Korean Journal of Remote Sensing, Vol.23, No.1, 2007, pp.43~53

Application of GML and X3D to 3D Urban Data Modeling: A Practical Approach

Hak-Hoon Kim and Kiwon Lee †

Department of Information System Engineering, Hansung University

Abstract : In this study, two standard specifications such as GML (Geography Markup Language) from OGC (Open Geo-spatial Consortium, Inc.) and X3D (extensible 3D) from Web3D consortium were dealt with for a web-based 3D urban application without using commercialized tools. In the first step of this study, DEM (Digital Elevation Model) and 3D GIS data sets were converted to GML structure with attribute schema. Then, these GML elements were projected onto a common coordinate system, and they were converted to the X3D format for visualization on web browser. In this work, a 3D urban data model, as a simple framework model, is extended to a framework model having further detailed information, depending upon application levels. Conclusively, this study is to demonstrate for practical uses of GML and X3D in 3D urban application and this approach can be applied to other application domains regarding system integrators and data sharing communities on distributed environments.

Key Words : GML, OGC, Urban Data, Web3D, X3D.

1. Introduction

In spatial sciences and engineering fields including remote sensing and GIS, the consideration regarding international level standards with technical specifications is necessary. The keys of those standards are increasingly extended to interoperability for system resources on heterogeneous platform, framework data models for defining common database schema within an application domain and data formats for distributed data sharing, and so forth.

Among those, two main concepts are dealt with: GML (Geography Markup Language) and X3D (Extensible 3D Graphics). These two standards provide fundamentals for interoperable data model and data formats for 3D graphics and XML (eXtensible Markup Language: W3C Consortium)-database building for a given target application in web environment.

GML, from OGC (Open Geo-spatial Consortium, Inc.), is a markup language for encoding both spatial and non-spatial geographic information over internet. One of the advantages is that GML enables one to leverage the whole world of XML technologies.

In developing integrated information management techniques, difficulty in linking data sets and analysis tools is one of the barriers to be overcome. XML technologies are regarded as a solution for this

Received 13 December 2006; Accepted 21 February 2007.

† Corresponding Author: K. Lee (kilee@hansung.ac.kr)

problem on the internet environment, because the internet, being an efficient means of communication, provides an attractive platform to link models and data sets each other. Communications between model and data resources and the model user is crucial in the concept of the open modeling framework. Compatibility and reliability of data transfer are vitally important for business applications servicing a large number of clients and requiring across to extensive data resources. Furthermore, Data stored in a relational database are transferred to an XML document, which is then fed to an application model (Kokkonen *et al.*, 2003).

In particular, GML builds on XML, XML Schema, XLink, and XPointer. GML data can also be easily mixed with non-spatial data. By providing an open, standard, meta-language for geospatial constructs, information can be semantically shared across various domains like forestry, tourism, archaeology, geology and telecommunications. Among them, 3D city modeling is regarded as an important application domain (Suveg and Vosselman; 2004; Takase *et al.*, 2004; Varshosaz, 2004; Zhou, 2004; Kolbe *et al.*, 2005; Peng, 2006).

The development of real-time communication of 3D data across all applications and network applications has evolved from its beginnings as the Virtual Reality Modeling Language (VRML) to the considerably more mature and refined X3D standard. X3D is a royalty-free open standard file format and a run-time architecture for representing 3D scenes and communicating with 3D objects using XML. It is an ISO ratified standard that provides a system for the storage, retrieval and playback of real time graphics content embedded in applications, all within an open architecture to support a wide array of domains and user scenarios. Currently, interests on VRML/X3D applications have been increased in the various domains. X3D has a rich set of componentized features that can tailored for use in engineering and scientific visualization, CAD and architecture, medical visualization, training and simulation, multimedia, entertainment, education, and more (Lim and Honjo, 2003; Choi, 2005; Goeta *et al.*, 2006; Yan, 2006).

The main objective in this study is the design and implementation of a prototype for X3D-based 3D visualization associated with GML-based data modeling scheme toward web-based 3D urban application system dealing with complex typed urban features.

The development and operation environment of this implementation is as follows: Visual C++ 6.0 MFC and MSXML 4.0 SDK as development environment of the main modules and user interface, MSXML 4.0 as parser, Octago Player (http://www.octaga.com/) for encoded X3D viewer running on Windows XP.

2. Brief Overview

1) GML (Geography Markup Language)

XML is a text-based markup language for describing what the datum is, rather than just specifying how to display it, as it is the case of HTML. Further, XML is an approved format for representing structured data on the web and it has been applied for establishing the link between data and model resources.

A document type definition (DTD) can be used to define the structure of an XML document. A DTD lists a set of legal element names, specifies the hierarchy of the elements, and gives a listing of allowed attribute names. With a DTD different parties can agree on a particular XML application for data exchange. A validating parser can verify that received XML documents conform to the structure defined in the DTD.

Two features of XML have contributed to its increasing acceptance in web-based applications requiring standard data transfer or storage formats. First, the markup tags identify the information and break up the data into logical parts. Thus, the same XML data document can be fed to different applications, which then can extract the relevant parts of the data from the document for further processing. Secondly, as a plain text markup language, XML is platform-independent, which means that the application development is not tied to certain software (Kokkonen *et al.* 2003).

Geography Markup Language (GML) is an XML-based encoding technique for transporting and storing both spatial and non-spatial properties of geographic features. It is a key information technique behind the geo-spatial Internet. Using the GML, you can deliver geographic information as distinct features, and then control how they are displayed in a web browser. In other words, one of the primary objectives of GML is to express shareable geographic objects over the Internet.

Features, as real world entities (e.g. rivers, buildings, boundaries, or distributions of quantities over geographical coverage), are the fundamental objects used in GML, and thus, GML features can be concrete and tangible, or abstract and conceptual.

GML features are also described in terms of their spatial (e.g. location, shape, and extent of geographic objects), temporal (e.g. speed), or non spatial temporal (e.g. color, speed, and density) properties of the features. Rather, GML concrete features must be defined by GML application schemas, which are created by database administrators (Peng and Tsou, 2003). Normally, GML users create application schema that suit their application domain, which is land use, traffic, telecom and etc, using GML core schema.

GML provides a set of core schema components (e.g. feature, geometry, topology, temporal, etc) together with a simple semantic model between objects and properties that is similar to Entity-Relationship diagrams or the class and property model of RDF (Resource Description Framework). Using the GML model and its schema components, users can describe geographic types, whether concrete or conceptual, which are used within their application domain. The set of objects is created in the form of one or more GML Application schemas, which is XML schemas that make use of the GML schema components, and which comply with the GML semantic model and syntactic rules. A key benefit of GML is that the application schemas can be published and shared over the Internet, something that would be critical to any regional, national or international information infrastructure. XML schema is regarded as the implementation encoding of GML. Therefore, GML is composed of two kinds of document: schema document of defining structure and schema instance document (Cox *et al.*, 2003).

In this works, GML version 3.0 was used. It supports for encoding spatial and non-spatial properties. It is defined over 20 schemas: base schemas, general syntax, feature model, metadata mechanisms, basic 1D/2D geometry and additional 3D geometric primitives, geometric composites, coordinate reference systems, units, measures and values, definitions and dictionaries, directions, temporal information, and dynamic features, topology, and so on (Lake *et al.*, 2004).

As mentioned before, the practical core concepts of GML are feature, geometry and xLinks. Feature describes properties (name, type, value). FeatureCollection is one of feature types for complex typed real objects. Geometry is used for describing geometry properties of feature. It supports both 2D and 3D geometry component, such as point, polyline, polygon, multipoint, mulitpolyline, solid and so on. While, xlinks is for reference relationship between objects. It is also an independent devices and environment because of simple data format based text. Thus, it make wide uses of exchanging data format in web environment and else where.

Based on this basic architecture, grid coverage is used to encode satellite imagery, aerial photo, and distribution of land, climate, or elevation.

2) X3D (Extensible 3D Graphics)

X3D is an emerging software standard for defining various interactive web-based 3D content including web 3D graphics which can be integrated with multimedia across a variety of hardware platforms, extending the earlier VRML. It is regarded as a universal interchange format for integrated 3D graphics and multimedia, since it is represented by the XML. Conceptually, every X3D application contains graphic objects which can be dynamically updated through a variety of means based upon delivery context and user's own design.

The basic unit of X3D runtime environment is a directed scene-graph, a acyclic graphs containing objects in 3D world, as well as relationships among objects. These relationships composed of a transformation hierarchy for spatial relationships and a behavior hierarchy for fields and event flows in a 3D space. In these hierarchies, nodes within the scene graph contain descriptive fields and one or more child nodes to produce the desired hierarchy of objects in the scene. Detailed base components of X3D architecture can be referred in ISO/IEC 19775, as X3D components and X3D profiles (Park *et al.*, 2006).

X3D supports 3D functions such as polygonal geometry, parametric geometry, hierarchical transformations, lighting, materials, multi-pass/multi-stage texture mapping, pixel and vertex shading, and hardware acceleration, in addition to text, 2D vector graphics, and 2D/3D compositing.

Fig. 1 shows relationships between XML (GML)

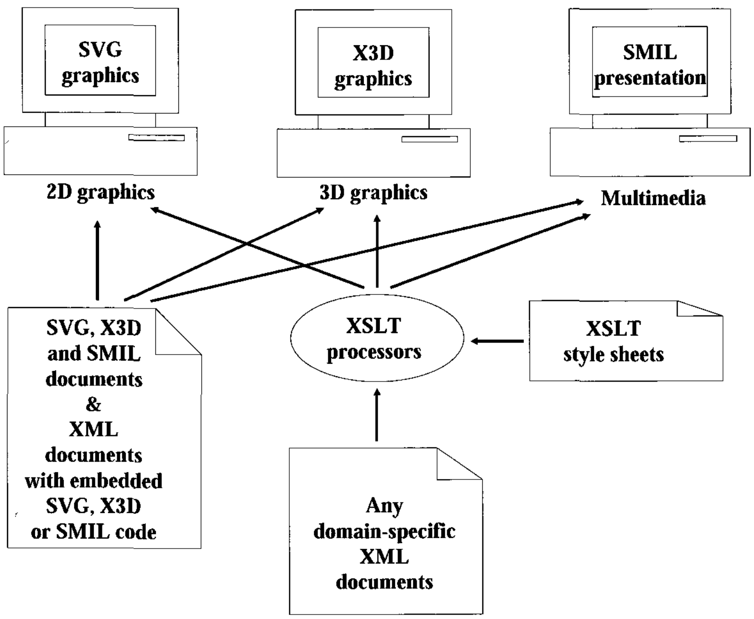

Fig. 1. XML(GML), X3D, and XSLT in web environment (Geroimenko and Chen, 2005).

and X3D with XSLT (eXtensible Stylesheet Language Transformations), which controls graphic attributes in visualization process by transforming an XML document into other formats.

3) XML-based Web Application Services

Using GML (XML) and X3D, an efficient web application services with web 2D/3D graphics is separation of data, logic and presentation (Geroimenko and Chen, 2005). Compared to a file-based system or a closed system, this type is categorized to the end-to-end XML system.

A file-based system or a closed system need multiple data locations in data, multiple data requests or batched data requests in logic, respectively. Furthermore, presentation processes are possible during the runtime. However, the end-to-end XML system characterizes reusable data or XML in data, and the logic is composed of three parts of aggregate, structure application, and transformation into XSLT.

In data presentation, XSLT is used to encode the logic to combine the programmed data interfaces to the rich presentation layer such as SVG (Scalable Vector Graphics) in 2D graphic objects and X3D in 3D graphic objects. Using XSLT, data-driven graphics, which one XML formatted data is transformed into another XML for presentation, can be implemented for web-based applications.

3. Design and Implementation Result

A prototype for modeling and visualizing 3D urban landscapes was designed and implemented using the standard specification mentioned above. Modeling process was carried out within the GML scheme and visualization was performed using the X3D and XSLT techniques.

This prototype is not a fully functioned system of urban system at this stage, and it only aimed for evaluation and practical consideration with actual data manipulation, regarding these two standards in the 3D urban modeling system, according to the end-to-end XML system.

At the first step for this work, we attempted to design GML application scheme for urban model, in UML (Unified Modeling Language) diagram.

Urban application using spatial database is a complicated task, and it is accomplished by a detailed data model. Moreover, establishment of the standard urban modeling is somewhat difficult, because the most urban area shows different structures along with the main function and the developing stage. Thus, application for a kind of simple urban model is performed in this study, and it is a basic level model with common urban components. It can be regarded as a kind of the framework model for further detailed application with actual data sets.

Fig. 2 shows a framework model linking GML for urban modeling. A city class is composed of basic sub classes such as road, building, facilities and elevation surface sub-class, as well as users' defined classes. The relation between city class and each sub class is inclusion, and the algebraic relation is one to many because one city has many buildings, road structures, facilities on elevation surface. Linking with GML, the city class, as a compound type including sub-classes is derived from FeatureCollection type of GML, and each sub-class is derived from feature type of GML.

Fig. 3 represents a diagram of the GML application schema in UML class diagram in Fig. 2. The city object, which is derived from FeatureCollection of GML, has elements of gml:description for its explanation, gml:name for its name, gml:boundedBy for its range and cityMember for containing structure objects. The structured object of City, as derived from feature of GML has its attribute and geometry property. For 3D geometric characteristics of these

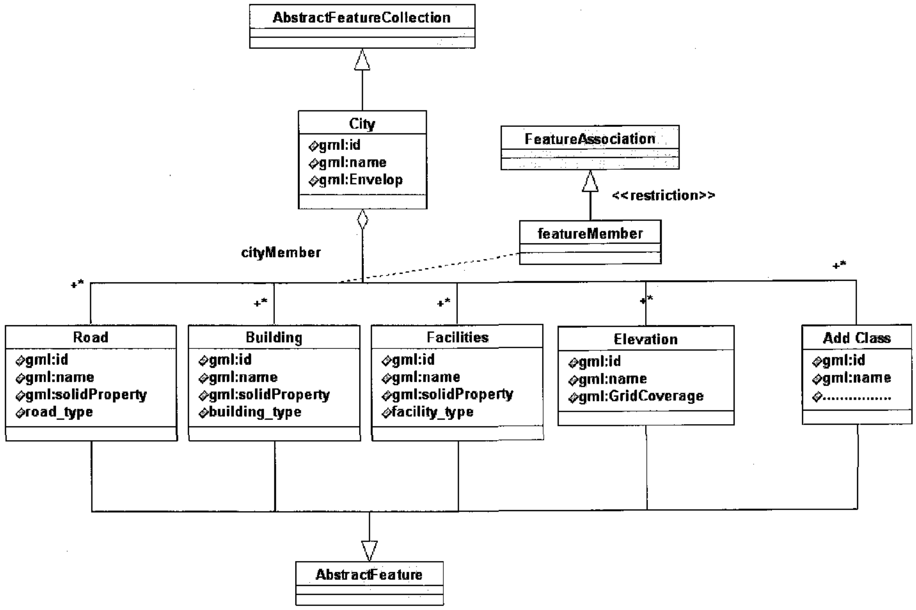

Fig. 2. Applied urban framework model in UML diagram for a prototype application.

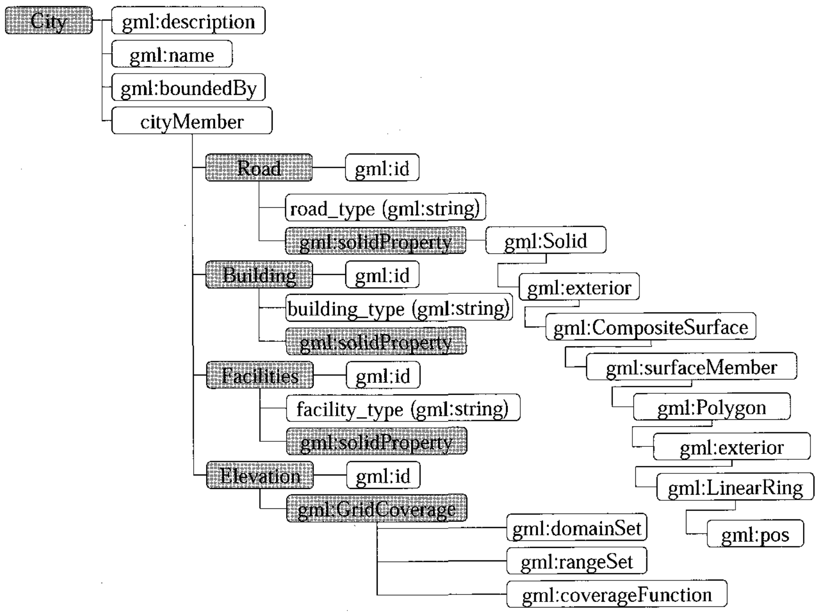

Fig. 3. A subset of GML application schema of the urban modeling.

objects, gml:solid as GML geometric component is used, and it has multi-facet by 3D coordinates. While, surface elevation uses gml:GridCoverage element with gml:domainSet, gml:rangeSet, gml:coverageFunction, to encode DEM data set into GML.

At the second step, we constructed an actual GML instance based upon the previous schema of Fig. 3. Fig. 4 shows a part of GML instance document in this framework model. A GML document can be written using a text editor, but manual GML typing is a time consuming work. To overcome this problem, implementation for semi-automatic generation of GML instance documents was carried out using DOM (Document Object Model) API by W3C. A DOM parser reads the entire XML document into memory and creates a navigable tree structure. Using DOM, data can be accessed out of order as well as manipulated to suit a particular task.

For building components, we produced ASCII files of GML documents with actual geo-based coordinates, which they were encoded using solid element, such as the 3D component of GML schemas, and they added building layers into an urban city model. Especially, for surface elevation components, a gridcoverage was used for encoding satellite images into the GML schema based upon the OGC specification. The gridcoverage may be stored and processed in both a ASCII and a binary format. Encoding gridcoverage of GML instance document includes filename, image size and resolution size of DEM data.

The third step is X3D document generation using DOM. The DOM is a platform- and language-neutral interface that will allow programs and scripts to dynamically access and update contents, structures and styles of documents. Documents can be further processed and the results of the processing can be incorporated back into presented pages.

Using DOM, we transform GML instance into X3D format for visualizing on a web browsing environment. The basic principle of the transformation is to extract image range or size and pixel information for web visualization from GML instance (Fig. 5). ElevationGrid and Indexedfaceset element, as X3D geometry components are used to represent terrains and buildings. The one is grid object including height. Range and height values of it are assigned by gridcoverage of source data (GML

```
- <Elevation>
  - <gml:GridCoverage>
    - <gml:domainSet>
      - <gml:Grid dimension="2">
        - <gml:limits>
          - <gml:GridEnvelope>
              <gml:low>0 0</gml:low>
              <gml:high>127 127</gml:high>
            </gml:GridEnvelope>
          </gml:limits>
          <gml:axisName>x</gml:axisName>
          <gml:axisName>y</gml:axisName>
        </gml:Grid>
      </gml:domainSet>
    - <gml:rangeSet>
      - <gml:File>
        - <gml:rangeParameters>
          - <gml:CompositeValue>
            - <gml:valueComponent>
                <Height>template</Height>
              </gml:valueComponent>
            </gml:CompositeValue>
            <gml:fileName>dem.dat</gml:fileName>
            <gml:fileStructure>Record Interleaved</gml:fileStructure>
          </gml:rangeParameters>
        </gml:File>
      </gml:rangeSet>
    - <coverageFunction>
      - <GridFunction>
          <sequenceRule order="+x+y">Linear</sequenceRule>
          <startPoint>0 0</startPoint>
        </GridFunction>
      </coverageFunction>
    </gml:GridCoverage>
  </Elevation>
```

```
- <ise:Building>
    <ise:Texture>D:\GML3D_X3D\Building\building8.bmp</i:
  - <gml:solidProperty>
    - <gml:Solid>
      - <gml:exterior>
        - <gml:CompositeSurface>
          - <gml:surfaceMember>
            - <gml:Polygon>
              - <gml:exterior>
                - <gml:LinearRing>
                    <gml:pos>570 70 550</gml:pos>
                    <gml:pos>600 70 550</gml:pos>
                    <gml:pos>600 70 580</gml:pos>
                    <gml:pos>570 70 580</gml:pos>
                    <gml:pos>570 70 550</gml:pos>
                  </gml:LinearRing>
                </gml:exterior>
              </gml:Polygon>
            </gml:surfaceMember>
          + <gml:surfaceMember>
          - <gml:surfaceMember>
            - <gml:Polygon>
              - <gml:exterior>
                - <gml:LinearRing>
                    <gml:pos>600 70 580</gml:pos>
                    <gml:pos>570 70 580</gml:pos>
                    <gml:pos>570 20 580</gml:pos>
                    <gml:pos>600 20 580</gml:pos>
                    <gml:pos>600 20 580</gml:pos>
                  </gml:LinearRing>
                </gml:exterior>
              </gml:Polygon>
            </gml:surfaceMember>
          - <gml:surfaceMember>
            - <gml:Polygon>
              - <gml:exterior>
```

Fig. 4. A part of GML Instance documents of urban modeling.

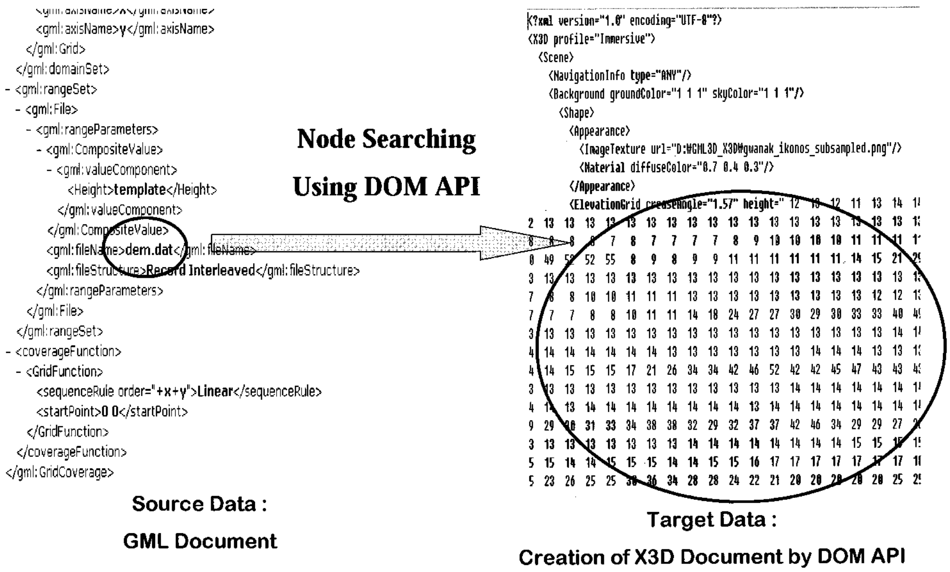

Fig. 5. A part of transformation process of GML into X3D by DOM API.

instance). The other is object that can create 3D geometry using coordinates of each surface of target. Coordinate values of it are assigned by solid of source data (GML instance).

Fig. 6 shows the schematic view of the workflow in this study, along with data or XML-database, logic and presentation in the end-to-end XML system.

This scheme can be applied to most web-based GIS applications; in this study, web-based 3D urban visualization system is considered for a primary target application. Data or XML-database are composed of core GML schema, 3D data model on 3D features

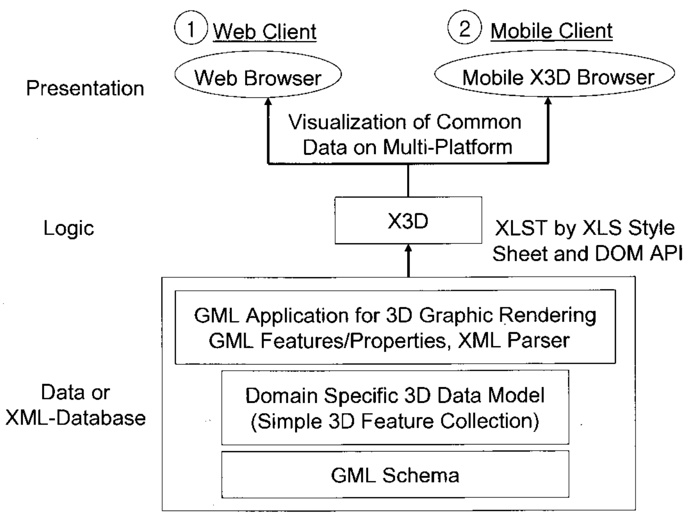

Fig. 6. The schematic view of the work flow and the base architecture.

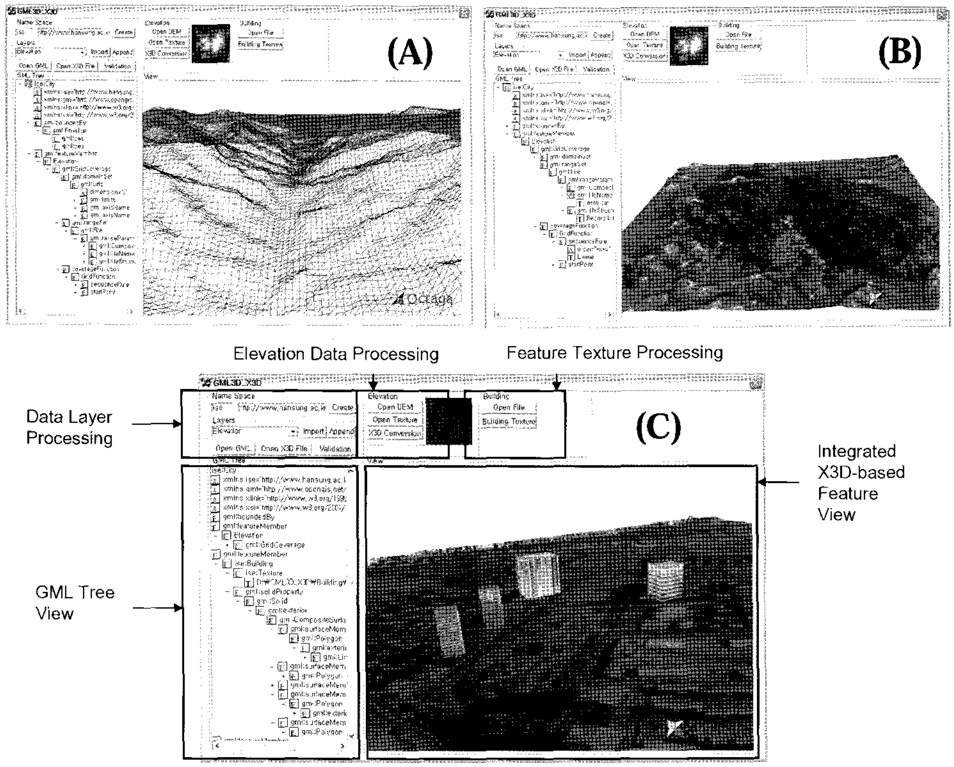

Fig. 7. A prototype development and its application cases: (A) wireframe, (B) texture mapping with satellite imagery, (C) user interface and 3D urban model with GML and X3D.

and their attributes for mapping GML features and properties later, and XML parser. Then, GML can be transformed into X3D by using XLST, as the logic part. Through this XLST processing, multiple presentations with same data model are possible. In the presentation part, plugged-in X3D browser such as Octago Player can be used without other commercialized viewers for web clients. As well, in the mobile environment, Pocket Cortona X3D Player (http://www.parallelgraphics.com/) or MobiX3D browser (Nadalutti *et al.*, 2006) on Windows Mobile can be used. For both web client and mobile client, the applicability of 3D visualization with respect to common data model on multi-platform is thought to be the main advantage in this approach.

Fig. 7 shows the implemented results of a prototype for urban 3D visualization system; (A) and (B) are 3D wire frame and texture mapped model using DEM and high resolution imagery in a certain area, respectively. Because the main purpose of this study is a generic system, the regional attributes on applied area are not considered in detailed.

Fig. 7(C) is the user interface of this prototype with other applied cases: ① Elevation data processing to read DEM and to encode X3D, ② Data layer processing to read the other X3D files, to read and validate GML documents, and to declare URL for layer definition, ③ GML tree view frame followed by elevation processing and data layer processing, ④ Feature texture processing for image loading and its draping on elevation surface layer, and ⑤ Integrated X3D feature scene view frame.

4. Conclusion

In this study, a prototype for a web-based 3D urban application is implemented with GML of OGC and X3D of Web3D consortium, both international information standards. Although this system can be developed without standard specifications, both GML and X3D are very practical specifications. But there is a few applications using these XML-database building and web-3D graphics. Some advantages of this approach are summarized as follows. This prototype does not need any commercialized tools for web 3D urban application. Applications on the dual platforms of a web environment and a mobile environment are possible by means of data sharing using X3D browser. Extensibility of XML, as internet data communication protocol, is available, because adopted cases of XML database into the enterprise or web-based GIS applications in the most domains are increasing.

However, some works are needed for more practical applications handling actual data sets. First, urban 3D data modeling to more detailed levels is necessary because the urban structure is very complicated and variable. Second, other OGC-based standards such as WMS (Web Map Service) and WFS (Web Feature Service) are needed with LOD (Level of Details) for handling large data sets, managing large and distributed data base and providing user interactive query functions. It is necessary to add to 3D complex feature types such as topology or temporal operators supported by GML.

Conclusively, this study is tentatively to demonstrate a prototype for practical uses of GML and X3D in further 3D urban applications and this approach can be also applied to other application domains with the architecture of web-based 3D graphics with XML database.

ACKNOWLEDGEMENTS

This research was financially supported by Hansung University in the year of 2007.

References

Choi, H., 2005. Virtual Reality and Internet GIS for Highway Simulation based on the ASE, *Korean Journal of Remote Sensing*, 21: 433-443.

Cox, S., P. Daisey, R. Lake, C. Portele, and A. Whiteside (Ed.), 2003. OpenGIS Geography Markup Language (GML) Implementation Specification, OGC-02-023r4, *OGC Implementation Specification*, 529p.

Geroimenko, V. and C. Chen, 2005. *Visualizing Information Using SVG and X3D: XML-based Technologies for the XML-based Web*, Springer, 298p.

Goeta, F., B. Ebmann, and T. Hampel, 2006. Collaboration by Illustration: Real-Time Visualization in Web3D, *Proceedings on Web3D, ACM*: 47-56.

Kokkonen, T., A. Jolma, and H. Koivusalo, 2003. Interfacing environmental simulation models and databases using XML, *Environmental Modeling & Software*, 18: 463-471.

Kolbe, T. H., G. Groger, and L. Plumer, 2005. CityGML-Interoperable Access to 3D City Models, *Proceedings on the First International Workshop on the Next Generation 3D City Models.*

Lake R., Burggraf, D. S., Trninic, M., and Rae, L., 2004. *Geography Markup Language: Foundation for the Geo-Web*, Wiley, 388p.

Lim, E.-M. and T. Honjo, 2003. Three-dimensional visualization forest of landscape by VRML, *Landscape and Urban Planning*, 63: 175-186.

Nadalutti, D., L. Chittaro, and F. Buttussi, 2006. *Rendering of X3D Content on Mobile Devices with OpenGL ES*, Proceedings of Web3D 2006: 19-26.

Park. K. B, K. I. Kang, and S. W. Kwak, 2006. *VRML & X3D for Virtual Reality*, 21 Century Press, in Korean, 422p.

Peng, C., 2006. *In-situ 3D concept design with a virtual city*, Design Studies, 27: 439-455.

Peng, Z.-R. and M.-H. Tsou, 2003. *Internet GIS: Distributed Geographic Information Systems for the Internet and Wireless Networks*, Wiley, 679p.

Suveg, I. and G. Vosselman, 2004. Reconstruction of 3D building models from aerial images and maps, *ISPRS Journal of Photogrammetry & Remote Sensing*, 58: 202-224.

Takase, Y., N. Sho, A. Sone, and K. Shimiya, 2004. Automatic generation of 3D city models and related applications, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, ISPRS, XXXIV-5/W10.

Varshosaz, M., 2004. True realistic 3D models of buildings in urban areas, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS*, XXXIV-5/W10.

Yan, W., 2006. Integrating Web 2D and 3D Technologies for Architectural Visualization: Applcation of SVG and X3D/VRML in Environmental Behavior Simulation, *Proceedings on Web3D*, ACM: 37-45.

Zhou, G., C. Song, J. Simmers, and P. Cheng, 2004. Urban 3D GIS from LiDAR and digital aerial images, *Computers & Geosciences*, 30: 345-353.