Triangulating The Netherlands on the fly using a Spatial DBMSs

Romulo Goncalves[1], Tom van Tilburg[2], Kostis Kyzirakos[3], Foteini Alvanaki[3], Panagiotis Koutsourakis[4], Ben van Werkhoven[1], and Willem van Hage[1]

[1]The Netherlands eScience Center, The Netherlands: {r.goncalves,b.vanwerkhoven,w.vanhage}@esciencecenter.nl
[2]Geodan, The Netherlands: tom.van.tilburg@geodan.nl
[3]Centrum Wiskunde & Informatica, The Netherlands: {kostis.kyzirakos,f.alvanaki}@cwi.nl
[4]MonetDB Solutions, The Netherlands: panagiotis.koutsourakis@monetdbsolutions.com

ABSTRACT

3D digital city models, important for urban planning, are currently constructed from massive point clouds obtained through airborne LiDAR (Light Detection and Ranging). They are semantically enriched with information obtained from auxiliary GIS data like Cadastral data which contains information about the boundaries of properties, road networks, rivers, lakes etc.

In this work we demonstrated a column-oriented SDBMS enhanced with a set of optimized operators to provide effective data skipping, efficient spatial operations, and interactive data visualization. Such features are exploited for 3D digital city models using latest topography of The Netherlands and the latest Cadastral information for The Netherlands. Through a web-interface exploiting X3D technology, the user requests 3D digital city models with predicates on the semantic attributes. The demo has been presented at ACM SIGSPATIAL 2016 [4].

Keywords

3D digital city models, Spatial DBMS, and Column-stores

1. INTRODUCTION

Technical advances in the LiDAR data acquisition systems made possible the rapid acquisition of high resolution topographical information for an entire country. Such data sets are now reaching the trillion points barrier. To cope with this data deluge and provide up-to-date 3D digital city models on demand current geospatial management strategies should be re-thought.

With Cadastral data being constantly updated and LiDAR data being extended with periodic scans, 3D digital city models must be reconstructed periodically, i.e., not being generated anymore with a onetime large pre-computation job, and allow user interaction. Such demands made us to re-think on how spatial computations, data management and data processing is performed in large scale.

Hence, our goal is to modernize the generation and manipulation of 3D digital city models by extending a Geospatial Database Management System (DBMS) with all necessary functionality to access directly raw data sets without requiring data to be pre-loaded and take advantage of GPUs to accelerate the 3D digital city models generation thereby vastly improving flexibility and performance.

This work presents a column-oriented Spatial Database Management System which provides in-situ data access, effective data skipping, efficient spatial operations, and interactive data visualization. It provides support for a flexible storage schema that allows 2D/3D geospatial datasets to store semantically rich objects that are needed for the customization (i.e., data re-generation with user defined parameters) of 3D digital city models on a large scale. Its efficiency and scalability is demonstrated using a dense LiDAR scan of The Netherlands consisting of 640 billion points and the latest Cadastral information, and compared with PostGIS.

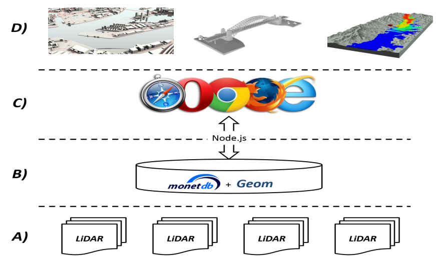


Figure 1: Architecture

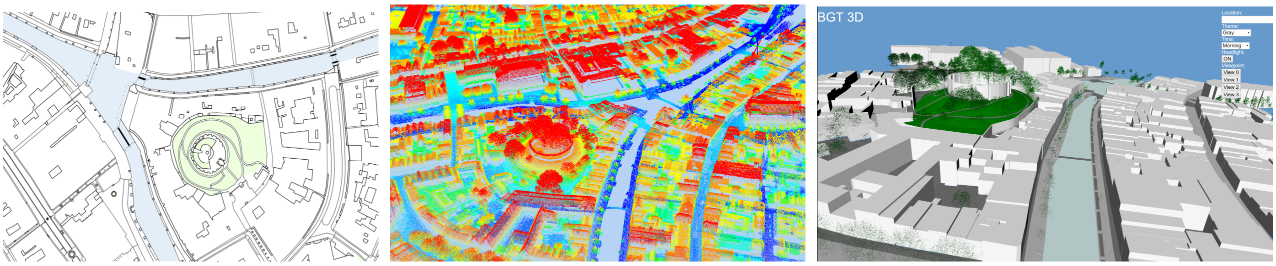
1.0.1 Spatial column-store

For our work we have extended the column-oriented SDBMS MonetDB [5], which steps away from traditional SDBMSs that are all record-oriented architectures. By being a column-oriented SDBMS, our solution offers vertical partitioning of relational tables which significantly reduces data access, improves data compression, simplifies data skipping strategies, suits well vector processing and the integration of GPU-accelerated operators (c.f., Section 1.0.3). In our case, vertical partitioning is further exploited to reduce the number of columns to be imported as we will explain in Section 1.0.2.

Our solution provides support for a flexible storage schema that allows 2D/3D geospatial datasets to store semantically rich objects that are needed for the customization (i.e., data re-generation with user defined parameters) of 3D digital city models on a large scale. It is the backend used in the architecture described in Figure 1. It caters for multiple types of applications; SQL can be used for analytical workloads, but also provides an integrated environment for R and Python. After processing the data our solution has the option to export the results to standard formats to be loaded into visualization tools. Examples are GeoJSON, LAS and X3D, the latter being used by the web-application of our demo.

1.0.2 In-situ data access

The nationwide LiDAR dataset AHN2 comprises 640 billion points averaging 6-10 points per square meter. For this dataset, the authors of [7] spent nearly 18 hours for extracting, transforming, and loading (ETL) the dataset into MonetDB because the data was



(a) Cadastral data (b) LiDAR data (c) Web-browser viewer

Figure 2: Data sets and web-browser viewer

initially converted to the internal format of MonetDB and then imported through a bulk loading operation. The same amount of time was spent for sorting and indexing AHN2 to be able to query it efficiently using the well-known LiDAR file-based solution Rapidlasso LAStools [1].

To reduce such a costly pre-query preparation step we decided to keep data in its original format and access it through a lazy and iterative import process that accesses the raw files directly. This approach is a LiDAR extension of the strategy described in [3] that discusses how to provide *in-situ* data access to large NetCDF data repositories. The general concept was developed in the context of the data-vaults framework [6].

1.0.3 Massive parallelism

The spatial query model that is used by MonetDB follows the well-established two-step approach of **filtering** and **refinement** [2]. The refinement step operates on the results of the filtering step that produced a superset of the solution. During this step, the spatial predicate is evaluated against the precise geometry G. The refinement step can be very expensive, especially when the geometries are complex. To circumvent the issue, we exploit GPU technology to speed up the refinement step.

Our work integrates GPU aware spatial primitives into the column-oriented kernel of MonetDB for efficient geo-spatial querying. Within a single query the user has column-store efficient I/O and massive GPU parallelism for spatial operators. MonetDB allows a seamless integration due to two special features of its architecture, the operator-at-the-time paradigm and late materialization.

2. DEMONSTRATION

We propose a demonstration with a live interactive experience using a X3D viewer to visualize 3D digital city models of The Netherlands. Using the interface illustrated in Figure 2c, the user can select a city, angle of view, resolution, etc.

2.1 Data sets

The data sets for the demo session are the latest topography of The Netherlands, *Actueel Hoogtebestand Nederland 3* (AHN3) and the latest Cadastral information for The Netherlands, Basisregistratie Grootschalige Topografie (BGT). The BGT data is a detailed 2D digital map of The Netherlands. It contains the location and footprint of objects such as buildings, roads, water, railways, and green areas as illustrated in Figure 2a. The BGT data is freely distributed as open data [1].

AHN3 dataset is the latest LiDAR scan of The Netherlands and it is also freely distributed as open data [2]. The sample density is 6 to 10 points per square meter for the entire country. Figure 2b presents a 3D visualization of the AHN2 dataset. During our demo session the latest scan AHN3 is used.

2.2 User interaction

In our demo, a user expresses an area of interest with a predicate on the semantic attributes she wants to visualize. The request is translated into an SQL query and through a MonetDB node.js module it is pushed down to the remote MonetDB server, layer *D), C) and B)* in Figure 1.

After query compilation, the database performs all the steps described in Section 1.0.2 to make the necessary data available, i.e., *in-situ* access to the file repositories in layer *A)* Figure 1. With all relevant data available, the filtering and the refinement steps are executed. The selected points are then grouped per object and each object is triangulated using Constrained Delaunay Triangulation. The query result is exported as X3D format and sent back to the browser for rendering.

In the web browser, the resulting scene shows a 3D extrusion from the Cadastral data where the terrain is modelled along its vertices and buildings are extruded to their average height. The scene can be rotated and lighting, and colors can be changed. The most important feature however is that the individual entities from the Cadastral data (like roads, houses, bridges) can be highlighted or otherwise individually dealt with. This gives way to more advanced interaction with the database in the future, where users can add information to the data and send it back to the database to run more advanced modelling (e.g. flood or noise modelling).

3. ACKNOWLEDGMENTS

The work reported here is partly funded by the NLeSC project, *Big Data Analytics in the Geo-Spatial Domain* (project code: 027.013.703), and the NWO valorization project for COMMIT/, *3D geospatial data exploration for modern risk management systems*.

4. REFERENCES

[1] rapidlasso GmbH. `http://rapidlasso.com/`, 2014.

[2] F. Alvanaki, R. Goncalves, M. Ivanova, and et al. GIS navigation boosted by column stores. *PVLDB*, 2015.

[3] R. Goncalves, M. Ivanova, F. Alvanaki, and et al. A round table for multi-disciplinary research on geospatial and climate data. *IEEE e-Science*, 2015.

[4] R. Goncalves, T. van Tilburg, and et al. A spatial column-store to triangulate the netherlands on the fly. In *ACM SIGSPATIAL 2016*, 2016.

[5] S. Idreos, F. Groffen, N. Nes, S. Manegold, and et al. Monetdb: Two decades of research in column-oriented database architectures. *IEEE Data Engineering Bulletin*, 2012.

[6] M. Ivanova, Y. Kargin, and et al. Data Vaults: A Database Welcome to Scientific File Repositories. SSDBM, 2013.

[7] P. van Oosterom, O. Martinez-Rubi, and et al. Massive point cloud data management: design, implementation and execution of a point cloud benchmark. *Computer Graphics*, 2015.

[1] `https://www.pdok.nl/nl/producten/pdok-downloads/download-basisregistratie-grootschalige-topografie`

[2] `https://www.pdok.nl/nl/ahn3-downloads`