

## A HIGHLY SCALABLE DATA MANAGEMENT SYSTEM FOR POINT CLOUD AND FULL WAVEFORM LiDAR DATA

A. V. Vo <sup>1,\*</sup>, D. F. Laefer <sup>2</sup>, M. Trifkovic <sup>2</sup>, C. N. L. Hewage <sup>1</sup>, M. Bertolotto <sup>1</sup>,  
N. A. Le-Khac <sup>1</sup>, U. Oferdinger <sup>3</sup>

<sup>1</sup> School of Computer Science, University College Dublin, Dublin 4, Ireland – anhvu.vo@ucd.ie,  
chamin.lokugamhewage@ucdconnect.ie, (michela.bertolotto, an.lekhac)@ucd.ie

<sup>2</sup> Center for Urban Science and Progress, New York University, Brooklyn, New York, USA – (debra.laefer, mt3786)@nyu.edu

<sup>3</sup> School of Natural and Built Environment, Queen's University Belfast, Belfast, Northern Ireland, UK – u.oferdinger@qub.ac.uk

**KEY WORDS:** LiDAR, Point Cloud, Full Waveform, Spatial Database, Big Data, Web Service, Web-based Visualisation

### ABSTRACT:

The massive amounts of spatio-temporal information often present in LiDAR data sets make their storage, processing, and visualisation computationally demanding. There is an increasing need for systems and tools that support all the spatial and temporal components and the three-dimensional nature of these datasets for effortless retrieval and visualisation. In response to these needs, this paper presents a scalable, distributed database system that is designed explicitly for retrieving and viewing large LiDAR datasets on the web. The ultimate goal of the system is to provide rapid and convenient access to a large repository of LiDAR data hosted in a distributed computing platform. The system is composed of multiple, share-nothing nodes operating in parallel. Namely, each node is autonomous and has a dedicated set of processors and memory. The nodes communicate with each other via an interconnected network. The data management system presented in this paper is implemented based on Apache HBase, a distributed key-value datastore within the Hadoop eco-system. HBase is extended with new data encoding and indexing mechanisms to accommodate both the point cloud and the full waveform components of LiDAR data. The data can be consumed by any desktop or web application that communicates with the data repository using the HTTP protocol. The communication is enabled by a web servlet. In addition to the command line tool used for administration tasks, two web applications are presented to illustrate the types of user-facing applications that can be coupled with the data system.

### 1. BACKGROUND

Laser scanning or Light Detection And Ranging (LiDAR) is one of the latest technologies for airborne topographic mapping. The most common type of airborne LiDAR uses a pulsed laser mounted on an airplane to measure the range to target objects on the ground at a high frequency (i.e. up to exceeding 1 MHz). The range measurement is based on the time of flight of the laser pulse reaching the target objects and returning. The ranging sensor works in conjunction with a scanning mechanism that sweeps the laser beam across a space to cover the target objects. In addition, a position and orientation system (i.e. POS) is required on an airborne LiDAR platform to capture the platform's motion. Post-flight, these data components from the ranging sensor, the scanning mechanism, and the POS are integrated to derive a three-dimensional (3D) representation of the scanned scene. Data derived from laser scanning is typically in the format of a point cloud, which is a spatially coherent group of discrete sampling points taken from the surfaces being mapped. Each sampling point consists of a tuple of the point's coordinates (e.g. x, y, z), a timestamp recording the acquisition time, and potentially several other scalar and vector attributes (e.g. signal intensity, colours, normal vector). Some LiDAR sensors allow recording and exporting full waveform (FWF) data, which contain more complete signal backscatter emitted from and received by the sensors (i.e. waveforms) (Mallet and Bretar, 2009). The waveforms are often digitised and recorded as individual timeseries of signal amplitude. The waveforms, together with auxiliary datasets (e.g. the sensor's position and orientation) offer valuable insight into the data's origin. Retention of full waveform

data is becoming a more common practice in LiDAR data acquisition.

Due to their high potential value and wide range of applications (US Geological Survey, 2020), airborne LiDAR data are being acquired at massive scales. Many countries have fully or partially completed their national LiDAR maps. They include Canada, England, Denmark, Estonia, Finland, Japan, the Netherlands, Slovenia, and Switzerland. In the United States, the US Geological Survey is leading the 3D Elevation Program (3DEP), a decade-long national project that aims to complete the acquisition of nationwide LiDAR mapping by 2023. Those large-scale LiDAR acquisition projects generate huge amounts of LiDAR data. For example, as of 2019, over 12 trillion LiDAR data points were made available to the public by 3DEP. In the Netherlands, the nationwide LiDAR acquisition is repeated every 6 years with increasing point densities. The first national LiDAR scan of the Netherlands (Actueel Hoogtebestand Nederland 1 - AHN1) completed in 2003 with most of the country mapped at a density under 1 points/m<sup>2</sup>. The second scan, AHN2, completed in 2012 and resulted in 640 billion data points at a density of 6-10 points/m<sup>2</sup>. AHN3 was launched in 2014 and due to complete by 2019 (Riveiro and Lindenbergh, 2020). AHN4 is set to start in 2020 and to complete by 2023 (<https://www.ahn.nl/ahn-4>). Those few selected examples of large-scale LiDAR projects illustrate the massive amounts of LiDAR data being collected worldwide.

Full waveform LiDAR data are more demanding in terms of storage and processing. FWF LiDAR has mostly been collected at regional scales. In the US, FWF data are collected by the National Ecological Observatory Network (NEON) and the

---

\* Corresponding author

National Center for Airborne Laser Mapping (NCALM). NEON publishes a large amount of its FWF LiDAR data through its data portal (<https://data.neonscience.org/home>). NCALM data are released to the public via OpenTopography, an open-access portal for topographic data in the US. Many datasets on OpenTopography were acquired by sensors with FWF capabilities (Krishnan et al., 2011). However, OpenTopography does not currently support raw waveform data access. Outside the US, the amount of FWF LiDAR data available in the public domain is limited even though FWF data acquisitions at regional scales are frequently reported in academic publications (e.g. Fieber et al., 2015; Mandlbürger et al., 2019). As per the authors' knowledge, there is not a FWF data repository in Europe that is comparable to the NEON data portal. The FWF LiDAR dataset collected over 2 km<sup>2</sup> in Dublin city in Ireland by Laefer et al. (2017) is the only large-scale FWF LiDAR dataset publicly available outside the US known to the authors.

## 2. DATA ACCESS CHALLENGES AND EXISTING SOLUTIONS

Airborne LiDAR data are a critical spatial resource that benefits many stakeholders as well as the general public. The huge amounts of LiDAR data being made publicly available enable a wide range of applications and create great opportunities for data exploration. However, as LiDAR data are massive, challenges involving data storage and management can go beyond the capacities of most data users. Local data storage and management are not efficient if not impossible. Hence, there is an obvious demand for a data management solution that equips users and computer applications with rapid and convenient access to the massive sources of data hosted at remote repositories managed by data publishers. Such approach allows decoupling the data usage on the users' end and the data storage and maintenance, which are handled by expert administrators. The development of applications that consume the data is separated from the data management tasks such as data encoding, compression, and indexing. The approach is referred to as a web service or a service-based approach (Butler et al., 2019). For a sheer volume perspective, employing a web service approach is critical in designing an efficient point cloud data access and dissemination system as recognised by the Open Geospatial Consortium (OGC) in its Testbed 14 Engineering Report (Butler et al., 2019).

There are currently at least five different implementations of a web service approach. They include ESRI's I3S<sup>(2)</sup>, Cesium's 3D Tiles<sup>(3)</sup>, Greyhound<sup>(4)</sup>, Potree<sup>(5)</sup>, and Entwine<sup>(6)</sup>. The first two implementations are OGC community standards. All of the implementations are geared towards serving data for web-based visualisation. Notably, none of the existing implementations employs a database management system (DBMS). Readers interested in a comprehensive comparison of the implementations can refer to the OGC Engineering Report (Butler et al., 2019). In addition to progressively streaming data for visualisation, region query [i.e. selection of data points within an area of interest (AOI)] is another frequently required data access pattern for LiDAR data. That type of query, also known as an AOI selection, is typically required in data dissemination systems. As outlined by van Oosterom et al. (2015), region query is a primary query required in point cloud databases. The query type is supported by major point cloud DBMSs, including Oracle, PostGIS, and MonetDB. However, only a few data dissemination systems are capable of supporting true AOI selection. One

example is the OpenTopography platform at the San Diego Supercomputer Center (Krishnan et al., 2011). In contrast, the majority of LiDAR data dissemination systems only provide a set of static files, each of which usually corresponds to a geographic tile. Examples of such systems include the Discover GIS Data New York by the New York State Office of Information Technology Services (2020) and the DEFRA Data Services platform in the United Kingdom (Department for Environment Food and Rural Affairs, 2020), just to name a few.

## 3. THE ARIADNE3D APPROACH

As an effort towards constructing a scalable spatio-temporal database to provide web service access to LiDAR point cloud and full waveform data, this paper presents the key LiDAR database components in a data system called Ariadne3D. Ariadne3D is a distributed, spatio-temporal data storage system which uses 3D laser scanning point clouds as the skeleton for remote sensing and other data integration. The system is built around the following criteria: integration of heterogeneous urban data, scalability, and user-friendliness, as well as full support for 3D data with an optional temporal dimension control. The architecture of Ariadne3D is presented in Figure 1. The key component of the system is the distributed database on the server side, which hosts LiDAR data and necessary metadata in HBase and HDFS (Hadoop Distributed File System), which are components within the Hadoop ecosystem for big data. Access to the database is performed in two ways. Web and desktop applications that stream data for visualisation or retrieve some fractions of the data can fetch the data using the HTTP protocol via a servlet. Applications that perform data administration tasks (e.g. data ingestion, indexing) or data analytics [e.g. simulations], which require high throughput batch processing, can access the data via the APIs native in HBase and HDFS.

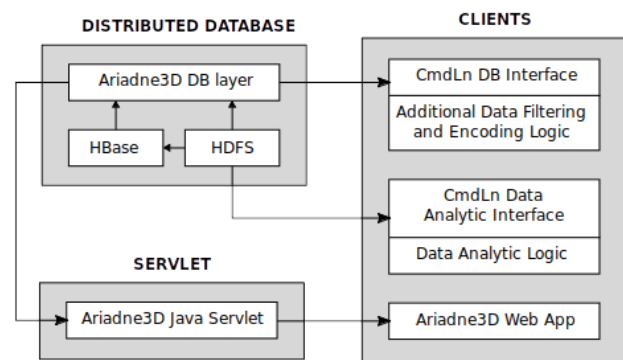


Figure 1. System architecture of Ariadne3D

### 3.1 Server-Side Database

The server side of the system is a distributed database management system built atop HBase and HDFS. HDFS allows storage of data on a cluster of distributed nodes, which can be made of low-cost, commodity hardware. HDFS provides fault-tolerance and high throughput access to a large amount of data. HBase is a non-relational, wide-column, key-value data store built atop HDFS to provide random read and write access to data stored in HDFS. As HBase does not have built-in spatial capabilities, extensions are developed in Ariadne3D to index LiDAR point cloud and FWF data stored in HBase and to enable spatial and temporal queries.

<sup>2</sup> <https://github.com/Esri/i3s-spec>

<sup>3</sup> <https://github.com/CesiumGS/3d-tiles>

<sup>4</sup> <https://github.com/hobu/greyhound>

<sup>5</sup> <https://github.com/potree/potree>

<sup>6</sup> <https://github.com/connormanning/entwine>

Presently, Ariadne3D aims to support three types of query for point clouds and two types of query for full waveform LiDAR data hosted in the database. The first type of query is 3D AOI selection, which can be performed on both the point and the FWF pulse data (i.e. region query). To support this type of query for a point cloud, the point data are indexed using a 3D Hilbert curve at a fixed resolution. More specifically, a Hilbert code is computed for each point, and the points are grouped by their Hilbert codes into voxels, each of which is stored in a tuple of an HBase table. The Ariadne3D database management system exploits the Hilbert codes to rapidly retrieve voxels matching a spatial predicate (i.e. intersecting a given 3D region) and the points contained in the voxels. Using space filling curves such as the Hilbert curve is a common technique for indexing spatial data in key-value data stores (Whitby et al. 2017). More details about the design considerations for point cloud data storage and indexing in Ariadne3D are available at Vo et al. (2018a). To support 3D AOI selection for FWF pulses, which are in the form of line segments, a 6-dimensional (6D) Hilbert curve is currently used. The approach is described in detail in the authors' previous work (Vo et al., 2018b). In brief, the approach models each laser pulse (i.e. a line segment) as a point in a 6D space defined by the x, y, z coordinates of the two end-points of the line segments. The 6D points are mapped to a 6D Hilbert curve, and the Hilbert codes are used as the indexing key in the HBase table. The index allows selecting laser pulses intersecting a given axis-aligned bounding box in 3D.

The second type of query includes lookup and range search by timestamp, which are applied for both the point and the FWF data. A query of this type returns the LiDAR data, including point cloud and FWF data, acquired at a given timestamp or within a specific temporal range. Additional filters can be applied to indicate the specific attributes the database returns (e.g. point coordinates, other point attributes, FWF pulses, waveforms). This type of query is useful for integrating point and FWF data of the same flight line and for analyses that involve the temporal components of the data. To support the query type, the timestamps of the point and FWF data are used as the indexing key in the key-value datastore. The timestamp can optionally be prefixed by the flight line ID, when temporal retrieval by flight line is desirable. Points, pulses, and waveforms of the same flight mission are stored in a single table in HBase.

The last type of query supported in Ariadne3D allows interactive 3D point cloud visualisation. The query returns selected portions of the point data that intersects a 3D viewing frustum. A level of detail (LoD) hierarchical structure is employed to return subsamples the point data based on their distance to the viewing camera in order to avoid overloading the renderer. Ariadne3D follows the OGC community standard 3D Tiles format<sup>(7)</sup> to implement this type of query. According to 3D Tiles, each node in the LoD hierarchy is called a tile, which is represented by a pair of JSON and binary PNTS files. The PNTS file stores the actual point data corresponding to the node while the JSON file contains pointers to the children of the node, a link to the PNTS file, as well as some metadata. Ariadne3D uses an in-house algorithm that maximises the use of the Hadoop MapReduce framework to generate a 3D Tiles structure for a point cloud dataset at a high speed. The generated tiles are stored in an HBase table, in which each node is stored as a tuple and indexed by a compound Hilbert key.

One caveat in the current system is that the non-relational key-value model provided by HBase only allows one primary index

per table. As such, multiple tables of the same dataset are needed to support the different types of queries. While the approach may be good in terms of querying performance, particularly in the context of multi-user access, the multiple copies of the same dataset require large storage overheads and maintenance costs. Ariadne3D uses a metadata table as a gateway for all queries. The metadata table registers each dataset with all corresponding indexed tables. The actual indexed tables are not exposed to users or client applications. Only the name of the dataset of interest is required in a query. Depending on the type of query, the metadata table directs the received query to the indexed table suitable for resolving the query. The metadata table also stores additional information about each dataset, such as the spatial extent, temporal extent, geographical projection, and indexing parameters.

In addition to the spatial and temporal indices that enable spatial querying capabilities, Ariadne3D employs a specific data encoding solution which aims to maximise the efficiency of the distributed file system in Hadoop while providing sufficient flexibility in terms of data schema. The support for flexible schema allows Ariadne3D to accommodate not only the raw LiDAR data originated from laser scanning, but also derived data in the point cloud or FWF format. An example of such derived data is the point cloud annotated with solar potential presented by Vo et al. (2019a). The primary data encoding solution in Ariadne3D is developed by combining Sequence Files and Google Protocol Buffers. Sequence File is the binary file format built in Hadoop that allows the data to be processed at a high level of parallelism. Google Protocol Buffers is a language-neutral, cross-platform, extensible data serialisation framework; details about the encoding mechanism and its efficiency have been published by the authors (Vo et al., 2019b).

### 3.2 Web Service Interface

In the current version, Ariadne3D contains a Tomcat servlet to provide a web interface to the database. The servlet translates web service requests, which are specified using the HTTP protocol, to one of the database queries described in Section 3.1. Upon receiving querying results from the database, the web service transforms the results to the format indicated in the request. GetPointCloud, GetFullwaveform, GetTile, and GetMetadata are the three types of web service requests currently supported in Ariadne3D.

Element	Description
http://localhost:8080/ariadne3d?	Base URL
service=WLDRS	Service type
version=0.1	Service version
request=GetPointCloud	Request type
dataset=d15	Dataset name
clipper=316000,234000,0, 316050,234050,100	Spatial clipping window (AOI)
temp_range=400061, 400062	Temporal range
srs=EPSG:29903	Requested spatial reference system
format=LAS	Requested format

Table 1. Description of a GetPointCloud request

A GetPointCloud request returns a segment of a point cloud within a spatial region and/or a temporal range. Table 1 presents the elements GetPointCloud request and example parameters. In this example, the spatial clipping window is specified by the six

<sup>7</sup> <https://github.com/CesiumGS/3d-tiles>

coordinates of the query bounding box ( $x_{min}$ ,  $x_{max}$ ,  $y_{min}$ ,  $y_{max}$ ,  $z_{min}$ ,  $z_{max}$ ). For more complex queries, where a non-axis aligned querying window is needed, Ariadne3D can perform the request by submitting the complex clipping window as a GeoJSON file via a POST request. GetFullwaveform requests return full waveform data structured similarly to what shown in Table 1, except that the request type is GetFullWaveform.

The GetTile request illustrated in Table 2 provides LoD data in the 3D Tiles format, which is compatible with the CesiumJS library for 3D web-based visualisation. A GetTile request returns either a binary PNTS file or a JSON file depending on the requested format. Data of a specific node can be requested by specifying the node's key in the request. Otherwise, the root node (key=root) can be passed to a CesiumJS client which can automatically traverse the LoD hierarchy to retrieve subsequent nodes.

In addition, GetPointCloud, GetFullwaveform, and GetTile, Ariadne3D provides GetMetadata request, which allows retrieving high level information about the data stored in the database. Using GetMetadata, users and client applications can list all datasets available in a host, their descriptions as well as their spatial and temporal extents. The list of indexed tables associated with each project is also available through GetMetadata requests.

Element	Description
http://localhost:8080/ariadne3d?	Base URL
service=WLDRS	Service type
version=0.1	Service version
request=GetTile	Request type
dataset=d15	Dataset name
key=root	Key (ID) of a node
format=PNTS	Requested format

Table 2. Description of a GetTile request

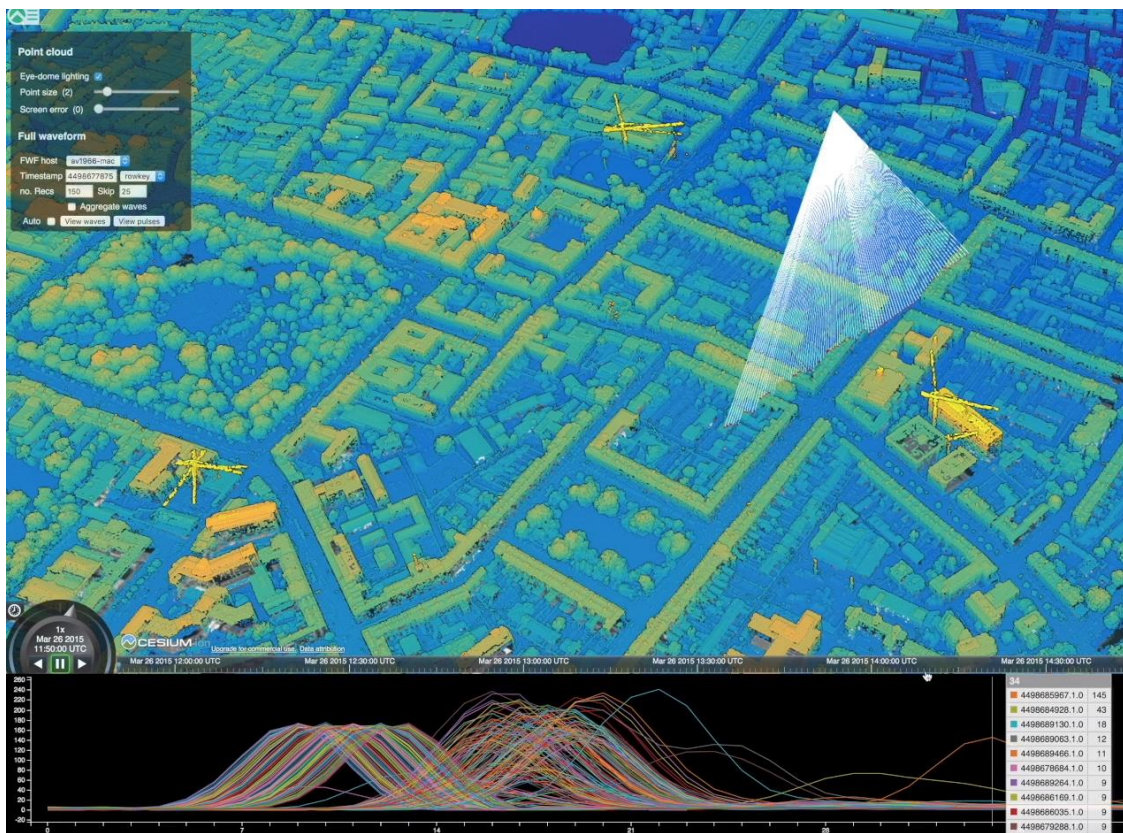


Figure 2. Three-dimensional web application for visualisation

### 3.3 Client Applications

Data administration tasks in Ariadne3D are performed using a combination of the Java API native in HBase and a set of in-house applications developed by the authors using the Hadoop MapReduce and Apache Spark frameworks. For example, computing Hilbert indices and generating 3D Tiles structures are implemented as Spark programs. Generating HFiles for bulk loading the data into HBase is implemented using Hadoop MapReduce.

Ariadne3D currently provides 2 different web applications for data dissemination and visualisation. These web applications are meant for non-expert users to conveniently access and explore the sources of LiDAR data hosted remotely from their web browsers. In other words, the web applications provide user-friendly graphical means to submit data service requests described in Section 3.1 to a database (see Section 3.2). The first web application uses Leaflet<sup>(8)</sup>, an open source Javascript library, to provide the web map graphical interface. The application allows users to visualise the spatial extents of LiDAR datasets available in a host and to download point cloud and/or FWF data

<sup>8</sup> <https://leafletjs.com/>

within an AOI where the data are available. Notably, the AOI is not constrained by the internal data partitioning mechanism in the database. The returned data can be downloaded from the web browser as files. Point cloud data can be downloaded in a text or LAS format while FWF data can be download in LAS, PulseWaves, or Autodesk DXF format (for visualisation). Point cloud data returned from an AOI selection can be automatically uploaded to Sketchfab, a platform to view and publish 3D contents on the web with virtual reality capabilities.

The second web application in Ariadne3D is a 3D web application shown in Figure 2, which allows overlaying point cloud and FWF data on a virtual globe for viewing and exploration. The 3D web application is based on CesiumJS<sup>9</sup>. A screenshot of the 3D web application is shown in Figure 2. The main window presents the point cloud data collected in Dublin by Laefer et al. (2017) rendered in a CesiumJS virtual globe environment. On top of the point cloud, there is a set of laser pulses spread down from the helicopter collecting the data. The laser pulses are part of the full waveform components of the dataset. The point cloud and FWF data in this example are fetched to the browser using two separate requests. The first one is a GetTile request, which is handled automatically by CesiumJS to render the 3D point cloud. The second request is GetFullWaveform, which returns the laser pulses and waveforms given a temporal range. The temporal range is specified using the input boxes in the control panel on the left of the window. The waveforms returned from the request are rendered in the D3.js chart at the lower segment of the window. The point and FWF data in all of the viewing windows are simultaneously synchronised to offer an integrated view of the data, which has not been seen elsewhere.

#### 4. CONCLUDING REMARKS

Compared to existing systems, the Ariadne3D data system is novel in multiple ways. First, the system is highly scalable and fast due to the use of the distributed database and the scalable computing frameworks (i.e. MapReduce and Apache Spark). Furthermore, the system can cope with high data volumes by elastic scaling. Namely, additional nodes can be added to the system to accommodate an increase in workload on demand. Such elastic scaling is ever more feasible with the increasing availability of cloud computing resources. The capability to accommodate and integrate both point clouds and full waveform data is the second feature signifying the novelty of the presented system. As per the authors' knowledge, there are no similar systems that can provide efficient and convenient access to integrated point cloud and full waveform data in the way the Ariadne3D system does. The third characteristic that distinguishes the presented data system from others is its user-friendliness. End users are provided with simple and intuitive graphical tools to view and access the data. In contrast, the data partitioning scheme, data indexing, encoding, and all other low-level implementation details are hidden from end-users. Additionally, users can access both 2D and 3D viewers directly from a generic web browser without the need of installing additional software locally. Such a feature is critical when LiDAR datasets are being acquired in mass at national and regional levels and are meant to be utilised by the general public, rather than being contained within a specialised community. Future research will investigate the use of OGC standard protocols (e.g. Web Map Service, Web Feature Service, Web Coverage Service) for LiDAR data services and further optimise the system's scalability, performance, and functionalities.

#### ACKNOWLEDGMENTS

Funding for this project was provided by the National Science Foundation as part of the project "UrbanARK: Assessment, Risk Management, & Knowledge for Coastal Flood Risk Management in Urban Areas" NSF Award 1826134, jointly funded with Science Foundation Ireland (SFI - 17/US/3450) and Northern Ireland Trust (NI - R3078NBE). The Hadoop cluster used for the testing was provided by NYU High Performance Computing Center. The implementation and testing were jointly enabled through computing resources provided under sub-allocation "TG-IRI180015" from the Extreme Science and Engineering Discovery Environment (XSEDE), supported by National Science Foundation grant ACI-1548562 (Townes et al., 2014). The aerial LiDAR data of Dublin were acquired with funding from the European Research Council [ERC-2012- StG-307836] and additional funding from Science Foundation Ireland [12/ERC/I2534].

#### REFERENCES

- Butler, H., Manning, C., Bourgon, J.F., Heazel, C., 2019. OGC Testbed-14: Point cloud data handling engineering report. Public Engineering Report 18-048r1, Open Geospatial Consortium. docs.ogc.org/per/18-048r1.html (1 May 2020).
- Department for Environment Food and Rural Affairs, 2020. DEFRA Data Services Platform. data.gov.uk/dataset/977a4ca4-1759-4f26-baa7-b566bd7ca7bf/lidar-point-cloud (1 May 2020).
- Fieber, K.D., Davenport, I.J., Tanase, M.A., Ferryman, J.M., Gurney, R.J., Becerra, V.M., ... Hacker, J.M., 2015. Validation of Canopy Height Profile Methodology for Small-footprint Full-waveform Airborne LiDAR Data in a Discontinuous Canopy Environment. *ISPRS J Photogramm Remote Sens*, 104, 144–157. doi.org/10.1016/j.isprsjprs.2015.03.001.
- Krishnan, S., Crosby, C., Nandigam, V., Phan, M., Cowart, C., Baru, C., Arrowsmith, R., 2011. OpenTopography: a services oriented architecture for community access to LiDAR topography. *Proceedings of the 2<sup>nd</sup> International Conference on Computing for Geospatial Research & Applications*, 1-8. doi.org/10.1145/1999320.1999327.
- Laefer, D.F., Abuwarda, S., Vo, A.V., Truong-Hong, L. and Gharibi, H., 2017. 2015 Aerial Laser and Photogrammetry Survey of Dublin City Collection Record. New York University Spatial Data Repository, New York, USA. doi.org/10.17609/N8MQ0N.
- Mallet, C. and Bretar, F., 2009. Full-waveform Topographic Lidar: State-of-the-art. *ISPRS J Photogramm Remote Sens*, 64(1), 1-16. doi.org/10.1016/j.isprsjprs.2008.09.007.
- Mandlbürger, G., Lehner, H., Pfeifer, N., 2019. A comparison of single photon and full waveform LiDAR. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-2/W5, 397–404. doi.org/10.5194/isprs-annals-IV-2-W5-397-2019.
- New York State Office of Information Technology Services, 2020. Discover GIS Data NY. orthos.dhSES.ny.gov (1 May 2020).
- Riveiro, B. and Lindenbergh, R., 2020. *Laser Scanning - An Emerging Technology In Structural Engineering*. CRC Press.

<sup>9</sup> <https://cesium.com/cesiumjs/>

Towns, J., Cockerill, T., Dahan, M., Foster, I., Gaither, K., Grimshaw, A., ... Wilkins-Diehr, N., 2014. XSEDE: Accelerating Scientific Discovery. *Computing in Science and Engineering*, 16(5), 62–74. doi.org/10.1109/MCSE.2014.80.

US Geological Survey, 2020. Lidar Applications and Business Uses Factsheets. [www.usgs.gov/core-science-systems/ngp/3dep/lidar-applications-and-business-uses-factsheets](http://www.usgs.gov/core-science-systems/ngp/3dep/lidar-applications-and-business-uses-factsheets) (1 May 2020).

Vo, A.V., Konda, N., Chauhan, N., Aljumaily, H., Laefer, D.F., 2018a. Lessons learned with laser scanning point cloud management in Hadoop HBase. *Workshop of the European Group for Intelligent Computing in Engineering (EG-ICE) 2018*, 10863, 231-253. doi.org/10.1007/978-3-319-91635-4\_13.

Vo, A.V., Chauhan, N., Laefer, D.F., Bertolotto, M., 2018b. A 6-dimensional Hilbert approach to index full waveform LiDAR data in a distributed computing environment. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-4, 671–678. doi.org/10.5194/isprs-archives-XLII-4-671-2018.

Vo, A.V., Laefer, D.F., Smolic, A., Zolanvari, S.M.I., 2019a. Per-point Processing for Detailed Urban Solar Estimation with Aerial Laser Scanning and Distributed Computing. *ISPRS J Photogramm Remote Sens*, 155, 119–135. doi.org/10.1016/j.isprsjprs.2019.06.009.

Vo, A.V., Hewage, C.N.L., Russo, G., Chauhan, N., Laefer, D.F., Bertolotto, M., Le-Khac, N.A., Oftendinger, U., 2019. Efficient LiDAR point cloud data encoding for scalable data management within the Hadoop eco-system. *IEEE International Conference on Big Data*, 2019, 5644-5653. doi.org/10.1109/BigData47090.2019.9006044.

van Oosterom, P., Martinez-Rubi, O., Ivanova, M., Horhammer, M., Geringer, D., Ravada, S., ... Gonçalves, R., 2015. Massive Point Cloud Data Management: Design, Implementation and Execution of a Point Cloud Benchmark. *Computers & Graphics*, 49, 92–125. doi.org/10.1016/j.cag.2015.01.007.

Whitby, M., Fecher, R., Bennight, C., 2017. GeoWave: Utilizing distributed key-value stores for multidimensional data. *International Symposium on Spatial and Temporal Databases 2017*, 10411, 105–122. doi.org/10.1007/978-3-319-64367-0\_6.