OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet

Johannes Heisig

Yu-Feng Ho

APACHE ARROW

Parquet

ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

OPEN EARTH MONITOR

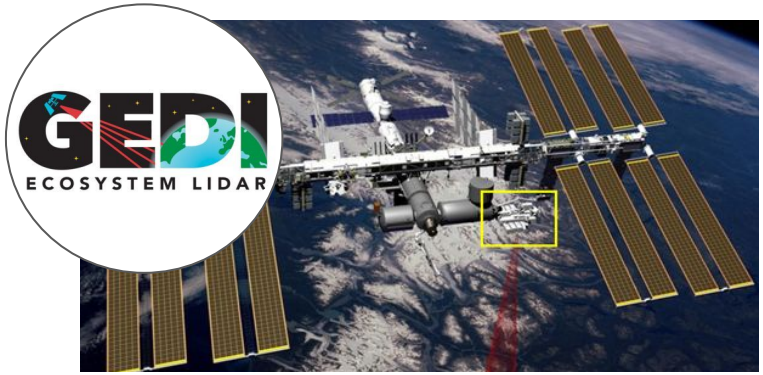
Outline

1. The data
2. Apache Arrow & Parquet
3. Cloud Setup
4. Notebook examples

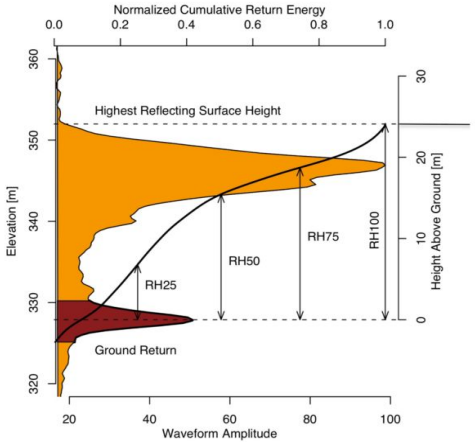
Funded by the European Union

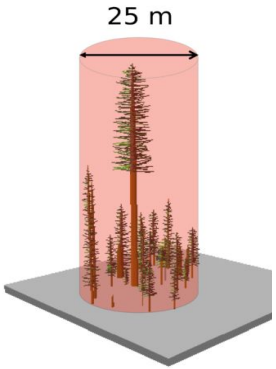
OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet

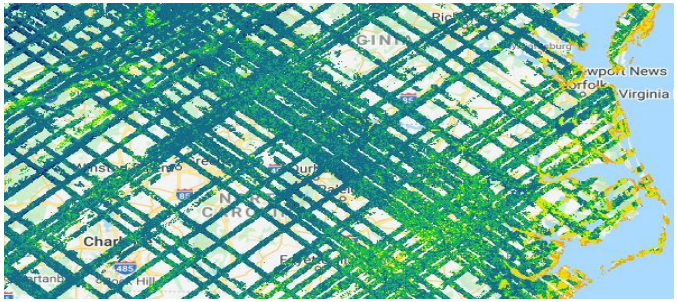


- Full-waveform LiDAR system
- Data since 2019
- N ~ 5.4 billion points
- Level 2A+B data, 94 variables
- Point geometries (lon/lat)





Sampling pattern



ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet

In this example of a Version 2 Level 01B product, the filename **GEDI01_B_2019110110221_O01997_03_T03335_02_005_01_V002.h5** indicates:

- **GEDI01_B** = Product Short Name
- **2019110** = Julian Date of Acquisition in YYYYDDD
- **110221** = Hours, Minutes, and Seconds of Acquisit
- **O01997** = O = Orbit, 01997 = Orbit Number
- **03** = Sub-Orbit Granule Number
- **T03335** = T = Track, 033335 = Track Number
- **02** = Positioning and Pointing Determination Syster
- **005** = PGEVersion = SDPS Release Number
- **01** = Granule Production Version
- **V002** = Version Number
- **.h5** = Data Format

GEDI ECOSYSTEM LIDAR

NASA

ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

Accessing Big Vector Data on the Cloud using Arrow Parquet

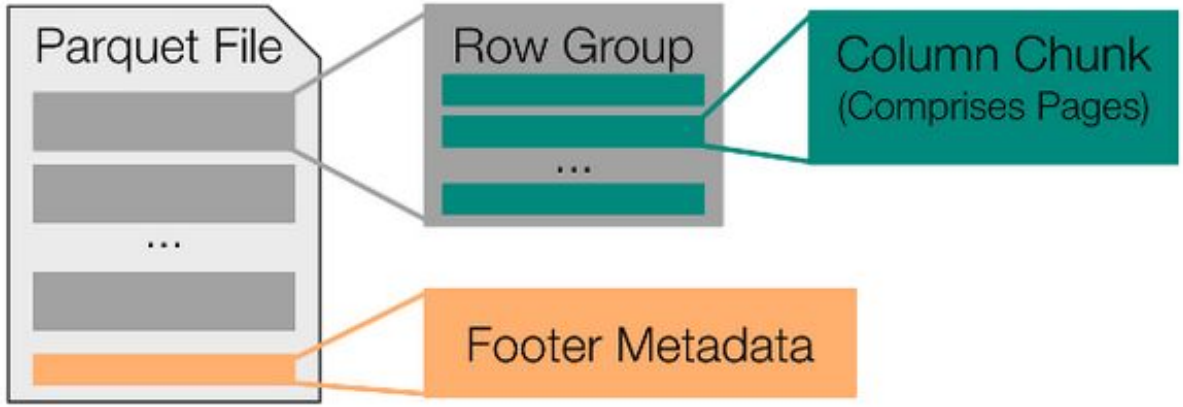


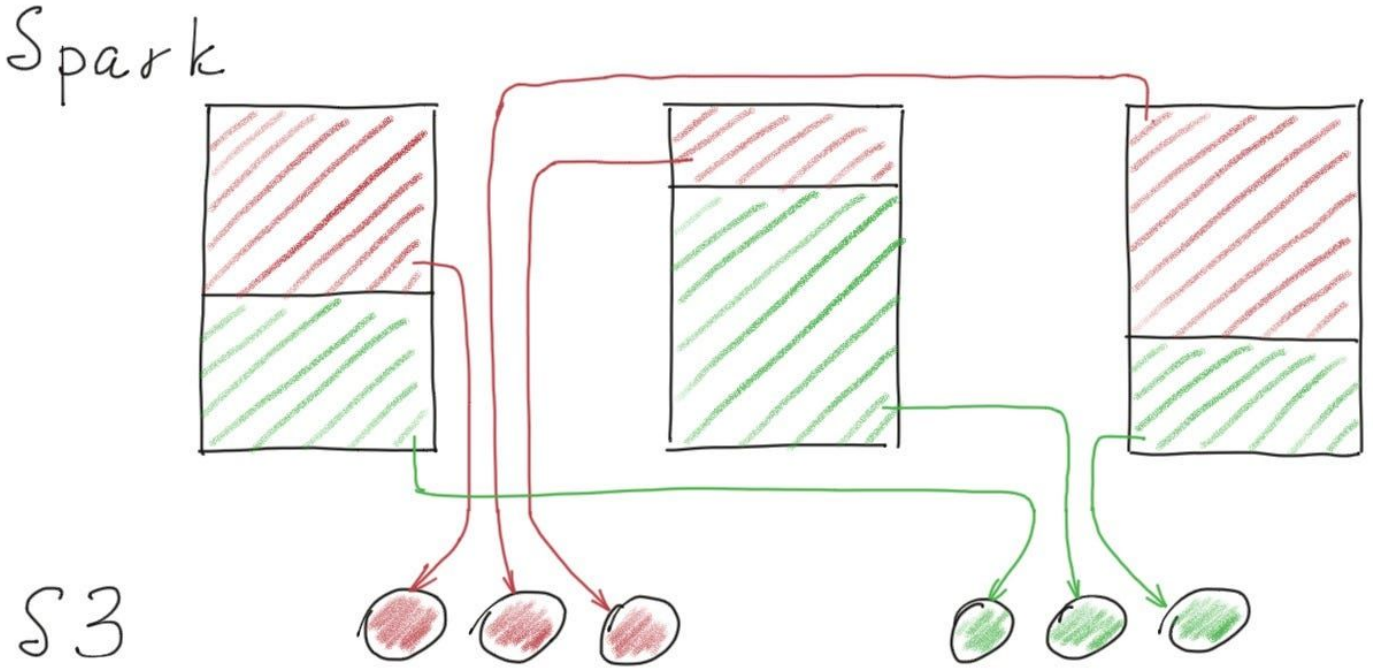
Figure 1. High-level visual representation of the Parquet file format

Dataset
|--- Partition
  |--- File
    |--- Row group
      |--- Column chunk
        |--- Page

```
httpfs = S3FileSystem(
      endpoint_url='https://s3.eu-central-1.wasabisys.com',
      anon=True
   )
pq_dataset = pq.ParquetDataset(
    path_or_paths='gedi-ard/level2/l2v002.gedi_20190418_20230316_go_epsg.4326_v20240622',
    filesystem=httpfs)
CPU times: user 634 ms, sys: 0 ns, total: 634 ms
Wall time: 1.35 s
```

Source:
(1) Optimizing Access to Parquet Data with fsspec

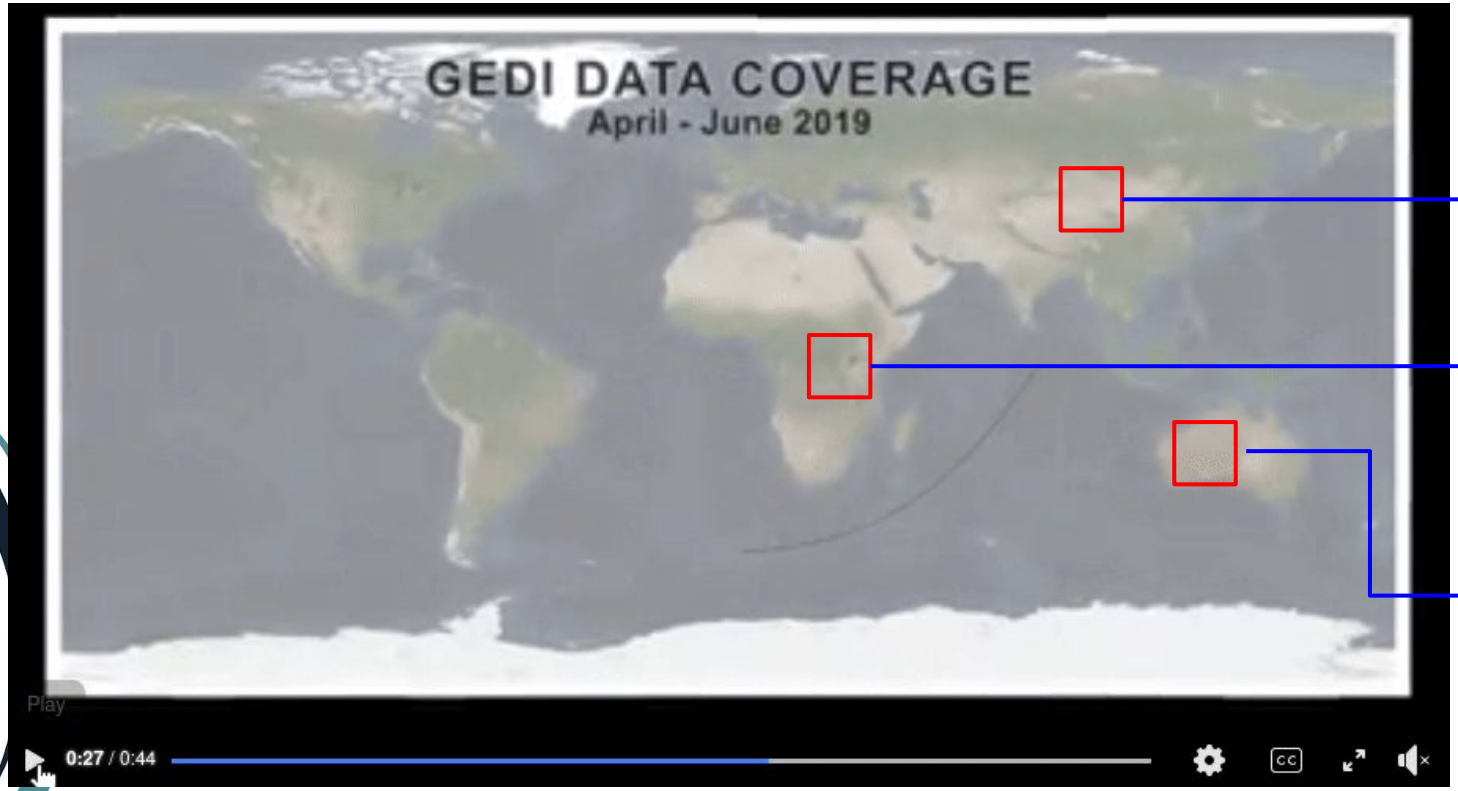
Accessing Big Vector Data on the Cloud using Arrow Parquet



Source:
[Spark partitioning: the fine print, Vladimir Prus, 2021]

OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet

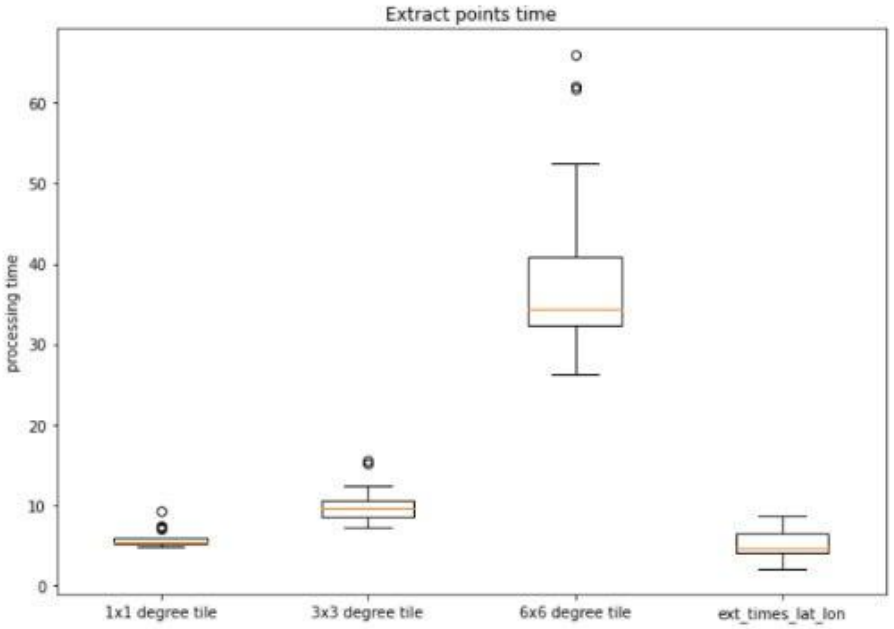


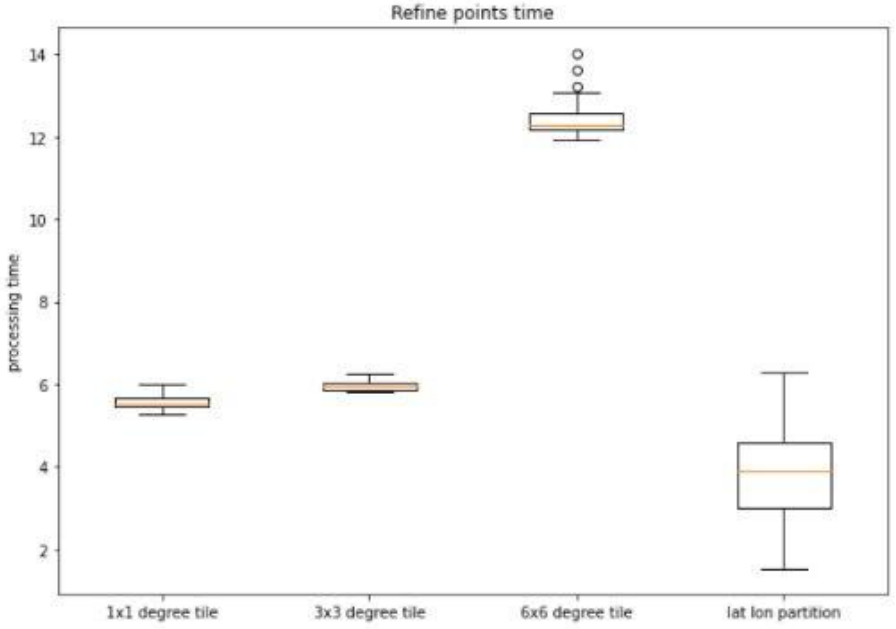
partition-1
- gedi-o1a.parquet
- gedi-o2a.parquet

partition-2
- gedi-o1b.parquet
- gedi-o2b.parquet
- gedi-o3a.parquet
- gedi-o4a.parquet

partition-3
- gedi-o3b.parquet
- gedi-o4b.parquet
- gedi-o5a.parquet

Accessing Big Vector Data on the Cloud using Arrow Parquet

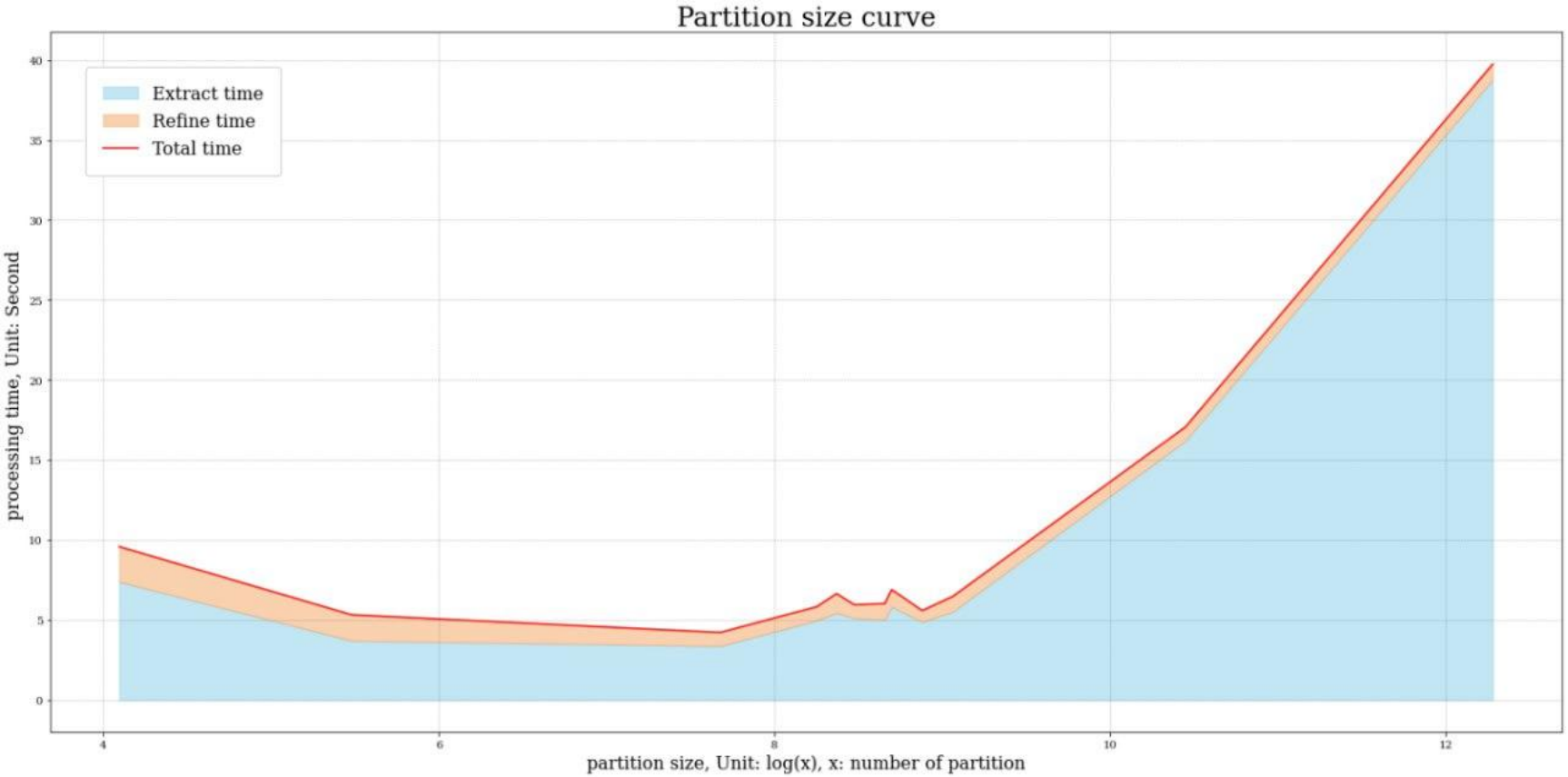




The multi depth structure(lat in 1 degree / lon in 1 degree) outperforms flattened structure (1x1 degree grid)

OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet



ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

Accessing Big Vector Data on the Cloud using Arrow Parquet

Comparison of Python Parquet Engine in Parallelisation and Cloud Object Storage

Stats for all stories

6 min read · Dec 1, 2023

5

Lifetime

Dec 1, 2023 – Today · Updated every 24 hours

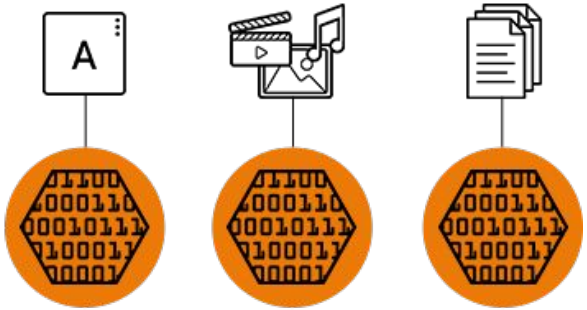
536	337	63%
Views	Reads	Read ratio

pyarrow.dataset.write_dataset

```
pyarrow.dataset.write_dataset(data, base_dir, *, basename_template=None,
format=None, partitioning=None, partitioning_flavor=None, schema=None,
filesystem=None, file_options=None, use_threads=True, max_partitions=None,
max_open_files=None, max_rows_per_file=None, min_rows_per_group=None,
max_rows_per_group=None, file_visitor=None, existing_data_behavior='error',
create_dir=True)                                                    [source]
```

Write a dataset to a given format and partitioning.

Accessing Big Vector Data on the Cloud using Arrow Parquet



Cloud Object Storage

```
        url = 'http://s3.eu-central-1.wasabisys.com/gedi-ard/level2/l2v002.gedi_20190418_20230316_go_epsg.4326_v20240614/lon=15/lat=5/
year=2021/gedi_l2_0.parquet'
        q =  pl.scan_parquet(url)
        if cols is not None:
            q = q.select(cols)
        q = q.filter((pl.col('longitude') >= bbox[0]) & (pl.col('longitude') <= bbox[2]) &
                     (pl.col('latitude') >= bbox[1]) & (pl.col('latitude') <= bbox[3]))
```

Source:

(1) What is object storage? and File storage, block storage, or object storage?

(2) Polars — DataFrames for the new era

OPEN EARTH MONITOR

Accessing Big Vector Data on the Cloud using Arrow Parquet

GlobalEarthPoint Public

Edit Pins | Watch 1 | Fork 0 | Star 1

main | 1 Branch | 0 Tags | Go to file t | Add file | Code

yu-feng-ho Update Readme.md — f0112ca · 12 minutes ago — 14 Commits

Python	python tutorial update	50 minutes ago
R	automatic package install	1 hour ago
.gitignore	rearrange repo structure	18 hours ago
LICENSE	Initial commit	5 months ago
Readme.md	Update Readme.md	12 minutes ago

README | Apache-2.0 license

Access the repository here:
https://github.com/Open-Earth-Monitor/GlobalEarthPoint

GlobalEarthPoint

GlobalEarthPoint is a **Open Source Data Service Library** that specializes in managing large datasets of geographical points, providing a portal to access cloud infrasturcture, subset and retrieve data efficiently.

Key Features

- **Cloud Optimization**: Data is stored using ArrowParquet in the partitioning structure, and retrieved with Lazy evaluaton.
- **High Efficiency**: Designed to handle massive datasets by Parquet format, minimizing data size, latency and maximizing throughput.
- **Easy Integration**: Compatible with popular data processing frameworks and geospatial tools, facilitating easy

About

This initiative is inclined to address global vector (point) data accessing and hosting in a cloud-native environment.

- Readme
- Apache-2.0 license
- Activity
- Custom properties
- 1 star
- 1 watching
- 0 forks

Report repository

eleases

Create a new release

Packages

No packages published
Publish your first package

Contributors 2

yu-feng-ho

joheisig Johannes Heisig

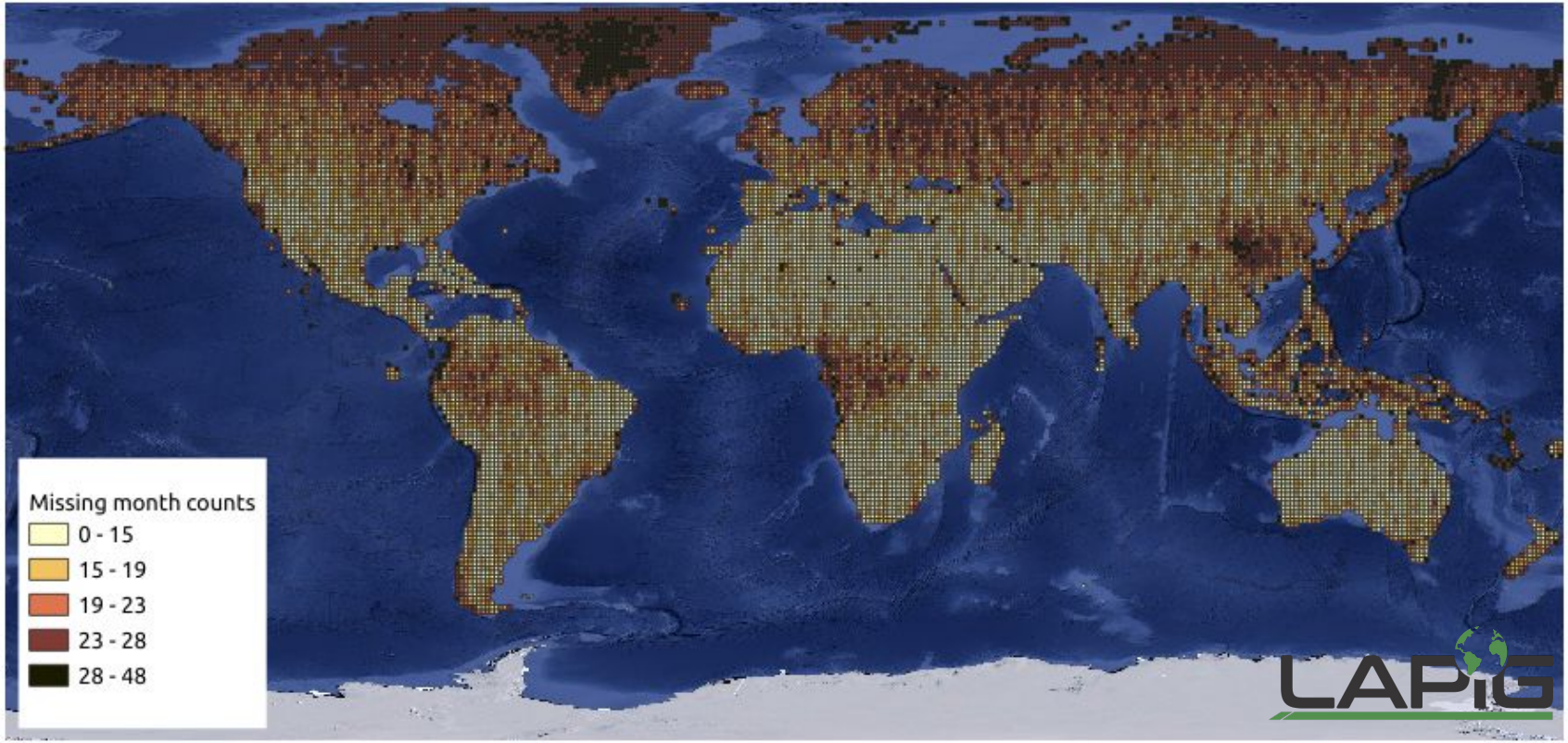
ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

OPEN EARTH MONITOR

Use Cases of Big Vector Data - Short vegetation mapping



ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

Use Cases of Big Vector Data - Short vegetation mapping

Icesat-2 ATL08v006 in 20m segment

24.8 billions

~ 4TB

Generating extra attributes

- **n_ph_20m**: Number of photons in 20m
- **med_ht**: Median of vegetation photons relative height

Filter

By:

- **n_ph_20m >= 3**: number of photons in 20m greater than or equal to 3
  *separate segments with valid data
- **med_ht <= 5**: median less than or equal to 5m
  *focus sample on areas dominated by low vegetation
- **night_flag == 1**: data acquired at night
  *reduction in RMSE of more than 1m

Processing in 2 hours with a HPC 96 cores

Icesat-2 ATL08v006 in 20m segment for Short Vegetation Mapping

4.9 billions

Production Process

LAPIG

ifgi Institut für Geoinformatik Universität Münster

OpenGeoHUB Connect · Create · Share · Repeat

Funded by the European Union

OPEN EARTH MONITOR

Questions & Feedback

ifgi
Institut für Geoinformatik
Universität Münster
OpenGeoHUB
Connect • Create • Share • Repeat
Funded by the European Union