



DELIVERABLE D2.3
PART VI: TESTING AND INTEGRATION OF THE
CROSSGRID APPLICATION PROGRAMMING
ENVIRONMENT

Task 2.5 Integration, Testing and Refinement

Document Filename:	CG2.5-D2.3-v0.3-TUM004-PrototypeDoc.doc
Work package:	WP2 Grid Application Programming Environment
Partner(s):	TUM, CYFRONET, FZK, USTUTT, UCY, CSIC, U.S.C
Lead Partner:	TUM
Config ID:	CG2.5-D2.3-v0.3-TUM004-PrototypeDoc
Document classification:	PUBLIC

Abstract: This document summarizes the testing and integration activities that have been carried out in WP2 in the reporting period M9-M12. It mainly presents aspects, which are relevant for several tasks of WP2. Task specific information on the testing of the individual tools developed in WP2 is presented in parts II, III, IV and V of this deliverable.

Delivery Slip

	Name	Partner	Date	Signature
From	Roland Wismüller	TUM	22/05/2003	
Verified by				
Approved by				

Document Log

Version	Date	Summary of changes	Author
0.1	22/05/2003	Draft version	Roland Wismüller
0.2	27/05/2003	Added executive summary	Roland Wismüller
0.3	27/05/2003	Added section 3 with technology vs. application matrix	Holger Marten

CONTENTS

EXECUTIVE SUMMARY	4
1. TESTING OF FIRST PROTOTYPES	5
2. FIRST INTEGRATION OF TOOL PROTOTYPES	6
3. TECHNOLOGY VERSUS APPLICATION MATRIX	8

EXECUTIVE SUMMARY

This document reports on the activities concerning testing and integration in WP2, which are grouped under the umbrella of task 2.5. The document is structured into two parts:

- Section 1 deals with the testing of the first WP2 prototypes. Since these testing activities are most closely related to the individual tasks of WP2 (task 2.2, 2.3, and 2.4), they are already reported in parts II, III, IV and V of this deliverable. Thus, Section 1 merely refers to these reports.
- Section 2 reports on the activities and results of the first WP2 integration meeting, which took place in Santiago de Compostela, on Feb. 7-9, 2003. The main result of this meeting was that all tools could be installed on some site of the CrossGrid testbed, and could successfully be demonstrated during the general meeting, which took place a few days later.
- Section 3 summarizes the current and future use of the WP2 tools by the applications of WP1 in the form of a technology/application matrix.

1. TESTING OF FIRST PROTOTYPES

Since according to Annex I, the first prototypes of the tools developed in WP2 should still be separated, independent tools, the testing of these tools within task 2.5 has been carried out mainly by those project partners responsible for their development. Thus, the presentation of the testing methodology and the results of the first tests has been integrated into parts II, III, IV, and V of this deliverable.

In particular, the tests and the resulting issued are described in the following parts and sections of this deliverable:

- **MARMOT (task 2.2)** Part II, Sections 6 and 7
- **GridBench (task 2.3)** Part III, Sections 7 and 8
- **G-PM (task 2.4)** Part IV, Sections 6 and 7
- **PPC (task 2.4)** Part V, Sections 6 and 7

Although some issues were identified for each of the tools, all of them essentially fulfilled the requirements defined for the first prototypes in deliverable D2.1.

According to the plans for deliverables of WP2, a more detailed report on testing, feedback, issues, new requirements and resulting extensions of the tools' design is contained in deliverable D2.4 which is due at project month M15.

2. FIRST INTEGRATION OF TOOL PROTOTYPES

The first integration of the tool prototypes into the CrossGrid testbed was carried out during an integration meeting in Santiago de Compostela on Feb. 7-9, 2003. The main purposes of this meeting were:

- Installing the WP2 tool at at least one site of the CrossGrid testbed,
- Testing the WP2 tools with at least one application from WP1,
- Integrating the G-PM tool with the OCM-G monitoring system developed in task 3.3,
- Preparing a demonstration of all WP2 tools for the CrossGrid General Meeting that took place the next days.

Essentially, all these goals could be achieved. The meeting turned out to be very productive and extremely helpful for the integration task. Thus, it was decided later to repeat these kinds of work-meetings regularly during the next phases of the project.

In particular, the following activities have been carried out:

- The MARMOT tool developed in task 2.2 was first installed on a RedHat 7.2 machine at IFCA, due to the compatibility with the gcc compiler version (MARMOT requires gcc version 3.1 or higher). On that first installation MARMOT was made working with MPICH-G2 on the IFCA cluster. It was tested with some test applications and with the HEP (distributed ANN) application from WP1. It was checked that the output from MARMOT had the expected shape.

In order to make MARMOT run on the current CrossGrid testbed, it was installed also on a RedHat 6.2 machine (UI at IFCA). It was necessary to install gcc-3.1 and reinstall completely Globus2 and MPICH-G2 using this compiler. Doxygen was also needed for making the documentation. Then, MARMOT was tested as before, using the HEP application. It was found that the size of the binary executable was 5 times greater when compiling with MARMOT.

After some modifications in the integration process, RPM packages have been created to simplify distribution.

Finally, the information about how MARMOT was used with HEP application has been commented to the application developers and they are planning to implement an option on the portal job submission to define if MARMOT is used or not and somehow show the output when used.

- The efforts concerning GridBench (task 2.3) were focused on three main tasks:
 1. Adaptation of the build process so that the prototype could run using MPICH-G.
 2. Packaging of the prototype into an RPM.
 3. Creation of RSL for specifying the execution of the prototype benchmark.

The build process for the prototype was modified so that it could build and run on the CrossGrid testbed. Before the integration meeting the prototype would build and run successfully on a local cluster using the stock distribution of MPICH (using the 'ch_p4' device), and a generic distribution of the 'libblas' linear algebra library. In order to run the prototype, which is an MPI application, we had to build it using the MPICH 'globus' device. Since MPICH is not part of the EDG distribution, it is not installed on many of the testbed machines. To overcome this problem we had to statically link the executable. The build process was also altered so that the prototype would compile using the 'libblas' that is available on the testbed machines and is part of the 'cern-0.1.rpm' package.

Once the compilation and linkage of the prototype was successful, the executable was packaged into an rpm. The target directory of the RPM was '/opt/cg' and the RPM package contains only two files:

`/opt/cg/bin/gb_site_hpl` – the statically linked executable
`/opt/cg/share/doc/gridbench-0.1/HPL.dat` – a sample input file

The RPM was installed on the development workstation.

During the integration meeting, it was decided to resort to RSL for executing the prototype since JDL did not support MPI jobs. Furthermore, the job management software of the Computing Elements is not specifically configured to support MPI jobs (the ‘count’ RSL attribute was not used correctly). To overcome this, a script was written to aid the generation of RSL (where each process was mapped to a Globus sub-job, and the executable was staged manually). Jobs were submitted directly to the Computing Elements and not the Resource Broker, since it lacked support for MPI.

- The PPC tool developed in task 2.4 has been installed at the user interface machine at CESGA. An issue that arose was that the packages PIPlot and TCL/TK had to be previously installed in order to compile the tool. This dependency will be avoided in the next version of the GUI, which will be implemented as a Java applet; this work is in progress. The implemented kernels were successfully tested by the tool assuming an homogenous system. No input from monitoring task were available at the time. An integration with the CrossGrid job submission services was not required, since this tool does not require the submission of any job to the Grid.
- The G-PM tool developed in task 2.4 has been installed at a user interface machine at CYFRONET. The compilation worked without problems, since G-PM uses GNU autoconf to figure out the software configuration of the target machine. However, there is a dependency to the GTK and GTK+Extra packages that must be available. These packages will have to be included in the CrossGrid distribution for the user interface machines. After OCM-G was installed at the compute nodes at CYFRONET, the first integration between G-PM and OCM-G took place. Several minor issues have been detected and could be fixed immediately, mostly they were caused by incorrect and/or incomplete use of the OMIS interface. In addition, some bugs in OCM-G were uncovered and fixed. After the successful integration, G-PM and OCM-G have been tested both with special test applications and with the HEP code (distributed ANN) from task 1.3. Again, some minor problems have been found and were fixed immediately.

Further integration issues, especially the detailed plan for the integration between the tools and the execution and programming environment, will be discussed in deliverable D2.4, in accordance with the plans outlined in Annex I.

3. TECHNOLOGY VERSUS APPLICATION MATRIX

The following table summarizes the current and future use of the WP2 tools by the applications of WP1 in the form of a technology/application matrix. Four stages of usage are distinguished: “applicable to” means that the tool can be applied to the program developed in that WP, “tested with” means that the tool prototype was tested with the program of that task by the tool developers, “used by” means that the tool prototype was used by the application developers and applied to the developed application of that WP, and “will be used by” means that the developers expressed the need or interest to use the tool.

Please note, that especially the columns “tested with” and “already used by” present a snapshot after project months 12-15, i.e. 6-9 months after starting the tools development and at a time, where, according to Annex 1, neither for the tools nor for the applications full (Grid) functionalities are available. More details on the usage of the tools and feedback from users will be presented in deliverable D2.4 after project month 15.

WP2 Tool	Application	Applicable to	Tested with	Already used by	Will be used by
WP 2.2 MARMOT	WP 1.1 BioMed	Yes	No ¹		No
	WP 1.2 Flooding	Yes	No ¹	No? ²	No? ²
	WP 1.3 HEP	Yes	Yes	Yes	Yes
	WP 1.4 METEO	Yes	Yes	Yes	Yes
	WP 2.3 Benchmarks	Yes	Yes ³	No ³	Yes
WP 2.3 GridBench	WP 1.1 BioMed	Yes	No ⁴		No
	WP 1.2 Flooding	Yes	No ⁴		Yes
	WP 1.3 HEP	Yes	No ⁴		Yes
	WP 1.4 METEO	Yes	No ⁴		No? ²
WP 2.4.1 G-PM	WP 1.1 BioMed	Yes			Yes
	WP 1.2 Flooding	Yes			Yes
	WP 1.3 HEP	Yes	Yes	Yes	Yes
	WP 1.4 METEO	Yes			No? ²
	WP 2.3 Benchmarks	Yes			Yes
WP 2.4.2 PPC	WP 1.1 BioMed	No ⁵			No
	WP 1.2 Flooding	Yes	Yes ⁶	No	Yes
	WP 1.3 HEP	Yes	No	No	Yes
	WP 1.4 METEO	Yes	Yes	Yes	Yes
	WP 2.3 Benchmarks	Yes	No	No	Yes
	WP 2.4.2 special kernel library Paraiso	Yes	Yes	Yes	Yes

¹ Application not available in cvs

² No user feedback has been provided yet

³ The focus of the first prototype was on functionality and availability for the applications of WP1. Performance tuning and testing with the benchmarks is a target for the next version.

⁴ Implementation of application kernels into benchmarks not yet finalized.

⁵ Application that does not have computational intensive kernels. So, no kernels will be provided. Their main topic is to communicate huge sets of data.

⁶ Application kernel available; testing currently under development.