss
(keine Leerzeichen im Dateinamen erlaubt, Octave ist ein Paket in Cygwin)

Bildverarbeitung

bmp2jpg options < pic.bmp > pic.jpg
tif2jpg options < pic.tif > pic.jpg
BMP/TIF-Grafik nach JPEG konvertieren
Optionen: -optimize -quality=xx (xx=1..100, 100 beste)

bmp2jpg-all options
tif2jpg-all options
alle BMP-Dateien im aktuellen Verzeichnis nach JPG konvertieren, Optionen wie bmp2jpg

tif2png < pic.tif > pic.png
tif2png1 image.tif

bsb2tif-all.bat *.kap

bsb2png-all.bat *.kap

BSB/KAP Bitmapgrafik nach TIF oder PNG konvertieren (setzt libbsb-tools voraus)

map-merge "*.png" reference.pgw output.tif

verbindet mehrere Einzelkacheln zu einer output.tif Gesamtkarte

Die Auflösung wird anhand eines Referenz-Worldfiles bestimmt

alle Grafiktypen von gdalwarp werden unterstützt (PNG, TIF, ECW, JPG) etc.

geotiffsplit mapname.tif sx sy

geotiffsplito mapname.tif sx sy "optionen"

teilt eine GeoTIFF referenzierte Karte in Kacheln der Größe (sx,sy), angepasst an geradzahlige Kachelzahl
die Kacheln sind getreu der Projektion wieder georeferenziert

d.h. die Ecken haben unabhängig von der Projektion wieder exakte Kalibrierpunkte

(die Glopus-geodetic-Projektion wird bei geteilten Mercatorkarten dadurch genauer)

Es wird gdal_translate aus der GDAL-Bibliothek benötigt

Optionen sind gdal_translate-Optionen, z.B. Kalibrierpunkte "-gcp pixel line easting northing"

Statt TIFF werden auch PNG, JPG und andere in GDAL zulässige Grafikdateien verarbeitet,

wenn zusätzlich die entsprechenden WorldFiles vorliegen (*.pgw, *.jgw, *.wld ...)

geoecwsplit mapname.ecw sx sy

geopngsplit mapname.png sx sy

geojpgsplit mapname.jpg sx sy quality

wie zuvor, jedoch per pgw/jgw Worldfiles georeferenzierte Karten, oder ecw-support

bei png/jpg wird auch in diesem Format gespeichert, bei jpg ist die JPEG-Qualitätsstufe anzugeben

Sonstige Scripts

bash Einstellungen

Batch-Dateien und Registry-Einstellungen mit denen die bash-shell komfortabel per
rechtem Mausklick im aktuellen Verzeichnis gestartet werden kann.

batchreplace "files" pattern replacementpattern

In Dateien "files" (auch Wildcard "*.kal", Anführungszeichen erforderlich)

eine Zeichenfolge durch eine andere ersetzen (z.B. Kartennamen in allen KAL-Dateien ändern)

files-moveno2nd "*.ext1" ext2 directory

Verschiebt Dateien mit Erweiterung ext1 in ein anderes Verzeichnis,

wenn die zugehörige Datei mit Erweiterung ext2 nicht existiert

(z.B. nach Löschung von Bilddateien überflüssige WorldFiles entfernen)

gmf-correct map-source.gmf map-destination.gmf

4 Kalibrierpunkte auf die Ecken setzen

Ungültige longitude/latitude-Werte korrigieren

Kachel-Namen korrigieren (kein "="-Sonderzeichen)

gmf-rename map-source.gmf map-destination.gmf "sourcepattern" "replacementpattern"

Kacheln nach Muster umbenennen

maps-rename "files" pattern replacementpattern

Glopus-Kacheln umbenennen, neue Dateinamen und neue KAL-Referenzen

files-rename "files" pattern replacementpattern

Dateien anhand von Mustern umbenennen (sed-Syntax)

filename-lowercase directory

rekursiv: Kleinschreibung für alle Dateinamen

filesize file1 file2 ...

filesizes file1 file2 ...

Dateigröße ausgeben (2.: mit Name)

csvswapll < in.csv > out.csv

CSV: longitude,latitude,"text"

longitude <-> latitude Einträge austauschen