Engineer

GDAL

bibliothèque qui lit et convertit des données géospatiales

Ce que c’est

GDAL, abréviation de Geospatial Data Abstraction Library et souvent écrit GDAL/OGR, est un logiciel libre qui ouvre des données cartographiques stockées dans un format et les écrit dans un autre, en les reprojetant au passage si on le lui demande. Sa documentation le présente comme une bibliothèque de conversion.

Les données spatiales prennent deux formes. Les données raster sont une grille de cellules : une image satellite, une photographie aérienne, un modèle altimétrique. Les données vectorielles sont des points, lignes et polygones accompagnés d’attributs : adresses, routes, limites parcellaires. GDAL fournit au programme qui l’appelle un modèle abstrait pour chacune de ces formes ; derrière ce modèle se trouve un pilote par format. GeoTIFF, JPEG 2000 et NetCDF ont leurs pilotes côté raster ; Shapefile, GeoJSON, GeoPackage, KML et les tables PostgreSQL ont les leurs côté vectoriel. Il en existe bien d’autres dans chaque catégorie. Un programme utilisant GDAL les lit tous par les mêmes appels.

La moitié vectorielle s’est appelée OGR pendant des années. Les deux moitiés ont conservé des structures de pilotes distinctes jusqu’à leur fusion dans GDAL 2.0. C’est pourquoi les programmes vectoriels portent toujours cet ancien préfixe : `ogrinfo` décrit un jeu de données vectorielles, et `ogr2ogr` le convertit. Frank Warmerdam a commencé à travailler sur la bibliothèque fin 1998. Elle est publiée sous une licence de type MIT et fait partie des projets de l’Open Source Geospatial Foundation, OSGeo.

Il existe trois façons de l’utiliser. La première est la ligne de commande, avec des programmes comme `gdalinfo`, `gdal_translate` et `gdalwarp` pour les rasters, `ogrinfo` et `ogr2ogr` pour les données vectorielles. Depuis la version 3.11, il

existe aussi une commande unique `gdal` avec des sous-commandes comme `gdal raster` et `gdal vector`, encore qualifiée de provisoire par le projet. La deuxième passe par le code, grâce aux liaisons pour Python et d'autres langages. La troisième passe par d'autres logiciels : QGIS, par exemple, lit ses formats vectoriels à travers les pilotes GDAL et propose les programmes GDAL dans sa boîte à outils de traitements. Une conversion typique tient en une ligne : ici, un shapefile devient un GeoPackage et est reprojeté dans la zone UTM utilisée dans tout le Danemark :

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

À la date de rédaction, la version courante est la 3.13.3. Peu d'outils sont en concurrence à ce niveau. ArcGIS dispose de ses propres outils de conversion, et certains formats ont leurs lecteurs légers dédiés. Mais une équipe qui doit accepter ce que lui envoie un client finit souvent par intégrer GDAL à son système, parfois à travers une bibliothèque qui en dépend.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que GDAL ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Tout ce qui se situe quelque part — navires, véhicules, parcelles, capteurs — sur une carte où chercher, mesurer et partager.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. Pipeline géographique accéléré de 50 fois — 50x d'accélération des pipelines de données géographiques

Pipeline géographique accéléré de 50 fois Contactez-nous →

Pourquoi c'est important

Tout système géospatial a une frontière par laquelle entrent les données extérieures : shapefiles d'une commune, GeoTIFF d'un géomètre, export GeoJSON d'un partenaire, fichier KML dessiné dans un navigateur. GDAL intervient dans le choix de ce qui assure ce passage : une personne cliquant dans un logiciel de bureau ou un script répétant les mêmes opérations à chaque livraison. GDAL rend ce script peu coûteux à écrire.

Une conversion formulée comme une commande `ogr2ogr` ou `gdalwarp` peut être relue, conservée dans le gestionnaire de versions et réexécutée à la livraison suivante. Une conversion manuelle est faite une fois, par la personne qui se

souvient des étapes. Pour des données livrées par lots, mois après mois, cette différence détermine si la qualité se maintient.

La licence compte aussi. Il n'y a ni redevance par poste ni autorisation d'un fournisseur à obtenir. Le même outil fonctionne sur un ordinateur portable, un serveur de compilation et dans un conteneur.

La troisième raison est l'accès au stockage. Les systèmes de fichiers virtuels de GDAL permettent à un pilote de lire les données là où elles se trouvent. Avec le préfixe `/vsicurl/`, il lit un fichier par HTTP en demandant des plages d'octets : la plupart des pilotes consomment ainsi un jeu de données distant morceau par morceau sans le télécharger entièrement. `/vsis3/` fait de même sur Amazon S3. Ajoutez un Cloud Optimized GeoTIFF, découpé en tuiles, contenant ses propres vues réduites et organisé pour limiter le nombre de requêtes : un raster de plusieurs gigaoctets peut rester dans un stockage objet et être lu fenêtre par fenêtre.

Les coûts sont réels eux aussi. GDAL est une grande bibliothèque native avec de nombreuses dépendances facultatives. Les pilotes présents dépendent de sa compilation : deux machines qui « ont GDAL » peuvent ouvrir des formats différents. Et sa conversion fidèle transmet tout aussi fidèlement les limites du format de sortie demandé.

À quoi cela ressemble en pratique

GDAL apparaît dans deux projets de ce site, avec deux rôles différents.

De 2015 à 2022, dans ses fonctions de responsable d'équipe de développement logiciel SIG, la fondatrice a assuré la livraison de plus de trente projets SIG, chacun accompagné en production après sa livraison. Ils reposaient sur PostgreSQL avec PostGIS pour les données spatiales. GDAL/OGR servait à transférer ces données entre formats : shapefiles, GeoJSON, GeoTIFF, KML et tuiles vectorielles. QGIS et JOSM servaient au travail sur les données elles-mêmes, et les cartes étaient construites avec Mapbox GL et Leaflet. Les deux moitiés de la bibliothèque étaient utilisées, puisque GeoTIFF est raster et les autres formats de cette liste sont vectoriels. Les projets allaient de la cartographie numérique 2D et 3D et du traitement LiDAR à la géorectification, à la vectorisation, à la cartographie intérieure et à la navigation.

De 2022 à 2024, alors que la fondatrice occupait un poste d'ingénieure données sur un projet d'analyse géospatiale à grande échelle, GDAL faisait partie des outils de détection et de correction des erreurs. Les erreurs spatiales étaient diagnostiquées et corrigées avec GDAL, QGIS et ArcGIS. Comme les mêmes nettoyages revenaient, ils ont été automatisés par des scripts Bash. PostGIS assurait les requêtes et les index spatiaux ; des procédures en PL/pgSQL et Transact-SQL transformaient les données dans PostgreSQL et SQL Server. Les données nettoyées alimentaient des visualisations interactives réalisées avec Mapbox. Avec GDAL parmi les outils, le nettoyage a résolu plus de 1 000 problèmes et a contribué à la réduction de 35 % des temps d'exécution des requêtes spatiales.

Où les problèmes surviennent

GDAL fait exactement ce qu'on lui demande. La plupart des difficultés commencent lorsque la demande diffère légèrement de l'intention.

Les points arrivent sur le mauvais continent. Depuis GDAL 3.0, la bibliothèque respecte l'ordre des axes fixé par l'autorité qui définit le système de coordonnées. Pour EPSG:4326, le système familier de latitude et longitude, la latitude vient en premier. Le code écrit pour GDAL 2 passe la longitude d'abord, tout comme GeoJSON. Mélangez les deux, et chaque paire est inversée : un point de Copenhague réapparaît dans la mer d'Arabie. Choisissez un ordre à la frontière du programme et explicitez-le. Pour le code qui veut la longitude en premier, le réglage documenté est `SetAxisMappingStrategy(OAMS_TRADITIONAL_GIS_ORDER)` sur la référence spatiale.

Les noms de colonnes sont tronqués et les dates perdent l'heure. Un shapefile limite les noms d'attributs à 10 caractères. GDAL raccourcit les noms plus longs à l'écriture. Si deux noms raccourcis entrent en collision, il les réduit à 8 caractères et ajoute un numéro de séquence : `abcdefghijkl` devient `abcdefgh_1`. Le format stocke les dates sans heure. GDAL recommande de garder les fichiers `.shp` et `.dbf` à 2 Go ou moins pour l'interopérabilité avec d'autres logiciels. Le texte danois ou cyrillique peut devenir illisible si son encodage est mal identifié en l'absence du fichier `.cpg` qui indique la page de codes. Écrivez du GeoPackage lorsque le destinataire l'accepte. Si un shapefile est exigé, renommez délibérément les colonnes avant la conversion.

Un raster distant est téléchargé en entier. Avec une simple adresse `https://` sans préfixe, GDAL choisit son pseudo-pilote HTTP, qui charge tout le fichier en mémoire avant qu'un pilote ne le lise. Cela convient à un petit fichier, mais pas à une orthophotographie nationale. Préfixez l'adresse par `/vsicurl/` pour ne récupérer que les plages d'octets nécessaires, et stockez les gros rasters en Cloud Optimized GeoTIFF pour réduire le nombre de ces plages. Puis laissez le fichier intact : modifier un COG sur place casse son organisation. Depuis GDAL 3.8, son ouverture en modification est refusée, sauf si cette altération est explicitement autorisée.

Les nombres d'un CSV arrivent sous forme de texte. Le pilote CSV lit chaque champ comme une chaîne, sauf instruction contraire. Une colonne de populations se trie donc alphabétiquement, et une somme en aval provoque une erreur de type. Définissez l'option d'ouverture `AUTODETECT_TYPE=YES`, ou livrez à côté du CSV un fichier `.csvt` qui précise le type de chaque colonne.

Une mise à niveau casse l'import Python. Le paquet `gdal` sur PyPI est compilé avec la bibliothèque GDAL déjà installée sur la machine : les deux versions doivent donc correspondre. Les consignes d'installation du projet les alignent :

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

Pour aller plus loin

- Documentation GDAL — la présentation par le projet de sa bibliothèque de conversion, de ses pilotes et de sa licence.
- Programmes — tous les outils en ligne de commande, de `gdalinfo` et `ogr2ogr` à la nouvelle commande `gdal`.
- Systèmes de fichiers virtuels GDAL — `/vsicurl/`, `/vsis3/` et les autres préfixes pour lire les données sur place.
- Tutoriel OGR sur les systèmes de référence de coordonnées et la transformation de coordonnées — notamment l'ordre des axes et les changements de GDAL 3.0.
- COG : générateur de Cloud Optimized GeoTIFF — l'organisation du fichier, ses vues réduites et les raisons pour lesquelles une modification sur place les compromet.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le processus de travail, les interfaces existantes et les utilisateurs qui ont besoin du changement. Nous pouvons définir une livraison testable et une transmission permettant à votre équipe d'assurer la maintenance des éléments livrés.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce que GDAL ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

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

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/gdal/