Future Generation Computer Systems 25 (2009) 326–336

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

# MediGRID: Towards a user friendly secured grid infrastructure[☆]

Dagmar Krefting[a,*], Julian Bart[b], Kamen Beronov[c], Olga Dzhimova[c], Jürgen Falkner[b], Michael Hartung[d], Andreas Hoheisel[e], Tobias A. Knoch[f,g], Thomas Lingner[h], Yassene Mohammed[i], Kathrin Peter[j], Erhard Rahm[k], Ulrich Sax[i], Dietmar Sommerfeld[l], Thomas Steinke[j], Thomas Tolxdorff[a], Michal Vossberg[a], Fred Viezens[i], Anette Weisbecker[b]

[a] *Institute of Medical Informatics, Charité - Universitätsmedizin Berlin, Germany*
[b] *Fraunhofer Institute for Industrial Engineering IAO, Stuttgart, Germany*
[c] *Lehrstuhl für Strömungsmechanik, Technische Fakultät Universität Erlangen, Germany*
[d] *Interdisciplinary Centre for Bioinformatics, University of Leipzig, Germany*
[e] *Fraunhofer Institute for Computer Architecture and Software Technology, Berlin, Germany*
[f] *Biophysical Genomics, Kirchhoff Institute for Physics, University of Heidelberg, Germany*
[g] *Biophysical Genomics, Cell Biology and Genetics Cluster, Erasmus Medical Center, Rotterdam, The Netherlands*
[h] *Institute of Microbiology and Genetics, University of Göttingen, Germany*
[i] *Universitätsmedizin Göttingen, Abteilung Medizinische Informatik, Germany*
[j] *Zuse Institute Berlin, Germany*
[k] *Department of Computer Science, University of Leipzig, Germany*
[l] *Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen, Germany*

ARTICLE INFO

*Article history:*
Received 29 November 2007
Received in revised form
15 April 2008
Accepted 14 May 2008
Available online 18 May 2008

*Keywords:*
Grid computing
Usability
Security
Healthgrids

ABSTRACT

Many scenarios in medical research are predestined for grid computing. Large amounts of data in complex medical image, biosignal and genome processing demand large computing power and data storage. Integration of distributed, heterogeneous data, e.g. correlation between phenotype and genotype data are playing an essential part in life sciences. Sharing of specialized software, data and processing results for collaborative work are further tasks which would strongly benefit from the use of grid infrastructures. However, two major barriers are identified in existing grid environments that prevent extensive use within the life sciences community: Extended security requirements and appropriate usability. To meet these requirements, the MediGRID project is enhancing the basic D-Grid infrastructure along with the implementation of prototype applications from different fields of biomedical research. In this paper, we focus on the developments for ease-of-use under consideration of different aspects of security. They encompass not only security within the grid infrastructure, but also the boundary conditions of network security on the site of the research institutions. For medical grids, we propose a strictly web-portal-based access to grid resources for end-users, with user-guiding, application specific, graphical interfaces. Different levels of authorization are implemented, from fully authorized users to guests without certificate authentication in order to allow hands-on experience for potential grid users.

## 1. Introduction

### 1.1. Biomedical grids

Grids have been globally used in life sciences for many years [1]. The famous first mapping of the human genome would not have happened without grid technology [2]. A closer look reveals the fact, that in most cases, grids have not been used in regulated environments but for fundamental research. Also in clinical research and healthcare, technological and scientific advances have developed a rising need for computational resources that grid networks might be able to meet. Furthermore, clinical trials and integrated care require an infrastructure for collaboration between distributed and dynamically changing health care actors. Another possible benefit of health grids is the provision of services for specialized computer aided diagnosis and therapy planning tools. This presumed, health grids or medical grids, are expected to have a major impact on the healthcare business in the coming years

[☆] This work is supported by the the German Federal Ministry of Education and Science, MediGRID(01AK803A-H) as part of the D-Grid initiative.

[*] Corresponding author. Tel.: +49 544 515; fax: +49 544 901.
*E-mail address:* dagmar.krefting@charite.de (D. Krefting).
*URL:* http://www.medigrid.de (D. Krefting).

0167-739X/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2008.05.005

**Table 1**
Classification of data to be processed in health grids regarding security requirements

| Processed data | Sec. Level | Application classes | User |
|---|---|---|---|
| Non-human data | low | basic research<br>Knowledge bases<br>Demoversions | researcher<br>all<br>all |
| Anonymized human data,<br>no risk of reidentification | low | basic research<br>clinical research<br>Demoversions | researcher<br>res./physician<br>all |
| Anonymized human data<br>with risk of reidentification | medium | basic research<br>clinical research | researcher<br>res./physician |
| Pseudonymized human data | medium<br>or high | clinical research<br>clinical application | res./physician<br>physician |
| Patient data | high | clinical application<br>telemedicine | physician<br>physician/patient |

and the way the various healthcare actors are interacting [3]. The number of publicly funded medical grid projects in the past years, for example the European EGEE, the U.S. cancer network caBIG, or MediGRID, as part of the German grid initiative D-Grid, shows the rising interest in grid technologies for medical applications today [4–7]. While the potential of grid technology for medical research is undoubted, within the course of the MediGRID project we have identified two community specific barriers that have to be overcome in order to enable the widespread use of grid infrastructures in life sciences: security and usability.

### 1.1.1. Security requirements

Applications involving any human data have to meet regulatory requirements, encompassing data protection, data safety and reliability. These issues have to be guaranteed by the grid infrastructure. The principles of confidentiality and privacy have to be respected at all times within a grid workflow. Fine grained access control with personalized authentication and authorization is required. Whereas medical applications within hospitals still take place under the umbrella of the physician-patient confidentiality, research computing requires some more technical effort. The patient – as owner of his data – has the right to be informed why, where and how long his data is processed and stored. Therefore, medical grid applications must be equipped with a comprehensible audit track in order to fulfill this requirement (a-posteriori). Furthermore, we have to guarantee to the patient, that his data will only be stored and processed in a trustworthy environment (Tracking, a-priori). This is a challenge in grid computing, as every grid node has to be assessed concerning the trustworthiness using trust metrics [8]. Current grid middleware cannot fulfil all these requirements, as standard security methods do not scale in heterogeneous, distributed environments [9]. But of course these security restrictions apply not for all biomedical research. In MediGRID, we also deal with applications of low or no security requirements, i.e. gene sequence prediction of animal data. For these cases, security issues like identification and authorization are mainly determined by the demands of the resource providers. Table 1 shows the identified classes of processed data and their use and users regarding security requirements.

### 1.1.2. User requirements

The majority of researchers in medical sciences are working in institutions like university hospitals. This implies two limitations for grid usage: (A) Protected networks in clinical environments: Clinical IT environments are highly secured networks with strict firewall regulations. Integrating a clinical computing resource into an external grid infrastructure like MediGRID is difficult to accomplish. Grid clients require a variety of TCP ports and transfer protocols [10]. For example, gridFTP, the de-facto standard of file transfer within grid infrastructures demands a portrange of 5000 ports to be opened bidirectionally. Even web-based solutions demand further TCP ports [11], while typical firewall configurations in clinical environments allow only http and https connections to the standard ports—at the most additional ftp and mail transfer. A sustainable health grid infrastructure has to cope with such requirements, and cannot leave it to potential users to realize a reconfiguration of the institution's firewall. (B) Non expert computer users: While the firewall problem is mainly of technical nature, health grids typically deal with a community that consists mainly of researchers being medical doctors and not computer scientists. The acceptance of software tools depends strongly on usability and ease-of-use. If long training periods and computer knowledge are required to use the application, it is unlikely that it will find widespread acceptance, even if the functional benefit is proven. This is a wide difference between the Life Science community and "classical grid communities" like high energy physics, where software developers and software users are almost identical. Such a personal union guarantees a much higher insight into information technology and therefore a higher tolerance for command-line based tools, manual installation and configuration of software clients or even the use of graphical user interfaces that often still require input of technical configuration data. To make a grid infrastructure – as a distributed system – manageable for inexperienced users, a high level of virtualization is necessary. This applies for main parts of the data processing, computing resources, data storage and transfer, metadata retrieval and security implementations.

The mentioned boundary conditions for health grids contain many challenges aside from the implementation of algorithms on the grid nodes. But a general tendency in current grid research towards service oriented grid infrastructures, mature front-end security concepts and web-based grid portals paves the way for productional medical grids. In the following section, we will describe the MediGRID architecture within the D-Grid framework and the developments made to close, or at least narrow, gaps in operability and usability.

## 2. The D-Grid framework and MediGRID extensions

The MediGRID project – as part of the D-Grid initiative – is based on the core D-Grid infrastructure: The different community grids can choose between Globus Toolkit, gLite and Unicore as basic grid middleware. The D-Grid supports several technologies on top of the middleware stack, such as OGSA-DAI for distributed database access [12], dCache for distributed data management, the GridSphere Portal Framework [13] for the setup of grid portals, the Grid Resource Registry Service (GRRS) and the VO Membership

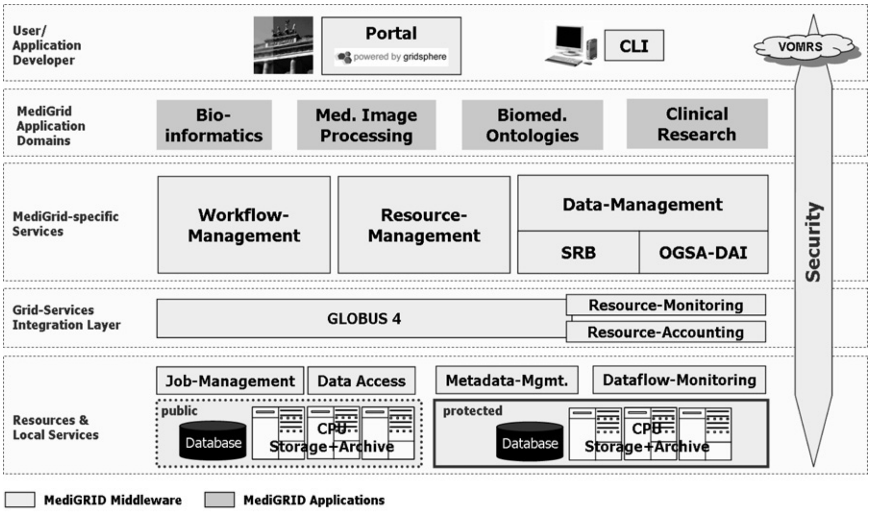

**Fig. 1.** Software architecture of MediGRID. The middleware layer splits into core D-Grid services and MediGRID specific services. The implemented applications are grouped into subdomains of medical research, to account for specific requirements and synergies. While user access is portal based, developers use regular client software. VOMRS-based security is enabled throughout all layers.

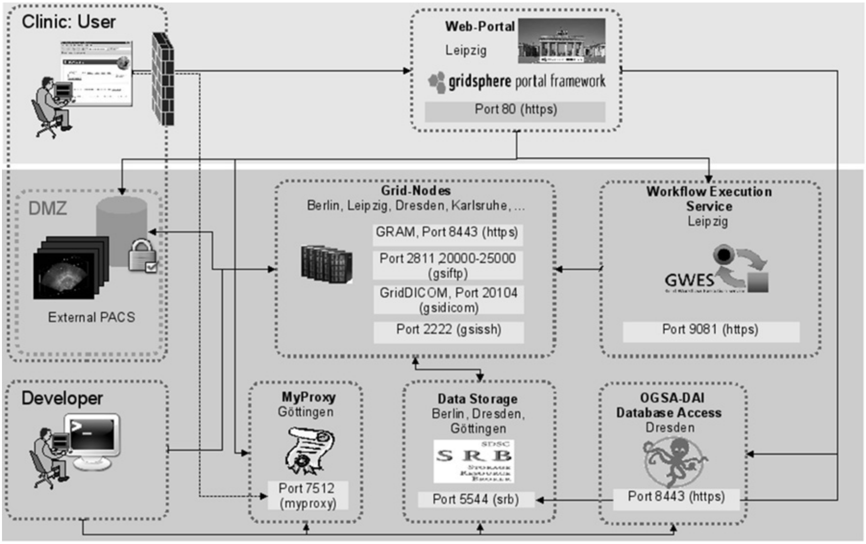

**Fig. 2.** Implemented MediGRID system architecture. User access is strictly webbased, while several transfer protocols and TCP ports are used within the grid environment. An exception is the weekly upload of the proxy certificate, which still needs outgoing connection to TCP port 7512 of the Myproxy server.

Registration Service (VOMRS) for resource and user management, repectively [14,15]. These technologies also include grid-wide monitoring services and (so far) rudimentary accounting. It also provides concepts for authentication and authorization as well as for the setup and management of firewall rules. D-Grid supports a public key infrastructure(PKI), accepting certificates from two certificate authorities. From D-Grid's portfolio, MediGRID uses Globus Toolkit, OGSA-DAI, GRRS, GridSphere, the monitoring services and the PKI-infrastructure. MediGRID focuses on fine-grained user management, using the provided VO and subVO structure, and the development of strictly portal-based graphical user-interfaces. On top of the core grid infrastructure, MediGRID integrates, enhances and develops a variety of further services and tools to meet the community specific requirements. They encompass enhanced resource management, the Grid workflow execution service (GWES) for process virtualization including basic resource brokering and scheduling [16], SRB data virtualization [18], and gridDICOM for medical image transfer [32]. The software layout and implemented system architecture of MediGRID are given in Figs. 1 and 2, respectivly. In the following sections, we present the MediGRID solution and developments in high level virtualization, user management and user interfaces towards a grid infrastructure suitable for the biomedical community.

## 3. Data and process virtualization

High-level virtualization of the grid is a prerequisite to allow inexperienced users full utilization of the grid potential. The key idea of a computing grid – the integration of distributed heterogeneous resources crossing administrative borders towards a single virtual computer – is even more important for users who are not experienced in distributed computing and network technologies. Data management and virtualization within the MediGRID is realized with SRB. For process virtualization, the Grid Workflow Execution Service (GWES) comes into operation.

### *3.1. SRB*

The Storage Resource Broker (SRB) is a Data Grid Management System (DGMS), based on a client-server architecture. It provides

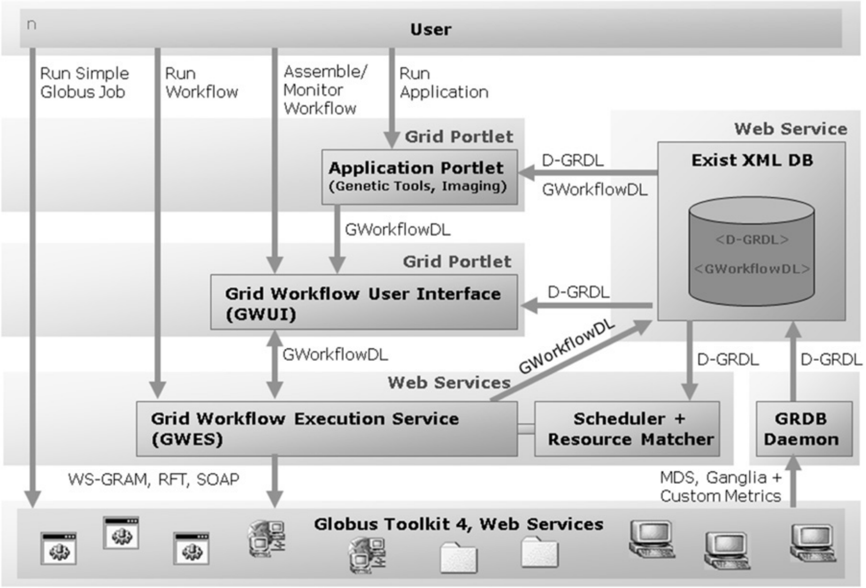
User
Run Simple Globus Job
Run Workflow
Assemble/ Monitor Workflow
Run Application
Grid Portlet
Application Portlet (Genetic Tools, Imaging)
Web Service
Exist XML DB
D-GRDL
GWorkflowDL
<D-GRDL>
<GWorkflowDL>
GWorkflowDL
Grid Portlet
Grid Workflow User Interface (GWUI)
D-GRDL
GWorkflowDL
GWorkflowDL
Web Services
D-GRDL
D-GRDL
Grid Workflow Execution Service (GWES)
Scheduler + Resource Matcher
GRDB Daemon
WS-GRAM, RFT, SOAP
MDS, Ganglia + Custom Metrics
Globus Toolkit 4, Web Services

**Fig. 3.** Components involved in job execution using GWES (see text).

a unified and transparent access to a high number of distributed heterogeneous storage resources. In contrast to dCache, which is still in development, SRB is a matured DGMS providing higher abstraction level as it presents the user with a single global logical namespace or file hierarchy. The SRB DGMS has features to support collaborative management of distributed data including: controlled sharing, publication, replication, transfer, attribute based organization, data discovery, and preservation of distributed data. Access is secured by using X.509 certificates instead of username and password, but SRB has a separate user management and is not connected to Globus mapfiles by default. Each user has an own home directory in SRB which is similar to the home directory in a local filesystem; user and group access rights like read and write can be configured for files and directories. As SRB is widely used in grid environments, there are many tools to access SRB. MediGRID runs an SRB installation with distributed resources in Berlin, Dresden and Göttingen, managing about 80 TB storage space. We have developed an automatic creation and mapping of SRB accounts to enable single sign-on. Collaborative data handling is managed by group accounts, while user access is realized by integrating the GridSphere portlet developed within the BIRN-project [19].

### 3.2. GWES

GWES is a workflow manager established within the K-WF-grid [20,17]. The core of the GWES is the grid Workflow Description Language (GWorkflowDL), which is a Petri net based standard for describing workflows using XML. A Petri net – as a mathematical formalism to describe discrete distributed systems – allows for simple and intuitive modelling of complex distributed workflows, especially parallel processing. GWES uses high level Petri nets (HLPN) for workflow description, as they can be used directly in order to model transfer and storage of input and output data as well as control data (e.g. the exit status of a workflow step). The resulting workflow description can be analyzed for certain properties such as conflicts, deadlocks, and liveliness using standard algorithms for HLPNs. High-Level Petri nets can do anything that can be defined in terms of an algorithm [21]. GWES descriptions can be realized on several abstraction levels, which are then concretized by scheduling and user interaction during runtime. As every process execution within a workflow can be confined to selected grid nodes by appropriate resource descriptions or even be constrained beforehand in a concrete workflow description, *a priori* tracking can be incorporated for every desired level of security. GWES offers persistent checkpointing and maintains the state at any stage in the workflow (transfer) execution. This feature enables process tracking as required for medical applications. An implementation of fault-tolerance strategies for reliable process execution is accomplished within the MediGRID project. If an execution step fails, the error is reported and the transition is rescheduled to another resource up to an adjustable number of retrials. All medical image and biosignal processing applications and most of the bioinformatics applications in MediGRID are now implemented as GWES workflows. The generic GWES portlet (GWUI) allows for upload of workflow-descriptions and monitoring of running workflows. Direct upload is possible from clinical environments. But as the formulation of workflow-descriptions require knowledge about GworkflowDL, only experienced users may use this option. The default way to initialize a workflow are the application specific portlets. Several workflow templates, defining data flow and software components, are deposited in the portal. The user has to select the input data (and if needed additional setup parameter). When initializing the workflow, the template is complemented and passed to the Execution Service. The decision, which computing resources are used for the individual steps of the workflow or the physical storage where the data is taken from, is left to GWES. GWES provides – as mentioned above – basic resource brokering and scheduling — based on the information provided by the D-Grid Resource Description Language (D-GRDL, Fig. 3).

The progress of the workflow execution is monitored within the workflow portlet component. An example is given in Fig. 9, Section 5.3.

## 4. User management and security

### 4.1. PKI-based login and access to application services

PKI based authentication and authorization provides all legal features for fully secured grid usage. Therefore, the D-Grid PKI and VOMRS infrastructure, provided for all communities, is chosen as the default way to register to MediGRID. The registration involves several steps a user has to go through (Fig. 4): The user first needs to request a PKI certificate at a trusted certification

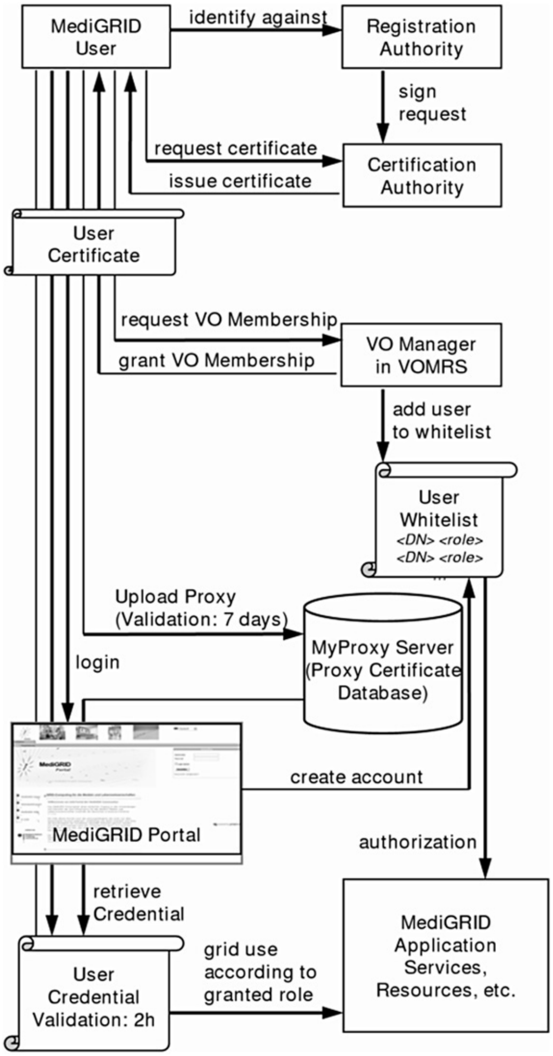

**Fig. 4.** User registration and management in Medigrid. Several processing steps have to be accomplished during registration.

authority (CA). This task can be accomplished even from clinical environments by using the webbased graphical user interface provided by the CA [22]. The private key is saved into the current browser. The identification against the CA usually requires a local trusted registration authority (RA), which the applicant physically has to visit. The approved certificate is sent back per mail and has to be loaded into the browser which was used for the request.

The next step is the login at the VOMRS. In case the validity check of the user certificate is passed, the user is identified by the system and can request membership in different virtual organisations, among them MediGRID. In MediGRID, the facility of a more fine-grained differentiation into sub-VOs (so called groups) is enabled by our user management for a simple modeling of roles as a first step towards role based access control within the grid. During the registration process the applicant has to accept the usage policies of the respective VO, so a certain legal basis for the provision and use of grid resources is given. Any membership application has to be granted or denied by the responsible VO and/or group managers. As the information from the VOMRS can be retrieved by trusted services, all grid resources are kept up-to-date with respect to the user registrations; and the necessary local accounts, role mappings and authorization rules are implemented automatically. Within the MediGRID project, the GridSphere Portal Framework has been extended in terms of user management functionality by linking it to the VOMRS (Fig. 4). Furthermore, an extension for certificate-based login reduces the portal login and registration effort and lowers usage barriers. Once the user possesses his grid certificate he or she will face the second large barrier on the way to use grid applications, services and resources. It is possible for the user to log on to the portal, as the primary user interface for MediGRID, but this does not mean that grid applications, services and resources can be instantly used. For authentification and authorisation between the middleware components, they need access to a complete certificate pair (public and private key) of the user, which is usually solved by issuing a grid credential, generated from intermediary proxy certificates stored on the MyProxy server [24]. The credentials can be retrieved from the MyProxy server via the grid portlets [23] provided with the GridSphere Portal Framework. The upload of grid proxy certificates to the MyProxy server can be performed in MediGRID by using the MyProxy Upload Tool [25]. It is implemented as a Java WebstartTM application which can be started from the grid portal: it allows for local conversion of certificates into the necessary formats and for the setup of a secure communication channel with the MyProxy server.

Within the described process, two main challenges for operability and usability are identified by practical experience within the MediGRID project: In a heterogeneous user community like MediGRID, with participants from several organisations and foreign research partners scattered all over the world, the setup of a RA for each potential participant is a barrier that is difficult to overcome. Especially if the processed data is not sensitive and the usage of the grid would imply just a few visits, the bureaucratic effort is not in line with the prospected benefit by the users, in particular if they have no experience with grid computing and are not able to explore the grid capabilities beforehand. A trust fabric as e.g. realized by *caBig* [26], is not compatible with current D-Grid policies. At least for potential users from Germany, a practical solution has been found with setup of registration authorities within medical societies or similar subcommunity associations.

The second major challenge is the grid proxy upload. Even the best currently available lightweight solution, the MyProxyUploadTool proved to have significant drawbacks in terms of usability and operability in practice. During the download of the tool, the user has to accept several security notifications as the tool needs to be executed locally. This usually causes uncertainties and concerns among the users. Furthermore, there are still a lot of configuration options to be set manually by the user. The MyProxy Upload Tool appeared to create a great demand for user support. The main problem is caused by the fact, that the MyProxy upload tool requires communication to TCP port 7512 of the MyProxy server, which will not be allowed by standard clinical firewall configurations, as mentioned above (see Fig. 2). Currently, users in clinical environments have to convince their IT-administrators to grant connection permission, or must transfer their credentials and accomplish the task e.g. at home. Today, a significant number of potential users showing great interest in the applications and services provided by MediGRID are discouraged or even deterred.

### 4.2. Guest accounts for least barrier access

In order to provide easy access for applications with low security requirements (see Table 1), a concept for a low barrier (but still personalized) guest user registration and access has been developed and implemented in MediGRID [27].

Guest users are not required to own a certificate. This avoids both mentioned obstacles on the user's path to the grid. Every person can register to the portal with username, email address and password. Activation of the account will be enabled automatically, when the email address is verified. The guest user gets a

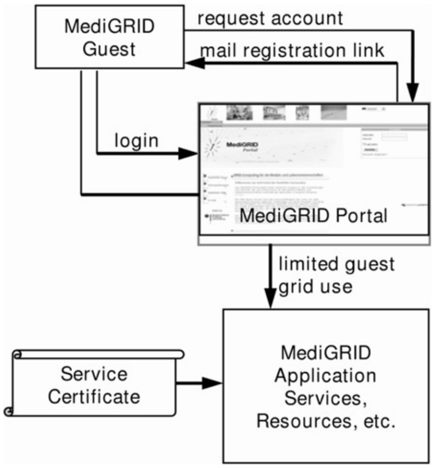

**Fig. 5.** Guest user registration and management in Medigrid without personal grid certicate. The suggested solution using service certificates is being discussed. Current resource provider policies still prefer personal certificates, as the grid-mapfile based authorization does not allow fine grained access control yet.

personalized account with guest status and limited rights and functionality (Fig. 5).

After registration, guest users can access defined services simply by logging on to the portal with their username and password. However, in the background these services still need to use credentials for communication with and access to the grid resources. In MediGRID this is realized completely transparent to the (guest) user. The respective services use service certificates for this purpose instead of user credentials. These service certificates are technically identical with machine certificates. In analogy to machine certificates, the CA registers an administrator during the process of issuing the certificate, who is responsible for this service. The GWES feature of passing arbitrary information within the workflow is used to pass the guest user ID as a parameter with each job executed in the grid. It allows for a tracing of resource and service usage down to a specific guest user for auditing purposes. In the case misuse should happen, the affected account can be closed down the same way as for regular user accounts. The email address obtained during the guest registration process also gives some chance of tracing back the user to his physical location in cases of significant misuse.

## 5. MediGRID portal development

As mentioned before, usability and operability within clinical environments is a vital prerequisite for acceptance of health grids. It encompasses easy access to the grid without elaborate client installations or system configurations. On the other hand, today virtually everybody is familiar with using an internet browser for search, email and e-commerce. Therefore, a web-based portal as main entry point into the grid is predestined for user acceptance. They can access the grid from workplaces with strict firewall requirements as well as from every computer providing internet access. The user may start, control and download grid jobs using a conventional internet browser. The user-side installation reduces to some freeware browser plug-ins for full exploitation of the provided MediGRID applications. At the moment Java and VRML plugins are recommended, but not vital. The portal is realized with the GridSphere Portal Framework. It comes with some predefined gridportlets for basic credential-, data- and job-management within a Globus-based grid infrastructure. The application specific portlets are developed in Java following the JSR168-portlet standard for portable web components. We want to emphasize, that the strict limitation to web-based connections to the user site makes complex interactivity with grid applications challenging. Existing client solutions for interactive and collaborative work within grid environments cannot be adopted to the portal, if they imply further transfer protocols. Therefore, a variety of desired grid functionality in MediGRID has to be integrated into the portal by development of generic portlets or portlet components and application specific portlets. To give insight into the achieved results, some are exemplarily described in detail in the following sections

### 5.1. Ontology components

In recent years, ontologies have emerged as a key concept to support understanding and exchange of information, especially in the life sciences [28]. They are primarily used to semantically and uniformly describe biomedical objects with structured domain knowledge in terms of ontology concepts. These concepts are connected through semantic relationships, principally *is-a* and *part-of*, and thus form specialization/generalization hierarchies (taxonomies) or more complex acyclic graph structures.

The rapid increase in the number of available ontologies in the life sciences leads to ontology access and integration problems which likewise affect applications in grid environments. In MediGRID varying applications of dissimilar life sciences domains (bioinformatics, imaging, clinical research) need a platform for a uniform and simple ontology accessibility within the grid and want to integrate information of these ontologies in their application portlets. Existing ontologies developed and managed in different projects, institutes or research programs present heterogeneity in source formats and syntax. Particularly, ontology sources range from relational databases, structured files like XML, OWL, OBO [29] or CSV to web services allowing a service-based access. Using and extending the OGSA-DAI framework, an ontology access middleware is developed within MediGRID [30]. Currently, 15 ontologies of different biomedical domains are uniformly accessible within the grid, including GeneOntology, NCIThesaurus, SequenceOntology, CellOntology and RadLex. The approach is flexible and generic; new ontologies are added and included within the middleware by simply adding or extending adaptors.

The central Ontology Access Portlet serves as a look up service and information resource for all ontologies integrated in MediGRID. Main entry point is the Search component. A simple list allows the selection of an ontology of interest. Currently, we offer different search possibilities for concept/term look up. Users with background knowledge about a specific ontology can directly input an accession number identifying a concept within an ontology. Furthermore, keyword-based search capabilities which optionally make use of suggestion functionalities to help users finding their desired ontology concepts are provided. After a search request is submitted, corresponding ontology information of found concepts is displayed in the result component (Fig. 6). MediGRID is using several display techniques to help users navigate and browse in available ontology information. In particular, users are supplied with information about ontology concepts, namely its ID, description, synonyms and references to other ontologies/data sources. Furthermore, the result component uses the semantic relationships between ontology concept to show the local environment of the concept (semantic neighborhood). Links on displayed concepts are used for navigation within the entire ontology graph, i.e. users can browse to concepts that are more special or more generic compared to the selected one. Finally, the use of Web 2.0 Ajax features (trees, asynchronous requests) enables users to dynamically navigate through ontology graphs (Fig. 6). Application specific portlets can interlink to the Portlet to retrieve ontology information about results or important concepts.