



EOOSC-hub

D7.2 First report on Thematic Service architecture and software integration

Lead Partner:	CINECA
Version:	V1
Status:	FINAL
Dissemination Level:	Public
Document Link:	https://documents.egi.eu/document/3412

Deliverable Abstract

The current deliverable describes the activities done by the various tasks of the work package 7 (Thematic Services: Integration, maintenance and Exploitation) during the first year of the project. Each task represents a Thematic Service, which is a single service or a set provided by a research community for its users. In particular, the deliverable focuses its attention on the architecture of those services and on their integration within the EOOSC-hub ecosystem.



COPYRIGHT NOTICE



This work by Parties of the EOSC-hub Consortium is licensed under a Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>). The EOSC-hub project is co-funded by the European Union Horizon 2020 programme under grant number 777536.

DELIVERY SLIP

<i>Date</i>	<i>Name</i>	<i>Partner/Activity</i>	<i>Date</i>
From:	Claudio Cacciari	INFN/WP7	
Moderated by:	Malgorzata Krakowian	EGI Foundation/WP1	
Reviewed by:	Marcin Plociennik	PNSC	27 Dec 2018
Approved by:	AMB		

DOCUMENT LOG

<i>Issue</i>	<i>Date</i>	<i>Comment</i>	<i>Author</i>
v.0.1	10/12/18		Claudio Cacciari, Dieter Van Uytvanck, Willem Elbers, Daniele Spiga, Tobias Weigel, Sandro Fiore, Paolo Mazzetti, Mattia Santoro, Anabela Oliveira, Alberto Azevedo, Mário David, Alexandre Bonvin, Antonio Rosato, Brian Jimenez Garcia, Marco Verlato, Christian Brieese, Michele Manunta, Marcin Gil, Simone Mantovani, Peter Baumann, Grega Milcinski, Fabrizio Pacini, Davor Davidovic
v.0.2	03/01/18	Reviewed by Marcin Plociennik	Claudio Cacciari
v.1	09/01/18		Claudio Cacciari

TERMINOLOGY

<https://wiki.eosc-hub.eu/display/EOSC/EOSC-hub+Glossary>

<i>Terminology/Acronym</i>	<i>Definition</i>
Thematic services	Scientific services (incl. data) that provide discipline-specific capabilities for researchers. (e.g. browsing and download data and apps, workflow development, execution, online analytics, result visualisation, sharing of result data, publications, applications)

Contents

1	Introduction.....	7
2	T7.1 CLARIN	8
2.1	Service description	8
2.2	Integration activities.....	9
2.3	Identified gaps.....	11
2.4	Future perspective	12
3	T7.2 DODAS.....	13
3.1	Service description	13
3.2	Integration activities.....	14
3.3	Identified gaps.....	18
3.4	Future perspective	18
4	T7.3 ECAS.....	20
4.1	Service description	20
4.2	Integration activities.....	21
4.3	Identified gaps.....	30
4.4	Future perspective	30
5	T7.4 GEOSS	32
5.1	Service description	32
5.2	Integration activities.....	32
5.3	Identified gaps.....	33
5.4	Future perspective	33
6	T7.5 OPENCoastS	34
6.1	Service description	34
6.2	Integration activities.....	35
6.3	Identified gaps.....	35
6.4	Future perspective	35
7	T7.6 WeNMR.....	36
7.1	Service description	36
7.2	Integration activities.....	37
7.3	Identified gaps.....	38
7.4	Future perspective	38

8	T7.7 EO Pillar.....	39
8.1	The Geohazards Thematic Exploitation Platform (GEP)	40
8.2	The EPOSAR Service.....	43
8.3	EODC JupyterHub for global Copernicus data.....	45
8.4	EODC Data Catalogue Service	46
8.5	Rasdaman EO Datacube	47
8.6	CloudFerro Data Collections Catalog	48
8.7	CloudFerro Infrastructure.....	50
8.8	CloudFerro Data Related Services - EO Finder	52
8.9	CloudFerro Data Related Services - EO Browser	52
8.10	Sentinel Hub	53
9	T7.8 DARIAH.....	56
9.1	Service description	56
9.2	<i>DARIAH Science Gateway</i>	56
9.3	<i>Invenio-based repository in the Cloud</i>	59
9.4	<i>DARIAH Repository</i>	61
10	T7.9 LifeWatch	65
11	Future plans	66
12	References	68

Executive summary

The current deliverable describes the activities done by the various tasks of the work package 7 (Thematic Services: Integration, maintenance and Exploitation) during the first year of the project. Each task represents a Thematic Service, which is, according to the above Terminology table, a single service or a set provided by a research community for its users. In particular, the deliverable focuses its attention on the architecture of those services and on their integration within the EOSC-hub ecosystem. Since not all the tasks started at and last the same time, differences in the advancement of the integration activities are expected. It is worth to note that, while the deliverable D7.1 (First Thematic Service software release) [1] included the details about the software releases of the components that implement the services, this deliverable reports the work done to develop those components and to enable those services.

The following Thematic Services are described:

- CLARIN: The Virtual Language Observatory (VLO) service.
- DODAS: Dynamic On Demand Analysis Service.
- ECAS: ENES Climate Analytics Service.
- GEOSS: Global earth Observation System of Systems Platform.
- OPENCoastS: On-demand oPERatioNal Coastal circulation forecast Services.
- WeNMR: Worldwide e-Infrastructure for NMR and structural biology.
- EO Pillar: a set of services related to Earth science.
- DARIAH: digital arts and humanities services.
- LifeWatch: environmental science services.

LifeWatch (T7.9) has no integration activity to report because of the task is on hold due to an administrative issue internal to the community. The Project Office is discussing possible solutions.

Aside from EO Pillar, none of the Thematic Services was expected to complete its integration within the first year; therefore, many activities are in progress, not finalized in production yet. This is why some of the integration results described here are not “visible” in the D7.1, which is a snapshot of the software in production.

These are the main integration aspects considered in this deliverable:

- **CLARIN:** usage statistics are harvested to provide metrics for the Virtual Access mechanism. The Virtual Language Observatory is published on the Marketplace and on the Service Catalogue.
- **DODAS:** it is currently integrated with several EOSC-hub service.
 - *Compute:* Infrastructure Manager, Paas Orchestrator.
 - *Security:* Identity and Access Management (IAM) and Token Translation service.
 - *Data:* OneData, CVMFS stratum 0 and 1.
 - It is published on the Marketplace and on the Service Catalogue.
 - A new dedicated instance for the Alpha Magnetic Spectrometer experiment has been deployed.

-
- CMS community collaboration.
 - Deploying of a new enabling facility at two INFN sites: [Cloud@CNAF](#) and [ReCaS@Bari](#).
 - **ECAS:**
 - *Security:* B2ACCESS, Indigo IAM and EGI Check-in.
 - *Data:* B2DROP, OneData.
 - *Accounting and monitoring.*
 - Integration with ESGF data infrastructure.
 - integration with the birdhouse open source framework used for data processing in Copernicus CDS
 - new ECASLab web portal
 - It is published on the Marketplace and on the Service Catalogue.
 - **GEOSS:**
 - *Compute:* EGI FedCloud. Porting of the discovery module of GEO DAB on EOSC-hub resources. Porting of the ECOPotential VLab on EOSC-hub resources.
 - It is published on the Marketplace and on the Service Catalogue.
 - **OPENCoastS:**
 - *Security:* EGI Check-in.
 - *Compute:* EGI FedCloud.
 - It is published on the Marketplace and on the Service Catalogue.
 - **WeNMR:**
 - *Compute:* DIRAC4EGI, EGI High-Throughput Compute.
 - *Security:* EGI Check-in.
 - *Data:* OneData, B2DROP
 - **EO Pillar:**
 - *Compute:* EGI FedCloud.
 - It is published on the Marketplace and on the Service Catalogue.
 - **DARIAH:**
 - *Security:* integration with the eduGAIN identity provider.
 - *Compute:* EGI FedCloud.
 - It is published on the Marketplace and on the Service Catalogue.

1 Introduction

In the following chapters the various Thematic Services provides information about their service architecture and level of integration with the EOSC-hub infrastructures. Each chapter starts with a short description of the services, an explanation about their architecture and main use cases and then there are the main paragraphs about the work done in the last year: the description of the integration activity done so far, the discovered gaps between the initial plan and the current status and the future steps.

2 T7.1 CLARIN

2.1 Service description

The Virtual Language Observatory (VLO) is a service provided by CLARIN ERIC offering uniform search and discovery functionality for language resources and tools. The metadata indexed is heterogeneous in terms of content and structure and sourced regularly, typically once or twice per week, from over forty CLARIN centres and other selected OAI-PMH endpoints. The VLO is openly accessible via the web to anyone and integrated with the Language Resource Switchboard and Virtual Collection Registry. A detailed description of the service is provided in D7.1 First Thematic Service software release, section T7.1 CLARIN [\[1\]](#).

2.1.1 Architecture

The VLO is a Java web application, using Apache Wicket and the Spring Framework for the server-side generated front-end. For the backend, there is an Apache Solr index in which the metadata is indexed, and a “harvester” command line application running periodically to update this index with harvested metadata from the providers. A detailed description of the architecture is provided in D7.1 First Thematic Service software release, section T7.1 CLARIN.

2.1.2 Main use cases

2.1.2.1 *Researchers*

The VLO is openly accessible via the web to anyone, though primarily aimed at scholars from the target disciplines, and allows the user to freely enter a search term and/or use a number of pre-defined *facets* to refine the search results, as shown in figure 1. This method of faceted browsing is as easy as using an online store and allows for quick filtering on basis of object language, nature of the resource, subject or organizations involved. Search results can be used to obtain a link to the associated resources and/or process the search result in the language resource switchboard.

Free text search

Faceted search

Fig. 1 – VLO faceted interface

2.1.2.2 Metadata providers and Repository Administrators

Repository administrators and other content ‘owners’ interested in integration of their metadata into the VLO can contact the support team. The main requirement for integration of records into the VLO is the availability of an OAI-PMH[2] endpoint and the use of a supported metadata schema (preferably, CMDI but Dublin Core, OLAC[3] and EDM[4] are also supported). A special status page is available to repository administrators to check the status of the latest harvest per endpoint.

2.1.2.3 Service and Resource Providers

The VLO service is provided by CLARIN ERIC as a central part of the CLARIN infrastructure. Providing this service includes development and management of the service, resulting in several instances of the VLO for different purposes. The production and development instances are hosted at a commercial resource provider and the beta instance is hosted at an academic resource provider. In an ongoing activity we are evaluating another academic resource provider for hosting a second production instance in order to enhance the service its availability and stability

2.2 Integration activities

One of the requirements for Virtual Access (VA) integration was collecting user feedback. This was not available in the VLO and was specifically developed for the EOSC-hub release. The chosen

solution is based on a commercial solution offered by Mopinion[5]. The user feedback form is shown after a configurable amount of time (currently set to 20 seconds) and shows icons representing satisfaction levels on a five step scale (not satisfied = 1 to very satisfied = 5). This can be submitted anonymously and if desired additional comments and/or contact details can be provided. This workflow is shown in figure 2.

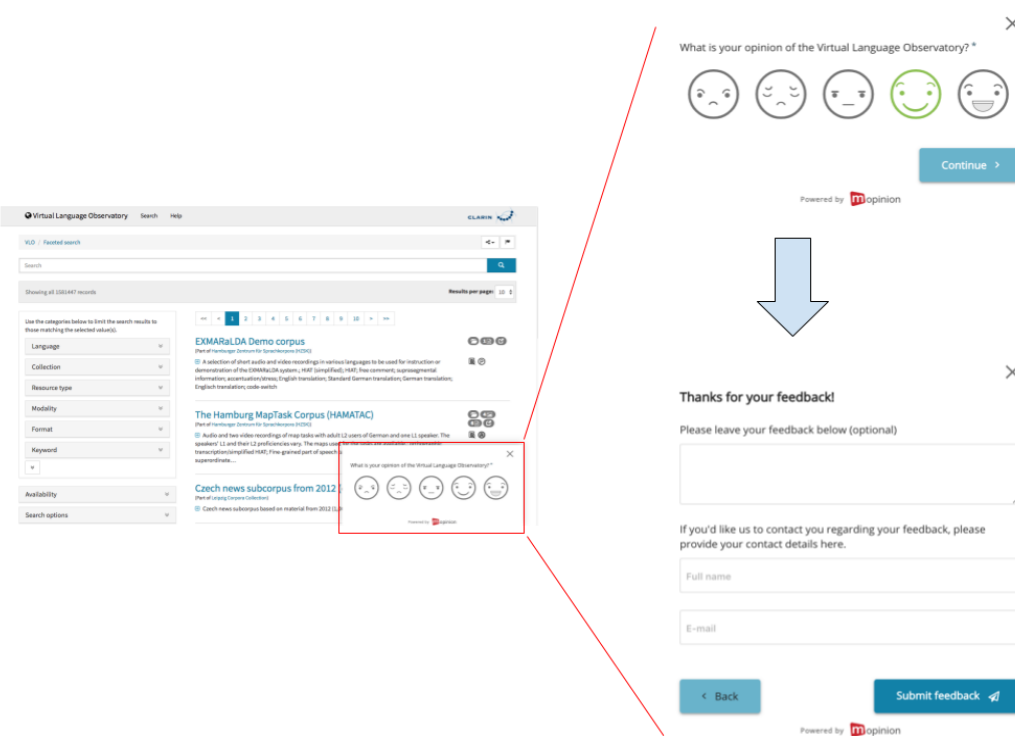


Fig. 2 – User feedback workflow

Our main goal when implementing the user feedback solution was to limit the required user interaction as much as possible while at the same time make sure we collect enough information to use the feedback to improve the product. Our first impression is that the rating value by itself might provide too little information but we understand that most users do not want to leave a comment and/or provide their contact details. We did configure the form in such a way that lower ratings lead to a more urgently phrased request for detailed feedback. Mopinion also offers features to include additional context collected from the user's session with the feedback rating. This can be useful to know for example on what kind of device the user was experiencing the service, or what search terms resulted in a specific piece of feedback.

Scores and other properties of all submitted feedback items can be inspected via the Mopinion platform. A dashboard allows us to quickly see the evolution of submitted satisfaction ratings in aggregated form (Fig. 3).

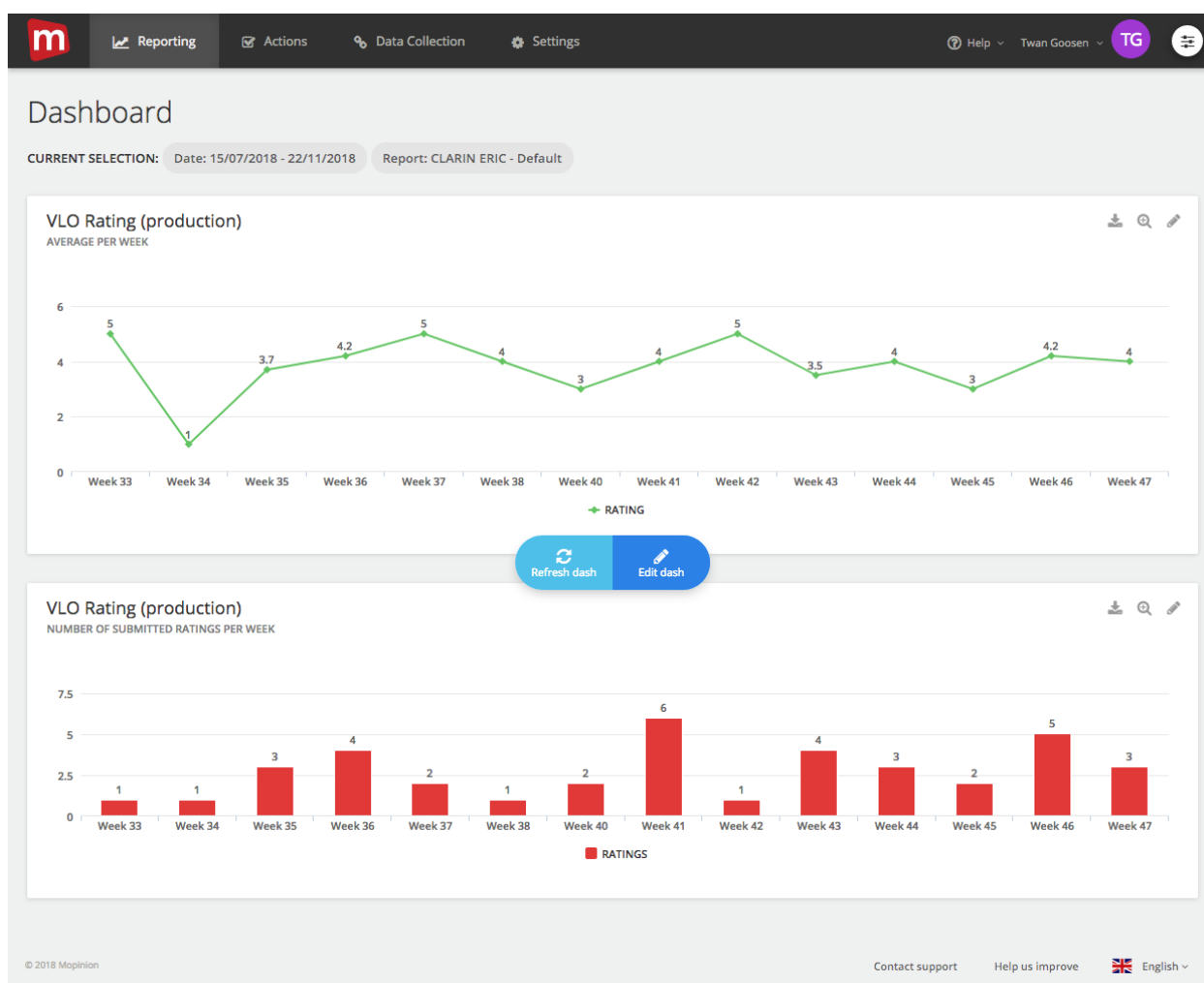


Fig. 3 – Mopinion dashboard

In addition to the user feedback plugin, a guided tour was added to better explain to VLO workflow to new users. The interactive ‘tour’ consists of a series of steps that demonstrates the most important functionalities of the VLO. This should allow the user to successfully carry out basic searches and use the results without having to consult the extended documentation. The tour can be triggered via a button on the entry page of the service, via its help page or by going to a dedicated URL.

All features, including the highlighted user feedback and guided tour, included in the EOSC-hub release are managed through GitHub issues associated with the “VLO 4.5 (EOSC-hub M6)” milestone. The VLO production instance is hosted on commercial infrastructure. We are looking to run a secondary, redundant, instance on academic infrastructure. For this reason we requested resources from EOSC-hub providers. Currently we are in the process of getting access to the resources provided by CESGA.

The VLO has been published to the EOSC-hub service catalogue and marketplace as well as the EOSC-portal marketplace.

2.3 Identified gaps

The main deviation from the original work plan is that due to the extra work required for the EOSC-portal launch event, we have agreed to push back the date for the first Virtual Collection

Registry release from M12 to M13. At this moment, we do not expect this to affect any other of the scheduled deadlines. The release of the VLO was also delayed with one month to M7, due to the availability of key personal. In the workplan we have defined integration with the EOSC-hub monitoring infrastructure. Internally we are using Icinga[6] and Statuscake[7] to monitor our services. We have to clarify what exactly is needed to integrate these systems with the EOSC-hub monitoring or that it is sufficient to periodically report based on the information collected from our internal systems.

2.4 Future perspective

Currently the VLO is the first out of three services that has been released under the EOSC-hub project. For this service the roadmap is focussed at improving the integration with EOSC-hub services. For the other two services the main focus is getting the first EOSC-hub release out, before working on improvements of the EOSC-hub integrations. The roadmap for the CLARIN thematic services is available in the CLARIN document archive, a second iteration is planned for M12/M13.

2.4.1 Virtual Language Observatory

For M24 a milestone has been defined for user community specific deployments. The current VLO implementation is focused on metadata coming from the CLARIN domain. The underlying CMDI schema is sufficiently flexible to support other domains as well, however the VLO must be updated to support this kind of flexibility. Integration with the EOSC-hub accounting and reporting services has already been finished under the VA framework. More advanced integration with the Virtual Collection Registry is also planned to be implemented for the M24 milestone. The aim is to make it easier for the end user to add resources found in search results to a new or existing virtual collection, and to feature this integration more prominently.

2.4.2 Virtual Collection Registry

The main focus for the virtual collection registry (VCR) is the first EOSC-hub release, scheduled for M13. For this release we are focussing on improvements of the integration with both the VLO and LRS. Another requirement for this release is the collection of user feedback. The same approach as the VLO will be followed; using Mopinion. The VCR is the only service of these three that requires authentication. Currently authentication is implemented using SAML via the CLARIN Service Provider Federation. We have to evaluate the best approach for integration with the EOSC-hub. This is not a technical but an organisational issue, because we need to understand the best way to comply with the security policies adopted by the CLARIN's community. After releasing this service in the EOSC-hub the focus shifts towards the integration with B2SHARE.

2.4.3 Language Resource Switchboard

The main focus for the Switchboard is the first EOSC-hub release, scheduled for M14. This release should already include the integration with B2DROP (roadmap LRS.3) as demonstrated at the EOSC Portal Launch event.

3 T7.2 DODAS

3.1 Service description

The Dynamic On Demand Analysis Services (DODAS) is an open-source Platform-as-a-Service tool, to deploy software applications over heterogeneous and hybrid clouds. DODAS instantiates on-demand container-based clusters through Apache Mesos. It offers a high level of abstraction to users, allowing exploiting any cloud infrastructure with almost zero effort since it requires a very limited knowledge of the underlying technologies.

DODAS completely automates the process of provisioning by creating, managing, and accessing a pool of heterogeneous computing and storage resources. As a consequence, it drastically reduces the learning curve as well as the operational cost of managing community-specific services running on distributed clouds.

Currently DODAS provides support to deploy:

- **HTCondor Batch System**, which in turn can be:
 - A complete and standalone HTCondor batch system (BatchSystem as a Service)
 - as such it includes all the HTCondor services: Schedd, Central Manager and executors (startds).
 - A HTCondor extension of an already existing Pool
 - this is about pre-configured HTCondor executors (startd) auto-join an existing HTCondor pool.
- **BigData Platform**
 - A Machine Learning as a Service. Currently this is about a Spark Framework, which can be coupled with a HDFS (either pre-existing or generated on demand) for data ingestion.

Both of the above configurations can be also deployed on several cloud infrastructures for and federated.

DODAS consists of multiple integrated components, currently integrating several services of the EOSC-hub portfolio. The technical details are described further in D7.1.

3.1.1 Architecture

DODAS has a highly modular architecture and its workflows are highly customizable. For this reason, it is very extensible, spanning from software dependencies up to the integration of external services, including also user tailored code management. Services composition is described through TOSCA language, software configuration and automation relies on Ansible. Authentication, Authorization and delegation is based on OpenID connect and IAM service is used. More details are contained in D7.1.

3.1.2 Main use cases

DODAS is a PaaS layer service aiming at providing a friendly solution for automating and simplifying the whole process of provisioning, creating, managing and accessing a pool of heterogeneous (possibly opportunistic) computing resources.

The main use cases are

- exploitation of opportunistic computing, intended as resources not necessarily or permanently dedicated to a specific experiment and/or activity;

- elastic extension of existing facilities, to absorb peaks of resource usage;
- generation of on-demand batch systems and/or Machine/Deep Learning facilities for data processing.

Regarding users groups, most relevant categories are described below:

- **User, researcher, who need to exploit opportunistic computing.**
An effective usage of compute and data resource, for medium to large data processing requires quite advanced IT skills. Core business of researchers is on data analysis and model definition and development rather than on site admin activities. Moreover, opportunistic resources, meant as those computing resources not necessarily or permanently dedicated to a specific experiment and/or activity as such do not provide a dedicated support for communities. DODAS, by provide automation, abstraction and self-healing capabilities represents an ideal solution.
- **Site manager who need to elastically extend existing facilities.**
Sites and facility already support communities and experiments for any related computing activity. Those might need to absorb peaks of resource usage. Alternatively they could provide specific setup for specific workflow requirements, a very simple example is a very high memory based setup, or high I/O jobs. In such case, DODAS represent an easy to use solution to deal with this kind of situations. Similarly, a site could need to accommodate a request for a mission specific facility where to give a priority to some activity for e.g. a scientific discovery. Again by far DODAS is a service perfectly fitting with this use case.
- **Researchers who need a data analytics Infrastructure as a Service.**
New approach to data analysis is more and more required to user communities. A frequent scenario is that users need to have access to facility where to develop, tests and training models. Not only also, but some facility where to perform inference as well. Some advanced community has already small facility for this type of activity, however need extension into the distributed on demand infrastructure suitable for large scale collaborations. DODAS provide the technical answers for such cases providing a platform for facilitating user access and supporting the testing and development of new Machine Learning applications

Finally, for what regard resource providers, anyone offering cloud interface can integrated DODAS as platform as a service layer for data and compute exploitation.

3.2 Integration activities

A dedicated repo[24] has been created for DODAS on GitHub. It is continuously evolving in order to include all the integration and new developments done. It includes and will include also training material and so on.

3.2.1 DODAS and integration with Data Management

Regarding the data management there are two main technical requirements that DODAS have got from communities.

- To provide solutions to implement transparent data access
- To provide solution for data ingestion and temporary store of input/output data.

The first one is needed in order to reduce possible source of inefficiency coming from the latency during read operations of data hosted outside the cloud provider where CPU is. To address this issue, a possible solution is to bring up a set of services acting as a proxy between the computing resources and the remote storage, and in addition to cache the served data.

As foreseen in the original workplan where we foresaw multiple solutions for the data management and data caching, based on distinct technologies as per requirements analysis, we collected requirements from CMS and AMS communities

After analyzing the CMS and AMS communities requirements two possible solution have been developed and integrated: One based on XrootD technology and one based on Onedata service.

Onedata is part of the portfolio of EOSC hub and a lot of integration and testing has been done during the first year of the project. Several tests have been carried on with different release candidate of Onedata and at different scale.

The very last version tested is rc11 and with this we concluded that performances within local setup is acceptable, while for the remote data access there is still need to improve the CPU efficiency. Stability of the system is very good with the latest test. It is important to highlight that most of the testing, mostly the scale testing of Onedata, has been done by DODAS over resources provided by Helix Nebula Science Cloud project[8].

The second solution relies on XCache and is based on XRootD, a software largely adopted in the High Energy Physics communities. As example, CMS computing model is largely based on “Anydata, Anytime, Anywhere” (AAA) system for data access. The choice of XCache integration has been also done after consulting the [XDC](#) project where we found a Work Package (WP4) mostly dedicated to these problematics.

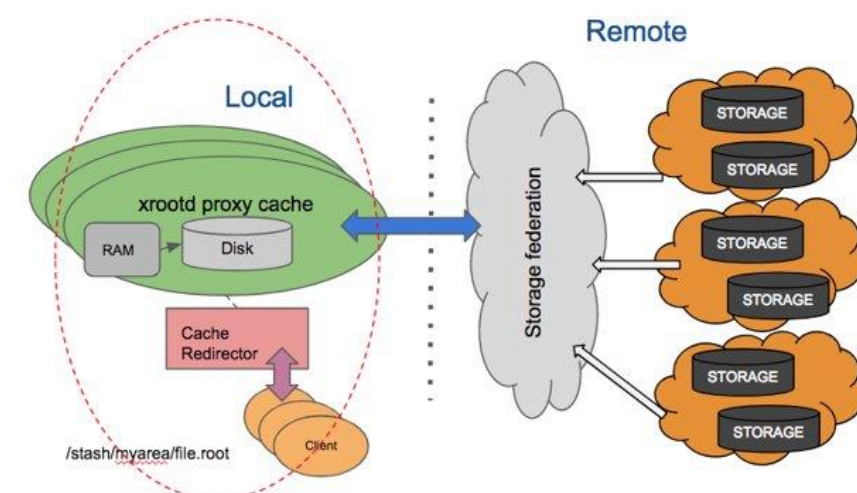


Fig. 4 – local data caching mechanism

Figure above shows a schema of local data caching mechanism with respect remote data source. In addition, in this case, a lot of testing has been done and results are really promising. A detailed study on simulation for a higher scale of remote data processing is ongoing.

Both Onedata and XCache testing has been done with CMS workflows used as benchmark. Regarding Onedata the integration has been carried on starting from the existing Ansible Roles for the services setup. Client specific roles have been extended and integrated at TOSCA level in to the DODAS recipes.

Concerning XCache, a complete new development for the Ansible Roles has been done. TOSCA integration is still ongoing. Both Kubernetes and Mesos plus Marathon have been validated and are both supported.

The XCache solution has been adopted by AMS experiment, which also tested and adopted xrootd based solution for the temporary data management of input and output data. The plan is to evaluate Onedata performances with AMS based workflows.

3.2.2 CVMFS Server Integration

In order to support AMS data analysis workflow a new service has been integrated within the DODAS automated configuration. CVMFS is a product of the EOSC-hub portfolio and DODAS integration has been done adapting Ansible roles previously developed. A new TOSCA template has been developed to provide a full automation. Moreover, the AMS TOSCA template has been extended to support a dynamic configuration of the client based on user define stratum0 parameters and keys.

3.2.3 AAI: IAM and ESACO Service integration

DODAS has a deep integration with IAM. Not only as solution for user authentication and authorization but also for delegation (services acting on behalf of users) and services authentication (in a distributed environment). In addition to IAM DODAS integrates WaTTS and, during this year of activity, it also integrated ESACO.

ESACO is a daemon that has the responsibility of checking validity and signatures of OAuth tokens for registered trusted OAuth authorization servers. The daemon exposes an OAuth token introspection endpoint compliant with RFC 7662[9] that can be used by authenticated clients to inspect tokens. The daemon can only introspect JWT access tokens that contain the iss claim. ESACO is registered as a client at one (or more) trusted OAuth authorization servers, and is used by client applications as a gateway for token validation and introspection. ESACO performs local JWT validation checks and leverages the introspection endpoints at trusted AS to inspect a submitted token. The result of a token introspection is cached, if caching is enabled. More info can be found [here](#).

ESACO has a key to solve DODAS problem due to the need to use multiple Openstack requiring federation also with DODAS. Currently the Apache2 openidc[10] doesn't allow to integrate multiple openid-connect endpoints thus adopting ESACO client registered at (one or more) trusted OAuth AS, which acts as a gateway for token validation and introspection allow to easily overcome the problem. ESACO has been installed on the Cloud providers supporting DODAS Thematic Service (Namely Cloud at CNAF and Cloud at ReCaS, Bari).

3.2.4 Scientific Communities integration: Alpha Magnetic Spectrometer experiment

AMS doesn't operate a central services to manage workflows and thus it relies batch systems and technology available on the sites where resources are made available to the collaboration. That said what AMS requires, at first stage, is to access a batch system in any Cloud to process remote data, and thus by adopting DODAS AMS will have an enabler for a Batch System as a Service.

To accomplish with this requirement a set of Ansible roles and TOSCA templates have been developed. TOSCA and Ansible development have been done extending the basic DODAS implementation of HTCondor batch system.

In the case of AMS, the HTCondor structure is made of a Central Manager, a submitter node and several worker nodes. The Central Manager is stateless and has the task to coordinate the jobs that the users want to do, so it connects the submitter node to the worker nodes. The submitter node is special because it has also the environment where each user can prepare own jobs and, of course, it is similar to the environment present in the worker nodes.

A second approach has been provided to the community by adding the possibility for remote job submission.

All the nodes are managed as Docker instances: there is a core image that is extended specifically for the HTCondor cluster. Both user compiled software and centrally managed libraries are distributed through CVMFS. Data ingestion is managed through XrootD, similarly to the CMS use case.

A relevant fraction of the effort has been spent to allow the AMS software to run on docker container and to validate it.

The overall implementation have been tested and validated by AMS users, two months long scale test was carried on running DODAS over Helix Nebula provided resources. More than 500k jobs have been executed producing more than 40TB of data.

3.2.5 CMS Data Preservation and Open Access workflow integration

One of the integration activities was done with the Data preservation and open access (DPOA) team having the objective of using DODAS generated batch system to execute CMS Open Data 2010 VM Monte Carlo generation example.

The DOPA group is in charge of maintaining and developing software so that data from any run period can be analyzed by a CMS member any time in the future. It also provides tools, like virtual machines and analysis examples, so that people from outside the collaboration can perform analysis on open legacy data. This is part of the open access project and it has currently released data from 2010 to 2012. Objective of this integration activity consisted in allow to use DODAS to create an HTCondor batch system capable of running regular CMS analysis jobs over legacy and open data., and example on how to perform MC simulations using Pythia in CMS Open Data 2010 virtual machine.

To allow this integration succeeding, DODAS Thematic Service facilities have been used. As a result of this activity, examples how to run Monte Carlo simulations are added to the OpenData portal. It is now possible to submit batch jobs through crab to analyse legacy data and the platform DODAS was successfully tested and used to launch a working batch system. Some things, remaining to be done concerning this subject, are: to properly document the MC examples on the Open Data site, and to create a platform template with DODAS to allow users outside CMS to create their own batch systems.

3.2.6 New Instance of DODAS provided by the project

Begin the mission of DODAS Thematic Service to provide support for the integration of new use cases and workflows required by possibly any scientific communities seeking to exploit Cloud resources to accomplish research activities. As foresaw in the workplan DODAS team provides not only the guidance for integrating the user community workflows, but also offers the possibility to test DODAS on a freely accessible cloud. This is the rationale behind the installation of a completely new instance of DODAS PaaS core service, the so called Enabling Facility.

In addition, a key point, in order to better support the development of real distributed solution the Data and compute resources of this new DODAS installation are offered by two distinct providers at INFN: [Cloud@CNAF](#) and [ReCaS@Bari](#). Both of them are based on Openstack middleware.

The Enabling Facility is freely accessible through DODAS PaaS core services, upon successful registration and authentication on <https://dodas-iam.cloud.cnaf.infn.it/>.

3.2.7 Integration of DODAS with third party Providers

DODAS has been exploited to create a CMS Grid Tier-3 site using resources hosted at Imperial College London (ICL), UK. The primary objective of this activity has been to perform a functional test of DODAS to run requirement-specific workflows. For the functional test, a small amount of quota has been reserved on ICL public OpenStack namely: 30 instances, 140 virtual CPUs, and 300 GB of RAM, with 1 TB of disk volume. The new Tier3 site has been called T3_UK_Opportunistic_dodas, and it relies on the Tier2 running at ILC (T2_UK_London_IC) as target storage to copy the produced output data.

The evolution of this activity is to use the same approach in order for the UK-ICL to exploit AWS cloud provider.

3.2.8 DODAS documentation

To better explaining DODAS, its components and user guides, we created a technical documentation hosted at <https://pages.github.com> and built from GitHub repository.

Links:

- <https://dodas-ts.github.io/dodas-doc/>

3.2.9 DODAS on the EOSC-hub marketplace

DODAS was also integrated with the EOSC-hub marketplace. Descriptions were added that indicate very concisely to users the value of the service, how to access the two instances and where to find further information. The marketplace entry for DODAS provides all necessary details for end users.

3.3 Identified gaps

There is one area where gaps in integration with EOSC-hub services portfolio had an effect on DODAS integration workplan. This is the Accounting component, which was originally foreseen as one of the first activity to carry on. Related to this two effects combined together, the lack of accounting training from the EOSC-hub project as required, and unforeseen contingencies such as communities requirements for integration and request of support caused some delay in DODAS accounting integration compared to the original plan. However, the issue is by no means so severe that they would ultimately stop delivery of a fully integrated DODAS, because all the data management activities have been anticipated so to balance a bit the delay. As a matter of fact this demonstrates that supporting new communities and new use cases, despite is the core business of the Thematic Services of EOSC-hub, requires a huge amount of effort not only to technically integrate services or component but also to understand requirement for a best mapping with features as well as to provide user support.

3.4 Future perspective

The integration and training activities of DODAS will continue during the next year. DODAS is already operational service since M4 however following an agile approach new features are made available when commissioned and gathering feedback allow to better evolve the service continuously. Integration with EOSC-hub service will continue and will surely focus on extending the AAI integration by meaning two things:

- Integration of IAM service as solution for a dynamic user mapping management of the HTCondor batch on demand.
- Federation, through IAM, with EOSC-hub AuthN/Z services.

Integration of the Accounting toolkit: this will require training for DODAS team first and then a design phase to be sure that all the identified cases are covered. Another priority is to further improve and integrate the MLaaS (Machine Learning as a Service) features of DODAS. New uses cases based on this flavour of the Thematic Service are approaching. As a matter of fact, this implies that we will need to integrate new features. In this respect we consider that EOSC-hub might provide benefits because we might find solutions already available. Dynamic extension of

cluster through clues: this is a rather important aspect that we actually did not mention in the original plan. In fact, we need it and thus we will integrate it in the next year. This integration relates to TOSCA, PaaS Orchestrator and Mesos configuration.

A huge amount of effort will be dedicated to support the communities using DODAS.

- Continuous support is required for user and communities as mentioned above. The amount of effort required for this task cannot underestimate. This is a lesson learnt after the first year. Support is meant not to be only at technical level but also in term of training, design and strategic decisions needed while porting static models to a geo distributed and federated architecture
- There are on-going requests from user to integrate commercial providers. This will require EOSC-hub IM and PaaS orchestrator further integration.

4 T7.3 ECAS

4.1 Service description

The ENES Climate Analytics Service (ECAS) enables scientific end-users to perform data analysis experiments on large volumes of multidimensional data (e.g. NetCDF data format), by exploiting a PID-enabled, server-side, and parallel approach. It aims at providing a paradigm shift for the ENES community with a strong focus on data intensive analysis, provenance management, and server-side approaches as opposed to the current ones mostly client-based, sequential and with limited/missing end-to-end analytics workflow/provenance capabilities.

ECAS consists of multiple integrated components, centred on the Ophidia Big Data Analytics framework, which is at the current state integrated with B2DROP, ESGF, IAM, JupyterHub, and the ECAS-Lab web portal. The technical details are described further in D7.1.

4.1.1 Architecture

The Ophidia framework provides the key scalable, parallel analytics capabilities. To enable easy data provisioning, it is integrated with B2DROP and ESGF. Integration with JupyterHub and IAM enables users to easily access the service and re-use existing Python scripts, modify them or create entirely new ones in a fully server-side approach. When scripts are executed using the Ophidia analytics framework, result data or other artefacts such as images as well as the scripts can be shared directly via B2DROP. More details are contained in D7.1.

4.1.2 Main use cases

ECAS provides server-side data analytics capabilities, which may be relevant for a wide range of users from many disciplines. The following is an estimation of the most relevant user groups for the upcoming dissemination and, in particular, training activities that ECAS will provide.

4.1.2.1 *Users with no directly available computing or data analysis resources*

By far not every researcher from the climate data domain, but also beyond, has local access to well-resourced computing and data facilities, which also provide easy and performant access to desired data sources. ECAS enables these users to work with data processing and analysis scripts independent of locally available computing resources.

4.1.2.2 *Users from the climate data downstream communities*

ECAS can be of value to those users from ‘downstream communities’ of climate data such as climate impacts researchers or for regional modelling, who are to some extent familiar with the design of climate data but are not as familiar with the usual access paths for such data e.g. via ESGF. For these users, ECAS offers easier access and the ability to work with multiple data sources independent from the researchers’ location and locally available support.

4.1.2.3 *Climate data users interested in performing cross-model/large-scale ensemble analysis*

A particular strength of the Ophidia framework and the approach underlying it is that one may devise a data analytics workflow and then execute it on large numbers of datasets many times without having to specify the workflow again each time. A particularly good example for such scenarios is the analysis of climate model ensemble data or cross-model analysis. This can be

another important benefit for climate impact researchers. Examples from other domains are targeted as well, e.g. from EMSO (European Multidisciplinary Seafloor and water-column Observatory), LBT (Large Binocular Telescope) and LifeWatch.

4.1.2.4 Users interested in prototyping applications with datacube concept, expecting speed-up compared to conventional approaches

The datacube concept underlying the Ophidia framework may have benefits for users performing massive array operations within their scientific analyses. It may be possible to provide a speed-up compared to conventional approaches using Ophidia and the fully server-side ECAS framework, which also exploits locality of data and processing.

4.2 Integration activities

Integration since the beginning of the project has focused on enabling a first exemplary user workflow, spanning from data input over login and processing to data output sharing. The workflow is provided at the two instances hosted at CMCC and DKRZ, which differ in their setup and available data and computing resources. Besides bug fixing, software and service maintenance, several integration and adaptation activities, as well as those related to documentation and training, were also performed by the ECAS team during the first year, reported as follows. A GitHub project (<https://github.com/ECAS-Lab>) has been set up to store all the developments, training material and integration performed in relation to the ECASLab. Extensions and adaptations to other tools have been published, instead, on the related Github repositories (i.e. for the Ophidia framework: <https://github.com/OphidiaBigData>).

Two instances of ECASLab have been deployed (at CMCC and DKRZ):

- <https://ecaslabor.cmcc.it/web/home.html>
- <https://ecaslabor.dkrz.de/home.html>

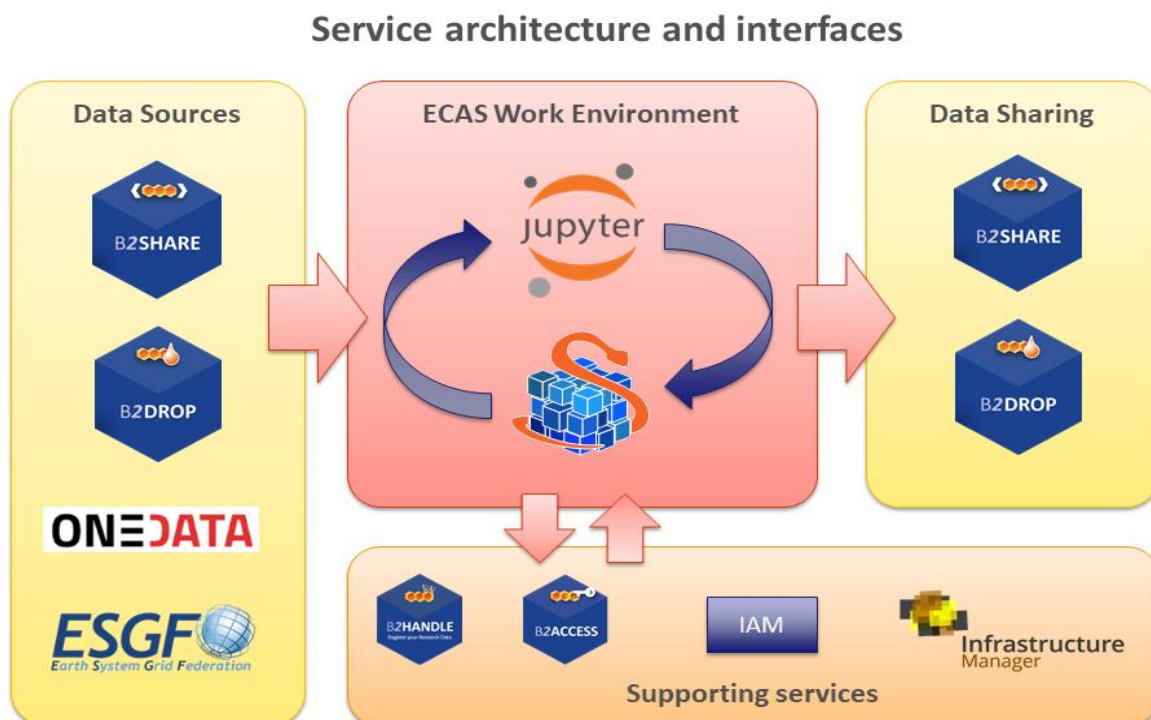


Fig. 5 – Service architecture and interfaces

4.2.1 Integration pathway for the first year of EOSC-hub

The integration activities for building the full ECAS service have followed a specific pattern for the first project year. The goal was to enable one full path following the user workflow, focusing on activating at least one component from each step in the workflow. The diagram shown above (Fig. 5) displays the desired final architecture for the fully integrated ECAS service. In the following sections, we explain the path that was enabled so far.

To enable input of data into ECAS, from the group of data sources ESGF was integrated. The work environment was set up with Ophidia and Jupyter, both of which were essential to deliver a complete workflow. For data sharing, B2DROP was integrated, and as such, it also became available as a first alternative data source. Concerning supporting services, integration focused on IAM as an AAI solution.

These essential integration activities have been complemented by additional actions on software containerization of the ECAS environment using Docker, providing good workflows for development and operations, integration with the EOSC-hub accounting and monitoring, documentation and extensive training using Jupyter notebooks in particular.

The first year activities provide a single full path through the ECAS environment. The integration activities of ECAS in the second project year will then complement this by adding more alternatives for data input and sharing, and integrating the other supporting services to enhance the workflow further.

4.2.2 Integration of Ophidia Framework and Jupyter with B2DROP

In order to easily take advantage of the EUDAT B2DROP service within ECASLab, some preliminary extensions to the Ophidia framework have been made. To this end, a new operator called OPH_B2DROP has been implemented to upload files from the ECASLab user folders to the B2DROP user account. Regarding authentication, the current version still requires users to provide credentials for the B2DROP service by means of a *netrc* file. Besides specifying the path of the netrc file, the operator allows the definition of the source (on ECASLab file system) and destination (on B2DROP) paths of the files to be uploaded to the service. An example of usage of this operator is the following:

```
oph_b2drop src_path=test.nc;
```

To make the operator more usable, both the destination path and the authentication file can be omitted, thus relying on default values. The complete documentation for this new operator is available on the Ophidia framework documentation website:

http://ophidia.cmcc.it/documentation/users/operators/OPH_B2DROP.html

To support an easier exploitation of the operator from JupyterHub, the PyOphidia module has also been extended to include the method for the B2DROP operator. Additionally, in order to target user experience, an additional method has been added to PyOphidia for an easy exporting of a datacube as a NetCDF file directly to the user B2DROP account.

An example of the method used to export the content of myCube datacube as a NetCDF file is the following:

```
myCube.to_b2drop()
```

B2DROP is also integrated at the Jupyter notebook dashboard. ECAS users can decide whether to share their files with other users or simply keep them in their private B2DROP repository. In each user workspace in the jupyter notebook the following directory are mounted:

- **B2drop-shared:** this directory is shared between all ECAS users and limited to 20GB. Therefore, this not recommended using it for storing big datasets. No further authentication steps are required to view the content of this directory (<https://b2drop.eudat.eu/s/gDyJjMeJ2Xiapwi>).
- **B2drop-private:** this directory can be created using B2DROP credentials and the content is only visible for the owner. The following steps describes how to mount the b2drop-private directory in the Jupyter notebook:
 - a. Generate app username and password from your B2DROP account.
 - b. Log in to ECASLab and start Jupyter notebook.
 - c. Go to **/conf** directory and put the credentials from (a) in the **env** file.
 - d. Open the **mount-your-b2drop.ipynb** and run it.

There are different ways to move files to the b2drop-* directories:

- A new extension is implemented for sharing notebooks using a *Share* button in the Menu (only for notebooks and b2drop-shared).
- Using the “Move” button and specify the B2DROP target (for both).
- Creating files directly in **b2drop-shared** or **b2drop-private**. These directories are synced with the B2DROP main repository.

4.2.3 Integration with ESGF

The Earth System Grid Federation (ESGF) is a global federated data infrastructure used to make climate datasets available for the wider climate community. Datasets from CMIP projects (5 and 6) are available (read-only) for ECAS users. Currently, the ESGF@DKRZ data pool is mounted in the compute node and accessible to the Ophidia framework. At the moment, data is not directly visible in the Jupyter notebook but technically possible upon request. Furthermore, we are working on enabling the import of NetCDF datasets via OpenDAP server.

4.2.4 Integration of Ophidia Framework with JupyterHub

In order to seamlessly exploit Ophidia through JupyterHub, several integration and adaptation activities (including system configuration and automated scripts definition) have been performed.

In particular users generally have to provide information regarding the endpoint (IP address and port) and the login (username/password or token) to connect to an Ophidia server. In the context of the ECASLab, the authentication to the system is already handled by the JupyterHub login interface, so that users do not need to set again the login and endpoint information once again; therefore, some environmental variables have been used to set up the ip/port of the Ophidia server (globally at the level of the ECASLab instance) and the username/password (locally at the level of the user account).

JupyterHub, by default, does not load the environmental variables in the Jupyter Notebooks. To this end, the JupyterHub instance has been configured to load the variables related to Ophidia from the environment before spawning an instance of JupyterHub for the user. In particular, a custom spawner has been defined; the following configuration has been used:

```
c.Spawner.cmd = ['/usr/sbin/jupyterhub-bash.sh']
```

The script jupyterhub-bash.sh is the following:

```
#!/bin/bash
source $HOME/.bash_vars
exec jupyterhub-singleuser $@
```

As it can be seen, the script loads the variables from the `.bash_vars` file available in the user's home and then spawns a new instance of JupyterHub. The `.bash_vars` file is automatically created by an automated script upon user account creation.

The Ophidia terminal can automatically read this information directly from the environment in order to connect to the proper Ophidia server. To make this behaviour also available in Python-based Jupyter Notebooks, the Ophidia python bindings (PyOphidia) have been extended to get the connection information directly from the environmental variables. More in detail, the connection method has been extended with a new argument, so that the user can use the more convenient command:

```
ophclient=client.Client(read_env=True)
```

instead of the traditional format:

```
ophclient=client.Client(username="oph-user", password="oph-
passwd", server="127.0.0.1", port="11732")
```

Another important extension to the Ophidia framework, which has been required to provide a stronger integration of the platform with JupyterHub, is related to user space management. In fact, former versions of Ophidia framework only provided support for virtual file system (i.e. management of datacubes) with the Ophidia server acting as a bridge from the (virtual) user requests to the actual execution, performed through a single (Linux) administrator user. However, since JupyterHub provides access to the physical Linux user account, Ophidia has also been enhanced to support the physical file system. To this end, the main extensions relate to the (i) Ophidia server, which has been extended to submit the jobs with the user's (Linux) account in order to preserve the permission on input/output files and integrate the load/download data features provided by the JupyterHub file manager with Ophidia, and (ii) the operator `OPH_FS`, which has been extended to improve support for basic physical file system management operation (e.g. creation, removal and renaming of folders, as well as showing folders content) from the Ophidia terminal and PyOphidia module. Documentation regarding the new operator is available at:

http://ophidia.cmcc.it/documentation/users/operators/OPH_FS.html

4.2.5 AAI Integration of Ophidia and JupyterHub with IAM

An early version of the support for IAM authZ/authN mechanisms was already available in Ophidia. During the first year of the project, the integration was subject to further testing and bug fixing in the context of ECASLab. The current implementation supports IAM tokens (OpenID-based) from both the Ophidia terminal and the PyOphidia module. Upon receiving a request, the Ophidia server, which is registered at the OpenID provider as IAM client, checks the token integrity and validity, asks the provider for user information and caches the data returned by the provider (thus avoiding to contact the provider for the next requests of the same user). If the user is authorized by the provider, then the Ophidia server can perform an additional local authorization check, granting access to the user only if some requirements are met: for instance, the user belongs to one of the

admitted organizations. Documentation describing the steps to activate and configure the OpenID-based authN/authZ in Ophidia is provided at:

http://ophidia.cmcc.it/documentation/admin/install/components/install_openid.html

Concerning the integration with JupyterHub, it has been performed by properly configuring the JupyterHub instance with the JupyterHub OAuth authenticator (<https://github.com/jupyterhub/oauthenticator>). In particular, the LocalGenericOAuthenticator class has been used to exploit IAM as the JupyterHub authenticator method.

A two-instance JupyterHub deployment supporting both local and IAM-based authentication is under testing at CMCC. This activity will be completed over the next weeks to provide a seamless security workflow from JupyterHub login to data analytics tasks execution.

4.2.6 Dockerization and operational development workflow

Several workflows and additional components have been put in place to make operations of the ECAS environment easier and facilitate easier transfer of updates from the development environment to the operational setting.

- **JupyterHub:** in order to avoid creating a Linux account for each ECAS user, the jupyter notebooks run in Docker containers and mounted volumes to allow persistence. A custom Docker image is created and contains all packages to connect with the Ophidia framework, use PyOPhidia and visualize (plot) the results. The image is continuously updated and the Docker build process is automated using GitHub and Docker Hub.

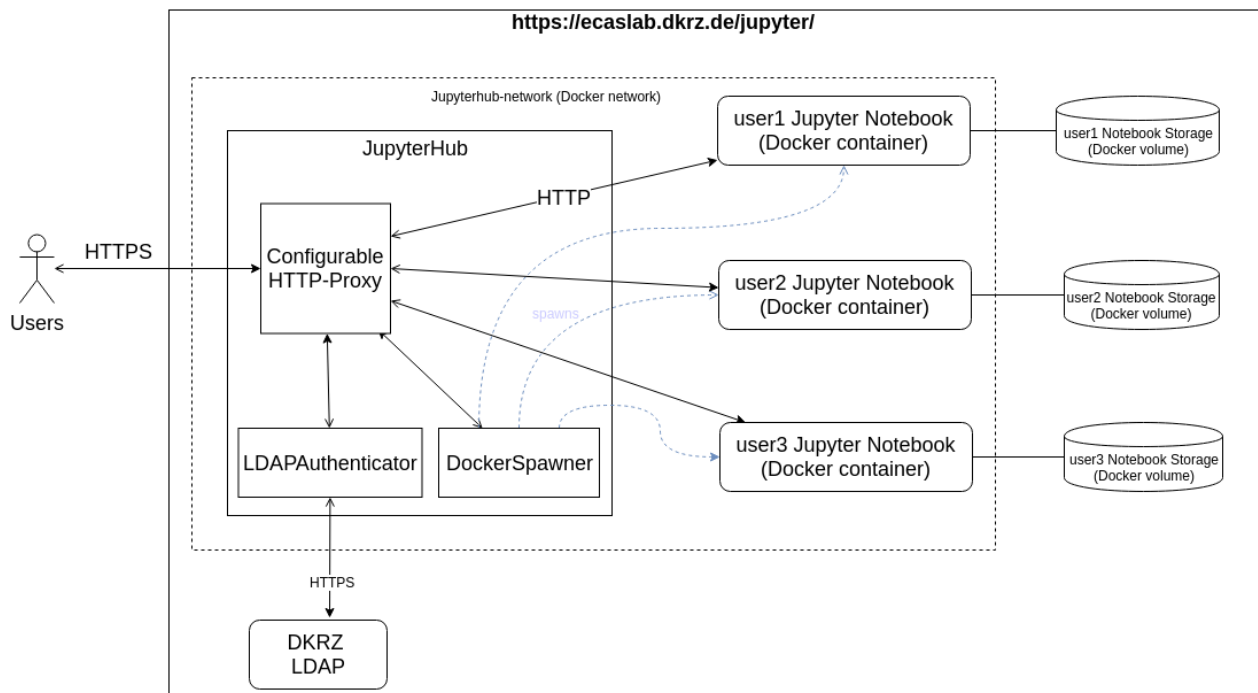


Fig. 6 – Workflow of the spawning of JupyterHub dockerized notebooks

- **The Elasticsearch, Logstash, Kibana (ELK)** stack is deployed internally principally to view and analyze Ophidia server and I/O logs instead of performing a login via ssh to the corresponding machines and looking for the logs. The setup is being extended to allow logs managements of the other existing ECAS components e.g. *JupyterHub* and also the coming ECAS AAI virtual machine. All components of the stack run in Docker containers except the beats tools, which run as system services.
- **Rancher (container orchestration framework):** we use Rancher mainly for monitoring the Jupyter notebooks, which run in Docker containers and also debugging all containers related to ECASLab.
- **ECAS AUTH VM (experimental)** in order to enable the EOSC-HUB AAI for ECAS/ECASLab an identity broker has been deployed and is connected to the JupyterHub instance. The identity broker allows users to authenticate against the three available EOSC-HUB AAI providers: B2ACCESS, EGI Check In and Indigo IAM.

4.2.7 Collaboration activities

- **WPS:** DKRZ also integrated ECAS with the birdhouse open source framework used for data processing in Copernicus CDS. An exemplary full ECAS workflow is already available via a standard WPS interface, accessible and executable via the birdhouse framework. The proof-of-concept use case concerns the calculation of the number of tropical nights (TN). This work is available at the ECASLab/JupyterHub.

- **New JupyterHub profile** for birdhouse integration mentioned above, a new Docker image is built and support Birdhouse functionalities. ECAS users can select the ECAS-Birdhouse profile. If they want to try the WPS-based TN workflow.
- **Workflows:** based on workflows/notebooks implemented by CMCC, the GitHub notebooks repository is extended with new use cases. Concretely, DKRZ designed new scientific examples for index calculation; resulted in training material and contributed to evaluation of the service as far as it is already integrated in a valid setting. These examples are available in the ECAS GitHub repository (<https://github.com/ECAS-Lab/ecas-notebooks>). There is also an ongoing work to analyse the performance of ECAS/Ophidia by calculating the climate indices with other existing tools like **CDO**.

4.2.8 Integration of ECAS with OneData

Initial discussions with the OneData team are ongoing to identify the best architectural solution and infrastructural setup at CMCC. Specific requirements (especially security constraints/policies) related to the data centre nature of the hosting site have been properly considered: each data centre has already its own software stack, hardware configuration and policies, therefore the integration with a new service, like OneData, needs to be implemented taking them into account.

4.2.9 Accounting integration

The main jobs executed on the ECASLab resources consist of Ophidia jobs (i.e. workflows of tasks or single tasks). In order to properly track resource usage on a user basis, the Ophidia framework has been extended to conveniently store the most important information in CSV format for accounting purposes. In particular, the Ophidia server stores information in two different files:

- **Workflow log:** tracks the workflow requests, including submission timestamp, workflow identifier, username, number of tasks, execution duration, etc.;
- **Task log:** tracks the single tasks (composing a workflow); among the information logged for each task, there are: submission timestamp, task identifier, workflow identifier (as stored in the previous file), operator name, number of cores, execution duration, etc.

An example of the second log files is provided in the following listing:

timestamp	idtask	idwf	operator	#cores	success_flag	duration
2018-11-14 20:10:39	244	197	oph_reduce2	20	1	1.360963
2018-11-14 20:10:39	247	197	oph_exportnc2	20	1	1.384643
2018-11-14 20:10:40	248	197	oph_explorecube	1	1	1.286801
2018-11-14 20:10:40	266	197	oph_set	1	1	0.001698
2018-11-14 20:10:40	245	197	oph_reduce2	20	1	1.560700

A detailed documentation about the accounting features has been published at:

<http://ophidia.cmcc.it/documentation/admin/management/accounting.html>

Additionally, to improve the system performance, the OphidiaDB system catalogue has also been improved to move all job information from a transient to a historical table upon job completion.

4.2.10 ECASLab monitoring integration

In order to support the system operation, a monitoring system has also been set up on the ECASLab at CMCC. The monitoring system exploits Grafana with an InfluxDB back-end. In particular, besides

traditional metrics (e.g. disk, RAM and CPUs usage) a set of application specific metrics (from Ophidia) and the related Grafana dashboard have been defined to monitor the executed workflows, the number of tasks, the users currently using the resources, etc. From the point of view of Ophidia, the extensions exploited for accounting have also been used as a base for the application-specific monitoring metrics. The following figure shows an example of this custom Grafana dashboard.

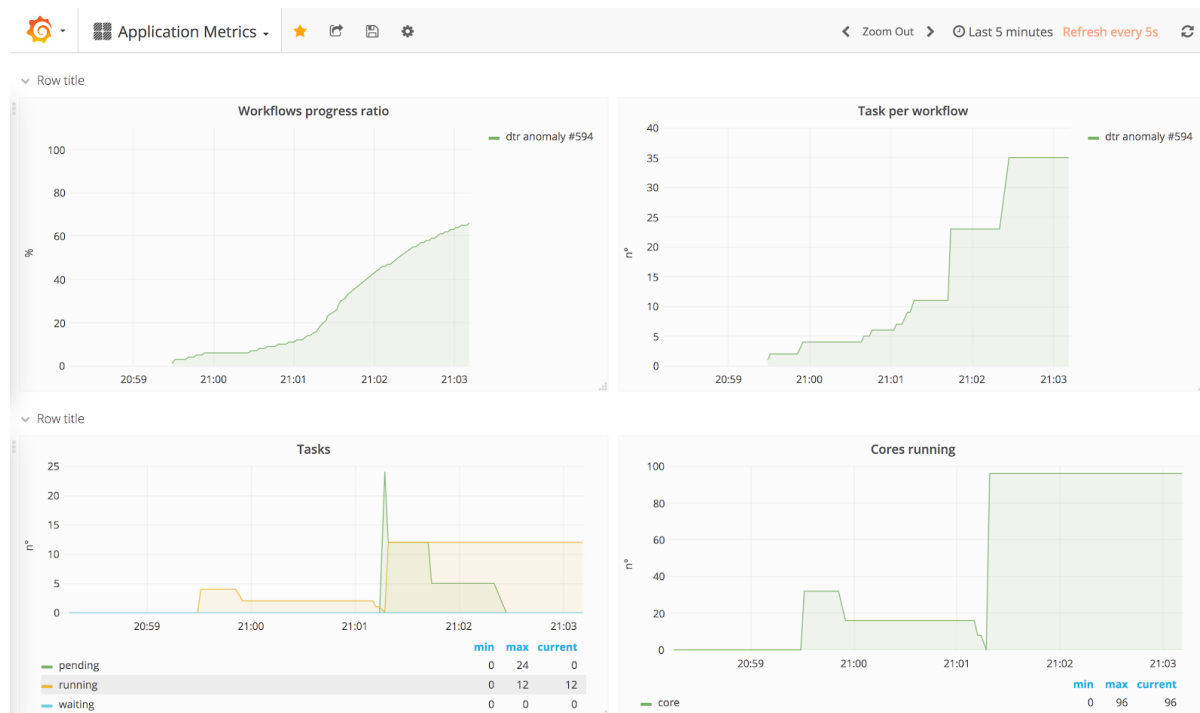


Fig. 7 – ECASLab monitoring Grafana dashboard

4.2.11 JupyterHub notebooks for training purposes

Training of users to exploit the ECASLab features has been a key activity for the first project year. Hence, to support user training, several demonstration Jupyter Notebooks, integrating some key features of the system, have been implemented. The Notebooks have been implemented and are available on the ECASLab GitHub repository [11]. Some of them address some basic features:

- ECASLab training: provides an overview of the ECASLab interface and some step-by-step instructions;
- Time series extraction: shows how to plot on a map a time series from an Ophidia datacube;
- Time series difference: shows how to perform the difference among two time series;
- Aggregated map: shows how to create a map from an Ophidia datacube;
- Subsetted map: shows how to create a map from a subset of a datacube.

Other notebooks address more complex computation:

- Tropical Nights: shows how to compute the Tropical Nights indicator;
- Frost Days: shows how to compute the Frost Days indicator;
- Summer Days: shows how to compute the Summer Days indicator;
- Icing Days: shows how to compute the Icing Days indicator;
- Daily Temperature Range: shows how to compute the Daily Temperature Range climate indicator.

These Notebooks have also been used as training material for the events held during the first year.

4.2.12 ECAS documentation

For better explaining ECAS and its related components, we created a technical documentation hosted at <https://readthedocs.org> and built automatically from GitHub repository and tested by Travis CI.

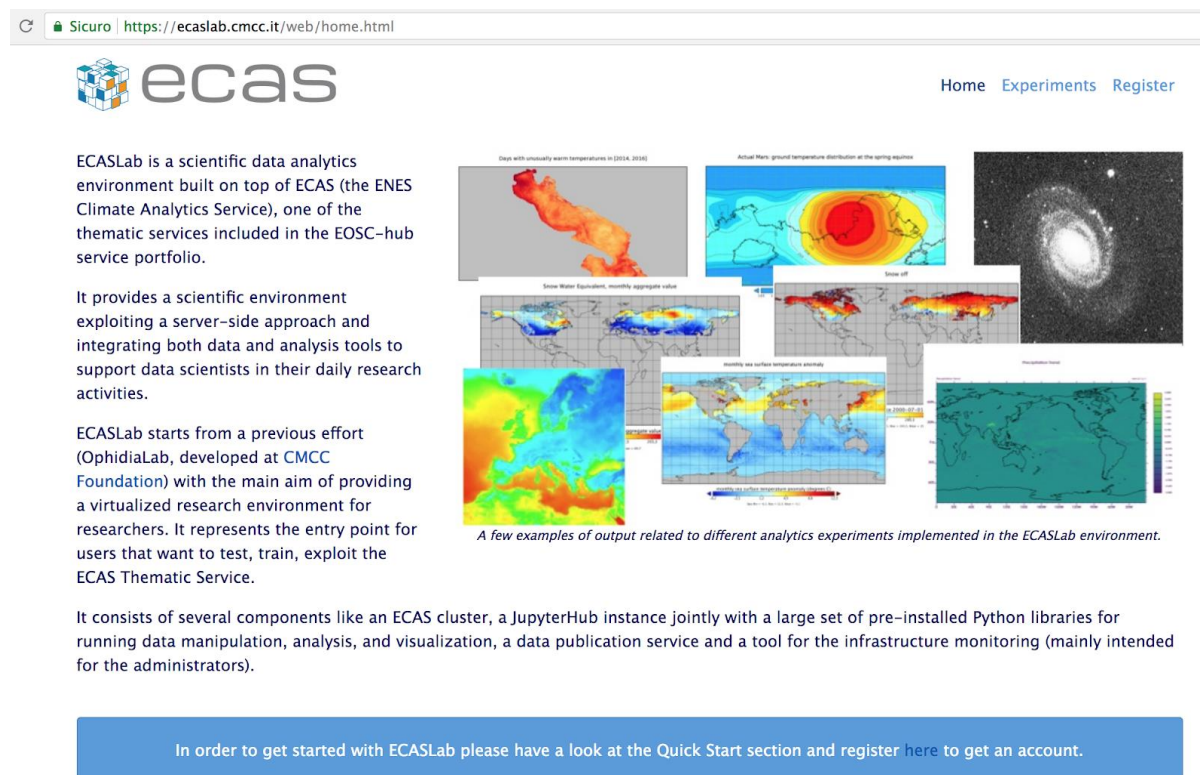
ECAS team on both sites can use the repository to add details on how to use the service.

Links:

- <https://ee-docs.readthedocs.io/en/latest/>

4.2.13 New ECASLab website for EOSC-Hub

A website portraying the main ECASLab features, including a quick start and documentation about climate change-based experiment, as well as the user registration form, has been developed. The website is based on the popular *LAMP* (Linux OS, Apache HTTP web server, MySQL RDBMS and PHP) software stack. It exploits JavaScript and a Bootstrap template for the layout and page style. The source code is available on the GitHub ECASLab repository: <https://github.com/ECAS-Lab/ecas-web>.



The screenshot shows the ECASLab website homepage. The browser address bar displays "Sicuro | https://ecaslab.cmcc.it/web/home.html". The page features the ECAS logo, navigation links for "Home", "Experiments", and "Register", and a central collage of scientific data visualizations including maps of temperature anomalies and satellite imagery. A blue banner at the bottom contains the text: "In order to get started with ECASLab please have a look at the Quick Start section and register here to get an account."

Fig. 8 – ECASLab web site

4.2.14 ECAS on the EOSC-hub marketplace

ECAS was also integrated with the EOSC-hub marketplace in its current early versions. Descriptions were added that indicate very concisely to users the value of the service, how to access the two instances and where to find further information. The marketplace entry for ECAS thus provides all necessary detail so that users are directed towards proper usage of ECAS and can take their first steps.

4.3 Identified gaps

There are two general areas where gaps in integration between other EOSC-hub services had an effect on ECAS integration, that are: EOSC-hub AAI and B2DROP service. Both cases were mitigated by putting workarounds in place that may hold for at least the integration phase of ECAS, which caused some delay in ECAS integration compared to the original plan. However, the issues are by no means so severe that they would ultimately stop delivery of a fully integrated ECAS service.

4.3.1 Missing common EOSC-hub AAI solution and operationally sane workflow

It must be assumed that users may already have an account for one of the AAI solutions used in EOSC-hub, i.e., IAM, B2ACCESS and EGI CheckIn. Thus, ultimately, ECAS should provide users access independent of the AAI solution they registered with, and should not require users to re-register with another solution. However, a common solution that provides seamless interfacing with AAI from ECAS point of view has not yet been provided by the project. This was already indicated in the beginning of the project and it was recommended that ECAS integrate with one of the solutions, pending resolution of the cross-connection issues by the AAI team. ECAS has opted to do so via the IAM service.

The basic technical integration on this is complete. However, a full operationally sensible workflow is not yet established pending additional setup and integration to be done in collaboration with the IAM team: it must be possible to activate users for the use of ECAS selectively, pending approval of the ECAS site managers, who need to be automatically informed of such new ECAS usage requests. Also, users need to agree to ECAS Terms of Use, which may also be specific to the ECAS site as they use local computing and storage resources. The workflows for this are not settled yet.

4.3.2 Less than ideal solution for B2DROP integration

The integration with B2DROP as done currently is less than ideal, ultimately resulting from not yet completed integration between IAM and B2DROP and lack of support for secured programmatic upload to B2DROP via secured WebDAV (the latter is a feature B2DROP would ultimately require from nextcloud). If users share results or scripts from the JupyterHub environment, they need to provide their B2ACCESS credentials separately from the original login to the ECAS environment, which seems cumbersome. A workaround was put in place that enables users to use B2DROP nonetheless if they provide their credentials via a separate file; this must only be done once by every user.

4.4 Future perspective

The integration and training activities of ECAS will continue during the 2nd project year. The goal is to have a pre-operational service ready and opened for wide usage by M18 and completes all integration activities by M22 latest.

Some ECAS components are already integrated with one of the EOSC-HUB AAI providers. We are working on providing a common AAI solution that possibly supports token propagations between ECAS components (e.g. Ophidia and JupyterHub).

ECAS will integrate B2SHARE most likely via the existing integration between B2DROP and B2SHARE. This will require defining the necessary metadata and a process to acquire it in the ECAS-Lab environments.

Regarding IM/Orchestrator an ansible role to support automated deployment of ECAS will be provided with the latest Ophidia release, by leveraging previous efforts. Such activity will be tested

and validated in the EGI-FedCloud. In this respect, future follow up will relate to integration of automated deployment procedures into Marketplace framework.

Over the next weeks, technical integration activities regarding OneData will proceed further. In particular an OneProvider instance will be set up at CMCC and integrated at the file-system level with the local ECASLab instance.

In PY2, ECAS will integrate B2HANDLE, putting persistent identifiers (PIDs) on output results and connecting them with input data, recording data lineage in the most basic way. This will require support for PID assignment and profiles by B2SHARE and OneData. Interaction with the services concerning these features has already been started.

In terms of future training, ECAS proposal for the EGU2019 is accepted as a Short Course session. Presentations and also Hands-on will be prepared to cover ECAS components and how to use them. Inter-thematic-service collaboration will be supported over the next months to build new integrated scenarios relying on multiple thematic services. In particular, based on preliminary interactions with openCOAST and the EGI team, joint ECAS & openCOAST use cases as well as training events may be planned/organised, pending evaluation of the use cases for possible common approaches.

5 T7.4 GEOSS

5.1 Service description

The GEO Discovery and Access Broker (GEO DAB) is a key component of the GEOSS (Global earth Observation System of Systems) Platform, transparently connecting GEOSS User's requests to the resources shared by the GEOSS Providers. GEO DAB scope is to simplify cross and multi-disciplinary discovery, access, and use (or reuse) of disparate data and information. It is a brokering framework that interconnects hundreds of heterogeneous and autonomous supply systems (the enterprise systems constituting the GEO metasystem) by providing mediation, harmonization, transformation, and QoS capabilities.

5.1.1 Architecture

The GEO DAB applies the broker pattern, which separates users of services (clients) from providers of services (servers) by inserting an intermediary, called a broker. When a client needs a service, it queries a broker via a service interface. The broker then forwards the client's service request to a server, which processes the request. The GEO DAB presently provides broker components for discovery, access, and semantics-enabled search.

The deployment of GEO DAB utilizes Cloud services (IaaS and PaaS) to ensure high availability, reliability and scalability of the system. The details of GEO DAB operational deployment are available in (Nativi et al., 2015, Big Data challenges in building the Global Earth Observation System of Systems, Environmental Modelling & Software 68, 1-26).

5.1.2 Main use cases

The target users of the service are:

- Single researcher
- Virtual Organization (VO)
- Research project
- Business

The main high-level use cases include:

- Discovery and Access of GEOSS datasets: GEO DAB is accessible via Web APIs to anyone and allows client applications to submit search and/or access requests to discover and/or download GEOSS datasets;
- Generation of new products utilizing GEOSS datasets: users can discover and execute workflows to generate new products useful for her/his analysis, utilizing the GEO DAB to discover and ingest input data. The ECOPotential[12] VLab[13] is used to orchestrate and run the workflow of interest.

5.2 Integration activities

During 2018, the Task 7.4 team developed

- the initial porting of GEO DAB on EOSC-hub infrastructure nodes, provided by CESNET. The porting currently includes the discovery module of the GEO DAB. The service was registered in EOSC-hub Marketplace (<https://marketplace.eosc-portal.eu/services/geo-dab>);
- the porting of the ECOPotential VLab on EOSC-hub infrastructure nodes, provided by CESNET;
- a demo which was demonstrated at the EOSC Launch event, held in Wien on the 23rd of November 2018, showing the use of GEO DAB and ECOPotential VLab to generate new added-value products relevant for Protected Areas management.

Integration with EOSC-hub infrastructure was implemented mainly utilizing Kubernetes APIs to manage and orchestrate the required software modules. In fact, as explained in the previous chapter, GEO DAB consists of multiple components that need to be orchestrated according to the workflow requested by the user, showed in high level step by step description here: <https://www.geodab.net>; hence Kubernetes allows to orchestrate them on the EOSC-hub computing cloud resources in the same fashion that GEO DAB administrators use to manage them on the Amazon cloud.

Documentation on GEO DAB was provided and is available in the training material section of EOSC-hub web site (<https://www.eosc-hub.eu/training-material/geo-discovery-and-access-broker-geo-dab>).

5.3 Identified gaps

Integration with AAI of EOSC-hub is missing. This could be particularly useful when extending the porting of GEO DAB and VLab, allowing users to utilize their AAI credentials to run workflows only on nodes already allocated to them.

Other main identified gaps deal with available Cloud services and in particular, queueing service and server-less computing.

5.4 Future perspective

The plan for next year is to enhance present integration. This will include: exploring solutions to fill the identified gaps in terms of required Cloud services; defining a clear use case for AAI integration; and implementing the AAI integration.

6 T7.5 OPENCoastS

6.1 Service description

The OPENCoastS service builds on-demand circulation forecast systems for user-selected sections of the North Atlantic coast and maintain them running operationally for the timeframe defined by the user. This service generates daily forecasts of water levels and 2D velocities over the spatial region of interest for periods of 48 hours, based on numerical simulations of the relevant physical processes (more information can be retrieved from EOSC-hub deliverable 7.1).

6.1.1 Architecture

The OPENCoastS service architecture includes a frontend with a user interaction component for forecast systems configuration and management, via a web application, a backend where models and mapping services run and a storage tier for preservation (more information can be retrieved from EOSC-hub deliverable 7.1).

6.1.2 Main use cases

The OPENCoastS service is available freely to anyone who plays a role in coastal areas, from coastal authorities to the general public. Presently, it has been applied to estuaries and coastal regions in Europe, Africa and North America with local and regional scale applications. Current users span all continents.

The OPENCoastS platform is dedicated to all entities with activity on coastal regions across Europe. It targets coastal managers, public institutions, research groups and private companies with responsibilities in emergency and monitoring purposes across Europe. National, regional or local coastal managers from the public and private sector with responsibilities in emergency and monitoring purposes need forecast systems to anticipate hazardous events and prepare emergency response. At the same time, these systems can support planning activities, from daily tasks to strategic interventions. Being able to reproduce the operational behaviour of coastal engineering interventions (even before they are implemented in the coast), the OPENCoastS service is a valuable tool for consultancy companies working in the field of coastal engineering to support engineering projects and their implementation (e.g. study the impact of maritime structure building and dredging interventions in coastal regions).

Given the flexibility and generic nature of the OPENCoastS service, research groups extend their limited use in specific sites to broad geographical scope studies of coastal processes, of the climate change and anthropogenic impacts in the coastal zone among other topics. This platform will also facilitate the access to circulation forecasts to research groups with little experience in numerical physical modelling of oceanic and coastal zones such as biologists, geologists and biogeochemists, which have strong needs in understanding the impact of water dynamics in water quality, ecology and sediments dynamics. By making the service available for deployment in any European coastal regions, OPENCoastS leverages the conditions for any entity to develop their responsibilities in a faster, efficient and high accuracy way.

6.2 Integration activities

During 2018, the OPENCoastS team integrated this originally national service with European and global services for water levels and meteorological forcing to provide the possibility for international service deployments. The service was integrated with both EGI CheckIn and the Marketplace, supported by implementation at EOSC-hub infrastructure nodes (INCD and ongoing at IFCA). Support for non-national users was provided through detailed documentation in English (manual) and two training activities (a hands-on course during the IMUM workshop in September and an e-tutorial to be held in December). The service was published both at EOSC-hub channels (Service Catalogue, Marketplace) and the EOSC Portal's Marketplace[\[14\]](#).

6.3 Identified gaps

The integration with the EGI CheckIn and the Marketplace has been completed. The main missing part is the handling of data for users. Access to service outputs is provided by the frontend of the service but for a limited amount of time. A persistent repository along with facilitated ways to access the data are required.

6.4 Future perspective

In late 2018 and during 2019, the following actions will be performed:

- OPENCoastS@IFCA: the OPENCoastS services will be created in the IFCA/CSIC cloud first as a duplicate instance of the whole service and later interconnected with the INCD service, towards a high availability, to have some replication/synchronization of files and databases.
- Integration with European forcing services for waves to permit wave-current interaction in the European deployments.
- Integration with other EOSC-hub core services for data preservation.
- Integration with EOSC-hub computing (Grid) services, through the DIRAC4EGI.

7 T7.6 WeNMR

7.1 Service description

The WeNMR Thematic Services [15] are a suite of web portals, providing user-friendly access to complex computational workflows and tasks. These allow inexperienced and experienced structural biologists to use state-of-the-art software for their data analysis while benefiting from the computational infrastructure provided through the EOSC-hub project. The services make use of high-throughput computing (HTC) resources [16], but some are also using accelerated computing (GPGPUs) grid resources and cloud computing. The WeNMR suite is composed of seven individual platforms (refer to Deliverable 7.1 for details):

- AMPS-NMR (<http://py-enmr.cerm.unifi.it/access/index>), a web portal for Nuclear Magnetic Resonance (NMR) structures.
- CS-ROSETTA (<http://haddock.science.uu.nl/enmr/services/CS-ROSETTA3>), to model the 3D structure of proteins.
- DISVIS (<http://milou.science.uu.nl/enmr/services/DISVIS>), to visualise and quantify the accessible interaction space in macromolecular complexes.
- FANTEN (<http://abs.cerm.unifi.it:8080>), for multiple alignments of nucleic acid and protein sequences.
- HADDOCK (<http://haddock.science.uu.nl/enmr/services/HADDOCK2.2>), to model complexes of proteins and other biomolecules.
- POWERFIT (<http://milou.science.uu.nl/cgi/services/POWERFIT/powerfit>), for rigid body fitting of atomic structures into cryo-EM density maps.
- SPOTON (<https://milou.science.uu.nl/services/SPOTON>), to identify and classify interfacial residues as Hot-Spots (HS) in protein-protein complexes.

7.1.1 Architecture

As already described in Deliverable 7.1, all portals are web-based, built on a variety of technological solutions (e.g. Python, Flask, Apache...), but all present a unified and well-recognizable front-end to users. They make use of the EOSC HTC resources to distribute jobs to the sites supporting the enmr.eu VO, using in most cases DIRAC4EGI for job submission, but also gLite in some specific cases, e.g. like the GPGPU-grid enabled DISVIS and POWERFIT portals that are sending jobs to specific CEs in Florence. These two applications make also use of udocker, a basic user tool to execute simple Docker containers in user space without requiring root privileges, developed by the INDIGO-DataCloud project and currently supported by EOSC-Hub.

7.1.2 Main use cases

As recently described in the 2nd edition of the EOSC-Hub magazine[17], WeNMR is serving the structural biology community at large. Structural biology studies the functions and interactions of proteins, nucleic acids and other biomolecules using experimental methods such as X-ray crystallography, Nuclear Magnetic Resonance (NMR) or cryo-electron microscopy (cryo-EM). All these methods generate data that needs to be processed, analysed and finally converted into three dimensional (3D) structures (or models) of biomolecules using a variety of computational

tools and techniques. Gaining access to 3D structures of biomolecules, their dynamics, and their interactions with other molecules is the key to a proper understanding of their function. It also allows, for example, rationalizing the effect of disease-causing mutations, to engineer better molecules for material, health or food applications and to obtain a starting point for drug design to combat disease. As such, structural biology has a strong socio-economic impact on many application fields from health, to food, to materials.

The services are making use of the computational infrastructure provided through the EOSC-hub project. The services make use of high-throughput computing (HTC) resources, but some are also using accelerated computing (GPGPUs) grid resources and cloud computing. This support has been formalized by a Service Level Agreement between EGI.eu and the enmr.eu VO (represented by the Faculty of Science – Department of Chemistry of Utrecht University). This SLA was established in 2016 and has been renewed in 2018, granting to enmr.eu VO until 31/12/2020 an amount of opportunistic computing time up to 53 Million of normalized CPU hours and opportunistic storage capacity up to 54 TB[18]. Five resource centres signed this last version of the SLA: INFN-PADOVA (Italy), TW-NCHC (Taiwan), SURFSara and NIKHEF (The Netherlands), NCG-INGRID-PT (Portugal).

7.2 Integration activities

As already reported in Deliverable 7.1, the WeNMR thematic services have been in operation from day 1 of the project, sending over the first eight months of the project over 5 million jobs to the EOSC HTC resources, most of which through the DIRAC4EGI service. During the first year of the project, a number of WeNMR portals have been migrated from the old gLite-based job submission to the EOSC DIRAC4EGI service. Further, all portals are now offering Single Sign On (SSO), either through the West-Life SSO which connects to both ARIA (the access management solution of the Structural Biology infrastructure INSTRUCT-ERIC) and the old legacy WeNMR SSO[19], or directly through the EGI Check-in. Users can now register and use the WeNMR services using the Check-in, allowing them to use a variety of identity providers. AMPS-NMR is currently using only the West-Life SSO, but it is in process of including EGI-CheckIn too.

7.2.1 EGI Check-in integration

A major part of the WeNMR Utrecht contributed services have been migrated to a new Flask-based architecture, with a central registration and authentication system (accessible from: wenmr.science.uu.nl). For these services, EGI Check-in has been integrated using the Flask-OIDC library, and EGI Check-in is now an active agent in the registration system. For those services not fully integrated, the EGI Check-in support has been enabled using the EGI provided JavaScript OIDC client. The contributed source code is freely available at https://github.com/WeNMR-EOSC/checkin_endpoint.

7.2.2 West-Life SSO integration

INFN developed a new plugin to enable West-Life SSO as authentication method for Onedata. This contribution was pulled upstream to Onedata software repository[20][21]. INFN has also enabled West-Life SSO as authentication method for accessing the cloud resources of its INFN-PADOVA-STACK instance of the EGI Federated Cloud. According to the SLA signed with EGI.eu, up to 100 VCPUS and 200 GB of VRAM are available to WeNMR users until the end of 2020.

7.2.3 West-Life Virtual Folder integration with Onedata

Virtual Folder (VF) is a tool developed in the context of West-Life project and now maintained by INSTRUCT-ERIC. Currently integrated in several WeNMR portals, it acts as a gateway for many storage systems, such as Dropbox, B2Drop and any other system accessible through the WebDAV protocol. INFN developed a plugin for integrating VF with Onedata, i.e. to enable Onedata storage system as additional back-end. The plugin is able to get information about files and directories from a given Oneprovider. It is also able to mount on demand any user space available on the provider, making use of the Oneclient command line tool.

Unfortunately, the testing phase of the plugin has shown some concerns: 1) there are several compatibility problems between Oneclient and Oneprovider, so that the plugin only works if the same release is installed on both sides; 2) there are dependencies on third party libraries which are not stable enough; 3) at the time of writing, Onedata software maturity is still at the level of release candidate.

For the above reasons, the current integration of VF with Onedata can only be considered as a prototype. It will hopefully become "production ready" only when a final release of Onedata is available.

7.3 Identified gaps

CheckIn and MarketPlace integration has been completed. The main missing part is a simple way for users to access data repositories (e.g. B2DROP) directly from web portal to either upload (provide a WebDAV address) data to the portal, or directly upload the results to their repository without have to transfer the data to a local device.

7.4 Future perspective

We are planning to connect some of our portals to data repositories such as the ones offered by EUDAT in order to allow user to directly upload and/or download data/results. The data generated by the WeNMR services are however very specific to a user/application and not globally reusable by third parties. This is very different for example from sky images collected by telescopes. We do aim, however, at facilitating data deposition directly into public repositories where relevant.

8 T7.7 EO Pillar

The EO-Pillar within the EOSC provides access to different services established in the field of Earth Observation (EO). The services are categorised into three main classes: data access and computing services, data exploitation services, general user services.

These services are:

- **GEP:**
 - High-Resolution Change Monitoring for the Alpine Region
 - EO Services for Earthquake Response and Landslides Analysis
- **EODC JupyterHub for global Copernicus data**
- **EODC Data Catalogue Service**
- **Sentinel Hub**
 - OGC compliant WMS, WCS and WMTS access to global archives of Sentinel-1 GRD, Sentinel-2 L1C, Sentinel-2 L2A, Sentinel-3 OLCI, Sentinel-5P, Envisat MERIS, Landsat-8, Landsat-5 (Europe archive), Landsat-7 (Europe archive)
 - Statistical API providing statistical data over long time-series (e.g. min/max/median value over point or polygon)
 - Configuration utility to expose various configurations over WMS/WCS/WMTS
 - Custom scripting for ad-hoc definition of algorithms
 - More info: <https://www.sentinel-hub.com/develop/documentation/api>
- **OSX-Sentinel** (<https://sentinel.eosc.grnet.gr>)
Integrated with EGI Check-in, this service is a child/leaf node of the Hellenic National Sentinel Data Mirror Site and it performs order management for Sentinel 1,2 & 3 using OpenSearch and OData APIs for browsing and accessing the EO data. The service is also supported with resources offered by GRNET's IaaS ~Okeanos [25] and ~Okeanos-Knossos [26]. GRNET is also planning to investigate how to integrate this service with its EGI Fedcloud Site.
- **MEA Platform** (Data access and exploitation service)
- **Rasdaman EO Datacube**
 - Sentinel 2, Landsat 5, Landsat 7 time series datacubes via OGC conformant service endpoints, accessible through a variety of clients
 - OGC WMS, WCS, and WCPS are supported
 - Federation with the CODE-DE precursor
 - Service endpoint: <http://eoschub.rasdaman.com:8080/rasdaman/ows>
- **CloudFerro Data Collections Catalog**
The Catalogue is being based on CKAN open source software, which is widely used for open data publications like e.g. European Data Portal, data.org.uk or danepubliczne.gov.pl. CKAN provides user-friendly web interface for all activities associated with data publication and subscription. It is capable of advanced data management. All datasets are organized and described with metadata, which allows it to be easily discoverable, with the use of search phrases and customizable filters (e.g.: tags, categories, data formats). It is possible to publish one dataset in different data formats, not only as downloadable files but also as links to web service, web API or links to external WWW resources. Datasets can be stored in CKAN, along with version history and dataset statistics, which allows monitoring the interest in datasets. CKAN also provides functionalities for collaboration, community participation and providing feedback, such as comments, ratings and sharing. CKAN is highly customizable in both terms

of Look&Feel and functionalities. CKAN provides very rich RESTful JSON API, which allows other applications to discover and access the datasets. It can be integrated easily with Semantic Web technologies such as RDF data model and SPARQL.

- **CloudFerro Infrastructure**

CREODIAS processing covers full set of virtual resources available in the solution: VM – Virtual Machines (or virtual computing servers) with several operating systems available (both free like CentOS, Ubuntu, Debian, Scientific Linux, and commercial like RedHat, SUSE, Microsoft Windows Server), virtual storage volumes that can be easily mounted to the VMs together with object storage solution, virtual networks, virtual appliances like firewalls (FWaaS) and VPN concentrators (VPNaaS), physical servers (baremetal) that can be integrated to the virtual world, Single Server VMs – full physical server with a single VM and very fast passthrough NVMe storage – a combination of advantages of a dedicated server and a cloud VM (high capacity, storage speed, no noisy neighbour problem).

- **CloudFerro Data Related Services - EO Finder**

The Finder tool allows finding data products stored in the repository, obtained or processed at selected times with selected cloud coverage levels and with other selection criteria.

- **CloudFerro Data Related Services - EO Browser**

CREODIAS EO browser allows browsing wide archive of Earth Observation products, created by ESA's Sentinel 1, Sentinel 2, Sentinel 3, ESA's archives of Landsat 5, Landsat 7, Landsat 8 and Envisat. It provides ability to visualize and download chosen products in .png and .jpg formats.

- **EPOSAR service**

The EPOSAR allows for a systematic generation of sub-centimetric ground displacement maps and time series by exploiting Sentinel-1 images.

8.1 The Geohazards Thematic Exploitation Platform (GEP)

8.1.1 Service description

8.1.1.1 GEP - High-Resolution Change Monitoring for the Alpine Region

This service provides an interferometric & coherence product at 50m resolution and 25m pixel spacing systematically generated every 6-days, for each new Sentinel-1 SLC pair, over the Alpine Region (<https://geohazards-tep.eo.esa.int/geobrowser/?id=eoschub-alpsmonitoring-app>).

It supports ground deformation monitoring, as well as rapid response to earthquakes occurring within the processing mask, by automatic generation of co-seismic interferograms that are published in a dedicated GeoBrowser, and made available for visualization and download.

- User's Manual: <https://terradyne.github.io/doc-tep-geohazards/>
- Blog: <https://geohazards-tep.eo.esa.int/#!/blog>
- <https://discuss.terradyne.com/t/interpreting-the-layers-of-dlr-s-insar-browse-service/216>

8.1.1.2 GEP - EO Services for Earthquake Response and Landslides Analysis

This is a thematic application of the Geohazards TEP, providing access to a set of on-demand terrain motion services supporting: interferogram generation, co-seismic displacement mapping, landslide rapid mapping and landslide displacement field monitoring with Sentinel-1 and Sentinel-2 data (<https://geohazards-tep.eo.esa.int/geobrowser/?id=eoschub-landslide-app>).

- Users Manual: <https://terradyne.github.io/doc-tep-geohazards/>
- Blog: <https://geohazards-tep.eo.esa.int/#!/blog>

-
- <https://terradue.github.io/doc-tep-geohazards/tutorials/index.html>

8.1.2 Architecture

The Geohazards TEP platform is a complex system composed by the following main components:

- **Web portal:** a community portal, focusing on sharing processing results amongst users, integrating social media and gathering users in communities. It offers a community workspace where users can interact amongst themselves and access processing job or datasets shared by the community. It offers also a completely integrated geobrowser, in which users can discover existing EO collection or EO products produced by the community, but also create new processing jobs using the available WPS services.
- **Data Agency** including
 - **Catalogue service:** the EO data catalogue allows quickly accessing metadata about main EO satellite missions all over the world. Metadata are continuously harvested and ingested into the catalogue. The catalogue is also used to publish user's data products (directly uploaded by the user, or retrieved from a WPS job).
 - **Data Gateway:** it manages the data access and storage. It automatically pulls data results created by the processes on the WPS provider to a dedicated storage, in order to create value-added (e.g. WMS layers, time series, ...), persistent and quickly available data results. It also supports programmable and systematic data caching, in order to have data ready for specific area of interest of the users
- **Production Center:** the PC is a set of components enabling the processing services on the Geohazards TEP. It contains several heterogeneous modules enabling several specific functions from service integration to bulk and massive processing in clusters over IaaS and cloud. In particular it delivers a dynamic multi-tenant cluster with the full Hadoop YARN stack
- **Development environment:** the Developer Cloud Sandbox PaaS is the environment to integrate scientific applications written in a variety of languages (e.g. Java, C++, IDL, Python, R), then deploy, automate, manage and scale them in a very modular way. The algorithm integration is performed from within a dedicated Virtual Machine, running initially as a simulation environment (sandbox mode) that can readily scale to production (cluster mode) and then be deployed on the Production Centre. It is accessed from a harmonized Shell environment and provides support tools to facilitate the data access and workflow management tasks.

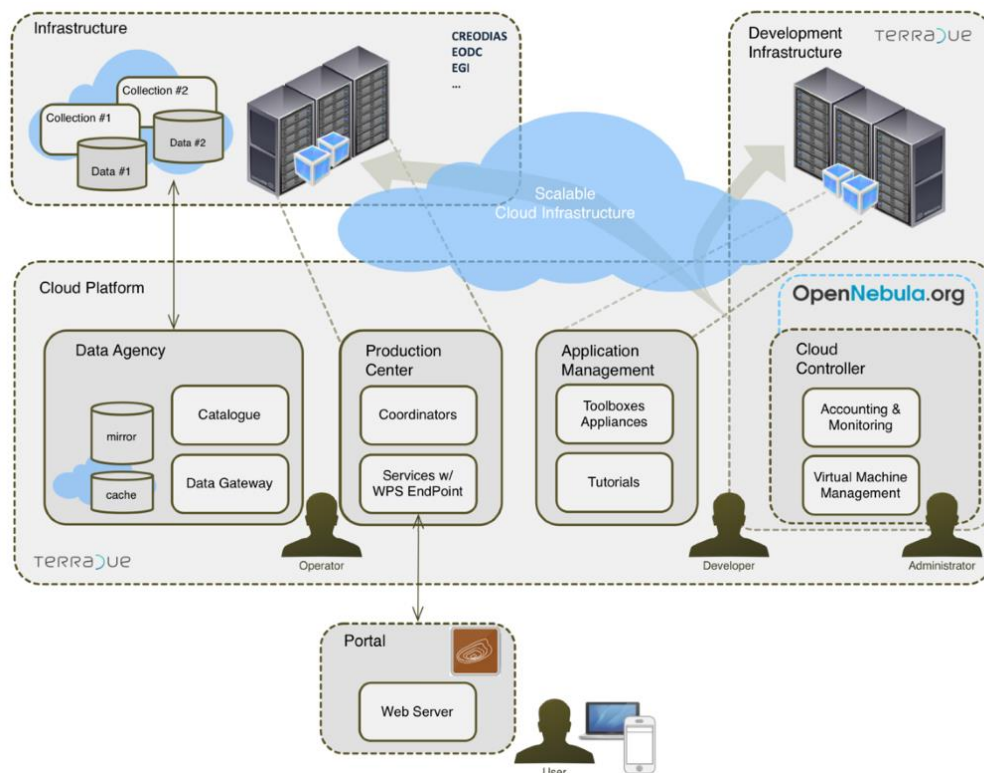


Fig. 9 – Geohazards TEP platform

8.1.3 Main use cases

The Geohazards TEP (GEP, <https://geohazards-tep.eo.esa.int/>) is an enhancement of the precursor platforms (G-POD, SSEP), and is designed to support the Geohazard Supersites (GSNL) and the Geohazards community via the CEOS WG Disasters. GEP is an ESA originated R&D activity on the EO ground segment to demonstrate the benefit of new technologies for large scale processing of EO data. Its goal is to apply a complementary operations concept, i.e. moving User activities to the Data: users access a work environment containing the data and resources required, as opposed to downloading and replicating the data 'at home'.

The Platform allows both on demand processing for specific user needs and systematic processing to address common information needs of the Geohazards community as a whole, as well as massive processing on multi-tenant computing resources on the Cloud.

Such capacities address the challenges of monitoring tectonic areas on a global basis, and of studying a range of Geohazards. To exploit the geo-information generated using the Platform, the GEP leverages open APIs for the integration of interactive processing and post-processing services.

The GEP provides innovative responses to the Geohazards community needs (services & support)

- On-demand processing services to address AOI-specific analysis.
- Systematic processing services to address needs for “common information layers”.
- Massive Cloud Compute power, managing multi-tenant resources.
- Access to Copernicus Sentinels-1/2/3 repositories.
- Access to 70+ TB of EO data archives (ERS and ENVISAT), and specific data collections from EO missions, such as JAXA’s ALOS-2, ASI’s Cosmo-SkyMed and DLR’s TerraSAR-X, provided under special arrangements in the framework for the CEOS WG Disaster and the GSNL.
- Support the generation strain rate estimates and the mapping of active faults at the global scale by providing EO InSAR and optical data and processing capacities.

- Supports development and demonstration of advanced science products for rapid earthquake response.

The GEP platform is operated by Terradue.

Service providers for the services offered in the context of EOSC are DLR, TRE-Altamira, CNRS EOST, ESA. Resources providers are CloudFerro (CREODIAS) and EODC (EODC Cloud).

8.1.4 Integration activities

The integration activities for the GEP during the first year of the project have focused on the integration and deployment of the platform Production Centre on the IaaS layer provided by two different EOSC-Hub Resource Providers (CloudFerro/CREODIAS and EODC Cloud). The Production Centre provisions a dynamic multi-tenant cluster with the full Hadoop YARN stack. In this context “dynamic” means that the cluster capacity can be scaled horizontally, i.e. nodes (VMs) can be added or removed on the fly, based on the cluster load and scheduled processing. To achieve this the Production Centre relies on two components, the capacity Manager and the Cloud Controller, that interact directly with the IaaS layer via a machine-to-machine interface (API) to add or remove nodes from the cluster. This required the development/adaptation of the Cloud Controller driver for OpenStack that is the solution adopted by both IaaS provider.

Moreover, some adaptations have been also applied to the Production Centre data management tools in order to enable access for the deployed services to the Sentinel data made available locally by the Resource Providers with different protocols (NFS, S3) and with different repository structures. All information for integration into EOSC-hub marketplace has been provided and publication is expected to be performed within the end of November.

8.1.5 Identified gaps

The integration and deployment of the platform Production Centre is still ongoing due to late availability of IaaS resource. The plan is however to complete the integration and start operations by mid-December. No specific gaps/issues are identified so far.

8.1.6 Future perspective

In order to enable the EOSC-HUB AAI for the Geohazards TEP, there are plans to develop/integrate an identity broker to allow users to authenticate against one or more EOSC-HUB AAI providers (EGI Check In, Indigo IAM and B2ACCESS). It is still under discussion which ones will be tackled and how they will be prioritized. Decision will be made within the end of the year.

8.2 The EPOSAR Service

8.2.1 Service description

EPOSAR is one of the services of the EPOS infrastructure (www.epos-ip.org). EPOSAR, based on the Small BAseline Subset (SBAS) DInSAR technique, is targeted to generate Earth surface displacement maps and time series with sub-centimetric accuracy by exploiting Sentinel-1 (S1) data of the Copernicus Programme. The service implements the whole processing chain, from SAR data retrieval until the generation of geocoded products. It is able to efficiently manage very large SAR datasets (hundreds or thousands of acquisitions) in very large computing environments (it has been used with 280 parallel AWS instances). The products are generated in automatic and systematic way over several selected areas of the Earth and are continuously updated when new S1 acquisitions are

available. The achieved products can be discovered, visualized and downloaded by users through the web interface of the EPOS infrastructure.

8.2.2 Architecture

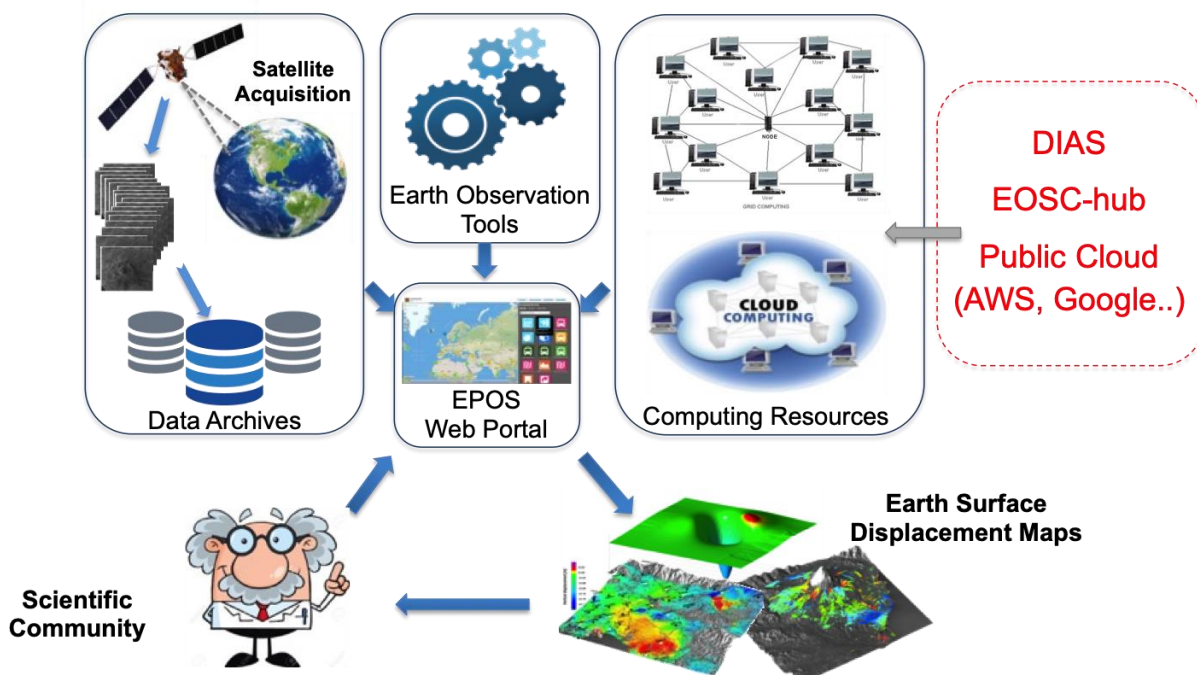


Fig. 10 – EPOSAR architecture

The EPOSAR service is designed to allow scientific users to jointly exploit Sentinel-1 SAR data, HPC resources and Earth Observation tools and processing techniques in order to easily generate value added products, such as Earth surface displacement maps and time series. In particular, it access the Sentinel-1 SAR data catalogues and archives made available by the Copernicus initiative (Copernicus Open Access Hub, DIAS, etc.) and is envisaged to exploit different cloud computing resources, such as those available within the DIAS initiatives (i.e. CloudFerro - CREODIAS) and also private cloud ones.

8.2.3 Main use cases

The EPOSAR service offers systematic processing to provide advanced interferometric products for the scientific community on several areas of the Earth that are of particular interest for geophysical dynamics dominating natural hazards such as earthquakes, volcanoes and landslides, and therefore to support all the activities related to risk mitigation and prevention.

The EPOSAR service is envisaged to exploit different cloud computing resources, such as those available within the DIAS initiatives (i.e. CloudFerro - CREODIAS) and also private cloud ones.

8.2.4 Integration activities

The main activities developed during the first year of the project have been aimed at deploying the EPOSAR service on the EOSC-hub IaaS layers. In particular, the cloud resources provided by CloudFerro were used, configured and tested in order to run in parallel the P-SBAS DInSAR algorithm. This testing phase was useful to tune the cloud computing resources needed to properly

run the P-SBAS processing chain and to optimize their usage. It allowed figuring out and setting the minimum requirements that will be necessary during the production phase. Moreover the interface and the access to the EO data catalogue were tested.

8.2.5 Identified gaps

The main issues encountered during the integration with EOSC-hub were related to the malfunctioning of some IaaS resources. In particular, several problems were experienced when trying to access the EO data catalogue directly from the CloudFerro resources. It is worth noting that the availability and robustness of the Sentinel-1 data catalogue is a fundamental prerequisite to deploy the EPOSAR service.

8.2.6 Future perspective

During the next months the full integration of the EPOSAR service within EOSC-hub is envisaged. This means to create the final configuration of all the cloud resources needed for the production phase, to consolidate the access to EO data catalogue and test its robustness, and to set up the EPOSAR service for the automatic and updated processing of the Sentinel-1 dataset acquired over the areas of interest. The production phase will address some areas relevant for the EPOS community and the achieved products will be accessible through the EPOS central hub and the EPOS Thematic Core Service Satellite Data access point.

8.3 EODC JupyterHub for global Copernicus data

8.3.1 Service description

This service is based on an implementation of JupyterHub at EODC. The service functions as a starting point to get free access to Copernicus Sentinel satellite data hosted at EODC. In addition, the service facilitates the development and realizations of algorithms. Functions and algorithms can be developed and executed directly on the data by utilizing well-known environments such as Jupyter notebooks and a bash Unix shell. The service can be accessed via <https://jupyterhub.eodc.eu/hub/login>. Support is provided via support@eodc.eu.

8.3.2 Architecture

The current architecture of the service is single virtual machine (VM) setup, provisioned within the EODC cloud environment based on OpenStack. The software stack deployed on the VM consists mainly of the Docker daemon running a customized JupyterHub container deployment. Customisation of the JupyterHub was needed in order to automatically mount the Copernicus Sentinel satellite data archive inside the container and provide various pre-installed Python packages. In addition, configurations with reference to a ready to use user federation interface have been undertaken with the objective to provide SSO via EGI. Accordingly, the EODC JupyterHub was connected to the EODC user identity and access management service. Within that service, an identity federation was established via OpenID Connect towards EGI.

8.3.3 Main use cases

The service is foreseen to attract users especially in the field of earth observation but also other interested in satellite imagery. Previous standard procedures to get access to those datasets were a direct access via a VM per user. This kind of data access is very resource demanding in addition to a

high entrance barrier because of the utilised VM access. JupyterHub simplifies the access to the data and provides a simple platform to develop first algorithms out of the box.

8.3.4 Integration activities

The service was developed within the framework of EOSC-hub and is deployed on the EODC Cloud. Because of its micro service architecture a migration to PaaS will be done in the further course of the project depending on the user uptake.

8.3.5 Identified gaps

Technical gaps identified can be summarised as follows. One missing component is the provisioning of a persistent storage in order to be able to store processed or tested data. Such a storage pool would be beneficial in order to be able to reconnect to the service and start from where the user has left. A concern related to that is the potential forwarding of the user information to the actual running container environment. At the moment the container is executed via a proxy account for all the users.

8.3.6 Future perspective

The service is currently not fully operational, accordingly one major goal is to make the service operational and provide access to it via the EOSC marketplace. The current proof of concept should be transferred to a more elastic compute environment (PaaS). Accordingly, a second goal is to migrate the entire service to the EODC PaaS currently running OpenShift to be able to scale the service up/down as requested by the users. Additional focus will be given to improve the usability of the JupyterHub with respect to earth observation use cases by providing additional need software libraries and viewing possibilities.

8.4 EODC Data Catalogue Service

8.4.1 Service description

The EODC Data Catalogue service allows querying the Copernicus Sentinel satellite data hosted at EODC. The service is available through a simple Web GUI, eomEX+, as well as an API. The back-end of eomEX+ is the EODC pycsw server, an implementation of an OGC CSW server. As a consequence, the eomEX+ API is accompanied by an expert level API provided by the EODC CSW server, located at <https://csw.eodc.eu>. Further details can be found here: <https://eomex.eodc.eu/manual>.

8.4.2 Architecture

The service is available through a simple Web GUI, eomEX+, as well as an API. The web-frontend is deployed on a single virtual machine hosted on the EODC Cloud environment. The Python Flask framework is utilised for the realisation of the Web GUI. Native python bindings to the gdal ogc api connect the EODC pycsw server with the front-end. The metadata infrastructure consists of a single POSTGRES database instance, which is exported via the pycsw server.

8.4.3 Main use cases

Search/query data assets provided by EODC. Results are provided in various output formats (atom+xml, json, xml) and schemas.

8.4.4 Integration activities

No specific integration with EOSC-hub has been implemented yet, however additional metadata fields have been added to the metadata schema in order to improve the search of the Copernicus Sentinel satellite data.

8.4.5 Identified gaps

No gaps have been identified.

8.4.6 Future perspective

Because of the enormous number of objects stored in the metadata database, several optimisations are foreseen to reduce the DB query response time. Those optimisations will focus on a load balancing DB infrastructure and on optimised DB schemas.

8.5 Rasdaman EO Datacube

8.5.1 Service description

The Rasdaman EO Datacube service allows querying the Copernicus Sentinel satellite data (Landsat 5, 7, 8, Sentinel 2, ...) hosted by Rasdaman datacube engine - a multi-parallel, federated array database system optimized for flexibility, performance, and horizontal/vertical scalability. Its interfaces consist of easy-to-use OGC WMS and WCS APIs plus high-level, declarative, standardized query languages (OGC WCPS and ISO Array SQL) allowing any query, any time, on any size on 1-D sensor time series; 2-D airborne/satellite image maps; 3-D x/y/t satellite image time series and x/y/z geo tomograms; 4-D climate and ocean data.

8.5.2 Architecture

The Rasdaman architecture consists of a full-stack implementation of a datacube analytics engine, with every component handcrafted and optimized for fast, scalable processing. Rasdaman runs multi-parallel rserver processes. On top of core Rasdaman is a geo-services layer, which offers OGC standards-based coverages, supporting regular as well as irregular grids, etc. OGC standards supported include WCS, WCS-T, WCPS, WMS.

8.5.3 Main use cases

Geospatial web service technologies, such as the OGC WC(P)S datacube suite, unleash new opportunities to access large volumes of geospatial data (Terabytes to Petabytes), especially popular satellite images like Landsat, Sentinel via the Internet and to process them at server-side by rasdaman. Users are not restricted any longer by available disc space and computing capacities of their local machines or organisations. Requests to an OGC WC(P)S can directly be integrated into existing processing routines, giving users better insights into data.

Further, this allows for faster development of WebGIS applications. The most powerful feature is WC(P)S service federations that combine access and processing of data from different coverages by different data providers. They allow establishing true interoperability of decentralised data repositories over the Internet between data service partners.

8.5.4 Integration activities

Not yet integrated into the EOSC-hub framework.

8.5.5 Identified gaps

The underlying DIAS archive is still suffering from performance and stability problems; this is being addressed currently by the providers.

8.5.6 Future perspective

The service will become part of the emerging European Datacube Federation.

8.6 CloudFerro Data Collections Catalog

8.6.1 Service description

8.6.1.1 Open search interface

Existing EO Cloud search interface, based on extended version of the open source RESTO software (<https://github.com/jjrom/resto>). It provides API-s in OpenSearch, GEOJson and ATOM standards allowing for easy data search. The interface exposes the operational metadata catalogue. It is used by EO Browser as well as exposed to external users.

8.6.1.2 C-SW interface

Standard OGC C-SW web service capable of publishing ISO19115 metadata records through the OGC standard Catalogue Service for Web interface.

8.6.1.3 SPARQL interface

SPARQL endpoint web service implementing core of W3C specifications: SPARQL 1.1 Query Language, SPARQL 1.1 Federated Query, SPARQL 1.1 Protocol. SPARQL endpoint exposes RDF metadata as Linked Open Data from DCAT Catalogue as well as from Discrete Global Grid System RDF catalog.

8.6.1.4 Linked Data URI dereferencing

According to W3C recommendation URIs should be dereferenceable. This means that the system should respond to incoming HTTP requests with information about the resources identified by the URI. To meet this requirement we propose to dereference GeoDCAT metadata with physical archives (or proxies containing downloadable scenes, and Discrete Global Grid System metadata to Sentinel HUB serving image data for requested cell.

8.6.1.5 EO Browser

EO Browser – is an advanced, specialized catalogue application allowing user not only to search but also display data in a georeferenced mode. The applications will be extended with the SPARQL client functionality, which will allow user to perform advanced queries using external Linked Data resources

8.6.2 Architecture

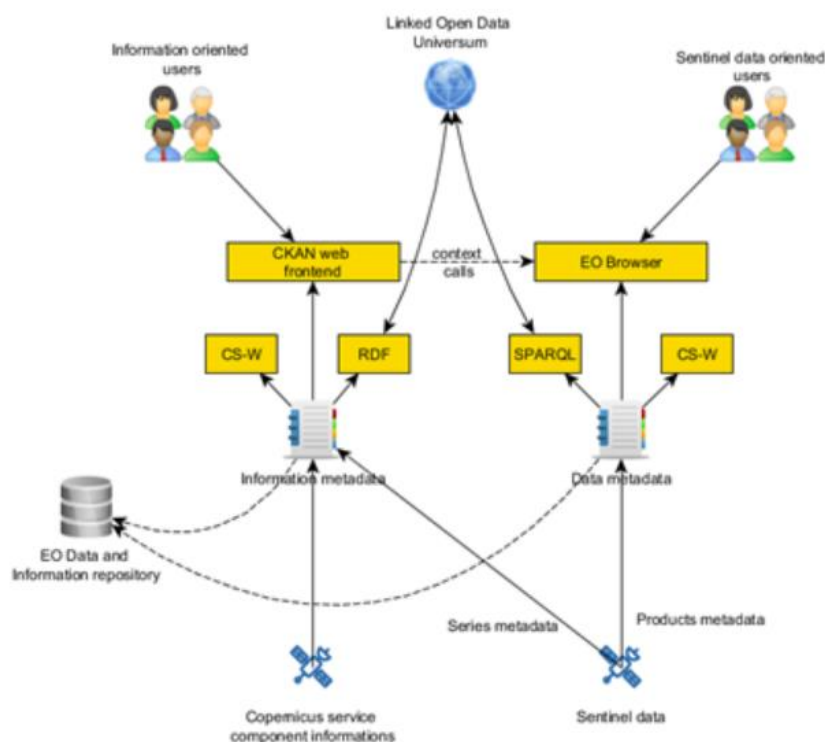


Fig. 11 – CloudFerro Data Collection Catalog architecture

8.6.3 Main use cases

8.6.3.1 Extended geo-search

It's possible to enrich the EO Cloud search engine or alternatively build external EO data explorer utilizing all spatial features stored in LinkedGeoData (LGD). LGD uses the information collected by the OpenStreetMap project and makes it available as an SPARQL endpoint according to the Linked Data principles. LGD is linked to LOD universe via GeoNames and DBpedia nodes. At the moment it stores more than 20 billion triples.

Sample search: find EO data for certain zip-codes, airport codes, points of Interests (POI), addresses, etc.

8.6.3.2 Environmental research

It's possible to streamline research tasks by combining knowledge stored in Wikipedia (published in LOD as DBpedia) with CREODIAS LOD catalogue to retrieve necessary EO data in a single step.

Sample search: find most current EO data for hurricanes sites in US between 1990 and 2017 of a category 4 or above in Saffir–Simpson scale.

8.6.3.3 Emergency response

It is possible to build an application, which could automatically retrieve imagery for sites where severe weather condition may require action of emergency response services. The app could combine data from live RDF weather feed with EO LOD catalogue and automatically access EO data for area/s of interest.

Sample search: find most current non-cloudy S-2 image and most current S-1 imagery for area of wind 80 knots and above.

8.6.3.4 *Image Intelligence support*

It is possible to build an application which could support image Intelligence analysts by automatic retrieval of imagery for the areas mentioned in the news in a certain context (e.g. terrorist attack, natural disaster, troops movements etc.). The app could use RDF news feed available through SPARQL in combination with GeoNames and EO LOD.

8.6.3.5 *External indices database*

It is possible to build an external knowledge base storing historical values of various indices calculated for reference grid cells. The app can calculate value of certain index (e.g. NDVI, NDWI, etc.) and store information in the database. Using EO LOD URI allows the linking of the index value with the grid cell as well as source imagery. The database could help to study annual cycles of indices values and its variations.

8.6.3.6 *Research work enrichment*

It is possible to use same concept as described above in research works to use imagery URI's instead of actual photo to save disk space and give access to a broader picture than just a small window.

8.6.4 **Integration activities**

No integration yet.

8.6.5 **Identified gaps**

No gaps identified.

8.6.6 **Future perspective**

Service is going to be added to the EOSC-hub Marketplace content (see: <https://discovery.creodias.eu/dataset>).

8.7 **CloudFerro Infrastructure**

8.7.1 **Service description**

8.7.1.1 *Virtual Machines*

Virtual Machines (VMs) are fully functional computational instances. They operate as if they were real physical entities with all the elements of a physical server. A user obtains his VM with full root access. He can fully manage it and install any software he has and needs.

In the EO Cloud Users can use Virtual Machines (VMs) by defining their different parameters and characteristics, including machine type (physical or virtual), RAM, CPU (vCores), Storage quantity and type, Operating System, middleware components, Virtual Networks connected to the machine.

Users determine the characteristics of a newly provisioned VM by selecting its flavour and base image.

All the VMs come fully configured (based on the image selected) and ready-for-use, with an administrative User account, network access, preconfigured toolboxes and software components. Volume Storage may be attached to running VM-s to extend the storage space available. VMs can be started, stopped, rebooted, paused, suspended and snapshotted. Live backup functionality is also

available, including server quiescing. VMs may also be attached to Virtual Networks. Virtual Networks may be system-defined or User-defined.

8.7.1.2 Bare Metal Dedicated Servers

Bare Metal Machines are physical servers with no virtualization. There is no any virtualization overhead on those servers. In the CREODIAS Cloud, Bare Metal Machines are nearly as easy to provision and run as the virtual ones. They can be connected to Virtual Networks and especially to EO Cloud network with EO Data. Users can perform actions such as start/stop/reboot/reinstall etc. Some of the standard actions available for VM-s such as snapshot are not available in Bare Metal Machines.

Bare Metal Machines are available for NBD (Next Business Day) provisioning and can be used in several predefined configurations as shown in the pricelist.

8.7.1.3 VM related storage

VM related-storage is a fast solid state SSD storage connected to individual Virtual Machines. It is directly available to the VM without the need for mounting or connecting network shares. The quantity of VM related storage reserved for a VM depends on the VM flavor selected.

VM storage is closely associated with a given VM, which has exclusive access to this type of storage. Once the VM is terminated, its VM storage disappears.

8.7.1.4 Orchestration

The CREODIAS Platform provides Users with an orchestration service to ease and simplify virtual infrastructure deployment and management. User can describe the infrastructure in a template file and deploy/delete it with one action. Templates can contain configuration of Virtual Networks, Virtual Routers, Floating IPs, Security Groups, VMs, Volume Storage and many other Resources together with their parameters to create a fully functional and replicable virtual environment. Templates can also specify relationships between Resources. Orchestration service manages the whole lifecycle of the virtual environment and can apply configuration changes in a smart way (without whole infrastructure redeployment). The OpenStack orchestration service (Heat) is based on the HOT templates (see OpenStack documentation). The Orchestration service can be used via API or via the Cloud Dashboard, it is compatible with that of Amazon Web Services (CloudFormation templates) and it is free of charge.

8.7.2 Architecture

Based on OpenStack, Ceph, Intel Servers, SSD and HDD Disks.

8.7.3 Main use cases

- Storing data in dedicated object storage.
- Listing and using CREODIAS Earth Observation data.
- Providing the continuity of computing important data in Docker or Kubernetes.
- Leasing dedicated servers designed for convenient networking.
- Assuring crucial data accessibility for a bigger group of employees.
- Using a multiplied power of hardware for most resource demanding processes.
- Transferring large amount of information in a closed, secure group of machines.
- Keeping stable performance of Voice over IP processing.
- Bare Metal Dedicated Servers.
- Volume and Object data storage.

- Powerful VMs with accessibility to EO Data.

8.7.4 Integration activities

Published on EOSC-hub Marketplace

8.7.5 Identified gaps

The single sign on and purchase functionality is not implemented yet.

8.7.6 Future perspective

The Service Catalogue, Marketplace and B2FIND integration are planned. Further expansion of the available products and data collections is planned in line with data acquisition roadmap.

8.8 CloudFerro Data Related Services - EO Finder

8.8.1 Service description

It is a catalogue and a search engine dedicated to Earth Observation products. Its main purpose is to handle EO satellite imagery but it can be used to store any kind of geospatialized data.

8.8.2 Architecture

Based on RESTO, CEPH, KeyCloak, RabbitMQ.

8.8.3 Main use cases

The main users are institutions and companies wanting to find various satellite products using various search criteria.

8.8.4 Integration activities

No integration yet.

8.8.5 Identified gaps

No gaps identified.

8.8.6 Future perspective

Service is going to be added to the EOSC-hub Marketplace content (<https://finder.creodias.eu/www>).

8.9 CloudFerro Data Related Services - EO Browser

8.9.1 Service description

It allows visualization and basic processing of selected data collections (like Sentinel-1 L1 GRD or Sentinel-2 L1C)

8.9.2 Main use cases

The main users are companies and institutions wanting to visualize and download Earth Observation products, created by ESA's Sentinel 1, Sentinel 2, Sentinel 3, ESA's archives of Landsat 5, Landsat 7, Landsat 8 and Envisat.

8.9.3 Integration activities

No integration yet.

8.9.4 Identified gaps

No gaps identified.

8.9.5 Future perspective

Service is going to be added to the EOOSC-hub Marketplace content (<https://browser.creodias.eu>).

8.10 Sentinel Hub

8.10.1 Service description

Sentinel Hub service provides immediate on-demand access to satellite imagery data, e.g. Sentinel, Envisat, Landsat, MODIS, etc.

8.10.2 Architecture

SERVICE OVERVIEW

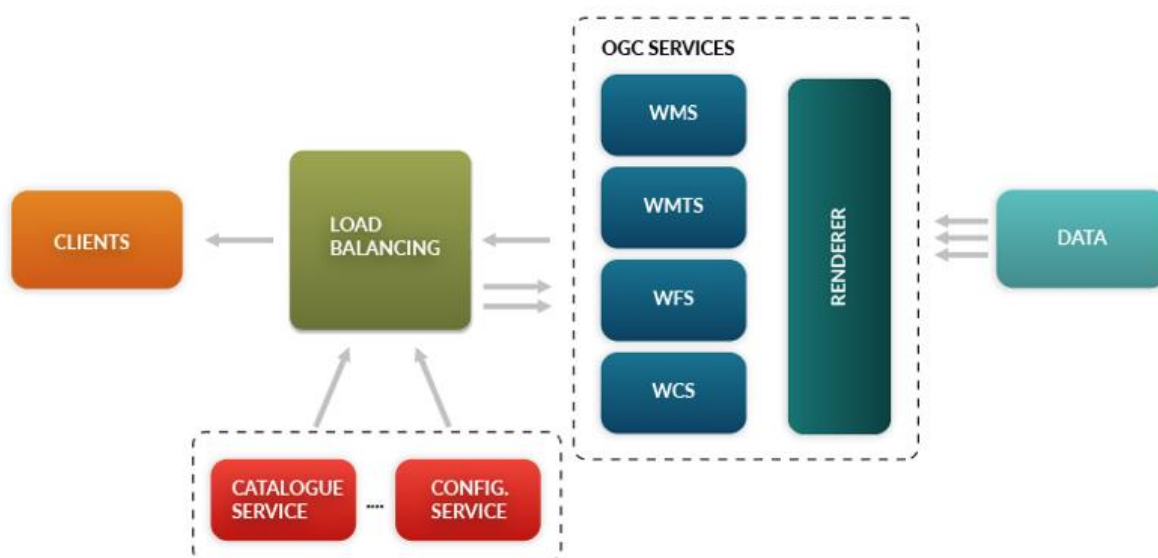


Fig. 12 – Overview of Sentinel-Hub services

The figure above shows the general overview of the Sentinel-Hub services. The data consists of original (raw) data and some additional data typically processed at ingestion time (file indices). The storage links to the data processing and rendering service directly. The users connect to these OGC API services. User's configurations used by the data processing and rendering services are provided by the configuration service. Configurations are, for the moment, managed by the Configuration utility application (WMS configurator). The catalogue and service on top of it is optimized to support fast querying and data retrieval by the rendering service.

8.10.3 Main use cases

The services have a strong focus on the Earth Observation (EO) community, but aim to reach out to further research disciplines. It supports machine learning applications, allowing to “prepare” the data, extracting from the data sources exactly what is desired [27] [28] [29]. With the same tools it is possible to implement web-based applications for precision farming and similar tasks and water resource monitoring [30], like the [BlueDot Water Observatory](#) service. It is based on the [Copernicus](#) satellite imagery that provides timely information about water levels of lakes, dams, reservoirs, wetlands and similar water bodies globally. The key benefit of the service is the accumulation of current and historic global water level data in one place. Because of its cost-effective approach, anyone is able to access water level information freely. Not only water

authorities but also citizens can now better understand the state of their local and global environment.

8.10.4 Integration activities

The service instance offered through the EOSC-hub marketplace was specifically deployed for EOSC users on the CreoDIAS cloud [\[31\]](#), provided by CloudFerro.

8.10.5 Identified gaps

Large area processing: the current computing resources allocated to the service in the CreoDIAS cloud do not allow analysing the big amount of data usually associated to large geographical areas.

8.10.6 Future perspective

- Inclusion of additional data sources.
- Data-cube like capabilities.

9 T7.8 DARIAH

9.1 Service description

DARIAH is a large and heterogeneous community regarding a variety of datasets and data types used, tools and applications utilized in research processes, needs for data archiving and storing, as well as metadata descriptions. The DARIAH TS aims to provide user-friendly solutions (services and applications) addressing the needs of different research groups within the DARIAH and digital arts and humanities in general. Although the DARIAH TS cannot address all the requirements, it provides a set of web-based services enabling end-users to seamlessly store, describe and share their datasets as well as to discover, browse and reuse datasets shared by the others and to perform elemental analysis on those data. The DARIAH TS services are specially tailored for the DARIAH-ERIC community providing free access to its members. More details on the DARIAH TS services refer to deliverable D7.1.

The DARIAH Thematic service provides three independent services:

- DARIAH Science Gateway.
- Invenio-based repository in the cloud.
- DARIAH repository.

9.2 *DARIAH Science Gateway*

9.2.1 Service description

Nowadays, the user communities have broad access to a diverse set of global research infrastructures to perform computations, data manipulation and storing. However, these actions require substantial in-depth knowledge and can be seen as a challenging task for the scientists and end-user, especially from those research domains that are still underutilizing those large, global research infrastructures such as researchers and scholars coming from the arts and humanities domains. To reduce or eliminate these requirements and challenges from DARIAH users, one may use scientific gateways a user-friendly, easy-to-use interface that enables end-users to run their experiments on those research infrastructures without the need to learn the particular features of the underlying infrastructure.

The DARIAH Science Gateway is a web portal based on the WS-PGRADE/gUSE technology that provides a set of generic and customized services and tools that enable end-users to exploit the research infrastructure easily. The gateway, with its modular framework, currently offers the following functions:

- ***Semantic Search Engine (SSE)***
Allows users to search in the e-Infrastructure Knowledge Base (Open Access Document Repositories and Data Repositories) and to discover new correlations about document and data and, ultimately, the creation of new knowledge. The queries to the Semantic Search Engine can be made in more than 110 languages, and the results are ranked according to the latest issue of Ranking Web of Repositories. Moreover, they are connected, whenever the information is available, to Google Scholar and Altmetric to provide users with additional information about versions and citations of a given resource found by the query. SSE exploits

LodLive API to allow users navigate/explore the Linked Data graph for each record found by a search.

- **Parallel Semantic Search Engine (PSSE)**

An extension of the SSE that allows simultaneously search across different online repositories, enabling users to semantically correlate contents in geographically distributed digital repositories across several domains. The service is currently configured to support the following platforms: e-Infrastructure Knowledge Base, Europeana, Cultura Italia, Isidore, OpenAgris, PubMed and DBpedia.

- **Simple Cloud Access**

Simple Cloud Access is a portlet that simplifies workflow creation and execution in the simplest case, i.e. when the workflow contains only one job. The service runs the workflow inside a virtual machine provided by the cloud provider. Its primary purpose is to give a quick and easy demonstration of how to submit and run simple jobs in the cloud environment. A simple use-case describing how to access and harvest the Project Gutenberg collection for a specified author and perform simple word analysis on this corpora is available on Gateway.

- **DBO@Cloud**

DBO is a Cloud-based repository presenting the work of a 100+ year's old collection of Bavarian dialects within the Austrian-Hungarian monarchy from the beginning of German language to nowadays, and the service is based on the gLibrary framework.

More information on the DARIAH Science Gateway and the provided services can be found on DARIAH Competence Centre wiki pages: https://wiki.egi.eu/wiki/Competence_centre_DARIAH.

9.2.2 Architecture

DARIAH Science Gateway is based on the WS-PGRADE/gUSE gateway technology, which is based on the Liferay portlet container framework. This allows science gateway developers to create user-friendly portlets easily. Together with the customization methodologies of WS-PGRADE/gUSE, science gateway developers can create user-friendly portlets for different scientific communities, which help to exploit the available compute and storage resources easily. The gateway provides an API for creating portlets, which are using the services of WS-PGRADE/gUSE, called the Application-Specific Module (ASM API). This API enables science gateway developers to call services of WS-PGRADE/gUSE for importing, modifying and running existing workflows. Parameter-sweep job wizard enables users to run their application processing large input data step by step following six simple steps: executable upload, static input upload, parameter-sweep input upload, command-line argument definition, resource selection, and definition of output files. Finally, Data Avenue is a set of services enabling users to manage their data located on remote storage resources easily. The set of available operations includes browsing, directory creation/renaming/removal, file up- and download, file management (rename/remove/move) and file transfer between different types of storages.

The core technology of the DARIAH SG is based on the following components:

- WS-PGRADE/gUSE science gateway frameworks,
- Application-Specific Module (ASM),
- Parameter-sweep job wizard,
- DataAvenue.

The architecture of the DARIAH Science Gateway includes the following components:

- DARIAH CC VO
- DARIAH CC portal hosting:

- eduGAIN login module
- Application portlets
- Workflow development portlets
- ASM API
- Cloud access portlet
- File transfer portlet

The figure outlines the architecture of the gateway.

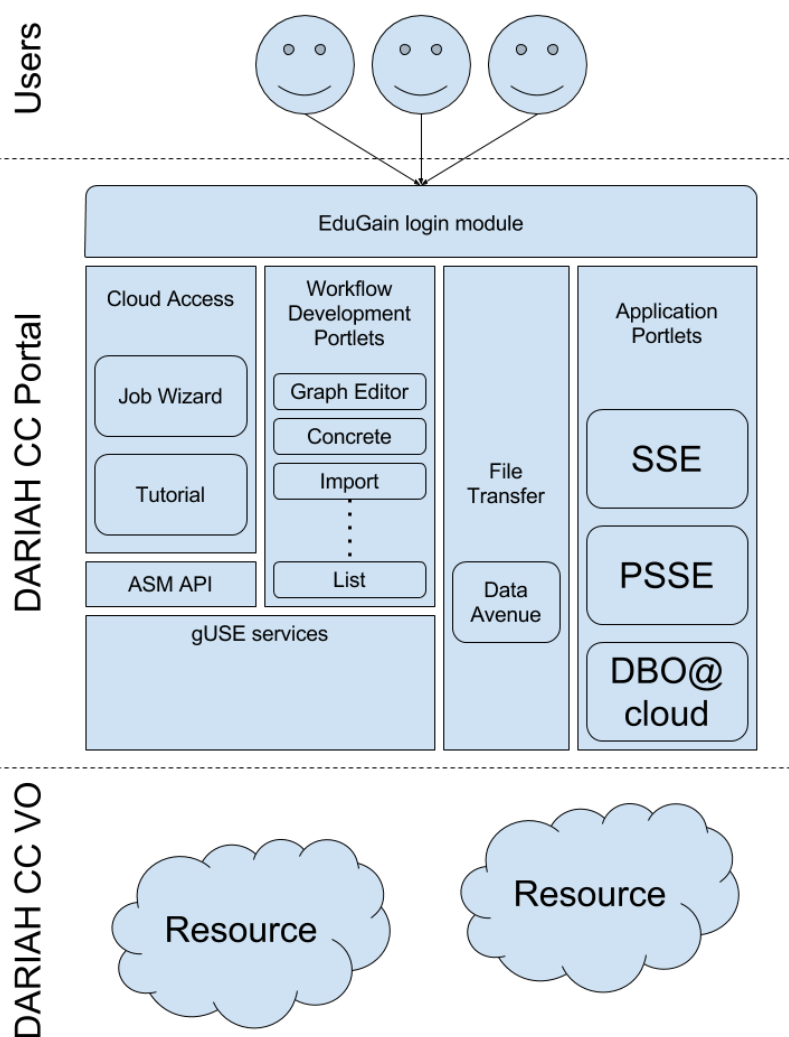


Fig. 13 – DARIAH Gateway architecture

9.2.3 Main use cases

The target users of the gateway are the end-users and scholars coming from digital arts and humanities domain and DARIAH community, which needs user-friendly and easy-to-use services and tools to browse, search and download records from digital collections and open repositories as well as to perform simple data analysis on remote compute resources.

The main use cases are:

- search across various geographically distributed open access repositories from a single point and semantically enrich the search queries (SSE and PSSE),
- search, browse and download in the collection of the Bavarian dialects database (DBO@Cloud),
- perform simple data analysis in the cloud and run a simple job (one executable with input/output parameters and data) in the cloud environment,
- create (complex) workflows and then submit to and execute them on various remote computing environments (cluster, grid, cloud)

The service is a virtualised instance of the WS-PGRADE portal and is hosted by the INFN-Bari FedCloud site. The providers of the DARIAH Science Gateway are MTA SZTAKI (Hungary) and Rudjer Boskovic Institute (Croatia).

9.2.4 Integration activities

In the first project year, the DARIAH Science Gateway was migrated from the local site in MTA SZTAKI to the INFN-BARI OpenStack cloud site. Currently, the gateway is running inside a dedicated virtual machine. The gateway uses the EduGain login module, thus enabling users to authenticate through Shibboleth and log in to the Gateway. Therefore, all users with DARIAH IdP as their identity provider can login to the Gateway.

The DARIAH SG is published in the EOSC-Hub Service Catalogue as well as in EOSC Marketplace.

9.2.5 Identified gaps

No gaps have been identified for the DARIAH Science Gateway compared to the initial plan (M7.1).

9.2.6 Future perspective

During the second project year, the plan foresees the work on integrated login with the Marketplace; in other words, if the user navigates to the Gateway from the EOSC Marketplace or requests the service via the Marketplace, then no authentication should be required from the Gateway. This integration step requires a more detailed technical consideration and is planned to be considered and implemented in the first half of the next project year. The benefits of this integration are that the users of the DARIAH SG will not need to authenticate multiple times, only once at the landing page, i.e. EOSC Marketplace.

9.3 *Invenio-based repository in the Cloud*

9.3.1 Service description

The Invenio-based repository in the Cloud is a web-based service which simplifies the process of configuring, creating, deploying, maintaining and managing the repositories of digital assets from various Arts and Humanities disciplines. The service deploys Invenio-based repository instance in the cloud. The end-user, who wishes to start a new instance of the Invenio-base repository, has to configure the repository by providing the basic options, such as hardware requirements (e.g. the number of CPUs and storage size) which are then automatically allocated in the EGI FedCloud environment and the all required components of the Invenio-based repository deployed. Upon successful deployment of the repository, the repository IP address is returned to the user. The user can then navigate directly to his/her repository starting page available in the given URL (IP). From

there, the user can define user roles for his/her repository, upload new records, browse, search and download, based on the given user roles and access rights.

9.3.2 Architecture

The service is a set of several integrated and connected components and depends on the following components:

- Indigo IAM identity and access management, which is responsible for handling the user authorisation. The users having a DARIAH IdP account can access the service,
- FutureGateway as a web-based user interface that allows users to configure and start deployment of their new repository instances,
- Prepared data repository images from the DockerHub,
- Pre-defined TOSCA template that is if filled with the data provided by the user via FutureGateway. The template describes all the necessary resource requirements and configuration required to deploy a new repository instance,
- Orchestrator,
- Mesos cluster + Marathon framework for resource scaling and container management of long running jobs,
- Invenio-based repository composed of several sub-components packed into five docker images.

9.3.3 Main use cases

The Invenio-based repository focuses on meeting the needs of individual researchers, small research groups and (research) projects that do not have adequate expertise nor in-depth know-how and experience in deploying the Invenio-based repositories or does not have financial support to acquire and maintain the resources required to host a repository.

The service is targeting two different types of users:

- **repository manager** – a user that via the FutureGateway configures, initialize and deploys a new repository instance in the cloud, this is an administrator of the repository instance and have an access to Marathon dashboard from where he/she can monitor the resource allocation, scale the resources and overview the separate Invenio components,
- **end-user** – a user that, once a new repository instance is deployed and running, have access to the repository. Based on his/her access rules can browse, search, upload and download the records from the repository

The main use cases of the service are:

- configure a new repository via a user-friendly web-interface,
- automatically deploy a repository instance in the cloud without in-depth knowledge in the cloud resource allocation or repository deployment process,
- monitor the repository instance status via web-browser,
- scale the resources (manually or automatically) based on the load (hardware requirements, storage volume, number of users, etc.).

The resource (compute and storage) provider is INFN-BARI, which provides enough resources for one repository instance. These are the test-bed resources. The plan is to make a better integration with the EOSC Marketplace, which will allow the service to automatically request compute resources from the Marketplace, under categories 'Compute', subcategory 'EGI Cloud Compute'.

9.3.4 Integration activities

The integration is still in progress.

The authentication is done via the Indigo-IAM service. A new instance of Indigo-IAM service is launched for the DARIAH TS as an IAM-as-a-Service and is hosted at the INFN-BARI site. Currently, the FutureGateway and the Orchestrator are in the process of deploying on the INFN-BARI cloud site. The plan is to finalize the deployments and integration by the end of 2018.

9.3.5 Identified gaps

We have experience some delays in deploying the Invenio-based repository service, which was planned for the end of September 2018 (see M7.1). The main reason was technical problem with deploying services (IAM, Orchestrator) and connecting them together. These technical problems delayed integration for a few months but did not jeopardize the entire integration process, which will be finished on time.

9.3.6 Future perspective

The future integration activities involve integration with the EOSC Marketplace. There are two integration ideas:

1. **Basic integration with the Marketplace.** The Invenio-based repository in the cloud will be published in the EOSC Marketplace such that the users, landing at the Marketplace can order and navigate (redirect) to the Invenio-based repository service. This integration includes the mutual authentication, which will allow users to log in once, at the Marketplace, and from there navigate to the service without further logins required. This deadline for this action is June 2019. The benefit of this integration will be an increased visibility of the service and a simpler access to the service for end users, which require to authenticate only once.
2. **Advanced integration with the Marketplace.** The Invenio-based repository integrates with the Marketplace allowing users to parametrize and configure a new repository instance (e.g. storage size, number of CPUs) from the Marketplace, without a need to redirect to the service configuration (remote FutureGateway site). This integration requires a more detailed technical consideration. At the end of this integration process, the Invenio-based repository in the cloud would be offered via Marketplace as an Application-as-a-Service with possible parametrization before launching.
From the end-user point of view, this integration will present a more transparent and simple process of deploying a new repository instance in the cloud, without an unnecessary navigation between Marketplace and the service's landing page. For the service provider point of view, this solution will decrease the maintenance effort, since no need to operate a separate, remote site (FutureGateway) that only provides a template for parameterizing a new repository instance.

9.4 DARIAH Repository

9.4.1 Service description

The DARIAH repository is provided by the German DARIAH-DE project. It is a digital long-term archive for human and cultural-scientific research data. The repository is a central component of the DARIAH Research Data Federation Infrastructure, which aggregates various services and applications and can be used comfortably by DARIAH users. The repository allows users to save their research data in a sustainable and secure way, to annotate it with metadata and to publish it. The collections

as well as each individual file are available in the DARIAH Repository and get a unique and permanently valid Digital Object Identifier (DOI) and EPIC PID through which the data can be permanently referenced and cited. In addition, users can register their collections within the DARIAH Collection Registry, which are then searchable through the DARIAH Generic Search.

9.4.2 Architecture

The overall architecture of the DARIAH Repository is depicted in the figure below. The repository consists of a number of sub-services:

- CRUD Service
- Publish Service
- PID Service
- OAI-PMH Service

The CRUD service is the core storage service for the DARIAH Repository. It talks to both the DARIAH OwnStorage (private) and PublicStorage (public). The CRUD service is used either via the Publish Service or via its own API (see also <https://repository.de.dariah.eu/doc/services/submodules/tg-crud/service/dhcrud-webapp-public/docs/index.html>). The main purpose of the CRUD service is to offer Create, Read, Update and Delete operations on data objects.

The Publish Service is the backend of the so-called Publikator (<https://repository.de.dariah.eu/publikator/mainView>). This is the entry point for importing collections, which allows users to prepare and manage their data objects for import into the DARIAH Repository (and for later publication). A collection created through the Publikator subsumes a number of related data objects that can be annotated with metadata. The Dublin Core standard is used there.

Each object published in the DARIAH Repository has got two persistent identifiers, which are created during import process: a DataCite DOI for citation, and an EPIC Handle PID for administrative use. DataCite DOI and EPIC Handle prefixes are institution specific, and the suffixes for DARIAH Repository DOIs and Handles are just same, such as:

- 10.20375/0000-000B-C8EF-7 (DataCite DOI)
- 21.11113/0000-000B-C8EF-7 (EPIC Handle)

Both DataCite DOIs and EPIC Handles can be resolved by any DOI or Handle resolver. With each EPIC Handle some administrative metadata is stored, that is providing access to all the object's data and metadata stored in the repository, and an URL that points directly to the object in the DARIAH-DE PublicStorage.

Each object is stored as a Bagit bag in the DARIAH-DE PublicStorage, where it can be accessed publicly. Access to the repository's objects is provided using HTTP content negotiation with the basic DOI or Handle. You can get:

1. The complete bag (as ZIP) setting the HTTP Accept-Header to application/zip.
2. The HTML landing page if requesting text/html.
3. The data object itself otherwise.

The DARIAH OAI-PMH Service is the service to harvest all metadata from the collections stored in the DARIAH-DE Repository. So the Generic Search (<https://search.de.dariah.eu/search/>) can index all the data that is entered into the Collection registry (<https://colreg.de.dariah.eu/colreg-ui/>).

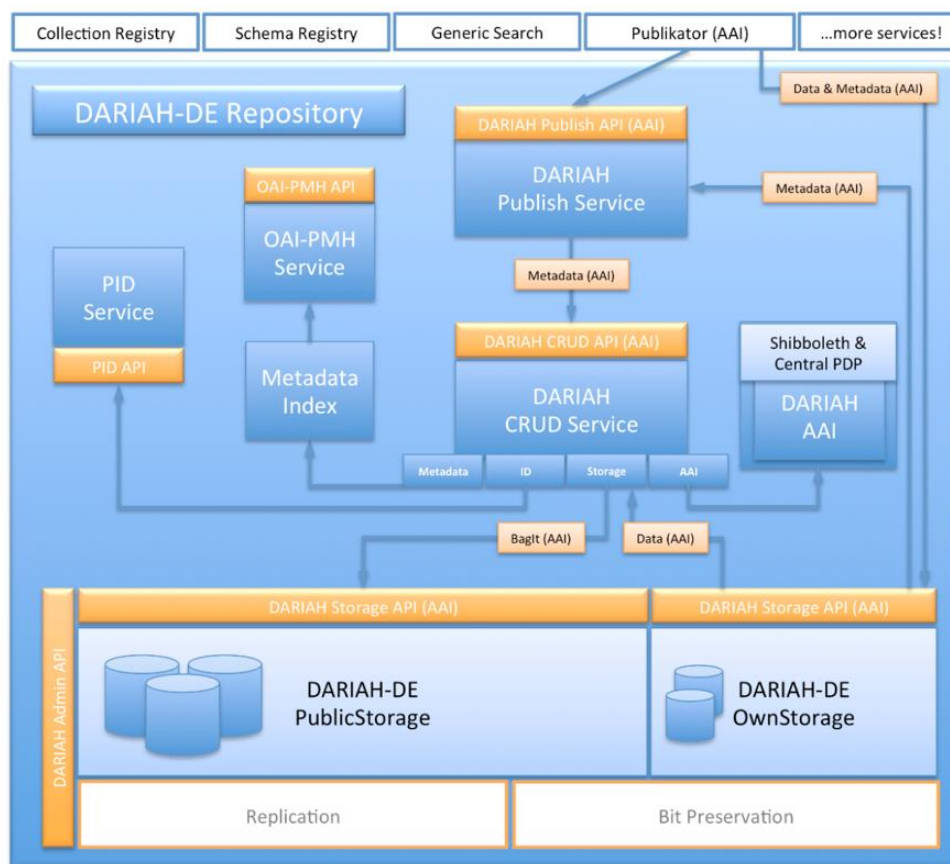


Fig. 14 – DARIAH Repository architecture

9.4.3 Main use cases

The DARIAH Repository is a service for the Arts and Humanities to store research data sustainably and securely, to add metadata to it and to publish it. Furthermore, collections and data objects receive unique and permanently valid persistent identifiers (DOI and EPIC PIDs) for long term identification and citation. So the main use cases are:

- Storing collections and data objects in a secure and sustainable repository specifically for the Arts and Humanities domain.
- Receiving a Persistent Identifier for reference and citation.
- Making research data publicly available.
- Adding collections to the DARIAH Collection Registry.
- Making data objects findable through the DARIAH generic Search.
- Making data available for subsequent use thus feeding it into the public research cycle.

9.4.4 Integration activities

The DARIAH TS is one of the Thematic Services that has been identified as a primary target for the integration of EOSC-hub and OpenAIRE Advance services. Following an initial meeting on February 9, 2018, in Pisa, Italy, between the two projects, which included a representative from the DARIAH TS, the integration of the DARIAH Repository with OpenAire Advance has been further planned by the DARIAH TS. A concise integration plan has been developed that include

- Specification and implementation of integration work-flows
- Implementation of metadata transformation processes
- Extension of the OAI-PMH Service to cover all metadata

- Testing and documentation

The plan implementation should start during the second year.

9.4.5 Identified gaps

No gaps identified.

9.4.6 Future perspective

With the growing number of services integrated it is essential for the thematic services to continuously evaluate integration potential. Regarding the DARIAH TS there is specifically potential regarding complementary services from the CLARIN TS.

10 T7.9 LifeWatch

The LifeWatch Thematic Service's integration activity has been on hold since July. This is due to an administrative issue internal to the community itself. The project office is investigating possible solutions.

11 Future plans

The work plan of the Thematic Services (TSs) was defined at the beginning of the year in M7.1[22] (Thematic Services Integration plan), but it is not static, it changes and the updates are reflected on a shared document[23]. Some Thematic Services activities have been delayed because of a delay in the provisioning of the required computing or storage resources, others due to a slow allocation of the human resources aimed at supporting the service integration. In this group are included GEOSS, DARIAH and EO Pillar. Other TSs have changed their schedule to cope with the unavailability of some required feature in the EOSC-hub ecosystem, but without delaying the overall task thanks to a re-shuffling of their sub-tasks. For this reason DODAS moved the accounting integration to the next year and the same did ECAS with the EOSC-hub AAI and the integration with B2SHARE. While CLARIN delayed the integration with B2SHARE because of it was involved in the EOSC Portal launch, whose preparation required more time than expected.

The following integration topics are those included in the plans for the next year:

- **CLARIN:**
 - *Data:* B2DROP.
 - *Security:* EOSC-hub AAI.
 - *Accounting and Monitoring.*
- **DODAS:**
 - *Security:* IAM
 - *Accounting and Monitoring.*
 - Improved flexibility to support new communities
- **ECAS:**
 - *Security:* EOSC-hub AAI.
 - *Data:* B2SHARE, OneData, B2HANDLE.
 - *Compute:* EGI-FedCloud.
 - Collaboration with OPENCoastS
- **GEOSS:**
 - *Security:* EOSC-hub AAI.
 - *Compute:* EGI-FedCloud.
- **OPENCoastS:**
 - *Compute:* EGI-FedCloud, DIRAC4EGI.
 - *Data.*
- **WeNMR:**
 - *Data:* EUDAT services.
- **EO Pillar:**
 - *Security:* EOSC-hub AAI.
 - *Compute.*
 - Publishing on the Marketplace.
 - Integrating new data sources.

- **DARIAH:**
 - *Security*: EOSC-hub AAI.
 - Collaboration with CLARIN
- **LifeWatch:**
 - It is not possible at the moment to provide a clear roadmap, since the task's integration activities are suspended, as explained in the Executive Summary.

12 References

No	Description/Link
1	https://wiki.eosc-hub.eu/display/EOSC/D7.1+First+Thematic+Service+software+release
2	https://www.openarchives.org/pmh/
3	http://www.language-archives.org/
4	https://pro.europeana.eu/resources/standardization-tools/edm-documentation
5	https://mopinion.com
6	https://icinga.com
7	https://www.statuscake.com
8	http://www.helix-nebula.eu
9	https://tools.ietf.org/html/rfc7662
10	https://www.mod-auth-openidc.org
11	https://github.com/ECAS-Lab/ecas-notebooks
12	http://www.ecopotential-project.eu
13	https://vlab.geodab.eu
14	https://marketplace.eosc-portal.eu/services/opencoasts-portal
15	https://www.eosc-hub.eu/catalogue/WeNMR%20suite%20for%20Structural%20Biology
16	https://eosc-hub.eu/catalogue/EGI%20High-Throughput%20Compute
17	https://eosc-hub.eu/eosc-in-practice-wenmr
18	Updated enmr.eu VO SLA and OLAs, https://documents.egi.eu/document/2751
19	WeNMR Thematic Service AAI report,EOSC-Hub Week, Malaga,16-20 April 2018: https://indico.egi.eu/indico/event/3903/session/32/material/0/4.pdf
20	https://github.com/onedata/oz-worker/pull/1
21	https://github.com/onedata/oz-gui-homepage/pull/1
22	https://wiki.eosc-hub.eu/display/EOSC/M7.1+Thematic+Services+Integration+plan
23	https://docs.google.com/spreadsheets/d/1C0T2oOBHqN8esfJp8q08fqYikIL8yLCSdkA8MvQSPYA/edit?usp=sharing
24	https://github.com/DODAS-TS
25	https://oceanos.grnet.gr/home
26	https://oceanos-knossos.grnet.gr/home
27	https://medium.com/sentinel-hub/land-cover-classification-with-eo-learn-part-1-2471e8098195
28	https://medium.com/sentinel-hub/introducing-eo-learn-ab37f2869f5c

29	https://medium.com/sentinel-hub/sentinel-hub-cloud-detector-s2cloudless-a67d263d3025
30	https://medium.com/sentinel-hub/bluedot-eo-solution-for-water-resources-monitoring-d7663c21af16
31	https://creodias.eu