31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

Network Collaborative Environment supporting 3D Medicine

Přemysl Kršek, Michal Španěl, Miroslav Švub, Vít Štancl, Ondřej Šiler and Radek Bartoň

Abstract— **This paper is focused on the virtual collaborative consultation system which is intended for support of 3D geometrical modelling applications in the field of clinical human medicine. The system allows uploading the CT/MR data and 3D tissue geometry models (prepared in advance). The data define a 3D scene, which allows for viewing of the data and consulting them between technicians and physicians over the medium of computer network. The system is conceived as a three layer client-server architecture. For communication between the server and a client, the HTTPS protocol is used. Test results in Czech republic and the world-wide tests as well confirm, that the system is practically applicable and beneficial.**

I. INTRODUCTION

Various advanced technologies, more and more complicated, have been recently used in human medicine. This demands increasing effectivity of collaboration between physicians and technicians who are able to make use of the modern technologies effectively. The problems of effective communication between physicians and technicians is the objective of this paper.

Current usage of diagnostic image data (CT/MR data) is in the majority of cases based on qualified radiological evaluation of obtained data. Verbal description of the finding (diagnosis) is then based on these data. After considering the diagnosis, the physician conducts a treatment. The diagnostic data have a further use in relatively lower number of cases. These are for example precise planning of surgeries, navigations or manufacturing of custom implants.

One of the main obstacles to wider clinical use of CT/MR data is (aside from financial aspects) the fact, that the physicians are not experts on effective processing of image data, on 3D modelling, on designing implants etc. In majority of cases the physician has neither the time, nor the required experiences. Therefore he needs someone else to do it for him. In the ideal scenario, the clinic has technicians available, who specialize on the data processing. This is however not the general case. Therefore, the physicians are relying on conventional methods and the advanced methods aren't used that much, although there is a large number of available software tools, both commercial and academical.

There are couple of commercial companies offering custom implant manufacturing. Physician sends the CT/MR data to the company and the implant is sent back after some time. However the collaboration between the company and the physician is very limited. This means that the physician has little if any influence on the implant design.

Authors are with Faculty of Information Technology, Brno University of Technology, Božetěchova 2, 612 66, Brno, Czech republic, krsek/spanel/svub/stancl/siler/ibarton@fit.vutbr.cz

The ideal solution would be providing a service for the physicians, which would allow them to effectively consult with the technicians, who are participating in the process of data processing. This service should be independent of time and spatial () relationship between the physician and the technicians, who should be able to independently consult not only with the technicians in other county, but with fellow physicians in other hospital, city, country etc.

For these purposes we have been developing our collaborative network system, which is designed to support clinical applications of 3D modelling in medicine.

II. RELATED WORK

For the last seven years, we have been devoting ourselves to solving the problems of 3D geometrical modeling of human tissues, based on CT/MR data [8]. The models are consequently used for individual planning, simulations or navigation of clinical applications. We have been contributing to realized application (more then 45 surgeries [5][6]) within the clinical fields of: orthopedy, maxilofacial surgery and neurosurgery. Therefore we have been collaborating with physicians from the whole Czech Republic.

Number of modern hospitals store all diagnostic data within the PACS (Picture Archiving and Communication System) digital archive. Furthermore it is possible to share and transmit data between hospitals through the medium of optical networks. However, the PACS systems allow only for transmission and archivation of the data. They fail to provide the functionality of online network consultations (VCE - virtual collaborative environment) [3][4].

Videoconferencing has recently become a standard tool of communication technology. However it has been focused on communication between humans only. Moreover, without demands on high-capacity network connection (for HD transmission), it is not possible to practically implement full VCE consultations through video-conference itself. The problem that remains is ensuring the interactivity of data manipulation for every participating person.

An alternative solution to this problem could be utilization of shared desktop system, such as MS Windows Remote Desktop, Adobe Connect etc. But the same problems with network connection quality emerge. Considering the medical applications, the visualization quality of the data for more than two people is also a matter of great importance.

There of course existing solutions that fit between videoconferencing and shared desktop, like VizGrid project [1] or 3D Web Services project [2]. But they are oriented a little bit differently. VizGrid is more focused on natural interaction and is heavy on special equipment (cameras, CPU power

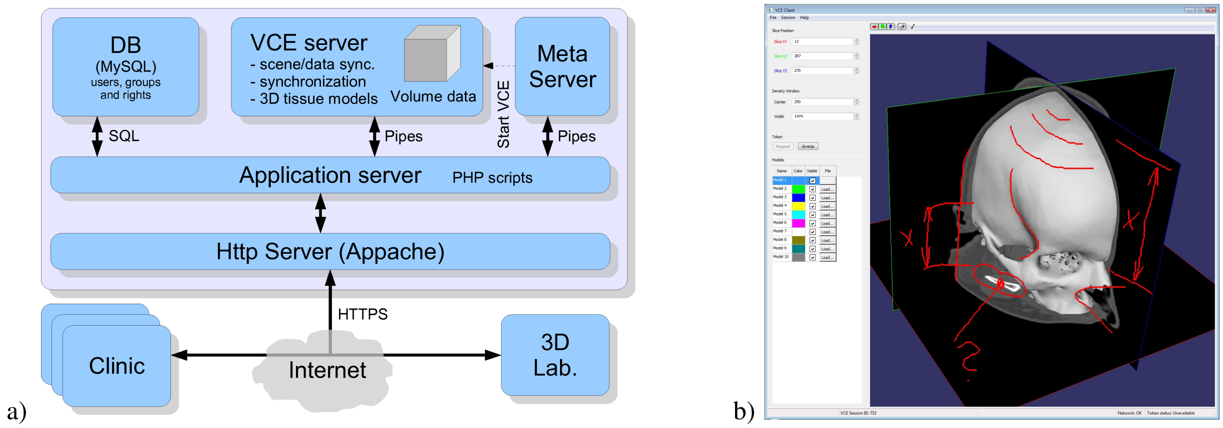

Fig. 1. VCE system: a) VCE system block scheme, b) Screen shot of VCE client.

etc.). Web Services project is more focused on clear service for creating 3D models than on being a collaborative tool.

Therefore, we have been devoting into developing our own VCE system, which would allow several people to interactively share and manipulate with 3D scene, which consists of CT/MR data and tissue 3D models. The first version of our system was based on Peer-to-Peer socket SSL communication between connected clients [7]. This system allowed for full real-time collaboration, but it was very sensitive to network lags. Second version of our VCE system was based on client-server architecture and HTTPS protocol [9] [10]. The main focus of the system was on the interactive consultations, to provide a support for 3D tissue modelling.

III. NETWORK COLLABORATIVE ENVIRONMENT

The current VCE system is built as a pure service, which doesn't archive CT/MR data. VCE server only maintains a database of users (rights, groups, sessions etc.). The server only runs active VCE sessions, which can run concurrently (dozens of them). Each VCE session allows for users to connect (again, dozens of users) to them according their group membership. The principles of working with the VCE system are following:

- A competent user sets up a concrete session on application server. The session is for one case (patient) only.
- Users, who want to participate (physicians, technicians etc,), connect to the session.
- CT/MR data are imported to the session, by their owner.
- Technicians import prepared 3D tissue and implant models, solution alternatives, navigations, etc.
- This session becomes the setting for interactive consultation. Marks can also be drawn into the scene.
- Consultation is being held again and again, until the result is confirmed by all the participants.

The system is implemented as a client-server architecture. Client applications connect to the server through a WWW server by HTTPS protocol. VCE session server holds a private copy of the data, maintains evidence of changes to the data and ensures synchronization of the clients. Graphical clients (thick client) synchronizes the data from the server (client has the whole copy) and provides displaying, processing of changes in the scene and sending the changes to the server. Application layer consists of several PHP scripts and it connects VCE session server and WWW server (Fig. 1a).

Interactivity of collaboration between the users of VCE session is resolved by active token holding. One client only can be handed a token, which allows him to make changes in the data and scene (rotate, move slices, display models etc.). Clients take turns in token holding according to their needs. The token holding client sends all the changes to the server. VCE server maintains the changes and it assigns a virtual time stamp to them. After a change has been accepted, the server virtual time is incremented. Clients who aren't currently holding the token are periodically demanding updates. VCE server sends them all the changes, that are corresponding to the difference between the virtual times (of client and server). In the case of interrupted connection with the token holding client, the token is seized by VCE server after some time has passed (60s).

To save time needed for volume data downloading and to start the collaboration as soon as possible, the data are downloaded progresively. Subsampled (4x) and compressed volume data($\sim$ 2MB) are downloaded first. Collaboration may then start in 10 $\sim$ 20 s. The full volume is then downloaded slice by slice at times when client is idle, which takes couple of minutes on average.

A. Server

The server part of the VCE system is implemented as a three layer client-server architecture. The part ensuring network communication is based on WWW server (Apache) which provides HTTPS communication. Clients login, groups, rights and actual running sessions are stored in small SQL dababase (MySQL). Metaserver is responsible for launching and shutting down of the particular VCE servers, because of security and robustness. Each running VCE server corresponds to one VCE session (Fig. 1a).

Application VCE server is implemented as a multi-threaded application (C++). This allows the server to service many clients concurrently. The communication follows the request-response pattern. Therefore the server is independent of the clients. Running individual VCE sessions as separate processes provides a good scalability of the whole system.

Between the WWW server and VCE server, there lies an application layer, which is implemented in a form of several PHP scripts. Connection with the database and the metaserver is done through PHP as well.

HTTPS communication between the server and the clients is binary coded in our own TGELD format (Time, Group, Element, Length, Data). **Time** represents a client virtual time of the item. **Group** (2 Byte) and **Element** (2 Byte) serve as operation identifiers (for example scene view change etc.). **Length** (4 Byte) corresponds to the length of transmitted data. **Data** item contains the data of current operation. The coding we have used allows us to effectively parse the incoming and outgoing data. It is a general, easily extendable format, which is inspired by the DICOM format.

B. Client

The second essential part of the VCE system is a graphic client application (VCE client). It displays the 3D scene, which consists of CT/MR data and 3D tissue models. The scene allows for basic 3D manipulation (3D rotation, scale, zoom etc.) Furthermore it is possible to move the multi-planar CT/MR slices and to define a density window (mapping gray-scale values to CT/MR intensities). It is also possible to draw (by mouse) free annotation curves (Fig. 1b).

VCE client (thick client) holds the entire copy of the scene data. These data (CT/MR data, 3D tissue models etc.) and parameters for their displaying (transformation matrix, projection matrix, density window etc.) are synchronized with the server. Synchronization (update) takes place periodically ($\sim$ 100 ms interval, adapted according to the parameters of the network). During the update, new changes are encoded as a stream of TGELDs and sent to the server in the case of a token holding client (the right to make changes). In the case of the other (not token holders) clients, a stream of new TGELDs is sent from the server during the update, selected by difference of virtual time on client and server.

Client is designed as a multi-threaded application: update thread periodically sends TGELDs, other thread has the task of rendering the scene and another thread makes changes to the local data according to incoming TGELDs.

In order to minimize dependency on the network latency and quality, the clients network interface itself is designed as multi-threaded (multi-connection). Requests on sending TGELDs are being lined up in an outgoing queue. A pool of threads is present (8 threads) and they periodically send the TGELDs in parallel fashion. Responses from the server are lined up in an incoming queue and processed. Due to this concept the network latency doesn't have a cumulative effect. The update frequency is maintained, so the response time of the whole system is more or less constant ($\sim$ 100 ms).

From the implementation point of view, the client application is based on the OpenSceneGraph 3D library, which ensures 3D rendering of the scene. The network HTTPS layer is based on the Curl library. Client application itself is written i C++ and its GUI is based on the wxWidgets toolkit.

C. Testing

We have been developing and debugging the whole system at our workplace FIT VUT in Brno. Therefore we have arranged testing in the conditions similar to the real setting. This means between different clinics and faculty hospitals cross the whole Czech Republic. The tests have been conducted on several levels:

- Local test at our workplace using fast network (1 Gbit, ping of under 1 ms). Test was conducted at several different spots (within our workplace).
- Tests within the metropolitan network of the city of Brno (10 – 100 Mb, ping of under 10 ms). Tests were conducted at our homes and at the faculty hospital of st. Anna in Brno, at the radiological clinic.
- Test between Czech cities (10 – 100 Mb, ping of under 25 ms). Tests were done in faculty hospital of Olomouc (80 km from Brno) and faculty hospital of Ostrava (200 km). Configuration with microwave connection was tested also (10 Mbit, ping of under 50 ms).
- Test within Europe. Currently, we have been preparing a test at the University Ray Juan Carlos in Madrid (10 – 100 Mbit, ping of under 50 ms). Another opportunity for testing should be a public demonstration of the VCE system at the Terena 2009 conference in Malaga, Spain.
- We have been preparing tests in Taiwan on the occasion of the Apan meeting 2009. This will be the test on the global scale (5-10 Mbit, ping of 320 ms).

The average results of the tests are shown in enclosed graphs (Fig. 2a). The measurements were done after the CT/MR data had been completely loaded (in the phase of scene state synchronization).

To test the share of individual parts of the VCE system on the overall response time, we measured the response time on the server as well. The measurement was done for different numbers of concurrently connected clients to a single VCE session. During that time, clients were connected within our network. The results are shown in enclosed graphs (Fig. 2b).

According to the local conditions, the data-flow of one VCE client side varies between 1kB/s $\sim$ 10kB/s during the standard scene synchronization. During the initial downloading of CT/MR data, the data-flow temporarily grows up to 100 kB/s $\sim$ 1 MB/s.

The data-flow on the server side depends on the number of concurrently connected clients. For 10 clients the average value of the data-flow is around $\sim$ 1MB/s during standard scene synchronization. During the initial data downloading phase, the server data-flow for 10 clients can reach up to 10 MB/s $\sim$ 50 MB/s. Progressive data downloading is the feature, that limits the data-flow peek during the downloading phase. Since the server is connected via 1 Gbit backbone, these data-flows present no problem at all.

The server itself runs on following configuration: 2xIntel Xeon 2.1 GHz (dual core), total 4GB RAM, Raid 5 disk array with the capacity of 4,3 TB. Server is connected to a 1 GBit backbone. The server is located at FIT VUT in Brno.

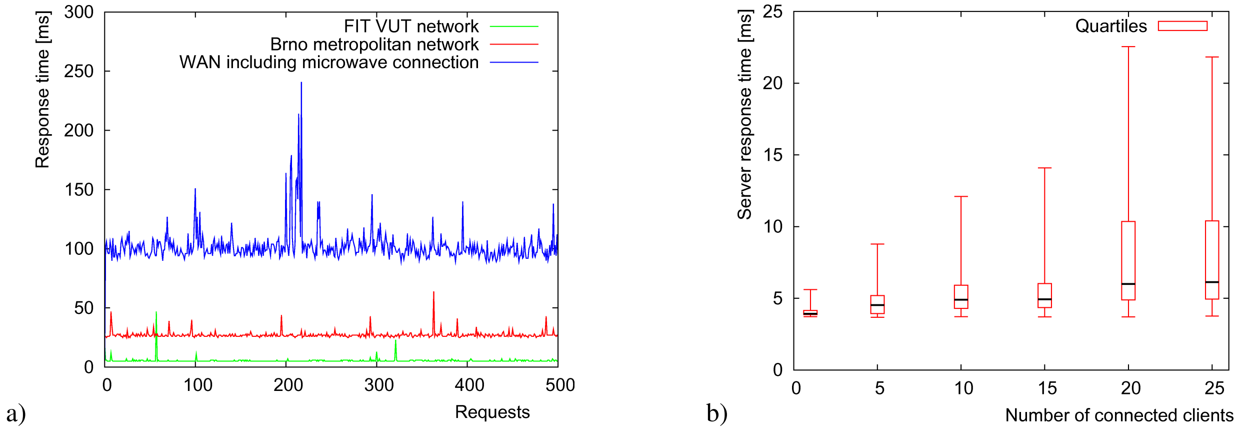

Fig. 2. Some testing reasults graphs: a) Response time on client side by type of network: local network on FIT BUT in Brno (ping ∼ 5 ms), Brno MAN (ping ∼ 20 ms) and Brno – Ostrava WAN including microwave connection (ping ∼50 ms) b) Internal server response time by number of clients, boxplot.

IV. CONCLUSIONS AND FUTURE WORKS

Based on large number of previous experiences with 3D modelling applications in medicine, we have designed, implemented and tested a support system for online network consultations, the VCE.

From the implementation point of view, we have build the system on the existing robust communication tools and instruments (Apache, MySQL, PHP, HTTPS, CurLib etc.) Therefore, we have been able to build a stable, reliable and affordable solution. We ourselves can focus on the problems of VCE system and development of VCE clients.

The test results of the VCE system in various conditions show, that the solution we developed is fully functional and that it is practically applicable for the task of online consultations in 3D medicine. After adding the complete functionality to the client applications (density measuring, distance measuring etc.) it will be possible to deploy the system in the clinical practice. Minimally within Czech Republic or similar conditions.

Even though we have used standard instruments (WWW server, PHP), the performance of the server is very good (the response time of 5 ms ∼ 10 ms). The response time is mainly influenced by the properties of network. Therefore, the whole solution is easily scalable. By using faster hardware or virtualization, we can adjust the system to serve dozens of clients concurrently connected to a single VCE session and dozens of sessions concurrently running.

Considering the properties of current networks, the data-flows on client and server are more or less insignificant. The main limitation from the network point of view is the latency. If the latency doesn't exceed ∼ 50 ms (common in Czech Republic), the response time of clients is sufficient. 5 ∼ 10 updates per second can be done without noticeable lags in communication. As the latency of the network grows, the response times remain the same, the communication is just slightly delayed. This is due to the multi-threaded (multi-connection) nature of our network interface.

The VCE system only provides collaboration on the 3D scene with CT/MR data. No other communication resources have been implemented so far. Therefore, it is appropriate to use the system in combination with other multi-medial resources (NetMeeting, Skype, H323/SIP etc.).

In the future, we would like to focus on generalizing the system not only for CT/MR data, but for other modalities as well (US, RTG, endoscopic video scanners, etc.)

V. ACKNOWLEDGMENTS

Supported by grand project No. MSM6383917201 CZ and MSM0021630528 CZ

REFERENCES

[1] Matsukura R., Koyamada K., Tan Y.: Virtualized Collaboration Environment with 3D images, In: Joho Shori Gakkai Kenkyu Hokoku, Japan, p. 33-38, 2004, ISSN0919-6072
[2] Kooper, R., Shirk, A., Lee, S., Lin, A., Folberg, R., Bajcsy, P.:3D Medical Volume Reconstruction Using Web Services, In: Proceedings of the IEEE international Conference on Web Services ICWS. IEEE, Washington, ISBN 0-7695-2409-5
[3] R. K. Hall: The role of CT, MRI and 3D imaging in the diagnosis of temporomandibular joint and other orofacial disorders in children. Aust Orthod J. 13(2): 86-94, 1994.
[4] G. Hamarneh, V. Chua, M. R. Bardalo, M. Schweitzrc: Deformation Analysis of Hoffa?s Fat Pad from CT images of Knee Flexion and Extension. Medical Imaging 5746: 527-534, 2005.
[5] Černochová P., Kaňovská K., Kršek P., Krupa P.: Application of geom. biomodels for autotransplant. of impacted canines, In: World Journal of Orthodontics 2005, Paris, FR, 2005, p. 1, ISBN 1530-5678
[6] Krupa P., Kršek P., Černochov P., Molitor M.: 3-D real modelling and CT biomodels pplication in facial surgery, In: Neuroradiology, Berlin, DE, Springer, 2004, p. 1, ISBN 0028-3940
[7] Peiva Jan: Active Transaction Approach for Collaborative Virtual Environments, In: ACM International Conference on Virtual Reality Continuum and its Applications (VRCIA), Chinese University of Hong Kong, HK, ACM, 2006, s. 171-178, ISBN 1-59593-324-7
[8] Španěl M., Kršek P.: Vector-based Medical Image Segment. using Adaptive Delaunay Triangul., In: Proceedings of the Sixth IASTED International Conference on Visualization, Imaging, and Image Processing, Mallorca, ES, ACTA Press, 2006, p. 6, ISBN 0-88986-600-7
[9] Kršek P., Španěl M., Švub M., Štancl V., Šiler O., Sara V.: Network collaborative environment for human tissues 3D modeling, In: Engineering in Medicine and Biology Society Proceedings, Lyon, FR, 2007, s. 4540-4543, ISBN 1-4244-0788-5
[10] Kršek P., Španěl M., Švub M., Štancl V., Šiler O., Sara V.: Consultation Virtual Collaborative Environment for 3D Medicine, In: Engineering in Medicine and Biology Society Proceedings, Vancouver, CA, 2008, s. 4540-4543, ISBN 1-4244-0788-5