GISELA

1ST INTERMEDIATE REPORT ON THE JOINT RESEARCH ACTIVITY

EU DELIVERABLE: D6.1

Document Full name	**GISELA-D6.1-v1.3**
Date	**26/08/2011**
Activity	**WP6 / Infrastructure and Applications-oriented Services for User Communities**
Lead Partner	**UFCG**
Document status	**APPROVED**
Classification Attribute	**PU (PUBLIC)**
Document link	http://documents.gisela-grid.eu

Abstract: This document contains an assessment of the work carried out in the context of the Joint Research Activity, Work Package 6 "*Infrastructure & Application-oriented Services for User Communities*" of the GISELA project during the first year of its execution. It describes the main achievements attained and compares these results with the outcome planned for the period in the Description of Work.



Delivery Slip

	Name	Partner/Activity	Date	Signature
From	WP6	CLARA / WP6 - Infrastructure and Applications-oriented Services for User Communities		
Reviewed by	Technical Board			
Approved by	Management Board		26/08/2011	B. Marechal Ph. Gavillet S. Jalife Villalón L. A. Trejo Rodriguez R. Barbera R. Ramos Pollán

Document Log

Issue	Date	Comment	Author
0-1	28/06/2011	First draft (template)	B. Marechal
0-2	06/08/2011	First draft of the contributions	F. Brasileiro, V. Hamar, D. Scardaci, H. Castro
0-3	07/08/2011	First review. Minor corrections and formatting	B. Marechal
1-3	26/08/2011	Final Review and approval	B. Marechal

Document Change Record

Issue	Item	Reason for Change

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1. INTRODUCTION

1.1. PURPOSE OF THE DOCUMENT

This document describes the main achievements of the research and development activities carried out in the context of the Joint Research Activity, Work Package 6 (WP6) of the GISELA project during its first year of execution - from 01/09/2010 (M01) to 31/08/2011 (M12). It also discusses the main difficulties that have been faced and how they have been addressed. Finally, it presents a summary of what is planned for the second year.

For a comprehensive view of the Project and of the GISELA Consortium, the Description of Work (DoW)[1] and the Consortium Agreement (CoA)[2] should be consulted.

1.2. DOCUMENT ORGANISATION

The work developed by the WP6 team is mainly related to the support of the application-oriented and infrastructure-oriented services developed in the context of the EELA-2[3] project, and the research leading to the development of new services that can enhance the experience of both application users and system administrators that use the GISELA infrastructure, and other similar systems. Before presenting the details of the work that has been carried out, we present an Executive Summary in Section 2. In Section 3, we describe the activities performed in the first year of the project execution. Then, in Section 4, the human effort invested to conduct these activities is presented. This is followed, in Section 5, by a discussion of the few deviations from the original plan that we had to make. We explain the reasons for the deviations and what have been done to address them. The plans for the next reporting period are outlined in Section 6. Finally, Section 7 presents some conclusions.

1.3. APPLICATION AREA

The target audience for this document is:

- The members of the Project;
- The European Commission Services;
- The Project Reviewers;
- The External Advisory Committee (EAC);
- The general public.

1.4. DOCUMENT AMENDMENT PROCEDURE

Amendments to this document can be requested by any Project Member to the Project Coordinator, via the Project Office (hlp-gisela@hlpdeveloppement.fr).

[1] http://documents.gisela-grid.eu/record/32?ln=en

[2] Consortium Agreement (CoA) available upon request to the GISELA Project Office (hlp-gisela@hlpdeveloppement.fr)

[3] http://www.eu-eela.eu/

1.5. GLOSSARY

API	Application Programming Interface
BeeFS	Beehive File System
CE	Computing Element
CLARA	Cooperación Latino Americana de Redes
CNRS	Centre National de la Recherche Scientifique
CPPM	Centre de Physique des Particules de Marseille
CoA	Consortium Agreement
CPU	Central Processing Unit
CVC	Customized Virtual Cluster
DCI	Distributed Computing Infrastructure
DIRAC	Distributed Infrastructure with Remote Agent Control
DoW	Description of Work
EAC	External Advisory Committee
GSAF	Grid Storage Access Framework
GUMA	Grid Uniandes Management Application
INFN	Istituto Nazionale di Fisica Nucleare
IP	Internet Protocol
JDL	Job Description Language
JSF	Java Server Faces
JSR	Java Specification Requests
LAN	Local Area Network
LHCb	Large Hadron Collider beauty
MPI	Message Passing Interface
NAS	Network Attached Storage
NFS	Network File System
OPeNDAP	Open-source Project for a Network Data Access Protocol
POSIX	Portable Operating System Interface for Unix
SAGA	Simple API for Grid Applications
SAGE	Storage Accounting for Grid Environments
UFCG	Universidade Federal de Campina Grande
UFRJ	Universidade Federal do Rio de Janeiro

UNAM	Universidad Nacional Autónoma de México
UNIANDES	Universidad de los Andes
UPorto	Universidade do Porto
VM	Virtual Machine
VRC	Virtual Research Community
WP2	Work Package 2: Dissemination and Outreach
WP3	Work Package 3: User Communities Support
WP4	Work Package 4: NGI / LGI Infrastructure Services
WP6	Work Package 6: Infrastructure and Applications-oriented Services for User Communities
XMPP	Extensible Messaging and Presence Protocol

2. EXECUTIVE SUMMARY

The goal of this activity is to develop and support grid services that facilitate the porting and execution of applications in the e-infrastructure. For that, on one side actions have been taken to foster the use of both application-related and infrastructure-related services that were already available, including the customization of these services for the particular needs of specific Virtual Research Communities (VRCs). On the other side, new services were developed, tested, deployed in the GISELA infrastructure and supported. Finally, actions to help in the dissemination of the services have also been conducted. The main achievements after the first year of execution of the project are briefly summarised below.

Two new services were developed, tested, and deployed:

- Beehive File System (BeeFS), a distributed file system that harnesses the free disk space of desktop machines already deployed in the corporation;
- Customised Virtual Cluster (CVC), a mechanism that allows the creation of Desktop Grids, taking advantage of the idle processing capabilities of desktop computers within computer labs, in a non-intrusive manner.

The following scientific papers were published:

- Brasileiro, Francisco; Gaudencio, Matheus; Silva, Rafael; Duarte, Alexandre; Carvalho, Diego; Scardaci, Diego; Ciuffo, Leandro; Mayo, Rafael; Hoeger, Herbert; Stanton, Michael; Ramos, Raul; Barbera, Roberto; Marechal, Bernard; Gavillet, Philippe. Using a Simple Prioritisation Mechanism to Effectively Interoperate Service and Opportunistic Grids in the EELA-2 e-Infrastructure. Journal of Grid Computing, p. 1-17, 2011.
- Brasileiro, Francisco; Andrade, Nazareno; Lopes, Raquel Vigolvino; Sampaio, Lívia Maria Rodrigues. Democratizing Resource-Intensive e-Science Through Peer-to-Peer Grid Computing. In: Xiaoyu Yang; Lizhe Wang; Wei Jie. (Org.). Guide to e-Science: Next Generation Scientific Research and Discovery. London: Springer-Verlag, 2011, p. 53-80.
- Barbera, Roberto; Brasileiro, Francisco; Bruno, R.; Ciuffo, Leandro; Scardaci, Diego. Supporting e-Science Applications on e-Infrastructures: Some Use Cases from Latin America. In: Nikolaos P. Preve. (Org.). Grid Computing. London: Springer-Verlag, 2011, p. 33-55.
- Hamar, Vanessa. DIRAC on GISELA. DIRAC User Community Meeting, 12th – 13th May 2011, Barcelona (Spain).
- Castro, Harold; Rosales, Eduardo; Villamizar, Mario; Jimenez, Artur: UnaGrid. On Demand Opportunistic Desktop Grid. CCGRID 2010, p. 661-666
- Souza, Carla; Lacerda, Ana Clara; Silva, Jonhnny W.; Pereira, Thiago Emmanuel; Soares, Alexandro S.; Brasileiro, Francisco. BeeFS: Um Sistema de Arquivos Distribuído POSIX Barato e Eficiente para Redes Locais (in Portuguese). In: XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2010, Gramado. Anais do XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (Salão de Ferramentas). Porto Alegre, Brasil : Sociedade Brasileira de Computação, 2010. v. 1. p. 1033-1040.

In the second year we will continue to support the services already available in the portfolio, and will work on the development of three new services, namely:

- Efficient execution of data-intensive applications based on the Map-Reduce paradigm;
- Seamless execution of CPU-intensive applications in hybrid e-Infrastructures augmented with the capability of interfacing with cloud computing providers;

- Development of specialised application portals based on the Distributed Infrastructure with Remote Agent Control (DIRAC) Web Portal project.

3. ACHIEVEMENTS

The objective of the work package WP6 is to increase the usability of Distributed Computing Infrastructures (DCIs) in general, and GISELA's infrastructure in particular, by:

- Offering a portfolio of application-oriented and infrastructure-oriented services, including both services already available and new services to be developed during the course of the project;
- Customising these services to attend particular requirements of the VRCs supported by GISELA;
- Providing, together with WP3, comprehensive support to the users of the portfolio of services offered;
- Helping WP2 to disseminate the GISELA's portfolio of services, so that the benefits yield from their use is as broad as possible.

In the following we describe the activities that have been performed to support the portfolio of services already developed, to develop new services for the GISELA toolbox, and to disseminate these services.

3.1. SUPPORT AND CUSTOMISATION OF ALREADY AVAILABLE SERVICES

3.1.1. gLite Application-Oriented Services

During the EELA-2 project, a set of services has been designed and developed to cope with all heterogeneous aspects of the application porting process and to meet the requirements of the applications. These services enhance the functionality of both the gLite and the OurGrid middleware giving the EELA-2 users access to a richer toolbox, reducing the amount of effort on application development, and generally accelerating the adoption of grid technologies.

GISELA adopted these EELA-2 additional services and continue to provide the necessary support so that more application developers can utilise these services.

The following list shows the Application-Oriented services developed in the context of the EELA-2 project:

- The Secure Storage, a service for the gLite middleware providing users with a set of tools to store in a secure way and in an encrypted format confidential data (e.g. medical or financial data) on the grid storage elements. The data stored through provided tools is accessible and readable by authorized users only preventing also the administrators of the storage elements to access the confidential data in a clear format (e.g. insider abuse problem);
- Open-source Project for a Network Data Access Protocol (OPeNDAP) Meta-Finder, a tool for searching geographical information data sets available at the Web through OPeNDAP servers. By filling a form, the user can search for data sets containing some wanted attributes and variables such as "spatial coverage", "temporal coverage", "atmosphere temperature", "precipitation", etc.. To make queries easier, "tag clouds" are shown with some suggestions of tags to classify published data sets;
- The Watchdog, a tool that allow users to watch the status of a running job when it runs on a working node tracing the evolution of produced files;
- The lcg-rec-* tools allowing users to perform recursive grid file operations such as copies and deletion;

- The Grid Storage Access Framework (GSAF), an Object Oriented Framework designed to access and manage Data Grid via APIs. It provides developers with a development tool to write application that adopts Grid as Digital Repository hiding the fragmentation and the complexity of the Data Grid Services. GSAF also provides a basic transaction layer for multi service operation (i.e. synchronization of data operations);

All these services can be downloaded from the project forge repository: https://forge.eu-eela.eu/.

Given that the GISELA infrastructure is using the last gLite version, 3.2, and all its grid sites have been updated, our first activity was to check if the services supported are compliant with gLite 3.2. The Secure Storage was the only service that needed to be update in order to be compliant with gLite 3.2. The new Secure Storage Client compliant with gLite 3.2 (64 bit) has been released and it is available in the forge site: https://forge.eu-eela.eu/frs/?group_id=7.

The GISELA project is continuing to support all the legacy services inherited from EELA-2, except the Grid Storage Access Framework (GSAF), considered obsolete now. GSAF has been replaced by another tool to manage data in gLite middleware, named gLibrary, compliant with gLite 3.2. gLibrary is a system with an easy-to-use Web front-end designed to save and organise multimedia assets on grid-based storage resources. For more information, the reader is invited to visit the web site of the tool available at http://glibrary.ct.infn.it/glibrary_new/index.php.

3.1.2. gLite Infrastructure-Oriented Services

Similarly to the application-oriented services, GISELA adopted and continue to support the infrastructure-oriented services designed and developed in the context of the EELA-2 project.

The infrastructure-oriented services have been developed to provide alternatives to ease the installation, management and use of the e-infrastructure. To this end, the following services have been built:

- A gateway between gLite and OurGrid; OurGrid is a simpler peer-to-peer technology to provide alternative ways to make resources available to the grid infrastructure and to simplify the access to the infrastructure for new users and applications;
- The Storage Accounting for Grid Environments (SAGE), a system to measure the usage of storage resources in a gLite based grid infrastructure whose main task is to collect information from physical devices and make this data available to system administrator to account the storage at higher levels.

GISELA provides an on-demand support for these services. System administrators asking to install these services in their farm have been properly supported.

3.1.3. DIRAC

The DIRAC Workload and Data Management System has been developed for the Large Hadron Collider beauty (LHCb) experiment to support simulation data production, processing and analysis. The system was generalised to support a wide range of applications in various domains.

In the GISELA context, the WP6 team has deployed a replicated installation of DIRAC, following the activities listed below:

- Installation and maintenance of a DIRAC main server http://dirac.eela.if.ufrj.br at the Universidade Federal do Rio de Janeiro (UFRJ), Brazil, including the following services:
  - Configuration Server

- Workload Management
- Data management
- Proxy management
- DIRAC Web Portal

- Installation and maintenance of two secondary servers, one, http://guaivira.lsd.ufcg.edu.br, at the Universidade Federal de Campina Grande (UFCG), Brazil, and the other, http://dirac.up.pt, at the Universidade do Porto (UPorto), Portugal, including the following services:
  - Configuration Servers
  - DIRAC Web Portal

Installation of each new DIRAC release is made in order to ensure than all the features including the new ones are working properly before the update in GISELA servers.

There has also been some work devoted to improve the DIRAC middleware. In particular, the DIRAC Message Passing Interface (MPI) Service and Agent code has been updated. Moreover:

- Command reference for each DIRAC command were created and added to the DIRAC main site, available at http://diracgrid.org/files/docs/diracindex.html ;
- Web portal references were created, and made available at: http://diracgrid.org/files/docs/UserGuide/WebPortalReference ;
- Tutorial wiki pages for beginners and advanced users were created, and made available at: https://github.com/DIRACGrid/DIRAC/wiki/DIRAC-Tutorials ;

The contributions from GISELA to the DIRAC middleware are available through a git repository at https://github.com/hamar/EELADIRAC.

DIRAC has been one of the most useful services provided by GISELA and several applications are currently using this service:

- An application from *Universidad del Valle* in Colombia, myLims.org system (Laboratory Information Management System), allows storing, manipulating and sharing virtually any type of experimental data and to predict molecular properties in a way that users can add features and integrate their own programs to predict more properties from structure, spectra, or any information already available from the system. The processes created by LIMS rely on DIRAC for sending jobs to the grid. A series of steps provide for the control of the runtime environment and the installation of the software needed to perform the tasks generated. After several tests, the results (output files) of recovered tasks are sent back to the LIMS platform to be presented to end-users in an easy and friendly way (http://www.cecalc.ula.ve/gisela/?p=320);
- The *bowtie* program (http://bowtie-bio.sourceforge.net/index.shtml) is another application, used in the *Centro de Ciencias Genómicas* – Universidad Nacional Autónoma de México (UNAM) in Mexico. This application has the following requirements:
  - 22 GB of databases;
  - Additionally 6.97 GB of query files to make alignments used as input files;
  - DIRAC JDLs and scripts were generated, 3290 jobs were executed in approximately one week in GISELA sites (9273.89 CPU hours), total output files size is 41.2 GB;
  - Users are preparing a large experiment to submit to the GISELA grid with the support of DIRAC.

- From *Universidad de Bucaramanga* in Colombia arises the necessity of calculating electrical, structural and photonic crystal alloys of Hg (1-x) Cd (x), proprieties. In order to fullfill this requirement ABINIT (http://www.abinit.org) application with MPICH2 support is being gridified using DIRAC MPI Service.

All the development made until May 2011 were presented at the Second DIRAC User Meeting in Barcelona (http://diracgrid.org/?p=120).

3.1.4. OurGrid

The development of the OurGrid middleware continues to evolve. The latest release of the middleware, version 4.2.4, was made on July 15th, 2011, featuring the following improvements:

- fixed some of the known bugs, including the need for re-ordering XMPP messages due to a known bug on the Openfire Jabber Server;
- the output of the Network of Favours balances is now available on the peer status command;
- the refactoring of the Aggregator component;
- a new script and image for the installation of vserver-based workers.

We have also developed a tool to extract statistics of the utilisation and availability of the system. The service is used in GISELA by WP3 and WP4, and is available at http://charts.ourgrid.org/.

Finally, support has been developed to allow OurGrid jobs to be run over gLite resources in a pilot-job approach. Following this approach, when an OurGrid bag-of-tasks job is submitted using the newly developed tool, an OurGrid system is deployed on-the-fly over the gLite infrastructure. The pilot jobs run OurGrid worker nodes that can be used transparently by the OurGrid broker to run the OurGrid application.

The idea is, for one side, to leverage on the job management capabilities of the OurGrid Broker to facilitate the execution of bag-of-tasks jobs over the gLite infrastructure, an on the other side, to allow OurGrid users to enjoy the more reliable services and less stringent security mechanisms provided by the gLite middleware; in particular, they are able to run jobs with much longer tasks, and allow tasks to be able to communicate with the external world, something that is not allowed by the sandboxing scheme that provides security to OurGrid.

3.2. DEVELOPMENT AND SUPPORT OF NEW SERVICES

The following new services have already been identified as important for the GISELA portfolio:

- Aggregation of spare disk space in multiple desktops in a single Storage Element to support the storage needs of data-intensive applications;
- Efficient execution of data-intensive applications based on the Map-Reduce paradigm;
- Seamless execution of CPU-intensive applications in hybrid e-Infrastructures augmented with the capability of interfacing with cloud computing providers;
- Development of specialised application portals based on the DIRAC Web Portal project;
- Support for the dynamic creation of customized virtual cluster, built on top of commodity and interconnected desktop workstations executing virtual machines through virtualisation technologies.

During the first project-year, we have concentrated our efforts on the development of the Beehive File System that aggregates spare disk space in a POSIX distributed file system that can be used to support

a gLite Storage Element. We have also developed the support for the customised and dynamic creation of virtual clusters. These developments are detailed in the next two subsections.

3.2.1. Beehive File System

The Beehive File System, or BeeFS for short, is a distributed file system that harnesses the free disk space of desktop machines already deployed in the corporation. Like some special-purpose rack-aware file systems, it uses a hybrid architecture that follows a client-server approach for serving metadata and managing file replicas, and a peer-to-peer one for serving data. This characteristic allows BeeFS to aggregate the spare space of desktop disks to build a single logical volume on top of which a general-purpose fully POSIX-compliant file system is implemented.

A BeeFS installation consists of a single *queen-bee* server that handles naming, metadata and replica management operations and a number of *honeycomb* servers that store the actual files. The queen-bee and the honeycombs provide service to many *honeybee* clients as shown in Figure 3. These components are arranged following a hybrid architecture that mixes aspects of client-server and peer-to-peer systems in a fashion that simplifies the design and facilitates the administration of the system.

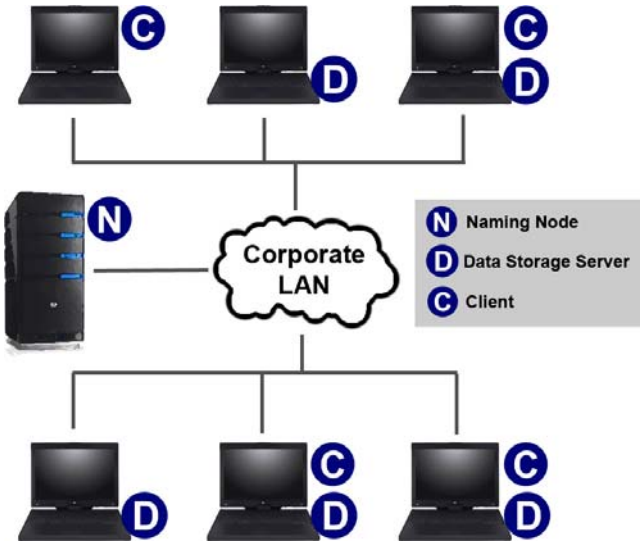

Figure 1: BeeFS Components Overview

The queen-bee server is deployed in a dedicated machine. It is responsible for providing a global file namespace with location-transparent access for files, access control, resource discovery and placement coordination services. On the other hand, it is not involved in data storage at all. Honeybee clients contact it in order to obtain the location of the honeycomb servers that store the files. After that, they fetch/send data directly from/to the appropriate honeycomb server.

The role of the honeycomb servers is to collaboratively store files, providing basic read and write primitives. Honeycomb servers are conceived to be deployed over a set of desktop machines belonging to the corporate LAN.

The honeybee client component normally coexists with a honeycomb server that runs on the same machine. A data placement strategy tries to keep data as close as possible from its users, allowing the scalable growth of the file system.

3.2.1.1. Data Operations

BeeFS does not rely on client side data caching. Caching is only performed at the honeycomb servers. As a result, there will be a single copy of a data file in the system at any time, avoiding the situation when concurrent writes update different copies of the data. This allows a strong consistency model - one-copy-serialisable. In this model, any sequence of data operations has a result that is equivalent to the execution of the same sequence in a local system. The non-existence of caching at the file system level can impose severe performance penalties. This can only be mitigated if a suitable data placement mechanism forces that only a tiny fraction of data access is performed in remote honeycomb servers. In the next section we discuss strategies that can be used to accomplish this goal.

A system workload measurement has demonstrated that 98% of the *stat* system calls issued in general purpose file systems are followed by another *stat* system call. Because of that, metadata caching is performed by the BeeFS at the honeybee client side. In particular, the fetching of the metadata associated to a directory causes the caching of the metadata of all the files that appear in the first level of this directory. Flushing of the metadata cache is done whenever operations that modify the file (eg. *write*, *truncate*, *rmdir* etc.) are called or when a timeout is reached. This metadata caching approach is similar to the one implemented on NFS.

3.2.1.2. Fault Tolerance

For tolerating faults of a honeycomb server, BeeFS employs a non-blocking primary-backup replication model. Following this model depicted in Figure 2, a honeybee client always performs data operations in the primary honeycomb server and eventually the data updates are forwarded to the secondary ones. Hence, the primary copy is always consistent, while the secondary ones are only eventually. There are two main reasons to apply this replication model in the case of file systems:

- The services operations are not delayed by waiting the data commit to secondary honeycomb servers;
- Most data blocks have a short lifetime, so there is a high probability of unnecessary commits.

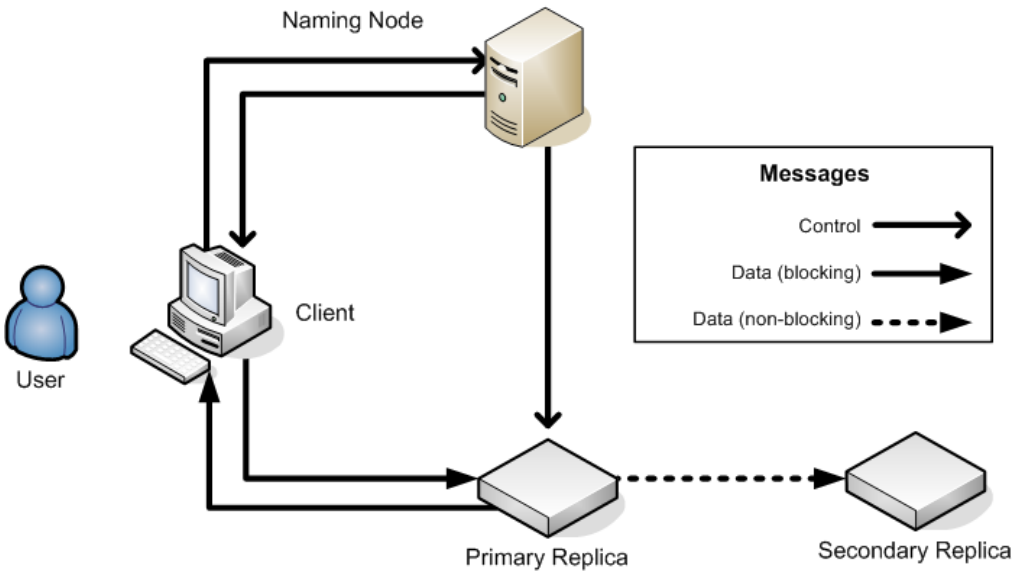

Figure 2: BeeFS Replication Model

The queen-bee server is responsible for orchestrating the update of secondary replicas. It keeps a view of the state of the replica groups associated to each file in the system. This view contains the version of each replica stored in the honeycomb servers. Following the POSIX semantic, the honeybee client sends a close call to the queen-bee server that updates its view about the replicas. In the end of the close call, the queen-bee server schedules a process to propagate the content from the primary replica to the secondary ones. The time that this propagation takes place is defined by a configurable coherence parameter.

The queen-bee server is responsible for monitoring the honeycomb servers. When a honeycomb server failure is detected, the replication group must be reorganised. A new honeycomb server must replace the faulty one on the replication group, in order to keep the desired replication level. If the faulty honeycomb server was a primary one, the replication group can be recomposed only if there is at least one secondary honeycomb server whose content is consistent with that of the faulty primary. The access to staled replicas is always denied, until an updated primary replica is brought back in operation, possibly from a previous consistent backup.

Regarding the queen-bee server, we distinguish two types of faults. For transient faults, BeeFS assumes a crash-recovery failure model. A transient failure of the queen-bee server will impact all operations that have changed file metadata and that had not been made permanent in the queen-bee server disk. Like most file systems (both centralised and distributed), BeeFS periodically flushes data to disk in order to reduce this window of vulnerability. Upon recovery, a consistency check is applied to identify inconsistencies and restart operation after these are healed. The redundant copies of the file attributes (one in the queen-bee server disk and another in the honeycomb server that stores its primary replica) are used to identify inconsistencies.

Differently from other file systems, BeeFS leverages on the redundant storage of file attributes to also recover from permanent queen-bee server failures (eg. due to a crash in the queen-bee server disk). In this case, a fresh instance of the queen-bee server is started in recovery mode. This service receives as input the list of honeycomb servers and contact them in order to rebuild its state.

3.2.1.3. File Attributes

The metadata stored by the queen-bee server includes attributes related to the files stored by the honeycomb servers. They are subdivided in common and extended. Common attributes keep basic information defined in the POSIX standard, such as size, owner, group, access permissions, change and access times, and so on. Extended attributes allow BeeFS to store additional information in the metadata associated to a file. Such information is useful for some applications as well as for BeeFS itself.

BeeFS makes use of extended attributes to keep information about the replication level of each individual file. Also, once modifications over primary data files are not immediately propagated to its replicas, the concept of version was introduced in order to control replica consistency. A file is said to be consistent if all its replicas share the same version number. The time elapsed since a modification is made into a file until the propagation process is triggered, is called time-to-coherence and is also stored as an extended attribute. In addition, a type descriptor is stored to distinguish between primary and secondary copies. For fault tolerance reasons, whenever the extended attributes of a file are changed, this information is sent to the primary replica to be redundantly stored. Note that the other attributes are already redundantly stored by the underlying local file system used by the honeycomb servers.

The POSIX file attribute operations are implemented by the queen-bee server. The amount of storage space required to maintain file attributes is remarkably small, as suggested by a study on file system

contents. Taking into consideration that a typical user stores about 6,000 files and that 100 bytes per file are enough for storing attributes, only 600 Kbytes are required per user. This way, the queen-bee server adopts a policy similar to the Google File System, keeping this information stored in-memory and periodically committing updates to the disk for fault-tolerance reasons.

3.2.1.4. Security

BeeFS scatters files over the corporation desktops. The files stored on these machines must be protected from inside abusers. Both privacy and integrity may be compromised by such malicious users.

One possibility to solve these problems is by using cryptography. BeeFS avoids this technique due to performance reasons. Instead, it leverages on the access control mechanism (eg. Access Control Lists, etc) of the underlining local file system that is used by the honeycomb server. This way, privacy and integrity can be easily guaranteed by assigning a permission mode to every file stored on a honeycomb server, in such a way that only the BeeFS components can access them.

3.2.1.5. Evaluation

We have deployed BeeFS on 50 machines of the Distributed Systems Lab at the UFCG, in Brazil. Our first experiments using standard benchmarks with this deployment show that, in average, BeeFS outperforms NFS execution time in 74% for write operations and 30% for read operations in the best case. In the worst case, BeeFS results in a 56% improvement in write operations and 20% for read operations when compared with NFS. Moreover, in all cases, BeeFS improves in at least 30% all metadata operations.

Moreover, BeeFS is not only more efficient than the state-of-practice that uses a dedicated server approach (eg. NFS), but also cheaper and naturally scalable. Reduced ownership cost is achieved by increasing the utilisation of desktop disks, while sudden increases in the demand for storage, normally caused by the arrival on new users, is usually matched by the extra disk space that is available in the machines allocated to the new users.

3.2.2. Virtual Cluster

In the first year of the project we have developed UnaGrid, a Desktop Grid that takes advantage of the idle processing capabilities of desktop computers within computer labs, in a non-intrusive manner, through the execution of Customized Virtual Clusters (CVCs). A CVC is a set of commodity and interconnected physical desktop computers executing virtual machines (VMs). While a student do his/her activities, a VM, playing a slave role of the CVC, is executed as a low-priority and background process on each desktop computer used by a student. A dedicated machine for the CVC plays the role of the cluster master. All of these VMs in execution make up a CVC, which has the operating system (mainly Linux), applications, and middleware required by the application user.

This model allows researchers to continue executing applications within their native environments, guaranteeing high usability of the infrastructure. The idea is to recreate an as exact as possible environment such as those used by the researchers in a real cluster (something similar to what it is achieved by cloud computing approaches like Open Nebula, but in dedicated machines). Users access a CVC through a remote shell connection to the CVC master. The use of virtualization tools such as VMware, allows adding and taking advantage of the capabilities of tens or hundreds of machines in computer labs, as well as the user to assign and limit the resources consumed by the VMs.

When a research group requires its CVC, they can deploy it on demand using a Web application called GUMA (Grid Uniandes Management Application). GUMA allows deploying on demand a previously configured CVC. A researcher securely access GUMA (using a Web browser) and defines the size

(number of VMs) of the CVC he/she requires. GUMA automatically deploy the VMs on selected desktops, hiding the complexities associated with the location, distribution and heterogeneity of computing resources, providing an intuitive graphical interface. GUMA also provides services for selection, shutdown and monitoring of physical computers and VMs. GUMA offers high usability to the UnaGrid solution, using the on-demand approach. UnaGrid does not guarantee any service agreement, because it uses a best-effort approach. UnaGrid allows taking advantage of the idle power processing in a 24x7 scheme and, additionally, can be deployed on desktops running Windows, Linux, or Mac operating systems, since the entire deployment is based on the use of virtual machines. The UnaGrid architecture is illustrated in Figure 3.

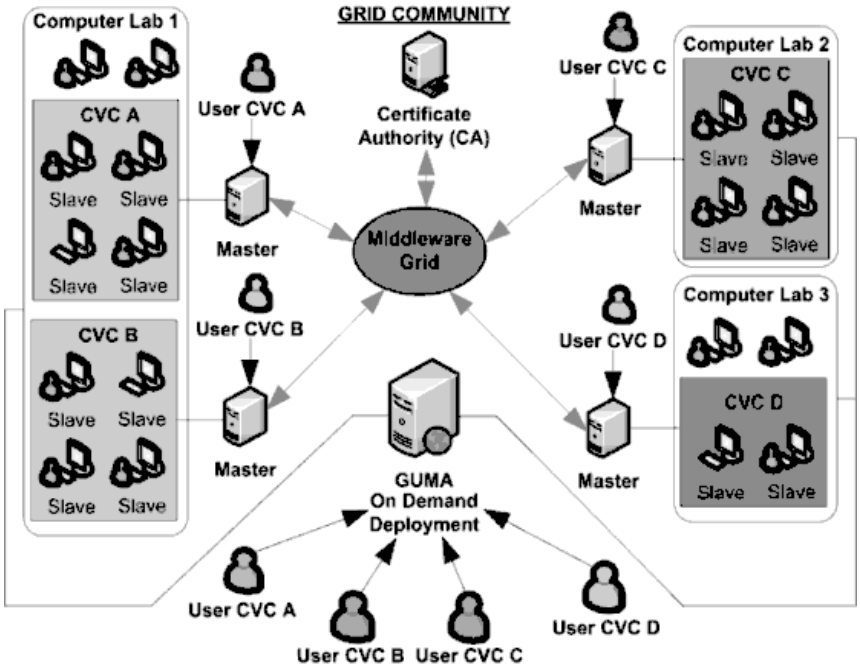

Figure 3: Architecture of the proposed opportunistic virtual grid infrastructure

UnaGrid has been deployed in three computer labs at the *Universidad de los Andes*, with Windows XP as the base operating system (see Figure 4). Each laboratory has 35 computers with an Intel i5 processor and 8GB of RAM memory. All desktops are interconnected through 1 GbE and 10GbE links. We use an NFS-NAS solution to store the information of all CVCs. In some CVCs the master nodes have installed the Globus middleware to allow that jobs, sent by researchers, can be distributed and executed in different CVCs.

The GUMA Web portal was developed using open technologies, such as Java Server Faces (JSF), Enterprise Java Beans (EJB), GlassFish, and MySQL. Regardless of the administrative domain, this application executes and manages the CVCs. Multiple execution tests, supported by the active directory services of two domain controllers (Windows 2003 and 2008 Server) have evidenced high level of performance in the CVC execution, such as the launching of 35 virtual machines in less than 5 seconds and their afterward shutdown in less than 4 seconds (assuming the VM image has been previously copied and configured to every single desktop).

GUMA manages the remote execution of the instances of the virtual machines executor (VMware Workstation), which runs on every desktop of a computer lab. Using a client-server scheme together with authentication, authorization, and confidentiality mechanisms, GUMA provides multiple services for managing the infrastructure.

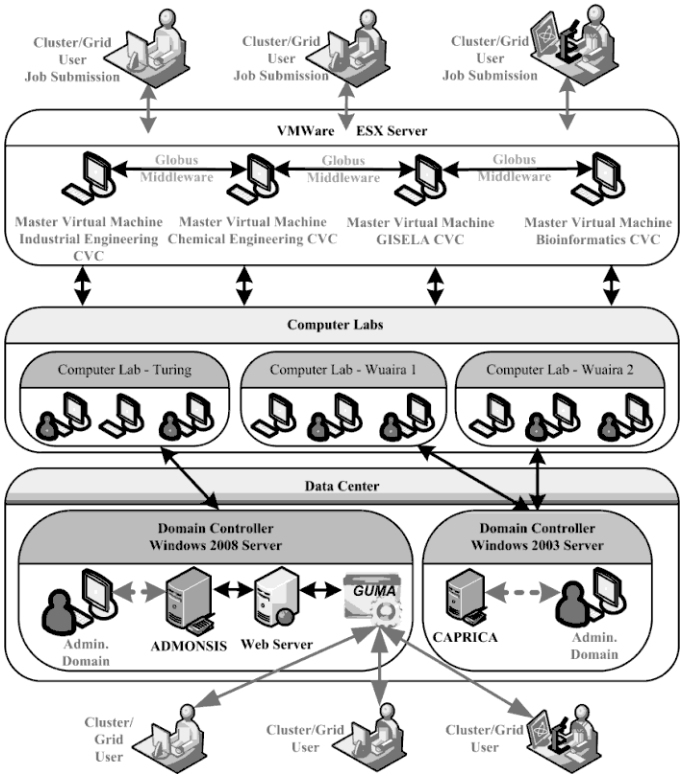

Figure 4: Opportunistic virtual grid infrastructure deployed

Although, launching a CVC is really fast, the deployment and configuration (IP and hostname assignment) of every single VM image is a very time consuming task. We then designed, implemented and tested a new mechanism to deploy and automatically configure as many instances as required by the user, from a single image stored in a centralized server. This mechanism is currently working and the time to launch a CVC now takes 4-5 minutes but the storage required and the human effort is vastly improved.

3.3. DISSEMINATION ACTIVITIES

Dissemination of the services supported by WP6 has been achieved through the communication channels put in place by the WP2 team, i.e. training events, grid schools, newsletter, Web site, etc.. In addition, some scientific papers, listed below, have been produced:

- Brasileiro, Francisco; Gaudencio, Matheus; Silva, Rafael; Duarte, Alexandre; Carvalho, Diego; Scardaci, Diego; Ciuffo, Leandro; Mayo, Rafael; Hoeger, Herbert; Stanton, Michael; Ramos, Raul; Barbera, Roberto; Marechal, Bernard; Gavillet, Philippe. Using a Simple Prioritisation Mechanism to Effectively Interoperate Service and Opportunistic Grids in the EELA-2 e-Infrastructure. Journal of Grid Computing, p. 1-17, 2011.

- Brasileiro, Francisco; Andrade, Nazareno; Lopes, Raquel Vigolvino; Sampaio, Lívia Maria Rodrigues. Democratizing Resource-Intensive e-Science Through Peer-to-Peer Grid Computing. In: Xiaoyu Yang; Lizhe Wang; Wei Jie. (Org.). Guide to e-Science: Next Generation Scientific Research and Discovery. London: Springer-Verlag, 2011, p. 53-80.
- Barbera, Roberto; Brasileiro, Francisco; Bruno, R.; Ciuffo, Leandro; Scardaci, Diego. Supporting e-Science Applications on e-Infrastructures: Some Use Cases from Latin America. In: Nikolaos P. Preve. (Org.). Grid Computing. London: Springer-Verlag, 2011, p. 33-55.
- Hamar, Vanessa. DIRAC on GISELA. DIRAC User Community Meeting, 12th – 13th May 2011, Barcelona (Spain).
- Castro, Harold; Rosales, Eduardo; Villamizar, Mario; Jimenez, Artur: UnaGrid. On Demand Opportunistic Desktop Grid. CCGRID 2010, p. 661-666
- Souza, Carla; Lacerda, Ana Clara; Silva, Jonhnny W.; Pereira, Thiago Emmanuel; Soares, Alexandro S.; Brasileiro, Francisco. BeeFS: Um Sistema de Arquivos Distribuído POSIX Barato e Eficiente para Redes Locais (in Portuguese). In: XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2010, Gramado. Anais do XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (Salão de Ferramentas). Porto Alegre, Brasil : Sociedade Brasileira de Computação, 2010. v. 1. p. 1033-1040.

4. HUMAN EFFORT

The current human resources allocated to WP6 are listed in Table 1. The data has been extracted from the GISELA timesheets system.

Table 1 – WP6 Human Resources

Name	Role	Partner
Francisco Vilar Brasileiro	WP6 Manager TWP6.1 and TWP6.3 Task Leader	UFCG
Vanessa Hamar	WP6 Deputy Manager TWP6.2 Task Leader	CNRS / CPPM
Lívia Campos	TWP6.2 staff	UFCG
Rodrigo Duarte	TWP6.2 staff	UFCG
Raquel Lopes	TWP6.3 staff	UFCG
Carla Souza	TWP6.3 staff	UFCG
Rodolfo Viana	TWP6.3 staff	UFCG
Harold Enrique Castro Barrera	TWP6.3 staff	UNIANDES
Mario Villamizar	TWP6.3 staff	UNIANDES
Germán Sotelo	TWP6.3 staff	UNIANDES
Arthur Oviedo	TWP6.3 staff	UNIANDES
Diego Scardaci	TWP6.2 staff	INFN

5. OPEN ISSUES AND / OR DEVIATIONS FROM THE WORK PLAN

We have planned the development of five new services to be incorporated in the GISELA portfolio. Instead of working in parallel on the development of all services, we decided to concentrate the work on two services in the first year. These two services, namely BeeFS and CVC, have been completely developed and tested. The other three services will be target in the second year. As a consequence, the first milestone was only partially achieved.

Science Gateways are considered as valid and innovative tools to increase grid adoption and usage. By hiding the complexity of the grid environment, Science Gateways can indeed allow large Virtual Research Communities to easily access the e-infrastructures, reducing the skills needed today to fully exploit them.

In collaboration with WP3, the WP6 team decided to design and develop a Science Gateway for the Industrial Applications. We chose a pilot application to be integrated in this Science Gateway, named Industry@Grid (see http://applications.gisela-grid.eu/application_details.php?l=20&ID=14) and we started to collaborate with its application developer team. This Science Gateway will be developed using international standard as JSR 168 e JSR 286 for Web Portal development and the Simple API for Grid Applications (SAGA) Core API, a high level, application-oriented, software library for grid application development specified by the Open Grid Forum. SAGA allows the creation of a unique interface towards different middleware stacks and makes Scientific Gateways able to exploit resources coming from different Grid worlds. This is particularly useful for a multi-middleware e-infrastructure such as GISELA's one.

6. PLANS FOR THE NEXT REPORTING PERIOD

For the next reporting period we will continue to provide support for all the services available in the GISELA portfolio and continue the development of the new services proposed. In particular, work will be concentrate on the following services:

- Efficient execution of data-intensive applications based on the Map-Reduce paradigm;
- Seamless execution of CPU-intensive applications in hybrid e-Infrastructures augmented with the capability of interfacing with cloud computing providers;
- Development of specialised application portals based on the DIRAC Web Portal project.

Along the second project-year we are also going to implement an elastic and transparent integration of the CVC with the gLite infrastructure. We are going to build a gLite CVC and, by modifying the local scheduler at the Computing Element (CE), we will allow a gLite site to automatically launch as many opportunistic nodes as needed to deal with jobs arriving to the site. This will allow service providers to handle much more GISELA jobs with no need to invest in new servers.

Finally, we will try to have an even closer interaction with both WP3 and WP4 teams in order to make sure that the services developed are used by both application users and system administrators, improving the experience with the GISELA infrastructure.

7. CONCLUSIONS

Overall, the outcomes from the first year of execution of the activity have been very good. We were able to upgrade in many ways the services already available in the GISELA portfolio and enhance the portfolio with the addition of two new services. Most importantly, the services supported are already fulfilling their aim of facilitating the implementation and deployment of applications.

According to Table 13 of the DoW, the accomplishments of WP6 are to be measured using the following Quality Metrics: number of scientific papers published, and percentage of applications using at least one WP6 service. We expected to have at least 2 scientific publications by M12. This expectation was more than doubled, as presented in Section 3.3. For the percentage of applications using services from the portfolio, our expectation was to have at least 30% of the applications using at least one of the services available in the portfolio. From the 13 new applications supported by GISELA, 4 of them are using the services provided by WP6. This corresponds to a little less than 31% of the applications supported.