**EGI-InSPIRE**

Interim report on the mini projects

**EU MILESTONE: MS801**

|  |  |
| --- | --- |
| Document identifier: | EGI-InSPIRE-D8.1-2147-v1.docx |
| Date: | 24/03/2014 |
| Activity: | **SA4** |
| Lead Partner: | **EGI.eu** |
| Document Status: | **DRAFT** |
| Dissemination Level: | **PUBLIC** |
| Document Link: | <https://documents.egi.eu/document/2147> |

|  |
| --- |
| Abstract  The EGI-InSPIRE SA4 Work Package has been set up as part of an amendment to the project’s DoW for PY4. This document provides the final report on the status of all individual, funded mini-projects. |

1. Copyright notice

Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration. EGI-InSPIRE (“European Grid Initiative: Integrated Sustainable Pan-European Infrastructure for Researchers in Europe”) is a project co-funded by the European Commission as an Integrated Infrastructure Initiative within the 7th Framework Programme. EGI-InSPIRE began in May 2010 and will run for 4 years. This work is licensed under the Creative Commons Attribution-Noncommercial 3.0 License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc/3.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, and USA. The work must be attributed by attaching the following reference to the copied elements: “Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration”. Using this document in a way and/or for purposes not foreseen in the license, requires the prior written permission of the copyright holders. The information contained in this document represents the views of the copyright holders as of the date such views are published.

1. Delivery Slip

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name** | **Partner/Activity** | **Date** |
| **From** | Michel Drescher | EGI.eu/SA4 |  |
| **Reviewed by** | **Moderator:**  **Reviewers:**  <<To be completed by project office on submission to AMB/PMB>> |  |  |
| **Approved by** | **AMB & PMB**  <<To be completed by project office on submission to EC>> |  |  |

1. Document Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Issue** | **Date** | **Comment** | **Author/Partner** |
| 1 | 17 Mar 2014 | ToC & Skeleton | Michel Drescher/EGI.eu |
| 2 | 24 Mar 2014 | Added TSA4.4, TSA4.8, TSA4.9 | Michel Drescher/EGI.eu |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |

1. Application area

This document is a formal deliverable for the European Commission, applicable to all members of the EGI-InSPIRE project, beneficiaries and Joint Research Unit members, as well as its collaborating projects.

1. Document amendment procedure

Amendments, comments and suggestions should be sent to the authors. The procedures documented in the EGI-InSPIRE “Document Management Procedure” will be followed:  
<https://wiki.egi.eu/wiki/Procedures>

1. Terminology

A complete project glossary is provided at the following page: <http://www.egi.eu/about/glossary/>.

1. PROJECT SUMMARY

To support science and innovation, a lasting operational model for e-Science is needed − both for coordinating the infrastructure and for delivering integrated services that cross national borders.

The EGI-InSPIRE project will support the transition from a project-based system to a sustainable pan-European e-Infrastructure, by supporting ‘grids’ of high-performance computing (HPC) and high-throughput computing (HTC) resources. EGI-InSPIRE will also be ideally placed to integrate new Distributed Computing Infrastructures (DCIs) such as clouds, supercomputing networks and desktop grids, to benefit user communities within the European Research Area.

EGI-InSPIRE will collect user requirements and provide support for the current and potential new user communities, for example within the ESFRI projects. Additional support will also be given to the current heavy users of the infrastructure, such as high energy physics, computational chemistry and life sciences, as they move their critical services and tools from a centralised support model to one driven by their own individual communities.

The objectives of the project are:

1. The continued operation and expansion of today’s production infrastructure by transitioning to a governance model and operational infrastructure that can be increasingly sustained outside of specific project funding.
2. The continued support of researchers within Europe and their international collaborators that are using the current production infrastructure.
3. The support for current heavy users of the infrastructure in earth science, astronomy and astrophysics, fusion, computational chemistry and materials science technology, life sciences and high energy physics as they move to sustainable support models for their own communities.
4. Interfaces that expand access to new user communities including new potential heavy users of the infrastructure from the ESFRI projects.
5. Mechanisms to integrate existing infrastructure providers in Europe and around the world into the production infrastructure, so as to provide transparent access to all authorised users.
6. Establish processes and procedures to allow the integration of new DCI technologies (e.g. clouds, volunteer desktop grids) and heterogeneous resources (e.g. HTC and HPC) into a seamless production infrastructure as they mature and demonstrate value to the EGI community.

The EGI community is a federation of independent national and community resource providers, whose resources support specific research communities and international collaborators both within Europe and worldwide. EGI.eu, coordinator of EGI-InSPIRE, brings together partner institutions established within the community to provide a set of essential human and technical services that enable secure integrated access to distributed resources on behalf of the community.

The production infrastructure supports Virtual Research Communities (VRCs) − structured international user communities − that are grouped into specific research domains. VRCs are formally represented within EGI at both a technical and strategic level.

1. EXECUTIVE SUMMARY

TABLE OF CONTENTS

1 Introduction 7

2 Mini projects status reports 8

2.1 Work Package management 8

2.2 TSA4.2: Massive open online course development 8

2.2.1 Results achieved 8

2.2.2 Mini project closure report 8

2.3 TSA4.3: Evaluation of Liferay modules 9

2.3.1 Results achieved 9

2.3.2 Mini project closure report 9

2.4 TSA4.4: Providing OCCI support for arbitrary CMF 10

2.4.1 Results achieved 10

2.4.2 Mini project closure report 10

2.5 TSA4.5: CDMI support in cloud management frameworks 11

2.5.1 Results achieved 11

2.5.2 Mini project closure report 11

2.6 TSA4.6: Dynamic deployments for OCCI compliant clouds 12

2.6.1 Results achieved 12

2.6.2 Mini project closure report 12

2.7 TSA4.7: Automatic deployment and execution of applications using cloud Services 13

2.7.1 Results achieved 13

2.7.2 Mini project closure report 13

2.8 TSA4.8: Transforming scientific research platforms to exploit cloud capabilities 14

2.8.1 Results achieved 14

2.8.2 Mini project closure report 15

2.9 TSA4.9: VO Administration and operations PORtal (VAPOR) 16

2.9.1 Results achieved 17

2.9.2 Mini project closure report 17

2.10 TSA4.10: A new approach to computing availability and reliability reports 17

2.10.1 Results achieved 18

2.10.2 Mini project closure report 18

2.11 TSA4.11: GOCDB scoping extensions and management interface 18

2.11.1 Results achieved 19

2.11.2 Mini project closure report 19

2.12 TSA4.12: Tools for automating applying for and allocating federated resources 20

2.12.1 Results achieved 20

2.12.2 Mini project closure report 20

3 Conclusion 21

4 References 22

# Introduction

# Mini projects status reports

## Work Package management

## TSA4.2: Massive open online course development

This task develops a MOOC (Massive Open Online Course), in which participants will learn to use Grid computing and storage services as well as other EGI services for their own projects. It focuses on users without any previous large scale computing experience and shows them different methods to use large scale computing facilities. This includes working on a local cluster, and using the Grid through the gLite middleware, pilot jobs and workflow management systems.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.3: Evaluation of Liferay modules

The objective of the mini-project is to evaluate the Liferay portal[[1]](#footnote-1) with its recently released modules Liferay Sync and Liferay Social Office as a replacement for some of the EGI back office services provided now by CESNET using a set of specialised software systems, and as a web portal platform for the EGI community. The outcome is expected to be best practices and recommendations for the EGI community.

The mini-project is divided among three partners: CESNET, currently operating EGI’s back-office, evaluating the service replacement and general portal options, and INFN and SZTAKI, both evaluating compatibility with their community portlets.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.4: Providing OCCI support for arbitrary CMF

This EGI-InSPIRE mini-project aims at providing a cloud interoperability framework based on OCCI with support for arbitrary cloud management frameworks. One of its key enabling scenarios is to be able to run a predefined virtual machine image at multiple sites of a federated cloud environment and, consequently, to manage the resulting virtual machine. As different cloud management frameworks currently exist and are actively used at different sites, enforcing a particular framework across all sites is neither practical nor desired for a plethora of organizational and technical reasons. Therefore a standardized, uniform interface for the management of virtual machines is needed. This mini-project maintains and further develops the rOCCI framework and rOCCI-server that are used in the EGI Federated Cloud infrastructure, with a particular focus on interoperability with other OCCI implementations present in the EGI Federated Clouds infrastructure testbed..

The mini-project efforts are divided into three main categories:

1. Organization
2. Design and implementation
3. Testing and documented deployment

### Results achieved

This section provides a summary of the achieved results. Details are available in the mini project wiki[[2]](#footnote-2), including task descriptions and rOCCI-server design documentation[[3]](#footnote-3). In the context of this report CMF refers to Cloud Management Framework, otherwise also known as Cloud Middleware or Cloud Management Platform. The results can be divided into groups according to its tasks as follows:

***Task 1: Mini-project Management***

Members of the team proposed and agreed on a work schedule, meeting schedule, reporting schedule and development tools, presented the mini-project at the EGI Community Forum 2013, EGI Technical Forum 2013 and are preparing a final presentation for the upcoming EGI Community Forum 2014. Reporting was performed on a weekly and quarterly basis, additional reports were provided for MS801 Interim Reports.

***Task 2: rOCCI framework changes***

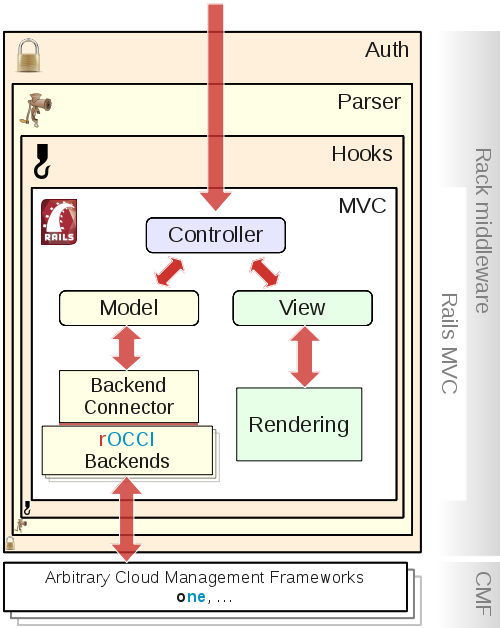
Members of the team identified changes that were required to harmonise authentication and authorisation mechanisms across all available OCCI implementations, implemented said changes and deployed updated version of all rOCCI cmponents within the EGI Federated Cloud Task environment. The previously monolithic rOCCI framework has been split into three easily maintainable components named rOCCI-core, rOCCI-api and rOCCI-cli in preparation for the re-design of rOCCI-server. All components are available as open source from the EGI-FCTF organization on GitHub. Aside from cosmetic changes, the rOCCI framework went through a series of major rewrites and extensions, adding features explicitly requested by members of the EGI Federated Cloud, most notably:

* Contextualization support
* Improved human-readable output rendering
* Support for linking networks and storages to running compute instances
* Support for creating storage instances (storage block devices)
* Support for CMF Synnefo

***Task 3: rOCCI-server re-design***

Members of the team proposed and agreed on a design of the new rOCCI-server and implemented said design in Ruby. rOCCI-server is available as open source from a repository on GitHub maintained under the EGI-FCTF[[4]](#footnote-4) organization. The implementation follows these basic design concepts (see also Figure 1):

* Modular authentication
* Modular back-end architecture
* Extensible core architecture

Figure 1: Architecture of the rOCCI server

The implementation also includes the following production-grade facilities and features:

* Advanced logging possibilities
* Improved performance
* Dummy back-end for testing purposes
* Configuration integrated with Apache2 Virtual Host configuration

***Task 4: Back-ends for CMFs***

Building on top of design principles stated in Task 3, members of the team implemented a fully-featured rOCCI-server backend for the open source CMF OpenNebula. This back-end is considered to be production-grade and ready for deployment in the EGI Federated Cloud environment. See also section 2.4.2.6 for information about additional back-ends originally proposed in the beginning of this mini-project.

***Task 5: Testing and Deployment***

The newly implemented rOCCI-server has been extensively tested internally at CESNET during its development and publicly, within the scope of the EGI Federated Cloud, in co-operation with CESGA during its beta stages.

Production deployment within the EGI Federated Cloud is scheduled for April 2014, packages for all supported platforms[[5]](#footnote-5),[[6]](#footnote-6) and installation instructions[[7]](#footnote-7), significantly simplifying deployment, are among the outputs of this mini-project.

***Task 6: Documentation***

Documentation is provided in the form of wiki pages publicly available on GitHub[[8]](#footnote-8) and code documentation using the RDoc format. It covers rOCCI-server architecture, deployment scenarios, installation, configuration, smoke testing and upgrade procedures. It will be extended in the future, based on user feedback.

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

The following objectives have been outlined by this mini-project; all were successfully achieved:

* Improve and maintain the rOCCI framework with all its components
* Design a new rOCCI-server with modularity and extensibility in mind
* Implement the newly designed rOCCI-server
* Provide documentation
* Deploy the newly implemented rOCCI-server within the EGI Federated Cloud

#### Benefits

The expected benefits of this mini-project represent improvements in three categories.

1. **Interoperability:** Changes implemented inside the rOCCI framework demonstrably improved compatibility with all major CMFs used within the EGI Federated Cloud.
2. **Support for new CMFs**: The modular architecture implemented inside rOCCI-server provides necessary building blocks for back-end developers, back-end for CloudStack is already in development.
3. **New features:** By extending the existing rOCCI framework components, we were able to offer new features requested by user communities and resource providers alike. These are all expected benefits.

An unexpected benefit of this mini-project is a SAM Nagios probe based on the rOCCI framework. It is currently used to monitor all OCCI endpoints in the EGI Federated Cloud.

#### Scope

The mini-project stayed within its original scope and tolerances. It did not exceed the expected timescale or budget. For information about minor changes within the project tolerances, see section 2.4.2.6.

#### Lessons Learned

Overall, the mini-project went well and we managed to provide expected outputs in a timely manner. The lessons learned during this mini-project were mostly related to task scheduling and the division of work among team members. It was a valuable experience for all members of the mini-project team.

#### Was the Project managed appropriately?

The mini-project and the whole work package were managed appropriately. The shepherd was helpful in every area, provided advice and leadership.

#### Risks

Over the duration of the mini-project we encountered only one unexpected and potentially risky situation, as follows:

The original mini-project proposal included unfunded participation of two current and one former member from GWDG (Gesellschaft fuer wissenschaftliche Datenverarbeitung mbH Goettingen). Unfortunately, both GWDG members did not participate at all and the former GWDG member indicated his unavailability for this mini-project in M6-M12. This had an impact on the original work schedule agreed upon in the beginning of the mini-project. To accommodate this change, we proposed and implemented the following changes to the original project work plan, in co-operation with our shepherd:

1. Focus more on the rOCCI framework and its end-user component rOCCI-cli instead of implementing multiple back-ends for the rOCCI-server in the scope of this mini-project;
2. Simplify rOCCI-server architecture by limiting the extent of the back-end abstraction;
3. Focus on implementing a rOCCI-server back-end for OpenNebula while assisting with any third-party back-end development;
4. Drop the objective to implement a native proof-of-concept OCCI client for Java.

Despite these unexpected organizational changes, the mini-project completed its term without major delays or problems and completed the initially stated goals with minor exceptions mentioned above.

## TSA4.5: CDMI support in cloud management frameworks

This task’s objective is to design and implement a SNIA/ISO CDMI-compliant storage service that integrates with the EGI core infrastructure, and extends the EGI service portfolio by offering a standards based object storage component.

The development aims at offering richer server-side processing functionality to simplify client creation. The initial preparation of this task consisted in setting up a development infrastructure for the project (Github projects[[9]](#footnote-9), RTD documentation[[10]](#footnote-10), CI, and Jira).

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.6: Dynamic deployments for OCCI compliant clouds

This task’s objective is to deliver to OCCI compliant clouds the possibility for users to dynamically provision complex multi-VM applications, with elements of elastic behaviour as well as an automatic image factory. For this, we take advantage of the open source SlipStream[[11]](#footnote-11) solution.

The project is split into the following subtasks:

* **Creation of the SlipStream OCCI connector**: This will allow SlipStream users to provision cloud resources on the EGI federated cloud service, using OCCI as the API.
* **Automatic and repeatable deployment**: this will prove that users can construct machine images and perform deployments automatically over the OCCI connector.

**Auto-scale foundations capabilities**: This will allow users to provision dynamic workloads on OCCI-compliant clouds with elements of auto-scale (i.e. elastic behaviour), based on user defined KPIs and trigger logic.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.7: Automatic deployment and execution of applications using cloud Services

This task’s objective is to design and implement a contextualization capability, which supports scientific communities in executing their computing workload through automating the deployment of scientific software on virtual machines, using the interfaces and standards used in EGI’s Cloud Infrastructure Platform. This new capability will allow VRC managers (or advanced users) to define a set of applications that the researchers can easily deploy in virtual machines relieving them from the overhead of setting up the computing environment.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.8: Transforming scientific research platforms to exploit cloud capabilities

The goal of this activity is the derivation of patterns and recipes that can be applied to make applications cloud ready. This is done by optimising several use cases that we see most promising to benefit from these actions. The lessons learnt will result in a collection of best practices of which new applications can make use to ease their uptake of cloud technologies. We do this by evaluating existing VM images provided by various user communities and trying to optimize how they make use of cloud resources. Our decisions are supported by questionnaires about the applications targeted at the individual use cases.

### Results achieved

For the WeNMR community, two use cases were considered. First of all, we crafted an image containing the Gromacs software along with PyMol for molecular visualization. This image is intended to be used for tutorial purposes, such that students can easily start off with an already running Gromacs environment. The second use case involved the VCING application for validating and improving biomolecular NMR structures. The original image provided by the community had been generated several years ago and was missing important updates. Secondly, the image was clearly a development environment including continuous integration server and related data payload of generated software artefacts. The excess data payload of the image accumulated to several gigabytes that would never be used when running instances of the image in the cloud. This use case was also lacking contextualization, such that the image was only really useful for a single set of computations. Detailed information has been documented in the project’s report (REF EGI DOC 1824-v3). We have never received any official or final feedback regarding our recommendations or implementation for any of the two use cases.

Engagement with the BNCweb use case of CLARIN started rather late in the project. It was unclear for several months what the intended use of the provided image was and whether or not a substantial data payload was really needed. After a meeting with community representatives, it was decided to remove the data payload and factor components of the appliance. Consequently, the database server was factored out and separated from the web frontend. Only later we discovered that the scaling characteristics of the application are not quite as simple as claimed by the user community. Additionally, there are technical requirements in BNCweb to host the MySQL server and web frontend on the same machine.

The PeachNote community was the most agile one to engage with. We were unable to improve anything in the OMR VM images they had provided to us, as they were based on MS Windows and already minimal. However, a very specific requirement from the community was to enable the object storage to provide scaled versions of their images. There were some alternative solutions to this problem, including scaling all music sheet images and offering these files. However, we developed a solution to do the scaling on the fly and integrated it in the object storage’s front end. This can become a general solution for user communities having specific demands on the object storage.

For the BioVeL community and their OpenModeller use case, we proposed to shrink the image by removing excess data payload, cleaning the operating system and using cloud-init for contextualization of instances. The recommendations are in the final phase of implementation and need to be verified by the user community.

### Mini project closure report

The activity started out with a set of use case specific questionnaires that were sent to the user communities. One of the goals was to find out which storage access patterns were prevalent for each of the applications, enabling us to recommend certain setups of the infrastructure and use either block of object storage for the applications. This would eventually lead to decreasing the huge data payload that was sent along with some of the application images.

In addition to clarifying the use of appropriate storage resources, applications within virtual machines need to be contextualized for the actual execution. For instance, given a generic image supporting an application, one would want to set certain parameters about how it is started. The most trivial example is the SSH access to the VM, for which public keys need to be installed such that users can log in.

We engaged with a number of user communities to see how we could help them make better use of the EGI federated cloud resources.

Further communities in addition to the above mentioned ones had initially been considered to engage with. However, this did not happen for several reasons. The WSPGRADE and GAIASpace use cases were poorly defined. The VMDIRAC tool is a framework and as such not a primary target of our activities. When interacting with the DCH-RP community, the goals in terms of technical implementation were too vague to make any recommendations at the time. Lastly, with only limited resources in the project, we focused on the most promising use cases.

#### Objectives Achieved

After thorough investigation of user communities’ images, we were able to determine which components within the images can be removed. This lead to generic patterns for optimization which have been documented in our Wiki as well as dedicated blog posts.

Another topic of optimization was the use of appropriate storage resources. Whereas many communities stored data within their images, we recommend using explicit storage resources, either attachable block storage or object storage, depending on the use case and data access pattern.

We have used the EGI Blog to publish information dedicated to image setup. In one posting we described how to keep application images minimal in terms of installed software and data. Not only is this important for performance, but depending on what image creators have installed before, security issues play a role here. Unneeded services installed in an image may provide serious attack vectors and should be avoided. In another blog post, we described how to shrink images without altering their contents, a measure that can always be taken and potentially avoid gigabytes of data being transferred per image or instance.

These results have also been incorporated in the generic images that we provide. The generic images contain the minimal operating system and are contextualized through cloud-init. These can be used as a basis for application specific images. They are available for Debian 7 and Ubuntu.

One of the generic capabilities that the BNCweb use case lead us to is a generic database server that can be instantiated and linked with arbitrary data provided through an attachable block storage device. However, we did not make use of all of the initially identified cloud capabilities. Partially because they were not appropriate for the use cases, and partially because introducing such capabilities usually results in changing the application, which would have required stronger support from communities.

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

Beyond the optimization of individual VM images for selected use cases, we originally also targeted the use of higher-level cloud services and allowing applications to make better use of the distributed infrastructure they run in. These higher-level services include messaging, auto-scaling, cloud orchestration. Partially, these have been developed in other mini projects. Also, some new standards and implementations appeared during the course of the project.

#### Lessons Learned

We learned several lessons about the interaction with user communities. It is very important to ascertain the commitment of the user communities when engaging with them. User’s engagements need to be revalidated during the course of interaction. In general, one can say that while user communities are quick to raise a hand when it is about offering services and resources to them, they’re less willing to provide the required information to better understand their use cases or to verify if the project developments meet their needs. Without a community’s commitment to the collaboration, one can still learn a lot, however the full potential of the interaction will only develop with mutual commitment to the goals.

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

As described above, the commitment of user communities was utterly important for this project. Whereas it is possible to work without them and only provide generic instructions and information, we could not have been certain to provide the required information for communities. As these communities had indicated real interest in making use of cloud resources, their heterogeneous commitment had not been foreseen.

## TSA4.9: VO Administration and operations PORtal (VAPOR)

VAPOR intends to help small and medium-sized grid communities perform daily administrative and operational tasks, by developing a generic tool to assist community managers and support teams in performing their daily activities. Such communities may typically have no or few dedicated IT support, have scattered scientific activities or fragmented user groups, and may possibly make an opportunistic usage of the resources.

The portal is expected to

* Facilitate administration and operations for VO with few IT support,
* Help communities sustain their model by making it possible to mutualise the administrative and operational cost with other communities,

Facilitate the outreach of new user communities by making it easier to start with the administration and operations of a VO.

### Results achieved

#### Functional specifications

In the first phase of the project (M1 to M3) the functional specifications and priorities of the project were defined with partners, along with the inventory of existing material that may be leveraged on. This phase resulted in *Deliverable D1 - VAPOR Functional Specifications* [R 8].

#### Developments

VAPOR consists of three major sets of features (details follow): (1) Resources status indicators and operational reports, (2) VO data management, and (3) Community users management. A fourth feature regarding VO accounting described initially was deemed of low interest by partners. Only the first two sets of features were completed and deployed during the project duration. Reasons for not developing the third item (deliverable D3) are given in section 2.9.2.1.

(1) The *Resource status indicators and operational reports* comprise several functions:

* A consolidated list of the resources provide VO administrators with an outlook of the resources that support the VO, joining information from the GOCDB and BDII. Another view presents resources with non production status or well known issues (erroneous publication of information in the BDII).
* The JobMonitor tool monitors any computing elements supporting the VO and reports graphical and tabular views as to the rate of successful, faulty or timed-out jobs. An average execution time for successful jobs is calculated.
* Results of the JobMonitor are used to compute a “white list” of computing elements, those CEs known to have been performing well during the last tests. This list can be used to feed a job submission system.
* Finally a view reports the evolution of the VO running and waiting jobs, that can help monitor the overall activity, and help investigate bad performances issues, in particular in the context of opportunistic usage of resources.

(2) The *VO Data Management* comprises two functions that help VO administrators track full storage elements or prevent them from filling up, track and clean up inconsistencies between the file catalog and storage elements, deal with the decommissioning of storage elements.

* The *catalog-based SE scan* periodically checks the filling rate of the storage elements (SE) supporting the VO. Those over a per-VO configurable threshold are scanned against the VO logical file catalog (LFC). Reports provide a list of heavy users according to the LFC along with their current VO membership status, and help the VO administrators to contact those users to ask them to clean up or migrate their files.
* The *cleanup of dark data and lost files* periodically checks the consistency between the VO file catalog and the files actually stored on the storage elements. Reports provide the list of inconsistencies that may be addressed manually. Optionally, the dark data files (older than a configurable age) can be cleaned up automatically.

The features of the *Resource status indicators and operational reports* (1) were delivered at M7 (deliverable D3.1). The *catalog-based SE scan* was delivered at M9, while the *Cleanup of dark data and lost* *files* was delivered at M12 (deliverable D3.2).

Along the project, a continuous bug fixing and improvement process helped update the functions, take into account feed-back from beta users, and integrate new functionalities periodically.

#### Deployment

An early release of VAPOR, including the web application and the data collecting services of the *Resource status indicators and operational reports* was deployed at M8.

Then, a beta release was open for biomed at M9, for test by the biomed support team. VAPOR's data collecting services were deployed on a virtualized server hosted at the I3S laboratory. The web portal was integrated with the EGI Operations Portal, and deployed on the same web server. Two reasons for this choice: (i) avoid to provide users with yet another portal, and (ii) benefit from the production web servers maintained at the CCIN2P3 Computing Centre (CC) in Lyon.

The shiwa-workflow.eu partner VO was enabled in VAPOR at M10. Then, VOs compchem, enmr.eu, vlemed and vo.francegrille.fr were enabled at M11 and M12.

#### Documentation

A detailed *Installation and Configuration Guide[[12]](#footnote-12)* has been written and updated all along the evolutions of the project. This document lists software dependencies, configuration files, the complete installation and deployment procedure, as well as the procedure to enable an additional VO.

A document *Apache2 Server securization guide lines* provides guidelines as to the safe configuration of an Apache server. It has been applied to the web server hosted at I3S which serves the data collected to the web application hosted on production web servers at the IN2P3 CC.

### Mini project closure report

#### Objectives Achieved

The main goal of VAPOR is the development and deployment of an administration and operations portal dedicated to help small and medium-size grid communities perform daily administrative and operational tasks.

The scope and functional specification of the features to be developed were discussed and refined at the beginning of the project, with partner VOs. Three sets of features were defined: (1) Resources status indicators and operational reports, (2) VO data management, and (3) Community users management.

(1) and (2) were successfully developed. They are now deployed and supporting 6 VOs: biomed, CompChem, Enmr.eu, shiwa-workflow.eu, Vlemed.

(3) could not be addressed at all during the project. This results from the fact that the project shifted by 4 months out of the total 12 months duration, as compared to the expected schedule. Below we identify the reasons of this shift.

The main developer of the project (recruited for 12 months) showed a serious lack of expertise in software engineering and development best practices, despite a 4 year experience. This problem was handled by setting up a joined work with the project manager, weekly if not daily meetings, proofreading of code produced, etc. Yet, development tasks were significantly delayed.

The development of feature (1) was longer than expected due to two reasons:

* The JobMonitor feature of VAPOR relies on the JSAGA API[[13]](#footnote-13). Initial exploitation results showed unexpectedly high job failure rates. The investigation took quite a long time, involving the VAPOR team, the JSAGA team, and site administrators. Ultimately this lead to figuring out many different problems: firewall configuration issues, computing element misconfigurations ( “true” failures), but also bugs in the JSAGA API itself or its dependencies (Globus).
* The data integration web service Lavoiser[[14]](#footnote-14) is a great tool, although its learning curve proved to be quite long, and the finalisation of the data integration views was tedious (but the developers were very supportive and helpful).

Within (2), the development of the *Cleanup of dark data and lost files* feature was delayed:

The feature is based on the development of a tool able to list all the files that belong to a certain VO on a given storage element, in an implementation-independent manner. However, it proved to be incredibly difficult to answer simple questions: how to recursively list the files of an SE for a given VO? How to find the protocol supported by an SE to list the files? How to build the access url? Etc. We discussed with site administrators from several NGIs, with GFAL2 API[[15]](#footnote-15) developers, we submitted GGUS tickets (101086, 101187, 97076), we tried many solutions. Ultimately, we got in touch with people from the BDII at CERN who helped us figure out the proper information to query from the Glue model.

#### Benefits

At the time of writing, VAPOR has been up and running for 4 months for the VO biomed and approximately one month for other VOs. The biomed VO support team now uses it on a daily basis: e.g. it helps check the status of resources, contact users with files on an SE that will be decommissioned, etc. Feed-back and questions from other VOs attest their interest in using it.

The JobMonitor feature proved to be an effective tool: it helped reveal several technical issues on computing elements, although these were not necessarily reported by standard VO Nagios probes.

During the last 2 years, the biomed VO has noticed irregular computing resource availability. It is suggested that this concern results from the opportunistic resource usage model. To test this hypothesis, biomed has submitted a request for resource allocation to EGI. We expect VAPOR to be helpful to assess the effect of such an allocation on the effectiveness of computing resources for the VO.

Besides, in the longer term, the portal is expected to provide benefits:

* Help communities sustain their model by making it possible to mutualise the administrative and operational cost with other communities

=> This will depend on the will and the need of VOs using VAPOR to mutualise the effort. Biomed has the experience of the difficulty of maintaining an active VO support team, and will push in the direction of such an effort mutualisation.

* Facilitate the outreach of new user communities by making it easier to start with the administration and operations of a VO.

=> The feedback of VOs currently using VAPOR will be significant here to help identify emerging communities that may fit in the model of VOs targeted by VAPOR, and encourage them to use it.

Lastly, discussions have been initiated as to the possible usage of VAPOR for different VOs or in different contexts (federating cloud, ER-flow, SCI-BUS).

#### Scope

The project scope remained very close to that defined initially with partners. The third set of features (Community users management) could not be developed within the time frame due to a time shift of 4 months. Reasons are detailed in section 2.9.2.1. The budget was respected.

#### Lessons Learned

The grid middleware consists of lots of components detailed in many documents here and there, but it is hardly possible to find up-to-date documents describing the global picture, or at least of get in touch with the people who have this global picture. The section “Risks” below illustrates this issue. Overall it attests how difficult it is to ensure the knowledge preservation within our community which is highly distributed, constantly evolving, consisting of many different sub-communities. But probably this is inherent to the academic world.

The recruitment of a trained software engineer proved to be difficult. We hardly received any response from the grid community, and a 12 month contract with few little further perspective is not attractive for an engineer from the industry. We feel like a recruitment media spanning the grid/cloud community would be helpful to reach a wider audience.

#### Was the Project managed appropriately?

The connection with the shepherd and the work package management was very easy and effective. The collaboration helped initiate discussions with regards to its applicability for different VOs or in different contexts (federating cloud, ER-flow, SCI-BUS), to the possible future extensions of VAPOR.

Also, tools provided (wiki, mailing list etc.) were used extensively.

#### Risks

One important issue that we had to deal with is the definition of the suitable technical solution to implement the *Cleanup of dark data and lost files* feature. This problem is detailed in section 2.9.2.1. We had anticipated the implementation by investigating several possible options through discussions with different NGIs and partner VOs. We were proposed solutions that were specific to some SRM implementation or to local deployment policies of some sites or NGIs. The problem we met is that only few people seem to have the full understanding of the various implementations and protocols involved in storage elements. We ended up with a suitable solution after discussing with the people who are very close to the information system definition (BDII and Glue schema).

Another risk we had not identified was the lack of expertise of the main developer. When this issue became clear, we deemed that it would be risky to try to recruit someone else, since the project completion date could not be postponed beyond April 2014, which was the end date of the EGI-InSPIRE project itself.

## TSA4.10: A new approach to computing availability and reliability reports

The goal of TSA4.10 is to implement a new availability and reliability reporting service that will replace ACE [R 14]. The new service will be implemented using open source components; it will be more flexible and extensible and it will allow the inclusion of more middleware services into the calculation of A/R metrics and by also adding VO-wise metric results (in addition to service-wise, site-wise and NGI-wise provisioning of results). Moreover, the profiles under which the calculations are done will be modular and a way to add or remove profiles will be made available and documented.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.11: GOCDB scoping extensions and management interface

The goal of this mini project was to

* Extend the current ‘EGI’ and ‘Local’ data scoping logic to introduce multiple, non-exclusive scope tags to encourage other projects to host their data within a single GOCDB instance, and
* Provide a supporting GOCDB management interface to simplify daily operational/admin tasks.

With these developments, the functionality of GOCDB would be extended beyond the current DoW so that topology data from multiple projects could be more effectively managed using a single GOCDB instance (e.g. EGI, EUDAT, PROJX). A management interface would help simplify and speedup daily operational tasks, especially for new service administrators and will help reduce on-going operational costs for EGI. Non-exclusive scope tags would allow sites/services to be scoped with both project-specific tags (e.g. ‘UK\_NES’) and with the wider ‘EGI’ scope tag.

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

## TSA4.12: Tools for automating applying for and allocating federated resources

This mini project directly supports one of EGI’s key strategic activities, by providing a tool that will allow automated provisioning of federated EGI resources. The tool is built collaborating closely with the Resource Allocation Task Force (RATF) [R 15]; the RATF in this relationship is the main coordination body, and this mini project serves as the technical implementation body. Details of the project plan are maintained with the RATF (see above).

### Results achieved

*Similar to MS801, provide here a summary of your project’s results over the entire project duration (i.e. including the last 6 months)*

*Expected amount: 1 A4 page*

### Mini project closure report

*This section – with its subsections – is intended to provide retrospection over the entire mini project, except the tangible results (which are already provided earlier). See instructions in the individual parts.*

*Expected amount: 2 A4 pages*

#### Objectives Achieved

*Outline the objectives of the project and indicate if they were achieved or not and if not, why not*

#### Benefits

*Outline any benefits that have been achieved and if these were expected or unexpected, and if there are any benefits which should be realised in the long run.*

#### Scope

*Did the project stay within its original scope? This is the project’s timescale, the budget and the projects tolerances.*

#### Lessons Learned

*Summarise the lessons learnt during the project, what went well and what you would do differently next time. Think about the project’s successes and areas which need to be improved and explain any recommendations you would have for future projects. You may then want to share these with the Planning Division.*

#### Was the Project managed appropriately?

*How would you describe the overall management of the mini project, your connection with the shepherd, and the overall work package management*

#### Risks

*Were there any risks which you felt you could have identified earlier on in the project. What do you feel prevented you from identifying them, or were they unavoidable and unpredictable?*

# Conclusion

# References

|  |  |
| --- | --- |
| R 1 | EGI Strategic Plan, Dx.y, DocDB |
| R 2 | EGI Platform Roadmap, MS514, <https://documents.egi.eu/document/1624> |
| R 3 | EGI-InSPIRE Quarterly Report 13, MS127, <https://documents.egi.eu/document/1928> |
| R 4 | Evaluation and work plan for Cloud Capability use cases, <https://documents.egi.eu/public/ShowDocument?docid=1824> |
| R 5 | Mini projects Wiki space, <https://wiki.egi.eu/wiki/Overview_of_Funded_Virtual_Team_projects> |
| R 6 | Mini projects meeting planner space, <https://indico.egi.eu/indico/categoryDisplay.py?categId=93> |
| R 7 | Mini projects document store space, https://documents.egi.eu/public/ListBy?topicid=51 |
| R 8 | D1 – VAPOR Functional Specification, https://wiki.egi.eu/wiki/VT\_VAPOR:VAPOR\_features\_description |
| R 9 | VAPOR feature priorisation, <https://wiki.egi.eu/wiki/VT_VAPOR:VAPOR_features_priorities> |
| R 10 | TSA4.10 SCRUM backlog, <https://code.grnet.gr/projects/ar-ng/issues?query_id=91> |
| R 11 | GRNET ~okeanos Cloud platform, <https://okeanos.grnet.gr/home/> |
| R 12 | Exit report for Task TSA4.11, <https://documents.egi.eu/document/1957> |
| R 13 | Agreemount framework, http://agreemount.com/, to be published soon |
| R 14 | Availability Computation Engine (ACE), <https://wiki.egi.eu/wiki/External_tools#Availability_Computation_Engine> |
| R 15 | EGI Resource Allocation Task Force (RATF), <https://wiki.egi.eu/wiki/Resource_Allocation_Task_Force> |
| R 16 | Liferay social office and synch evaluation plan and report, <https://documents.egi.eu/document/1737> |
| R 17 |  |
| R 18 |  |

1. <http://www.liferay.com/> [↑](#footnote-ref-1)
2. <https://wiki.egi.eu/wiki/VT_OCCI_for_CMF> [↑](#footnote-ref-2)
3. <https://wiki.egi.eu/wiki/ROCCICMFDocs> [↑](#footnote-ref-3)
4. <https://github.com/EGI-FCTF/rOCCI-server> [↑](#footnote-ref-4)
5. <https://appdb.egi.eu/store/software/rocci.cli> [↑](#footnote-ref-5)
6. <https://appdb.egi.eu/store/software/rocci.server> [↑](#footnote-ref-6)
7. <https://github.com/EGI-FCTF/rOCCI-server/wiki/rOCCI-Server-Admin-Guide> [↑](#footnote-ref-7)
8. <https://github.com/EGI-FCTF/rOCCI-server/wiki> [↑](#footnote-ref-8)
9. https://github.com/stoxy [↑](#footnote-ref-9)
10. https://stoxy.readthedocs.org/ [↑](#footnote-ref-10)
11. <http://sixsq.com/products/slipstream.html> [↑](#footnote-ref-11)
12. <https://redmine.i3s.unice.fr/svn/vapor/Docs/VAPOR%20Install%20and%20Configuration%20guide.pdf> [↑](#footnote-ref-12)
13. http://software.in2p3.fr/jsaga/dev/index.html [↑](#footnote-ref-13)
14. http://software.in2p3.fr/lavoisier/ [↑](#footnote-ref-14)
15. http://dmc.web.cern.ch/projects/gfal-2/home [↑](#footnote-ref-15)