# A COM BASED FRAMEWORK FOR MANAGEMENT AND VISUALIZATION OF LARGE-SCALE DEMS

**WANG Yongjun***
**GONG Jianya****

*Dr. Candidate
National Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS),
Wuhan University, Wuhan 430079, China

**Prof.
National Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS),
Wuhan University, Wuhan 430079, China

**KEYWORDS:** COM, RDBMS, Visualization

**ABSTRACT**

As the developing of Geographic Information System (GIS) and Virtual Reality (VR), digital elevation model (DEM) plays more and more important roles in virtual reality generating, surface analyzing, orthophoto generating, etc. We presented here a Component Object Model (COM) based framework for management and visualization of mass multi-scale DEMs. The Framework consists of a data management component which based on RDBMS/ORDBMS, a scene management component and a scene rendering component. The data manager can manage mass (up to 30GB or more) multi-scale data expressed at variable reference frame within a pyramid hierarchy and supports fast access to data at variable resolution. The scene manager provides view-dependent data paging with the support of the underlying pyramid database and can organize the potential visible data at different levels of detail into rendering while the rendering component answers for the rendering of the scene using the standard graphic library such as OpenGL or DirectX. Due to the advantage of the COM, each component of the framework is able to be used both separately as independent module and in an integrated way. It has applied in the latest version of Geostar—a GIS software which has run to the fourth version now. At last, one of its applications which is a demo system now in National Geomatics Center of China was showed.

## INTRODUCTION

As the developing of Geographic Information System (GIS) and Virtual Reality (VR), digital elevation model (DEM) plays more and more important roles in virtual reality generating, surface analyzing, orthophoto generating, etc. China State Bureau of Surveying and Mapping (SBSM) had brought DEM into the production planning (Zhu et al, 1999). Due to different producer and software incompatibility, mass DEM data stored in various file formats raises a series of urgent problems on data management, analysis and visualization. ESRI ArcSDE demonstrated its effective to manage mass spatial data including DEM

(ESRI; Fan, 2002), but is poor to give power support to the generating of virtual scene and real-time walkthrough or flyover at interactive rate.

This contribution presents here a kind of RDBMS/ORDBMS based framework, which is complied with Component Object Model (COM), to manage and visualize mass multi-scale DEM data. Based this framework, a prototypical toolkit was implemented as a part of latest version of Geostar—a GIS software which has run to the fourth version now.

The reminder of this paper is organized as follows. Sect. 2 Introduces the Concept and rules of Component Object Model. And then briefly describe the structure of our framework. The RDBMS/ORDBMS based data management component was presented in Sect. 3. Sect. 4 describes the Scene management Component which organized the data flow in main memory and followd by Sect. 5 which discussed the Rendering of Virtual Scene. In Sect. 6, we give the implementation of the framework in National Geomatics Center of China and the results. At last, Sect. 7 contains some concluding remark and outlines further work of this framework.

## FRAMEWORK OVERVIEW

The Framework is targeted to be a community with broad and demanding range of functional requirements: the data management, visualization and so on. A major challenge in designing and building such a system is not only to develop basic system capabilities but to provide a framework where the best available tools can be integrated into the system with minimal effort and that each of these components can communicate with each other to create new applications. With this mind, we designed our framework as a component-based system, which is compliable with COM and will be able to support continuous increase of functionality and portability as new and more sophisticated tools become available.

Component Object Model (COM) was a binary software architecture brought forward by Microsoft Software Inc. COM is a platform-independent, distributed, object-oriented system for creating binary software components that can interact (Sara, 1994). As a binary standard which specified an object model and programming requirements that enable COM components to interact with other objects, COM components interact through a set of functional set called interface. By interface, COM defines a rule between component and their user. User access component's data and function through interface, instead of component itself. This rule separates the function from its program implementation and makes a different view between component developer and application developer about application. Component developer can pay more attention to single function set and application developer can assemble their application at a high level of abstraction by use various component regardless what language they used.

Our framework complies with COM. The architecture conforms to client-server mode and basically contains three components: data management component which can manage mass multi-scale data presented at variable reference frame within a pyramid hierarchy, scene management component which mainly answers for the management of DEM data in main memory and scene rendering component which answers for the rendering of virtual scene. We defined a set of interfaces adapted to describe geo data such as geometry, coordination system, etc. Each of components supports different interface and interact through them. Application or other component can use them separately or together whilst hold same high efficiency at data transmission and redundancy control. Fig 1 shows the architecture of the framework.

## DATA MANAGEMENT COMPONENT

The traditional way to manage DEM data was based on file system in which data was produced and stored in standard man-partitioned maps, each map was corresponding to a file. File system has inevitable default at multi-user operation, data security, concurrency control and so on (Gong, 2001). In the meantime, Object Oriented Database System is still unacceptable to the management of mass important national infrastructure data due to its default at reliability, maturity, standardization and also high costs. Contrast to this, The relational database management system (RDBMS) and object relational database management system (ORDBMS) is becoming the main stream in GIS area.

Data Management Component (DMC) mainly answers for the establishment and maintenance of the DEM database based RDBMS such as Oracle or ORDBMS such as SQL Server. As a middleware between client application and database, on the one hand, DMC organizes data in database as a manager, mainly on how to organize data, to build index and to retrieve data according different database manage system; on the other hand, DMC serve the application or other parent owner as a data server, mainly on data submitting, dispensing and maintenance.

To data, RDBMS/ORDBMS have got extensive application in various fields including geo-science. They have higher advantage over file system at multi-user operation, data security, concurrency control and so on. In comparison with this, most of DEM data was still keep in files in China now. Aiming at this target, our framework designs specific structure for the storage of DEM data in RDBMS/ORDBMS.

We take into account following facts and orientation in design the storage structure: 1.) the DEM data production was partitioned by map; 2.) maps have different scale generally corresponds to different resolution, extent and coordinate system, but have no relevant strict rules on them; 3.) most of RDBMS support the BLOB field in data structure; 4.) there is no ripe spatial SQL language adapted for geometry query; 5.) to manage all kinds of DEM data presented by regular square grid in database; 6.) to support high effective data access which is necessary when walk through or fly over virtual environment generated from the DEM data.

We designed a pyramid structure for the storage of DEM data. Each pyramid include several data layer, each layer was tiled by many single DEM map which has same scale but not always same resolution, and each was partition by regular cells which include equivalent number of vertex points and correspond to a record in database. Because most of maps in same layer have same extent, they were indexed by rectangle grid in each layer. Each layer has a virtual origin point. The partition of cell in each map may use the left-bottom corner of the map (or the virtual layer origin point lied on the coordinate system) as the origin and encode. The upper layer data can be primary data directly submitted to the database or auxiliary data sampled from lower layer. By this layer-map-cell architecture, DMC can easy retrieve any data in all kinds of resolution from the pyramid. User can get DEM data at specified resolution directly from relevant layer, or if there have no corresponding resolution in the pyramid, DMC will get the required resolution data from the nearest lower layer by interpolation resample. Both data submitting and data dispersing work with the interface describing the DEM data.

## SCENE MANAGEMENT COMPONENT

Due to the mass DEM data and current hardware resource, its impossible to model and render the virtual scene by loading all terrain data (including DEM and phototexture) in main memory when walk through or fly over the entire database. The Scene Management Component (SMC) is to answer for the data management in main memory. SMC creates visible data buffer and potential visible data buffer for the loading and updating of visible and potential visible data.

At first, SMC calculate the visible area by using the distance viewpoint can reach and viewangle. We call it here render buffer which is the area scene rendering component is to draw. A Certain extension to both sides and ahead of the render buffer is the area we called data buffer, where collect the data potential visible to viewpoint. If considering the backward walkthrough/flyover, the data buffer must contain areas some extent behind to viewpoint.

SMC partitions the virtual scene by regular cell. Each cell consists of a single hierarchical R-tree. SMC keep the root node of all R-tree at a list. The size of lowest layer cell of hierarchy is $n = 2^k+1$. SMC specify the resolution of lowest cell based on the highest resolution of the pyramid database, in most of case, the resolution of lowest cell equal the highest resolution of the pyramid database. When walking through virtual scene, the distance between center point of each cell and viewpoint was considered to determinate the resolution of cell. SMC loads the highest resolution of those cells which were nearest to viewpoint whilst loads the lowest resolution of cell which were far away from viewpoint. The resolution decreased as the distance to viewpoint become far and far. We limit the difference of layer number of R-tree between adjacent cells up to one to avoid undesirable crack.

In a multi-thread operating system, SMC and SRC reside in different thread and work synchronously, which make it possible to walk through or fly over the whole area the database covers. They communicate with each other by using event. When user walk through virtual scene, SMC load data from DMC for each cell in data buffer by the order they enter the buffer at different resolution based on the distance they away from viewpoint. As viewpoint moving, those cells have no longer contribution to current virtual scene was cast away and new cells had just enter the data buffer was load by SMC. SMC keep the equivalence between data obliterating and data loading, reduce the burden of main memory greatly.

## SCENE RENDERING COMPONENT

Scene Rendering Component (SRC) is used for the modeling and rendering of the visible virtual scene. SRC maintains the list of pointer to all data cell in render buffer. When viewpoint moving, SRC keep querying to the SMC whether each cell had ready for rendering and if got "Yes", just do it. SRC is designed to have the capability to use different graphic library such as OpenGL and DirectX based on special application environment. At present, we had finished the development using Silicon Graphics OpenGL library in the prototype system. We will try to using DirectX as the underlying graphic library in same interface to provide SRC strong flexibility to run at different graphic hardware platform.

## PROTOTYPE SYSTEM SAMPLE AND RESULTS

The data management and visualization framework was mainly consists of three components mentioned above. They can be used on different development platform and easy to embed in many applications. We present here two prototype systems which were developed using our framework and the component it contains. One is GeoVBSample which is a test application developed in Visual Basic. Another is demo system in National Geomatics Center of China. We call it here GeoDEMSpace. GeoDEMSpace was developed using our framework and GeoImageDB—One image component of Geostar—In Visual C++ environment. We built the 1:50000 demo DEM database which contains data about 750MB by using Oracle 8.1.6 as database management system and built the demo image database which contains data up to 20GB by using SQL Server 2000. When the system runs on general personal computer, user can load image data at high resolution (1 meter) from image database as phototexture of the terrain and can walkthrough or fly over entire virtual scene at interactive rate. Fig 1 present one snapshot of flying over virtual scene.

## CONCLUSION AND FUTURE WORK

As one part of Geostar, our framework and its component work well together with other toolkit within the software architecture, e.g. the image component mentioned above. The result of the experiment and the exemplificative sample argued the framework's effectiveness in management and visualization of mass multi-scale DEMs

The framework presented here is just at its prototypical phase and contains only several functional components primary to management and visualization of mass DEMs. Our future work contains developing new component and extending to old ones. The extending to DMC is to make it can using more other RDBMS/ORDBMS such as DB2, Sybase, etc. For the SMC, the most challenging problem is the employments of appropriate Levels of Detail which take into account not only the view relevant feature but also the surface character of terrain. As Complied with COM, the framework can be extended to the web-based solution. In addition, we are studying the management and rendering of 3D object with similar architecture.

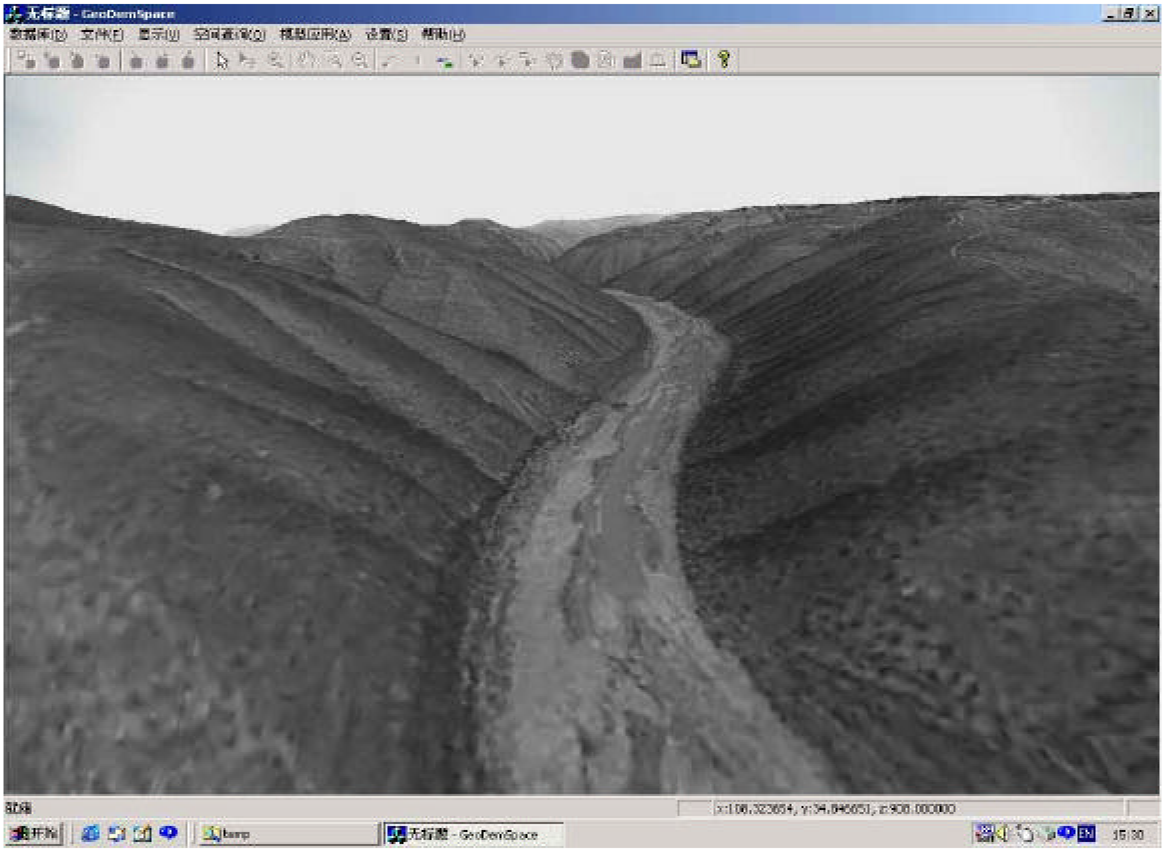


Fig 1  One snapshot of virtual scene when user fly over entire database.

**REFERENCE**

ESRI, ArcSDE Online, available on http://arconline.esri.com/ArcSDEOnline/default.asp?URL=/ArcSDEOnline /ArcSDE /Whats_ArcSDE.htm

Fan Hong, Zhan Xiaoguo, 2002. Application and Development Technology of ARC/INFO, Wuhan University Press, Wuhan, pp. 556-564

Gong Jianya, 2001. The Concept and Direction of Spatial Database Management System. Science of Surveying and Mapping, 26 (3), pp. 4-9.

Sara Williams and Charlie Kindel, Developer Relations Group Microsoft Corporation, 1994. The Component Object Model: A Technical Overview. MSDN Library, July 2000.

Zhu Qing, Li Zhilin, Gong Jianya, Sui Haigang, 1999. Review on the Scheme for "The Updating and Database Building of 1:10000 Digital Elevation Model. Journal of Wuhan Technical University of Surveying and Mapping, 24 (2), pp. 129-133.