Cloud Optimized Geospatial Formats

Status Report

March 2024

Context

This report has been developed as supporting material within the framework of the Swiss Geoinformation Strategy (SGS). It reviews the current status of cloud optimized formats in the geospatial domain. eCH-0056 which is produced by GeoStandards.ch serves as a guide for the implementation of geoservices in Switzerland. In this context, the potential of cloud optimized data formats for enhancing the access to geospatial information in Switzerland is being explored. This report supports this effort by summarizing the current state of technology and standards.

Basics

Cloud optimized formats are becoming increasingly more popular. These formats allow publishing large amounts of data in a way that subsets can be accessed by clients like web, desktop or mobile applications with a minimal infrastructure. In contrast to traditional WxS or OGC API X services, they do not need a server side application to be running. Instead, simple files are stored on storage accessible through the http protocol which is cheap to run and low in maintenance, often – a bit misleading – referred to as serverless.

Cloud optimized formats are also one of the areas in the IT and geospatial IT industry where we see a lot of movement, so things evolve rapidly.

Technical

Cloud optimized formats rely on a technology called streaming, made possible by RANGE requests. This allows a client to access data at specific positions in a file. This is well known from web mp3 or video players where it is possible to directly skip to specific positions within a track or clip without downloading the complete file.

Cloud optimized formats make use of smart organization of data within the file to leverage streaming. By putting a table of content (or index) at the beginning of the file, a client can download just a few bytes from this table of content which contains information of where inside the file certain data is located. In the geospatial context, files will also normally be organized in a way that information that is nearby on the earth is also stored next to each other inside the file, a typical example for this is tiling.

Another typical buzzword that is being mentioned in the area of cloud native formats is zero copy. This means that a client does not need to interpret the data and copy the interpreted version at another place in the memory. Zero copy cloud optimized formats are normally binary formats, and not directly human readable.

Motivation

One of the main reasons for using cloud optimized files is that the infrastructure cost and maintenance effort is very low, as it is created around basic building blocks of the web (like http).

Cloud optimized formats are optimized for … the cloud. They therefore allow making archives of data from different sources and domains available in a way that is machine readable and ready to use in cloud computing. This will allow for new use cases where diverse data can be combined and drive innovation. This is most interesting for big quantities of data, this is a reason why a lot of the development in this sector has been driven by the earth observation sector.

Basic considerations, recommendations and caveats

- Cloud optimized formats are well suited for publishing static data. This is data that is written once and then read many times. It should not be used for living data that is being updated.
- Cloud optimized formats are well suited for publishing complete datasets. Authorization can be done on file level but not on individual objects, regions, attributes.
- The global GIS industry is currently discussing cloud native formats and implementing tools at a high speed. Many of the resources considered in this report are from 2023 and as such quite young. It is to be expected that in the next months and years the community will learn and evolve these technologies.
- As part of the global GIS industry movement, the OGC is having working groups and events on the topic. It is worth keeping an eye on OGC activities and standards[1][2] and follow this movement for interoperability and synergies.
- Cloud optimized formats' real strength is in big data, where traditional access patterns like downloading a complete file is not appropriate. For small sized data a publication in more readable formats like CSV or GeoJSON or database formats like GeoPackage only or as a complement is worth considering.
- For distributing data over http, a CDN can be interesting as this reduces latency for data access and will help to handle sudden peaks in data access. As multiple requests will regularly need to be made, often round-trip latency, rather than throughput, is the limiting factor. A caching CDN can be especially helpful here. Fetching a subset of a file over HTTP utilizes Range requests. If the page accessing the data is hosted on a different domain from the CDN, Cross Origin policy applies,

[1] https://www.ogc.org/standards/

[2] https://www.ogc.org/standards/community/

and the required `Range` header will induce an `OPTIONS` (preflight) request. If using a CDN, make sure this can also be cached.[3]

Raster data

Cloud Optimized GeoTIFF

Cloud Optimized GeoTIFF (COG) was one of the first formats to be developed and one of the most obvious candidates.

As raster data can very well be tiled and overviews can be generated, the approach is a natural evolution step.

One of the advantages of COG and a driver for the quick market adoption was its backward compatibility. A COG file can still be downloaded as a complete file and be opened by applications that can read TIFF files, even without making use of its cloud optimized characteristics through streaming.

It is also an approved OGC standard and therefore a good choice for usage.

Vector data

When it comes to vector data, the requirements towards a cloud optimized format are a lot more diverse.

With both tiling and overviews additional challenges arise.

For creating overviews, a generalization needs to be made, which will involve eliminating features at smaller scales. The decision which features to eliminate is non-trivial.

For creating tiles, larger features can overlap many tiles and can therefore not simply be assigned to a single tile.

We therefore have often very different format specifications for particular use cases of “visualization” and “analysis”. The decision for cloud native vector formats also sometimes depends on specific requirements for datasets, like geometry-less features and other edge cases for which no easy answer can be provided.

FlatgeoBuf

FlatgeoBuf is a format for storing vector data features with attributes. It comes with the properties of a spatial index in the file header and allows for streaming. It supports typical

[3] https://flatgeobuf.org/

geometries defined by the OGC Simple Features specification[4] and a basic range of attribute types. It is focussed on analysis.

It has broad support in the open source geospatial stack through GDAL and javascript libraries being published.

While not natively supported by ArcGIS Pro, it can consume FlatgeoBuf through its GDAL driver.[5]

FlatgeoBuf is internally uncompressed[6], this means that there is potential for reducing the storage requirements. The potential for using SOZip in combination with FlatgeoBuf and how it compares to http transfer encoding needs to be further analyzed.

FlatgeoBuf does not have any support for overviews builtin and is mostly focussed on publishing data for analysis.

FlatgeoBuf is in the process of becoming a community standard of OGC.[7]

GeoParquet

GeoParquet is a format for storing vector data in a column-oriented way. This allows for a series of optimizations like keeping column statistics in metadata and better compression. It also allows for having multiple geometry columns.

GeoParquet is an extension of the Parquet format which is famous in big data and excels for specific access patterns. It is generally well implemented in a wide range of tools from the data science and data engineering world. GDAL has support for it but requires adding extra drivers in some environments, so it's not easily available everywhere.

---

[4] https://www.ogc.org/standard/sfa/

[5] https://community.esri.com/t5/arcgis-pro-ideas/vector-data-flatgeobuf-support-in-arcgis-pro/idi-p/1155670

[6] https://flatgeobuf.org/#why-not-use-compression-as-part-of-the-format

[7] https://www.ogc.org/requests/public-comment-requested-justification-document-for-flatgeobuf-as-an-ogc-community-standard/

GeoParquet allows for a lot of optimization in internal data organization, making use of this is important as otherwise usage can impacted[8]. By implementing appropriate formatting like row groups it can perform well in cloud native scenarios[9].

This format has a big potential for specific use cases but it's not the first weapon of choice for many scenarios. It is definitely a candidate to keep an eye on and consider if you have big data and an interest in fine tuning data for your use cases. It has to be seen if this evolves into generic recommendation and best practices for general purpose data preparation.

GeoParquet is in the incubation process for an OGC standard[10]

PMTiles

PMTiles is a tile format optimized for visualization. It can hold vector and raster data, its main strength lies in the vector data domain though.[11]

In contrast to the other vector formats in this document, PMTiles supports overviews and tiles. It is by itself a generic container for tiled data and often used as a container for MVT (Mapbox Vector Tiles)[12]. It is an efficient way to publish vector data that has been optimized for visualization. By clipping and generalizing geometries and reducing the number of attributes on different scales it can greatly reduce the size and improve fetching data for visualization purposes but renders it inappropriate for analysis.

This is a very interesting format, it is currently only lacking an endorsement by a standards organization.

Cloud Native Shapefile

Shapefile is one of the longest living formats in the geospatial industry. It was one of – if not the – first vector format on which a proof of concept implementation for cloud optimized vector formats has been performed. One of the advantages this has is that it offers a full backwards compatibility with a broad range of applications. There are however a number of

---

[8] https://www.postholer.com/articles/Parquet-Is-Not-A-Cloud-Native-Format

[9] https://medium.com/radiant-earth-insights/the-admin-partitioned-geoparquet-distribution-59f0ca1c6d96

[10] https://www.ogc.org/press-release/ogc-to-form-new-geoparquet-standards-working-group-public-comment-sought-on-draft-charter/

[11] https://mapscaping.com/podcast/planet-scale-tiled-maps-without-a-server/

[12] https://guide.cloudnativegeo.org/pmtiles/intro.html

reasons that flag the Shapefile format as legacy[13] and the cloud optimized Shapefile has not seen any market adoption. It is listed here mostly for its historic importance.

Multi Dimensional Data

GeoZarr

GeoZarr is a file format for multi dimensional data, it is similar to raster data but with extra dimensions like time or hyperspectral.[14] It builds on the Zarr standard which is optimized for storing N-dimensional arrays in object stores and efficient I/O[15]. This data is often produced by either remote sensing or models, and its primary use cases are in earth observation or meteorology.

In preparation for a future adoption as a standard, the OGC has adopted Zarr V2.0 as a community standard. This standard has no spatial properties in itself and should not be confused with the GeoZarr specification.[16] The GeoZarr specification is currently being discussed in an OGC working group[17]. The standard is currently evolving and being fine tuned and implemented in various tools[18].

Point Cloud Data

COPC

Point Cloud data is produced by airborne LIDAR scanners and is used to produce data sets like swissSURFACE3D[19] COPC is an emerging specification for storing and serving point cloud data in a cloud native way, which is based on the LAZ 1.4 specification. It has been implemented in desktop and mobile applications as well as web viewers. This allows for using one published dataset for distribution of data for analysis as well as using it as a backend for visualization purposes.

---

[13] http://switchfromshapefile.org/

[14] https://radiant.earth/blog/2023/06/exploring-the-potential-of-geozarr-for-storage-and-analysis/

[15] https://zarr.dev/

[16] https://portal.ogc.org/files/100727

[17] https://www.ogc.org/requests/ogc-to-form-geozarr-standards-working-group-public-comment-sought-on-draft-charter/

[18] https://radiant.earth/blog/2023/06/exploring-the-potential-of-geozarr-for-storage-and-analysis/

[19] https://www.swisstopo.admin.ch/en/height-model-swisssurface3d

Market adoption in the open source world was relatively quick, it still lacks integration in well-known tools like ESRI ArcGIS Pro and Cesium, which means that a publication in other formats alongside should currently be evaluated.

Other

3D Data

3D Tiles and I3S

3D Tiles and I3S are designed for accessing 3D data like buildings, photogrammetry or instanced features.[20] This data is optimized for visualization and prepared for modern hardware accelerated rendering. It makes use of tiling and overviews, where overviews are provided through different hierarchical level of detail representations of objects.[21]

- 3D Tiles is listed as an OGC standard within the standards section[22] and also adoption as a community standard is communicated[23]
- I3S is an OGC community standard

An in depth comparison of the two formats could not be found and would need more investigation. There is a conversion tool available[24] and some experiments on comparison have been done[25].

The formats are to our knowledge not cloud optimized in a way that would completely leverage streaming, however they can be hosted as static files and through this be hosted serverless. Ideas for an extension towards a fully cloud optimized format are being considered[26].

For a decision for which of the two formats to use, tooling for creating and maintaining as well as the desired application for accessing and analyzing data should be considered. It is also important to keep track of the direction the industry and community will pick in the future.

---

[20] https://www.ogc.org/standard/3dtiles/

[21] https://cesium.com/blog/2015/08/10/introducing-3d-tiles/

[22] https://www.ogc.org/standard/3dtiles/

[23] https://www.ogc.org/press-release/ogc-adopts-3d-tiles-v1-1-as-community-standard/

[24] https://www.youtube.com/watch?v=0C2fvQXqODQ

[25] https://docs.ogc.org/per/17-046.html

[26] https://github.com/CesiumGS/3d-tiles/issues/399

SOZip

SOZip is a specification for a seek optimized zip file. This format helps to store compressed data in a way that it can still be streamed. It is a profile of the ZIP format and with this property completely backwards compatible which means that any reader that does not support SOZip will still be able to download the ZIP and extract it[27]. This can be used to reduce the storage requirements for uncompressed data like FlatgeoBuf. It is implemented in GDAL which means that it can be used with a lot of the existing tooling. At the time of writing, there is no known javascript implementation, which is a limitation for certain applications. It needs to be considered how it affects transfer speed as native http transfer compression has similar properties.

This format is interesting, and it has to be seen in which applications the global community will find its best use cases.

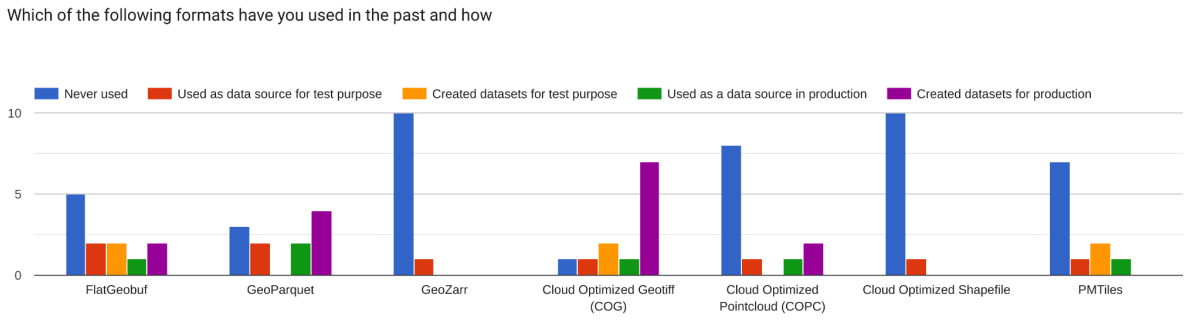
Survey

A survey has been conducted to get community feedback. It has been published on the OPENGIS.ch account on linkedin, mastodon, X and facebook and has also been published on geowebforum.ch.

12 responses have been recorded. The most valuable feedback from this survey was to get an idea of additional formats that need to be considered as well as assembling a catalog of existing data.

Some numerical analysis can be performed as seen in the chart below.

As this was not a representative sample, this can not be used to assess market adoption or distribution in general and is influenced by a self-selection of participants in that survey and the domains in which OPENGIS.ch and geowebforum.ch operate.



[27] https://github.com/sozip/sozip-spec

Catalogs

As cloud native formats are often released in multiple versions over time or for multiple topics, it is an obvious question how they can be exposed with a web friendly approach. This means that they should be discoverable for search engines and other client software. The catalogs are tightly coupled to cloud optimized formats but different in design, as they come in readable formats (JSON) and are not necessarily static like the cloud optimized formats, as new items will often be added continuously.

STAC

The spatio-temporal asset catalog (STAC) has been developed for making large imagery datasets available. As the first available catalog, it has been adopted by the industry as a means to provide metadata and make collections discoverable. Through this, the limitations of the STAC specification have been tested which has led to a better definition of the scope of this specification, specifically also with respect to its limitations on dataset collection level and its advantages in dataset granule handling[28]. STAC defines a core specification[29] which can be extended to specific domains, which has been done for example for SAR[30] or point cloud[31] among many others[32].

The STAC specifications consist of 4 parts, 3 of which can be provided as flat files, whereas the STAC API provides a searchable endpoint which cannot be provided serverless.

STAC is proposed as an OGC community standard[33]

OGC API Records

The OGC API Records specification is part of OGC's strive to standardize services in the geospatial in a modern and web friendly way, providing services in a RESTful way. The OGC API Records specification defines different deployment patterns, one of which is the

---

[28] https://github.com/radiantearth/stac-spec/blob/v1.0.0/best-practices.md#representing-vector-layers-in-stac

[29] https://stacspec.org/en

[30] https://github.com/stac-extensions/sar

[31] https://github.com/stac-extensions/pointcloud

[32] https://github.com/orgs/stac-extensions

[33] https://www.ogc.org/requests/ogc-seeks-public-comment-on-adoption-of-stac-and-stac-api-as-community-standards/

“Crawlable Catalog” which is designed in way that it can be hosted serverless[34]. It is fully optimized for metadata handling of dataset catalogs.

The OGC API Records specification is currently in draft status.[35]

Throughout the last years, the community behind OGC API Records and STAC have undertaken steps to align the two specifications[3637].

[34] https://github.com/opengeospatial/ogcapi-records

[35] https://ogcapi.ogc.org/records/

[36] https://www.ogc.org/blog-article/bringing-stac-into-ogc/

[37] https://github.com/stac-utils/stac-crosswalks/tree/master/ogcapi-records#crosswalk-between-stac-and-ogc-api---records

	FlatGeobuf	GeoParquet	GeoZarr	COG Cloud Optimized GeoTiff	COPC Cloud Optimized Point Cloud	Cloud Optimized Shapefile	PM Tiles
Primary purpose	Vector Data	Vector Data	Multi-Dimensional Grid	Raster	Point Cloud	Vector Data	Vector Data / Raster Data
Status	3.0.1	1.0.0	0.4	-	1.0	POC	3.0
Web reference	https://flatgeobuf.org/	https://github.com/opengeospatial/geoparquet	https://github.com/zarr-developers/geozarr-spec	https://github.com/cogeotiff/cog-spec/blob/master/spec.md	https://copc.io/	https://blog.cleverelephant.ca/2022/04/coshp.html	https://docs.protomaps.com/pmtiles/
Software support	QGIS OGR Geopandas OpenLayers	OGR	GDAL Xarray	GDAL OpenLayers	QGIS PDAL QField		Tippecanoe GDAL QGIS
Specification	https://github.com/flatgeobuf/flatgeobuf?tab=readme-ov-file#specification	https://github.com/opengeospatial/geoparquet	https://github.com/zarr-developers/geozarr-spec	https://github.com/cogeotiff/cog-spec/blob/master/spec.md	https://copc.io/copc-specification-1.0.pdf	-	https://github.com/protomaps/PMTiles/blob/main/spec/v3/spec.md