Service-Based 3D Rendering and Interactive 3D Visuali ation

Benjamin Hagedorn
Hasso-Plattner-Institut for
Software Systems Engineering
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam, Germany
benjamin.hagedorn@hpi.uni-potsdam.de

Jürgen Döllner
Hasso-Plattner-Institut for
Software Systems Engineering
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam, Germany
doellner@hpi.uni-potsdam.de

Abstract

This report describes the subject and preliminary results of our work in the context of the HPI Future SOC Lab. This work generally aims on exploiting high performance computing (HPC) capabilities for service-based 3D rendering and service-based, interactive 3D visualization. A major focus is on the application of HPC technologies for the creation, management, analysis, and visualization of and interaction with virtual 3D environments, especially with complex 3D city and landscape models.

1 Introduction

Virtual 3D city models represent a major type of virtual 3D environments. They can be defined as a digital, geo-referenced representation of spatial objects, structures and phenomena of a distinct geographical area; its components are specified by geometrical, topological, graphical and semantic data and in different levels of detail.

Virtual 3D city models are, e.g., composed of digital terrain models, aerial images, building models, vegetation models, and city furniture models. In general, virtual 3D city models serve as information models that can be used for 3D presentation, 3D analysis, and 3D simulation. Today, virtual 3D city models are used, e.g., for urban planning, mobile network planning, noise pollution mapping, disaster management or 3D car and pedestrian navigation.

In general, virtual 3D city models represent prominent media for the communication of complex spatial data and situations, as they seamlessly integrate heterogeneous spatial information in a common reference frame and also serve as an innovative, effective user interface. Based on this, virtual 3D city models, as integration platforms for spatial information, represent building blocks of today's and future information infrastructures.

1.1 Complexity of 3D city models

Virtual 3D city models are inherently complex in multiple dimensions, e.g., semantics, geometry, appearance, and storage. Three major complexities are described in the following.

Massive amounts of data: Virtual 3D city models typically include massive amounts of image data (e.g., aerial images and façade images) as well as massive amounts of geometry data (e.g., large number of simple building models as well as smaller number of buildings modeled in high detail). Vegetation models represent another source of massive data size; a single tree model could contain, e.g., approximately 150,000 polygons.

Distributed resources: In today's so called geospatial data infrastructures (GDIs), the different components (i.e., base data) of virtual 3D city models as well as functionalities to access, and process (e.g., analyze) virtual 3D city models can be distributed over the Internet. In specific use cases such as emergency response scenarios, they need to be identified, assembled, and accessed in an ad-hoc manner.

Heterogeneity: Virtual 3D city models are inherently heterogeneous, e.g., in syntax (file formats), schemas (description models), and semantics (conceptual models).

As an example, the virtual 3D city model of Berlin contains about 550,000 building models in moderate and/or high detail, textured with more than 3 million single (real-world) façade textures. The aerial image of Berlin (covering an area of around 850 km^2) has a data size of 250 GB. Together with additional thematic data (public transport data, land value data, solar potential) the total size of the virtual 3D city model of Berlin is about 700 GB.

1.2 Service-based approach

The various complexities of virtual 3D city models have an impact on their creation, analysis, publishing, and usage. Our overall approach to tackle these complexities and to cope with these challenges is to design and develop a distributed 3D geovisualization system as a technical framework for 3D geodata integration, analysis, and usage. For this, we apply and combine principles from Service-Oriented Computing (SOC), general principles from 3D visualization systems, and standards of the Open Geospatial Consortium (OGC).

To make complex 3D city models available even for small devices (e.g., smart phones, tablets), we have developed a client/server system that is based on server-side management and 3D rendering [1]: A portrayal server is hosting a 3D city model in a preprocessed form that is optimized for rendering, synthesizes images of 3D views of this data, and transfers these images to a client, which (in the simplest case) only displays these images. By this, the 3D client is decoupled from the complexity of the underlying 3D geodata. Also, we can optimize data structures, algorithms and rendering techniques with respect to specialized software and hardware for 3D geodata management and 3D rendering at the server-side.

Our project in the context of the HPI Future SOC Lab aims on research and development of how to exploit the Lab capabilities for the development and operation of such a distributed 3D visualization system, especially for 3D geodata preprocessing, analysis, and visualization. The capabilities of interest include the availability of many cores, large main-memory, GPGPU-based computing, and parallel rendering.

2 Project Work & Next Steps

During the last project phase, we continued our research and development on fundamental concepts and techniques in the area of service-based 3D rendering and service-based, interactive 3D visualization. The techniques developed so far rely on and take advantage of multi-core/multi-threading processing capabilities, the availability of large memory, and GPGPU systems as provided by the HPI Future SOC Lab. By exploiting these capabilities, we are able to accelerate processing, management and visualization of massive amounts of 3D geodata in a way that new applications in the area of 3D geovisualization become feasible.

Project work was done in three major areas – processing of massive 3D point clouds, processing of massive 3D city models, and assisted 3D camera control.

2.1 Processing massive 3D point clouds

We have continued our research on the processing, analysis, and visualization of massive 3D point cloud data based on multi-CPU and multi-GPU approaches including algorithms for the spatial organization of massive 3D point cloud data and for the computation of simplified representations of this data.

During last project phases, we could improve speed and quality of the spatial organization and the rasterization of 3D point clouds:

Spatial organi ation: Organization of massive point clouds is required to efficiently access and spatially analyze 3D points; quadtrees and octrees represent common structures for their organization. For this task we used a PARTREE algorithm, which creates several sub trees, which are combined to a single one.

Rasteri ation: Rastered 3D point clouds are a central component for visualization techniques and processing algorithms, as they allow efficient access to points within a specific bounding box. Rasterization transforms arbitrary distributed 3D points into a gridded, reduced, and consolidated representation; representative points are selected and missing points are computed and complemented. For this task we implemented a four-step process including: detection of the raster cell a point should be assigned to; ordering the points according to their raster cells; computing representative points for each cell; interpolating points for empty cells. A CUDA-based version of this algorithm was tested with the Future SOC Lab's TESLA system, which resulted in the rasterization of 30.5 million 3D points in 22 minutes in contrast to more than 5 hours on a single-threaded CPU version.

Since then, we have further improved our algorithms and have been working on segmentation algorithms, i.e., analysis algorithms, which allow for the classification of 3D points as part of, e.g., build infrastructure, vegetation, or terrain. Next steps include test and optimize these algorithms by help of the Future SOC Lab to make them ready to serve as a building block of an SOA-enabled 3D point cloud processing and analysis pipeline.

2.2 Processing massive 3D city models

In the field of 3D city model visualization and distribution, we continued our research on the processing of very large 3D city model data from CityGML data, which includes data extraction, geometry optimization and texture optimization. In a first iteration of adjusting our algorithms and testing those with HPC servers, we could significantly reduce the time to process very large sets of texture data (550,000 buildings with real façade textures) from more than a week on a standard desktop PC (2.8 GHz, 8 logical cores, 6 GB RAM) to less than 15 hours on the Future SOC Lab's RX600-S5-1 (48 logical cores, 256 GB RAM).

Also, we continued refactoring of the algorithms and processes used for texture optimization and consolidation, which allows us to generate and manage several texture trees in parallel. Next steps include testing and using this improved technique on the Future SOC Lab's HPC-servers to develop a fast data preprocessing tool chain, which is crucial for coping with continuous updates of the underlying data and for ensuring that a 3D visualization is up to date as much as possible.

The data preprocessed this way forms the basis for the client/server-based rendering and visualization concept and system described above. Based on this technology, 3D geodata could be served in a scalable way to various platforms [2].

In the field of 3D city model processing, we are also planning to exploit the HPC capabilities for the integration of different massive data sources for 3D city models and additional relevant information, namely data from CityGML models and data from OpenStreetMap to create an information-rich 3D city model, which can be used as a spatial user interface to the original underlying data. Tasks will include to integrate data from CityGML files and OpenStreetMap to come up with an integrated, rich database, which should be as up to data as possible.

Additionally, we plan to flank this work by researching techniques for software-based and parallel rendering of massive 3D city models based on our previous work; we expect new insights in how to efficiently organize and preprocess massive 3D models (geometries, appearance information, thematic data) for cloud-enabled visualization and distribution of complex 3D models and for object-related information retrieval.

2.3 Assisted 3D camera control

In the last project phase, we also worked on algorithms and methods for service-based technologies for assisted interaction and camera control in massive virtual 3D city models. As a first step in such a process we have been developing heuristics, algorithms and a machine learning based process for generating and classifying so called "best views" on virtual 3D city models. Heuristics include geometrical and visual characteristics.

As next step, the algorithms and techniques developed need to be adapted and implemented for even better exploiting the Future SOC processing capabilities. This will allow us to compute a multi-dimensional "navigation space" for complete 3D city models very fast as well as to query in real-time user-specific and task-specific camera positions and paths.

3 Conclusions

This report briefly described the subject and preliminary results of our research and development in the context of the HPI Future SOC Lab as well as intended future work. Work and results were mainly in the areas of processing massive 3D point clouds, processing massive 3D city models, and assisted 3D camera control. Here, we could, e.g., reduce the time required to preprocess raw 3D geodata (CityGML data with geometry and textures; also massive 3D point clouds) and to make this data ready for visualization, analysis, and use. Also, we identified additional opportunities for optimizing these algorithms. More generally, our work leads to new opportunities for research and development on advanced and innovative technologies for the exploitation (e.g., analysis and visualization) of massive spatial 3D data sets.

Acknowledgements

We would like to thank our partner 3D Content Logistics for providing access to their 3D visualization platform as a basis for our work in this field.

References

[1] J. Döllner, B. Hagedorn, J. Klimke: Server-Based Rendering of Large 3D Scenes for Mobile Devices Using G-Buffer Cube Maps. 17th Int. Conference on 3D Web Technology, 2012.

[2] J. Klimke, B. Hagedorn, J. Döllner: Scalable Multi-Platform Distribution of 3D Content, 8th Int. 3D GeoInfo Conference, 2013. *(accepted)*