Engineer

GDAL

et bibliotek til at læse og konvertere geodata

Hvad det er

GDAL, en forkortelse for Geospatial Data Abstraction Library og ofte skrevet GDAL/OGR, er open source-software, der åbner geodata gemt i ét format og skriver dem ud i et andet, og som omprojicerer dem undervejs, hvis man beder om det. Projektets egen dokumentation kalder det et »translator library«, et oversætterbibliotek.

Geodata findes i to former. Rasterdata er et gitter af celler: et satellitbillede, et luftfoto, en højdemodel. Vektordata er punkter, linjer og polygoner med egenskaber knyttet til sig: adresser, veje, matrikelskel. GDAL giver det program, der kalder det, én abstrakt model for hver form, og bag modellen ligger en driver pr. format. GeoTIFF, JPEG 2000 og NetCDF hører til rastersiden, Shapefile, GeoJSON, GeoPackage, KML og PostgreSQL-tabeller til vektorsiden, og der findes mange flere af begge slags. Et program skrevet mod GDAL læser dem alle gennem de samme kald.

Vektordelen hed i mange år OGR, og de to halvdele havde hver deres driverstruktur, indtil GDAL 2.0 slog dem sammen. Derfor bærer vektorprogrammerne stadig det gamle præfiks: `ogrinfo` beskriver et vektordatasæt, og `ogr2ogr` konverterer det. Arbejdet på biblioteket begyndte i slutningen af 1998 med Frank Warmerdam. Det udgives under en MIT-lignende licens og er et projekt under Open Source Geospatial Foundation, OSGeo.

Det bruges på tre måder. Fra kommandolinjen med programmer som `gdalinfo`, `gdal_translate` og `gdalwarp` til raster og `ogrinfo` og `ogr2ogr` til vektor. Siden version 3.11 findes der også én samlet `gdal` -kommando med underkommandoer som `gdal raster` og `gdal vector`, som projektet stadig betegner som foreløbig. Fra kode via bindinger til Python og andre sprog. Og inde i anden software: QGIS læser for eksempel sine vektorformater gennem

GDAL-driverne og stiller GDAL-programmerne til rådighed i sin Processing-værktøjskasse. En typisk konvertering fylder én linje, her en shapefil, der bliver til en GeoPackage og samtidig omprojiceres til den UTM-zone, der bruges i Danmark:

```
ogr2ogr -f GPKG -t_srs EPSG:25832 matrikler.gpkg matrikler.shp
```

I skrivende stund er den aktuelle udgave 3.13.3. Der er få konkurrenter på dette lag. ArcGIS har sine egne konverteringsværktøjer, og enkelte formater har små læsere for sig, men et team, der skal tage imod det, kunden nu engang sender, ender som regel med GDAL et sted i stakken, ofte fordi et bibliotek, det har valgt, bygger på det.

Brug begrebet i en beslutning

Brug »Hvad er GDAL?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Alt, der sker et sted — skibe, køretøjer, matrikler, sensorer — sat på et kort, I kan søge i, måle på og dele.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har gjort en geodatapipeline 50 gange hurtigere — 50x hurtigere geografiske datapipelines

Har gjort en geodatapipeline 50 gange hurtigere Kontakt os →

Hvorfor det betyder noget

Ethvert geodatasystem har en grænse, hvor data udefra kommer ind: en kommunes shapefiler, en landinspektørs GeoTIFF-filer, en samarbejdspartners GeoJSON-eksport, en KML-fil, som nogen har tegnet i en browser. Beslutningen her handler om, hvem der står ved den grænse. Det kan være en person, der klikker sig gennem et skrivebordsprogram, eller et script, der gør det samme hver gang. GDAL gør scriptet billigt at skrive.

En konvertering skrevet som en `ogr2ogr` - eller `gdalwarp` -kommando kan læses i review, ligge i versionsstyring og køres igen, når næste levering kommer. En konvertering lavet i hånden bliver lavet én gang, af den, der husker trinene. For data, der kommer i portioner måned efter måned, afgør den forskel, om kvaliteten holder.

Licensen tæller også. Der betales hverken pr. bruger eller pr. maskine, ingen leverandør skal godkende noget, og det samme værktøj kører på en bærbar, en byggeserver og i en container.

Den tredje grund er adgangen til lageret. GDAL's virtuelle filsystemer lader en driver læse data der, hvor de ligger. Med præfikset `/vsicurl/` læser den en fil over HTTP ved at bede om bestemte byteintervaller, så de fleste drivere kan læse et fjernt datasæt stykke for stykke uden at hente det hele, og `/vsis3/` gør det samme mod Amazon S3. Kombineret med en Cloud Optimized GeoTIFF, som er delt op i fliser, har sine egne oversigtsniveauer og er lagt ud, så antallet af forespørgsler holdes nede, kan et raster på mange gigabyte ligge i objektlager og stadig læses ét udsnit ad gangen.

Omkostningerne er også reelle. GDAL er et stort, nativt bibliotek med mange valgfrie afhængigheder, og hvilke drivere en bestemt installation indeholder, afhænger af, hvordan den er bygget. To maskiner, der begge »har GDAL«, kan altså åbne forskellige formater. Og det oversætter trofast, hvilket også betyder, at det trofast tager alle begrænsninger med fra det format, det bliver bedt om at skrive.

Sådan ser det ud i praksis

GDAL optræder i to projekter på dette site, i to forskellige roller.

Fra 2015 til 2022 var rollen teamlead for GIS-softwareudvikling med ansvar for mere end tredive GIS-projekter, som rollen fulgte helt ind i driften. De spatiale data havde hjemme i PostgreSQL med PostGIS, og hver gang de skulle skifte format, var det GDAL/OGR, der gjorde arbejdet: shapefiler, GeoJSON, GeoTIFF, KML og vektorfliser. QGIS og JOSM tog sig af selve dataarbejdet, og webkortene ovenpå var bygget med Mapbox GL og Leaflet. Bag den liste af formater lå opgaver som kortlægning i 2D og 3D, behandling af LiDAR-data, georektifikation, vektorisering samt kortlægning og navigation indendørs, og hvert projekt blev supporteret i produktion, efter det var leveret.

Fra 2022 til 2024, som dataingeniør på et storstilet geospatialt analyseprojekt, var GDAL et af værktøjerne til at finde og rette fejl. Spatiale fejl blev diagnosticeret og rettet med GDAL, QGIS og ArcGIS, og fordi de samme oprydningsopgaver blev ved med at vende tilbage, blev de automatiseret i Bash-scripts. PostGIS stod for de spatiale forespørgsler og indeks, procedurer i PL/pgSQL og Transact-SQL omformede data i PostgreSQL og SQL Server, og de

rensede data endte i interaktive visualiseringer bygget med Mapbox. Over 1.000 problemer blev løst, og arbejdet bidrog til en reduktion på 35 % i køretiden for spatiale forespørgsler.

Hvor det går galt

GDAL gør præcis, hvad det får besked på. De fleste problemer opstår, når det får besked på noget lidt andet end det, der var ment.

Punkter havner på det forkerte kontinent. Siden GDAL 3.0 respekterer biblioteket den akserækkefølge, som myndigheden bag et koordinatsystem har fastlagt, og for EPSG:4326, det velkendte system med bredde- og længdegrad, kommer breddegraden først. Kode skrevet til GDAL 2 sender længdegraden først, og det gør GeoJSON også. Blandes de to, bliver hvert par byttet om, og et punkt i København dukker op i Det Arabiske Hav. Vælg én rækkefølge ved programmets kant, og skriv den ned. I kode, der vil have længdegraden først, er den dokumenterede omskifter `SetAxisMappingStrategy(OAMS_TRADITIONAL_GIS_ORDER)` på den rumlige reference.

Kolonnenavne bliver klippet, og datoer mister klokkeslættet. En shapefil begrænser attributnavne til 10 tegn. GDAL forkorter længere navne, når det skriver, og når to forkortede navne falder sammen, skærer det dem ned til 8 tegn og sætter et løbenummer på, så `abcdefghijkl` bliver til `abcdefgh_1`. Formatet gemmer datoer uden klokkeslæt. GDAL anbefaler at holde `.shp`- og `.dbf`-filer på højst 2 GB af hensyn til samspillet med anden software. Æ, ø og å kan blive til tegnsuppe, hvis kodningen fejlfortolkes, når den `.cpg`-fil, der angiver tegnsættet, mangler. Skriv GeoPackage, hvor modtageren tillader det. Hvor en shapefil er et krav, så omdøb kolonnerne med vilje før konverteringen.

Et fjernt raster bliver hentet i sin helhed. Får GDAL en almindelig `https://`-adresse uden præfiks, vælger det sin HTTP-pseudodriver, som henter hele filen ind i hukommelsen, før nogen driver læser den. Det passer til en lille fil og er det forkerte valg til et landsdækkende ortofoto. Sæt `/vsicurl/` foran adressen, så driveren kun henter de byteintervaller, den har brug for, og gem store rastre som Cloud Optimized GeoTIFF, så intervallerne er få. Lad så filen være: redigerer man en COG på stedet, går dens layout i stykker, og siden GDAL 3.8 afvises et forsøg på at åbne en til opdatering, medmindre man udtrykkeligt tillader bruddet.

Tal i en CSV-fil kommer ind som tekst. CSV-driveren læser hvert felt som tekst, medmindre den får andet at vide, så en kolonne med indbyggertal sorteres alfabetisk, og en sum længere nede i kæden ender i en typefejl. Slå åbningsindstillingen `AUTODETECT_TYPE=YES` til, eller læg en `.csvt` -fil ved siden af CSV-filen med hver kolonnes type.

En opgradering ødelægger Python-importen. Pakken `gdal` på PyPI bygges mod den GDAL, der allerede er installeret på maskinen, så de to versioner skal stemme overens. Projektets installationsvejledning låser den ene til den anden:

```
pip install gdal[numpy]=="$(gdal-config --version).*"
```

Læs mere

- GDAL documentation — projektets egen beskrivelse af oversætterbiblioteket, dets drivere og licens.
- Programs — alle kommandolinjeværktøjerne, fra `gdalinfo` og `ogr2ogr` til den nye `gdal` -kommando.
- GDAL Virtual File Systems — `/vsicurl/` , `/vsis3/` og de andre præfikser til at læse data der, hvor de ligger.
- OGR Coordinate Reference Systems and Coordinate Transformation tutorial — med afsnittet om akserækkefølge og det, der ændrede sig i GDAL 3.0.
- COG – Cloud Optimized GeoTIFF generator — fillayoutet, oversigtsniveauerne og hvorfor redigering på stedet ødelægger dem.

Ordliste

Vælg ud fra jeres faktiske vilkår

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er GDAL?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/gdal/