CONTRACT NUMBER RI-222919

DEISA 2
DISTRIBUTED EUROPEAN INFRASTRUCTURE FOR SUPERCOMPUTING APPLICATIONS

European Community Seventh Framework Programme
RESEARCH INFRASTRUCTURES
Integrated Infrastructure Initiative

Initial Report on Environment and User Related Application Support

Deliverable ID: DEISA2-D6.1
Due date: October 31st, 2008
Author: Denis Girou, IDRIS-CNRS

Project start date: May 1st, 2008
Duration: 3 years

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2011)		
Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
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 Contents

List of Figures

1 Introduction

1.1 Executive Summary

The *Environment and User Related Application Services* work package (WP6) has two major complementary objectives: first, to maintain and improve the applications environment; and second, to provide support to users in the applications area.

To achieve the first objective, a software layer has been defined and deployed since the beginning of the DEISA1 project. It is named the *DEISA Common Production Environment* (DCPE), which is one of the major features in DEISA. It defines a coherent set of software, accessible on the various sites of the infrastructure, and it also offers a common interface to the users, independent of the target platform employed. And, to permanently check the availability of the software included in the DCPE, a monitoring framework has been deployed and is actively maintained to provide a complete view of the status of the software availability on the different platforms integrated in the infrastructure. The tasks T6.1 and T6.2 are those related to these activities.

Concerning the second main objective, the support provided to the users is based on the usage of a Help Desk service, which manages the requests for information or assistance. The key tool used to manage this service is the *DEISA Trouble Ticket System* (DTTS), which is also heavily used internally by staff members, and is maintained by the Operations Service Activity (WP3). Further, to complement the user support activities outside the classical management of the requests sent to the Help Desk, special efforts are made to promote the various DEISA middleware and to help users to efficiently use the DEISA hardware and software infrastructure, by means of dedicated advanced application production assistance. In addition, support is also provided by the traditional way of documentation related to all the DEISA services accessible to the users, which is updated when needed and, of course, created when new user services are provided. The tasks T6.3, T6.5 and T6.4 are those related to these activities, respectively.

This document gives a detailed overview of all the work achieved or currently undertaken in this work package, for the five different tasks mentioned. But, as most of them have been active for several years inside the DEISA1 project, we particularly emphasize the ways any implementation redefinitions or changes that were recently introduced. These changes improve the organization of some tasks and the services provided to the users, using the huge experience and feedback acquired in DEISA1.

1.2 References and Applicable Documents

[1] DEISA2 Description of Work (Annex I of the Grant Agreement)
[2] DEISA web pages: http://www.deisa.eu
[3] Deliverable DEISA2-D1.1: Initial Report on Management
[4] DEISA2-D3.1: WP3 deliverable: Initial Report on Operations and Quality Plan
[5] DESHL (*DEISA Services for the Heterogeneous management Layer*) http://forge.nesc.ac.uk/projects/deisa-jra7/
[6] Globus Toolkit: http://www.globus.org/toolkit
[7] INCA (Test Harness and Reporting Framework): http://inca.sdsc.edu/
[8] Modules: http://modules.sourceforge.net/
[9] Request Tracker: http://www.bestpractical.com/rt
[10] TeraGrid: http://www.teragrid.org/
[11] UNICORE (UNiform Interface to COmputing REsources): http://unicore.sourceforge.net/

1.3 Document Amendment Procedure

This document is prepared according to the guidelines defined by the management of DEISA2. These rules can be found in section 2.7 of the deliverable DEISA2-D1.1 [3].

1.4 List of Acronyms and Abbreviations

ATaskF	Applications Task Force
BSCW	Basic Support for Cooperative Work
DART	DEISA Accounting Report Tool
DATS	DEISA Applications Test Suite
DCPE	DEISA Common Production Environment
DEC	DEISA Executive Committee
DEISA	Distributed European Infrastructure for Supercomputing Applications
DESHL	DEISA Services for the Heterogeneous Management Layer
DLLTS	DEISA LoadLeveler Test Suite
DoW	Description of Work (Annex I of the Grant Agreement)
DTTS	Open Science Grid (an US Grid Initiative)
GridFTP	File Transfer Protocol in Grid Computing Networks
gsi-SSH	Globus Security Infrastructure Secure SHell
PI	Principal Investigator
SDSC	San Diego Supercomputer Center
UNICORE	UNiform Interface to COmputing REsources
WP	Work Package

2 Management of the Environment and User Related Application Service

2.1 Introduction

As said in the introduction, the activities in this work package are divided in five different tasks. In this Section we explain how these tasks are managed and what are the special relationships established with some of the other work packages.

Nearly all partners are involved in the same manner: 12 PM of contributed effort from each partner except IDRIS, with 24PM as the coordinator of the work package, EPCC with 16 PM and ECMWF with 2 PM.

It must also be emphasized that there is a strong relationship between this work package (especially its tasks T6.1, T6.2 and T6.3) and task T3.5 (User-Related Services) of Work Package 3 (Operations).

2.2 Management organisation

2.2.1 WP6 Tasks

Each of the five tasks defined inside this work package has a dedicated task leader, under the responsibility of the work package leader. Each task leader is in charge of:

- the specific management of his task,
- the definition of the work,
- monitoring the progress against the initial objectives,
- reporting progress and any problems and/or difficulties encountered.

The responsibilities are the following:

- T6.1 – Evolution of the DEISA Common Production Environment: IDRIS
- T6.2 – Monitoring of Applications availability: LRZ
- T6.3 – Help Desk: SARA
- T6.4 – User Documentation: EPCC
- T6.5 – Advanced Application Production Assistance: CSC

2.2.2 Relationships with the other work packages

This work package (WP6) has a strong connection with several others. The strongest relationship is with the Operations work package (WP3), which is in charge of all the operational aspects of the infrastructure. It is obvious that all the operational services of DEISA in WP3 directly concern and impact the users, that is to say, they have strong relevance for WP6. This concerns all the subtasks of the operational services and especially the User-Related Services (T3.5), for which the task leader was explicitly chosen to be the WP6 leader to ensure the closest possible relationship.

But strong connections also exist with other work packages: WP2, for the training activities in particular, which involve current or expected future users; WP4, for the work on technologies which can potentially become operational services accessible to the users; WP5, for the enabling of applications which will become applications for production mode operation, using both the dedicated user environment and user support; and WP7, in charge of the coordination of the projects of the *DEISA Extreme Computing Initiative* (DECI).

2.2.3 Organisation of videoconferences and face to face meetings

It has been decided to organize monthly videoconferences to discuss the status of the activities, as well as the current problems diagnosed and the planned work and events. The first was set up in July. Minutes are written after each of these videoconferences, summarizing the main points discussed and the actions planned, as well as the status of the actions decided during preceding videoconferences. When needed, special videoconferences are organized between the work package manager and the task leaders to discuss the coordination of the different tasks and to prepare the regular working group videoconferences.

No regular face to face meetings were initially planned; however, the experience of the first joint WP3/WP4 meeting on July 16-17, 2008, where important discussions concerning the T3.5 task (User-Related Services) occurred without some WP6 being present. As these were highly relevant for the WP6 activities, it is clear that a strong coordination with easy and frequent exchanges are required between the operational and non operational aspects of the common topics (maintenance of the DCPE, monitoring framework, Help Desk). This is why, after discussion with the WP3 (Operations) and WP4 (Technology) leaders, as well as with the Technical Coordinator, it was decided that in the future WP6 will join these regular meetings, which will become WP3/WP4/WP6 meetings. This will be clearly more efficient than to organize WP6 dedicated meetings, and this would have the benefit to enforce the cooperation and the exchanges between the three work packages, and between all the people working within.

Concerning the mailing lists, a dedicated list for WP6 of course exists since the start of DEISA2 and is used for all exchange of information between the staff members involved in its activities. And, for the reasons emphasized above for the meetings, with the strong connections between the task T3.5 and tasks T6.1, T6.2 and T6.3, it was decided not to define a special dedicated mailing list for T3.5 but to alias it to the WP6 mailing list.

2.3 Videoconferences and meetings

Three monthly videoconferences were set-up during the reporting period, on July 15th, September 5th and October 9th. Detailed minutes of the discussions are available.

It must also be noticed that WP6 has participated in the PRACE WP4 meeting in CSCS (Lugano, Switzerland) on October 13-14 concerning the user environment (definition of the *Software Stacks* and description of the interface developed to access them across the platforms, based on the *Modules* tool), to explain in detail the design choices and implementation done inside DEISA.

3 Evolution of the DEISA Common Production Environment

The unified *DEISA Common Production Environment* (DCPE) is a major feature in DEISA, as in all distributed infrastructures of this kind, defining a coherent set of software accessible on the various sites of the infrastructure. It offers a common interface to the users, independent of the target platform employed. Of course, the level of coherence is not the same everywhere in the infrastructure, ranging from very high inside each subgroup of homogeneous computers to a lower level across the other subgroups.

The main components of the DCPE are:

- a coherent set of software packages divided into six categories: environment, shells, compilers, libraries, tools and applications;
- a uniform interface to access the software, provided by the *Modules* tool [6].

As this is a major requirement in such an infrastructure, the definition of the DCPE was done at the very early beginning of DEISA1. In the summer of 2004, both defining the initial *Software Stack* (at that moment only for the so-called *core sites* running IBM Power 4 computers under the AIX system) and choosing the *Modules* tool for the user interface. In the next period, the DCPE was enlarged to include other systems (SGI Altix, IBM Linux and, later, CRAY Linux), defining dedicated *Software Stacks* for them and building for them the required interfaces with the *Modules* tool.

The management of the *Modules* tool itself is done inside WP3 (task T3.5a *Maintaining the Modules Framework*) – see the deliverable D3.1 [2] for explanations and technical details. After four years of experience, we decided to maintain our own flavour of the *Modules* tool, allowing us to circumvent the problems found in the maintenance of this public domain software. The task T6.1 is itself in charge of the development and maintenance of the dedicated environment built using this tool, based on a set of scripts, called *modulefiles*, to interface each software included in the *Software Stacks*.

The activity in this task during the reporting period was concentrated in two directions:

- the redefinition of the *Software Stacks* ;
- the installation of the renewed *Modules* environment prepared at the end of DEISA1.

3.1 The Software Stacks

Concerning the *Software Stacks*, it has become clear from our experience that our initial definition was not flexible enough and that the evolutions in such a rigid context were problematic. In our original design, all sites with the same computer and the same operating system must keep a very strong synchronization between their *Software Stacks*, all including exactly the same software and at the same major version level for each of them (discrepancies were only allowed on patch levels). But in practice such constraints appeared to be too rigid to be fulfilled. Four main problems were encountered:

- Some commercial software packages are very expensive on huge parallel supercomputers and some sites were not prepared to purchase these packages only to be compliant with the general rules, without guarantees that these packages would be used significantly on their platform;
- There was disagreement between the partners about the performances of a few software packages (this concerns third party applications), where some partners had higher demands on the minimal performance of the applications installed on their computer than some other sites;

- Some specific projects running on a single site (or in some cases only among the homogeneous set of IBM AIX computers) required specific software (mainly libraries and special applications), for which other sites were not interested in spending time installing them without any planned usage at that moment;
- Different sites have different schedules and priorities to install new software, or new versions of existing software, which sometimes meant a long delay before the expected synchronization between the *Software Stacks* of an architecture has been reached.

In view of our experience and in light of the problems given above, it was decided to modify our definition of the *Software Stacks*, relaxing various constraints and introducing more flexibility. In particular, we have introduced different categories:

- *core*: the small set of software common to all sites and strongly synchronized (as in the original definition),
- *core architecture*: the set of software common to one specific architecture,
- *optional*: software optionally included, without shared constraints.

Such redefinition of the *Software Stacks* was the topic of a proposal submitted for approval to the *DEISA Executive Committee* (DEC) in October, 2008.

3.2 The renewed Modules environment

In this area too, we used our four years of experience in DEISA to get the feedback and opinion of all the partners, and after a rather long process that started at the end of 2007 we agreed upon some changes and improvements in the *Modules* environment itself. This has concerned our choice to now support our own enhanced version of the *Modules* tool itself, as previously stated, and to simplify and improve our environment. The main enhancements concern:

- the removal of some unused features, which were initially introduced for greater power but which were not really used and in fact complicate the maintenance of the framework;
- a notably higher flexibility in the initialisation process, with less default choices and easy customization by a personal configuration file (it was a general complaint that the default initialization choices were too difficult to change in case of need);
- the support of different sets of compilers, when only one was initially accessible (which does not matter on some systems, but was a real drawback on others where one kind of compiler must be used for some applications, and better another one for some other applications).

4 Monitoring of Applications Availability

In a distributed hardware, but also software, infrastructure like DEISA, it is an absolutely necessity that the administrators, the user support services, and in some aspects the users themselves, have an updated view and a detailed status of the software environment. The monitoring of such a distributed software environment has become a requirement in all important grid projects, especially in a heterogeneous context.

The main features required in such tools are the ability:

- To verify the accessibility of the compilers, libraries, tools and applications (including version numbers) installed on the different computers and also, when possible, their current behaviour. Under this aspect, it is important that some basic tests, repeated at specified intervals of time, guarantee not only that a specific piece of software is installed and currently accessible, but also that its main functionalities are working correctly.
- To verify the status of the installations after software upgrades. Often, large pieces of software (mainly libraries and applications) include their own system of tests, but some changes in a software product may badly impact other installed software, so it is of interest to integrate more specific and cross functional tests.
- To offer both to the centre's staff and, with fewer details, to the users, an updated view of what is installed and available on the various computers of the distributed infrastructure. Both the administrators and the persons in charge of the operation must be alerted when some DEISA software component is not available or not working correctly. Also, user support teams will obviously need this information too.

For this purpose, the choice of the INCA tool (*Test Harness and Reporting Framework*) [5], developed by the San Diego Supercomputer Center, initially for the TeraGrid project [8], was made in DEISA1 in early 2005. During the following year, it was progressively deployed on the whole infrastructure and is used now in full production mode for around three years, including the switch to the new, completely rewritten, version 2.0 at the beginning of 2007. INCA is specifically designed to periodically run a collection of validation scripts, called *reporters*, with, for what concerns the software, the purpose of collecting two different kinds of information:

- the version of the software installed (*version reporters*),
- the availability and the correct operation of this software (*unit reporters*).

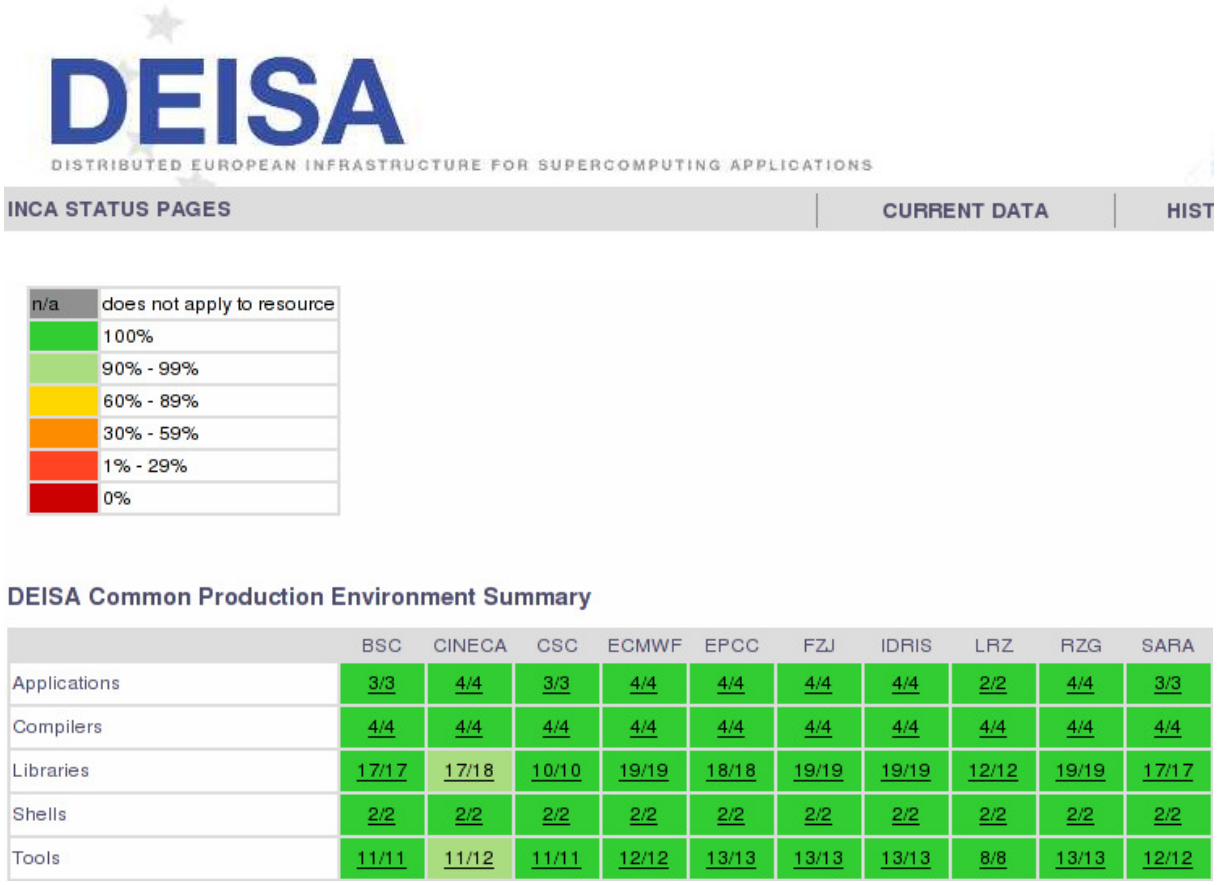
As explained in the deliverable D3.1 [1], the usage of INCA inside DEISA was progressively extended to also monitor the availability and status of various middleware, or at least to display in the unified INCA presentation views the information collected by some other tools dedicated to the monitoring of some services: LoadLeveler batch filters for the IBM platforms using the *DEISA LoadLeveler Test Suite* (DLLTS), UNICORE [9] using its dedicated Simon monitoring tool, the major Globus [4] services deployed on the infrastructure (namely gsi-SSH and GridFTP), the LDAP administration service, displaying, for each site, both the status of the local LDAP server and the result of a precise consistency check of the user's registration information stored in the LDAP database. And as stated in the D3.1 deliverable, the monitoring of other services is expected to be integrated in the near future.

Concerning the software availability and correct status, the tests integrated in INCA as *unit reporters* are based on the *DEISA Applications Test Suite* (DATS), developed in DEISA1 to check the behaviour of some major software components (compilers, linkers, some scientific libraries, parallel libraries and execution framework). It is currently composed of:

- a test to check the compilation of C and Fortran codes, and their linkage with an external scientific library,

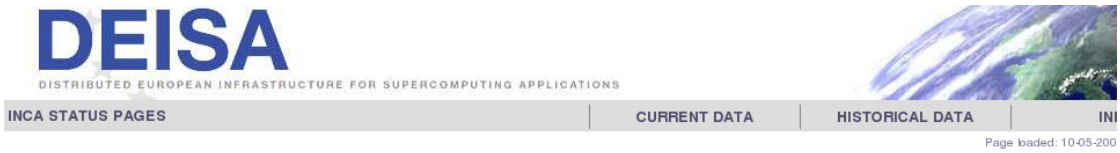
- a test to check the compilation of a C++ code and of an IDL (interface definition language for CORBA) code, and their linkage with the CORBA libraries,
- a test of execution of an OpenMP code,
- two tests of execution of MPI codes,
- a test of execution of a CORBA client / server application,
- a test of execution of a packaged application (CPMD).

We give here in the following screenshots the summary and detailed views of the monitoring of the application availability on our infrastructure, at the date of the deliverable.



	BSC	CINECA	CSC	ECMWF	EPCC	FZJ	IDRIS	LRZ	RZG	SARA
Applications	3/3	4/4	3/3	4/4	4/4	4/4	4/4	2/2	4/4	3/3
Compilers	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4
Libraries	17/17	17/18	10/10	19/19	18/18	19/19	19/19	12/12	19/19	17/17
Shells	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2
Tools	11/11	11/12	11/11	12/12	13/13	13/13	13/13	8/8	13/13	12/12

Figure 1 – INCA DCPE Summary View



n/a	does not apply to resource
n/i	is not installed on resource
	missing (not yet executed)
incaErr	internal Inca error
pass	passed
	version mismatch
error	error
*	result is stale

DEISA Common Production Environment

- cluster.application.cpmd.unit
- cluster.application.cpmd.version
- cluster.application.cpmd2cube.version
- cluster.application.gopenmol.version
- cluster.application.torb.version
- cluster.compiler.icc.version
- cluster.compiler.icpc.version
- cluster.compiler.ifort.version
- cluster.compiler.java.version
- cluster.compiler.pgcc.version
- cluster.compiler.pgCC.version
- cluster.compiler.pgf.version
- cluster.compiler.xlc.version
- cluster.compiler.xlC.version
- cluster.compiler.xlf.version
- cluster.library.acml.version
- cluster.library.blacs.version
- cluster.library.blacssmp.version
- cluster.library.compilelink1.unit
- cluster.library.essl.version
- cluster.library.esslsmp.version
- cluster.library.fftw.version
- cluster.library.gmalloc.version
- cluster.library.hdf5.version
- cluster.library.hydro.unit
- cluster.library.iobuf.version
- cluster.library.lapack.version
- cluster.library.libsci.version
- cluster.library.mass.version
- cluster.library.mkl.version
- cluster.library.nag.version
- cluster.library.netcdf.version
- cluster.library.pessl.version
- cluster.library.pesslsmp.version
- cluster.library.scalapack.unit
- cluster.library.scalapack.version
- cluster.library.tmqcd.unit
- cluster.library.uw.unit
- cluster.library.wsmp.version
- cluster.shell.bash.version
- cluster.shell.tcsh.version
- cluster.tool.compilelink2.unit
- cluster.tool.corba.unit
- cluster.tool.craypat.version
- cluster.tool.emacs.version
- cluster.tool.gmake.version
- cluster.tool.hpm.version
- cluster.tool.nedit.version
- cluster.tool.omniorb.version
- cluster.tool.openssh.version
- cluster.tool.perl.version
- cluster.tool.python.version
- cluster.tool.tcl.version
- cluster.tool.tk.version
- cluster.tool.totalview.version

Applications	BSC	CINECA	CSC	ECMWF	EPCC	FZJ	IDRIS	LRZ	RZG	SARA
cluster.application.cpmd2cube.version	apr06	apr06	apr06	apr06	apr06	apr06	apr06	n/a	apr06	apr06
cluster.application.cpmd.version	3.11.1	3.11.1	3.13.2	3.11.1	3.11.1	3.11.1	3.13.1	3.11.1	3.11.1	3.11.1
cluster.application.cpmd.unit	pass	pass	n/a	pass	pass	pass	pass	pass	pass	pass
cluster.application.gopenmol.version	n/a	2.32	3.00	2.32	3.00	2.32	3.00	n/a	3.00	n/i
Compilers	**BSC**	**CINECA**	**CSC**	**ECMWF**	**EPCC**	**FZJ**	**IDRIS**	**LRZ**	**RZG**	**SARA**
cluster.compiler.icc.version	n/a	n/a	n/a	n/a	n/a	n/a	n/a	10.1	n/a	n/a
cluster.compiler.icpc.version	n/a	n/a	n/a	n/a	n/a	n/a	n/a	10.1	n/a	n/a
cluster.compiler.ifort.version	n/a	n/a	n/a	n/a	n/a	n/a	n/a	10.1	n/a	n/a
cluster.compiler.java.version	1.4.2	1.5.0	1.4.2	1.4.2	1.5.0	1.4.2	1.5.0	1.4.2	1.5.0	1.5.0
cluster.compiler.pgcc.version	n/a	n/a	7.2-4	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.compiler.pgCC.version	n/a	n/a	7.2-4	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.compiler.pgf.version	n/a	n/a	7.2-4	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.compiler.xlc.version	8.0.0.0	8.0.0.19	n/a	8.0.0.0	8.0.0.13	9.0.0.3	9.0.0.2	n/a	9.0.0.4	10.1.0.0
cluster.compiler.xlC.version	8.0.0.0	8.0.0.19	n/a	8.0.0.0	8.0.0.13	9.0.0.3	9.0.0.2	n/a	9.0.0.4	10.1.0.0
cluster.compiler.xlf.version	10.1.0.0	10.1.0.8	n/a	10.1.0.0	10.1.0.8	11.1.0.2	11.1.0.3	n/a	11.1.0.4	12.1.0.0
Libraries	**BSC**	**CINECA**	**CSC**	**ECMWF**	**EPCC**	**FZJ**	**IDRIS**	**LRZ**	**RZG**	**SARA**
cluster.library.acml.version	n/a	n/a	3.6.1	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.library.blacs.version	3.2.1	3.2.0.1	3.0	3.1.0.2	3.3.0.2	3.3.0.2	3.3.0.2	n/a	3.3.0.2	3.3.0
cluster.library.blacssmp.version	3.2.1	3.2.0.1	n/a	3.1.0.2	3.3.0.2	3.3.0.2	3.3.0.2	n/a	3.3.0.2	3.3.0
cluster.library.compilelink1.unit	pass	pass	n/a	pass	pass	pass	pass	pass	pass	pass
cluster.library.essl.version	4.2.1	4.2.0.5	n/a	4.1.0.1	4.3.0.3	4.3.0.3	4.3.0.3	n/a	4.3.0.2	4.3.1
cluster.library.esslsmp.version	4.2.1	4.2.0.5	n/a	4.1.0.1	4.3.0.3	4.3.0.3	4.3.0.3	n/a	4.3.0.2	4.3.1
cluster.library.fftw.version	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5	2.1.5
cluster.library.gmalloc.version	n/a	n/a	1	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.library.hdf5.version	1.6.5	1.6.3	1.6.7	1.6.5	1.6.5	1.6.2	1.8.1	1.6.4	1.6.6	1.6.5
cluster.library.hydro.unit	pass	pass	n/a	pass	pass	pass	pass	pass	pass	pass
cluster.library.iobuf.version	n/a	n/a	1.0.2	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.library.lapack.version	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0
cluster.library.libsci.version	n/a	n/a	10.0.1	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.library.mass.version	4.3.0	4.4.0.1	n/a	4.1.0.0	4.4.0.1	4.4.0.1	4.4.0.1	n/a	4.4.0.1	5.0.0
cluster.library.mkl.version	n/a	n/a	n/a	n/a	n/a	n/a	n/a	9.1.1	n/a	n/a
cluster.library.nag.version	n/a	20B	n/a	21	n/i	21_1	21	21	21_1	21
cluster.library.netcdf.version	3.6.0	3.6.0	3.6.2	3.6.1	3.6.2	3.6.2	3.6.3	3.6.1	3.5.0	3.6.2
cluster.library.pessl.version	3.2.1	3.2.0.1	n/a	3.1.0.2	3.3.0.2	3.3.0.2	3.3.0.2	n/a	3.3.0.2	3.3.0
cluster.library.pesslsmp.version	3.2.1	3.2.0.1	n/a	3.1.0.2	3.3.0.2	3.3.0.2	3.3.0.2	n/a	3.3.0.2	3.3.0
cluster.library.scalapack.version	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0	1.7.0
cluster.library.scalapack.unit	pass	pass	n/a	pass	pass	pass	pass	pass	pass	pass
cluster.library.tmqcd.unit	pass	pass	n/i	pass	pass	pass	pass	pass	pass	n/i
cluster.library.uw.unit	pass	error	n/a	pass	pass	pass	pass	pass	pass	pass
cluster.library.wsmp.version	n/a	n/i	n/a	6.2.28	4.8.5	4.8.5	8.8.20	n/a	7.1.25	n/i

Figure 2 – INCA DCPE Detailed View

Shells	BSC	CINECA	CSC	ECMWF	EPCC	FZJ	IDRIS	LRZ	RZG	SARA
cluster.shell.bash.version	2.05b.0(1)	3.00.0(1)	3.1.17(1)	3.1.0(3)	2.05a.0(1)	3.00.16(1)	3.2.0(1)	3.1.17(1)	3.00.16(1)	3.1.17(1)
cluster.shell.tcsh.version	6.12.00	6.11.00	6.14.00	6.14.00	6.11.00	6.11.00	6.15.00	6.14.00	6.11.00	6.14.00
Tools	BSC	CINECA	CSC	ECMWF	EPCC	FZJ	IDRIS	LRZ	RZG	SARA
cluster.tool.compilelink2.unit	pass	pass	n/a	pass	pass	pass	pass	n/a	pass	pass
cluster.tool.corba.unit	pass	error	n/a	pass	pass	pass	pass	n/a	pass	pass
cluster.tool.craypat.version	n/a	n/a	3.2.3	n/a	n/a	n/a	n/a	n/a	n/a	n/a
cluster.tool.emacs.version	21.3.1	21.1.1	21.3.1	21.3.1	21.3.1	21.3.1	21.3.1	21.3.1	20.2.1	21.3.1
cluster.tool.gmake.version	3.80	3.80	3.80	3.80	3.80	3.80	3.80	3.80	3.80	3.80
cluster.tool.hpm.version	n/a	n/i	n/a	n/i	3.1.5	3.2.1	3.2.2	n/a	3.2.2	3.2.2
cluster.tool.nedit.version	5.5	5.5	5.5	5.5	5.5	5.5	5.5	5.5	5.5	5.5
cluster.tool.omniorb.version	4.1.0	4.0.6	4.1.0	4.0.6	4.0.6	4.0.6	4.1.3	n/a	4.0.6	4.1.1
cluster.tool.openssh.version	4.1p1	4.7p1	4.2p1	3.6.1p2-CERT-patched	4.7p1	4.3p2	4.7p1	4.2p1	4.3p2	4.2p1
cluster.tool.perl.version	5.8.3	5.8.2	5.8.8	5.8.0	5.8.2	5.8.2	5.8.2	5.8.8	5.8.2	5.8.8
cluster.tool.python.version	2.4.2	2.4.1	2.4.2	2.4.2	2.4.3	2.4.3	2.6	2.4.2	2.4.2	2.5.1
cluster.tool.tcl.version	8.4.6	8.4b1	8.4.12	8.4.12	8.4.12	8.4.12	8.5.5	8.4.12	8.4.15	8.4.12
cluster.tool.tk.version	8.4.6	8.4b1	8.4.12	8.4.12	8.4.12	8.4.12	8.5.5	8.4.12	8.4.15	8.4.12
cluster.tool.totalview.version	n/a	8.0.2-0	7.3.0-0	7.3.0-2	8.1.0-0	8.4.0-0	8.6.0-1	n/a	8.3.0-1	n/i

Figure 3 – INCA DCPE Detailed View (continued)

5 Help Desk

The main objective of T6.3 is the continuation and consolidation of the Help Desk Service as set up in DEISA1. At this time, the DECI users referred for help to the site assigned to the management of their project, called their *Home site* (the one representing the country to which their organisation belongs, or an allocated country when their country did not have a DEISA partner). In special cases, they could directly contact the user support of the site(s) on which their application run, called the *Execution Site(s)*. A trouble ticket system already existed at this time but was only used to keep track of internal problems, not for user related ones.

In the reporting period, which was also the beginning of the DEISA2 project, a general discussion was started within WP6 on how to change, refine and improve the User Support Service as practiced in DEISA1. Also WP3 T3.5f was closely involved in these discussions as there was a close relationship between the Help Desk and the definition of the renewed *DEISA Trouble Ticket System* (DTTS), based on the *Request Tracker* [7] tool, also used within the Operations group.

The objectives were to define an organisation both easy and convenient to use for the users, allowing distribution of responsibilities, speeding up of replies, escalation mechanisms and archiving of information. It appeared also important to offer to the users a more centralized view, as this is convenient for a unified project, with a common user interface for user support when the applications are in production mode, but, nevertheless, keeping great flexibility. This is why we tried to build a Help Desk organization *centrally managed but locally supported*.

The new organization of the Help Desk was the subject of a proposal submitted in October to the *DEISA Executive Committee* (DEC or ExeCom). In this proposal, a number of practices were formalized and common procedures were formulated for the reporting of user problems and the subsequent handling of these. A central Help Desk e-mail address and the usage of the *DEISA Trouble Ticket System* were introduced in this proposal as the key tools to facilitate the handling of user problems. A revolving Help Desk duty is proposed to monitor the DTTS and make sure that all user problems are handled appropriately.

In the proposal, we also described the separation of responsibilities between the DECI project support that is provided by the *Application Task Force* (WP5/7) and the problem support that is provided by the Help Desk, to be used when the applications have reached the *production* mode, that is to say, after the installation and enabling phases of the projects.

Upon acceptance of this proposal by DEC, the revolving Help Desk duty roster will be implemented immediately. Accordingly, the procedures will of course be described in the user documentation.

6 User Documentation

The first version of the user documentation was initially released in February 2005, as the *DEISA Primer*. It has been continuously updated since this moment, around two times a year, with six updates in total, according to the evolution of the hardware and software infrastructure, the evolution and improvements of the existing services and the introduction of new ones.

This process is of course an absolute requirement and will be continued during the complete lifetime of DEISA2. But two important changes are currently ongoing. Firstly, it has become clear for some time that to regularly include new material in a single document becomes unmanageable. Indeed, a huge documentation is rather dissuasive for a user. And, secondly, a long and bothersome process was required to convert the original file format, used to automatically generate the PDF file for printing, into the XML file usable for the on-line version which must be accessible on the Web site, as this process was only semi- automatic, and required that many tiny customizations and conversions be introduced manually.

This is why it was first decided to split the documentation in various different documents, keeping only a light Primer which describes:

- the organization of the project,
- the hardware infrastructure,
- the procedures to access to it,
- the organization of the file systems,
- the way to access to user support and documentations.

And separate manuals will be devoted to each of the main services (DART, DCPE, Batch systems, UNICORE, DESHL, Globus supported services, etc.), giving both the presentation of the service and all the relevant technical details.

In parallel, a new tool, having the functionalities of a collaborative editor, is currently studied to ease both the maintenance of the documents between several writers and reviewers, and the conversion process between the two required PDF and XML formats.

7 Advanced Application Production Assistance

7.1 Objectives

As described in the *Description of Work*, the main objective of this task T6.5 (*Advanced Application Production Assistance*) is to support the DEISA interfaces and portals which hide the complexity of the infrastructure, as well as the workflow tools like UNICORE and DESHL [3], and to provide assistance for data handling, including the development of procedures for data-analysis if needed.

The work in this task is dependent on both the internal development work in DEISA2 (WP3 Operations and WP4 Technology) and eDEISA (for the portals activity) and the user projects (DECI projects) and their needs, as well as the needs of the *Virtual Communities*. The work is also related to application enabling (WP5) but not concerning the applications themselves.

7.2 Expected strategy and actions

During the reporting period, the DECI-2007 projects have been given assistance in DEISA tools, such as UNICORE, DESHL, based on the projects' assignments in 2007 at each site. As the use of these tools is not mandatory to run the DECI projects, only a selection of projects have needed this assistance.

The DECI-2008 proposals will be used as a basis for future work in T6.5. The *Technical evaluations* of the proposals, which have partly been carried out by the WP6 staff during the reporting period, provide the necessary preliminary data for each site to provide assistance in advanced DEISA services, such as the UNICORE, DESHL and portals. The need for these tools has been explicitly assessed in the evaluation forms. However, the proposal about the DECI-2008 projects to accept was made at the end of September by the *Application Task Force* (ATaskF) and the decisions will be made by the DEC at the end of October. The respective Home sites are responsible for the work in task T6.5. The ATaskF of the Home site shall contact the *Principal Investigators* (PI) of the DECI projects and plan for the support.

The first organized *Virtual Communities* in 2008 will be EFDA (*European Fusion Development Agreement*), EUFORIA, and VIROLAB. Most of the users, especially in EFDA and EUFORIA are well experienced. The respective Home sites of the *Virtual Communities* (RZG for both EFDA and EUFORIA and EPCC for VIROLAB) will have the main responsibility from the T6.5 point of view and will take in charge the contacts, discussions and actions in coordination with the task leader.