



WP 4.3 ILDG File Catalogue Security

Project Title: QCDgrid

Document Title: WP 4.3 ILDG File Catalogue Security

Document Identifier: QCDGRID2-FC-SEC

Document Filename: QCDGrid2-Security-Specification.doc

Distribution Classification: Public

Authorship: Radoslaw Ostrowski and George Beckett

Approval List: QCDgrid Project Management Board

Distribution List: Public

Document History:

<i>Personnel</i>	<i>Date</i>	<i>Summary</i>	<i>Version</i>
MGB/RHO	20/FEB/07	First Release	1.0

Contents

1	Introduction	3
1.1	The QCDgrid Project	3
2	Background to the ILDG file catalogue.....	4
2.1	Glossary	5
3	Security Considerations	6
3.1	User classification and a grid-mapfile.....	6
3.1.1	What is a grid-mapfile?.....	6
3.1.2	Users types.....	6
3.1.3	Pooled accounts and multiple pooled accounts	6
3.1.4	Generation of the grid-mapfile.....	7
3.2	Access control on the file level	7
3.3	Access control to the RLS	8
3.4	Trust delegation mechanism (for the Web Service).....	9
3.4.1	Security on the client side	9
3.4.2	Security of the transport layer.....	9
3.4.3	Security on the server side	10
3.4.4	Why not use a host certificate to query the RLS?	10
4	Conclusions.....	11
5	References.....	12

1 Introduction

1.1 The QCDgrid Project

The QCDgrid project [6] is a core activity of UKQCD [5], a collaboration of UK academics and researchers that aims to procure and jointly exploit computing facilities for lattice field theory (commonly referred to as Lattice QCD) calculations. The primary aim is to increase the predictive power of the *Standard Model of elementary particle interactions* through numerical simulation of *Quantum Chromodynamics*. Such numerical simulations produce significant amounts of data and the purpose of the QCDgrid project is to provide software and supporting infrastructure that simplifies the management, storage, and manipulation of this data.

In the first three years of the project (2002—2004), software engineers at EPCC developed QCDgrid—a data management system that combines the distributed resources of the collaborators into a robust facility called the *UKQCD Grid*. The result is a multi-terabyte storage facility over seven sites at: University of Columbia, University of Edinburgh (including the UoE Advanced Computing Facility), University of Liverpool, Rutherford Appleton Laboratories (RAL), University of Southampton, and University of Wales Swansea. The University of Glasgow is also a member of the consortium.

The facility is based on commodity hardware and open-source software. The hardware consists primarily of high specification, PC-based servers running the Linux operating system and managing large RAID storage arrays. On top of this infrastructure, the QCDgrid software (built using components from the Globus Toolkit [2], EGEE application stack [1], and an XML database) provides *Datagrid* management and user functionality – furnishing a simple and intuitive environment that hides the complexities of the underlying grid and presents a standard file system to the user. It incorporates a robustness metric that automatically disperses datasets across the grid, providing a resilience that ensures data is not affected by the loss of one (or possibly more) storage nodes. Security is leveraged from the Globus Toolkit, based on X.509 digital certificates issued by an approved Certificate Authority. The result is a reliable and secure data management system.

UKQCD is an important contributor to the International Lattice Data Grid (ILDG) [3], a group of like-minded scientists, working around the world, who aim to share their data to accelerate scientific progress in the field of Lattice QCD. The ILDG was initiated in 2002 by UKQCD and, at the time of writing, has significant representation from research groups in Australia, France, Germany, Italy, Japan, UK and USA.

The ILDG infrastructure is being assembled as a web services layer that will aggregate the resources of each contributing collaboration (that is, regional grid such as the UKQCD Grid) for the benefit of the wider community. To achieve its objectives, ILDG has established two working groups:

- Metadata Working Group – to facilitate data sharing, through the standardisation of the format and content for Lattice QCD scientific data and associated metadata.
- Middleware Working Group – to produce a set of specifications that define an architecture for an international *Grid of Grids* for Lattice QCD.

This document describes the security considerations for the *ILDG File Catalogue*, a key component of the ILDG infrastructure. In Section 2, we explain the purpose of the file catalogue in the context of the wider infrastructure. Then in Section 3, we consider several security aspects like user authentication, security of the stored data, access control to the RLS and trust delegation. In Section 4, we draw together conclusions, based on the work to date, and establish the next steps towards a functioning implementation.

2 Background to the ILDG file catalogue

In a *grid* context, a file catalogue is a registry that binds unique file identifiers (commonly referred to as Logical Filenames (LFNs)) to one or more instances of (internet) locations, or URLs, where a copy of the specific file exists.

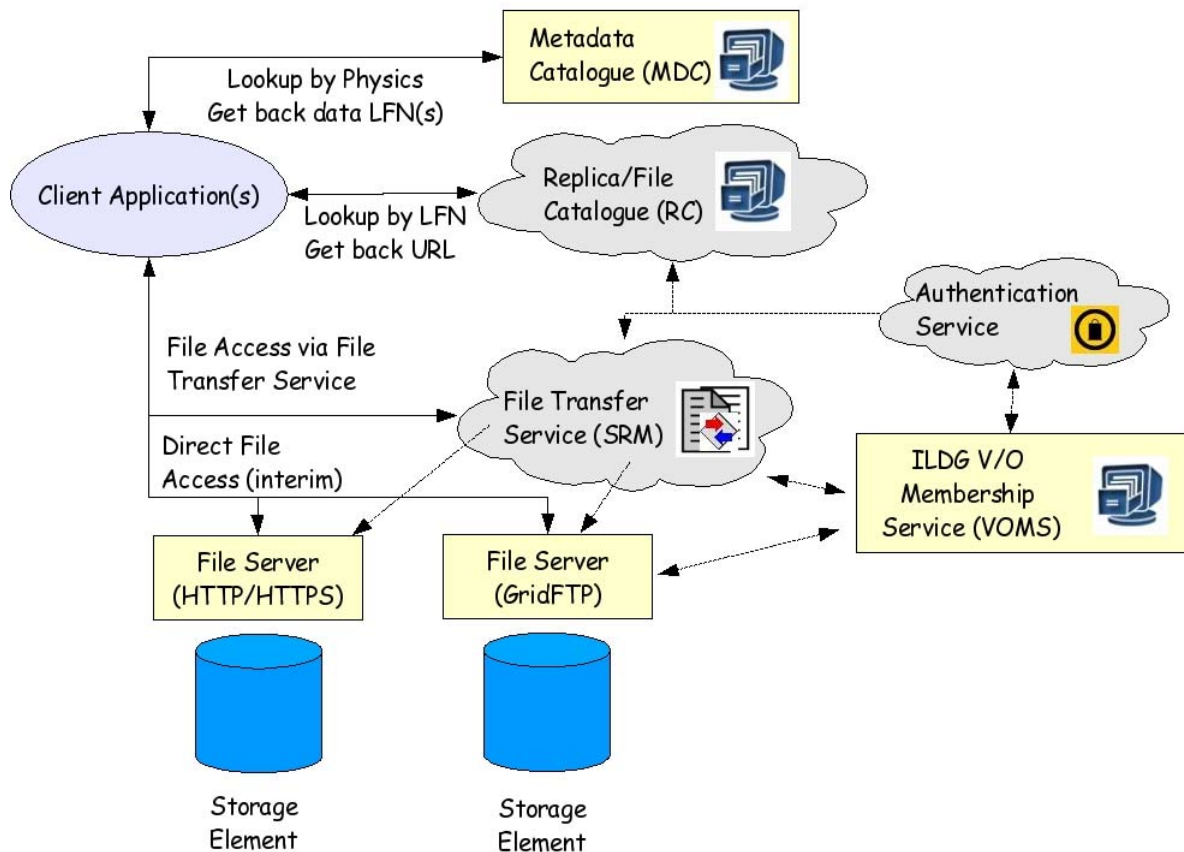


Figure 1: Modular overview of the ILDG infrastructure

A file catalogue is a core component of the ILDG infrastructure (see Figure 1) that is used to track the location of Lattice Gauge Configuration datasets within the storage resources provided by the regional grids. It is most powerful when combined with the ILDG Metadata Catalogue (see QCDgrid project deliverables from Work Package 3 [6]), which lists the lattice gauge configurations that are available, described using meaningful and standardised physics terminology and marked up using QCDML. The file catalogue contributes to the ILDG’s “Search and Retrieval” use cases [4][7].

At the time of writing, each regional grid has implemented – or is in the process of implementing – a local file catalogue to manage the datasets owned by the particular collaboration. Some groups, such as UKQCD, CSSM (Australia) and LDG (Germany), have elected to use a third-party catalogue: UKQCD and CSSM use the Globus Replica Location Service [2], while LDG use the LCG File Catalogue Service [1]. Other collaborations, such as USQCD (USA) and JLDG (Japan), have elected to build their own file catalogues from scratch.

Both of these approaches (using a third-party catalogue or writing one’s own from scratch) have advantages for the individual collaborations. However, for the ILDG as a whole, the different approaches imply that member collaborations present incompatible interfaces and provide different functionalities to the potential user. This is not a new situation for the middleware working group: it was previously encountered and successfully overcome during the specification of the ILDG Metadata Catalogue. As for the metadata catalogue, the middleware working group propose to overcome these

incompatibilities by introducing a web service layer that will present a standard interface on top of the regional catalogues.

The determination of the form and function of this web service layer has been a focus of discussions within the working group over the previous 12 months. The QCDgrid project team, as representatives for UKQCD to the middleware working group, have been heavily involved in this process through both face-to-face meetings (Japan, October 2005; Germany, July 2006; and USA, December 2006) and monthly teleconferences of the working group. During the December 2006 meeting, the working group arrived at an agreed specification for the file catalogue that is to form the basis for the design and implementation activities of the regional grids.

2.1 Glossary

Logical file name – a unique identifier for the contents of a file.

Logical name – a unique identifier for the contents of a data item.

Physical file name – the address or the location of a copy of a file on a storage system.

Replica Location Service (RLS) – a registry that keeps track of where replicas exist on physical storage systems. The job of the RLS is to maintain associations, or mappings, between logical names for data objects and one or more target or physical names for replicas. Users or services register data items in the RLS and query the RLS to find copies of data items.

Authentication – a process of identifying a user. Usually, it is done by requiring the user to provide credentials (username and password) which are then compared with those stored in a database. In the Grid Security Infrastructure (GSI), these credentials are provided by X.509 certificates.

Authorisation – a process of enforcing policies. Once a user has been authenticated, decisions can be made as to what types of services or resources this particular user is allowed to utilise.

Delegation – an act of transferring rights and privileges to another party.

Delegatee – a requestor of delegation, receives delegated credentials from Delegator.

Delegator – an entity that delegates the abilities and/or rights to the Delegatee.

TLS – Transport Layer Security is a protocol which ensures privacy between communicating parties on the Internet. When a server and client communicate, TLS ensures that no third party may eavesdrop or tamper with any message sent over the network. TLS has superseded the Secure Sockets Layer (SSL).

Proxy Certificate – X.509 certificates issued by the end-user (the *Delegator*) to a process acting on the end-user's behalf (the *Delegatee*). The proxy certificate carries the identity of the Delegator: that is, it can be used to establish TLS connections and is interpreted using the GSI as authority to perform work on the Delegator's behalf.

3 Security Considerations

Importantly for the security of the UKQCD Grid infrastructure, there is a need to control user access to it. Firstly, we have to distinguish different users who have and have not the right to use the grid. This is done in a process of authentication. Secondly, having identified the user and their role, we can provide them with an appropriate set of permissions to utilise any of the available services or resources, or deny such access. This process is called an authorisation. In this section, we consider both authentication and authorisation, as well as other security issues. Moreover, we highlight any specific features for UKQCD.

3.1 User classification (authorisation) and the grid-mapfile

3.1.1 What is a grid-mapfile?

A grid-mapfile is a list of distinguished names (user identifiers found in the X.509 certificates) that are allowed access to a service in the grid environment. Additionally, the file maps each distinguished name (DN) to a local UNIX user account.

The file format consists of two columns separated by a whitespace (DN and a corresponding user account) and as many rows as necessary for each user. DN is surrounded with quotation marks ("DN"), as it may contain whitespaces.

An example line from a grid-mapfile is presented below:

```
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=radoslaw ostrowski" radek
```

3.1.2 Users types

There are three types of user for the UKQCD Grid infrastructure: known as special, UKQCD and ILDG.

- Special users – two examples:
 - the `qcdgrid` user – which runs the control processes on the Control Node. A user necessary for proper running of a QCDgrid-powered infrastructure.
 - administrators of UKQCD Grid infrastructure
- UKQCD users – scientists who belong to the UKQCD collaboration
- ILDG users – scientists who belong to the ILDG collaboration, though who are not part of UKQCD.

Special users have normal UNIX accounts. Administrators have to have their own private accounts on each machine and possibly `sudo` access. In the grid-mapfile, their DN will be mapped to their local UNIX username.

UKQCD and ILDG collaborators use so called *pooled accounts*, described in the following section.

3.1.3 Pooled accounts

Pooled accounts allow a dynamic allocation of local UNIX usernames to grid users. A pool of local accounts is created by the system administrator using a normal account creation method, and these are leased to incoming grid users. Additionally, the leases for the accounts are revoked after a specified time and made available to other users. For more information about pooled accounts in general and about their generation process please refer to [8].

In order to use pooled accounts, the grid-mapfile has to be modified. Every DN, instead of being mapped to a UNIX username, must be assigned to a “.” (dot) character, as in the following example:

```
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=radoslaw ostrowski" .
```

DNs of users which correspond to a “.” in a grid-mapfile, will be automatically assigned to one of the pooled accounts. The same pooled account will be used for a particular user as long as the lease is valid. Upon the expiration of the lease, the account will be freed and returned to the pool.

It is also possible to use more than one pool of users. Multiple pools of accounts can be used in the UKQCD Grid infrastructure to distinguish the collaborators from UKQCD and ILDG groups. To achieve this, usernames with the same three letter prefix should be created followed by a unique number, as in the following example:

```
ukq000, ukq001, ukq002, ...
```

```
ild000, ild001, ild002, ...
```

The mapping for multiple pooled accounts in the grid-mapfile would then be, as follows:

```
"DN of UKQCD collaborator" .ukq
```

```
"DN of ILDG collaborator" .ild
```

Note, that the environment variable `GRIDMAPDIR` has to be exported prior to running the RLS service:

```
export GRIDMAPDIR=/etc/grid-security/gridmapdir
```

in order for pooled accounts to be supported. It should be done in the starting script, depending on how Globus is installed (`$GLOBUS_LOCATION/sbin/SXXrls` or `$VDT_LOCATION/setup.sh`). Exporting this variable will activate the pooled account patch and allow the RLS to properly understand the *dot-username* mappings.

3.1.4 Generation of the grid-mapfile

Creation of the grid-mapfile is a multi-step process. It is so, because the information about the users allowed on the grid has to be collected from different sources, depending on the type of the user. Additional complexity is introduced by employing both “normal” and pooled accounts.

Information about special users has to be stored in a separate file called “grid-mapfile-local”. This file has the same format as the grid-mapfile. It should consist of DNs and corresponding UNIX accounts for the `qcdgrid` user and the administrators of the UKQCD Grid. The file has to be created and maintained manually by one of the administrators.

A list of collaborators of UKQCD and ILDG can be downloaded from the VOMS server and appropriately parsed so it populates the grid-mapfile with the correct mappings to the pooled accounts – `.ild` for ILDG and `.ukq` for UKQCD collaborators.

A script can then be used to combine the information from the grid-mapfile-local and the VOMS server and store it in a grid-mapfile. If anyone is in both user lists (for example, an administrator), the mapping from the grid-mapfile-local should be used. This script should also provide a security feature that informs the administrators whenever new members are added to the VOMS service. Additionally, an appropriate mechanism should be used to store current version of the grid-mapfile before creating another one in case of unavailability of the VOMS server. On this occasion, UKQCD Grid administrators should also be notified. The grid-mapfile should be generated on a regular basis, to help keep the local list synchronised when users are added or removed from the collaborators list on the remote VOMS server. The proposed interval for polling the VOMS server should not be bigger than 24 hours.

Table below presents an example usage of the grid-mapfile mappings to UNIX accounts:

DNs	Corresponding mappings
"DN of qcdgrid"	qcdgrid
"DN of admin" (radek, jamesp etc.)	admin (radek, jamesp)
"DN of UKQCD collaborator"	.ukq
"DN of ILDG collaborator"	.ild

3.2 Access control on the file level

As there will be different types of users allowed on the grid, we have to make sure that data files which are considered *private* are only accessible by users with appropriate rights. In order to achieve this, we will protect the files using their UNIX file permissions.

We will consider two cases (see Table 1 below). The first case is for files that are private to a certain group (in this case, UKQCD). The second case is for files that are public: that is, available to any ILDG members.

Cases	OWNER (qcdgrid)	GROUP (UKQCD)	OTHERS (ILDG)
1. Files private	RW-	R--	---
2. File public	RW-	R--	R--

Table 1: Proposed file permissions for UKQCD specific and public data files.

All files on the grid should be owned by the `qcdgrid` user and belong to UKQCD group. The `qcdgrid` user should be a member of UKQCD group. By default, files are set to not be readable by users outside of the UKQCD group: that is, they are considered as private to UKQCD group – case 1.

In case 2, all users should be allowed to read the data. Files can be made public by any of the UKQCD collaborators, by running a command such as `chgrp <groupID> <LFN>`.

3.3 Access control to the RLS

As there are three kinds of users in the UKQCD Grid infrastructure, we would like to ensure that everyone has the appropriate access to the RLS.

First of all, the control thread, running on the central node, should be able to perform every operation which is enabled on the RLS server. Similarly, so should administrators of the infrastructure.

Secondly, collaborators of UKQCD and ILDG should not be allowed to change the content or the configuration of the RLS server, therefore, they should only be granted read access to the catalogue.

However, the most important security feature is to disable access to the RLS for any user that is not a member of ILDG: that is, users who are not registered with the VOMS service, nor are in the local `grid-map` file.

The configuration file of the RLS server is located in: `$GLOBUS_LOCATION/etc/globus-rls-server.conf`. It allows the configuration of different permissions for different users using so called access control lists (ACL). ACL is a list of permissions attached to an object (RLS server in this case). The list specifies who or what is allowed to access the object and what operations are allowed to be performed.

ACL entries, in the RLS configuration file, may be a combination of distinguished names (DNs) and local UNIX usernames [9]. If a DN, of a user who is trying to access the RLS, is not found in the configuration file, then the `grid-map` file is used. It is searched for a mapping of this particular DN to the local user account, which in turn is matched against the entry in the ACL list to determine the user's permissions. There may be multiple ACL entries, with the first match found used to determine a user's privileges.

There are seven permission values, as follows:

- `lrc_read` – allows client to read an LRC.
- `lrc_update` – allows client to update an LRC.
- `rli_read` – allows client to read an RLI.
- `rli_update` – allows client to update an RLI.
- `admin` – allows client to update an LRC's list of RLIs to send updates to.
- `stats` – allows client to read performance statistics like number of LFNs.
- `all` – allows client to do all of the above.

To summarise, the RLS allows authorisation based on DN or local usernames (if a `grid-map` file is used) to make calls to the server.

3.4 Trust delegation mechanism (for the Web Service)

The ILDG File Catalogue Web Service (detailed description can be found in [10]) offers two different modes of access: unsecured and secure. UKQCD must use the secure mode, because only authenticated users (with valid X.509 certificate) will be permitted to access the UKQCD Grid file catalogue and file servers. Moreover, access rights are user-dependent. In order to achieve this, a delegation of authority by a user to the web service must be performed. As this has significant security implications, both for the user and the service, it is considered in detail in this section.

These implications were considered carefully and, to this end, a number of security experts have been consulted including: David Bianco, a cyber-security analyst from Jefferson Laboratories, USA; plus Rachana Ananthakrishnan and Charles Bacon from the Globus Security team.

The security of the trust delegation mechanism will be analysed separately for the client, the server and the transport layer in between them. Note that readers unfamiliar with the basic concepts of the Grid security are advised to consult [13] for an introduction on this matter.

3.4.1 Security on the client side

Importantly, the private key used to sign a user's X.509 certificate must be kept secret. It is a user's responsibility to protect it. If the private key is compromised, the user's identity is effectively stolen and anyone could pretend to be this particular user on the grid.

Another security risk is when a user tries to contact a server, but doesn't verify if the server is actually the one it claims it is. Should this happen, the client might delegate its authority to a malicious server which might then use the user's credential against their will. Therefore, if an act of delegating credentials is to be carried out, the destination server has to be authenticated first.

This feature is offered by the Transport Layer Security (TLS) protocol (successor of Secure Sockets Layer – SSL). It can be configured so that mutual user and server authentication is required, prior to initialising the connection. This is commonly referred to as a *handshake*. Client's and server's identities are confirmed basing on their X.509 certificates.

3.4.2 Security of the transport layer

This section investigates the most appropriate method for data transfer between the client and the server. It is essential to send the data through an encrypted channel to prevent eavesdropping, tampering, and message forgery. To achieve this, the TLS protocol should be used.

Moreover, the security of the transport layer also depends on the mechanism used for delegation of the authority. The current approach is listed below:

- a. Client creates a proxy certificate (contains newly generated public and private keys).
- b. Client sends it over an encrypted channel to the server.
- c. Server can use the proxy to query the RLS server.

There is a potential security risk involved in the above. If the proxy is intercepted, someone could use it and pretend to be the original client, as it contains the private key.

An alternative mechanism is described in the delegation protocol used by both Globus Toolkit [14] and gLite Delegation Service [11][12]:

- a. Client asks server to create a proxy for them (server generates public and private keys).
- b. Server sends the proxy stub to the client (only certificate with the public key is transmitted, the private key stays on the server).
- c. Client signs the proxy and sends it back to the server.
- d. Server can use this signed-by-client proxy together with the private key to query the RLS server.

The risk existing in the previous case is here avoided. However, if TLS protocol is used and is confirmed to be secure, then these two cases are equivalent.

Note that there is also a concern of re-using the keys generated for the proxy certificates in case they are compromised. This can be remedied by generating a new key pair for every new delegation.

3.4.3 Security on the server side

Not surprisingly, once the delegated credentials are transferred to the server, there are still some security risks. Firstly, they can be stolen by someone who has access to this machine and used in any way. Secondly, delegated credentials can be used by the service to carry out an operation not agreed to by the user. And finally, a malicious system administrator has full access to the server and can easily steal the credentials.

It is relatively simple to solve the first case. The credentials should be stored securely, so that they can only be used by the service. Setting appropriate permissions would enable the protection offered by the file system, in the same manner as for any proxy certificate.

The second and third cases are more difficult to deal with and cannot be completely avoided. However, their effects can be mitigated. One way to limit the misuse of a compromised proxy certificate is to set it up with a short lifetime, only long enough to complete its task. Additionally, enforcing some further restrictions policies on the proxy certificate would minimise the negative effects. Disabling the proxy from delegating further as well as limiting its use only to querying the RLS would be the most desirable restrictions, however such security restrictions are not widely supported by grid middleware providers, at the time of writing.

3.4.4 Why not use a host certificate to query the RLS?

The main reason for this is that the RLS has its own system for controlling access: if the host certificate is used, then a new mechanism needs to be implemented. Not only will it be time consuming and represent a burden for the web service, but it also might not work as well as the existing solution. Therefore, we believe it is best to leave the RLS to control access to its resources.

4 Conclusions

The specification of the file catalogue needs to accommodate more complex security requirements than the metadata catalogue and this has yielded some challenges in determining a robust process for trust delegation that is independent of the choice of middleware. Several solutions have been considered in detail and proposed in this work package.

There will be three types of users on the Grid: special users (`qcdgrid` and administrators), collaborators from the UKQCD, and collaborators from the ILDG. All files will be owned by the `qcdgrid` user, but UKQCD collaborators will have access to them. All files will be – by default – private to the UKQCD group, but any collaborator from this group can make the files public. Public files can be read by the ILDG collaborators. Access controls on the data will be handled by the file system.

As far as the access control to the RLS is concerned, it will be done by the RLS server itself based on its Access Control Lists (ACL) and the `grid-mapfile`.

A user's proxy credentials will be used to query the RLS server. An authority delegation to the web service will be done employing Transport Security Layer (TSL) protocol. Moreover, to minimise a risk of stealing the proxy credentials, they will be short-lived.

To sum up, this document reviews a security of the File Catalogue and its Web Service access. All the specified information should be considered when designing architecture for QCDgrid implementation of ILDG interface to File Catalogue (Work Package 4.4).

5 References

- [1] EGEE, *gLite – Lightweight Middleware for Grid Computing*. Project homepage at <http://glite.web.cern.ch/glite/> (2006).
- [2] Globus Alliance, Globus Toolkit. Homepage at <http://www.globus.org/>.
- [3] International Lattice Data Grid. Homepage at <http://www.lqcd.org/ildg/>.
- [4] M. Sato, *Lattice QCD Data Grid Middleware: The Metadata Catalogue*, Presentation to ILDG Workshop (2004). Electronic copy available at <http://www.rccp.tsukuba.ac.jp/workshop/ILDG-4/pdf/MitsuhsaSato.pdf>.
- [5] UKQCD Collaboration. Homepage at <http://ukqcd.epcc.ed.ac.uk/>.
- [6] UKQCD, *QCDgrid: Probing the building blocks of matter with the power of the Grid*, QCDgrid project homepage at <http://www.gridpp.ac.uk/qcdgrid/>.
- [7] T. Yoshie, *Status of ILDG Activity*, Presentation to ILFTN (Edinburgh, 2005). Electronic copy available from http://www.nesc.ac.uk/talks/464/Session7/ILFTN2_yoshie.pdf.
- [8] Pool Accounts patch for Globus, <http://www.gridsite.org/gridmapdir/>
- [9] GT 4.0 RLS: System Administrator's Guide, <http://www.globus.org/toolkit/docs/4.0/data/rls/admin-index.html>
- [10] R.H. Ostrowski & M.G. Beckett. WP 4.2 Functional specification of ILDG File Catalogue
- [11] GridSite, Delegation protocol, http://www.gridsite.org/wiki/Delegation_protocol
- [12] Gridsite Delegation, <https://twiki.cern.ch/twiki/bin/view/EGEE/GridSiteDelegation>
- [13] GT 4.0 Security: Key Concepts, <http://www.globus.org/toolkit/docs/4.0/security/key-index.html>
- [14] GT4 Delegation Service Developer's Guide, <http://www.globus.org/toolkit/docs/4.0/security/delegation/developer-index.html>