Engineer

ArcGIS

Esris kommersielle GIS-plattform

Hva det er

ArcGIS er navnet Esri gir en hel familie av geospatial programvare, ikke ett enkelt program, og delene er lettest å skille fra hverandre etter hvor hver av dem kjører.

På skrivebordet heter dagens program ArcGIS Pro, og systemkravene lister opp Microsoft Windows og ingenting annet. Før Pro kom ArcMap, som var en del av det som ble solgt som ArcGIS Desktop. Esris side om livssyklusen oppgir ArcMap som utfaset og i såkalt moden støtte, med 10.8.x som den siste serien av utgivelser.

På en server er midtpunktet i plattformen en portal: stedet der innhold, brukere, grupper og datatjenester opprettes og deles. Esri drifter én selv som ArcGIS Online, et abonnement levert som programvare som tjeneste, og en organisasjon kan kjøre sin egen med ArcGIS Enterprise. Data i en portal publiseres som tjenester, hver med sitt eget REST-endepunkt. En feature service gir tilgang til romlige og ikke-romlige data i lag (feature layers) og tabeller, og lar en klient legge til, oppdatere, slette og hente ut objekter. Tjenester for vektorfliser og kartfliser lagrer og leverer fliser, og en bildetjeneste behandler og setter sammen rasterdata fortløpende. Hvert lag har en egen adresse på formen `…/FeatureServer/0` og en `query`-operasjon som tar en `where`-betingelse, en liste med felt og et utdataformat: HTML som standard, JSON, GeoJSON eller protocol buffers.

På klientsiden publiserer Esri Maps SDK-er for JavaScript, Kotlin, Swift, Flutter, .NET, Qt og Java, og i tillegg programtillegg for Unity og Unreal Engine. Den samme utviklerdokumentasjonen dekker også åpne biblioteker som kan tegne ArcGIS-tjenester, nemlig Leaflet, MapLibre GL JS, OpenLayers og CesiumJS. For skripting finnes ArcPy, Python-pakken for geoprosesseringsverktøy og

automatisering av kart inne i ArcGIS Pro, ved siden av ArcGIS API for Python og ArcGIS REST JS for arbeid med portaler og tjenester utenfra. Tilgang til alt som ikke er offentlig, går gjennom API-nøkler, OAuth 2.0 eller en innlogget bruker.

På disk er formatet som følger ArcGIS-data, en file geodatabase: en katalog med navn som slutter på `.gdb`, der hver tabell ligger som `.gdbtable`-filer. Utenfor Esris egen programvare leser GDALs innebygde OpenFileGDB-driver slike databaser, har kunnet skrive og oppdatere dem siden GDAL 3.6 og har håndtert rasterlag i dem siden 3.7.

Bruk begrepet i en beslutning

Bruk «Hva er ArcGIS?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Alt som skjer på et sted — fartøy, kjøretøy, tomter, sensorer — lagt på et kart du kan søke i, måle på og dele.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Gjorde en geodatapipeline 50 ganger raskere — 50x raskere pipelines for geografiske data

Gjorde en geodatapipeline 50 ganger raskere Ta kontakt →

Hvorfor det er viktig

Når en organisasjon allerede bruker ArcGIS, er spørsmålet sjelden om den skal ta det i bruk. Spørsmålet er hvor grensen går: hvilke data som blir værende i portalen som tjenester, hvilke som kopieres til en database resten av stakken styrer, og hvilken vei endringer flyter mellom de to.

Innenfor den grensen kommer mye ferdig. Redigering, symbolsetting, deling med navngitte brukere og grupper, analysetjenester og verktøy for å bygge apper med lite kode virker alle mot den samme portalen, og ansatte som kan ArcGIS Pro, kan fortsette å arbeide slik de allerede gjør. Når et lag publiseres som en feature service, får alle andre systemer en adresse å lese det fra.

Grensen kan krysses fra begge sider, og det gjør valget til et reelt valg og ikke en innlåsing. En feature service snakker JSON og GeoJSON over vanlig HTTP. GDALs ESRIJSON-driver leser URL-en til en feature service som om den var en fil og blar gjennom resultatet på egen hånd, så `ogr2ogr` kan kopiere et driftet lag inn i PostGIS eller en GeoPackage med én kommando. Esris egen dokumentasjon viser Leaflet og MapLibre GL JS som tegner tjenestene. Den andre veien har GDAL selv kunnet skrive formatet file geodatabase siden 3.6, så data kan også gå tilbake i Esris eget format.

Det som koster innsats, er alt som lever på begge sider samtidig. Et lag kopiert ut av en portal er en kopi nummer to, og en kopi nummer to trenger en tidsplan, en eier og en måte å oppdage at den har glidd bort fra originalen. Skrivebordsprogrammet kjører bare på Windows og krever kommersiell lisens, og det avgjør hvem på et blandet team som kan åpne et prosjekt. ArcPy følger med ArcGIS Pro og ArcGIS Server, og selv installert i et annet Python-miljø trenger den ArcGIS Pro på maskinen, så en pipeline bygd på ArcPy kjører bare der ArcGIS gjør det.

Slik ser det ut i praksis

ArcGIS dukker opp i to prosjekter på dette nettstedet, og i ingen av dem utgjør det hele stakken.

Fra 2015 til 2022 hadde rollen som teamleder for utvikling av GIS-programvare ansvaret for over tretti GIS-prosjekter, hvert av dem fulgt fra utvikling til drift. I de prosjektene var ArcGIS noe kartene kunne kalle, og ikke grunnmuren. Webkartene, bygd med Mapbox GL og Leaflet, gikk noen ganger mot API-ene til ArcGIS eller HERE, og applikasjonslaget rundt dem var skrevet i JavaScript, Python, PHP og SQL. De romlige dataene bodde i PostGIS på PostgreSQL, og dataarbeidet skjedde i QGIS og JOSM, så ArcGIS sto ved siden av stakken som et sett API-er og ikke under den.

I et storskala prosjekt for geospatial analyse fra 2022 til 2024 hadde dataingeniørrollen ArcGIS i verktøykassen sammen med GDAL og QGIS, og med de tre i kombinasjon ble de romlige feilene i datasettene funnet og reparert. Også her var ArcGIS et verktøy for å rette data, og ikke stedet dataene bodde. De geografiske og tidsmessige dataene lå i PostgreSQL og SQL Server, der prosedyrer i PL/pgSQL og Transact-SQL transformerte dem, PostGIS sto for romlige spørringer og indeksering, og Bash-skript tok de gjentakende vaskeoppgavene. Resultatet var over 1 000 løste problemer og 35 % kortere kjøretid for romlige spørringer.

Hvor det går galt

Det meste av trøbbelet ligger på grensen mellom ArcGIS og alt annet, der hver side tar for gitt at den andre oppfører seg som den selv.

Nedlastingen av et lag stopper på et rundt tall. En feature service returnerer høyst `maxRecordCount` objekter per forespørsel, en grense som settes per tjeneste. Når en spørring treffer flere, setter svaret `exceededTransferLimit` til `true`, og en klient som leser objektene og overser flagget, sitter med et halvt lag som ser komplett ut. Der laget oppgir `supportsPagination`, blar klienten med `resultOffset` og `resultRecordCount`, eller med `resultPaginationToken`, til flagget kommer tilbake som `false`. GDALs dokumentasjon legger til at bladingen bare er pålitelig når resultatet er sortert på et felt som `OBJECTID`:

```
url='https://gis.example.org/arcgis/rest/services/Parcels/FeatureServer/0'
q='query?where=1=1&outFields=*&orderByFields=OBJECTID+ASC&f=json'
ogr2ogr -f GPKG parcels.gpkg "ESRIJSON:$url/$q"
```

Koordinatene kommer i meter der det var ventet grader. Uten parameteren `outSR` returnerer en spørring geometri i det romlige referansesystemet til tjenestens kart, som kan være et projisert system målt i meter. For GeoJSON-utdata sier referansen at resultatet fra 10.8 følger RFC 7946 bare når `outSR` er utelatt eller satt til 4326. Enhver annen verdi gir projiserte koordinater i en fil som fortsatt kaller seg GeoJSON, og et webkart som leser den, plasserer objektene langt fra der de hører hjemme. Be om `outSR=4326` hver gang mottakeren venter lengde- og breddegrad.

Et ArcMap-skript slutter å virke i ArcGIS Pro. ArcGIS Desktop, som omfatter ArcMap, brukte versjoner av Python 2, mens ArcGIS Pro kjører Python 3, et skifte Esri beskriver som bare delvis kompatibelt. I ArcPy ble modulen `arcpy.mapping` fjernet og erstattet av `arcpy.mp`. Nå som ArcMap er utfaset, er skript skrevet for den en flytting som bare venter på å skje. Verktøyet Analyze Tools For Pro i Pro markerer problemene som ville stoppe Python-kode eller Python-baserte geoprosesseringsverktøy fra å kjøre, og det lønner seg å gjøre den gjennomgangen med vilje før den dagen en maskin byttes ut og problemene dukker opp av seg selv.

En file geodatabase åpnes bare for lesing. GDAL før 3.6 kunne lese en file geodatabase gjennom OpenFileGDB, men bare skrive en gjennom den separate FileGDB-driveren, som trenger Esris FileGDB API-bibliotek når GDAL bygges. Rasterlag inne i en `.gdb` krever GDAL 3.7 eller nyere. Når en pipeline skriver en `.gdb` på én maskin og feiler på en annen, bør du sammenligne GDAL-versjonene og driverlistene før du gir dataene skylden.

Les mer

- Developer documentation — Esris kart over plattformen: SDK-er, portal- og datatjenester, og sikkerhet.
- Query (Feature Service/Layer) — hver parameter til en spørring, med `outSR`, utdataformatene og de tre måtene å bla på.
- Python migration from 10.x to ArcGIS Pro — hva som endret seg mellom Python 2 og 3, og i selve ArcPy.
- ArcMap Life Cycle — støttestatusen til ArcMap, og hvorfor 10.8.x er den siste serien.
- ESRI File Geodatabase vector (OpenFileGDB) — GDALs driver for `.gdb`-kataloger, med versjonene som la til skriving og støtte for raster.
- ESRIJSON / FeatureService driver — slik leser GDAL resultater fra en feature service, og slik virker den automatiske bladingen.

Ordliste

Velg ut fra deres faktiske rammer

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er ArcGIS?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/arcgis/