D3.5 Test Suite

Project acronym: *MAPPER*

Project full title: Multiscale Applications on European e-Infrastructures.

Grant agreement no.: 261507

Due-Date:	30.9.2012
Delivery:	30.9.2012
Lead Partner:	LMU
Dissemination Level:	Public
Status:	Final
Approved:	Q Board, Project Steering Group
Version:	1.0

DOCUMENT INFO

Date and version number	Author	Comments
1.8.2012 v1	M. Schiffers	Initial draft
13.9.2012 v2	R. Pajak and M. Schiffers	Added appendix and several corrections
19.9.2012	Internal reviewers	Comments and remarks
28.9.2012	M. Schiffers	Final version after integration of the internal reviewer's comments

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

Executive Summary

The purpose of the MAPPER Test Suite is to provide a framework for examining MAPPER multiscale applications related to the MAPPER Multiscale Base Case (MBC) [2] against a set of objectives expressed by the MAPPER Profile [2]. As MAPPER applications are focusing on solutions requiring computational and storage Grid resources (provided as part of the European PRACE and EGI infrastructures [3, 4]), a well-defined and standards compliant access to and usage of these resources is an important requirement. Consequently, the MAPPER Test Suite provides the context for testing such applications against the standards defined by the MAPPER Profile [2].

Hence, the MAPPER Test Suite is not a newly developed test conduction engine. Rather, it is a methodology for claiming conformance with the MAPPER Profile [2].

1 Introduction

The purpose of any test suite is to provide a (more or less standardized) framework for examining a system under test (SUT) against a set of objectives, typically expressed by test cases. Test suites may be applied manually or automated, the latter usually requiring the functionality of a test harness and a sufficiently detailed level of communication between the various components.

As MAPPER applications are centring on multiscale applications with computational and storage resources to facilitate efficient model coupling, the coordinated and secure access to resources is an important requirement. Consequently, the MAPPER Test Suite (MTS) provides the context for testing such applications against the standards defined by the MAPPER Profile [2]. Thus, MTS is a key ingredient for ensuring the quality of MAPPER applications. However, MTS is not a newly developed test conduction engine. Rather, it is a methodology for claiming conformance with the MAPPER Profile [2].

Technically speaking, MTS consists of a collection of test cases that are intended to be used for assessing the behaviour of MAPPER services and applications together with the respective documentation. The test cases themselves are represented by sets of conditions or variables (typically expressed as <input, output> pairs) which determine whether a MAPPER application is behaving as expected or not relative to the MAPPER Multiscale Base Case and the MAPPER Profile [2].

Section 2 describes the test suite background by briefly recapping the MAPPER Multiscale Base Case and the European e-Infrastructures MAPPER application are intended to be deployed to. Section 3 describes the compliance testing for MAPPER applications. Section 4 gives an example of applying the test suite to the Multiscale Application Skeleton (MASK). Finally, section 5 concludes the deliverable. Detailed test case descriptions for the MASK example are listed in the appendix (section 9).

2 Background

This section sets the context for the MAPPER Test Suite. It briefly describes the MAPPER Multiscale Base Case, the MAPPER target infrastructure, and the MAPPER Profile [2]. The profile consists of basic execution services, job description definitions, data staging definitions, security definitions, and conformance claiming rules.

2.1 MAPPER Multiscale Base Case

The MAPPER Multiscale Base Case (MBC) is specified in detail in [2]. It describes the simplest MAPPER use case as a single submodel of a multiscale application running as a job on a single Grid Computing element (which may be a high performance computing (HPC) system). A typical multiscale application consists of software modules simulating certain phenomena in certain time or space scale (scaleful), typically requiring HPC resources and often (but not always) implemented as parallel programs, and software modules that convert data from one scaleful module to another, all based on appropriate communication structures (master/worker, peer-to-peer, pipe, hybrid) and execution paradigms (stateless, stateful).

MBC assumes that applications are preinstalled, and often optimized, by the user at the target site where the quantitative and qualitative properties of the system resources (e.g., by means of Service Provider documentation) are well-known and well described. MBC also assumes that failing jobs must be resubmitted by the client (as a new job) because the job scheduler will not automatically rerun the job. In addition, MBC considers several other aspects to be out-of-band or out-of-scope:

- Users are assumed to possess the necessary credentials needed to submit both data and work to the Grid components.
- The Grid's job scheduler is assumed to be reachable at a well-known communications endpoint.
- Resource allocation is out of band.
- Scheduling policies is out-of-scope.
- Service Level Agreement (SLA) handling is out-of-scope.
- From the point of view of a single Resource Provider jobs are completely independent of each other. Dependencies among jobs are managed at a higher level.
- A job consists of a single program running on a single Grid element, whether it is a sequential program, a single node application parallelized using OpenMP, or a multi-node MPI job. The process of reservation is out-of-scope of the MAPPER Base Case. It might be a call to the system supporting advance reservation interface (e.g. QCG-Computing[1]) or a manual process involving human interactions with the system administrator (e.g., via mail or phone call – a classic out of bound channel). MBC only assumes that reservation identifiers are known to the system in advance.

[1] http://www.qoscosgrid.org/trac/qcg-computing

2.2 Quality-Assured Deployment to the MAPPER Target Infrastructure

MAPPER applications require the integration of multiple computational and storage resources under the constraints of efficient model couplings. To fulfil these requirements the MAPPER project utilizes the capabilities of European e-Infrastructures augmented by components and services necessary for facilitating the execution of multiscale applications (known as the MAPPER e-Infrastructure). The MAPPER e-Infrastructure is described in detail in [2] with the MAPPER software infrastructure as depicted in Figure 1 and the e-Infrastructures as shown in Figure 2. To deal with the coupling requirements of the applications [1], the MAPPER e-Infrastructure deploys a selection of services (see Figure 1) into resource centres (sites) distributed among the partners of the project in several European countries (see Figure 2).

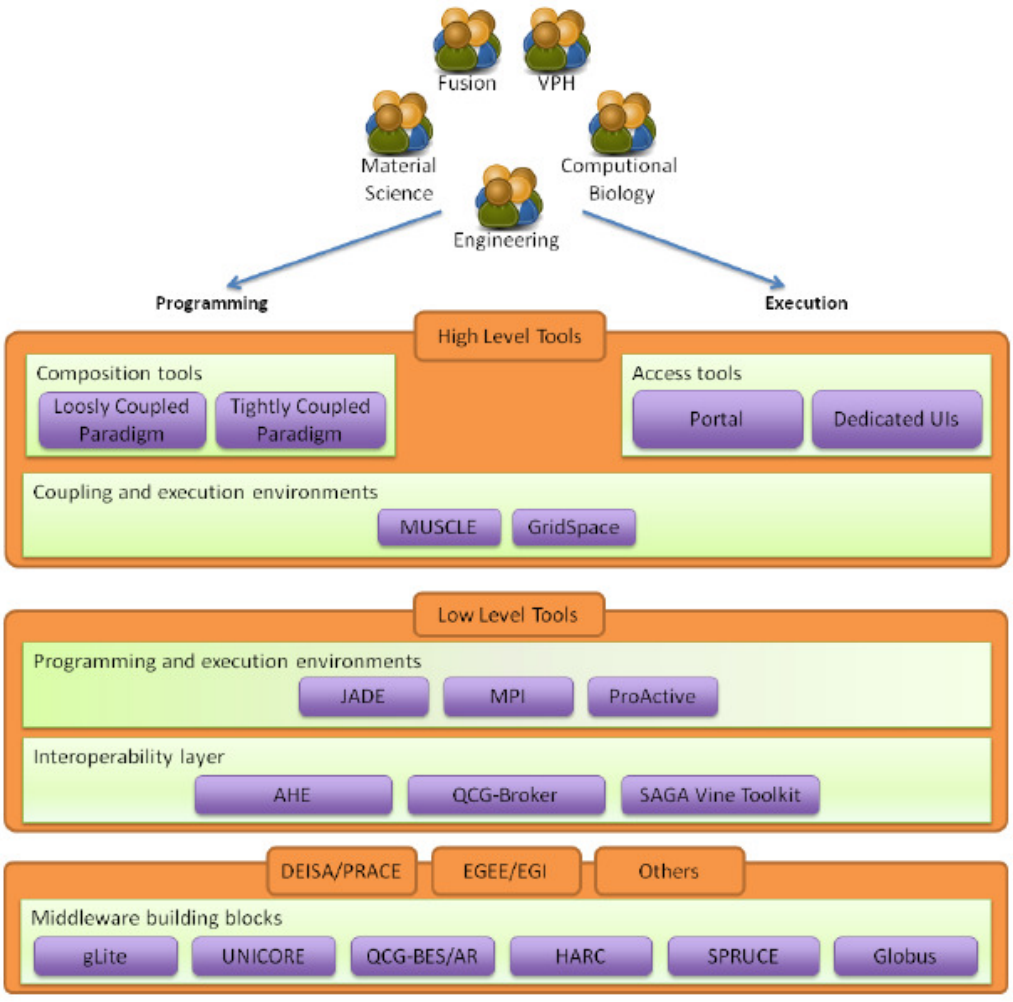

Figure 1: Overview of the MAPPER software infrastructure [2]

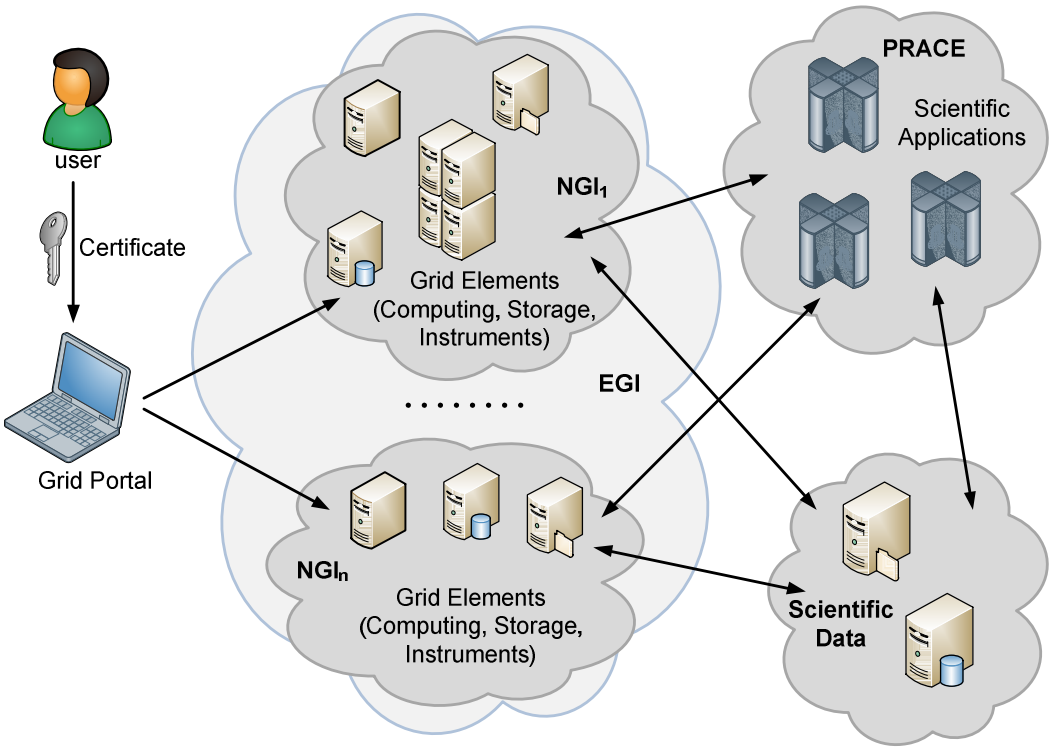

Figure 2: E-Infrastructures

The middleware architecture proposed for MAPPER aims at setting up a highly configurable, user-centric and open source development environment that adopts best practices from a number of previous Grid projects as far as its Grid support and the development of applications/services are concerned. The middleware architecture consolidates tools, services and solutions not only for multiscale applications but also for the respective resource management, the application development, the workflow management and for the parallelization of applications.

3 MAPPER Profile Compliance Testing

Conformance claims against the MAPPER Base Case Profile can only be made using the mechanisms described in the WS-I Conformance Claim Attachment Mechanisms (http://www.ws-i.org/Profiles/ConformanceClaims-1.0-2004-11-15.html) provided the applicable profile requirements have been met. Using these mechanisms developers need to check the compliance with the MAPPER Base Case Profile, described in sections 3.1-3.2 of deliverable D3.3 (MAPPER Profile) [2]. Compliance checking thus does not refer to functional testing. It rather primarily relates to standards compliance, either de facto ones or project defined ones. An example of the latter category is the compliance of the HemeLB

application with the MPWide light-weight communication library[2] under the open source Jenkins integration tool[3]. An example of the former category is the conformance of Grid job executions according to the Open Grid Services Architecture (OGSA) Basic Execution Services (BES) Factory requirements [2]. Accordingly, appropriate test procedures and test setups have to be developed either by standardization bodies or by software developers (external to MAPPER or internal). Because MAPPER relies on both standardized and non-standardized interfaces, both categories will be discussed briefly.

3.1 Testing the Conformance to Grid Standards

MAPPER adheres to several Grid standards (see [2] and [3. These are maintained by the Open Grid Forum (OGF) and their respective working groups. Especially, MAPPER leverages the work of the following working groups:

- Distributed Resource Management Application API (DRMAA)
- Simple API for Grid Applications (SAGA)
- High Performance Computing Profile (HPCP)

Consequently, testing the conformance to standards means compliance testing as required by the respective working groups. Unfortunately, as of this writing, not all working groups provide a respective test suite. If they do, however, MAPPER adheres to them. If they do not, MAPPER provides best practices to follow. An addendum to this deliverable at the end of the project will list the as-then-available test suites.

3.1.1 DRMAA

“The DRMAA OGF working group maintains a testsuite, which allows to check existing DRMAA C implementations for compliance with the specification. A DRMAA implementation should point out (e.g. in a README file) to which version of the test suite it complies. The version number contains of major version, minor version, and patch level. The major version expresses the regarding version of the DRMAA specification. The minor version expresses the amount and semantics of tests performed by the suite. The patch level is increased in case of bug fixes or adjustments of the implementation. A DRMAA implementation shall call itself "DRMAA-compliant" only if all tests of the suite are passing.”[4]
The DRMAA test suite is available from http://www.drmaa.org/testsuite.php (from which the citation is borrowed as well). Figure 3 contains a sample DRMAA test case.

[2] http://castle.strw.leidenuniv.nl/software/mpwide.html

[3] http://jenkins-ci.org/

[4] Cited from the DRMAA test suite site.

```
[drmaa@grass1 drmaa_testsuite]$ ./test_drmaa
usage: test_drmaa <test_case>
  <test_case> is one of the keywords below including the enlisted test case arguments
        ALL_AUTOMATED                                <sleeper_job> <exit_arg_job> <kill_arg_job> <email_addr>
        ST_MULT_INIT
        ST_MULT_EXIT
        ST_SUPPORTED_ATTR
        ST_SUPPORTED_VATTR
        ST_VERSION
        ST_DRM_SYSTEM
        ST_DRMAA_IMPL
        ST_CONTACT
        ST_EMPTY_SESSION_WAIT
        ST_EMPTY_SESSION_SYNCHRONIZE_DISPOSE
        ST_EMPTY_SESSION_SYNCHRONIZE_NODISPOSE
        ST_EMPTY_SESSION_CONTROL                     DRMAA_CONTROL_*
        ST_SUBMIT_WAIT                               <sleeper_job>
        ST_BULK_SUBMIT_WAIT                          <sleeper_job>
        ST_BULK_SINGLESUBMIT_WAIT_INDIVIDUAL         <sleeper_job>
        ST_SUBMITMIXTURE_SYNC_ALL_DISPOSE            <sleeper_job>
        ST_SUBMITMIXTURE_SYNC_ALL_NODISPOSE          <sleeper_job>
        ST_SUBMITMIXTURE_SYNC_ALLIDS_DISPOSE         <sleeper_job>
        ST_SUBMITMIXTURE_SYNC_ALLIDS_NODISPOSE       <sleeper_job>
        ST_EXIT_STATUS                               <exit_arg_job>
        ST_SUBMIT_KILL_SIG                           <kill_arg_job>
        ST_INPUT_FILE_FAILURE                        <sleeper_job>
        ST_OUTPUT_FILE_FAILURE                       <sleeper_job>
        ST_ERROR_FILE_FAILURE                        <sleeper_job>
        ST_SUBMIT_IN_HOLD_RELEASE                    <sleeper_job>
        ST_SUBMIT_IN_HOLD_DELETE                     <sleeper_job>
        ST_BULK_SUBMIT_IN_HOLD_SESSION_RELEASE       <sleeper_job>
        ST_BULK_SUBMIT_IN_HOLD_SINGLE_RELEASE        <sleeper_job>
        ST_BULK_SUBMIT_IN_HOLD_SESSION_DELETE        <sleeper_job>
        ST_BULK_SUBMIT_IN_HOLD_SINGLE_DELETE         <sleeper_job>
        ST_SUBMIT_POLLING_WAIT_TIMEOUT               <sleeper_job>
        ST_SUBMIT_POLLING_WAIT_ZEROTIMEOUT           <sleeper_job>
        ST_SUBMIT_POLLING_SYNCHRONIZE_TIMEOUT        <sleeper_job>
        ST_SUBMIT_POLLING_SYNCHRONIZE_ZEROTIMEOUT    <sleeper_job>
        ST_ATTRIBUTE_CHANGE
        ST_SUBMIT_SUSPEND_RESUME_WAIT                <sleeper_job>
        ST_USAGE_CHECK                               <exit_arg_job>
        MT_SUBMIT_WAIT                               <sleeper_job>
        MT_SUBMIT_BEFORE_INIT_WAIT                   <sleeper_job>
        MT_EXIT_DURING_SUBMIT                        <sleeper_job>
        MT_SUBMIT_MT_WAIT                            <sleeper_job>
        MT_EXIT_DURING_SUBMIT_OR_WAIT                <sleeper_job>
        ST_GET_NUM_JOBIDS                            <sleeper_job>
        ST_BULK_SUBMIT_TNCRPH                        <sleeper_job>
```

Figure 3: Sample DRMAA test case

3.1.2 SAGA

SAGA does not provide a standardized conformance test suite. Therefore, conformance to the SAGA API may be tested using several methods. One example is given in https://github.com/saga-project/saga-adaptors-bes, another example can be found in http://grid.in2p3.fr/software/jsaga-dev/testers-guide.html. All such SAGA Test Suites eventually prove the conformance of a SAGA application with the SAGA standard.

3.1.3 HPCP and others

There is no standardized test suite for the HPC Basic Profile[5] and other OGF profiles like the Basic Security Profile[6]. Conformance testing against these profiles is to be performed manually.

[5] http://www.ogf.org/documents/GFD.114.pdf

[6] http://www.ws-i.org/profiles/basicsecurityprofile-1.0.html

3.2 Testing the Conformance to Non-Grid Standards

MAPPER applications fall into the categories fusion, hydrology, physiology, nanomaterial and computational biology [1]. In order to develop and execute applications in these categories, MAPPER looks at standards, quasi standards and resources discussed in the bodies and organizations like (see also [3])

- SBML.org for the Systems Biology Markup Language (SBML) standard
- BioPAX.org for the Biological Pathways Exchange (BioPax) standard
- SBGN.org for the Systems Biology Graphical Notation (SBGN) standard
- There are also several ontology bases considered as quasi standard:
    - The Gene Ontology (GO) Consortium provides ontologies of defined terms representing gene product properties
    - the Open Biological and Biomedical Ontologies (OBO) support the development and publication of ontologies in the biomedical domain
- The BioModels Database for peer-reviewed, published, computational models
- The Medical Imaging and Technology Alliance (MITA), a division of the Association of Electrical and Medical Imaging Equipment Manufacturers (NEMA) for the standard on Digital Imaging and Communications in Medicine (DICOM)
- The FieldML (Field Modelling Markup Language) for supporting the building of hierarchical models represented by generalized mathematical fields in a standardized way
- The CellML language as an open standard maintained by the Auckland Bioengineering Institute at the University of Auckland and affiliated research groups. The purpose of CellML is to store and exchange computer-based mathematical models.

Typically, there are no standardized test suites for these standards; conformance testing thus has to be performed manually. However, there is an increasing number of templates to base these tests upon.[7]

3.3 Testing in Non-Standards Environments

Testing in non-standardized environments (like e.g., the nano-materials coupling suite or the AHE) is not based on standardized procedures. Rather, it is based on best practices and state-of-the-art methodologies well-known from general software engineering. A comprehensive description is therefore out-of-scope of this report.

[7] For example CellML offers a template for DOM tests (http://cellml-api.hg.sf.net/hgweb/cellml-api/cellml-api/diff/35be026fa6f5/tests/DOMTest.cpp). DICOM offers test images under http://www.dicom-solutions.com/testsuite.shtm?lang=en.

3.4 Test Procedure

Although compliance test procedures form an integral part of all MAPPER development activities, MAPPER does neither prescribe any specific development paradigm (like agile development, extreme development or similar ones). Nor does MAPPER prescribe any specific methodology for software development and software testing. Generally speaking, every evaluation process starts at the development sites. The component developers follow their best practices in coding and preparing the compliance tests as well as the documentation accompanying all compliance claims for their components. All test cases are well documented and saved for reproducibility purposes.

All test cases will be saved in the restricted project wiki (http://www.mapper-project.eu/web/guest/wiki) which also serves as the project's bug tracker system. In case of urgent and critical issues MAPPER accepts a shortcut via e-mail, telephone, or Skype.

4 Example

This section contains an example of how to apply MTS for compliance testing.

4.1 MASK

As described in [4] an auxiliary Multiscale Application Skeleton (MASK) tool for creating multiscale applications skeletons (i.e., "empty" multiscale applications with the same structure as real ones (number and type of modules, execution type etc.)) has been developed. The functionality of the skeleton is placed between the structure description and the implementation of the application. From the point of view of multiscale programming and execution tools, the created skeleton is a running application. MASK is used to create a simple multiscale application to test the MAPPER tools (and can thus be used as part of the compliance test suite).

The MASK application consists of three scaleful single scale modules and two MAPPER modules used for connecting them. It simulates a very simple behaviour of an ant and an elephant. The structure of the application is hybrid as the loosely coupled part is followed by a tightly coupled one as shown in Figure 4.

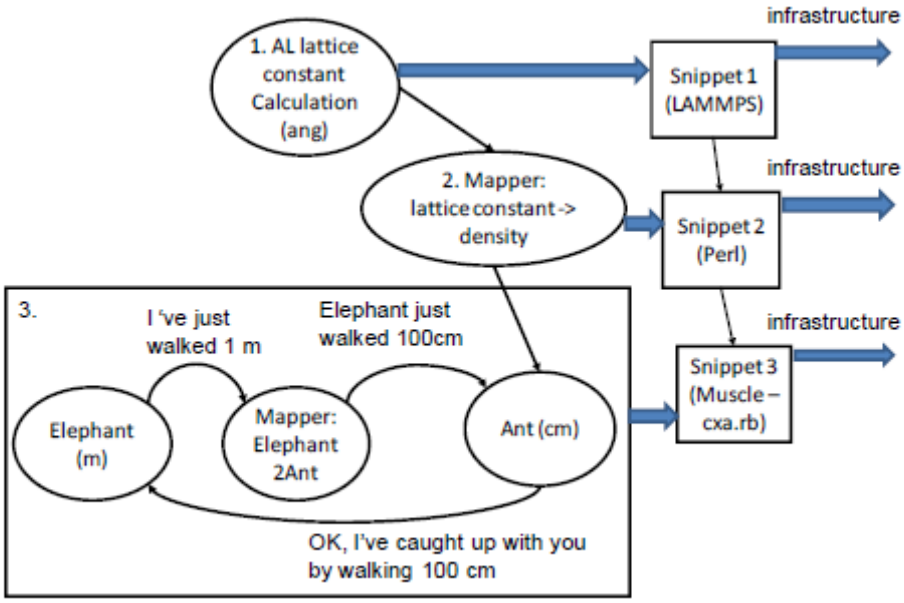

Figure 4: Ant and elephant test application generated by MASK

A typical testing scenario would then be as follows (using nano materials as an example):

1. The module in nano scale calculates the lattice constant of aluminium atoms.
2. The scaleless MAPPER transforms the nano scale information into g/cm scale (density of aluminium)
3. The module called Ant calculates the weight of aluminium it carries from density obtained in the previous step (scale: centimetres)
4. The Ant walks after the Elephant in a tightly coupled manner:
    a. The Elephant walks a 1m/step and sends information back to the Ant (scale: meters).
    b. The Ant walks 100cm and confirms its progress to the Elephant.

Some remarks:

1. The codes of the MASK prototype are freely available from https://github.com/kzajac/mask.
2. An example of the DSL (Domain Specific Language) for the simple application scenario can be found at https://github.com/kzajac/MASK/tree/experimental/examples.
3. MASK was also deployed in the current version of the GridSpace Experiment Workbench http://gs2.mapper-project.eu as one of the available interpreters.
4. The demonstration videos on how to use the skeletons are available at http://www.youtube.com/user/dicecyfronetpl.

4.2 Conformance Claiming

Conformance claims against the MAPPER Base Case Profile can now be made using the mechanisms described previously. The corresponding test cases are listed in the appendix (test cases TC-1 through TC-8). The complete list of the appendix test cases is:

Case ID	Description
TC-1	Conformance to the HPC Basic Profile, Version 1.0 specification
TC-2	Implementation of the *CreateActivity* operation
TC-3	Implementation of the *GetActivityStatuses* operation
TC-4	Implementation of the *TerminateActivities* operation
TC-5	Service accessibility over an HTTPS or HTTPG (HTTP over GSI) endpoint
TC-6	Support of notifications of activity statuses over a WS-Notification
TC-7	Support/provision of an advance reservation management web service interface
TC-8	Acceptance of the activities described using the JSDL Version 1.0 standard
TC-9	Capability of submitting a job into an advance reservation
TC-10	Availability of the GridFTP servide at the target site
TC-11	Access to the file system granted to the GridFTP service
TC-12	X.509 proxy certificates accepted by the target service
TC-13	Certificates issued by the CAs accepted by the target site
TC-14	The existence of the one incoming open port on the frontend/interactive node.
TC-15	Not restricted connectivity from worker nodes to the frontend/interactive node

In the following we demonstrate a typical conformance test using Web Service Definition Language (WSDL) compliance as an example.

4.2.1 WSDL 1.1 Conformance Claim

Conformance claims can be attached to a `wsdl:port` element in a WSDL 1.1[8] description as a child of its `wsdl:documentation` element, using the WS-I Conformance Claim Schema.

```
wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl"
  xmlns:tns="http://example.org/myservice"
  xmlns:soapbind="http://schemas.xmlsoap.org/wsdl/soap"
```

[8] http://www.w3.org/TR/wsdl.html

```
  xmlns:wsi="http://ws-i.org/schemas/conformanceClaim/"
  targetNamespace="http://example.org/myservice">
  <wsdl:portType name="MyPortType">
      ...
  </wsdl:portType>
  <wsdl:binding name="MyBinding" portType="MyPortType" >
      ...
  </wsdl:binding>
  <wsdl:service name="MyService" >
    <wsdl:port name="MyPort" binding="tns:MyBinding" >
      <wsdl:documentation>
        <wsi:Claim
         conformsTo="http://ws-i.org/profiles/basic/1.0" />
      </wsdl:documentation>
      <soapbind:address
       location="http://example.org/myservice/myport" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

4.2.2 Test Cases

A test case is usually a single step, or occasionally a sequence of steps, to test the correct behaviour or functionalities of an application. Accordingly, an expected result or expected outcome needs to be specified. Additional information that may be included in the specification contains:

- test case identifier
- test case title
- test case purpose
- test case description
- specification (the profile is based upon) to test against
- expected results
- author
- rights
- test category/group
- additional references

For protocol purposes other fields are important as well:

- observed results
- pass/fail indicator
- additional remarks

A detailed description of some test cases for MASK as performed by CYFRONET can be found in the appendix (section 9.).

5 Conclusion

The MAPPER Test Suite (MTS) provides a framework for examining MAPPER multiscale applications related to the MAPPER Multiscale Base Case against a set of objectives expressed by the MAPPER Profile. MTS relies on both standardized procedures and non-standardized best practices. As MAPPER applications are focusing on solutions requiring computational and storage Grid resources, a standards compliant access to and usage of these resources is necessary for claiming conformance to the MAPPER Profile. The MAPPER Test Suite provides the corresponding context. Profile test cases are collected in the MAPPER wiki.

6 Intellectual Property Statement

MAPPER takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the MAPPER Project Office.
MAPPER invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice the recommendations expressed in this document. Please address the respective information to the MAPPER Project Office.

7 Disclaimer

This document and the information contained herein is provided on an “As Is” basis and MAPPER disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

8 Abbreviations and References

8.1 Abbreviations

DRMAA	Distributed Resource Management Application API
HPCP	High Performance Computing Profile
MASK	Multiscale Application Skeleton
MBC	MAPPER Multiscale Base Case
MTS	MAPPER Test Suite
OGF	Open Grid Forum
SAGA	Simple API for Grid Applications
SUT	System Under Test

8.2 References

[1] MAPPER Description of Work, 2010

[2] MAPPER Deliverable 3.3: MAPPER Profile, http://www.mapper-project.eu/documents/10155/23400/D3.3-MAPPER+Profile.pdf, 2012

[3] MAPPER Deliverable D3.2: Standardization Roadmap and First Sustainability Plan, http://www.mapper-project.eu/documents/10155/23400/Deliverable+D3.2+Standardization+Roadmap+and+First+Sustainability+Plan?version=1.1, 2011

[4] MAPPER Deliverable D8.1: Description of the Architecture and Interfaces, http://www.mapper-project.eu/documents/10155/23479/D8.1+-+Architecture+and+Interfaces.pdf, 2011

9 Appendix: Description of profile compliance test cases for MASK

This appendix lists the MASK test cases in detail. The test cases are:

Case ID	Description
TC-1	Conformance to the HPC Basic Profile, Version 1.0 specification
TC-2	Implementation of the *CreateActivity* operation
TC-3	Implementation of the *GetActivityStatuses* operation
TC-4	Implementation of the *TerminateActivities* operation
TC-5	Service accessibility over an HTTPS or HTTPG (HTTP over GSI) endpoint
TC-6	Support of notifications of activity statuses over a WS-Notification
TC-7	Support/provision of an advance reservation management web service interface
TC-8	Acceptance of the activities described using the JSDL Version 1.0 standard
TC-9	Capability of submitting a job into an advance reservation
TC-10	Availability of the GridFTP servide at the target site
TC-11	Access to the file system granted to the GridFTP service
TC-12	X.509 proxy certificates accepted by the target service
TC-13	Certificates issued by the CAs accepted by the target site
TC-14	The existence of the one incoming open port on the frontend/interactive node.
TC-15	Not restricted connectivity from worker nodes to the frontend/interactive node

Identifier	TC-1
Title	Conformance to the HPC Basic Profile, Version 1.0 specification
Purpose	The interface must adhere to the HPC Basic Profile, Version 1.0 specification
Description	Claims of conformance to the HPC Basic Profile 1.0 can be made using the following mechanisms, as described in Conformance Claim Attachment Mechanisms, when the applicable Profile requirements associated with the listed targets have been met: The conformance claim URI for the Basic Profile 1.0 is as follows, as per

	the discussions in the "Security Considerations" section of this document: • Username Token - http://ogf.org/profiles/hpc-basic/1.0/username-token • X.509 Certificate Token - http://ogf.org/profiles/hpc-basic/1.0/x.509-certificate-token A claim of conformance MUST be made with at least one of these two tokens. A claim of conformance MAY be made with both of these two tokens. In addition, a claim of conformance MUST be made for the WS-I basic profile (http://ws-i.org/profiles/basic/1.1).
Specification tested	HPC Basic Profile, Version 1.0 specification
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	HPC Basic Profile / Basic Execution Service Requirements
See Also / References	http://www.ogf.org/documents/GFD.114.pdf

Identifier	TC-2
Title	Implementation of the *CreateActivity* operation.
Purpose	*CreateActivity* operation must be implemented by the *BES-Factory* port.
Description	The service provider must provide access to the hosted resources over the OGSA Basic Execution Service Version 1.0 compliant interface. The service must implement at least the *BES-Factory* port type. The *BES-Factory* port must implement the following operation: *CreateActivity* – an operation that creates a new activity (i.e. submit a job).
Specification tested	OGSA Basic Execution Service Version 1.0
Inputs	

Expected Results	The following code is present in Input:
	
	`<bes:ActivityDocument>`
	`<jsdl:JobDefinition>`
	`...`
	`</jsdl:JobDefinition>`
	`<xsd:any/> *`
	
	`</bes:ActivityDocument>`

Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	*BES-Factory* port type / Basic Execution Service Requirements
See Also / References	http://www.ogf.org/documents/GFD.108.pdf

Identifier	TC-3
Title	Implementation of the *GetActivitiesStatuses* operation.
Purpose	*GetActivitiesStatuses* operation must be implemented by the *BES-Factory* port.
Description	The service provider must provide access to the hosted resources over the OGSA Basic Execution Service Version 1.0 compliant interface. The service must implement at least the *BES-Factory* port type. The *BES-Factory* port must implement the following operation: *GetActivitiesStatuses* – an operation that provides the status of the previously submitted activity.
Specification tested	OGSA Basic Execution Service Version 1.0
Inputs	

Expected Results	The following code is present in Output:
	
	`<ActivityStatus>`
	`<ActivityIdentifier> {wsa:EndpointReferenceType}`
	`</ActivityIdentifier>`
	`<ActivityStatus> ActvityStateType </ActivityStatus> ?`
	`|`

	<SOAP-1.1:fault> fault </SOAP-1.1:fault> ? </ActivityStatus>*
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	*BES-Factory* port type / Basic Execution Service Requirements
See Also / References	http://www.ogf.org/documents/GFD.108.pdf

Identifier	TC-4
Title	Implementation of the *TerminateActivities* operation.
Purpose	*TerminateActivities* operation must be implemented by the *BES-Factory* port.
Description	The service provider must provide access to the hosted resources over the OGSA Basic Execution Service Version 1.0 compliant interface. The service must implement at least the *BES-Factory* port type. The *BES-Factory* port must implement the following operation: *TerminateActivities* – an operation that requests cancellation of the previously submitted activity. It is left to the implementation whether this operation has synchronous or asynchronous semantics.
Specification tested	OGSA Basic Execution Service Version 1.0
Inputs	
Expected Results	The following code is present in Output: <TerminateActivityResponse> <ActivityIdentifier>EPR</ActivityIdentifier> <Terminated> xsd:Boolean </Terminated> ? | <SOAP-1.1:fault> ... </SOAP-1.1:fault> ? </TerminateActivityResponse> *
Actual Results	

Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	*BES-Factory* port type / Basic Execution Service Requirements
See Also / References	http://www.ogf.org/documents/GFD.108.pdf

Identifier	TC-5
Title	Service accessibility over an HTTPS or HTTPG (HTTP over GSI) endpoint.
Purpose	The service must be accessible over an HTTPS or HTTPG (HTTP over GSI) endpoint.
Description	
Specification tested	The HTTP(S,G) and SOAP Server/Framework – Code and Usage Description
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	HTTPS or HTTPG accessibility / Basic Execution Service Requirements
See Also / References	http://www.nordugrid.org/documents/HTTP_SOAP.pdf

Identifier	TC-6
Title	Support of notifications of activity statuses over a WS-Notifications.
Purpose	The service provider may support notifications of activity statuses over a WS-Notifications.
Description	
Specification tested	Web Services Base Notification 1.3, OASIS Standard
Inputs	

Expected Results	The following code is present:
	`<wsnt:Notify>`
	`<wsnt:NotificationMessage>`
	`<wsnt:SubscriptionReference>`
	`wsa:EndpointReferenceType`
	`</wsnt:SubscriptionReference> ?`
	`<wsnt:Topic Dialect="xsd:anyURI">`
	`{any} ?`
	`</wsnt:Topic>?`
	`<wsnt:ProducerReference>`
	`wsa:EndpointReferenceType`
	`</wsnt:ProducerReference> ?`
	`<wsnt:Message>`
	`{any}`
	`</wsnt:Message>`
	`</wsnt:NotificationMessage> +`
	`{any} *`
	`</wsnt:Notify>`
	
	Example SOAP Encoding of the Notify Message:
	`<s:Envelope ... >`
	`<s:Header>`
	`<wsa:Action>`
	`http://docs.oasis-open.org/wsn/bw-2/NotificationConsumer/Notify`
	`</wsa:Action>`
	`...`
	`</s:Header>`
	`<s:Body>`
	`<wsnt:Notify>`
	`<wsnt:NotificationMessage>`
	`<wsnt:SubscriptionReference>`
	`<wsa:Address>`
	`http://www.example.org/SubscriptionManager`
	`</wsa:Address>`
	`</wsnt:SubscriptionReference>`
	`<wsnt:Topic Dialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">`
	`npex:SomeTopic`
	`</wsnt:Topic>`

	<wsnt:ProducerReference> <wsa:Address> http://www.example.org/NotificationProducer </wsa:Address> </wsnt:ProducerReference> <wsnt:Message> <npex:NotifyContent>exampleNotifyContent</npex:NotifyContent> </wsnt:Message> <wsnt:NotificationMessage> </wsnt:Notify> </s:Body> </s:Envelope>
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	WS-Notifications / Basic Execution Service Requirements
See Also / References	http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf

Identifier	TC-7
Title	Support/provision of an advance reservation management web service interface.
Purpose	
Description	The service provider may support advance reservation management web service interface.
Specification tested	There is currently no normative extension of the BES interface for advance reservation management, however any technology provider wishing to provide such interface may derive from existing works: Mamoński M.: Smoa Computing HPC Basic Profile Adoption – Experience Report.
Inputs	
Expected Results	An example of creating an advance reservation in Advance Reservation Description Language. Similar to the Advance REServation Factory the Advance Reservation Description Language (ARDL) was modelled upon another OGF standard: the Job Submission Description Language (JSDL) specification. An example ARDL

	document is presented in the below listing: <ardl:ReservationDefinition> <ardl:ReservationDescription> <ardl:ReservationIdentification> <ardl:ReservationName>SampleReservation</ardl:Reservation nName> </ardl:ReservationIdentification> <ardl:TimeWindow> <ardl:StartTime>2010-03- 21T11:00:00+01:00</ardl:StartTime> <ardl:EndTime>2010-03-21T15:00:00+01:00</ardl:EndTime> </ardl:TimeWindow> <ardl:Resources> <ardl:ReservedSlotsCount>1</ardl:ReservedSlotsCount> <ardl:UserName>jsmith</ardl:UserName> </ardl:Resources> </ardl:ReservationDescription> </ardl:ReservationDefinition> This document describes request for creating an advance reservation: • bearing human readable name SampleReservation, • starting on 11.00 (CET) 21st March 2010, • ending on 15.00 (CET) 21st March 2010, • for one slot (which usually corresponds to one cpu core), • with Access Control List set to local user jsmith.
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Advance reservation management web service interface / Basic Execution Service Requirements
See Also / References	http://www.ogf.org/documents/GFD.179.pdf

Identifier	TC-8
Title	Acceptance of the activities described using the JSDL Version 1.0 standard.
Purpose	The *CreateActivity* operation of the *BES-Factory* port must accept activities described using the Job Submission Description Language Version (JSDL) 1.0 standard.
Description	We distinguish normative JSDL elements from non-normative vendor extensions. List of normative JSDL document elements that must be accepted by the target system are listed below:

	• *ApplicationName* – an abstract name of the application (e.g. MUSCLE) that is mapped to the physical absolute path by the underlying system, • *JobIdentification/JobName* – an opaque name of the job, • *HPCProfileApplication/Argument* – a vector of arguments passed directly to application, • *HPCProfileApplication/Input* – a name of the application standard input file, • *HPCProfileApplication/Output* – a name of the application standard output file, • *HPCProfileApplication/Error* – a name of the application standard error file • *HPCProfileApplication/Environment* – a vector of the key value pairs that denote environment variables that must be set before starting the application • *HPCProfileApplication/WorkingDirectory* – an absolute path of the working directory where the application should be started, • *Resources/TotalCPUCount* – a number of job slots (cores) allocated for the job, • *Resources/IndividualCPUCount* – a number of job slots (cores) allocated for the job on single node.
Specification tested	Job Submission Description Language Version (JSDL) 1.0 standard.
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Normative JSDL elements / Job Description Requirements
See Also / References	http://www.gridforum.org/documents/GFD.56.pdf

Identifier	TC-9
Title	Capability of submitting job into an advance reservation.
Purpose	The target system must be capable of submitting job into an advance

	reservation.
Description	We request one additional non-normative extension of the JSDL document: the target system must be capable of submitting job into an advance reservation. Therefore, the system must accept, probably as a vendor extension of the Resource element, a local reservation identifier in the job description document.
Specification tested	Job Submission Description Language Version (JSDL) 1.0 standard.
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Non-normative JSDL elements / Job Description Requirements
See Also / References	http://www.gridforum.org/documents/GFD.56.pdf

Identifier	TC-10
Title	Availability of the gridFTP service at the target site.
Purpose	The gridFTP service must be available at the target site.
Description	Although we assume that the application is preinstalled there is still a need for transferring input and output data of the application. For this purpose the profile requires that the gridFTP service is available at the target site.
Specification tested	GridFTP v2 Protocol Description, Report GFD-R-P.047
Inputs	
Expected Results	
Actual Results	
Pass/Fail	

Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	gridFTP / Data Staging Requirements
See Also / References	http://www.ogf.org/documents/GFD.47.pdf

Identifier	TC-11
Title	Access to the file system granted to the gridFTP service.
Purpose	
Description	The gridFTP service must have access to the file system that is available from cluster worker nodes, and thus accessible directly within a job.
Specification tested	GridFTP v2 Protocol Description, Report GFD-R-P.047
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	gridFTP / Data Staging Requirements
See Also / References	http://www.ogf.org/documents/GFD.47.pdf

Identifier	TC-12
Title	X.509 proxy certificates accepted by the target service.
Purpose	The target service must accept X.509 proxy certificates.

Description	As both the Tightly and Loosely coupled scenarios involve additional services (e.g., the QCG-Broker or AHE) between the user and the infrastructure it implies that the target site must accept delegated credentials. Therefore in the MAPPER profile we request that the target service must accept X.509 proxy certificates
Specification tested	Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile, RFC 3820
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Certificates / Security Requirements
See Also / References	http://www.ietf.org/rfc/rfc3820.txt

Identifier	TC-13
Title	Certificates issued by the CAs accepted by the target site.
Purpose	To prevent users from the necessity of presenting different credentials while accessing different resources, the target site should accept certificates issued by any of the Certificate Authorities that are member of EUGridPMA.
Description	
Specification tested	EUGridPMA
Inputs	
Expected Results	
Actual Results	

Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Certificates / Security Requirements
See Also / References	www.eugridpma.org/

Identifier	TC-14
Title	The existence of the one incoming open port on the frontend/interactive node.
Purpose	
Description	This non-normative requirement is related only to the Tighly Coupled Application Scenario as it involves parallel computations spanning across multiple sites. The cross-cluster communication is facilitated with the help of the additional user-space daemon deployed on the frontend machine under the assumption that the sites follow the following firewall policy: • there exists one incoming open port on the frontend/interactive node (the port must be accessible from the other clusters involved in cross-cluster simulation).
Specification tested	
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Requirements specific to the Tightly Coupled Application Scenario

See Also / References	

Identifier	TC-15
Title	Not restricted connectivity from worker nodes to the frontend/interactive node.
Purpose	
Description	This non-normative requirement is related only to the Tighly Coupled Application Scenario as it involves parallel computations spanning across multiple sites. The cross-cluster communication is facilitated with the help of the additional user-space daemon deployed on the frontend machine under the assumption that the sites follow the following firewall policy: • the connectivity from worker nodes to the frontend/interactive node is not restricted.
Specification tested	
Inputs	
Expected Results	
Actual Results	
Pass/Fail	
Remarks	
Author	ACK CYFRONET AGH
Rights	PUBLIC
Category / Group	Requirements specific to the Tightly Coupled Application Scenario
See Also / References	