INTERNATIONAL SYMPOSIUM ON
"MODERN INFORMATION AND GPS TECHNOLOGY – ASPECTS AND IMPLICATIONS OF THEIR APLICATION"
Sofia, 11 - 12 November, 1999

An alternative for a 3D GIS

Siyka Zlatanova
ITC, 7500 AA P.O. Box 6, Enschede, the Netherlands
zlatanova@itc.nl

Summary

An increasing need for a 3D GIS is observed in many areas in the last several years that challenges researchers and vendors to find a solution for maintenance and analysis of 3D data. Advances in the field of software and hardware development warranty the "luxury" to operate with the third dimension. Based on some recent software achievements, the paper presents a client-server approach for a 3D GIS. The approach permits query and analysis of spatial and semantic information by remote users across the Web. The data are hosted in a DBMS according to a specially designed conceptual schema, which allows storage of geometry, attributes, spatial relationships and behaviour of object. The approach is based on established Web mechanisms to control the access to the database and create documents on the fly. The new aspect is the elaborated employment of VRML for visualisation, query and exploration of 3D data. A GUI on the client site provides the users with means (fill-out forms) to specify SQL queries and visualise 3D outcomes in virtual reality environment. A prototype system and a number of examples validate the proposed concepts.

1. Introduction

The city is usually the centre of education, trade, industry, cultural life, governmental activities, etc. It offers a variety of opportunities, which always have been attracted people from the villages (see Eisner et al 1993). The increase of population inevitably leads to the increase of the complexity of tasks, which have to be tackled, and the information, which has to be processed. In many cases, the need for 3D geo-referenced information is extremely high, e.g. large private and public construction works, environmental impact analysis, tourism and shopping information, real estate market, management and preservation of cultural cities.

The commonly established systems dealing with spatial information are 2D GIS's and 3D CAD systems. CAD systems, originally designed to create, edit and display small 3D graphics models, show weaknesses in 3D visualisation of large models. Recent developments (power processors, graphics accelerators and standard hardware-implemented graphics libraries) have opened new horizons for 3D visualisation named virtual reality (VR) techniques. Conventional equipment or sophisticated devices allow complete immersion with the model, simulating actions and observing reactions similar to those in the real world. Nowadays computer technologies are capable to handle, process and display more graphics entities than ever before. Despite the various tools to manipulate data, CAD and VR systems have two conceptual specifics that result in insufficient means for GIS analysis, i.e. 1) most of them are not design to deal with semantic information and 2) spatial analysis is hardly in focus. Some attempts to overcome the first shortcoming are made by CAD and VR systems by establishing links to databases to attach attribute information to the objects (see Bentley 1999, Intergraph 1999, Autodesk 1999). Maintenance of semantic information becomes feasible but in a separated information system, most commonly DBMS.

The generic idea of GIS's is to incorporate geometric and semantic information in one system and support analysis in both domains (see Maguire et al 1991, Aronoff 1995). Current commercial GIS's, however, support 2D topology (ESRI, 1999) and face problems supplying 3D spatial analysis. The existing possibility to create separate vertical layers with explicitly recorded 2D topology only per layer (e.g. ArcCAD, ATKIS, SPRAD) gives some extensions to 3D, yet spatial operations between two layers create problems. 3D-display and real time navigation is still supported only for 2.5D data (e.g. 3D Analyst, ArcView, ESRI).

Moreover, GIS and CAD tend to be conceived as a monolith, stand-alone or Intranet system that, however, does not correspond to the recent business trend to exchange information on a Web electronic market. The Web electronic market is considered by many as a new information revolution and tremendous amounts of money and manpower have been invested in Web-based commerce systems (see Bichler et al 1998). New technologies (HTTP, CORBA, Java) and standards (VRML, HTML) make possible the development of software kernels, which can be extended, linked and aggregated, in component-based systems on the Web. The significance of electronic market is appreciated already by many GIS vendors: although the limited functionality, several extensions for remote access and query of spatial data are already available (see bellow).

The current status to maintain and analyse 3D spatial data thus can be characterised as: a lack of appropriate commercial software and an increasing demand for systems processing and analysing 3D spatial information gathered from urban environments. Some vendors co-ordinate their efforts in order to offer a suitable solution on the base of their own products. Typical example is the integrated software package of ESRI and AutoDesk companies ArcCAD, AutoCAD and ArcVIEW. Although improved capabilities of such hybrid systems, the maintenance of separate databases remain problematic and either comprehensive 3D spatial analysis can not be performed or efficient 3D visualisation cannot be offered.

The design, utilisation and maintenance of a *new* 3D GIS comprises wide spectrum of questions concerning a 3D model, data collection, analysis, manipulation, visualisation and remote access of geo-referenced data (see also Raper et al 1998). In this paper, we present an alternative for a 3D GIS on the Web. The concentration will be on three important aspects of the 3D GIS development: the system architecture for remote access and query, the Graphics User Interface (GUI) and the conceptual model for data organisation. The paper is organised in four sections. First, the system architecture is elaborated, second the approach for visualisation is discussed, third the conceptual schema for organisation of 3D spatial data is presented and finally, some examples of queries are discussed.

2. System architecture

The remote access to 3D spatial information is one of the newest topics of research. The Web has already shown a great potential in improving accessibility to 2D spatial information (raster or vector maps) hosted in different computer systems over the Internet (e.g. http://www.visa.com, http://www.mapquest.com, http://www.mapguide.com, etc). 3D graphics was not transferable across the Web until very recently. The new Web standard Virtual Reality Modelling Language (VRML) has opened up the ability to distribute and navigate in 3D virtual worlds. Many virtual towns, biological and geological structures, etc., become possible to explore on the Web in a relatively short time. Most of the models, however, are uploaded on the servers as Web documents. The utilisation of VRML as a front-end visualisation engine to a database system is on a very early stage. The research on spatial query and 3D visualisation in VRML has resulted in a few prototype systems (see Coors et al 1998, Lindenbeck et al 1998). The design criteria, however, is more visualisation rather than spatial analysis oriented.

The system architecture used in our approach is a server-oriented, i.e. the Common Gateway Interface (CGI) mechanism is utilised to access remote information. The overall structure of the system comprises a Web browser with a VR plug-in on the client site and a Web server and a database system on the server site (see Figure 1). The system relies on HTML documents for compositions of queries and visualisation of other data then 3D graphics. VRML documents are intended for pointing (selection) of objects to query and visualisation of spatial analysis. CGI scripts establish the protocol between the client and the server. They are responsible for the assembling of SQL queries, the access to the RDBMS and the creation of documents (HTML or VRML) on the fly with respect to the result of the query. Although the principal similarity to the approach presented by Lindenbeck et al 1998, our system intends more sophisticated tasks for CGI scripts, HTML and VRML documents and thus goes a step ahead toward analysis (spatial and semantic) of data.

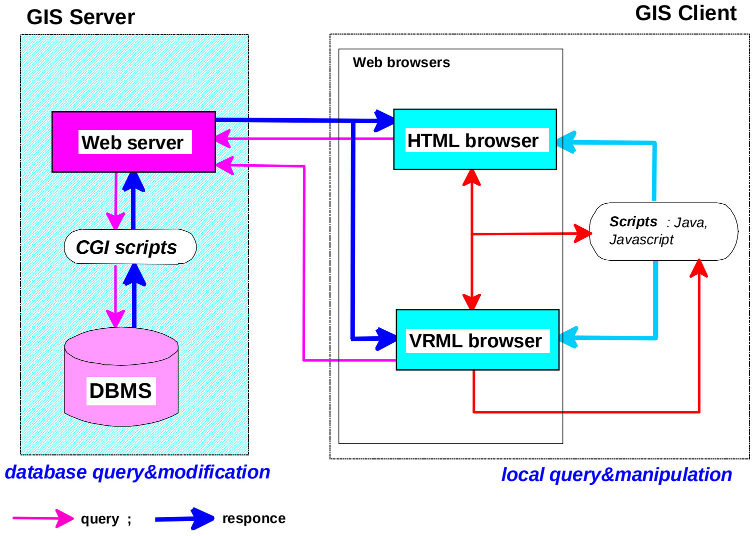

Figure 1: Client-server architecture based on CGI scripts, HTML and VRML

The CGI mechanism with its specific client/server protocol subdivides the process of query, visualisation and modification of data into several different stages:

Query. The request for information per object (a particular building, an owner or a list of co-ordinates of a parcel) can be organised in two steps: first, the object has to be identified and second, the type of the information has to be specified. For example, the simple query "who is the owner of this building" will require: 1) means to point the building and 2) an appropriate interface to specify the needed information. The first step can be formulated either in a VRML or an HTML document. In the HTML document, the query can be specified in a fill-out form (pull-down menu, multiple choice or directly typing SQL queries). This means that the user has to be aware of the identification number (ID) of the object in advance, which often is impossible. The alternative, i.e. pointing the object in the VRML document is much more attractive. Despite the lack of real pointing interface, combinations of sensors and scripts simulate the same operation. The user visually chooses the object of interest as in a standard CAD system or GIS.

Both ways of identification of an object activate a CGI script on the server. The script delivers an initial HTML form, where the user makes further clarifications and sends it back to the server. First, the form parameters are processed, the needed according to the request data are extracted, a document is created and sent back to the client station. Since the CGI client/server architecture is stateless, each new query initiates the same process and results in a completely new document created on the fly. Very often, however, some information of already filled-out forms has to be forwarded to the next query. In this case, the information (some values of particular parameters) has to be memorised. Since the server has no memory for previous connections, parameters are kept on the client station. The intermediate document created contains the needed parameters in hidden for the user fields and passes them back to the server with the new query.

The two-step schema described above is appropriate only to query information (non-spatial and spatial) about a particular object. Many queries and analysis cannot fit in the schema due to impossible clarification of objects in advance. Examples of such queries are "show the highest building in the town", "show all the administrative buildings", "show the common walls", " who are the owners of the buildings along this street". The composition of such queries is rather complex to be organised in VRML documents. Therefore special HTML fill-out forms, where the user types either the necessary SQL statement or other appropriate parameters, have to be created. Some of the examples discussed later focus these queries.

Data visualisation. The information delivered at the client site is displayed either in an HTML document (text, 2D graphics, etc.) or in a VRML document (3D graphics and text). For example, the query "show the way between the hotel and the nearest shop for shoes " will result in a subset of objects (streets and surrounding houses) that can be displayed in a VRML document. Animation may even route the user from the hotel to the shop. The result of a more complex request "show the way between the hotel and the nearest shop for shoes and the prices of the shoes", will be separated into two steps. Fist, the VRML document with the geometry will be displayed. A second user action, e.g. click with the mouse on the shop will care the creation of an HTML document delivering information about prices.

This limitation is related to the CGI mechanism to create documents on the fly. The first line sent by the server is the MIME type of the document. The creation of two documents on the fly is impossible.

Data modification. Using a similar approach, changes in the database on the server can be formulated and executed. In general, changes can be formulated in VRML document, however they do not effect the data on the server. As was specified above, the VR browser is not capable to save the modification. The link between the DBMS and the VRML document in this respect is one-way. For example, if the operation “drag-with the mouse” is provided by the VRML document, the user can move the object "inside the browser" but he/she cannot send the new designed position back to the server. The way out of this limitation is an execution of CGI scripts, which will deliver an HTML fill-out form to describe the changes.

Local query and manipulation (exploration) means temporary changes of the data on the client station. The local query can be understand as an exploration of some properties, which are described in the VRML or HTML document, but can be activated and/or visualised only by user action. For example, one may what to compare several architectural plans for a reconstruction of existing facades. The existing facades will be stored in the database and the user will operate with several new views of the facades available as a separate image files on the server. Another typical example is the design of vegetation and evaluation of its future development. Different types and sizes of trees can be prepared and send to the client and the user can compare and evaluate the overall effect in different views. A way to make such temporary design of geometric information is provided by the VRML, i.e. the suit of *sensors*, *interpolators*, and embedded *scripts*. The interpolators and scripts run only on the subset of data at this particular moment at the client station and, consequently, do not influence the information on the server neither other clients.

3. 3D visualisation in VRML

Advances in the area of computer graphics have made visual media a major ingredient of the current interface and it is likely that graphics will play a dominant role in the communication and interaction with computers in the future. 3D visualisation within 3D GIS requires a number of specific issues to be investigated, e.g. appropriate means to visualise 3D spatial analysis, tools for an effortless exploration and real-time navigation through large models and texturing of the geometry. Observations on the demand for 3D City models (see Gruber et al 1994, Leberl et al 1994, Gruber e al 1995, Razinger et al 1995) show user preferences to photo true texturing, due to improved performance of detail and better orientation in the model. Trading photo realistic texture rises new topics for research, i.e. collection (methods, automation), storage (original images vs. separate pieces), mapping onto the "geometry". Specific functions of objects modelled in VR systems and referred to as behaviours, gain an increased popularity as tools for walkthrough the model, exploration of particular phenomena and improving of the cognitive perception (see Kraak 1998, Raper et al 1998).

In our approach, we pursue a maximal utilisation of 3D graphics and VR techniques to query and explore 3D models. Since the approach relies on Virtual Reality Modelling Language (VRML), a short introduction to its characteristics will be given as well as possibilities to describe realistic urban models will be discussed.

The history of VRML has started in 1994 with the first attempts of leading vendors to establish a Web standard for 3D graphics. The language has passed several stages, i.e. VRML 1.0, VRML 2.0 VRML97, before it was endorsed for an Internet standard in December’97. The first edition, VRML 1.0 was capable to describe static 3D models and link them with VRML and HTML documents. The second version added interaction, dynamics, scripting and multimedia interface (Web3D Consortium Inc., 1999).

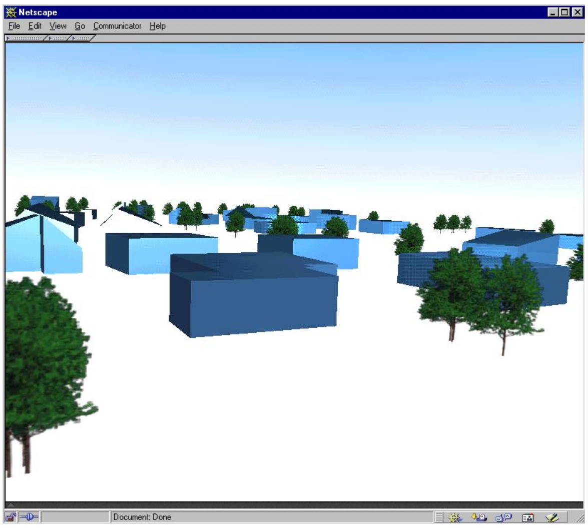

Figure 2: Rozenburg: buildings and trees (represented by *billboards* mapped with an image of a tree)

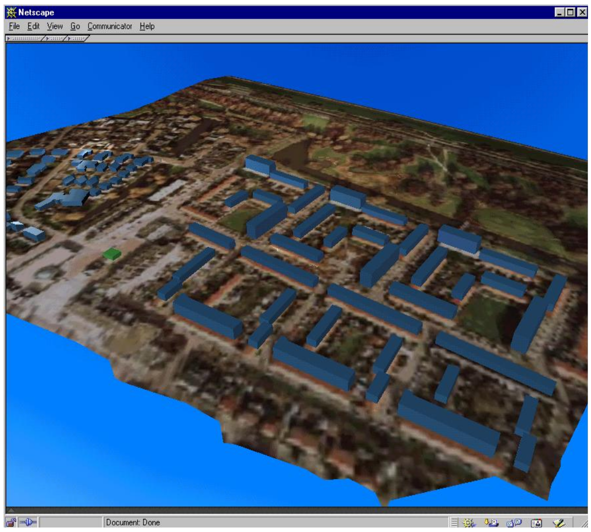

Figure 3: Rozenburg: buildings and terrain draped with a rectified image

Designed for a standard Web protocol, it enables hyperlinks to other types of documents supported by the Web, e.g. movie, sound, HTML, VRML, and scripts. It is capable to represent geometry, texture, materials and lighting, i.e. all the components needed for the rendered scene. With respect to visualisation of real objects, VRML provides sufficient means to achieve high realism. VRML supports two geometric representations: bounding faces and predefined primitives. Between these two representations, our approach gives preferences to face description due to three major reasons. First, the GIS models maintaining topology are usually based on boundary representations, i.e. they operate with faces, points, arcs, etc. Second, most of the 3D models are re-constructed from surface measurements that are difficult to be combined with simple shapes (cone, sphere, cone). The third reason refers the texturing of the geometry. VRML supports the two mechanisms to attach image to geometry, i.e. *texture mapping* and *texture draping*. The more precise adjustment, (i.e. *texture mapping*), is permitted only for individual faces (see Figure 4). This is to say that there is not a way to map, for example, six different images onto the six sides of a predefined shape cube. The operation in this case is texture draping, i.e. the image covers the entire shape according to a rule. Texture draping is an appropriate technique for large surfaces, represented by a set of faces, e.g. terrain (see Figure 3). Often, real objects used to be represented as lines and points in the GIS model. Although VRML support descriptions of lines and points, utilisation of predefined primitives is recommendable. For example, lines and points can be displayed as tiny cylinders and little spheres or symbols (see Figure 2). The substitution usually increases the readability of the VRML world.

The dynamics supported by the language ranges from techniques to play complex animation to means to detect user actions and perform consequent events. The ability to represent real worlds and their dynamics places VRML among the second-level modelling languages (such as Open Inventor). Since the level of realism and dynamics is very high, the 3D VRML models are named *worlds* or *virtual worlds*. Every world can be almost an unlimited composition of smaller worlds distributed on different servers all over the Web (see VRML specifications). All these features make VRML an attractive alternative for a front-end interface to 3D GIS data on the Web.

Once designed in VRML, the model can be visualised in a VR browser. The potentiality of VRML and VR browsers to manage dynamics and interact with the model is still underestimated. The understanding about the VRML and VR browsers is mostly as a system for visualisation of 3D graphics on the Web allowing real-time navigation. The specific manner to provide dynamics and maintain interaction contributes to this opinion. It is important to realise that the dynamics introduced has to be described in advance. If one wants to be able to click on a building, a special operator (*sensor*) has to be attached to this building in the VRML document. If one wants to have animation of a walk on a street, the route and the speed of walking have to be specified in VRML. The VR browser provides the freedom to move inside the world while the particular dynamics runs. These features of the VRML concept are further explored for an experimental GUI for 3D GIS.

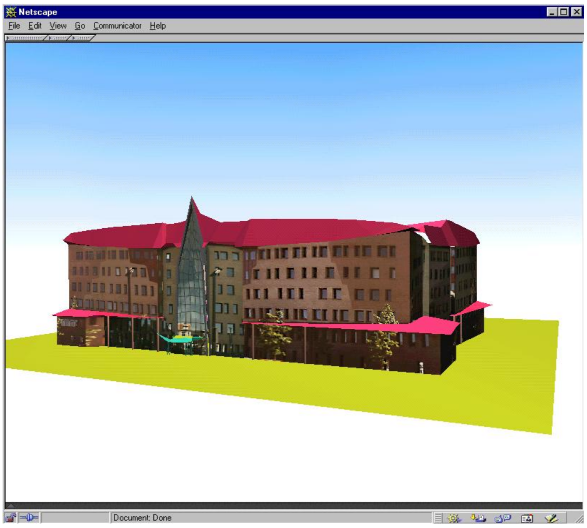

Figure 4: Enschede: the ITC building textured with real photo images

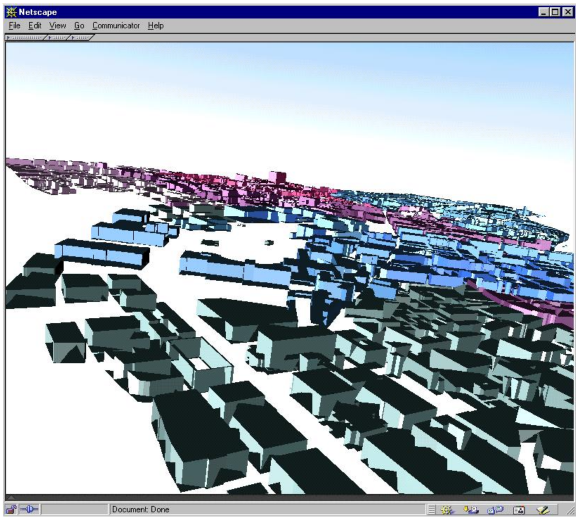

Figure 5: Vienna: 1600 buildings (1/3 of the town)

4. 3D GIS conceptual model

The common understanding is that conceptual model (and its corresponding logical model) is the key element in a 3D GIS. The conceptual 3D model integrates information about 3D geometry, semantics and 3D spatial relationships (3D topology). The conceptual model provides methods for the description of real objects and spatial relationships between them. The design of a conceptual model is a subject of intensive investigations and several 3D models have already been reported (see Brisson 1990, Molenaar 1992, van Oosterom et al 1994, Pigot 1995, Pilouk 1996, Kofler 1998). Since specialists with different background conducts the research the emphasis varies. The research in 3D computer graphics is toward algorithms and structures for processing of large 3D data in real time, photo-realistic visualisation and extended VR tools for interaction with the model. The research in the GIS community is directed to the development of models for maintenance of 3D topology. Little evidences of investigation on both 3D spatial analysis and visualisation are discovered (e.g. Coors et al 1998).

Molenaar 1989 introduces a concept for a 2D GIS named "Formal Data Structure", which follows an integrated approach for describing geometry, semantic and spatial relationships in one spatial model. Later it is extended to comprise 3D information and texture in several successor models. The model is one of the mostly studied (see Rikkers et al 1993, Bric 1994, Pilouk et al 1994, Pilouk 1996, Baucic 1995, Kofuniyi 1995). The investigations in 2D and 3D spatial analysis have shown promising results, which motivated the initial consideration of one of the variants, i.e. 3D FDS (see Figure 6). Experiments with 3D FDS and its suitability as a conceptual model for our approach are discussed in Zlatanova and Tempfli 1998.

Here, we will mention shortly the most important conclusions. 3D FDS supplies sufficient data for rendering, it can be easily extended to accommodate data about behaviour and geometric attributes, however the time for creation of VRML documents is rather long. The following conceptual characteristics influence the time for documents delivery:

- lack of explicit boundary information per objects, i.e. an additional algorithm to collect all the faces per object is required;
- explicit storage of relationships *arc on face* and *node on face* and the implicit description of holes, which create rendering pitfalls;
- maintenance of texture per face, which imposes the limitation to texture a face with only one image.

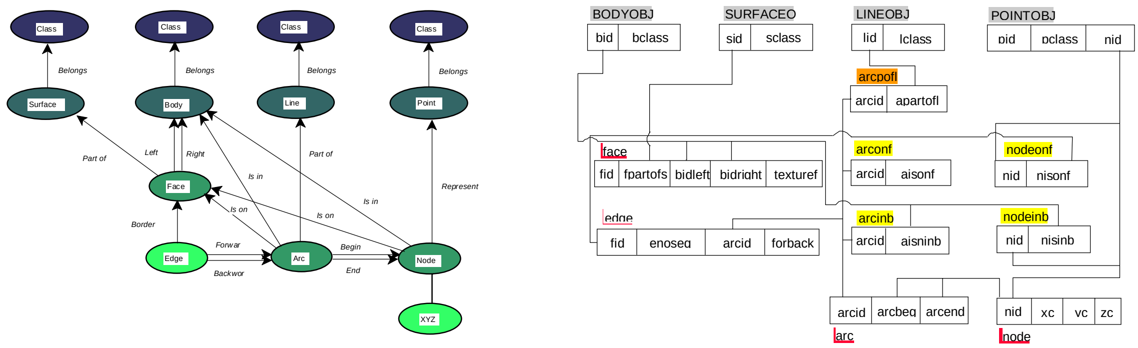

Figure 6: 3D FDS - conceptual and relational schemas

Analysis of the disadvantages of 3D FDS for visualisation, limitations of rendering engines (OpenGL, DirectX) utilised by the VR browsers, as well as the data organisation in VRML has resulted in a new Simplified Spatial Model (SSM). The relational implementation of the model, named Simplified Spatial Schema (SSS) is shown on Figure 7. According to the schema, four relational tables can represent each geometric object, i.e. point, line, surface and body. The _D table contains description of geometry, the _A table contains information about radiometric parameters (i.e. colour), the _B table contains behaviour parameters and the _T table intends semantic information. Note that SSS is an elaboration only in the geometric domain. Semantic characteristics of objects are only indicated in the _T tables with only one field (tema). The behaviour of objects is represented by two fields, i.e. *event initiator* (e.g. "click with the mouse") and *event response* (e.g. "start animation"). *Nodes* and *faces* form the four geometric objects. The introduction of SSS is reported in Zlatanova and Gruber 1998.

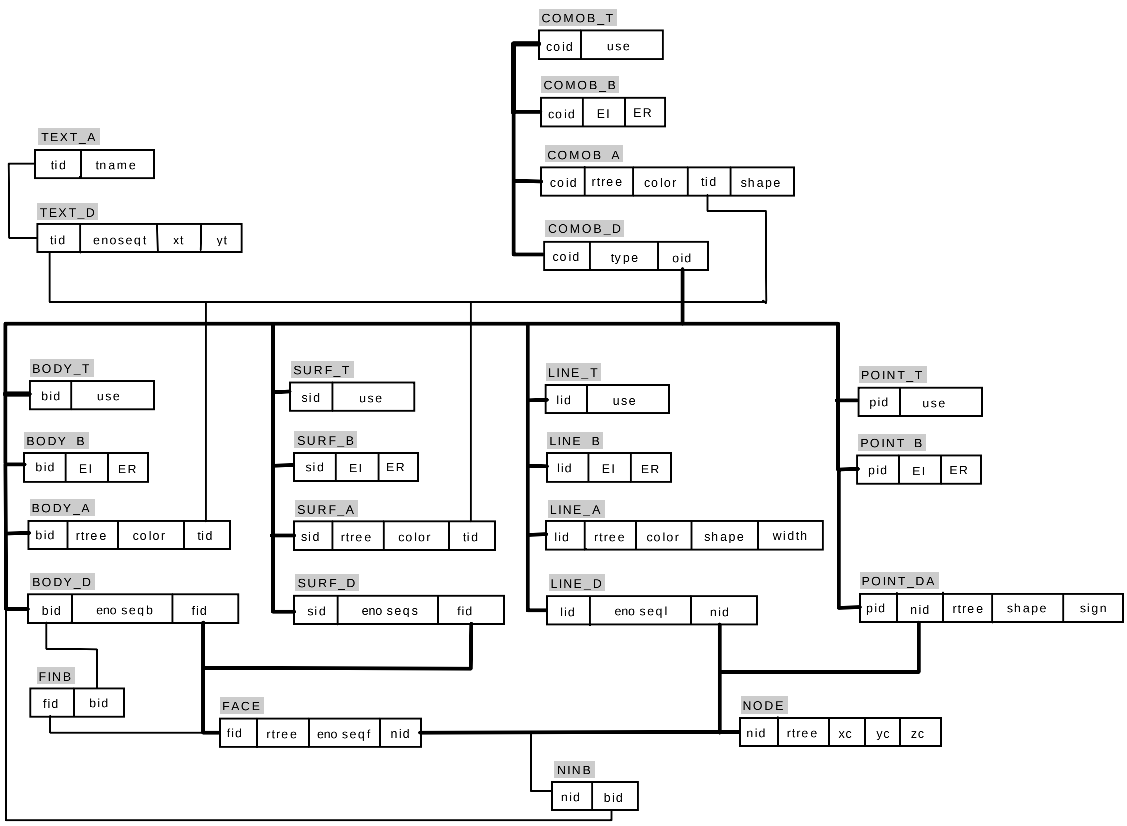

Figure 7: Simplified spatial schema (SSS)

SSS has a number of new features, which preserve the potentiality to maintain 3D topology and at the same time facilitate the visualisation process needed for our approach. Some of these features are listed bellow:

- parameters describing behaviours of objects are introduces (e.g. a door can be open)
- parameters describing radiometric properties of objects are included (e.g. colour, texture)
- the information about *texture* is maintained per surface and body objects, which simplifies the storage of images
- *arcs* are not maintained, which benefits the traversing time and the storage space
- the geometric object *body* is explicitly described by *faces*, which facilitates VRML composition

- sequence of the *nodes* in a *line* is indicated, which allows beginning of a line to be specified
- *arc-on-face*, *arc-in-body* and *node-on-face* relations do not exist, but a new relation *face-in-body* is introduced, which ensured the correct rendering (the arc is incorporated in the face)
- composite objects are introduced, which extend the potentiality to manipulate objects.

5. Prototype system

Besides validation of the concepts presented (the system architecture and the conceptual model), the prototype system aimed at a simple, low cost, easy for implementation solution. A study on user requirements summarised in Zlatanova and Bandrova 1998 has revealed needs for 3D data which, however hardly can be satisfied due to the rather high cost of the third dimension concerning each phase: data acquisition, maintenance, analysis and visualisation. The prototype system had to investigate whether the user could be supplied with reliable non-commercial (or low cost) software and moderate hardware to complete the variety of tasks.

Apache is the Web server selected for the prototype system. The Apache server is a freely available server written by a non-profit team of developers, i.e. the Apache group (see The Apache Group, 1999). Officially released as Apache in April 1995, the daemon was already the most popular one on the Web based on HTTP protocol. Since that time it gained a lot of popularity with its stable work, many advanced features, a relative easy set up (see Stein 1997). Apache works under many operation systems (Windows, UNIX, Linux), on microcomputers and workstations, does not require large disk space (1.5Mb). The processor and memory requirements are unpretentious (it can be installed on 486DX with 16Mb RAM). The server has already been quite a long time in use and most of the bugs are already fixed. All these considerations influenced the election of the Apache server.

MySQL is client-server DBMS supporting SQL (see T.c.X. DataKonsultAB, 1999). MySQL consists of a server, client programs and libraries. The freeware server is launched for the first time in 1996 with the idea to deal with very large databases, which no vendor can provide. Although the limited functionality (hosted SELECTs and views are still missing), the system has a lot of attractive characteristics: available for almost all kinds of platforms, variety of API's (C, C++, Perl, Python, Java, TCL), very fast JOINs, mixture of tables from different databases in the same query, very fast B-tree disk tables with index compression, in-memory hash tables which are used as temporary tables, etc. Several benchmarks have showed significantly better speed performance than many RDBMS. The database lacks of GUI (e.g. "query by example"), which however was not needed for our system architecture.

Perl language is the favourite for CGI scripting. Developed originally as a Unix language, it is now available for most of the computer platforms and can be freely downloaded from Comprehensive Perl Archive Network (CPAN, 1999). Although the CGI scripts can be written in any language, usually, preferences are given to interpreter languages when command-line-based operations has to be performed. A typical example of such programs is the submission of SQL queries to the database. Another crucial advantage of the language is the large number of freeware CGI scripts, libraries and API's, which save program efforts. Two of them, i.e. CGI.pm and DBI.pm to create fill-out forms and access MySQL database were used for the implementation of the system. More information about them can be found in Stein, 1998 and Wiedmann, 1998.

The Web server and the RDBMS are installed under LINUX on Pentuim 133 MHz, 96 Mb RAM. The experiments were conducted from a client station equipped with *Netscape* and *Cosmo player*, running under Windows 95. The understanding was that an acceptable performance on such a configuration with test data will be an encouraging indication for acceptable performance of large data sets on a better hardware configuration. The first test site, i.e. the central part of Enschede, contains photo-textured buildings DTM, trees, lampposts, streets and parking lots (see Figure 8). The model is relatively small but contains most of the tables according to the conceptual schema. The second test site, i.e. Vienna, consists of only buildings (see Figure 5). In contract to the first model, the second one is relatively simple but with a size compatible to size expected for urban models. The buildings have an arbitrary number of walls and have no texture. Whilst he first data set is used mostly to illustrate the GUI and suggested manner to compose queries, extract data and perform results, the second one was exploited for speed performance of the system (reported in Zlatanova 1999). The prototype systems as well as all the examples are available on **http://barley.itc.nl**.

Example1: SELECT queries

As it was discussed before the CGI script not only executes the spatial query but also represents the result in an appropriate VRML file. The interface in this case is based on a two sections in a framed HTML document (see Figure 9): the left part is reserved for typing SELECT statements and right part is used to display either HTML or VRML documents. Since the syntax of the VRML requires a structuring of the geometric data different than the conceptual schema, the SQL statement has to ensure sufficient data and an efficient ordering for the VRML creation. Therefore the data and the order required are displayed in the fill-out form. The form correctly filled is send to the server

(Submit button 1) and an HTML document is created as a first document (on the left side of the frame). On the basis of the result obtained, the user decides whether to continue with VRML creation. The intermediate step can be avoided with a control over 1) the fields in the form and 2) the data extracted from the RDBMS. Such control, however, will restrict the form to creation of only VRML documents and therefore is not performed. Figure 9 shows a snapshot of the Web browser after VRML visualisation.

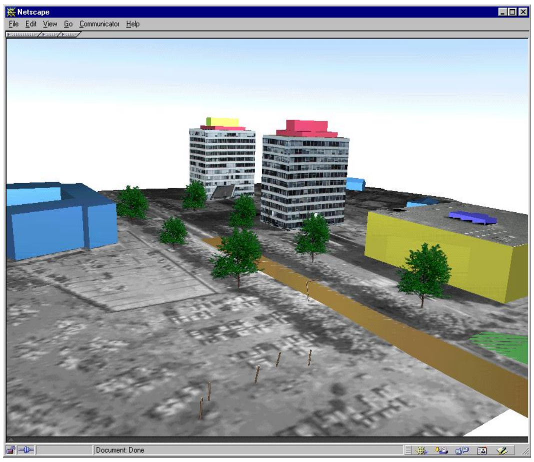

Figure 8: Enscehde: the experimental model

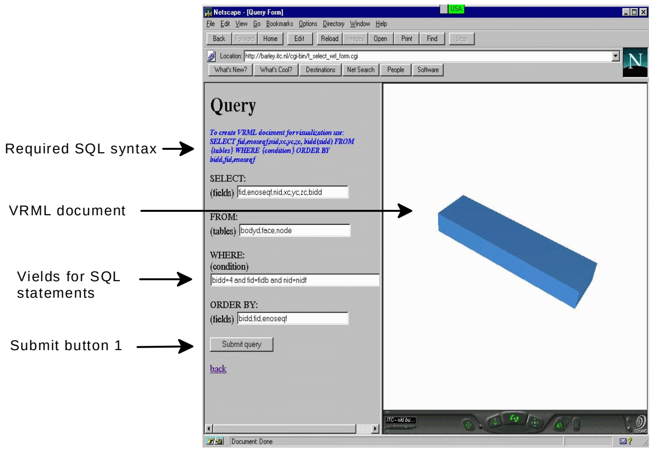

Figure 9: SELECT and visualise in VRML document

The next example deals with queries and analysis which cannot be expressed by one SQL statement, e.g. "find the nearest building", "find shortest way", "visibility check"", etc. In such cases, the development of series of specialised HTML fill-out forms is inevitable. All the forms designed to treat a specific query can be offered to the user in an individual HTML document.

Example2: Common faces

Two fill-out forms are created to illustrate a query of neighbourhood relationships, i.e. "find common nodes" and "find common faces". The user has to type the ID of the objects, i.e. the user has to be aware of the objects ID. As was discussed above, the ID can be provided with a VRML document. An option to analyse more than two objects is offered, as well. An asterisk, instead of ID, extends the search among all the objects in the database. For example, the query "visualise all the common walls" is realised using asterisks in both ID fields (see Figure 10). The result of the query is visualised in the right section of the form. In the example, two bodies have a common face (invisible). CGI scripts processing such queries acts has an extended function, i.e. a completion of a query, which cannot be obtained using only SQL statements.

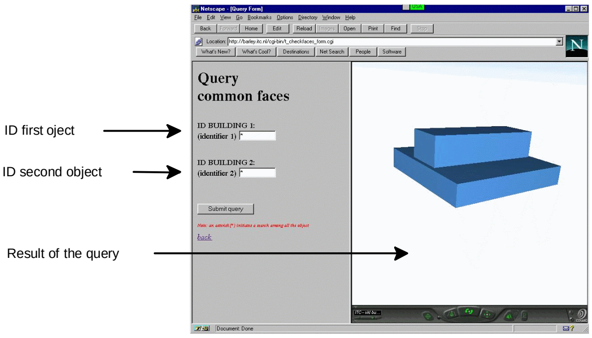

Figure 10: Query of common faces

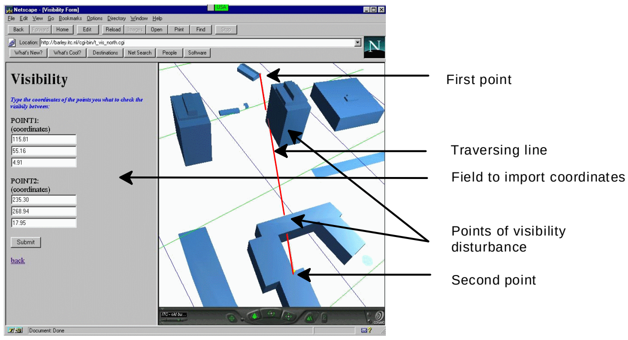

Figure 11: Visibility control

Example3: Visibility check

The last example demonstrates the potential of VRML to facilitate and simplify complex analysis. The field of vision (or line of vision) is an important information for telecommunication, geodetic, military etc. companies and organisations. For example, a mobile telephone company could be interested in the verification of the actual position of a transmitter. This can be translated to a query "check the visibility between the position of the transmitter and the roof of that building" or "show the range of the transmitter". Two 3D co-ordinates or ID of objects or co-ordinates and range of the transmission (e.g. represented as a cone) may be the required input for such queries. The outcome of the query must be a set of objects, which disturb the view that, in fact, will force complex 3D intersection algorithms. The answer to this question, however, can be obtained by a visual inspection of the actual path of a traversing line between the two points. A form to illustrate the idea is shown on Figure 11. The user has to import co-ordinates of two points, and as a result he/she gets a VRML document with a subset of the model surrounding the points of interest. A line trough the points traverses the direction. The user can navigate around the disturbing object, inspect and evaluate the situation. Appropriate *sensors* (elaborated variant of *Touchsensor*) attached to the objects will provide identification information, e.g. the ID of the objects or the name of the owner (company or private person). More examples can be found in Zlatanova 1999 or on the Web site.

6. Conclusions

The paper presented system architecture of a 3D GIS on the Web and a corresponding appropriate conceptual schema for database organisation.

The approach has several important advantages compare to the attempts for extensions of commercial 2D GIS and CAD systems or the development of a 3D GIS from scratch. First, the system is built on components and mechanisms commonly used and available on the Web. This implies that the approach can be implemented in short terms with minimum efforts. The actual implementation work is related to the development of suitable CGI scripts Java applets or ECMAscripts. In fact, the software developer is released from the data management and rendering tasks, which are responsibilities of respectively DBMS and browsers. Second, the combination of HTML, VRML and other types of transferable on the Web documents (movies, images, sounds, etc.) extend the functionality of the information system toward an utilisation of different data formats for data organisation. Moreover the data can be distributed on different servers all over the Web. The basic principles of the Web are practically spread over the 3D GIS. This is a feature, which hardly can be observed in a single GIS or CAD system. Third, the utilisation of VRML as a front-end engine ensures 3D visualisation of spatial analysis and real-time navigation through the model. Fourth, such an approach does not require any specific software on the client site. Web and VR browsers are freely available on the Web. Fifth, the approach assumes a database organisation according to unified conceptual schema of all the information of interest per real object. Although discussed for spatial objects, the conceptual schema can be easily extended to comprise non-spatial objects. The incorporation of spatial and non-spatial objects will enable complex interrelated analysis between them.

The work on the approach has revealed the critical role of the structuring of data on database level. In this respect, a conceptual schema, which preserves the traditional functionality of GIS systems and extends it to 3D, ensures a fast traversal of the relational tables and facilitates the exploration of the model by storage of object's behaviours was introduced and successfully experimented. The conceptual schema takes into consideration both topologic and visualisation requirements.

Despite of the promising results, the approach has a specific limitation, i.e. CGI scripting restricts the functionality on the client site. Currently, editing operations can be performed by several sequential connections to the server. A possible way out is the utilisation of Java applets instead of CGI scripts. Further investigations are necessary to specify the types of behaviours, which can be maintained in the database. In case of multiple behaviour per object, the conceptual model needs further elaboration.

References

Aronoff, S., 1995, Geographic Information Systems: A management perspective, WDL publications, Ottava, Canada

Autodesk, 1999, http://www.autodesk.com/

Baucic M, 1995, Spatial data for electric utilities, MSc-thesis, ITC, the Netherlands

Bentley, 1999, http://www.bentley.com/

Bichler, M., A. Segev and J.L. Zhao, 1998, Component-based E-commerce: Assessment of current practice and future directions, in: ACM SIGMOD record, Vol. 27, No.4, 8 p.

Bric, V., 1994, 3D Vector data structure and modelling of simple objects in GIS, MSc-thesis, ITC, the Netherlands

Coors, V. and V. Jung, 1998, Using VRML as an interface to the 3D data warehouse, in: Proceedings of VRML'98, New York

CPAN, 1999, The Perl language, http://www.perl.com/pace/pub/

Brisson, E., 1990, Representation of d-dimensional geometric objects, PhD Thesis, University of Washington, USA

Doyle, S., M. Dodge and A. Smith, 1998, The potential of Web-based mapping and virtual reality technologies for modelling urban environments, in: Computers, Environment and Urban Systems, Vol. 22, No. 2, pp. 137-155

Eisner, S. and S.A. Eisner, 1993, The Urban Pattern, New York, Van Nostrand Reihold

ESRI, 1999, http://www.ersi.com/

Gruber, M., M. Pasko and F. Leberl, 1995, Geometric versus texture detail in 3D models of real world buildings, in: Automatic extraction of man-made objects from aerial and space images, Birkhauser Verlag, Basel, pp. 189-198

Gruber, M. and E. Wilmersdorf, 1997, Urban data management–a modern approach, in: Computers, environment and urban systems, Vol. 21, No.2, pp. 147-158
Intergraph, 1999, http://www.intergraph.com/
Isdale, J.   , 1998, What is Virtual Reality, http://isdale.com/jerry/VR/WhatIsVR/frames/WhatIsVR.html
Kofler, M., H. Rehatschek and M. Gruber, 1996, A Database for a 3D GIS for urban environments supporting photo-realistic visualisation, in: ISPRS, Commission III, Vienna, Austria
Koller, D., P. Lindstrom, W. Ribarsky, L. Hodges, N. Faust and G. Turner, 1995, Virtual GIS: A real-time 3D Geographic Information System, Technical report 95-14, Georgia Institute of Technology
Kraak, M.J., 1998, The cartographic visualisation process: from presentation to exploration, in: Cartographic Journal 35, 1, pp 11-16.
Kufoniyi, O.1995, Spatial coincidence modelling automated database updating and data consistency in vector GIS, PhD-thesis, ITC, the Netherlands
Leberl F., M Gruber, P. Uray, F. Madritsch, 1994, Trade-offs in the reconstruction and rendering of 3-D objects, in: Mustererkennung'94, Vienna, Austria
Lindenbeck, Ch. and H. Ulmer, 1998, Geology meets virtual reality: VRML visualisation server applications, in: WSCG'98, Vol. III, Plzen, Czech Republic, pp. 402-408
Maguire, D.J., M.F. Goodchild, and D.W. Rhind, 1991, Geographical Information Systems, Vol. 1, Harold, Longman Scientific&Technical
Molenaar, M., 1989, Single valued vector maps; a concept in Geographic Information Systems, in: GIS, Vol. 2, No. 1, pp. 18-27
Molenaar, M., 1992, A topology for 3D vector maps, ITC Journal 1, pp. 25-33
van Oosterom. P., W. Vertegaal, M.van Hekken and T. Vijlbrief, 1994, Integrated 3D modelling within a GIS, in: Advanced Geographic Data Modelling: spatial data modelling and query languages for 2D and 3D applications, Netherlands Geodetic Commission, No.40, pp. 80-95
Pigot, S., 1995, A topological model for a 3-dimensional Spatial Information System, PhD Thesis, University of Tasmania, Australia
Pilouk, M., 1996, Integrated modelling for 3D GIS, PhD-thesis, ITC, the Netherlands
Pilouk, M. and K.Tempfli, 1994, Integrating DTM and GIS using a relational data structure, in: Proceedings of the eight annual symposium on Geographic Information Systems (GIS'94), Vol. 1, Vancouver, Canada, pp. 163-169
Rikkers R., M. Molenaar and J. Stuiver, 1993, A query oriented implementation of a 3D topologic data structure, in: EGIS'93 Vol.2, Genoa, Italy, pp. 1411-1420
Raper, J.F., T. McCarthy and D. Unwin, 1998, Multi dimensional Virtual Reality Geographic Information System, (VRGIS): Research guidelines, in: Proceedings GISRUK 98, Edinburg, UK, 6p.
Razinger, M. and G. Gleixner, 1995, Changing the city: data sets and applications for 3D urban planning, in: GIS Europe, March, pp. 28-30
Stein, L.D, 1997, How to set up and maintain Web Site, Addison Wesley Longman, Inc.
The Apache Group, 1999, Apache HTTP server project http://www.apache.org/
Tempfli, K. and M. Pilouk, 1996, Practical Photogrammetry for 3D GIS, in: ISPRS, Vol. XXXI, Part B4, Vienna, pp. 859-867
Tempfli, K., 1998, 3D topographic mapping for urban GIS, ITC Journal 3/4, pp. 181-190
T.c.X. DataKonsultAB, 1999, MySQL, http://www.tcx.se/index.html
VRML specifications, 1997, http://www.web3d.org/technicalinfo/specifications/vrml97/index.htm
Watt, A., 1993, 3D Computer Graphics, Wokingham, etc: Addison-Wesley
Wiedmann, J., 1998, Msql, Mysql modules, http://www.arcana.co.uk/cgi/dbi/moduledump?module=Msql-Mysql-modules
Zlatanova, S, 1999, VRML for 3D GIS, in: Proceedings of the 15th SCCG, 28 April-1May, Budmerice, Slovakia, pp. 74-82
Zlatanova, S. and T. Bandrova, 1998, User requirements for the third dimensionality, in: E-mail seminar of Cartography 1998: Maps of the future, Vol. 1, Sofia, pp. 61-72
Zlatanova, S. and M. Gruber, 1998, 3D GIS on the Web, in: ISPRS, Com. IV, 7-10 September, Stuttgart, Germany, pp. 691-699
Zlatanova, S and K. Tempfli, 1998, Data structuring and visualisation of 3D urban data, in: Proceedings of AGILE conference, 23-25 April, Enschede, The Netherlands, CD ROM, 12 p.