

Medical images simulation storage and processing on the European DataGrid testbed

Johan Montagnat, F. Bellet, Hugues Benoit-Cattin, Vincent Breton, Lionel Brunie, Hector Duque, Yannick Legre, Isabelle Magnin, Lydia Maigne, Serge Miguet, et al.

▶ **To cite this version:**

Johan Montagnat, F. Bellet, Hugues Benoit-Cattin, Vincent Breton, Lionel Brunie, et al.. Medical images simulation storage and processing on the European DataGrid testbed. Journal of Grid Computing, 2004, 2, pp.387-400. 10.1007/s10723-004-5744-y . in2p3-00023294

HAL Id: in2p3-00023294
https://hal.in2p3.fr/in2p3-00023294

Submitted on 23 Nov 2004

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Medical images simulation, storage, and processing on the European DataGrid testbed

J. Montagnat[1], F. Bellet[1], H. Benoit-Cattin[1], V. Breton[4], L. Brunie[2], H. Duque[1,2], Y. Legré[4], I.E. Magnin[1], L. Maigne[4], S. Miguet[3], J.-M. Pierson[2], L. Seitz[2], T. Tweed[3]

[1] *CREATIS, CNRS UMR5515-INSERM U630, INSA, 20 av. A. Einstein, Villeurbanne, France*

[2] *LIRIS, CNRS FRE 2672, INSA, 20 av. A. Einstein, Villeurbanne, France*

[3] *LIRIS, CNRS FRE 2672, Université Lyon 2, Bron, France*

[4] *LPC, CNRS/IN2P3, 24 avenue des Landais, 63177 Aubière Cedex, France*

Abstract. The European IST DataGrid project was a pioneer in identifying the medical imaging field as an application domain that can benefit from grid technologies. This paper describes how and for which purposes medical imaging applications can be grid-enabled. Applications that have been deployed on the DataGrid testbed and middleware are described. They relate to medical image manipulation, including image production, secured image storage, and image processing. Results show that grid technologies are still in their youth to address all issues related to complex medical imaging applications. If the benefit of grid enabling for some medical applications is clear, there remain opened research and technical issues to develop and integrate all necessary services.

Keywords: Medical imaging, grid computing and storage, European DataGrid project, simulation

1. Context

Medical images play a key role in medicine for diagnosis, therapy planning and treatment follow-ups. All major medical imaging modalities today produce digital images (Acharya et al., 1995). Digital medical images represent an enormous amount of distributed data for which automated processing is increasingly needed. Most recent medical imaging devices produce 3D images. A standard 3D Computed Tomography scan (CTscan) of Magnetic Resonance Image (MRI) represents tens to hundreds of MB of data. A single radiology department in a medium size hospital is estimated to produce tens of TB of digital images each year. Medical images are distributed over the medical acquisition centers throughout the territory. Although national regulation concerning medical images are heterogeneous in Europe, the current trend is: (i) a free access of patients to their medical data, and (ii) the long term archiving (from 20 to 70 years) of all medical data for pathology and epidemiology studies.

Automated medical image analysis and processing tools have been developed in computer science and signal processing laboratories for more than 15

years. Beyond the low level processing for signal filtering or 3D reconstruction internal to medical imagers, medical image processing algorithms proved to be useful for image enhancing, visualization, comparison, quantitative evaluation, and various simulation processes. Medical image processing algorithms provide diagnosis assistance, therapy planning tools, and a way of performing tedious image analysis tasks which are not human tractable for large datasets. In addition, some medical image analysis tools require very large computing power.

Grid technologies, that have recently emerged as a data intensive manipulation tool, are promising for medical image management. They offer large scale and distributed storage associated to better use of computing power. They permit to share data and resources which is important for clinical practice since hospitals and clinics usually do not own much computing power. Beyond the obvious interest of grids for clinical practice, these technologies favor research by allowing scientists to share datasets and image processing algorithms more easily than ever. All these facts made the awareness about grid technology benefits raise in the medical community these very last years.

The European DataGrid IST project main objective was to develop a *middleware* layer capable of addressing application requirements coming from three different communities: High Energy Physics, Earth Observation, and Biomedical applications (EDG, 2001). It was a pioneer in identifying the biomedical applications as a candidate for grid enabling. The requirements identified by the Biomedical applications working group early revealed to be the most complex and the most challenging for the middleware developers. As a result, all of them could not be addressed within the project lifetime. Early in the project, two communities were identified inside the biomedical applications working group: the bioinformatics and the medical imaging communities. This paper exclusively focuses on the later and does not address all work done on genomics, proteomics, and phylogenetics among the bioinformaticians participating to this working group.

This paper summarizes medical image processing application requirements identified during the project in section 2. It further details the need for complex and distributed medical datasets management on which specific effort has been allocated in section 3. Section 4 describes and reports on several medical image processing related applications that illustrate the interest of grid technologies in this field.

2. New trends in medical imaging and grid promises

Grids make the promise of large computing power and data storage space, but more benefits are expected in the medical imaging domain beyond these capabilities. Indeed, grids are a vector for permitting the creation of large scale

distributed datasets, enforcing the use of common standards, and permitting the medical communities to share computing resources and algorithms. Grids are likely to have a deep impact on health related applications by playing a key federative role (Breton et al., 2003). They provide a logical extension to regional health networks (Huang, 1996) by allowing distant sites to collaborate and exchange their data for specific research purposes.

Medical imaging applications that can benefit from grid technologies often involve large and/or distributed datasets. However, their successful deployment requires to tackle specific needs related to medical data manipulation and computations that we detail thereafter. The level of maturity of the EDG middleware regarding all these requirements is indicated.

2.1. Data-related requirements

Medical data security. The primary concern when distributing medical data over a grid is privacy. Medical applications often deal with patient data that are confidential and should only be accessible to the patient himself, the medical team involved in his health care, and, under some restrictions, for research purposes. Therefore, a medical grid, opened to a wide community of users, should enforce strict access right control. The lack of data security integration is today a major weakness of the EDG middleware to address medical requirements. Section 3.2 further comments on the needed security infrastructure.

Medical data semantics. Another particularity of medical data is their strong semantic content. As illustrated in section 3, a medical image itself is often of low interest if it is not related to a context (patient medical record, other similar cases...). Tools to manipulate metadata attached to the data are a first step in this direction. Metadata and application metadata facilities have been integrated lately within the EDG middleware.

Traceability. Another related requirement for a medical data management system is traceability. It should always be possible to know, for a given image where it originates from (which algorithm and which input image(s) were used to produce it). Indeed, physicians often need to come back to the unaltered data when studying a processed image. Conversely, for each input data it is of interest for optimizing computations to record which output has already been processed using various algorithms (computation results cache). Only low level logging is performed by the EDG middleware and medical traceability has to be implemented at the application level today.

2.2. Computation-related requirements

Pipelining computations. Medical application usually require more than a middleware offering batch job submission services and data access. A medical experiment often involves not a single algorithm but a set of processings that

can sometimes be executed concurrently. Processing pipelines are compound jobs composed of several elementary stages Stages are chained but not necessarily linearly. The EDG project has developed a Directed Acyclic Graph (DAG) job submission service allowing the user to describe compound jobs as DAGs of elementary processes. The DAG job manager is a computation flow controller. However it does not implement a data flow manager yet. Pipelines are of real interest when processing a large number of input data rather than a single input. Through pipelines, the user can describe once for all the chain of transformations that each element of the input dataset should undergo.

Parallel computations. Some image processing, simulation, and modeling algorithms are very compute intensive and need a parallel implementation in order to get executed in a reasonable amount of time compatible with clinical practice constraints. Local area parallelism is widely available today through message passing interfaces. The EDG project has lately developed a parallel job interface on top of the MPICH-G2 (MPICH for Globus Toolkit 2, (Karonis et al., 2003)) implementation.

Interactive applications. Interaction with the user may be needed for controlling an algorithm, to solve legal issues when dealing with medical data, or for the application itself (*e.g.* therapy simulator). Data compression and high-bandwidth networks should ensure a limited response time which is mandatory for interactive usage. Interactive feedback often involves 3D visualization of medical scenes. This is challenging due to the large size of 3D medical images and the complexity of meshes used for realistic 3D modeling (Montagnat et al., 2002). The EDG middleware allows the user to specify outbound connectivity as a requirement for job execution to ensure possible communication between running jobs and the user interface.

2.3. Future trends and opened doors

Sharing data sources will facilitate research on pathologies and epidemiology. Connecting distributed data sources will allow researchers to assemble *virtual data sets* suited for statistics extraction or study of rare diseases. With a proper grid infrastructure, experiments can be led at a scale never reached before. Sharing resources will facilitate the access of health centers to image processing services even though they might involve computation. Finally, sharing algorithms will ease the access to such image processing tools for the end user and foster collaboration, comparison, and algorithms assessment on the software developer side. Grid technologies are not only providing additional computing and storage power but they are also an opportunity to address new medicine challenges.

3. Managing medical data in a grid environment

The Digital Image and COmmunication in Medicine (DICOM) specification has recently emerged as the standard for image storage (DICOM, 1996). DICOM describes an image format, a communication protocol between an image server and its clients, and other image related capabilities. On top of such a standard, Picture Archiving and Communication Systems (PACS) are deployed to manage data storage and data flow inside hospitals. However, medical images by themselves are not sufficient for most medical applications. A physician is not analyzing images but he needs to interpret an image or a set of images in a medical context. The image content is only relevant when considering the patient age and sex, the medical record for this patient, sociological and environmental considerations, etc. Beyond simple diagnosis, many other medical applications are concerned with the data semantics and require rich metadata content. Therefore, medical metadata carrying additional information on the images are mandatory.

In addition to PACS, hospitals have a need for Radiological Information Systems (RIS). The PACS archives the images and performs image transfers. The RIS contains full medical records: image-related metadata and additional information on the patient history, pathology follow-up, etc. Although some vendors propose integrated PACS and RIS, there exists no open standards for the data structure and the communication between the services in this architecture. Moreover, they are usually designed to handle information inside an hospital but there is no system taking into account larger data sets nor the integration with an external component such as a computation/storage grid. Inside the EDG, we have been working on interfacing DICOM servers with the grid *Storage Element* specification in order to build a high level medical information system benefiting from the grid data storage and metadata management services.

3.1. Medical images distributed storage and retrieval

The DataGrid data manager identifies files through a Grid Unique IDentifier (GUID). To each GUID is associated one or several physical instances of the file named *replicas*. The data manager manipulates files that are stored in different Mass Storage Systems (MSS) through a unified *storage interface*. To ensure fault tolerance and to provide an efficient access to data, files are registered into the data manager and may be replicated transparently by the middleware in several identical instances, on different MSS. When a file is needed, the grid middleware will automatically choose its best available copy. To solve consistency problems, replicas are accessible in read only mode.

To easily manipulate medical images from the EDG testbed, we have designed a storage interface to DICOM medical servers. This proved to be

difficult since DICOM data are not structured as flat files but as collection of image slices (DICOM series) and DICOM slices are containing both raw image data and metadata. The *Distributed Medical Data Manager* (DM^2) (Duque et al., 2003) that we are developing therefore defines an abstraction for medical images and split raw image data from metadata. A DM^2 connected to the DataGrid data manager is depicted in figure 1. The DICOM interface to the DM^2 has been implemented today. The storage interface is still under investigation.

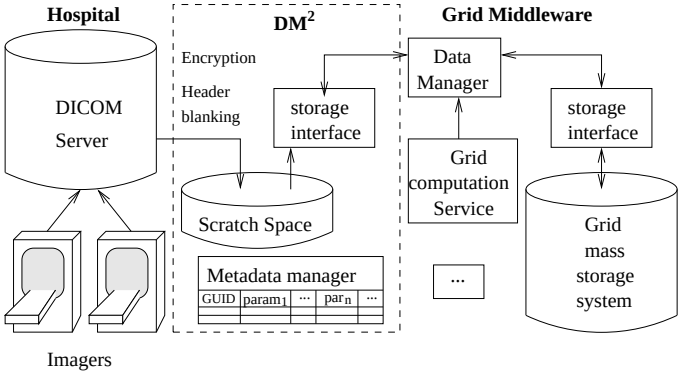


Figure 1. DM^2 interface between medical imagers and the grid

Although image processing algorithms are manipulating 3D images, possibly made of a set of DICOM slices, the DataGrid storage interface only deals with data at the file granularity level. Therefore, each 3D image has to be recognized as a single file by the system. When sets of DICOM slices are registered into the hospital DICOM server, the structure of this data set is interpreted and one or several GUID are associated to virtual image files. From the grid point of view, these files will therefore be published and accessible to any grid service through the storage interface. However, the physical image files are not assembled until requested through the storage interface. On demand, the requested image file is assembled on a scratch space by querying the DICOM server for the set of DICOM slices composing the image and extracting the image content from these files. It is then returned to the querier. The image can be replicated to any classical MSS or downloaded from a worker node for computation. For efficiency, assembled files are cached on the scratch space for future use.

The DM^2 also extracts metadata from all DICOM files registered in the DICOM server and store them in an SQL database to ease query on metadata. A link between each image GUID, the composing DICOM slices, and the associated metadata are stored in the same database. The metadata structure is designed to be extensible: the user can associate any complementary

metadata needed for a medical application to the image. Later versions of the DataGrid data manager also permit registration of metadata associated to data files. However, the granularity is not necessarily sufficient in this case, and integration with the metadata facility of the data manager is not possible today for security reasons. Indeed, medical metadata is the most critical part of the data as it may contain patient private and identifying information. The metadata database stored inside the DM^2 also contains additional security elements detailed in the next section.

The DM^2 is able to register and provide a grid interface to data coming from several distributed DICOM servers. It enables the DICOM server with a storage interface that makes it visible as any MSS. However, the DM^2 is a read-only MSS as it does not allow external grid data to be stored on the sites it controls: new medical images are registered internally when produced on the medical imagers and DICOM servers are not intended to store any other kind of data.

3.2. Security and privacy

Preserving patients privacy is a major concern for medical data processing systems. The distribution of data over a grid makes data control much more difficult than on closed systems. Data on grids may be replicated but all storage sites are not accredited to receive medical data. Therefore, their administrators should not have read access to the data content. Some identifying metadata are not accessible to non accredited users as well. Achieving a high security level is mandatory but security is always a trade off between inconvenience for the users and the desired level of protection. In order to convince users (physicians and patients) to use grids for their data storage and processing needs, many functionalities need to be provided such as:

- Reliable authentication of users.
- Secure transfer of data from one grid element to another.
- Secure storage of data on a grid element.
- Access control for resources such as data, storage space or computing power.
- Anonymization of medical records to make them available for research.
- Tamper-proof logging of operations performed on medical files.
- Robustness against denial-of-service attacks

Note that we have not included *secure processing of data* in this discussion. Performing computations on encrypted data without the without explicit decryption is a burning research area today. These techniques can accommodate to simple arithmetic operations but they are not mature enough to handle the complexity of image processing, not to mention the efficiency problems. To remain realistic, the features that should protect data while it is being processed on a grid are based on best effort technologies, *i.e.* on-disk encryption,

access control, and anonymization. Users need to trust the servers on which their data is to be processed, to our knowledge no systems for data processing on untrusted resources exist.

Our proposal for addressing all these requirements are detailed below.

Authentication is not a grid-specific problem. It is well researched and standard solutions exist. The use of a public key infrastructure (PKI) with certification authorities (CA) and X.509 certificates is a reasonable way to handle authentication in grid environments. The EDG middleware relies on Globus (Foster and Kesselman, 1997) and its public key-based infrastructure (Foster et al., 1998).

Secure transfer is also a well researched area independently of grid technologies. It is addressed in various standardized protocols such as *SSL/TLS*, *IpSec* or *SSH*. Data transfers are handled by GridFTP (Allcock et al., 2002) in the EDG middleware and can be encrypted although this functionality is not used in the EDG testbed.

For **secure storage of data**, encryption and signing is an obvious solution. The problem in grid environments is that mechanisms are required to share decryption keys between users authorized to access data. Common encrypted storage systems lack the flexibility to deal with the dynamic nature of grid access permissions. We have therefore proposed an architecture with a generic interface to grid access control mechanisms, that provides access to decryption keys based on access permissions. For further details on this system see (Seitz et al., 2003a)

Authorization and access control raises the most problematic issues for medical data processing in grid environments. Classic access control techniques are not designed to deal with the problems arising from the decentralized, cross-organizational nature of grid access permissions. The medical field of applications adds another inherent problem. Grid applications such as nuclear physics deal with data that has relatively low confidentiality and that is accessible for large groups of users. Classical grid access control mechanisms such as CAS (Pearlman et al., 2002) are satisfactory. Nevertheless these systems fail to provide sufficient permission granularity and flexibility for ad hoc permission granting that is required in medical applications. Furthermore such systems use centralized permission databases. We want to avoid this since they are a single point of failure.

An alternative approach is to manage grid access control using decentralized permission checking through attribute certificates. Such certificates permit resource administrators to issue permissions in a simple way without having to resort to third party services. Local servers can easily verify the permissions granted in such certificates, using a local database that specifies the *sources of authority* (SOA) of the resources on their systems. The attribute certificates enable the local servers to trace a permission from SOA of the concerned resource to the user requesting it. The database that specifies the

SOAs is managed by the access control system itself and is updated, when new resources (e.g. files) are added to the system.

The EDG security model is based on Virtual Organizations (VO, (Foster et al., 2001)). Resource providers assign permissions to those VOs, and the VOs have policies to dispatch the resources they have been assigned between their members (Alfieri et al., 2003). Our access control system supports this cooperation model by providing role based access control (RBAC) (Ferraiolo and Kuhn, 1992). Using RBAC, administrators can manage user groups (VOs) that are assigned sets of permissions and the membership of users within those groups. Our access control system also provides a generic program execution interface, that permits users to run their own specific programs in a sandbox environment prior to giving access to a resource. For details on our proposed access control system see (Seitz et al., 2003b).

Anonymization is required to provide large sets of data for medical research. Legislation imposes severe regulations as to what can be considered an anonymized information. The main problem is that even if obvious sections such as name and address of the patient have been removed a medical document could be re-identified with secondary information. We have not yet addressed that problem within the project, however our access control system is designed to provide an interface where a data filtering software can be plugged in before a medical file is delivered in order to ensure privacy protection. A promising approach to deal with anonymization is described in (Claerhout and De Moor, 2004).

Traceability is clearly another important factor in medical grids. The pre-access program execution interface, integrated in our access control system, can be used to plug in a log-keeping system. Through this interface the system will be able to get the necessary information about access requests to keep the log-file. However since the logs and the programs that create them are located on the distant storage element, users have to trust the administrators of this storage element not to interfere with the log-file creation.

The availability of services may be a critical factor in medical environments. Most measures to prevent denial-of-service attacks are not specific to grid architectures. However it is important to realize that centralized services are very vulnerable to such attacks. Therefore none of our proposed security services relies on a single centralized service.

4. Processing medical images

4.1. Magnetic Resonance Images simulation

4.1.1. *MRI physics simulation*

The simulation of Magnetic Resonance Images (MRI) is an important counterpart to MRI acquisitions. Simulation is naturally suited to acquire theoretical understanding of the complex MR technology. It is used as an educational tool in medical and technical environments (Torheim et al., 1994). By offering an analysis independent of the multiple parameters involved in the MR technology, MRI simulation permits the investigation of artifact causes and effects (Olsson et al., 1995; Brenner et al., 1997). Likewise simulation may help in the development and optimization of MR sequences (Brenner et al., 1997). Simulated MR images also provide an interesting assessment tool (Kwan et al., 1996) since it generates 3D realistic images from medical virtual objects which structure is perfectly known while this ground truth is usually not available when dealing with clinical data.

The CREATIS laboratory (CNRS-Inserm) develops, in collaboration with the CNRS LRMN-MIB[1] lab and with CEMAGREF/TEA[2] research unit, a 3D MRI simulator named SIMRI that is designed to simulate realistic high resolution 3D MR images and includes magnetic susceptibility and chemical shift artifacts from a virtual object and an MRI sequence (describing the succession of the magnetic events) as illustrated in figure 2.

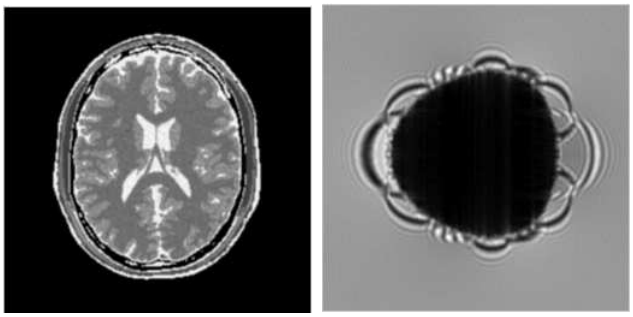


Figure 2. Simulated image of a brain (left) and simulated image of an air bubble into water showing the susceptibility artifact (right).

Since simulation of the MR physics is computationally very expensive (Brenner et al., 1997), parallel implementation is mandatory to achieve performances compatible with the target applications. The magnetization computation kernel is based on the solving of the Bloch equations (Bittoun et al., 1984) which describes the local spin magnetization. It requires the use of 3D rotation matrices with trigonometric and exponential functions. It can be shown that the

[1] LRMN-MIB UMR CNRS 5012, Lyon, France. http://jade.univ-lyon1.fr/

[2] CEMAGREF TEA Research Unit, Rennes, France. http://www.rennes.cemagref.fr/tere/tere.htm

overall volume simulation time is proportional to the object size ($X \times Y \times Z$) multiplied by the image size ($M \times N \times P$). As an example, the simulation of a 128^2 image takes only 3 minutes on a P4-2.6GHz PC, but multiplying by two all the dimensions of the virtual object and the MR image leads to a simulation time multiplication by 16 in two dimensions and by 64 in three dimensions (Benoit-Cattin et al., 2003). Therefore, we turn toward Grid technologies that promise a virtually unlimited computing power and we propose a gridification of our MRI simulator.

4.1.2. *Gridification strategy and results*

The parallelization of the magnetization kernel has been done using the MPI version for Globus (MPICH-G2). Because all the spin magnetization vectors are independents and because the signal acquisition process is linear, a parallelization scheme of type "divide & conquer" (see figure 3) has been implemented. It consists in distributing the magnetization computation of a subset of spin vectors. This subset can be fixed to a given size or adapted to the number of active nodes.

All the computation nodes have the MRI sequence knowledge and they receive from the master node a part of the virtual object. They compute the magnetization evolution of the corresponding spin vector subset. At the end of each acquisition step, the master node collects and adds all the Radio Frequency (RF) signal contributions. At the end of the MRI sequence, the master node applies the reconstruction algorithm to generate the MRI simulated image.

When using an homogeneous grid, the virtual object portion distributed to the nodes has a maximal size equal to the object size divided by the number of nodes. Only one distribution is done at the process begin which limits the communication between the master node and the computation nodes. When using an heterogeneous grid, to avoid to be penalized by the slowest node, the distributed object portion is reduced. In this case, the lowest node will receive one portion of the object to process while the fastest nodes will receive several portions.

The third line of table I gives computation time values for different object and image sizes obtained using a small grid based on an 18 PC cluster (8 PIII 1GHz, 10 P4 2.6GHz). These simulation results show that with a small cluster, MRI simulation of high resolution (1024^2) 2D images is possible within one day. Concerning 3D images, it is not realistic to simulate on such a small set of processors over 64^3 MRI. Nevertheless, it is possible to simulate within a week 3D multi-slice images (32 slices of 512^2 pixels). The simulation of high resolution 3D images should be tractable on full scale grids. However, large scale experiments were not possible on the DataGrid testbed due to the limited deployment of MPI-enabled nodes at the end of the project. This application remained at the testing phase and could not scale up to production.

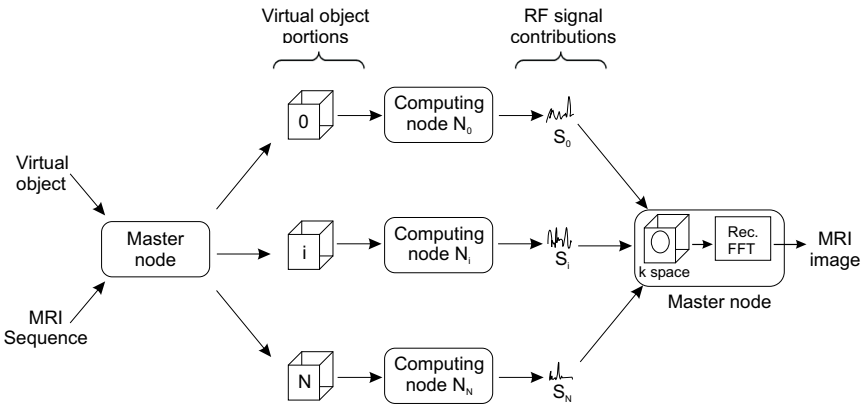


Figure 3. Data and Process distribution to the grid nodes: A "divide & conquer" scheme.

To better analyze the performance of the simulator, the sixth row of table I shows the computation time needed when executing a sequential version of the code on a single P4-2.6GHz processor. Since all values could not be measured because of the computation time needed, some of them (in italics) have been estimated using the theoretical time coefficient applied when the image size doubles (see fourth row of table I). The fifth row of table I shows the time coefficients measured on the parallel version of the simulator. It validates the use of a coefficient of 16 in 2D and 64 in 3D (only 8 in the last case since the object size doubled but not the image size). Therefore speed-ups could reasonably be estimated in the seventh row. The raw speed-up (except for the very small 64^2 image where communication times dominate) is in the order of 10 to 12. Remember that the 18 processors are heterogeneous and composed of 10 P4-2.6GHz processors (as in the sequential case) plus 8 PIII-1GHz processors. To compensate for the lower performance of PIII processors, a compensated speed-up has been roughly estimated by applying a 1.4 coefficient to the computed speed-up (it corresponds to the simple ratio of processor clock frequencies for 8 out of the 18 processors). The compensated speed-up, in the order of 14 to 16 shows that the parallelization on 18 processors does not cause any significant performance issue and that the resources are fully exploited.

4.2. Monte Carlo simulation for radiotherapy

4.2.1. *Radiotherapy simulation*

Monte Carlo simulations are increasingly used in medical physics, especially to elaborate cancer treatment. The principle is to simulate the radiation transport knowing the probability distributions governing each interaction of particles

Table I. 2D and 3D MRI simulation computation time on a cluster of 18 PC.

Object size	64^2	128^2	256^2	512^2	1024^2	32^3	64^3	128^3
Image size	64^2	128^2	256^2	512^2	1024^2	32^3	64^3	64^3
Time	2.2s	17.2s	4min10	1h09	18h50	1min15	1h13	9h41
Theoretical time factor	-	16	16	16	16	-	64	8
Measure time factor	-	12.7	16.7	16.8	16.3	-	60	7.7
Monoprocessor computation time (*theoretical*)	11.1s	3min05	49min23	*11h58*	*7d 23h30*	714	*12h41*	*4d 5h33*
Speed-up (*theoretical*)	5.04	10.8	11.85	*10.4*	*10.16*	9.52	*10.4*	*10.4*
Compensated speed-up (*theoretical*)	7.09	15.21	16.69	*14.64*	*14.30*	13.40	*14.64*	*14.64*

in the patient body to deliver the required dose deposit near the tumor and sensitive organs (see figure 4).

We know that some dosimetric studies for radiotherapy-brachytherapy treatments in complex body structure or at interfaces of tissue using analytic calculations have shown some limits. Indeed, most of the commercial systems, named TPS (Treatment Planning Systems), used for clinical routine use an analytic calculation to determine these dose distributions and so, errors near heterogeneities in the patient can reach 10 to 20%. Such codes are very fast comparing to Monte Carlo simulations: the TPS computation time for an ocular brachytherapy treatment is lower than one minute, thus allowing its usage in clinical practice, while a Monte Carlo framework could take 2 hours. Thus, there is a real interest for parallel and distributed Monte Carlo simulations in order to provide accurate medical studies for a clinical usage.

Medical radiotherapy treatment planning has been performed on the EDG Testbed, from pre-processing and registration of medical images on the Storage Elements (SEs) of the grid to the parallel computation of Monte Carlo simulations GATE (Geant4 Application for Tomographic Emission (Jan et al., 2004; Santin et al., 2003; Assié et al., 2003)).

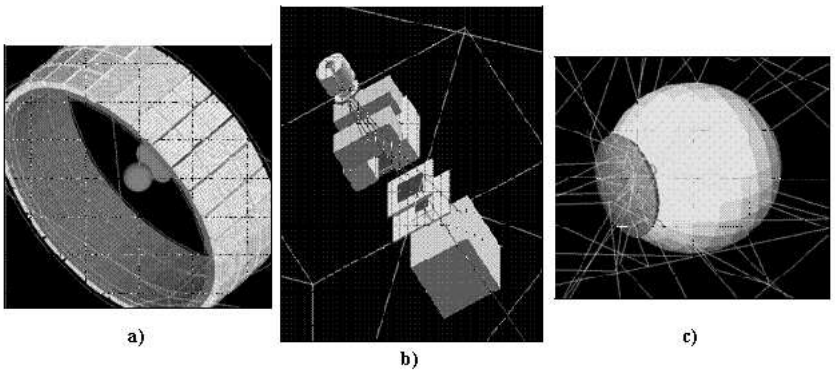


Figure 4. GATE Monte Carlo simulations: a) PET simulation; b) Radiotherapy simulation; c) Ocular brachytherapy simulation

4.2.2. *Medical images treatment*
The application framework is depicted in figure 5. Sets of 40 DICOM slices or so, 512^2 pixels each, acquired by CT scanners are concatenated and stored in a 3D image file format(see section 3). Such image files can reach up to 20 MB in size for our application. To solve privacy issues, DICOM headers are wiped out in this process.

The 3D image files are then registered and replicated on the sites of the EDG testbed where GATE is installed in order to compute simulations (5 sites to date). During the computation of the GATE simulation, the images are read by GATE and interpreted in order to produce a 3D array of voxels whose value is describing a body tissue. A relational database is used to link the GUID of image files with metadata extracted from the DICOM slices on the patient and additional medical information. The EDG Spitfire software (EDG WP2, 2001) is used to provide access to the relational databases.

4.2.3. *The parallelization of GATE simulations on the DataGrid testbed*
Every Monte Carlo simulation is based on the generation of pseudo-random numbers using a Random Numbers Generator (RNG). An obvious way to parallelize the calculations on multiple processors is to partition a sequence of random numbers generated by the RNG into suitable independent sub-sequences. To perform this step, the choice has been done to use the Sequence Splitting Method (Traore and Hill, 2001; Coddington, 1996; Maigne et al., 2004). For each sub-sequence, we save in a file (some KBs) the current status of the random engine. Each simulation is then launched on the grid with the status file.

All the other files necessary to run Gate on the grid are automatically created: the script describing the environment of computation, the GATE

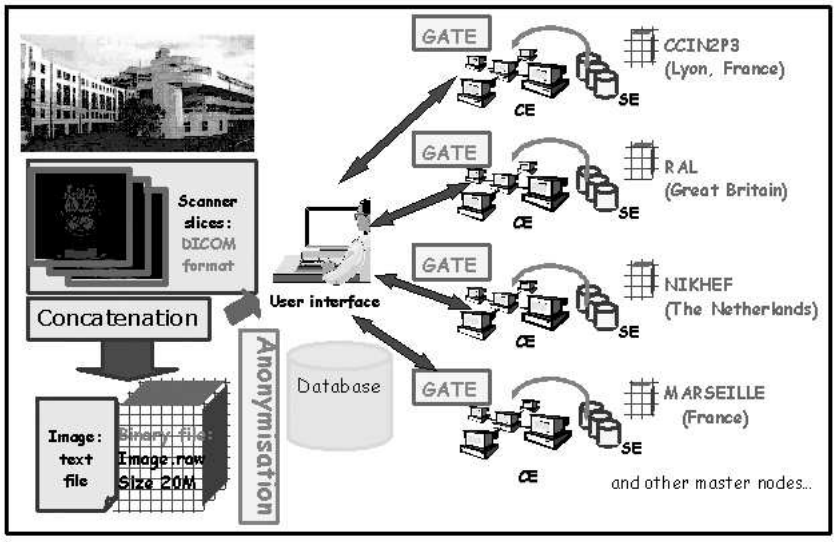


Figure 5. **Submission of GATE jobs on the DataGrid testbed**

macros describing the simulations, the status files of the RNG and the job description files.

4.2.4. *Results*

In order to show the advantage for the GATE simulations to partition the calculation on multiple processors, the simulations were split and executed in parallel on several grid nodes. Table II illustrates the computing time in minutes of a GATE simulation running on a single P4 processor at 1.5GHz locally and the same simulation splitting by 10, 20, 50 and 100 jobs on multiple processors.

Table II. Sequential versus grid computation time using 10 to 100 nodes

Number of jobs	1 (local)	10	20	50	100
Computation time (in min)	159.0	31.0	20.5	31.0	38.0
Speed-up	-	5.12	7.75	5.12	4.18

The results show a significant improvement in computation time although this is to be improved for clinical practice as the computing time using Monte Carlo calculations should stay comparable to what it is currently with analyt-

ical calculations. The next challenge is to provide daily for the user the best resources to compute his simulation on the grid.

This example also shows that the computing time is not proportional to the number of jobs running in parallel. In particular, the total computing time and the speed-up achieved drop after 20 processors used. This performance loss is due to two factors:

- The pay-off induced by the grid to manage each job.
- The constant computation time per job induced by the building of the geometry at the beginning of each simulation.

The geometry computation time is rather negligible (about 1.5% of the job computation time) and is therefore not responsible for the performance loss. The pay-off induced by the grid middleware decomposes into the job submission time and its queuing time. Job submission itself is rather negligible. The problem is the queuing time that is really dependent on the queuing policy of the site accepting the jobs. In the current DataGrid middleware, there exists no queue optimized for very short jobs and, since the jobs computation time decreases with the splitting, a too large number of processor yields to an unacceptable pay-off. Their is a trade-off to find between jobs splitting and grid pay-off. This is dependent on the queuing policy of the middleware. In the future, dedicated schedulers for short jobs are expected to improve the application performance by a large factor.

4.3. Searching medical databases

4.3.1. *Medical images indexing and content-based retrieval*

One of the primary expectation of physicians regarding medical information systems is the ability to access distributed patient medical records for diagnosis and for comparison with known records. Indeed, a physician may wish to confirm his diagnosis by comparison of a medical case he is studying to other known cases. Metadata query is the first way of searching for similar medical cases. However, metadata are often not sufficient for that purpose and image analysis tools dedicated to detection of a specific pathology are needed. Medical images indexing and content-based retrieval of images is very important in the medical field.

A simple way to compare medical images is to use similarity measures (Penney et al., 1998; Montagnat et al., 2004). Although each measurement is not very compute intensive, the comparison of a sample image against a complete database is intractable, in a reasonable time, on a single computer due to the size of medical databases. The actual cost of such a computation depends on several parameters such as the input image size and the computation precision desired.

More image or pathology specific comparison criteria may be extracted from the images. For instance, in the case of mammograms analysis, the physician

is mainly interested in the detection of tumors, their classification (malignant or benign), and their location.

4.3.2. *An application to computer aided diagnostic in mammograms*
Breast cancer is one of the most common cause of women mortality. In France, a systematic screening for breast cancer is generalized for women between 50 and 74 years old, in order to detect the early signs of change that could point out the presence of a malignant tumor. The number of mammograms to be analyzed is in constant increasing; the data corresponding to mammograms and medical diagnosis reports are distributed among several medical sites. Thus, an early detection using mammographic screening is essential. In order to be specific, a computer-aided diagnosis system (CAD) is an ideal tool in assisting a radiologist, and can be used as a second opinion or a second reading. Those tools, based on a segmentation or a detection, then a feature extraction, and finally a classification or decision making (Bick and Doi, 2000), need to be trained among different databases.

Our aim in this application is to evaluate the grid possibilities in order to build a distributed system of stored mammographic data and metadata, that will work as a CAD tool. This system must allow the different users, specifically physicians or researchers, to analyze and index the images distributed among the different geographic medical sites, to do some content-based requests on the image databases, and then offer an assistance to the diagnosis, based on the research of a set of images that are similar to a request image according to some extracted features. Two types of scenarii of content-based request were considered:

- A physician has doubts about a particular region in the mammogram he is analyzing. He can search for images of the database that contain regions having similar properties, based on similarity measures. Two types of requests can be submitted to the system: find the set of images containing regions that are similar to the query zone, or find the set of images that contain regions previously detected as cancerous and that are similar to the query zone.
- Without the help of a specialist, a new image is compared with a set of images in the database, for example in the case of a second reading. The idea is to highlight in the image all the zones that are close to the regions that have previously been noted as cancerous. This can be helpful for attracting the attention of a specialist to a precise region that he could have missed during his first reading.

For our tests, we are working on a digital database containing 2620 patients, divided into three groups: benign, malignant and normal. This database is composed of 230 GB of mammographic data and comes from the University of South Florida (Heath et al., 1998). Each case/patient includes two views of

each breast and information about the patient, the study date, or the scanner used for digitalization. In the case of benign and malignant cases, a description of the malign regions, delimited by a specialist and confirmed by later examinations, is given using the ACR BI-RADS lexicon (BI-RADS Committee, 1998). The whole image database is stored on a mass storage system called HPSS (High Performance Storage System) in the IN2P3 Computing Center, which is one of the major resource provider in the EDG project.

The heart of our algorithms in this application is the comparison of elementary regions in images. For computation optimizations, this comparison is not done on the image data themselves, but on feature vectors extracted from the regions. We have developed an indexing algorithm that describes elementary regions of the images by the way of feature vectors based on gray level distribution as well as texture analysis. We have reported in our previous works (Tweed and Miguet, 2002) the indexing process we use to describe image regions. This indexing is compute intensive: from 8 to 30 minutes per case (4 images) on 2.4GHz P4 to 750MHz PIII based machines.

We have then experimented several proximity criteria on these feature vectors: simple thresholds on the histograms, and Euclidean distances on the texture attributes, in order to make requests as described above. We have been working on the optimization of the database indexing on the EDG testbed. We have developed jobs that transfer the image data from the storage elements to the workers and that perform image indexing.

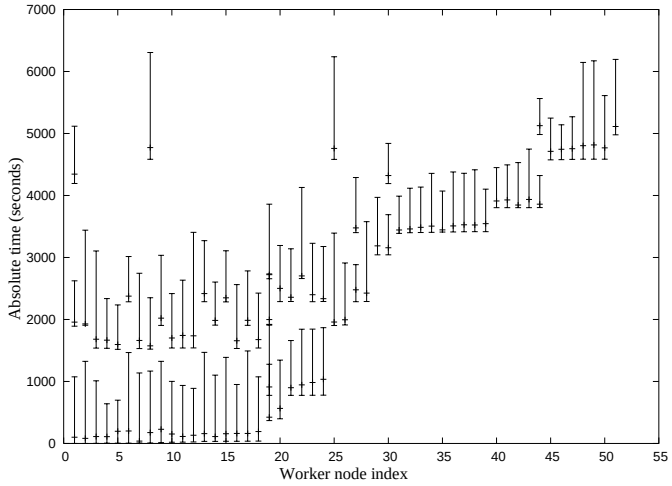


Figure 6. distributed indexing of 83 patients

Figure 6 shows a typical experiment on 83 patients (332 images) indexing. Each vertical bar represents the starting time and the duration of one job (in seconds). On the horizontal axis are represented each node selected by the

scheduler for the computations. The results are very conclusive: the computing time execution is of less than two hours using the EDG testbed. The sequential computing time for the same experiment is 23 hours. The speed-up in this case is 13.14 using up to 51 processors. Although this is far from linear, one should notice that this experiment has been ran on a production computing infrastructure and other jobs were scheduled on the same processors. Indeed, the starting time of some jobs is delayed by the late availability of processors. We observe that several jobs start at the same time, but not all according to the availability of the resources at the time the experiment was led. More than 800 processors could theoretically have been used but the racing conditions with other users lead to a consumption of 51 processors and the re-use of some worker nodes for jobs once the first indexing/task was finished. Given the parallel nature of this application, almost linear speed-up can be obtained by dedicating resources to this task.

5. Discussion

The EDG middleware and testbed provide a basic grid infrastructure for testing grid-enabled medical applications. As reported in section 4, different kind of applications could be experimented to some level and real benefits in terms of computation time and size of datasets processed have been demonstrated. This platform is still in its youth though, and most advanced developments, only recently made available, could hardly be tested. More services are expected in order to cover all medical image application requirements.

Privacy and security remain primary concerns for deploying large scale applications that involve real patient data. Medical data security requirements are complex: several category of users with different access rights, encryption to avoid accessibility to data for non accredited system administrators on storage sites, security controls to prevent intrusion on data storage sites, etc. However, these constraints have to be enforced in order to be able to interconnect medical information systems with patient data to the grid resources. Another important related aspect is the confidence the medical users will put in the system. As long as the system is not trusted by the community, progress on grid-enabling medical applications will remain slow.

Data and metadata management is another domain that requires further investigation. Medical data are widely distributed due to their acquisition in different centers spread out the territory. The management of medical data requires an information system capable of dealing with data sets rather than flat files. Moreover, processing often concerns full data sets rather than single data. The semantics of data and metadata should be taken into account by the data manager to ease meaningful retrieval of medical data.

Other computational aspects can also improve medical data processing in the future. A pipelines computation system such as the EDG DAG jobs manager does not cover all application requirements for instance since it does not take into account the processing of full datasets. Parallelism is another point that has been well studied on cluster architectures but for which grid-wide implementation on a large scale and heterogeneous infrastructure is still to be investigated.

A key factor in the success of grid technologies in the medical domain will be its accessibility to non computer scientists. End users are often non specialists who need well designed interfaces and algorithms applying to precise medical analysis needs. Grid technologies will only be adopted once it has proved to be more useful and as easily accessible as existing PACS and RIS.

Considering the medical application themselves, all development and deployment of medical applications made during the EDG project have been performed in parallel to the middleware development. This has made things difficult as applications were supposed to adapt to a continuously moving target. As a consequence, mostly simple applications with a rather straight forward capability for parallel execution could be ported in the project lifetime. The real impact of grid technologies in porting large scale applications is still to be investigated. We are just beginning this exploration now that the basic tools are available for development and testing.

6. Conclusions

The EDG project was a pioneer in identifying the biomedical domain as a relevant area of application for grid technologies. Within the project 3 years lifetime, the awareness of these technologies has raised in the medical image processing community and, to some extend, in the medical community. Medical informatics, and more generally biomedical informatics and life sciences are now well established candidates with a clear interest for grid enabling. Several clues testify of the growth of this emerging community such as conferences and workshops organized in this domain and the creation of international bodies such as the HealthGrid association (HealthGrid, 2003) or the GGF Life Science research group (LSG-RG, 2003) aiming at federating research projects in this field. The European Community is eager to develop grids as a high level infrastructure for e-Health and funds many research projects and *networks of excellence* in the domains of biomedical informatics and grid infrastructures. Among these, the EGEE (EGEE, 2004) project will deploy a production testbed for which biomedical applications are identified candidates.

Acknowledgements

The authors are grateful to the European IST DataGrid project (EDG, 2001) for financial and operational support in the various experiments related in this paper. This paper synthesizes the work performed in several laboratories developing grid technologies for health applications that are supported by several national and international research programs. The authors express their gratitude to the ACI GRID research program funded by the French ministry for research (ACI MEDIGRID, ACI GLOP), the Rhône-Alpes regional support (RAGTIME project), the French-Algerian CMEP agreement (Avicenne Grid project), and the French-Colombian ECOS Nord Committee (action C03S02). The LPC Clermont also acknowledges fruitful discussions with David Hill and the GATE collaboration.

References

Acharya, R., Wasserman, R., Sevens, J., and Hinojosa, C. (1995). Biomedical Imaging Modalities: a Tutorial. *Computerized Medical Imaging and Graphics*, 19(1):3–25.

Alfieri, R., Cecchini, R., Ciaschini, V., dell'Agnello, L., Frohner, Á., Gianoli, A., Lörentey, K., and Spataro, F. (2003). VOMS, an authorization system for virtual organizations. In *Proceedings of the 1st European Across Grids Conference*.

Allcock, B., Bester, J., Bresnahan, J., Chervenak, A., Foster, I., Kesselman, C., Meder, S., Nefedova, V., Quesnal, D., and Tuecke, S. (2002). Data Management and Transfer in High Performance Computational Grid Environments. *Parallel Computing Journal*, 28(5):749–771.

Assié, K., Breton, V., Buvat, I., Comtat, C., Jan, S., Krieguer, M., Lazaro, D., Morel, C., Rey, M., Santin, G., Simon, L., Staelens, S., Strul, D., Vieira, J., and Van de Walle, R. (2003). Monte Carlo simulation in PET and SPECT instrumentation using GATE. *Nucl. Instr. and Methods*.

Benoit-Cattin, H., Bellet, F., Montagnat, J., and Odet, C. (2003). Magnetic Resonance Imaging (MRI) simulation on a grid computing architecture. In *IEEE CGIGRID'03 - BIOGRID'03*, pages 582–587, Tokyo.

BI-RADS Committee (1998). *Illustrated Breast Imaging Reporting And Data System*, American College of Radiology edition.

Bick, U. and Doi, K. (2000). *Computer Aided Diagnosis Tutorial*. CARS 2000 Tutorial on Computer Aided-Diagnosis, Hyatt Regency, San Francisco, USA.

Bittoun, J., Taquin, J., and Sauzade, M. (1984). A computer algorithm for the simulation of any nuclear magnetic resonance (NMR) imaging method. *Magnetic Resonance Imaging*, 3:363–376.

Brenner, A., Kürsch, J., and Noll, T. (1997). Distributed large-scale simulation of magnetic resonance imaging. *Magnetic Resonance Materials in Biology, Physics, and Medicine*, 5:129–138.

Breton, V., Medina, R., and Montagnat, J. (2003). DataGrid, Prototype of a Biomedical Grid. *Methods MIMST*, 42(2).

Claerhout, B. and De Moor, G. (2004). Privacy protection for healthgrid applications. to appear in the Methods of Information in Medcine journal.

Coddington, P., editor (1996). *Random Number Generators For Parallel Computers, Second Issue*. NHSE Review.

DICOM (1996). Digital Imaging and COmmunications in Medicine, http://medical.nema.org/.

Duque, H., Montagnat, J., Pierson, J., Brunie, L., and Magnin, I. (2003). DM2: A Distributed Medical Data Manager for Grids. In *Biogrid'03, proceedings of the IEEE CCGrid03*, Tokyo, Japan.

EDG (2001). European DataGrid IST project, FP5, jan. 2001-feb. 2004, http://www.edg.org/.

EDG WP2 (2001). Spitfire. http://edg-wp2.web.cern.ch/edg-wp2/spitfire/.

EGEE (2004). European IST project of the FP6, Enabling Grids for E-science and industry in Europe, apr. 2004-mar. 2006, http://www.eu-egee.org/.

Ferraiolo, D. and Kuhn, D. (1992). Role based access control. In *15th NIST-NCSC National Computer Security Conference*, pages 554–563.

Foster, I. and Kesselman, C. (1997). Globus: A Metacomputing Infrastructure Toolkit. *International Journal of Supercomputer Applications*, 11(2):115–128.

Foster, I., Kesselman, C., Tsudik, G., and Tuecke, S. (1998). A Security Architecture for Computational Grids. In *Proc. 5th ACM Conference on Computer and Communications Security Conference*, pages 83–92, San Francisco, CA, USA.

Foster, I., Kesselman, C., and Tuecke, S. (2001). The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of Supercomputer Applications*, 15(3).

HealthGrid (2003). HealthGrid Association, http://www.healthgrid.org/.

Heath, M., Bowyer, K. W., and Kopans, D. (1998). Current Status of the Digital Database for Screening Mammography. In *Digital Mammography*, pages 457–460. Kluwer Academic Publishers. http://marathon.csee.usf.edu/Mammography/Database.html.

Huang, H. K. (1996). *PACS: Picture Archiving and Communication Systems in Biomedical Imaging*. Hardcover.

Jan, S., Santin, G., Strul, D., and et al. (2004). GATE (Geant4 Application for Tomographic Emission): a simulation toolkit for PET and SPECT. *to appear in Phys. Med. Biol.*

Karonis, N., Toonen, B., and Foster, I. (2003). MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface. *Journal of Parallel and Distributed Computing*, 63(5):551–563.

Kwan, R.-S., Evans, A. C., and Pike, G. B. (1996). An extensible MRI simulator for post-processing evaluation. In *International Conference on Visualization in Biomedical Computing, VBC'96*, pages 135–140.

LSG-RG (2003). Global Grid Forum Life Sciences Grid Research Group, http://forge.gridforum.org/projects/lsg-rg.

Maigne, L., Hill, D., Breton, V., and et al. (2004). Parallelization of Monte Carlo simulations and submission to a Grid environment. *to appear in Parallel Processing Letters*.

Montagnat, J., Breton, V., and I.E., M. (2004). Partitionning medical image databases for content-based queries on a grid. In *Healthgrid'04*, Clermont-Ferrand, France.

Montagnat, J., Davila, E., and Magnin, I. (2002). 3D objects visualization for remote interactive medical applications. In *3D Data Processing, Visualization, Transmission*, Padova, Italy.

Olsson, M. B. E., Wirestam, R., and Persson, B. R. R. (1995). A Computer-Simulation Program For Mr-Imaging - Application to Rf and Static Magnetic-Field Imperfections. *Magnetic Resonance in Medicine*, 34(4):612–617.

Pearlman, L., Welch, V., Foster, I., Kesselman, C., and Tuecke, S. (2002). A community authorization service for group collaboration. In *Proceedings of the 2002 IEEE Workshop on Policies for Distributed Systems and Networks*.

Penney, G., Weese, J., Little, J., Desmedt, P., Hill, D., and Hawkes, D. (1998). A Comparison of Similarity Measures for Use in 2D-3D Medical Image Registration. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 1496 of *LNCS*, pages 1153–1161, Cambridge, USA. Springer.

Santin, G., Strul, D., Lazaro, D., Simon, L., Krieguer, M., Vieira Martins, M., Breton, V., and C., M. (2003). GATE, a Geant4-based simulation platform for PET and SPECT integrating movement and time managment. *IEEE Trans. Nucl. Sci.*, 50:1516–1521.

Seitz, L., Pierson, J., and Brunie, L. (2003a). Key management for encrypted data storage in distributed systems. In *Proceedings of the second Security In Storage Workshop (SISW)*.

Seitz, L., Pierson, J., and Brunie, L. (2003b). Semantic access control for medical applications in grid environments. In *Euro-Par 2003 Parallel Processing*, volume LNCS 2790, pages 374–383. Springer.

Torheim, G., Rinck, P., Jones, R., and Kvaerness, J. (1994). A simulator for teaching MR image contrast behavior. *MAGMA*, 2:515–522.

Traore, M. and Hill, D. (2001). The use of random number generation for stochastic distributed simulation: application to ecological modeling. In *13th European Simulation Symposium, Marseille*, pages 555–559, Marseille, France.

Tweed, T. and Miguet, S. (2002). Automatic Detection of Regions of Interest in Mammographies Based on a Combined Analysis of Texture and Histogram. In *International Conference on Pattern Recognition*, pages 448–552, Qubec City, Canada.