CONTRACT NUMBER 508830

DEISA
DISTRIBUTED EUROPEAN INFRASTRUCTURE FOR SUPERCOMPUTING APPLICATIONS

European Community Sixth Framework Programme
RESEARCH INFRASTRUCTURES
Integrated Infrastructure Initiative

SA3 activity report and roadmap evaluation

Deliverable ID: DEISA-DSA3-8
Due date : May, 1, 2007
Actual delivery date: May, 16, 2007
Lead contractor for this deliverable: Organisation, Country

Project start date : May 1st, 2004
Duration: 4 years

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of Content

Table of Content....1
1. Introduction....3
1.1 Executive Summary....3
1.2 Document structure....3
1.3 References and Applicable Documents....3
1.4 List of Acronyms and Abbreviations....4
2. General SA3 strategy overview and new targets....6
2.1 Activity summary....6
3. SA3 activity status....7
3.1 LL-MC infrastructure and system administration activity....7
3.1.1 Introduction....7
3.1.2 LL-MC, production customization update tuning and evolution....7
3.1.3 DECI jobs management....8
3.1.4 LL-MC cluster user mapper....8
3.1.5 LL-MC cluster metric enhancement....8
3.1.6 LL-MC multi -step jobs management....8
3.1.7 LL-MC DEISA llsubmit filters enhancement....8
3.1.8 LL-MC DEISA test-suite....8
3.2 DEISA License Server solution....9
3.2.1 Introduction....9
3.2.2 Short description of all possible solution....9
3.3 DEISA UNICORE infrastructure....10
3.3.1 Current DEISA UNICORE infrastructure....10
3.3.2 Objectives reached....11
3.4 DEISA RMIS infrastructure and GT4 MDS evaluation....12
3.4.1 Introduction....12
3.4.2 Evaluation of GTK4 MDS....12
Access to information....12
Security....13
WSRF data services....13
MDS4....13
3.4.3 New Architecture for the RMIS....13
3.4.4 Improved RMIS schema....13
RMIS version....14
Site....14
1.0....14
ECMWF, EPCC, SARA....14
2.0....14
BSC, CSC, CINECA, FZJ, IDRIS, RZG....14
Not deployed....14
HLRS, LRZ....14
3.5 DEISA co-allocation....14
3.5.1 Introduction to co-allocation and co-reservation topics for DEISA....14
3.5.2 Platform solution for co-allocation and co-reservation....14
3.5.3 Current status....18
3.6 DEISA data federation....19
3.6.1 Overview: the need of data federation in Deisa....19
3.6.2 Prototype version 1.0 and 1.1....19
3.6.3 Prototype version 2.0....19
3.6.4 Prototype version 3.0....19
3.7 DESHL deployment....20
3.7.1 DESHL v3 final deployment across all DEISA sites....20

3.7.2 DESHL v4.0 deployment across DEISA JRA7 sites (EPCC, FZJ, CINECA, ECMWF) 20
3.8 DCRMP 20
3.9 Interoperability 21
3.9.1 Introduction 21
3.9.2 Activity results 21
4. Roadmap evaluation and further plans 22
4.1 SA3 overall roadmap 22
4.2 SA3 further plans 22
4.2.1 LL-MC 22
4.2.2 Unicore 22
4.2.3 RMIS 23
4.2.4 Co-allocation 23
4.2.5 Data federation 23
4.2.6 DESHL 23
4.2.7 DCRMP 23
4.2.8 Interoperability 24

1. Introduction

1.1 Executive Summary

As stated in the DEISA Technical Annex (DEISA TA) [1], this deliverable is an *"Activity report and roadmap evaluation".* The last *"Activity report and roadmap evaluation"* was done at PM24[3], so this document will contain the activity report of just the last twelve months and an update of the roadmap.
The document deals with progresses obtained in the deployment of DEISA resource management and middleware software stack on new functionalities deployed from month 24 to 36.
Finally it mentions and refers documents that cover in deep some aspects of the SA3 activities. Such documents are:

- the DEISA SA3 Quality Plan for DEISA, the document organizes the activity inside the SA3 amongst SA3 partners;
- The SA3 Licence server possible solutions document that addresses topics concerning license server management inside DEISA;
- The Universus acceptance tests document for the co-allocation and co-reservation tool.

Major results of the last 12 months of the SA3 activity can be summarized as follow:

1. Upgrade and new DEISA LL-MC filters release developed and deployed.
2. New DEISA License server solutions and deployment
3. Upgrade of the DEISA Unicore infrastructure improvement and evolution
4. Upgrade of the DEISA Information system development and deploying
5. Extended DEISA data federation software development and deployment.
6. Latest DESHL evaluation and deployment
7. Delivery of the SA3 Quality Plan

1.2 Document structure

This section outlines the document structure and intended audience for the document.

- Section 2. Summarizes the activity that SA3 has to tackle and new targets have to be addressed for the last next 12 months of the project

- Section 3. Sums up the results obtained during the period from month 24 to month 36 of the DEISA project.

- Section 4. Marks out the actions for the last 12 months of the project keeping in account the SA3 roadmap.

1.3 References and Applicable Documents

[1] DEISA Annex I – "Description of Work", November 5th 2003.

[2] DEISA public web site, http://www.deisa.org

[3] D-SA3-6

[4] D-SA3-5

[5] D-SA3-4

[6] http://work.deisa.org/bscw/bscw.cgi/d66814/Midterm-report-eSA4-Dec-2006.doc

[7] License solution for DEISA
http://work.deisa.org/bscw/bscw.cgi/d28864/License-DEISA.doc

[8] Co-allocation in a heterogeneous environment
http://work.deisa.org/bscw/bscw.cgi/d4898/Coallocation%20in%20an%20heterogeneous%20environment.doc

[9] Heterogeneous batch scheduling
http://work.deisa.org/bscw/bscw.cgi/d4906/SA3%20-%20Proposal%20n.3%20-%20heterogeneous%20batch%20scheduling%20-%20rev.1.3.doc

[10] Advanced reservation
http://work.deisa.org/bscw/bscw.cgi/d4902/SA3%20-%20Report%20n.2%20-%20advance%20reservation%20-%20rev.0.5.doc

[11] Requirements Specification Technical description of Universus for DEISA
http://work.deisa.org/bscw/bscw.cgi/d65794/Requirements_Specification_Technical_description_of_Universus_for_DEISA.doc

[12] Universus acceptance test
http://work.deisa.org/bscw/bscw.cgi/d65814/UNIVERSUS%20Acceptance%20test.doc

[13] "Using Platform LSF MultiCluster", Version 6.2, Platform Computing, INC, September 2005

[14] http://www.deisa.org/applications/projects2005-2006/liams.php

[15] Application Hosting Environment AHE,
http://www.realitygrid.org/AHE/index.shtml

[16] RealityGrid, http://www.realitygrid.org/

[17] D. Frank and T. Soddemann Rechenzentrum Garching der MPG (RZG). Interoperable job submission and management with GridSAM, JMEA, and UNICORE

[18] https://forge.gridforum.org/projects/gin

[19] http://edg-wp2.web.cern.ch/edg-wp2/security/voms/

1.4 List of Acronyms and Abbreviations

AOB	Any Other Business
AHE	Application Hosting Environment
BSD	Berkeley Software Distribution
CINECA	Consorzio Interuniversitario
CVS	Concurrent Versions Systems
DEC	DEISA Executive Committee
DEISA	Distributed European Infrastructure for Supercomputing Applications
DESHL	DEISA Services for Heterogeneous management Layer

ECMWF	European Centre for Medium-range Weather Forecasts
EPCC	Edinburgh Parallel Computing Centre
FZJ	Forschungszentrum Julich
GGF	Global Grid Forum
HPC	High Performance Computing
HSM	Hierarchical Storage Management
HSM	Heterogeneous Service Management (now superseded by DESHL)
INCA	Grid Monitoring Framework
JRA7	Joint Research Activity 7 – Access to Resources in Heterogeneous Environments
LSF	Load Sharing Facility
OGSA	Open Grid Services Architecture
PM	Project Month
QP	Quality Plan
SA2	Service Activity 2 – Data management with global file systems
SA3	Service Activity 3 – Resource Management
SA4	Service Activity 4 – Applications and User Support
SA5	Service Activity 5 - Security
UNICORE	Uniform Interface to Computing Resources
VOMS	Virtual Organization Membership Service
XML	eXtensible Markup Language

2. General SA3 strategy overview and new targets

SA3 deals with Resource Management and Middleware. SA3 organisation and strategy can be found in all SA3 deliverables, in the DEISA Technical Annex and in the periodical “DEISA next 12 months plan”.

2.1 Activity summary

The following Gantt chart - Figure 1 - shows the SA3 activity for the 3rd year of the project.
We see that there are two kinds of activities:

- activities that conclude by the 3rd year
- activities that continue for all DEISA project

Activities belonging to the first group focus their actions of particular software for the DEISA infrastructure, activities belonging to the second group are actions that continue for all DEISA project and guarantee the correct operation of the infrastructure.

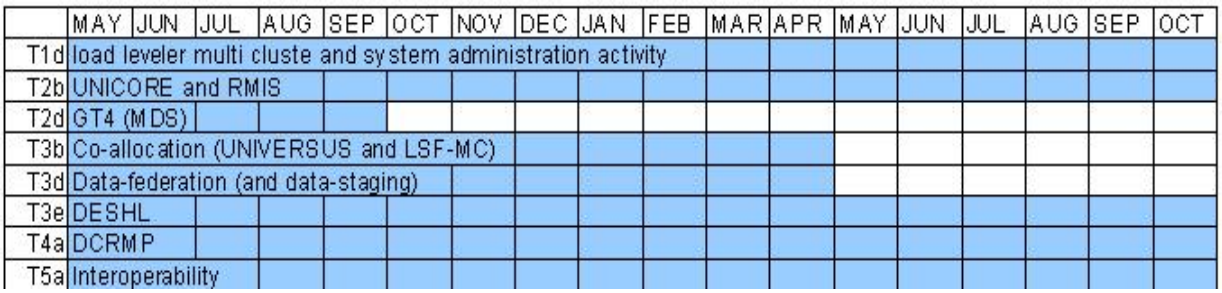

Figure 1 - SA3 Gantt chart

T1d operates, maintain and upgrade the DEISA LL-MC infrastructure field and has addressed and deployed the license server related-issues.

T2b operates, maintain and upgrade the DEISA UNICORE infrastructure and DEISA RMIS infrastructure.

T2d has evaluated GT4 MDS functionalities for the DEISA RMIS 2.0 implementation.

T3b concerns the evaluation of Universus for co-allocation implementation developed by Platform

T3d has provided a stage-in and a stage-out script aimed to transfer files to and from the Deisa GPFS using standard file-transfer mechanisms such as cp, scp, ftp and sftp, has implemented a new remote staging feature in order to pro-vide a mechanism to perform a stage-in/out operation from a remote Deisa site; has decided to interface these scripts with a site batch scheduler to allow a job to perform data-staging before and after the execution.

T3e activities work in strict cooperation with JRA7 for test and deploy DESHL over the DEISA infrastructure.

T4a had produced new packages for applications and the latest RMIS release.

T5a supports and encourages the cooperation of the DEISA Grid with other Grids around the world.

The Table 1 sums up the status of the DEISA software on the DEISA infrastructure.

Service	IDRIS	FZJ	RZG	CINECA	EPCC	CSC	SARA	ECMWF	LRZ	BSC	HLRS
UNI-G	FP	FP	FP	FP	FP	FP	FP	FP	FP	FP	FP
UNI-N	FP	FP	FP	FP	FP	FP	FP	TR	FP	TR	FP
UNI-T	FP	FP	FP	FP	FP	FP	FP	TR	FP	PP	FP
DESHL	FP	FP	NA	FP	FP	FP	FP	TI	TI	FP	TI
LL-MC	FP	FP	FP	FP	NA	FP	NR	NR	NR	NR	NR
LL	NR	NR	NR	NR	FP	FP	NR	FP	NR	FP	NR
LSF	NR	NR	NR	NR	NR	NA	FP	NR	TI	FP	NR
RMIS	FP	FP	FP	FP	FP	NA	FP	FP	TI	FP	NA

Table 1 - SA3 software stack status

Services status codes:

- **FP** full production, users are accessing the service
- **PP** pre-production: selected users are testing, with limited quality of service
- **TR** test reliable, but only DEISA staff accesses the service
- **TI** test initial, not yet reliable, only DEISA staff accesses the service
- **NA** not available yet, the service is not yet installed
- **NR** not relevant, the site is not concerned by the service

Service codes:

- **UNI-G** Unicore Gateway
- **UNI-N** Unicore NJS
- **UNI-T** Unicore TSI
- **DESHL** Unicore DESH
- **LL-M** AIX LL 3.3.x with multi-cluster functionality for job migration
- **LL** AIX LL 3.2.x standard
- **LS** LSF batch scheduler
- **RM** Resource Management Information System

3. SA3 activity status

This chapter describes the activities performed during the 3rd year of the DEISA project and shows the results and outcomes achieved.

3.1 LL-MC infrastructure and system administration activity

3.1.1 Introduction

This section describes the DEISA LL-MC status and outlines the tasks over the period PM25 to PM36.

3.1.2 LL-MC, production customization update tuning and evolution

Since PM25, 7 new releases of IBM LL-MC have been installed on all sites that are parts of the AIX super-cluster, with taking care of possible inter-site release dependencies.
DEISA LL-MC production customization is also maintained in order to fix problems encountered and to improve the multi-cluster management. For this purpose 4 filters releases have been tested and distributed.
The AIX super-cluster has been extended to CSC (another AIX DEISA sites), which has finalized its customization. During this integration all other sites needed to update their configuration to in order to communicate with the new incoming site

3.1.3 DECI jobs management

DECI (Deisa Extreme Computing Initiative) jobs that are extremely resource demanding can be automatically remotely submitted. These very particular demands are observed carefully, because they are out of the resource ranges usually managed for regular site jobs. For this reason, their execution needs production environment adaptation. Several DECI projects using this functionality with hundreds of jobs already executed [6].

3.1.4 LL-MC cluster user mapper

Cluster user mapper has been deployed as an additional security layer. It prevents non DEISA users to interfere with the DEISA infrastructure. It prevents also remote *uid* from being directly interpreted locally without verifying their link with the correct defined DEISA user name.

3.1.5 LL-MC cluster metric enhancement

As additional routing cluster selection factor, we setup a DEISA workload updated by each site within a small scale (0 to 4) and giving a level of its workload, because without sites interpretation, the AIX nodes workload or the length of the queues is meaningless for a remote site.
This *exit* is also managing a new maintenance flag during its routing site selection.

3.1.6 LL-MC multi -step jobs management

The management of multi-step jobs allows a better usage of the infrastructure. With this kind of jobs we can offer to users the possibility to bring the files transfer out of the application step, reducing the time of unused reserved computing resources.
The level of complexity of such object is increased and all the filters (*llsubmit_filter, remote_job_filter and deisa_llsubmit_filter*) with other basic level interfaces need modifications for allowing multi-step jobs management.
Available since the v2.0 release of the filters, this feature is deployed everywhere.

3.1.7 LL-MC DEISA llsubmit filters enhancement

Remote resources necessary for DEISA job execution can be temporary missing for scheduled reasons (sites maintenance) or because of failures (network, power, operating system, middleware...).
We setup a maintenance flag, also used during the filters execution to prevent remote DEISA job execution and thus automatically saving new DEISA jobs from resource missing problems. Available since the v2.01 release of the filters, this feature is deployed everywhere.

DEISA filters customization has been simplified by managing outside from the filters the main necessary site's setup. This new feature had also been introduced within V2.01 release.

Multijob-steps management had been improved with the V2.03 release by reducing the deisa keywords checking in step zero.

This last distributed release allows checking remote sites release filters, in order to ensure full compatibilities between installed DEISA filters.

3.1.8 LL-MC DEISA test-suite

A test suite had been developed in order to check the compliance of new LL-MC releases and DEISA filters releases with the DEISA job management defined policy.
A subset of this test suite has been adapted to check on a regular basis the behaviour and the status of the AIX Super Cluster job management. The results of the executions are automatically collected and consolidated with INCA.

The filters. Table shows the status of the LL filters on some DEISA sites.

SITE/ SYSTEM	CSC	RZG	FZJ	CINECA	IDRIS
PROD SYSTEM	LL-MC 3.3.2.7 + deisa filters V2.02/ V2.03 + local filters	LL-MC 3.4.0.3 + deisa filters V2.02/ V2.03 + local filters	LL-MC 3.3.2.6 + deisa filters V2.2/ V2.03 + local filters	LL-MC 3.3.2.7 + deisa filters V2.2/ V2.03 + local filters	LL-MC 3.3.2.8 + deisa filters V2.2/ V2.03 + local filters

Table 2 - Deployment status of LL filters.

3.2 DEISA License Server solution

3.2.1 Introduction

The aim of this activity is to inspect possible solutions for deploying software for managing Independent Software Vendor (ISV) third party codes requiring a license to be run within the DEISA infrastructure.
The start of several DECI projects, and the consequent use of almost all sites' resources offered by the DEISA project, is making a reality the job migration belonging to the DEISA standard Users (formerly DEISA implicit users). One of the main problem related to the job migration is that some applications (mainly those developed by ISVs), in order to be run, depend on the availability of "licenses" or "tokens". Since different sites have different security policy, it is not guaranteed that such licenses can pass through the network firewalls. The DEISA dedicated network seems to be a possible answer to all the security problems. More details concerning license server solution can be found on the DEISA repository at [7].

3.2.2 Short description of all possible solution

The first solution is to use existing license server available at each site over the internet.
The second solution is to install a dedicated DEISA license server at each site using DEISA licenses.
The third solution is to use a centralized license repository
The fourth solution is to connect the existing license server to the DEISA network

The first solution is the simplest one, because it doesn't request any additional hardware or extra management efforts, but it requires only routers reconfiguration to perform NAT on compute nodes. Obviously, as most of the presented solutions, such solution is strictly dependant on the site's specific security policy, nevertheless CINECA and IDRIS have successfully performed a proof of concept of this scenario. A minor negative aspect of this solution is that it doesn't address any security inconvenient, in any case it can be pointed out that license managers usually don't represent a threatening menace.
The second solution requires a bigger effort on the economic and managing side, but can solve the security problems, because it would avoid a network reconfiguration in the sites and it would route the whole traffic on the DEISA trusted network.
The third one is surely the weakest proposal, because it introduces an additional point of failure because all migrated jobs using licenses would then depend not only from GPFS-MC file system but also from this centralized machine, without adding any further improvement in security.
The fourth one, even if it requires some extra-configuration and is site' policy dependent, seems to be the best solution because it does not request big

development efforts. The need for additional hardware and its configuration (differently to the first solution) can be considered a minor drawback (especially if it is compared to the scenarios where fully fledged new license server is needed and all the needed policy decisions need to be addressed).
In all these cases it should be considered also that license management it is strongly dependent from vendor daemon implementation. This means that the solutions proposed are valid if it is true the hypothesis that all the license managers work in the same way that is how it has been working until now.

	Pros	**Cons**
Solution 1	• easy implementation • no need for additional hardware or licenses	• require network configuration (NAT + firewall) • traffic over Internet
Solution 2	• no security problems, all lies in the DEISA network • no particular network configuration required	• additional hardware and eventually also licenses needed
Solution 3		• introduces a single point of failure
Solution 4	• no complete hardware required, only Ethernet card	• requires modification of the /etc files

Table 3 – License server solution scenarios

Further observation: the idea to have “project licenses” is not good because it raises all the problems connected to the license ownership, so the unique applicable solutions are 1 and 4.

Project partner has selected and implemented solution 4.

3.3 DEISA UNICORE infrastructure

3.3.1 Current DEISA UNICORE infrastructure

All DEISA Sites have installed and configured the UNICORE server components Gateway, NJS and TSI to access the local supercomputer resources of each site in the heterogeneous DEISA cluster.

The DEISA UNICORE infrastructure is up and operates in production mode. Current situation is depicted in Figure 2.

The actual installed UNICORE software at all sites are:

- Gateway 4.1.3
- NJS 4.6.2 build 1
- TSI 4.1.0 build 2 (with DEISA specific extensions)
- DEISA UUDB 1.0

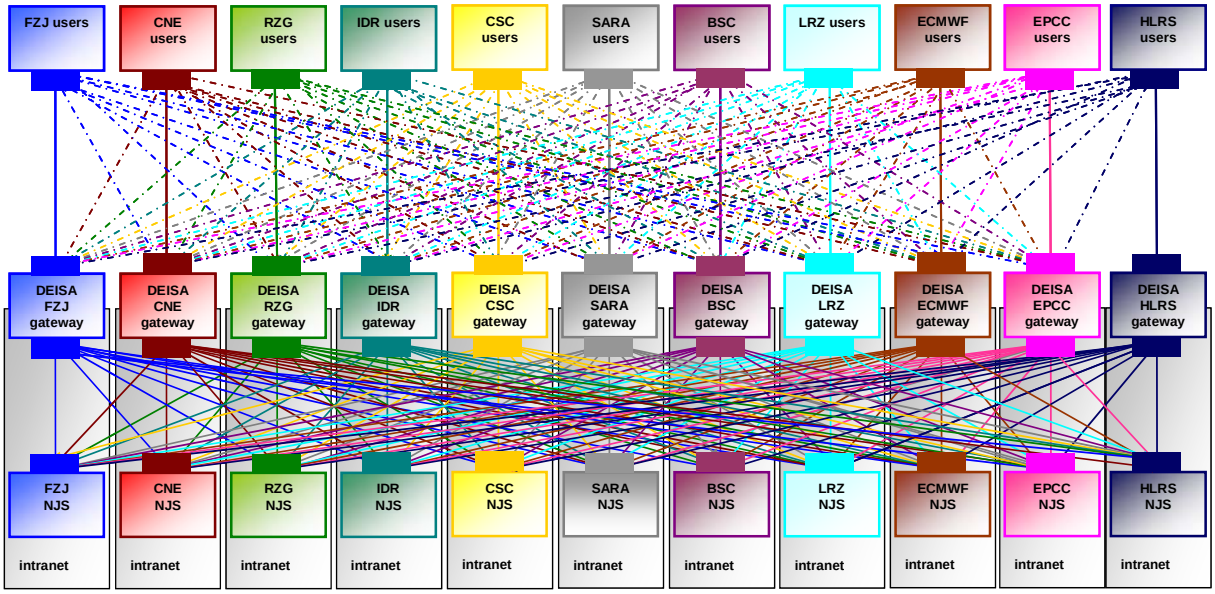

Figure 2 - DEISA UNICORE infrastructure

3.3.2 Objectives reached

Objectives reached can be summarized as follow:

- Certificates Control Revocation List Gateway extension,
- VOMS investigation for DEISA Unicore infrastructure,
- DEISA UNICORE infrastructure test suite development,
- SIMON development and test,
- UUDB development and test.

In June 2006 a new Gateway release has been developed to fix some bugs with the certification revocation list (CRL) checking of incoming user and server certificates. Now all incoming user and server certificates are checked if they have not been revoked for any reason.
Most of the DEISA certificates contain a CRL-URL. This URL is selected and interpreted by the Gateway. The Gateway makes a connection to the CRL and checks if the certificate is still valid. If so the job is passed through to the NJS. If not the job is rejected and the user gets a corresponding warning in the UNICORE client. Some EU-Grid-PMA Certification Authorities (CA) does not provide the CRL-URL within the certificates. For this reason an additional fallback location has been implemented in the Gateway. This location contains all the URLs of all DEISA CRLs. If a UNICORE job is signed with a certificate without this CRL-URL the Gateway checks every CRL in the fallback location until the corresponding CRL-URL is found.

On behalf of the SA5 DEISA security service activity the UNICORE group has done a first evaluation of the ongoing work in connecting VOMS (Virtual Organization Management System) [19] with UNICORE. VOMS allows defining finer user roles and attributes and could be interesting for the future DEISA user management. The UNICORE - VOMS connection will be addressed in the OMII Europe project. On the joined SA3 / SA5 meeting in Amsterdam at the beginning of December 2006 it was decided that all partners define their local requirements to an eventually UNICORE – VOMS connection and these will be forward to the OMII Europe developers as design input.

A requirement for the UNICORE working group was to develop a UNICORE test suite. The test suite contains a couple of one- and multi-site jobs to test the basic UNICORE functions at all sites and to guarantee that every site has the same UNICORE status. The test suite contains simple job scripts as well as complex workflows. The following issues are covered by the test suite:

- Is the DEISA Production Environment (DPE) and especially are the users' directories $DEISA_HOME and $DEISA_DATA available and accessible?
- Do the file import / export to and from the different DEISA file spaces work?
- Does the creation of remote files work and the downloading of the results to the local workstation?
- Does a multi-site job behave as expected?

Another tool, called SIMON (Site Monitor for UNICORE Grids), has been developed to check the health status of all UNICORE components automatically. SIMON is based on implementations coming together with DESHL from the JRA7 group. It monitors the state of a UNICORE based Grid deployment by executing the developed test suite periodically and automatically. SIMON records both expected and unexpected outcomes and this information can be used to build detailed reports regarding the health of the Grid. The test results might be extracted and prepared in various final presentation formats or for instance it can be emailed to the UNICORE administrators.
The UNICORE test suite is used in SIMON and submitted to the target systems to check the status of the server components.
A first prototype has been finished in PM28. Until end of December the developers worked on an enhancement of SIMON concerning an easy configuration and installation method. A graphical installer has been developed which allows the UNICORE site administrator to set all required values in a comfortable and easy way.

Further on all partners concentrated in investigating in a working and running UNICORE infrastructure serving actual and new projects coming from DECI.

A new release of the DEISA UNICORE user database (UUDB) has been published. This now allows mapping one user's certificates distinguished name (DN) to more than one user's login on the target system and/or to one or several projects. This got necessary because DEISA users e.g. from DECI sometimes work in more than one project and so now they don't have to apply for several certificates any more.

3.4 DEISA RMIS infrastructure and GT4 MDS evaluation

3.4.1 Introduction

The activity of the information system group inside SA3 for the 3rd year of the project was to first investigate on MDS4, in order to design a new architecture of DEISA RMIS. Also, an open source XML native database, called eXist database, has been investigated and used to store information data.

3.4.2 Evaluation of GTK4 MDS

The GTK4 MDS (or MDS4) is the monitoring and discovery service for the Globus tool kit version 4. It is based on the Web Service Resource Framework (WSRF) implementation of GTK4 and implements the WS-ServiceGroup specification. So it can be said that, it is a framework rather than a complete package This makes its use more versatile although use-cases have to be implemented in some way. While it is quite trivial to query an MDS4 service, information has to be fed into it.

Access to information

In principle the framework can deal with three different kinds of data sources:

- it can query web services with information
- it can subscribe to WS-Notification compliant web services
- it can periodically execute scripts which gather information

Although the last possibility seems to be the simplest, it is certainly the least favourable in terms of maintainability and performance. So, in the long term,

administrators making use of this framework should consider to use one of the first two ways for getting data into the MDS4 system.

Security

MDS4 supports the configurable security given by the Globus WSRF framework. Here the complete installation can be secured using PKI means. Furthermore, different methods of each application can be separately configured and provide an increased security over the default configured one for the installation. The Globus WSRF framework support point to point security with encryption and authentication via TLS, as well as message level security employing the WS-Security specification. Here, security can be either on the data integrity level by digital sign of messages or on data privacy by encrypting each message.
Resource level security, which has to be programmatDically implemented, is not directly supported. In order to use it, the WS-ServiceGroup implementation needs to be slightly extended, by having the Resource classes implement an additional interface (SecureResource). This does not break the compatibility with clients, since a client does not need to know about secure resources.

WSRF data services

The GTK4 WSRF core is implemented in Java and C. As far as RMIS is concerned, only the Java implementation is considered.
For RMIS the GTK4 WSRF core is to implement services with WSRF resources which represent the content of an eXist data storage and the information contained therein. This setup has been described in the last deliverable. With representation we mean both the logical representation where information is divided into logical objects modelled as WSRF resources and their content which is based on the RMIS schema (see below).

MDS4

The MDS4 is being employed for delivering data to the outside. The aggregator framework allows the connection to the WSRF services which directly connect to the eXist data storage and allows clients to retrieve the overall picture of the data available for each of the DEISA sites. The MDS4 acts as a registry for information while the original information is represented by the underlying WSRF services described above and the information residing in the eXist data storage.

3.4.3 New Architecture for the RMIS

The new DEISA RMIS infrastructure has already been described quite extensively. For completeness, since the transition from RMIS v1 to RMIS v2 has almost been finished, we would like to briefly describe the main points here.
The cache for all information consists of a eXist XML database which with its new engine is seems to be fast enough to answer to queries within an acceptable latency. Information on the system hardware status are gathered from ganglia, converted into XML and written into the XML database. Later, the MDS4 does not require to XML encode data again, since they can natively be fetched from the database and included into the document to be sent to the client. Information on the LoadLeveler and other batch systems are currently gathered employing a script which is periodically executed and again converts the output of the batch scheduling system into the usage record format and store them in the XML database. Also this information finds its way natively via the MDS4 to the requesting client application.

3.4.4 Improved RMIS schema

Originally the RMIS schema was exactly developed with the DEISA infrastructure in mind and hence was able to exactly map it. With more and more Grids interoperating and the current GIN efforts it turned out that mapping of the DEISA schema to other schemata is in some cases quite difficult since data are structured completely

differently. E.g. in DEISA there is no concept of Storage elements since GPFS is the main storage sink accessible from almost every supercomputer.
Furthermore, for the accounting initiative in SA5 the Usage Record format has been employed to describe jobs to be accounted. The UR-WG format in the meantime became a specification recommendation by the OGF and hence it would be almost natural to use it for describing job entities.
The move from the LDAP system to a WSRF based system requires the formulation of the RMIS schema in terms of a XML Schema definition. Hence, the RMIS schema has been re-factored. While doing so, two objectives were always kept in mind: More interoperability by including established information and avoiding ambiguities. The re-factored schema will be updated in minor revisions steps so that a possible transition towards UNICORE6 will be possible without much effort.

The below table summarizes that status of the RMIS deployment on the DEISA sites.

RMIS version	***Site***
1.0	ECMWF, EPCC, SARA
2.0	BSC, CSC, CINECA, FZJ, IDRIS, RZG
Not deployed	HLRS, LRZ

Table 4 – RMIS deployment status

3.5 DEISA co-allocation

3.5.1 Introduction to co-allocation and co-reservation topics for DEISA

For co-allocation we intend the capability of a software to interact and in some way control different resources management software allowing to request different resources on different machines. The co-allocation acts on the scheduler of a Batch System.
For co-reservation we intend the capability of a software to create a pool of resources that can be used in a specific timeframe by a pool of specific users.
These resources can belong to different organizations. It interacts with the resource manager of a Batch System.
Further details about preliminary studies concerning co-allocation, co-reservation and advanced reservation can be found in [8][9][10].

3.5.2 Platform solution for co-allocation and co-reservation.

As stated before in order to accomplish co-[allocation, reservation] an additional software is needed.
The following Figure 3 illustrates a possible software stack that enables DEISA infrastructure to address co-reservation and co-allocation.
The solution proposed is based and leverages Platform LSF and Universus
In this figure:

- LSF Universus is the "back-end" of LSF MC;
- LSF Universus acts as "translator" from the LSF job syntax to Load Leveler backfill job syntax, but also for other schedulers like SGE, NQS, PBSpro ;
- batch scheduler polices are set in Load Leveler backfill, SGE, NQS, PBSpro and LSF HPC;
- LSF MC can act as "single point of job submission";
- LSF MC can offer "floating" advance reservation or "co-scheduling";
- only one instance of LSF MC can be present in the infrastructure;

- There are accompanying tools coming out from the LSF Universus suite able to translate jobs syntax from Load Leveler backfill, SGE, NQS, PBSpro to LSF so to facilitate users in translating their jobs.

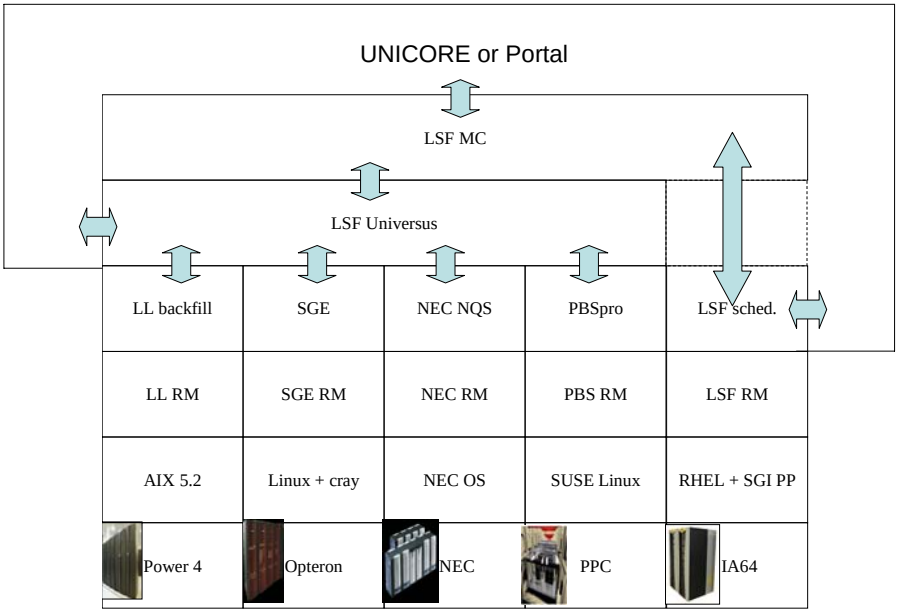

Figure 3 - Software stack for co-allocation and co-reservation

LSF and Universus are products developed by Platform. The DEISA consortium has commissioned Platform for implement a solution enabling co-allocation and co-reservation for DEISA infrastructure. A complete feasibility study can be found at [11].

Pons-Universus (or 'Universus') is a generic framework to cater for submission, monitoring and control of jobs from LSF to foreign resource managers (or 'schedulers'), such as PBS, SGE or LL.

Universus can operate on different execution scenarios.

Figure 4, Figure 5, Figure 6, Figure 7 illustrate four deploying scenario for Universus feasible for DEISA infrastructure.

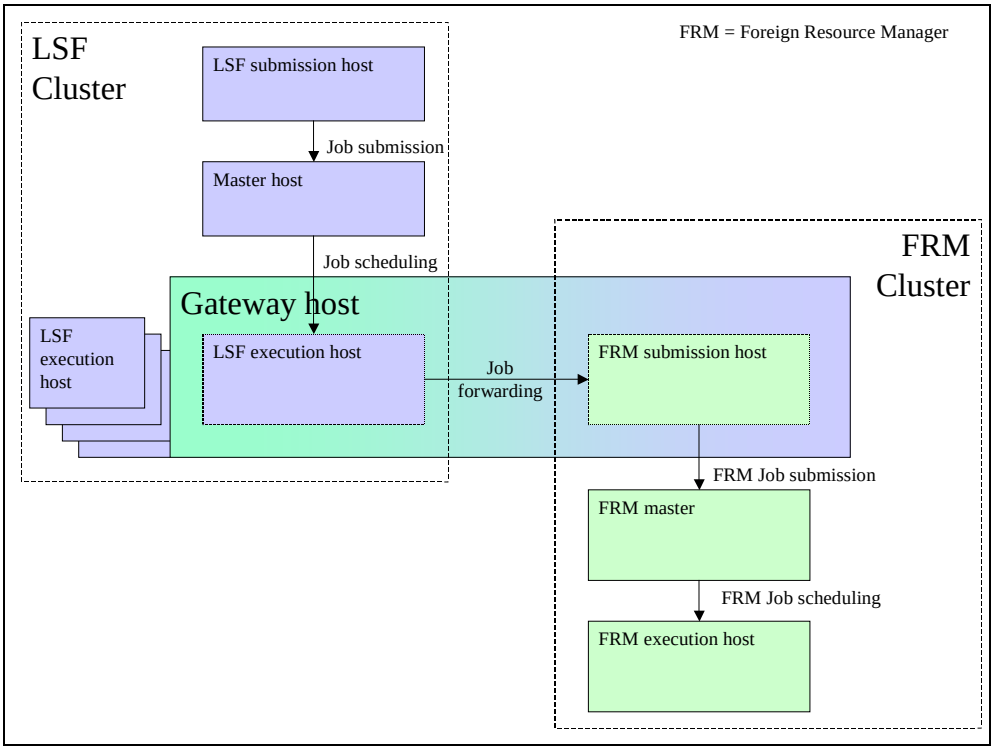

Figure 4 – Default physical layout of hosts in a Universus cluster

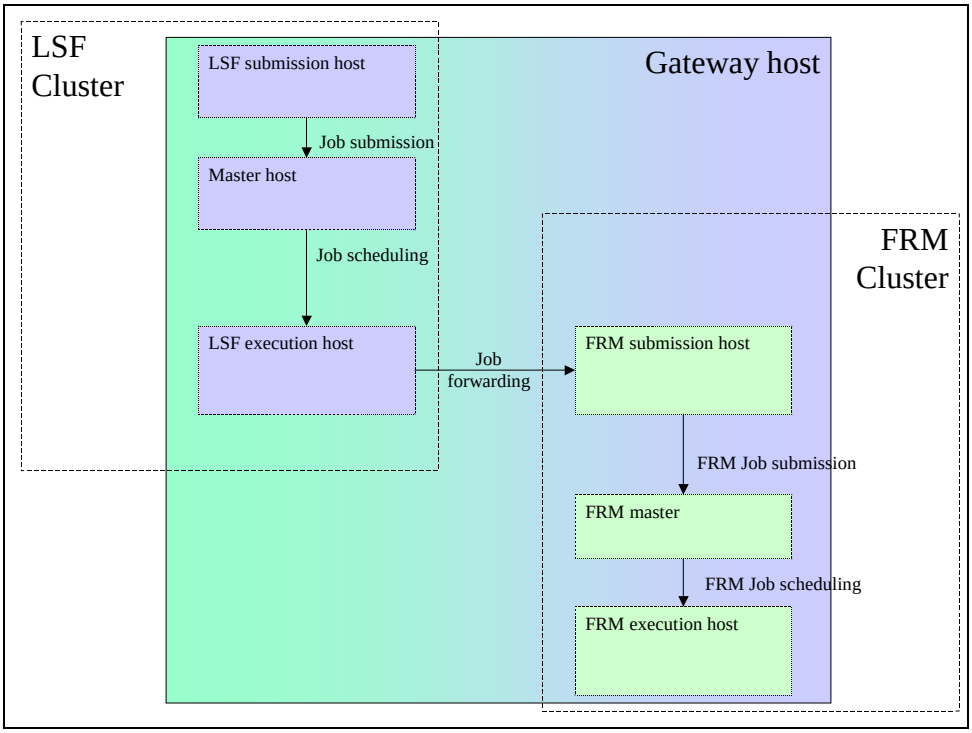

Figure 5 - Physical layout of hosts in a Universus cluster where all components are deployed on the same host

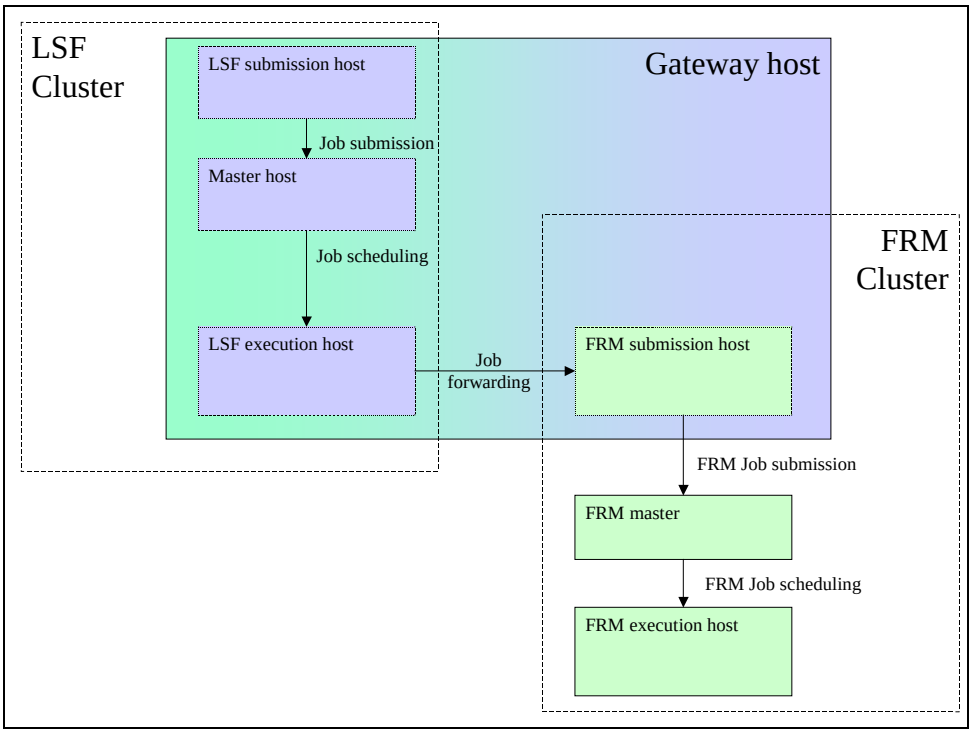

Figure 6 - FRM master and execution host reside in a different machine

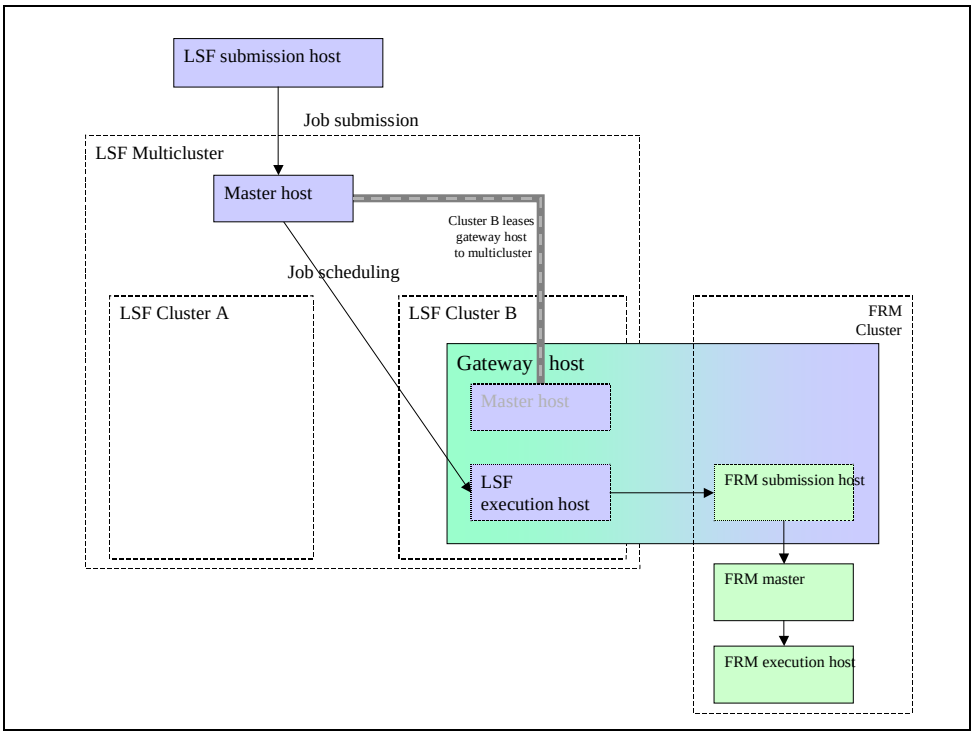

Figure 7: Multicluster configuration with Universus

Universus allows to implement co-reservation functionality by LSF MC leasing model. For details refers to [13].
Co-reservations in LSF MC are implemented as follows:

- Provider cluster:
  - Exports all or part of the gateway host to the consumer (remote) cluster(s)
  - The Reservation Translation Adapter (RTA) on the gateway host is configured to query MBD(s) of the consumer(s) (remote) cluster(s) . This way RTA on the imported gateway tracks advanced reservation in the remote cluster(s)

- Consumer (remote) cluster:
    - Defines queue(s) that use imported gateway hosts
    - Advanced reservations on imported gateway hosts are created through this cluster

RTA is responsible for keeping track of existing LSF reservations and updating (creating/deleting/modifying) corresponding reservations in the FRM.
Further technical details about co-reservation can be found in [11].

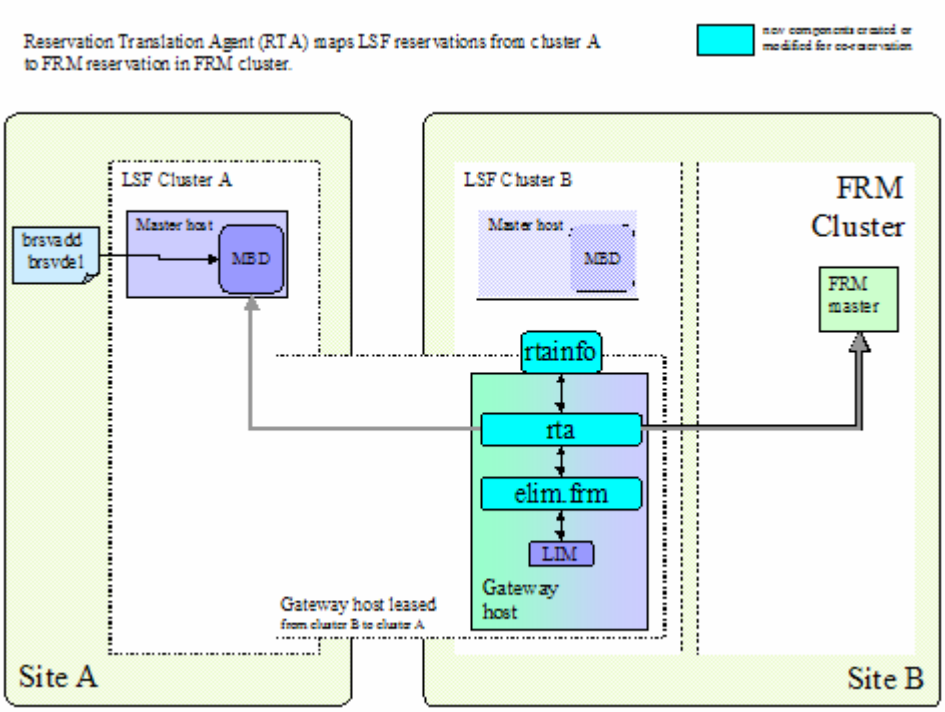

Figure 8 Co-reservation creation in Universus

3.5.3 Current status

In Phase I (Multi cluster job submission via Universus):
- Universus bugs were reported to Platform by the working group, they have been fixed;
- Enhancement / correction were asked for dealing with queues.

In Phase II (co-allocation and co-reservation via cluster leasing):
A- LL delivery:

- Coding and integration testing has been completed by Platform;
- Platform had two main problems that hindered deployment on site for the working group to test: 1) the package delivered had not been built entirely properly. Platform slightly changed their process, so this should not happen. 2) The LSF patch delivered was very probably causing problems on some commands on the CINECA LSF cluster; Platform redirected CINECA to another patch for the Multicluster problem.

B- PBS Pro:

- Coding and integration testing has been completed;
- for testing live in the LRZ site, Platform has been delayed by some slow synchronization on problems in the environment (mainly *sudo* access problems). First run was done, small corrections have been done and Platform is doing integration tests again.

C- NQS

- Platform is working with HLRS to get NQS reservation commands. The idea was to ask to get them as close to PBS Pro commands as possible.

In Phase III (scenario deployment):
- Platform and the DEISA working group worked on starting the setup or help setting up:
- The CINECA site
- The LRZ LSF single cluster for the phase I test setup
- The HLRS test environment.

Today, this task has not finalised as expected. The work will continue in the next project year.

3.6 DEISA data federation

3.6.1 Overview: the need of data federation in Deisa

Within the scope of data staging and data federation functionalities, the command-line data federation represents one of the possible way to access the DEISA infrastructure. To fulfil this requirement a UNIX-like command-line interface has been designed and implemented. The tool also provides an effective way of being integrated with LoadLeveler and PBSPro batch schedulers in order to automate data stage operations.

3.6.2 Prototype version 1.0 and 1.1

On PM28 a first prototype has been provided. The prototype was comprised of two Perl scripts meant to be the basic means for data transfers.
Possible transfer methods were cp, scp, ftp, and sftp.
Though the tool was able to transfer files between two generic machines, it did not implement the ‘remote staging’ functionality.

3.6.3 Prototype version 2.0

On PM31 a second prototype has been developed. The scope of the scripts has been reduced to allow data transfers between a generic file server and a DEISA GPFS mounting cluster only. It has been introduced the ‘remote staging’ feature allowing file transfer commands to be run from a remote DEISA cluster through a SSH channel.
Security issues have also been addressed. SSH interactive login has been disabled as specified in the initial requirements as well as any other command other than the data staging ones.
ECMWF, CINECA and LRZ tested this version successfully on their production machines. However, installation was a bit cumbersome due to the plethora of Perl-modules that were necessary.

3.6.4 Prototype version 3.0

The last version of the tool has been implemented on PM35.
This prototype can be integrated within a LoadLeveler job command file and it is compliant with DEISA job migration specifications. That required significant changes in the previous releases because every single file transfer operation must occur without password typing. This has been accomplished by setting automatic SSH file transfers by means of SSH key crossing between DEISA clusters.
Automatic file transfers must also be set up between a DEISA cluster and its file server that holds the file to be processed.
Security measurements have been maintained from version 2.0 by means of the SSH ‘force command’ feature.
ftp method has been removed for security reasons.
The core concept of the tool is the integration with a batch scheduler, in our case LoadLeveler. The DEISA job migration scenario in particular has been addressed.
The LoadLeveler job command file must be comprised of three job steps. The first one is the stage-in step, the second one is the computational step and the third one

is the stage-out step. The three steps are dependant from each other in the sense that each one can start only if and when the previous step terminates successfully.
That guarantees that we are not wasting computational resources or network bandwidth when one of the steps fails.
Data stage steps, one and three, are serial LL steps in order not to allocate unnecessary resources.
When a job is migrated, it gets executed remotely in a different DEISA cluster. The problem of the input file being still hold by the file server at the submitting site is dealt with by connecting via SSH and by running the local instance of the stage-in script. This operation moves the file to the shared GPFS file system allowing the computational step to run at the site where the job has been migrated to.
If everything is configured properly, a DEISA user just needs to submit its job and wait for the result. The job may be migrated or not and the output file is staged out where the user has specified in the job command file.
Although the tool had been designed to work with LoadLeveler, LRZ tested it successfully with PBSPro batch scheduler by just modifying accordingly the job submission script

3.7 DESHL deployment

This section describes the current status of the DESHL deployment to PM36.
DESHL provides standards-based access for users and their applications to manage jobs and transfer files in the DEISA heterogeneous supercomputing infrastructure.

3.7.1 DESHL v3 final deployment across all DEISA sites

Members of the SA3 DESHL WG have all installed the DESHL client at their respective sites and tested it against the DEISA production infrastructure. Of these sites there are outstanding issues with the installation at HLRS that mean testing against the DEISA infrastructure has as yet not been successful. This requires further investigation. At LRZ and RZG, v4 rather v3 of the DESHL client has been installed and partially tested against the DEISA infrastructure.

3.7.2 DESHL v4.0 deployment across DEISA JRA7 sites (EPCC, FZJ, CINECA, ECMWF)

DESHL v4.0 was made available by DEISA JRA7 in November 2006. This has been installed and tested at the subset of DEISA sites comprising the participants in DEISA JRA7. This deployment was solely for production testing against the DEISA operation platforms at these sites with DEISA JRA7 responsible for the actual production testing. The outcomes from this testing has fed into the JRA7 DESHL development process. In addition IDRIS under the auspices of DEISA SA4 have also installed and tested DESHL v4.0 in readiness for the DESHL's inclusion in the DEISA Primer sometime in 2007. Finally, BSC also installed and tested DESHL v4.0. This version was used by attendees at the DEISA User training session in Barcelona in February 2007 with feedback being passed on to DEISA JRA7

3.8 DCRMP

DCRMP task has been establish to create a standard way to distribute and to easy the deployment of software across the DEISA Grid. To this extent in the last year it has produced the DPC (DEISA Package Creator), a simple shell based application able of creating self contained executable file to install applications on UNIX like systems.
Further the DCRMP task has worked along with SA4 to produce several packages for the distribution of CPE applications as well as some software needed by DECI projects. The following list presents the packages created with the help of SA4:

- Python 2.4.3 64-bit version with included:
    - Readline 5.0
    - GDBM 1.8.3
- omniORB 4.0.6 64-bit (CORBA implementation)

- NAMD 2.6 (a Molecular Dynamics software)
- LAMMPS (another Molecular Dynamics software)

At the same time the DCRMP task has created several packages to distribute its middleware components, in particular:

- RMIS v2 which consisted of three packages:
    - Ganglia MBean
    - LoadLeveler MBean
    - MDS 4 front-end

During the creation of all these packages the DCRMP task has been releasing several versions of the DPC, in order to patch bugs and add features. Considering all this effort the DPC can finally claim a production level quality. Differently from what originally thought, this activity will not continue during the last project year.

3.9 Interoperability

3.9.1 Introduction

This activity concerns actions with aim to promote DEISA integration to European and extra European Grids. The activity, performed during the 3rd year, moved in two directions. First surveying evolution of standards developed by standardization organism and second implementing solution for making real grid integration.

3.9.2 Activity results

During the third year of the project an important demonstration of DEISA integration has been reached and shown to the community at SC06.
That is a demonstration of interoperability of three major Grid environments.
The combination of GridSAM, JMEA, and UNICORE offers users a great opportunity to submit jobs in a coherent way to different Grid infrastructures offering GridSAM as their gateway. This was actually exploited by a DEISA extreme computing initiative project.
The DECI project LIAMS [14] was carried out in 2006 by the Centre for Computational Science, UCL, London, UK. In two studies of biological macromolecules large-scale Molecular Dynamics (MD) simulations were performed on resources provided by DEISA, TeraGrid, and the National Grid Service (NGS) in the UK. In order to increase the ease of use, LIAMS simulation relied on the use of the Applications Hosting Environment [15] developed within the RealityGrid project [16] in the UK. AHE's focus is to provide scientists with application specific services to utilize grid resources in a quick, transparent manner with the scientific objective as the main driver of the activity. The job submission and management backend employed by AHE is GridSAM.
Figure 9 sketches the deployment of the different components grouped by organization. The AHE is deployed at the UCL and used by several users from the department of chemistry. These users interface the AHE with a rich client application. With its help they can make simulation datasets available and formulate job submissions interactively, which are upon submission converted into JSDL. The AHE takes care of submitting a job request to the selected computing infrastructure. For that purpose, the AHE interfaces GridSAM in the various deployment scenarios at NGS, TeraGrid, and DEISA.
As a result interoperation is achieved by providing access to these Grid infrastructures by giving the users the opportunity to submit and manage jobs with the same components.
This multi Grid infrastructure could be demonstrated at the SC|06 conference in Tampa, FL, USA [33]. For that purpose dedicated resources within DE-ISA, TeraGrid, and the NGS were accessed during several presentations held at various booths of participating partners in these Grid projects on the exhibition floor

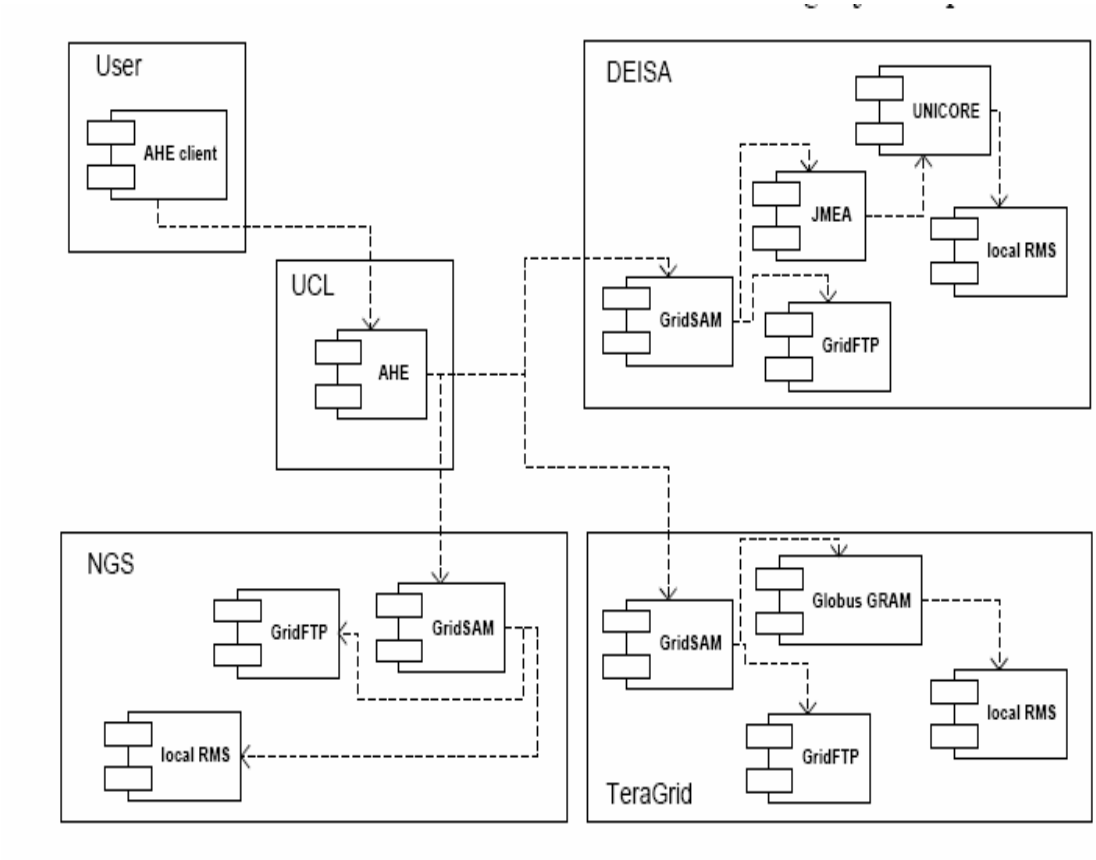

Figure 9 - AHE

For details about AHE and the demo refer to the article [17].
An other important outcome of the interoperability activity is the participation of a DEISA developer inside the GIN group of OGF [18]. Dr. Thomas Soddemann (RZG) is co-chairing inside OGF's Grid Interoperation Now Community Group (GIN-CG).

4. Roadmap evaluation and further plans

4.1 SA3 overall roadmap

The main goals of the SA3 activity for the last year will be

- consolidate the deployed infrastructure and facilitate the user exploitation,
- increase infrastructure exploitation by challenging application,
- promote interoperability among worldwide grid-empowered infrastructures.

4.2 SA3 further plans

4.2.1 LL-MC

Continue LL-MC, production update tuning and evolution by installing new releases of IBM LL-MC and new releases of DEISA filters to fix problems.
Stay open to SA4 new features requirements and manage the hand checking and interface with other DEISA services setup in an upper layer.

4.2.2 Unicore

CINECA and FZJ will do first tests with UNICORE and GridFTP to verify the technical possibility of GridFTP adoption within UNICORE. Before GridFTP will be then installed at all other sites the UNICORE group awaits the results delivered by the eSA3 – Global Data Management group which evaluates the general adoption of GridFTP in DEISA.
Actually UNICORE 5 is used for DEISA in production. Originally it was planned to evaluate and install the new web services based version of UNICORE coming from

the UniGrids Project which finished mid of 2006. The software developed in this project was called UNICORE GS. Due to decisions of the UNICORE community which caused an essential redesign of the following UNICORE software UNICORE/GS moved to UNICORE 6. This redeployment has delayed the delivery of the final release of UNICORE 6. Actually there is an alpha 7 release of UNICORE 6 available. But still there are a lot of changes in every new alpha release. Therefore the DEISA UNICORE group decided to install and evaluate UNICORE 6 once there is a first stable beta release. This is planned for June 2007.
As an additional service for all DEISA users it is planned to generate a special preconfigured UNICORE client which makes the installation of UNICORE even more comfortable. Also DEISA related UNICORE documentation, FAQs, job examples, etc. are planned to be collected and published on [2].

4.2.3 RMIS

Upgrade, maintain and operate the DEISA RMIS infrastructure at production quality level.

4.2.4 Co-allocation

For what concerns co-allocation and co-reservation SA3 will analyze the solution proposed by Platform and will conclude all tests expected by the Acceptance document. Universus will be installed and tested on LL-AIX and on PBS machines

4.2.5 Data federation

Data federation has successfully completed its on PM 36. The work done has been appreciated by sites that have tested and deployed.
For that SA3 has decided to operate for the last year of the project

- consolidating the software installed and deploying on new sites;
- deepen the tool configuration for batch schedulers different from LL;
- produce the technical documentation.

4.2.6 DESHL

Over the remainder of 2007, the intention is to install DESHL v4.1 across all DEISA sites. The SA3 DESHL Working Group will therefore undertake the following tasks from PM 37 to PM48.
The main goals for DESHL group are:

- Install DESHL v4.1 release due end of April 2007 (PM36). it will initially be installed and tested at the subset of DEISA sites comprising the participants in JRA7 with subsequent testing by JRA7 to ensure that these deployments are able to access the production platforms at all the DEISA sites. This is expected to have been completed by PM 38 (June 2007).
- On successful completion of DESHL v4.1 installation and production testing by the DEISA JRA7 sites, DESHL v4.1 will then be installed and tested across all the remaining DEISA sites. This is expected to be completed end of DEISA PM 42 that is October 2007. To the end of PM 48, members of the SA3 DESHL WG will be expected to answer occasional queries on DESHL usage from users at their respective sites.

4.2.7 DCRMP

In the last six months there have not been significant developments nor enhancements and nor request for packing are came from SA4. That confirms the maturity of the tools developed. This activity has probably exhausted its initial spin. The tools made available can be considered mature. For that, the activity remains hang for packing request. Next SA3 deliverable will not address the latest release of DCRMP but a SA3 status report.

4.2.8 Interoperability

In the third year DEISA infrastructure will be consolidated that will facilitate the cooperation and integration on world spread Grid.
Interoperability activity will continue to survey the standardization process concerning grid in order to address the final software development and facilitate the integration.