EGI-Engage

ELIXIR Competence Centre

Demonstrator for ELIXIR workflows implemented in the EGI Federated cloud

D6.15

Date	27 September 2017
Activity	WP6
Lead Partner	CSC
Document Status	FINAL
Document Link	https://documents.egi.eu/document/3019

Abstract

The ELIXIR Competence Centre (CC) of the EGI-Engage project facilitates collaboration between EGI and ELIXIR service developers and service providers. During its 24-month lifetime the CC collected, analysed and compared life science community requirements with EGI technical offerings, and designed and implemented pilot application setups from the life science community using EGI cloud services. This document is the final deliverable of the CC: the description of the 6 scientific use cases, their implementation in a joint EGI-ELIXIR federated cloud infrastructure, possible further development of the demonstrators, lessons learnt from the CC activities and recommendations for future work for ELIXIR and EGI concerning cloud services.

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

DELIVERY SLIP

	Name	***Partner/Activity***	***Date***
From:	Kimmo Mattila	CSC / SA2	19.09.2017
Moderated by:	Malgorzata Krakowian	EGI Foundation/WP1	
Reviewed by	Yannick Legre	EGI Foundation/WP1	15.09.2017
Approved by:	AMB and PMB		27.09.2017

DOCUMENT LOG

Issue	***Date***	***Comment***	***Author/Partner***
v.0	07/17	TOC created Part of META-pipe information added.	Gergely Sipos / EGI.eu Kimmo Mattila / CSC
V0.1-0.9	07/17	Partner contributions added: • cBioPortal: Miroslav Ruda (CESNET) • META-Pipe: Kimmo Mattila (CSC) • Marine Metagenomics: Steven Newhouse (EMBL-EBI) • Insyght: Christophe Blanchet (CNRS, IFB) • PhenoMeNal: Steven Newhouse (EMBL-EBI) • JetStream: Enol Fernandez (EGI.eu)	
V0.99	11/08/17	Update structure; Add abstract, executive summary, lessons learnt and future work	Gergely Sipos / EGI.eu
V1	18/08/17	Clean version for external review	Gergely Sipos / EGI.eu
V2	04/09/17	Typos fixed	Gergely Sipos / EGI.eu
FINAL		Final version after external review	

TERMINOLOGY

A complete project glossary and acronyms are provided at the following pages:

- https://wiki.egi.eu/wiki/Glossary
- https://wiki.egi.eu/wiki/Acronyms

Further information about the ELIXIR Competence Centre is available at:

- https://wiki.egi.eu/wiki/CC-ELIXIR

Contents

Executive summary

The ELIXIR Competence Centre (CC) of the EGI-Engage project aimed at evaluating, adopting and promoting technologies and resources from EGI to the wider ELIXIR research community. This was done with an incremental approach:

1. Bringing together designated life science experts from ELIXIR and technical experts from EGI within the CC.
2. Identifying life science use cases that could benefit from EGI services, primarily from the EGI Cloud service, a federated cloud. Analysing the e-infrastructure requirements of the use cases.
3. Implementing the use cases as demonstrators based on EGI and ELIXIR services as required. Collaborating during implementation with relevant EGI and ELIXIR partners.
4. Demonstrating and evaluating the implementations. Disseminating the experiences gained with the use cases towards ELIXIR, EGI and other relevant communities. Planning the long-term adoption of EGI services within ELIXIR based on the pilot experiences.

This document is the final deliverable of the CC: the description of the 6 scientific use cases that were selected[1] for implementation in the EGI Cloud service. The report starts with a description of the federated cloud infrastructure that was setup by the CC members and external partners to serve the demonstrators (Section 1), continues with the description of the 6 demonstrators, (Section 2-7) and concludes with the lessons learnt from the activities and recommendation for future work based on these findings (Section 8).

A joint ELIXIR-EGI federated cloud infrastructure was setup for the demonstrators, combining core components of the EGI Federated Cloud with ELIXIR cloud providers and with the ELIXIR Authentication and Authorization system. This infrastructure is expected to become a core element of the ELIXIR Compute Platform, envisaged as a federation of compute services to serve various life science applications and services from the boarder ELIXIR community.

The 6 demonstrators and their outcomes (indicated after '→') were the following:

1. cBioPortal (from CESNET): porting a long-running service (a portal) to the EGI-ELIXIR federated cloud technology, and hosting it within the federated ELIXIR cloud. → The demonstrator did not reach the level of using a federated cloud because the user community's priority has changed during the project: they do not see demand for additional/portable portal instances, but rather the need of hosting a single portal instance as a high availability service in a cloud. Such as setup was achieved in the CESNET cloud, using following local access policies and using local cloud interfaces.
2. META-Pipe metagenomics pipeline (from CSC): Burst out the compute-intensive part of an analysis pipeline to third party cloud resources. → The pipeline was able to use the CESNET resource of the federated cloud testbed, as well as the cPouta cloud of CSC (in non-

[1] EGI-Engage M6.3 Life science requirements analysis and driver use case(s) with implementation roadmap

federated mode, via local interfaces and policies). The setup was demonstrated at various training courses and workshops.

3. Marine metagenomics (from EMBL-EBI): Offloading the compute part of the EMBL-EBI hosted online service to cloud resources, to ensure the sustainability of the service in terms of compute capacity. → The infrastructure was ported across six providers (AWS, Google Cloud Platform, Azure, EMBL-EBI Embassy Cloud, IN2P3-IRES and UK Cloud). Two of these were used from the federated EGI-ELIXIR testbed (EMBL-EBI and IN2P3-IRES).
4. Insyght Comparative Genomics (from CNRS IFB): enable life-scientist to instantiate their own instance of Insyght analysis environment in the cloud as a cluster of VMs, offloading users from the central, public Insyght instance. → Two different configurations have been implemented on the EGI-ELIXIR federated cloud testbed, and have been demonstrated to members and partners of the French ELIXIR node.
5. PhenoMeNal (from EMBL-EBI): Porting the PhenoMeNal Virtual Research Environment (VRE) to various cloud platforms, including the EGI Federated Cloud; and provide all the tools required for users to deploy the VRE in those clouds on demand. → EGI Federated Cloud was initially considered as a deployment target but later exploration proved that support for the orchestrator chosen by the project (Terraform) was not available. Terraform integration has in the meantime been added by EGI, but not in a timescale compatible with the deliverables of the Phenomenal project. The EGI-ELIXIR federated cloud will be considered again as a deployment target in the forthcoming releases of the platform.
6. JetStream (from University of Indiana, US): This interoperability use case aimed to assess what are the technical requirements to enable sharing VMs images between the JetStream cloud and ELIXIR federated cloud VO so, users in the US and Europe can easily access the same tools and run them on clouds they have access to. Such interoperability would lower the cost of developing and maintaining virtualised tools and applications for life sciences. → Images from EGI/ELIXIR can be executed on JetStream, but images from JetStream are KVM-specific so not portable across all EGI providers.

The partners captured 10 lessons and recommended future work items from the performed activities: 5 of them directly relating to specific demonstrators, and another 5 as generic observations relating to the ELIXIR/EGI federation model and infrastructure. Our recommendations span across a broad range of topics, from technical activities (such as extending AAI on cloud sites) to policy-networking activities (such as setup of Operation Level Agreements). The recommendations will be proposed for implementation to the 'ELIXIR Compute Platform' group (group leaders are involved in this CC), and to the second phase of the ELIXIR Competence Centre, which is expected to run between 2018 and 2020 (3 years) within the EOSC-hub H2020 project.

From the point of view of life science research community the key result of this Competence Centre was gathering up these six use cases and setting them up to collaborate with the EGI infrastructure. The use cases are examples of research collaborations that have much longer tine span than the CC. However, the ELIXIR Competence Centre project enabled these research

collaborations to recognize and give feedback how e-Infrastructures such as EGI.eu are able to deliver data analysis and storage for the increasingly data intensive life science research.

Introduction

1.1 Background and motivations

ELIXIR[2] is a pan-European research infrastructure in agreement between 20 European governments to build a sustainable European infrastructure for biological information, supporting life science research and its translation to medicine, agriculture, bio-industries and society.

EGI[3] is a pan-European e-infrastructure that delivers integrated computing services to European researchers, driving innovation and enabling new solutions to answer the big questions of tomorrow.

Data analysis on life sciences, that is the focus area of ELIXIR, is a fast moving field. For the EGI services to become relevant and help keep European Life Sciences competitive globally, it is important to develop mechanisms that allow the research infrastructure to flexibly meet new challenges and respond to new scientific and technical developments.

The ELIXIR Competence Centre (CC) of the EGI-Engage project aimed at evaluating, adopting and promoting technologies and resources from EGI to the wider ELIXIR research community. This was done with an incremental approach:

1. Bringing together designated life science experts from ELIXIR and technical experts from EGI within the CC.
2. Identifying life science use cases that could benefit from EGI services, primarily from the EGI Cloud service, a federated cloud. Analysing the e-infrastructure requirements of the use cases.
3. Implementing the use cases as demonstrators based on EGI and ELIXIR services as required. Collaborating during implementation with relevant EGI and ELIXIR partners.
4. Demonstrating and evaluating the implementations. Disseminating the experiences gained with the use cases towards ELIXIR, EGI and other relevant communities. Planning the long-term adoption of EGI services within ELIXIR based on the pilot experiences.

This document is an ELIXIR Competence Centre deliverable that summarizes the results of the 6 use cases that helped the CC members to assemble a federated cloud testbed infrastructure and demonstrated the usage of ELIXIR and EGI services within the testbed. The use cases, discussed here were using a variety of services provided by EGI. Some of the tools and functionalities were developed during the life span of this Competence Center within the CC (e.g. Terraform interface),

[2] http://www.elixir-europe.org/
[3] http://www.egi.eu/

and beyond the CC (e.g. ELIXIR AAI, EGI CheckIn). The results demonstrate that many of the EGI services (e.g. EGI Federated Cloud computing environment and AppDB virtual appliance repository and VM Management Dashboard) are already feasible for use by ELIXIR life science communities.

However in several areas, (e.g. user in authentication, accounting and data management) more development and integration work is needed.

1.2 The testbed infrastructure

The ELIXIR Competence Centre established a federated cloud using the EGI Federated Cloud technology and partner clouds from ELIXIR and EGI. The established infrastructure is called 'vo.elixir-europe.org' Virtual Organisation (VO) within the EGI service management systems, and it served as a testbed infrastructure for the 6 demonstrators. In the long term VO[4] is expected to serve as the core of the ELIXIR Compute Platform, a federated infrastructure that joins together cloud compute and storage resources from ELIXIR and EGI providers to make these available for life science applications.

What is the EGI federated cloud?

The EGI Federated Cloud is a standards-based, open cloud system that federates institutional clouds to offer a scalable computing platform for data and/or compute driven applications and services in research and science. The EGI Federated Cloud currently includes 23 cloud sites from all across Europe. These clouds are available for users through community allocations, so called Virtual Organisations. Each Virtual Organisation includes a subset of the federated cloud sites, and makes those available for the given community through generic and/or community-specific policies and protocols. Members of a scientific community have to join the VO to access the cloud capabilities offered by the federated VO sites. The EGI federation model ensures single-sign on (i.e. after a user registers to the VO he/she is able to access every VO cloud); uniform interfaces (i.e. each VO cloud can be accessed via the same/harmonized interfaces) and application portability (i.e. every VO cloud uses the same Virtual Machine (VM) image and contextualization format). VO members can deploy new VMs on the cloud sites through the EGI AppDB VM marketplace, and can instantiate VMs and block storages via the graphical AppDB VMOps Dashboard or using the API and command line interfaces offered by the cloud sites. High level tools, such as orchestrators and application portals can offer additional, and science domain-specific capabilities for users.

The EGI Federated Cloud is built from open source software components, maintained by an open consortium, the Federated Cloud Task Force. The technology stack is currently capable of federating OpenStack, OpenNebula and Synnefo clouds. EGI promotes the federated cloud technology stack for scientific communities who want to establish community-specific cloud federations, and assists them through this process. The EGI security and operational policies and service management practices offer a baseline, but customizable framework for operating the community specific cloud federations.

Why EGI Federated cloud for ELIXIR?

The EGI Federated Cloud, the underlying technology stack and the related operational and security processes enable scientific communities to (1) share resources and applications across institutes and national borders; (2) develop portable, standard-based applications; (3) operate high-quality services for

[4] Technical information about the VO is available in the VO ID card in EGI Operations Portal: http://operations-portal.egi.eu/vo/view/voname/vo.elixir-europe.org

science; and ultimately to (4) establish sustainable e-infrastructures for large-scale, digital science.

During 2015 the ELIXIR community – in collaboration with various e-infrastructures and other service providers – initiated the development of the reference architecture for ELIXIR, called the 'ELIXIR Compute Platform' (ECP). The prime role of the ECP is to support the use cases of the ELIXIR-EXCELERATE H2020 project, however, the platform is expected to serve other ELIXIR-related use cases from ELIXIR and other biomedical sciences Research Infrastructures. The ECP is envisaged as a federation of compute and storage resources, operated according to community-agreed principles and with the use of centrally provided federator services (e.g. Authentication and Authorisation combined with centralized user identity). The concept is very similar to the EGI Federated Cloud, therefore the ELIXIR CC was setup to assess the EGI Federated Cloud, and to deploy the first ECP implementation based on the EGI federated cloud architectural implementation.

The ELIXIR federated Cloud is currently supported by the following resource providers:

- CESNET-MetaCloud: OpenNebula cloud (interface support: OCCI)
- IN2P3-IRES: OpenStack cloud (interface support: OCCI and OpenStack)
- EMBL-EBI: OpenStack cloud (interface support: OCCI and OpenStack)
- GRNET: Synnefo cloud (interface support: OCCI)

The table below summarizes the different access options to interact with the providers:

	Options to interact with the clouds that provide OCCI interface:	Options to interact with the clouds that provide OpenStack Interface:
Graphical Interface	AppDB VMOps Dashboard	OpenStack dashboard
API level	OCCI with jOCCI (Java) or rOCCI (Ruby)	OpenStack API with python SDK
Command line	rOCCI-cli	OpenStack CLI with VOMS plugin
Orchestrator	Terraform OCCI plugin Infrastructure Manager OCCOPUS	Terraform EGI-OpenStack plugin Infrastructure Manager OCCOPUS

These ELIXIR providers were federated using the following centrally provided federator services:

- EGI GOCDB service registry: A database where basic information are stored about the federated cloud compute and storage resources (such as endpoint URL, system administrator contact, exposed capabilities).
- EGI AppDB: a database where ELIXIR-specific Virtual Machine Images and contextualisation scripts are registered and made available for the ELIXIR cloud sites for download.
- EGI AppDB VMOps dashboard: A graphical portal that provides interfaces for ELIXIR VO users to instantiate and to manage Virtual Machines and block storages on the federated cloud resources.

- ELIXIR Authentication-Authorisation Infrastructure and the EGI CheckIn service: together they enable users to register in the ELIXIR VO and use the VO clouds with personal ELIXIR IDs[5].
- RCAuth Master Portal: translates ELIXIR IDs to X.509 proxy credentials. These credentials are recognised and used by the cloud sites to authenticate and authorize users.
- EGI Monitoring system (ARGO): to test the availability and reliability of the federated cloud sites using a well-defined set of probes. The monitor also raises alarms towards the operators in case tests fail on their sites.
- EGI Accounting system and accounting portal: collects usage statistics from the federated cloud sites and offers various views to browse users' resource consumption data.

Full integration details about the ELIXIR federated cloud are available in the previous ELIXIR CC deliverable, D6.10[6], which also provides information for interested providers on how to federate their cloud resources into the ELIXIR infrastructure. These instructions are kept up-to-date on a Wiki page[7].

Users who want to access the ELIXIR VO need to ask for VO membership[8]. An ELIXIR account is required for this process so the user is redirected to create one (or login with his/her existing ELIXIR account). VO membership is received and evaluated by the VO Managers and a notification email about the approval/rejection of the VO membership request will be sent. Full documentation for users is provided in the ELIXIR CC Wiki[9].

2 cBioPortal replication use case

2.1 Overview

Name	cBioPortal replication in EGI Federated cloud
Responsible person within the CC	Miroslav Ruda, CESNET
URL	*https://cbio.cerit-sc.cz/*
Description	The aim of the use case was to package the cBioPortal service into Docker images that are compatible with EGI Federated Cloud. This would allow the service to be cloned on-demand, according to the needs of existing and future user communities.

[5] https://www.elixir-europe.org/intranet
[6] D6.10 Infrastructure tests and best usage practices for life science service providers. https://documents.egi.eu/document/2802
[7] Cloud Resource Centre Installation Manual https://wiki.egi.eu/wiki/MAN10
[8] ELIXIR VO Registration: https://perun.elixir-czech.cz/registrar/?vo=elixir&group=EGI:vo.elixir-europe.org
[9] https://wiki.egi.eu/wiki/ELIXIR_Virtual_Organisation

Value proposition	Users could easily launch their own copies of this service. This would enable e.g. ensuring the availability of the computing capacity when needed and providing dedicated cBioPortal service for analysis cases that can't be uploaded to a public server.
Customer/user of the demonstrator	This use case was targeted for European cancer researchers using the EurOPDX data and service providers supporting them.
Envisaged scenario	Creating a new cBioPortal instance in the cloud if one of these two conditions hold: (1) Capacity of the cBioPortal server is not sufficient for the expanding user community OR (2) a user group wants to use a dedicated cBioPortal service for uploading and analysing data.
Demonstration success criteria	Success criteria: EGI Federated Cloud compatible Docker images of cBioPotral are available and tested. The above success criteria were not achieved. An instance of the cBioPortal dedicated to EurOPDX data visualisation was deployed on Czech MetaCloud site. After that the development work was halted, as the user community did not need any additional cBioPortal servers.
User Documentation	*NA*
Technical Documentation	*NA*
Developer team	*NA*
License	*NA*
Source code	*NA*

2.2 Architecture

The cBioPortal allows users an exploration of multidimensional cancer genomic data (http://www.cbioportal.org/, Gao et al. Sci. Signal. 2013, Cerami et al. Cancer Discov. 2012). In this pilot stage of the EurOPDX cBioPortal (available at https://cbio.cerit-sc.cz) 5 colorectal or breast cancer PDX models cohorts (or "studies") from 4 laboratories of the consortium are included, and it is our aim to gradually include PDX data from the whole consortium. A study summary is available for each of them, and one can also exploit the cBioPortal tools for browsing DNA copy-number data, mRNA expression data and mutation data for a single study, or across studies by querying for instance series of genes such as EGFR, ERBB2, ERBB3, EGF, EREG, KRAS.

The service is deployed as virtual machine, which contains several Docker containers, one with MySQL machine, second with java application (portal in Tomcat environment). Data can be supplied either from publicly available cohorts (studies) or from internal dataset, which cannot be available publicly.

2.3 Demonstration

2.3.1 Scenario

During the ELIXIR CC project the cBioPortal provider group's priorities changed: instead of making a fully portable and cloneable portal setup, they preferred hosting a single cBioPortal in a single cloud site (CESNET). This was achieved using the access policies and interfaces that the CESNET cloud exposes to users.

2.4 Future plans

The operation of the cBioPortal will continue within the CESNET cloud, supported by the Czech NGI, for the benefit of the EurOPDX collaboration[10]. The partners have no plan to port the portal server to the federated ELIXIR cloud resources.

3 META-Pipe metagenomics pipeline

3.1 Overview

Name	Integrating the META-pipe analysis pipeline with cloud services from EGI
Responsible person within the CC	Kimmo Mattila, CSC
URL	*NA*
Description	META-pipe, developed at the University of Tromsö, is an analysis pipeline that is designed to fulfil the needs of marine metagenomics data analysis. META-pipe integrates existing biological analysis frameworks, and compute and storage infrastructure resources to provide an easy to use but effective analysis platform. META-pipe is also an important component in one of the four scientific use cases in the ELIXIR-EXCELERATE H2020 project[11].

[10] http://europdx.eu/ongoing-work-publications.html
[11] https://www.elixir-europe.org/news/elixir-accelerates-major-horizon-2020-funding

	The use case demonstration aimed at showing how the computationally demanding parts of META-pipe can be executed on federated EGI-ELIXIR cloud resources.
Value proposition	The possibility to shift the computationally demanding parts of the META-pipe analysis pipe line to the EGI federated cloud allow opening up the tool for wider user community: A new user group could apply the computational resources from its partner clouds and then utilize these resources easily through the interface provided by the META-pipe developers. The technologies and protocols used in the EGI-ELIXIR federated cloud would ensure compatibility and portability of the META-pipe analysis part across clouds.
Customer/user of the demonstrator	The service is aimed for academic laboratories, research groups or institutes doing marine metagenomics. During the ELIXIR CC period it was used as a back-up service on a Marine Metagenomics course hosted by ELIXIR Finland.
Envisaged scenario	Use case scenario: A research project or training course wants to use META-pipe analysis pipeline. The technical manager of the research group/course launches a temporary META-pipe server that the end users can use through the normal META-pipe interface. Once the analysis phase or course is over, the virtual cluster is closed.
Success criteria	The success criteria of the use case was the ability to set up a META-pipe server in the EGI Federated Cloud so that it is able to process analysis tasks submitted from the META-pipe interface. This was achieved, but so far the service has been tested only in one EGI Federated Cloud site (CESNET). At the moment the pipeline is available only for Norwegian academic users.
User Documentation	https://github.com/cduongt/mmg-cluster-setup-CESNET
Technical Documentation	https://github.com/cduongt/mmg-cluster-setup-CESNET
Developer team	Cuong Duong Tuan (CESNET), Aleksander Agafonov(UiT),Lars Ailo Bongo (UiT), Inge Aleksander Raknes (UiT), Kimmo Mattila (CSC)
License	*NA*
Source code	https://github.com/cduongt/mmg-cluster-setup-CESNET

3.2 Architecture

META-pipe is an automatized workflow for analysing sequence data obtained from marine metagenomics samples. Pipeline is developed by Willassen *et al* at ELIXIR-Norway. META-pipe takes raw sequencing data as an input and outputs taxonomic classification and functional annotation of the sample. META-pipe analysis steps can be divided in three modules as shown in figure 1: pre-processing and assembly (blue), classification (yellow), and functional annotation (green).

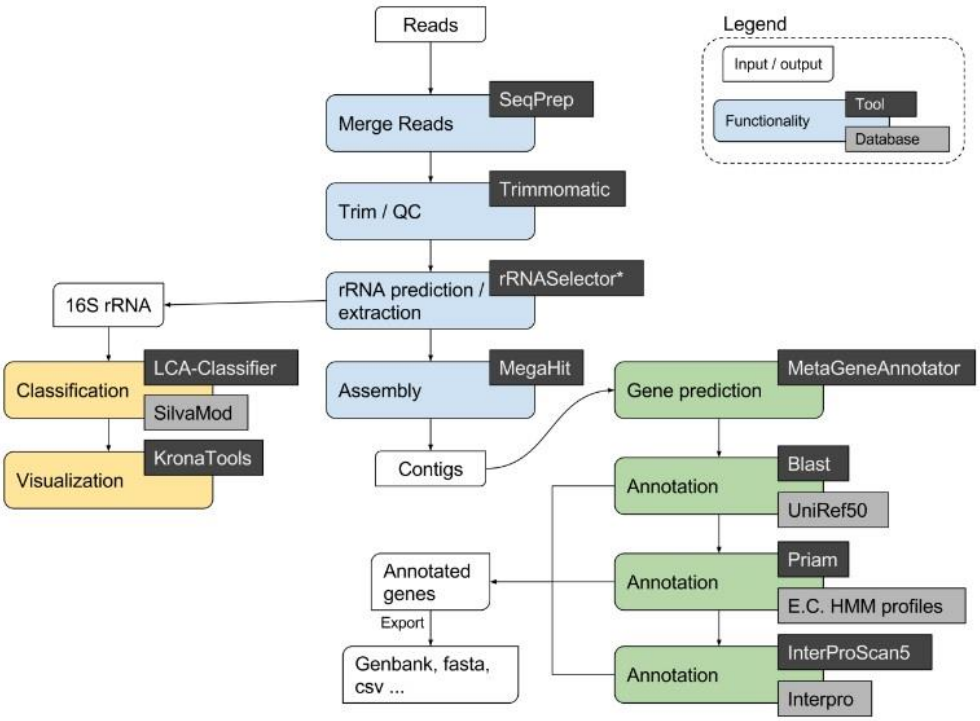


Figure 1. META-pipe analysis workflow. The computationally heavy part of the pipeline, that is executed in the cloud environment, includes the four annotation steps, displayed as green boxes, in the lower right corner of the flow chart.

The group of Willassen maintains META-pipe interface and service, but due to heavy computing that is required by the annotation steps, the service has been opened for Norwegian users only. To enable the extension of the user community the META-pipe developers have encapsulated the annotation steps into a virtual cluster environment that can be launched with simple command line tools. In ELIXIR Competence Centre we have tested launching and using this annotation server in the OpenStack cloud environment of CSC (Elixir Finland) and then in the federated ELIXIR-EGI cloud testbed:

The OpenStack version of the launcher tool was developed by Aleksander Agafonov (UiT). This tool was then fitted into the EGI Federated cloud by Cuong Duong Tuan (CESNET). The EGI FedCloud compatible launcher is a command line tool that requires following components:

- Linux distribution (tested on Fedora 24)
- rOCCI Client
- X509 VOMS certificate
- Python 3
- Terraform and OCCI plugin
- Ansible

The manager of the server must provide a contextualization file and Terraform configuration file that define the technical features of the virtual cluster. When the launching command is issued

the tool first builds the virtual cluster to the given endpoint and then automatically installs the software components and reference datasets to the new virtual cluster.

The end users (researchers, students) will submit analysis tasks from the META-pipe interface to the META-pipe server running in EGI Federated cloud. The end users do not need certificates, VO membership or the tools required to launch the META-pipe server. Instead, the end users just authenticate to the META-pipe web interface using ELIXIR AAI.

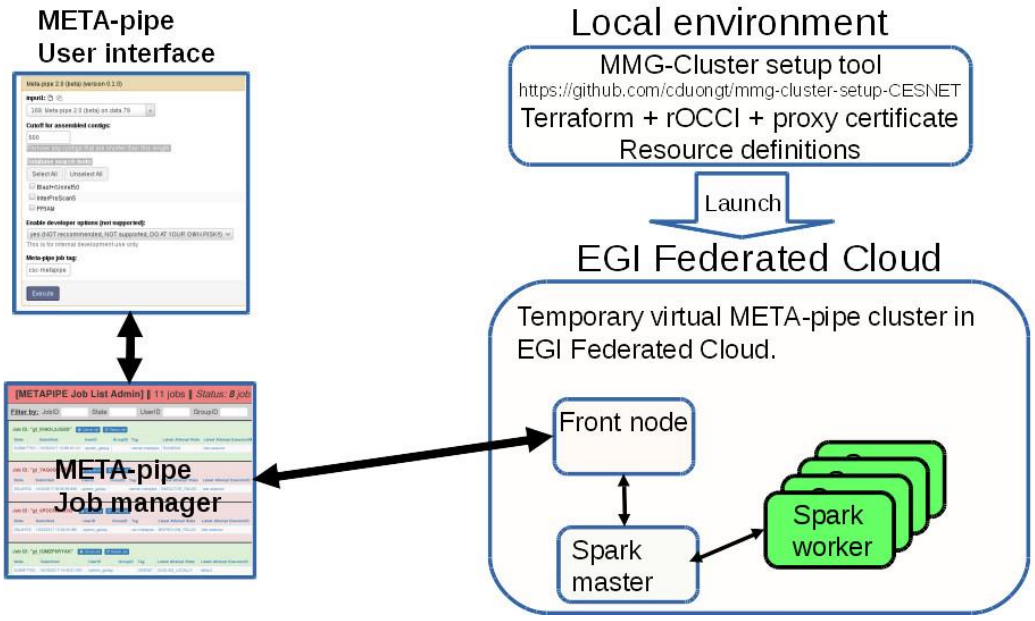


Figure 2. Using virtual META-pipe server. The local manager uses a MMG-Cluster setup tool to launch a temporary META-pipe annotation server to the EGI Federated Cloud. In the user interface, the end user adds server specific name tag to the analysis task submitted. Based on the name tag, the job manger submits the task to a specific META-pipe annotation server.

3.3 Demonstration

3.3.1 Scenario

A set-up where the computationally heavy parts of the META-pipe analysis pipeline were executed in external cloud environment was utilized in a metagenomics course organized by the Finnish ELIXIR node on April 2017 (https://www.csc.fi/web/training/-/metagenomics). Two temporary METApipe-servers were launched for the course: the main sever was running in the cPouta cloud environment at CSC and a backup server that was running in EGI Federated Cloud. During the course the students used a METApipe web interface so that the analysis tasks were computed in the cloud environment instead of the local (i.e. UiT) servers.

These METApipe servers we set up by the course organizers and so the students didn't need to any technical preparations to use the cloud services. Instead, they only needed to define one extra parameter in the web interface to guide their analysis tasks to a specific external METApipe sever.

3.3.2 Feedback

42 students participated to the metagenomics course. The participants were not aware and didn't notice that when they executed the annotation part of the exercises, they were actually using a virtual cluster running in a cloud environment. Thus they could not give any feedback about using a cloud service.

However, in the course feedback and requests submitted later on, many users have been asking, when they cloud start using this service, which is a rather promising reaction.

3.4 Future plans

The development of the META-pipe workflow will continue in the ELIXIR community even after the EGI-Engage project. In the future, the key issue will be obtaining computing resources for META-pipe end users. Resources from the EGI Federated Cloud or the ELIXIR federated cloud testbed are ideal candidates, if access guarantees (SLAs, OLAs) are arranged. The process of how resources will be obtained, managed and allocated to the individual end users is still to be decided. Possible solutions could be e.g. that ELIXIR nodes host META-pipe servers (in local cluster or in EGI Federated Cloud) for their local users. Other possibility is that each user group applies resources for their own needs (e.g. through ELIXIR VO) and maintain a temporary META-pipe server in some cloud environment. In both cases an up-to date and highly automatized tool for launching a META-pipe server in EGI Federated Cloud will be very useful.

To achieve this following things are needed:

- META-pipe server launching tool that is up-to-date and tested widely in EGI-ELIXIR federated clouds.
- Method/politics for applying resources for running a META pipe environment.
- Instructions, guides of META-pipe service providers and end users.

4 Marine metagenomics use case

4.1 Overview

Name	Marine Metagenomics
Responsible person within the CC	Steven Newhouse, EMBL-EBI

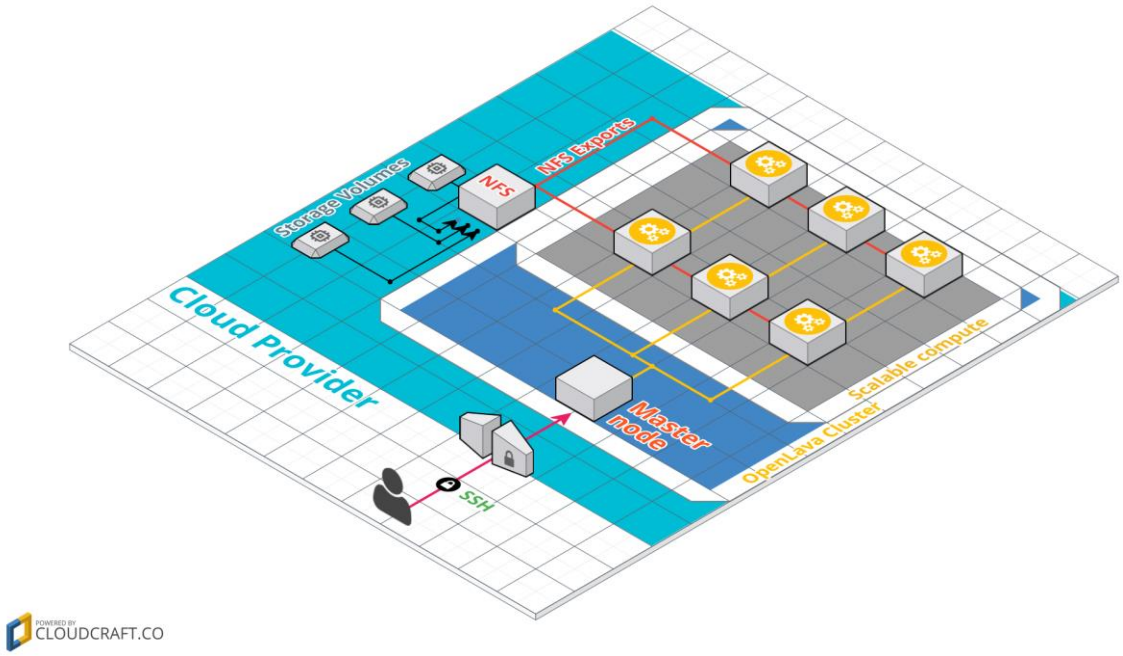
URL	*NA*
Description	The Metagenomics pipeline, developed by the EBI Metagenomics team, currently underpins the EBI Metagenomics Service (https://www.ebi.ac.uk/metagenomics/) at EMBL-EBI. This service, which stores more than 1000 publicly-accessible studies for a total of more than 300 billion nucleotide sequences, is widely adopted by the metagenomics community for the analysis and long-term storage of their sequences. With the amount of metagenomics data deposited increasing on a daily basis, the possibility of offloading at least part of the compute to cloud resources would represent a very important step forward to ensure the sustainability of the service in terms of compute capacity. The aim of this demonstrator was to port the Metagenomics pipeline to cloud resources, the EGI Federated Cloud included, and to test the feasibility of running the pipeline at production scale.
Value proposition	The Metagenomics pipeline is currently sustained at production scale by the EMBL-EBI compute resources, specifically the EMBL-EBI production cluster. However, with the increase in demand due to the ever-growing amount of data deposited in the archives, the pipeline is now limited in throughput due to its very high IOPS needs. Being able to offload at least a fraction of this compute to cloud resources would help delivering results back to the metagenomics community in a more timely fashion as well as freeing up resources on the EMBL-EBI systems for other users to take advantage of. Also, a complete porting to cloud resources would allow third parties, both scientists and industries, to run their own instance of the Metagenomics pipeline further improving the turnaround time.
Customer/user of the demonstrator	Scientists, SMEs and industry interested in the metagenomics field. The EMBL-EBI Metagenomics service.
Scenario	The Metagenomics team wants to process some of the new datasets in their backlog taking advantage of cloud resources. Being the pipeline now fully ported to run in the cloud, a user can use the EBI Cloud Portal (https://cloud-portal.ebi.ac.uk) to instantiate batch systems in the cloud with the pipeline pre-installed in a matter of minutes, providing as inputs the number of nodes to be deployed and the ENA study to be processed. Once the processing

	is completed and the results transferred back to the EMBL-EBI data centre, the batch system is destroyed.
Success criteria	Success of this demonstrator will be achieved when it will be possible to deploy on-demand from the EBI Cloud Portal a batch system pre-installed with the Metagenomics pipeline onto different cloud resources, including the EGI-ELIXIR federated cloud.
User Documentation	*NA*
Technical Documentation	*NA*
Developer team	Technology and Science Integration Team (EMBL-EBI) EBI Metagenomics Team (EMBL-EBI)
License	*NA*
Source code	*NA*

4.2 Architecture

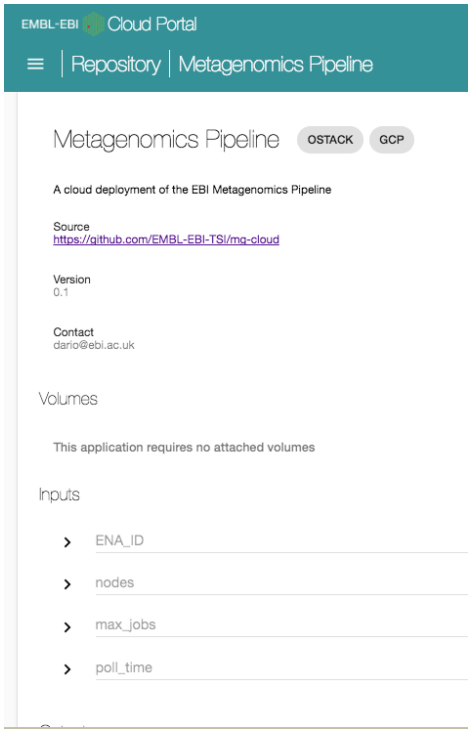
The EBI Metagenomics pipeline is composed by several tools dealing with sequences preparation, QC and analysis. A graphical representation of the steps is shown in the figure below (more details are available on the EBI Metagenomics website https://www.ebi.ac.uk/metagenomics/pipelines).

As this pipeline was initially designed to run in a batch system environment, it was decided to replicate a similar environment in the cloud to help reducing the initial porting efforts. OpenLava, based on a fork of LSF, was chosen as scheduler to be adopted in the cloud, as it offered an almost drop-in replacement for the on-site system. A single fat node served as NFS server to provide shared storage across the cluster. A graphical representation of the components is visible in the figure below:

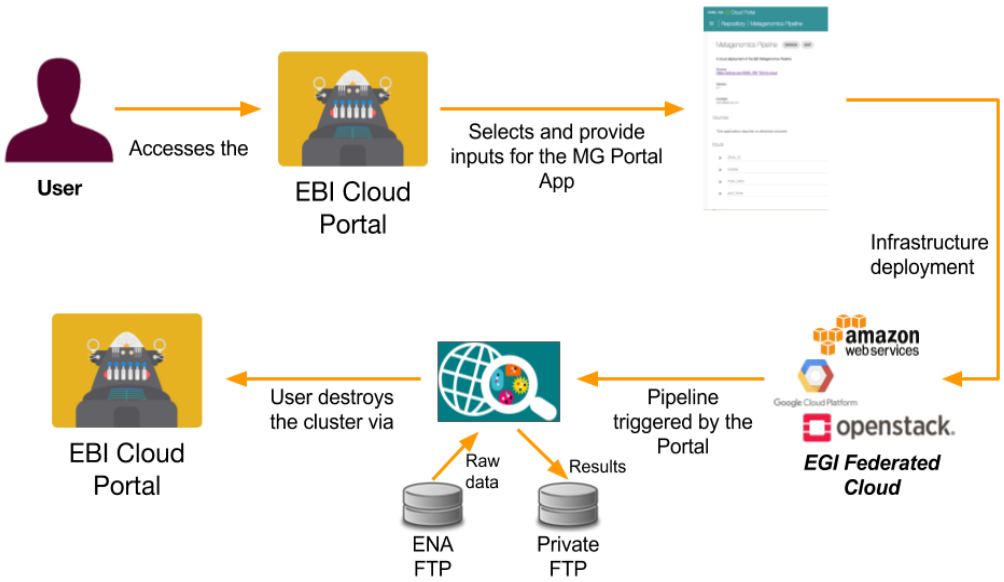


The provisioning of resources was orchestrated via Terraform (https://www.terraform.io/) while the configuration of the deployed systems was defined via Ansible (https://www.ansible.com/). This DevOps-based approach helped us in porting the infrastructure across four different clouds and five providers (AWS, GCP, Azure and two OpenStack providers: EMBL-EBI Embassy Cloud and UK Cloud) with minimal adjustments mainly linked to the network configuration. As for the compute infrastructure, also the pipeline install scripts were ported to Ansible code to provide a cross-cloud deployment. At the time of writing, the same codebase is in charge of deploying the pipeline on-site and in the cloud.

The next step the demonstrator took was to include the Infrastructure-as-Code developed in the first step and incorporate it into an Application for the EBI Cloud Portal, which is specifically designed to support inexperienced users in deploying complex stacks in cloud environments. A screenshot of the resulting Metagenomics Application in the EBI Cloud Portal is visible below.



The user is required to provide four easy inputs: the `ENA_ID` of the study that needs to be processed, the number of `nodes` to be provisioned, and parameters related to how many concurrent run can be processed at any given time (`max_jobs` and `poll_time`). After defining these parameters, the user can select the cloud where the pipeline should be deployed to (OpenStack and Google Cloud Platform in the example above) and the system will take care of provisioning the infrastructure and trigger the pipeline. In the course of this demonstrator, additional modules of the Metagenomics Pipeline were developed to deal with data ingestion (from a public FTP endpoint maintained by the European Nucleotide Archive) and with the transfer of the final results back to a private FTP server. Once the processing is complete, the user can destroy the provisioned cluster from the EBI Cloud Portal interface. A figure describing the complete end-to-end flow of the processing of an ENA dataset via the Portal app is available below.



4.3 Demonstration

4.3.1 Scenario

The Metagenomics Pipeline App has been successfully used within the EMBL-EBI Hybrid Cloud Working Group to provision batch systems of up to 80 nodes in three different providers (Google Compute Platform - GCP, EMBL-EBI Embassy Cloud, UK Cloud) and carry out scalability test of the pipeline with inputs up to 14GB. In earlier tests, deployments of this scale were also achieved in AWS. Initial scaling tests are currently underway in Azure. Thanks to the development work carried out by EGI to extend Terraform to support VM orchestration in EGI Federated Cloud Openstack providers, proof of concepts were also carried out in two Federated Cloud sites: EMBL-EBI Embassy Cloud and IN2P3-IRES.

The Metagenomics Pipeline App was also demonstrated multiple times in the context of the "ResOps: Delivering Bioinformatics Across Clouds" workshop organised by EMBL-EBI comprising internal and external participants.

4.3.2 Feedback

Development efforts for this demonstrator were required both from the EBI Metagenomics Team (EMBL-EBI) and the Technology and Science Integration Team (EMBL-EBI). We believe that this demonstrator has achieved all its objectives, and has strongly helped to define the way forward in porting on-site pipelines to cloud resources.

A strong feedback is also represented by the fact that, even if only at a proof-of-concept level, we managed to exploit two Federated Cloud sites via the Terraform plugin for OpenStack developed

by EGI as part of the activities of this CC. However, it must be noted that at the time of development the EGI Federated Cloud still relied on X509 certificates for authentication, which is a different AAI mechanism than the other clouds use and which the EBI Cloud Portal is compatible with. Because it has proven unfeasible to properly integrate the X.509-based AAI into the EBI Cloud Portal in the given timescale the user account management is not properly integrated, the setup is suitable for demonstration purposes only. The Federated Cloud, through the EGI CheckIn service is expected to offer non-certificate based access to federated cloud sites in early 2018. When this happens, the full AAI integration between the EBI Cloud Portal and the EGI-ELIXIR federated cloud could be implemented in any of the following ways:

Certificate-less:

1. EBI Cloud portal uses OpenID tokens to interact with sites. The portal will initiate a standard OpenID interaction with EGI CheckIn to obtain an access token as defined in the OpenID Connect specification. Access tokens identify uniquely each user ad allow access to the infrastructure. They can be refreshed using OpenID Connect standard mechanisms. This requires providers to fully support OpenID Connect natively. At the time of writing this is not yet in production, the monitoring integration is missing[12].

Certificate based:

2. EBI Cloud portal interacts with a single identity with the EGI-ELIXIR infrastructure. This identity is an X.509 robot certificate issued for the portal service by an IGTF CA. The robot is registered in the vo.elixir-org.eu VO, and the portal is using this robot for all the interactions with the federated clouds, transparently to the end users. On the cloud resources level all users will be mapped to a single 'portal robot identity', or – using the so called Per-User-Sub-Proxy mechanism[13] within the portal server – the distinction of users within the clouds is also possible.
3. EBI Cloud portal obtains temporal X.509 proxies for each user from the RCAuth portal. This is an OpenID based interaction that will ask for consent to the user the first time the service is accessed. Each user proxy corresponds to a single user and therefore the cloud infrastructure will see the individual users. Users do not need to see, and access the proxies.
4. And another option, that is not currently allowed by ELIXIR AAI policies[14], would be a mixture of above but without the need for a robot certificate: EBI Cloud Portal has a service account that can be used with OpenID Connect to interact with RCAuth to obtain a temporal proxy or with EGI CheckIn to directly interact with the sites. All users would be mapped to the portal account.

[12] https://github.com/ARGOeu/nagios-plugins-fedcloud/pull/16

[13] This is how user AAI is implemented in the EGI Applications On Demand Service: https://wiki.egi.eu/wiki/Applications_on_Demand_Service_-_architecture

[14] Because service accounts are not allowed in the ELIXIR AAI. (Only user accounts)

An initial evaluation of the development efforts required to support this type of authentication is foreseen in the next few months.

4.4 Future plans

The porting efforts around the Metagenomics pipeline after the end of this Competence Centre will be focused on three different fronts:

1. assess the feasibility of adopting the Federated Cloud as a backend for the Portal App
2. optimisation of the cloud resources usage
3. initial adoption of native cloud components in the pipeline (e.g. autoscaling, object storage)

The knowledge coming out of this work, along with the one that will be developed in the future activities, constitutes a solid ground of which other pipelines (both from EMBL-EBI and others) will take advantage in their own porting process. A key part of the EMBL-EBI mission is constituted by knowledge transfer and training, and both of these areas are already taking advantage of the results of this demonstrator thanks to the organisation of multiple dates of the “ResOps: Delivering Bioinformatics Across Clouds” workshop, in which attendees gain theoretical and hands-on experience on ResOps (DevOps for Research) and pipeline porting.

5 Insyght Comparative Genomics use case

5.1 Overview

Name	Insyght Comparative Genomics
Responsible person within the CC	Christophe Blanchet, CNRS IFB
URL	http://genome.jouy.inra.fr/Insyght/
Description	Insyght is a comparative genomic visualization tool providing users with a browser that helps navigate among abundant homologies, syntenies and genes annotations.
Value proposition	The Insyght comparative genomic visualization tool will provide scientists with an efficient and user-friendly tool to assist them in comparative genomics analyses (i.e. conservation of gene neighbourhood, presence/absence of orthologous genes, phylogenetic profiling, etc.) of large amounts of data. The cloud integration will provide life-scientist with their own instance of Insyght on devoted resource analyses, and to unload the public server.
Customer/user of the demonstrator	Any life-scientist who needs to perform bacterial comparative genomics analyses.
Scenario	Users can deploy their own instance of Insyght to: 1. visualise their own genomic data with the reference data provided by Insyght 2. compute their own genomic files and include them in the local Insyght reference data for further analysis
Success criteria	Users being able to • launch on-demand their own instance of the Insyght portal • visualise their own genomic data • launch in one-click the whole application (cluster of VMs for the computations and one VM for the visualization) • compute their own genomic files
User Documentation	https://migale.jouy.inra.fr/redmine/projects/insyght/documents
Technical Documentation	NA
Developer team	IFB-core and IFB-migale team, in collaboration with IPHC team.
License	This project is open source under the CeCILL-B licence.
Source code	https://migale.jouy.inra.fr/redmine/projects/insyght/repository

5.2 Architecture

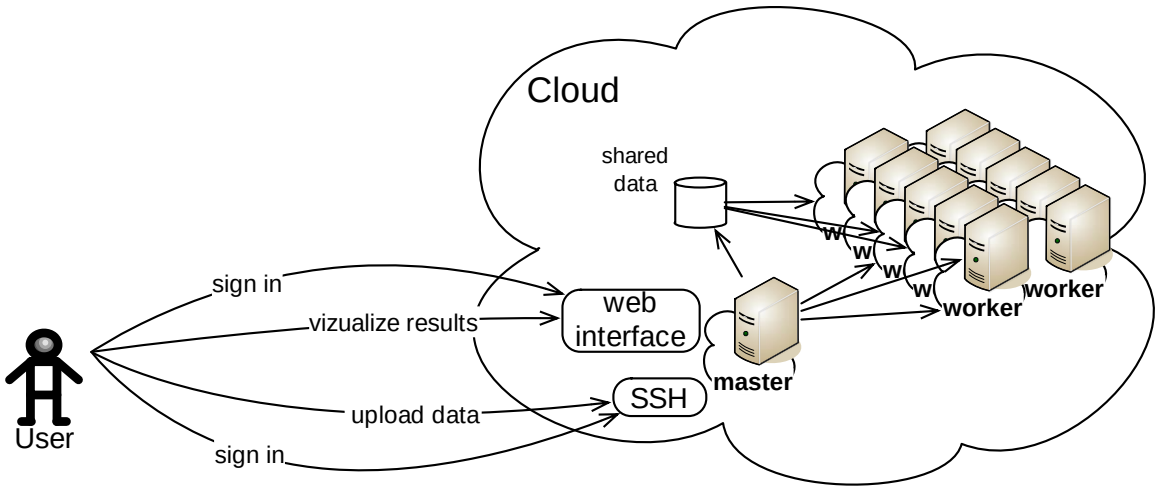
Insyght tightly integrates three complementary views: (i) a table for browsing among homologs, (ii) a comparator of orthologs' functional annotations and (iii) a genomic organization view that combines symbolic and proportional graphical paradigms to improve the legibility of genomic rearrangements and distinctive loci. Insyght benefits from an easy and smooth navigation between these 3 views and provides users with a powerful search mechanism.

Several virtual machines are required to deploy the whole application:

1. A database. Data is stored in a PostgreSQL relational database. This database contains three types of data: (i) primary data such as genomic annotations extracted from genome files (obtained from EMBL-EBI's Ensembl Bacteria), (ii) secondary data that results from the cross comparison of the proteomes using BLASTp, and (iii) tertiary data such as the synteny regions.
2. A pipeline. The database is populated by a pipeline of Perl scripts that (i) process the genome files, (ii) run the BLASTp jobs on a cluster, (iii) parse the results, and (iv) execute the program that determines the syntenies between all the pairs of bacterial proteomes
3. A Web interface.

Users need to:

- have access to the required appliances, published in the marketplace
- run a single virtual machine to visualize their genomic data
- deploy in one-click the complete environment for more complex cases: one virtual machine for the visualization, one for the database, and a cluster of virtual machines for the computations.



5.3 Demonstration

5.3.1 Scenario

The virtual appliance is available in the EGI AppDB as a live demonstrator for the deployment by users on the EGI Fed Cloud. We also did a demonstration internally to IFB users to evaluate the two scenarios:

1. Users can deploy their own instance of Insyght to visualise their own genomic data
2. Users can deploy in one-click several virtual machines to compute their own genomic files and include them in the local Insyght reference data for further analysis

For the first scenario, the application Insyght Comparative Genomics is available in the AppDB for the self-deployment by users. As the deployment requires advanced skills (need of a X.509 certificate[15], usage of the command line interface, and a lot of commands or scripting), it was done by the IFB-core team. Then, users were able to sign in the deployed Insyght instance, upload their data, perform their data analysis and visualise the results.

For the second scenario, the application requires several virtual machines: a web interface, a database, a computing cluster with a master and several computing nodes. The deployment of the whole application was done manually with scripts and the use of the jOCCI tool by the IFB-core team.

5.3.2 Feedback

The demonstration was partly successful for the first scenario. Indeed, users were able to visualise their own genomic data in Insyght, but they were not able to deploy themselves the application. The way of using the EGI FedCloud is not suitable, technically too demanding for most of the users of Insyght, who are mostly biologists, so not comfortable with the use of electronic certificates, scripts and OCCI. Unfortunately the simplified user interfaces for the Federated Cloud (Certificate-less access, AppDB VMOps Dahsboard) became available only by the end of the project.

5.4 Future plans

The foreseen improvement will be to evaluate the new AppDB VMOps Dashboard for the deployment of the whole application made of many virtual machines providing the Web portal, the database and the virtual computing cluster. We will also evaluate the possibility to use the OCCI connector on EGI clouds in conjunction with a cloud broker like SlipStream, as the Insyght

[15] Certificate-less access to the ELIXIR federated cloud testbed (i.e. with ELIXIR IDs) became available in 2017, only after this use case was implemented.

application is already available as a recipe for complex deployments in that framework. The criteria of success will be the usefulness of these features to our users.

6 PhenoMeNal project use case

6.1 Overview

Name	*PhenoMeNal*
Responsible person within the CC	Steven Newhouse, EMBL-EBI
URL	*http://phenomenal-h2020.eu/home/* *http://portal.phenomenal-h2020.eu/home*
Description	Deploy a Cloud Research Environment (CRE) for Metabolomics data analysis on private and public cloud providers.
Value proposition	Metabolomics, as many other Life Sciences, is experiencing a dramatic increase both in the size and the types of data they're required to analyse. The aim of the PhenoMeNal project is dual: to provide a robust Virtual Research Environment (VRE) able to process metabolomics data at scale using standardised tools and workflows; and to provide all the tools required to deploy such VRE across different clouds, both private and public. Achieving the second objective will unlock interesting scenarios where researchers will be able to deploy their own VRE to perform their research and other institutions, hospitals among them, to exploit standardised and robust workflows to foster adoption of cutting-edge metabolomics approaches into clinics.
Customer/user of the demonstrator	Both scientific researches looking for a platform able to process metabolomics data at scale taking advantage of several different cloud backends, and more clinical settings (e.g. hospitals) looking for a standardised and reliable approach to metabolomics analyses.
Scenario	The Phenomenal VRE and the Phenomenal VRE Gateway, relying on the EBI Cloud Portal APIs developed by the TSI team (https://www.ebi.ac.uk/about/people/steven-newhouse) for the cloud deployments, has seen its first beta release in February 2017. Since then, it has been reliably deployed into AWS, Google Cloud Platform (GCP) and several OpenStack clouds, to all using the

	Terraform orchestrator. The next release of the PhenoMeNal platform is due in August 2017, and will further push forward the capabilities of both the VRE and the deployment infrastructure.
Success criteria	Being able to deploy the VRE onto different cloud infrastructures, both private and public. EGI Federated Cloud was initially considered as a deployment target but later exploration proved that support for the orchestrator chosen by the project (Terraform) wasn't available. Terraform integration has in the meantime been added by EGI, but not in a timescale compatible with the deliverables of the Phenomenal project. The EGI-ELIXIR federated cloud will be considered again as a deployment target in the forthcoming releases of the platform.
User Documentation	https://github.com/phnmnl/phenomenal-h2020/wiki#phenomenal-users https://www.ebi.ac.uk/training/online/course/phenomenal-accessing-metabolomics-workflows-galaxy
Technical Documentation	https://github.com/phnmnl/phenomenal-h2020/wiki#phenomenal-developers
Developer team	Phenomenal Dev Team (EMBL-EBI and Uppsala University) Technology and Science Integration Team (EMBL-EBI)
License	Individual licenses, covering the different parts constituting the PhenoMeNal VRE and the deployment stack are available in their respective repositories (see below)
Source code	https://github.com/phnmnl

6.2 Architecture

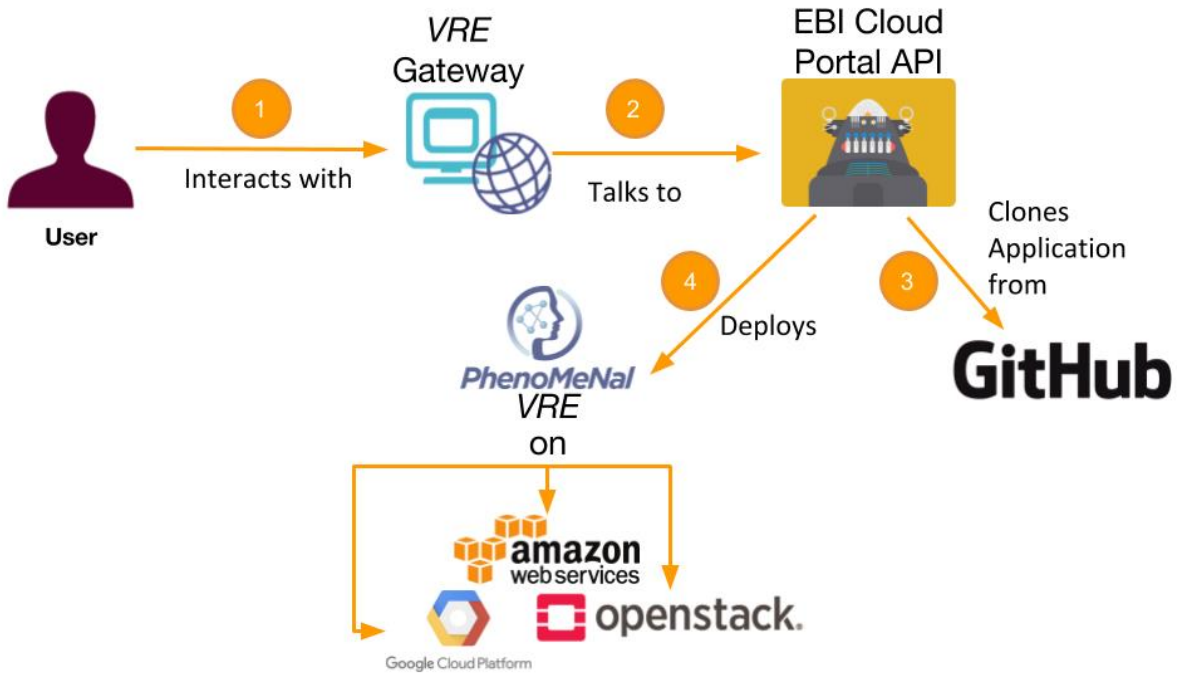
The architecture of Phenomenal, from a deployment point of view, is constituted by 3 parts:

- **the Phenomenal VRE Gateway.** This is an Angular web application that allows the user to define to which cloud the PhenoMeNal VRE should be deployed to, provide credentials for it and create an initial account on the deployed system. Once the user confirms that he or she wants to go ahead, the Gateway offloads the deployment to the EBI Cloud Portal APIs.
- **the EBI Cloud Portal APIs.** This is a REST API service developed by the Technology and Science Integration team at EMBL-EBI that allows users to add, share and manage the lifecycle of cloud applications and credentials. Applications consist of a git repository containing the Terraform (provisioning - https://www.terraform.io/) and Ansible

(configuration management - https://www.ansible.com/) to deploy an application in multiple clouds. In this context, the PhenoMeNal VRE is one of the many applications the EBI Cloud Portal is able to deploy. The PhenoMeNal VRE Gateway interacts with the EBI Cloud Portal to deploy the VRE, check the deployment status and manage its lifecycle.

- t**he Phenomenal VRE.** This is the Virtual Research Environment users will interact with to perform their analysis. At the time of writing, users can access it through Galaxy (https://usegalaxy.org/) and Jupyter Notebooks (http://jupyter.org/).

Once the deployed VRE is no longer required, users can remove it using the VRE Gateway that will request to the EBI Cloud Portal REST APIs to destroy it. A diagram clarifying the interactions between each of the aforementioned components and with the final user is provided below:



6.3 Demonstration

6.3.1 Scenario

The Phenomenal VRE deployment on OpenStack, AWS and GCP has now been presented and demoed at many different conferences and workshops. An extensive list of the Phenomenal outreach efforts can be found at the following URL: http://phenomenal-h2020.eu/home/about/presentations-posters/

The current version of the VRE can be autonomously deployed by all the users having access to one of the supported clouds from the PhenoMeNal Gateway portal (http://portal.phenomenal-h2020.eu - requires an ELIXIR account).

Because the EGI-ELIXIR cloud was lacking support for Terraform at the time of the first PhenoMeNal release, these demonstrations did not use the testbed which was setup by the CC. The Terraform support is now available[16], and future releases will consider also the EGI-ELIXIR federated cloud as a target platform. However, it should be noted that the same cloud AAI integration issue, which is described under the Marine Matagenomics (EMBL-EBI) section holds for PhenoMeNal too, given that both use cases interact with clouds through the EBI Cloud Portal.

6.3.2 Feedback

The demonstrator involved both PhenoMeNal developers (especially those based at EMBL-EBI) and the Technology and Science Integration team (EMBL-EBI). In general, the demonstrator has been very successful, as the PhenoMeNal VRE can now be reliably deployed through the EBI Cloud Portal APIs onto three different cloud backends.

While the original plan also comprised the EGI FedCloud as a possible source of cloud resources, this has not proven to be obtainable in the allocated timeframe. The main factor driving this was that the provisioning tool chosen by PhenoMeNal, while providing cutting-edge support for most of the cloud providers, lacked support for the FedCloud. Thanks also to the feedbacks provided by the Demonstrator within the CC, this has now been addressed by EGI creating a customised OpenStack provider exploiting the FedCloud authentication mechanism based on X509 certificates. However, due to other constraints in the development roadmap of PhenoMeNal the adoption of this backend hasn't yet been assessed.

Similarly, the Demonstrator was originally aiming to take advantage of AppDB to store both the VRE and all the containerised scientific tools that have been developed by the other partners of the project. However, as AppDB doesn't currently support shipping images to public cloud providers (i.e. AWS and GCP), PhenoMeNal eventually decided to store imaged directly in each cloud. Adoption of AppDB as registry for the containerised PhenoMeNal tools has also been temporarily suspended due to missing support for storing metadata to describe containers.

6.4 Future plans

Even after the end of this Demonstrator, PhenoMeNal will continue to assess, in accordance with its own development roadmap, the feasibility of adopting the FedCloud as a backend for the VRE. However, the fact that authentication still requires X509 certificates represents an issue that will require specific workaround to be implemented at the EBI Cloud Portal API level. At the time of writing, this is still in the inception phase.

In the same way, when AppDB will provide support to properly store and describe Containers, the project will assess the feasibility of adopting it as a backed to store its containerised tools. Nonetheless, as this would require a considerable change in the architecture currently adopted to deliver tools to the VREs, the roadmap for such an uptake cannot be defined *a priori*.

[16] https://wiki.egi.eu/wiki/ELIXIR_Virtual_Organisation#Terraform_orchestrator

7 JetStream interoperability use case

7.1 Overview

Name	JetStream interoperability
Responsible person within the CC	Robert Quick, Indiana University and Open Science Grid
URL	*NA*
Description	JetStream provides IaaS based computing resources to US researchers. It provides both API and GUI for managing cloud resources and library of VM images with the software ready to be used. The interoperability use case tries to assess what are the technical requirements to enable sharing these VM images between the JetStream and ELIXIR federated cloud VO so users in the US and Europe can easily access the same tools and run them on clouds they have access to.
Value proposition	Allow users to share scientific tools via Virtual Appliances on different computing infrastructures (JetStream ane EGI)
Customer/user of the demonstrator	ELIXIR VO Service providers, JetStream provider Developers of virtualised tools and applications
Scenario	ELIXIR VO is interested in offering a given VM image available in JetStream to European users and vice versa.
Success criteria	Images can be moved across infrastructures and started on them. Finding was that images from EGI/ELIXIR can be executed on JetStream, but images from JetStream are KVM-specific so not portable across all EGI providers. (some support KVM, others do not)
User Documentation	*NA*
Technical Documentation	*NA*
Developer team	Jeremy Fischer (Indiana University), Enol Fernández (EGI Foundation)
License	*NA*
Source code	*NA*

7.2 Architecture

The EGI AppDB provides a virtual marketplace for Virtual Appliances (VAs), which are virtual machine images designed to run on a virtualization platform, that provide a

software solution out-of-the-box, ready to be used with minimal or no set-up needed within the EGI Federated Cloud infrastructure. These images use OVA as preferred format, which is an open standard for packaging and distributing these virtual appliances. VOs can manage VA lists that are automatically distributed to the resource providers supporting the VO.

JetStream is based on the Atmosphere cloud-computing platform, and extended to support science and engineering research. The operational software environment is based on the OpenStack cloud operating system. Similarly to AppDB, JetStream provides a library of pre-configured virtual machines to support analysis and collaboration in a variety of disciplines. These VMs are tuned to specific scientific research types, taking the tools you typically need for a research area and tailoring them – along with libraries and other dependencies – so everything is ready to go once the VM is launched. The images are packaged as raw disk images.

7.3 Demonstration

7.3.1 Scenario

Scenario 1:

An image from JetStream is extracted and provided to EGI. The image is registered in EGI AppDB and providers are subscribed to the image so is downloaded and made available to users.

Scenario 2:

A VA from ELIXIR VO image list is downloaded in JetStream. The image is extracted from the OVA file and converted to raw format.

7.3.2 Feedback

Scenario 1:

The image is correctly downloaded and boots at most EGI providers that use KVM hypervisor since JetStream images are specifically crafted for it. For other hypervisors (e.g. Xen), the image fails to boot since the disk is partitioned in an incompatible format. Conversion to OVA would not fix these issues directly, the image needs to be built with a compatible disk layout beforehand.

Scenario 2:

The image boots without issues in JetStream.

7.4 Future plans

The demonstrator has shown that standards for creating and distributing images are needed in order to allow seamless sharing or images between e-infrastructures. EGI has adopted OVA

format, but this is not enough to ensure that hypervisor-specific images are ready to run on all providers. Automated conversion or clear guidelines to image builders should be provided.

The demonstrator mainly involved manual steps and administration privileges at providers, as future work a user driven process with selection of images in a simple GUI and automated triggering of the replication process should be provided.

8 Lessons learnt and recommended future work

The Elixir Competence Center project was just a starting point for the collaboration between the ELIXIR community and EGI. In general, the use cases showed how EGI resources can be utilized at the moment and the work should be continued to scale up and streamline the services for wider user community instead of just few pilot users.

We learnt the following lessons from the demonstrators, and suggest the following work in the future for EGI and ELIXIR. The recommendations will be proposed for implementation to the 'ELIXIR Compute Platform' group (group leaders are involved in this CC), and to the second phase of the ELIXIR Competence Centre, which is expected to run between 2018-2010 (3 years) within the EOSC-hub H2020 project:

1. From cBioPortal: Long running services that are relevant/specific to a small user base and/or can be hosted by a single operator do not need a federated cloud infrastructure, they can be served more economically by individually operated cloud sites. The entry barrier for a single cloud is typically lower than for a federated cloud, and because a single instance can serve the whole user base, the extra investment in the broadly available federated technology would not pay back.
2. From META-Pipe metagenomics: The critical elements of the federated ELIXIR cloud infrastructure are assembled, the federation is ready for broader update. Now it's time for the ELIXIR community to discuss the rules of engagement with cloud and application providers, and with the users. Some of the key questions to answer: What are the motivations and conditions for cloud providers to contribute to the federation? What are the rules for application providers and users to consume resources? What can be a sustainable business model for the federation?
3. From EMBL-EBI metagenomics and PhenoMeNal: The fact that federated cloud sites still rely on X509 certificates for authentication puts an extra barrier for adoption of the platform. This is especially true if the community specific cloud front end (e.g. EBI Cloud Portal) is already using commercial clouds with non-X509 AAI. The EGI Federated Cloud Task Force started working on certificate-less support in the federated cloud interfaces. This work should continue and conclude as soon as possible.
4. From PhenoMeNal: The EGI AppDB is currently tightly coupled with the EGI Federated Cloud, by which we mean it can act as a front-end only for clouds that use the EGI FedCloud technology, and is focussed on VM management. A possible and useful extension of AppDB

can be adding support for non-EGI clouds (e.g. starting with the most popular public clouds, such as Amazon, GCP, Azure), and for containers (enabling the deployment and management of containers in clouds).

5. From JetStream: EGI has adopted OVA format for Virtuam Machine Images, but this in itself is not enough to ensure that hypervisor-specific images are ready to run on all providers. Automated conversion or clear guidelines to image builders should be provided. In case of higher automation, a simple GUI to trigger the conversation of images, and replication onto JetStream and AppDB repositories will be also needed.
6. Generic observation – Need for an application hosting environment: Would be very beneficial if the EGI Federated Cloud had an 'application hosting' service in which developers of scientific applications could easily 'plug in' their applications and the hosting service would instantiate the application for end-users on demand (following some pre-defined rules and policies). EGI could operate a single, centrally managed application hosting service for the EGI Federated Cloud VOs, and offer assistance for RIs to establish their dedicated, similar hosting services that are connected to RI-specific cloud federations. The EBI Cloud Portal is more or less acting as such a system, serving EMBL-EBI and life science users across different commercial and private clouds.
7. Generic observation – Possible future demonstrator about data management: None of the demonstrators was really focussing on data management in the cloud, thus such aspects of the federated cloud model remained un-assessed (for example the object storages). Despite interaction with EUDAT, and integration with EUDAT services was initially foreseen by the CC, this aspect of the work was not realised. However, even without data-demonstrators it can be stated that life science applications in a federated cloud environment would benefit from (1) data movement services that applications can use to move data between sites of a federation, and in and out the cloud federation and (2) a replication service that could replicate datasets from a central place to every (or to selected) clouds of the cloud federation – similarly how the VMs are replicated from AppDB VM Marketplace to every federated cloud site.
8. Generic observation – Possible future demonstrator about graphical interfaces: The AppDB VMOps Dashboard – a simplified, graphical interface to manage VMs and block storage – became available for use only by the end of the EGI-Engage project but unfortunately it the demonstrators didn't had enough time to test it yet. It would be beneficial to adopt this as a GUI for future demonstrators and production users.
9. Generic observation – Taking the federated ELIXIR federated cloud to the next level with OLAs: Driven by the CC the project established the initial version of the ELIXIR Compute Platform in the form of the federated cloud testbed with 4 sites, described in Section 1. EMBL-EBI and EGI started preparing OLAs (Operational Level Agreement) that would define the condition for providers to participate in the infrastructure, and the level and quality of the services they offer, and they receive from EGI and EBI. Two types of OLAs are under

preparation. The work on these should continue and conclude, enabling ELIXIR to turn the testbed into a production service:

a. Resource Centre OLA: one OLA between each cloud provider and ELIXIR, defining the functional capabilities and support that the cloud providers expose to users.
b. Infrastructure Provider OLA: one document between EGI and ELIXIR, defining the services and support that EGI provides for ELIXIR so it can operate its cloud federation[17], and the responsibilities of ELIXIR when consuming these services.

10. Generic observation – Taking the federated ELIXIR federated cloud to the next level with more providers: During the EGI-Engage project the ELIXIR community setup its own service catalogue on the ELIXIR Website[18]. This catalogue includes also Compute Services (13 at the time of writing). The providers of cloud services from this list[19] should be approached and invited into the ELIXIR cloud federation. Their feedback and preferences for a federation model can help ELIXIR finalise the testbed into a production federation.

[17] Such as GOCDB, AppDB, Monitoring and Accounting systems, CheckIn
[18] https://www.elixir-europe.org/services
[19] CSC Cloud, de.NBI Cloud, French Academic Cloud (in development),