

Recollections of the ILDG Middleware Convenors Face-To-Face Meeting, Jefferson Lab, Dec 11-13, 2006

Bálint Joó

May 10, 2007

1 Introduction

These notes summarize the points of discussion and in session activities that took place at the 3rd ILDG Middleware Convenors Face-To-Face Meeting that was held at the Jefferson Lab from through the 11-13 December, 2006. Where appropriate the report may also be supplemented by presentation material.

2 Dramatis Personae

The following persons were present for most or all of these discussion

- George Beckett, EPCC, U.K. for UKQCD & QCDGrid (now DiGS)
- Noriyoshi Ishii, CCS Tsukuba, Japan for JLDG
- Bálint Joó, Jefferson Lab, U.S for USQCD
- Radoslaw Ostrowski, EPCC, U.K for UKQCD & QCDGrid (now DiGS)
- Dirk Pleiter, DESY, Germany, for LDG
- Mitsuhsa Sato, CCS Tsukuba, Japan for JLDG
- James Simone, Fermilab, U.S. for USQCD
- Chip Watson, Jefferson Lab, U.S. for USQCD

3 High Level View Of Discussion Items

The following key topics were discussed or addressed through joint activity in session:

- Brief Status of Regional Grids (Monday)
- Namespace Awareness of Metadata Catalogs (MDC) (Monday)

- Metadata Catalog Operation, Usage and Experiences (Monday)
- VOMS Operation Reliability redundancy and VOMSRS (Monday)
- File Transfer Trials Between the sites (Tuesday)
- Discussion on access control (Tuesday)
- SRM Developers (Tuesday)
- Discussion about File Catalog Interface and Delegation (Tuesday, Wednesday)
- Lattice 2007 Milestones, client tools,distribution (Tuesday, Wednesday)
- Monitoring the ILDG Components (Wednesday)
- Miscellaneous Infrastructure: Software repositories, distribution of software via RPMs etc.

Apart from the discussions in session we have had the pleasure of discussing some item on the telephone. In particular we spoke to:

- dCache and SRM developers at Fermilab regarding the prospects for SRM.
- David Bianco, a Cybersecurity Analyst at the Jefferson Lab, about delegation and proxies
- Members of the Globus Security Development Team (visiting the University of Edinburgh)
- Paul Coddington, about the status of the ILDG effort in Australia.

Some of the topics spanned several sessions of discussion. In the discussion below, I attempt to summarize each topic, rather than provide a chronological transcript.

4 Brief Status of the Regional Grids

4.1 Australia - via phone with Paul Coddington

The status of the ILDG in Australia has been described in ILDG 9 however some pertinent issues were mentioned by Paul on the Phone.

In Australia, a dCache system has been set up, but it is not yet serving ILDG data. A new person has been hired to work on the grid infrastructure: Shunde Zhang. Shunde has already subscribed to the ILDG Middleware Mailing lists. He will set up the dCache to serve ILDG data through the SRM interface and at that time Australia can join in the File Service trials.

4.2 Japan

Japan has now a new domain: `jldg.org` and the Japanese ILDG effort now has a new website: `www.jldg.org`

Japan has a simple replica catalog deployed (using the old replica catalog interface) which is a one-to one map of the Logical File Name (LFN) to the Transfer URL (TURL).

File downloads are provided through HTTP. Users need to register at the web site and are given a user name and a password. These can then be used in tools like *wget*. The authentication then is simple HTTP Password authentication.

The metadata catalog (MDC) provides a low level *raw query* interface.

The web client for accessing the metadata catalog and the files is being revamped. It provides an interactive interface that can be used to search through the Japanese configurations. The search queries now go through the web services rather than through direct access to the MDC database.

Japan plans to have a two tier system of access: a public one and a Japan wide one. Each kind of access will have a separate Certificate Authority (CA). International certificates will be issued by the NAREGI/KEK CA while Japanese only certificates will be issued by JLDG. Having and maintaining the 2 CAs is essentially required by politics.

When data is made public it will be published in the public grid and be available to all.

Future plans are as follows:

- MDC - Work will focus on interactive ensemble queries and for querying the ensembles of other regional grids
- The file services will move to GFarm - using GSI certificate authentication - a Grid File system developed in Japan. A prototype between KEK and Tsukuba has been tested and will be rolled out to Kyoto, Hiroshima and Kanazawa.
- The move from prototype to production is expected in March 2007.

A question arose as to whether there would be any issues deploying a GridFTP service on top of GFarm and whether dCache is needed to provide the SRM interface. Since GFarm provides already a file system interface it does not seem useful to deploy dCache on top of GFarm. A standalone SRM-implementation may be considered in case SRM server functionality is considered necessary.

4.3 LDG

LDG has a fully vertically integrated setup its main features are summarized below.

In terms of Security LDG uses GSI authentication as implemented in the Globus Toolkit version 2.4. Anyone who is in the ILDG VO currently has read and write access to storage elements. There is an access control service which gives user control over access rights, however this only currently works in the MDC and the File catalog. A future plan is to extend this to also extend this to the storage elements themselves which will involve implementing Access Control Lists (ACLs) in dCache.

The MDC is built on top of a Hibernate Object Relational Database. Documents are mapped to Java objects through JAXB and JAXM technologies and these objects are persisted into Hibernate. Searches can be carried out by either mapping XPath queries into SQL and performing a database query, or if this is not possible, by materializing the Java Objects and querying them. Both authenticated and non authenticated access is provided.

The File Catalog used is the File Catalog component of LCG/EGEE called LFC. This provides a file system view of LFNs and it has authenticated access with POSIX like ACLs.

LDG Provides a VOMS service hosting the ILDG VO. However, the roles feature is not being used. Instead access rights are implemented or will be implemented via ACLs.

The storage elements (SEs) are run the dCache software in various setups. Some are directly attached to Hierarchical Storage Management (HSM) systems and some are not. The storage elements provide an external SRM v1 interface

which will move to SRM v2.2 soon, with GridFTP being the download protocol. While it may be possible to download with GridFTP directly, some sites have more than one GridFTP server and so SRM is needed to negotiate the transfer. In terms of uploading, currently the solution is that the client tools attempt to pre-register data in the file catalog before placing it on the SE. This fails if a user does not have sufficient privileges to add the configuration to the SEs. Of course this is not 100% foolproof since one could go directly to the SEs and the grid relies on users being responsible at this time.

Client software is provided in the form of JAVA applications and PERL scripts. Easy installation is provided through distributing the tools as Red Hat Package Manager (RPM) packages. The binaries are static to avoid shared library issues and the RPMs correspond to RPM v3. The installation uses a private package database and so does not need root privileges - in other words, a user can install the RPM into his/her home directory.

Several flavors of Linux are supported. These currently are: Red Hat Linux 7, Scientific Linux 3, Suse Linux 10, Debian 3 (Sarge), and Fedora Core 3. The Debian RPMs also work on Ubuntu systems.

Managing all these platforms is a little painful, especially for the GLite data management clients. Vitalization may help here in the future.

Some of the basic tools provide the following functionality:

lget - get a configuration or an XML document based on LFN

linit - set up a new ensemble

lput - upload a new configuration.

While there have not yet been explicit head to head trials, the Object Relational Database solution appears robust and speedy enough.

4.4 UKQCD

The UKQCD grid infrastructure is currently made up of 7 sites and is altogether capable of hosting some 80Tbytes of data. Currently there are about 50,000 configurations on line although only about 3000 have associated metadata.

The operation is driven by a control thread running on a central control node. Each site hosts dedicated data elements. There is also a backup node in case the central control node were to fail.

The Data Grid portion of the grid is implemented with a software called DiGS which used to be called QCDGrid. The renaming took place to widen funding opportunities and to move the software more into the mainstream (away from just QCD) so that it can be appropriately hardened.

The middleware technology has been migrated to Globus Toolkit version 4. The file catalog is implemented with Globus Replica Location Services (Globus RLS) and file transfers are managed through Grid FTP. A Job submission interface has been provided that allows staging of files to the jobs via the Data Grid. This is implemented over the GRAM component of Globus.

The MDC is implemented through the eXist native XML database, version 0.9. After a long wait version 1.1 release is available and the MDC may migrate to this.

Security is provided through the use of x509 certificates through mapfiles and pooled accounts. The mapfiles are nowadays generated from the VOMS server in LDG, although some still go through the old EDG/LDAP mechanism. Data is currently only readable to people with UK issued certificates.

In terms of client software, there is a command line interface and a GUI browser. The MDC has an ILDG compliant interface.

The grid is monitored through GiTS (Grid Interaction Test Suite) which is a PERL script based system for monitoring grid installations. Typical problems are to do with firewalls, the sensitivity of Grid Services to failure and that diagnosis can be difficult and time consuming.

The future work plan for ILDG focuses on the following packages

- ILDG Compliant File Catalog (by 2007, Q1)
- ILDG access to storage elements (by 2007, Q2)
- ILDG browser to include FC support (by 2007, Q3)
- SRM and tape support (by 2008, Q2)

with some UKQCD specific (DiGS v2) items:

- various performance improvements (by 2007, Q1)
- enhancements for file catalog (by 2007, Q2)
- implementation of Globus Reliable File Transfer (RFT) service (by 2007, Q4)
- improved monitoring (possibly through Globus MDS?) (2008 and beyond)

Funding is secured until Jan 2008. There is an application in process to extend this funding until August 2008, after which funding may be sought through the rolling grant system of the PPARC successor funding agency. The up side is that the funding will be long term, but the down side is that the effort funded may be quite low.

Using the VOMS server to generate the map files rather than the old LDAP strategy was straightforward to implement and no real problems were encountered with certificate revocation/deletion.

4.5 USQCD

The US infrastructure consists of an MDC and FC interface hosted at the Jefferson lab with storage elements being hosted at Fermilab.

The MDC back end is an exist 1.0 beta native XML database with an ILDG Compliant MDC front end web-service. The FC currently is a script that uses string and URL manipulation to map an LFN into an SRM URL pointing to the SRM server of the public dCache in Fermilab.

In Fermilab there are 2 dCache storage elements. There is the public dCache with an HSM where Carleton has been storing MILC configurations for ensembles with lattice spacings of 0.12fm, 0.15fm and the 0.09 fm.

There is also a volatile dCache which is used as a parallel file system for clusters.

In order to have read access to the storage elements you need to be a member of the ILDG VO. In order to write to the storage elements you need to have an account that can grant you a Kerberos ticket and this requires you to be a member of the US LQCD VO. (This is a subset of the FermiGrid and maps you to a Fermilab Kerberos certificate).

US experience has included difficulty with certificates. On most of the FermiGrid storage elements keeping the list of accepted certificates up to date on a storage nodes is now automated. However for the public dCache this level of automation has not yet been reached.

Other sites have also reported having difficulties keeping root certificates up to date. Some kind of automatic diagnosis for this would be desirable.

5 MDC Related Discussions

5.1 Namespace Awareness for Metadata Catalog

This discussion arose from the request by the Metadata Working group, for us to decide how we support namespaces in the Metadata Catalog, since every non backward compatible extension of the Metadata Schema is placed in a new namespace with an increased version number. So that versions 1.3.x should be backward compatible, but when an extension is made so that new documents can no longer be validated under the 1.3 schema the version gets increased to version 1.4. The main question is therefore: Do we wish to support queries under several namespaces in the middleware?

The discussion here revolved around several issues. The version 0.9 release of the *eXist* database supports queries within a namespace through its *XQuery* interface. However, experience in Japan suggests that this is quite slow although the most recent release of *eXist* promises performance improvements. Also at any one time, only one namespace can be the *default namespace*. A namespace aware query needs to *register* the namespace in which the query is performed and, unless the query itself includes the namespace prefixes in tags the namespace has to be in the default namespace. For example if we register the namespace:

```
xmlns:ensem = http://www.lqcd.org/ildg/QCDml/ensemble1.3/QCDmlEnsemble1.3.0.xsd
```

then queries need to be formulated using the the `ensem:` prefix eg as

```
/ensem:markovChain
```

and in order for us to be able to formulate queries without the prefix as `/markovChain` the namespace has to be registered as a default namespace. In this sense, supporting several namespaces would involve, for each supported namespace, the registering of that namespace as the default, executing the user query for each one and aggregating the results at the end. Alternatively, one could register all the supported namespaces, perform text processing on the query and insert the relevant namespaces into the queries and then execute each query. There are efficiency concerns with the first approach and the second one is complicated and error prone. The fact that not all MDCs are *eXist* based weighs further against supporting multiple namespaces. There is also the argument that documents in different versions of the schemata may not be backward compatible and so queries in one namespace may not be well defined in others.

For the various reasons above, it has been decided that the MDCs in regional grids ought not to support multiple incompatible namespaces. Rather the metadata should be transformed to adhere to the latest version. Essentially it is expected that only the ensemble documents will change which are few in number compared to configuration documents. We point out, that configuration and ensemble schemata need not change versions in lockstep. However, in order for this strategy to be successful, most metadata catalogs should update their documents to new versions of the schemata in a coordinated fashion so that the same query will work universally across all ILDG MDCs.

: Action Item: Bálint to formulate a response to the metadata working group outlining this position.

5.2 Operation and Usage experiences with MDC

Currently most ILDG MDCs do not provide web services for write access. In order to increase usage and uptake of the MDCs it would be desirable to add features to add/remove documents. This brings with it the questions of authentication/authorization and delegation. A standardization of such MDC services at ILDG level is not considered.

Our Australian colleagues have noted a feature of the US Metadata Catalog that returned the full contents of the database in response to a simple query.

Some implementations of the MDC service contain duplicate attributes in the SOAP messages returned from the MDC (Australia) which can cause failures in some SOAP parsers (eg the PERL one).

There is an issue with the availability of the Schema documents. In particular the *eXist* database required schema documents declared in the XML instance documents to be available via the web in order to perform automatic validation the instance documents. This is in general not a problem, but recently there have been times when the ILDG web site where the Schemata are was down and this affected *eXist* databases.

5.2.1 Response times of MDC Catalogs

This was a sufficiently large discussion to merit a separate section. It was found that server side clients can have poor response times when talking to the MDCs of regional grids. Several reasons have been identified for this

- When large data sets are returned there was a noticeable overhead in what is believed to be the construction and parsing of soap messages. For example UKQCD measured that a query to the UKQCD MDC took several minutes to return a soap message to a client but a direct connection to the database, without SOAP returned the query results in seconds. Bálint noted that in test usage of the MDC infrastructure in the JLab - for purposes other than MDC - had problems when retrieving very large result sets via the Web Service Interface. In fact the queries could fail to return or cause a machine to be heavily loaded. Going through the *eXist* command line client alleviated this to some degree but not completely.
- The LDG MDC showed slow response times to some queries. Dirk explained that this may be because the queries are formulated in such a way that they do not get mapped to the more SQL implementation. In this case the entire database needs to be materialized as Java objects. Timing queries to the LDG MDC was consistent with the time required to materialize 80000 documents as Java Objects. Further, it was verified in session, that both the UKQCD client and the USQCD server side clients sent queries that do not map to SQL queries in the LDG service.

Action Items: USQCD and UKQCD to modify their clients to issue the SQL friendly version of ensemble search queries.

George also noted that at one point in the past, UKQCD carried out stress tests for *eXist*. This showed that the database had trouble when inserting O(100,000) documents at once. The Java application crashed in this case and corrupted the database. For this reason, XML documents are also backed up as separate files on the QCDGrid (DiGS). Backing up *eXist* is straightforward.

6 VOMS Operation

DESY has now been operating a VOMS service which hosts the ILDG VO for several months. We have had the following experiences:

In UKQCD, authentication is based on the 'grid-mapfile' mechanism that is frequently used by Globus installations and others. Each UKQCD server has a copy of the grid-mapfile, which is refreshed automatically on a daily basis.

The grid-mapfile includes entries for users who are registered with the ILDG VO and listed as members of the UKQCD sub-group. The list of such users is retrieved from the VOMS server using the standard web interface and mapped to a generic pooled account within the grid-mapfile for each resource. In addition to actual users, UKQCD servers (storage and control) also need to be listed in the grid-mapfile; this allows background housekeeping to be performing automatically. Since servers are not recorded in the ILDG VO, certificate subjects for these are captured in a static, local mapfile, which is merged into the grid-mapfile alongside user information from VOMS.

6.1 Pooled Accounts

Mitsuhisa asked about pooled accounts at this point. Pooled accounts are a way to allow many grid users to use services without creating many accounts. Essentially there is a pool of user accounts with arbitrarily chosen user IDs and grid users are randomly assigned to accounts in this pool. Usually a user will be assigned to the same pooled account for some reasonably long period so that two successive requests can build on each other rather than executing in different pooled accounts. If there is over subscription and insufficient user accounts in a pool to meet demand, services decline requests.

Pooled accounts are not a necessary feature. The advantage is that one does not need to create an account explicitly for every grid user. However, the down side is that logging by user can get complicated. There are also administrative down sides. As an example, if there is a pool of accounts and users just get allocated to one - possibly without even knowing that they are mapped to an actual user account within a service, then questions may arise such as: "Has the user read and signed the appropriate computing regulations?" Other security restrictions may also exist, for example, U.S. DOE rules may prohibit the deployment of pooled accounts on DOE resources.

In Fermilab the VO Server for the ILDG VO is the DESY VOMS server as far as the FermiGrid is concerned. However, the public dCache at Fermilab is not yet automatically updated and currently managed by hand. Fermilab promised to automate this service in approximately 1 month.

6.2 Decentralized VO Administration

It has generally been agreed that the current decentralized approval of VO requests is reasonably straightforward. If a user requests to join the VO, an email is sent to all the VO admins and is processed by the regional admin from whose regional grid the request originated. A minor technicality arose as to how to who should approve VO admins whose certificates have expired or who have to add a new certificate. In the most recent case Bálint had to re-register. Dirk approved this on a specific request from Bálint. However, according to George no email of this event was sent. Dirk explained that there is not an automatic email notification to inform admins of the approval of requests. He does this manually currently and must have forgotten to do it. It was decided that some form of automatic notification would be good, even if it was just archived to a mailing list archive.

6.3 VOMS Fault Tolerance and VOMSRS

Dirk described the VOMSRS (VOMS Registration Service) service which could be used to make the VOMS service more fault tolerant. Currently ILDG exists only in a single VOMS server hosted at DESY. If this service were to fail, grid services that rely on it would be adversely affected.

There is a new service called VOMSRS (VOMS Registration Service) the purpose of which is to deal only with VO Registrations. A user registers their certificate in a VOMSRS service and this service then registers that certificate in

possibly several VOMS servers. The VOMSRS service uses the standard VOMS interface so the VOMS services do not need to change, but simply must accept the VOMSRS service's certificate as an admin. This mechanism allows a single VOMSRS service to maintain several mirrors of a VO in several VOMS servers. If the VOMSRS service fails, only new registrations are held up. If one of the VOMS servers fails, one can fall back to one of the other mirrors VOMS servers. In this way the VOMS service can gain redundancy through duplication.

Politically this setup may be quite attractive, as each regional grid may hold their own local copy of a VOMS server.

Client tools would have to be adapted to take advantage of the multiple VOMS services. One possibility to aid this is to add VOMS server addresses into the ILDG Services file. In this case the ILDG services file is a single point of failure. However, since that file tends to be fairly static, a locally cached copy can provide fault tolerance.

It is possible that a VOMSRS service can be kept up to date by another VOMSRS service, since the VOMSRS services have a standard interface.

Should we choose to adopt the VOMSRS service for fault tolerance of the VO, we would try to set it up so that users would not need to reregister.

VOMSRS also offers different levels of privilege from the VO. It has for example the concept of a group representative who is not a full admin, but can for example only approve requests.

In the end we'd have to agree where the ILDG VOMSRS service runs. Currently DESY is looking at VOMSRS and Fermilab already runs one. Also Australia is also using a VOMSRS service. There may be a political dimension as to where the ILDG VOMSRS runs in particular with U.S. DOE. However, the Atlas and CMS experiments must have solved this already, since they need to use extended VOMS certificates and need a highly available VOMS service all the time.

On our phone conversation with Paul C, he mentioned that the Australian National Grid also uses VOMSRS and so the Australian end of ILDG would have no problem with this technology for the ILDG VO.

6.4 VOMS and Certificates and Policies

We currently agree to accept IGTF certificates. This was placed in front of local admins at service providing sites and to our best knowledge there has not been opposition to this.

Also currently on the VOMS Server when a user registers, they agree to the EGEE VO Policy rules rather than our own one.

Action Item: Dirk to change the VOMS service screen so that when joining the ILDG VO, the users is asked to agree to the ILDG VO rules rather than the EGEE one.

7 File Transfer Tests

Most of Tuesday morning was spent attempting File Transfer Trials. Dirk posted a list of test files to

<http://www-zeuthen.desy.de/~pleiter/ildg/fttest.txt>

The test contained files from

- DESY Zeuthen – SRM URL

- ZIB Berlin – SRM URL and GridFTP URL
- ZAM Jülich - SRM URL and GridFTP URL
- UKQCD - GridFTP URL
- FNAL - SRM URL only
- JLDG - GridFTP URL

We also installed *srmcp* clients of various versions. We built up the following matrix of File Transfer tests

From/To	LDG	JLab	JLDG	UKQCD	FNAL
Zeuthen(SRM)	OK	OK	OK	OK	OK
Berlin (SRM)	OK	OK	OK	OK	OK
Berlin (GridFTP)	OK	OK	OK	OK	OK
Jülich (SRM)	OK	OK	OK	OK	OK
Jülich (GridFTP)	OK	OK	OK	OK	OK
UKQCD (GridFTP)	OK	OK	OK	OK	OK
USQCD (SRM)	NO	NO	NO	NO	OK
JLDG (GridFTP)	NO	NO	OK	NO	NO

To summarize, file transfer tests failed on transfers from Fermilab and Japan. The reason for the failure of the transfers from Fermilab was due to infrastructure issues at Fermilab. The Japanese GridFTP service has just been set up by Mitsuhsa during our discussions and its remote access was probably inhibited by firewall issues.

8 SRM issues

8.1 SRM Clients

During the file transfer trials we encountered some minor annoyances with *srmcp* clients. First of all almost all of us had different versions of the client. Secondly the client was quite sensitive to the Java Run Time environment that was available. In particular the version 1.25 client needs Java 1.5 while previous versions work with other version of Java. Several clients also attempted a file transfer with more than one stream which may not be supported by firewall configurations or SRM servers. In general the following options needed to be set on *srmcp* clients

- the `-streams_num` option needed to be set to 1
- the `-use_url_copy_script` needed to be set to `true` in order that the file transfer proceed with the a copy script rather than the Java GridFTP client object

Also we noted that while `globus_url_copy.sh` can handle HTTP downloads, currently *srmcp* does not. However *srmcp* is just a script and may be modified to handle the HTTP protocol. Alternatively we could write a wrapper that would call *wget* under the hood if necessary.

Further discussion with SRM developers revealed that v1.25 is now the most advanced version. It no longer relies on the configuration file (unless that is desired) and handles proxy and CA certificate detection better than before. It also fixes some number of bugs.

As a response to the client issues, after the meeting came to an end Dirk packaged up a version of the *srmcp* client v 1.25 and placed the RPM on the LDG Ltools site.

8.2 SRM Service

Our discussion with the SRM developers revealed the following information about SRM. There are 4 SRM implementations which are part of storage management software: dCache (DESY/FNAL), Castor (CERN), DPM (CERN) and DRM/HRM (LBL). The STORM project in Italy implements SRM v2 on top of GPFS and also over a standalone UNIX filesystem.

The dCache implementation needs only Java and the Postgres database to be installed. There are discussions with VDT, to automatically install dCache as part of their standard installation.

Fermilab distributes a standalone SRM implementation. This is a little bit limited in functionality but it essentially uses the same code base as the dCache SRM and can be deployed on top of a regular UNIX filesystem. It serves out files with GridFTP.

A good place to find out information or to get the Fermilab standalone client is to <http://srm.fnal.gov>. There are instruction here to check out the code from CVS and have a test drive.

UKQCD asked about SRM on top of Globus 4 and no other middleware. The recommendation was to try the standalone SRM service. It should allow UKQCD to get and put files.

The roadmap for SRM v2.2 is heavily driven by LHC experiment requirements. New requirements were introduced last may and these are being implemented at this time. The changes are related to storage classes in particular to Quality of Retention and Access Latency. The functions have been implemented but scaling tests are still pending. Alpha version prereleases are expected in a month or two. Some version 2 features that do not relate to space reservation or directories, are already present in dCache. Deployment of production systems with SRM v2.2 is expected for spring 2007.

All this having been said, in the current version of `srmcp` the ability for communicating with an SRM v2 interface has already been implemented.

SRM version 2 will probably remain useful until the end of the LHC experiments, although evolution of the version is expected. SRM version 3 is planned for the far future, maybe for the time frame of the International Linear Collider.

9 Access Control Discussion

Dirk showed a presentation relating to the implementation of access control within LDG using Access Control Lists (ACLs)

The requirements were principally that ACLs be defined with respect to ensembles and groups. Access states were the following:

- World readable, write access restricted
- Both read and write access are restricted

The following authorization levels were mentioned

- *Administrator* has full rights everywhere
- *Project Managers* have read write access to data owned by projects such as Ensembles, Groups and the list of project managers

- *Group Member* has rights defined by Access Control Lists with respect to his/her group.

The administration of ACLs was implemented within the Administration Interface of the MDC web service. The authentication is implemented using the GLite trust manager. Clients use a proxy to establish an SSL connection and through this use GSI authentication/authorization.

Currently the storage elements have no ACL implementation. This is expected to come in a work package of dCache called Chimera which is currently in development to replace PNFS. Chimera should/would use POSIX ACLs adapted for grid credentials.

There was in the end no real agreement as to whether such access control is universally needed within the ILDG and the discussion was suspended.

10 File Catalog Discussion

Recent trends in terminology have replaced the term Replica Catalog with File Catalog. The MDC is relatively successful, and SRM clients and servers are available. The last tier of the ILDG Middleware remains the File Catalog (FC; formerly Replica Catalog RC).

During the spring/summer, a File Catalog interface service was developed at DESY the purpose of which was

- Provide a shared software component which can be deployed at each site (Uniformity)
- Provide a way to query arbitrary back end catalogs and save developers/admins from the burden of writing the web service front end for their catalog.
- Provide for proxy delegation in case that is needed by the back end

The main service provided by the service interface is the `getURL()` function, which takes as input a Logical File Name and returns a list of URLs which can be used to download the relevant data file. This is done by passing the LFN onto a back end `geturl.sh` script which has to be provided by the local site to access their own particular back end database.

Additionally the following functions are provided for proxy management:

- `proxyInit()` - is used to delegate a local proxy to the web service
- `proxyInfo()` - is used to query the status of the delegated proxy
- `proxyDestroy()` - is used to destroy the delegated proxy

In particular, the `proxyInfo()` function maps onto a local `proxyinfo.sh` script.

The service of course can also be deployed so as not to need a proxy, as is done currently by USQCD. However, both LDG and UKQCD may need a valid proxy to access their replica catalog. UKQCD uses Globus RLS and LDG uses the LCG File Catalog.

The biggest bone of contention in this discussion was the procedure for delegating a proxy. Everyone was concerned about simply uploading a proxy into a web service. We brought this concern up in our discussion with JLab Cybersecurity Analyst David Bianco and with our phone call to Globus Security Specialists visiting EPCC.

We received the following feedback from David: Proxy delegation is potentially dangerous. A proxy credential is a full representation of a user. One particular problem is the “malicious administrator” attack. At the site where the delegated proxy is held, a malicious administrator or other attacker, could gain access to the proxy and use it to impersonate the end user. The risk imposed by this kind of attack can be reduced by creating only short lived proxies. This can be achieved in several ways:

- We could post guidelines and advice to users to use short lived proxies – this advice risks being ignored
- We could make our client tools generate a short lived proxy out of an existing user proxy and delegate the short lived proxy to the service. This can be circumvented by writing a client which delegates a long lived client - something relatively difficult for the average user.
- The service could reject delegated proxies that have too long a lifetime

David told us, that in particular, the generation of a short lived proxy from a longer lived one and then the delegation of the short term proxy to an intermediate service is similar to a strategy followed by Kerberos (but with public-key infrastructure rather than Kerberos tickets) and that we could probably learn a lot from how Kerberos does things.

On a phone call, the Globus security developers explained to us that the process of delegation is defined by a standard called WS-Trust. They worried about our use of the phrase “uploading a proxy to a web service” because a proxy contains private keys which should not be sent over the network. Instead the following scheme is in common use:

Alice on computer A wants to delegate her certificate to Bob’s machine B. The way this works is that Bob’s machine generates a certificate request and sends it to Alice’s machine. Alice’s machine signs the certificate request and sends it back to Bob’s machine. Bob’s machine has then got a signed certificate without Alice’s private key being transmitted over the network, and can generate a proxy certificate for Alice.

We were told, that delegation was a service already supported in Globus. In particular the Globus delegation service allowed us to delegate proxies to any other service within the same Globus container. A Globus container in this instance is a Java platform server (essentially Tomcat modified to allow it to host grid services which have persistence, state etc) in which the Globus Grid Services are hosted.

As another alternative if we didn’t want to use Globus, they suggested we try MyProxy (<http://grid.ncsa.uiuc.edu/mypr>) developed by NCSA. This is a service which can be used to hold grid credentials. However, to us this is not really solving our delegation problem - essentially we still need to load up our credential to a service (the MyProxy service) and our intermediate services still need to generate proxies so that issue is not solved. Even worse, we may now need to run MyProxy as an ILDG central service.

In terms of WS-Trust there is a concern that the relevant signing protocol now may need to be implemented in the clients.

One underlying issue is whether we can trust a remote service into which we delegate our grid credentials. This level of trust may be established based on examining the certificate the web service sends us.

Despite the bewildering array of options there was an agreement that there is still a genuine desire to form a File Catalog Interface Service and to share in its development and deployment.

Action Items:Radek and George to investigate mechanisms for trust delegation (in gLite and Globus) that do not require the transfer of a proxy certificate (or private key). Dirk to look into options for generating a short-lived proxy from an existing proxy cert.

11 Grid Monitoring

Our work with the File Transfer Trials showed us that the grid infrastructure is capable of degrading. Some kind of monitoring is probably required so if a component fails, we learn about it in short order. Radek demonstrated to us his use of the GITS scripts to monitor the services on DiGS (QCDGrid). Essentially this exercised various Globus features such.

A more high powered approach has been designed by the Teragrid called Inca. (inca.sdsc.edu). This abstracts and formalizes the concepts of tests and provides a framework for constructing monitoring infrastructure. We should investigate how easy this tool is to use. A nice feature of Inca would also be to persist the status information into a database so that we could get a time history of behavior should this be desirable.

An alternative scheme, is to run a local script by cron job and place the result on an endpoint (or RSS feed). This could then be aggregated by some central web site. This works in a 'pull' mode. Essentially the central web-site would poll its list of known endpoints for information.

Action Items: George and Radek to investigate Inca. Bálint to investigate more low brow method.

12 Development Environment and Web Site

Not having a Wiki is becoming noticeably felt. Several key documents are missing and need to be collected. These documents include for example

- The ILDG File Format Definition Document
- The ILDG VO Policy Document
- The ILDG VO Usage Instructions
- The ILDG MDC WSDL Document
- The ILDG MDC Specification Document
- The ILDG MDC test suite document
- The ILDG Operations Specification
- Presentations from this meeting
- Minutes from previous meetings.

Further several other files would be useful to have under revision control such as:

- Services File Schema and Services File
- QCDML ensemble and configuration Schemata
- ILDG FileFormat XML Document Schema and example

Further to this it would be useful to have a shared source code area.

It was decided that all these requirements can be provided by the NeSCForge, which already has an ILDG project and an associated CVS repository. It was decided to make better use of this resource. An initial layout for the CVS repository is proposed below (but may be slightly different depending on who implements it)

```
/doc
/doc/external

/src
/src/external
/src/schemas
/src/mdc
/src/mdc/wsdl
/src/mdc/client
/src/mdc/tests
/src/fc
/src/fc/wsdl
/src/fc/client
/src/fc/tests

/projman
/projman/meetings
/projman/meetings/jlab-12.2006
/projman/minutes
```

Files under `src` would essentially be source code, files under `doc` would be documentation including external documentation and files under `projman` would relate to project management.

13 Resources

Resources are fairly pitiful. European guys need to investigate US Can sponge off SciDAC. Japan is waiting for an opportunity to submit a proposal. EU have framework 7. Dirk and George to take actions to talk to people they know who know about Framework 7.

Various other application areas which use the grid were being used such as microbiology and astronomy. If we could take these areas to the level of a project then it would improve support within the EU. At the JLab it is hoped that OSG may be coming up and that we can leverage off this.

14 Miscellany

14.1 Grid Operations

We meandered into the issue of grid operations. Users have potentially got a legitimate expectation of knowing the status of the grid. How to inform our users if some part of the grid is being maintained? One suggestion was to post email to the list of users in the ILDG VO. However, those of us who receive regular 'status' emails from various places wondered whether this is a good idea. We fear being deluged with mail. One suggestion was just to mail a mailing list that no one reads but is archived. Interested parties could always check the archive.

We also discussed briefly reliability of MDC clients. Currently some server side clients can fail if not all the MDCs are up. This problem is particularly acute in the case of server side clients accessed through web browsers.

Some of these problems may be alleviated by running automated test suites to identify problems which then may need to be fixed by hand.

One question that arises is whether we are overstepping our remit as a middleware interface standardization body. While it is possible that we are in fact overstepping our bounds, really all that could be done would be for the board to create another working group, which would be staffed by us yet again, so we might as well consider these issues.

14.2 RPMs and Techniques

Dirk showed us how he rolls his RPMs and he has also showed us how RPMs can be installed as a regular user rather than root. These can be found at the URL:

<http://www-zeuthen.desy.de/latfor/ldg/doc>

14.3 Java Development with Maven

Bálint discussed the use of Maven for Java builds and unit testing. Maven is a build system for Java that works at a higher level than ant. Its chief strengths are that due to its standardized directory layout and conventions it is fairly easy to build Applications and Web Services with it. Bálint walked everyone through the setup of the MDC Client application build.

Maven relies on a repository of Jar files so that it can fulfill dependencies needed by the sources. A large repository is available globally on the *Ibiblio* web site. However, some Jar files (say by Sun Microsystems) are not available there due to licensing restrictions. In a shared development environment it would be pleasant to be able to set up a shared Maven repository or at least to set up a system so that everyone can synchronize a local repository and that everyone can build everyone else's code.

Maven also integrates nicely with the Eclipse Java Development Environment in use in both DESY and EPCC/

Interested parties should download and consult a freely available book on Maven 2 (http://www.mergere.com/m2book_dow)

15 Milestones for Lattice 2007

We assembled the following major Milestones to be reached by the lattice 2007 conference:

- *by end of December, 2006* - Write this report and circulate by end of Dec 2006.
- *by 15, January 2007* - Setup CVS repositories and collate documentation. Check into the possibility of a VOMSRS service (NB: One already running at Fermilab)
- *by 31, January 2007* - Decide on delegation mechanism, explore concepts for monitoring
- *by 01, March 2007* - We would like to have regional grids synchronized with the VOMS service. Also we should ensure that the file transfer matrix is filled only with OKs (ie correct what doesn't work right now). It would be desirable to automate this and post the results through reporting infrastructure.
- *by 01, June, 2007* - Complete File Catalog Interface, public data should be accessible on all grids.
- *by 01, July, 2007* - Have user friendly client tools ready.
- *by 30, July, 2007* - All systems go for Lattice'07 in Regensburg

Although not present at the meeting, we managed to reach Paul Coddington on the telephone. He agreed that these are good milestones and that the ILDG effort in Australia is happy to work towards them.

Action Items: Paul C, and Derek L, should register in the ILDG VO. Shunde Zhang should also probably register.

16 Conclusions

This meeting discussed several pertinent issues, some going beyond mere middleware and into the area of general grid operations. We considered the one outstanding web service – the replica catalog – and have had a long discussion on how to deal with certificate delegation. We have discussed making our VO service fault tolerant through VOMSRC. We have carried out our most extensive file transfer trials to date. We have considered efficiency issues in our current metadata catalog. We have discussed the future of the SRM service and clients as well as monitoring our grids. We have reaffirmed our commitment to cooperation on software development and have nominated the NeSC Forge site to host our software and backups of our documents. We consider this meeting to have been pleasant, productive and successful.

17 Thanks and Acknowledgements

Bálint would like to thank Roy Whitney for his pleasant welcome of the delegates and for keeping the meeting fueled with coffee and pastries and cookies. Bálint would also like to thank staff services especially Cynthia Lockwood and Noel Vermeire for organizing the room and the delivery of the cookies and snacks and for dealing with any local paperwork needed for this meeting. Thanks must also go to Melissa at the Residence Facility for her help in arranging accommodation, and also to David Bianco for joining us for dinner to discuss security issues.