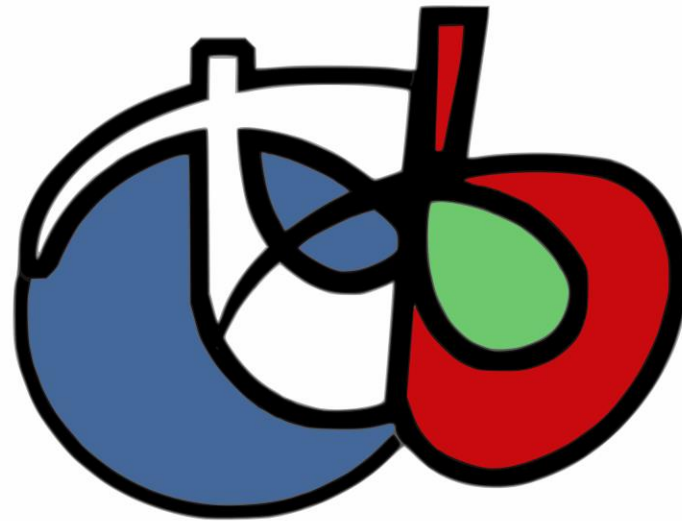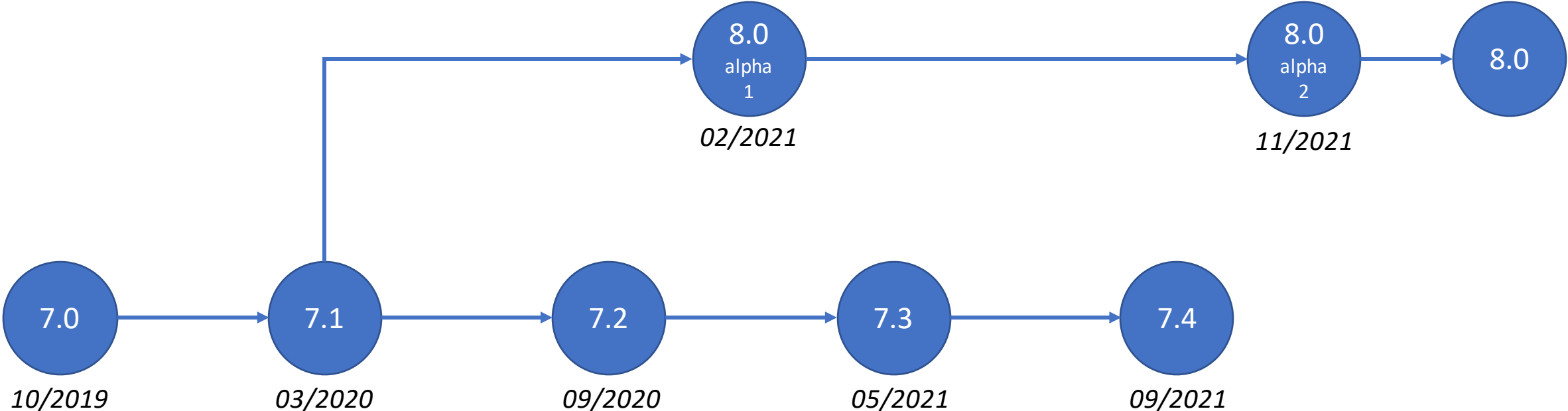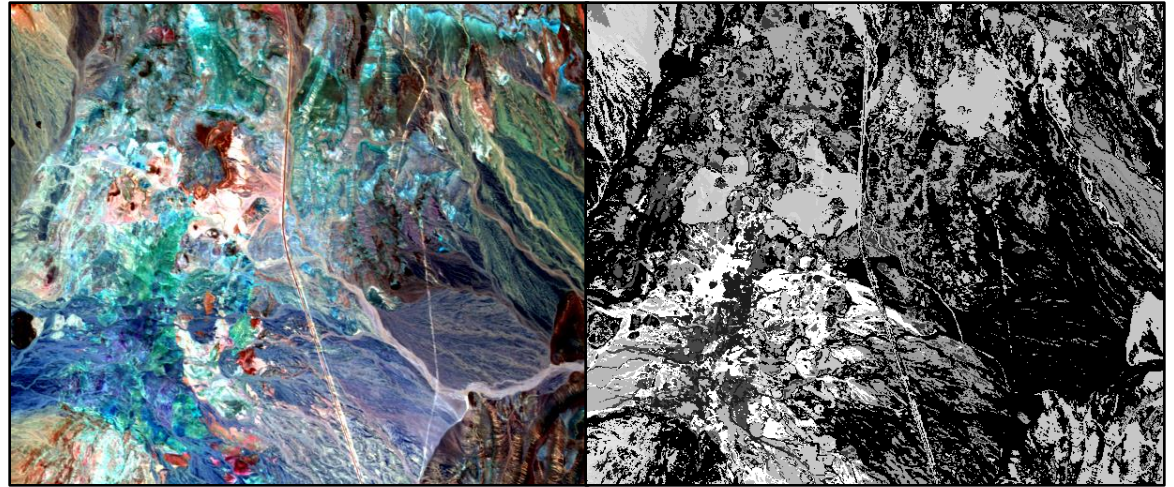# What's new in OTB ?

User days 2021

OTB releases

OTB 7: New applications

- Regression framework

  - TrainVectorRegression (7.0)

  - TrainImagesRegression (7.0)

  - VectorRegression (7.0)

  - ImagesRegression (7.0)

- Hyperspectral Image processing

  - LocalRxDetection (7.0)

  - EndmemberNumberEstimation (7.0)

  - SpectralAngleClassification (7.2)

Spectral angle classification

What's new in OTB 7 ?

OTB 7: New applications

- Image processing

  - FastNLMeans (7.1)

  - PantexTextureExtraction (7.2)

- Utility

  - ZonalStatistics (7.0)

  - ResetMarging (7.2)

  - Synthetize (7.2)
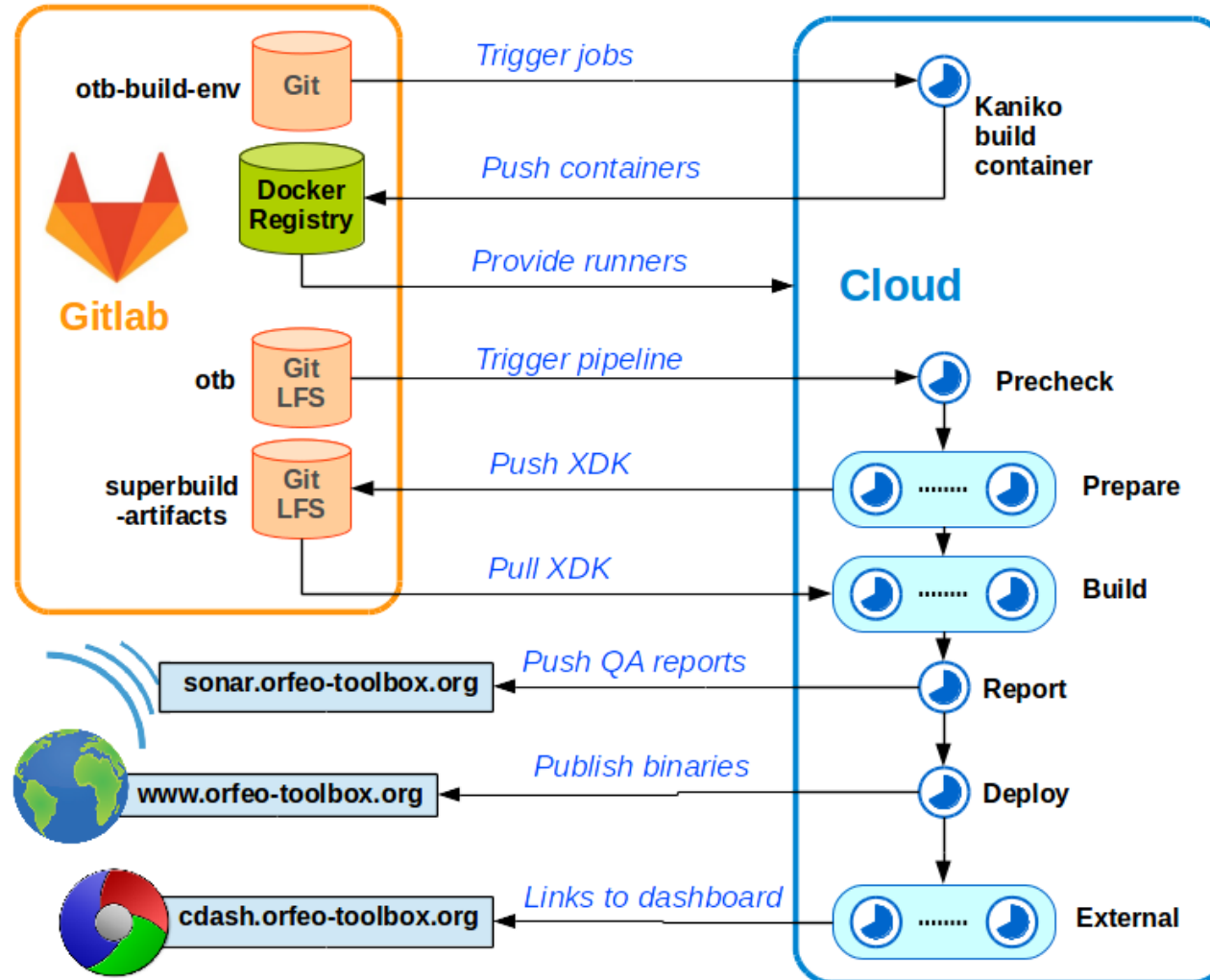
What's new in OTB 7 ?

OTB 7: SAR

- New SAR sensor models :
  - CosmoSkyMed
  - TerraSAR-X

- Remote modules:
  - Diapotb
  - S1-tiling

- Improvement of the Sentinel 1 model

OTB 7: New OTB Continuous Integration platform

What's new in OTB 7 ?

OTB 7: Functor Image Filter

A generic filter for pixel based operations

- Takes any number of images or vector images as input
- Output an image or a vector image
- Operation based on pixel, neighbordhood or a mix of both
- Operation defined using a lambda, functor or a function pointer

```cpp
// Define the filter
auto ndvi = [](double nir, double red) -> double {(nir - red) / (nir + red)};
auto ndviFilter = NewFunctorFilter(ndvi);

// Set inputs (otb::Image)
ndviFilter->SetInput<1>(nirImage);
ndviFilter->SetInput<2>(redImage);

// Use the filter
ndviFilter->Update();
```

Check out the "[Functor application template](#)" project on the Gitlab

What's new in OTB 7 ?

And also ....

- Support for GDAL 3

- Switch from Python 2 to Python 3

- Logs for the Python wrapper

- NoData extended filename for output images :

  &nodata=(double) value

- The Java wrapper has been removed

OTB 8.0

Coming soon !

Ossim

An open source library for geospatial image processing

Ossim usage in OTB
- Spatial reference (refactored in OTB 7.0)
- Product metadata parsing
- DEM handling
- Time points and duration
- Sensor models :
  - SAR Sensor models
  - RPC model

Ossim removal

Why are we removing Ossim from OTB ?

- Hard to package (Debian, Conda, Superbuild)

- Hard to follow Ossim development cycle

- Many Ossim functionalities are also implemented in GDAL

Ossim Removal

Metadata parsing refactoring

Goal : Use GDAL as much as possible to handle metadata

- **Metadata parsing**

    - Read metadata from GDAL drivers when possible and
    from the product metadata files when needed

    - Support for geom files from OTB 7.0

- **Supported sensors**

    - SAR: Sentinel-1, TerraSAR-X, CosmoSkyMed, Radarsat-2

    - Optical: Pleiades, Formosat, Worldview 2, Quickbird,
    Ikonos, Spot 5/6/7

What's new in OTB 8 ?

Ossim Removal

Metadata parsing refactoring

Metadata writing

- Metadata are written in the output image using GDAL

- OTB does not create geom files anymore

```
> gdalinfo pleiades_roi.tif
Driver: GTiff/GeoTIFF
Files: pleiades_roi.tif
Size is 100, 100
Coordinate System is:

[...]

Metadata:
  AcquisitionDate=2012-02-25T00:25:59.9Z
  AREA_OR_POINT=Area
  BlueDisplayChannel=2
  DataType=1
  Extra.ImageID=PRG_FC_5855-001
  GeometricLevel=ORTHO
  GreenDisplayChannel=1
  Instrument=PHR
  InstrumentIndex=1A
  METADATATYPE=OTB
  Mission=Pléiades
  OTB_VERSION=8.0.0
  ProductionDate=2013-01-31T09:50:51.821Z
  RedDisplayChannel=0
  SatAzimuth=284.284
  SatElevation=88.8475
  SensorID=PHR 1A
  SunAzimuth=53.6912
  SunElevation=49.7424
  TileHintX=2048
  TileHintY=2048

[...]
```

Ossim Removal

Metadata parsing refactoring

Internally ...

- Ossim keywordlists have been replaced by **ImageMetadata**

- **ImageMetadata** stores different types of metadata:
    - MDNum: numerical values
    - MDStr: string
    - MDTime: dates
    - MDL1D and MDL2D: Lookup tables
    - MDGeom: RPC, SAR model, GCP, projection

```
otb::ImageMetadata & imd = image->GetImageMetadata();

std::string id = imd[MDStr::SensorID];
bool hasDate = imd.Has(MDTime::ProductionDate);
imd.Add(MDNum::NoData, 999);
```

Ossim Removal

DEM Handler

Use GDAL to read DEM and geoid files :

- Any monoband raster format supported by GDAL can be used as DEM

- The application elevation API is the same:
  - -elev.dem: path to a directory containing DEM tiles
  - -elev.geoid: path to a geoid file
  - -elev.default: value used when no elevation info is available

- The C++ API is still based on a singleton

```
otb::DEMHandler::GetInstance()::OpenDEMDirectory(directoryPath);
double elevation = otb::DEMHandler::GetInstance()::GetHeightAboveEllipsoid(lon, lat);
```

Ossim Removal

RPC model

- RPC coefficient parsing is done by GDAL
- RPC transformation class based on GDALRPCTransform
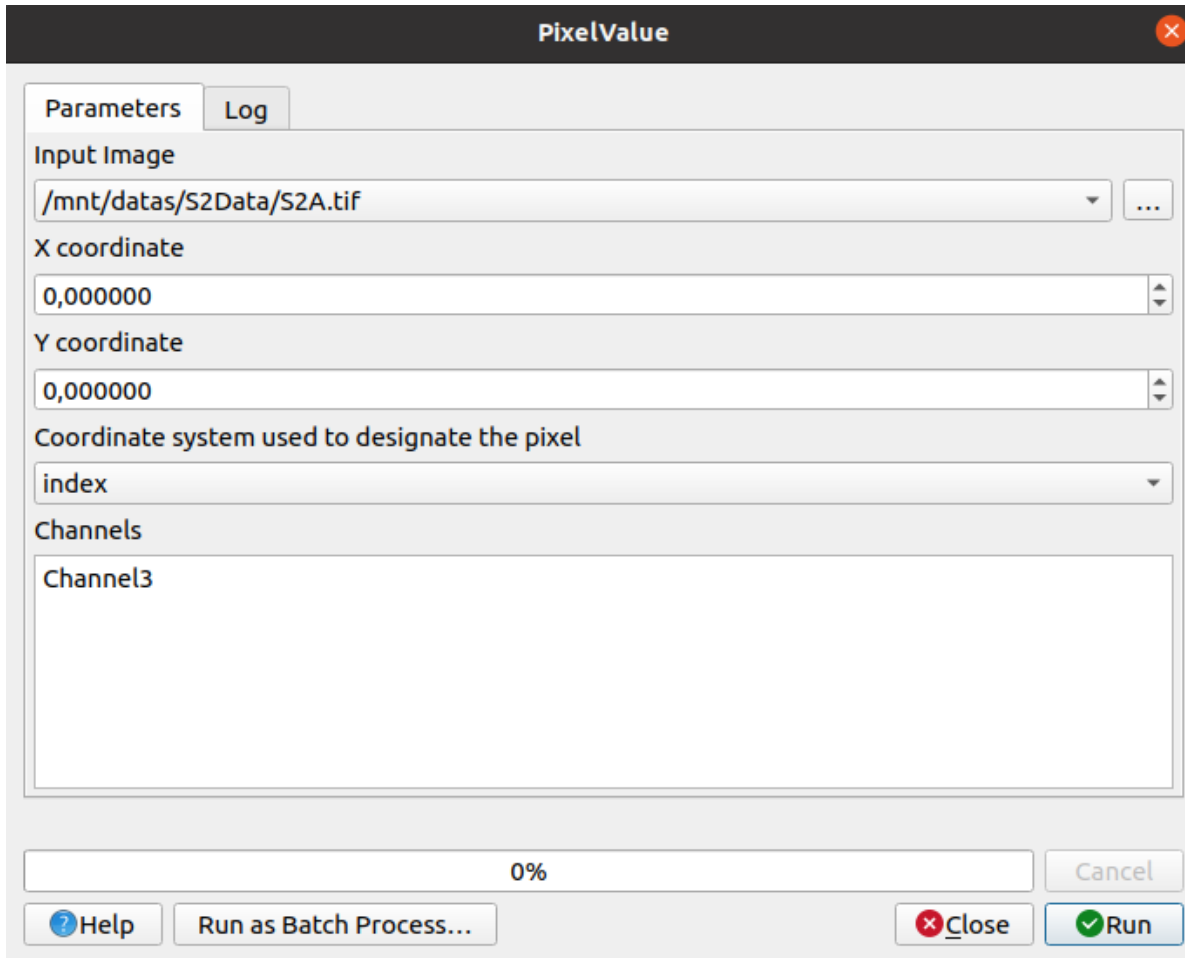
SAR model

- SAR functionalities available in Ossim plugins have been reimplemented in OTB in SarSensorModel
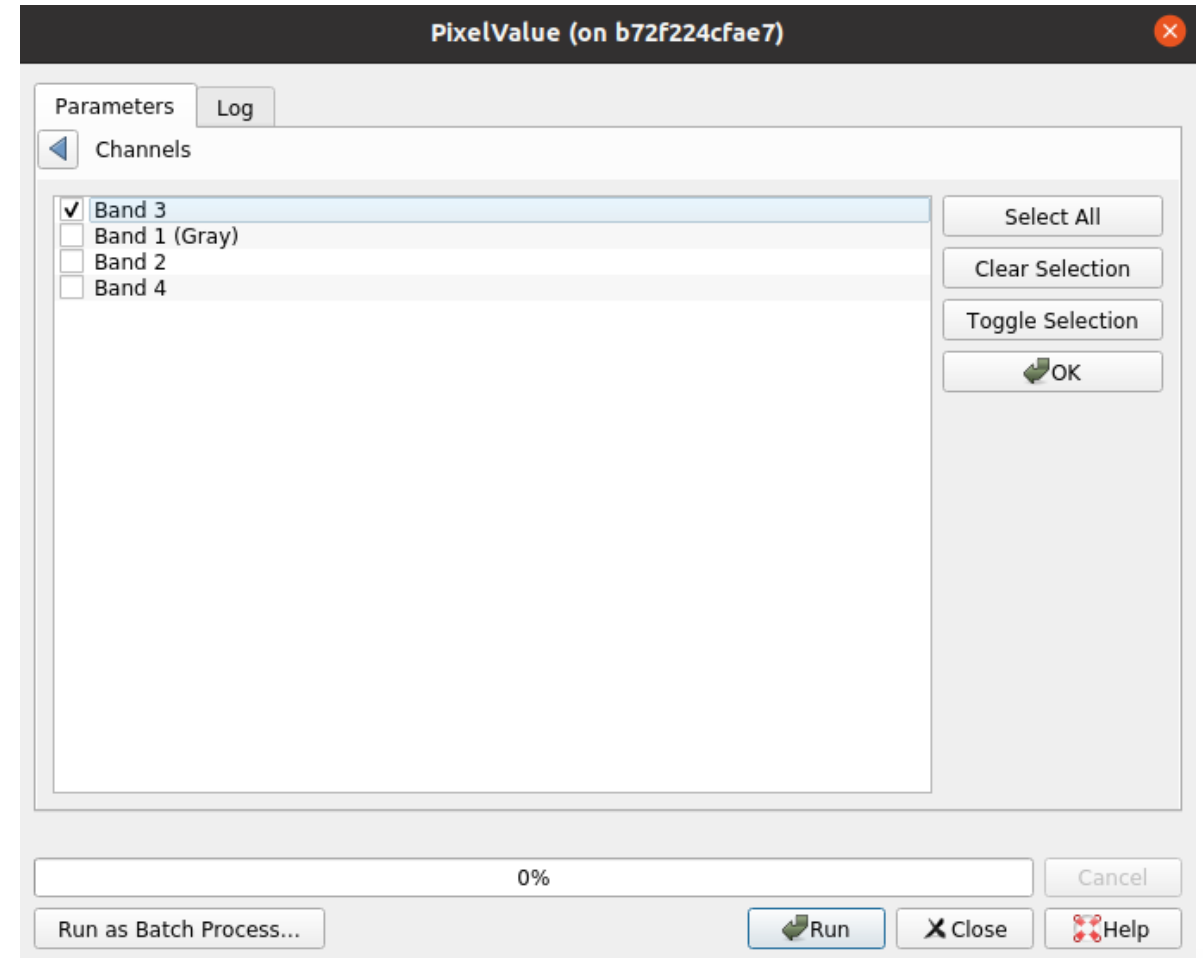
Time points and durations

- Usage of std::chrono and of the Howard Hinnant date library

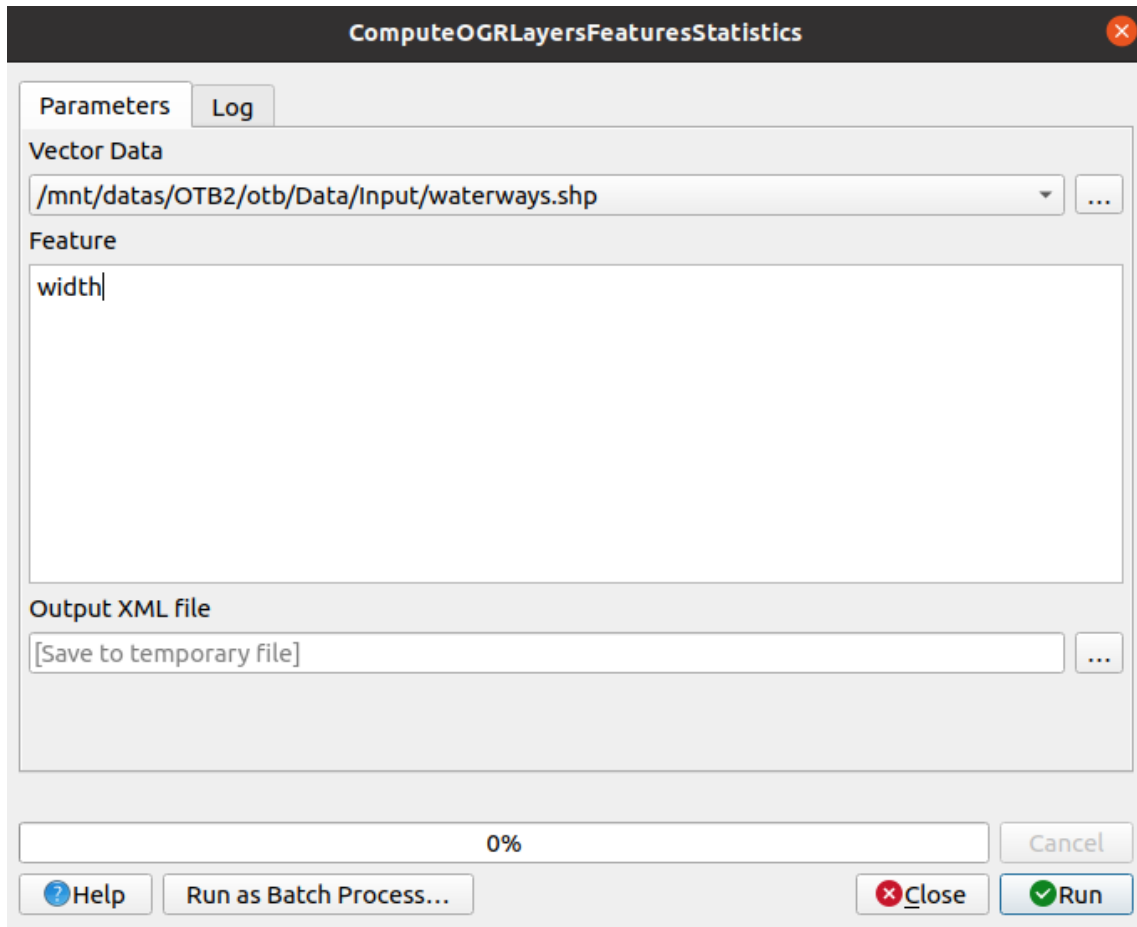# Improvements of the QGIS plugin
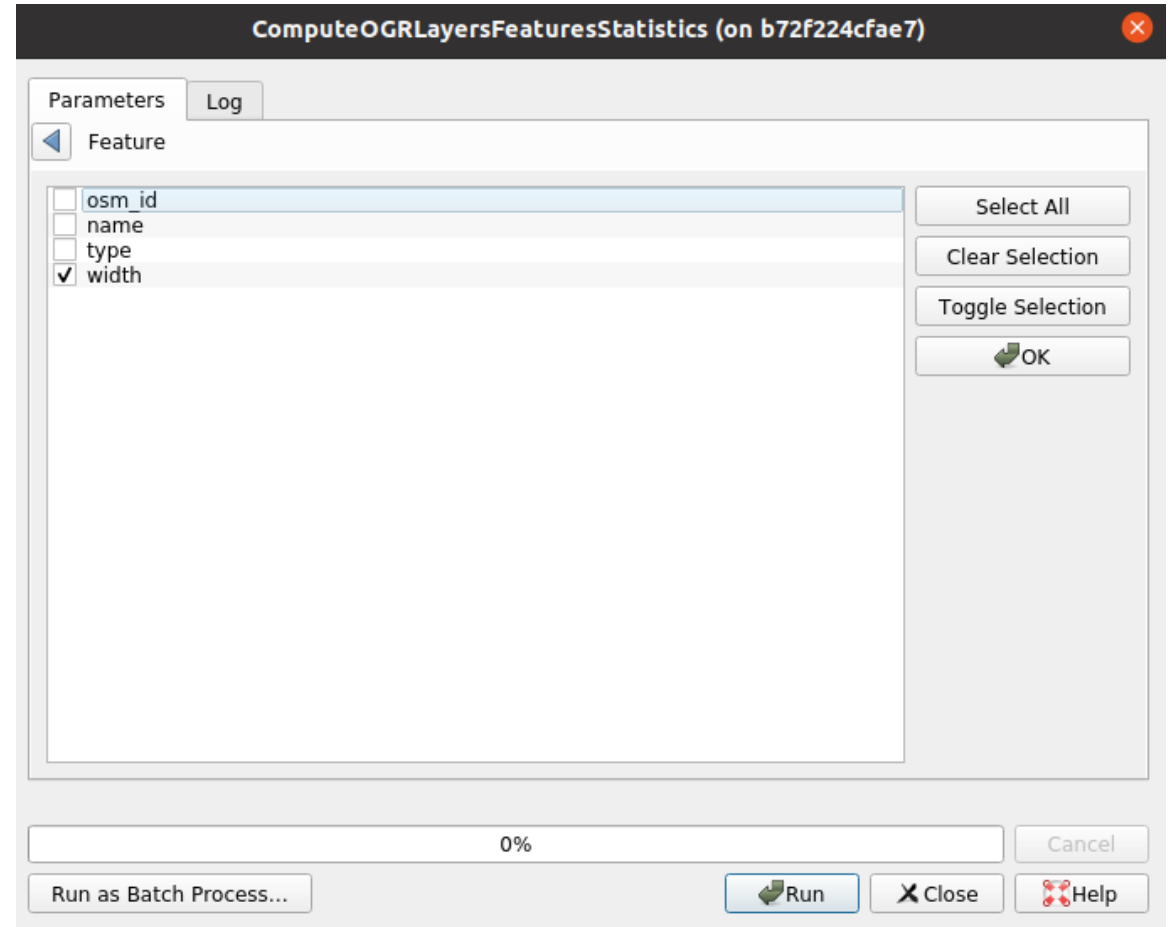

OTB 7.4

OTB 8.0

What's new in OTB 8 ?

Improvements of the QGIS plugin

OTB 7.4

OTB 8.0

What's new in OTB 8 ?

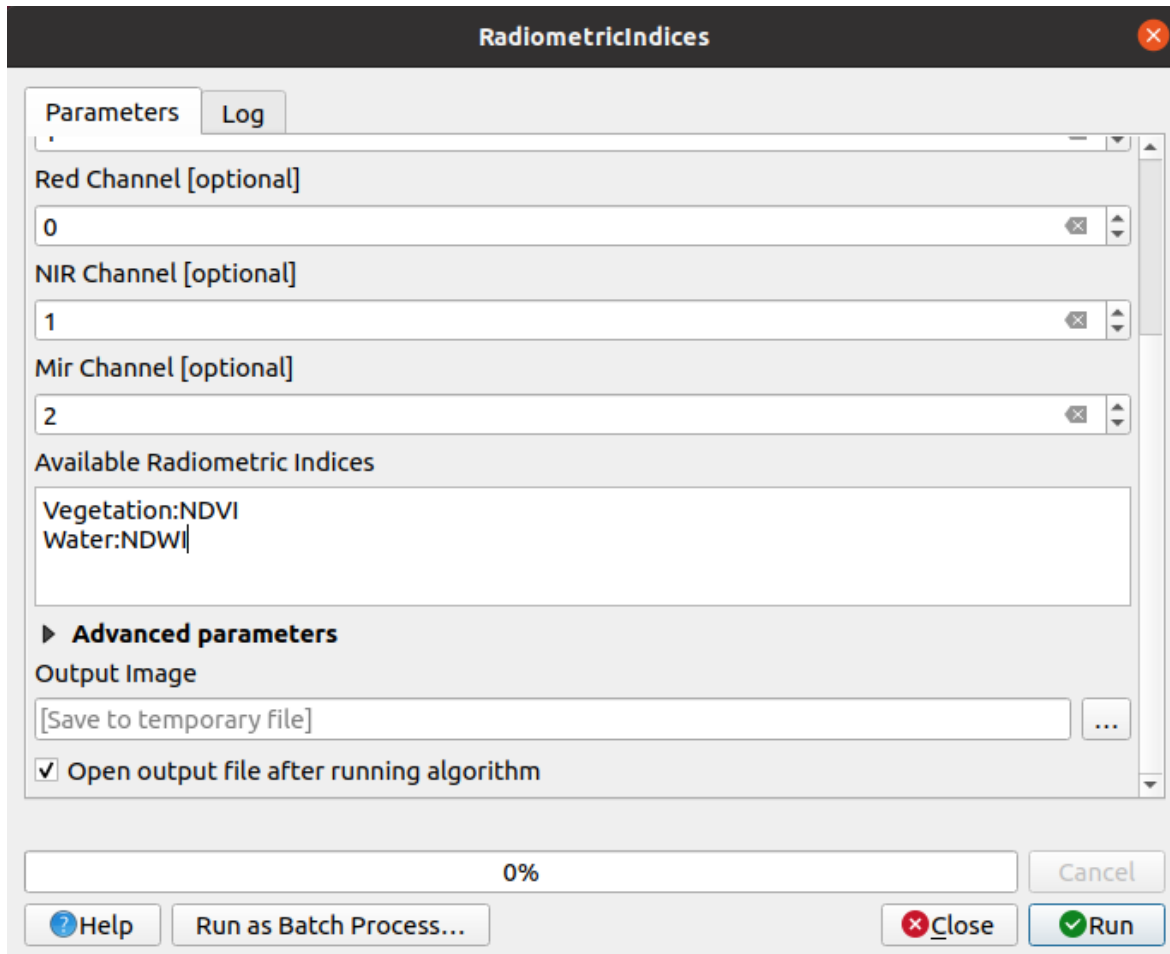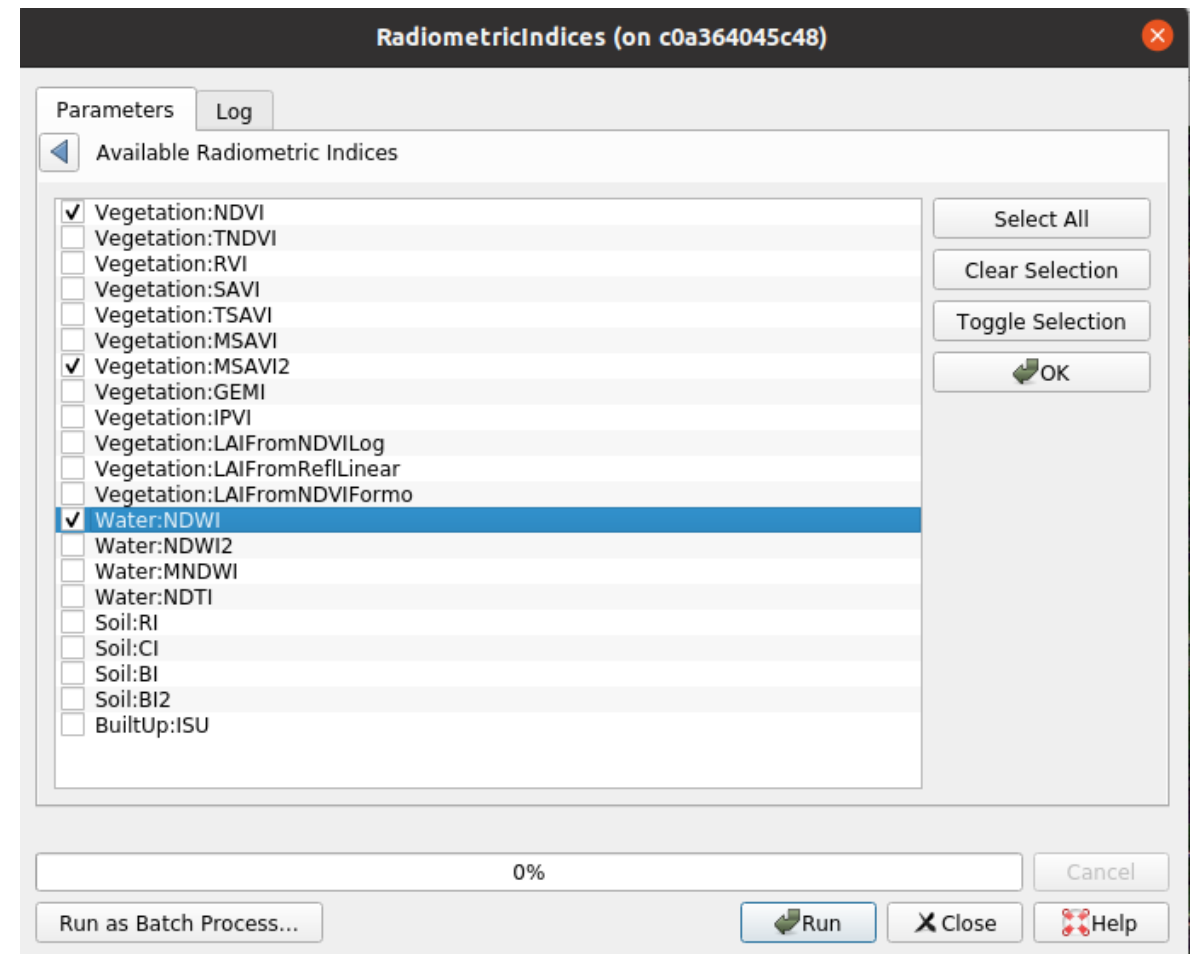# Improvements of the QGIS plugin

OTB 7.4   OTB 8.0

What's new in OTB 8 ?

Let's test the new release on Wednesday !