

Smart database and transmission techniques for fast rendering of large 3D datasets in web clients

Marian de Vries

TU Delft, Faculty of Architecture, section GIS Technology

Abstract:

The direct interactive use of 3D geo-information in web applications still lags behind that of 2D geo-information. There are several reasons for this. While for 2D maps there is the light-weight option to send raster images to the web client (OGC's WMS, Google Maps etc.), for 3D data a comparable easy display alternative does not exist (yet). Hence the handling and visualization of 3D geodata asks for more specialized web clients. In addition, 3D datasets are by their very nature much larger than their 2D counterparts. 3D web clients therefore have to deal with potentially very large data streams.

In this presentation a strategy for fast transfer and rendering of 3D geodata is discussed. It is based on multi-representation and replication using generic database technology, combined with streaming techniques and progressive rendering in the web client.

Core characteristics of the approach are: the amount of data that goes from server to web client is reduced as much as possible; secondly, the runtime transformation from data storage format to displayable format is minimized; and thirdly, use is made of strengths of object-relational databases where complex ('nested') data types can be defined, and processing can be carried out in the database itself.

More specific the solution consists of the following parts:

- Only transfer to the client what is needed for initial visualization; so not all attributes, only the geometry and an identifier attribute for each object;
- Have an efficient transfer encoding. X3D is a good candidate (in two variations, either as X3D or as X3DOM) because it has a more efficient storage structure than the GML 3D geometry types: vertices are only stored once per object, and an index list per object holds the order of the vertices per face;
- Transform the geometry from database format (for Oracle Spatial this is SDO_GEOMETRY) to transfer format already in the database. To accomplish this, the X3D complex types 'IndexedFaceSet' and 'IndexedLineSet' were created in Oracle as part of this research, so that for each object also the X3D representation can be stored in the database;
- Use of database triggers to keep the two geometry attributes of each 3D object synchronized, so that inconsistencies will not occur (instant replication);
- Chunk-wise transmission of data from server to client: this makes it easier to control the incoming data stream, and helps in preventing slowing down or failure of the web client.

Together these methods will lead to faster and more robust web applications for 3D geodata. Other techniques, such as switching between levels-of-detail when zooming, or the use of tiles, can be added and do not conflict with the presented approach.