

CONSIDERATIONS WHEN CHOOSING APPLICATION DEVELOPMENT PLATFORMS

Emerging web and game engine tech for 3D cities



The increasing number of solutions for developing 3D city model applications poses a challenge for developers. This article outlines key considerations that help them to create applications efficiently while utilizing the potential benefits of the 3D data.



3D city modelling has become an established activity in major cities. As a part of this development, city models are increasingly developed to support numerous applications that involve large user groups, such as collaborative urban planning, pedestrian navigation, event planning and the discovery of services and businesses in cities.

The 3D city models themselves, produced by cities, public organizations and private companies, are increasingly widely

available, in some cases even as open data (Figure 1). Furthermore, thanks to the huge amount of open software, tutorials and other educational material that is available from online communities, learning how to utilize geospatial data has become easier than ever before.

Applications that utilize 3D city models can be developed with an increasing number of software solutions, ranging from simple tools to complex systems. This alone poses a challenge for developers when choosing the right solutions. Creating applications efficiently while utilizing the potential benefits of the 3D data remains the key challenge in 3D city modelling.

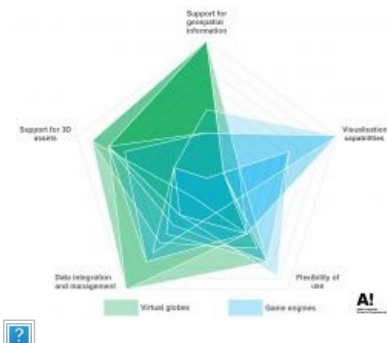


Figure 1: Helsinki's 3D city models are available as open data assets – the reality mesh model (left) and the semantic city information model (right).

For developing applications, two categories of platforms have been especially topical in recent years: browser-based 3D environments and 3D game engines. While both of these platforms offer 3D visualization capabilities, the game engines have been at the forefront of the virtual reality (VR) and augmented reality (AR) boom. At the same time, browser-based 3D has made it possible to create applications that can be easily distributed to large user groups and operated without separate software installations.

Browser-based platforms

The development of browser-based 3D rendering has enabled the transition of interactive 3D applications from desktop systems to browsers. Browser-based solutions often rely on WebGL technology and are limited by the performance constraints inherited from web browsers and the internet. At the same time, however, they are inherently close to easy sharing, online participation and the visualization of live data streams (e.g. from Internet of Things systems).

For geospatial online 3D systems, numerous use cases exist globally, applying 3D virtual globes as either the generic publishing platforms of 3D city models or for more dedicated use cases such as energy (e.g. the Helsinki Energy and Climate Atlas, see Figure 2).

In 3D GIS and virtual globes, browser-based systems have matured into entire software ecosystems that have come to include standardized data formats such as 3D tiles. The most commonly applied open-source components, such as CesiumJS, GeoServer or 3DCityDB, have become starting points for several commercial products (e.g. virtualcityMAP) or open-source frameworks (e.g. TerriaJS).



Figure 2: The Helsinki Energy and Climate Atlas contains energy-related building information in a visual and informative form and allows the user to explore Helsinki in a browser while retrieving information about buildings.

3D game engine platforms

A second significant development track has been the emergence of game engines as platforms for professional applications (e.g. visualizations, virtual experiences, simulations and gamified training) involving GIS data and building models. Game engines still lack the native support for geographic coordinate systems or map projections, but they excel in the visual quality and performance provided by the real-time 3D rendering that allows for engaging real-time interactions and experiences, VR and AR. Major software vendors have introduced tools that simplify the utilization of urban datasets on game engine platforms.

The most popular game engines have vast developer communities and offer extensive documentation that flattens the learning curve and makes them easier to access. Nowadays the game engine companies have started to invest heavily in expanding the utilization of their platforms outside the traditional fields of digital gaming and entertainment, such as in the automotive and AEC industries. The games industry is also showing a growing interest in utilizing 3D spatial data (e.g. the massive use of globally generated 3D data in the new Microsoft Flight Simulator).

Unity and Unreal Engine have arguably been the most popular game engine platforms for creating interactive visualizations and experiences based on 3D city data. The increasing variety of data conversion tools and plugins makes the game engines an attractive and flexible data integration platform for generating 'digital twin'-style applications. Game engines have been widely used for photorealistic 3D visualizations (see Figure 3 for an example), typically covering culturally significant sites, planned future city areas and construction projects containing participatory elements for citizens.



Figure 3: A photorealistic and highly detailed 3D model of Helsinki in Unreal Engine.

Technical features and aspects to consider

When creating 3D city model applications, the developer should consider the technical features and capabilities of the platform according to the application. These can be related to the support for 3D geoinformation (i.e. geographical coordinate systems and APIs), visualization capabilities (e.g. real-time rendering, support for VR/AR), flexibility of use (e.g. web use, openness, the level of interactivity), data integration and management (e.g. integration of semantics, sensor data or other meaningful data types) and support for 3D data assets (e.g. mesh models, point clouds, CityGML, BIM models, etc.). Some of these properties are explained and briefly discussed below.

Support for geospatial information

As the application platforms are likely to be utilized with various geospatial datasets, their ability to support these is often quite critical. This may include aspects such as the support for geographic coordinate systems and support for large coordinate spaces that are required for citywide, national or global datasets. In addition to storage formats like GeoTIFF or CityGML, being able to fetch data via interfaces commonly used to exchange GIS data (e.g. WFS and WMS) can be extremely useful and reduce the need for additional data preparation.

Visualization capabilities

The level of visualization is essential for communicating the desired information to the users. Contemporary game engines and the WebGL technology used in web browsers both rely heavily on real-time computer graphics. Real-time 3D graphics allow developers to create engaging interactive visualizations and experiences such as, for example, by enabling the user to move freely in the scene with dynamic real-time illumination.

Many of the most advanced visualization techniques are driven by and applied in the video games industry. Modern rendering techniques such as physically based rendering (PBR) can increase the photorealism of the scene even further. Additionally, the support for immersive visualizations via VR and AR can be extremely powerful tools for communicating the city environment using a real-life scale and dimensions. With applicable display devices – such as head-mounted displays (HMDs) or mobile devices – VR and AR can be natively applied using game engines or, at least to some degree, on the web (e.g. via the WebXR standard).

Flexibility of use

In the case of an individual 3D application development platform, the flexibility of its use consists of several components:

- The ability to deploy the application on a web browser increases its accessibility and removes the need for separate installations and plugins. Needless to say, this is a significant advantage for many of the browser-based platforms.
- 3D city model applications can become powerful tools for communication and information dissemination. Building on this, the ability for multiple users to simultaneously add comments and interact can be extremely useful, especially in participatory use cases. This approach is most pronounced in real-time multiplayer applications and is typically realized in game engines.
- The licensing of the software and its 'openness' also play an important role in the flexibility of use. Both open-data and open-source movements are strongly present in the city modelling scene. This is interesting for application development, as a growing number of platforms for distributing geodata and developing applications are available as open-source platforms. For example, Unreal Engine is free to use and even to monetize under a gross revenue of US\$1 million.

Data integration and management

Data integration is at the core of a meaningful application and can be seen as a key enabler for urban data analysis and the creation of digital twins. By generating a semantic information model of a city, integrating attribute information (e.g. data related to a building cadastre, city plans, utility networks) or other additional information (e.g. statistics, city services and businesses), the 3D city model can be more than just a visualization of a city environment. Furthermore, an increasing amount of data is being collected using real-time sensors related to traffic, weather and building HVAC systems, for example. 3D city models can act as a platform for integrating and visualizing these data sources.

In addition to containing individual points of dynamic data, cities themselves are dynamic and constantly changing environments. As a result, it can be useful to communicate the changes in the city environment using time series data, for example. This time-dependent data can be historic data, or simulated or planned data describing the future.

In terms of data management, it is possible for 3D city models to include data at various scales and levels of detail. Simply managing the sheer volume of data is a common challenge. It is therefore wise to consider a platform's ability to handle multiple levels of detail and apply optimization methods in storage and loading in order to facilitate big data volumes.

Support for 3D assets

Needless to say, for building applications that utilize 3D data, the support for various 3D data assets used in the urban modelling context is crucial. The different 3D asset types include dense, coloured point clouds (e.g. LAS, E57), large textured mesh models (e.g. OBJ, FBX, glTF), various semantically enriched city information models (e.g. CityGML, CityJSON) or building information models (e.g. IFC), offering a highly detailed view of an individual building. Furthermore, support for flexible data formats that are suitable for several asset types, such as 3D tiles, is becoming increasingly important.



Figure 4: An overview of the capabilities offered by the available virtual globes and game engines for each technical feature type, illustrated for a set of openly available platforms.

Conclusions and likely future developments

Considering the complex and diverse nature of 3D city modelling, there is no single platform to meet all the requirements related to producing, upkeeping, maintaining and utilizing 3D models of city environments. The virtual globes and game engines currently available tend to highlight slightly different technical aspects (see Figure 4). The application development requires flexible platforms that enable the development of accessible, open, visual and interactive applications that respond to user needs. Achieving this flexibility often calls for solutions that are available as open-source solutions or that are at least free to learn.

One of the most interesting developments in the context of application platforms for 3D city models is the progress of browser-based game engines. This holds the promise of not only bringing game engine-like features to browser-based 3D applications, but also of integrating the visualization capabilities, such as VR, with future online GIS applications. The same process of integration can be observed in the increasing use of game engines for professional applications, even resulting in the integration of game engines and virtual globes. Examples of this are already emerging in the software ecosystem.

Technologies and the software landscape are constantly changing with promising outcomes. Studying and advancing the utilization of the available 3D city data is crucial in order to fully exploit the potential benefits provided by the data. Despite the often-stated challenges in data quality, interoperability, standardization and management on both technical and organizational levels, there is an increasing amount of feasible and openly available data that is waiting to be utilized beyond the scope of cities themselves.

Further reading

On development of browser-based interactive 3D applications:

Julin, A.; Jaalama, K.; Virtanen, J. P.; Maksimainen, M.; Kurkela, M.; Hyypä, J.; Hyypä, H. Automated multi-sensor 3D reconstruction for the web. *ISPRS Int. J. Geo-Inf.* 2019, 8(5), 221. <http://dx.doi.org/10.3390/ijgi8050221>

On current platforms for 3D city models:

Julin, A.; Jaalama, K.; Virtanen, J.P.; Pouke, M.; Ylipulli, J.; Vaaja, M.; Hyypä, J.; Hyypä, H. Characterizing 3D City Modeling Projects: Towards a Harmonized Interoperable System. *ISPRS Int. Geo-Inf.* 2018, 7, 55. <https://doi.org/10.3390/ijgi7020055>

On utilizing 3D geospatial data in virtual reality:

Virtanen, J.-P.; Daniel, S.; Turppa, T.; Lingli Z.; Julin, A.; Hyypä, H.; Hyypä, J. Interactive dense point clouds in a game engine. *ISPRS Journal of Photogrammetry and Remote Sensing* 2020, 163(162), 375—389. <https://doi.org/10.1016/j.isprsjprs.2020.03.007>