VLAM-G: A Grid-based virtual laboratory

H. Afsarmanesh[a], R.G. Belleman[a], A.S.Z. Belloum[a], A. Benabdelkader[a], J.F.J. van den Brand[b,c], G.B. Eijkel[d], A. Frenkel[a], C. Garita[a], D.L. Groep[b], R.M.A. Heeren[d], Z.W. Hendrikse[a,*], L.O. Hertzberger[a], J.A. Kaandorp[a], E.C. Kaletas[a], V. Korkhov[a], C.T.A.M. de Laat[a], P.M.A. Sloot[a], D. Vasunin[a], A. Visser[a] and H.H. Yakali[a]

[a]*Computer Science Department, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands*

[b]*NIKHEF, P.O. Box 41882, 1009 DB Amsterdam, The Netherlands*

[c]*Vrije Universiteit, de Boelelaan 1081, 1081 HV Amsterdam, The Netherlands*

[d]*FOM Institute AMOLF, Kruislaan 407, 1098 SJ Amsterdam, The Netherlands*

The Grid-based Virtual Laboratory AMsterdam (VLAM-G), provides a science portal for distributed analysis in applied scientific research. It offers scientists remote experiment control, data management facilities and access to distributed resources by providing cross-institutional integration of information and resources in a familiar environment. The main goal is to provide a unique integration of existing standards and software packages.

This paper describes the design and prototype implementation of the VLAM-G platform. In this testbed we applied several recent technologies such as the Globus toolkit, enhanced federated database systems, and visualization and simulation techniques. Several domain specific case studies are described in some detail. Information management will be discussed separately in a forthcoming paper.

1. Introduction

Recent developments in integrating parallel and distributed computing, combined with improvements in overall network bandwidth have made it possible to add a new dimension to distributed computing: the Grid [1]. Nowadays, an impressive number of workstations and super computers can be connected in an efficient way, offering users unprecedented computational power. The part of the scientific community that relies on high-performance and/or high-throughput computing machinery needs an ever growing amount of CPU-cycles. Moreover, huge amounts of data are generated from experiments conducted world-wide. Scientists want to be able to smoothly connect the data producing equipment to the compute, storage and visualisation centers, which are usually not on the same site. Collaborative data processing from big science laboratories, such as CERN, to home institutes and universities is a necessity as well.

The Grid-based Virtual Laboratory AMsterdam (VLAM-G) offers such a distributed analysis platform for applied experimental science [2]. It is distinguished by

- the development of specialized upper middleware, meant to bridge the gap between the applications and grid-services layer,
- unique federated information management provisions supporting/modelling experiments [3],
- means for resource integration & collaboration.

Therefore, the VLAM-G is *a science portal for remote experiment control and collaborative, Grid-based distributed analysis in applied sciences, using cross-institutional integration of heterogeneous information and resources*.

The VLAM-G design promotes the use of services offered by the Globus toolkit,[1] which is the *de facto* standard in Grid computing nowadays. Its 'bag of services' approach meets the requirements of the VLAM-G design very well. However, since the Globus toolkit constitutes a low-level middle-tier, an additional layer is needed to open up Grid/Globus technology to applications of scientists. This layer should hide them from low-level Grid issues, such as e.g. security, resource management and networking. The importance is un-

*Corresponding author: Z.W. Hendrikse.

[1] http://www.globus.org/.

Scientific Programming 10 (2002) 173–181
ISSN 1058-9244 / $8.00

derlined by the fact that scientists may benefit from geographically dispersed resources, thereby enhancing their experimentation capabilities considerably and by the fact that they form the most important group of Grid users thus far.

Domain specific application case studies have been developed for physics, bio-informatics, and medical and systems engineering. Bearing these application domains in mind, three different science portals are being designed and implemented simultaneously, sharing a common technology.

2. The VLAM-G toolkit

The bag of services offered by the VLAM-G toolkit may to some extent be used independently by a scientist to conduct his experiment(s). It provides a GUI, structures to encapsulate and interface to experiment specific software and hardware, data storage and visualisation systems. It offers an assistant to guide a scientist through the process of structuring his study by giving templates to outline the different steps to perform the entire data handling process. It uses the Grid services standardized by the Global Grid Forum (GGF) to tie everything together.

In these distributed environments advanced networks play an essential role. Access to the better than best effort Quality of Service (QoS) network resources is done via the appropriate Grid services. Portals are envisioned to encompass several organisational domains. Since it is very well possible that competing organisations will be using the same Grid infrastructure using their specific portals, support for authentication, authorisation, accounting (AAA) and security are essential.

2.1. The concept of a study

One of the fundamental challenges in experimental science is the extraction of useful information from large data sets. This triggers the need for cooperation of multi-disciplinary teams located at geographically dispersed sites. VLAM-G meets these challenges by:

- providing a collaborative application environment (for which new technology is under development) and
- integrating Grid (software) tools and packages in a unique way, resulting in an added value when compared to using these Grid services directly.

To achieve these goals, experiments are embedded in the context of a *study*. A study is about the meaning and the processing of data. It includes descriptions of data elements (meta-data) and process steps for handling the data. A study is defined by a formalized series of steps, also known as process flow, intended to solve a particular problem in a particular application domain. The process steps may generate raw data from instruments, may contain data processing, may retrieve and store either raw or processed data and may contain visualisation steps. A Process Flow Template (PFT) is used to represent such a formalized workflow. The information management issues behind this model will be published in a forthcoming paper.

A study is activated by instantiating such a PFT. This instantiation will be called a process flow instantiation (PFI). A user is guided through this PFI using context-sensitive interaction. An example of a PFT is shown in Fig. 3. This particular flow describes the MACS application, which will be discussed in Section 3.1.

The context of the operational steps (represented by ovals) is provided by entity meta data (represented by boxes). For one-time experiments, a trivial process flow should be defined, consisting of one operational step only, and limited contextual meta data associated with the result.

A careful analysis of the application domains (see Section 3) driving the VLAM-G development has enabled us to identify generic aspects in these application domains. Combining these generic features with those of specific case-studies, it became possible to develop a database model for the Experiment Environment (EE). The EE data model underlies the definition of a PFT and allows for a random ordering of processes and data elements (which enables one to construct an arbitrary PFT).

The process steps in the PFT represent the actual data flow in an experiment. This usually entails the data flow stemming from an instrument, through the analysis software to data storage facilities. Consequently, an experiment is represented by a data flow graph (DFG). This DFG usually contains experiment specific software entities as well as generic software entities. From now on we will call these self-contained software entities modules. Examples of experiment specific modules are those meant for instrument control. An example of members of the generic class are the visualisation modules, which visualize incoming data formatted in the NetCDF format. The modules constituting the data-flow have a strong analogy to those used in IRIS explorer.[2]

A scientist may connect these modules to conduct a specific experiment. Separate modules may be grouped together, to form a larger aggregate module. Construction of an experiment topology takes place in a GUI, as will be explained in Section 2.2. There it will also be shown how this topology is executed on the resources offered by the Grid, using the services of the Globus toolkit.

Our first prototype implementation integrates various filters developed for physics experiments, modules for accessing High Performance Storage Systems (HPSS) and databases, modules for control of external devices and modules specifically geared for visualization tasks. These modules are stored in the module repository, see Fig. 1.

2.2. *The VLAM-G architecture*

We define components as the various services offered by the VLAM-G toolkit. The interaction of the various components is depicted in Fig. 1. Six main components can be distinguished:

– a GUI,
– a run time system (RTS),
– a collaboration system,
– an assistant,
– a module repository and
– VIMCO, the Virtual laboratory Information Management for COoperation.

The GUI and the RTS form the core components and constitute the interface to the underlying Grid services. The GUI is the only component that users interact with. Once an experiment topology has either been loaded or composed, its associated data-flow graph (DFG) can be dispatched to the RTS using XML. The RTS takes care of the scheduling, instantiation and monitoring of the modules comprising the experiment, thereby making extensive use of Globus services such as the gatekeeper and the Grid Information Service (GIS).[3] In the future it is planned to use the Globus scheduling services of DUROC too. By using these existing components from the Grid-services layer, we take full advantage of many Grid-related developments. More details on the instantiation of the DFG can be found in the next section.

The assistant supports the composition of an experiment by providing templates and information about previously conducted experiments. The collaboration system provides communication mechanisms for collaboration during an experiment.

– *Collaboration:* The collaboration system will offer audio and video communication between participating scientists by supporting amongst others a whiteboard for scientists and an experiment logbook. Therefore, the collaboration system will closely follow the developments in the ACCESS-Grid.[4]
– *The assistant:* The assistant is a subsystem that assists users during the design of an experiment, e.g., it may suggest module definitions or PFTs of previous experiments. Decisions of the assistant are supported by knowledge gathered from the RTS database (it may suggest previously conducted experiments) and the application database.

The VIMCO information management system provides access to the RTS database and the PFT databases (usually, each application domain has its own PFT database). It is important to point out that the PFT databases only store meta-data: the raw data coming from experiments is stored using high performance storage systems (HPSSs). Archiving and manipulation of these large data sets is facilitated by VIMCO as well. Finally, VIMCO enables cross-institutional data sharing and exchange, based on a federated database approach.

The RTS database stores topologies of experiments, user sessions and module descriptions. The PFT databases contain data models specifically designed for a particular application domain.

Since the RTS connects to the Grid services, the next section explains in more detail how an experiment is actually executed.

2.3. *The VLAM-G RTS*

The RTS manages the topology representing an experiment, which it receives from the GUI. The RTS enables data transfer between modules. These data-streams are established between appropriate modules only, using typed streams. Not only are modules in-

[2] http://www.nag.co.uk/Welcome_IEC.html.

[3] For information on GIS, the reader is referred to http://www.globus.org/mds/.

[4] http://www.accessgrid.org/.

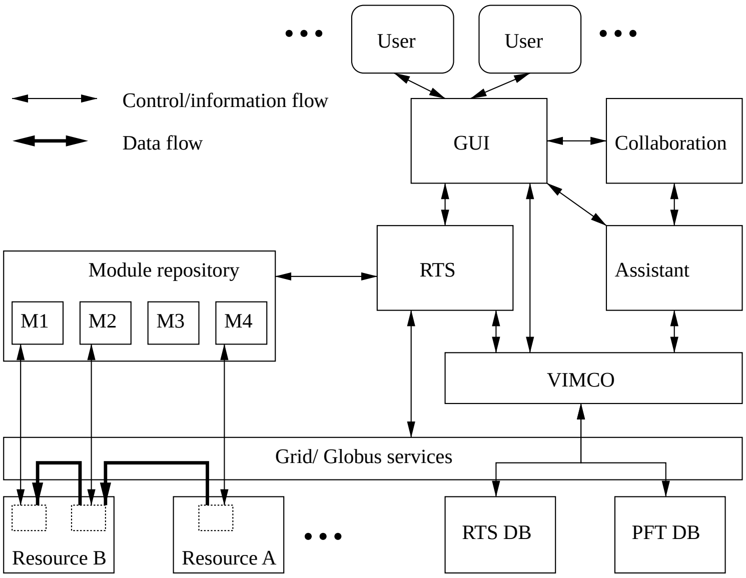

Fig. 1. The components of the VLAM-G toolkit and their relations. It is also shown how an experiment topology is instantiated.

stantiated by the RTS, also parameter and state values may be changed, cached and or propagated.

From case studies it has been shown that at least three types of connectivity have to be supported:

– interactions among processing elements located on the same machine,
– interactions among two or more machines and
– interactions with external devices, where some of the processing elements are bound to a specific machine.

Moreover, practice has not only shown the need for one-to-one communications, but also for one-to-many (fan-out).

Communication is established using uni-directional ports. The sole use of unidirectional, typed streams assures only that proper connections can be established and intentionally precludes the possibility of control & monitoring communication. The latter is only allowed between the RTS and a module in the form of events that exchange state and parameter changes between the modules and the RTS.

The RTS consists of three components: the module launcher, the module connector and module controller, see Fig. 2. The *module launcher* initiates the execution of an experiment, which it receives via the GUI in XML. For the time being it makes use of the `globus-job-run` utility of the Globus toolkit, where the resources are statically specified. *The module connector* implements third party arbitration which builds the data communications channels between modules. For each data channel, the *module connector* selects a communication port (TCP ports) and sends specific commands to the producer and the consumer to initialize the communication. Thereafter, input and output ports may exchange data using the Grid FTP protocol. This choice is motivated by the fact that experiments require a reliable and effective data transfer mechanism, including features for high-performance parallel data transfers.

In this scheme, a module acts like a GridFTP server that presents input and output ports as files with appropriate permissions. Consequently, it may be connected by any GridFTP client, such as a Grid Security Infrastructure (GSI) enabled HPSS.

Finally, the *module-controller* monitors the state of modules, once they have been instantiated. Changes of state are relayed to the GUI. Parameter changes are communicated between the module-controller and the GUI as well.

We support two ways of communicating data among remote processing modules. The first way makes use of the *eXternal Data Representation* standard (XDR) to define a typed data stream. The second method uses an external data typing mechanism that supports data types provided by the application programmer. An example is the NetCDF format, which we use for visualisation purposes. Here the RTS uses plain GridFTP streams, that are compatible with standard C++ streams.

Modules are always active, i.e. once instantiated they enter a continuous processing loop, blocked if waiting for arrival of data on input ports.

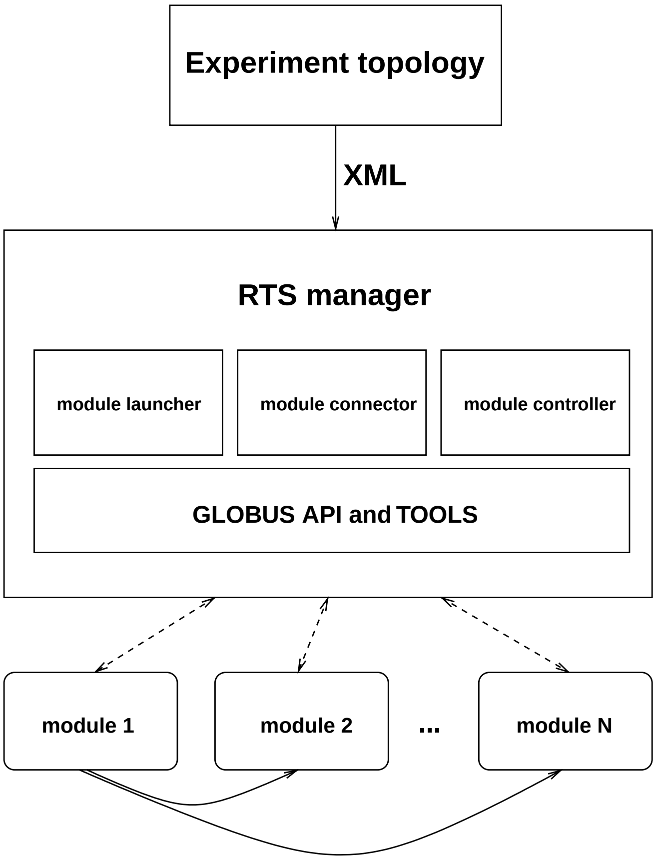

Fig. 2. The VLAM-G RTS.

3. The VLAM-G science portals

To guide the development of the VLAM-G platform, five application cases have been selected from different domains in applied science: chemo-physical analysis of material surfaces, simulated bio-medical vascular reconstruction using immersive visualisation techniques, correlation of gene expression data from heterogeneous databases, and a simulation/analysis environment for road-traffic measurement data. As a fifth, we created a small distance learning portal allowing the development group to do proof of concept studies. These case studies each use their own portal, that provides them with a localized set of PFTs and appropriate supportive tooling. Each case study highlights a different aspect of the VLAM-G environment. Steering of remote measurement apparatus and value-adding by combining data sets from different sources is demonstrated by MACS, see Section 3.1; the simulation and system engineering case shows the interplay between commodity analysis environments and the VLAM-G middleware; the bio-informatics portal stresses the workflow support provided by the PFT; and the surgical planning system demonstrates the usefulness of man-in-the-loop type dynamic data exploration environments. The application cases and science portals are described in detail below.

3.1. A virtual material analysis laboratory

The use of advanced physical methods to study the properties of material surfaces has proven to be a powerful tool, for both fundamental and applied research. Fields as diverse as art conservation, cancer therapy and cosmology all benefit strongly from the availability of sub-micron techniques that can elucidate the spatial organization of elements and (macro-)molecules in complex (organic) surfaces. A selection of these methods is embedded in a section of the VLAM-G referred to as “Material Analysis of Complex Surfaces” or ‘MACS’.

The MACS center comprises three large instruments: an imaging infra-red Fourier-transform spectrometer (FTIR [4]), a 4 MeV Nuclear Microprobe for elemental analysis [5], and an imaging mass spectrometer.

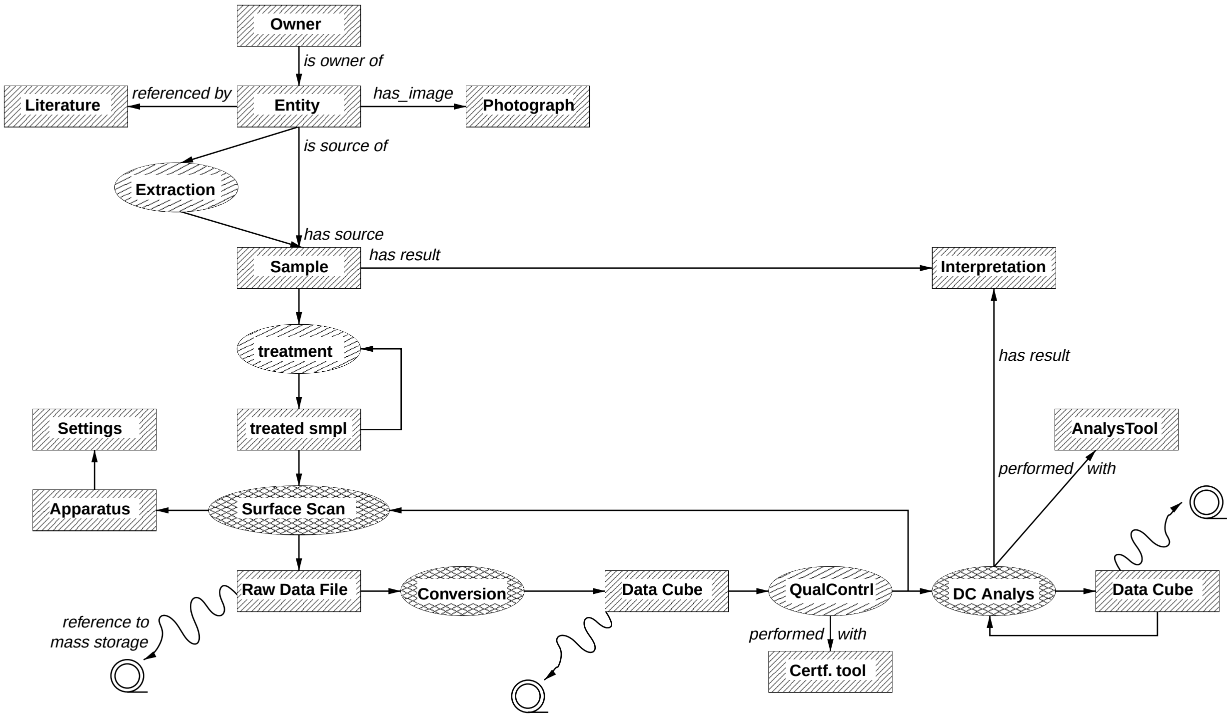

Fig. 3. Typical steps in a surface-analysis study, as represented by the MACS [8] process flow. The hatched rectangles represent meta data associated with objects (either physical objects or bulk data on mass storage). Hatched ovals represent operations: cross-hatched ovals are performed using the VLAM-G Run Time System, the single-hatched ovals represent meta data associated with manually performed process steps.

These devices produce largely complementary data; FTIR imaging is primarily used to elucidate and classify the spatial distribution of chemical functionalities (IR functional groups), whereas PIXE [6,7] and specific nuclear reactions quantitatively identify the elemental composition of the sample. Existence and spatial distribution of (large) macromolecules and polymers may be identified with imaging mass spectrometry. Correlation of these complementary data would significantly add to the scientific merit of these experiments.

VLAM-G provides all these experiments with a uniform environment for data analysis, since most if not all experiments share a similar PFT (in this particular case preparation of a sample, surface analysis, visual inspection and possibly re-analysis). A simplified version of such a data flow is shown in Fig. 3.

The processes in the PFT may represent the data gathering phase of any of the devices mentioned, either by real-time access to these resources or by way of human intervention. The subsequent data analysis can be modelled as a directed data-flow, that may be processed by the VLAM-G RTS.

All resulting data, n-dimensional 'data cubes', are represented in a common standard format (NetCDF-3 with locally established conventions). Generic visualization modules for such data cubes are provided as part of the core VLAM-G module set.

3.2. Simulation and systems engineering

Evaluation of communication systems in various ICT application areas, like traffic management strategies and telematic services, can greatly profit from simulations. Although field tests provide an opportunity to measure the benefits of new applications in real-world environments and convey vital information for system design and evaluation, the scale and the range of the conditions for which the system performance can be tested is severely limited in practice. Using simulation, both key operations and performance issues may be investigated under a wide range of conditions.

Available compute power is the determining factor with respect to the level of detail that can be simulated and, consequently, lack of it leads to more abstract models. With the resources provided on Grid, we can afford more detailed simulations.

In addition to simulation, research on traffic management and the subsequent analysis generates significant amounts of data. In order to provide a comprehensive environment for data analysis, an interface between VLAM-G and Matlab is being developed in addition to generic traffic analysis and validation tools [9, 10]. In this environment, users will be able to start their VLAM-G applications from Matlab and get their simulation/experimentation results directly back into their Matlab environment, where they may do subsequent analysis.

3.3. Information management for gene expression studies

One of the application domains in VLAM-G is biosciences. The systematic growth of research efforts in biosciences resulted in vast amounts of data scattered all around the world. The major problems in this domain are related to the management of distributed and heterogeneous experimental data/information. This heterogeneity is due to the wide variety of genomic information types and various formats used to store this information. Moreover, on the single experiment level, all information about the experiment steps and the results must be stored for analysis and reproducibility of the experiment. On a higher level, experimental information from one area in biosciences must be linked to information from other areas (e.g. gene sequence, expression, metabolic pathways, etc.), so that information can be extracted from the scattered collected data. To support experimentation in biosciences, infrastructures for systematic definition and execution of experiments and federated management of information are required. Currently the focus is on the information management of DNA micro-arrays. The results of this case study will be further extended to the other areas of biosciences.

DNA micro-arrays allow genome-wide monitoring of changes in gene expressions, where changes may be induced by response to some stimuli. Depending on the organism and the exact design of the experiment, the number of useful data points produced per experiment ranges from around 12000 for a simple organism like yeast upwards to 300000 for a complex organism such as human.

In order to support the information management requirements for the DNA micro-array case study, a gene expression database called EXPRESSIVE has been developed [11]. EXPRESSIVE database model complies with the Minimum Information About a Microarray Experiment (MIAME)[5] standard. It allows to represent any kind of expression pattern, and is scalable and open to support future requirements. EXPRESSIVE provides:

- Definition and management of information to support the storage and retrieval of the steps and information involved in DNA microarray experiments, to support investigation and reproducibility of experiments using the VLAM-G.
- Storage and retrieval of results from DNA microarray experiments (both raw and processed) using the VLAM-G, to facilitate both analysis and scientific collaboration with external partners.

Summarizing, the DNA micro-array case does not only benefit from the information management functionality offered by VLAM-G, but also from the friendly and uniform user interfaces. Moreover, it can share a wide variety of attached software tools for analysis and visualization, and Grid-enabled resources, such as distributed and high-performance computing facilities, and high-speed and secure networking.

3.4. Interactive simulated vascular reconstruction

The purpose of vascular reconstruction is to redirect and augment blood flow or perhaps repair a weakened or aneurysmal vessel through a surgical procedure. The optimal procedure is often obvious but this is not always the case, for example, in a patient with complicated or multi-level disease. Pre-operative surgical planning will allow evaluation of different procedures *a priori*, under various physiological states such as rest and exercise, thereby increasing the chances of a positive outcome for the patient. The aim of this case study is to provide a surgeon with an environment in which he/she can explore the effect of a number of different vascular reconstruction procedures before it is put to practice. Our approach combines parallel flow simulation, interactive virtual reality and high performance computing techniques into an interactive dynamic exploration environment that together allows for human-in-the-loop types of experimentation [16].

Generic visualization modules in the VLAM-G allow surgeons to inspect patient specific computerized tomographic angiography (CTA) and/or magnetic resonance angiography (MRA) data using an interactive stereoscopic virtual environment. Through the VLAM-G, this medical data can be processed on remote high performance computing (HPC) systems to simulate blood flow through the patient's vascular system. High speed networking initiatives such as the GigaPort[6] project

[5] http://www.mged.org/Workgroups/MIAME/miame.html.

[6] http://www.gigaport.nl/.

allow hospitals to make interactive use of HPC techniques without having to invest in these (often expensive) machines [14].

The simulated vascular reconstruction operating theatre provides visualisation and interaction methods to simulate a vascular reconstruction procedure and visualize the effect of that procedure on a patient's blood circulation in real time. The environment can be used in a CAVE [12] but also on low cost commodity PC hardware in conjunction with a projection display and tracking hardware [15], allowing the system to be used in the radiology department. Multi-modal interaction methods such as speech recognition, hand gestures, direct manipulation of virtual 3D objects and measurement tools allow researchers to explore simulation and visualization results [13].

This environment will be validated through a comparison of fluid flow simulation results and the results of other simulation methods as well as *in vivo* measurements of blood flow through phantom structures and pre- and post-operative MRA scans.

3.5. Distance learning proof of concept setup

A small setup consisting of a real mechanics physics experiment controlled by a computer is used to test all concepts of the VLAM-G environment. The setup contains real time control parts, data taking modules, data analysis, visualization and interpretation software and cameras for remotely inspecting the experiment when measurements are ongoing. This setup uses several different computers which have to work together to complete the experiment. Apart from providing for a proof of concept in the VLAM-G, the portal may also be used to provide students in high schools with access to advanced environments. Also security and authentication/authorization issues can be tested in this environment.

4. Conclusions

We designed an upper middleware layer to establish a connection between experiments and the (Globus) Grid services layer. The services offered by the VLAM-G middleware shield users from the complexity of binding different infrastructures together so that they can concentrate on their profession. Moreover, it offers scientists easy access to Grid resources. The prototype implementation of this concept has been shown to work very well for four separate scientific domains: bioinformatics, systems engineering, physics and distance learning.

It has also been shown that the VLAM-G facilitates integration of existing components:

– it elaborates on existing (Globus) Grid technology,
– it allows scientists from different disciplines to reuse generic modules and
– it abstracts the usage of advanced networks and experiment infrastructure.

.

Particular attention has been paid to construct a user friendly environment, using context sensitive information, automated resource management and assistance during the design process. We have profited from the bag of services approach from the Globus toolkit. These services are useful for experimental scientists, if an additional upper middleware layer is provided. The various application domains discussed in this paper turned out to have enough common ground to successfully design and implement this additional upper middleware layer.

Acknowledgments

This work was supported by an ICES/KIS grant from the Dutch ministry of education and sciences (OC&W) and the ministry of economic affairs (EZ). It is also part of the research program of the Foundation for Fundamental Research of Matter (FOM), which is financially supported by the Netherlands Organization for Scientific Research (NWO). This project includes software developed by and/or derived from the Globus project (http://www.globus.org/).

References

[1] I. Foster, C. Kesselman and S. Tuecke, *The anatomy of the Grid: Enabling scalable virtual organizations*, The International Journal of Supercomputer Applications, 2001.

[2] A. Belloum, Z.W. Hendrikse, D.L. Groep, E.C. Kaletas, A.W. van Halderen and L.O. Hertzberger, The VL-Abstract-Machine: a Data & Process Handling System on the Grid, in *Proceedings of the 9th international conference,* HPCN Europe, 2001, pp. 81–93.

[3] C. Garita, H. Afsarmanesh and L.O. Hertzberger, The PRODNET federated information management approach for virtual enterprise support, *Journal of Intelligent Manufacturing* **12** (2001), 151–170.

[4] G.B. Eijkel, H. Afsarmanesh, D. Groep, A. Frenkel and R.M.A. Heeren, Mass spectrometry in the Amsterdam Virtual Laboratory: development of a high-performance platform for meta-data analysis, in *Proceedings of the 13th Sanibel Conference on Mass Spectrometry: informatics and mass spectrometry,* Sanibel Island, Florida, USA, 2001.

[5] R.D. Vis, J.L.A.M. Kramer, G.H.J. Tros, F. van Langevelde and L. Mars, The upgraded Amsterdam nuclear microprobe, in *Proceedings of the 3rd international conference on microprobes*, Uppsala, Sweden, 1992.

[6] S.A.E. Johansson and J.L. Campbell, *PIXE- A Novel Technique for Elemental Analysis*, ISBN-0 471 92011 8, 1988.

[7] S.A.E. Johansson, J.L. Campbell and K.G. Malmqvist, *Particle-Induced X-Ray Emission Spectrometry* (*PIXE*), S.A.E. Johansson, J.L. Campbell and K.G. Malmqvist, eds, John Wiley and Sons, ISBN-0 471 58944 6, 1995.

[8] A. Frenkel, H. Afsarmanesh, G.B. Eijkel and L.O. Hertzberger, Management for material science applications in a virtual laboratory, in *Proceedings of the 12th international conference on database and expert systems applications*, Munich, Germany -DEXA, 2001.

[9] A. Visser, A.J. van der Wees and L.O. Hertzberger, Discrete event modelling methodology for intelligent transport systems, in *Proceedings of the World congress on intelligent transport systems*, Torino, Italy, 2000, pp. 2016.

[10] A. Visser, H.H. Yakali, A.J. van der Wees, M. Oud, G.A. van der Spek and L.O. Hertzberger, *An hierarchical view on modelling the reliability of a dsrc-link for etc applications*, submitted to IEEE Transactions on Intelligent Transportation Systems, Technical Report CS-99-02, University of Amsterdam, 1999.

[11] E.C. Kaletas, H. Afsarmanesh and L.O. Hertzberger, *Virtual Laboratories and Virtual Organizations supporting biosciences*, to appear in the PRO-VE'02 3rd IFIP working conference on infrastructures for virtual enterprises.

[12] C. Cruz-Neira, D.J. Sandin and T.A. DeFanti, *Surround-Screen Projection-Based Virtual reality: The Design and Implementation of the CAVE*, SIGGRAPH '93 Computer Graphics Conference, ACM SIGGRAPH, 1993, pp. 135–142.

[13] R.G. Belleman, J.A. Kaandorp, D. Dijkman and P.M.A. Sloot, GEOPROVE: Geometric Probes for Virtual Environments, in: *High Performance Computing and Networking* (*HPCN'99*), P.M.A. Sloot, M. Bubak, A. Hoekstra and L.O. Hertzberger, eds, Springer-Verlag, Amsterdam, the Netherlands, 1999, pp. 817–827.

[14] R.G. Belleman and P.M.A. Sloot, The Design of Dynamic Exploration Environments for Computational Steering Simulations, in: *Proceedings of the SGI Users' Conference*, M. Bubak, J. Mościński and M. Noga, eds, Academic Computer Centre CYFRONET AGH, Kraków, Poland, 2000, pp. 57–74.

[15] R.G. Belleman, B. Stolk and R. de Vries, Immersive Virtual Reality on commodity hardware, in: *Proceedings of the seventh annual conference of the Advanced School for Computing and Imaging*, R.L. Lagendijk, J.W.J. Heijnsdijk, A.D. Pimentel and M.H.F. Wilkinson, eds, Advanced School for Computing and Imaging (ASCI), 2001, pp. 297–304.

[16] R.G. Belleman and P.M.A. Sloot, Simulated vascular reconstruction in a virtual operating theatre, in: *Computer Assisted Radiology and Surgery* (*Excerpta Medica, International Congress Series 1230*), H.U. Lemke, M.W. Vannier, K. Inamura, A.G. Farman and K. Doi, eds, Elsevier Science B.V., 2001, pp. 938–944.