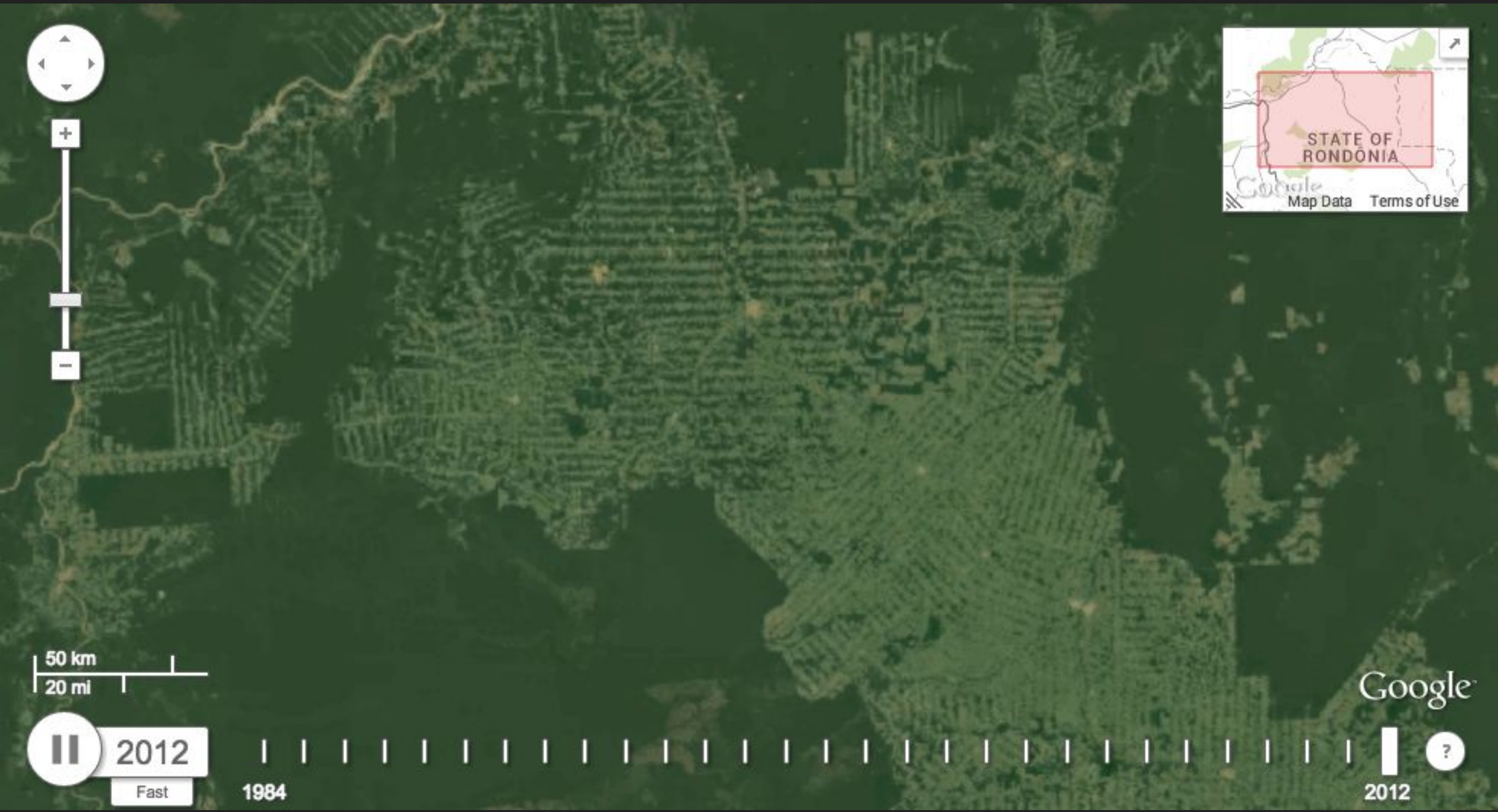
Image Data Cube

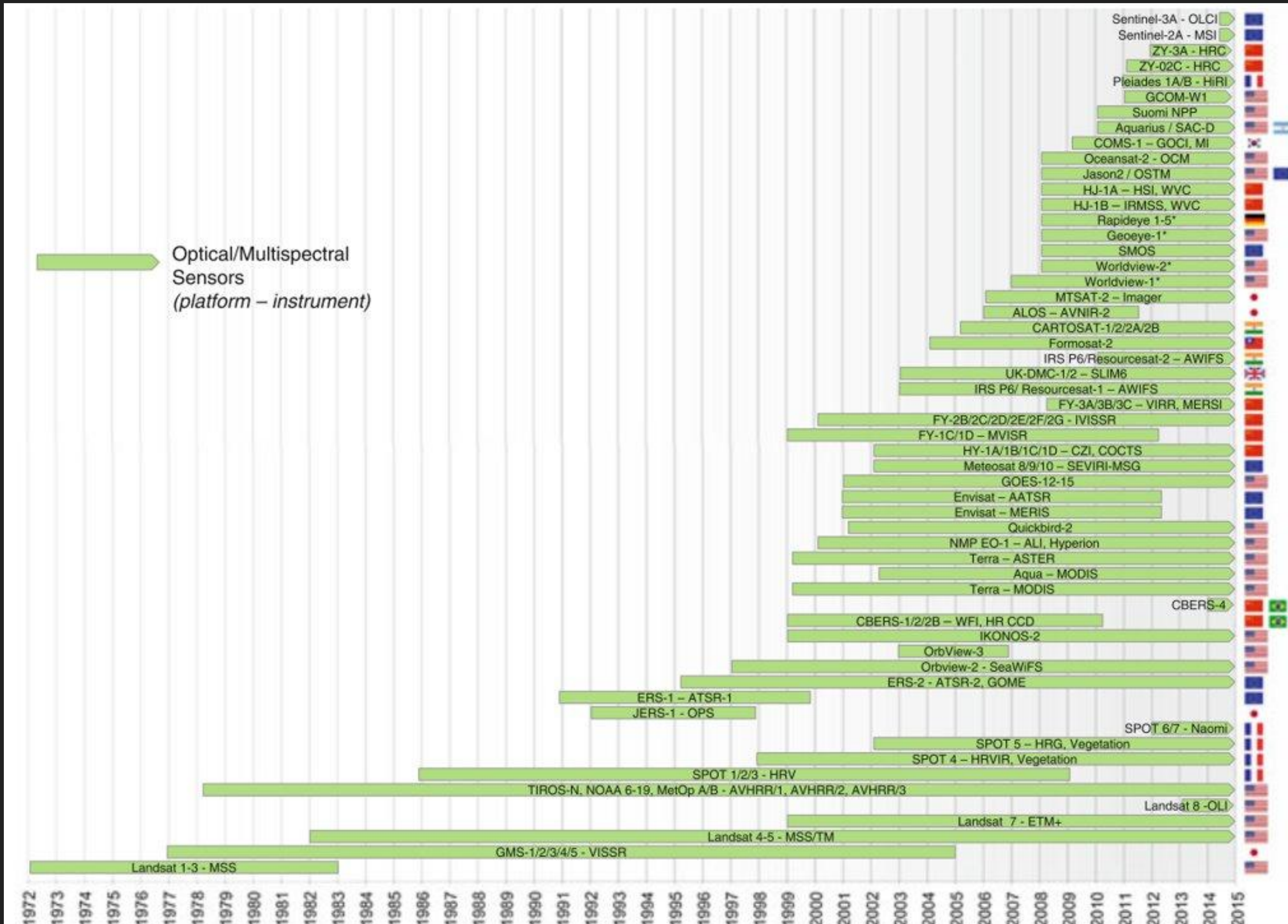
2019-Oct-08

Rennan Marujo

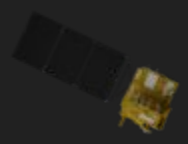

STATE OF RONDÔNIA
Map Data
Terms of Use
Google
50 km
20 mi
2012
Fast
1984
2012
Google

Spatial Ground Resolution	Ultra High < 1m	Very High 1 – 4 m	High 4 – 10 m	Medium 10 – 50 m	Low 50 – 250 m	Very Low > 250 m

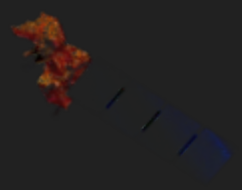
Source: Ehlers et. al (2002)

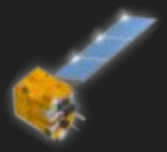


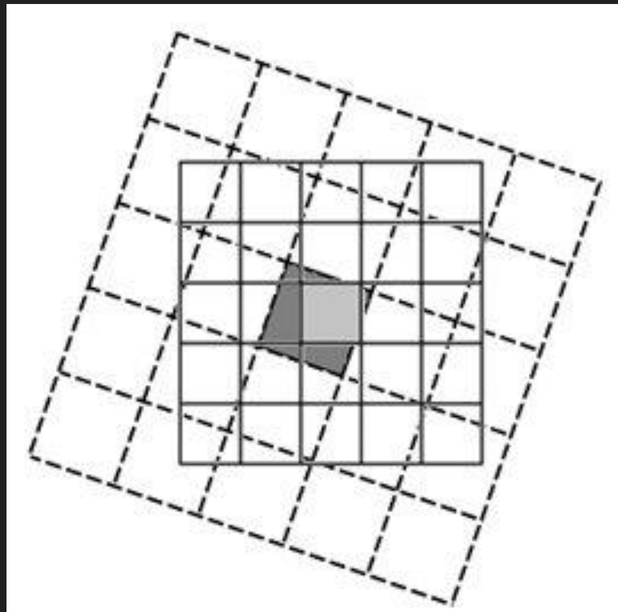
Source: Kuenzer et. al (2015)

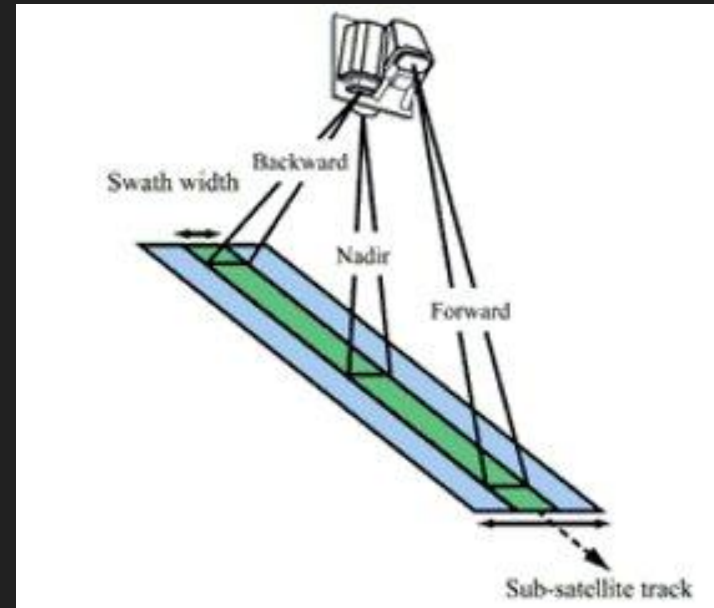


DATA MUST BE COMPATIBLE

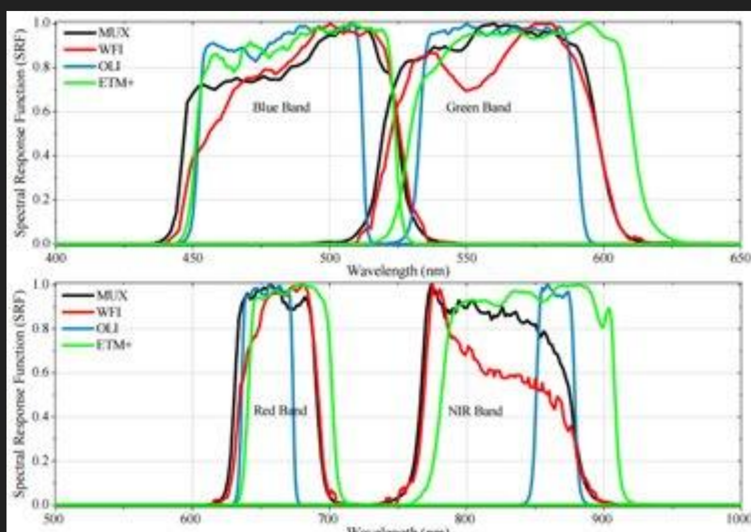




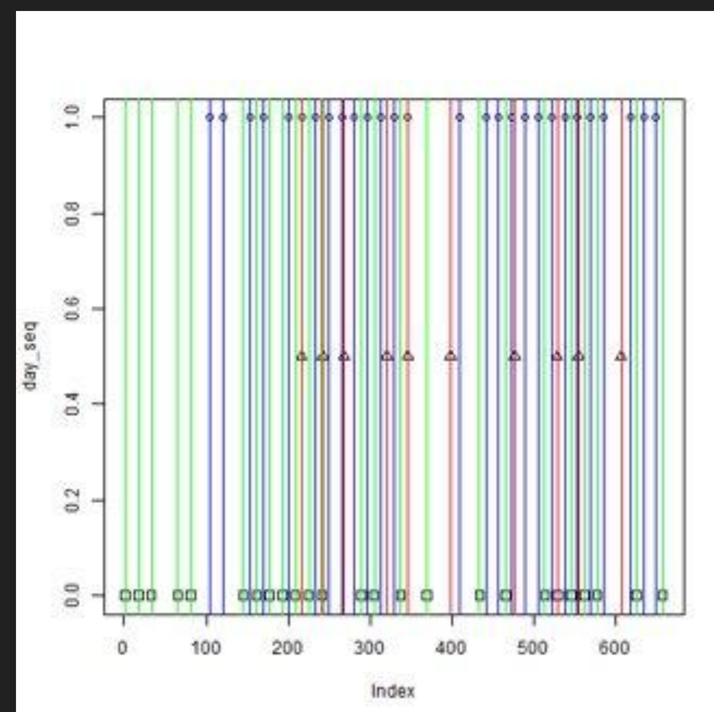




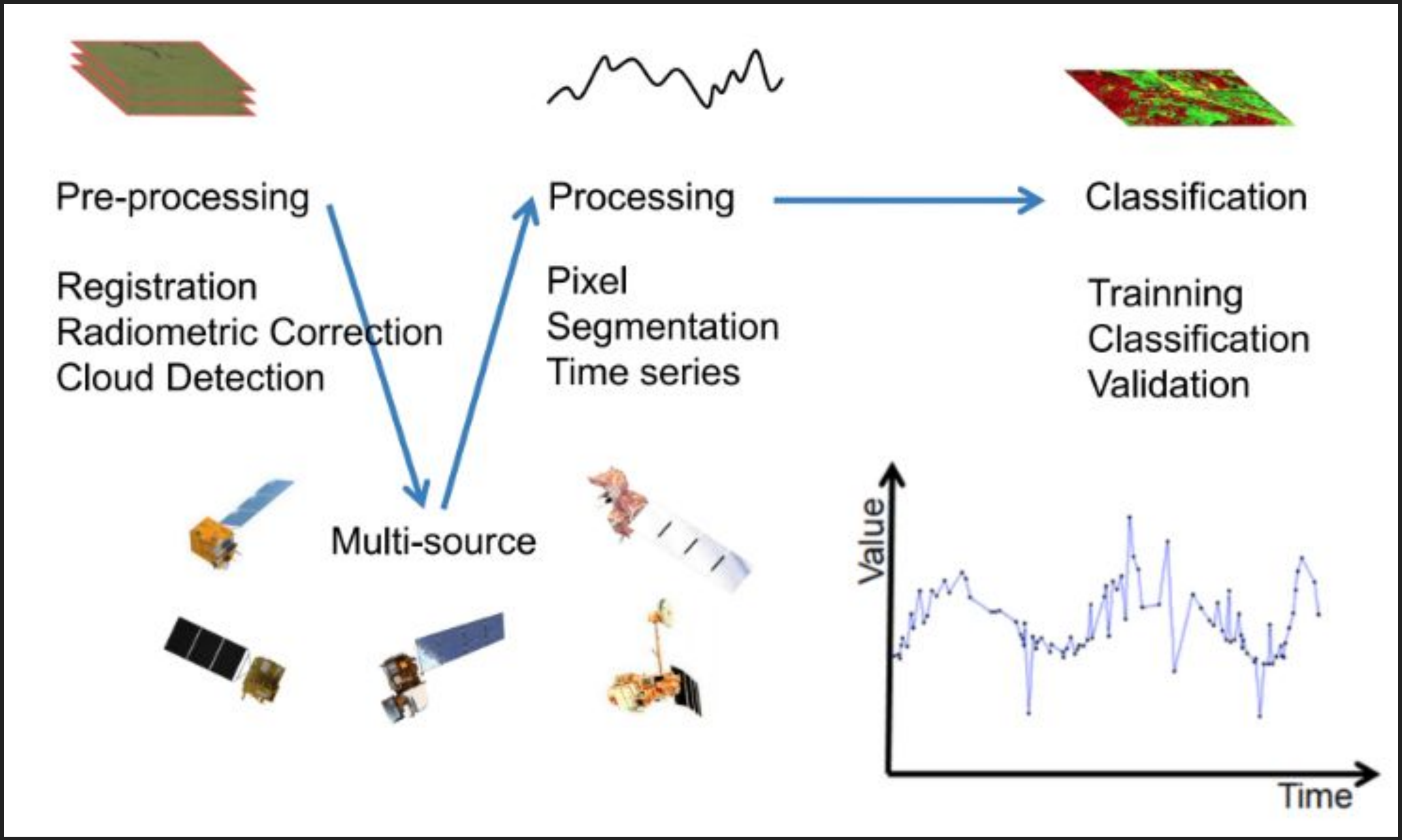
Swath width
Backward
Nadir
Forward
Sub-satellite track

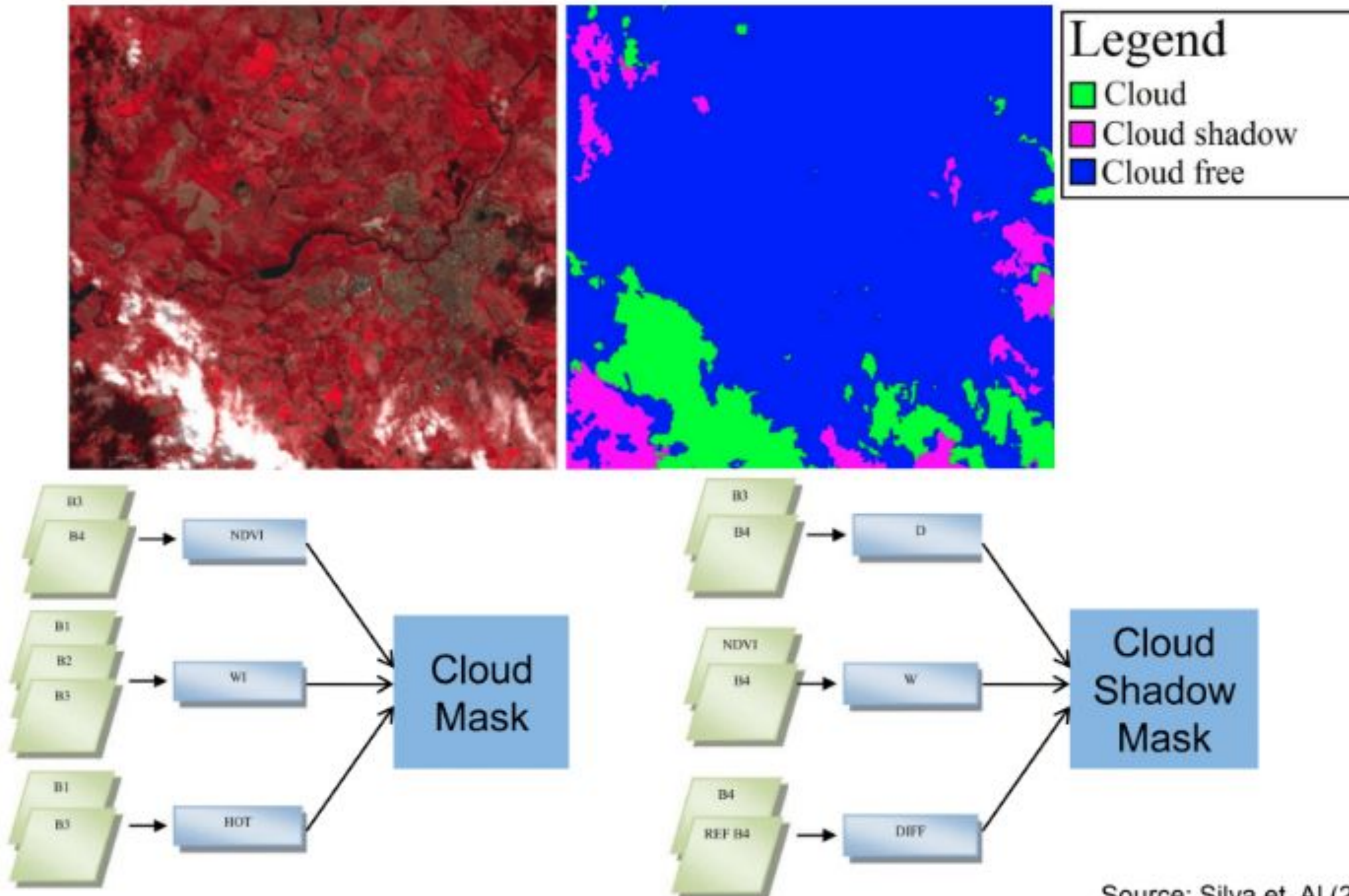


MUX
WFI
OLI
ETM+
Blue Band
Green Band
Spectral Response Function (SRF)
Wavelength (nm)
Red Band
NIR Band

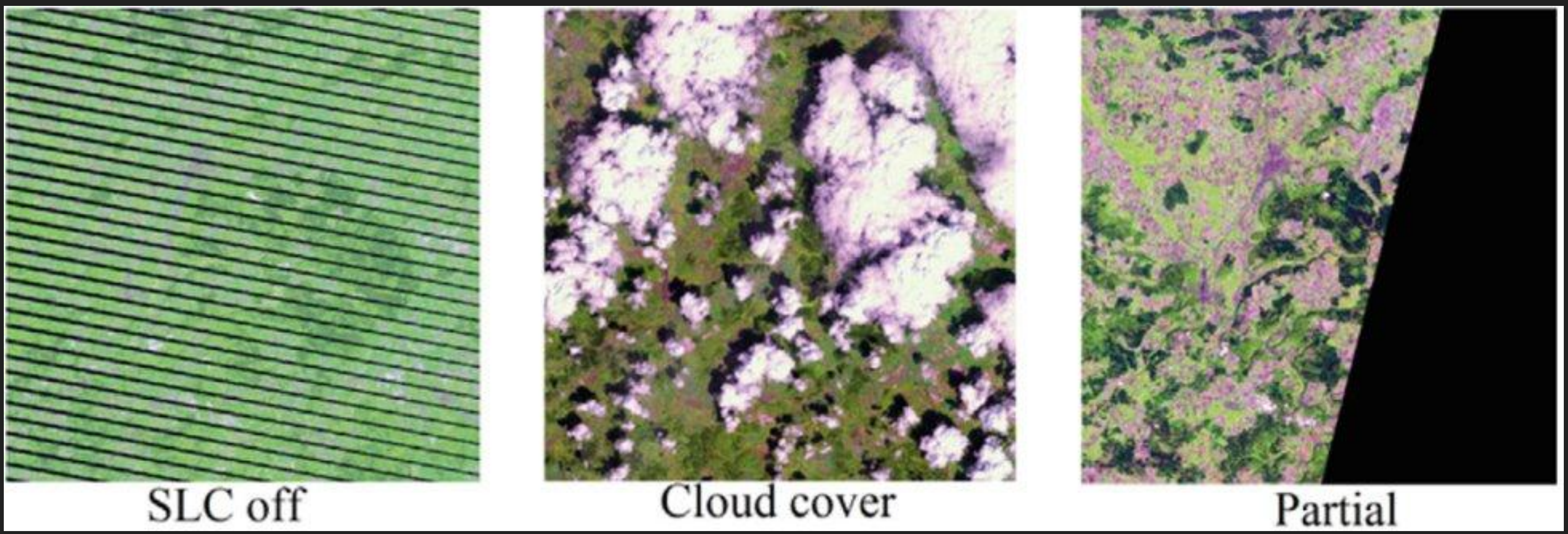


day_seq
Index

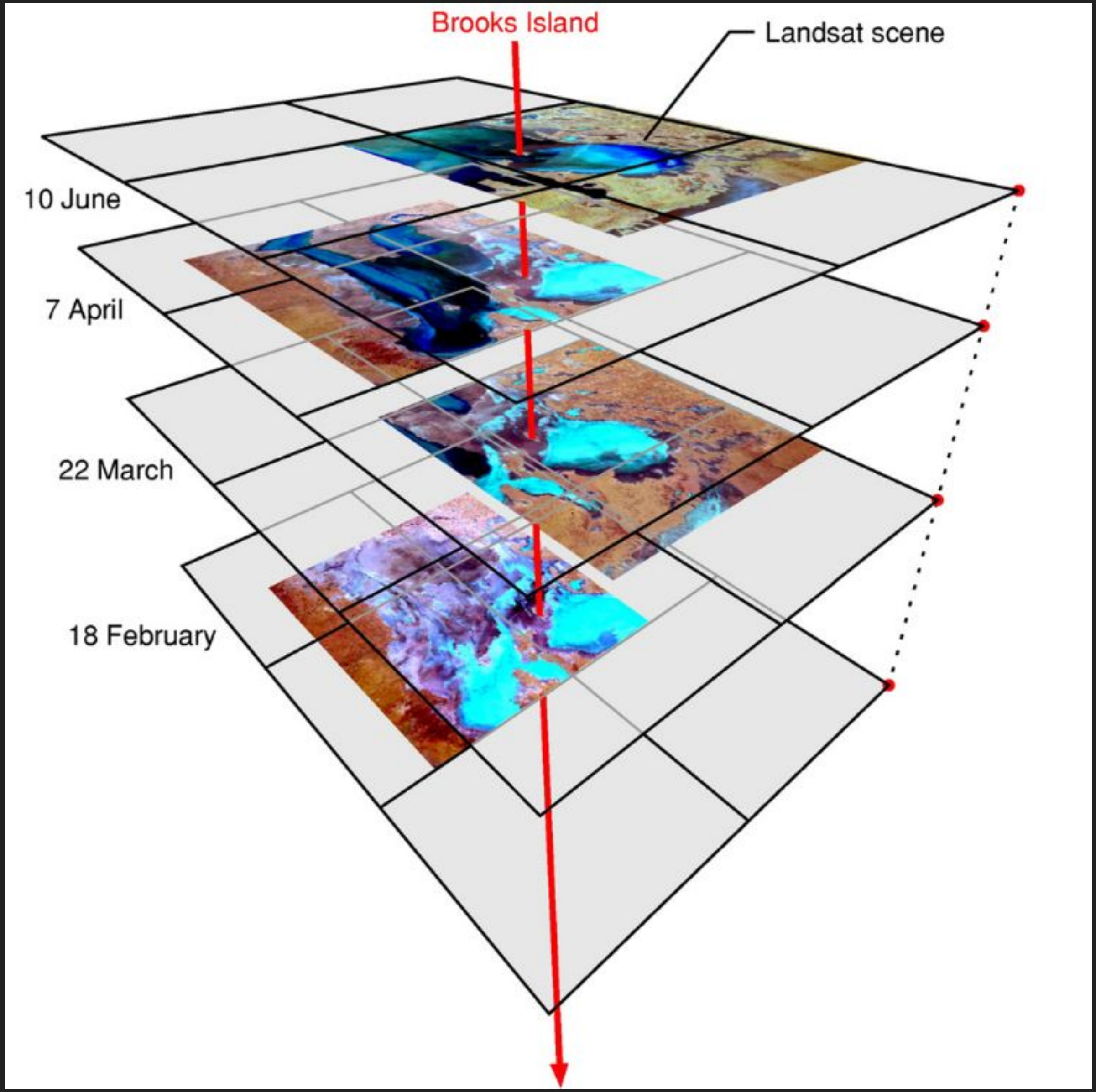

Pre-processing
Registration
Radiometric Correction
Cloud Detection
Processing
Pixel
Segmentation
Time series
Classification
Trainning
Classification
Validation
Multi-source
Value
Time



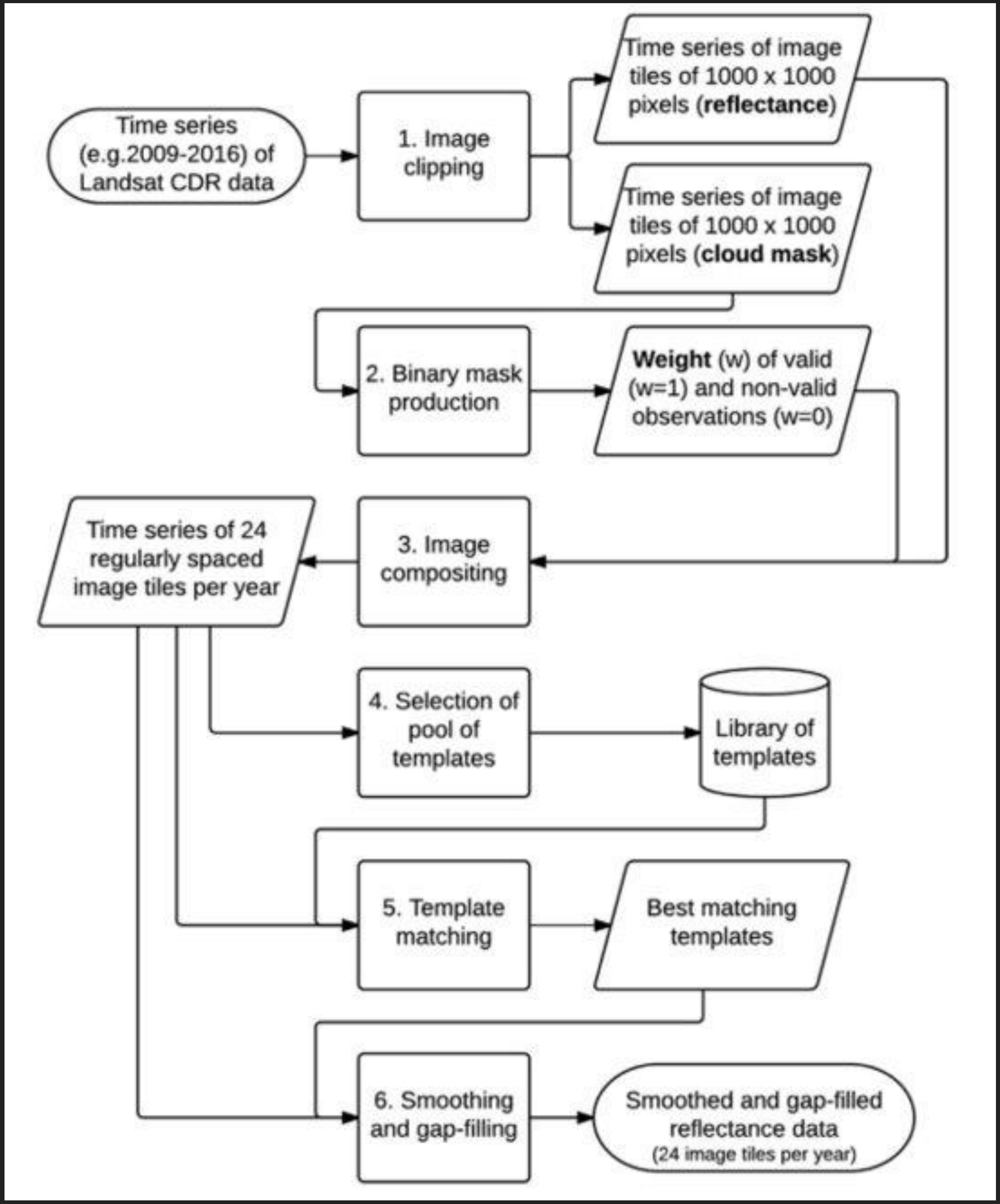
Source: Silva et. Al (2016)



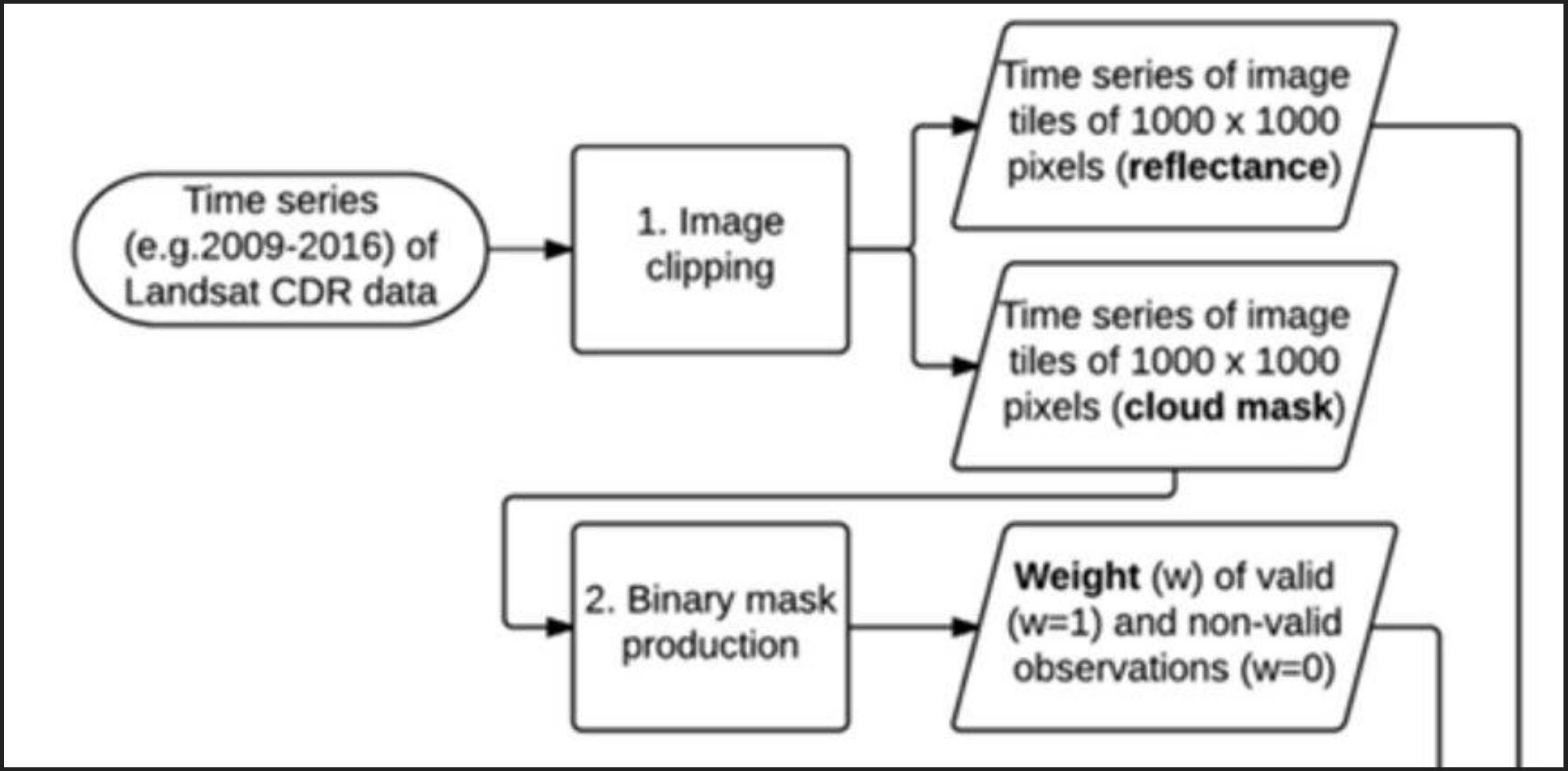
Source: Vuolo et. al (2017)



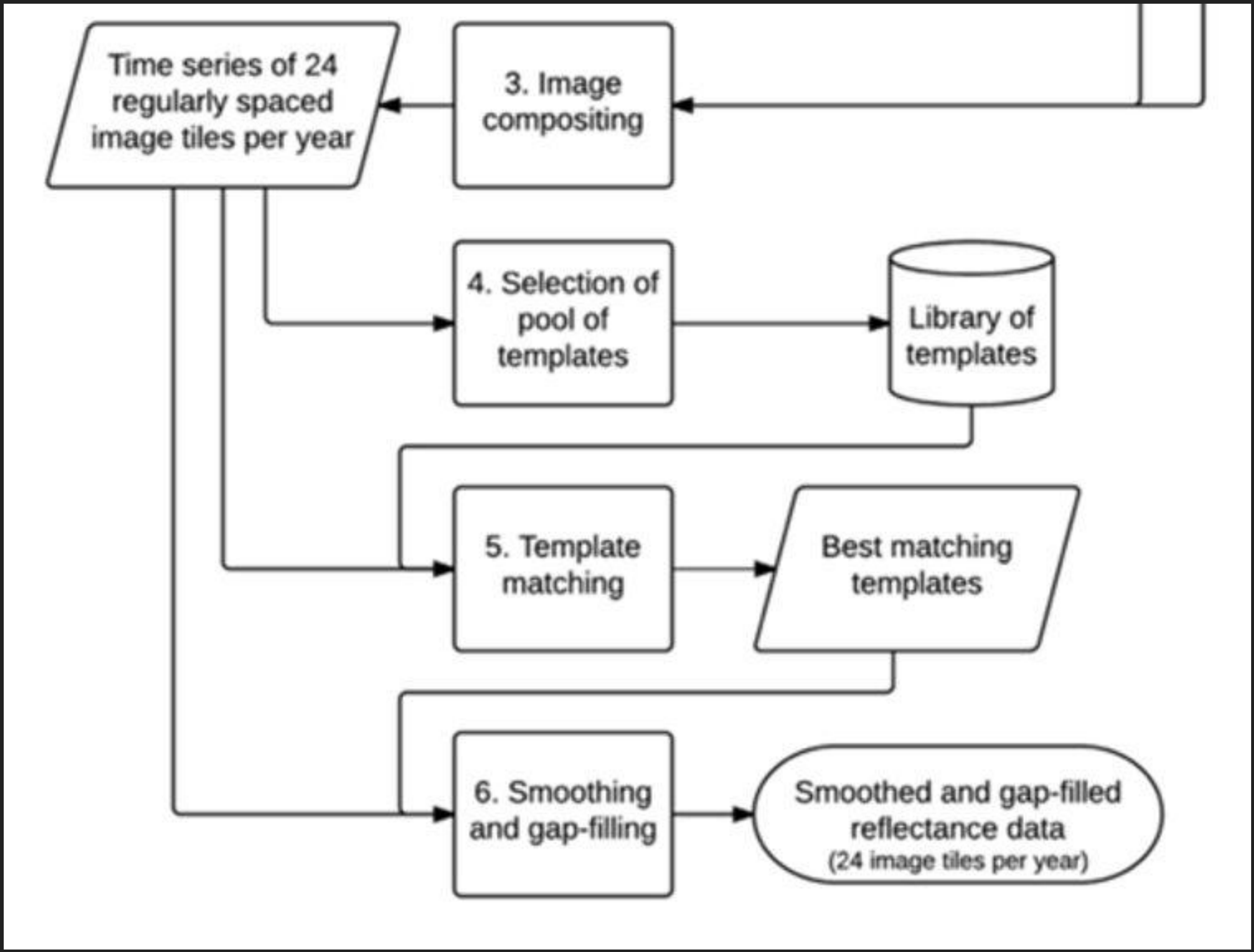
Source: Lewis et. al (2017)



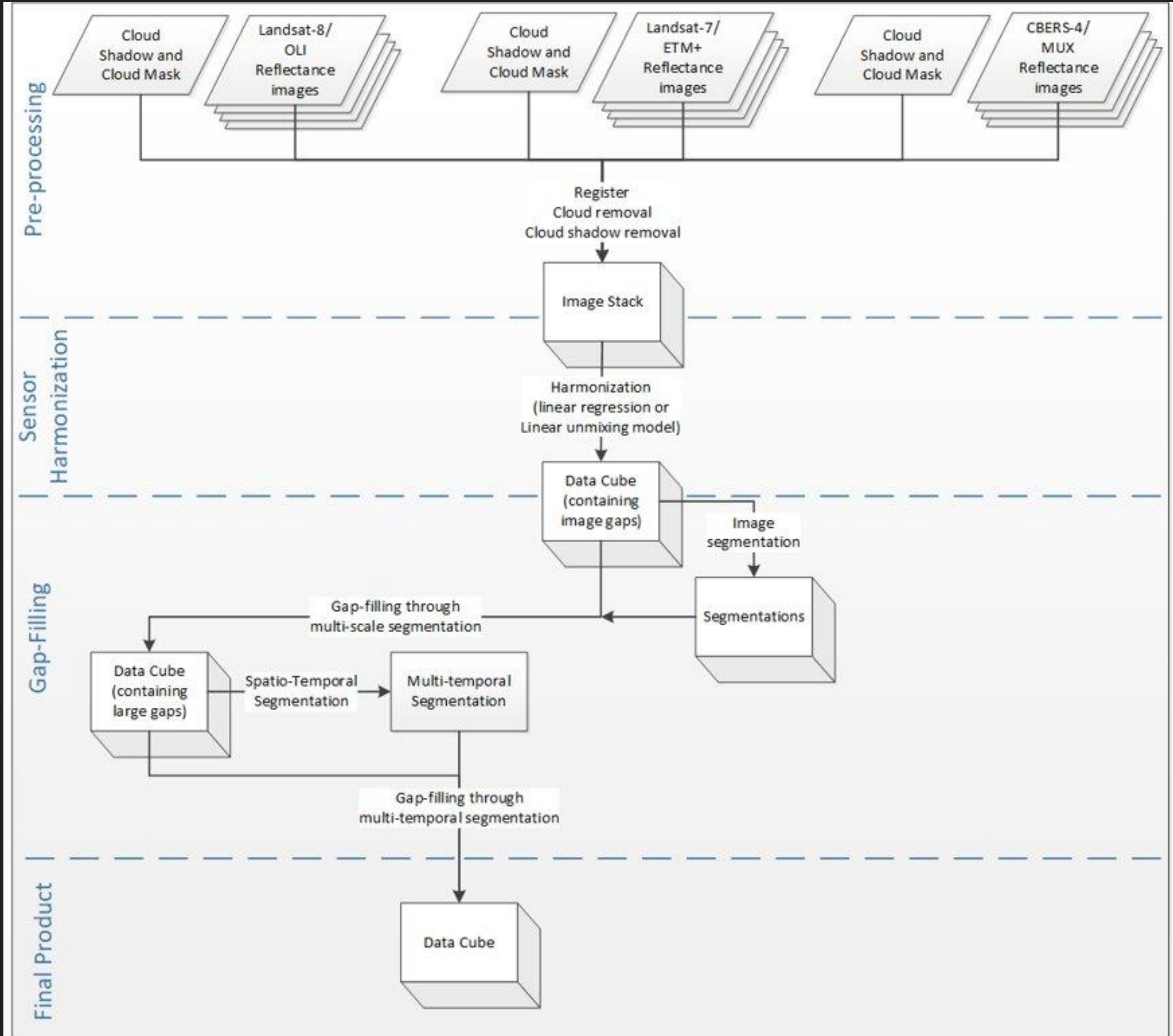
Source: Vuolo et. al (2017)



Source: Vuolo et. al (2017)



Source: Vuolo et. al (2017)



Pre-processing
Cloud Shadow and Cloud Mask
Landsat-8/ OLI Reflectance images
Cloud Shadow and Cloud Mask
Landsat-7/ ETM+ Reflectance images
Cloud Shadow and Cloud Mask
CBERS-4/ MUX Reflectance images
Register
Cloud removal
Cloud shadow removal
Image Stack
Sensor Harmonization
Harmonization (linear regression or Linear unmixing model)
Data Cube (containing image gaps)
Image segmentation
Segmentations
Gap-Filling
Gap-filling through multi-scale segmentation
Data Cube (containing large gaps)
Spatio-Temporal Segmentation
Multi-temporal Segmentation
Gap-filling through multi-temporal segmentation
Final Product
Data Cube

Open Data Cube (ODC)

Remote Sensing of Environment 202 (2017) 276–292

Contents lists available at ScienceDirect

Remote Sensing of Environment

journal homepage: www.elsevier.com/locate/rse

The Australian Geoscience Data Cube — Foundations and lessons learned

Adam Lewis [a,*], Simon Oliver [a], Leo Lymburner [a], Ben Evans [b], Lesley Wyborn [b], Norman Mueller [a], Gregory Raevksi [a], Jeremy Hooke [a], Rob Woodcock [c], Joshua Sixsmith [a], Wenjun Wu [a], Peter Tan [a], Fuqin Li [a], Brian Killough [d], Stuart Minchin [a], Dale Roberts [a,b], Damien Ayers [a], Biswajit Bala [a], John Dwyer [e], Arnold Dekker [c], Trevor Dhu [a], Andrew Hicks [a], Alex Ip [a], Matt Purss [a], Clare Richards [b], Stephen Sagar [a], Claire Trenham [b], Peter Wang [c], Lan-Wei Wang [a]

[a] *Geoscience Australia, GPO Box 378, Canberra, ACT 2601, Australia*
[b] *Australian National University, Canberra, ACT 2601, Australia*
[c] *Commonwealth Scientific Industrial Research Organisation (CSIRO), GPO Box 1700, Canberra, ACT 2600, Australia*
[d] *NASA Langley Research Center, 1 Nasa Dr, Hampton, VA 23666, United States*
[e] *U.S. Geological Survey Earth Resources Observation and Science (EROS) Center, 47914 252nd Street, Sioux Falls, SD 57198, United States*

A R T I C L E   I N F O

Article history:
Received 27 July 2016
Received in revised form 7 February 2017
Accepted 12 March 2017
Available online 12 April 2017

Keywords:
Landsat
Time-series
Big data
Data cube
High performance computing
High performance data
Collection management
Geometric correction
Pixel quality
Australian Geoscience Data Cube

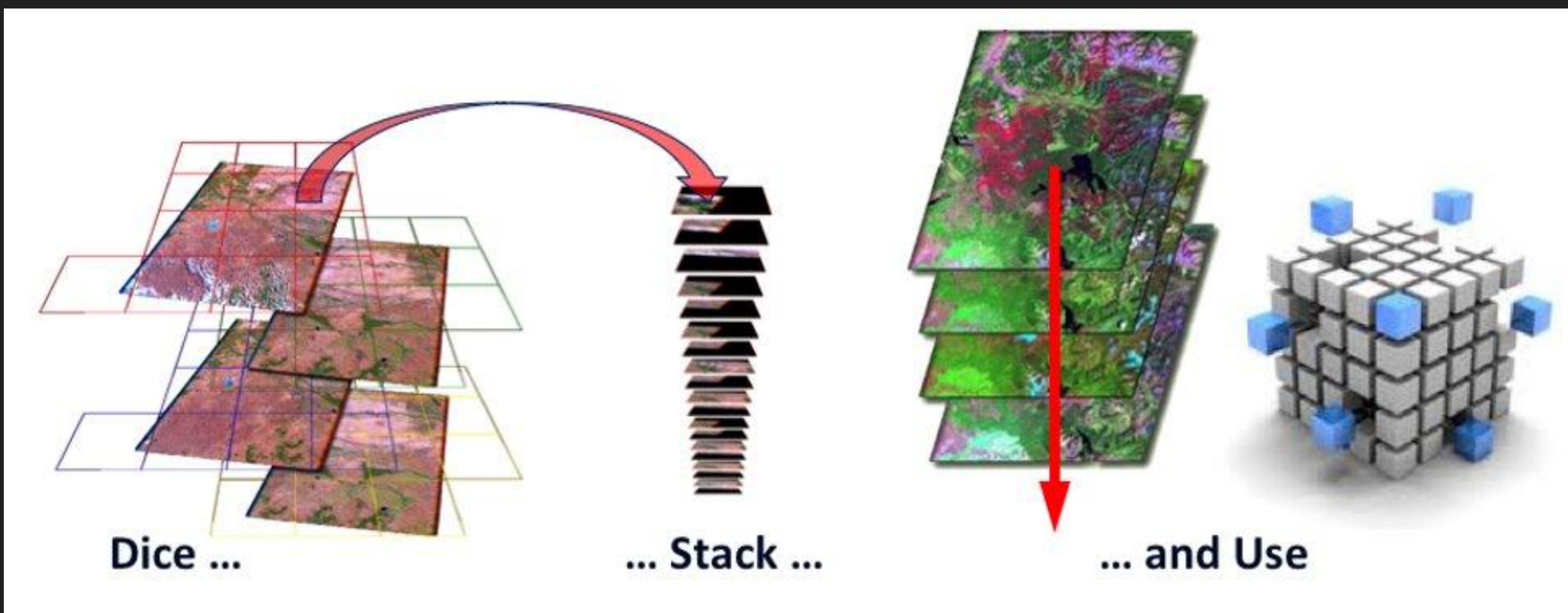
A B S T R A C T

The Australian Geoscience Data Cube (AGDC) aims to realise the full potential of Earth observation data holdings by addressing the Big Data challenges of volume, velocity, and variety that otherwise limit the usefulness of Earth observation data. There have been several iterations and AGDC version 2 is a major advance on previous work. The foundations and core components of the AGDC are: (1) data preparation, including geometric and radiometric corrections to Earth observation data to produce standardised surface reflectance measurements that support time-series analysis, and collection management systems which track the provenance of each Data Cube product and formalise re-processing decisions; (2) the software environment used to manage and interact with the data; and (3) the supporting high performance computing environment provided by the Australian National Computational Infrastructure (NCI).

A growing number of examples demonstrate that our data cube approach allows analysts to extract rich new information from Earth observation time series, including through new methods that draw on the full spatial and temporal coverage of the Earth observation archives. To enable easy-uptake of the AGDC, and to facilitate future cooperative development, our code is developed under an open-source, Apache License, Version 2.0. This open-source approach is enabling other organisations, including the Committee on Earth Observing Satellites (CEOS), to explore the use of similar data cubes in developing countries.

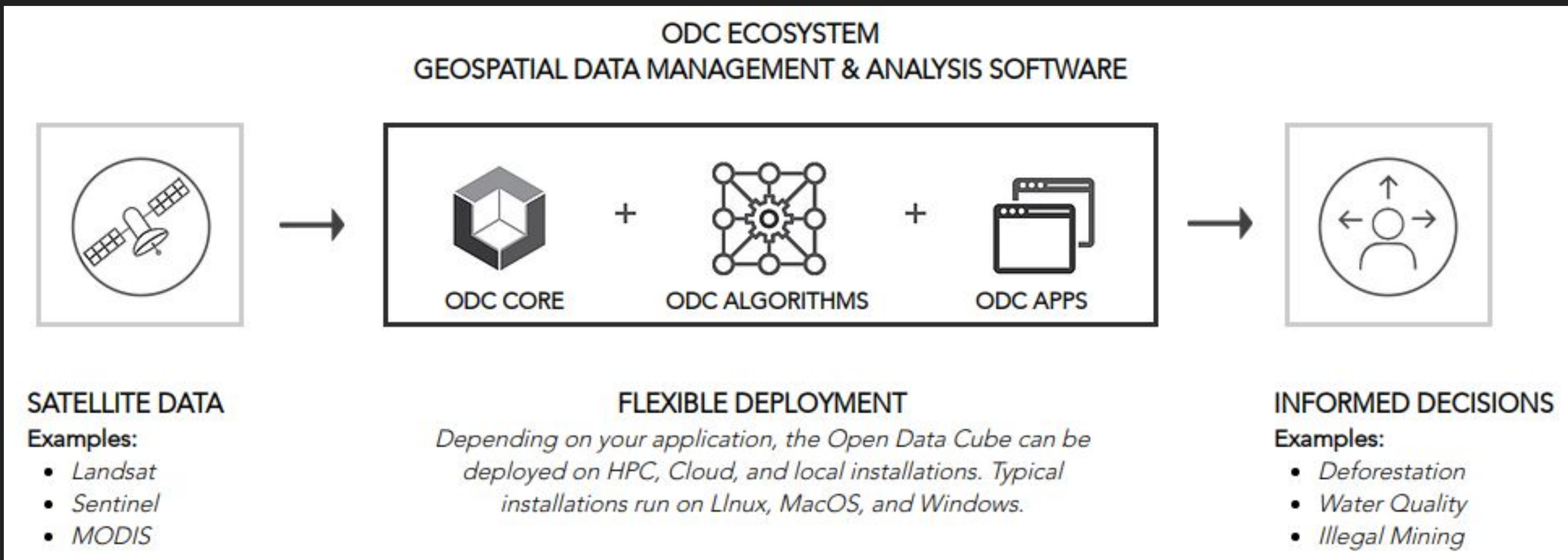
Australian Geoscience DATA CUBE

python™

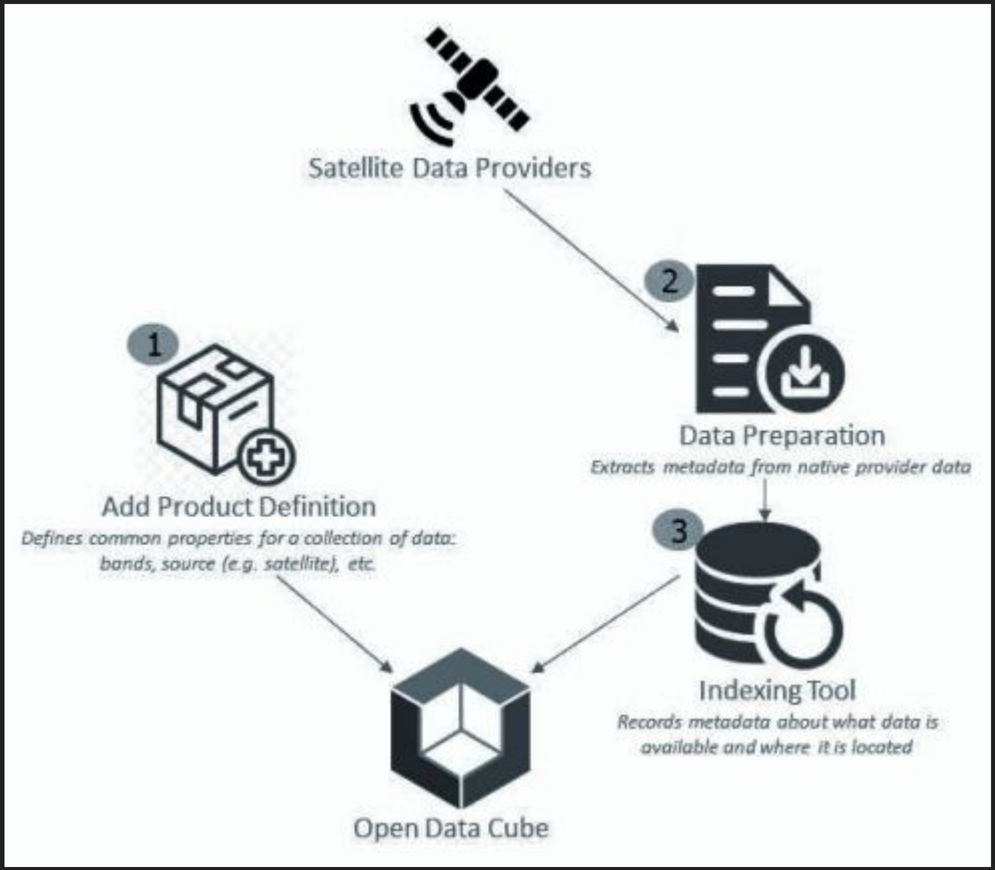


Open Data Cube (ODC)

Open Data Cube (ODC)



Open Data Cube (ODC)



Google Earth Engine (GEE)

Pixel evaluated on visible map
Computation is highly reduced by sub-sampling the data
GEE does not store image data as a data cube but provides cube-like operations, such as reduction over space and time.

GDAL Cubes

data

MDPI

Article

On-Demand Processing of Data Cubes from Satellite Image Collections with the gdalcubes Library

Marius Appel * and **Edzer Pebesma**

Institute for Geoinformatics, University of Münster, Heisenbergstraße 2, 48149 Münster, Germany

* Correspondence: marius.appel@uni-muenster.de

Received: 29 May 2019; Accepted: 26 June 2019; Published: 28 June 2019

check for updates

Abstract: Earth observation data cubes are increasingly used as a data structure to make large collections of satellite images easily accessible to scientists. They hide complexities in the data such that data users can concentrate on the analysis rather than on data management. However, the construction of data cubes is not trivial and involves decisions that must be taken with regard to any particular analyses. This paper proposes on-demand data cubes, which are constructed on the fly when data users process the data. We introduce the open-source C++ library and R package gdalcubes for the construction and processing of on-demand data cubes from satellite image collections, and show how it supports interactive method development workflows where data users can initially try methods on small subsamples before running analyses on high resolution and/or large areas. Two study cases, one on processing Sentinel-2 time series and the other on combining vegetation, land surface temperature, and precipitation data, demonstrate and evaluate this implementation. While results suggest that on-demand data cubes implemented in gdalcubes support interactivity and allow for combining multiple data products, the speed-up effect also strongly depends on how original data products are organized. The potential for cloud deployment is discussed.

Keywords: earth observations; satellite imagery; R; data cubes; Sentinel-2

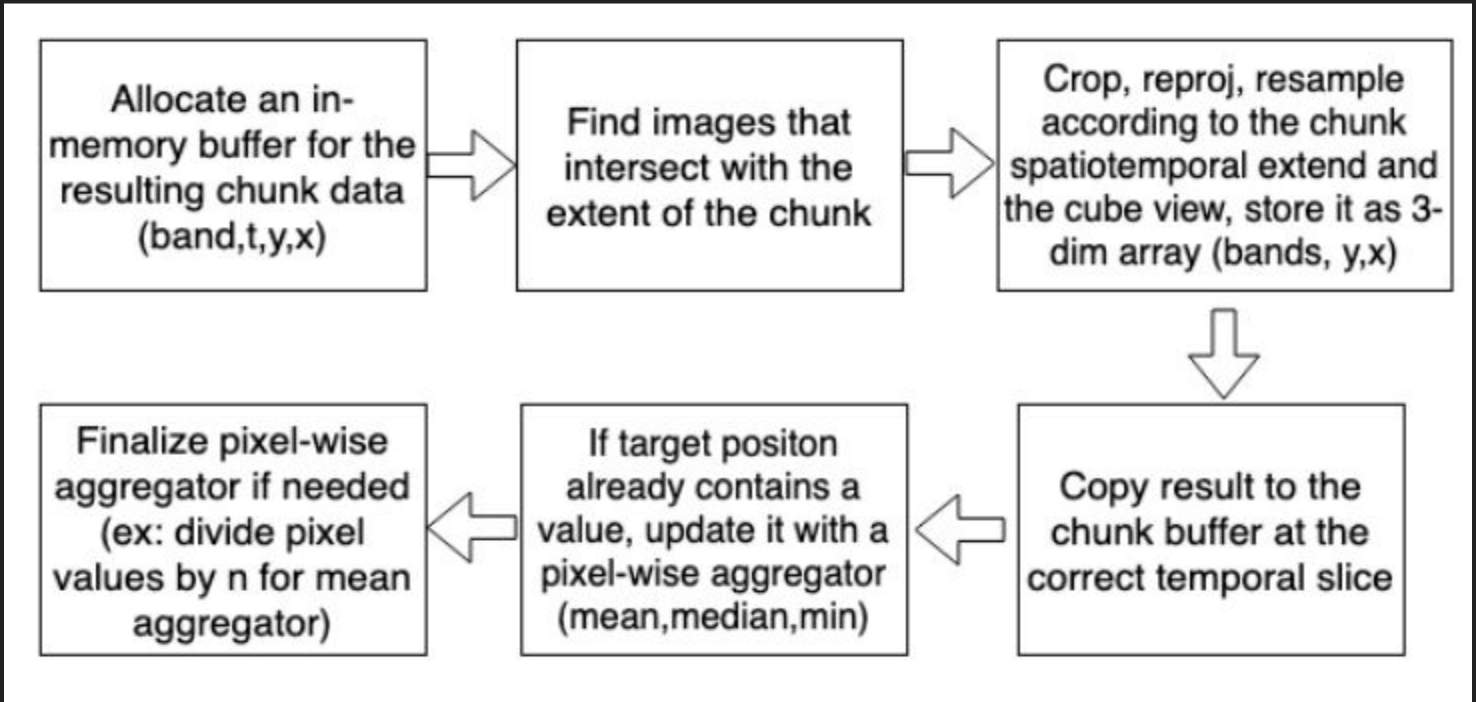
GDAL Cubes

Definition 1. *A regular, dense raster data cube is a four-dimensional array with dimensions x (longitude or easting), y (latitude or northing), time, and bands with the following properties:*

(i) *Spatial dimensions refer to a single spatial reference system (SRS);*
(ii) *Cells of a data cube have a constant spatial size (with regard to the cube's SRS);*
(iii) *The spatial reference is defined by a simple offset and the cell size per axis, i.e., the cube axes are aligned with the SRS axes;*
(iv) *Cells of a data cube have a constant temporal duration, defined by an integer number and a date or time unit (years, months, days, hours, minutes, or seconds);*
(v) *The temporal reference is defined by a simple start date/time and the temporal duration of cells;*
(vi) *For every combination of dimensions, a cell has a single, scalar (real) attribute value.*

Definition 2. *An image collection is a set of n images, where images contain m variables or spectral bands. Band data from one image share a common spatial footprint, acquisition date/time, and spatial reference system but may have different pixel sizes. Technically, the data of bands may come from one or more files, depending on the organization of a particular data product.*

GDAL Cubes


Allocate an in-memory buffer for the resulting chunk data (band,t,y,x)
Find images that intersect with the extent of the chunk
Crop, reproj, resample according to the chunk spatiotemporal extend and the cube view, store it as 3-dim array (bands, y,x)
Copy result to the chunk buffer at the correct temporal slice
If target positon already contains a value, update it with a pixel-wise aggregator (mean,median,min)
Finalize pixel-wise aggregator if needed (ex: divide pixel values by n for mean aggregator)

Brazil Data Cube (BDC)

Brazil Data Cube (BDC)

CUBE COLLECTION: A collection of CUBEs (see bellow) with the following constraints: (a) it is associated with only one Geographic Reference System (GRS) that organizes the spatial dimension; (b) it is associated with a temporal schema (COMPOSITE SCHEMA) that organizes the temporal dimension; (c) it is associated with an attribute set (BANDS) that specifies the metadata of the ASSETs values.

CUBE: A four-dimensional dense matrix (x, y, t, bands), in which any cell have only one value. A CUBE is associated with a TILE (see next).

ASSET: A reference to a two-dimensional dense matrix (x, y) associated to a time (t) and a band of a CUBE ITEM.

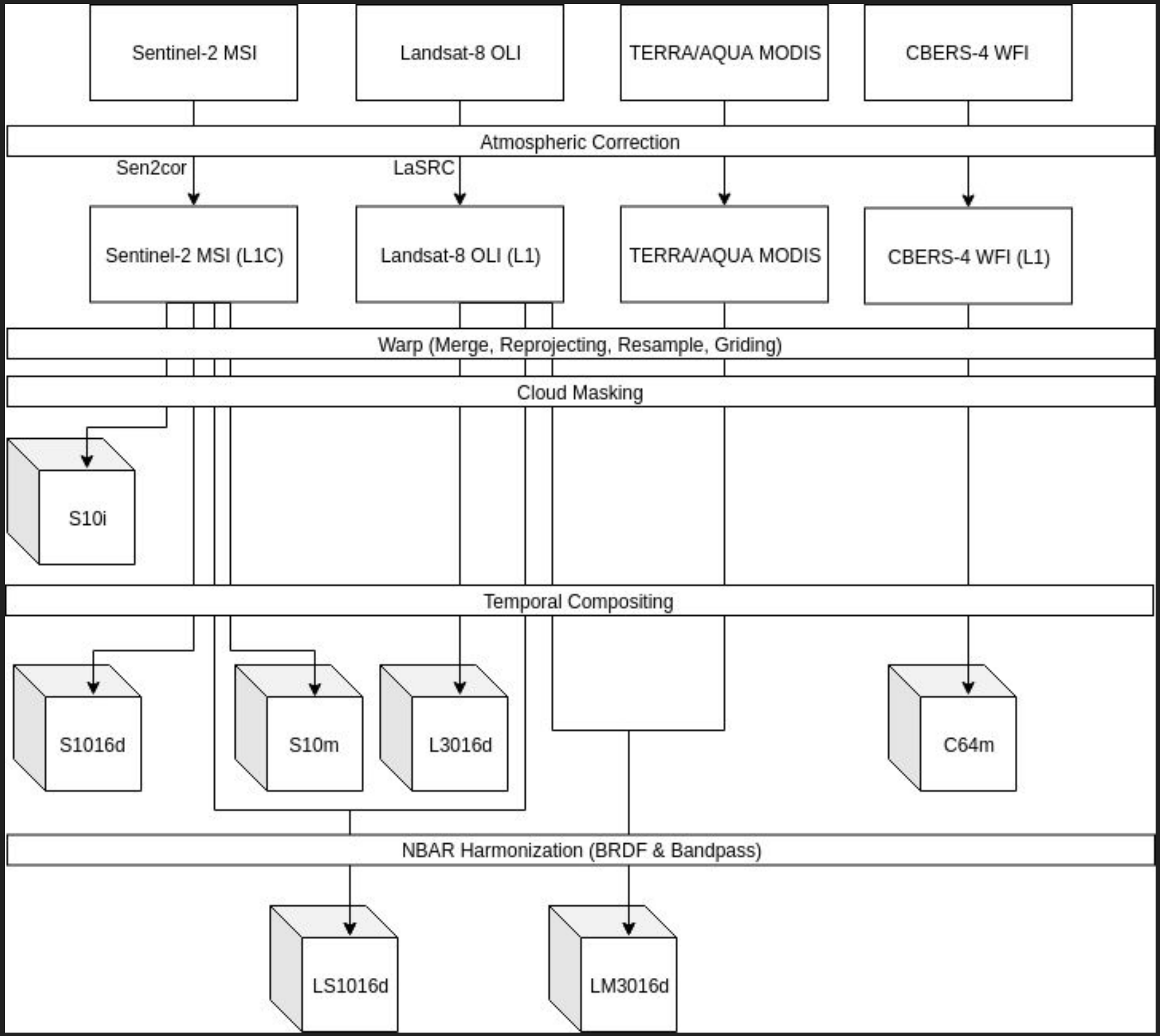
BAND: A description of the matrix band values and data storage.

GRS: A geographic reference system that divides the space into regions (TILE) and uses a code system, specified by a set of numbers, letters, or symbols, to identify an unique region.

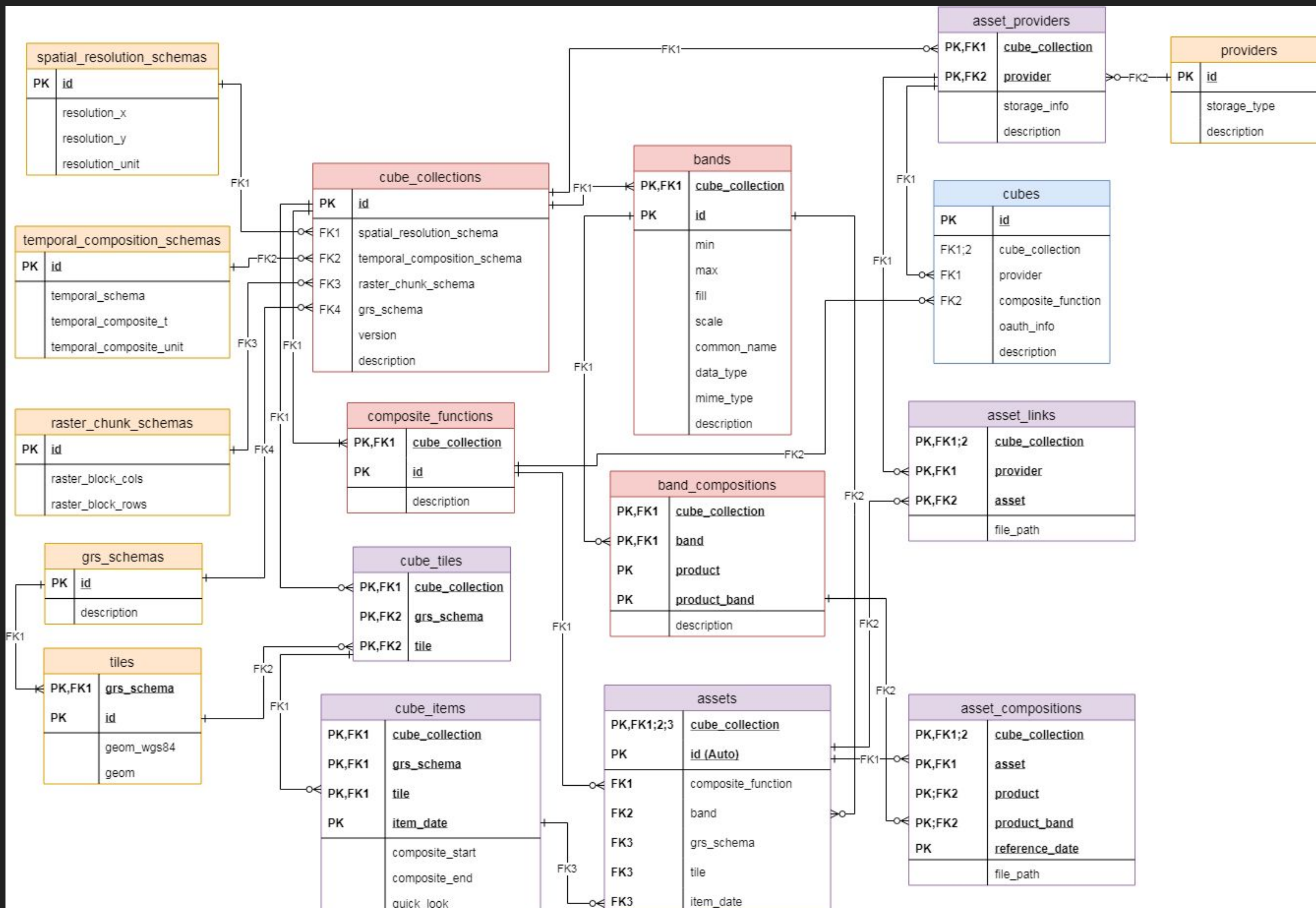
TILE: It is an element of a GRS, associated with a unique code, that represents a geographic region.

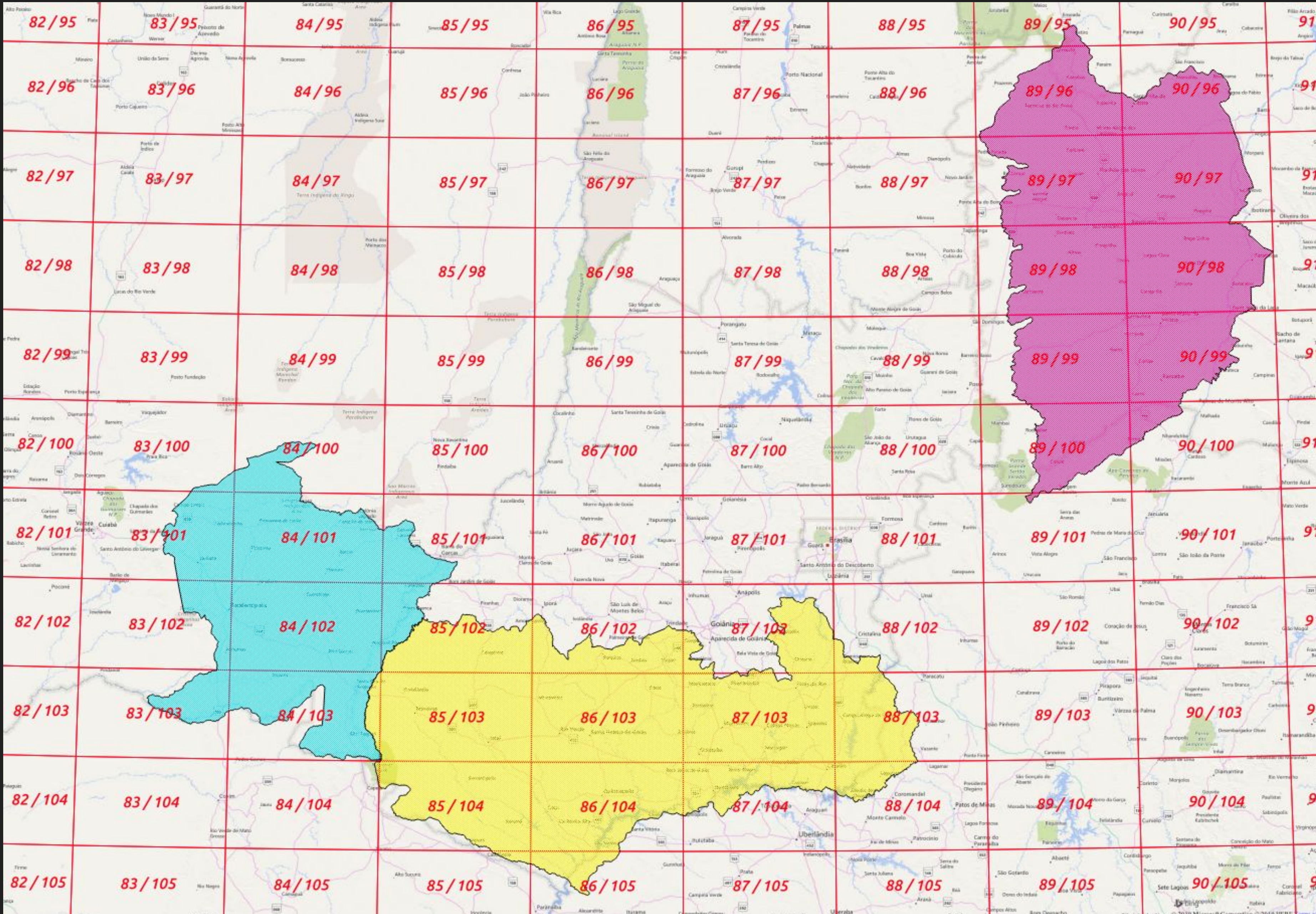
COMPOSITE FUNCTION: Is a function that chooses a pixel value, according to a heuristic, from a set of pixels contained within a time interval.

Brazil Data Cube (BDC)



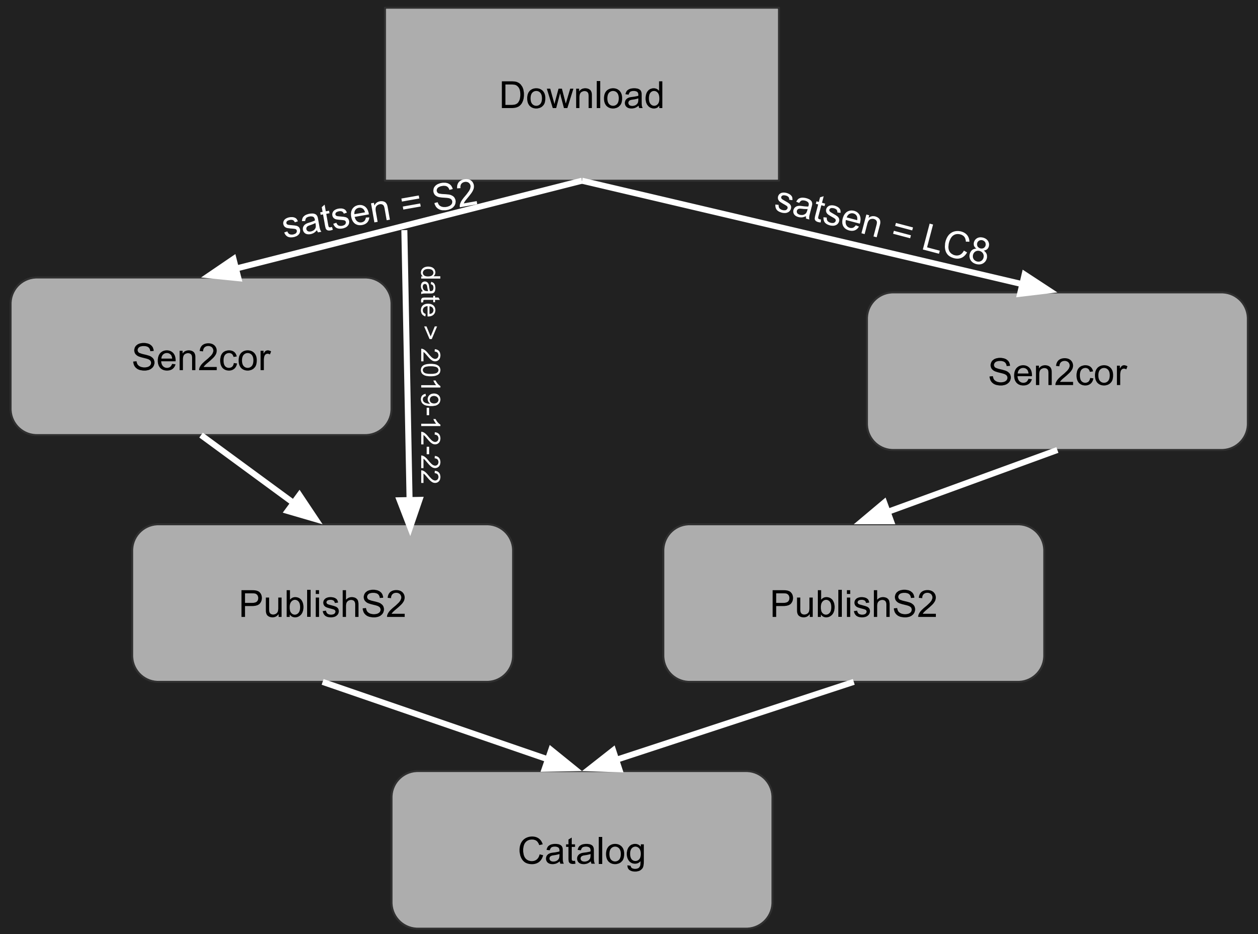
Brazil Data Cube (BDC)



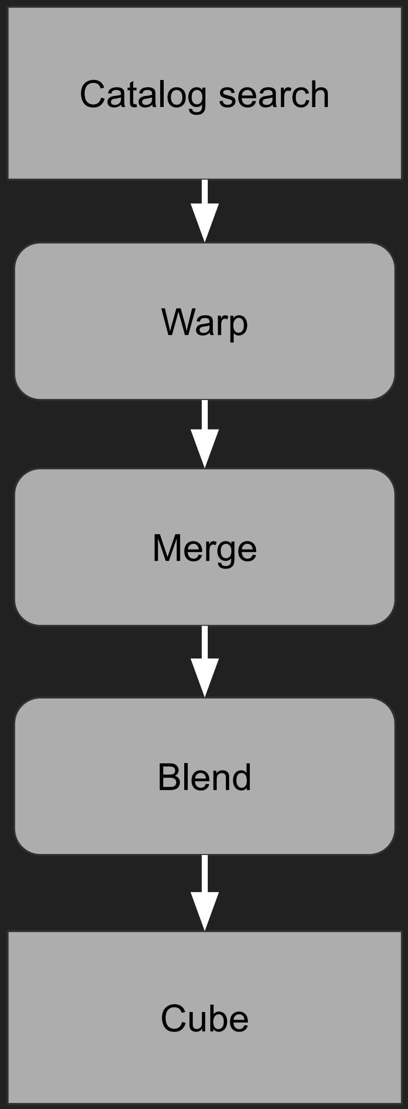


82 / 95
83 / 95
84 / 95
85 / 95
86 / 95
87 / 95
88 / 95
89 / 95
90 / 95
82 / 96
83 / 96
84 / 96
85 / 96
86 / 96
87 / 96
88 / 96
89 / 96
90 / 96
82 / 97
83 / 97
84 / 97
85 / 97
86 / 97
87 / 97
88 / 97
89 / 97
90 / 97
82 / 98
83 / 98
84 / 98
85 / 98
86 / 98
87 / 98
88 / 98
89 / 98
90 / 98
82 / 99
83 / 99
84 / 99
85 / 99
86 / 99
87 / 99
88 / 99
89 / 99
90 / 99
82 / 100
83 / 100
84 / 100
85 / 100
86 / 100
87 / 100
88 / 100
89 / 100
90 / 100
82 / 101
83 / 101
84 / 101
85 / 101
86 / 101
87 / 101
88 / 101
89 / 101
90 / 101
82 / 102
83 / 102
84 / 102
85 / 102
86 / 102
87 / 102
88 / 102
89 / 102
90 / 102
82 / 103
83 / 103
84 / 103
85 / 103
86 / 103
87 / 103
88 / 103
89 / 103
90 / 103
82 / 104
83 / 104
84 / 104
85 / 104
86 / 104
87 / 104
88 / 104
89 / 104
90 / 104
82 / 105
83 / 105
84 / 105
85 / 105
86 / 105
87 / 105
88 / 105
89 / 105
90 / 105

Brazil Data Cube (BDC): Radcor

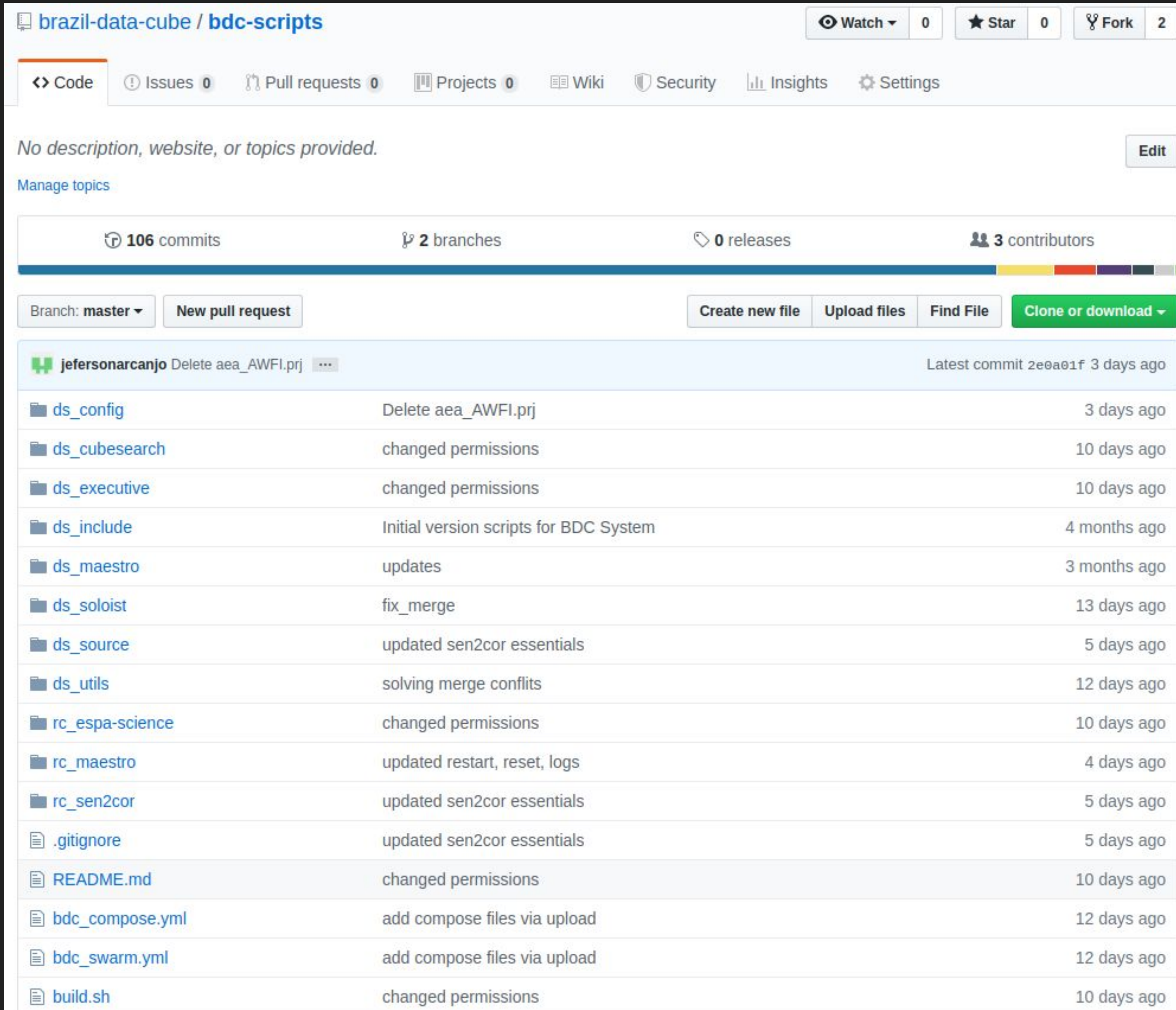


Brazil Data Cube (BDC): Data Storm



GitHub

https://github.com/brazil-data-cube/bdc-scripts



brazil-data-cube / bdc-scripts
Watch 0
Star 0
Fork 2
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Security
Insights
Settings
No description, website, or topics provided.
Edit
Manage topics
106 commits
2 branches
0 releases
3 contributors
Branch: master
New pull request
Create new file
Upload files
Find File
Clone or download
jefersonarcanjo Delete aea_AWFI.prj ···
Latest commit 2e0a01f 3 days ago
ds_config Delete aea_AWFI.prj 3 days ago
ds_cubesearch changed permissions 10 days ago
ds_executive changed permissions 10 days ago
ds_include Initial version scripts for BDC System 4 months ago
ds_maestro updates 3 months ago
ds_soloist fix_merge 13 days ago
ds_source updated sen2cor essentials 5 days ago
ds_utils solving merge conflits 12 days ago
rc_espa-science changed permissions 10 days ago
rc_maestro updated restart, reset, logs 4 days ago
rc_sen2cor updated sen2cor essentials 5 days ago
.gitignore updated sen2cor essentials 5 days ago
README.md changed permissions 10 days ago
bdc_compose.yml add compose files via upload 12 days ago
bdc_swarm.yml add compose files via upload 12 days ago
build.sh changed permissions 10 days ago

README.md

BDC-Scripts - Brazil Data Cube

Infrastructure for creating brazil data cube environment in a local server machine

Structure

- `ds_config` : Definitions and system configs (Reference Grids, Spetral bands mapping, etc)
- `ds_cubesearch` : Web interface to view datacubes, download products, etc.
- `ds_executive` : Web interface for management of datacubes (definition, creating, execution)
- `ds_include` : Include files needed by the system
- `ds_maestro` : Docker and components that manage the cube generation system (manage queues, tasks)
- `ds_opensearch` : API opensearch to the original files repository
- `ds_soloist` : Docker and components that execute datacube tasks
- `ds_source` : Python source codes
- `ds_utils` : Python API for accessing data cube and general tools

Requirements

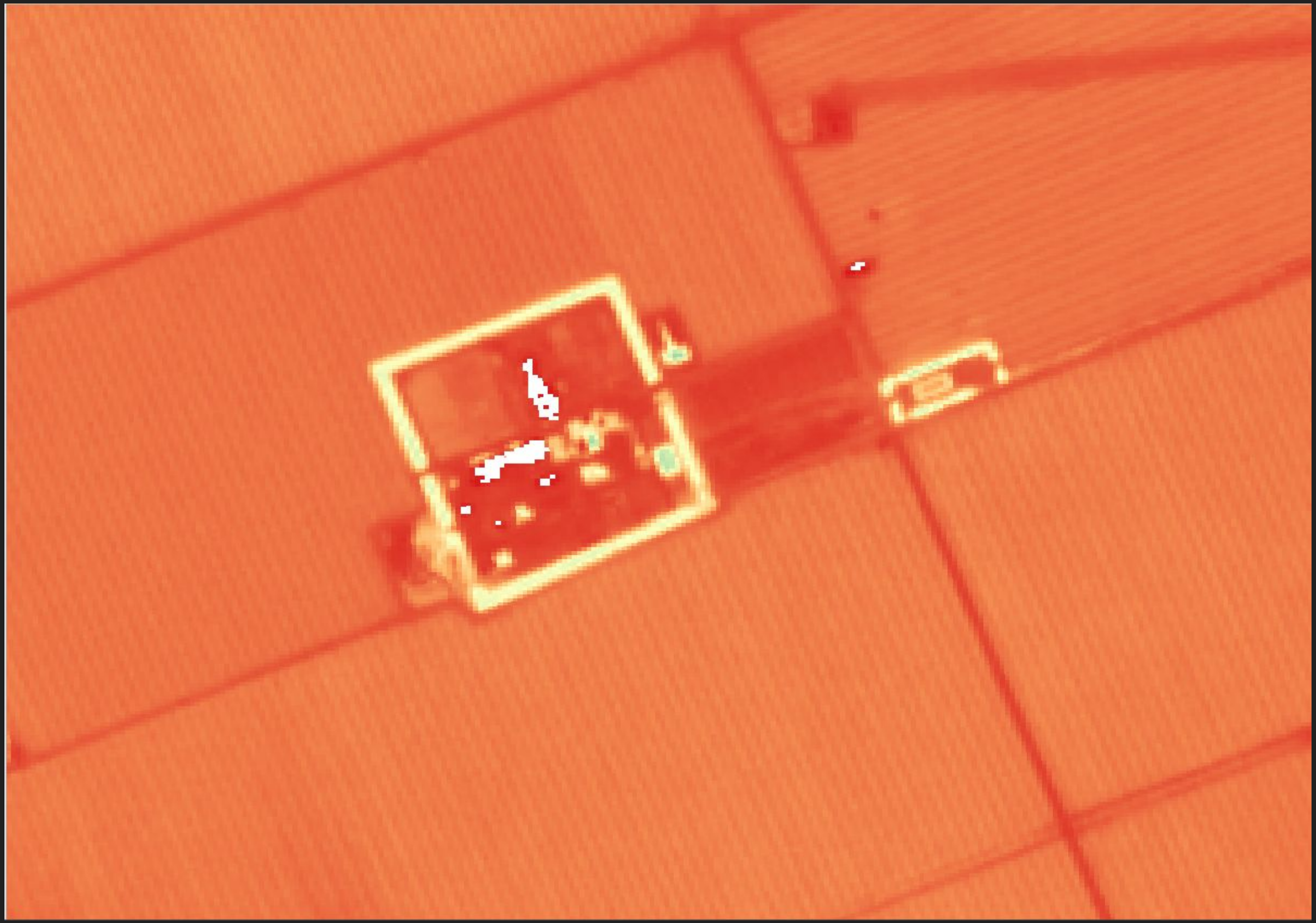
Make sure you have the following libraries and tools installed:

- `Python 3`
- Docker Docker version 18.06.1 or later
- Docker Compose docker-compose version 1.22.0 or later

Installation

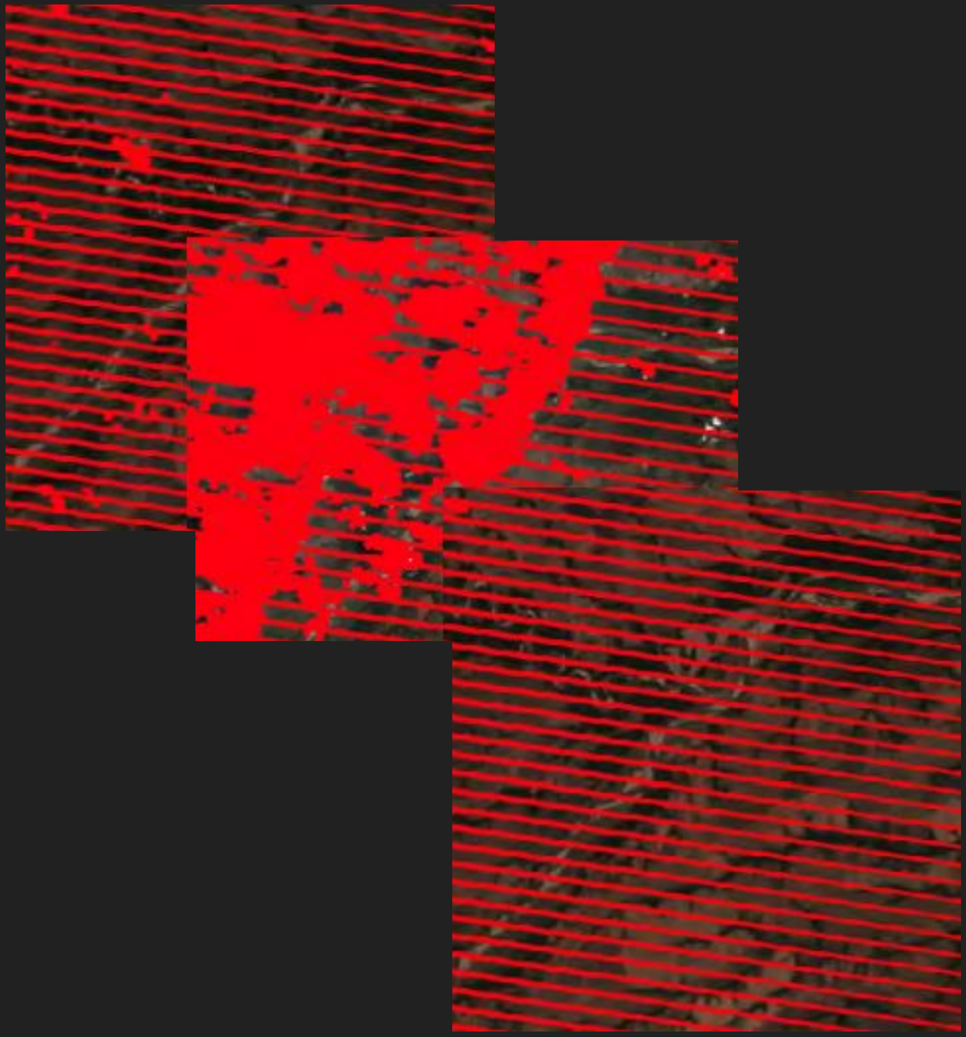
Running with docker

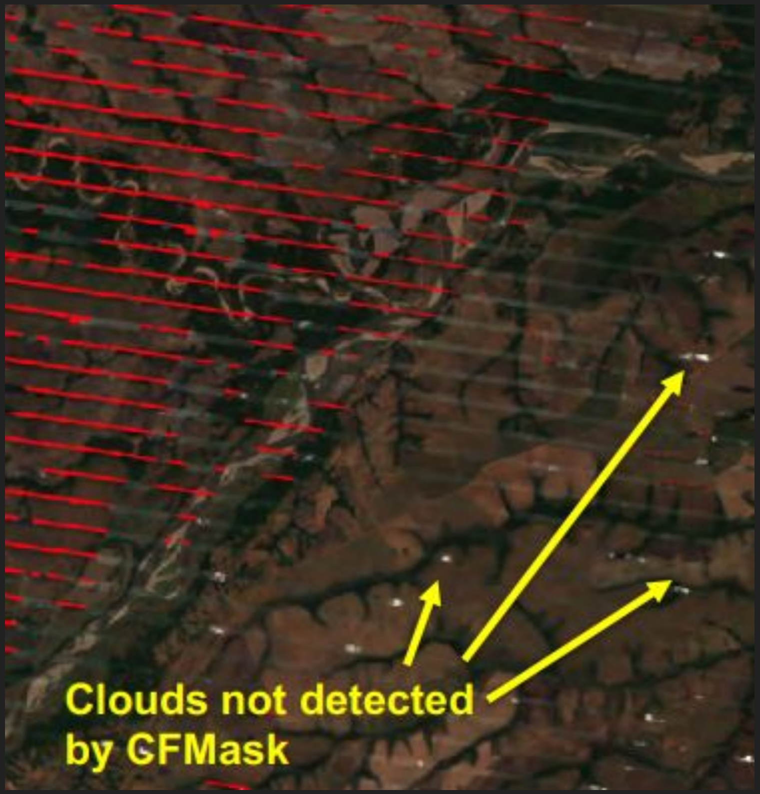
Cloud mask removing



Register

Cloud Mask

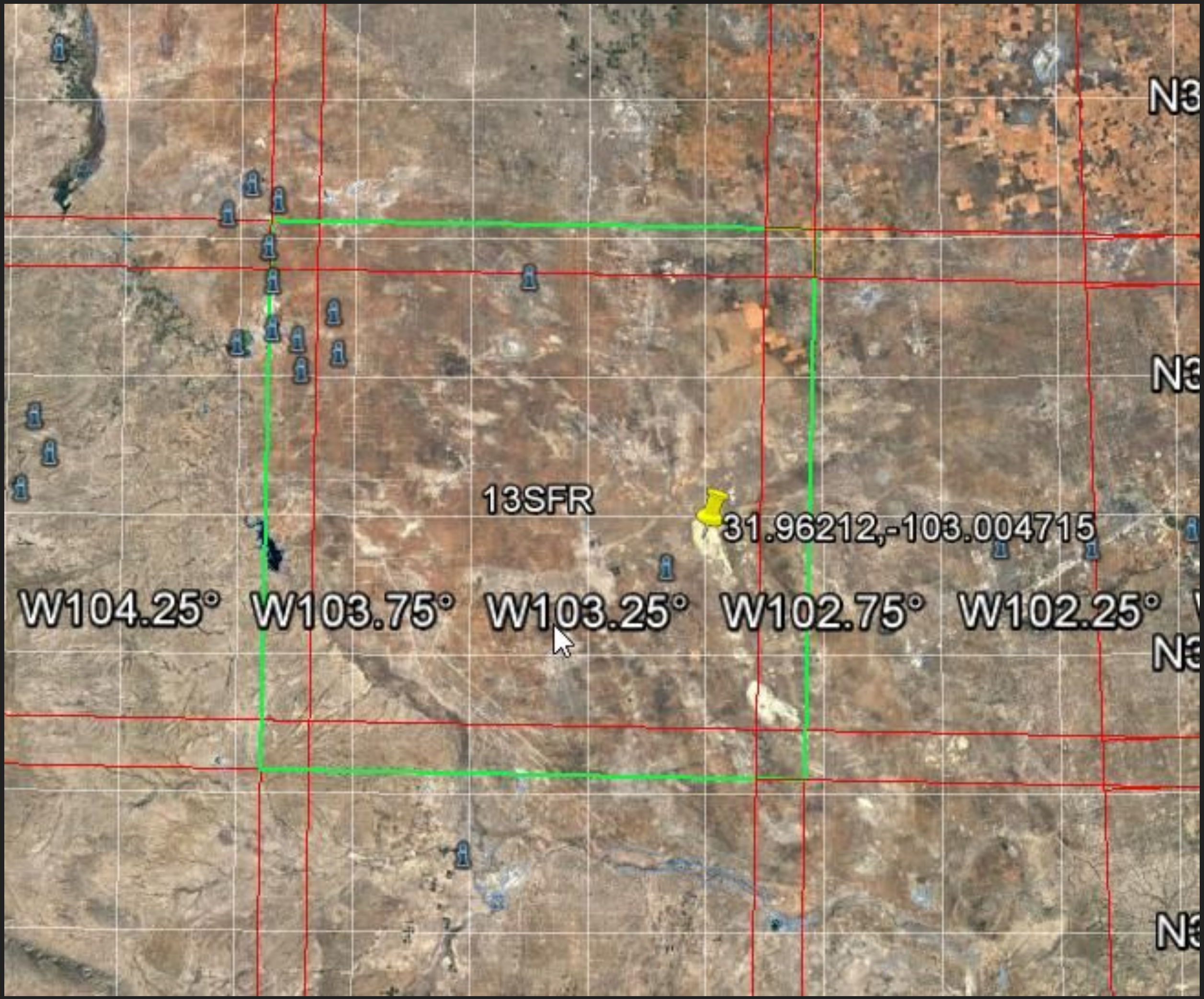




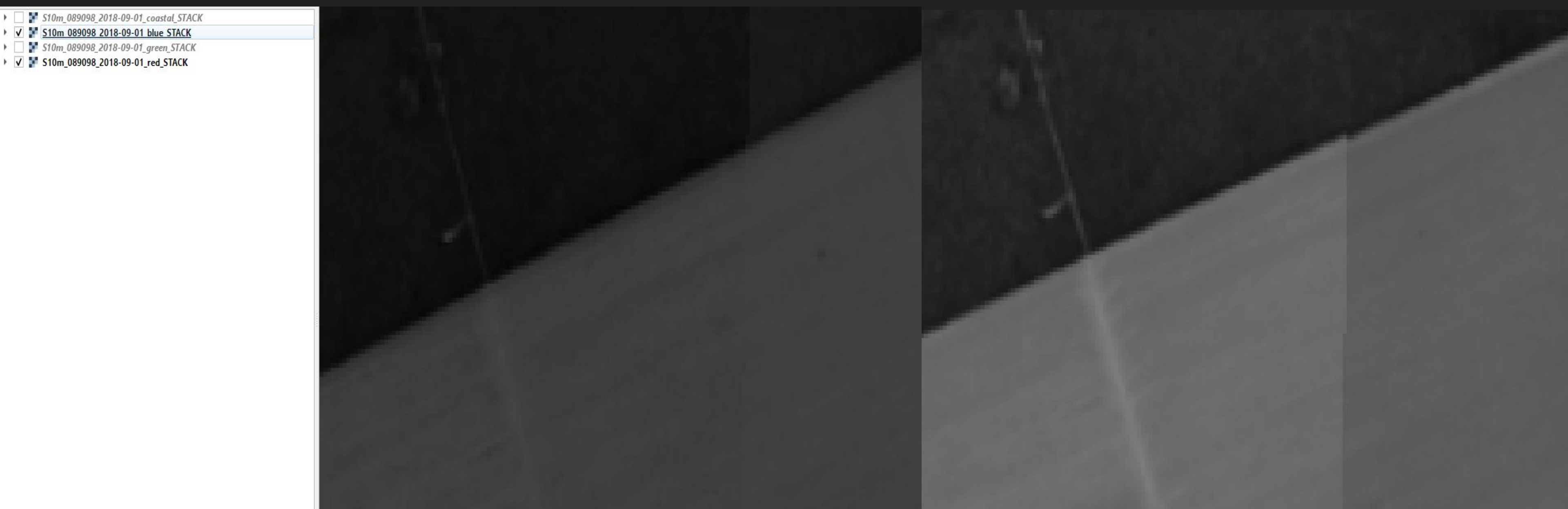
Cloud Mask

Cloud Shadow Mask

Sentinel Overlap



Sentinel Overlap



Data Provenance