

LCG Technical Workshop

KEK, Japan, November 17 – 18, 2005.

Marco La Rosa

mlarosa@physics.unimelb.edu.au

<http://epp.ph.unimelb.edu.au>

Outline

Thursday 17th November.

- Introduction to the LCG Grid middleware
 - components of the middleware
 - Grid services
- Installation and Configuration of two testbed Grids
- Using the Grid.

Friday 18th November

- Detailed introduction to Grid concepts and the LCG Grid middleware
- Review of the previous day
- Installation, configuration and federation of computing resources at participating sites
- Belle MC demonstration on the Grid (if we have time)

Acknowledgements

Takashi Sasaki

Yoshiyuki Watase

Go Iwai

Hiroshi Sakamoto

Glenn Moloney

Lyle Winton

Who am I?

- School of Physics, The University of Melbourne, Melbourne Australia.
- LCG deployment as part of the Australian National Grid Program supported by the Australian Partnership for Advanced Computing (APAC)

Australian deployment so far:

- **Victorian Partnership for Advanced Computing (VPAC)**
Compute element gateway to 2 clusters
Brecca: 194 CPU RedHat Linux cluster (Intel Xeon 2.8 Ghz)
Edda: 144 CPU Suse Linux cluster (IBM Power5)
- **The University of Melbourne, Advanced Research Computing**
Compute Element, Storage Element (DPM) and R-GMA_MON
Charm: 26 CPU Scientific Linux cluster (Intel Xeon 2.8 Ghz)
- **School of Physics, The University of Melbourne**
Resource Broker, R-GMA registry / schema server, VOMS / VOMRS,
Compute Element, Storage Element, User Interface
Xen Virtual Machine Monitor (VMM) – Ubuntu and Scientific Linux

Introduction to the LCG Grid Middleware

User Interface

tools to use the grid: submit jobs, manage data

Resource Broker

accepts job control requests and takes action to satisfy them

Information System

information on available resources, state etc

Compute Element

Grid gateway to CPU resources

Storage Element

Grid gateway to storage (disk, tape) resources

Worker Node

where the work gets done

User Interface

A user's point of access to the Grid

Allows the user to interact with the Grid at a 'high level'

resources are available for use - the system manages who and how

Globus tools

globus-job-run, globus-job-submit, globus-job-status, globus-job-get-output

EDG / LCG tools

edg-job-submit, edg-job-status, edg-job-get-logging-info, edg-job-get-output

Information System queries

lcg-infosites, lcg-info, ldapsearch

Data Management

lcg-cr, lcg-aa, lcg-cp, lcg-del, lcg-la, lcg-lr etc.
edg-gridftp-ls, edg-gridftp-exists, edg-gridftp-mkdir, edg-gridftp-rename etc.
glite-catalog-ls, glite-catalog-mkdir, glite-catalog-mv, glite-catalog-rm etc.
dpns-ls, dpns-mkdir, dpns-rm, globus-url-copy, srmcp, srm-get-metatdata etc.

Allows the user to interact with the Information, Workload and Data Management Systems

Resource Broker / Workload Management System

Central component in the LCG Grid middleware – provides:

Network Server: accepts job control from the client (UI) and passes them on to the workload manager if appropriate

Workload Manager: core component – satisfies valid request for resources

Resource Broker: a.k.a match maker – matches resources with requests

Job Adaptor: prepares the job for submission

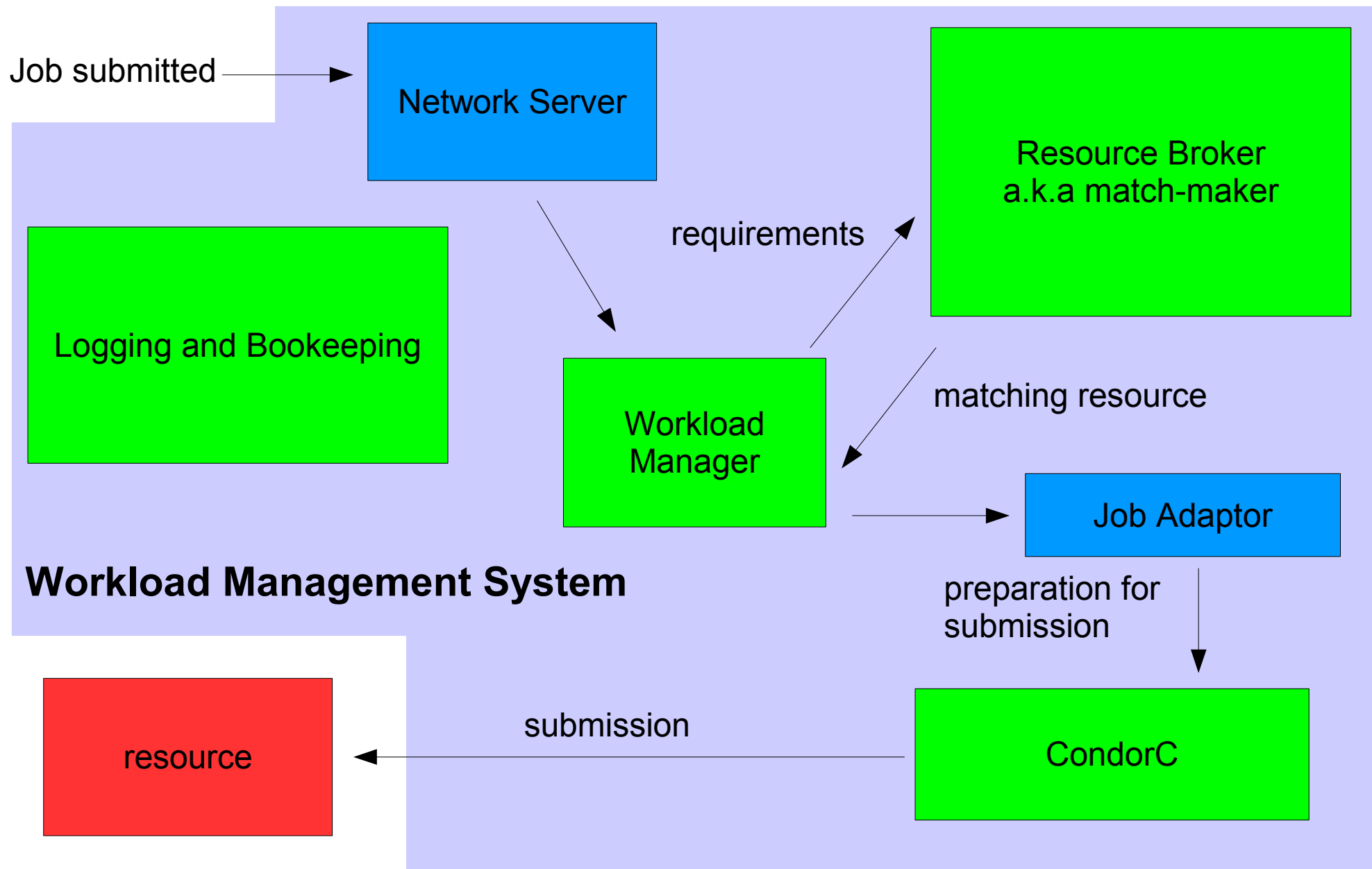
Proxy renewal service: ensures a valid proxy exists for the life of the job

CondorC: performs the job management on request of the workload manager

Logging and bookkeeping: job monitoring functionality

Logmonitor: watches the CondorC logfile and triggers actions based on interesting events

Resource Broker / Workload Management System



Information System

Information about the Grid

- provides information about Grid resources and their status
- published information also used for monitoring and accounting purposes
 - namely performance and usage analysis
- Available resources
 - CPU: location, state (free cpu's, currently running jobs)
 - Storage: type, total capacity, available capacity
 - Metrics: Is the resource available? Is it accepting jobs?

Virtual Organisation's supported by the resource – VERY IMPORTANT!

- Based on the GLUE Schema
 - Grid Laboratory for a Uniform Environment – v 1.2

Information System – Globus Monitoring and Discovery Service (MDS)

Globus Monitoring and Discovery Service (MDS)

- currently the main provider of information

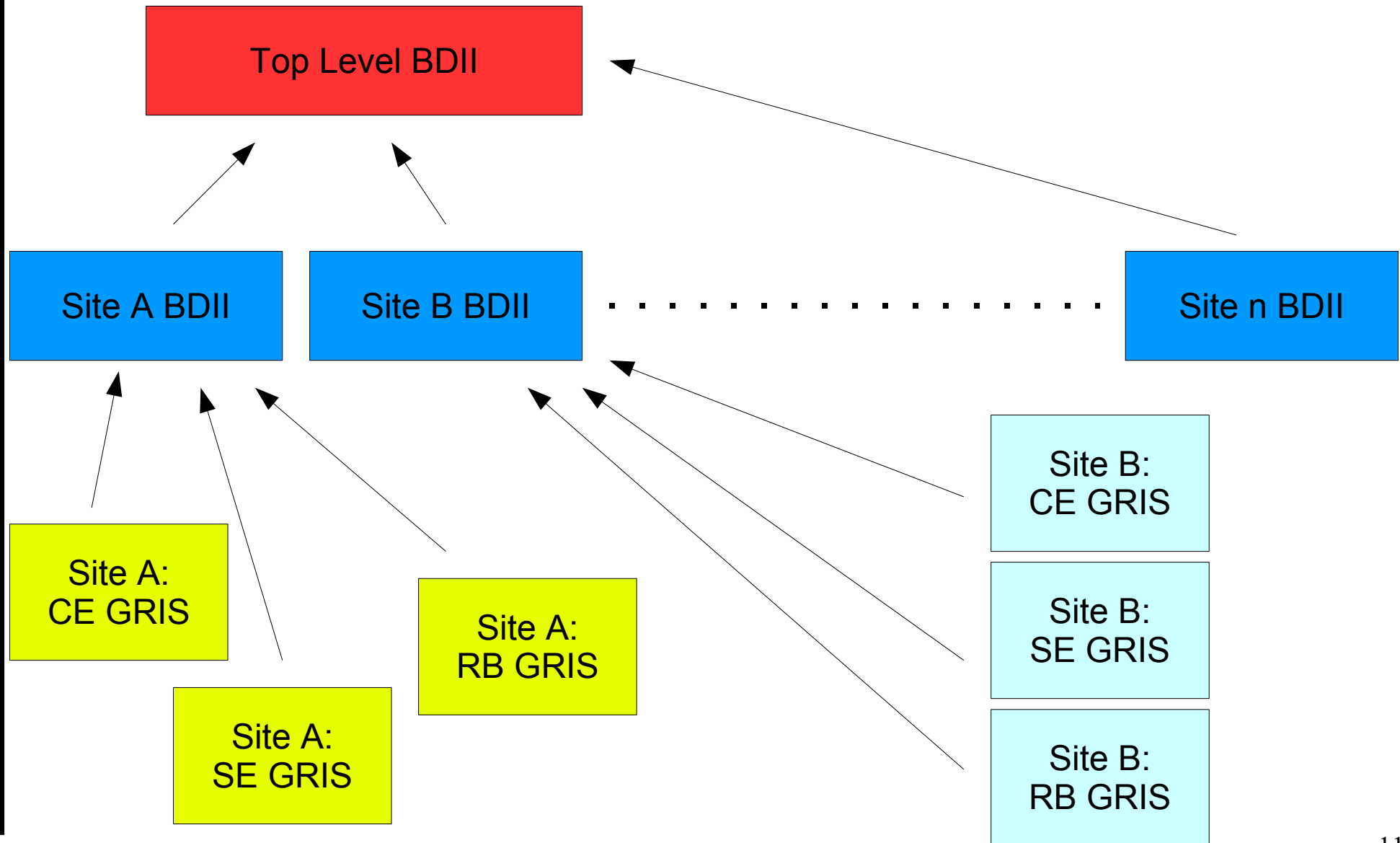
openLDAP – the information conforms to the GLUE Schema

- Only anonymous access is allowed to the catalog – all users can browse the catalogs and all services can enter information into it
- Information providers generate the relevant information from static configuration files and dynamically gathered information
- this information is published via Grid Resource Information Servers
- Site Berkeley Database Information Indexes (BDII) compiles the information from the different GRISes at a site and publishes it

used instead of Globus GIIS (Grid Information Index Server) for stability reasons

- Top level BDII queries the site-BDIIs acting as a cache of information about the Grid
- Up-to-date information can be gathered via querying of the site-BDIIs (or even the GRISes) directly

Information System – Globus Monitoring and Discovery Service (MDS)



Information System – Relational Grid Monitoring Architecture (RGMA)

Relational Grid Monitoring Architecture

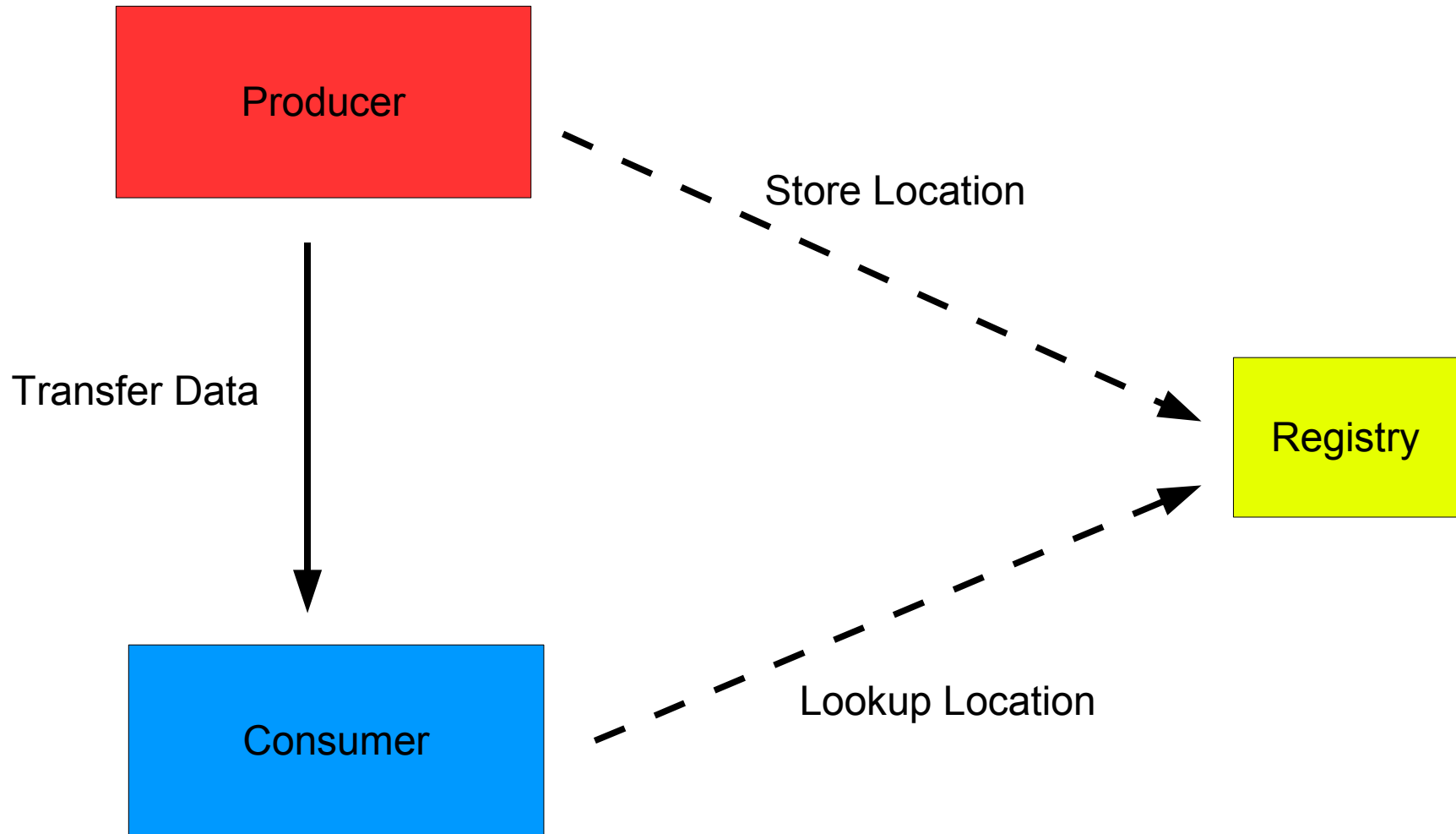
- information is presented as being in a global, distributed relational database

- Supports more powerful query operations than LDAP
- Based on Producers and Consumers
- Producers register themselves with the registry and describe the type and structure of information they want to make available to the Grid
- Consumers can query the registry to find out what information is available and where to locate the producers of that information
- Registry mediates the communication of information from producers to consumers

MDS is the main source of information in LCG2.

In time, R-GMA is expected to completely replace it

Information System – Relational Grid Monitoring Architecture (RGMA)



Compute Element

Grid gateway to computing (CPU) resources

- Gatekeeper service
- Security (GSI) enabled
- Job manager's – perl modules
 /opt/globus/lib/perl/Globus/GRAM/JobManager
- GRIS (Grid Resource Information Server)
 Information about the computing resource
- Site BDII
- Local Resource Management System (LRMS)
 - PBS – Portable Batch System (Torque LRMS)
 - LSF – Load Sharing Facility
 - Condor-G
 - Grid Canada is a Condor based Grid system with an LCG 'front-end'
 - Maui – Fine grained scheduling and policy definition

Compute Element

- A computing element is defined as a Grid batch queue and is identified via a contact string of the form:

<hostname>:<port>/<batch-queue-name>

So:

lcgce.kek.jp:2119/jobmanager-lcgpbs-short

lcgce.kek.jp:2119/jobmanager-lcgpbs-medium

lcgce.kek.jp:2119/jobmanager-lcgpbs-long

**are all different computing elements
even though they are on the same physical host**

- Computing Elements are built on farms of nodes called Worker Nodes, a LRMS and a Grid Gate (Gatekeeper)
- Gatekeeper's must be accessible from outside the site as it will need to receive work from Resource Brokers on the Grid.

Each LCG-2 site will run at least one Compute Element and a farm of Worker Nodes

Storage Element

Grid gateway to storage (disk, tape) resources

Classic Storage Element (default up to LCG-2.4, LCG-2.6)

simple disk server,

Services: GSIFTP, insecure RFIO, no SRM interface

Mass Storage System (MSS) (LCG-2.4, LCG-2.6??)

front end disk, back end tape,

Services: MSS, GSIFTP, insecure RFIO, usually have an SRM interface

dCache Storage Element (LCG-2.6)

pools of disk with a 'disk pool manager'

Services: Disk pool, GSIFTP, gsidcap, SRM interface

Disk Pool Manager Storage Element (LCG-2.6)

recently introduced, aims to replace the classic SE

will offer much of the functionality of dCache whilst avoiding its complexity.

Aims to be easier to install, configure and maintain than dCache

Services: Disk pool, GSIFTP, secure RFIO, SRM interface

Storage Element

Disk Pool Manager Storage Element (v 2.6)

From the Admin Guide:

<https://uimon.cern.ch/twiki/bin/view/LCG/DpmAdminGuide>

**Lightweight solution for disk storage management
implementation of SRM (Storage Resource Manager) v1.1 v2 spec.**

- Handles the storage on disk servers
- Pools (groups) of filesystems - pools can be on one or more disk servers
- Filesystems can be:
 - volatile – files may be removed by the system at any time unless pinned by the user
 - permanent – files cannot be removed by the system.
- Security (GSI) enabled

DPM is designed to be easier to install, configure and manage than dCache whilst providing similar services – specifically, an SRM compliant interface to storage

Worker Node

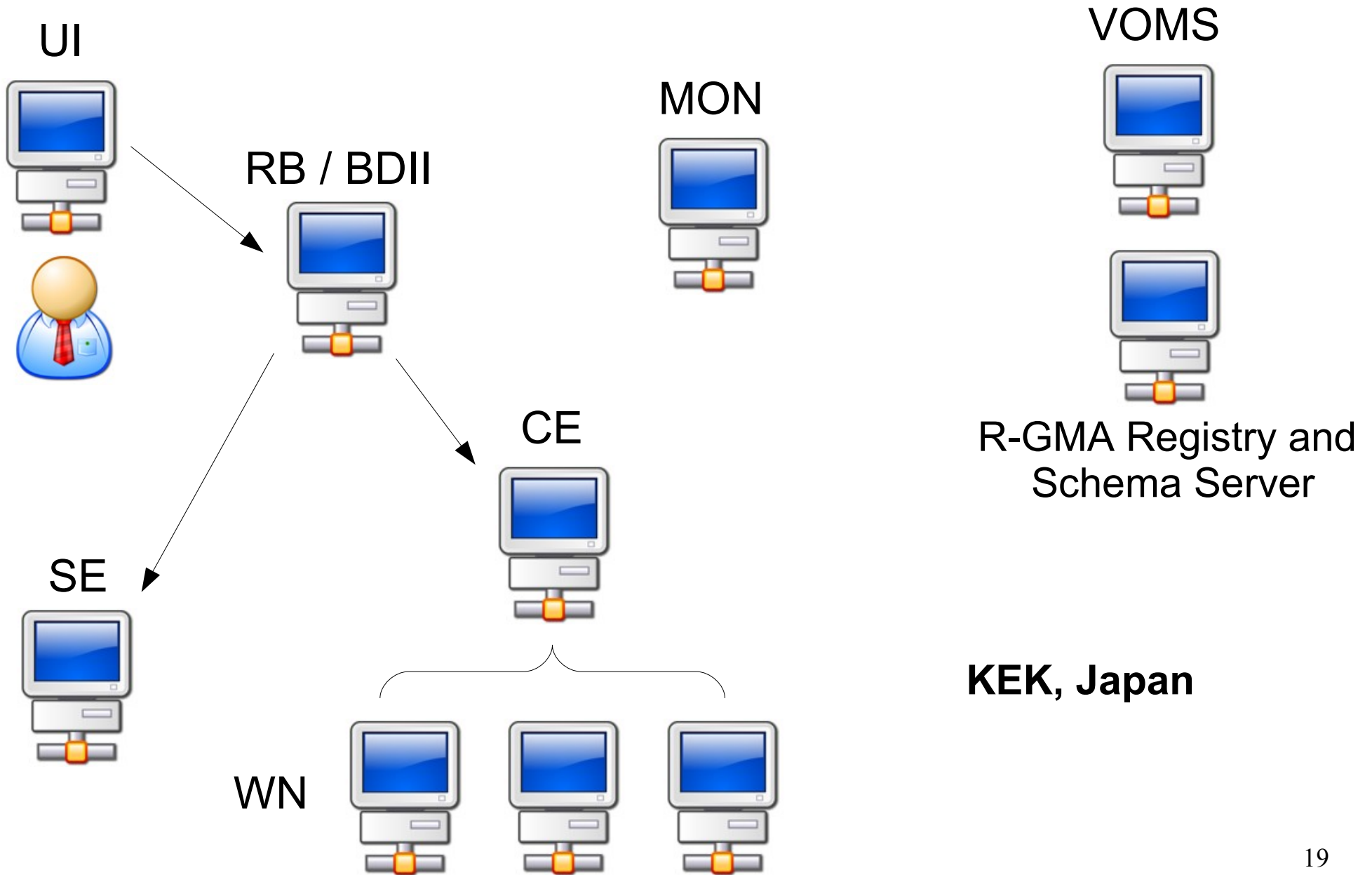
- The computing power of the Grid
- Sandboxes
 - Input sandbox retrieved from the Resource Broker using globus-url-copy
 - Output sandbox copied back to the Resource Broker at job completion
- Data files retrieved from / stored directly on a Storage Element
- Grid tools available to running jobs
 - the same tools as on the User Interface

Generally, cluster compute nodes are on a private network and are not allowed to access the external network.

LCG Grid cannot function in this configuration.

Compute node's need to be NAT'ted through the headnode

What did we do yesterday?



Certificate Authorities

What is a Certificate Authority?

- It is an authority which issues and manages security credentials and public keys for message encryption and decryption
- Public Key Infrastructure (PKI) – asymmetric encryption
- Authentication of the identity of an individual / host / service
- Manages the issuance of new certificates
- Revokes certificates of individuals / hosts / services that have lost their authorisation
- Maintains a list of these entities

Grid Security Infrastructure (GSI)

X.509 certificates offer four primary pieces of information:

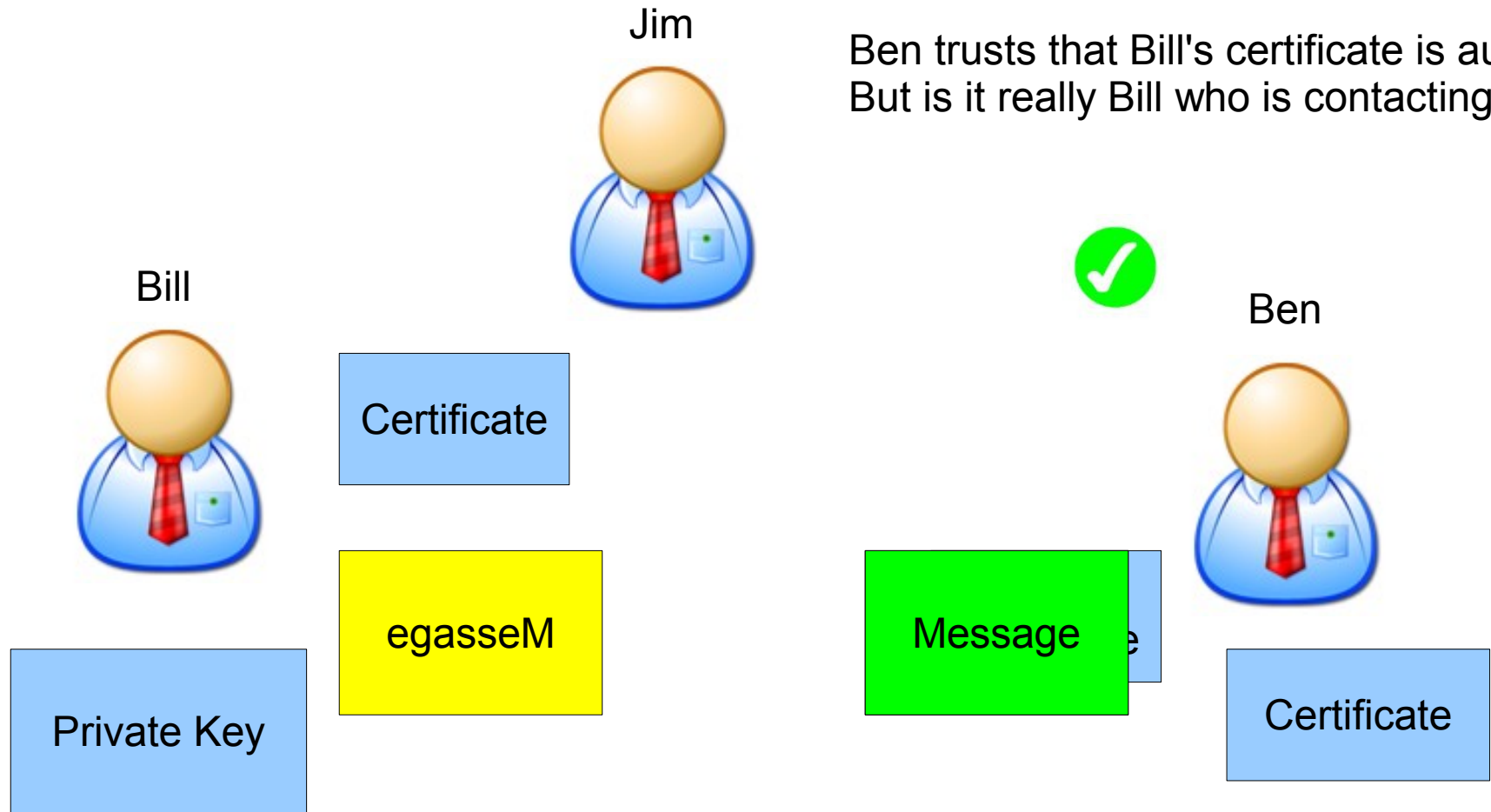
- a subject name which identified the person or service
- public key belonging to the subject
- identity of a certificate authority that has signed the certificate
- digital signature of the CA

The CA certifies the link between the public key and the identity

Mutual authentication

If two parties have certificates, and both parties trust the CA's that signed those certificates, then the two parties can prove to each other that they are who they say they are.

Grid Security Infrastructure (GSI) – Mutual Authentication



Ben trusts that Bill's certificate is authentic.
But is it really Bill who is contacting Ben?

Ben trusts that it is really Bill he is talking to.

Now the same must happen in reverse – Bill must check Ben's authenticity

Grid Security Infrastructure (GSI)

Mutual Authentication

- Person A establishes a connection to a Service B and gives B his Certificate.
- B checks the CA's digital signature on the certificate to make sure that the certificate is valid and that it hasn't been tampered with. - similar to digitally signed emails – PKI.
- If B trusts the CA that signed A's certificate, and the certificate is authentic and valid, then B must determine that A really is the person he says he is.
- So, B generates a random message and sends it to A – asking him to encrypt the message with his private key.
- A encrypts the message and sends it back to B.
- B decrypts the message with A's public key. If the message is the original message sent to A, then B can trust that A is who he says he is.
- The reverse process is then initiated.
- If successful, then A and B have established a connection and can trust each other.

Grid Security Infrastructure (GSI)

Confidential Communication

- GSI does not establish encryption communication between parties by default.
- GSI does provide communication integrity by default – third party may be able to read the communication but is not able to modify it in any way.

Delegation and Single Sign-On

- an extension of the standard SSL protocol
- a proxy is created to act on behalf of the user
- new certificate, public and private keys
- contains the owner's identity but is modified to indicate that it's a proxy
- the proxy has a lifetime – after which it should not be accepted by others
 - default lifetime = 12 hours
- the proxy is signed by the user's certificate not the CA

Grid Security Infrastructure (GSI)

- Mutual Authentication with proxies is slightly different
- The remote party now receives the proxy and the users certificate
- The owners public key (from their certificate) is used to validate the signature on the proxy
- The CA's certificate is then used to validate the signature on the certificate

This process establishes a chain of trust from the CA to the proxy through the owner

Proxies and Lifetimes

- Proxies have a lifetime
- What happens if the proxy expires before the job is complete?

The job fails.

Two options:

- long lifetime proxies
the longer the lifetime of the proxy, the longer someone can impersonate you if they get hold of it – after all, it is stored on the filesystem
- some sort of proxy renewal service

MyProxy proxy server

- user generates a long term proxy and stores it in the proxy server
- the WMS periodically uses this proxy to renew the proxy for a submitted job before the proxy expires and until the job is complete.

Virtual Organisations

- A compulsory requirement – anyone who wishes to use the Grid must be a member of a supported VO
- VO membership allows the user to access their VO's resources
e.g. an ATLAS VO member can read ATLAS data files
- Must be a member of the collaboration to become a member of the VO which defines the collaboration
- Users must comply with the rules of the VO – membership can be revoked as easily as it is granted
- VO may have access to facilities they do not own or manage, those facilities have their own security and policy concerns, they may serve other VO's and other user communities

“A GRID IS A COMPLEX NETWORK OF ORGANISATIONS
AND CONNECTING POLICIES”

Winton, 2005

Virtual Organisations

Some requirements from the community:

- Users may be members of multiple VO's
- A resource can support / participate in more than one VO
- VO's must be able to specify membership policy
- A users VO membership must remain confidential
- It should be possible to list resources and actions which a VO member or role has access to
- It should be possible to list resources to which a VO member or role has access to carry out specific actions
- Authorisation decisions **MUST** be consistent within a VO
- It must be possible to disable a users VO authorisation
- The VO must be able to specify security requirements on any resource for specific roles

VO, users / members, groups and roles, resources, priorities, authorities

Virtual Organisations

CA's effect the conditions for resource sharing:

- VO owned services / tools must trust all CA's associated with member's
- Participating resources must trust all VO member's CA's
- VO services / tools and participating resources must trust all resource CA's

Why should resources trust the VO's list of CAs?

- If they want to participate, they must
- Generally, the VO knows it's member institutions and who can best certify their members.
- If a CA starts issuing bad certificates the VO can “untrust” them. (until CRLs are fixed?)

**VO's are a complicated beast. Resource sharing, trust, usage policies.
In development – working to a degree – but still work to be done**

Virtual Organisation Membership Service (VOMS)

- A service to manage authorisation information in a VO scope
- The service is in use – although not yet fully functional
- The VOMS system is used to include a users VO membership and any other related information in a users proxy certificate
- These proxies are said to have “VOMS extensions”
- VOMS allows finer grained control of rights, roles and capabilities within a VO

Current situation:

- User DN mapped to a pool account at each facility via a grid-mapfile

Future situation:

- User will initiate sessions with VOMS proxies detailing their requested right, role, group
- User gets dynamically mapped to an appropriate pool account which will allow that user to carry out their task

Virtual Organisation Membership Registration Server (VOMRS)

An extension to the VOMS system

Design goal:

To provide an easy to use interface for users to register with the system and request sign-in credentials

- Integrated with VOMS – and adds authorisation features in addition to registration
- Users must have Grid credentials – VOMRS / VOMS do not supply them

What does VOMRS allow users to do?

- register,
- associate a grid credential with an authorised account
- which institution the user belongs to
- contact information

Requests are forwarded by email to an admin and validated by an institutional representative

Users are notified by email when their accounts are ready to be used

Virtual Organisation Membership Registration Server (VOMRS)

VOMRS works with VOMS

When an account is created by VOMRS, the data is uploaded into the VOMS database

The VOMS database is used to issue credentials to the user when they go to use the Grid

Authorisation information in addition to the authentication information in the user's Grid credentials

Virtual Organisation Membership Registration Server (VOMRS)

File Edit View Go Bookmarks Tools Help

https://voms.ph.unimelb.edu.au:8443/vo/belle/vomrs

The Mozilla Organiza... Latest Builds Google Australia EPP Home

belle VO Registration

- belle Registration Home
 - . Registration (Phase I)
 - . Groups and Group Roles
 - . Institutions & Sites
 - . Required Personal Info
 - . Certificate Authorities

Welcome to the belle VO Registration Service!

This site is used for registering belle VO grid resource users and for managing information about their affiliation with the belle VO and their permissions regarding use of the grid resources. Registration consists of two phases. In phase I, a visitor completes the registration form labelled "Registration (Phase I)". Next, the visitor becomes a "candidate" for VO membership. After an initial confirmation of identity, the candidate moves to phase II. In phase II, the candidate completes the registration form labelled "Registration (Phase II)" and confirms intent to comply with [the KEK Usage Rule document](#). At this point the candidate becomes an "applicant". Once the applicant is approved, he or she becomes a "member". The [VO administrator](#) may grant you administrator rights, as appropriate.

Below we list the VOMRS functions available to users at each stage:

Visitors to the belle VO may:

- ◆ Browse groups
- ◆ Browse institutions and sites
- ◆ Browse required personal information
- ◆ Browse CAs recognized by belle VO
- ◆ Complete Registration Phase I

In addition to the visitor functions, **candidates** to the belle VO may:

- ◆ Browse their personal information
- ◆ Browse their certificate information

In addition to the candidate functions, **applicants** to the belle VO may:

- ◆ Select group and group role assignments
- ◆ Re-sign usage rules
- ◆ Browse their own authorization
- ◆ Unsubscribe and resubscribe to personal event notification

In addition to the applicant functions, **members** to the belle VO may:

- ◆ Enter/delete additional certificates
- ◆ Select a different primary certificate
- ◆ Change their own personal information

Additional functions reserved for approved belle VO **Administrators** :

- ◆ Add, change, delete information for members, groups, institutions,sites, CAs

[Help about the site can be found here](#)

You are logged in as /C=TW/O=AS/OU=PHYS/CN=Marco La Rosa/Email=m.larosa@physics.unimelb.edu.au
/C=TW/O=AS/CN=Academia Sinica Grid Computing Certification Authority

Data Management

On the Grid, data can be replicated to where it is needed

Users or applications should not know where the data is located – data management services map logical filenames to physical files

A file can always be recognised by its Grid Unique Identifier (GUID), furthermore, all replicas of the file will share the same GUID – anywhere on the Grid

guid:<unique_string>

- Users or applications refer to files using logical file names

lfn:<any_alias>

- Storage URL (SURL) provides information on the physical location of files

sfn:<SE_hostname>/<local_string> or srm:/<SE_hostname>/<local_string>

- Transport URL (TURL) provides the necessary information to retrieve a replica, including:

hostname, path, protocol and port

The mappings of these objects need to be stored. Currently there is the Replica Location Service (RLS) used by all of the VO's so far and the LCG File Catalog (created to overcome performance and functionality issues in RLS)

LCG File Catalog (LFC)

- High performance file catalog based on lessons learnt during the data challenges
- Fixes performance and scalability problems seen with the EDG catalogs
- Significant improvement over EDG catalogs, including:
 - GUID identifiers and LFN identifiers like EDG catalogs, but, mappings between them are stored in the same database
 - operations which span both sets of mappings are faster
- API similar to UNIX filesystem API – create, mkdir, chown
- Supports bulk operations with transactions
- Timeouts and retries supported
- Authentication via Kerberos 5 or GSI
- Planned integration of VOMS authentication

Only a secure version is available

Job Flow

- User logs onto a UI and creates a proxy certificate to be used in authenticating himself
- The job is submitted to the WMS – files defined in the job description file are copied to the resource broker and these become part of the Input Sandbox.
The event is logged – **state: SUBMITTED**
- WMS looks for the best available CE to submit the job to – queries BDII, File Catalogs.
The event is logged – **state: WAITING**
- WMS prepares the job for submission, creates the wrapper.
The event is logged – **state: READY**
- CE receives the request and sends the job to the LRMS for execution.
The event is logged – **state: SCHEDULED**
- LRMS handles the job. User files copied (globus-url-copy) from the RB to the WN.
The event is logged – **state: RUNNING**

while the job is running, data can be accessed from a closeSE using RFIO or gsiftp, or from remote SE after copying them locally to the WN filesystem with the data management tools

Job Flow

generated data can be uploaded to the Grid using the data management tools – it can be stored in a close SE and registered in a File Catalog

- if the job completes successfully, then the output (small data files specified in the job description file as being part of the Output Sandbox) are copied back to the RB.
The event is logged – **state: DONE**
- The user can now retrieve the output of the job.
The event is logged – **state: CLEARED**

Input / Output sandboxes should be used to transfer small data files necessary to start the job or check its results.

Large data files should be read and written directly to SE's and registered in a File Catalog

Belle MC: Australian Grid

- Initialise job: look for Belle MC scripts in SRB
- Process the script locally to determine required input files
- Submit the job to the resource
- Stage-in (from SRB) the required data onto the Grid gateway of the resource
- Submit the job to the LRMS
- Stage-out (to SRB / gsiftp) the data from the Grid gateway after the job is complete
- Transfer the output of the job to the local submission host
- Process the output to determine the status of the job – success / failure
- Organise the data within SRB



General Grid Issues

- Grid tools on the cluster worker nodes (compute nodes)
 - most clusters have their compute nodes on a private network with no external communication
- Middleware / Globus bugs and instabilities
 - I won't even bother trying to list them here
- Distributed management
 - version differences, constantly changing API's, how do we keep it configured?
- Firewalls / Network ACL's
 - Grid services / clients generally require public IP's – open ports
 - many applications require access to DB's or remote data whilst processing
- Staging work arounds
- User access barriers
 - user has a certificate – congratulations – what now?
 - access to facilities can be complicated
 - education – start now and it might be understood within the next year

Recommendations

- Start talking to your System / Network admin's

education, education, education

- Have dedicated admin's for the OS, middleware

DON'T UNDERESTIMATE THE TIME REQUIRED

- Have a regression test script
- Run the script from the local site
- Run the script from a remote site
- Automate general management tasks – people are fallible
 - authentications lists (VO's)
 - CA files – especially CRL's
 - Host cert checks and imminent expiry warning
 - Service status checks
 - File clean up

The end....

Thank you

Marco La Rosa
mlarosa@physics.unimelb.edu.au
<http://epp.ph.unimelb.edu.au>

Distribution Concerns

- Distributed as a set of binary RPM packages for Scientific Linux 3

(gLite states compatibility with any distribution which is binary compatible with RedHat Enterprise 3 – in principle the LCG release should also be)

- Easily installable as a binary RPM component – installed via the APT package management tool
 - Binary RPM package limited to RHEL3 compatible distro's
- UI and WN components also distributed as a tarball
- more portable – users can install it on their own desktops
 - on any distribution – potentially
- RHEL3 is based on a 2.4 kernel
 - better I/O with a 2.6 kernel
 - applications want Scientific Linux 3

Should the middleware and applications be distribution independent?

- Some sites will not change their distribution
 - vendor support, local expertise, manpower

Some details...

Workshop divided into 4 sections

Preparing the node

- The nodes have a base installation of Scientific Linux 3.0.5 and now need to be prepared for the Grid middleware installation

Download the LCG installer and prepare for installation

- The LCG YAIM installation scripts need to be installed. Customised scripts for this installation are provided

Install and configure the middleware components

- There are 6 components to be installed – in two Grid configurations. You will work in groups and install multiple middleware components per group

Testing

- Everyone can log in to the User Interface in their group guest account and run the tests. If any of the tests fail, then the first group to find out should notify everyone else, then su to root and fix the problem.

Section 1 - Node Preparation

- Is APT installed?
- APT sources file
- Update the System
- Install Java
- Configure network time synchronisation

Section 2 –

Download the LCG installer and prepare for installation

- Download the installer
- Configure the site information files

keksite-info-Grid[1 or 2].def

- key = value pairs
- standard bash syntax – you should be able to source the file
- shared amongst all of the different node types

kekusers.conf

- list of 'pool' (generic) user accounts to be created on the nodes
- we will only be configuring this test bed for the apdg VO

kekwn-list-Grid[1 or 2].conf

- if using LCG to configure worker nodes, then write their names in this file – for configuration of the LRMS

Section 2 –

Download the LCG installer and prepare for installation

We need to make some modifications - keksite-info-Grid[1 or 2].def

```
LFC_HOST=lfc-host.$MY_DOMAIN
```

```
GRID_TRUSTED_BROKERS="" "  
    openssl x509 -in hostcert.pem -noout -subject
```

```
SE_TYPE=srm_v1
```

```
DPMDATA=$CE_CLOSE_SE1_ACCESS_POINT
```

```
DPMGR=dpmuser
```

```
DPMUSER_PWD=verySecret
```

```
DPM_HOST=$SE_HOST
```

```
DPMPOOL=se-storage
```

```
BDII_HTTP_URL="http://epp.ph.unimelb.edu.au/epp/lcg2-central.conf"
```

```
BDII_REGIONS="CE SE RB"
```

Section 3 – Install and configure the middleware component

- Installing the components
- Generic installation procedure
- User Interface
- Storage Element (DPM MySQL)
 - although the notes also detail how to install and configure a classic Storage Element
- Compute Element (with Torque Server)
- Mon-box
- Resource Broker / BDII
- Worker Node (with Torque Client)

Section 4 – Test the Grid

- Proxies
- Run a simple job
- Is the User Interface correctly configured to access the Resource Broker?
- Data Management