

Visualization Service Based on Web Services

Bojan Blazona¹ and Zeljka Mihajlovic²

¹Ericsson Nikola Tesla, Zagreb, Croatia

²Department of Electronics, Microelectronics, Computer and Intelligent Systems,
Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

Scientific visualization evolved up to the stage where collaborative work in virtual coexistence of various experts has become unavoidable in identifying and solving specific visualization tasks. These problems encouraged us to research and define key aspects of visualization service which will enable collaborative work over the Internet. In this paper we define architecture of the whole system and unified interface for all requirements in such environment. Unified requests are sent to visualization processing unit to execute required visualization routine. Collaborative work on the volumetric medical data is a specially demanding task, so we present development of VMI (Visualization of Medical data over the Internet) system, where segmentation for visualization is based on joint histogram.

Keywords: visualization, web services, service-oriented architecture

1. Introduction

Vendor adoption of web service's concepts over the Internet, with new enhancements in security concepts and fast computer networks, is a reasonable base for development of the system which will enable collaborative work on visualization task definition from distant locations over the Internet.

Data visualization and the routine distribution with the ability to process request on distant place, is called Visualization Over the Internet (VOI). Concept of the visualization over the network is not new, but central data and visualization routine repository enabling collaborative work on visualization routine is new, and architecture and interfaces for that purpose are proposed in this paper. Also, we focus on visualization problems where the amount of data is significant. Medical volumetric data such as CT or MR sequences usually contain a huge

amount of data, but isosurface rendering could also generate significant amount of data.

Some pioneer concepts are presented as server-based visualization approaches developed for specific visualization needs like brain mapping [10] or building anatomical 3D scenes [2]. Our approach follows server-based visualization that relies on the processing routine [7] where script is used to define processing routine and VRML is used to present generated visualized data.

The project behind this paper is focused on medical data visualization, but the concept is applicable to all visualization tasks from other areas of science and human work. Problems about security, huge amount of generated data, privacy and interaction at any stage of the visualization process are new tasks in desired system and this is described in Section 2.

Development of distributed visualization systems for medical purpose can decrease overall cost of multiple desktop applications. This can be done by using central system (medical visualization unit) over network communication channels. Medical staff and computer scientists can collaborate to produce better health care in such system, using one central system from a different geographical location.

Protocols such as SOAP (Simple Object Access Protocol) for data transfer and standards like VRML or X3D for virtual worlds are very helpful in the definition of client side. Rendering the objects using these standards is the responsibility of the client viewers. These technologies are presented in Section 3.

Otherwise, new visualization patterns, from different areas of science, have to be simply adopted

in the system with respect to recent standard systems. New requirements in visualization have to change standards and protocol first, or development of the custom client is necessary.

Needs such as generating animation sequences based on visualization of anatomical human parts [2] in corporation with [1] can be mapped to similar process routine using base visualization functionalities (surface or volume rendering) with connectors to the Internet data centers where sequences are stored.

Emerging use of strict definition scheme for processing routine with ability of enhancements can be found in XML as the routine script and XML Schema standard for the strict functionality set which can be simply checked and finally executed. Our definition of such scheme can be found in Section 4. Finally, in Section 5, we present our VMI system. Developed system corresponds with defined key aspects, following the proposed architecture and communication concepts described in the previous four sections.

2. Concept of the Visualization over the Internet

A system that enables visualization over the Internet is a component-based system. Identified components are client and visualization service. Visualization service is then granulated to interface, unified interface and visualization processing unit (Figure 1).

Client and visualization service are the main parts of the system presented here. Client's re-

sponsibility is sending volume or other specific data, which must be accepted, processed and returned from visualization service.

Client or interface (for direct connection to standards e. g. DICOM or HL7 CDA) should provide a way to define processing routine for a given data to manage the visualization process. Client is responsible for the presentation of the visualized data. This can be provided by development of the new rendering software or integrating software applications to render specific responses from visualization service (3D viewer, graph generator, 3D printer).

At first, client sends the data to the visualization service. Collaboration of scientists from different areas is evident in process routine creation on visualization service. More users can be members of the visualization project (medical case). Processing routine can be changed without resending data to visualization service. Visualization services are responsible to process data according to the created processing routine upon the client's request. Processed data is retrieved on client, performing action of getting project by project's temporal ID (Figure 2).

To manage and process requests from client, visualization service has strict system architecture. This architecture consists of the interface, unified interface and visualization processing unit (Figure 1). Communication between interface and clients should be based on well used communication protocols and standards (such as HTTP, HL7 or DICOM). All requests should be mapped to unified interface of the visualization service.

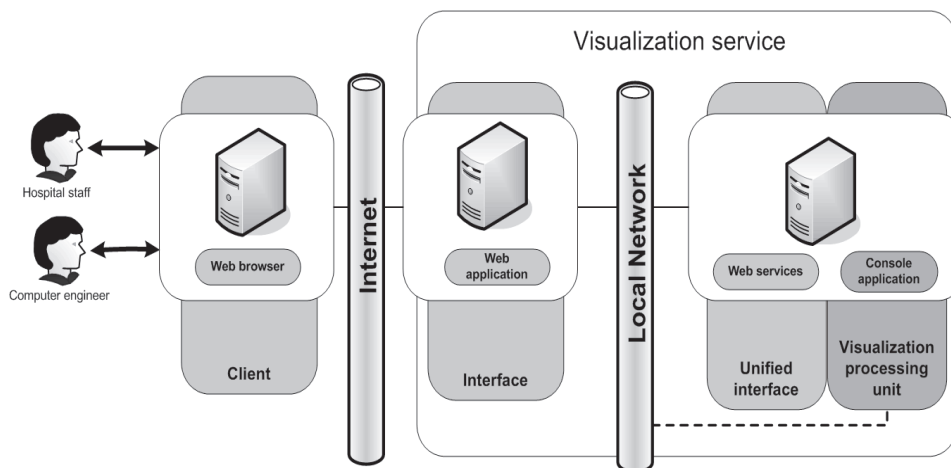


Figure 1. Overview of the system enabling visualization over the Internet.

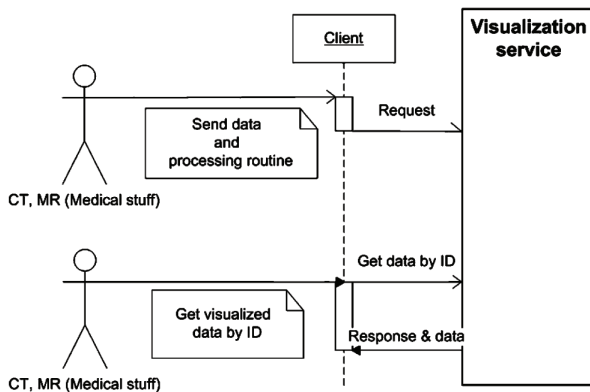


Figure 2. Client sends data and processing routine to visualization service.

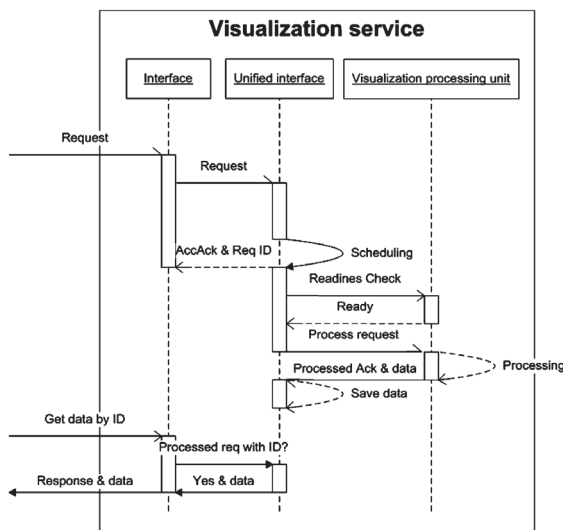


Figure 3. Client communicates with the interface according to selected protocol. Requests are unified and sent to unified interface.

Interface is responsible for accepting the requests according to the implemented protocol (Figure 3). Therefore, requests go through unification process.

Unified request is forwarded to unified interface. Temporal ID is returned to the interface after acceptance of the unified request on the unified interface. Requests are scheduled on unified interface if visualization processing unit is down or another process is running (this is the case when only one visualization processing unit is connected to unified interface). The time between invocations of processing request is set by operating system scheduling functionality. If visualization processing unit is ready, the process starts and generated data are returned to unified interface where it should be

saved. Client invokes get data by ID to see if specific request is processed through interface in event-based manner. Finally, generated data are returned if the request is processed.

3. Visualization of Medical Data over the Internet

The use of described VOI in the environment of exchange and visualization of medical data is called Visualization of Medical data over Internet (VMI). According to the defined key aspects, security is the most considerable task on the level of implementation.

Sending medical data, such as data sets or other reports, must go through a process of anonymization. All personal data, that can identify patient, must be moved before sending the data through the communication channel. Identification should be managed in another way like temporal ID. Use of secured communication channel is demanded even if anonymization is used.

Security threats are almost present because of the provisional server settings without advanced knowledge of the security concept of the underlying operating system.

Modern medical communication infra-structure consists of computer networks and other direct communication links. Generally, DICOM [8] (Digital Imaging and Communications in Medicine) is widely used and accepted norm for storing medical data and transferring them through computer network. DICOM's usage focused on radiology is overridden by new communication concepts, such as HL7 (Health Level 7) [9]. HL7 [3] can transfer all type of medical data through CDA (Clinical Document Architecture) [4]. Future system should implement DICOM or HL7 interfaces to VMI system. In this paper, we have chosen and developed Web Portal called VMI for sending DICOM files through HTTP interface (web application) to unified interface.

X3D [11] is an open XML standard for exchanging 3D objects, scenes and for real time collaboration through network applications in 3D environment. Defined profiles encapsulate functionalities from fundamentals, like 3D object's and animation sequence definitions (Interchange profile), to NURBS (Non uniform rational B-spline) and humanoid animation (Full

profile). Web3D Consortium has formed special group to develop MedX3D standard for medical use as enhancement to the X3D [12].

4. Processing Routine and Project Definition

Modern visualization toolkits use processing routine defined as script in plain text file. This simple approach is good, but it cannot accommodate the abstraction of the visualization process through the automatic imports of defined routines or special semantics according to the place of the visualization process subroutine definition. For example, generation of histograms must be implemented as the different functions with different parameters, but with our approach, the meaning of a histogram is defined by its place in the processing routine tree.

4.1. Project XSD Scheme

XML is the best standard to define processing routine. A strict scheme for XML processing routine defines allowed processing routine functionalities. It defines virtual object-oriented approach. Some tags define creation of objects and the other tags define invocation of methods with specific parameters. The scheme enables validation of lexical and semantic meaning of processing routine and demands strict implementation for vendors to implement defined functionalities.

Unification of all requests is verified by the Project XSD scheme. Project file should define input file definition, input file data, security specific data, status and processing routine. Project files should be processed by visualization processing unit (Figure 1).

4.2. Web Service Interface

Web service implemented on unified interface should employ methods such as Request (input is XML data according Project XSD Scheme) and GetProject (Project ID). Exceptions are thrown if request is not valid according to Project XSD Scheme. If exception isn't thrown, response parameter is project ID. Project ID identifies visualized data. Invocation of GetProject() method with returned ID will return visualized data.

Large datasets transferred through XML enlarge the size of initial dataset. Web services allow use of SOAP attachments to transfer binary data as they are without baseCode64 encoding. All binary files are attachments to SOAP message [13].

5. Technical Overview of our VMI System

Client side of the system consists of web browser with plug-in to render X3D 3D objects and scenes generated by visualization service. Communication between client and visualization service is secured with SSL protocol.

Visualizations service is based on .Net platform developed by Microsoft. We have developed web application as interface to visualization service. Interface is available as web application through Microsoft's web server IIS (Internet Information Services). Unified interface is implemented as web service with functionalities specified by concept of VOI, presented in Section 2. Also, project scheme follows definition for transferred data.

Unified interface is defined in WSDL (Web Service Definition Language). Visualization processing unit is C#.NET console application triggered by windows task manager which processes requests stored in project repository (project repository is local or remote folder) managed by the unified interface. Processing visualization unit saves results in project's folder in the project repository. Client's request for getting visualized data is forwarded through interface to unified interface, according to defined VOI sequence diagram for visualization service.

5.1. VMI XSD Scheme and Implemented Functionalities

VMI XSD scheme follows "object-oriented" design. All objects and methods are defined in XML elements. Attributes of these elements are parameters of the object constructor or parameters of the method that invokes. We developed DICOM parser to extract text data like volume data size, pixel or color presentation of sliced images and other equipment parameters from binary formatted DICOM dataset. Processing raw datasets requests input of the dataset volume size and real resolution.

5.2. Visualization and Segmentation Framework

In the visualization processing unit, we have implemented segmentation framework. Segmentation framework can be used for surface rendering based on the original Marching cube algorithm [6] with one isolation value parameter (Figure 4). We enhance segmentation framework to enable surface rendering based on the 2D area defined on 2D joint histogram. The idea is similar to the idea presented in [5] for volume rendering. Modification of marching cube is necessary to enable the same usage in surface rendering (Figure 5).

Processing routine could generate 1D histogram or 2D joint histogram (Figure 6; without area selection). Display of histogram is very important for correct segmentation and visualization. Selected area in 2D join histogram defines transfer function as shown in Figure 6. Isovalue is not defined by one threshold value, but combined values of intensity and derivative are used for segmentation in transfer function.

Segmentation framework presents segmented area of the same sized binary array. True presents vertex in a given area (according to 2D transfer function manifested as selection on 2D join histogram) and false for vertex out of selected area.

This approach generates hashed vertex data that presents segmented area and should be somehow interlaced to produce 3D objects. Binary selection produces artifacts on the result, so we develop separation method which filters segmented area with Gauss filter kernel in three dimensions. This interpolation of segmented data results in smoother models (Figure 7). It is a good example of adding a new functionality to visualization processing unit.

According to specific segmentation, some specific operations are required such as area selection. X3D shape writer creates X3D object as result of Isosurface rendering on segmented volume data. Object writer creates obj file as simpler presentation without specific appearance definition.

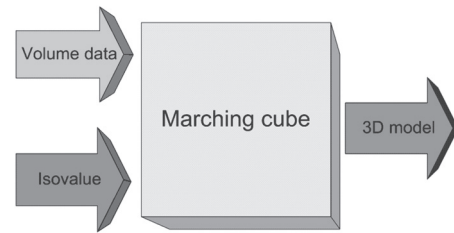


Figure 4. Marching cube algorithm overview.

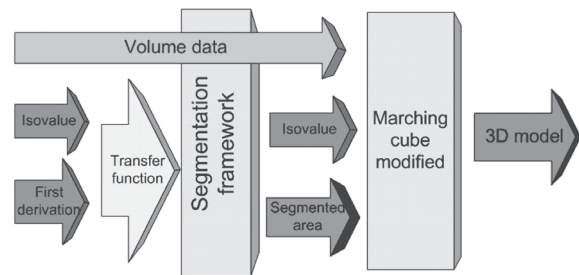


Figure 5. Segmentation framework and modified marching cube algorithm.

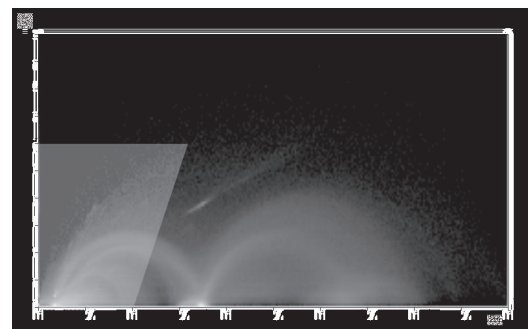


Figure 6. 2D join histogram (2D Histogram) of volume data. X axis – intensity level, Y axis – magnitude of first derivative, area presents occurrences $N(x,y)$ in logarithmic scale. Selected area extracts sinuses and air.

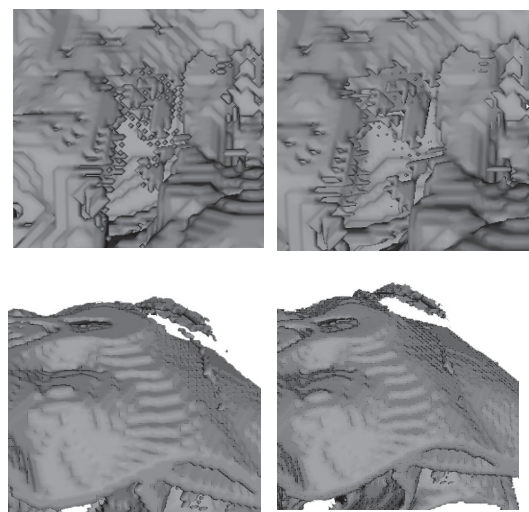


Figure 7. Left side – original object generated by marching cube; Right side – Gauss separation.

6. Results

Logging to the VMI is required. The user can select one of previously created projects to check project status and get processed data. Also the user can create a new project (Figure 8). Creation of the new project goes through the wizard to collect project details like definition and selection of input file type and file which will be uploaded to web application. Input file

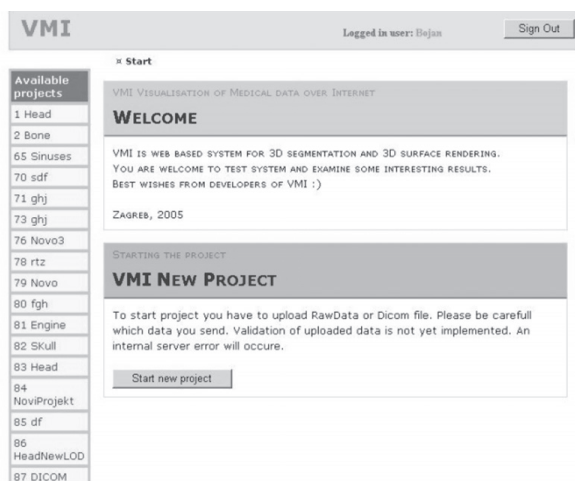


Figure 8. ASP.NET interface main screen with available projects (create new project option).

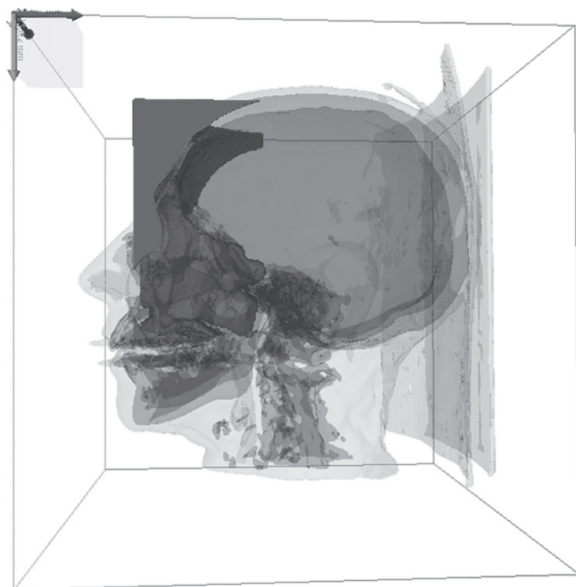


Figure 9. X3D scene generated from sample processing routine defined by VMI XSD Scheme. Sinuses & air, bones and soft tissue are presented. 1 825 394 polygons in three high detailed X3D objects.

type can be “Raw data” or DICOM file. Raw data input file requires additional parameters. Project is then created and all data is saved on ASP.NET Application host server.

Finally, sample processing routine generates X3D scene consists of the X3D objects in 2 LOD levels. Sample generated X3D Scene presented in higher level of details consists of the 1 825 394 polygons (Figure 9).

7. Conclusion

We have found that proposed architecture and given concepts of communication are acceptable according to key aspects we define. Strict definition of the communication based on presented architecture and XSD Scheme processing routine definition should encourage research of the separated parts of the system to find out the most effective way of implementation and functionality enrichment. Our VMI can simply adopt new interfaces such HL7 or direct DICOM connectors.

Medical technicians or biomedical engineers can be introduced as human support in sequence of visualization process. If we define this approach as a new way of virtual medical environment, the future research of presented architecture and communication concept of VOI is yet to be explored and developed.

References

- [1] R. BHATTI, B. SHAFIG, M. SHEHAB, A. GHAFOR, Distributed Access Management in Multimedia IDCs. *IEEE Computer*, Vol. 38, No. 9 (2005), 60–69.
- [2] F. EVESQUE, S. GERLACH, R. D. HERSCH, Building 3D anatomical scenes on the Web. *Journal of Visualization and Computer Animation*, **13** (2002), 43–52.
- [3] HL7 ORGANIZATION: HL7 Version 3, (2005), <http://www.hl7.org/>.
- [4] HL7 ORGANIZATION: The Clinical Document Architecture, (2005), <http://www.hl7.org/>.
- [5] J. KNISS, G. KINDLMANN, C. HANSEN, Multi-Dimensional Transfer Functions for Interactive Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, **8** (2002), Issue 3, 270–285.

- [6] W. E. LORENSEN, H. E. CLINE, *Marching Cubes: A High Resolution 3D Surface Construction Algorithm*. Presented at the *Proceedings of SIGGRAPH '87*, Vol. 21, No. 4 (1987), 163–169.
- [7] B. MAROVIC, Z. JOVANOVIĆ, *Visualization of 3D fields and medical data and using VRML*. *Future Generation Computer Systems*, **14** (1998), Issue 1–2, 33–49.
- [8] NEMA: DICOM STANDARD, (2004), <http://medical.nema.org/dicom/2004.html>.
- [9] H. OOSTERWIJK, *The relationship between DICOM and the CDA*. *Proceedings of 2nd International Conference on the Clinical Document Architecture*, (2004) Acapulco.
- [10] A. POLIAKOV, E. ALBRIGHT, D. G. CORINA, R. F. OJEMANN, J. F. BRINKLEY, *Server-based Approach to Web Visualization of Integrated 3-D Medical Image Data*. Presented at the *Proceedings of AMIA2001*, (2001).
- [11] WEB3D CONSORTIUM: *The X3D ISO standard*, (2005), <http://www.web3d.org/>.
- [12] WEB3D CONSORTIUM: *Medical real-time visualization and communication using X3D*, (2005), <http://www.web3d.org/>.
- [13] WORLD WIDE WEB CONSORTIUM: *SOAP 1.2 Attachment Feature*, (2004), <http://www.w3.org/TR/2004/NOTE-soap12-af-20040608/>.

Received: June, 2007

Accepted: September, 2007

Contact addresses:

Bojan Blazona
Ericsson Nikola Tesla
Krapinska 45
10 000 Zagreb, Croatia
e-mail: bojan.blazona@ericsson.com

Željka Mihajlović
University of Zagreb
Faculty of Electrical
Engineering and Computing
Department of Electronics,
Microelectronics, Computer
and Intelligent Systems
Unska 3
10 000 Zagreb, Croatia
e-mail: zeljka.mihajlovic@fer.hr

ŽELJKA MIHAJLOVIĆ is an associate professor at the Department of Electronics, Microelectronics, Computer and Intelligent Systems, University of Zagreb, Croatia. She received the B. S. degree in Electrical Engineering in 1988, the M. S. and Ph. D. degrees in computer science in 1993 and 1998 respectively, from the Faculty of Electrical Engineering and Computing, University of Zagreb. Her current research interests include visualization techniques, especially medical visualization, reconstruction in volumetric space and surface reconstruction.

BOJAN BLAZONA graduated in 2005 from the Faculty of Electrical Engineering and Computing (FER), University of Zagreb. He was awarded with several outstanding awards: "Stanko Turk Prize"; first prize in the Student Competition of the Croatian Medical and Biological Engineering Society (CROMBES); Best student paper award ITI 2007. He contributed to several national and EU projects: Integrated Health Care Information System (CRO); eMpower project-FP6 (EU); Service Delivery Platform (BIH); m-Commerce platform (BIH); Joint Information System for Land Registry and Cadastre (CRO); working for Ericsson Nikola Tesla Company.
