



EGI-Engage

Fully Integrated MoBrain web portal

D6.4

Date	29 February 2016
Activity	WP6
Lead Partner	GNUBILA
Document Status	FINAL
Document Link	https://documents.egi.eu/document/2664

Abstract

The MoBrain Web Portal (<http://mobrain.egi.eu>) is the entry point to the MoBrain Competence Center, it is meant to aggregate information from the WeNMR and N4U communities and link to relevant content on the respective sites and other related projects such as INSTRUCT and the WestLife Virtual Research Environment.

The MoBrain portal has been deployed as a Liferay-based portal in the EGI Federated Cloud. Its deployment and configuration are managed using the Software Configuration Management tool Puppet and is called from a contextualization script launched at the end of the Virtual Machine creation. This portal provides the basis from which the integration/aggregation process will be continued under EGI-Engage with input from the various MoBrain partners.



This material by Parties of the EGI-Engage Consortium is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

The EGI-Engage project is co-funded by the European Union (EU) Horizon 2020 program under Grant number 654142 <http://go.egi.eu/eng>

COPYRIGHT NOTICE



This work by Parties of the EGI-Engage Consortium is licensed under a Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>). The EGI-Engage project is co-funded by the European Union Horizon 2020 programme under grant number 654142.

DELIVERY SLIP

	<i>Name</i>	<i>Partner/Activity</i>	<i>Date</i>
From:	Baptiste Grenier	GNUBILA	14/01/2016
Moderated by:	Małgorzata Krakowian	EGI.eu/WP1	18.02.2016
Reviewed by	Sergio Andreozzi	EGI.eu/WP2	18.02.2016
	Johan Montagnat	FR CNRS IDGC/PMB	18.02.2016
Approved by:	AMB and PMB		3.03.2016

DOCUMENT LOG

<i>Issue</i>	<i>Date</i>	<i>Comment</i>	<i>Author/Partner</i>
v.1	02/02/2016	First version	Alexandre Bonvin Marco Verlato Jérôme Revillard
v.2	18/02/2016	Revision after external review	Alexandre Bonvin Baptiste Grenier
FINAL	29/02/2016	FINAL version after external review	Baptiste Grenier Alexandre Bonvin

TERMINOLOGY

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

Contents

1	Introduction.....	6
2	Service architecture	7
2.1.1	High-Level Service architecture	7
2.1.2	Integration and dependencies	8
3	Release notes.....	14
4	Feedback on satisfaction	16
5	Future plans.....	16

Executive summary

The original aim of building an integrated web portal¹ for the MoBrain competence center² was to provide a common entry point that aggregates information from the WeNMR³ and N4U⁴ communities and link to relevant content on the respective sites and other related projects such as INSTRUCT⁵ and the WestLife Virtual Research Environment⁶. There have been however a number of developments since the start of EGI-Engage that question the need and usefulness of such an integrated portal:

- 1) The WeNMR community and its associated services are being integrated in the West-Life Virtual Research Environment e-Infrastructure project. The West-Life portal⁷ is tightly integrated with and hosted by the INSTRUCT ESFRI project, which should ensure its sustainability beyond the end of EGI-Engage and West-Life.
- 2) NeuGrid4You has launched a company, CEREBRO⁸, which is taking over some of their activities and services.

These developments raise the question if a fully integrated WeNMR/N4U portal would still respond to a need of the respective user communities. Because of this, in developing this integrated portal described in this deliverable we made the following choices:

- 1) We chose to develop a lightweight version of the portal, which provides an entry point to the MoBrain CC, based on Liferay technology for which the gnúbila partner has a strong expertise (the technology behind the N4U portal). This easily allows to extend its functionality when needed.
- 2) The portal contains mainly information about the MoBrain CC, its partners and achievements, and links to external respective sites for support and services. It offers a content management system with editing capabilities so that partner can keep adding content to the website as the project progresses (one additional reason to choose for the Liferay technology).
- 3) We chose to host the portal in a Virtual Machine on the EGI federated cloud. This avoids reserving hardware to run the portal, allowing for flexibility on where to run it as the EGI FedCloud provides mechanism to provision resources.

¹ <http://mobrain.egi.eu>

² https://wiki.egi.eu/wiki/EGI-Engage:Competence_centre_MoBrai

³ <https://www.wenmr.eu/>

⁴ <http://neugrid4you.eu/>

⁵ <http://www.structuralbiology.eu/>

⁶ <http://www.west-life.eu/>

⁷ <http://www.west-life.eu/>

⁸ <http://cerebro.pro/>

- 4) Its deployment and configuration are managed using the Software Configuration Management tool Puppet, called from a contextualization script launched at the end of the Virtual Machine creation.

The MoBrain portal provides the basis from which the integration/aggregation process can be continued under EGI-Engage with input from the various MoBrain partners. The choices made for its implementation should allow to react quickly to future needs of the communities, which might also mean it might get decommissioned if it appears it does not add value to the landscape of developing services and portals of the respective communities.

1 Introduction

Tool name	<i>MoBrain Web Portal</i>
Tool URL	https://mobrain.egi.eu
Tool wiki page	NA
Description	<i>Portlet-based Web portal by the MoBrain Competence Centre. The portal is the entry point, it will aggregate information from the various web portals of the WeNMR, INSTRUCT and N4U research communities.</i>
Customer of the tool	<ul style="list-style-type: none"> - Researchers - Public
User of the service	<i>TBD but mainly MoBrain CC partners / users</i>
User Documentation	NA
Technical Documentation	<i>Described in this document</i>
Product team	<i>Gnúbila</i>
License	https://www.liferay.com/downloads/liferay-portal/license-ce
Source code	https://www.liferay.com/downloads/liferay-portal/available-releases

The original aim of building an integrated web portal⁹ for the MoBrain competence center was to provide a common entry point that aggregates information from the WeNMR and N4U communities and link to relevant content on the respective sites and other related projects such as INSTRUCT and the WestLife Virtual Research Environment. There have been however a number of developments since the start of EGI-Engage that question the need and usefulness of such an integrated portal:

- 1) The WeNMR community and its associated services are being integrated in the West-Life Virtual Research Environment e-Infrastructure project. The West-Life portal is tightly integrated with and hosted by the INSTRUCT ESFRI project, which should ensure its sustainability beyond the end of EGI-Engage and West-Life.
- 2) NeuGrid4You has launched a company, CEREBRO, which is taking over some of their activities and services.

These developments raise the question if a fully integrated WeNMR/N4U portal would still respond to a need of the respective user communities. Because of this, in developing this integrated portal described in this deliverable we made the following choices:

⁹ <http://mobrain.egi.eu>

- 1) We chose to develop a lightweight version of the portal, which provides an entry point to the MoBrain CC, based on Liferay technology for which the gnúbila partner has a strong expertise (the technology behind the N4U portal). This easily allows to extend its functionality when needed as a lot of portlets are available, providing an easy to use Content Management System used for maintenance by multiple people and teams, a lot of additional features can be added and integrated very quickly, like a WiKi or a media library. The fact that the Liferay Portal is also following the portlets specifications (JSR 168 and JSR 286) is also a guarantee that other third-parties or custom portlets can be run and integrated easily making the portal very versatile.
- 2) The portal contains mainly information about the MoBrain CC, its partners and achievements, and links to external respective sites for support and services. It offers a content management system with editing capabilities so that partner can keep adding content to the website as the project progresses (one additional reason to choose for the Liferay technology).
- 3) We chose to host the portal in a Virtual Machine on the EGI federated cloud. This avoids reserving hardware to run the portal, allowing for flexibility on where to run it as the EGI FedCloud provides mechanism to provision resources.
- 4) Its deployment and configuration are managed using the Software Configuration Management tool Puppet, called from a contextualization script launched at the end of the Virtual Machine creation.

This document is meant to present the deployment of the portal at a technical level and try to be simple while exhaustive. The main target audience is therefore portal operators and system administrators responsible for its deployment. It should be clear that following the deployment procedure described in this document won't allow a user to deploy a personal instance of the portal as the configuration is managed by the Puppet tool and associated configuration file and only an allowed node will be able to get configuration information.

The following section describes the service architecture.

2 Service architecture

The MoBrain Web portal is built on multiple mature, robust and well known technologies. As explained above, we chose to host the portal deployment on the EGI federated cloud infrastructure. Deployment and configuration of the web portal on the cloud is fully automated enforcing a validated and reproducible deployment procedure. A single instance of the portal can be deployed at any time though.

2.1.1 High-Level Service architecture

The EGI federated cloud infrastructure is an e-infrastructure federating resources from multiple academic private clouds accessible using open standards such as OCCl. Multiple sites provide

cloud resources (mostly storage, computing and network resources) shared among multiple Virtual Organizations, allowing their users to easily manage it using well-known interfaces and compatible tools. Thanks to the OCCI compatibility, multiple heterogeneous infrastructures (some are OpenStack-based, others are OpenNebula-based) can be managed using a common set of tools, such as the rOCCI client whose usage is documented in the HOWTO11 on the EGI.eu wiki¹⁰.

The MoBrain web portal is a Liferay portal instance benefitting from the OCCI abstraction over multiple cloud providers. In fact during the deployment tests, the web portal was first published at INFN-PADOVA-STACK cloud site that was an OpenStack-based IaaS and is now in production at CESENT-MetaCloud, build using OpenNebula. A SLA¹¹ has been signed that ensures access to those resources for the coming year.

The Liferay portal is configured to be used by the MoBrain Competence Centre, using the mobrain.egi.eu domain name. A certificate for mobrain.egi.eu is also used to provide access to the portal over HTTPS only, thus the portal is accessible at <https://mobrain.egi.eu>.

A MoBrain-specific theme is deployed allowing quickly identifying and efficiently presenting the MoBrain Competence Centre.

2.1.2 Integration and dependencies

The MoBrain Web Portal has been deployed in a Virtual Machine running at CESNET MetaCloud, one of the sites integrated into the EGI Federated Cloud. CESNET-MetaCloud site was chosen as it:

- Provides the required storage, computing and network resources along with the desired OS template;
- Accepts compatible rules of usages for public services, allowing long lived and public services to be operated; and
- Supports the enmr.eu VO under which the VM is running.

An overview of the deployment is shown in Figure 1: the operator, shown on the right, is responsible of creating the puppet “manifests” that are expressing the desired state of the virtual machine configuration and are stored on the puppet master.

Puppet is working in a client/server model. On the managed node, a client, the puppet agent, is installed and interacts with the server, the puppet master, to receive its personal “catalog” based on its facts (certname, Fully Qualified Domain Name, IP, role, location,...), describing every resources (packages, services, configuration files,...) statuses (installed, running, containing that particular option/line...) and their dependencies. Once the puppet agent will have received its catalog it will apply the required steps to access to a state of “convergence” between the desired state of the node (expressed in the catalog) and the state of the local resources.

¹¹ <https://documents.egi.eu/document/2751>

The operator is also responsible for launching the Virtual Machine creation on the cloud using an OCCI-based tool, like the rOCCI a command line client. At the end of the node deployment a contextualization script is called in order to automatically install the puppet agent and run it.

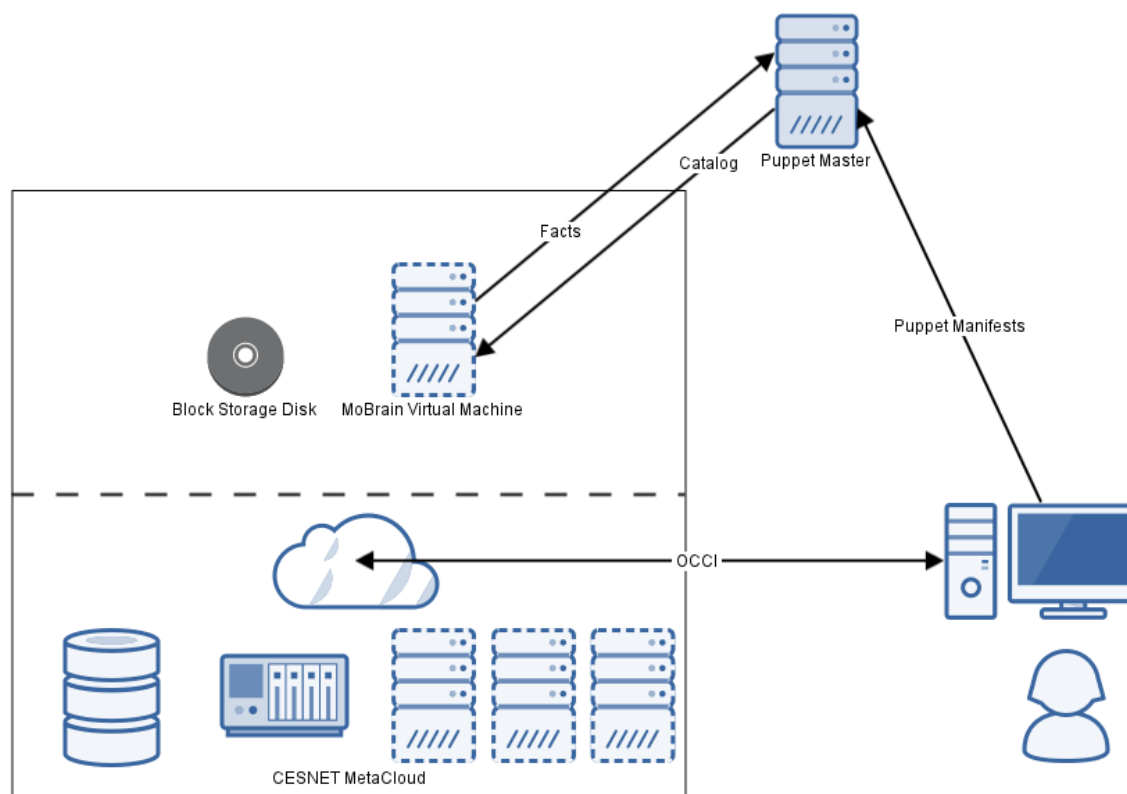


Figure 1: Deployment overview

2.1.2.1 The OS Template: EGI CentOS 6 base image

For the running the portal, we chose for the EGI CentOS 6 base image, an Operating System template created and endorsed by the EGI team and available from the AppDB¹². It provides a clean and solid base image that can be used to deploy services confidently on the EGI Federated Cloud. This template provides a mature, stable, tested and well known RedHat-based Operating System with an extremely long support period ending in November 2020.

2.1.2.2 Software dependencies

As shown in Figure 2, multiple components are used to run the MoBrain web portal:

¹² <https://appdb.egi.eu/store/vappliance/egi.centos.6>

- Liferay Portal 6.2 bundled with Apache Tomcat 7 servlet and JSP container
- MySQL 5.1 relational database
- Apache HTTPD 2.2 http server
- OpenJDK 1.7 providing Java 7 execution environment
- Puppet 4 for automating the deployment and configuration of the portal

The last piece of software used is backupninja, a set of shell scripts allowing to create incremental backups of the MySQL database and Liferay files on a server operated by gnúbila and located outside of the CESNET cloud site. The backup data could be used to quickly deploy another instance of the MoBrain portal in case of trouble.

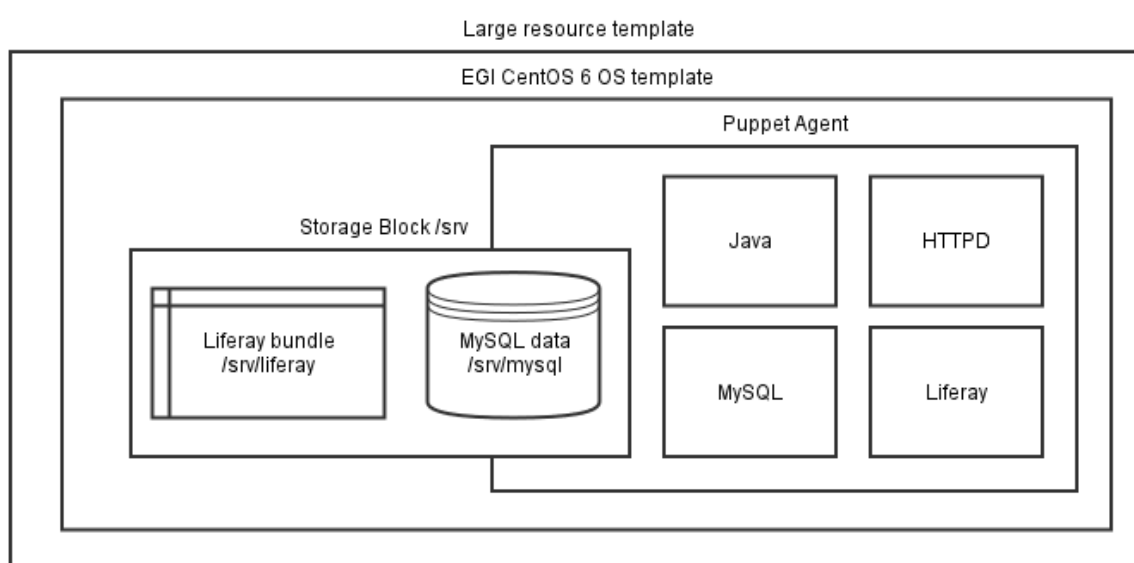


Figure 2: Overview of MoBrain web portal

2.1.2.3 EGI federated cloud: CESNET MetaCloud site

CESNET-MetaCloud site was chosen as it provides the required storage, computing and network resources along with the desired OS template as well as compatible rules of usages for public services, allowing long lived and public services to be operated.

2.1.2.4 The resource template: large

When creating a system image for deployment on the EGI FedCloud, a VM resource template is first selected. The resource template defines what will be the hardware resources available to the Virtual Machine, i.e. the number of Virtual CPUs and memory that will be assigned to the Virtual Machine instance. As the Liferay portal is java-based and uses a local MySQL database, it will benefit from running on a powerful Virtual Machine having multiple cores and a large amount of

memory, so the large resource template¹³ provided by CESNET-MetaCloud was chosen as it is powered by 4 CPU cores and 8 GB of memory.

2.1.2.5 Deployment

The deployment on the EGO Federated Cloud¹⁴ has been automatized, with only two manual steps required using the rOCCI tool:

- Creation of a storage block using the EGI Federated Cloud Storage (see below).
- Creation and launch of a Virtual Machine using the previously selected OS and resource templates and accessing the storage block.

When the Virtual Machine is started, a cloud-init contextualization script is run and is responsible for bootstrapping the MoBrain portal.

2.1.2.5.1 Creation of the block storage device

The block storage was created using rOCCI command line instructions as documented on the EGI Wiki in the HOWOT11¹⁵.

The block storage device is mounted in the Virtual Machine filesystem under /srv, and the Liferay bundle and MySQL files stored under /srv, allowing to easily separate the data from the operating system.

In EGI FedCloud, authentication is based on grid certificates, and a valid proxy must first have been issued to be passed to the occi command (-x command line option). The command used to create this 50 GB storage block named mobrain_data using rOCCI is the following:

```
occi -n x509 -x /tmp/x509up_u0 -X -e https://carach5.ics.muni.cz:11443 \
  --action create --resource storage \
  --attribute occi.storage.size='num(50)',occi.core.title=mobrain_data
```

This command returns the storage ID, in the form of /storage/<id>, usable to reference the storage block when linking it on the virtual machine.

2.1.2.5.2 Creation of the Virtual Machine

Once the Storage Block has been created, the Virtual Machine can be instantiated. It is created using the OS and resource templates found in the EGI AppDB as described above.

The rOCCI command to create the Virtual Machine, link the storage block and run the contextualization script is:

```
occi -n x509 -x /tmp/x509up_u0 -X -e https://carach5.ics.muni.cz:11443 \
  --action create \
```

¹³ http://schemas.fedcloud.egi.eu/occi/infrastructure/resource_tpl#extra_large

¹⁴ https://wiki.egi.eu/wiki/HOWTO09_How_to_use_Federated_Cloud_Storage

¹⁵ <https://wiki.egi.eu/wiki/HOWTO11>

```

--resource compute \
--mixin 'os_tpl#uuid_enmr_centos_6_fedcloud_warg_XXX' \
--mixin 'resource_tpl#large' \
--attribute occi.core.title="mobrain" \
--link /storage/<id>
--context user_data="file:///data/mobrain-bootstrap.cfg"

```

where `/storage/<id>` is the identifier of the block storage device created previously and XXX stands for the template version, currently 151.

Three points are noticeable in this command line example. The first one is the usage of the `mixin` parameter to specify the OS template and the resource template specifying the hardware resource and Operating System image to be used by the Virtual Machine. The second important point is the linking of the previously created storage block, which will allow accessing the storage block as a local disk. The last important point is the usage of a contextualization script available locally: `mobrain-bootstrap.cfg`. This script, only available to the operator of the portal, will be responsible for bootstrapping the MoBrain Web Portal.

2.1.2.5.3 Bootstrap using cloud-init contextualization script

The cloud-init script `mobrain-bootstrap.cfg`, used at the end of the Virtual Machine creation to configure the MoBrain Web Portal does the following:

- Creation of a `cloudadm` administrator user, allowing him to execute all commands using `sudo`, and associating to it the ssh keys of gnúbila's administrators, allowing them to connect securely on the Virtual Machine over ssh.
- Creation and formatting of a partition on the block storage device, only if not already done.
- Mounting of the storage partition on `/srv`.
- Downloading of a `bootstrap.sh` script from gnúbila's server.
- Launching of the `bootstraps.sh` script using the appropriate parameters: the role (`liferay_mobrain`) and location (CESNET) of the Virtual Machine.

The contextualization script is shown in the following, except that the XXX has to be replaced by the operator SSH key.

```

#cloud-config
#
# http://cloudinit.readthedocs.org/en/latest/topics/examples.html

users:
  - name: cloudadm
    sudo: ALL=(ALL) NOPASSWD:ALL
    lock-passwd: true
    ssh-authorized-keys:
      - ssh-rsa XXX

```

```

# Bootstrapping the installation
# * partition and format storage disk if not already done
# * mount mobrain storage disk under /srv
# XXX storage disk device id is hardcoded!
runcmd:
  - test -b /dev/vdc -a ! -b /dev/vdc1 && echo -e "o\nn\np\nl\n\n\nw" |
  fdisk /dev/vdc && sleep 1 && mkfs.ext4 -L mobrain_data /dev/vdc1
  - mount /dev/vdc1 /srv
  - curl -L -o /tmp/bootstrap.sh
  https://web.gnubila.fr/unattended/bootstrap.sh && chmod +x
  /tmp/bootstrap.sh
  - test -x /tmp/bootstrap.sh && /tmp/bootstrap.sh --apply-final-role=0 -
  -role=liferay_mobrain --location=cesnet --hostname=mobrain.gnubila.fr
  - test -x /opt/puppetlabs/bin/puppet && /opt/puppetlabs/bin/puppet
  agent -tv

```

In this process, the bootstrap.sh script, which is executed as last, sets up the required components: mostly installing and running the puppet agent to setup the server using Puppet.

Puppet is an IT automation ecosystem, enforcing the state of an infrastructure using a declarative language. Software Configuration Management allows to specify what software needs to be installed on a particular server and how it should be configured. Puppet is following an agent/master approach: an agent runs on the node (server, virtual machine...), this agent collects first information about the local node (e.g. network and kernel information, Operating System details and location) and connects securely to a master (puppet.gnubila.fr for the time being). Then the master, based on the information provided by the agent, builds and sends back to the agent a catalogue of resources with their configuration and interdependencies, applies it to the local node, putting every resource (package, file, service...) in the state described in the catalogue. So once the Puppet run has completed, the state of the virtual machine should reflect the desired state that was described and configured on the puppet master.

For our particular MoBrain portal case, the puppet agent downloads a Liferay tarball, deploys it under /srv on the block storage device, installs MySQL, installs and configures an Apache HTTP server and bootstraps the initial Liferay configuration. It also setups backups and applies some security-related configuration such as closing all unnecessary ports and adjusting the SSH configuration.

At this point the only missing final action is to connect the MoBrain Web Portal domain name, mobrain.egi.eu to the IP address that was assigned to the Virtual Machine upon creation.

2.1.2.5.4 Updating the public IP of the MoBrain Web Portal

During the Virtual Machine creation a public IP is added to the host, which needs to be linked to the domain name for the portal: mobrain.egi.eu. The CNAME alias of mobrain.egi.eu points to mobrain.gnubila.fr, and the mobrain.gnubila.fr domain is the one pointing to the public IP of the

Virtual Machine. The latter can be quickly updated if the VM is redeployed and gets a new IP; this is done using a homemade script and the appropriate API provided by OVH, the registrar of the gñubila.fr domain.

Updating the domain name with the public IP of the compute resource:

```
curl -L -O https://web.gnubila.fr/ovh/update_ip.sh
chmod +x update_ip.sh
./update_ip.sh XXXXX XXXXX XXXXX mobrain.gnubila.fr XXX.XXX.XXX.XXX
```

Here the first three XXXXX blocks are the secret API key and the last XXX.XXX.XXX.XXX argument is the IP address assigned to the VM and that can be retrieved using the OCCI interface and the describe action on the compute resource (the Virtual Machine).

Once this has been done, <https://mobrain.egi.eu> can be used to access the MoBrain Web Portal.

3 Release notes

The initial version of the web site has been populated with some content and is organised with the following sections:

- Main Page: This page contains an overview of the project as well as a pointer to the N4U and WeNMR communities. Last but not least, a Blog module has been added in order to be able to publish news about the project.
- Partners: contains the name of the project partners
- Documentation: For the moment, this section is decomposed in two sub-section:
 - Technical: will contain project specific technical documentation
 - Project: contains the project description as well as the different tasks and deliverables planned during the project.
- Contact Us: This is a pointer to the community contacts as well as support pages.



Figure 3: Web site main page overview

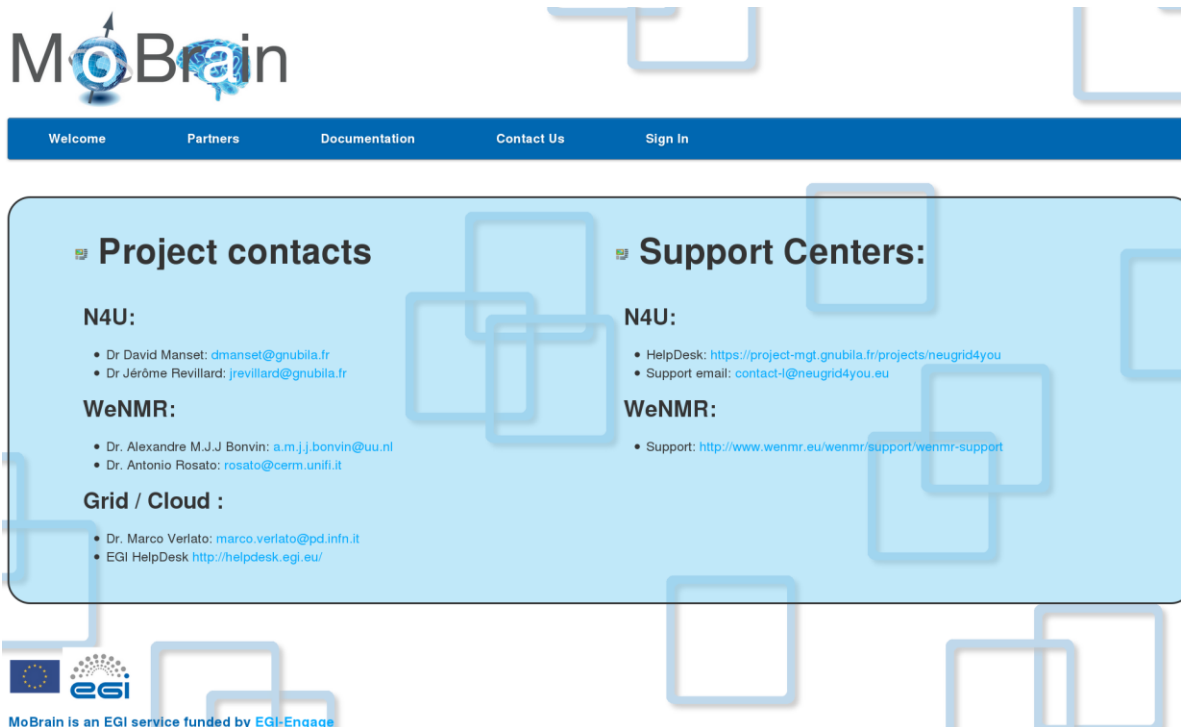


Figure 4: Contact page

4 Feedback on satisfaction

The MoBrain portal has only been running for a limited amount of time and as such user satisfaction is difficult to monitor (no stats available). It does however provides the required functionalities (Content Management System and online editing capabilities) that have already allowed MoBrain project partners to successfully add and modify content. Since, however, most of the content relevant to the respective communities is hosted on external sites (N4U, CEREBRO, West-Life) to which the web site is pointing to, we do not foresee a heavy usage of the tools provided.

5 Future plans

We will monitor the use of the portal by the MoBrain CC partners and ask for their feedback to identify missing components. However, as explained already, we should avoid duplicating efforts with other e-Infrastructure projects. In particular, the West-Life VRE web portal, which will integrate the WeNMR services, should become the main entry point in the future for the structural biology community. Being hosted and linked to the INSTRUMENT ESFRI project, its long-term perspectives for sustainability are better than for the MoBrain portal described in this document. The future will thus tell if it is worth running this portal using the EGI Fed Cloud resources, or rather only the use the EGI wiki pages for the MoBrain CC and directly direct our users to the respective community portals (N4U and West-Life).