



## TASK 3.5 SRS DOCUMENT

PUBLIC

### Test and Integration

---

Document Filename:	<b>CG-3.5-SRS-0002 Public.doc</b>
Work package:	<b>WP3 New Grid Services and Tools</b>
Partner(s):	<b>PSNC, CYFRONET, UAB, CSIC</b>
Lead Partner:	<b>CSIC</b>
Config ID:	<b>Task 3.5 SRS Public document</b>
Document classification:	<b>Public</b>

---

Abstract: This document described the WP3 Software Integration, test and deliver Process, focussing in particular on first milestone is in month 12.



### Delivery Slip

	Name	Partner	Date	Signature
<b>From</b>				
<b>Verified by</b>				
<b>Approved by</b>				

### Document Log

Version	Date	Summary of changes	Author
1.1 DRAFT	5 April 2002	Draft version	Santiago González de la Hoz
1.2 Final version	7 May 2002	First version	Santiago González de la Hoz
1.3 Final version	30 May 2002	Final version	Santiago González de la Hoz

---

## CONTENTS

<b>1. INTRODUCTION.....</b>	<b>4</b>
6.1.1. OBJECTIVES OF THIS DOCUMENT.....	4
7.1.2. APPLICATION AREA.....	5
8.1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS.....	5
9.1.4. REFERENCES.....	6
10.1.5. OVERVIEW.....	6
<b>2. TESTS AND INTEGRATION PROCESS.....</b>	<b>7</b>
6.2.1. PREPARATION.....	7
7.2.2. TASK3.5 RESPONSIBILITIES.....	7
8.2.3. SOFTWARE TASKS RESPONSIBILITIES.....	7
9.2.4. INTEGRATION TEAM RESPONSIBILITIES.....	8
10.2.5. INTENSIVE INTEGRATION.....	8
11.2.6. SEMI-PRODUCTION.....	9
12.2.7. PREPARATION OF SCHEDULE.....	9
14.2.8. PLATFORM SUPPORT.....	10
<b>3. COMPOSITION OF INTEGRATION TEAM.....</b>	<b>11</b>
<b>4. SUPPORT.....</b>	<b>12</b>
6.4.1. USER SUPORT.....	12
7.4.2. SITE ADMINISTRATOR SUPPORTS.....	12
<b>5. APPENDIXES.....</b>	<b>13</b>
6.5.1. INTEGRATION TEAM MEMBERS .....	13
7.5.2. WP3-PROVIDED RESOURCES.....	13
8.5.3. RESOURCES NEEDED FOR THE INTENSIVE INTEGRATION PHASE (SITE KARLSRUHE).....	15
9.5.4. SOFTWARE-PROVIDE RESOURCES.....	16

## 1. INTRODUCTION

### 6. 1.1. OBJECTIVES OF THIS DOCUMENT

The main goals of the tests and integration are to deliver an official prototype version of the whole set of tools of the WP3 – middleware. This has to be done before each prototype to provide a functional computing platform for the application software using the CrossGrid testbed sites. Such a release will be a base for further development in other workpackages, especially for WP1 and WP2. Task 3.5 is very strongly correlated with the prototypes released in WP4 as is shown on Fig.1.

Task 3.5 will deliver all information about possible errors detected while the system tests and required corrections which should be done, either as including changes in the architecture or the API interfaces between the WP3 tools and other modules (WPs). The relations between Task 3.5 and other WPs have been presented on Fig.1. The middleware software official version prepared by Task 3.5 will be used by other WPs, especially the WP1 and WP2. The tools developed in Tasks 3.1 and 3.4 will be used directly by the applications (WP1) under the scope of optimised data access (input data and results of computations) and simplified user access to each of the applications deployed in WP1. Task 2.4 will use the Grid Monitoring tool from Task 3.3. The testbed supported by WP4 will be used in Task 3.5 as a testing environment, allowing testing separate tools (Tasks: 3.1,...,3.4) as well as the whole prototype of WP3 (middleware).

The four partners of this task: CYFRONET, PSNC , CSIC and UAB will test all tools and services of WP3 to be able to deliver a middleware layer in the CrossGrid testbed, which will be used by the applications (WP1) and the Grid Application Programming Environment (WP2). The integration phase is strictly dependent and correlated with WP4 tasks and its time scheduling.

The integration process begins with preparation of the infrastructure (CVS server, RPM releases, Web server, etc.) by task 3.5 and ends with the beta–release of an integrated WP3 system by the Integration Team. Task 3.5 will concentrate on module tests and tests of dependencies between WP3 tasks. The integration phase must be done using the Testbed provided by WP4.

Because Prototype 0 will be the first time all of the software code will be integrated, the integration process for Prototype 0 will differ from that in later Prototypes, particularly in the preparation phase. Consequently, this version specifically targets the integration of software for the Prototype 0. It is a working document and will be updated for the later Prototypes.

The objective of this document is to describe these processes.

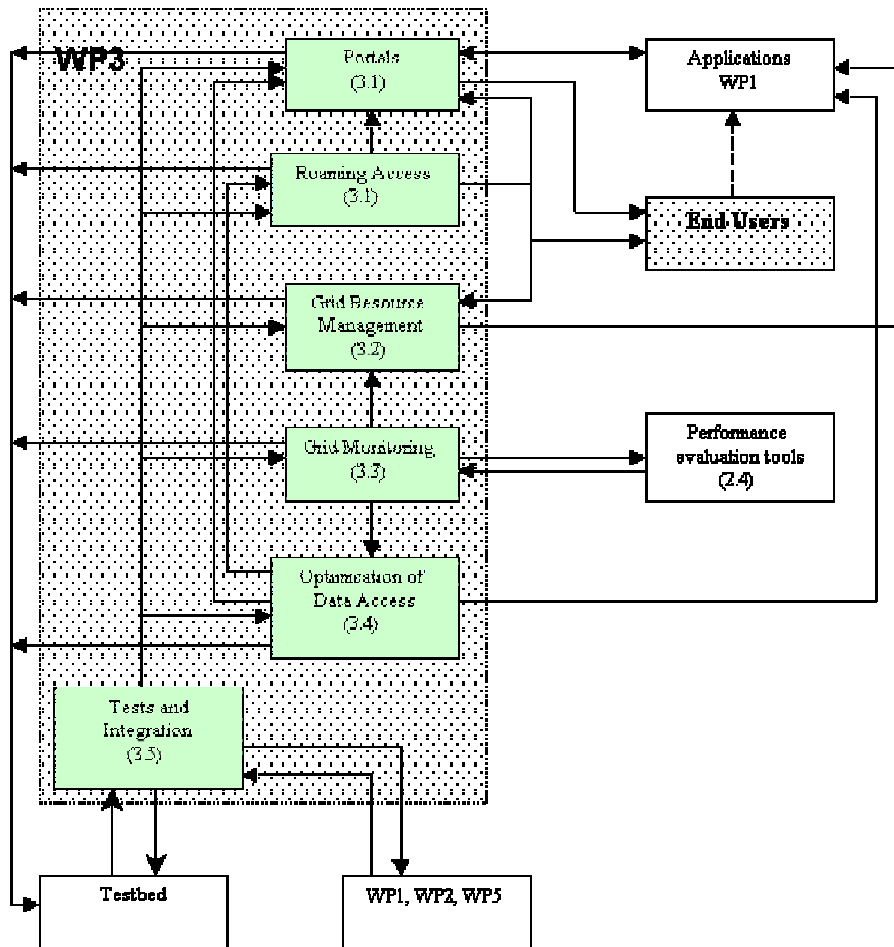


Figure 1 WP3 schema

## 7. 1.2. APPLICATION AREA

The document applies to the New Grid Services and Tools (WP3). It's audience is all those involved in the tests and integration process leading to delivery and subsequent validation of Prototype 0.

## 8. 1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

### Definitions

Integration	Combining WP3 software, DataGrid middleware and Globus to provide a functional computing platform for the application using CrossGrid testbed sites
Validation phase	A phase in which the design and implementation of the CrossGrid platform is tested, follows the integration.
Tests	Verifying the correctness of installation and configuration of each service or tool delivered by each task. Validating its basic functionality and checking its interaction with other software on which it depends.

---

Production	After passing the initial testing, the WP3 software will be ready for more widespread deployment. The WP3 software will be incrementally expanded to testbed sites.
Globus	The Globus Toolkit is a set of software tools and libraries aimed at the building of computational grids and grid-based applications.

### Acronyms/Glossary/Abbreviations

CVS	Concurrent Versions System
API	Application Programming Interface
IT	Integration Team
MPI	Message Passing Interface
RPM	Red Hat Package Manager

## 9. 1.4. REFERENCES

- [1] Testbed Software Integration Process, DataGrid-6-D6.1-0101-3.3  
<http://marianne.in2p3.fr/datagrid/documentation/>
- [2] Testbed Software Test Plan, DataGrid-6-TED-0111-0\_5  
<http://marianne.in2p3.fr/datagrid/documentation/>
- [3] Minimum Hardware Requirements, WP4  
<http://www.cyfronet.krakow.pl/eucgc1/materials.html>
- [4] Detailed Planning for Testbed Setup, Crossgrid D4.1-doc-0001-1-0-DRAFT-C

## 10. 1.5. OVERVIEW

Section 2 provides an overall description of the tests and integration process, section 3 details the composition of integration team and section 4 provides user and administrator support from the Integration Team.

## 2. TESTS AND INTEGRATION PROCESS

The integration and validation process will require significant effort from all of the partners of this task. The following describes the effort and resources required from all tasks in WP3 at various stages of the integration process.

An initial process will be based in the schema adopted at the DataGrid project [1].

### 6. 2.1. PREPARATION

During Month 10, the Integration Team should prepare for Prototype 0 (Month 12). This section describes the responsibilities of the Task3.5 funded effort, of the software tasks (task3.1–task3.4), and of the Integration Team. In Month 10, the Integration team will begin to integrate the available alpha–release software code with the Globus 2 version and the corresponding DataGrid middleware [2].

### 7. 2.2. TASK3.5 RESPONSIBILITIES

Task3.5 must provide the necessary infrastructure to support Prototype 0, including in particular, the software servers and the support system.

Task3.5 will provide a central, reliable, publicly accessible, backed–up CVS server. There will be a central CrossGrid repository in Karlsruhe and a backup copy for repositories will be located in Valencia. Read–write repository of WP3 will be localised in Poznan. This server will provide access to all of the WP3 software. Tasks in WP3 are invited to move their source repositories to this server at their convenience. The server will provide a read–only mirror of that work task package’s repository. For access to the CVS server WP3 members should email [miron@man.poznan.pl](mailto:miron@man.poznan.pl) for a password and further details.

Task3.5 will also provide a central HTTP repository for the packaged source and binary distributions of the WP3 software. The source distributions will contain the exact version of the code used to build the corresponding binary distribution. The application task packages may also use this repository to server their application kits.

Task3.5 will also provide a central website from which support information can be obtained. This includes contact information for Testbed site administrators, site user support personnel, applications support personnel, and the Integration Team. This site will also host the software used to track bugs in the CrossGrid or WP3 software.

### 8. 2.3. SOFTWARE TASKS RESPONSIBILITIES

The deliverables from each of the software tasks is expected in Month 10. At the latest, the source code from each software task must be available to all CrossGrid collaborators, particularly the Integration Team, at the same time. Each partner should test well its own software before releasing the code to the Integration. It is very important to guarantee that all interfaces will be defined correctly.

The software tasks must provide the detailed specifications for any public package they provide or require. The term "public package" refers to any API, protocol, service, etc. Intended to be used by other task software or by applications.

If the task software requires a specific operating system, architecture, compiler, etc., the task must provide this information.

## 9. 2.4. INTEGRATION TEAM RESPONSIBILITIES

The Integration Team should follow the progress of each task in WP3 and develop a fallback solution if the software cannot be delivered in time for Prototype 0.

The preparation of the infrastructure includes an evaluation of Globus packaging scheme and DataGrid software. This is clearly a vital evaluation with serious consequences for the success of the Testbed and Prototype 0. The Integration Team will do this evaluation before the end of Month 10.

One of the tasks of the integration team to define package names, in accordance with other tasks and WP's, to establish "standard" versioning rules among packages to be deployed with the CrossGrid software.

From the information obtained from the software tasks and from the needs of the applications groups, the Integration Team together the Technical Architecture Team should define the "reference platform (operating system, kernel, packages)". The reference platform is the one which will be used to build, integrate, and deploy the WP3 (CrossGrid) software. The complete specification of this includes the exact versions of the operating system, architecture, kernel, and needed external packages (from DataGrid).

The Integration Team must prepare a detailed schedule for when the CrossGrid testbed sites will be expected to install the WP3 (CrossGrid) software. This schedule should be available by mid-month 12.

## 10. 2.5. INTENSIVE INTEGRATION

The bulk of the tests and integration work takes place in this phases. It begin Month 10 and will continue for approximately 2 months.

To produce the first functioning beta-release of the WP3 software, all of the members of the Integration Team should be physically in the same place and have enough resources to build a small "grid". WP3 doesn't have this place to host this phase of the integration effort and providing the required resources. It is a point of later discussion according to the real needs. The appendix lists in detail the resources required.

During this phase, the Integration Team will continuously build, deploy, and test the WP3 software until a beta-release of sufficient quality is obtained. Each of the partners should have in place a minimum set of machines with the basic software packages (Globus, DataGrid,..) on top of which they'll develop their own tools (This would be a local testbed, month 6 in the project), but we need a testbed site to do the test and the integration, in order to put all WP3 software together, test it and deliver it. We will probably use the one delivered by WP4 (in Karlsruhe) and one or two people of the Integration Team should take part in integration phase, to be a "bridge" between WP3 and WP4.

The detailed integration process is:

- The reference platform will be installed and verified on all machines.
- Globus and DataGrid software installations should be tested to verify that the installations are correct. The configuration of this farm should represent typical examples from Testbed site.

- The WP3 and CrossGrid software should be installed from the CVS server. Once everything functions, the distribution packages should be made. Installation from these distribution packages must be verified.
- This version of the WP3 code is the initial beta-release of the WP3 software. The integration team should fully document the installation process for this beta-release. The Integration Team should consider a formal tutorial or training for the remote site administrators.

Note that the integration effort should be an open process allowing the participation of those not formally on the Integration Team. This includes particularly site administrators of CrossGrid Testbed sites.

## 11. 2.6. SEMI-PRODUCTION

After passing the initial testing, the WP3 software is now ready for more widespread deployment. The Integration Team will pair the additional remote sites. The site administrators will install and verify the WP3 (CrossGrid) beta-release using the WP (CrossGrid) software packages and documentation provided by the Integration Team.

## 12. 2.7. PREPARATION OF SCHEDULE

Time schedule assuming first milestone is in month 12

- Month 3 (31/05/2002):
  1. Definition of the software requirements (tools and services) for each task in WP3.
  2. Definition of the reference platform.
  3. CVS central CrossGrid in Karlsruhe and WP's servers available.
  4. HTTP servers available.
- Month 6 (31/08/2002):
  1. Co-ordination established with the other WP's and Grid projects.
  2. First local (in each site) testbed should be in place with the Globus and DataGrid's tools installed, where each task develops and testing their tools and services.
- Month 10 (31/12/2002):
  1. Collection of information from software tasks by the Integration Team.
  2. Software tasks should begin releasing beta-version of code to the Integration Team.
  3. Bug-reporting system in place.
  4. Reference platform install in Karlsruhe. The initial tests and integration must be done in WP4 TestBed.
- Month 12-15 (28/02/2003-31/05/2003):
  1. First prototype is delivered.
  2. Website is populated with support names and contacts.
  3. The integration team will provide documentation, training, and direct supporting for system administrators who are installing the WP3 software.
  4. Backup copy for repositories in Karlsruhe will be located in Valencia.

## 13.

---

## 14. 2.8. PLATFORM SUPPORT

The eventual goal of the grid is to make a large number of heterogeneous computing platforms at different sites available as a single computing platform. To accomplish this, the CrossGrid software will need to be as platform-independent as possible. However, most of the testbed sites and work packages should run Linux on an Intel architecture. As the principle motivation of Prototype 0 and 1 is testing the design and implementation of the CrossGrid software, the Integration Team will primarily support the CrossGrid software on Intel-based Linux platforms. This point should be discussed with the Technical Architecture Team and DataGrid project to have a common procedure.

### 3. COMPOSITION OF INTEGRATION TEAM

The detailed composition of this team should be determined via discussions with the task coordinators and WP3 leader. The following should guide the selection of the Integration Team members. (The appendix lists the members of the Integration Team for prototype 0).

- One member from each of the software tasks (task3.1–3.5) and for each partner of this task. These people will be the ones primarily responsible for building and modifying their task code during the integration process.

The team members should be able to cover the following needs:

- Authorisation management (certificates, gridmap files etc.)
- Write access to Task code.
- Management of core services (CVS server, distribution server, web management, bug-tracking, etc.)
- Co-ordination including management of tests, allocation of resources.
- Packaging expertise (in terms of task packaging and of packaging support).
- All aspects of Globus and DataGrid software operation.

It is a point of later discussion to the real needs have a local testbed to do the integration and test the WP3 software. We will probably use the one delivered by WP4. One or two people should be ready to spend approximately 1 month at Karlsruhe during the intensive integration phase to be a "bridge" between WP3 and WP4 The responsibilities of this team are:

- ♦ building the base WP3 (CrossGrid) release,
- ♦ maintaining and updating it, and
- ♦ supporting it.

It is explicitly not responsibility of this team to enforce coding rules and software management practices except to the point that interfere with the integration process. The integration team is intended to be a long-term body, continuing to maintain and support the WP3 software for the lifetime of the Testbed (Prototype 0 and 1). The members of the integration team must be able and willing to spend a significant amount of time on integration work through the production of the WP3 (CrossGrid) beta-release.

## **4. SUPPORT**

The Integration Team will begin to provide support to site administrators and to users when the WP3 software has first been released.

### **6. 4.1. USER SUPORT**

The WP3 user will have two primary contacts, the local system administrator and the application representative for operational problems and for application problems, respectively.

For bugs in the WP3 (CrossGrid) software, the user should submit a bug report. Karlsruhe will provide a bug tracking mailing list. The bug-report will assigned for a member of the Integration Team. If the Integration Team cannot resolve the problem, then the bug will be refereed to experts or appropriate work package.

### **7. 4.2. SITE ADMINISTRATOR SUPPORTS**

The integration team will provide documentation, training, and direct support for system administrators who are installing the WP3 (CrossGrid) software. Karlsruhe will provide a web-interface to central repository (Source-Forge based on CVS, with CVSweb inteerface and bug tracking mailing list, news and others). The installation documentation should be available at the same time as the beta-release of the WP3 (CrossGrid) software. It should be actively maintained and improved given feedback from the site administrators who have used the documentation.

## 5. APPENDIXES

### 6. 5.1. INTEGRATION TEAM MEMBERS

For prototype 0, the Integration Team is composed of the following people:

- Pawel Wolniewicz (PSNC, Task 3.1)
- Miquel A. Senar (UAB, Task 3.2)
- TBD (Task 3.3)
- Lukasz Dutka (CYFRONET, Task 3.4)
- Santiago González (CSIC, Task 3.5)
- More people are needed to test WP3 software!!!!

### 7. 5.2. WP3-PROVIDED RESOURCES

WP3 should provide the following resources:

- A central, reliable, publicly accessible, backed-up CVS server to host (or mirror) all work package code, core Globus code, DataGrid software, and needed external packages (see figure 2).



Figure 2 Example of a CVS Repository (DataGrid CVS Repository)

- A central, reliable, publicly accessible, backed-up repository for the RPM releases of the WP3 (CrossGrid) software as is shown in figure 3. (This may be mirrored by other institutions).



Figure 3 Example of a RPM repository (DataGrid RPM Repository)

- A webserver which will contain (see figure 4)
  - a) Instructions for building the WP3 (CrossGrid) software, support contact lists for users and other team members,
  - b) Statements on overall policies of using the Testbed, and
  - c) The bug-tracking software.. Note that the bug-tracking will be done only for WP3 (CrossGrid) software and not the application software

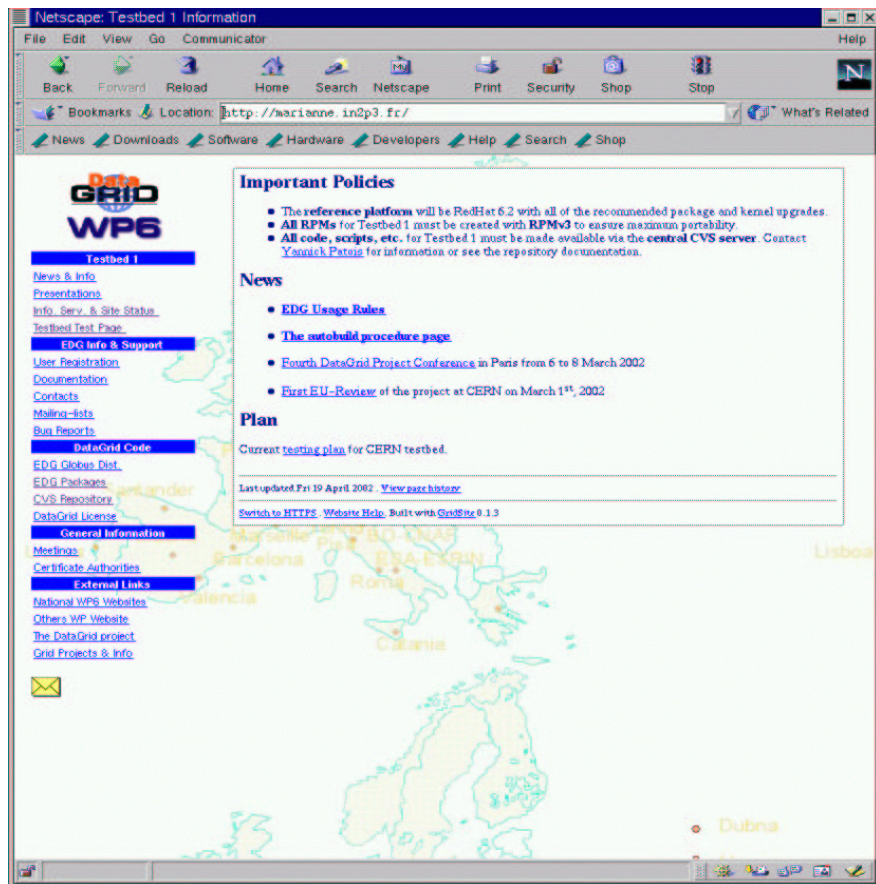


Figure 4 Example of a Webserver (DataGrid Webserver)

- A group for taking care of authorisation, certificates, etc.

## 8. 5.3. RESOURCES NEEDED FOR THE INTENSIVE INTEGRATION PHASE (SITE KARLSRUHE)

Testbed site will host the intensive integration phase. For Prototype 0, will provide the following resources [3] (see figure 5):

- 4 PIII>1Ghz, 512 MB RAM and 40 GB IDE disk, 100 MBs BasT
- 4 additional, similar machines
- Space for housing these machines and space for the integration team to work. The workspace should include fast network connections for team's terminals/portables.
- Fast networking between these machines and to the external network. These machines must also be externally visible.
- IT support for the provided hardware.

### *Requirements from WP4 (TESTBED SETUP)*

The official prototype version of the whole set of tools of the WP3 – middleware is prepared with the aim of running on an interactive Grid framework. The WP4 will primarily support the CrossGrid software on Intel-based on local linux clusters interconnected via high-speed network [4].

Interactive use will proceed through User Portals developed in task 3.1, and support roaming access. This will require setting up the corresponding services in the testbed side, likely sharing the hardware resources with those used for implementing the Virtual Organisation.

Scheduling agents are proposed in task 3.2. The architecture is an open issue, with two possibilities: one based on a tight design where all the components are integrated, or a modular one with subcomponents distributed across the Virtual Organisation. Existence of Resource Brokers and Job Submission Services is assumed. All these points have to be considered by this task on the testbed architecture.

Monitoring middleware (task 3.3) does not impose any extra constraint on testbed setup, and they will provide very valuable information for its operation, as the Jiro technology will monitor infrastructure components.

The optimisation of Data-Access task (3.4) includes support for local-site optimisation for data storage strategy. Related hardware nodes (Storage Elements, Metadata and Replica Catalog) should have a least one public IP address and there are no special hardware requirements for nodes sustaining Storage Element software.

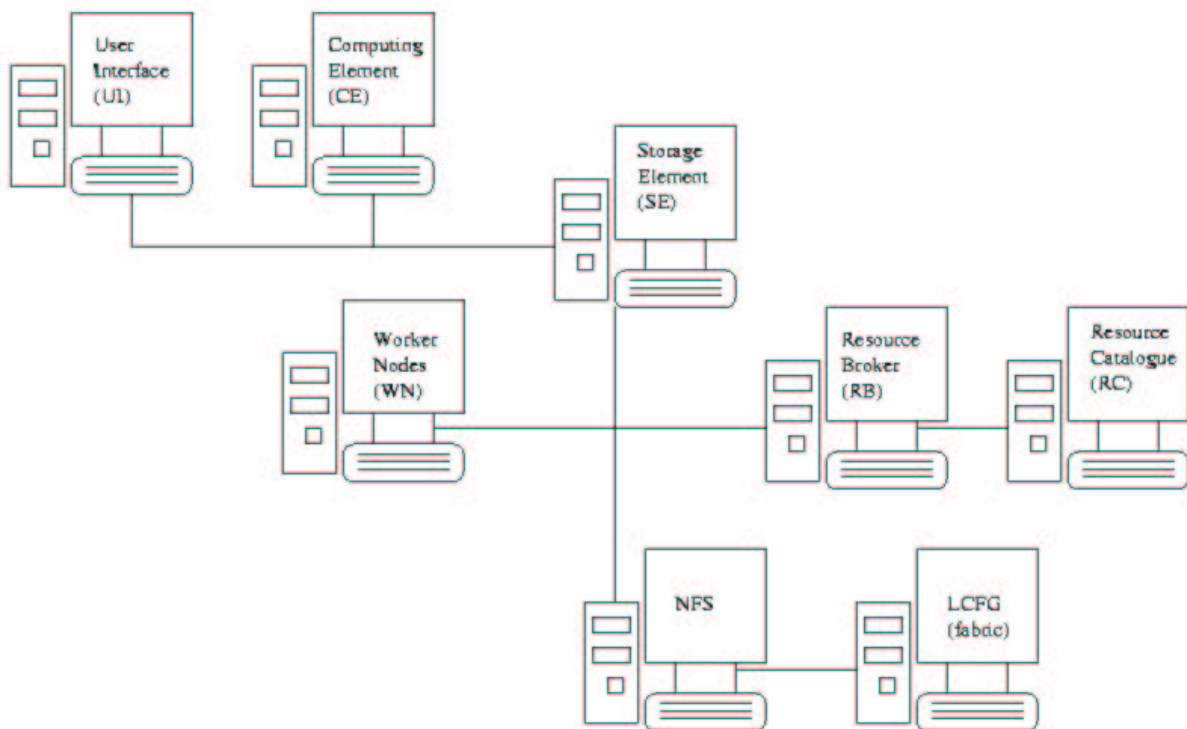


Figure 5 Minimum hardware required for a Testbed

## 9. 5.4. SOFTWARE-PROVIDE RESOURCES

- A complete list of the public packages (Globus, DataGrid, packages which will be used by other software packages or application software) that will be provided.
- For each package, a complete description of the public interface to the package (the API, protocol, etc.)
- List of dependencies on Globus core services, DataGrid software or other software and any external packages.

- Any operating system, kernel, or compiler requirements of the package (including the exact version).
- Detailed and clear instructions for building and configuring the task's software.
- Verification scripts as part of the installation so that they can be run to find out whether the basic configuration is working.
- Any internal tests which the task has written.
- Tasks must provide all information necessary to build the distribution packages. This currently means that the procedure for extracting a tarball from the packages sources and a .spec file must both be provided (it will be announced before Month 6). In addition, the software tasks should verify that a complete usable RPM package results from the produced tarball and .spec file (It will be announced before Month 6).

Tasks should package their software in as finely-grained way as possible without introducing circular dependencies. Software source code must be available to the WP3 and collaboration through the central repository. The software repositories can either be moved to the central repository of CrossGrid project or mirrored by it.

#### ***Requirements from WP1 (CrossGrid application development)***

The architecture of the WP3 middleware will be general, independent from the applications. The essence of the work in CrossGrid, and especially in the application tasks is to realise a system capable of supporting distributed interaction and visualisation.

The application tasks will not need components from WP2 and WP3 in the first year, but will be prepared to integrate these components for testing after one year. For the first year, the only common development identified is the datamining that is shared between tasks 1.3 and 1.4b

In later years, most of the applications need the visualisation toolkit.

All applications will use MPI somewhere, but won't be Single Program applications. Breaking off jobs, starting new jobs and modifying job parameters (which can influence job scheduling properties – run time, memory, number of tasks, etc?) are kinds of interaction envisaged by applications.

Tasks 1.1 and 1.2 primarily address the High Performance Computing and interactive aspects of the use of the grid. Tasks 1.3 and 1.4 tend more to High Throughput Computing, through sort response times remain important. The scheduling and monitoring capabilities realised in Tasks 3.2 and 3.3 are vital in this respect. Optimisation of data access (Task 3.4) which are distributed have to be used by the Tasks 1.2, 1.3 and 1.4.

#### ***Requirements from WP2 (Grid Application Programming Environment)***

WP2 will analyse the types of run time data needed by the tools to be developed in this WP and examine which of the needed data is already available via the DataGrid services. Since the access to all run time data will occur through the unified monitoring interface developed in Task 3.3, the results of the analysis will define the major requirements for this.

The tools allow application developers to measure, evaluate, and visualise the performance of Grid applications with respect to data transfer, synchronisation and I/O delay as well as CPU, network and storage utilisation. The tools will support any level of granularity, from the Grid level down to the process level. Acquisition of raw performance data will be based on the monitoring infrastructure developed in WP3. The performance evaluation tools will exhibit four distinctive and novel features:

1. The tools operate on-line. This means that the user can select the performance data to be measured at run-time, based on the results of previous measurements. Thus, the need to acquire, store and transmit large amounts of trace data is avoided.

- 
2. the tools include an automatic component that can extract high-level performance properties from the measured raw performance data.
  3. A tool will be provided to extract the relationships between the application execution features, i.e. the performance of the grid (network speed, CPU speed, memory bandwidth), the problem sizes (vectors and matrices) and the real execution time for some selected kernels in the applications, in a form of an analytical model.
  4. The tools will use a standardised tool/monitor interface which will be an extension of the OMIS specification for Grid applications