Using Databases for 3D Data Management –
From Point Cloud to City Model

Hans Viehmann¹

¹ORACLE Corporation, Server Technologies Division, Hamburg, Germany,
hans.viehmann@oracle.com

Abstract

With the cost of 3D data acquisition constantly falling, many use cases based on the availability of 3D models have become affordable in recent years. This trend gives rise to very large volumes of data being collected which must be managed, processed and analyzed in conjunction with both the associated metadata as well as complementary data such as raster imagery.

Increasingly, geospatial databases are being used for these purposes, which leverage the scalability and information lifecycle management capabilities of standard databases while at the same time allowing the integrated database logic to specifically store and analyze point clouds, TINs, solids, 3D vector data and geo-referenced raster data sets. This enables the derivation of more compact, generalized 3D models from laser scanning data without unnecessary data movement or transformation. The results can then be cast into data models which in turn allow easy data visualization, analysis or dissemination through standards such as CityGML or KML.

In this paper a solution for the end-to-end processing, storage and analysis of point cloud data is presented, which uses an ORACLE database with the additional functionality of the Spatial and Graph option. To illustrate the technical capabilities, two case studies are included - one project is focused on managing asset data in a railway infrastructure company, while the other is a city modeling solution.

1. Introduction

Today, three-dimensional geospatial data are being used across almost all industries in a wide variety of use cases. In many countries the public sector commissions data collection at high resolution covering large areas, sometime entire countries. The main drivers for most of these kinds of projects in Europe are either related to risk management (eg. determining flood plains or areas endangered by land slides) or renewable energy, for instance determining the optimal location for wind turbines or suitable rooftops for the placement of photovoltaic panels. On a smaller scale, 3D data are widely used on a city level for urban planning or disaster management. Other industries such as engineering and construction which have been using 3D CAD data for decades for planning and design purposes, increasingly use up-to-date sensor data for continuous site surveying or asset management.
At the same time a “consumerization” of 3D data is taking place. Companies such as Google are making 3D representations available to the wider public, current personal navigation devices include pseudo-3D views and initiatives around indoor mapping in public buildings, particularly shopping malls or airports, create a level of familiarity with this kind of visualization which can be expected to lead to more demand in other fields much like the advent of Google Maps has generated a wider demand for digital mapping in 2D.

Many of the applications mentioned above have only become economically viable after data acquisition has become faster, more automated and, most importantly, significantly less costly. For example, modern Light Detection and Ranging (LiDAR) devices are capable of capturing large areas by rapidly generating point clouds consisting of billions of points each of which comes with multiple attributes, up to “full-waveform” measurements which include a highly resolved sampling of the entire laser pulse per point.

The immediate challenge that comes with this kind of data acquisition is the need to manage and process vast amounts of raw data and to subsequently provide visualization technologies and further analytic capabilities on the resulting data sets. For these purposes database systems provide significant advantages over conventional file-based approaches, as they allow the joint management of the various types of 2D and 3D data together with the associated metadata and other attribute information. Search and (integrated) analysis are enabled by indexing mechanisms and can be implemented easily using well-known query languages such as SQL in conjunction with pre-packaged functionality for geospatial analysis. In the case of specialized database systems such as an ORACLE database with the Spatial and Graph option, much of the data-intensive computational logic specific to 3D geospatial data can be executed in the database server without additional data movement across a network.

In addition to the spatial capabilities database technologies deliver standard functionality for data management such as compression, indexing, etc., as well as scalability through clustering or multi-processor support, security or data access through standard tools and interfaces. This can help to reduce operational cost, simplify development and speed up the actual analysis.

In the following sections typical data processing workflows will be explained in more detail. Point cloud data generated by LiDAR will be used as the primary example, but the results are largely applicable to seismic data or 3D data sets generated by other sensors.

2. Point Cloud Data Processing Workflow

As a first step in the processing workflow point cloud data, together with the associated metadata, need to be loaded into the database, transformed into the appropriate data structure and indexed for efficient data access. Storing the entire point cloud data set from one survey in a single physical structure might be technically feasible, however, if every operation on the data would
require retrieving the entire object this would incur a massive overhead. Therefore, in order to deal with this kind of data structure the approach that has been taken with the SDO_PC point cloud data type in the ORACLE database is to separate the data set into a single logical point cloud object which contains all common metadata and a collection of physical blocks each of which contains a chunk of the actual measurement points. Within each block the point data are stored in a compact binary form together with their respective attributes and can be further compressed using standard database mechanisms in order to save storage space. For each block its spatial extent is stored for fast access through a spatial index. With this structure it is possible to retrieve a relevant spatial subset of the data for visualization or further processing very efficiently.

Creating these data structures from a variety of input formats can be largely automated, for instance by using tools such as Safe Software’s FME, open source libraries such as libLAS in the case of input files in LAS format, or the Point Data Abstraction Layer (PDAL), a BSD licensed library used to translate and manipulate point cloud data from different input sources.

Once the raw data is transformed into the target data structures, it is available to automated processing workflows in the database. In ORACLE Spatial and Graph there are functions available to extract parts of a data set, run filtering operations or derive secondary products such as triangulated irregular networks (TINs) from the raw point cloud data or parts thereof. Even a function to calculate contour lines from SDO_PC objects is included. Also, the processing capabilities of the database engine can be used to implement specific object recognition algorithms or pattern matching techniques in procedural languages such as PL/SQL or Java. If the resulting geometry is a 3-dimensional solid it can be modelled as such using the SDO_GEOMETRY data type which in turn can be rendered by a variety of commercial and open source GIS tools, CAD systems and visualization components.

2.1 LiDAR data management for asset management in a railway company

This database-centric approach to point cloud data processing has been used in a project in the railway industry in Europe where a combination of airborne and railborne laser scanning in conjunction with aerial photography was primarily used for the purpose of asset maintenance. The resulting data sets are also being made available within the enterprise to support the planning and design processes by providing an exact and up-to-date representation of the as-built status.

At present some 8bn measurement points are kept online, together with GIS data, CAD drawings and geo-referenced raster imagery, leading to a total data volume of about 2TB. 2D vector data are used to extract relevant parts of the survey data and subsequently object recognition algorithms in the database are used to determine the position of the centre of the track, the profile of track and track bed as well as various features in the vicinity of the tracks such as noise protection walls, utility poles and more. Selected portions of the data can be passed to an interactive 3D viewer for
An integral part of the project is metadata management, not the least because of the requirement to maintain the history of all features and surveys. Data from all sources - point clouds, raster imagery and vector data - is made available for efficient search based on the associated metadata, which are collected and managed using the ISO 19115 standard and published through an OGC CSW webservice. For this purpose the XML-related capabilities of the underlying database were used in order to significantly reduce the development effort.

2.2 Creating a city model from LiDAR data

A similar processing workflow can be used to generate building models in 3D from LiDAR surveys. Usually the building footprints can be derived from cadastral maps so that a rough approximation of the shape of the building can be determined by extruding the footprint to an average measured building height. Care must be taken, however, to avoid distortions caused by incorrect measurements due to obstacles in the vicinity of buildings, especially tree canopies. Also, for a meaningful building height it is necessary to determine the ground level first of all.

In a city modelling project with a major city in Europe this methodology was implemented covering about 250000 buildings in a total area of over 160km². LiDAR data were collected with a resolution of >30pts./m² which were complemented by stereogrammetric photograpy and oblique imagery to provide building textures. In a first step the bare earth surface model (digital terrain model, DTM) was derived from the LiDAR data set to provide a ground level. Subsequently, all measurement points attributed to buildings were extracted, classified into roof and wall surfaces to determine the roof shapes and then combined with the building footprint to deliver the wall surfaces. This process was largely automated with some limited manual verification. Significant effort was invested in quality assurance, making sure that no artificial gaps between buildings or other artefacts are generated. For complex structures such as buildings with curved walls or roof shapes the 3D shape was determined manually using stereo restitution. This procedure was also applied to bridges. The resulting 3D features are managed as objects in an ORACLE Spatial and Graph database in a data model which can be easily accessed by various systems or exported in formats such as CityGML or DGN for external use.

2.3 Using CityGML for City Modelling

Beyond a feature based model to store and manage 3D objects in an ORACLE Spatial and Graph database there is also a semantically rich, hierarchically structured higher level data schema readily available under the name of “3DCityDB”. It was originally developed on the basis of the CityGML information model by the Institute of Geodesy and Geoinformation of the University of Bonn in cooperation with lat/lon GmbH on behalf of the Senate of Berlin and Berlin Partner GmbH (http://www.ikg.uni-bonn.de/index.php?id=253) and has been published as open source under the
terms of LGPL3. This data model allows storage and maintenance of 3D features at five different levels of detail together with DTMs and geo-referenced raster imagery and can provide a good starting point for any city modelling project. It has been proven in a large number of implementations such as the 3D model of the City of Berlin, which includes 550000 buildings including textures which were extracted from oblique photography. The complete code, documentation as well as a set of import/export tools is available for download on www.3dcitydb.org.

2.4 Publishing 3D Data using Database Technologies

Besides specific tools for data export there are standard mechanisms within the ORACLE database which can be used to generate either GML or KML to publish the data. Converting SDO_GEOMETRY objects into GML is done by means of a single utility function call which is part of the standard database functionality. For KML the XML capability of the ORACLE database can be used. This allows the synthesis of a KML document from a set of geometries by aggregating SDO_GEOMETRY objects converted into KML together with styling information and attributes into a single XML document. This can in turn be published directly from within the database over the HTTP or WebDAV protocol so that applications such as a Google Earth client has direct access to the result without an intermediary file being generated.

3. Conclusion

Spatially-enabled databases are delivering a number of benefits to the entire point cloud processing workflow as they are well suited to store, manage and analyze the large data sets involved. Significant value results from the combined analysis of point cloud data with related 2D, 3D or attribute information as illustrated in the two examples above. With all that in mind it can be expected that over the coming years an increasing amount of algorithms specific to point cloud and 3D data types will be made available inside databases and that a growing ecosystem of 3rd party tools and solutions will address the broad set of needs of the various user communities.