



Cloud Optimized Point Cloud (COPC) allows reduced resolution access to data.

# Cloud Native Geospatial Lidar with the Cloud Optimized Point Cloud

Lidar consultancy offers efficacious access to big lidar datasets in the Cloud

**M**ost point-cloud processing tasks do not require all the data, but commonly used lidar formats require programs to read it all—whether over a network or directly from disk. In the case of compressed formats such as LAZ, reading it all means extra effort to decompress everything too. An ideal format is widely supported, is openly specified, and eliminates the need to read and decompress all the data for applications that desire only a spatial or reduced resolution subset.

The lidar domain to date has lacked a widely supported and openly specified data format with these features.

Compression and geospatial metadata are well supported by the venerable LAZ format from Martin Isenburg, which builds on ASPRS LAS and has been available in the industry since 2012, but LAZ on its own has not supported allowing readers to perform spatial partitioning. The newly released Cloud Optimized Point Cloud (COPC) draft specification from Hobu, Inc. augments LAZ to provide these features in an opt-in way.

Cloud Native Geospatial, or CNG, is a term used to describe data organization and formats that support data extraction and processing from massive at-rest data archives in cloud object storage.

BY HOWARD BUTLER



COPC is designed to assist lidar software developers in accessing large lidar data sets in the cloud.



Hobu, Inc., the creator of COPC, is an open-source software consultancy located in Iowa City, Iowa.

CNG requires mixing data format compression, indexing, and organization according to the constraints that cloud object storage imposes:

- Accessing information from the cloud is the same as accessing it from a disk, except many times slower
- Cloud formats work around this limitation by putting the pieces of data that applications want close to each other in the file, dramatically reducing the number of accesses required
- For spatial formats, the pieces of data applications want are usually close together in space, so “cloud formats” push together data in spatial clusters—tiles for imagery or small volumes for point clouds

A canonical example of a data format with these features in the geospatial raster data domain is the Cloud Optimized GeoTIFF (COG). COGs allow three things at once. First, the raster data is stored in the widely implemented and standard TIFF format. Second, geospatial metadata is provided using the standard GeoTIFF OGC specification. Third, the “cloud optimized” part organizes the data

to allow software to incrementally access data with as little processing and access as possible when partitioning, filtering, or sampling across the data. These features allow raster users to opt-in to COG’s capabilities as their software implements it rather than requiring an abrupt retooling or software development.

Two key features of LAZ enable this opt-in ability in the point-cloud domain with COPC. First, the LAZ format supports partial decompression by storing data in a series of data chunks. The second feature is the Variable Length Record (VLR) concept of LAS/LAZ, which can store application-specific support data of any kind. By combining LAZ chunking with VLRs that describe the octree structure, COPC allows data to be written in a LAZ file structured as a clustered octree. When data are then accessed according to the tree, software clients can fetch and decompress only what they need at the moment they need it.

The design approach means that clients that do not read the VLRs describing the COPC structure can still read all the LAZ content without impact. This crucial feature enables software to export COPC data and allows LAZ-reading software to consume it without providing

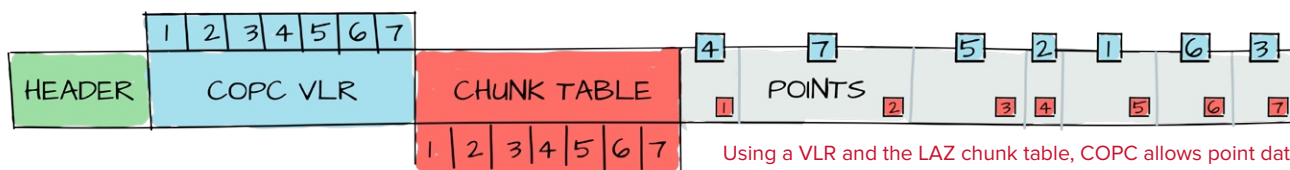
special implementations of COPC software. As with COG, a COPC file is, in essence, “just an LAZ file”.

Kevin Murphy, developer of Applied Imagery’s Quick Terrain Modeler, responded when asked about COPC as a format in the geospatial lidar ecosystem:

*COPC offers explicit support at near-zero cost (in terms of storage or backwards compatibility) for many of the most challenging issues for indexing and management of large lidar data archives. With COPC it is trivial not just to seek<sup>1</sup> through file headers to find files of interest but to do the same within large files. If anything it reduces the need to pre-cut your data into digestible tiles, as you can quickly and easily do overviews and pull tiles or chips on-the-fly. And the best part is, none of your exploitation software needs to change. You can take advantage of the advanced organizational structure if you want, or just access it like any old LAZ and go to town.*

Software that reads LAZ already supports reading COPC point-cloud content, even if it cannot consume its organization. A data provider can deliver

*continued on page 40*



Using a VLR and the LAZ chunk table, COPC allows point data to be stored as octree nodes that clients can directly seek to and decompress as needed.

<sup>1</sup> The word “seek” in computing parlance has the same meaning as skip, but it also means to skip in such a way so that the moving forward doesn’t incur a cost—in our case it is direct i/o or reading of the file AND decompressing the bytes along the way.



Ager guides the reader gently, explaining concepts in words, introducing technical terms only when necessary, and illuminates the path with a stream of simple, well designed graphics, presumably developed for his own teaching, together with excerpts from SAR images and derived products. The material is not oversimplified, however, and deriving maximum benefit from his efforts requires concentration, re-reading of sections and referring back to concepts, symbols and acronyms used earlier. Indeed, your reviewer plans to read the book, cover-to-cover, a second time! Ager's goal of rendering the technology comprehensible is unarguably met, yet along the way some formidable

equations, for example for pulse repetition frequency and the signal-to-noise ratio, are derived in easy steps. Moreover, towards the end readers find themselves believing that they could say something intelligent during the design of an airborne or satellite SAR sensor.

The concepts are illustrated by examples from a wide variety of airborne and spaceborne SAR sources. A table of these, as an appendix, would be convenient and would, in the spirit of the book, be limited in scope, giving just launch dates, principal specifications and products, and country of origin. The danger, obviously, is that such information inevitably would quickly become outdated.

*The Essentials of SAR* is self-published and available on Amazon. It is pleasing to the eye and your reviewer found few errors or typos, although the author has indicated some minor improvements he would make if there were a second edition or reprint. Readers interested in SAR casually, professionally or scientifically, as well as consumers of SAR products who want to understand their purchases better, ought to buy this book. Ager has succeeded magnificently in his mission to explain SAR in terms that almost everyone can understand. He has given thought to a sequel—*LIDAR Magazine* would welcome it. ■

---

Stewart Walker, Managing Editor, *LIDAR Magazine*.

#### **Butler**, continued from page 36

COPC-organized content to clients, and those with updated software can leverage its advanced capabilities, while those without can continue to use the format as before. The incremental opt-in approach of COPC will enable software systems to catch up at their own pace while allowing the early adopters to take advantage.

A major feature of COPC that early adopters can leverage is the ability to “range read” the data they require when they need it over HTTP. Range read capability is a key feature of COG and it is necessary for incremental access over cloud-storage solutions such as Amazon S3 or Azure Blob Storage. Incremental access means that a browser-based visualization client or an adaptive-processing technique can pan through the data over the internet,

over the local file system, or over cloud object storage and control the data access resolution and extent efficiently.

Proprietary point-cloud formats exist that provide some of COPC's features, especially those combining compressed storage with data organized as clustered octrees. None of them is openly specified, and none provides open-source software APIs to consume and produce them, however. Importantly, none of them is a valid delivery format that meets the USGS Lidar Base Specification.

The COPC specification is open source and available at [copc.io](http://copc.io). Open-source tooling such as Potree, PDAL, and Untwine are being updated to support reading and writing COPC data. Thus we hope that software vendors

and data providers will see it as a useful enhancement to apply to LAZ data being delivered for tackling the large-scale data management and processing challenges that lidar data provides. ■



**Howard Butler** is the founder and president of Hobu, Inc., an open-source software consultancy located in Iowa City, Iowa. Hobu focuses on point-cloud data management solutions. He is an

active participant in the ASPRS LAS Committee, a Project Steering Committee member of both the PROJ and GDAL projects, a contributing author to the GeoJSON specification, and a past member of the OSGeo Board of Directors. With his firm, Howard leads the development of the PDAL and Entwine open-source point-cloud processing and organization software libraries.