3D GEOSPATIAL DATA VISUALIZATION IN VR

Michal Mitter[1], Jaromír Landa[1]✉, Ivo Pisařovic[1], David Procházka[1], Michael Varga[1], František Dařena[1]

[1] *Mendel University in Brno, Czech Republic*

EUROPEAN JOURNAL OF BUSINESS SCIENCE AND TECHNOLOGY

Volume 11 Issue 1
ISSN 2694-7161
www.ejobsat.com

ABSTRACT

The use of Virtual Reality (VR) to visualize data is increasingly common in various fields, from medical science to education. However, visualization of geospatial data in VR is still not handled well. GIS data comes in various formats and converting them to 3D can be challenging. The choice of a format capable of streaming large amounts of spatial data in the VR scene is crucial. However, the challenge lies in ensuring the reusability of the presented scenes with different GIS data. Most applications (including the 3D scene models) are created for a single use case and do not allow for simple interchangeability of presented data. This article proposes a reusable architecture of a general-purpose web service for processing and visualizing spatial data in VR using 3D Tiles as the chosen format. The goal of the architecture is to allow fast VR scene generation from GIS data. The critical aspect is the conversion of standard GIS format to 3D Tiles. Several conversion tools were tested and compared using publicly available point and polygon feature layers in order to choose which can be used in the proposed architecture.

KEY WORDS

virtual reality, spatial data, 3D Tiles, data conversion

JEL CODES

L63, L86

MITTER, Michal, LANDA, Jaromír, PISAŘOVIC, Ivo, PROCHÁZKA, David, VARGA, Michael, and DAŘENA, František. 2025. 3D Geospatial Data Visualization in VR. *European Journal of Business Science and Technology,* 11 (1): 129–140. ISSN 2694-7161, DOI 10.11118/ejobsat.2025.005.

1 INTRODUCTION

Visualization of geospatial data within Geographic Information Systems (GIS) is a common task in various applications. The most frequent method is to visualize the data in 2D using widely used software like the open-source application QGIS or ESRI's ArcGIS Pro. Visualization of geospatial data in 3D is also common, referred to as 3D GIS (Boulos et al., 2017). 3D GIS is primarily a visualization tool that concentrates on topological models and frameworks for portraying spatial relationships (Antoniou et al., 2020). The subsequent level of visualization entails the utilization of Virtual Reality (VR). The fusion of VR and GIS is referred to as VR GIS (Virtual Reality Geographic Information System) (Lv et al., 2016). Using VR with geospatial data offers users a heightened level of immersiveness that is lacking in conventional GIS applications. Research indicates that immersive virtual reality (VR) provides users with a higher level of embodiment compared to traditional 3D GIS (Bagher et al., 2022). VR can reduce distractions through immersive experiences and offer more space for users with its 360-degree capabilities (Spur et al., 2020; Wang et al., 2020). Additionally, it can improve spatial analysis and information query abilities (Zuo, 2020).

To create a VR visualization of GIS data, two commonly used approaches are game engines (Du et al., 2018; Halik, 2018) and WebXR technologies (Sermet and Demir, 2022). However, all VR GIS applications share a common challenge – the production of 3D geospatial data for visualization in VR. Unlike traditional GIS applications that only require the use of 2D information, VR GIS necessitates a large amount of data in all three dimensions. This presents a significant challenge to address as the source data is often only available in conventional GIS formats, like Shapefile or GeoJSON, or in three-dimensional formats, such as CityGML (Rahman et al., 2019), that are not optimized for efficient visualizations, but rather just for the representation of three-dimensional models. As a result, to create VR GIS, the data must be converted into a format that is optimized for the streaming of large three-dimensional geospatial data. The two most commonly used formats today are 3D Tiles from Cesium and Esri Indexed 3D Scene Layer (i3S) (Würstle et al., 2022). Both Cesium and Esri offer software development kits (SDKs) for creating virtual reality (VR) applications using Unity or Unreal Engine.

Nevertheless, Unity and UE are only visualization tools and must be provided with 3D data (most commonly in the form of standard 3D model format). The most common approaches create the 3D scene models in specialized software and then import them to game engines for visualization. This poses a challenge in situations, where the underlying data often change and the entire scene must be recreated from the beginning. This article outlines the architecture of the VRGIS application for visualizing geospatial data without the need for the intermediate step (i.e., the 3D model creation in specialized software manually). The application's primary objective is to translate standard GIS formats and present them as 3D visualizations in VR, thus allowing to create the scene from new data easily. The main format used for visualization in the proposed application is 3D Tiles. The 3D Tiles format was chosen because it can be easily created using many available converters, unlike the i3S which can be created primarily only using ESRI software.

The article is structured as follows: Section 2 presents a literature review of approaches to visualizing geospatial data in VR, including commonly used spatial data visualization and streaming formats for virtual reality. Section 3 describes the proposed architecture of the system for geospatial data visualization in VR. Section 4 outlines the crucial part of the application, the data conversion from standard GIS formats to formats suitable for VR visualization. In section 4, the tests were conducted on the available tools for converting geospatial data from GeoJSON to 3D Tiles. Finally, Section 5 discusses the proposed architecture and the selected technologies.

2 LITERATURE REVIEW

This chapter first reviews the use of various geospatial data sources for different visualization purposes. The second part outlines the technical details of data formats suitable for storing and transfer of geospatial data.

2.1 Use Cases for Geospatial Data in VR

The visualization of spatial data in VR is an established field, with numerous examples available. For instance, the PeakLensVR application developed by La Salandra et al. (2019) facilitates the capture of panoramic images using mobile devices, which can then be augmented with data from OpenStreetMaps. Meanwhile, Wang et al. (2019) utilizes 360-degree panoramic images to enhance a system designed to assist with flood risk management through VR visualizations. To increase the visualization element, the addition of cars and trees was implemented (León et al., 2023). The study employs virtual reality (VR) to evaluate the selection of evacuation shelters during emergencies, specifically tsunamis. The application was developed to examine the preference for certain types of buildings during emergency situations. Another application of VR is in aiding the urban decision-making process for the Rafah planning authorities (El Halabi et al., 2019). Data is prepared through traditional GIS, imported to Esri City Engine, and visualized using the 360 VR Experience. The findings suggest that virtual reality (VR) can enhance comprehension amongst planners and other stakeholders. Nonetheless, it has been noted that editing virtual models can be challenging.

Another example is presented in Lindquist and Campbell-Arvai (2021). The Land.Info software was designed to aid in co-designing vacant lots with 3D visualization. The study details the development of a series of virtual environment prototypes, each of which was tested in a series of workshops with the public with the goal of increasing stakeholder engagement. Yang et al. (2018) explored various methods of rendering globes in virtual reality. The exocentric globe, flat map, egocentric globe, and curved map were all tested. The results indicate that the exocentric globe is typically the optimal choice for visualizing the globe in virtual reality (VR). Establishing a spatial presence is a crucial aspect of visualization. The user must recognize the virtual environment as their primary space (Hruby et al., 2018). VR can be used to visualize not only the globe but also traditional 2D maps (Spur et al., 2020). The authors sought to determine the optimal method for visualizing a two-dimensional (2D) map in virtual reality (VR). They proposed a three-dimensional (3D) stack of 2D maps. The projection of a 2D map in VR can be found in (Letić et al., 2018).

Indoor building presentations are also commonplace. The 4D immersive presentation of a virtual museum was developed by (Kersten et al., 2017). The authors reconstructed the museum's 3D model through terrestrial laser scanning. Additional modeling was required for the model to meet the necessary requirements. The VR environment was created using the Unreal Engine (Choromański et al., 2019). Campanaro and Landeschi (2022) fused VR technology with eye-tracking to monitor user visual attention within the Pompeian house. The 3D model of the house was imported into the Unity engine, while the eye-tracking data were analyzed in ArcGIS Pro. BIM is one of the many areas where the proposed research could be utilized.

Large-scale areas can also be visualized through virtual reality (VR), as demonstrated by the work of (Halik, 2018). They produced a 3D representation of a built-up area in Poland, converting the original vector data to the fbx format and visualizing it in the Unity engine. Although the conversion to 3D model formats is widespread, it can result in the loss of attribute and spatial information. Antoniou et al. (2020) used UAV-based photogrammetry to create a high-resolution 3D model of the Metaxa Mine in Greece. The 3D model was subsequently imported into GeoVR software for advanced GIS analysis. Vanden Broucke and Deligiannis (2019) created a VR application for visualizing heterogeneous, smart city data of Brussels. A comparison was made between the VR application and more conventional web platforms, with the results indicating a decrease in frustration levels when exploring the city.

Sermet and Demir (2022) created the GeospatialVR which is an open-source collaborative virtual reality framework. The framework goal is to generate the virtual reality scene in real time from common data sources using terrain data, elevation mode, and many more. The data is visualized from the Bird's eye view and allows users to collaborate and see other users' positions in the scene. To enhance the visualization, the animation of elements such as water flow or fire are used.

2.2 Formats for Visualization of Geospatial Data in VR

Geospatial data is typically visualized using standard 3D formats, such as fbx or obj, due to their compatibility with game engines. However, these formats are unsuitable for streaming large amounts of data. The two most commonly used formats for streaming geospatial data are 3D Tiles and I3S.

I3S is an open standard developed by Esri for storing and streaming large 3D geospatial data. It supports various types of data, including 3D models, BIM data, photogrammetry, point features, and point clouds. The standard is designed to be cross-platform and scalable. It defines a REST API for accessing the data. Additionally, the data can be stored in a `.slpk` file. The Open Geospatial Consortium (OGC) adopted the specification as a community standard in 2017. Since then, Esri has released several versions of the standard (Belayneh, 2022; Esri, 2024; Open Geospatial Consortium, 2023).

Another option is the 3D Tiles format, which is an open standard with the same purpose as I3S. Cesium introduced it in 2015, and it supports photogrammetry, 3D models, and point clouds. A single 3D Tiles dataset is called a tileset. The tileset is composed of tiles that are arranged hierarchically into a spatial data structure. The children of a tile fit into its bounding volume. Bounding volume can be an oriented bounding box, a bounding sphere, or a geographic region defined by minimum and maximum latitudes, longitudes, and heights. Bounding volume hierarchy can be defined explicitly by providing bounding volumes for every tile or it can be defined implicitly using common spatial division structures such as quadtrees or octrees. Implicit tiling enables random access to the tiles. The 3D content of 3D Tiles is stored in an open glTF standard, commonly known as the 'JPEG of 3D'. 3D Tiles enables the attachment of metadata (feature table in GIS) to various parts of a dataset, such as a texel or tile (CesiumGS, 2024a). The OGC accepted this standard in 2019 (Open Geospatial Consortium, 2019). The standard is supplemented by the Quantized Mesh standard for visualizing terrain data (CesiumGS, 2024b).

3 THE PROPOSED SYSTEM ARCHITECTURE FOR SPATIAL DATA VISUALIZATION IN VR

In order to visualize geospatial data in VR, we propose a system architecture (see Fig. 1). The architecture was designed based on the review performed in Section 2 and consists of three main parts: a) backend service for storing and processing spatial data, b) frontend application for managing spatial data and preparing scenarios for various use cases, and c) client application for displaying the data in VR headsets. Unlike traditional approaches where the main focus is on the visualization part (La Salandra et al., 2019; Lindquist and Campbell-Arvai, 2021), our architecture is designed to cover all aspects of the visualization pipeline. The purpose of the architecture is to enable the users to upload prepared files in a

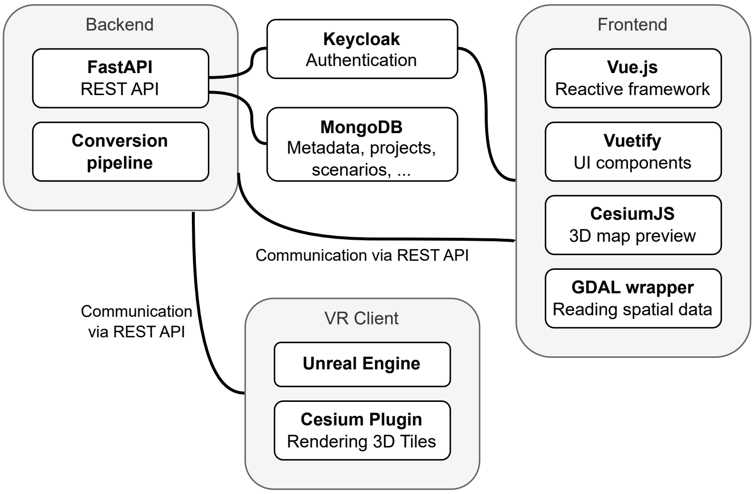

Fig. 1: A diagram of the system's architecture along with the used technologies for each part

standard GIS format. The upload is done using a front-end web application. The GIS data is converted to 3D Tiles at the backend and based on the provided user scene definition, a VR scene configuration is sent to the visualization part (Unreal engine). This is in contrast to approaches such as Kersten et al. (2017) and El Halabi et al. (2019) where a single-purpose scene is created. This approach has also an added advantage in the possibility to easily add other elements to the scene, such as trees or cars. This can be very beneficial to user experience as mentioned in León et al. (2023) and Lindquist and Campbell-Arvai (2021).

3.1 Backend Service for Storing and Processing Spatial Data

A crucial aspect of the backend is the conversion of spatial data to 3D Tiles. As most libraries for this task are written in Python, it is beneficial to implement the entire backend in Python. FastAPI, a commonly used framework for building API services, was used.

Spatial data are typically large textual or binary files and are stored using a standard file system. Metadata about assets, camera positions, or scenarios will be saved in MongoDB. The data structure is simple with few relations. Horizontal scaling of NoSQL databases may be beneficial when deploying our solution for multiple worldwide purposes. Additionally, user management and authentication will be outsourced to an open-source service Keycloak.

3.2 Frontend Application for Managing Spatial Data and Preparing Scenarios for Various Use Cases

The frontend application is intended for administrators to upload spatial data assets, convert them into spatial data, combine them using layers, and define projects with scenarios based on these data.

To ensure long-term stability, Typescript will be used as our proposed solution. The reactive framework Vue.js, currently in version 3, will also be used. Vuetify offers a collection of reusable UI components that simplify the implementation of user interfaces. The frontend application will be connected to Keycloak to manage authentication and access rights.

Additionally, libraries such as CesiumJS will be utilized to provide 3D previews of assets that users are working with. Furthermore, the GDAL wrapper for Javascript (`https://github.com/azavea/loam`) enables access to spatial data files directly in the browser.

3.3 Client Application for Displaying the Data in VR Headsets

The application for displaying spatial data to the user is created within the Unreal Engine version 5.3, which brings a wide range of supported platforms. The client application will be optimized for Meta Quest Pro VR glasses.

Within the UE, a plugin called Cesium for Unreal is used to display 3D Tiles. This plugin is connected to the Cesium Ion platform, from where it is possible to use a large number of freely available pre-prepared data (Imagery, Terrains, 3D buildings, etc.), or use your own content generated by the conversion from standard GIS format on the backend.

Spatial data will be rendered according to the data provided during the client application communication with the backend service. A standard JSON format is used. This file contains the definition of scenarios (i.e., the 3D scene) and the layers in them. Furthermore, this file contains additional information about individual layers. Layers can be of two types, namely layers located on the backend of the application and layers located on the Cesium Ion platform. According to the individual scenarios, layers will be imported into the application, so that the user can work within individual scenarios or move freely between them.

4 GIS DATA CONVERSION TO 3D TILES EVALUATION

As mentioned earlier, the crucial aspect of the backend is data conversion from a standard GIS format to 3D Tiles To test the possibility of effective conversion of GIS data from GeoJSON to 3D Tiles, two datasets were selected from the official GIS portal of the city of Brno. The two datasets represent common conversion and visualization scenarios: a) *points* layer with single objects, b) *polygon* layer representing box models of buildings. *Point* and *polygon* layers were chosen as the most used ones. Other types, like *line* layers, are not considered in this research as almost everything can be represented as point and polygons (e.g. lines can be represented as thin polygons).

Each dataset was converted using one or more publicly available tools for 3D Tiles conversion. The results were visually inspected using CesiumJS with the 3D Tiles inspector widget enabled and validated using the Cesium 3D Tiles validator[1]. The initial dataset[2] consists of a point layer displaying the location of 41,704 streetlamps in Brno. The second dataset[3] is a polygon layer containing 127,601 building bases. The first dataset is converted into a 3D Tiles layer that contains individual 3D models positioned at the points' locations. The second dataset is converted into a box representation of the building bases stored in the polygon layer.

The conversions were tested on a PC with an AMD Ryzen 7 3700X (16) running at 3.599 GHz, an NVIDIA GeForce RTX 2080, 32 GB of RAM, and the Ubuntu 22.04.3 LTS x86_64 operating system.

4.1 Point Layer Conversion to 3D Tiles

The streetlamp dataset in Brno was converted into instanced 3D models of 3D Tiles using `i3dm.export`[4]. This tool is currently the only available option for converting point features into instanced 3D models. It exports the `i3dm` and `tileset.json` from the PostGIS table and is mostly developed and maintained by a single individual. See the whole pipeline in Fig. 2.

Each lamp object is represented by a single glTF model. The 3D model instances were ac-

[1] https://github.com/CesiumGS/3d-tiles-validator
[2] Street lights [Stožáry veřejného osvětlení], https://data.brno.cz
[3] Buildings research in Brno [Průzkum budov v Brně] 2018–2020, https://data.brno.cz
[4] https://github.com/Geodan/i3dm.export

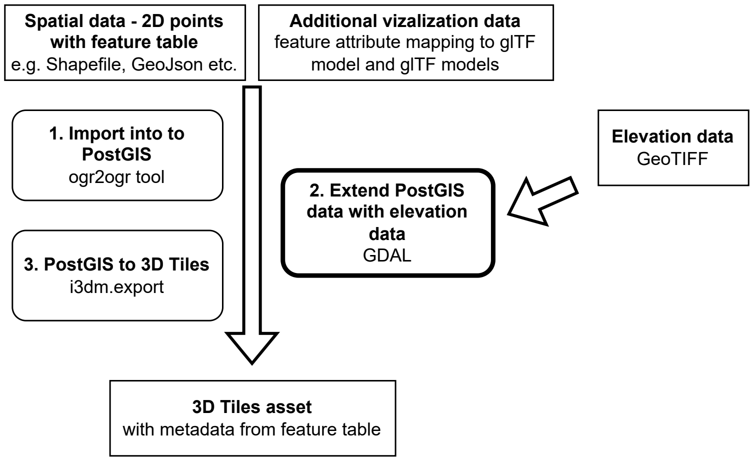

Fig. 2: The proposed pipeline for processing point spatial data

curately placed at their corresponding latitude and longitude coordinates on the globe. It is worth noting that the `i3dm.export` experimentally supports the EXT_mesh_gpu_instancing extension from 3D Tiles 1.1. Furthermore, it also supports export using `.i3dm` or `.cmpt` extensions, which have already been deprecated in the 3D Tiles specification.

The conversion of the test point features took a few seconds. The data size resulting from the model was approximately 10 MB, which is significantly smaller than the model size (128 kB) multiplied by approximately 40,000. Additionally, the size could be further reduced by using an external model option. After conversion, the data was tested using the Cesium 3D Tiles validator and no errors were found.

4.2 Polygon Layer Conversion to 3D Tiles

Four open-source tools were selected to test the generation of 3D Tiles from a polygon layer: `pg2b3dm`[5], `fanvanzh/3dtiles`[6], `py3dtilers`[7], and `viz-3dtiles`[8]. Since all the solutions had issues such as missing documentation, we conducted tests to determine their suitability for our purposes. The four selected tools were the only ones that demonstrated at least some potential. Other tools were either outdated or incomplete and were therefore excluded from this study.

The tools were evaluated based on their features, interpretation of the bounding volume, usage of glTF, and visual representation of element location using CesiumJS. The results of the evaluation are presented in Tab. 1.

The `pg3b3dm` tool is used to convert PostGIS 3D geometries to 3D Tiles. The tool `tesselate_buildings`, created by the same author, is used to create 3D geometries from 2D polygons. The conversion process and its options are explained in the documentation (Geodan, 2024). The conversion steps are as follows: use the `ogr2ogr` tool (from the GDAL package) to convert from GeoJSON (or any 2D polygon format that is supported by GDAL) to a PostGIS table, use `tesselate_buildings` to create 3D geometries, and finally create 3D Tiles from the 3D geometries with `pg2b3dm`. The tool relies on widely used PostGIS and GDAL. It offers implicit tiling and supports glTF content from 3D Tiles 1.1. However,

[5] `https://github.com/Geodan/pg2b3dm`
[6] `https://github.com/fanvanzh/3dtiles`
[7] `https://github.com/VCityTeam/py3dtilers`
[8] `https://github.com/PermafrostDiscoveryGateway/viz-3dtiles`

Tab. 1: Polygon layer to 3D Tiles conversion tools test results

	pg2b3dm	fanvanzh/3dtiles	py3dtilers	viz-3dtiles
Position on the globe in CesiumJS	Correct	Correct	Incorrect	Incorrect
Per feature height	Yes	Yes	Yes	No
Per feature altitude	No	No[9]	Yes	No[10]
Per feature color	Yes	No	Yes	No
Bounding volume	Region	Region	Box	Box
Supports metadata	Yes	Yes	Yes	Yes
GPU Draw calls per tile	Few	Hundreds	Few	Few
glTF as content	Yes, optional	No	No	No
Divides data into tiles	Yes	Yes	Yes	No
Implicit tiling	Quadtree	–	–	–
3D Tiles Validator	Errors when glTF as content enabled	Errors	Errors	Errors
Conversion duration	1m 47s	12s	5m 20s	1m 40s
Programming language	C#	C++	Python	Python
Github stars	272	1.7k	141	4
Number of main code contributors	Single	Single	Single	Few

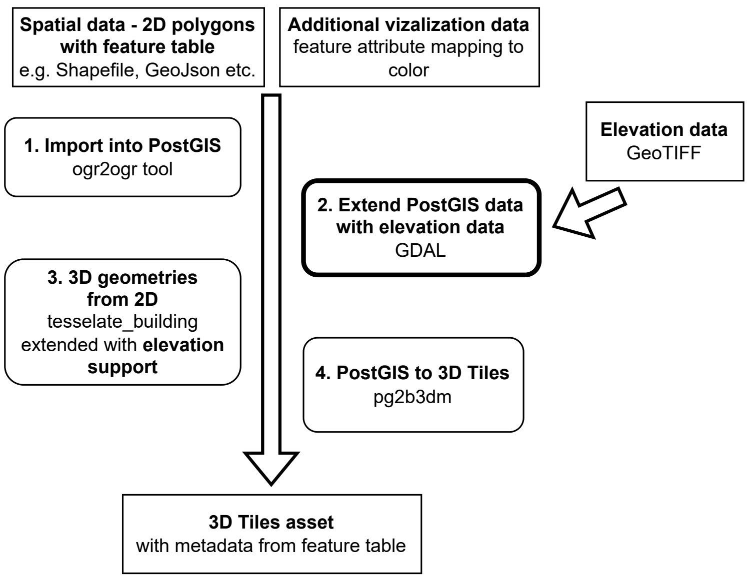

Fig. 3: The proposed pipeline for processing polygon spatial data

the glTF content option does not support feature metadata, and per-feature altitude is also not yet supported. See the whole pipeline in Fig. 3.

The second tested was `fanvanzh/3dtiles` tool. The `fanvanzh/3dtiles` tool has limited documentation compared to other similar tools, which can make it difficult to understand. Addi-

[9] `https://github.com/fanvanzh/3dtiles/issues/256`
[10] `https://github.com/PermafrostDiscoveryGateway/viz-3dtiles/issues/18`

tionally, it only accepts shapefiles for polygon conversion and lacks the option to set feature altitude. While the conversion process is fast, the resulting 3D Tiles tileset has poor rendering performance in CesiumJS due to hundreds of draw calls per tile.

The third tool tested was `py3dTilers`. It is well-documented. The resulting tileset is rendered in CesiumJS, with the latitude and longitude coordinates swapped.

The last tested tool was `Viz-3dtiles`. The forked version of `py3dtiles` was used, but the authors did not contribute changes to the upstream repository. This version shares the same issue as `py3dtiles` with swapped coordinates. When visualizing in CesiumJS, the geometry appeared jittery when the camera was moved. Unfortunately, there is no documentation available on how this version works. Additionally, only a single tile was created, resulting in a 120 MB `b3dm` file containing all the geometry.

5 DISCUSSION AND CONCLUSION

The visualization of geospatial data in VR is a common problem. One of the challenges is to effectively visualize large amounts of GIS data in the form of 3D models. Common approaches use standard 3D model formats (e.g., fbx) created for specific scenarios or scenes. However, these approaches lack the ability to easily switch models or scenes and are mainly used for single-purpose applications. Another challenge is that the standard 3D model formats are missing the location information and cannot be easily combined with any geospatial data. In this paper, we presented a system architecture for the universal visualization of large amounts of GIS data in VR. The proposed architecture can be used for any use case where the data or the focus of the scene can change and thus, in a traditional scenario, the scene and all objects would have to be recreated manually.

The main challenge is the generation of 3D data. The 3D Tiles format was chosen for its versatility. The i3S format was not chosen because of the vendor lock. Several tools for converting spatial data into the 3D Tiles format were tested. Out of the tested tools, the pg2b3dm was chosen. The main reason is the support of many features (implicit tiling, glTF content) from the latest 3D Tiles version 1.1. It is also easy to understand and can be extended to support elevation per feature. If the input data is missing elevation information, it is necessary to retrieve the elevation from a terrain model to be used in the visualization.

For importing data into PostGIS, the Ogr2ogr documentation suggests increasing the number of INSERT statements in transactions when populating a database with a lot of data.[11] The external model option can be used to save even more disk space, the only drawback being that the glTF model must be available from a web server along with the 3D tiles data. To support visualization of terrain, the elevation property of the given terrain would need to be queried for each feature in GeoJSON during the format conversion. It is important to remember that 3D Tiles can be used for all types of 3D spatial content, and extruded 2D polygons are only of one type.

This pipeline for processing 3D spatial data was wrapped into a web service based on commonly used technologies like FastAPI, Vue.js, and Typescript. Users may use this application to upload their geospatial data, convert them into an efficient format, and prepare a customized visualization in a user-friendly way via GUI. The web service is accompanied by a VR client implemented in Unreal Engine to visualize the provided data.

Our approach's primary contribution is the on-the-fly generation of VR scenes from large GIS datasets. Unlike common approaches that create and use a single-purpose application with a VR scene, our system allows for easy exchange of scene layers when visualizing new data.

[11] `https://gdal.org/programs/ogr2ogr.html#performance-hints`

6 ACKNOWLEDGEMENTS

This paper was supported by the project CZ.02.1.01/0.0/0.0/16_017/0002334 Research Infrastructure for Young Scientists, this is co-financed from Operational Programme Research, Development and Education and also supported by grant No. IGA-PEF-TP-22-006 (Opportunities to use metaverse technology to support business processes) of the Internal Grant Agency FBE MENDELU.

7 REFERENCES

Antoniou, V., Bonali, F. L., Nomikou, P., Tibaldi, A., Melissinos, P., Mariotto, F. P., Vitello, F., Krokos, M. and Whitworth, M. 2020. Integrating Virtual Reality and GIS Tools for Geological Mapping, Data Collection and Analysis: An Example from the Metaxa Mine, Santorini (Greece). *Applied Sciences*, 10 (23), 8317. DOI: 10.3390/app10238317.

Bagher, M. M., Sajjadi, P., Wallgrün, J. O., LaFemina, P. C. and Klippel, A. 2022. Virtual Reality for Geospatial Education: Immersive Technologies Enhance Sense of Embodiment. *Cartography and Geographic Information Science*, 50 (3), 233–248. DOI: 10.1080/15230406.2022.2122569.

Belayneh, T. 2022. Indexed 3D Scene Layers (I3S) – An Efficient Encoding and Streaming OGC Community Standard for Massive Geospatial Content. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022, 349–355. DOI: 10.5194/isprs-archives-xliii-b4-2022-349-2022.

Boulos, M. N. K., Lu, Z., Guerrero, P., Jennett, C. and Steed, A. 2017. From Urban Planning and Emergency Training to Pokémon Go: Applications of Virtual Reality GIS (VRGIS) and Augmented Reality GIS (ARGIS) in Personal, Public and Environmental Health. *International Journal of Health Geographics*, 16 (7). DOI: 10.1186/s12942-017-0081-0.

Campanaro, D. M. and Landeschi, G. 2022. Re-Viewing Pompeian Domestic Space Through Combined Virtual Reality-Based Eye Tracking and 3D GIS. *Antiquity*, 96 (386), 479–486. DOI: 10.15184/aqy.2022.12.

CesiumGS. 2024a. *3D Tiles Specification* [online]. GitHub. Available at: `https://github.com/CesiumGS/3d-tiles`

CesiumGS. 2024b. *Quantized-mesh Specification* [online]. GitHub. Available at: `https://github.com/CesiumGS/quantized-mesh`

Choromański, K., Łobodecki, J., Puchała, K. and Ostrowski, W. 2019. Development of Virtual Reality Application for Cultural Heritage Visualization for Multi-Source 3D Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W9, 261–267. DOI: 10.5194/isprs-archives-xlii-2-w9-261-2019.

Du, J., Zou, Z., Shi, Y. and Zhao, D. 2018. Zero Latency: Real-Time Synchronization of BIM Data in Virtual Reality for Collaborative Decision-Making. *Automation in Construction*, 85, 51–64. DOI: 10.1016/j.autcon.2017.10.009.

El Halabi, A. S., El Sayad, Z. T. and Ayad, H. M. 2019. VRGIS as Assistance Tool for Urban Decision Making. *Alexandria Engineering Journal*, 58 (1), 367–375. DOI: 10.1016/j.aej.2018.07.016.

Esri. 2024. *Scene Layers: Service and Package Standard* [online]. GitHub. Available at: `https://github.com/Esri/i3s-spec`

Geodan. 2024. *Pg2b3dm: Tool for Creating 3D Tiles from PostGIS Geometries* [online]. GitHub. Available at: `https://github.com/Geodan/pg2b3dm`

Halik, Ł. 2018. Challenges in Converting the Polish Topographic Database of Built-Up Areas into 3D Virtual Reality Geovisualization. *The Cartographic Journal*, 55 (4), 391–399. DOI: 10.1080/00087041.2018.1541204.

Hruby, F., Ressl, R. and de la Borbolla del Valle, G. 2018. Geovisualization with Immersive Virtual Environments in Theory and Practice. *International Journal of Digital Earth*, 12 (2), 123–136. DOI: 10.1080/17538947.2018.1501106.

Kersten, T. P., Tschirschwitz, F. and Deggim, S. 2017. Development of a Virtual Museum Including a 4D Presentation of Building History in Virtual Reality. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W3, 361–367. DOI: 10.5194/isprs-archives-xlii-2-w3-361-2017.

La Salandra, A., Frajberg, D. and Fraternali, P. 2019. A Virtual Reality Application for Augmented Panoramic Mountain Images. *Virtual Reality*, 24 (1), 123–141. DOI: 10.1007/s10055-019-00385-x.

León, J., Gubler, A., Catalán, P. A., Correa, M., Castañeda, J. V., Beninati, G. and Ogueda, A. 2023. Assessing Potential Tsunami Vertical-Evacuation Practices: A Study of Four Cases in Chile Using Virtual Reality and GIS. *International Journal of Disaster Risk Reduction*, 98, 104098. DOI: 10.1016/j.ijdrr.2023.104098.

Letić, M., Nenadić, K. and Nikolić, L. 2018. Real-Time Map Projection in Virtual Reality Using WebVR. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1439–1443. DOI: 10.23919/mipro.2018.8400259.

Lindquist, M. and Campbell-Arvai, V. 2021. Co-Designing Vacant Lots Using Interactive 3D Visualizations – Development and Application of the Land.Info DSS. *Landscape and Urban Planning*, 210, 104082. DOI: 10.1016/j.landurbplan.2021.104082.

Lv, Z., Yin, T., Zhang, X., Song, H. and Chen, G. 2016. Virtual Reality Smart City based on WebVRGIS. *IEEE Internet of Things Journal*, 3 (6), 1015–1024. DOI: 10.1109/jiot.2016.2546307.

Open Geospatial Consortium. 2019. *3D Tiles Specification 1.0* [online]. Available at: `https://docs.ogc.org/cs/22-025r4/22-025r4.html`

Open Geospatial Consortium. 2023. *Indexed 3D Scene Layers (I3S)* [online]. Available at: `https://www.ogc.org/standard/i3s/`

Rahman, A. A., Rashidan, H., Musliman, I. A., Buyuksalih, G., Bayburt, S. and Baskaraca, P. 2019. 3D Geospatial Database Schema for Istanbul 3D City Model. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W16, 11–16. DOI: 10.5194/isprs-archives-xlii-4-w16-11-2019.

Sermet, Y. and Demir, İ. 2022. GeospatialVR: A Web-Based Virtual Reality Framework for Collaborative Environmental Simulations. *Computers & Geosciences*, 159, 105010. DOI: 10.1016/j.cageo.2021.105010.

Spur, M., Tourre, V., David, E., Moreau, G. and Le Callet, P. 2020. MapStack: Exploring Multilayered Geospatial Data in Virtual Reality. In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, pp. 88–99. DOI: 10.5220/0008978600880099.

Vanden Broucke, S. and Deligiannis, N. 2019. Visualization of Real-Time Heterogeneous Smart City Data Using Virtual Reality. In *2019 IEEE International Smart Cities Conference (ISC2)*, pp. 685–690. DOI: 10.1109/isc246665.2019.9071699.

Wang, C., Hou, J., Miller, D., Brown, I. and Jiang, Y. 2019. Flood Risk Management in Sponge Cities: The Role of Integrated Simulation and 3D Visualization. *International Journal of Disaster Risk Reduction*, 39, 101139. DOI: 10.1016/j.ijdrr.2019.101139.

Wang, X., Cong, G., Yuen, D. A. and Luo, G. 2020. GeoVReality: A Computational Interactive Virtual Reality Visualization Framework and Workflow for Geophysical Research. *Physics of the Earth and Planetary Interiors*, 298, 106312. DOI: 10.1016/j.pepi.2019.106312.

Würstle, P., Padsala, R., Santhanavanich, T. and Coors, V. 2022. Viability Testing of Game Engine Usage for Visualization of 3D Geospatial Data with OGC Standards. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W2-2022, 281–288. DOI: 10.5194/isprs-annals-x-4-w2-2022-281-2022.

Yang, Y., Jenny, B., Dwyer, T., Marriott, K., Chen, H. and Cordeil, M. 2018. Maps and Globes in Virtual Reality. *Computer Graphics Forum*, 37 (3), 427–438. DOI: 10.1111/cgf.13431.

Zuo, H. 2020. The Construction of Stratigraphic Structure Model in Mining Area under Virtual Reality–Geographic Information System. *Arabian Journal of Geosciences*, 13 (17), 853. DOI: 10.1007/s12517-020-05844-3.

AUTHOR'S ADDRESS

Michal Mitter, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: xmitter@node.mendelu.cz

Jaromír Landa, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: jaromir.landa@.mendelu.cz (corresponding author)

Ivo Pisařovic, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: ivo.pisarovic@node.mendelu.cz

David Procházka, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: david.prochazka@node.mendelu.cz

Michael Varga, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: xvarga@node.mendelu.cz

František Dařena, Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: frantisek.darena@node.mendelu.cz