

# An Interactive Grid for Non-Invasive Vascular Reconstruction

Peter M.A. Sloot<sup>1</sup>, Alfredo Tirado-Ramos<sup>1</sup>, Alfons G. Hoekstra<sup>1</sup>, Marian Bubak<sup>2,3</sup>

<sup>1</sup>*Faculty of Sciences, Section Computational Science, University of Amsterdam  
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands*

<sup>2</sup>*Institute of Computer Science, AGH, al. Mickiewicza 30, 30-059 Krakow*

<sup>3</sup>*Academic Computer Centre –Cyfronet, Nawojki, 30-950, Krakow, Poland  
{ sloot | alfredo | alfons @science.uva.nl }, bubak@uci.agh.edu.pl*

## Abstract

*We conduct computer simulation experiments in pre-operative planning of vascular reconstruction with a physician in the experimental loop. We constructed a problem-solving environment, which offers an integrating approach for constructing and running complex interactive systems. Grid resources are used for access to medical image repositories, segmentation resources, simulation of blood flow, and visualization of the simulated results, together with medical data obtained from MRI/CT scanners. This case study in vascular reconstruction planning has been validated via contextual analysis in a number of hospitals, resulting in a set of new requirements gathered for future versions of the system.*

## 1. Introduction

Processing, visualisation and integration of information from various sources play an increasingly important role in modern healthcare. Information sources may be widely distributed, and the data processing requirements can be highly variable, both in the type of resources required and the processing demands put upon these systems. Grid technology is a major cornerstone of today's computational science and engineering. By offering a unified means of access to different and distant computational and instrumental resources, unprecedented possibilities and benefits are expected. Connectivity between distant locations, interoperability between different kinds of systems and resources, and high levels of computational performance are some of the most promising characteristics of the Grid.

For medical applications, issues such as remote access to patient data, medical knowledge bases, advanced visualization technologies and specialized

medical instruments are of utmost importance [1]. For these applications, Grid technology provides dedicated support such as strong security, distributed storage capacity, and high throughput over long distance networks. Besides these immediate benefits, the computational resources of the grid provide the required performance for large-scale simulations, complex visualization and collaborative environments, which are expected to become of major importance to many areas of medicine.

Over the past 20 years we have witnessed a shift in the scientific methods, from observation, hypothesis and experimentation to a substantial emphasis on computer simulation as a new scientific method paradigm. One could say that in addition to in-vivo and in-vitro, also in-silico nowadays plays an important role in biomedical research. Problem Solving Environments (PSEs) [2] are a type of collaborative environments that have naturally evolved from the traditional scientific method and aim to support this new paradigm.

The grid provides a powerful medium to achieve the integration of large amounts of experimental data and computational resources, from simple parameters and highly distributed networks, into complex interactive PSEs that allow the researchers to efficiently run their experiments, by optimizing overhead and performance.

In this paper we present a Grid-based PSE for pre-treatment planning in vascular interventional and surgical procedures, developed by the University of Amsterdam. We lay out a base architecture for PSEs using the Grid as a medium, with a validated case study in vascular reconstruction, and provide an extensive functional update of the work we presented in the First Healthgrid conference [3]. For additional background, motivation, and the latest Grid-based results, we refer to [4].

## 2. Vascular Reconstruction as a Case Study

Cardiovascular diseases are considered as a leading cause of death in the developed world, and are becoming more prevalent as well in developing countries [5]. About every 30 seconds someone in the United States suffers a coronary event [6]. Vascular diseases affect arteries and veins. Vascular disorders in general fall into two categories: aneurysms and stenoses. An aneurismal disease is balloon like swelling in the artery. Stenosis is a narrowing or blockage of the artery. The purpose of vascular reconstruction is to redirect and increase blood flow or repair a weakened or aneurismal artery if necessary.

There are several imaging techniques that can be used to detect vascular disorders. 3D data acquired by computed tomography (CT) or magnetic resonance imaging (MRI) is converted into a set of 2D slices that can be displayed and evaluated from various perspectives and at different levels. Magnetic resonance angiography (MRA) is a technique for imaging blood vessels that contain flowing blood. It is very popular among specialists working in cardiovascular diseases because of its ability to non-invasively visualize the disease.

In vascular reconstruction surgical operations can be performed in different ways. They include adding shunts, bypasses and stent placement in the case of aneurysms and applying thrombolysis techniques, balloon angioplasty, bypasses, etc. for a stenosis.

The best treatment is not always obvious because of the complicated vascular disease of a patient and sometimes of other diseases that the patient may have. The verification of the operation plan can be the most complicated task in the process. A PSE, in which patient specific situations are simulated and several treatments can be tested in real time, can provide useful clues for surgeons and assist student surgeons in understanding the complexity of the treatment, during the diagnosis and planning phase of the process [Fig 1].

The Virtual Radiology Explorer (VRE), which is being developed at the University of Amsterdam, is part of a PSE that puts a user at the center of an experimental cycle controlled by a computer and allows him to apply his expertise to find better solutions for treatment of vascular diseases [6][7]. The aim of the VRE is to provide an end user with an intuitive virtual simulated environment to access medical image data, visualize it, and explore patient's vascular condition. Furthermore, a procedure for adding a bypass, where a bypass is a graft rerouting a

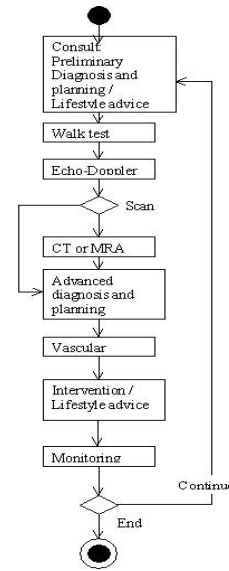


Figure 1. Simplified task trajectory of vascular diagnosis and treatment

blood flow around blockages, is of most interest as it can be used both for treatment of aneurysms and stenosis. In the VRE, Grid resources are used both for the simulation of the blood flow and for the visualization of the simulated results together with medical data about the patient obtained from a scanner in a hospital. Since this kind of medical image processing is usually a complicated and resource intensive task, additional computational resources are needed.

## 3. A Simulation Environment for Virtual Vascular Reconstruction

The criterion of success of a vascular treatment is the normalization of blood flow in the affected area. VRE contains an efficient parallel Computational Hemodynamics solver [9][10][11] that computes pressure, velocities, and shear stresses during a full systolic period. The simulator is based on the Lattice-Boltzmann method (LBM). LBM is a mesoscopic approach for simulating fluid flow based on the kinetic Boltzmann equation [12][13][14]. To convert the medical scans into LBM-grids, the raw data is first segmented so that only the arterial structures of interest remain in the data set. The segmented data set is then converted into a grid that can be used in LBM; boundary nodes, inlet and outlet nodes are added to the

grid using a variety of image processing techniques. The simulator generates patient's blood flow using Grid resources. To allow for parallel execution, the simulation volume is divided into several sub-volumes, and each sub-volume is processed concurrently. For visualization we use a semi-immersive wall as a projection environment [15][16].

The VRE system uses a virtual reality environment where the patient's data obtained from a CT/MRI scanner is visualized as a 3D stereoscopic image together with the graphical interpretation of the simulation results. A user can then manipulate the 3D images of arteries, patient's body and blood flow structures in virtual reality [Fig 2]. The concept of the interaction in a stereoscopic immersive virtual reality was selected as a basis for building the interaction component of the VRE: stereovision is the normal way almost everyone sees in the real world. Virtual reality is an environment where users interact freely in a 3D space with entities within it. The working prototype of the VRE is provided with a multi-modal interface described in [17]. It combines natural input modes of context sensitive interaction by voice, hand gestures and direct manipulation of virtual 3D objects. We called it the Virtual Operating Theatre, as a user can 'play virtually' a role of a real vascular surgeon planning and conducting treatment of a vascular disease on a simulated patient.

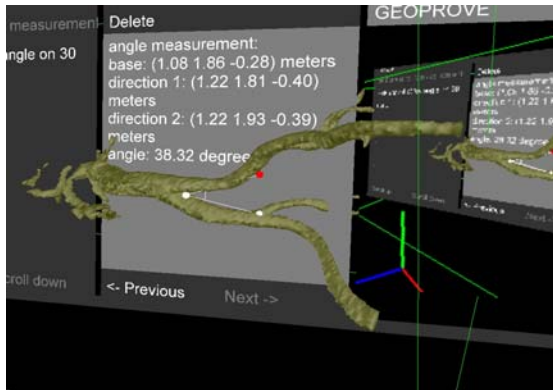


Figure 2. 3D-image manipulation in the VRE Virtual Operating Theatre

Within the VRE a user can conduct measurements, pick up a position on the artery for creating a bypass and modify the shape and size of selected bypass by mesh editing. Mesh editing provides the possibility to edit the geometry of an artery, i.e. add or remove mesh elements and restore elements lost during the segmentation or conversion stages.

Measurements are important for diagnoses. Clinical decision-making relies on evaluation of the vessels in terms of the degree of narrowing for stenosis and dilatation (increase over normal arterial diameter) for aneurysm. The selection of a bypass (its shape, length, diameter) depends on sizes and geometry of an artery. The interactive measurement component of the VRE provides the possibility to measure quantitatively distances, angles, diameters and some other parameters characterizing 3D objects in a virtual world, where markers are building blocks of distance, angle, and linestrip measurements [18]. For our initial experiments with Grid resources, a desktop version of the VRE system is developed in the framework of the CrossGrid project (the DesktopVRE), to make the biomedical application available in both immersive (CAVE [19], DRIVE, DesktopVR) and non-immersive (PC, PDA) projection modalities. The DesktopVRE has further advanced capabilities, such as changing the initial geometry and creating a virtual bypass [Fig. 3].

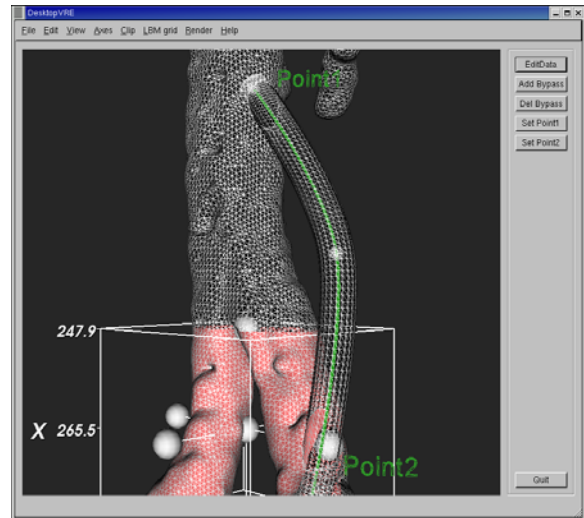


Figure 3. Creation of a virtual bypass within the DesktopVRE

#### 4. Integration with the CrossGrid Testbed

The CrossGrid [20] pan-European distributed testbed shares resources across sixteen European sites. The sites range from relatively small computing facilities in universities, to large computing centers, offering an ideal mixture to test the possibilities of an experimental Grid framework. National research networks and the high-performance European network, Geant [21], assure interconnectivity between all sites. The network includes a local step, typically inside a

University or Research Center, via Fast or Gigabit Ethernet, a jump via a national network provider at speeds that will range from 34 Mbits/s to 622 Mbits/s or even Gigabit, to the national node, and a link to the Geant network at 155 Mbits/s to 2.5 Gbits/s.

The CrossGrid team focuses on the development of grid middleware components, tools and applications with a special focus on parallel and interactive computing, deployed across 11 countries. The added value of this project consists in extension of the Grid to interactive applications. Interaction, in this context, refers to the presence of a human in a processing loop, and a requirement for real-time response from the computer system. The CrossGrid testbed largely benefits from the DataGrid [22] experience on testbed setup and Globus [23] middleware distributions [Fig. 4]. The efforts to establish an integrated CrossGrid testbed started with the release of EDG 1.2.0; currently LCG-1 [24] is deployed in the testbed.

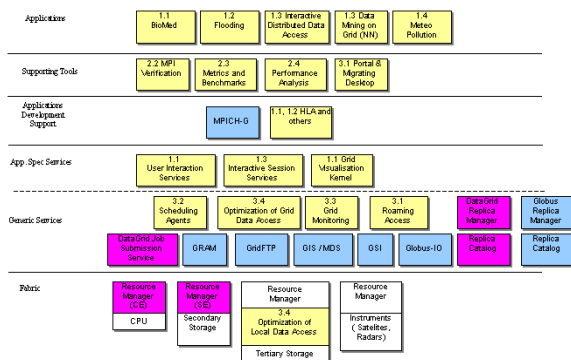


Figure 4. CrossGrid layered architecture, in the fashion of Foster et al. [25]. The yellow blocks indicate CrossGrid components, the blue blocks are Globus components, and the red components stem from the DataGrid project.

According to the CrossGrid testbed architecture and minimum hardware requirements, each site has at least five system components:

- Gatekeeper: the system that provides the gateway through which jobs are submitted to local farm nodes. The gatekeeper is the interface through which grid enabled systems can use the local resources.
- Worker Node (WN): a local farm computing node where jobs are actually executed. Jobs received by the Gatekeeper are sent to the WNs through the local batch scheduling system. Many WNs can exist behind a single Gatekeeper. The combination

of a Gatekeeper with its WNs is usually called a Computing Element (CE).

- Storage Element (SE): the generic name used for any storage resource that includes a Grid interface ranging from large Hierarchical Storage Management Systems to disk pools.
- User Interface (UI): the system used by end-users to submit jobs to the grid CEs. The job submission is usually performed through a resource broker (RB).
- LCFG installation server: used to install, configure and maintain the above systems from a single management system. The installation server is required for the proper installation and ease of maintenance of the EDG and LCG middleware.

The testbed includes a set of tools and services such as monitoring tools, development tools, a remote access server, portals and a prototype of the parallel resource broker [Fig 5].

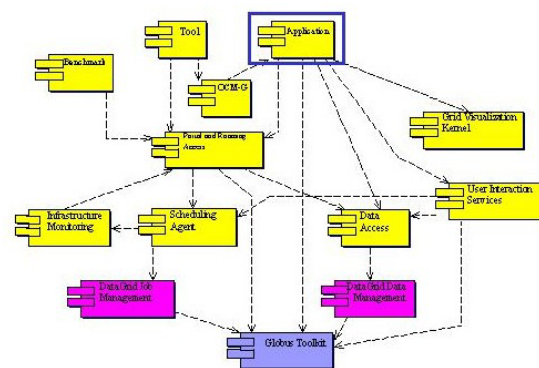


Figure 5. CrossGrid Component Architecture

Since the EDG RB does not support parallel applications, the program executions have been performed using the Globus job submission capabilities directly. The MPI package used for the tests was MPICH-G2, a variant of the MPICH implementation [26] using Globus IO for inter-process communication and Globus DUROC for resource co-allocation. The Resource Broker is the component of the Datagrid WMS that has been mainly modified in CrossGrid.

A key component of the Crossgrid architecture is the grid portal offered by the Poznan team, the Migrating Desktop (MD) [27]. The MD produces a transparent user work environment, independently of the system version and hardware. The MD allows the

user to access Grid resources and local resources from remote computers. It allows the user to run applications, manage data files, and store personal settings, independently of the location or the terminal type. This tool extends the functionality of the portal by providing a specialized advanced graphical user interface.

We have incorporated our VRE system into the Grid via the MD grid portal. We achieved secured grid access, node discovery and registration, grid data transfer, application initialization, medical data segmentation, segmented data visualization, Lattice-Boltzmann mesh creation, job submission, distributed blood flow visualization, and bypass creation. The DesktopVRE runs on a local machine, but it is launched and initialized through the MD. The input for the DesktopVRE is the segmented or non-segmented medical data produced at the Leiden Medical Center (LUMC); the CrossGrid testbed provides access to the medical data of interest from a medical image repository acting as a Grid SE in Leiden. The generation of an accurate computational mesh from such data, for the flow solver, is another part of the DesktopVRE functionality.

## **5. Interactive Grid-based Problem Solving Environment**

Consider the following scenario: A patient walks into a Medical Center scanning room to get his blood flow measured, the technician scans the abdominal aorta area, and the resulting image is stored in the Radiology Information System repository to be pre-examined and segmented. Later, a physician (user) in Amsterdam logs into the CrossGrid Portal using his Grid certificate and private key. The user checks if there are segmented or non-segmented medical data ready for analysis and simulation in one of the virtual nodes to work with them locally, and securely transfers a few. The user then starts the DesktopVRE from within the MD, loads the segmented medical data, selects a region of interest, crops image, adds a bypass, and creates LBM mesh. The user selects the Biomedical Application icon within the MD (parameters and files are taken from user's profile), and submits the job to the CrossGrid, to the nearest/most adequate CE in the Grid. The user may then check job submission or progress via the MD. After the job has been completed, the velocities, pressure, and shear stress are transferred to the local SE or to the appropriate visualization engine to be rendered and reviewed by the user.

This scenario implies at this moment downloading the MD (Poznan Supercomputing and Networking Center) to a storage element, secured access to the testbed, virtual exploration of available SEs throughout the Grid, secure data transfer from an image repository SE (Leiden Medical Center), the preparation of the data for the blood flow simulation within the DesktopVRE version of the biomedical application, job submission to the LBM solver via the RB (Lisbon Instrumentation and Experimental Particle Physics Laboratory), and visualisation of the simulation results using the GVK-server [28]. All processes are transparent to the user. For a complete recording of this scenario see [29].

### **5.1 Grid Portal Download**

MD Java-based download allows the user to access the Grid resources from machines like stand-alone PCs, notebooks or desktop workstations. It allows to run applications, manage data files, store personal settings independently of the localization or the terminal type. It allows users to handle Grid and local resources, run applications, manage data files, and store personal settings. The MD provides a front-end framework for embedding some of the application mechanisms and interfaces, and allows the user virtual access to Grid resources from other computational nodes.

### **5.2 Grid Authentication**

Access to the CrossGrid testbed is based on Globus GSI. GSI [30] uses public key encryption, X.509 certificates, and the Secure Sockets Layer (SSL) [31] communication protocol. Extensions to these standards have been added for single sign-on and delegation. The GSI provides a delegation capability: an extension of the standard SSL protocol which reduces the number of times the user must enter his pass-phrase. If a Grid computation requires that several Grid resources are used (each requiring mutual authentication), or if there is a need to have agents (local or remote) requesting services on behalf of a user, the need to re-enter the user's pass phrase can be avoided by creating a proxy.

### **5.3 Virtual Node Browsing**

The MD uses the Roaming Access Server (RAS) package, which provides a set of back-end portal services that are used by the MD front-end. RAS is also responsible for communicating with other modules, and offers a set of services such as an LDAP manager service responsible for storing the user profiles, a job submission service that provides an

interface that makes accessible the submission mechanisms, an scheduling agent, and a session manager service responsible for managing application-user session.

#### 5.4 Grid Data Transfer

Data transfer in the testbed is based on Globus GridFTP [32], a common data transfer and access protocol that provides secure, efficient data movement in Grid environments. We have registered Amsterdam and Leiden sites for testing secure GridFTP medical data transfer [Fig. 6].

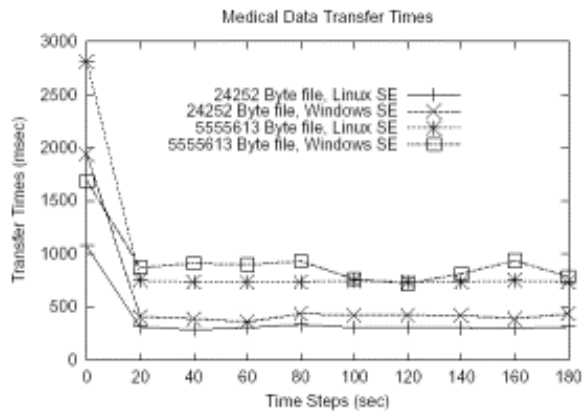
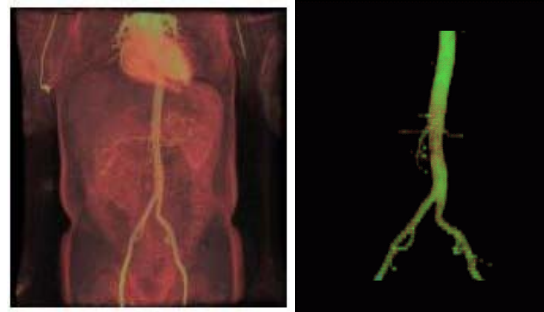


Figure 6. GridFTP transfer times to roaming nodes via the MD

We performed a set of experiments with the transfer of a segmented medical datasets, ranging from 24252 to 5555613 Byte loads, from registered storage elements to computing elements, as well as to roaming access nodes running both Linux and Windows NT acting as temporary storage elements, and back. We executed sets of image transfers via the MD grid portal every 20 seconds, and recorded on-the-wire transfer times, using a monitoring facility provided by the portal grid-FTP implementation. When comparing the transfer times of the data, at time steps of 20 seconds, we found that average transfer times to both Linux and Windows roaming storage elements running nodes, once taking into account the Globus caching mechanism, did not vary much above 200 milliseconds for the smaller size files and no more than 350 milliseconds for the larger size files, which we believe should be acceptable for the intended users.

#### 5.5 Medical Image Segmentation

Medical images are acquired in a scanner, e.g. by Magnetic Resonance Angiography. The raw data is stored in a database for further analysis. Next, using advanced image segmentation techniques, relevant 3D structures such as arteries, are extracted from the raw



data [33][Fig 7].

Figure 7. Segmentation of medical images

The segmentation software, originally developed for Windows environments, has been ported to Linux, to make it available to the testbed.

#### 5.6 Medical Image Visualization, Bypass Creation, and LBM Mesh Creation

The segmentation step is then connected to the DesktopVRE-based reconstruction of a 3D model of an artery. A geometrical modeling tool allows the interactive manipulation with 3D geometry and procedures, such as the clipping operation, editing of LBM mesh, creation/modification of problematic areas and interactive placement of a bypass. The LBM mesh editing compound of the DesktopVRE also allows indicating boundary conditions.

#### 5.7 Simulation Job Submission

Within the MD, application-specific information can be described using XML schemas. In order to integrate our visualization libraries into the computational grid testbed, we created and posted application XML schemata for job submission, to be dynamically linked to the MD via a job submission wizard. Then XSLT is used in order to transform the schemas into appropriate XHTML. MD sends the job request to the RAS, which then is sent to a job submission service, which then sends the job to a RB and logs all operations. The RB starts a job on the target CE. Before the job is started, though, a job-submission-script downloads all necessary files for simulation from a virtual node.

## 5.8 Simulation Job Monitoring

Grid Monitoring in CrossGrid includes services for application monitoring, as well as services for monitoring instruments and infrastructure. Application monitoring is substantially different from monitoring infrastructure and instruments, so separate approaches are offered, with application monitoring aimed at observing a particular execution of an application. The collected data is useful for tools for application development support, which are used to detect bugs, bottlenecks or just visualize the application's behavior, in the context of a particular execution.

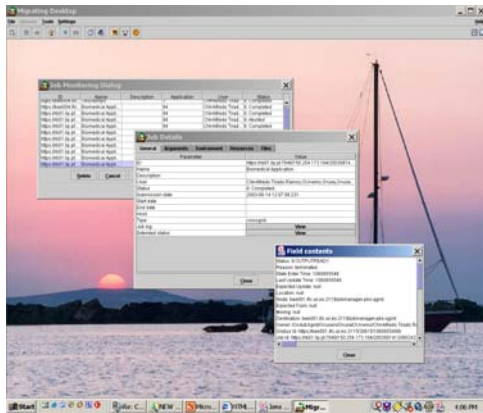


Figure 8. Simulation Job Monitoring within the MD

For our purposes, we use initially the MD capabilities for monitoring job submission. MD allows the user to restore settings, submit and monitor Grid jobs. We are also currently in the process of integrating the infrastructure and more advanced CrossGrid application monitoring toolbased on OCM-G [34], to allow for more interactive usage of collected information.

## 5.9 Blood Flow Visualization

The Grid Visualization Kernel (GVK) [28], addresses the combination of grid applications and corresponding visualization clients on the grid. While grids offer a means to process large amounts of data across distant resources, visualization aids in understanding the meaning of data. For this reason, the visualization capabilities of GVK are implemented using Globus services, thereby providing grid visualization services via dedicated interfaces and protocols while at the same time exploiting the performance of the Grid for visualization purposes. A

resource intensive module of the visualization pipeline is instantiated on a high-performance computer. Then, the VTK [35] pipeline on a graphics workstation connects (via re-direction through the GVK portal service) to this module, uses the power of the high-performance computer to generate the visual results, and downloads them to the visualization device.

We created links within the MD for initialization of both the GVK client and server startup applications, and experimented with rendering the flow both remotely and locally in the access storage element. This way, GVK remote visualization and local rendering are fully linked via the MD.

## 6. System Usage Validation

In order to ensure the successful adoption of VR-based PSEs, the context in which the system will be used has to be considered in system design and development. We have designed and performed contextual analysis to validate the VRE system within its target user base. A set of semi-structured interviews were coupled with observation in an ethnographic approach to requirements gathering in the daily work environment of (interventional) radiologists and vascular surgeons [36].

After heuristic usability evaluation of the VRE system, a need was identified to investigate the demands of real end-users. The study aimed to aid the direction of the VRE's future development by answering a number of questions about the real-life tasks and the context the VRE should support, functionality of such a system and its degree of immersiveness. Nine hospitals were involved in the study, both academic and non-academic. Fourteen specialists participated in the study, of which seven (interventional) radiologists and seven vascular surgeons were individually interviewed and/or observed. A number of radiologists, vascular surgeons, other medical specialists, technicians, nurses and assistants were observed during vascular meetings, surgery and informal discussions on scan assessment. The observed participants were usually unaccustomed to working with VR systems, but had experience in computing.

Most physicians were interested in new technology and were sometimes involved in technological research projects. An unexpected finding was the identification of an extra potential user group, the technicians who currently use diagnosis and planning systems to prepare scan images for radiologists and surgeons, the assessment of these images can then be performed as

fast as possible by experts. In some cases, technicians and radiologists performed the first assessment of these images together. Technicians form a prospective end-user group for the VRE.

## 7. Discussion and Prospective

The Grid-based simulation of vascular reconstruction introduced in this paper is a framework for rapid prototyping of an exploration environment that permits a user to explore interactively the visualized results of a simulation and manipulate the simulation parameters in near real-time, where a pre-operative planning of a vascular reconstruction procedure is a test case for making experiments.

The central idea of the VRE system is to use a set of hardware and software resources available on the Grid for building a specific framework to support vascular surgeons in their pre-operative decision-making. With such a system, medical experts will be able to prepare for certain interventions in the vascular system, using a combination of advanced simulation and visualization tools.

There is currently a lot of effort going on to define standard "inter-Grid" protocols to create something more than a set of non-interoperable distributed systems, which may be used to build Grid-based PSEs. The Open Grid Services Architecture (OGSA) [37] is one of the results of this effort. In OGSA, the primary purpose is to extend the Web Services to include Grid concepts, and to manage the creation and termination of resources as services [38]. In OGSA, the main focus is on the definition of abstract interfaces that allow services to cooperate without too much concern about the actual protocols being used. As such OGSA is a very general approach; it does not distinguish interactive services from batch services, and does not provide special support for near-real time interactive systems. This is left to higher level services build on top of it. One of the existing solutions for distributed and interactive simulation systems that may be used as a framework for interactive PSEs is the High Level Architecture (HLA) [39]. The existing HLA implementations are based on Common Object Request Broker Architecture (CORBA) middleware [40] and does offer many features for developers of interactive and distributed applications. However, existing implementations lack the flexibility we believe essential for Grid-based interactive applications, and the intelligent support required for interactive, near real-time systems. So far, we have found that the current OGSI implementation of the OGSA Registry PortType, even though unstable, works well for simple

ServiceData queries. In the near future, we plan to investigate the use of OGSA discovery topologies and intelligent searching agents for more complex dynamic discovery within HLA.

We are currently working on an approach for extending existing HLA-based PSE frameworks for interactive simulations to Grid services [41][42], based on our ISS-Conductor [43], an agent-oriented software architecture based on HLA for implementing and interconnecting distributed interactive simulation components [44][45]. In the ISS-Conductor, we use a layered interconnection mechanism: at the lower-level, messages between modules are carried by Communication Agents (ComAs), and at the higher-level, application logic is controlled by Module Agents (MAs). The software bus is normally the run-time infrastructure of the communication middleware adopted by the ComAs. The interaction scenarios between modules are represented as knowledge bases, which can be bound to MAs at run-time. ISS-Conductor is part of Polder [46], a distributed computing environment built by the University of Amsterdam.

The requirements gathered from the contextual usability analysis led to the conclusion that under well-defined conditions, a desktop version of our VRE PSE might be an alternative solution to the full immersive VRE. The identified usability issues for a fully immersive prototype, coupled with the needs, requirements, and real-life environment of the end-users gives us guidelines for the development of a PSE system based on a semi-immersive desktop VR environment [47][48][Fig 8].



Figure 8. The Desktop VR based PSE



## 8. Final Remarks

Among the lessons learned we found that configuration issues within Grid technologies in these early times of Grid development are not trivial. The functionality between packages and components in this early versions of the support technologies seemed to be prone to instability and synchronization issues, particularly when relating to large numbers of components and partners, such as is the case in projects like the CrossGrid.

For our next development steps, we will work on full integration with CrossGrid replication services, as well as experiment with advanced MPI support and monitoring functionalities that will allow us to fine-tune, monitor on the fly, and predict better performance results for our solver, on-line.

## 9. Acknowledgements

This research is sponsored partially by the EU CrossGrid Project IST-2001-32243 and the Dutch NSF Token 2000 project "Distributed Interactive Medical Exploratory for 3D Medical Images". The authors would like to thank the Dr. Elena Zudilova and Henriette Cramer for their contribution to the user evaluation and the European CrossGrid team for their software support and usage of the CrossGrid testbed.

## 10. References

- [1] P.M.A. Sloot: Simulation and Visualisation in Medical Diagnosis: Perspectives and Computational Requirements, in A. Marsh; L. Grandinetti and T. Kauranne, editors, *Advanced Infrastructures for Future Healthcare*, pp. 275-282. IOS Press, 2000.
- [2] Houstis E.N., Rice J.R., Weerwama S., Papachio P., Wang K. Yang and Gaitatzes M.: *Enabling Technologies for Computational Science – Frameworks, Middleware and Environments*, Chapter 14, pages 171-185. Kluwer Academic Publishers, 2000.
- [3] P.M.A. Sloot, G.D. van Albada, E.V. Zudilova: Grid-based Interactive Visualisation of Medical Images, in S. Nørager, editor, *Proceedings of the First European HealthGrid Conference*, January, 16th-17th, 2003, pp. 57 - 66. Commission of the European Communities, Information Society Directorate-General, Brussels, Belgium, 2003.
- [4] <http://www.science.uva.nl/research/scs/HotResults/>
- [5] World Health Organisation, 2002, *The World Health Report 2002, Reducing risks, promoting healthy life*. <http://www.who.int/whr/en/>

- [6] *Cleveland Clinic Heart Book*, Hyperion, 1st Edition, 2000.
- [7] Belleman R.G., Sloot P.M.A.: *Simulated Vascular Reconstruction in a Virtual Operating Theatre*. CARS 2001 Conference, Berlin, Germany, June 2001
- [8] Zajac K., Bubak M., Malawski, M. Sloot, P.M.A. Tirado-Ramos A., Zhao Z. *A Proposal of the Services For Managing Interactive Grid Applications in:* Bubak M., Noga M., Turala M: *Cracow Grid Workshop '02* December 11-14 2002:155-163.
- [9] A.M. Artoli, *Mesoscopic Computational Haemodynamics*, PhD. Thesis, Univeristy of Amsterdam, The Netherlands, 2003.
- [10] A.M. Artoli, A.G. Hoekstra, P.M.A. Sloot, *Simulation of a systolic cycle in a realistic artery with the Lattice Boltzmann BGK method.*, *Int. J. Mod. Phys. B*, Vol. 17, Nos. 1-2 (2003) 95-98.
- [11] A.M. Artoli, A.G. Hoekstra, P.M.A. Sloot, *Mesoscopic simulations of systolic flow in the human abdominal aorta*, accepted for *J. Biomech.*
- [12] S. Succi, *The Lattice Boltzmann Equation for fluid dynamics and beyond*, Oxford Science Publications, Clarendon Press, 2001.
- [13] Artoli A.M., Hoekstra A.G., Sloot P.M.A.: *3D Pulsatile Flow with the Lattice Boltzmann BGK Method*, *International Journal on Modern Physics C*, October 2002; 13:8.
- [14] Kandhai B.D., Koponen A., Hoekstra A.G., Kataja M., Timonen J. and Sloot P.M.A.: *Lattice Boltzmann Hydrodynamics on Parallel Systems*, *Computer Physics Communications*, 1998; 111: 14-26.
- [15] Belleman R.G., Stolk B., de Vries R.: *Immersive Virtual Reality on commodity hardware*, *Proceedings of the 7th annual conference of the Advanced School for Computing and Imaging*, May 2001: 297-304.
- [16] The University of Amsterdam Distributed Real-time Interactive Virtual Environment (UvA-DRIVE) <http://www.science.uva.nl/~robbel/DRIVE/>
- [17] Zudilova E.V., Sloot P.M.A., Belleman R.G.: *A Multimodal Interface for an Interactive Simulated Vascular Reconstruction System*, *Proc. of the IEEE International Conference on Multimodal Interfaces*, Pittsburgh, Pennsylvania, USA, October 2002: 313-319.
- [18] Belleman R.G., Kaandorp J.A., Dijkman D., Sloot P.M.A.: *GEOPROVE: Geometric Probes for Virtual Environments*, *Lecture Notes in Computer Science*, nr 1593 pp. 817-827. Springer-Verlag, Berlin, April 1999. ISBN 3-540-65821-1.

- [19] <http://www.evl.uic.edu/pape/CAVE/>
- [20] <http://www.eu-crossgrid.org/>
- [21] <http://www.dante.net/server.php?show=nav.007>
- [22] Data Management in an International Data Grid Project, Wolfgang Hoschek, Javier Jaen-Martinez, Asad Samar, Heinz Stockinger, Kurt Stockinger, IEEE/ACM International Workshop on Grid Computing Grid'2000 - 17-20 December 2000 Bangalore, India
- [23] Globus: A Metacomputing Infrastructure Toolkit. I. Foster, C. Kesselman. Intl J. Supercomputer Applications, 11(2):115-128, 1997.
- [24] <http://grid-deployment.web.cern.ch/grid-deployment/cgi-bin/index.cgi?var=homepage>
- [25] The Anatomy of the Grid: Enabling Scalable Virtual Organizations. I. Foster, C. Kesselman, S. Tuecke. International J. Supercomputer Applications, 15(3), 2001.
- [26] MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface. N. Karonis, B. Toonen, and I. Foster. Journal of Parallel and Distributed Computing, 2003.
- [27] <http://ras.man.poznan.pl/crossgrid/>
- [28] Dieter Kranzlmüller, Paul Heinzleiter and Jens Volkert: <http://grid.cesga.es/eabstracts/kranzlmuller.pdf>
- [29] Tirado-Ramos A., Ragas H., Shamonin D., Rosmanith H., Kranzlmüller D.: Integration of Blood Flow Biomedical Application Grid Visualization Kernel via a Grid Portal, Second European Across Grids Conference, Nicosia, Cyprus, Jan. 28-30, 2004.
- [30] Security for Grid Services. V. Welch, F. Siebenlist, I. Foster, J. Bresnahan, K. Czajkowski, J. Gawor, C. Kesselman, S. Meder, L. Pearlman, S. Tuecke. Twelfth International Symposium on High Performance Distributed Computing (HPDC-12), IEEE Press, to appear June 2003.
- [31] <http://www.openssl.org/>
- [32] GridFTP Protocol Specification. W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder, S. Tuecke. GGF GridFTP Working Group Document, September 2002.
- [33] P. F. Lelieveldt, S. C. Mitchell, J. G. Bosch, R. J. van der Geest, M. Sonka, and J. H. C. Reiber, "Time continuous segmentation of cardiac image sequences using Active Appearance Motion Models," proc. IPMI, Lecture Notes in Computer Science, vol. 2082 pp.446-452, 2001.
- [34] Balis, B., Bubak, M., Funika, W., Szepieniec, T., and Wisniewski, R.: An Infrastructure for Grid Application Monitoring In: Kranzlmüller, D., Kacsuk, P., Dongarra, J., Volker, J. (Eds.): Recent Advances in Parallel Virtual Machine and Message Passing Interface, Proc. 9th European PVM/MPI Users' Group Meeting, Linz, Austria, September/October 2002, LNCS 2474, pp. 41-49, 2002
- [35] <http://public.kitware.com/VTK/>
- [36] Heuristic evaluation and context analysis to aid the development of the simulated vascular reconstruction system", H.S.M Cramer, V. Evers, E.V. Zudilova, and P.M.A. Sloot, accepted at the International Workshop Designing and Evaluating Virtual Reality Systems", 22-23 January 2004, University of Nottingham, UK.
- [37] The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. I. Foster, C. Kesselman, J. Nick, S. Tuecke, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.
- [38] Grid Service Specification. S. Tuecke, K. Czajkowski, I. Foster, J. Frey, S. Graham, C. Kesselman; Open Grid Service Infrastructure WG, Global Grid Forum, Draft 2, 7/17/2002.
- [39] Defence Modelling and Simulation Office (DMSO), <http://hla.dmsomil/>
- [40] Ryan C.O., and Levine D.L.: Applying a Scalable CORBA Events Service to Large-scale Distributed Interactive Simulations In: Proceedings of the 5th Workshop on Object-oriented Real-time Dependable Systems. Monterey, CA, 1999.
- [41] Zajac K., Tirado-Ramos A., Zhao Z., Sloot P., Bubak M: A Grid Service Approach to HLA-based Distributed Simulation Frameworks for Interactive Problem Solving Environments, First European Across Grids Conference, Santiago de Compostela, Spain, 13-14 Feb, 2003.
- [42] A. Tirado-Ramos; K. Zajac; Z. Zhao; P.M.A. Sloot; G.D. van Albada and M. Bubak: Experimental Grid Access for Dynamic Discovery and Data Transfer in Distributed Interactive Simulation Systems, in P.M.A. Sloot; D. Abrahamson; A.V. Bogdanov; J.J. Dongarra; A.Y. Zomaya and Y.E. Gorbachev, editors, Computational Science - ICCS 2003, Melbourne, Australia and St. Petersburg, Russia, Proceedings Part I, in series Lecture Notes in Computer Science, vol. 2657, pp. 284-292. Springer Verlag, June 2003. ISBN 3-540-40194-6.
- [43] Z. Zhao; G.D. van Albada; A. Tirado-Ramos; K. Zajac and P.M.A. Sloot: ISS-Studio: a prototype for a user-friendly tool for designing interactive experiments in Problem Solving Environments, in P.M.A. Sloot; D. Abrahamson; A.V. Bogdanov; J.J. Dongarra; A.Y. Zomaya and Y.E. Gorbachev, editors, Computational Science - ICCS 2003, Melbourne, Australia and St. Petersburg, Russia, Proceedings Part I, in series Lecture Notes in Computer Science, vol. 2657, pp. 679-688. Springer Verlag, June 2003. ISBN 3-540-40194-6.

[44] Zhao Z., Belleman R.G., van Albada G.D., Sloot P.M.A.: State Update and Scenario Switch in an Agent Based Solution to Constructing Interactive Simulation Systems, Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference, 2002

[45] Zhao Z., Belleman R.G., van Albada G.D., Sloot P.M.A.: AG-IVE an agent based solution to constructing Interactive Simulation Systems, Proceedings of the second inter-action conference of computational science (ICCS02), Amsterdam, NL, 2002

[46] Iskra K.A., Belleman R.G., van Albada G.D., Santoso J., Sloot P.M.A., Bal H.E., Spoelder H.J.W. and Bubak M.: The Polder Computing Environment, a system for interactive distributed simulation, Concurrency and Computation: Practice and Experience, (Special Issue on Grid Computing Environments) 2002; 14: 1313-1335.

[47] Poston T., Serra L.: Dextrous Virtual Work, Communications of the ACM, 39:5, pp. 37-45, May 1996.

[48] J.D. Mulder, and R. van Liere: The personal space station: Bringing interaction within reach. In S. Richer, P. Richard, and B. Taravel, editors, Proceedings of the Virtual Reality International Conference, VRIC 2002, pp. 73-81 (2002).

[49] <http://www.cs.vu.nl/das2/>

## 11. APPENDIX

### Initial Testbed Setup

We developed a full Problem Solving Environment for our cardiovascular reconstruction test case application, using the CrossGrid testbed for experimentation. We solved a number of technical integration problems related to blood flow visualization in the Grid-enabled local roaming machines, e.g., support for hardware acceleration, and we also created and posted on the web a set of biomedical application XML schemata for job submission to our solver, via the MD Job Wizard. We also integrated our DesktopVRE with our Grid-based visualization engine, within the testbed. We finally registered a few sites in Amsterdam and Leiden for testing GgridFTP data transfer, and performed a set of experiments as discussed in this paper. The experimental setup contained the following software and hardware components:

#### DAS2 SEs

The Distributed ASCI Supercomputer 2 (DAS-2) [49] is a wide-area distributed computer of 200 dual nodes. The machine is built out of clusters of workstations, which are interconnected by SurfNet, the Dutch university Internet backbone for wide-area communication, whereas Myrinet, a popular multi-Gigabit LAN, is used for local communication. We used a set of DAS-2 nodes as SEs for our experiments, in UvA, NIKHEF, (Amsterdam) and LUMC (Leiden), with the following configuration:

- EDG 2.0 release candidate with VDT-1.1.8-6, installed and configured manually instead of LCFG (because is a shared system).
- No R-GMA running, using the MDS interface on port 2135.
- Distributed ASCI Supercomputer 2 (DAS-2).
- Myrinet multi-Gigabit wide-area distributed computer of 200 Dual Pentium-III nodes.
- Fast Ethernet used as OS network.

Each node contains two 1-Ghz Pentium-IIIs, and at least 1 GB RAM. Also, at least 20 GByte local IDE disk with Myrinet and Fast Ethernet interface cards. The system runs on Linux 2.4.7-10.

### CrossGrid CEs

The CrossGrid international distributed test-bed shares resources across a number of sites in Europe, ranging from relatively small computing facilities in universities to large computing centers. The CrossGrid production test-bed, initially based on EDG 1.2.2 and 1.2.3 deployed in the production test-bed, has moved now to LCG-1 running RH Linux 7.3. We have used a number of resources from this test-bed, which is composed of 15 Computing Elements, 69 Worker Nodes, 14 Storage Elements, 115 CPUs and around 2.7TB Storage Capacity, with this capacity growing larger as more resources are added by the project partners.

### MD

We used the CrossGrid portal Migrating Desktop (MD), a local repository for remote navigation, data transfer and application initialization. The configuration and requirements for launching the MD are in general a Web Browser, Java Plug-in 1.4 or newer with JAR cache disabled, firewalls open for 8080 port, pool 13000-17000 and port 2811 open in both directions between local workstation and remote SE, and a set of valid X509 certificate credentials.