



# Diligent

## A **D**igital **L**ibrary Infrastructure on **G**rid **E**Nabled **T**echnology

Deliverable No 4.4.2:

“Updated Demonstration plan and activity report - interim”

Jan 2007

## Document Information

### Project

Project Title:	DILIGENT, A <b>DI</b> gital <b>L</b> ibrary <b>I</b> nfrastructure on <b>G</b> rid <b>EN</b> abled <b>T</b> echnology
Project Start:	1 Sep 2004
Call/Instrument:	FP6-2003-IST-2/IP
Contract Number:	004260

### Document

Deliverable number:	4.4.2
Deliverable title:	Updated Demonstration plan and activity report - interim
Contractual Date of Delivery:	Dec 2006
Actual Date of Delivery:	30 Jan. 2006
Editor(s):	ENG
Author(s):	Andrea Manieri, Pasquale Pagano, Enrico Hauer, Michael Springmann
Participant(s):	ENG, CNR, FhG, UNIBAS
Workpackage:	4.4
Workpackage title:	Demonstration
Workpackage leader:	ENG
Workpackage participants:	ENG, CNR, UoA, UNIBAS, FhG/IPSI
Est. Person-months:	3
Distribution:	Public
Nature:	Report
Version/Revision:	1.0
Draft/Final	Final
Total number of pages: (including cover)	42
File name:	DILIGENT-D4.4.2.v03.doc
Key words:	

## Disclaimer

This document contains description of the DILIGENT project findings, work and products. Certain parts of it might be under partner Intellectual Property Right (IPR) rules so, prior to using its content please contact the consortium head for approval.

In case you believe that this document harms in any way IPR held by you as a person or as a representative of an entity, please do notify us immediately.

The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated the creation and publication of this document hold any sort of responsibility that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of DILIGENT consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 25 Member States of the Union. It is based on the European Communities and the member states cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors. (<http://europa.eu.int/>)



**DILIGENT is a project partially funded by the European Union**

## Table of Contents

Document Information .....	2
Disclaimer .....	3
Figures .....	6
Tables .....	7
Executive Summary .....	8
1 Introduction .....	10
1.1 Outline of the document .....	10
1.2 Methodology .....	10
1.3 Sources of Information .....	11
2 Demonstration Strategy.....	12
2.1 Understanding Customers point of view.....	12
2.1.2 Some cases .....	13
2.2 Demonstrations selection .....	16
3 Report on Demonstrators development .....	17
3.1 Demonstrator 1: Dynamic Service Instance Deployment .....	17
3.1.1 Demonstrator design.....	17
3.1.2 Overview of current development status .....	18
3.1.3 Target Audience .....	19
3.1.4 Where to find it and how to use it .....	20
3.2 Demonstrator 2: Watermarking and Encryption of Videos .....	20
3.2.1 Demonstrator design.....	20
3.2.2 Overview of current development status .....	21
3.2.3 Where to find it and how to use it .....	22
3.3 Demonstrator 3: Batch Feature Extraction .....	23
3.3.1 Demonstrator design.....	23
3.3.2 Overview of current development status .....	24
3.3.3 Target Audience .....	25
3.3.4 Where to find it and how to use it .....	25
3.4 Demonstrator 4: User management and Security mechanism in Rendering.....	27
3.4.1 Demonstrator design.....	28
3.4.2 Overview of current development status .....	29
3.4.3 Previous experiences and Target Audience.....	31
3.4.4 Where to find it and how to use it .....	31
3.5 Demonstrator 5: Content-based Multimedia Retrieval .....	32
3.5.1 Demonstrator design.....	32
3.5.2 Overview of current development status .....	33
3.5.3 Target Audience .....	33
3.5.4 Where to find it and how to use it .....	34
3.6 Demonstrator 6: Dynamic Infrastructure Deployment .....	36
3.6.1 Demonstrator design.....	37
3.6.2 Overview of current development status .....	37

3.6.3	Target Audience .....	39
3.6.4	Where to find it and how to use it .....	39
4	Conclusions .....	40
4.1	Evaluation of Process & Results .....	40
	References .....	42

## Figures

Figure 1 - Demonstrator Dynamic Service Instance Deployment Architecture .....	18
Figure 2 - Demonstrator Dynamic Service Instance Deployment Graphical User Interface ..	19
Figure 3 - Demonstrator Video Watermarking Design.....	21
Figure 4 - Demonstrator Video Watermarking Development status.....	22
Figure 5 - Demonstrator Video Watermarking GUI interface at GILDA .....	23
Figure 7 – Demonstrator Batch Feature Extraction Design .....	24
Figure 8 – Demonstrator Batch Feature Extraction Statistics .....	25
Figure 9 - DILIGENT VO Model .....	28
Figure 10 - User management and Security mechanism in Rendering deployment view .....	29
Figure 11 - User management and Security mechanism in Rendering Homepage .....	32
Figure 12 - Content-based Multimedia Retrieval Design.....	33
Figure 13 - Content-based Multimedia Retrieval Statistics.....	33
Figure 14 - Demonstrator Dynamic Infrastructure Deployment Architecture .....	37
Figure 15 - Demonstrator Dynamic Infrastructure Deployment Graphical User Interface .....	38

## Tables

Table 1 - Digital Libraries Market segments.....	12
Table 2 – Target coverage of DILIGENT demonstrators .....	16

## Executive Summary

The objective of this document is to report the activities performed in the Demonstration workpackage (WP 4.4) until M24, including the final description of planned demos and instruction for their access and use.

In particular it provides a detailed description of the activities and objectives planned until the end of the workpackage that will be also included in the New Implementation Plan M25-36. The aims of this document is to have a clear picture of the status of this WP, the problem encountered, any deviation from original plans and the corrective actions adopted. Some information already submitted as IMR for the past periods here are better represented and discussed.

Demonstration activities have the objective to promote the main features of the DILIGENT project to potential customers. Target customers have been analysed within WP 4.3 activities (Task 4.3.1 Virtual research organisation aggregation and analysis) and results of this analysis served as starting point for the subsequent demonstration identification.

As described in the D4.4.1 *Preliminary Demonstration Plan*, demonstration of concrete results involves business aspects, content selection strategies, organisational and technological aspects. The main objective is to highlight innovative and interesting results and present to potential users in an attractive way. This rationale led to the development of a set of demonstrators related to three different contexts:

- Cultural Heritage of the historical and humanities activities performed in the ARTE context of the Scuola Normale Superiore (SNS) university of Pisa,
- E-Science context emerged by the European Space Agency and EGEE communities
- Multimedia processing performed by RAI Educational mainly related to the Mediateca and Prometeo tools as described in the D4.4.1 Preliminary Demonstration Plan document.

Demonstrators are targeted to the three different audience categories:

- Developers, members of a research organisation, which are in charge to develop a DL, or are involved in evolution or maintenance of a grid middleware
- Decision Makers, whose people that have the responsibility to make strategic chooses comparing DILIGENT approach to other solution based on non-only-technical characteristics
- End Users, the people that will experience the daily use of the system, uploading and accessing the information, creating new contents and exploiting the DILIGENT solution in their business work.

Starting from these target categories, six demonstrators have been planned and are being implementing for the DILIGENT project. They are briefly introduced in the following.

The *Dynamic Service Instance Deployment* demonstrator aims to prove that location and re-location of a Web Service (WS) on a cooperating remote host is not only feasible but makes possible to simplify the management of a Service Oriented Application developed on top of a

grid middleware. It is mainly targeted to Developers and it's already available on the CNR infrastructure.

The *Watermarking and Encryption of Videos* demonstrates the process of digital watermarking as a method to verify the unauthorized copy of digital contents. It is targeted to all three audience categories and already available through the GILDA platform.

The *Batch Feature Extraction* demonstrator illustrates distributed parallel batch processing on the Grid for enhanced performance of time consuming data processing. It provides the technological foundations for content-based searches atop pre-extracted image features. It is targeted to Decision Makers and End Users and available at the UMIT installation.

The *User management and Security mechanism in Rendering* shows the opportunities of the DILIGENT security Model and provides a video rendering application on the Grid based on the gLite middleware. It is targeted to all three audience categories and already available on the ENG infrastructure.

The *Content-based Multimedia Retrieval* shows the integration of existing Grid-enabled technology for parallelized Content-Based Image Retrieval (CBIR). It illustrates the search effectiveness using sophisticated similarity measures, comprehensive query models, and expressive feature types. It is mainly targeted to End Users and available at the UMIT installation.

The *Identification and Configuration of Dynamic Services* demonstrator aims to prove that the co-deployment task of an infrastructure of interoperable services, appropriately configured, is feasible. By exploiting the Package Repository features, this demonstrator identifies all the mandatory services/packages and organises them in a simplified DL Map. Despite such map is not optimised, e.g. does not reduce the number of needed DILIGENT Hosting Nodes, it assures the compatibilities among the services allocated on the same DHN while preserves their package dependencies. This demonstrator is targeted to Developers and already available on the CNR infrastructure.

The following step to perform in order to improve the usability of demonstrators is to integrate them in a single uniform application accessible through the GridSphere portal. In order to do this, two steps could be performed until the end of the workpackage activities (March 2007). The first one consists of the creation of an access page linking all demonstrators. A brief description of each one will clarify the purpose and functionalities of the demonstrator to users. The second step aims to integrate all demonstrators in a single portal application. Accessing this applications, users will be able to test functionalities of supplied demonstrators in a uniform environment.

## 1 INTRODUCTION

The purpose of this document is to report the activities carried out in Demonstration workpackage and to update the plan for demonstrating main features of the DILIGENT test-bed.

### 1.1 Outline of the document

The last phase of DILIGENT focused on developing scenarios for DILIGENT services customized for different type of audiences, integrating and adapting results from technical work-packages into usable demos available on-line.

Two other main chapters, in addition to this one, compose the document:

Chapter 2 provides the update of the strategy adopted for the whole demonstration activities, different technical and business aspects considered and success criteria; the strategy, already introduced in D4.4.1 "Preliminary Demonstration plan", has been adapted to address new issues raised in the last period;

Chapter 3 describes the set of demonstrators under development and those available on the web for immediate use; for each demonstrator is given a short description of purpose, technical results used, potential target audience and success indicators;

The document is completed by a "Conclusion" chapter where we reflect upon work done and plan the final phase.

### 1.2 Methodology

In the previous phase, this WP was mainly devoted in selecting relevant features of the system under design and, starting from an initial analysis of potential user categories, in developing simple demonstrators accessible on-line.

This phase, due to delays in design and develop DILIGENT components implementing selected features, is far to be conclude. Only four out of six demonstrators have been put on-line by the end of 2006.

In this chapter we review the demonstration strategy based on lessons learned during the last two years.

In particular, given that DILIGENT project aims at creating "an advanced test-bed that will allow members of dynamic virtual e-Science organizations to access shared knowledge and to collaborate in a secure, coordinated, dynamic and cost-effective way" [1], particular attention will be given in approaching user communities and providing them with usable on line tools to demonstrate main DILIGENT features.

In order to reach the expected impact with Users Communities a single-stop page in the DILIGENT web-site will be provided from which the users could i) understand the main features of DILIGENT accessing the six on-line demonstrators, ii) access any training material available (technical documentation as well as the DILIGENT training DL).

The creation of material in the DILIGENT DL and its maintenance, as well as the installation and maintenance of training version of DILIGENT System are not part of the Demonstration activities.

### **1.3 Sources of Information**

The descriptions of demonstrators contained in chapter 3 of this document have been provided by DILIGENT partners involved in demonstrator development. Information about user groups has been gathered from other project deliverables and from informal meetings with people of these groups.

All these documents are listed at the end of the document and they are available on the project web repository (BSCW).

## 2 DEMONSTRATION STRATEGY

This chapter describes the new demonstration strategy that will be adopted by DILIGENT project to show its results to future potential customers. Such strategy depends on the creation of a real DILIGENT Community and the analysis of the impact of project results on Users that will be performed respectively in Dissemination and Exploitation activities.

### 2.1 Understanding Customers point of view

Following the market segmentation made within the WP4.3 activities and reported in [2] potential customers for DILIGENT are listed in the following table:

Market segment	User group members	Segment characteristics	Comments
Libraries	Public and corporate libraries	<ul style="list-style-type: none"> <li>Digital libraries are business critical</li> <li>Small budgets</li> <li>Large amounts of data and low computational requirements</li> </ul>	Familiarity with Digital Libraries, low IT competency.
e-Science	Private and public research organizations	<ul style="list-style-type: none"> <li>Digital libraries are business critical</li> <li>Large budgets</li> <li>Large amounts of data and high computational requirements</li> <li>Requires different types and sizes of Digital Libraries</li> </ul>	Familiar with Digital Libraries, high IT competence, drives Grid development
Media/publishing companies	Private media companies and public broadcasters	<ul style="list-style-type: none"> <li>Large amounts of data</li> <li>Unknown budgets</li> </ul>	Demand for Digital Libraries is questionable.

Table 1 - Digital Libraries Market segments

For each market segments different user profiles or interests have been selected to better focus the demonstrators. In the rest of the chapter such profiles are detailed.

#### 2.1.1.1 Developers

Developers are those members of a research organisation, which are in charge to develop a DL, or are involved in evolution or maintenance of a grid middleware. They are interested in demonstrations of technical DILIGENT features.

Demonstrations should be focused on

- How grid services are used to implement basic capabilities of a DL either from the perspective of a DL developer or a Grid-middleware architect and
- How to develop or integrate third-party services on DILIGENT platform (API, security mechanism, etc).

### **2.1.1.2 Decision-Makers**

Decision-Makers are people with the responsibility to make strategic choices comparing the DILIGENT approach to other solution based on non-only-technical characteristics.

Each decision-maker will have its own perspective that could vary strongly depending on their interest in the technology. Nevertheless some common issue can be highlighted:

- Reduction of initial and maintenance costs or optimisation of overall IT expenditure;
- Reduction of effort to address users needs (or to improve number of final users);
- Valorisation of institution assets;
- Improving the user experience interacting with the institution;
- Address unsatisfied requirements.

### **2.1.1.3 End Users**

End Users are the people that will experience the daily use of the system, uploading and accessing the information, creating new content and exploiting the DILIGENT solution in their business work.

In general End users are interested to verify that:

- The new system services will perform in the same way, or better, than their previous system (fastest or better response);
- The new system reduces the user effort (or attention requirements) in performing an activity;
- The new system allows the user to do more complex or new things.

## **2.1.2 Some cases**

Different target groups have different interests and decision-making processes. These interests and processes have been analyzed interacting with user communities representatives in the project.

For each Community are described the context of work, the developer requirements, the decision-makers perspective and user interests.

### **2.1.2.1 The ESA case**

The European Space Agency has been using grid technology for a while now, executing time consuming multi-part tasks on their Grid On-Demand system. All these processes are related to image analysis and data integration. The so-called "products" are geo-referenced images with visual representation of historical data.

The nature of earth observation implies that huge amounts of data are produced daily and scalable systems are required to deal with this information. While the current On-Demand system provides scalable data analysis, ESA expressed the need for scalable information management.

Developers are internal (or external) software developers that have the role to maintain and improve the Information Infrastructure (the On-Demand system is not the only information infrastructure available in ESA). The main interest is in how to allow DILIGENT services to gather information or products (security and API features) from the Grid On-Demand system and how allow interoperability with DILIGENT and the LCG II/gLite internal installations.

Decision-Makers in ESA are department directors that actually are well aware about grid technology. Their main interest is to reduce the effort in organizing and aggregating the data they have to accomplish their institutional objectives.

End-users are ESA employees working on several departments that are using On-Demand Systems to access its "products". They are interested to have some automatic or semiautomatic ways to produce different reports, aggregating existing information or "products". The most interesting issue is to update a "product" (i.e. a document containing some aggregated data and geographical references) with new (fresh) data or maps.

### 2.1.2.2 The RAI case

RAI Educational is a department of RAI, the Italian broadcasters, mainly dealing with educational challenges.

During the last years several tools and systems have been supplied to general public, two of them are particularly interesting in the DILIGENT perspective because represent potential future scenarios where the DILIGENT technology can be adopted:

- "Mediateca" is the Multimedia content repository related to the transmission "Mediamente" that for six years presented technological, cognitive and societal aspects of ICT. In the Mediateca users can find interviews (in text and audio format), sample video, hyperlink to related contents either in the same repository or on the web.
- "Prometeo" is the Rai web site dedicated to secondary school teachers interested in select RAI educational content (mainly video) for their lessons. Through the web site the registered user select the content required filling a request form. The Rai backstage find the video, and plan it broadcasting through the satellite channel (hot bird 1 or 2), then contact by email the teacher to advice him/her of the date and time of the transmission.

The concept of Developers in RAI is very complex, in fact within this category we could consider either developers of the software application or people in charge of the creation of a services and its design in terms of infrastructure (web site, TV or satellite channels, Archives, etc). While the former are often third-party software developers, the latter are people with ICT and Media competences. We choose to address only the second group, as they are interested in solutions to improve multimedia management using the grid technology. Demonstrators 4 (*User management and Security mechanism in Rendering*) and 2 (*Watermarking and Encryption of Videos*) have been created to address these interests.

Decision-makers, as in ESA, are the directors of each department with their institutional mission and assigned budget. Their interest is in identifying the most economic solution to address their objectives. A clear understanding of concrete interest has not been possible since no interview was possible to organize with such figures. No demonstrator for them has been designed.

End-users of RAI Educational services are the general public, primary and secondary teachers, academic researchers on Visual Arts, History, Social studies, Communication techniques, thus non-IT experts, that are interested in the quality of services (performance in accessing and retrieving the expected content, usability of the interface).

### 2.1.2.3 The SNS case

Scuola Normale Superiore is a university in Pisa, specialized in Humanities and strongly active in the historical field. One of the main interests is related to the ARTE project, a national initiative funded by MIUR, the Italian Ministry for Higher Education and Research. *"This is a community of scholars, distributed all over the world, who have decided to start working together in order to set up the basis for a new research discipline that merges together experiences from the medical, humanity, social science, and communication research areas"*[3]

The analysis of this Community have been made through the representatives that currently work in DILIGENT, hence exploiting their knowledge of the people they work with.

In the SNS environment developers are only third-party programmers that are called on-demand to realise the required applications.

Decision-makers are professors and heads of departments that at different levels have the autonomy to select the most appropriate tool to support their educational and research activity. Both figures are strongly interested in the reduction of costs in sharing resource and facilities within several research projects, and along different departments.

End Users are students, researchers and professors not experts in IT that look for – at least – the same facilities they have with their usual information system. Again, the key factors are related with the usability of the interface and the performance and quality of service supplied.

### 2.1.2.4 The EGEE case

About 70 institutions between research centres, academia and industries compose the EGEE project. The main objective of the EGEE project is to support the amount of data produced, and their consequent processing, by the Large Hadron Collider that will start its activities at CERN from 2007.

Within the EGEE activity the Joint Research Activity 1 (JRA1) is dealing with the integration and evolution of existing grid-middleware to create a unique, homogeneous grid-layer, generally considered the parent of the future European grid middleware.

Another Joint Research Activity (JRA3) is related with security issues: i.e. certificates, secure storage, multi-institutional access.

Within JRAs we have identified our target audience.

Developers are ICT people of several institutions involved in the extension of existing grid services or in the development of completely new one. They are interested in the experience of using their components, to detect possible errors, wrong performing, etc ...

Decision-Makers are designers of the components and the architects of the possible evolution. They have required to understand how gLite services are used within different application and scenarios, which features are useful, which not and which are needed in future releases.

End-users are other gLite communities using the middleware for their application. They're mainly the other Virtual Community within the EGEE project and the others in related initiatives (SEE-grid, Nordu-grid, Balkan-grid, etc). They're interested in experience the way we use gLite how we manage security, user management, and how we develop grid-services on top of the basic gLite services.

## 2.2 Demonstrations selection

Analyzing target group's interests and the technology development plan (which is an internal document not publicly available) we have identified some concrete demonstrators to develop.

Some demonstrators have been selected to be usable with multiple target groups and will be designed to be cross-domain, in order to reduce the effort and investment required.

Based on an iterative process, technical partners presented software components availability that might be used to develop demonstrators. On the basis of User requirements and scenarios, this WP partners identified potential scenarios that would be possible with the promised software components.

The result is a set of demonstrators that cover at least one requirement expressed by each target group. The following table summarizes these findings:

	Cultural Heritage	E-Science	Multimedia
<b>Developers</b>	Dem. 1 "Dynamic Service Instance Deployment"		
	Dem. 6 "Dynamic Infrastructure Deployment"		
	Dem. 4 "User management and Security mechanism in Rendering "		
			Dem. 2 "Watermarking and Encryption of Videos"
<b>Decision Makers</b>	Dem. 3 "Batch Feature Extraction "		
<b>End-Users</b>	Dem. 5 "Content-based Multimedia Retrieval"		

Table 2 – Target coverage of DILIGENT demonstrators

A detailed description of each demonstrator will be given in the next chapter.

## 3 REPORT ON DEMONSTRATORS DEVELOPMENT

In the following chapter the activities performed for each demonstrator are described as well as each final demonstrator, along with practical info about the location and some initial guidelines on how to use it.

Further development and the activities still to be performed are described separately.

### 3.1 Demonstrator 1: Dynamic Service Instance Deployment

This demonstrator aims to prove that location and re-location of a Web Service (WS) on a cooperating remote host is not only feasible but makes possible to simplify the management of a Service Oriented Application developed on top of a grid middleware. This demonstrator (i) transport a packaged WS together with all required libraries to a remote host; (ii) reserve the hosting resources for a given period of time; (iii) deploy the WS in the container; and, finally; and (iv) activate it.

The demonstrator will be executed in the following scenario: through Demonstrator portlet it is possible to select the service to be deployed and the hosting node where the deployment task has to be performed. The user will select the service and the hosting node by relying on their descriptions as reported in the Information System. The status of the deployment task can be monitored by means of a pop-up window reporting the current stage (deploying, running, failed) the process is. Finally, the effective creation of the new service instance can be verified by accessing the Information System and discovering the Running Instance entry.

#### 3.1.1 Demonstrator design

The demonstrator has been designed to exploit the capabilities provided by the Hosting Node Manager (HNM), by the Package Repository, and by the DL Management services. The information provided by the Diligent Information System (DIS) developed by the DILIGENT Collective Layer is used in order to support the described scenario.

The demonstrator architecture is mainly composed by two components: the demonstrator user interface (DSIS-Portlet) and the demonstrator logic (DSIS-Logic) as reported in Figure 1.

The DSIS-Portlet exploits the Google Web Toolkit (GWT) to provide a comprehensive user interface allowing selecting a service from the list of available services, to ask for its deployment on a node, and to monitor the activity performed by the DL Management by means of a monitoring user interface.

The DSIS-Logic interacts with the DIS and the DL Management services by means of the appropriate software libraries. It performs a query against the DIS, manages an internal cache, performs the streaming of the resources, and hides to the DSIS-Portlet all the complexity related to the DILIGENT framework.

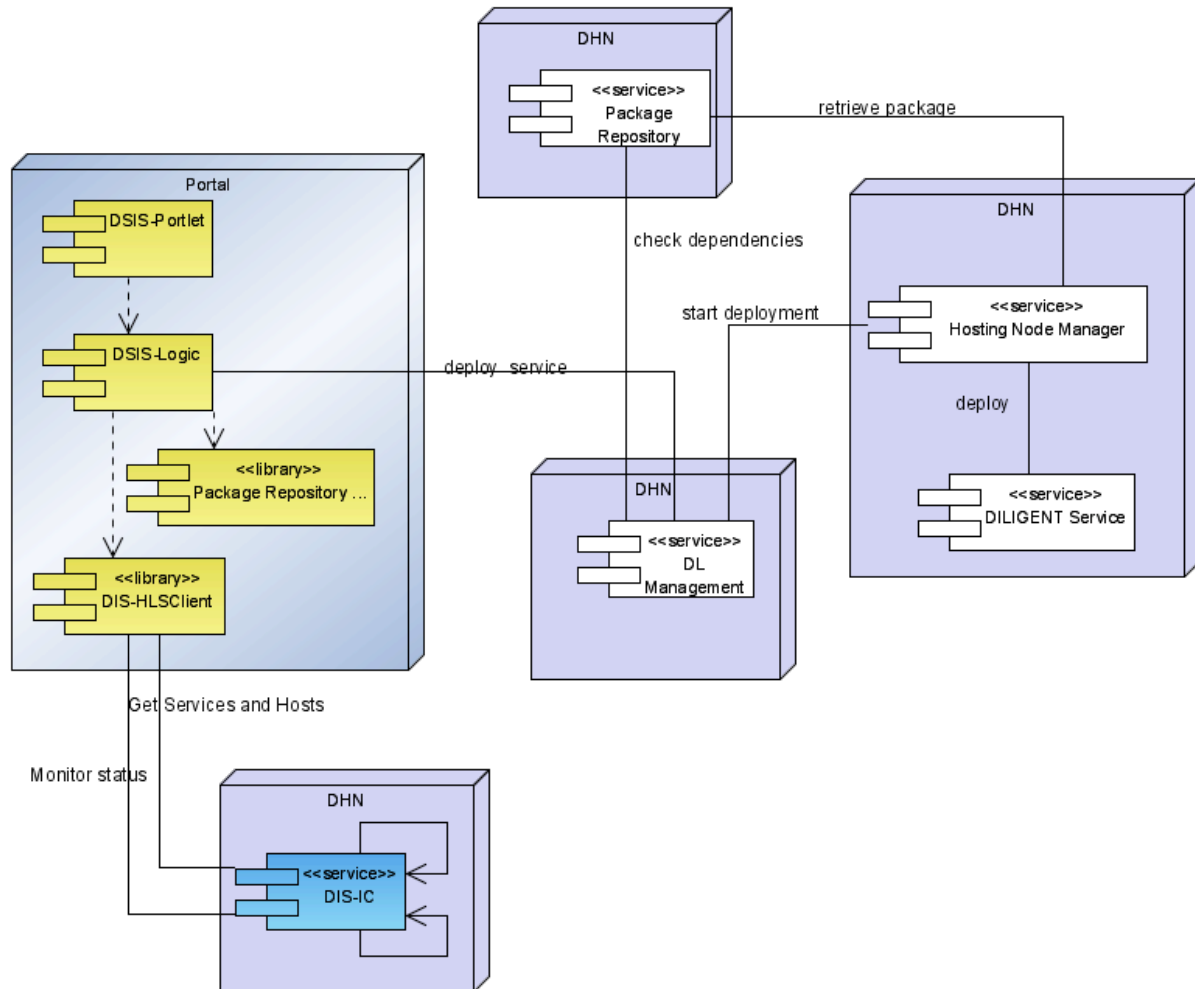


Figure 1 - Demonstrator Dynamic Service Instance Deployment Architecture

### 3.1.2 Overview of current development status

The Dynamic Service Instance Deployment demonstrator has been completely developed. The DSIS-Logic logic has been implemented and an internal interface isolates the libraries provided by the underlining services. This approach supports the easy integration of new versions of the same DIS and DL Management libraries without changing the core part of the demonstrator.

The DSIS-Portlet representing the visualization of the DILIGENT resources has been implemented as well as the activation of the DILIGENT service deployment process and its monitoring

Figure 2 shows the graphical user interface provided by the DSIS-portlet. Three main areas characterize it:

1. The *Deployment Panel* represents the main console of this demonstrator. It allows (i) to select the category of resource to browse, i.e. services and hosting nodes, and (ii) to start the deployment task. Moreover, it reports the current selected items involved in the deployment task to help the user in its activity, i.e. it reports the name of the selected hosting node and services that, once clicked, provides their profile and offers a link enabling the user to discover the service instances currently deployed on the selected hosting node.
2. The *Resources Browser* represents the area enabling the user to browse the resource instances of a certain type, e.g. service, hosting node. It shows the

distinguishing features of each resource instance, e.g. its identifier, its name, and allows to navigate through this potentially huge list of entries via an easy to use and understand mechanism based on previous/next page.

3. The *Profile Visualizer* represents the panel reporting detailed information on the resource currently selected in the Resources Browser panel or in the Deployment Panel, i.e. it is in charge to display the profile describing a service or a Hosting Node.

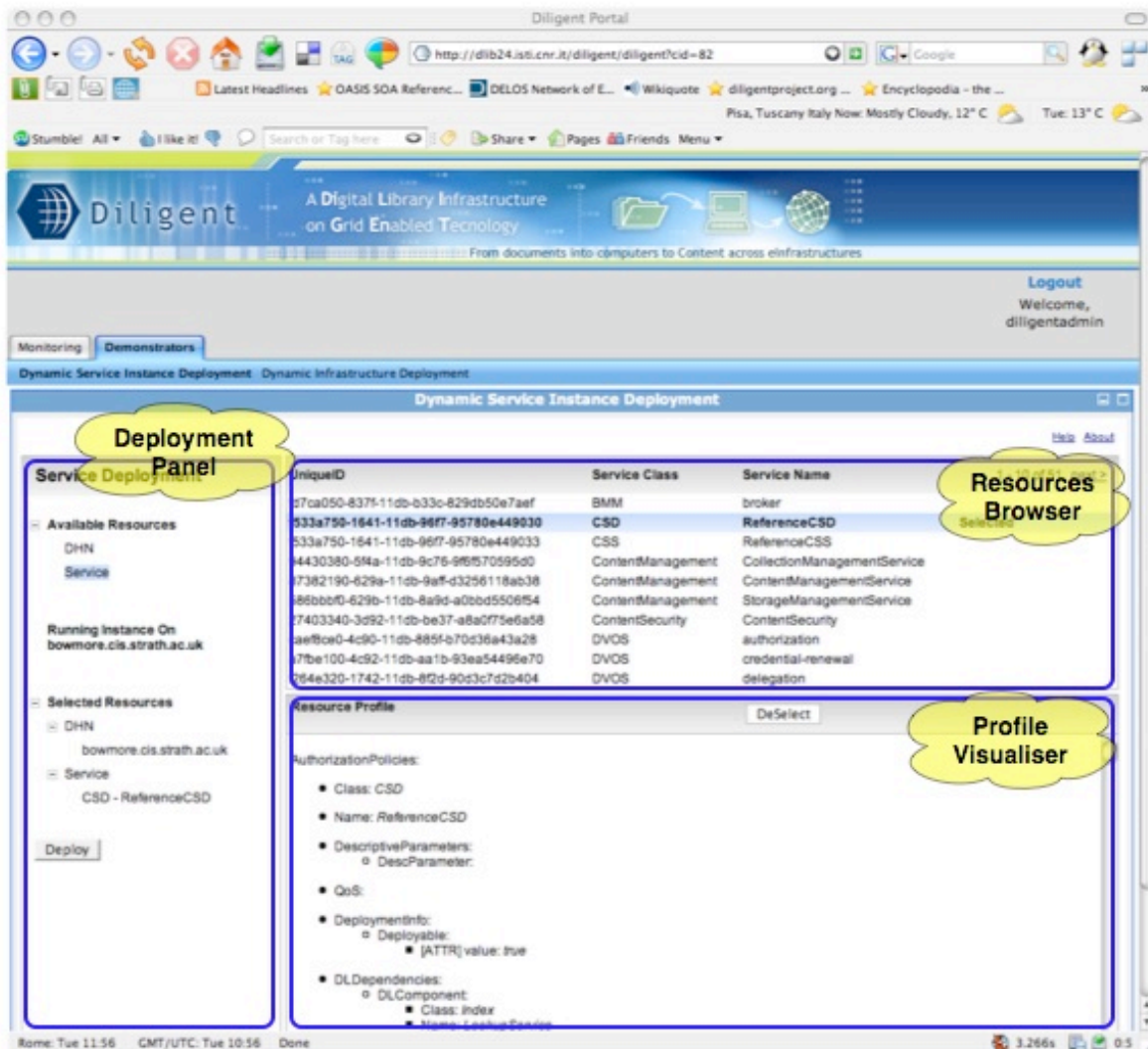


Figure 2 - Demonstrator Dynamic Service Instance Deployment Graphical User Interface

### 3.1.3 Target Audience

This demonstrator aims at demonstrating that well packaged Web Services can easier be deployed in a grid environment using the services of the DILIGENT Collective Layer.

It must rely on a demonstrator infrastructure equipped with the Diligent Information System and the Keeper services. It will target:

- High-level services developers of the DILIGENT community
- The gLite development and design team
- Other middleware designers and developers
- Grid application developers, such as the ones belonging to the ETICS team.

### 3.1.4 Where to find it and how to use it

The demonstrator can be accessed at <http://ddportal.isti.cnr.it/diligent/diligent>.

The infrastructure is currently provided by CNR-ISTI. However, a DILIGENT demonstration infrastructure or a third-parties infrastructure, e.g. the Gilda infrastructure, should be adopted to ensure the proper number of resources.

The demonstrator will be linked from the project web site and it will be accessible through the demonstrator portal. The access to the portal will be guaranteed to authenticated users. The registration process is out of the scope of the demonstrator and therefore it relies on the native authentication mechanisms provided by GridSphere portal engine.

## 3.2 Demonstrator 2: Watermarking and Encryption of Videos

The demonstrator shows the process of digital watermarking. It is a technique which allows an individual to add hidden copyright notices or other verification messages to digital audio, video, or image signals and documents. In this demonstrator media data are mpeg video files, which are marked by a digital watermark. Digital watermarking is an accepted security technology to verify the unauthorized copy of digital contents.

The demonstration is included into the GILDA demonstration platform. The applications of the demonstrator are a set of standalone applications. The applications are executed in a fixed order to complete the watermarking application. Every application will be submitted in form of a grid job. Based on these demonstrator properties GILDA is an adequate platform to apply the complete demonstration.

### 3.2.1 Demonstrator design

The demonstrator was re-designed. In D4.4.1 the idea of the demonstrator was a simulation of the grid system at a local workstation. The aspects of the advantages of the grid network can only be anticipated with the GILDA platform. GILDA provides the grid network aspects.

The demonstrator is composed by a set of applications which are applied in combination of grid jobs on a grid network. The grid jobs are combined to a DAG job. To create the DAG job a user interface is available in the GILDA platform. Through this interface the user can select the set of necessary input parameter for the watermark algorithm.

The user has to login to get access to the demonstrator. Before the demonstration starts the user has to input the necessary parameter security key, watermark message and video file. After pushing of the submit button the demonstrator application will start. The last point is the monitoring activity of GILDA to know the state of the started service.

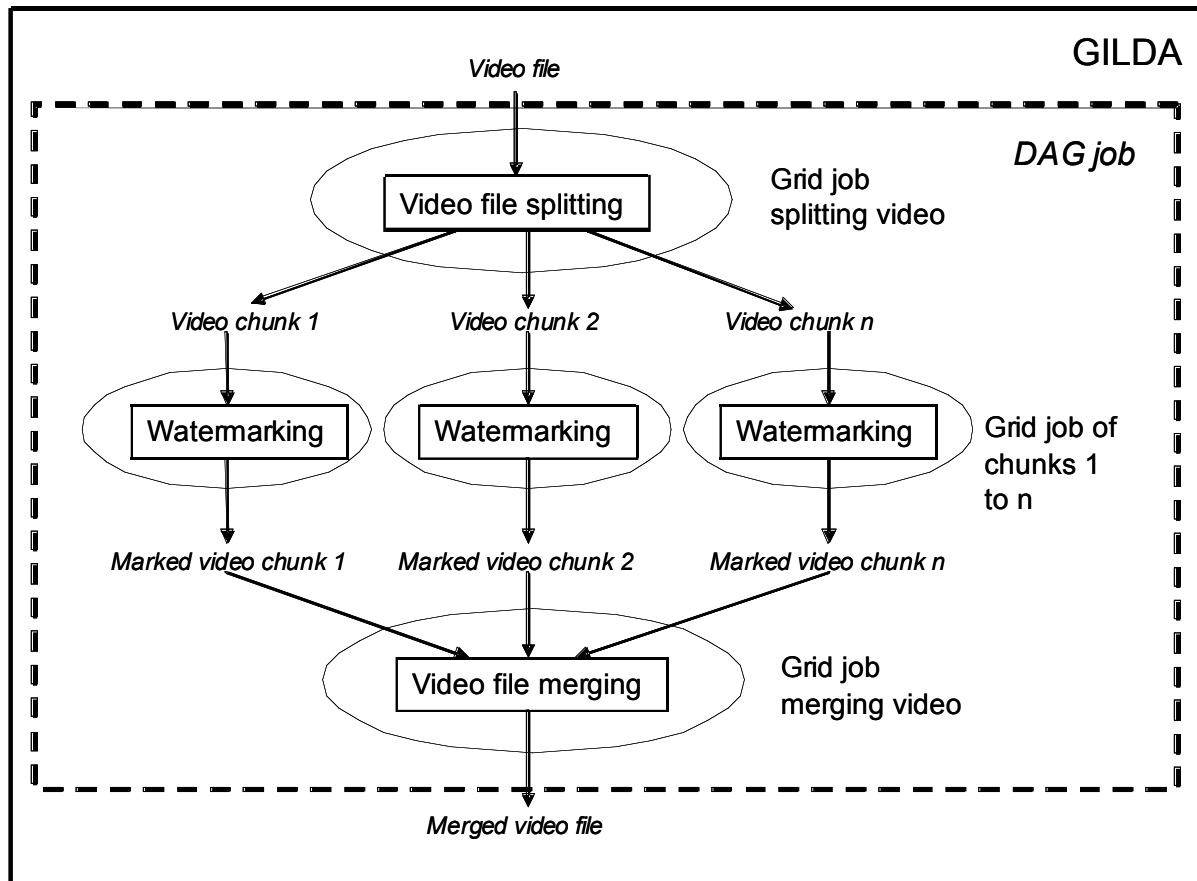


Figure 3 - Demonstrator Video Watermarking Design

Figure 3 shows the design of the demonstrator. The complete workflow consists of three sub processes: the splitting of the file into chunks, embedding of the watermark into every chunk and merging of all chunks into a final video file.

Every of the three operations (rectangles in Figure 3) will be submitted as a grid job to the grid network behind the GILDA platform. Because the submitted grid jobs have a fixed order to complete the operation a DAG is created for a dependency control between all sub processes. The demonstrator is restricted to the video file format MPEG. The applications use a MPEG decoder to get the information of the video frames and especially the watermarking application also uses a MPEG Encoder to encode the marked video frames back to a conform MPEG video file. If the video files consist of another video standard it is necessary to add a previous conversion process to convert the video file. This process can also be added as a grid job member of the complete DAG job. The three operations are command line tools. The tools are sent as a part of input sandbox of the grid job.

### 3.2.2 Overview of current development status

The status of the development is displayed in Figure 4:

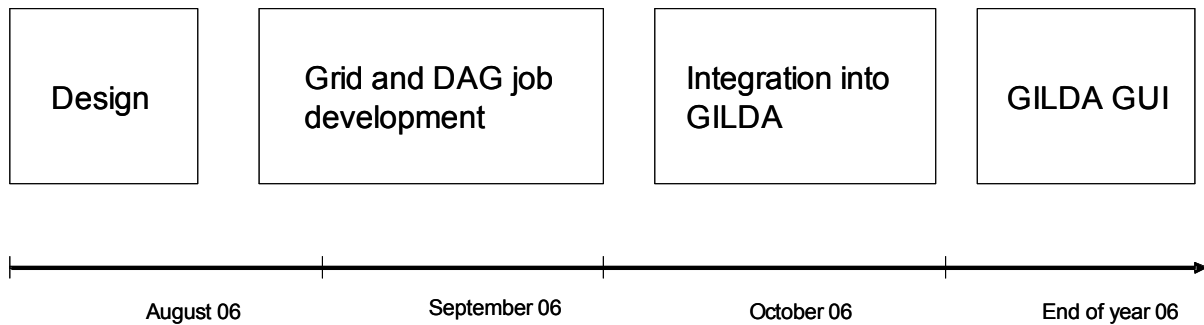


Figure 4 - Demonstrator Video Watermarking Development status

The design of the demonstrator was done mostly up to December 2006. This included I) discussions with the contact persons of GILDA to identify the necessary work for the demonstrator development, ii) development of the grid jobs shell script, its design and test on the grid networks iii) the watermarking and video splitting/merging application testing on video data provided by RAI, iv) integration of the demonstrator into the GILDA platform.

The core watermarking application was presented during the Cebit 2006. With the development of information material interested people were informed about watermarking and the aspect in the project. During this event it already existed a web application of watermarking in grid network, but it was not DILIGENT conform, and the service was not implemented to comply with the WSRF standard. At the end of June 2006 at the T-COM meeting in Frascati a stateless WSRF service was presented with the core applications.

The demonstrator includes a web presentation of the content security mechanisms to inform people about media security techniques. Thus it is possible to demonstrate the media security applications of diligent in practice.

The challenge of the demonstrator is to distribute still existing security technologies and apply them in digital library based on grid networks. Through grid networks it is possible to improve performances of complex video watermarking algorithms and also video standards.

### 3.2.3 Where to find it and how to use it

The demonstrator can be accessed at <https://glite-demo.ct.infn.it/>.

After the login, the user has to select the virtual organization of the watermarking demonstrator. In the GUI interface of the demonstrator (Figure 5) the user has to input required parameters. Starting from these parameters the security key and watermark message are generated. The video file path has to point to an existing video file at the storage element. Clicking on the submit button, a grid job is initialized. The result of job submission is the generation of a DAG job. The DAG job includes the following three operations:

- (a) video file splitting
- (b) watermark embedding
- (c) video file rejoining.

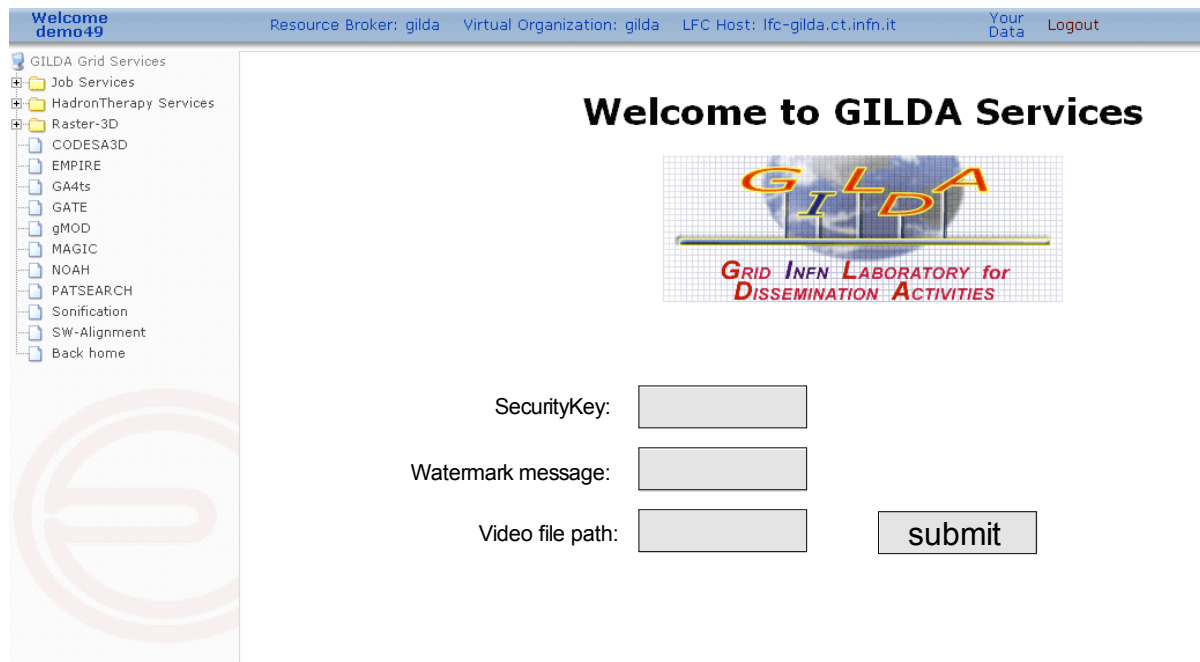


Figure 5 - Demonstrator Video Watermarking GUI interface at GILDA

Example videos are stored at a storage element of the GILDA platform. The user has to select one of the video files. The result file is also stored at the storage element.

### 3.3 Demonstrator 3: Batch Feature Extraction

This Demonstrator illustrates distributed parallel batch processing on the Grid for **enhanced performance** of time consuming data processing.

It provides the technological foundations for **content-based searches** atop pre-extracted image features.

It **integrates technology** of Grid middleware with Feature Extraction (WP1.4), Process Management (WP1.5) services, and Content Management (WP1.3).

#### 3.3.1 Demonstrator design

The demonstrator is still under development and will be released in two versions: The first one is released in a comparably early stage of the project to give a first feedback on the capabilities of distributed feature extraction, where the outcome can still affect the design and implementation of DILIGENT components. Therefore it is necessary to use pre-existing or non-DILIGENT components, until the required functionality has been implemented in DILIGENT. The second (and final) version of the demonstrator is relying on DILIGENT components, only.

The initial design of the demonstrator is as follows:

- *DILIGENT components*
  - **User Interface** provides interaction facility for end users (selection of image collection, parameterization, result presentation)
  - **Splitter** starts (static) workflow of parallel service invocations (feature extractors) processing sub collections; combines returned results
  - **Feature Extractors** fetch images from storage elements, compute localized colour signatures
- *gLite components*

- **FiReMan** catalog lists the content of a selected image collection, returns unique identifiers
- **gLite I/O** (client and server) provides file-based interface to fetch files from storage elements

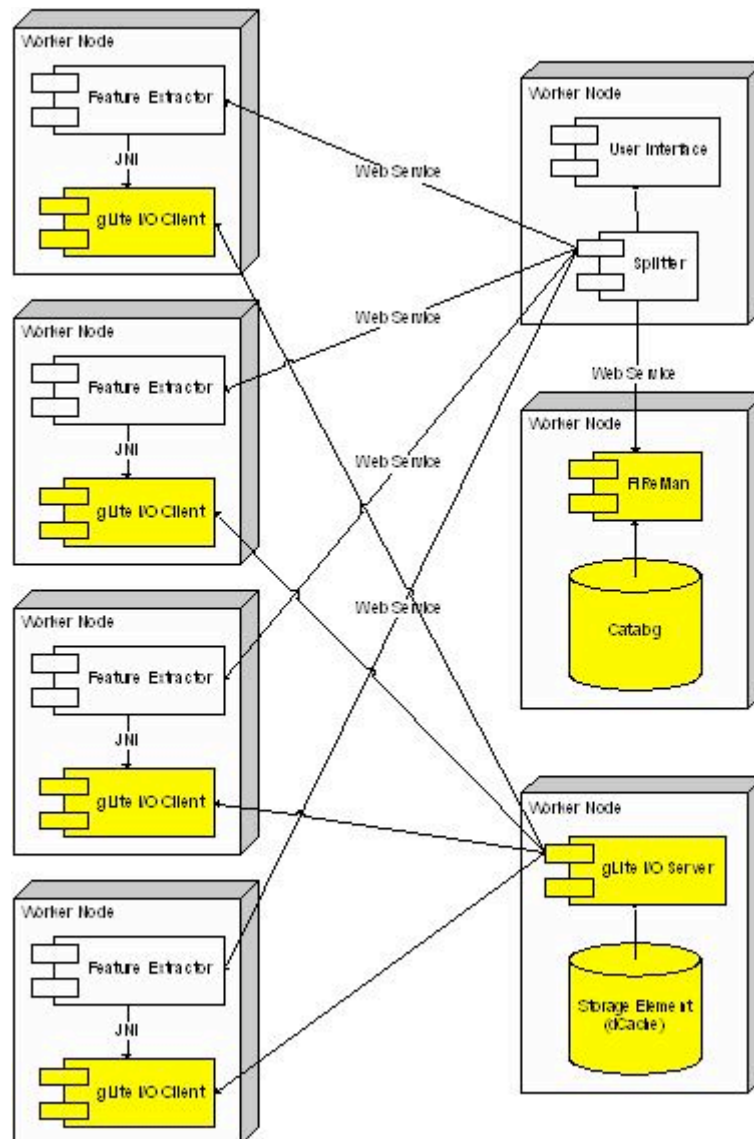


Figure 6 – Demonstrator Batch Feature Extraction Design

Due to major changes in data management from gLite 1.5 to gLite 3.0, gLite I/O, FiReMan, and dCache have been replaced by LCG components. The second version of the demonstrator will use Content Management functionality (WP 1.3), so this change will be transparent to the demonstrator. The splitting will result in a process, which will be optimized and executed by Process Management (WP 1.5).

### 3.3.2 Overview of current development status

The first version of the demonstrator has been made available before the first review (3<sup>rd</sup> quarter '05), using some non-DILIGENT components. The graph shows the results of a simple evaluation using one particular image feature (colour signatures), which can be parameterized with the number of different colour signatures to generate ( $k$ ). The range of nodes used in the experiment was from one (the whole extraction takes place on a single node, no parallelization) to 4 nodes. The test-run with 9 images (JPEG files of paintings) shows almost linear speed-up for increasing number of nodes.

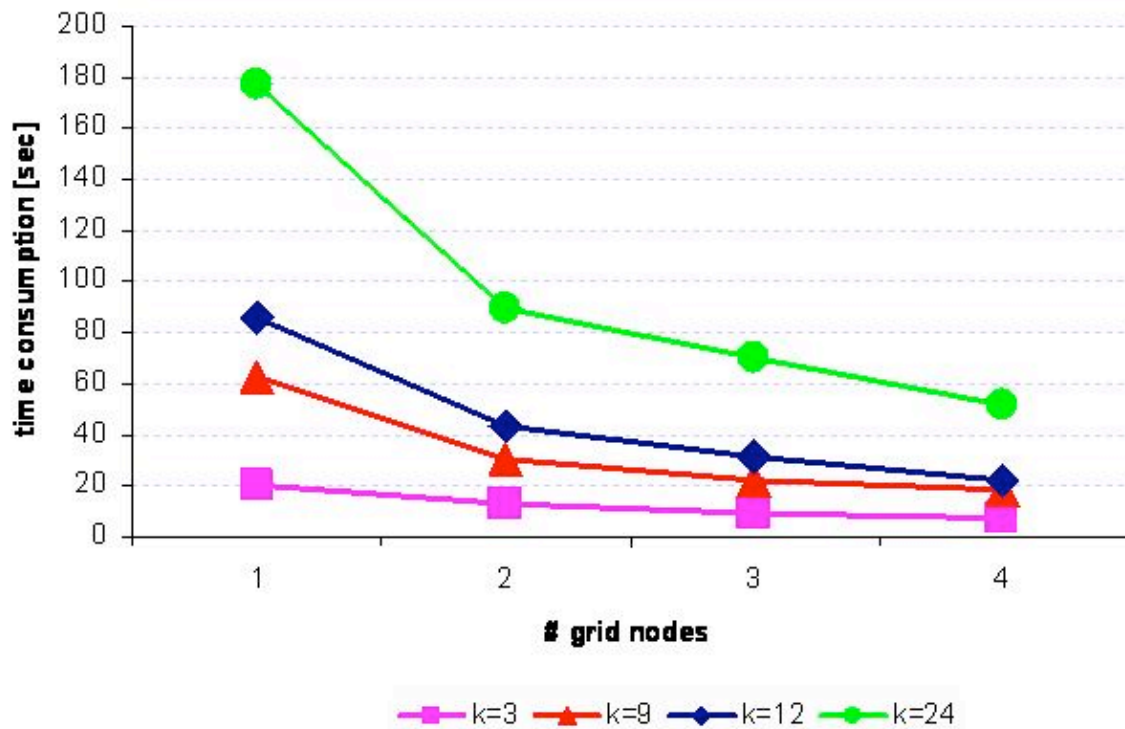


Figure 7 – Demonstrator Batch Feature Extraction Statistics

The second version of the service requires the availability of all DILIGENT components, of which use is intended. Since e.g. process optimization is planned for the BETA release of DILIGENT, the second version of the demonstrator cannot be made available before the BETA release.

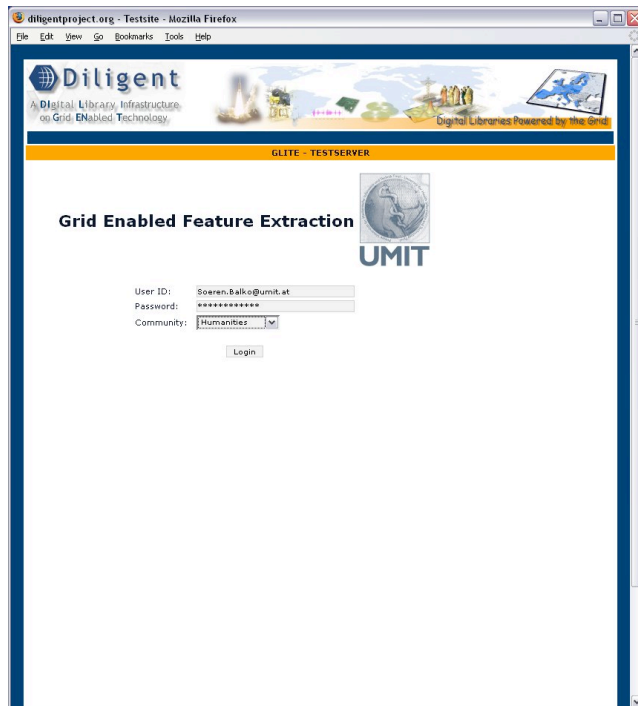
### 3.3.3 Target Audience

This demonstrator is mainly addressed to following target groups:

- Decision makers (performance improvements in batch processing)
- ARTE User Community (addressing of CBIR use cases)

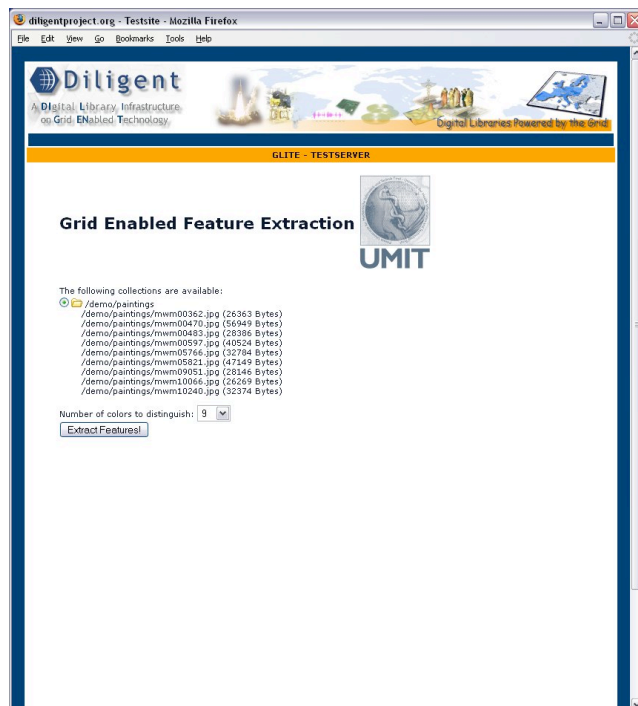
### 3.3.4 Where to find it and how to use it

The first version of this demonstrator has been deployed at UMIT and is currently not more available. The following screenshots show the process to start the feature extraction.



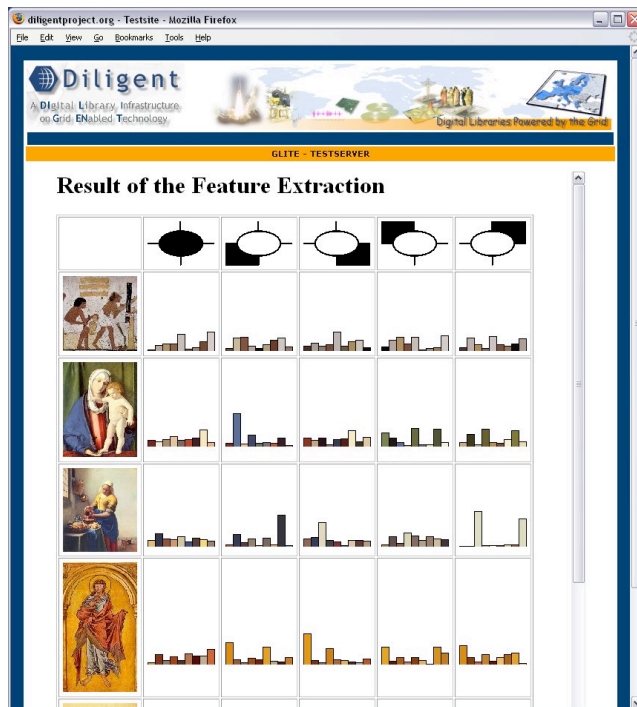
## Step 1

- ◆ Login into the demo
- ◆ Select from two user-community-related image collections (*earth sciences* and *humanities*)



## Step 2

- ◆ View catalog entries for selected image collection
- ◆ Parameterize feature extractor by specifying number of colors
- ◆ Start workflow



- Step 3
  - ◆ Inspect results (5-region color signatures)

Since UMIT is no longer a partner of the DILIGENT project, this version of the demonstrator can no longer be provided using their infrastructure. For this reason the demonstrator is currently unavailable. The second version of the demonstrator will be deployed at UNIBAS before the end of March 2007.

### 3.4 Demonstrator 4: User management and Security mechanism in Rendering

This demonstrator aims at testing animation rendering, a computational intensive process, on a secure grid middleware. Rendering is the process of generating an image from a model, by means of computer programs. The model is a description of three dimensional objects in a strictly defined language or data structure. It contains geometry, viewpoint, texture and lighting information. The image is a digital image or raster graphics image. Animation rendering is the process of calculating effects in a video editing file to produce final video output. As this process is really CPU consuming, the centralized approach cannot solve the problem.

In this Demo the animation to be rendered is split into fragments, each one constituted by some frames. Fragments are rendered concurrently on different grid nodes. When all fragments are rendered they are merged all together to get the complete animation. This demonstrator aims also at showing functionalities of the DILIGENT security model described below. It allows to face Authentication and Authorization issues on a grid environment.

The DILIGENT Security Model mainly deals with two different aspects:

- **Authentication:** Authentication support deals with user identity and credentials management issues. DILIGENT Authentication model is based on Grid Security Infrastructure. This implies the adoption of a Public Key Infrastructure (PKI) and the usage of X.509 End Entity Certificates (EEC) released by a Certification Authority (CA) in order to authenticate DILIGENT users.

- **Authorization:** DILIGENT authorization is based on the DILIGENT VO model reported in the diagram below.

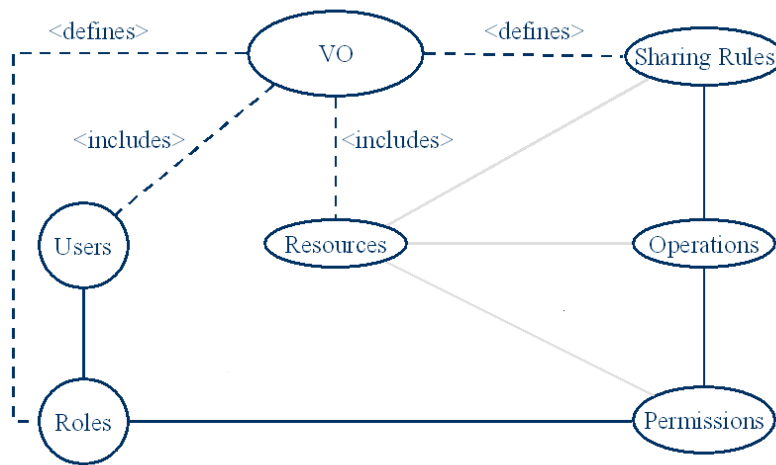


Figure 8 - DILIGENT VO Model

A VOMS VO containing all users and roles backs the DILIGENT VO. The VOMS server is used to create signed Attribute Certificates (AC) to be included in the Proxy Certificate of DILIGENT users. AC contains roles currently held by the user.

The DILIGENT authorization service is in charge to manage associations between roles and permissions in DILIGENT. When a service request is received, the DILIGENT Authorization service is queried to confirm that one among the roles held by the user are entitled to perform the requested operation.

The demonstrator is executed in a scenario where it is possible to choose between two different users having each one a different role. Depending on the identity you used to log on the demo, you'll be able to perform different operations.

### 3.4.1 Demonstrator design

The *User management and Security mechanism in Rendering* is developed on GridSphere Framework Portal that provides an open-source portlet based Web portal. The portal contains all portlets needed to manage renderings and to summarize security aspects of the demo based on the DILIGENT Security Model.

The demonstrator architecture is basically made up of following components

- *Demonstrator GUI:* The GridSphere application containing the graphical user interface of the Demonstrator deployed on an Apache TOMCAT installation.
- *Rendering Service:* A WSRF service able to submit and manage rendering jobs to the grid middleware deployed on a Java WS Core Node.
- *Workload Management System:* a gLite service in charge of receives and dispatch jobs to the grid, deployed in a gLite Node.
- *MyProxy server:* it is in charge to store credentials of users registered in the DILIGENT VO
- *Authorization Service:* The DVOS service in charge to manage and verify authorizations for DILIGENT roles developed for the Collective Layer.
- *User&GroupManagement service:* The DVOS service in charge to manage users and groups for DILIGENT VO developed for the Collective Layer.

The demonstrator architecture is shown in the following diagram.

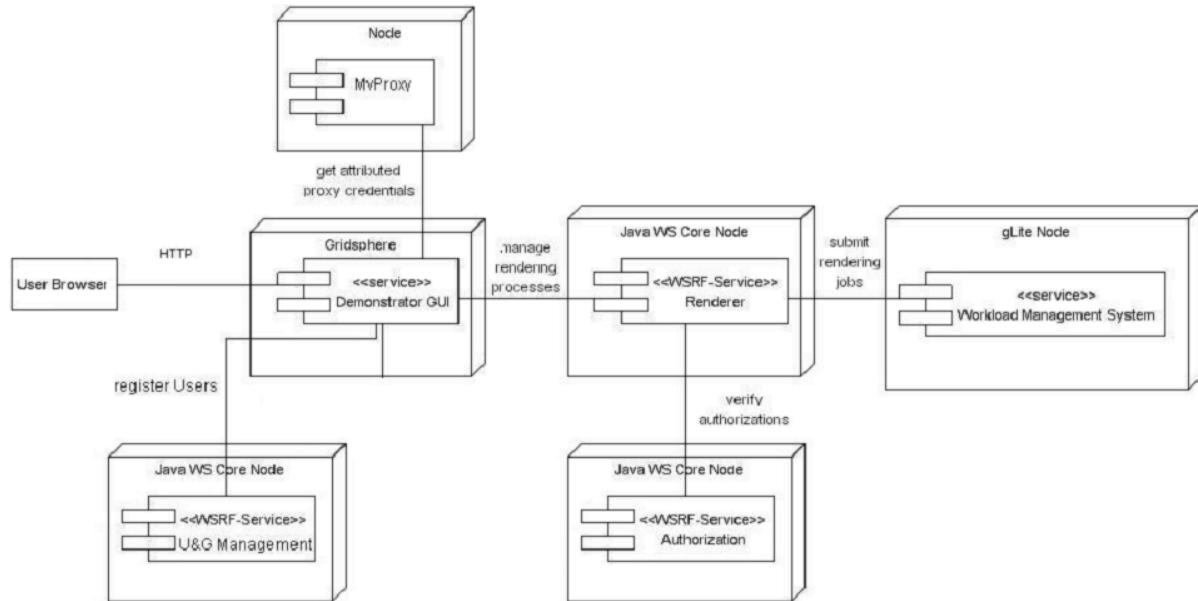


Figure 9 - User management and Security mechanism in Rendering deployment view

A user through his browser can contact the demonstrator user interface (Demonstrator GUI) and log on to the portal by using one of the users predefined. All the authentication aspects are managed by using a Proxy certificate retrieved from a MyProxy server that allows to the logged user to act on demo infrastructure.

An authenticated and authorized user can submit requests to a WSRF service (Renderer service) able to manage animation renderings. This service is able to submit jobs towards the gLite 3.0.0 middleware, in particular to Workload Management System (the gLite service in charge of receive and dispatch jobs).

The general behaviour of this demo is described by following steps:

- Each time the user login to the Demonstrator GUI, a proxy credentials containing his role is created automatically contacting the MyProxy server.
- Each time a new user is registered the User & Group Management service is invoked to add it to the DILIGENT infrastructure.
- The Authorization on the Rendering service is enforced using Authentication API and Authorization service of the DVOS component.
- Each time a new rendering is created, a corresponding WSRF resource is also created in the Rendering service.
- Once started, the resource is in charge to submit new rendering jobs to the Workload Management System (of the gLite 3.0.0 release) and to manage existing jobs until the animation is completed.

### 3.4.2 Overview of current development status

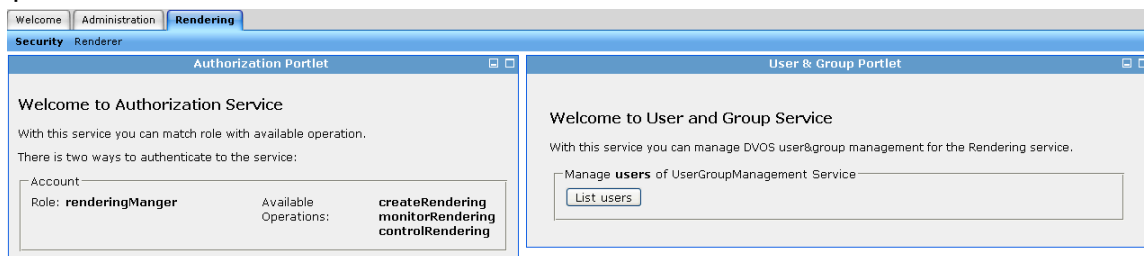
The service providing rendering management functionalities has been implemented as well as portlets in charge to access it. The GUI of the demonstrator is composed by two different groups of portlets: the first one is in charge to present the information provided by DVOS

services, while the second one allows managing the creation and the control of the rendering processes. Portlets created has been divided in two different pages: Security and Renderer.

In fact under the "Rendering tab" two pages are displayed:

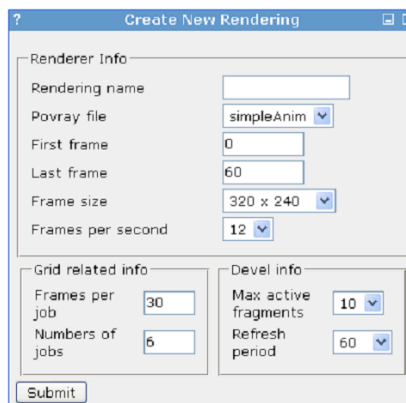
- The *Security* page representing the interface with the DVOS services. It contains two portlets:
  - Authorization portlet summarizing the user information about his current role and authorized operations on that portal, i.e. create a new rendering, monitor an existing rendering.
  - User and group portlet reporting the current users list registered to User & Group management DVOS service and that can be authenticated by the portal.

Figure below shows the graphical user interface provided by the demo portal for security aspects.



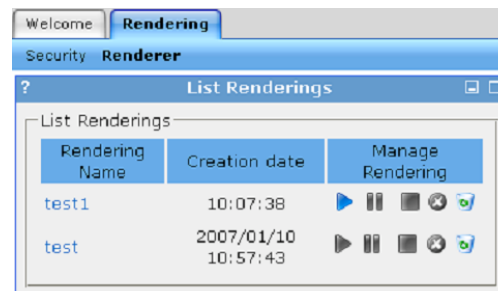
January 29, 2007

- The *Renderer* page represents the area enabling the user to manage all the aspects related to rendering. Depending on the identity used to perform the demo, the user will be able to perform different operations. If access with "guest" role, the user can only monitor renderings, on the other hand, if user is logged as "rendering\_manager" role, he will be able to create, manage and monitor renderings. This page contains three portlets:
  - Create new rendering portlet: this portlet allows submitting new renderings. To create a new rendering some information must be provided. In that portlet you have to pass some parameters required by Povray rendering engine, like frames per second, and some others required by gLite infrastructure, like frames per job.

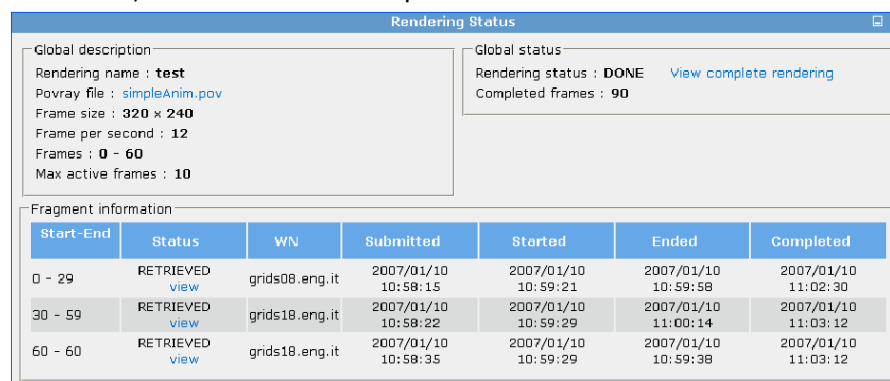


- List renderings portlet: this portlet allows user to monitor and, if he is an authorized user, to manage existing renderings, managed by the WSRF service. Clicking on each rendering name, detailed information will be shown on the Rendering Status portlet on the right of the page. On Manage rendering column portlet provides all actions supported for a

rendering and managed by this portlet, like start, pause, stop, abort, delete rendering.



- Rendering status portlet: this portlet shows the detailed information of a selected rendering (on List renderings portlet). It contains three sections:
  - **Global description** contains the summary of rendering settings selected during its creation;
  - **Global status** contains principal information about selected rendering;
  - **Fragment information** shows detailed description of every fragment: its current status, WN where it was assigned, and time when it is submitted, it started, it ended and it is completed.



### 3.4.3 Previous experiences and Target Audience

The Demo is mainly addressed to:

- Developers: interested in evaluating design and implementation solution as well as technologies used to create the demo.

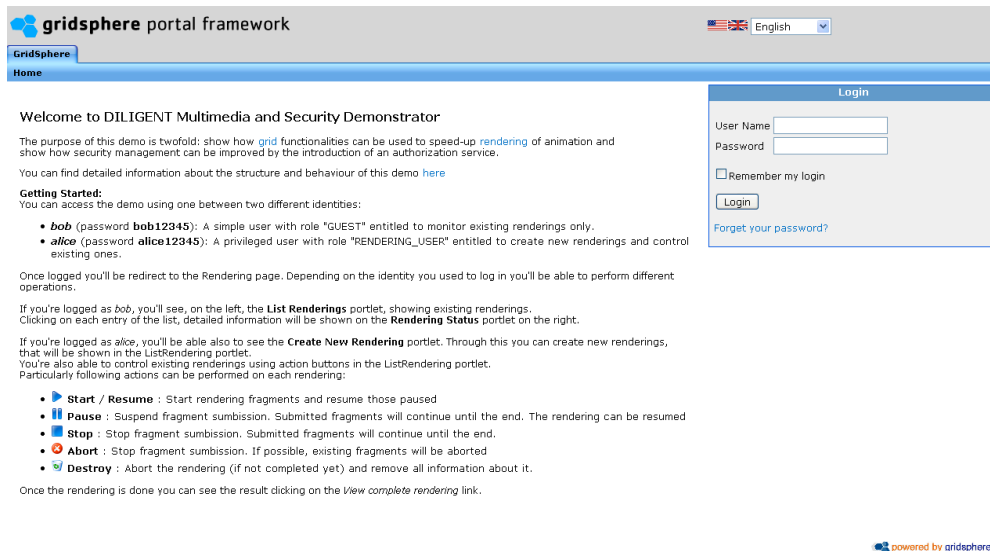
It relies on a demonstrator infrastructure equipped with the gLite 3.0.0 middleware and DVOS services.

### 3.4.4 Where to find it and how to use it

The demonstrator can be accessed at <http://grids16.eng.it/gridsphere>.

In the homepage a link points to a description of functionalities of this demonstrator.

The access to the portal is guaranteed to authenticated users registered to diligent VO.



**gridsphere portal framework** English

GridSphere  
Home

Welcome to DILIGENT Multimedia and Security Demonstrator

The purpose of this demo is twofold: show how [grid](#) functionalities can be used to speed-up [rendering](#) of animation and show how security management can be improved by the introduction of an authorization service.

You can find detailed information about the structure and behaviour of this demo [here](#)

**Getting Started:**  
You can access the demo using one between two different identities:

- **bob** (password **bob12345**): A simple user with role "GUEST" entitled to monitor existing renderings only.
- **alice** (password **alice12345**): A privileged user with role "RENDERING\_USER" entitled to create new renderings and control existing ones.

Once logged you'll be redirect to the Rendering page. Depending on the identity you used to log in you'll be able to perform different operations.

If you're logged as *bob*, you'll see, on the left, the **List Renderings** portlet, showing existing renderings. Clicking on each entry of the list, detailed information will be shown on the **Rendering Status** portlet on the right.

If you're logged as *alice*, you'll be able also to see the **Create New Rendering** portlet. Through this you can create new renderings, that will be shown in the ListRendering portlet. You're also able to control existing renderings using action buttons in the ListRendering portlet. Particularly following actions can be performed on each rendering:

- ▶ **Start / Resume** : Start rendering fragments and resume those paused
- ⏸ **Pause** : Suspend fragment submission. Submitted fragments will continue until the end. The rendering can be resumed
- ⏹ **Stop** : Stop fragment submission. Submitted fragments will continue until the end.
- ⏹ **Abort** : Stop fragment submission. If possible, existing fragments will be aborted
- 🗑 **Destroy** : Abort the rendering (if not completed yet) and remove all information about it.

Once the rendering is done you can see the result clicking on the [View complete rendering](#) link.

powered by gridsphere

Figure 10 - User management and Security mechanism in Rendering Homepage

The demonstrator infrastructure is provided by Engineering.

## 3.5 Demonstrator 5: Content-based Multimedia Retrieval

The demonstrator shows the integration of existing Grid-enabled technology for **parallelized Content-Based Image Retrieval (CBIR)**.

It illustrates the **search effectiveness** using sophisticated similarity measures, comprehensive query models, and expressive feature types.

It achieves **querying performance** through scalable parallel query execution on the Grid, using efficient index structures.

### 3.5.1 Demonstrator design

Like demonstrator 3, this demonstrator will be made available in two versions, where only the second will entirely use DILIGENT components. The initial design is as follows:

■ Components

- ◆ **Web Interface** (JSP) provides user interaction facility for query formulation and result presentation
- ◆ **Query Workload Partitioner** (Java) splits query into sub queries processing disjoint partial data stocks
- ◆ **Retrieval Servers** (Java) accept incoming sub queries and pass back partial results

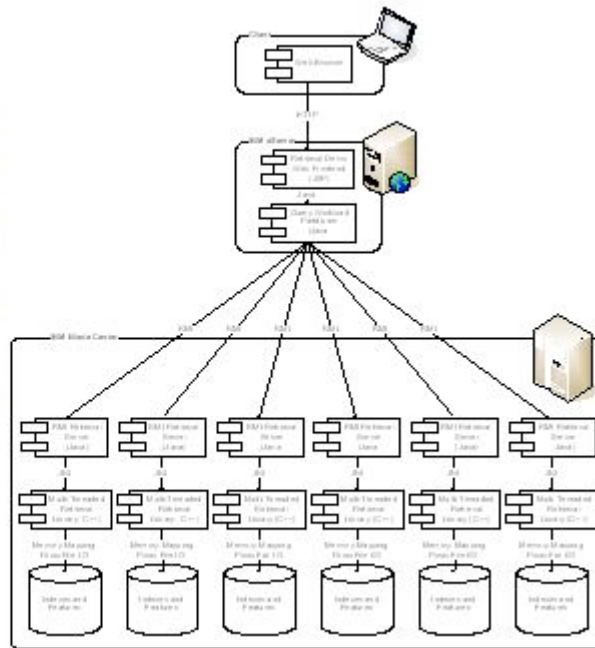


Figure 11 - Content-based Multimedia Retrieval Design

### 3.5.2 Overview of current development status

The first version of the demonstrator has been made available before the first review (3<sup>rd</sup> quarter '05), using some non-DILIGENT components. The graph shows the results of a simple evaluation performed using a set of about 100,000 colour images from previous projects.

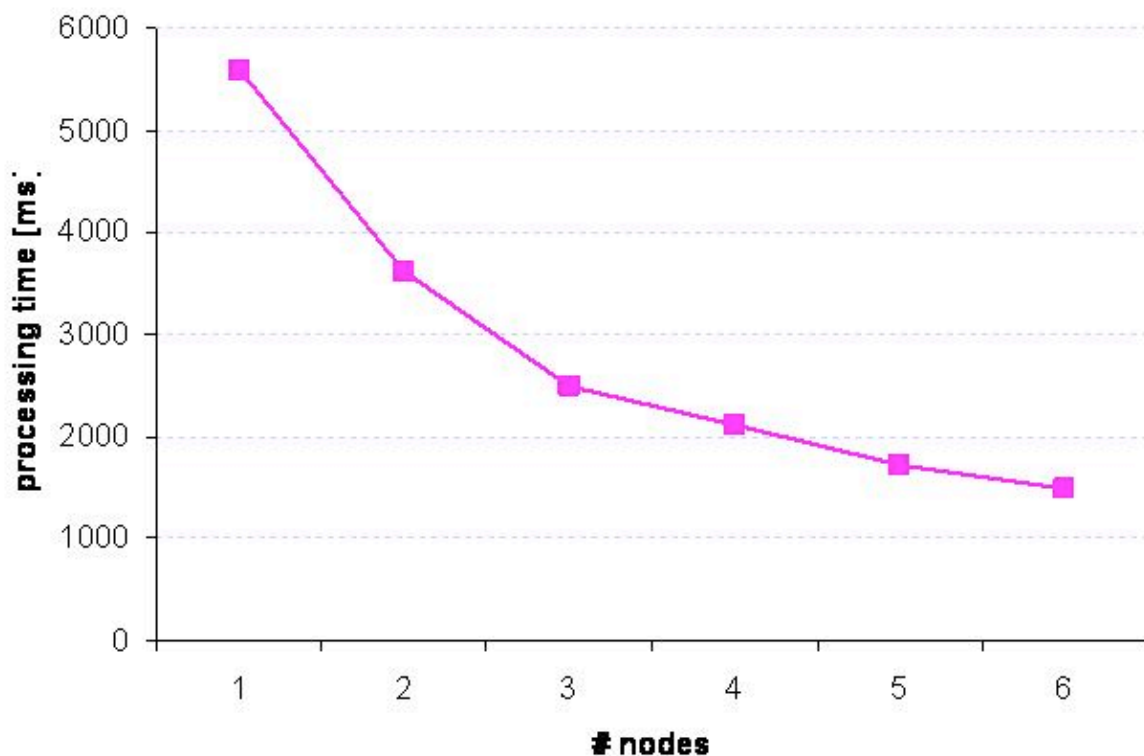


Figure 12 - Content-based Multimedia Retrieval Statistics

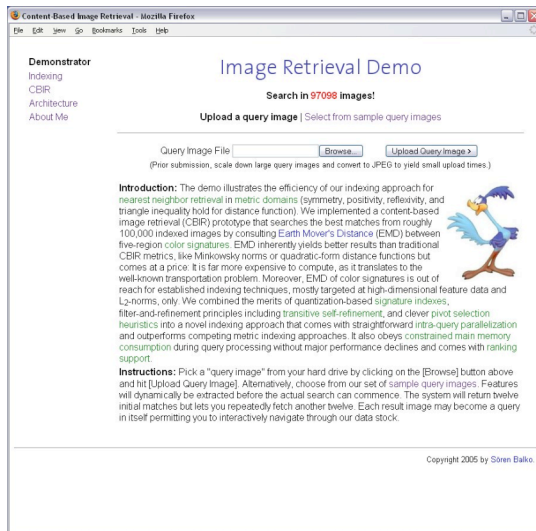
### 3.5.3 Target Audience

This demonstrator is mainly addressed to following targets:

- End users (illustrate virtues and potential of CBIR)

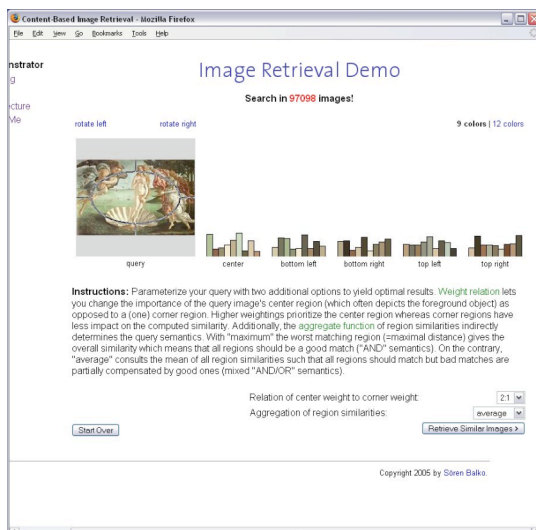
### 3.5.4 Where to find it and how to use it

The first version of this demonstrator has been deployed at UMIT and is currently not available. The following screenshots show the process to start querying the collection with some image from the collection or a file, which is uploaded from the local drive to the server.



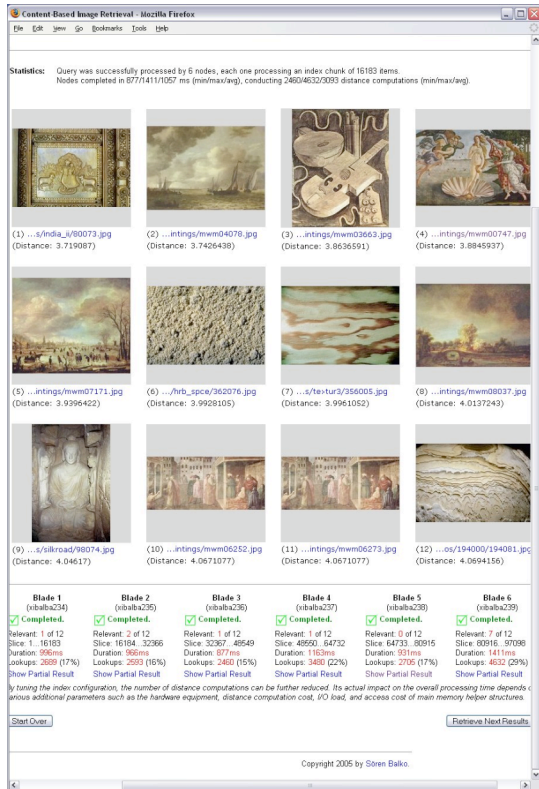
#### Step 1

- ◆ Query image can be uploaded from a local file
- ◆ Alternatively, users may select from pre-selected query images
- ◆ Start feature extraction of query image



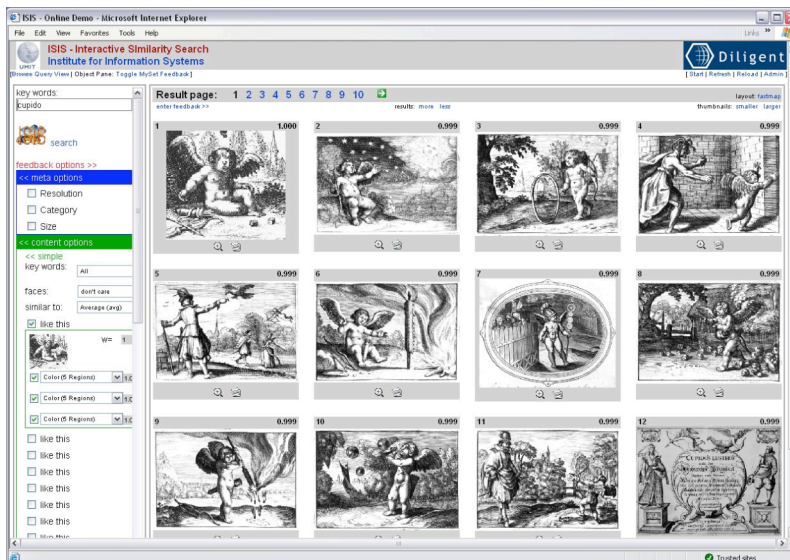
#### Step 2

- ◆ Inspect query image features
- ◆ Switch between different feature granularities
- ◆ Parameterize query (weightings and aggregates)
- ◆ Start query processing

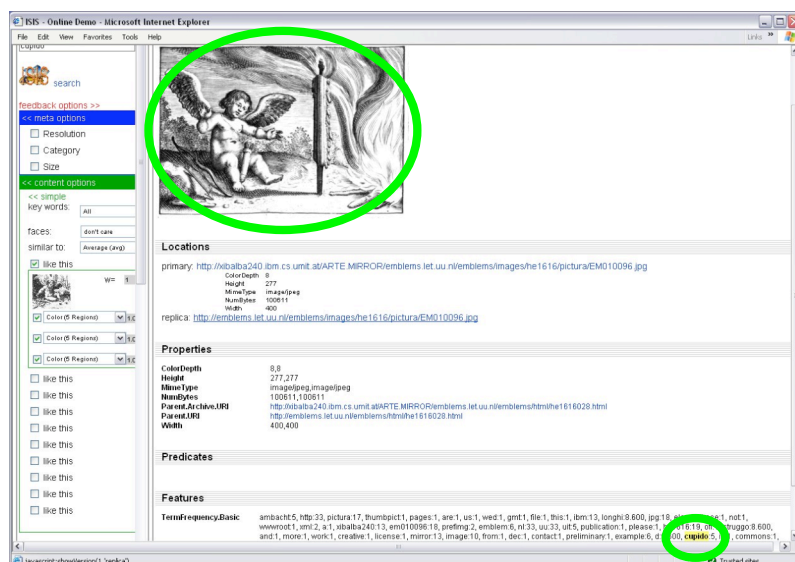


- Step 3
  - ◆ Inspect query results
  - ◆ Fetch subsequently ranked results
  - ◆ Pick alternative query image

The next screenshots show a different aspect of the demonstrator, which is, the expressiveness of queries using textual queries combined with image similarity search.



- Complex queries of multiple weighted feature types (*color, texture*) and exact term matches
- Example: color distribution and exact match on key word “cupido”



- Result documents contain key word “cupido”, approximately match color (*black/white*) and texture (*drawing*)
- High-level image semantics is appropriately supported by low-level features and textual annotation

### 3.6 Demonstrator 6: Dynamic Infrastructure Deployment

This demonstrator aims to prove that the co-deployment task of an infrastructure of interoperable services, appropriately configured, is feasible.

The demonstrator is based on a simplified scenario to avoid unnecessary complexity. In particular, it considers that:

- a) A set of Web Services (WS) tailored to provide digital library functionalities are described in and their properties are available through the Information Service;
- b) Some of these Web Services are configurable with respect to the Collections they acts on;
- c) There exist a number of known dependencies among the registered Web Services constraining their deployment/co-deployment on hosting nodes.

The demonstrator will be executed in the following scenario: through the Demonstrator portlet it is possible to select the pool of services, the pool of collections, and the pool of hosting nodes constituting the infrastructure the user is interested in. The demonstrator, by exploiting the Package Repository features, identifies all the mandatory services/packages and organises them in a simplified DL Map. Despite such map is not optimised, e.g. does not reduce the number of needed DHNs, it assures the compatibilities among the services allocated on the same DHN while preserves their package dependencies. The status of this complex deployment task can be monitored by means of a pop-up window reporting the deployment stages the various services (Deploying, Deployed, Failed) and their related packages are (Deployed, Failed). Thanks to this graphical tool it is also possible to perceive the additional services, respect to the selected ones, that have automatically been selected by the Demonstrator in order to properly run the selected services. Finally, the effective

creation of the so created infrastructure can be verified by accessing the Information System and discovery the service instances.

### 3.6.1 Demonstrator design

The demonstrator has been designed to exploit the capabilities provided by (i) the Package Repository, to download the appropriate software packages and discover their dependencies, (ii) the Information System, to discover and monitor the available resources, (iii) the DL Management, to orchestrate the deployment process, and (iv) the HNM, to implement the deployment task in a decentralised and distributed modality.

The demonstrator architecture is mainly composed by two components: the demonstrator user interface (DDLMapIdentification-Portlet) and the demonstrator logic (DDLMapIdentification-Logic), see Figure 13.

The DDLMapIdentification-Portlet exploits the Google Web Toolkit (GWT) to provide a comprehensive user interface enabling the user to select the resources forming the infrastructure from the list of available Diligent resources, to ask for the effective creation of such infrastructure, and to monitor the activity performed to satisfy the request by means of a monitoring user interface.

The DDLMapIdentification-Logic interacts with the Package Repository by means of the appropriate software libraries. It performs a query against the DIS via the DIS-HLS Client library, manages an internal cache, performs the streaming of the resources, and hides to the DDLMapIdentification-Portlet all the complexity related to the DILIGENT framework.

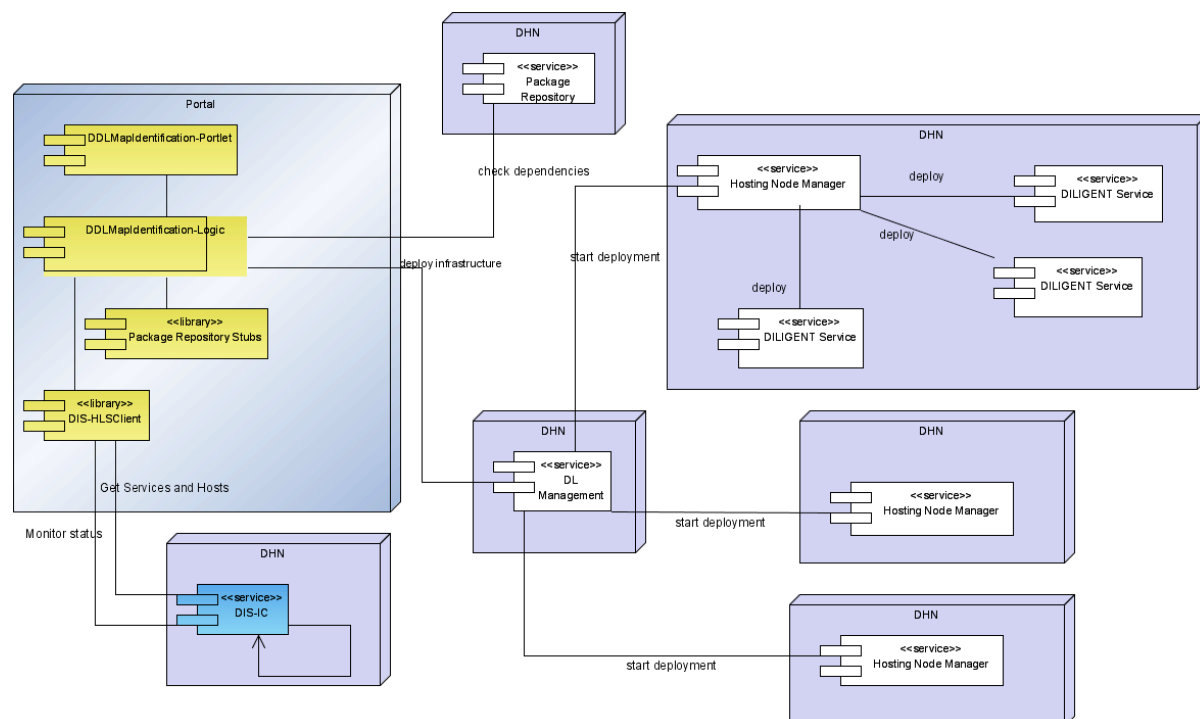


Figure 13 - Demonstrator Dynamic Infrastructure Deployment Architecture

### 3.6.2 Overview of current development status

The Dynamic Service Instance Deployment demonstrator has been completely developed. The DDLMapIdentification-Logic has been implemented and the interaction with the other components and services has been finalized.

The visualization of DILIGENT resources has been implemented through the DSIS-Portlet as well as the mechanisms allowing selecting them and starting the deployment task.

Figure 14 shows the graphical user interface provided by the DSIS-portlet. Three main areas characterize it:

1. The Deployment Panel represents the main console of this demonstrator. It allows (i) to select the category of resource to browse, i.e. services, hosting nodes, and collections, and (ii) to start the deployment task. Moreover, it reports the current selected items involved in the deployment task to help the user in its activity, i.e. it reports the name of the selected services, hosting node and collections that, once clicked, provides their profile in the Profile Visualizer panel.
2. The Resources Browser represents the area enabling the user to browse the resource instances of a certain type, e.g. service, hosting node, collection. It shows the distinguishing feature of each resource instance, e.g. its identifier, its name, and allows to navigate through this potentially huge list of entries via an easy to use and understand mechanism based on previous/next page.
3. The Profile Visualizer represents the panel reporting detailed information on the resource currently selected in the Resources Browser panel on in the Deployment Panel, i.e. it is in charge to display the profile describing a service or an Hosting Node.

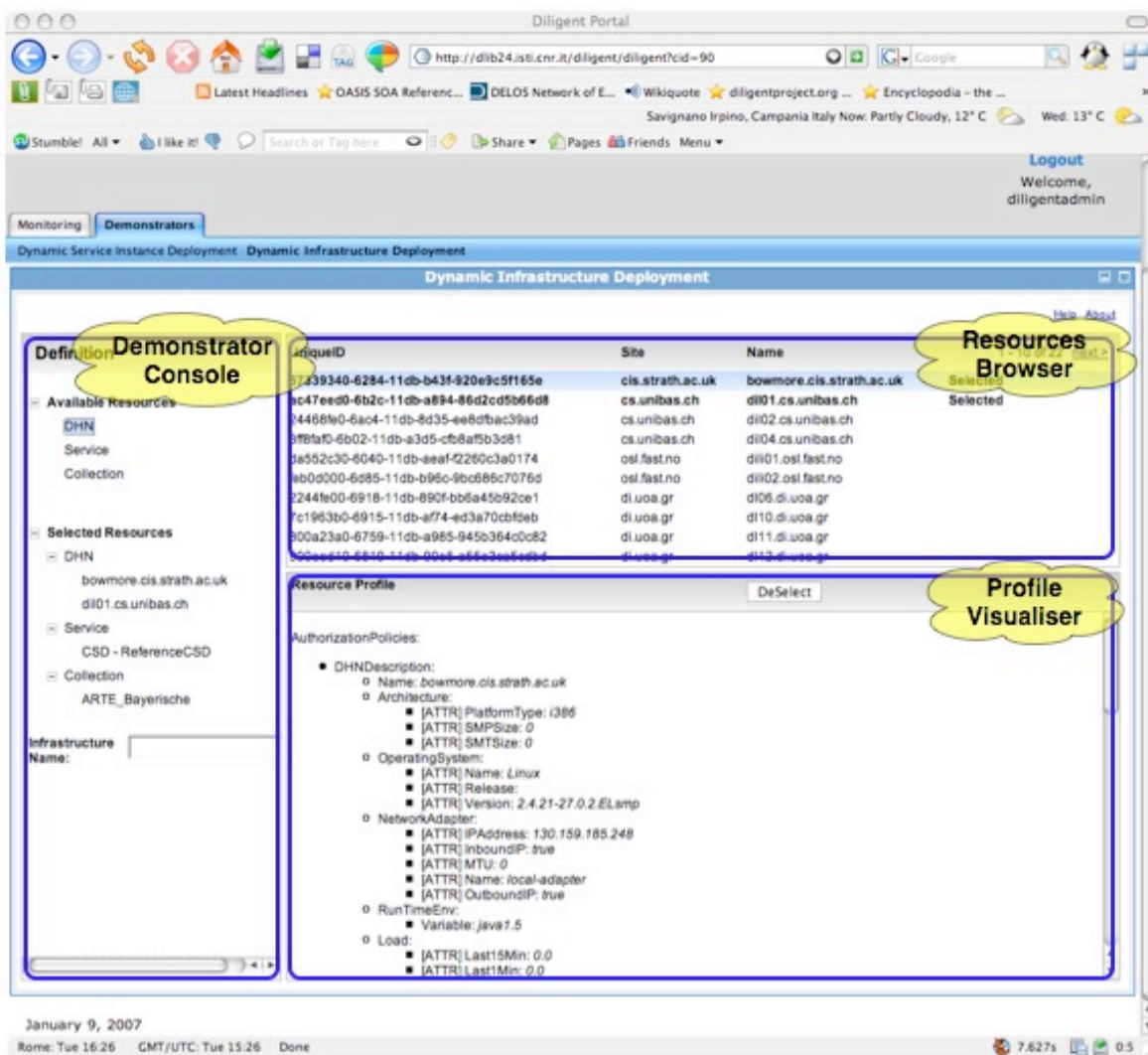


Figure 14 - Demonstrator Dynamic Infrastructure Deployment Graphical User Interface

### 3.6.3 Target Audience

The demonstrator is mainly addressed to developers of any target groups. It aims at demonstrating that (i) well described Web Services can easier be selected to satisfy the needs of specific user communities and (ii) a complex Service Oriented Architecture can automatically and easier be designed, deployed, and put in operation by using the services of the DILIGENT Collective Layer.

### 3.6.4 Where to find it and how to use it

The demonstrator can be accessed at <http://ddportal.isti.cnr.it/diligent/diligent>.

The infrastructure is currently provided by CNR-ISTI. However, a DILIGENT demonstration infrastructure or a third-parties infrastructure, e.g. the Gilda infrastructure, should be adopted to ensure the proper number of resources.

This demonstrator will be linked from the project web site and it will be accessible through the demonstrator portal. The access to the portal will be guaranteed to authenticated users. The registration process is out of the scope of the demonstrator and therefore it relies on the native authentication mechanisms provided by GridSphere portal engine.

## 4 CONCLUSIONS

The present paragraph represents an interim evaluation of the activities performed until M28 in this workpackage. It also provides useful suggestion for future assessment of the workpackage activities.

### 4.1 Evaluation of Process & Results

Demonstration activities aim at proving the viability of developed technologies that offer a potential economic advantage to the target communities, but which cannot be commercialized directly. In the first two years of the project those communities have been studied and following their potential interests some demonstrators have been designed and developed.

The resources so far allocated to these activities were not well distributed and impede to provide direct access to final users. The filter made by Communities representatives (including RAI and EGEE) allowed to effectively use of the resources and, at the same time, reduce the details on real expectations of final users.

All completed demonstrators can be accessed using URL contained in the demonstrators' descriptions. In some cases the access to some functionality has been restricted for two reasons. First of all they require an in-depth knowledge of the demonstrator itself and of the infrastructure where it is running; secondly they deeply alter the state of the infrastructure itself, so the execution of these functionalities cannot be performed arbitrarily by users. An example of this approach is the demonstrator 1 "*Dynamic Service Instance Deployment*". It can be performed for interested audience during DILIGENT presentations, as it needs to be performed with some help from the DILIGENT developers. Despite these limitations, demonstrators allow users to experiment core functionalities provided by the DILIGENT infrastructure.

Some difficulties to create demonstrators have been faced trying to interface them with the gLite 3.0.0 grid middleware. These difficulties are mainly related with different versions of libraries used by Java-WS-Core and the gLite API. These issues have been solved using temporary workarounds; an example is given by the demonstrator 4 "*User management and Security mechanism in Rendering*". In that case the gLite command line used to submit jobs has been wrapped and invoked using the Java Native Interface. These solutions are valid only for demonstrators, but more stable and clean ones need to be found for the final DILIGENT infrastructure.

Some difficulties have also been faced using the GridSphere portal engine capabilities, these are mainly related with the lack of documentation of the portal functionalities and with a high number of bugs discovered and notified to the GridSphere team.

Despite these difficulties, all planned demonstrators have been completed with the exception of demonstrators 3 "*Batch Feature Extraction*" and 5 "*Content-based Multimedia Retrieval*". For the first one an update to the gLite 3.0.0 release of the middleware is required as well as the integration with the GridSphere portal, for the second one contents and graphics need to be integrated with the portal. All other demonstrators are available to the users, currently deployed on infrastructures of partners that developed them. The integration among different demonstrators could be improved to ease the access and usage of them from the users' point of view. This integration will be the main objective of future plans of the WP 4.4 workpackage.

The plan adopted so far, based on the concept of reuse of software components as soon as they were ready from technical workpackages, was not so efficient. It resulted in continuous delays in demonstrators' release.

The future work will be devoted to the consolidation of the current demonstrators and the realisation of a common point of access for all of them, homogenise the accesses and provide self-contained information to be used through the DILIGENT portal.

## References

- [1] DILIGENT Description of Work
- [2] D4.3.2 Market and Technology trends Analysis
- [3] The ARTE project – <http://www.sns.it>; & DILIGENT DoW, part B, p. 55