Web 3D GIS for Urban Environments

Alias Abdul Rahman
Universiti Teknologi Malaysia

Sisi Zlatanova
UNSW Sydney

Web 3D GIS for Urban Environments

Georg Held[1], Alias Abdul-Rahman[1], Siyka Zlatanova[2]

[1]Department of Geoinformatics and
Institute for Geospatial Science and Technology
Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
qmann@gmx.net (Georg Held)
alias@fksg.utm.my

[2]GIS Technology, OTB
Delft University of Technology, 2600 GA Delft, The Netherlands
s.zlatanova@otb.tudelft.nl

Abstract. *Urban environments like city centers are recognized as one of the most complex systems for modeling. Because of their high density of big buildings as well as their complicated processes, there is the strong need for planning and managing properly. Here, the heavy trend towards web-enabled systems can help to improve communication, organization and decision-making in favor. In the field of urban environments, 2D Geographic Information Systems (GIS) have proven to be a very useful tool. However, since real objects are in 3D, GIS should offer sufficient functionality for dealing with the third dimension as well. Therefore, the intention of our research is to integrate 3D GIS in web-orientated environments in order to provide appropriate applications to urban planners.*
The aim of this paper is to introduce the current status of Web 3D GIS and most recent trends and developments. First, we will present technologies linked to web-enabled systems. Here, the field of distributed computing in relation to tasks in the field of Geo-Information is most interesting. Therefore, corresponding standards recommended by e.g. the Open GIS Consortium (OGC) will be discussed, too. After that, the paper focuses on 3D data management and corresponding functionality. The third aspect is 3D visualization for web-environments. Different techniques like VRML as well as the problematic task of a Graphical User Interface (GUI) to access and query data will be clarified. Finally, the paper discusses bottlenecks of Web 3D GIS and proposes new aspects to solve them.

1 Introduction

Recent developments in GIS are showing a general movement towards web-enabled GIS. The gap between Desktop-GIS and Web-GIS is closing. Applications based on network environments have already shown great potential in relation to geo-information. Examples can be online city maps and finding places (respectively routing) between points (MAP 24, 2004). The developments in web-enabled GIS are driven by user requirements and technology developments. But is the third dimension sufficiently exploited by Web-applications?
In general, the need of 3D geo-data is rising more and more. Especially people involved in urban and landscape planning, cadastre, real estate, utility management, geology, tourism, army, etc. are keen on taking advantages of the third dimension. Since real objects are in 3D it is obvious to extend GIS to the third dimension as well. However, the acceptance of 3D applications depends heavily on the profits of these. Therefore, one can say the number of users is increasing by introducing new and additional 3D functionality.
The steadily growth of urban environments worldwide is challenging our society. In order to avoid chaos and confusion, urban scenarios like cities and their complex streams have to be planned well. Therefore, geo-information and corresponding spatial data are able to support planners and their decision makers heavily (Laurini, 2001). Possible fields of application are comprised in Table 1.

Table 1 Possible Fields of Application within Urban Environments for web-based 3D GIS (based on Altmaier and Kolbe, 2003)

Sector	**Description**	**Example**
Event management	Simulation of the event to attract people	Offering the possible 3D view of a certain seat in a stadium
Facility management	Management of big building complexes	Organizing the room availability of a hospital
Navigation support	Car and pedestrian navigation systems	Location-based service displaying the recent position and its environment
Environment	Environmental Topics in Cities: noise characteristics, air flows, emission dispersions, etc.	Visualizing the emission dissemination
Disaster/emergency	Organizing the workflows if there is an emergency	Directing rescue teams through complicated environments with support of real-time data
Supply engineering	Management of supply related tasks	organizing the power network

Table 1 shows many useful scenarios for 3D applications in urban environments. Whereas some of them – e.g. even management – mainly deals with visualization only, there are applications involving spatial analysis. Particularly the topic around disaster and emergency management enjoys great popularity recently.
On the technological side, state-of-the-art computer hardware is already offering a reasonable means to deal with the third dimension such as improved 3D visualization techniques. Among others, there is photo-realistic texturing, advanced lightning or real-time navigation. These in return attract more users and applications. We firmly believe, the Web offers the possibility to make the third dimension widely accessible.
The aim of this paper is to provide an overview about web-oriented 3D GIS. Since we consider system architecture, data management, 3D GIS functionality and visualization (respectively user interaction) critical for Web 3D GIS, we address them in detail. The paper explains needed system components and their importance with respect to the requested Web 3D GIS functionality. System architectures and possible approaches for implementing a web-enabled 3D GIS are reviewed and profoundly explained. Finally directions for further research are outlined.

2 Web 3D GIS

Traditionally, any Geographic Information System is based on the principles of data input, management, analysis and representation. Within a web-enabled environment, these principles are represented by or implemented within the components as shown in Table 2.

Table 2 GIS Principles and their Corresponding Web Component

GIS Principle	**Web Component**
Data Input	Client
Data Management	DBMS possibly extended by a spatial component
Data Analysis	GIS Library on Server
Data Representation	Client/Server

In order to achieve communication between the different components in a web environment, a web server is common. Since the geo-data is a very specific type of data, different standards, e.g. the

OpenGIS Consortium (OGC) specifications are already developed and their utilisation has to be considered (see below). A system composed of these components is called here Web-GIS. It should cover a complete GIS workflow within a Web environment. Fig. 1 shows the general system architecture which is mostly "Client-Server".

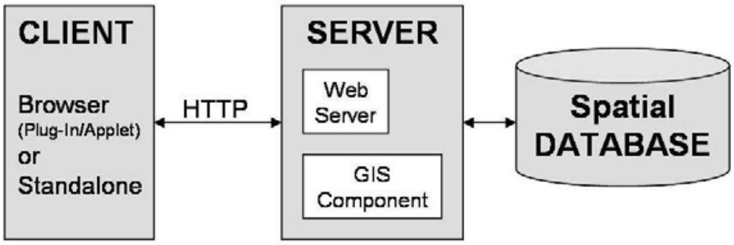

Fig. 1. **Typical Web-GIS Architecture**

Fig. 1 shows the minimum system architecture of Web-GIS. The Client is an application, which can communicate with the Server through a standard web protocol, for example HTTP. This application can either be in form of a web browser or standalone utility. In order to view and interact with GIS data, the browser needs to be extended by using an adequate Plug-In, Java Applet or both. Instead a standalone application can be used. This can be for example any GIS, which is supporting the appropriate protocol to access other computers in computer networks.
The web server is responsible for processing the request from the client and delivering the corresponding response. In Web-GIS architecture, the web server is also communicating with the server-side GIS component. This is adding spatial analysis functionality to the system. Moreover, server-side components are responsible for the connection to the spatial database, such as translating queries into SQL and creating appropriate representations to be forwarded to the server. In reality, GIS components are software libraries, which are offering classes to do spatial analysis on data.
Besides the components, a very critical aspect is the functionality offered by the client- or server-side within Web-GIS. Fig. 2 shows possible distributions of functionality for a client-server system based on the concept of the visualization pipeline (OGC, 2003b).

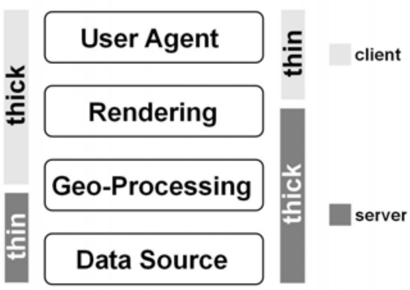

Fig. 2. Thin vs. Thick within Client Server Systems

Fig. 2 shows that a client is considered "thick" or "fat", if the main GIS functionality and the data rendering are client-side hosted. Consequently the server in this specific system would be called "thin". The server is called "thick" if GIS functionality and pre-rendering is hosted server-side. Within this system, the client would be called "thin". Altmaier and Kolbe (2003) exclude rendering for interactive 3D worlds on the server since real-time navigation in static images would not be possible anymore.
However, it is still an interesting question how to find the balance between server and client. Because of the system complexity, required functionality, type of application, data sets, even available funds for implementing one or another solution and user experience, no ordinary rules can be specified. The question has to be answered for each system individually. Regarding the general system architecture, 2D/3D Web-GIS don't have many differences. The setup shown in Fig. 1 can be used for both.

2.1 Differences between 2D and 3D

Traditionally, most GIS spatial operations are very expensive and more complex compared to for example administration numerical and textural type of data. This is especially the case if systems are dealing with the third dimension. Since calculations on 3D geo-information are by far more expensive than those in 2D, developers have to choose very carefully which system component is hosting certain GIS functionality. As stated before there is no general rule. Chapter 5 is discussing concrete implementations and gives answers for individual approaches.

On the operational/functional side, the differences between 2D and 3D calculations are most critical. Typical common operations for 2D- and 3D GIS are accessing attributes or further information on objects, calculating distances and areas, buffering, routing and nearest neighbour analysis. Whereas operations like volume calculations are 3D only. Because 3D information is much more complex and has a higher quantity, the processing is much more complex and therefore is taking by far more time and resources. 3D buffering for example needs much more efforts than the corresponding operation in the second dimension. These operations are done by the GIS component; either server- or client-side. In this respect third party tools or an individual developed component can be used. However, there are very few available third-party tools which are supporting 3D functionality. Therefore the available systems have to be customized. Individual implementations can be realized in any programming language, the server environment is supporting. Here, Java in conjunction with Servlets Technology is one approach (Vries and Stoter, 2003).

2.2 Important Aspects of Web 3D GIS

At the moment, regarding fundamental spatial analysis, database management systems are offering spatial extensions, too. There are spatial extensions for databases like Oracle, PostgreSQL, Informix, DB2, Ingres and most recently MySQL available. Unfortunately these do not support 3D sufficiently (Vries and Stoter, 2003).

In order to provide the development of analysis functionality at a database level, many DBMS are supporting procedural languages as well. Oracle's DBMS for instance offers two possibilities to create individual operations at the database level. First there is PL/SQL a procedural language. Second it has integrated its own Java Virtual Machine in order to process Java classes at the database level. The advantage compared to external spatial analysis will mainly be in terms of a better querying performance. In addition, operators on database levels can be used by anyone who has access to the database. Therefore, basic spatial analysis operations can be reused within other applications (Jansen, 2003). Systems implementing a spatial extension are called integrated systems (Oosterom *et al*, 2002).

Overall, the trend towards GIS in Web environments is still ongoing. Recently, the term of Distributed GIS has been introduced. Here, a GIS will be completely distributed in a computer network. The corresponding functionality, data and certain clients are operating like nodes in an object-oriented application (Peng and Tsou, 2003). However, there are not Distributed GIS for the third dimension available so far.

Furthermore, since geo-data is a very specific type of data, standards have to be considered. Therefore the OGC has developed a wide range of specifications/documents which should be considered for utilization. The base for OGC conform GIS are defined spatial data types and their relationships (Simple- and Abstract Feature Specification). In addition, "implementation specifications" are describing interfaces and rules of exchanging/transferring data between components. In context of web mapping, the Web Map Service Implementation Specification (OGC, 2001) has to be considered. It is defining an interface for requesting maps. The corresponding Web Map Service (WMS) is creating maps of geo-information. It has to support the operations of "GetCapabilities" and "GetMap". The operation "GetFeatureInfo" is optional but necessary for retrieving further information about objects through user interaction. Whereas "GetCapabilities" is returning information about the Web Map Service itself, "GetMap" is returning the map/figure. Since editing/manipulating of data is one GIS principle, the Web Feature Service Implementation Specification (OGC, 2002) is a must as well. Operations of a Web Feature Service (WFS) are insert, update, delete, query and discover data. The data is represented in form of GML, another OGC standard for exchanging geo-information (OGC, 2003a). Both, WMS and WFS, are based on the HTTP protocol for transferring data. Among others, GML3 is including 3D geometry and therefore suitable for Web 3D GIS. Besides, there is an implementation specification regarding 3D terrain scenes (Web Terrain Service). Altmaier and Kolbe realized that there is no specification or standard to describe interactive 3D worlds. Therefore, they

introduced the W3DS portrayal service for 3D spatial data (Altmaier and Kolbe, 2003).
OGC standards or others like the ISO/TC211 are important for the communication between components within complex GIS – especially Web-GIS. Therefore systems can be extended easily by additional components confirming to the standards (Vries and Zlatanova, 2004).
Another critical aspect is the performance of the system. If there is one bottleneck, the whole system will be affected. Therefore, system architects have to think about any – at first stage even unimportant – aspect. First, the base of a system should be state-of-the-art computer hardware and proved applications or environments, e.g. powerful 3D visualization techniques. Because of large data amounts, the data transfer between the components should be reduced to a minimum. Because of low band-width, there can be a critical bottleneck between the client and the server. Streaming techniques, which are allowing the data transfer in parts, are becoming very popular should be in favour. In order to achieve acceptable system performance, spatial analysis has to be done on top of a reasonable concept of storing data. Consequently databases have to largely employed, preferably with maintenance of topology (see Chapter 3). In Relation to 3D, issues can be further 3D object reconstruction and real-time navigation (Stoter and Zlatanova, 2003). In many cases, bottlenecks have to be solved for each system individually.

3 Management of 3D Spatial Data

In order to manage 3D geo-information, at least the use of databases and their managements systems (DBMS) is required. Object-relational modeling is most common since relational databases are not very appropriate for storing spatial data. The object-oriented database approach faces the problem that the general acceptance and knowledge is not available so far (Connolly and Begg, 2002). The field of Geo-information adopts both approaches and comprises them into Object-Relational DBMS (Shekar and Chawla, 2003). As stated in Chapter 2 the additional integration of spatial extensions is compulsory for GIS applications. Furthermore, because operations of 3D functionality are different from 2D, a reasonable concept of data storage is inevitable. Therefore the two aspects of 3D geometry and 3D topology have to be regarded. Geometry is holding the 3D coordinates of objects. In contrast to this, topology is holding their spatial relationships. The OGC is proposing the separation between geometry and topology within databases. The reason for this is to perform certain queries on geometry others on topology (Oosterom *et al*, 2002).
Regarding geometry, there are several DBMS available which already have the ability to handle spatial data types. These are divided in to the geometric primitives of point, line and polygon. The OGC calls them as simple features. However, 3D primitives like polyhedrons are missing and have to be implemented individually. Stoter and Zlatanova showed how to store a polyhedron within Oracle 9i using multiple polygons (Stoter and Zlatanova, 2003).
In contrast to geometry, the topological part is more critical. State-of-the-art DBMS are not offering any support for 3D topology. Shi *et al* (2003) and Zlatanova *et al* (2004) provide a brief overview about developed topological models including additional performance tests. However, recently Oracle announced the integration of topology up to 4D in its database spatial extension of Oracle 10g (Lopez, 2003). Besides, the corresponding OGC specification (complex feature specification) is not finished yet - in terms of implementation specifications for complex features. However, topology is the base for reasonable querying of 3D spatial data. Since there is no unique topological model available topology has to be implemented individually. Van Oosterom *et al* (2002) are giving an overview about available approaches. How to choose an appropriate model for a system is querying, application dependent (Zlatanova *et al*, 2002a). Moreover the technique of visualization is another factor for the question of selecting a topological model. There is no general rule of selecting in favour. Topological models should fulfil tasks like covering all possible relationship and extensibility (Oosterom *et al*, 2002).
Beside the geometry and topology, the spatial querying language for the third dimension is challenging the database community as well. Güting concluded in addition to SQL, a spatial query language has to provide fundamental spatial operations and reasonable ways of representing the result (Güting, 1994). Here, 3D operators on top of an ingenious data model are not available so far. For representing the result, tables are not appropriate. A standard-based way in order to illustrate the query result can be found in the GML3 standard.
Furthermore, spatial indexing is one main key to improve querying performance on geometric data. Several different indexing methods are common while mainly R-tree, Quad-tree and P-tree are used.

Furthermore, indexes are often used in conjunction with LOD implementations (Coors, 2003; Kofler, 1998).

4 GUI for 3D Visualization and Editing on the Web

4.1 Basic Concepts

In order to interact and communicate with information, a Graphical User Interface (GUI) has to be designed and created. Because geographic information is usually very complex, this task is difficult to achieve. Moreover, the user interface is most critical due to the fact that this is the "main gate" to the application. If a GUI is implemented poorly, an application will not be accepted by critical users. In contrast to user interaction in 2D, a GUI for the third dimension is different (Cöltekin, 2002).
To develop a GUI for 3D Visualization, different aspects are important. First of all, the virtual world has to be sufficient. To do so, core features of creating a 3D world are needed. This means appropriate modeling of physical objects, lighting and shadowing, definition of viewpoints, photo-realistic texturing. As soon as interaction has to be involved, using events, linking and internal/external scripting are becoming more important. In fact, 3D worlds including real-time interactive navigation like walkthrough, flying, panning and sliding are requirements today – similar techniques are widely used in computer games. In order to examine singular objects, rotating is another important real-time navigation attribute. More advanced characteristics of virtual worlds are the maintenance of Levels of Detail (LOD) or multi-resolution texturing implementation are improving the performance. Furthermore, culling algorithms should be provided in order to make sure that invisible back-faces will not be rendered. Overall, the amount o rendered polygons is a factor for the smooth navigation. Any technique which is reducing the amount while keeping the world realistic should be used (Kofler, 1998).
Intuitive editing of 3D data is much more complicated than visualization. In order to provide a human readable GUI for editing, high efforts have to be done. This is the reason why mainly common CAD or GIS software products are used as front-ends at the moment (Stoter and Oosterom, 2002; Zlatanova *et al*, 2002b).

4.2 3D Visualization Techniques for the Web

VRML/X3D

VRML (Virtual Reality Modeling Language) respectively its successor X3D (Extensible 3D) were introduced by the Web3D Consortium to distribute interactive virtual worlds on the web. Both are mark-up languages and standardized. Whereby X3D is fulfilling the concepts of XML. Besides X3D is specified more modular. The rendering concept is mainly based on a scene graph definition and a node structure (Web3D Consortium, 2004). VRML andX3D are accomplishing the basic concepts for a 3D GUI (Dykes *et al*, 1999). To list all the features would take too long. Concepts of constructing a core virtual world and especially the external authoring interface (EAI) grading the possibilities around X3D/VRML up. By using the EAI, one can add individual functionality to virtual worlds. Developed either by scripting or higher programming languages, 3D scenes can get highly interactive. One good example is accessing a database from VRML worlds in order to retrieve new data (Zhu *et al*, 2003).
Realized VRML clients in combination with HTML have already proven their ability to react as GIS user agents in many examples and prototypes (see Chapter 5). However, no well-known commercial implementation is available. The most common use of VRML is within a client-side browser/plug-in implementation. Unfortunately plug-in vendors are hesitating with shipping X3D browsers.

Java3D

Another instrument for creating 3D world on the Web is Java 3D. The Java3D library is a freely available API for developing Virtual Worlds in Java (Sun Microsystems, 2004). Therefore Java3D classes can be used by Java Applets within HTML pages. Java3D's functionality is almost the same as VRML/X3D are providing. Savarese introduces them briefly (Savarese, 2003). One big advantage compared to plug-in based solutions is that developers have more control about rendering and user interaction. Another is the transformability. Compiled Java3D classes can either be used as standalone application or applet. In contrast to the mark-up languages of VRML or X3D, Java3D requires much more programming knowledge (Diehl, 2001). This is probably one reason, why only a few solutions have been realized using Java3D. One example for implementing Java3D within a geo-related application is the DEMViewer by Taddei (Taddei, 2003).

5 Review of Current Web 3D Systems

As mentioned above, recent 3D GIS implementations are mainly covering 3D visualization and simple interactive components like accessing additional information. Other general GIS principles, like data analysis are still missing. The reason for this is that the related data management is not suitable for real 3D functionality (Nebiker, 2003). However there are a couple of prototypes available which are pointing towards real 3D GIS. The following brief descriptions are introducing browser based and standalone front-ends.

5.1 Realized browser based solutions

As stated in Chapter 2, browser based solutions are almost represented by some kind of browser + plug-in approaches. These have the big advantage of good availability to the user. However, sometimes applications are developed just for one specific plug-in while other fail. Even if different plug-in's can handle the virtual world, almost all of them have a different GUI in terms of real-time navigation. This is the reason why they can be difficult to use for inexperienced users (Kofler, 1998). The following examples are almost using VRML embedded in HTML based web pages.

A prototype system of 3D GIS (Zlatanova, 2000)

The developed system is a typical example of a very thin client, i.e. based on HTTP, CGI scripting (realized in Perl), VRML and HTML documents which are created on-the-fly. The VRML delivers the 3D graphics information obtained as a result of spatial queries or/and provides means to query graphically the objects observed in the 3D scene (by standard VRML nodes). HTML documents are used to visualise text and images, to specify SQL queries or introduce new values for edited elements. Web and VR browsers on the client stations are utilised to interact with the 3D model(s) and specify queries. The data are structured according to the topological model SSM are maintained in a RDBMS, namely MySQL.

Requesting information about a particular object can be done either by typing its ID in a HTML form or by clicking on the corresponding object in VRML (its graphical representation). For example, a click with the mouse on a building activates a CGI script, which delivers a "Query-Result" section (HTML). The user selects the needed information from "pull-down" menu that is created on the fly with all the information available for the object in the database.

Extracting a group of objects according to a criteria is completed by directly typing SQL query at the "Query" section. The result of the query is displayed either in an HTML or in a VRML document. These documents are created on the fly only with the information related to the objects of interest. The same mechanism is used to create DELETE, UPDATE, and INSERT forms to edit data. The free access to the database provides a mechanism to specify and display a wide range of spatial queries. Examples of such queries are "which is the highest building?", "show the buildings in a particular area", "show all the streets", "show all the administrative buildings".

An advantage of the system is that clients practically do not use any specific software besides Web browser and VR plug-in. The system also does not have a specific GIS component since the SQL queries are directly sent to the database. The spatial functionality is provided by operations at database level. The major disadvantage is eventual overload of the server in case of many users. The performance of the system has not been tested for multi-user access. Another disadvantage is increased complexity of the VRML file if elaborated point-and-click operations are needed. To be able to work with freeware VR browsers, all the interaction with objects is incorporated in the VRML (using special VRML nodes). Therefore, in many cases the size of the VRML file can increase drastically.

GOOVI 3D (Coors and Jung, 1998)

The system architecture is a medium client-server where most of the functionality is provided at the server side but some functionality is also kept at the client side. The components of the system are

VRML, HTML, Java and warehouse. The warehouse consists of files organised on the server. The interface to the data warehouse is done by COBRA IDL and is based on IIOP protocol.
The two kinds of queries, i.e. obtaining additional information about a selected object and extracting several objects as a result of specific queries, are also implemented. In the first case this is done by attaching to the objects in the VRML files hyperlinks to a HTML page (the pages are stored in the warehouse) or, more dynamically, by Java script nodes. In the second kind of data queries (objects, which meet specific conditions), the server has to access at database in order to perform the queries. The results are represented by highlighting the objects of interest in the current VRML scene using Java and IIOP protocol. Thus no new VRML file is created. Since the system is indented for discussing urban plans, editing/modification operations are not implemented.
The authors make a suggestion for SQL node in VRML that can be directly used to connect to DBMS and extract information. First implementations of the system use RDBMS to store objects as VRML nodes and information about them as HTML pages. Later implementations made use of more generic representations in Oracle, using the topological model UDM (Coors, 2003).
The advantage of the system is that it is a relatively thin client-server system, allowing implementations without large resources at the client. Part of the functionality (data query) is performed at the server but highlighting of the objects of interest is at the client. In this respect the system is better balanced than the previous one. The system however is a bit dependent on the file organisation in the warehouse (i.e. mixture between files and DBMS storage). The major disadvantage is that the extended protocol IIOP is used (not available overall).

SALIX (Lammeren and Hoogerwerf, 2003; Wachowicz *et al*, 2002)

SALIX is a typical example of a thick client. The system is intended for interactive landscape planning, i.e. planning trees and bushes and simulating their growing. The GUI is based on the Cortona environment, using VRML and java to provide all the functionality. DBMS is used only to store the objects of interest (a variety of tree and bush). The objects are manually placed in the field of view. A large number of toolbars give the users the possibility to inspect certain constraints, the distance between the planted trees in different stage of their life, to simulate growing, to create conglomerates of objects from the same type, etc.
The significant aspect of this system is the extended functionality in terms of interaction and manipulation. There are still more improvements necessary toward making real use of functionality available at DBMS (currently used only for object storage).

Accessing Geo-DBMS Using Web Technologies (Vries and Stoter, 2003)

Vries and Stoter (2003) described two prototypes using a web environment to query 3D spatial data and their attributes. Moreover the implemented applications are focusing on reasonable ways to visualize the query results within a web browser. Because the operations are hosted server-side, the system is represented by a thin client and thick server. The realized prototypes can be differentiated by the implementation technologies as follows.

- **VRML and Microsoft-specific technologies**
  This implementation uses common web technologies to achieve a 3D GIS. Geo-data is already available within VRML files, whereas its attributes can be queried dynamically. These are stored in Microsoft's Access database system. Active Sever Pages (ASP) technology combined with the Internet Information Server (IIS) as web server environment is used to offer interaction with the database. The served VRML world is embedded within the main frame of the HTML Page. User interaction is possible in form of querying each objects attribute data. If the user clicks on an object in the VRML world a request is sent to the server. After connecting to the database, ASP is creating an appropriate HTML fragment which is holding the requested attribute data in a table. This is supposed to be embedded in the second frame of the application. This approach is vendor specific. It is only working properly on MS components in favour.

- **X3D, Java Servlets, XSQL and Oracle9i**
  This prototype system is based on an integrated database architecture. The underlying DBMS hosts 3D spatial data as well as their attributes.Oracle9i and its spatial extension are used in favour. Server-side, the system is based on a Java Servlet Container, like Apache Tomcat, and the Apache HTTP server. In detail, the prototype uses XML specific Java libraries to query (XSQL)

and exchange data. The libraries are part of Oracle's XML Developer Kit's (XDK) which are integrated in Tomcat here. Among the XSQL servlet and others, the XDK is providing a XML parser and XSLT processor. In order to visualize the queries, the XML response of the database is transformed to X3D using XSLT style sheets on-the-fly. On the client the browser window is separated into three frames. The main frame for showing the virtual world, another for displaying the object's corresponding attributes using HTML tables. The third frame is offering HTML forms in order to query the database for spatial objects. Once a query is performed, the main frame will visualize the new scene.
This state-of-the-art implementation is demonstrating the advantages a fully XML based system nicely. Furthermore, it can be integrated into any platform which is supporting the Java programming language. Fig. 3 shows the prototype's client interface.

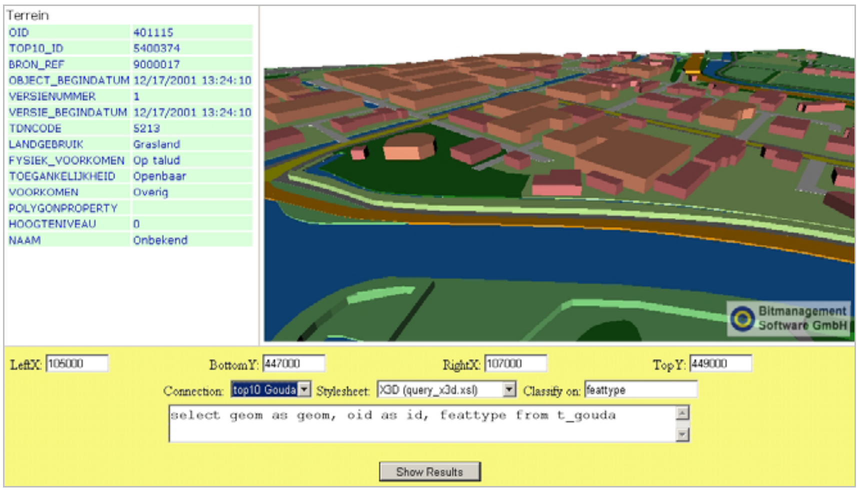

Fig. 3 A prototype system using web technologies to access Geo-DBMS

Pilot 3D of the GDI NRW

The Special Interest Group (SIG) 3D of the Geo-Data Infrastructure North-Rhine Westphalia, Germany (GDI NRW) proposes their first prototype. The 3D city model is based on the geometrical objects point, line, surface and body and has been presented in Gröger *et al*. The corresponding application logic – realized in Java programming language - is offering a standard based (OGC and ISO19107) solution to visualize 3D urban data (Gröger *et al*, 2004). The proposed data model is used in 3D city models, virtual flights and other projects which are able improve planning processes. For interactive 3D visualization, VRML is used at the moment. A first published result has been presented by the SIG 3D and is available online (SIG 3D, 2004).
Overall, the "Pilot 3D" project can be seen as a prototype scenario in order to prove the value of a standard-based Spatial Data Infrastructure. Most important here is that the SIG 3D is proposing their own extension of the Web Terrain Service called Web 3D Service (W3DS).

5.2 Standalone Solutions

Most CAD or GIS can be integrated into a web environment. They can be used as a user agent on the client. Stoter and Zlatanova are describing approaches using ESRI's ArcScene and Bentley's GeoGrapics iSpatial to visualize and edit data. These examples are not covering the integration into a web environment (Stoter and Zlatanova, 2003). However, one can do so. Because mean software products are difficult to use, they are not very suitable for inexperienced users. Therefore, different institution/companies have created special 3D applications. Geonova's Digital Landscape Server (DILAS) product line is one promising approach. The following paragraph is giving a brief description about the application and its components.

DILAS 3D (Nebiker, 2003)

Geonova's commercial product line DILAS offers a large variety of modules which can be seen as 3D Web-GIS. The DILAS server and manager are the system's main components. They are responsible

for characteristics like data storage, -management, representation and scene reconstruction. The DILAS modeler is an extension on Bentley's Microstation V8. This component integrates the creation and edition of new 3D objects and their corresponding styles. Moreover, the modeler benefits from the possibilities of Microstation due to the fact it is using its Java API. The server module is the connection for the database at this. In order to publish 3D worlds on the web, the DILAS scene generator is the key component. In conjunction with the visualization product G-VISTA it is generating complex 3D scenes like city models. These can be served by any web server. Most recently Geonova announced the new OGC conform Web Map Service. Therefore any client which is supporting this specification can be used (GEONOVA, 2003).
The whole concept and the already implemented features are looking very promising for the use in urban planning. Based on a state-of-the-art object-relational DBMS, DILAS offers managing, editing, reconstructing/visualizing and publishing virtual worlds. However, editing and managing of 3D scenes is only possible within an intranet network. Furthermore there is no 3D functionality offered by default. Nevertheless the shown examples are impressing.

GIERS (Kwan and Lee, 2003)

Kwan and Lee (2003) describe a developed GIS-based intelligent emergency response system (GIERS) which is implementing 3D routing features up to the inside of buildings for rescue teams in real-time. As a result, a navigable 3D GIS which is including building internal navigation as well the association to ground transportation possibilities of a city is presented. The underlying 3D data concept comprises a topological node-relation structure which is used for the routing operations. It has been transformed in to relational database model. On the technological side of the implementation, mainly Microsoft specific technologies are used. Furthermore, depending to its purpose, they system is able to communicate with mobile devices as well as through the Internet (Kwan and Lee, 2003).

6 Conclusions and Outlook

This paper made an overview on system architecture, data management and GUI visualization for Web 3D GIS.
Due to the fact that there are not truly Web 3D GIS systems available, further developments and research is still needed. The paper outlined the following important directions:

In Web 3D GIS the client can try to access the system from different devices (desktop computer, laptop, pocket PC, telephone, etc.). Therefore, it might worthy to consider thin clients and concentrate most on the functionality on the server side or in middleware implementations. However, the system has to be aware of the device type used by the client. In this respect, an important research direction is an intelligent automatic simplification (generalisation) and adaptation of the 3D vector data for the different clients. The client has to be able not only to request and visualise but also identify sufficiently itself. In addition, standards are necessary in order to improve interoperability.
Another critical question which has to be addressed is the edition of 3D data over the Web. Most of the systems discussed here focus on 3D visualisation and navigation as few or no tools are provided for modifications of data (both thematic and geometric). 3D editing requires a GUI extended with tools for pointing and selecting objects, parts of objects and constructive elements (vertices, edges, polygons) and a corresponding interface for editing. The currently provided 3D tools for navigation within the 3D model and querying their attributes are only the first step.
The most challenging 3D topic remains the maintenance of data. Implementation of true 3D primitive (e.g. polyhedron) is the first urgent development. Research on 3D primitives with curved surfaces and curved edges has to be initiated in short terms to be able to maintain urban objects created in CAD systems (e.g. complex buildings and bridges). 3D topology still requires a lot of developments, implementations and agreements on standards.
3D functionality has to be made available. This means that, first, advanced means towards specifying queries and analysis has to be provided and, second, algorithms for 3D spatial analysis have to be developed (i.e. 3D buffering, 3D navigation, etc.). Important improvements on performance of querying, analysing and visualising of 3D data are needed. An efficient organisation of LOD and images for textures will definitely speed up visualisation and navigation of 3D data.
Finally, further research should target the field of 3D standards (especially on the Web). Many

standards are already available but still the third dimension is not in the focus.

References

Altmaier Kolbe (2003). *Applications and Solutions for Interoperable 3D Geo-visualization.* Proceedings of the Photogrammetric Week 2003, Stuttgart, Germany.

Cöltekin A. (2002). *An Analysis of VRML-based 3D Interfaces for Online GIS: Current Limitations and Solutions.* Surveying Science in Finland, Vol.20, No: 1-2, p.80-91.

Connolly, T., and Begg, C. (2002). *Database Systems – A Practical Approach To Design, Implementation and Management.* 3rd Edition, Pearson Addison-Wesley, Menlo Park, California.

Coors V. (2003). *3D-GIS in Networking Environments.* Computer. Environments and Urban Systems 27/4, 2003, Special Issue 3D cadastre, pp 345-357.

Coors V. and Jung V. (1998). *Using VRML as an Interface to the 3D Data Warehouse*. Proceedings of the third symposium on the Virtual reality modeling language, Monterey, California, United States , pp 121 – 140.

Diehl S. (2001). *Distributed Virtual Worlds.* Springer-Verlag, Berlin Heidelberg New York.

Dykes J.A., Moore K.E., and Fairbairn D. (1999). *From Chernoff to Imhof and Beyond: VRML and Cartography.* Proceedings of 4th Int. Conference on the VRML and Web3D Technologies (VRML99), Paderborn, Germany.

GEONOVA (2003). *Newsletter Q3/2003.*

Gröger, G., Kolbe, T., Dress, R., Müller, H., Knopse, F., Gruber, U., and Krause, U. (2004). *Das interoperable 3D-Stadtmodell der SIG 3D der GDI NRW.* Version 2. Stand: 10.5.2004 (URL: *http://www.ikg.uni-bonn.de/sig3d/docs/Handout_04_05_10.pdf*).

Güting, H. (1994). *An Introduction to Spatial Database Systems.* VLDB Journal 3, Hans-J. Schenk (Editor), pp.357-399, 1994.

Jansen R. (2003). *Oracle, Java, XML: Integration in Oracle9i.* Frankfurt, Germany.

Kofler M. (1998). *R-trees for Visualizing and Organizing Large 3D GIS Databases.* Dissertation TU Graz, Austria.

Kwan, M.-P., and Lee, J. (2003). *Emergency response after 9/11: the potential of real-time 3D GIS for quick emergency response in micro-spatial environments.* In Press, Corrected Proof, Available online 28 September 2003. (*http://dx.doi.org/10.1016/j.compenvurbsys.2003.08.002*).

Lammersen R. van, Hoogerwerf T. (2003). *Geo-Virtual reality and Participatory Planning.* CGI Report 2003-07, Wageningen, The Netherlands.

Laurini, R. (2001). *Information System For Urban Planning – A hypermedia co-operative approach.* London New York.

Lopez, X. (2003). *Oracle Database 10g: A Spatial VLDB Case Study.* Oracle Cooperation Whitepaper. (URL: *http://otn.oracle.com/products/spatial/pdf/customer_success/papers/spatial_10g_ow2003.pdf*).

MAP24 (2004). (*http://www.map24.de/*).

Nebiker S. (2003). *Support For Visualization and Animation in a Scalable 3D GIS Environment: Motivation, Concepts and Implementation.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Science, Vol. XXXIV-5/W10.

Oosterom P. van, Stoter J., Quak W., and Zlatanova S. (2002). *The Balance Between Geometry and Topology.* Proceedings of Spatial Data Handling, Ottawa, Canada.

OGC (2001). *Web Map Service Implementation Specification.* (*http://www.opengis.org/docs/01-068r2.pdf*)

OGC (2002). *Web Feature Service Implementation Specification.* (*www.opengis.org/docs/02-058.pdf*).

OGC (2003a). *Geographic Markup Language (GML 3).* (*http://www.opengis.org/docs/02-023r4.pdf*).

OGC (2003b). *OpenGIS Reference Model.* (*http://www.opengis.org/docs/03-040.pdf*).

Peng, Y.-R., Tsou, M.-H. (2003). *Internet GIS – Distributed Geographic Information Services for the Internet and Wireless Networks.* Hoboken, New Jersey, USA.

Savarese D.F. (2003). *Learning to Fly.* Java Pro Magazine. June issue. *http://www.fawcette.com/javapro/2003_06/magazine/features/dsavarese/.*

Shekar, S., and Chawla, S. (2003): *Spatial Databases – A Tour.* Pearson Education, New Jersey.

Shi, W., Yang, B., and Li, Q. (2003). *An object orientated data model for complex objects in three-dimensional geographical information systems.* International Journal of Geographical Information Science, Vol. 17, No.5, pp. 411-430.

Special Interest Group (SIG) 3D (2004). *Pilot 3D der GDI NRW – Ergebnisse.* (URL: *http://www.ikg.uni-bonn.de/sig3d/docs/040109_Flyer_Endergebnis_3D-Pilot.pdf*).

Stoter J., and Oosterom P. van (2002). *Incorporating 3D Geo-Objects into a 2D Geo-DBMS.* Proceedings of ASPRS/ACSM, Washington, USA.

Stoter J., and Zlatanova S. (2003). *3D GIS, where are we standing.* Joint Workshop on Spatial, Temporal and Multi-Dimensional Data Modeling and Analysis, Quebec City, Canada.

Stoter J., and Zlatanova S. (2003). *Visualisation and editing of 3D objects organised in a DBMS.* Proceedings of the EuroSDR Com V. Workshop on Visualisation and Rendering, Enschede, The Netherlands.

Sun Microsystems (2004). *The Java 3D API. http://java.sun.com/products/java-media/3D/.*

Taddei U. (2003). *DEMViewer. http://www.geogr.uni-jena.de/~p6taug/demviewer/demv.html.*

Vries M.E. de, and Zlatanova S. (2004). *Interoperability on the Web: the case of 3D geo-data.* Paper submitted to IADIS International Conference on e-Society, Spain, July 2004.

Vries M.E. de, and Stoter J. (2003). *Accessing 3D geo-DBMS using Web technology.*

Wachowicz M., Bulens J., Rip F., et al (2002): *GeoVR construction and use: The seven factors.* Proceedings of the 5th AGILE, Palma.

Web3D Consortium (2004). *http://www.web3d.org/* (2004).

Zhu C., Tan E.C., Chan K.Y. (2003). *3D Terrain visualization for Web GIS.* Map Asia 2003, Kuala Lumpur, Malaysia, October 2003.

Zlatanova S. (2000). *3D GIS for Urban Development.* PhD Thesis, ITC Dissertation Series No. 69 , The Netherlands.

Zlatanova, S., Abdul-Rahman, A., and Shi, W. (2004). *Topological models and frameworks for 3D spatial objects*. Journal of Computers & Geosciences, May, Vol 30, Issue 4, pp. 419-428.

Zlatanova S., Abdul-Rahman A., and Shi W. (2002a). *Topology for 3D spatial objects.* International Symposium and Exhibition on Geoinformation, Kuala Lumpur, Malaysia.

Zlatanova S., Abdul-Rahman A., and Pilouk M. (2002b). *3D GIS: Current Status and Perspectives.* Proceedings of the Joint Conference on Geo-spatial theory, Processing and Applications, 8-12 July, Ottawa, Canada.