



CONTRACT NUMBER 508830

DEISA
**DISTRIBUTED EUROPEAN INFRASTRUCTURE FOR
SUPERCOMPUTING APPLICATIONS**

European Community Sixth Framework Programme
RESEARCH INFRASTRUCTURES
Integrated Infrastructure Initiative

SA4 Service Definition and Operation

Deliverable ID: D-SA4-1

Due date : September 15, 2004

Actual delivery date: September 15, 2004

Lead contractor for this deliverable: IDRIS – CNRS, France

Project start date : May 1st, 2004

Duration: 5 years

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	X
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of Content

Table of Content	2
1. Executive Summary	3
2. Introduction	4
3. Service Definition	6
3.1 Remarks on the DEISA Operational Model.....	6
3.2 Common Production Environment	6
3.2.1 Shells	7
3.2.2 Compilers, libraries, tools and applications.....	7
3.2.3 Tool to manage the software components.....	7
3.2.4 Way to set up the user environment	8
3.3 Framework to test the availability and the coherency of the Common Production Environment.....	8
3.4 User Support Service.....	11
3.4.1 Current organisation of the User Support Services of the partners.....	11
3.4.2 Organisation of the DEISA User Support Service	12
3.5 Co-ordination aspects	13
4. Service Operation	14
5. Conclusion	15

1. Executive Summary

The main objective of this service activity is to deploy all the actions required to enable the scientific users adoption and utilisation of the DEISA supercomputing infrastructure. This is mainly done by providing and maintaining a *Common Production Environment* on all the platforms of the infrastructure, and providing also documentations on its usage, training sessions to help the port and the optimisation of the applications and an *Help Desk* service.

This document describes how the user environment will be defined and set-up on all the platforms, how its coherence and availability will be checked, how the user support will be organised and what kind of actions will be proposed to the various users communities to help them to uptake the infrastructure.

This document is publicly available.

2. Introduction

The Applications and User Support Service Activity constitutes the privileged interface of the DEISA research infrastructure with its main users, namely, the European scientific community. This Service Activity is in charge of all actions that will enable or enhance the access to the supercomputing resources of the Consortium and their impact on computational sciences.

The Applications and User Support activity is directly driven by the policies and strategies related to the scientific management of the DEISA research infrastructure. This activity will therefore be the privileged vector for the implementation of the scientific policies established by the Executive Committee of the Consortium to enhance and reinforce high performance computing in Europe. It has to provide direct support to the major scientific initiatives launched by the Consortium.

The major responsibilities of the Applications and User support activities are:

- ? Providing consultancy services, training, and support documentation to end users on the access and the usage of the DEISA supercomputing environment
- ? Contributing to enable and deploy new, challenging scientific applications, following the scientific priorities established by the Consortium
- ? Setting and maintaining a global Common Production Environment (in some aspects, in collaboration with the *Resource Management and Middleware* activity SA3)

The DEISA operational model, based on the strong integration of national supercomputers, requires a very high level of harmonization of the production environment of the different sites. Just having the same operating system and software versions (which by itself is already an accomplishment) is not sufficient. A number of procedures and tools needed in batch processing, administration and security must be coherent with one another. This requires:

- ? a *Common Production Environment* on all the computers integrated in the infrastructure,
- ? a framework to automatically verify the consistency and the availability of the various components of this *Common Production Environment*.

On all these points, the decisions described in this document about the technical choices and the definition of the organisation of the service, for the lifetime of the project, have been taken on the basis of :

- ? The new context created by the DEISA project and its innovative operational model.

- ? The strong and long experience of all the partners, specially of their own user support services.
- ? The study of the experiences and choices of some other projects which build a distributed infrastructure, specially those of the American TeraGrid project, which federate the computing resources of some of the biggest academic centres in the USA.

3. Service Definition

3.1 *Remarks on the DEISA Operational Model*

To explain some of the choices done to organise the *User Support Service*, we must highlight one key point of the *DEISA Operational Model*. Users will have a unique entry point to the global infrastructure, called *Reference Site*, which is the one where they work usually independently of the DEISA project. This implies that users will have an interactive access only to this site and not to the others. Moreover, all their support requests will be primarily addressed to the user support service of this site. For end users, this is sufficient given the single system image of the infrastructure and the transparent services it provides.

The different national User Support service will pool their efforts and work in a global and coordinated way, operating on the full infrastructure. Unlike users, the members of these services will have the possibility to open interactive connections on the other sites and to directly access all the distributed computers integrated in the DEISA platform.

3.2 *Common Production Environment*

A distributed infrastructure like the DEISA supercomputing platform must offer an environment as homogeneous as possible, with a high level of coherence across the integrated computing platforms. This is especially true among subgroups with the same hardware and software architecture, where not only the executable softwares (as those frequently used by the chemists) and the high level libraries must be the same, but also the same compilers and specific tools of each manufacturer must be available.

For this purpose, a *Common Production Environment* has to be defined, set up and deployed on each computer integrated in the platform. It includes:

- ? the *shells*,
- ? the *compilers*,
- ? the *libraries*,
- ? the *tools*,
- ? and the *applications* available.

The subset made of the parallel and distributed libraries, the Grid Applications Toolkits, the input/output libraries and the scientific libraries is called the *Application Development Environment* (ADE).

This *Common Production Environment* must at the same time benefit from a high level of stability and remain in permanent evolution, both to upgrade the existing versions of the components when it will be necessary and to integrate new components, according to the evolution of the needs and the arrival of new users of the platform. The upgrade of the default versions of the components will be done in a coordinated way and according to the internal procedures defined, to guarantee the coherence of the software environment across the sites.

We must note the special case of some licensed softwares, coming from the industrial or third party world, which will be available on the DEISA platform and therefore accessible inside the *Common Production Environment*, but nevertheless not installed on each platform but only on one or some, obviously after negotiation and agreement with the software vendors.

3.2.1 Shells

It has been decided that the shells supported will be:

- ? the *Bourne Again Shell* (bash),
- ? the *Enhanced C Shell* (tcsh).

The motivations are that these two shells are available in source form, which is not the case of some of the other ones. This will allow us to install them everywhere in the future if needed. These shells are already well known and heavily used in the Unix community, and include all or nearly all of the features of the most frequently used other shells (*Korn Shell* for bash and *C Shell* for tcsh).

3.2.2 Compilers, libraries, tools and applications

The discussion to define the list of compilers, libraries, tools and applications which will be integrated in the initial version of the *Common Production Environment* started some weeks ago, taking into account the current situation in all the sites and the opinions of the partners. A preliminary list, divided between *Compilers*, *Libraries*, *Tools* and *Applications*, has been established, which will of course be updated each time that evolutions will appear necessary.

3.2.3 Tool to manage the software components

A generic tool is necessary, both to manage the software components and to allow the users to require the ones they need. When a user needs a software product, he should be able to issue a simple high-level command, not giving any technical detail. This command would make all the necessary modifications in the user environment.

The modifications may be very different from one case to another. In the case of a library, for instance, some linker options will be modified, specifying the location and the exact name of the file used, and possibly also the environment variable which defines the search path for the dynamic libraries. In the case of a huge executable software, like a chemistry code, a lot of actions may be required, both specifying several environments variables and executing some initialisation scripts.

Such a tool must be as simple as possible to use and nevertheless flexible enough to allow to handle various situations. Specially, without any other information, it will allow to access the default version as currently defined in the target computer, and with the default configuration set up for it.

But the users must also be able to make some special requests, like using a version of the software other than the default one (if several are available), or using a configuration of the software other than the default one (for instance with a different addressing mode or with a different kind of internal representation of numbers, when several are supported on the platform used).

A brief study was made on the tools available in the public domain to satisfy these needs. It appeared that only two are really interesting, the other ones having only part of the required functionalities:

- ? the first one is *Modules*, originally developed by SUN, and later heavily used on CRAY systems, but also on many other platforms. This tool is today available on SourceForge,
- ? the second one is *SoftEnv*, developed at the Argonne National Laboratory.

SoftEnv is the tool chosen in the TeraGrid project, but nevertheless all the partners have agreed to use *Modules* in DEISA. Our study shows that the functionalities of the two products are very close and without significant advantages for one of them, and most of the partners already have a long experience with *Modules*.

3.2.4 *Way to set up the user environment*

Even if the tool is defined, there are several ways to configure it. This is a rather simple question in the case of the homogeneous IBM configuration, but a study has also been made to prepare for an extended heterogeneous configuration. The implementation that we will define here very soon for a homogeneous configuration must be extensible to a heterogeneous configuration.

The characteristics to manage in the homogeneous case are the versions of the components and, in a few cases, the addressing mode (32 or 64 bits) used. In the heterogeneous case, we must also manage the representation of default integer, real and double precision numbers, and the internal representation of binary numbers (in the modes called *big-endian* or *little-endian*). Those characteristics must be managed in the simplest possible way from the user point of view.

3.3 ***Framework to test the availability and the coherence of the Common Production Environment***

On a distributed platform like DEISA, it is not sufficient to have a *Common Production Environment*. Administrators, people in charge of the operation of the computers, user

support teams and even for some aspects the users, must all be able to view an updated and detailed status of this environment.

The actions required here are:

- ? To verify the availability of the compilers, libraries, tools and applications (including version numbers) installed on the different computers and also, when possible, their current behaviour. This last point is obviously unreachable in a generic way, as some libraries or applications have their own huge system of tests, which sometimes include hundreds or even thousands of various elementary tests, to check as many functionalities of the softwares as possible. Even if it is not realistic to offer a dynamic view of such detailed behaviours (these huge sets of tests are mainly run when a new version of the software is installed), some basic tests which guarantee the main functionality of a software (or a few functionalities) would be really useful.
- ? To verify the status of the installations after software upgrades. As previously said, sometimes the softwares (mainly the libraries and applications) include their own system of tests, but some changes in a software product may badly impact another software. Think for instance to the upgrade of a compiler, to say nothing of the upgrade of the operating system, which can have all possible sort of border effects.
- ? To offer both to the centre's staff and to the users an updated view of what is installed and available on the various computers of the distributed infrastructure. The part of this information that the users will be able to access still needs to be defined, but in any case it will be a subset of the information accessible to the centre's staff. Both the administrators and the persons in charge of the operation must be alerted when some DEISA software component is not available or not correctly working. User support teams will obviously use these informations for diagnostic purposes.

The study of the state of the art in this field and of the capabilities of the available tools has already been carried out and decisions will be taken soon.

Further developments have been made on monitoring tools, to display the status of all the machines of a computer grid, sometimes adding detailed informations obtained by:

- ? queries to the various resource managers and schedulers,
- ? computation of performance metrics

The framework of such monitoring tools is not very different to the framework required to test and report the status of distributed software components, with automatic submission of repetitive queries, at specified time intervals.

Tools which give an overview of the status of software components on a distributed infrastructure have been developed mainly for grids operated by the Globus toolkit. These tools generally have very limited capabilities.

Nevertheless, the most ambitious work has been done in the *Test Harness and Reporting Framework* called *Inca*, developed mainly by the Argonne National Laboratory and the San Diego Supercomputer Center, and currently used inside the TeraGrid project.

This tool, not directly dependent on the Globus toolkit, is divided between:

- ? *Reporters* (more than one hundred have been defined), which each define the way to test a specific component,
- ? *Harnesses*, which are daemons which automatically execute the *Reporters* at the moments chosen and store all the collected informations in a central repository,
- ? and *Clients*, which are Web interfaces to allow to query and display the various informations collected.

In the DEISA project, the strategic and technical choices in this area are planned for the next months, in co-ordination with the *Resource Management and Middleware* activity (SA3).

3.4 User Support Service

3.4.1 Current organisation of the User Support Services of the partners

The main characteristics of user support organisations of the partners are described in Table 1 below.

	CINECA	CSC	EPCC	FZJ	IDRIS	RZG	SARA
Nb of users	700	1500	400	500 ¹	1500	500 (HPC) 3000 (general)	150
Nb of persons	12	30 ²	15 ³	14 ⁴	17	3-6 ⁵	20
Hot line	Yes	Yes ⁶	Yes	Yes	Yes	Only limited	Yes
Trouble Ticket System	No ⁷	No ⁸	Yes ⁹	Yes	No	Yes	Yes ¹⁰
Training Sessions	Yes	Yes	Yes	Yes	Yes	No	Yes
Special Competencies	HPC, CFD, oil and gas applications, industry support in general	HPC, coupled applications, scientific department with scientific support (in various fields)	HPC, tera-scaling ¹¹ , astrophysics, QCD	HPC, virtual reality, steering performance tools, comp. chemistry, methods for long range interactions, comp. physics, QCD	HPC, coupled applications, computational steering, astrophysics, chemistry, numerical analysis	HPC, computational steering, material sciences, plasma physics, astrophysics, bioinformatics	HPC, virtual reality, high performance networks, life sciences / BioASP

Table 1. Main characteristics of user support organisations.

¹ Each project has a responsible assigned.

² No dedicated user support organisation.

³ User support split across EPCC and CLRC.

⁴ 10 persons for the help desk, but organisation in three levels (help desk, computing support, science support).

⁵ 3-6 persons full time manpower spread over many groups/persons.

⁶ The help desk only assigns problems reported to specialists.

⁷ One is used internally for issues related to the machines, but no one for user support.

⁸ No *Trouble Ticket System* but a log of all emails sent to the help desk address is kept.

⁹ In-house developed system to track all problems submitted by email, web or phone. Used to generate statistics concerning quality of service of support.

¹⁰ Automatically generated for problems submitted by emails; manually generated for problems submitted by phone calls.

¹¹ Optimisation of applications on a large number of processors.

3.4.2 Organisation of the DEISA User Support Service

The DEISA *User Support Service* will be based on a close co-operation between the existing services of each of the partners, but without a centralised human interface, that is to say without a unique help desk and hot line.

After discussions between the partners, there was a common agreement on this position, which appears to be flexible, and also easy to build and to integrate in the current organisations in a non-disruptive way. This organisation will be fully coherent with the DEISA *Operational Model*, in which each user has a unique *Reference Site*, the only one on which he will have an interactive access, even if his batch jobs can be migrated on other sites.

Therefore, each centre will be fully responsible of its (known) users and its *User Support Service* will be the natural interface for their requests. Nevertheless, a strong coordinated action between the different User Support services will be deployed, as all the problems reported which will not concern only the primary site must be investigated and solved by the global *User Support Service*.

Although the help desk will not be unique, some other services must be centralised. First, all the DEISA specific documentations (Primer Guide, FAQs, specialised manuals), must be available on the central location of the DEISA Web site, even if local links to these documentations in the information system used by each partner will be obviously helpful for users.

Furthermore, it seems necessary to centralise the informations collected on the problems and questions which will be submitted by the users. For that, a *Trouble Ticket System* (TTS) must be chosen and implemented. As reported in the preceding paragraph, some sites already use such system for their own usage, but some others not.

It will be used to:

- ? Give to all sites a global and complete view of the situation of the virtual platform. It will help each *User Support Service* to solve the problems reported to them that concern in fact the computers of other sites, having access to the information on problems reported to other sites. It will also help everybody to understand globally how to improve some things, considering on the last period the possible repetition of the same questions or problems, which will show for instance the necessity to improve some parts of the documentation available.
- ? Allow to keep the whole memory of the situations and problems reported.
- ? Allow to generate reports and statistics, to have a general view of the *User Support* activity and of its historical evolution.

Such a tool must be chosen before the end of 2004, with subsequent installation and usage, as it will be useful before the arrival of the first *real* users, integrating the experience of the *User Support* teams of the partners during the preparation phase.

3.5 Co-ordination aspects

An important aspect of the activity on *Applications and User Support* in DEISA, and a key point for its success, will concern its co-ordination with the other activities, specially:

- ? the *Dissemination* activity,
- ? the activities on the *Infrastructure Services*,
- ? all the scientific *Joint Research Activities* (JRA 1-6).

User Support will rely on instruments deployed by the *Dissemination* team and especially on the Web server maintained by it, which will centralise all documentations and informations available. The *Dissemination* team will play an important role in providing support to the preparation and diffusion of documentation, user training, etc.

All the usage of the distributed infrastructure will be obviously completely dependent of the low-level layers introduced (networks, global file systems, middlewares, multi-clusters batch managers, etc.). A close collaboration with the *Service Activities* will be mandatory.

Activities of JRAs 1-6 can be seen as science gateways in which advanced support services are deployed for specific research activities. The JRAs will introduce the first *real* users and their applications. This will greatly help us to diagnose and solve many possible portability problems which can occur in the usage of the new software components used to provide single system image of the whole distributed infrastructure. But the special activities first deployed for the JRAs will be normally integrated in the future in the current *User Support* activities accessible to all the users of the distributed platform.

4. Service Operation

According to the *Service Definition* detailed in the preceding section, the *Service Operation* will include:

- ? The implementation and the maintenance of the defined *Common Production Environment* by all the partners, on each computer integrated in the distributed infrastructure.
- ? The implementation and the maintenance of the framework to test the availability and the coherence of this *Common Production Environment*, in collaboration with the *Resource Management and Middleware* activity (SA3).
- ? The writing and updating of the DEISA user documentation (preliminary one will be the content of the deliverable D-SA4-2). Documentation must be defined and its content chosen, but the preliminary versions must obviously include at least a Primer Guide, FAQs and specialised manuals. These will all be accessible on the DEISA Web site.
- ? The access to a distributed *User Support Service*, based on the strong co-operation of the relevant services of all the partners and with a centralised storage of the various questions and problems reported, using a *Trouble Ticket System*, to allow a global and historical view of all the preceding actions.
- ? The organisation of DEISA training sessions and workshops on *High Performance Computing* topics, based on the long and strong experience of most of the partners, mainly on the subjects of parallelisation, optimisation, grid computing and applications coupling. It must help both to facilitate the uptake of the DEISA infrastructure by the scientists and to help them to use efficiently this infrastructure.
- ? The special care given to applications with a high level of parallelism (that is to say which use a huge number of processors).
- ? The possibility for users to access a *specialised* local support, for a limited period of time, after agreement on their requirements, to help them to port or to optimise their applications on the DEISA infrastructure.

5. Conclusion

Even if the *real* users will only arrive during the first semester of 2005, a lot of work is being done in advance. It is obvious that a careful examination of user feedback will be needed later, starting in the middle of 2005, and that our operational model and all our procedures may have to be adapted.

It has also been emphasised that a strong co-ordination with other domain activities, and specially the *Dissemination*, *Service* and *Joint Research Activities* will be a key point for the success of the whole project.

The main actions scheduled for the next months are the following:

- ? The definition of the way to set up the user environment, based on the *Modules* software, in co-ordination with the *Resource Management and Middleware* activity SA3 (technical discussions started in June 2004).
- ? The definition of the content of the initial *Common Production Environment* and the building of the *Modules* interface (based on the so-called *modulefiles*) for each version of the softwares to include (technical work started in July 2004).
- ? The definition of the framework to test the availability and the coherence of the *Common Production Environment*, in co-ordination with the SA3 group (technical study started in July 2004).
- ? The first experiments to be done by the members of the application teams of the partners, before the availability of any special component of the distributed infrastructure, to test the portability of the codes involved in the scientific *Joint Research Activities* (JRAs 1-6) between the sites (technical work started in July 2004).
- ? The writing of the first user documentation (in the second semester of 2004).
- ? The choice of a *Trouble Ticket System* software to register all the problems reported, its installation and set up (in the second semester of 2004).
- ? The opening of the *User Support Service* when first *real* users arrive (in the first semester of 2005).

References and Applicable Documents

- [1] DEISA: <http://www.deisa.org>
- [2] Globus: <http://www.globus.org/>
- [3] Inca: <http://tech.teragrid.org/inca>
- [4] Modules: <http://modules.sourceforge.net/>
- [5] SoftEnv: <http://www-unix.mcs.anl.gov/systems/software/msys>
- [6] TeraGrid: <http://www.teragrid.org/>

List of Acronyms and Abbreviations

ADE	Application Development Environment
BioASP	Bioinformatics Application Service Provider
CFD	Computational Fluid Dynamics
CLRC	Council for the central Laboratory of the Research Councils
HPC	High Performance Computing
QCD	Quantum Chromodynamics
TTS	Trouble Ticket System