

Algorithms providers

Klasy Qgis pozwalające na zbudowanie nakładek, wywołujących natywne polecenia algorytmów QGIS ([qis](#)) lub innych aplikacji, w taki sposób, aby imitować algorytmy georzetwarzania QGIS, umieszczać te algorytmy w skryptach i modelach QGIS oraz ładować wyniki do środowiska aplikacji.

- **natywny (QGIS)** Dostarcza narzędzia zaimplementowane przez API QGIS oraz pluginy
- **GDAL** Dostarcza algorytmy będące nakładkami na aplikacje GDAL.
- **SAGA** Dostarcza algorytmy programu SAGA (nie wszystko działa)
- **GRASS** Dostarcza algorytmy programu GRASS GIS (większość nie działa)
- **OTB, TauDEM**

Skrypty tworzone w R

Qgis pozwala tworzyć skrypty w języku R oraz budować dla nich interface użytkownika, na innych zasadach niż skrypty tworzone w Pythonie:

https://docs.qgis.org/3.34/en/docs/training_manual/processing/r_intro.html

Data Provider

Kod, który umożliwia czytanie określonych formatów danych. Najczęściej jest to GDAL/OGR obsługujący większość formatów SIG, postgis i inne formaty geoprzestrzenne, dane w pamięci operacyjnej (memory), wybrane struktury kafelkowe (vectorTile i rasterTile) chmury punktów, sceny satelitarne oraz siatki 3D (mesh), WFS (wektor) i WCS (raster)

Data providers zostały wprowadzone z kilku powodów

- integracja systemu QGIS z algorytmami odstepnymi w innych aplikacjach
- unikanie nadmiarowego tworzenia kodu algorytów
- udostępnienie jednolitego systemu tworzenia skryptów geoprzetwarzania w systemie Windows

OSGeo4W

OSGeo to organizacja opiekująca się licznymi otwartoźródłowymi projektami z zakresu Geoinformacji i geoinformatyki. Flagowe projekt to QGIS, GRASS, GeoServer, SAGA oraz biblioteki: PROJ, GDAL, GEOS. Dostarcza również konsolę pozwalającą na uruchamianie skryptów.

<https://www.osgeo.org/projects/>

OSGeo4W (OSGeo for Windows) to instalator tworzący zintegrowane środowisko analityczne obejmujące wszystkie niezbędne komponenty, szczególnie biblioteki jak GDAL, tak aby unikać wielokrotnej instalacji zależności. Pozwala też na integrację oprogramowania, która, w przypadku osobnych instalacji była by utrudniona.

Obecnie instalacja QGIS zawiera wszystkie niezbędne komponenty wchodzące również w skład Osgo4w

Co to jest GDAL/OGR/OSR

GDAL jest wysokowydajną biblioteką napisaną C++ służącą do konwersji pomiędzy ponad 200 geoprzestrzennymi formatami rastrowymi i wektorowymi. Została wydana na licencji Open Source X/MIT przez Open Source Geospatial Foundation. Głównym zadaniem jest dostarczanie silnika uniwersalnego dostępu do danych geoprzestrzennych obecnie praktycznie dla wszystkich liczących się technologii i systemów informacji geoprzestrzennej

- Prezentuje pojedynczy abstrakcyjny model danych rastrowych i pojedynczy abstrakcyjny model danych wektorowych zgodny z OGC Simple Features i
- oferuje interfejs API dla różnych języków, takich jak C, C++, Python, Perl, C# i Java.
- Opiera się na silniku PROJ oraz formacie Well KNown Text (WKT) do reprojektacji danych
- Obsługuje duże formaty danych, powyżej 4GB

Zawiera również wiele przydatnych narzędzi wiersza poleceń (CLI) do tłumaczenia i przetwarzania danych, przystosowanych do zautomatyzowanej pracy w środowisku powłoki systemu UNIX/LINUX. Narzędzia te są przede wszystkim przeznaczone do zarządzania dużymi kolekcjami danych geoprzestrzennych.

najważniejsze narzędzia linii poleceń

GDAL

- gdalinfo
- gdalwarp
- gdal_translate
- gdaldem
- inne: gdal_merge, gdalbuildvrt, gdal_rasterize, etc.

OGR

- ogrinfo
- ogr2ogr
- ogrmerge, ogr_layer_algebra, etc.

gdalinfo

prezentuje formie tekstowej (format tekstowy, WKT lub JSON) informacje o pliku:

- informacje geoprzestrzenne: położenie, rozdzielczość, układ odniesienia
- wielkość pliku, pasma (*bands*), bloki i inne informacje
- statystyki rastra, histogram, NODATA, informację o kategoriach i tabeli kolorów (jeśli są)
- punkty kontrolne (jeżeli są)
- metadane

```
gdalinfo dem.tif
```

`gdalinfo --formats` zwraca informację o wszystkich formatach rastrowych obsługiwanych przez system

gdalwarp

Narzędzie, którego podstawowym zadaniem jest reprojekcja danych pomiędzy `s_srs` a `t_srs`, czyli układem odniesienia źródłowego i docelowego pliku. W ramach konwersji układu lubtego samego układu odniesienia może dokonywać zmian:

- rozdzielczości i rozmiaru
- zasięgu i przycięcia, w tym przycięcia przy pomocy pliku wektorowego
- przepróbkowania (z reguły połączone ze zmianą rozdzielczości)
- typu danych dla pasma i reprezentacji NODATA

```
gdalwarp -t_srs EPSG:4326 input.tif output.tif # konwersja do WGS
gdalwarp input.tif output.tif -s_srs EPSG:4326+5773 -t_srs EPSG:4979 # źródłowy srs
podajemy jeżeli dane nie mają pro_jekcji## Dostęp do narzędzi QGIS w skryptach Pytona
```

gdal_translate

Narzędzie, którego podstawowym zadaniem jest konwersja z jednego formatu rastrowego do drugiego. W ramach konwersji lub tego samego formatu może dokonać w większości tych samych czynności co [gdalwarp](#):

- rozdzielczości i rozmiaru
- zasięgu i przycięcia
- próbkowania (z reguły połączone ze zmianą rozdzielczości)
- typu danych dla pasma i reprezentacji NODATA
- pozwala na przeskalowanie wartości danych

```
gdal_translate -of GTiff input.jp2 output.tif # zamiana formatu jpeg2000 do geotiff.  
gdal_translate -projwin 258460 152400 258600 152200 input.tif output.tif # przycięcie  
rastra do okna zasięgu  
gdal_translate -tr 10 10 -r cubic input.tif output.tif # zmiana rozdzielczości do 10 x  
10 m z próbkowaniem sześciennym  
gdal_translate -scale 0 65535 0 255 input.tif output.tif # przeskalowanie wartości od  
16bit do 8bit
```

Podział zadań pomiędzy `gdal_translate` i `gdalwarp` nie jest jasny, część czynności się nakłada, narzędzia są nadmiernie rozbudowane, licznymi parametrami, ma to jednak swoje uzasadnienie

gdaldem

Jest zbiorem narzędzi przeznaczonym do podstawowych analiz terenu. Każde z narzędzi ma własny zestaw parametrów. Podstawowe analizy to:

- cieniowanie i mapa kolorów
- nachylenie i ekspozycja
- szorstkość i nierówność
- TPI - topographic position index czyli różnica względem średniej

```
gdaldem slope input.tif slope.tif  
gdaldem hillshade -s 2 -alg ZevenbergenThorne -az 180 input.tif hillshade.tif #  
przeskalowany, oświetlony z 180 stopni
```

Narzędzia `gdaldem` są bardzo uproszczone w porównaniu z innymi zawartymi np. w SAGA GIS

Inne narzędzia

- `gdal_merge.py` - tworzy mozaikę ze zbioru obrazów
- `gdalbuildvrt` - tworzy raster wirtualny zarówno w formie mozaiki (jedna warstwa) jak i stosu warst. Bardziej zaawansowany niż `gdal_merge`
- `gdal_create`, `gdal_edit.py` - tworzenie pustego zbioru danych lub edycja istniejącego
- `gdal_rasterize`, `gdal_polygonize.py`, `gdal_contour` - konwersja między formatami
- `gdal_grid`, `gdal2xyz` - interpolacja i zamiana w punkty
- `gdal_calc.py` - prosty kalkulator rastrów
- inne: <https://gdal.org/programs/index.html>

OGR i OSR

OSR to biblioteka odstarczająca funkcji dostępu do układów odniesienia, oparta o PROJ, będąca częścią GDAL. OGR do wersji GDAL 2.1 rozwijał się jako niezależny projekt, od wersji 2.1 jest pobiblioteką GDAL. Najważniejsze narzędzia to:

ogrinfo

Odpowiednik gdalinfo, zwraca informację na temat warstwy wektorowej, w tym również struktury danych atrybutowych

```
ogrinfo data.gpkg
```

ogr2ogr

Konwerter formatów łączący funkcjonalność `gdalwarp` i `gdal_translate` dla warstw wektorowych. Pozwala na mn:

- zmianę formatu danych
- usunięcie części pól atrybutowych
- selekcję obiektów na podstawie atrybutów i położenia przestrzennego
- przycinanie obiektów
- zmianę projekcji

```
ogr2ogr output.gpkg input.shp # output jest jako pierwsza
ogr2ogr -s_srs EPSG:4326 -t_srs EPSG:3857 output.gpkg input.gpkg # zmiana projekcji
ogr2ogr -where "\"POP_EST\" < 1000000" output.gpkg input.gpkg input_layer selekcja
ogr2ogr -spat -13.931 34.886 46.23 74.12 output.gpkg input_vector.gpkg # przycięcie
```

SAGA

System for Automated Geoscientific Analyses. Orginalnie rozwijana jako otwartoźródłowy projekt na Uniwersytecie w Getyndze. orginalna wersja dedykowanana dla systemu Windows, od wersji 2.0 projekt wieloplatformowy. Obsługuje większość zadań analiz geoprzestrzennych, wybrane operacje geostatystyczne i teledetekcyjne:

- dostęp do plików (GDAL i własne filtry)
- filtry przeszezenne
- interpolacja i geostatystyka
- klasyfikacja obrazów wielospektralnych i przetwarzanie obrazów
- symulacje
- analizy terenu (wiodące narzędzie)
- praca z wektorami i chmurami punktów
- wizualziacja 3D

Kiepski interface graficzny i narzędzia wizualizacji danych

Linia poleceń SAGA i Provider

W systemie Windows saga_cmd uruchamia konsolę. W systemie Linux i konsoli MacOS exeutor saga_cmd jest uruchamiany z linii poleceń. Wadą linii poleceń jest ograniczony dostęp do dokumentacji.

W praktyce wywołuje się kolejne krogi tworzenia interface, a SAGA podpowiada listę opcji do wyboru.

Składania mocno niespójna.

Pogrupowanie i nazewnictwo modułów oraz parametrów SAGA uruchamianych w QGIS nie jest zgodne z nazewnictwem oryginalnym

GRASS GIS

Geographic Resources Analysis Support System to zestaw programów (narzędzi) przeznaczonych do analiz goprzestrzennych. W przeciwieństwie do SAGA rozwijany był przez U.S. Army - Construction Engineering Research Laboratory jako narzędzie linii poleceń dla systemu Unix/Linux. Od wersji 4 jako projekt otwartoźródłowy, a od wersji 6 również multiplatformowe razem Interface użytkownika (mocno kontrowersyjnym). Oferuje pełen zestaw analiz GIS wewnątrz własnej bazy danych:

- zarządzanie danymi
- analizy rastrowe (mapcalculator)
- analizy wektorowe
- analizy 3D
- analizy obrazów (wybrane narzędzia)
- współpraca z bazami danych
- analizy czasowo-przestrzenne

Mocne strony GRASS

- praca w regionie
- `r.sun`
- `r.stream.*`
- `mapcalculator` i `r.neighbour`
- system wtyczek
- współpraca z R, obecnie wypierana przez Python

Każdy moduł w GRASS jest osobno skompilowanym programem. Oznacza to że nowe narzędzia tworzy się bez związku z istniejącymi, co ułatwia proces ich tworzenia i utrzymywania

koncepcja regionu (GRASS) i siatki (SAGA)

Zarówno GRASS jak i SAGA pracują we własnych środowiskach geoprocessingu. Oznacza to konieczność zaimportowania danych do środowiska pracy. Środowisko GRASS nazywa się `region`, środowisko SAGA `grid`. Obie koncepcje są do siebie podobne, określają rozmiar, zasięg i rozdzielczość siatki analizy oraz mają przypisane odniesienie geoprzestrzenne. Są jednak pomiędzy nimi różnice:

- GRASS może w locie zmieniać zasięg i rozdzielczość regionu, SAGA wymaga zbudowania osobnej siatki i skopiowania danych
- Ustawienia regionu są niezależne od danych, siatka SAGA tworzona jest w momencie importu danych
- MAPASET i LOCATION - podstawowe jednostki katalogowania w ramach których tworzony jest `region` mają jeden i niezmienny układ odniesienia. Proces reprojektacji GRID w sadze jest prostrzy niż w GRASS

GRASS a QGIS

Od początku rozwoju QGIS dostęp do bazy danych GRASS był proiorytetem. Na pewnym etapie rozwoju planowana była pełna integracja obu narzędzi. QGIS daje dostęp do bazy danych GRASS i pozwala na edycję danych (istotne dla danych wektorowych) oraz stosowanie narzędzi GRASS wewnątrz bazy danych (natywnie).

Provider GRASS to osobne narzędzie, geoprocessing realizowany jest poprzez zastosowanie możliwości odczytu danych zewnętrznych a następnie eksport wyników i usunięcie danych pośrednich

Skrypty w języku R

Wymagają instalacji pakietów geoprzetwarzania jak `sp`, `raster` oraz ich nowsze odpowiedniki `sf`, `terra` czy `stars` w środowisku R.

Zastosowanie skryptów w R to głównie wizualizacja oraz zastosowanie narzędzi obliczeniowych niedostępnych w Pythonie.

Skrypty R są wolne! i uzależnione od starych bibliotek

symbol `##` oznacza zdefiniowanie elementu interface graficznego oraz parametrów.

```
##Vector processing=group
##showplots
##Layer=vector
##X=Field Layer
boxplot(Layer[[X]])
```

Dostęp do narzędzi QGIS w skryptach Pytona

- **Linux:** `export PYTHONPATH=/<qgispath>/share/qgis/python`
- **macOS:** `export PYTHONPATH=/<qgispath>/Contents/Resources/python`
- **Windows:** `set PYTHONPATH=c:\<qgispath>\python`

```
from qgis.core import *

QgsApplication.setPrefixPath("/path/to/qgis/installation", True)

qgs = QgsApplication([], True)
qgs.initQgis()
# kod aplikacji
qgs.exitQgis()
```