



Deliverable 3.7

Updated Request for Enhancements of gLite to support bio-NMR applications

SEVENTH FRAMEWORK PROGRAMME Research Infrastructures

INFRA-2007-1.2.2 - Deployment of eInfrastructures for scientific
communities

**Grant agreement for: Combination of Collaborative projects &
Coordination and support actions**

Proposal/Contract no.: 213010 – **e-nmr**

Project full title: Deploying and unifying the NMR e-Infrastructure in System Biology

Project coordinator: Prof. Dr. Harald Schwalbe

Project website: <http://www.enmr.eu/>

Period covered: from **01-05-2009** to **30-4-2010**

Table of contents

1. INTRODUCTION	4
1.1. PURPOSE	4
1.2. DOCUMENT ORGANISATION	4
1.3. REFERENCES	4
1.4. TERMINOLOGY	4
2. EXECUTIVE SUMMARY.....	6
3. SURVEY OF APPLICATIONS CURRENTLY AVAILABLE IN E-NMR GRID	7
3.1. APPLICATION DESCRIPTION	7
3.2. APPLICATIONS CHARACTERISTICS	8
3.2.1. <i>Is the application parallel (MPI) or sequential?</i>	8
3.2.2. <i>Is the application CPU intensive, data intensive or both?</i>	8
3.2.3. <i>Is the application interactive or batch oriented?</i>	8
3.2.4. <i>Does the application have security requirements?</i>	9
3.2.5. <i>Does the application have encryption requirements for data storage or transfer?</i>	9
3.2.6. <i>Does the application make use of grid enabled data storage for input and/or output data?</i> 9	
3.2.7. <i>What is the typical size of the application software package?</i>	10
3.2.8. <i>Is the application installed locally in each cluster or is it downloaded with the job submission?</i>	10
3.3. RESOURCE REQUIREMENTS.....	10
3.3.1. <i>What is the typical CPU time consumption per run?</i>	10
3.3.2. <i>What is the amount of RAM required at run and compile time?</i>	11
3.3.3. <i>What is the typical number of concurrent running/queued jobs?</i>	11
3.3.4. <i>Disk space requirements per run?</i>	12
3.3.5. <i>Does the application require direct network connectivity inbound/outbound or both?....</i>	12
3.3.6. <i>Typical input data size</i>	13
3.3.7. <i>Typical output size</i>	13
3.4. SOFTWARE DEPENDENCIES.....	13
3.4.1. <i>Operating system and version?</i>	13
3.4.2. <i>Compilers and versions?</i>	14
3.4.3. <i>Databases and versions?</i>	14
3.4.4. <i>Other required libraries?</i>	14
3.4.5. <i>System or general tools required at run or compile time?</i>	15
3.4.6. <i>Does the application need party commercial software to run?</i>	15
3.5. GLITE REQUIREMENTS	16
3.5.1. <i>Which grid elements other than CE/WNs your application make use of? (e.g. WMS, SE, LFC, AMGA, HYDRA, MYPROXY, CREAM)</i>	16
3.5.2. <i>Does the application use standard Data management functionality? (i.e. file registration into LFC file catalogue, file access through LFN, file transfer/replication through standard lcg- utils)</i> 16	
3.5.3. <i>How many files per run are registered with the LFC file catalogue?</i>	17
3.5.4. <i>Does the application use advanced functionality for Data Encryption? (EDS / Hydra servers)</i>	17
3.5.5. <i>Are the normal users submitting the jobs from a UI or through a web portal?</i>	17
3.5.6. <i>Does the application use WS, Java or C/C++ APIs for job submission and/or data management, or simply glite-wms-*, lcg-*and lfc-* command line tools?</i>	18
3.5.7. <i>Does the application make use of Collection, Parametric or DAG job type functionalities?</i> 18	
3.5.8. <i>Did you find any limitation on gLite current functionalities?</i>	18
3.5.9. <i>Did you find any limitation on gLite current performances?</i>	18
3.5.10. <i>Did you expect any functionality from gLite that was completely missing in the current release, and which could be of benefit for your application?</i>	19
4. GLITE ENHANCEMENTS ACHIEVED.....	20

4.1.	DGAS SENSORS.....	20
4.2.	HLRMON.....	22
4.3.	MPI SUPPORT.....	23
5.	CONCLUSIONS	25

Document log			
Issue	Date	Comment	Author
0- 1	15-04-10	First Draft	Marco Verlato
0- 2	21-04-10	BMRZ contribution	Peter Guenter, Anurag Bagaria, Victor Jaravine
0- 3	22-04-10	BCBR contribution	Nuno L. Ferreira, Alexandre Bonvin, Tsjerk Wassenaar
0- 4	29-04-10	Document Review	Marco Verlato, Eric Frizziero
1- 0	30-04-10	Final Revision by MB	Project Management Board

1. Introduction

1.1. Purpose

This document is the project deliverable D3.7 due by Month 30. It aims at reporting the activities of the task T3.2: Enhancements of grid middleware for e-NMR support, assigned to the Work Package 3 of the e-NMR project. It is the updated version of the document D3.3 previously delivered at Month 18.

1.2. Document organisation

The document is organised as follows:

Section 1 contains the purpose of the document, its references and a glossary of terms and acronyms;

Section 2 summarizes the content of the document;

Section 3 reports the results of a survey distributed inside the consortium partners about the bio-NMR applications currently deployed on the e-NMR grid;

Section 4 describes some enhancements introduced in the accounting components and planned in MPI support which are of interest for e-NMR applications;

Finally, Section 5 tracks the conclusions.

1.3. References

[R1]	http://knowledge.eu-egi.eu/knowledge/index.php/UMD	EGI / UMD Knowledge Base
[R2]	https://edms.cern.ch/document/855382	Grid Policy on the Handling of User-Level Job Accounting Data
[R3]	http://www.iop.org/EJ/article/1742-6596/119/5/052012/jpconf8_119_052012.pdf	“HLRmon: a Role-based Grid Accounting Report Web Tool”, Journal of Physics: Conference Series 119 (2008) 052012
[R4]	http://grid.ie/mpi/wiki/WorkingGroup?action=AttachFile&do=get&target=EGEE-II-MPI-WG-TEC.doc	EGEE-II MPI WG Report
[R5]	http://grid.ie/mpi/wiki/WorkingGroup?action=AttachFile&do=get&target=MPIWG-recommendation-final.doc	EGEE-III MPI WG Recommendations

1.4. Terminology

This subsection provides the definitions of terms, acronyms, and abbreviations required to properly interpret this document.

Term	Definition
ACL	Access Control List
AMGA	gLite Metadata Catalogue service
BCBR	Bijvoet Centre for Molecular Research, University of Utrecht, The Netherlands
BMRZ	Centre for Biomolecular Magnetic Resonance, Goethe University, Frankfurt, Germany
CE	Computing Element
CIRMMMP	Interuniversity Consortium for Magnetic Resonance on Metalloproteins, Florence, Italy
CREAM	Computing Resource Execution And Management
DAG	Direct Acyclic Graph, basic job workflow implemented by gLite
DGAS	Distributed Grid Accounting System
DoW	Description of Work
EGEE	Enabling Grids for e-Science
EDS	Encrypted Data Storage
EGI	European Grid Initiative
EMI	European Middleware Initiative
gLite	Codename of the middleware software suite developed by EGEE
HLR	Home Location Register
Hydra	Keystore server for encrypted file storage solution
IGI	Italian Grid Infrastructure
INFN	National Institute of Nuclear Physics, Italy
JSPG	Joint Security Policy Group
LCG	LHC Computing Grid
LFC	LCG File Catalogue
LFN	Logical File Name
MPI	Message Passing Interface
NGI	National Grid Initiative
NMR	Nuclear Magnetic Resonance
SAGrid	South African National Grid Initiative
SE	Storage Element
UI	User Interface
UMD	Unified Middleware Distribution
VO	Virtual Organization
VOMS	Virtual Organisation Membership Service
WMS	Workload Management System
WP	Work Package
ZA-UCT-ICTS	University of Cape Town – Information and Communication Technology Services

2. Executive summary

This document mainly describes the requirements of the new bio-NMR applications deployed on the e-NMR grid after Month 18 and before the end of Month 30.

The same survey of the previous D3.3, with 31 questions about characteristics of the new applications made available on the e-NMR platform was circulated within the partners of the consortium. The questions were grouped into 5 items: Application description, Application characteristics, Resource requirements, Software dependencies and gLite requirements.

It turned out that also the new applications deployed on the e-NMR grid had not very stringent requirements and could quite easily be interfaced with the latest gLite middleware exploiting its very basic functionalities.

However, some enhancements on the grid middleware for what concerns the accounting components were implemented by the gLite developers upon request of e-NMR project, in order to allow per-application accounting. A description of such enhancements is also given in this document.

As pointed out in the previous D3.3, for some of the applications there was a plan for the second phase of the project to exploit more advanced gLite capabilities like MPI support and one-shot job collections submission. However, the progresses made by the EGEE MPI working group during the last year were mainly directed on providing recommendations rather than quickly implement enhanced MPI support on the production grid, so it was not possible up to now to make available reliable MPI-enabled grid components to e-NMR infrastructure. A summary of the results of the EGEE MPI working group and future perspectives for e-NMR are given here.

3. Survey of applications currently available in e-NMR grid

3.1. APPLICATION DESCRIPTION

TALOS+

TALOS+ is a program which predicts torsion angles using information derived from chemical shift, amino acid type, and the area of the Ramachandran map where the residue is likely to reside in. This information is compared to information stored in a large database, currently 200 proteins. Moreover, it includes a neural network component whose output is used as an additional empirical term. The neural network component of the program relies on a well established computational framework that optimizes the relation between a large number of input variables. The input data for this web server are the $^{13}\text{C}^\alpha$, $^{13}\text{C}^\beta$, $^{13}\text{C}'$, ^{15}N , $^1\text{H}^\alpha$ and $^1\text{H}^\text{N}$ NMR backbone chemical shifts of the query protein, either in TALOS or in BMRB format.

3D-DART

The 3D-DART server generates 3D structural models of double stranded DNA in either A-form or B-form. The modelling process requires an initial template base-pair parameter file representing the 3D conformation of a DNA starting structure. This parameter file is subsequently used in further modelling actions. The 3D models generated by 3D-DART can be used as starting structures for the macro-molecular docking program HADDOCK.

Glycans

Polysaccharides (glycans) are polymers made of several monosaccharides sequentially ramified via glycosidic bonds. These molecular entities are often conjugated to proteins or lipids and intervene in many different biological processes such as folding pathway, signal transduction or cell communication. This web server proposes a toolbox which helps the build up of glycans making use of CNS program.

GARANT

GARANT is an algorithm for automatic resonance assignment using as input the primary structure of a protein and lists with observed cross peaks from various spectra; knowledge about magnetization transfer pathways in the NMR experiments used is read from a library. The basic concept of GARANT is the matching of observed cross peaks to expected cross peaks derived from the sequence and the magnetization transfer pathways.

MARS

MARS is a program for robust automatic backbone assignment of $^{13}\text{C}/^{15}\text{N}$ labelled proteins. MARS simultaneously optimizes the local and global quality of assignment to minimize propagation of initial assignment errors and to extract reliable assignments. It can be applied independent of the assignment complexity, it does not require tight thresholds for establishing sequential connectivity or detailed adjustment of these thresholds, it can work with a wide variety of NMR experiments and it is robust against missing chemical shift information.

3.2. APPLICATIONS CHARACTERISTICS

3.2.1. Is the application parallel (MPI) or sequential?

TALOS+

Sequential

3D-DART

Sequential

Glycans

Sequential

GARANT

Sequential or parallel

MARS

Sequential

3.2.2. Is the application CPU intensive, data intensive or both?

TALOS+

None

3D-DART

None

Glycans

CPU intensive – runs CNS

GARANT

CPU intensive

MARS

CPU intensive

3.2.3. Is the application interactive or batch oriented?

TALOS+

Batch oriented after input from the user via the web form

3D-DART

Batch oriented after input from the user via the web form

Glycans

Batch oriented (runs CNS) after input from the user via the web form

GARANT

Batch oriented

MARS

Batch oriented

3.2.4. Does the application have security requirements?

TALOS+

No

3D-DART

No

Glycans

No

GARANT

No

MARS

No

3.2.5. Does the application have encryption requirements for data storage or transfer?

None of the applications considered have such requirement

3.2.6. Does the application make use of grid enabled data storage for input and/or output data?

TALOS+

No

3D-DART

No

Glycans

No

GARANT

Yes

MARS

Yes

3.2.7. What is the typical size of the application software package?

TALOS+

24 MB

3D-DART

1 MB

Glycans

250 MB – mainly CNS

GARANT

0.5 GB

MARS

0.2 GB

3.2.8. Is the application installed locally in each cluster or is it downloaded with the job submission?

For all of the applications considered the software is installed locally on each CE

3.3. RESOURCE REQUIREMENTS

3.3.1. What is the typical CPU time consumption per run?

TALOS+

Depends on the size of the system, from seconds to minutes

3D-DART

Depends on the size of the system, from seconds to minutes

Glycans

CPU requirements vary depending on the type of calculations performed, typically from a few minutes to several hours

GARANT

200 – 2000 hours

MARS

Small for small proteins (minutes); combinatorial time (days) for large proteins

3.3.2. What is the amount of RAM required at run and compile time?

TALOS+

Negligible

3D-DART

Negligible

Glycans

1 GB

GARANT

1 - 2 GB

MARS

1 GB

3.3.3. What is the typical number of concurrent running/queued jobs?

TALOS+

A few

3D-DART

A few

Glycans

A few

GARANT

10

MARS

1

3.3.4. Disk space requirements per run?

TALOS+

Depends on the system size, typically a few hundred KB's

3D-DART

A few MB

Glycans

A few MB

GARANT

500 MB

MARS

Small

3.3.5. Does the application require direct network connectivity inbound/outbound or both?

TALOS+

No

3D-DART

No

Glycans

No

GARANT

No

MARS

No

3.3.6. Typical input data size

TALOS+

Depends on the system size, typically a few KB's

3D-DART

Depends on the system size, typically a few KB's

Glycans

No file input data – only via web forms

GARANT

50 MB

MARS

Small

3.3.7. Typical output size

TALOS+

Depends on the system size, typically a few hundred KB's

3D-DART

Depends on the system size, typically a few MB's.

Glycans

Depends on the system size, typically a few MB's.

GARANT

500 MB

MARS

Very small (several text files)

3.4. SOFTWARE DEPENDENCIES

3.4.1. Operating system and version?

TALOS+

Scientific Linux on 32 or 64 bits Intel processors

3D-DART

Tested on Scientific Linux and MacOSX. Should run on every certified POSIX system

Glycans

Scientific Linux on 32 or 64 bits Intel processors and MacOSX

GARANT

Linux

MARS

Linux

3.4.2. Compilers and versions?

TALOS+

No need. Pre-compiled in static mode

3D-DART

No need – python code

Glycans

CNS pre-compiled in static mode using Intel compilers

GARANT

gcc, g++, gfortran or Intel Fortran compiler

MARS

gcc

3.4.3. Databases and versions?

None of the applications considered makes use of databases

3.4.4. Other required libraries?

TALOS+

Python 2.6; Numpy; pylab

3D-DART

Python >= 2.3 ; Numpy ; 3DNA package

Glycans

No

GARANT

No

MARS

No

3.4.5. System or general tools required at run or compile time?

TALOS+

Tar, gzip

3D-DART

Gzip

Glycans

Tar, gzip, gunzip

GARANT

None

MARS

No.

Other software: requires installation of **PALES**

3.4.6. Does the application need party commercial software to run?

TALOS+

No

3D-DART

No

Glycans

CNS which is free for non-profit organizations

GARANT

No

MARS

No

3.5. GLITE REQUIREMENTS

3.5.1. Which grid elements other than CE/WNs your application make use of? (e.g. WMS, SE, LFC, AMGA, HYDRA, MYPROXY, CREAM)

TALOS+

Runs locally on the web server, the grid power is not needed

3D-DART

Runs locally on the web server, the grid power is not needed

Glycans

CPU intensive option (not the default usage) might be ported to the grid.

GARANT

WMS, SE

MARS

WMS, SE

3.5.2. Does the application use standard Data management functionality? (i.e. file registration into LFC file catalogue, file access through LFN, file transfer/replication through standard lcg-utils)

TALOS+

No

3D-DART

No

Glycans

No

GARANT

No

MARS

No

3.5.3. How many files per run are registered with the LFC file catalogue?

TALOS+

None

3D-DART

None

Glycans

None

GARANT

None

MARS

None

3.5.4. Does the application use advanced functionality for Data Encryption? (EDS / Hydra servers)

None of the applications considered makes use of such functionality

3.5.5. Are the normal users submitting the jobs from a UI or through a web portal?

TALOS+

Web portal

3D-DART

Web portal

Glycans

Web portal under construction (beta version)

GARANT

Web portal

MARS

Web portal

3.5.6. Does the application use WS, Java or C/C++ APIs for job submission and/or data management, or simply glite-wms-*, lcg-*and lfc-* command line tools?

All applications make use of the command line tools

3.5.7. Does the application make use of Collection, Parametric or DAG job type functionalities?

TALOS+

No

3D-DART

No

Glycans

No

GARANT

No

MARS

No

3.5.8. Did you find any limitation on gLite current functionalities?

All applications answered no

3.5.9. Did you find any limitation on gLite current performances?

TALOS+

No

3D-DART

No

Glycans

No

GARANT

No

MARS

No

3.5.10. Did you expect any functionality from gLite that was completely missing in the current release, and which could be of benefit for your application?

All applications answered no, at the current stage

4. gLite enhancements achieved

As pointed out in the conclusions of D3.3, all of the gLite enhancements required by the bio-NMR application developers were found to be in common with the ones already identified and well detailed in many documents produced in the past by the EGEE Life Science cluster. The most update list can be found at the link **[R1]** set up by the EGI_DS project with the goal of providing input from the users' communities towards the definition of future middleware developments in the EGI era, after the end of EGEE project phase III expected in April 2010.

Nevertheless, after the issue of D3.3 in April 2009, a new requirement emerged from the project interim review: to implement per-application accounting on the e-NMR grid resources. It was chosen to implement it exploiting the granularity of the VOMS service, defining a VO group for each application run over the e-NMR grid. Such implementation however has led to a request for enhancement to the DGAS middleware component and its visualisation tool HLRMon, as described in the following sub-sections 4.1 and 4.2.

Moreover, during the last year, the gLite support to parallel applications via MPI protocol has shown its limitations with respect the production use, especially from the scientific communities in the field of Theoretical Physics, Chemistry and Life Sciences. MPI support is of interest also for e-NMR, since a number of bio-NMR applications currently running on the grid in sequential mode can also benefit of running in parallel mode. Thus, the INFN team has closely followed the activity of the MPI task force, set up by EGEE in September 2009 with the goal of overcoming such limitations, in order to be ready to deploy the identified solutions into the e-NMR grid. A summary of the current status of MPI support in EGEE is given below in the sub-section 4.3.

4.1. *DGAS sensors*

The Distributed Grid Accounting System (DGAS) was originally developed by INFN within the EU-DataGrid Project (until March 2004) and later maintained and re-engineered within the EGEE project.

Currently the DGAS system is deployed on the IGI production grid, and is one of the four accounting systems available for grid infrastructures, the others being APEL (used in EGEE), SGAS (used within the North European countries), and Gratia (used in the Open Science Grid US project).

All of the four systems are interfaced with the EGEE Accounting Portal, a graphical front-end to accounting data. DGAS has also its own graphical front-end called HLRmon.

It has been used since the beginning in the e-NMR grid infrastructure in order to account the usage of computing resources supporting the enmr.eu VO.

Its deployment schema has been widely described in the document D2.2 delivered at Month 18. An updated version of this schema at Month 30 is given in figure 4-1.

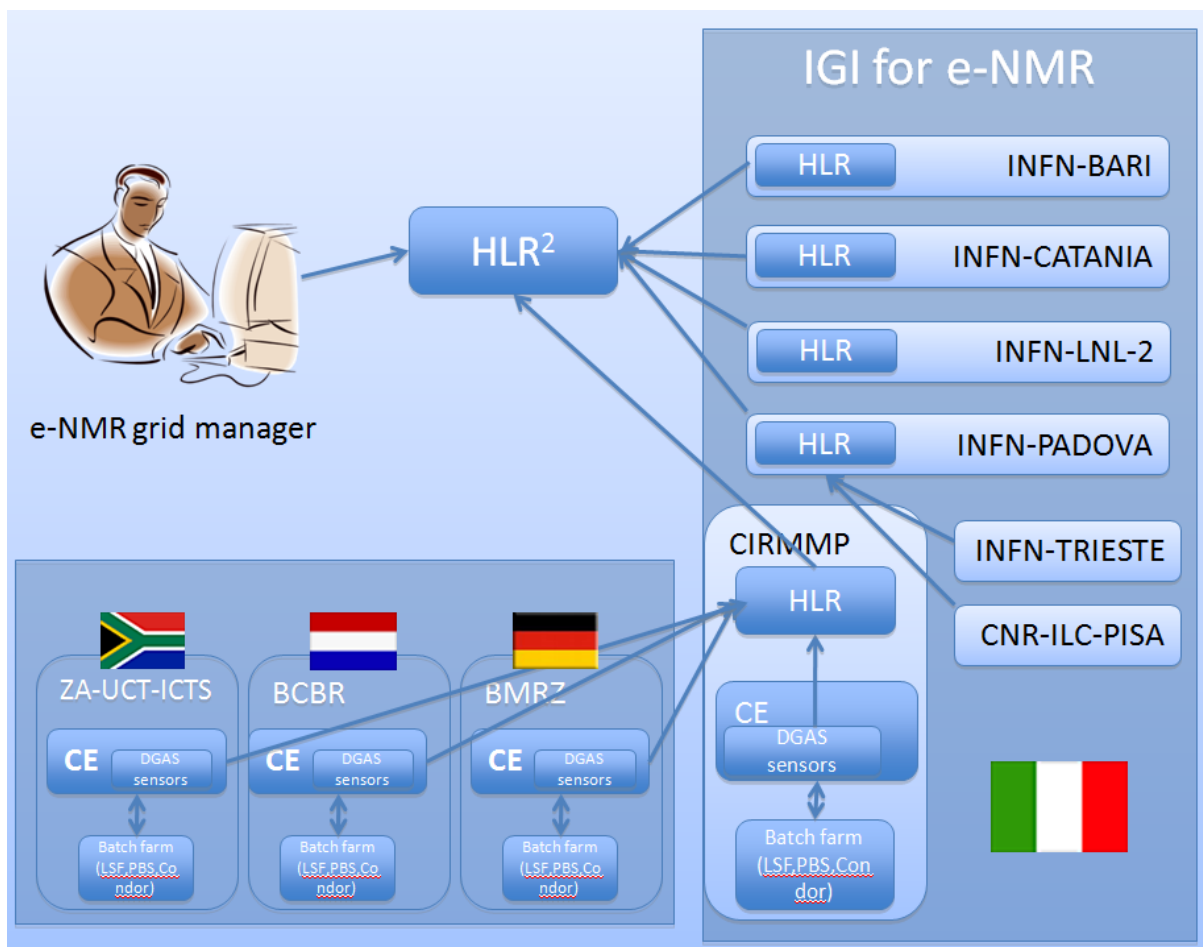


Figure 4-1: Overview of DGAS components deployed for serving e-NMR grid

The new layout includes the recently added SAGrid site ZA-UCT-ICTS, which installed the DGAS sensors on its Computing Element.

The standard DGAS deployment architecture actually would recommend a first level HLR (Home Location Register) at each site, which in turn send usage records to a second level HLR managed by the NGI or a VO. It is thus fully compliant with the “Grid Policy on the Handling of User-Level Job Accounting Data” document [R2] approved in 2009.

However, in order to make easier the work of the grid site administrators outside IGI, the above simplified architecture shown in figure 4-1 was proposed, where DGAS sensors of some CEs directly send usage records to the first level HLR hosted at CIRMMP of Florence.

This has led to a request for enhancement of the DGAS sensors functionality in order not to break the grid accounting policy rules defined in document [R2], stating e.g. that the access to aggregated data at the VO level may be public information only if authorized by the VO. DGAS sensors initial versions were sending usage records to HLR irrespectively of their VO.

After having acknowledged the above requirement, the DGAS team implemented in the sensors code the possibility of selecting the VO for which to send the usage records. Such an enhancement was put in production since the DGAS version 3.4.0, released in February 2010 with the INFNGRID gLite 3.2/x86_64 Update 07 (SL5) and gLite 3.1/i386 Update 59/60 (SL4).

4.2. HLRmon

HLRmon is the visual web component of DGAS, enabling access to accounting information for managers, administrators and users [R3]. HLRmon gathers the accounting information via the DGAS command line tools to query the HLR accounting database and displays a variety of accounting information such as Jobs per VO, Wall time per VO, CPU time per day, etc.

Information is presented in a graphical or tabular form in order to satisfy different user needs. The tool has been already described in the document D2.2 delivered at Month 18.

In order to implement the per-application accounting on the e-NMR grid, the solution of defining a VO group in the VOMS server for each application was adopted.

At the same time, a request for enhancement was directed to the HLRmon team in order to enable the visualisation of accounting information aggregated by VO groups/roles.

The request was acknowledged and implemented in the last release of HLRmon, which has been put in production in January 2010. The first results were published in the document D2.4 delivered at Month 27.

In the figure 4-2 and 4-3 below are shown the number of jobs per site (including the new SAGrid site results) and per VO group/role accounted in the last 3 months.

Jobs per Site [unit: Jobs number] 2010/02/01-2010/04/30

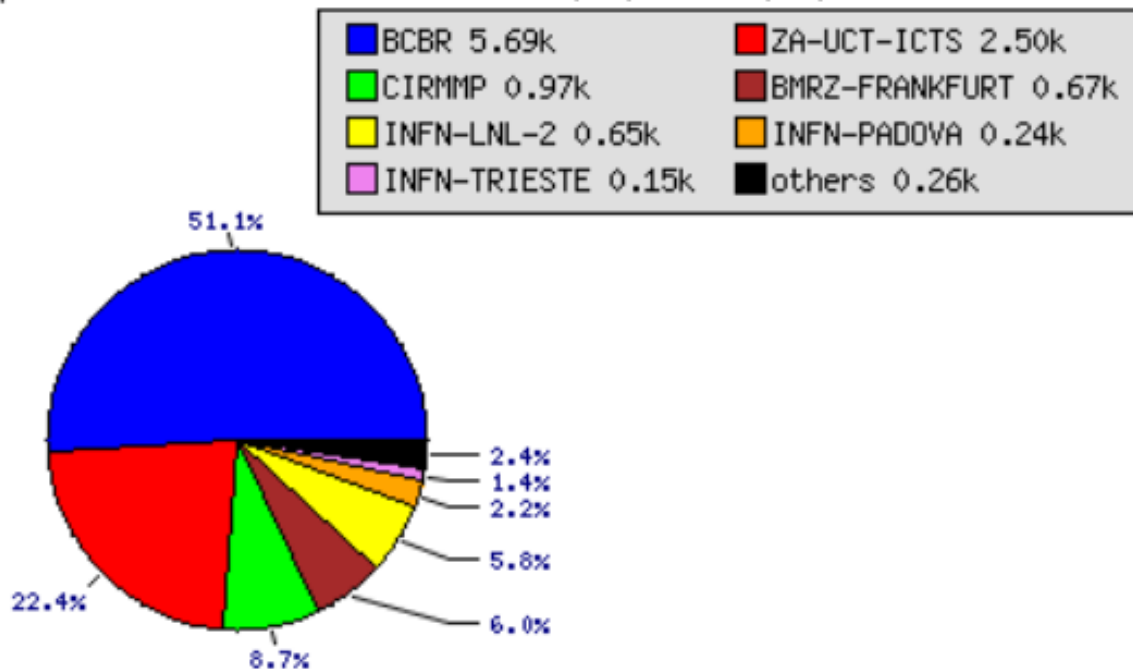


Figure 4-2: Job number distribution among sites accounted by DGAS

Jobs per VOMS role [unit: Jobs number] 2010/02/01-2010/04/30

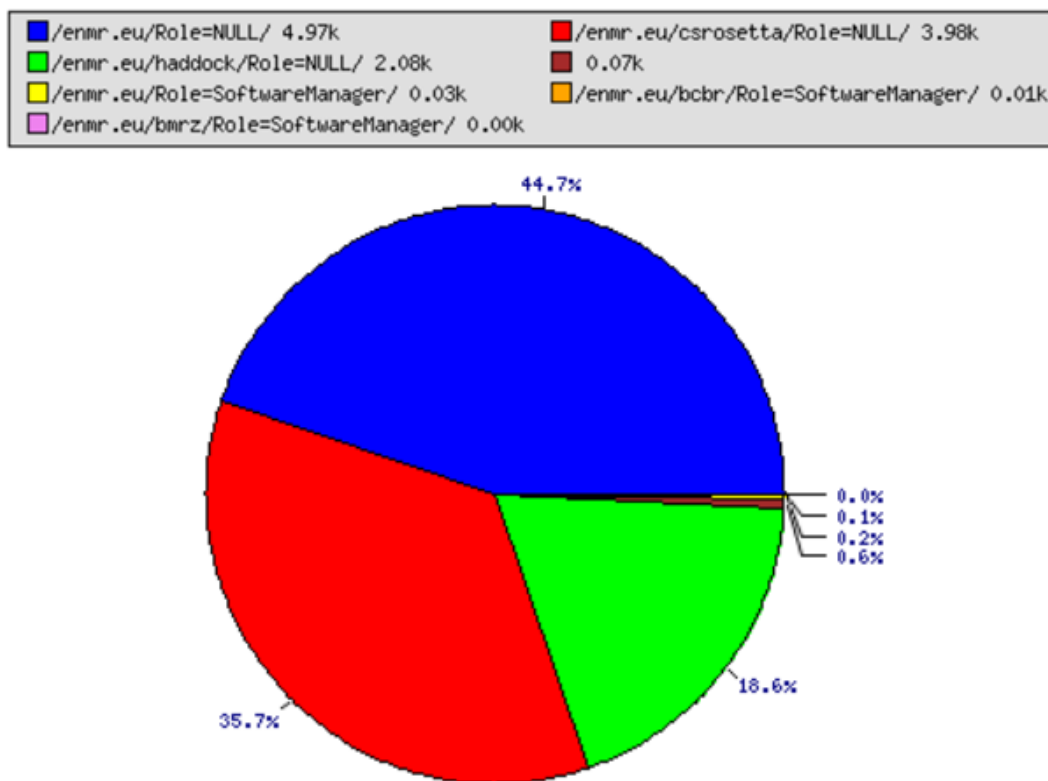


Figure 4-3: Job number distribution among VO groups. The jobs accounted under /enmr.ru/csrosetta and /enmr.eu/haddock groups correspond to CS-Rosetta and Haddock applications.

4.3. MPI support

The gLite middleware has been initially designed for serial job farming, perfectly addressing the High Energy Physics (HEP) needs. However, EGEE rapidly expanded beyond HEP, and currently more than 10% of the CPU-time provided by the EGEE production grid is consumed by scientific disciplines other than HEP, like Chemistry, Biology, Medical Imaging etc. In these communities the use of parallel programs is quite common and therefore the need for MPI on the EGEE grid increased.

However, the heterogeneity of grid infrastructures in terms hardware architectures, batch schedulers (LSF, PBS, SGE, etc.), network types (GigaEthernet, Infiniband, Myrinet, etc.), together with the availability of several MPI flavours (LAM, OpenMPI, MPICH), has made complex the provisioning of a reliable support to parallel applications in a grid environment.

The first working group set up to improve MPI support of gLite dates back to EGEE-II in 2006. Its results were reported in a document [R4] containing an analysis of existing limitations and proposing solutions in the areas of site configuration, middleware, and user submission methodology. The MPI working group made a setup procedure for site administrators to setup their site for use with MPI. Via this procedure several MPI flavours are installed and the site information system is updated to publish the MPI flavours available to the users. The MPI-start package, developed in the context of the Int.eu.grid FP6 project,

became also part of this procedure. Minor modifications have been made to the middleware, so that nowadays multiple CPUs can be requested in normal jobs. However, such recommendations were implemented through changes in the middleware incorporated only in the gLite WMS update 3.1.12-0 of 25 February 2009.

Still, in EGEE-III phase, the number of MPI enabled clusters was small, and even for those few sites the MPI functionality was marked as not satisfactory by the users. A new working group was then set up in EGEE-III project to investigate why so few sites have MPI enabled. Two questionnaires were proposed: one for (potential) grid users and one for system administrators. The initial scope of the working group was to come up with a set of recommendations, rather than to solve the issues. However, during the session organised by the group on the EGEE'09 conference in September 2009, it became clear that the time had come to act. Therefore, during the plenary, two people were appointed as Task Force members, receiving a mandate to get things done. The result of this effort, together with a deep analysis of the above surveys are the basis of the new recommendations produced by the working group and published in a document [R5] at the end of EGEE-III project in April 2010.

The most relevant news with respect to the report of 2006 concerns the detailed analysis of implications on the gLite middleware to enable the running of MPI jobs on multiple cores within a node. Today in fact most (if not all) computing elements on the grid have nodes which are equipped with multi-core processors and could therefore run very tightly coupled codes, at least in a limited fashion (using typically up to 8 CPU-cores).

Thus, the working group has suggested the introduction of three new JDL attributes in order to allow users to specify: how cores should be distributed over the cluster (*SMPGranularity*); whether whole (full) nodes should be used (*WholeNodes*); how many nodes should be used (*NodeNumber*).

Guidelines on how to modify the gLite WMS and CE middleware components have also been provided to the corresponding production teams of the gLite Open Collaboration, established between the EGEE partners involved in the middleware activity as a new framework for the maintenance and future evolution of the gLite middleware, beyond the end of the EGEE series of projects. Such activity in fact is planned to continue in the context of EGI within the European Middleware Initiative (EMI), a project aiming to provide a production quality grid middleware for EGI.

INFN is playing a leading role in the gLite WMS and CE product teams. This put e-NMR project in the best position to quickly deploy on its grid infrastructure the new MPI enhanced components as soon as these will be released in the coming months. Tests of bio-NMR applications running over the grid in parallel mode will be carried out immediately after.

5. Conclusions

A gap analysis between the current grid middleware and the e-NMR requirements has been performed focusing on five new applications ported to grid in the last year. Together with the eleven applications analysed in the previous D3.3 document, a comprehensive picture of the bio-NMR grid-enabled applications is now available.

This showed that gLite middleware is largely adequate for e-NMR, although in the future increased MPI support would be desirable. The consortium is closely following the activity of the MPI working group set up in EGEE and planned to continue its effort in the context of EMI, so that future new developments can be quickly incorporated in the e-NMR grid as soon as ready for deployment.

During the reporting period, a number of enhancements had been requested to gLite developers to enable operational support for per-application accounting. These were promptly acknowledged and have been quickly implemented in the production services, thanks to an effective collaboration with the middleware development group of EGEE.

The e-NMR project has confidence in keeping alive such communication channel with the gLite Open Collaboration, which is planning its future activity in the context of the recently approved EMI project starting in May 2010.