EGI-InSPIRE

HUC SOFTWARE ROADMAP

EU MILESTONE: MS602

Document identifier:	EGI-MS602-V10
Date:	**06/01/2011**
Activity:	**SA3**
Lead Partner:	**INFN**
Document Status:	**FINAL**
Dissemination Level:	**PUBLIC**
Document Link:	https://documents.egi.eu/document/230

Abstract

This document gives a first overview of the Roadmap for the development and deployment in the reference user communities of the software included in EGI-InSPIRE WP6. The Heavy User Communities who are part of the project are the primary target of the document, which is intended to give them information on the features available now and in the future, and offer the opportunity to interact with the planned developments so that they can best fit their needs.

However the document is open to other EGI-InSPIRE users and potential users, who can be interested in adopting parts of the software for their uses and in suggesting developments to this effect.

This document is a first step in the elaboration of a full roadmap (the HUC Software Roadmap is a Milestone due each year in SA3) as the material included in it is the result of the effort of the developers and of the single communities, while the interaction between the developers and between the communities has still to be strengthened, also on the basis of the information included here.

This goal of providing the first step in the development of a fully -fledged roadmap is reflected in the level of elaboration of the material included in this document which, albeit rather inhomogeneous and including in some cases a planning with detailed milestones and in other cases just a few indications for the future planning, fits the purposes of this document.

I. COPYRIGHT NOTICE

II. DELIVERY SLIP

	Name	Partner/Activity	Date
From	Laura Perini, Jamie Shiers	INFN/SA3	14/10/2010
Reviewed by	**Moderator:** Gergely Sipos **Reviewers:** **:** Jan Kmunicek Nuno L. Ferreira	EGI.eu	09/11/2010
Approved by	**AMB & PMB**		20/12/2010

III. DOCUMENT LOG

Issue	Date	Comment	Author/Partner
1	14/10/2010	First draft	Laura Perini/INFN
2	16/11/2010	Revised Draft after Reviewers comments	Laura Perini/INFN Jamie Shiers/CERN
3	17/12/2010	Revised for new template	Laura Perini/INFN

IV. APPLICATION AREA

This document is a formal deliverable for the European Commission, applicable to all members of the EGI-InSPIRE project, beneficiaries and Joint Research Unit members, as well as its collaborating projects.

V. DOCUMENT AMENDMENT PROCEDURE

Amendments, comments and suggestions should be sent to the authors. The procedures documented in the EGI-InSPIRE "Document Management Procedure" will be followed: https://wiki.egi.eu/wiki/Procedures

VI. TERMINOLOGY

A complete project glossary is provided at the following page: http://www.egi.eu/results/glossary/.

VII. PROJECT SUMMARY

To support science and innovation, a lasting operational model for e-Science is needed − both for coordinating the infrastructure and for delivering integrated services that cross national borders.

The EGI-InSPIRE project will support the transition from a project-based system to a sustainable pan-European e-Infrastructure, by supporting 'grids' of high-performance computing (HPC) and high-throughput computing (HTC) resources. EGI-InSPIRE will also be ideally placed to integrate new Distributed Computing Infrastructures (DCIs) such as clouds, supercomputing networks and desktop grids, to benefit user communities within the European Research Area.

EGI-InSPIRE will collect user requirements and provide support for the current and potential new user communities, for example within the ESFRI projects. Additional support will also be given to the current heavy users of the infrastructure, such as high energy physics, computational chemistry and life sciences, as they move their critical services and tools from a centralised support model to one driven by their own individual communities.

The objectives of the project are:

1. The continued operation and expansion of today's production infrastructure by transitioning to a governance model and operational infrastructure that can be increasingly sustained outside of specific project funding.
2. The continued support of researchers within Europe and their international collaborators that are using the current production infrastructure.
3. The support for current heavy users of the infrastructure in earth science, astronomy and astrophysics, fusion, computational chemistry and materials science technology, life sciences and high energy physics as they move to sustainable support models for their own communities.
4. Interfaces that expand access to new user communities including new potential heavy users of the infrastructure from the ESFRI projects.
5. Mechanisms to integrate existing infrastructure providers in Europe and around the world into the production infrastructure, so as to provide transparent access to all authorised users.
6. Establish processes and procedures to allow the integration of new DCI technologies (e.g. clouds, volunteer desktop grids) and heterogeneous resources (e.g. HTC and HPC) into a seamless production infrastructure as they mature and demonstrate value to the EGI community.

The EGI community is a federation of independent national and community resource providers, whose resources support specific research communities and international collaborators both within Europe and worldwide. EGI.eu, coordinator of EGI-InSPIRE, brings together partner institutions established within the community to provide a set of essential human and technical services that enable secure integrated access to distributed resources on behalf of the community.

The production infrastructure supports Virtual Research Communities (VRCs) − structured international user communities − that are grouped into specific research domains. VRCs are formally represented within EGI at both a technical and strategic level.

VIII. EXECUTIVE SUMMARY

This report provides a first snapshot of the status and planning of the services and tools developed and supported for the needs of the Heavy User Communities (HUCs): High Energy Physics, Life Sciences, Astronomy and Astrophysics, Earth Sciences. The choice of these specific communities as HUC's was done when preparing the EGI-InSPIRE Proposal, with the collaboration of the representatives of the communities. They have provided information both on the software that is today specific for each one of them and for the software which is already of interest for more than a single community.

This last category consists of the Dashboards, the GANGA and Diane applications, the HYDRA and GrelC services, the Kepler, Gridway, SOMA2 workflow schedulers, and the enabling of MPI applications, which receives important contributions also by the Computational Chemistry HUC.

This report is a checkpoint for the goals of WP6, recalled briefly in the points below

- To transition the services and tools from the communities that have already adopted DCIs, to where their services are part of the general service infrastructure provided through EGI or are sustained by other means – either through their own community or through external software providers (e.g. middleware projects such as EMI).
- To use the experiences obtained by these early adopting communities in integrating new data sources, tools and services to improve the experience for all user communities.
- To ensure that all the user communities supported by EGI should experience no disruption as they move from their current e-Infrastructure provider.

The initial planning of the services and tools, sketched in this report, is still mostly oriented to the community who is the main user, and often originator, of the tool.

The interaction between the different Heavy User Communities and between them and the generality of the EGI users has started and is becoming more organized; the next step of the software roadmap will be able to more fully reflect the shaping of these services and tools for more general use within WP6 and also outside it.

TABLE OF CONTENTS

1. INTRODUCTION

This document gives a first overview of the Roadmap for the development and deployment in the reference user communities of the software included in EGI-InSPIRE WP6. The Heavy User Communities who are part of the project are the primary target of the document, which is intended to give them information on the features available now and in the future, and offer the opportunity to interact with the planned developments so that they can best fit their needs.

However the document is open to other EGI-InSPIRE users and potential users, who can be interested in adopting parts of the software for their uses and in suggesting developments to this effect.

This document is a first step in the elaboration of a full roadmap (the HUC Software Roadmap is a Milestone due each year in SA3) as the material included in it is the result of the effort of the developers and of the single communities, while the interaction between the developers and between the communities has still to be strengthened, also on the basis of the information included here.

This goal of providing the first step in the development of a fully fledged roadmap is reflected in the level of elaboration of the material included in this document which, albeit rather inhomogeneous and including in some cases a planning with detailed milestones and in other cases just a few indications for the future planning, fits the purposes of this document.

2. ROADMAP FOR THE SA3 SHARED SOFTWARE SERVICES AND TOOLS

2.1. Dashboards (1)

The Worldwide LHC Computing Grid (WLCG) provides data storage and computational resources for the High Energy Physics (HEP) community. The Experiment Dashboard system provides monitoring of the WLCG infrastructure from the perspective of the LHC Virtual Organizations (VOs) and covers the complete range of the LHC computing activities: data distribution, job processing and site commissioning.

It works transparently across various middleware flavours. The Experiment Dashboard services are widely used by the LHC virtual organizations. Some of the Experiment Dashboard applications are generic and can be used outside the scope of the HEP community.

The main directions of work are focused on the applications which are generic and can be shared by multiple VOs. Among those applications are job monitoring, site status board and site usability.

At the time of writing this deliverable the list of actions and roadmap had been identified:

1) July 2010
    a. Develop a prototype of the job processing historical application. The job processing historical application shows job processing monitoring parameters as a function of time
    b. Test and deploy, in collaboration with the ATLAS VO, the Site Status Dashboard, in its specific version for ATLAS computing operations
2) August 2010
    a. Deploy new version of the job processing historical application to the validation server for the CMS VO
    b. Enable new functionality - resource utilization view for the CMS VO

3) September 2010: Enable import of the ATLAS PANDA work load management system into Dashboard schema
4) October 2010: Deploy prototype of the generic job monitoring for ATLAS VO, including Interactive view, Historical view and Task monitoring
5) November 2010: Test of the prototype, in collaboration with the ATLAS VO End of 2010: Deploy into production the generic job monitoring for ATLAS VO

During 2011 the new version of visualization for data transfer will be developed and deployed in production for Atlas Data Management monitoring.

Service Availability Monitor (SAM) is currently migrating to the new implementation. Dashboard Site Usability application based on the results of SAM tests is being redesigned to comply with the new SAM implementation. The new version of Dashboard Site Usability application should be deployed in production by the middle of 2011.

Development of other Dashboard applications will be driven by the requests of the user community.

2.2. Applications

2.2.1. GANGA (2)

GANGA is an easy-to-use frontend for job definition and management, implemented in Python. GANGA is a mature product with 500 unique end-users every month. The project planning follows a yearly cycle and is discussed at the GANGA Developers Meetings. The detailed technical work plan will be devised during the next GANGA Developers Meeting, however the main work areas i foreseen for further work are:

1. Consolidation and support of existing infrastructure of the GANGA project through an integration of the existing documentation.
2. Support and further develop the persistency solution for GANGA to be able to serve all user communities in a scalable way.
3. Extend GANGA with a plug-in-structure for data input and output in the same way as it currently has for access to different computational resources.
4. Integration with monitoring services, notably Dashboards, by reusing common software elements in the presentation layers which will increase the coherence of the look and feel of the tools for the end-users. This should allow for improved productivity for end-users and more natural integration of monitoring and task processing services at the application level.

2.2.2. DIANE(3)

DIANE is a lightweight job execution control framework for parallel scientific applications. DIANE improves the reliability and efficiency of job execution by providing automatic load balancing, fine-grained scheduling and failure recovery DIANE is an established software tool, used by power-users in several communities to implement complex and demanding task processing use-cases. The planning process involves the interested power-users and is typically discussed during EGEE/EGI events. The mains work areas will include:

1. Integration of application-level monitoring information with existing GANGA/DIANE Dashboard.
2. Improvement of failure detection algorithms to make it easier for users to specify correct values and avoid deadlocks.
3. Consolidation and support of existing infrastructure of the DIANE project through an integration of the existing documentation.
4. Maintenance of the release infrastructure and procedures following the internal migration to new development services (e.g. SVN).

2.3. Services

2.3.1. HYDRA

Hydra is a file encryption/decryption tool developed as part of the gLite middleware. Hydra is a special secure metadata catalogue designed to hold encryption keys. The Hydra functionality is accessible in the regular gLite User Interface and Worker Nodes through command line tools. Hydra may be deployed as a single key store or as a distributed key store, implementing the Shamir's secret key sharing algorithm, for improved availability and higher robustness against attacks.(4)

A Hydra catalogue will be deployed within the first year of the EGI-InSPIRE project as a service for the life sciences community. Hydra is mature software and the only planned developments for this service are bug fixes and updates to preserve the hydra functionality in the future versions of the gLite middleware. Bug reports and problems encountered with the service can be reported through the regular GGUS global gLite support portal.(5)

The roadmap for this service provision is (in EGI-InSPIRE Months):

- January 2011 (M9): installation of a test Hydra catalogue and validation of the functionality delivered, including:
    - Encryption key registration and removal.
    - Encryption key access control.
    - File encryption / decryption from User Interfaces and Worker Nodes.
- April 2011 (M12): delivery of a distributed Hydra catalogue for production use within the Life Sciences area
- Unless specific intervention is required due to changes in the gLite middleware Data Management System, a functionality check is then planed every semester from month 15:
    - July 2011 (M15): in preparation for D6.4, revalidation of the functionality delivered with performance and scale tests (encryption overhead measurements, number of keys that can be stored and number of accesses that the Hydra catalogue can handle).
    - January 2012 (M21): revalidation of the functionality delivered.
    - July 2012 (M27): in preparation for D6.7, revalidation of the functionality delivered with performance and scale tests.
    - January 2013 (M33): revalidation of the functionality delivered.

2.3.2. GRelC(6)

The GRelC service is a grid database management service aiming at providing access and management functionalities related to relational and non-relational databases in a grid environment.

It offers a unique front-end to store, manage and retrieve data, hiding several physical details such as database location, database name, user login, password, etc. The aim of this data grid service is to efficiently, securely and transparently manage databases on the grid, across virtual organizations.

The GRelC service provides dynamic binding to different versions of MySQL, PostgreSQLand SQLite for the relational data model as well as to eXists and Xindice for the hierarchical one. A command line interface and a web based environment (the GRelC Portal) are available to the end users to interact with the GRelC service. Support is also available to Java and C developers (in terms of JDK/SDK, documentation and examples) through the GRelC web site (www.grelc.unile.it).

During the project, the GRelC system (the network of GRelC services deployed within EGI) will be enhanced to support the EGI communities with a new set of functionalities. These will be accessed by the end-users through the GRelC Portal, a seamless, ubiquitous and web-based environment for the management of geographically spread and heterogeneous grid data sources.

Support in terms of management to the user communities will be provided through the Dash-G interface. It will provide a set of views to monitor, control, provide access to and interact with the GRelC services and the associated data sources. Such a framework will be available through the GRelC Portal by means of a new set of web pages exploiting the dashboard approach. The Dash-G will provide several views (including charts, reports, tables) about the GRelC deployment, the status of the services, the list of available grid-databases, the supported VOs, related keywords/tags, user's comments, etc. A small set of sensors will be able to collect the needed information, storing them into a relational database (named *system catalogue*). The dashboard will be designed and implemented taking into account the Web2.0 paradigm to really create “communities” around the available data sources. Accordingly, users will be also able to rate resources, post comments, etc. Yet, an important feature will be the capability to “export” into other web applications the views provided by the Dash-G just with few HTML lines of code.

The Dash-G will basically represent an important “database-oriented” tool providing effective views (like widgets). Additional management/control functionalities will be added too, like for instance some widgets related to the ECA (Event-Condition-Action) rules mechanisms.

The roadmap for the GRelC service foresees the creation of the EGI Database of Databases, a registry accessible through the Dash-G that will contain all of the information about the grid-databases available in the GRelC system, the associated VOs, a description, some keywords and other useful metadata.

Through the registry, users will be able to:

- *Submit a query* asking for specific databases, filtering by VO, keywords, domain, etc. This will help people working in a specific domain to quickly find out available resources, identify key people working on specific subjects, easily contact them to establish collaborations, etc.
- *join a specific grid-database*, submitting via web the join-request to the administrator of the grid-database;
- *add comments and rate data sources,* following the “community-oriented” approach of Web2.0 applications like Twitter, Facebook, etc.

The registry will complement the functionalities provided by the EGI Application Database and will represent a distributed and multi-VO system.

In few words, the Dash-G will provide a *cross-VO grid-database system* giving the user community both a unified view about the available grid-databases and several tools to interact with them.

During the project a set of user-community oriented use cases, involving grid-databases, will be defined starting from user needs and requirements.

Support will be provided to the interested VOs to “gridify” their data sources and to use these experiences to drive the design and implementation of new functionalities provided through the GRelC service.

A questionnaire will be also prepared and sent to the user communities to gather relevant database-oriented requirements and needs. This kind of support will help in (i) starting discussion and interaction with the user communities, (ii) classifying needs and requirements, (iii) planning the support strategy/roadmap prioritizing the coming requests.

Concerning the Earth Science community, the Climate-G testbed will represent one of the test cases related to the GRelC service. In this testbed the GRelC services will be exploited to manage metadata information distributed across several countries and stored into both relational and XML databases. The Climate-G testbed is a climate-change oriented research effort started in 2008 in the context of the EGEE Earth Science Cluster Community and presented in 2009 during the first year of the EGEE-III review (invited demo). New requirements in terms of metadata management coming from this community (e.g. harvesting functionalities) will be collected and implemented. Support will be also provided through the Climate-G portal, the scientific gateway of the testbed that provides search & discovery functionalities, metadata web pages, lists of experiments, datasets, projects, access to data, etc.

Additional use cases related to other communities (e.g. Life Science) will be defined, jointly with them during the project.

The roadmap of the GRelC service foresees in the first year of the project a strong support to the user communities, related to the porting in grid of new/existing relational and XML databases. The questionnaire mentioned before, will strongly help in this phase identifying key resources and users. Moreover, support will be also provided by means of new functionalities available via the GRelC Portal (and Command Line Interface too) to ease the porting in grid of existing data sources. The Dash-G will represent another important milestone of the first year (release 1.0). The first version of the Dash-G will allow users to access to basic views about the available GRelC services deployed in the EGI infrastructure. The registry implementation will be finalized in the second year of the project (release 2.0) while during the third year the final release (2.2) will basically offer few new functionalities, but a higher level of robustness. Comments, feedback and requirements coming from the user communities will be taken into account (and will be essential) to improve the software from a functional and non-functional points of view. This work will obviously impact on the GRelC service and on the GRelC Portal (for instance regarding the “publishing” functionality) leading to new releases of these software, compliant with the Dash-G functionalities.

Along with *cross-community support* related to common needs, additional effort will be dedicated to implement specific community-oriented use cases. This activity will be performed during the three years of the SA3 and will be carried out through F2F meetings, via emails, phone conferences, by means of new and updated documentation on the GRelC website, new tutorials, etc. In particular for the Earth Science community, in the first year of the project a new version of the Climate-G Portal will be released with map-enabled search and discovery of climate change datasets related to projects, models and scenarios. Support will be also provided in terms of physical resources, by deploying a new GRelC service instance at the SPACI site for use cases needing hosting and complete management of their data sources.

2.4. Workflow and Schedulers

2.4.1. Kepler and Gridway

Kepler(7) is a software application for the analysis and modeling of scientific data.

Kepler allows scientists to create their own executable scientific workflows by simply dragging and dropping components onto a workflow creation area and connecting the components to construct a specific data flow,
GridWay(8) is a Metascheduler that automatically performs all the **submission steps** and also provides the **runtime mechanisms** needed for dynamically adapting the application execution

The developments performed here are primarily aimed at the support of the Fusion Community, however could be indeed useful for other communities whose applications have a similar structure.

Fusion, as a HUC (Heavy User Community) has to perform developments on Kepler workflow engine and on Gridway metascheduler. These two tools have been previously extensively used and will be at work in the future, so they have become de fact standards.

The main goal of the activities described in the present roadmap is to build workflows among applications that run both on the grid and on HPCs (High Performance Computers). These workflows are all scientifically relevant and needed for several studies.

A list of possible workflows among applications is shown here:

- VMEC (grid – HPC) + **DKES** (grid – HPC)
- VMEC (grid – HPC) + ISDEP (grid – HPC)
- **FAFNER** (grid – HPC) + ISDEP (grid – HPC)
- ASTRA (HPC) + **TRUBA** (grid – HPC)
- ASTRA (HPC) + **TRUBA** (grid – HPC) + FAFNER (grid – HPC)
- EUTERPE (HPC) + ISDEP (grid – HPC)
- **GEM** (grid – HPC) + ISDEP (grid – HPC)
- ASTRA (HPC) + **GEM** (grid – HPC)
- ASTRA (HPC) + **GEM** (grid – HPC) + **TRUBA** (grid – HPC)
- …

The applications in bold are those that can run using the Gridway metascheduler. The ones that run on HPCs have this name in brackets, while the ones that run both on the grid and on HPCs are also marked in brackets. Furthermore, some of these applications (for example VMEC and ASTRA) are already using Kepler.

The architectures of these workflows are very different. There are the simple linear ones in which an application just run after reading the output of other application, which is the case of numbers 1, 2, 3, 6, and 7, and more complex workflows which imply an application generating input for another application and waiting some input from the later one without finishing its execution. All the cases in which ASTRA is present follow this second scheme. ASTRA calculates the time evolution of the plasma and needs to call other applications that run both on the grid and on HPCs. Examples of this case are numbers 4, 5, 8 and 9. Hence, by being able of running some of these workflows, a wide range of possibilities is covered. These possibilities are not only useful for the fusion community, but also for any other community.

The roadmap for the project will be the following:

1) To build and exploit scientifically the linear workflows.
   This activity implies the use of Kepler in its standard way, by designing the workflow and preparing and testing the predefined template workflow for each case.
2) To build and exploit scientifically workflows with a more complex structure.
   Again Kepler will be used in, but now the structure is much more complex. This will require effort in designing, construction, development and testing of complex use cases.
3) Integration of Kepler and Gridway.
   As stated, there are several applications that run using Gridway, since there are specific codes that cannot bear the missing of jobs on the grid and this metascheduler is able to relaunch jobs that are lost. It would be desirable to have Kepler integrated in Gridway in order that the applications that run using this metascheduler can be incorporated in the workflows without strong changes.
4) To enable Kepler to run applications that produce results in very disparate time scales. So in the workflow it will be necessary to keep an application in standby until another one has finished its task and sent the results. A communication mechanism is required to indicate

when an application requires input from another application and in order to restart the execution in the same point as the application was.

2.4.2. SOMA2(9)

SOMA2 is a versatile modelling environment for computational drug discovery and molecular modelling. SOMA2 is operated through WWW-browser. The SOMA2 environment offers a full scale modelling environment from inputting molecular data to visualisation and analysis of the results.

A brief description of the activities for SOMA2 in EGI-Inspire SA3 (TSA3.2.4) till end August 2010 is given here, as introduction to the roadmap:

- Targeted development to create 1st version of grid enabled SOMA2
  - processing of user's personal X509 certificate, examine existing / possibility to upload one
  - defining grid entity in SOMA2 capsule program XML-descriptions
  - SOMA2 service authorization scheme extended to take grid entities into account, based on
- middleware specific resource request against user's X509 certificate
  - recognize included grid resource in user's workflow and in such situation, create a proxy certificate upon
- project submission (in workflows, both "local" and grid resources can be used)
  - extend SOMA2 project stop scheme to take grid execution into account
  - extend SOMA2 job execution scheme to make use of job submission with grid middleware

- currently supports Nordugrid Arc middleware, should be easily expanded to support other middleware as well

First milestone in SOMA2 development (http://www.csc.fi/soma) for EGI-Inspire has been to extend existing SOMA2 gateway to interface with job submission to grid infrastructure. During the first four project months design and targeted development to achieve this has been made. As a result, a support for processing user’s personal X509 certificate has been added in SOMA2. System is also capable of examining existing certificate and if no certificate is found for a user, one can be attached into the system. Also, we are now able to define grid entities in SOMA2 program description scheme. Grid entities are integrated into existing SOMA2 service authorization scheme so that available resources (scientific applications) for a user can be resolved according to user’s certificate information. The system has also been made to detect usage of grid resource in user’s workflow in which case a grid proxy certificate will be created upon computation project submission. For the end user, it is now possible to include both local and grid resources in user’s application workflow. All grid jobs can be cancelled via the SOMA2 interface. In addition, SOMA2 program execution scheme has been extended to make use of grid middleware in actual job submission.

All grid related features have been implemented as optional setting and functionality is based on using user’s personal X509 certificate, which in technical implementation is handled with VOMS-Lite

Perl package (http://search.cpan.org/~mikej/VOMS-Lite-0.11/). Functionality also relies on existing grid middleware on the portal server. Currently this implementation supports Nordugrid Arc middleware but support for other middleware should be straightforward to add.

First milestone with working 1st grid enabled version has been reached in end of August. There are only few minor issues and update of technical documentation remaining.

Due to end of first project year we will release a new version of SOMA2, which will include all newly implemented grid related features. Also, we will build a first pilot service on top of grid enabled SOMA2. This will involve investigating suitable scientific applications to be used in the grid and creating SOMA2 program descriptions for them. Preferably the selected software should be open source licensed to be able to avoid license negotiations at this point. The pilot service will be set up on national level into existing SOMA2 service which CSC currently provides for Finnish academic researchers. During the second project year, we will expand the service so that it will be available for other user communities as well. We will investigate possibilities to set up this service. Also, our goal will be to expand scientific applications selection in the SOMA2 service, and integrate application services from different grid entities into SOMA2. This should be set as an important milestone because from the end users point of view, this would make using scientific applications in different grids very easy and transparent. During the third project year, we will continue to maintain and operate the SOMA2 service and seek possible scientific applications to be added as part of the service. During all project years, we will support using SOMA2 service. Also development of SOMA2 gateway will continue according to feature roadmap including possible feature requests from the user community, bug fixes and other enhancements.

2.5. MPI

The MPI sub-task shall focus on a number of goals over a 36 month period. This sub-task will produce numerous MPI workbenches of increasing complexity with specific high impact on the Computational Chemistry, Fusion and Astronomy and Astrophysics (A&A)communities. These products will also have impact on other User Communities. It will focus on ensuring that the user communities and site administrators benefit from several rudimentary improvements in well defined methodologies and documentation. Many of these objectives are iterative, often requiring updates or fine-tuning. Other objectives, such as EGI User Forum participation, will be repeated at regular intervals. The core sub-task objectives are:

- Improved end-user documentation, addressing MPI application development and job submission in ARC, gLite and UNICORE,
- Quality controlled MPI site deployment documentation,
- Outreach and dissemination at major EGI events and workshops,
- User community, NGI and site engagement, gathering direct input,
- Participation in selected standardisation bodies.

UNIPG, CSIC and INAF partners have a great wealth of experience in designing, producing and deploying MPI applications under gLite. These range from relatively simple codes, to large scale production workflows using multiple externally provided (and widely used) MPI-enabled libraries. TCD and CSIC will engage with the ARC and UNICORE communities, and will produce high-level documentation for MPI application development and submission under these middleware. In project year one, the first MPI "cookbook" will be produced addressing MPI application development and MPI job submission. This shall be reviewed and updated during project years two and three, as expected middleware changes and new features, such as generic "Parallel" application support, OpenMP support, and GPU application support are introduced.

As part of User Community engagement effort, the MPI team will regularly survey Virtual Organisations, Users and Site administrators for critical feedback. This will also act as a means to gather information about current deficits and future requirements. There will be at a minimum two surveys over the 36 months, one expected to be prepared in project month 16, and a second survey in project month 30.

3. ROADMAP FOR THE COMMUNITY SPECIFIC SOFTWARE

3.1. Services for the HEP Virtual Research Community

Services for High Energy Physics are covered in detail in MS603. Described below is only the software roadmap for the WLCG Persistency Framework, the main software package developed specifically for HEP.

Persistency Framework

The *Persistency Framework* is one of the projects set up within the Application Area (AA) of the LHC Computing Grid (LCG) to provide common software solutions to the LHC experiments at CERN, as originally defined in the LCG architecture blueprint. It consists of three software packages (POOL, CORAL and COOL) that are used by three of the LHC experiments (ATLAS, CMS, LHCb) for storing and accessing several different types of scientific data. CORAL is a generic abstraction layer with an SQL-free API for accessing data stored using relational database technologies. POOL is a generic hybrid technology store for C++ objects and object collections, using a mixture of streaming and relational technologies. COOL provides specific software components to handle the time variation and versioning of the conditions data of the LHC experiments. All packages are written in C++, but python bindings are also provided for CORAL and COOL. Further details about the three packages are described in the EGI-Inspire document covering HEP Services.

The Persistency software has been developed over several years (since 2003 for POOL, since 2004 for CORAL and COOL) through the well established collaboration of developers from the LHC experiments with a team in the CERN IT department (now partly funded by the EGI Inspire project), which has also ensured the overall project coordination. The collaboration with other LCG AA projects (especially the SPI and ROOT projects) has also been particularly important. Progress has been regularly reported to the LCG AA management. All of CORAL, COOL and POOL have been used in production by the LHC experiments since the first data-taking in 2009. While the software is by now mature in its development cycle, a large development and support effort (approximately 4 FTEs) is still required for user support, service operation and maintenance tasks. The support load is expected to decrease with time, as the issues encountered during the initial phase of LHC data taking are sorted out. The work in progress to consolidate the three packages using a more uniform infrastructure will also be relevant to this goal.

The release process is well established. Regular production releases are prepared whenever one of the LHC experiments demands it, leading to one release per month on average. This is generally motivated either by urgent bug fixes and functionality enhancements in the Persistency software, or by upgrades in the versions of the 'external' dependencies (ROOT, Boost, Python, Oracle...). These external versions, which are different from those installed on a predefined O/S version, vary quite frequently because they must match those chosen by the three experiments for their frameworks (Gaudi for LHCb, Athena for ATLAS and CMSSW for CMS), into which the Persistency packages are linked to build data-processing client applications. The software is supported on many production platforms on Linux, Mac OSX and Windows, using several compilers on each O/S (e.g. gcc3.4 and gcc4.3 on Linux SLC4). To improve the quality of the software and speed up the early adoption of new external versions, automatic builds and tests of CORAL, COOL and POOL are performed every night on all production platforms and a few development platforms to test new compilers (such as gcc4.5 or llvm).

Service operation incidents and user support requests normally result in bug fixes in the Persistency code, but often end up in the need for a more global analysis involving other software packages (such as Oracle, the Grid middleware or ROOT). In particular, while the Persistency software only provides client software components (with one notable exception, the CORAL server), understanding service operation issues almost always require a detailed troubleshooting on the server side, typically on the Oracle clusters where the LHC experiment data has been stored using CORAL.

The development of new functionalities, motivated by explicit experiment requests and/or to overcome specific limitations observed during service operation, is also not over. Recent examples include the addition of one new use case to the COOL relational, several enhancements in the POOL collection packages, or the implementation of update functionalities with proxy certificate authentication in the CORAL server component. Some R&D work is also in progress to evaluate new technologies relevant to data access optimization.

3.2. Services for the Life Science Virtual Research Community

The Life Science community invests effort in setting up a sustainable operation model for the Virtual Research Community (VRC). Indeed, the fragmentation of the former EGEE infrastructure and management into National Grid Initiatives calls for a new decentralized model, that serves the community well while remaining compatible with the overall European Grid managerial structure.

At the Life Science VRC meeting organized in Paris in conjunction with the HealthGrid 2010 conference on June 28, a broad consortium representative of the community decided to push forward the emergence of an international VRC implemented through a large-scale pan-European Virtual Organization to foster international collaborations and facilitate grid adoption.

This operational model requires defining:

1. Governance policies;
2. Secure sustainable funding; and
3. Design technical tools for daily operations.

The technical tools developed and installed to manage the VO are part of the software roadmap described in this document. The exact tools and functionality required depend on the precise governance model adopted and they are still under discussion with all stakeholders of the Life Sciences VRC. At the time of writing this deliverable, here is the list of action and roadmap identified:

- July 2010 (M3): define first proposal and distribute it to the community stakeholders
- August 2010 (M4): collect feedback from the community
- August 2010 (M4): deployment of the LS VRC wiki
- August 2010 (M4): design of the LS VRC user database
- September 2010 (M5): analysis of the “VO admin tool” capability
- October 2010 (M6): implementation of the LS VRC user database and its web front-end; link with the applications database.
- October 2010 (M6): implementation of the file access control to the VO LFC (notion of user home directory)
- November 2010 (M7): first release of the tools supporting VO members management
- November 2010 (M7): automatic updates of the VO mailing lists from the DB and tools
- December 2010 (M8): implementation of the group (projects, national…) management procedures
- June 2011 (M14): second release of the VO tools including groups management, preparation for MS611

- December 2011 (M20): third release of the VO tools including updates based on usage and evolutions
- June 2012 (M26): fourth and final release of the VO tools

3.3. Services for the A&A Virtual Research Community

The main areas of interest for the A&A community are related to the VisIVO tools, the use of MPI, and the Databases. Each of this area is included in a specific A&A sub-task

3.3.1. VisIVO & Visualization Tools

This subtask deals with VisIVO as well as with other potential data exploration tools. VisIVO is a suite of software tools aimed at creating customized views of 3D renderings from many types of datasets. Their peculiar characteristic is that there is no limit for what concerns the size of input tables containing data to be processed, thus they are able to support very large scale datasets (tens of Terabytes and more). VisIVO can be easily used to visualize and monitor data simultaneously, as they are produced by the jobs running on the grid; data under production can also be used to create movies.

The VisIVO server has got a good level of maturity and reliability, and a dedicated network of portals allow the user to access and make use of VisIVO directly via website (http://visivoweb.oact.inaf.it). The server is currently able to run both on HPC platforms as well as on Windows and MAC operating systems. Moreover it is Grid-enabled.

Recently the VisIVO server has been successfully ported on the EDGeS infrastructure as well.

VisIVO depends on several software packages but this does not represent a problem as all this software is open source. VisIVO has been thought and designed focusing on astrophysical datasets but now it is a general purpose data exploration tool; it can be considered a grid service enabling users to generate data previews and movies, not only for astrophysics but for in many other scientific domains.

VisIVO Server consists of three core components: VisIVO Importer, VisIVO Filter and VisIVO Viewer respectively. To create customized views of 3D renderings from astrophysical data tables, a two-stage process is necessary. First, VisIVO Importer is used to convert user datasets into VisIVO Binary Tables ((VBTs). Then, VisIVO Viewer is invoked to display customized views of 3D renderings. VisIVO Filters are collections of data processing modules able to explore datasets enhancing and highlighting their hidden properties. Filters support a range of operations such as: scalar distribution, mathematical operations, selection of regions, decimations and so on.

3.3.1.1. Parallelization with MPI

Depending on the structure and size of datasets, the Importer and Filters components can take several hours of CPU to create customized views, and the production of movies could last several days. For this reason the MPI parallelization of VisIVO plays a fundamental role. The concept of conditional compilation represents the basis for this activity; it will be possible to run the tool both on serial nodes and also in parallel or by using HPC oriented Grid nodes.

3.3.1.2. Development with CUDA

CUDA (Compute Unified Device Architecture) is the computing engine in NVIDIA GPUs (Graphics Processing Units) that is accessible to software developers through variants of industry standard programming languages. Because VisIVO is developed in C++, the environment of CUDA can be effectively used to develop some heavy modules of VisIVO Filters and Importers. This activity will enable the use of GPUs whenever they are available on worker nodes. The use of Viewer (based on VTK) in conjunction with CUDA will be evaluated as well.

3.3.1.3. Integration with the Grid Catalogue

VisIVO I/O operations are mainly based on the local file system; for this reason an I/O system, based on existing tools, will be developed allowing VisIVO to directly read/write data in the Grid catalogue. The outcome of this activity will be a tighter integration of VisIVO with the Grid storage system.

3.3.1.4. Usage of Portals for VisIVO on the Grid

Network-based portals for VisIVO are currently available in INAF and located at the Catania and Trieste Astronomical Observatories; a portal is also located at the University of Portsmouth (UK). Other portals will be installed soon at the Westminster University (London) and at the CINECA supercomputing center. One of the targets to evaluate is also the possibility to access and use VisIVO through these portals or other portals (i.e. Genius) allowing user to transparently use it on the Grid, everywhere it is available.

3.3.1.5. Milestones

Milestone	M	Title	Description
MSA3.5.2.1	11	VisIVO-MPI prototype	Prototypes for Importers and Filters implemented
MSA3.5.2.2	14	VisIVO-MPI final version	Final version of VisIVO for MPI in Grid deployed
MSA3.5.2.3	22	Prototype of VisIVO for CUDA	First prototype of VisIVO for CUDA generated through GPUs on the worker nodes
MSA3.5.2.4	25	Final version of VisIVO for CUDA.	Final version of VisIVO for CUDA generated through GPUs on the worker nodes
MSA3.5.2.5	31	I/O in Grid catalogue	Final version of VisIVO allowing I/O operation directly on the Grid Catalogue
MSA3.5.2.6	36	Portal for VisIVO	Final version of a portal that allows the usage of VisIVO on the Grid.

3.3.2. Grid and Supercomputing

People involved in TSA3.5 of EGI-Inspire will continue to operate and collaborate within the project to achieve the goal of a full support to MPI by Grid infrastructures.

One of the main goals is also the accomplishment of a good level of interactivity among different technologies related to supercomputing, i.e. HPC/HTC, Grid and Cloud and test different operational environments that combine all of them (or a part thereof) to identify those that better fit the needs of A&A applications.

3.3.2.1. EGI-Inspire Working Groups

After having established/agreed the level of collaboration, INAF will participate at the EGI-Inspire Working Groups and initiatives dedicated to HPC, HTC and Cloud computing. The main target of this collaboration is to contribute to the development of those grid services that can meet A&A application requirements using HPC and parallel systems.

3.3.2.2. A&A Requirements, Use-Cases and Workflows

The main target of this activity will consist in collecting and identifying the most important requirements of A&A applications (most of them use HPC systems) to make the Grid able to meet them and this is a mandatory precondition for a more efficient penetration of the Grid in Astrophysics.

At the same time this activity aims also at identifying some general and significant A&A use-cases that imply massive runs using HPC on the Grid. Cosmological simulations represent the most important class of A&A applications requiring HPC resources. Several steps have to be taken into account in this case: a) the preparation of the initial dataset whose size may be of hundreds of Gigabytes; b) the data production phase, generally performed through parallel code whose execution involves hundreds of CPU/cores; c) the post-analysis phase: this is generally carried out through ad-hoc code depending on the amount of data to be checked; d) the database and all aspects related to data persistency. The outcome of this activity will therefore identify the most significant applications (or classes thereof) as well as a typical workflow encompassing all the above steps to be executed on the grid. This task also impacts activity 3.1.2 as it could generate requirements of A&A applications.

3.3.2.3. Test/Validation Plan

The test/validation plan will be sketched out by taking into account the global architecture offered by the Grid. The test will be run for different types of use-case: small, medium and large size. Some indicators will be determined to monitor the level of usage, the efficiency of the entire system and the level of integration among HPC runs and Grid components.

3.3.2.4. Milestones

Milestone	M	Title	Description
MSA3.5.3.1	12	Requirements, Use-Cases and Workflows defined	Design of a number of Use-Cases and a complex workflow encompassing the complete set of use-cases
MSA3.5.3.2	15	Validation Plan in place	Definition of the indicators allowing to gauge the usage of HPC resources in Grid
MSA3.5.3.3	36	Global Test and results analysis completed	Complete test involving all workflows: medium and large size

3.3.3. Databases

Since early Grid projects, interactions and interoperability aspects between Grid infrastructures and databases are one of the hot topics for A&A applications; starting from this consideration the participation to EGI-Inspire is another excellent chance to go ahead in this direction.

Several options will be evaluated ranging from further developments of the G-DSE prototype to custom specializations of tools already in production (e.g. AMGA) possibly expanded and tailored according to the needs of A&A applications.

3.3.3.1. Status of Database-related Tools/Services

The resumption of the development activity on G-DSE prototype is only one of the options that will be considered for what concerns tools and services able to ensure the requested interoperability between Grid infrastructures and Databases. At the time the G-DSE was conceived and designed, no tools able to meet the requirements of the A&A applications were in place. Unfortunately the G-DSE prototype is frozen for a long time so it is advisable to check now the evolution of other tools to evaluate possible adaptations to make them fully compatible with A&A application; such adaptations should be done at a reasonable effort and human resources.

The main goal of this activity aims at evaluating the current state of the art; the status of the G-DSE prototype will be analyzed; other tools currently in production (OGSA-DAI, Spitfire, GRelC, AMGA) will be analyzed to understand the effort requested to adapt them to A&A applications.

3.3.3.2. Customization of Identified Tools

Once the tools have been chosen some development on them could (and probably will) be necessary to add the whole set of functionalities requested by A&A applications.

At this stage it is still not possible to identify the tools that better suit A&A applications and what kind of development they require.

3.3.3.3. User-dedicated Front-Ends and APIs

To be actually useful, tools and services able to make Grid infrastructures and databases mutually interoperable require to be supported by the most popular Grid portals (P-GRADE and others) and gateways; they have to be easily exploitable to run complex workflows. Moreover good APIs should be in place to be used by developers of other tools and services.

3.3.3.4. Milestones

Milestone	M	Title	Description
MSA3.5.4.1	10	Tools and services evaluated	Tools and services evaluated. Specific activity planned
MSA3.5.4.2	22	Mid-term release	Mid-term release of new/customized tools and services
MSA3.5.4.3	36	Final release	Final release of new/customized tools and services

3.4. Services for the Earth Science Virtual Research Community

The failure of the DCI user community related proposal for Earth Science (ES) has forced a reorientation and downscaling of the community efforts. Still, the community is alive and supported independently by organisations and NGIs. It will put effort mainly into fostering the community and providing essential services around EGI:

- Community coordination

Due to the broadness and size of the ES Grid community the coordination of effort is crucial.

- Contact point for new and existing users and basic first consultancy to help with orientation in the field of DCIs
- Conservation of knowledge about proven practices in the field, e.g. through maintenance of a wiki or document database.
- Management and maintenance of the "catch-all" VO, the common VO for new members of the ES community (user management, resource coordination, contact to providers, coordinated installation of ES specific software on resources of the VO etc.)
- Dissemination activity to make the ES Grid activity trackable and public

As a consequence of the national funding, however, the situation differs between the different NGIs. The support of the ES experts is more or less a best effort service.

Funded effort will be put into surveying options to integrate access to Earth Science data infrastructures, such as specialised catalogues or e.g. other data center services. Cooperation with ESA and the GENESI-DEC project for access to Earth Science Data is already under way (http://www.genesi-dec.eu/). Based on the effort in the EGEE III project, which worked on interoperability with the GENESI-DR infrastructure, it will be evaluated how to make features of the infrastructure available to other users. A MoU is currently in preparation. The next step will be a discussion of possible use-cases and a review of the architecture and technology.

Also, a demand for access to OPeNDAP (Open-source Project for a Network Data Access Protocol) resources has been noted, which will be evaluated independently.

In the next months, the community tries to move forward through the following action points:

- Re-activate the VRC and formulate core goals and alternate organisation structure
- Preparation and submission of proposals to fund the core of the VRC activities, to continue collaboration among the teams
- Collect feedback from the ES Grid community to clarify and document the global and individual situation
- Broaden community and foster community relations across the different ES disciplines

4. CONCLUSIONS

All the Heavy Users Communities included in the WP6 work-package have contributed to this report with information on the status and planning of their services and tools. The communication between the different communities has started; the planning for their software of potentially more general interest, which in some cases is rather well defined while in others is still in an early phase of development, has been exposed to the other communities of heavy users and to the general users communities.

The feedback from the different kind of users, the sites and the software developers will now be an important next step toward the long term planning of these services and tools.

5. REFERENCES

R 1	http://dashboard.cern.ch/
R 2	http://Ganga.web.cern.ch/Ganga/
R 3	http://it-proj-diane.web.cern.ch/it-proj-diane/
R 4	*Encrypted storage of medical data on a grid.* **L. Seitz, J.M. Pierson, and L. Brunie.** 2006, Methods of Information in Medicine, , pp. 44-45.
R 5	www.ggus.org
R 6	http://grelc.unile.it/home.php
R 7	https://kepler-project.org/users/documentation
R 8	http://www.gridway.org/doku.php?id=start
R 9	http://www.csc.fi/english/pages/soma