Integrated Arctic Observation System

Research and Innovation Action under EC Horizon2020
Grant Agreement no. 727890

Project coordinator:
Nansen Environmental and Remote Sensing Center, Norway

Deliverable 5.8

iAOS Platform and Tools
V2

Start date of project:	01 December 2016	Duration:	60 months
Due date of deliverable:	31 May 2020	Actual submission date:	10 June 2020
Lead beneficiary for preparing the deliverable:		TERRADUE	
Other contributing partners		AWI, ARMINES	
Person-months used to produce deliverable:		0,5 pm	

Authors: Hervé Caumont (Terradue), Ingo Schewe (AWI), Fabien Ors (ARMINES)

Reviewers: Pedro Goncalves (Terradue), Torill Hamre (NERSC), Hanne Sagen (NERSC)

Version	DATE	CHANGE RECORDS	LEAD CONTRIBUTOR
0.1	24/04/2020	Structure updates from D5.2	Hervé Caumont
0.2	20/05/2020	Version for internal review	Hervé Caumont
0.3	25/05/2020	First internal review	Pedro Goncalves
1.0	26/05/2020	Version submitted on EMDesk for approval	Hervé Caumont
1.1	29/05/2020	Second internal review	Hanne Sagen, Torill Hamre
1.2	03/06/2020	Version submitted on EMDesk for approval	Hervé Caumont
1.3	05/06/2020	Third internal review	Hanne Sagen, Torill Hamre
1.4	10/06/2020	Final version submitted on EMDesk	Hervé Caumont
1.5	19/06/2020	Technical review and submission to EU	Kjetil Lygre

Approval **X**	Date: 19 June 2020	Sign. Stein Sandven Stein Sandven, Coordinator

USED PERSON-MONTHS FOR WRITING THIS DELIVERABLE					
No	***Beneficiary***	***PM***	***No***	***Beneficiary***	***PM***
1	**NERSC**	**-**	**24**	**TDUE**	***0,3***
2	UiB		**25**	GINR	
3	**IMR**	**-**	**48**	UNEXE	
4	MISU		**27**	NIVA	
5	**AWI**	***0,1***	**28**	CNRS	
6	IOPAN		**29**	U Helsinki	
7	**DTU**	**-**	**30**	GFZ	
8	AU		**31**	**ARMINES**	***0,1***
9	**GEUS**	**-**	**32**	IGPAN	
10	**FMI**	**-**	**33**	U SLASKI	
11	UNIS		**34**	BSC	
12	NORDECO		**35**	DNV GL	
13	SMHI		**36**	RIHMI-WDC	
14	USFD		**37**	NIERSC	
15	**NUIM**	**-**	**38**	WHOI	
16	IFREMER		**39**	SIO	
17	MPG		**40**	UAF	
18	EUROGOOS		**41**	U Laval	
19	EUROCEAN		**42**	ONC	
20	UPM		**43**	NMEFC	
21	UB		**44**	RADI	
22	UHAM		**45**	KOPRI	
23	NORCE		**46**	NIPR	
			47	PRIC	

DISSEMINATION LEVEL		
PU	Public, fully open	**X**
CO	Confidential, restricted under conditions set out in Model Grant Agreement	
CI	Classified, information as referred to in Commission Decision 2001/844/EC	

EXECUTIVE SUMMARY

The “Integrated Arctic Observation System” (INTAROS) is a 5-year project funded by Horizon 2020 under the Blue Growth Programme. The overall objective of INTAROS is to build an efficient integrated Arctic Observation System (iAOS) by extending, improving and unifying existing systems in the different regions of the Arctic. Within INTAROS, WP5 (Data integration and management) is tasked with designing and implementing evolutions of the cloud platform and tools for the iAOS.

This document defines version 2 of the **iAOS cloud platform and tools**. The cloud platform has the objective to integrate multidisciplinary and distributed data repositories and provide a set of tools for data analysis, transformation and visualization. The cloud platform offers services to access observations and derived parameters, including the new observations from WP2-3-4. In addition, it offers seamless integration with geo-statistical methods for interpolation of spatiotemporal datasets, and services to store the generated datasets in iAOS enabled repositories

The cloud platform offers scalable allocation of data storage and computer power for integrative data processing and analysis. Selected applications established in WP6 will use the services and tools provided by the platform, and thereby demonstrate the usefulness and functionality of the platform. These applications will showcase the benefit of enhanced integration of data from Arctic observing systems. Examples of services developed using the cloud platform are presented in D5.7. Processing Services for iAOS v1.

The iAOS cloud platform provides tools to users with essential functionality for data discovery view, data download, data transformation (e.g. sub-setting, projection, interpolation), and facilitate for development and streamlining data processing workflows. The platform tools address the challenges of Big Data (Volume, Velocity, Variety, Veracity, Value). The cloud platform architecture is supported by a federated cloud infrastructure to facilitate the management of the applications’ lifecycle (integration, validation and deployment in production).

The version 2 (D5.8) of the iAOS cloud platform and tools presented in this deliverable augments the version 1 (D5.2) with the following descriptions:

- Clarifications on the scope of the iAOS cloud platform within the overall iAOS;
- New section presenting the evolutions of the Cloud Platform Software Development Kit (SDK) Handbook;
- Enhanced user guidance as part of the sections presenting the Cloud Platform's Ellip Solutions portfolio;
- More detailed tools descriptions and examples for the developer user guidance in section 3. “Catalog client tools” (OpenSearch Catalog client, OpenDAP/THREDDS Catalog client, Pangaea Catalog client).

Table of Contents

Acronyms and abbreviations

API	Application Programming Interface
CPU	Central Processing Unit
DevOps	Development to Operations
DIAS	Data and Information Access Service
EGI	European Grid Infrastructure
EO	Earth Observation
FRAM	FRontiers in Arctic marine Monitoring project
HTTP	HyperText Transfer Protocol
IaaS	Infrastructure-as-a-Service
iAOS	integrated Arctic Observation System
ICT	Information and Communication Technologies
IDE	Integrated Development Environment
NERSC	Nansen Environmental and Remote Sensing Center
OGC	Open Geospatial Consortium
OpenDAP	Open Data Access Protocol
OSDD	OpenSearch Description Document
OWS	OGC Web Services
PaaS	Platform-as-a-Service
SDK	Software Development Kit
THREDDS	Thematic Real-time Environmental Distributed Data Services
VM	Virtual Machine
VPN	Virtual Private Network
WMS	Web Map Service
WPS	Web Processing Service

Definitions

Application
A program or group of programs designed for end users. An application, application program or application software is a computer program designed to help people perform an activity.

Infrastructure as a Service (IaaS)
IaaS is a standardized, highly automated offering, where compute resources, complemented by storage and networking capabilities are owned and hosted by a service provider and offered to customers on-demand. Customers are typically able to self-provision this infrastructure, using a Web-based graphical user interface that serves as an IT operations management console for the overall environment. Application Programming Interface (API) access to the infrastructure may also be offered as an option.

Platform as a Service (PaaS)
PaaS provides users with the capability to deploy user-created or acquired applications developed using programming languages and tools supported by the provider onto an infrastructure, typically cloud- based. The user does not manage or control the underlying infrastructure, including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

Service
When an application software has been integrated and tested for cloud computing operations, it is deployed on production servers and made available as a "service": the application functionality can be delivered via a simple invocation interface, making the full application functionality available to others as a service of the iAOS.

Software as a Service (SaaS)
SaaS is characterizing application software that is owned, delivered and managed remotely by one or more providers. Typically the application provider delivers the software based on one set of common computing code and data definitions, that is consumed in a one-to-many model by all contracted customers, at any time, and on a pay-for-use basis or as a subscription based on the monitoring of service usage metrics.

Software Development Kit (SDK)
A set of software tools installable on the developer Integrated Development Environment (IDE) and usable as software dependencies in the developer's application code.

1. Introduction

This document provides guidelines for the integration and deployment of applications using observations made available through the overall iAOS, leveraging the cloud computing environment provided by Terradue. It describes in detail how INTAROS partners can use the cloud platform and tools to build services using a federation of distributed compute and data resources.

The guidelines include step-by-step descriptions that explain the Cloud Platform usage from the perspective of service providers who develop new processing services. The aim is to support coming users of the iAOS cloud platform, enabling them to deploy services according to identified requirements for the access, processing, and integration of data.

1.1. Purpose and scope of the document

This document covers the activities of the INTAROS Task 5.2 “IAOS platform deployment and operation”, with a focus on the Cloud-based exploitation environment (Ellip Solutions) delivered and operated by Terradue, and on the data access and data processing tools that are made available on this environment to application providers (e.g. ARMINES, NERSC in Tasks T5.4 and T5.5) in order to make use of the integrated data from existing iAOS repositories (as outcome of Task 5.2).

1.2. Intended audience of this document

The main target audience of this document is made of the INTAROS Consortium Partners (especially the WP5 partners integrating processing services and the WP6 partners developing showcase applications) and European Commission services.

1.3. Document Structure

This section describes the document structure:

Section 1 is this introduction section.

Section 2 provides application developer guidance for their Processing Service Integration & Deployment activities in INTAROS. It presents how to access and use the Ellip Solutions and provides additional information about Terradue Cloud Platform.

Section 3 describes the tools made available to application developers when integrating their software application with using the integration environment provided by Ellip.

1.4. What is the iAOS?

The overall objective of INTAROS is to develop an integrated Arctic Observation System (iAOS) by extending, improving and unifying existing and emerging systems in the different regions of the Arctic. The overall iAOS is comprised of observing infrastructures, data systems managing the data, and platforms for integration, service development and deployment. The iAOS cloud platform in INTAROS utilizes data collections from existing and new observing systems made available through online data repositories. The cloud platform and the cloud computing infrastructures are made available by Terradue.

A synoptic view of the logical connections and integration of main iAOS data systems and computer services with the iAOS cloud platform are shown in **Figure 1**. It shows how the iAOS cloud platform supports the development and maintenance of processing services including data access, integration, and analysis, using the cloud platform “Ellip Solutions” operated by Terradue to empower these processing services with:

- A service integration environment, where the processing services are under test and validation running on a “Development Infrastructure”;
- A service deployment environment, where the processing services are deployed in production running on a highly scalable “Hosting Infrastructure”;
- An embedded capacity for each Ellip-powered processing service to interact, whatever the infrastructure it runs on, with remote repositories and Web Portal(s).

INTAROS partners have used the Cloud platform to develop showcase applications e.g. Geostatistics in Task 5.4 (ARMINES), processing and analysis of acoustic data in Task 5.5 (NERSC) and other services provided by WP6 partners.

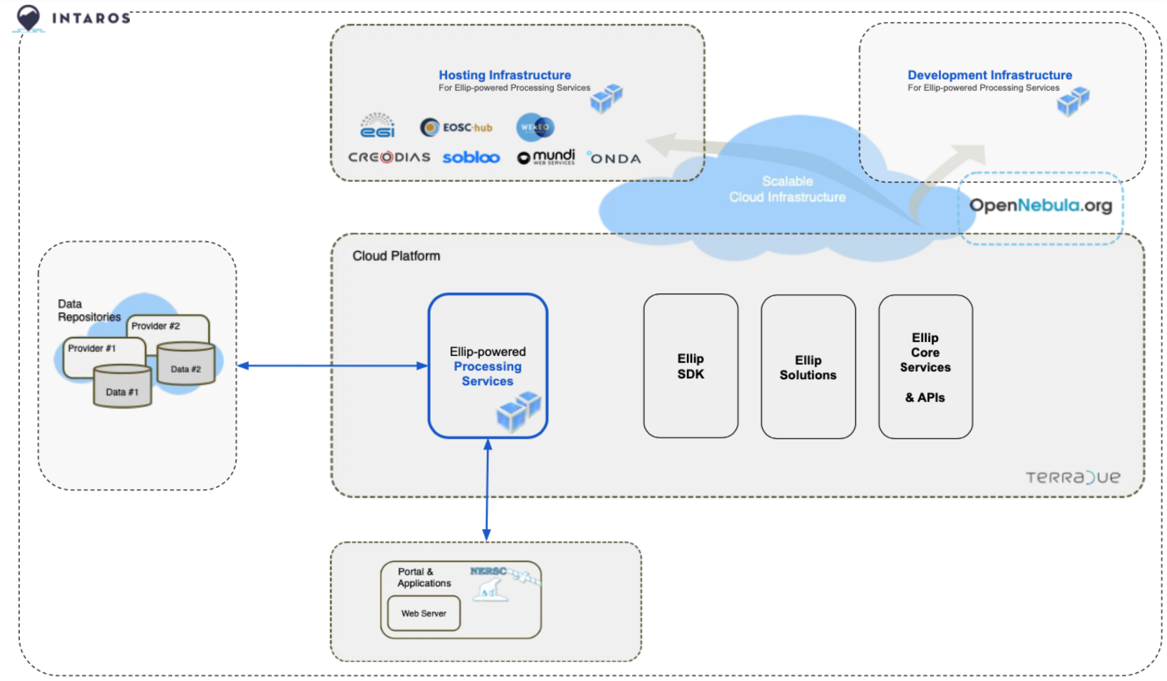


Figure 1. Synoptic view of how the Cloud platform is linked to other components of the iAOS e.g. data repositories, iAOS portal, as well as development and deployment (hosting) infrastructures.

The architecture of the iAOS cloud platform builds on standards, technologies and tools for data discovery, data access, scalable data processing, data visualization, Cloud resources provisioning, and APIs for integration of the iAOS Portal & Applications delivered via the Portal.

Going into some more details, we can split these main building blocks into a few sub-groups.

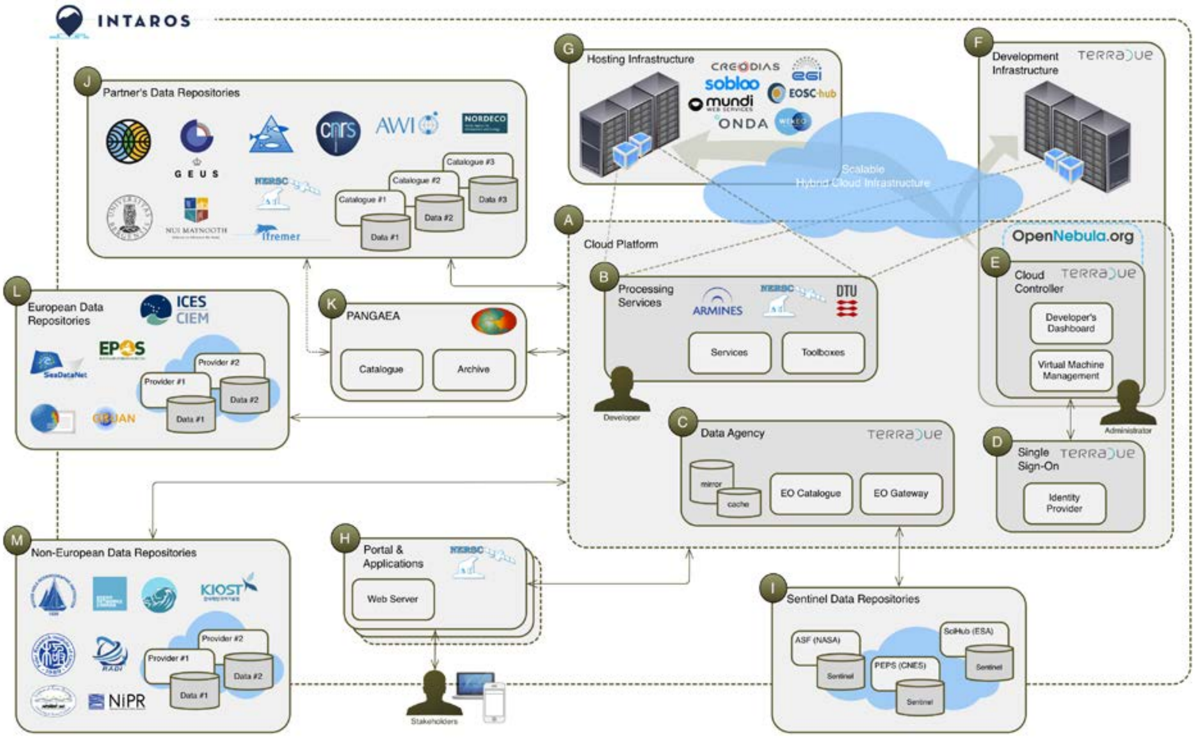


Figure 2. The architecture of the Information and Communication Technologies (ICT) components of the overall iAOS.

In **Figure 2** the ICT components of the iAOS are depicted in more details, where each of the elements are labeled A to M in the diagram:

- **(A)** is the set of iAOS Cloud Platform services, a Collaborative workspace operated as a domain hosted on Terradue Cloud Platform, and accessed by the authorized partners via the use of the Ellip Solutions, described in the next section of this document.

- **(B)** are the software toolboxes and the processing services maintained on the Platform over their full lifecycle, from the initial versions for integration and testing to the released ones, deployed in operation on production servers.

- **(C)** is the set of Catalog and Data Gateway services in charge of the data discovery and access, enabling to feed processors with specific collections or product types, based on dataset files mirroring, caching and remote access strategies. It includes mechanisms for data registration. The Data Agency supports programmatic discovery and access to distributed EO data repositories for scalable data processing services. Processing Service developers can request data mirroring operations into their allocated Cloud storage on the Platform (data buckets), and exploit a standard interface to define data staging operations from the referenced repositories (mirrored or remote) to their Application processing nodes.

- **(D)** is the Identity provider for the management of Platform user accounts, and the Single sign--on (SSO) service for authentication and authorization across all the Platform resources federated under the SSO.
- **(E)** is the Cloud Controller service performing the management of Virtual Machines allocated to the iAOS platform domain and users, and providing a Cloud Dashboard interface for Platform users to create (within their quota), stop and restart their Virtual Machines, as well as to monitor the allocated Cloud resources to their VMs (CPU, RAM, Disk layout, also within user quota). The Cloud Controller operates both the application integration environment where applications or services are first integrated and validated as “Cloud Appliances”, and the Production Centers in which applications are moved for their exploitation on a Cloud Computing infrastructure.
- **(F)** and **(G)** are the 'background facilities' running the services through their lifecycle, from integration, testing and packaging (the main purpose of the ‘Development’ private Cloud environment) to Hosted Processing operations (the main purpose of the ‘Hosting” environment using public Cloud providers, commercial or institutional).
- **(H)** is defining the ‘end-user exploitation’ environment (available at the iAOS Portal), from where users can access federated data repositories, as well as the Processing Services integrated and deployed on the Platform.
- **(I)** is the pool of Copernicus Sentinel distributed data repositories (e.g. Copernicus scientific hub, Copernicus collaborative ground segments) that are federated on Terradue Cloud Platform, thus available for access from the iAOS processing services.
- **(J)** is the pool of INTAROS Partner’s data repositories that have been federated into the iAOS via interoperability arrangements (data and metadata formats, online access protocols).
- **(K)** is the pool of the PANGAEA data repository, along with the PANGAEA catalog entries, that have been federated into the iAOS cloud platform via interoperability arrangements (data and metadata formats, online access protocols).
- **(L)** is the pool of European data repositories with relevance for the Arctic areas that have been federated into the iAOS cloud platform via interoperability arrangements (data and metadata formats, online access protocols).
- **(M)** is the pool of Non-European data repositories with relevance for the Arctic areas that have been federated into the iAOS cloud platform via interoperability arrangements (data and metadata formats, online access protocols).

Overall, the iAOS cloud platform and its access to distributed databases will enable application developers and data providers to collaboratively work on service development of relevance for communities and operators in the Arctic. The development of a federated iAOS cloud computing environment depends on:

- A network of interoperable databases, enabling search and access to multidisciplinary data, with links to relevant Copernicus and European Space Agency programmes (CMEMS, ESA CCI, …);

- Access to existing multidisciplinary observing systems (in situ and space) and their data repositories from Europe, the USA, Canada and other international partners, through open standards and protocols;
- Availability of Arctic data products that are compliant to free and open data policies, in agreement with the GEO data sharing principles.

1.5. What is the iAOS cloud platform?

The iAOS cloud platform provides a set of services and tools for data processing, data analysis, and data visualization with the objective to integrate multidisciplinary and distributed data repositories.

The iAOS cloud platform offers access to observations and derived parameters and seamless integration with geo-statistical methods for interpolation of spatiotemporal datasets, including the new observations from WP2-3-4, and store the generated datasets in iAOS enabled repositories.

In the sections 1.5.1. and 1.5.2. hereafter, we introduce the key aspects of the cloud platform for iAOS: the use of the Ellip Solutions as the work environment for the INTAROS partner users, for them to integrate and deploy data processing services, and the access to the INTAROS distributed repositories based on interoperability standards.

1.5.1. Using the Ellip Solutions to integrate and deploy iAOS processing services

Here we introduce the services previously referenced in the **Figure 2** elements from labels (A) to (G). The related iAOS Platform and tools (platform services and tools currently in scope to support iAOS partners) are described in the **section "2. Processing Service Integration & Deployment Guide"**.

The Ellip Solutions[1] are accessed online by the INTAROS partners, as referenced from the **Figure 2** element (A) "Cloud Platform".

The Ellip Solutions are operated by the INTAROS partner Terradue. With the Ellip Solutions, authorized users are provided with specialized Cloud services to build, maintain, deploy in production, run and monitor data processing services, using production servers from selected Cloud Computing providers. The Ellip Solutions provide tools to interact with the Cloud Platform APIs enabling key operations such as:

- Accessing and installing specialized software toolboxes (e.g. a Geostatistical library), integrating, maintaining and packaging processing services, as referenced from the **Figure 2** element (B) "Processing Services". More details are provided in the INTAROS Deliverables:
    - **D5.6 - Geostatistical library for iAOS V1** (delivered)
    - **D5.10 - Geostatistical library for iAOS V2** (upcoming)

[1] https://www.terradue.com/ellip

  - **D5.7 - Integration of new processing services V1** (delivered)
  - **D5.11 - Integration of new processing services V2** (upcoming)
- Performing data discovery and access from an application code, as referenced from the **Figure 2** element (C) "Data Agency" for EO data access, (K) for the Pangaea catalogue resources and (J, L, M) for other catalogue providers.

The user authentication mechanism for accessing the Cloud Platform services is managed on the Platform, as referenced from the **Figure 2** element (D) "Single Sign-On".

With the iAOS Platform, a hybrid Cloud approach is moreover supported, as referenced from the **Figure 2** elements (E, F, G). This hybrid Cloud approach is providing state of the art Cloud services for implementing, deploying and running applications. The primary purpose of such a hybrid Cloud Platform is to facilitate the management of elastic compute resources, with scale-out capabilities delivered at low cost.

The Ellip Solutions rely on the concept of an application integration environment (PaaS, or Platform-as-a-Service) and a production environment:

- **Integration environment**: Private Cloud infrastructure with specialized services for application integration, test, and packaging (PaaS from Terradue Cloud Platform)
- **Production environments**: Public Cloud infrastructures providing CPU and Storage as-a-Services (IaaS from EGI.eu federated Cloud, Copernicus DIAS providers, etc...)

The transition from the integration environment to a selected production environment is technically supported by Terradue Cloud Platform (Cloud bursting APIs).

1.5.2. Accessing iAOS federated remote repositories

Here we introduce the "data repository" services previously referenced in the **Figure 2** elements from labels (I) to (M). To make use of the data from selected online data repositories, tools have been developed and are detailed in **"3. Client software tools"**.

The iAOS approach to federate distributed repositories is based on interoperability conventions. These conventions must be implemented on both the repository provider side (data access protocols) and the data consumer side (catalogue services, client software tools). The technical implementations rely on standard APIs.
The added value for the iAOS cloud platform to access federated remote repositories is that users can access data from multiple data stores from a single portal or client application, while the data remains within the control and ownership of the data provider.

More details on the distributed repositories operated by the iAOS partners (or by other organizations in charge of data management activities relevant for iAOS) are provided in the **INTAROS Deliverables:**

- **D5.3 - Data integration from existing repositories V1** (delivered)
- **D5.9 - Data integration from existing repositories V2** (upcoming)

2. Processing Service Integration & Deployment Guide

The integration and deployment of new processing services using the iAOS cloud platform facilities are demonstrated through a set of activities performed in Task 5.4 “Geostatistical methods” (ARMINES) and Task 5.5 “Integration of new processing services” (NERSC). These services were developed with support from Terradue for the use of the Ellip Solutions via Terradue Cloud Platform.

The Cloud Platform’s core services (e.g. access to virtual machines, software repositories) are hosted on Terradue’s private cloud infrastructure, where the maintenance activities are directly performed by Terradue’s DevOps team. This Private Cloud environment supports a Platform-as-a-Service (PaaS) solution for application integration, and provides the capability to deploy in production (on a selected IaaS provider) the integrated & validated services.

Integration environment (PaaS from Terradue)

Terradue Cloud Platform builds on three major outcomes of the recent developments in Computer Science and Web technology - Cloud Computing, Open Data repositories, and Web Services interoperability. They constitute the foundation of the PaaS capabilities delivered to developers and service integrators.

They provide the backbone and the core services on top of which Terradue Cloud Platform solutions are delivered to the PaaS users, and that we will introduce in this guide:

- **Ellip Notebooks**, to create interactive laboratory notebooks
- **Ellip Workflows**, to design scalable processing chains
- **Ellip Launchpads**, to select a data processor and deploy it as a Web Processing Service, on a given Processing cluster, hosted on a selected IaaS
- **Ellip Infohubs**, to connect a Web Processing Service to a custom Geobrowser client application, to be accessed as part of a user community.

The present platform and tools guide provides technical descriptions in support to the INTAROS Partners that are developing and delivering their Processing Service via the iAOS Cloud Platform.

The Platform equips Application developers with a virtualized computing environment, to develop, test and validate their EO data processing applications. Developers benefit from a cost effective PaaS environment, to develop and simulate their Application behaviour on computing clusters. Once deployed, the resulting Application is made available as a highly scalable data processing service.

For each Processing Service, a final deployment in production is supported by the Platform, to run the Processing Service back-end onto a selected IaaS (providing distributed computing resources), and optionally to also run the other Processing Service components (e.g. Front-end software) on selected ad-hoc ICT resources hosted on a IaaS provider.

Production environment (IaaS from Cloud providers)

Deployments in production are managed from Terradue Cloud Platform, towards selected IaaS providers, in the perspective of the operational exploitation of these Applications at scale (that will ingest large volumes of data to process), and for a competitive cost (typically on a commercial Cloud Computing cluster, or possibly on Cloud Computing resource available for research organizations).

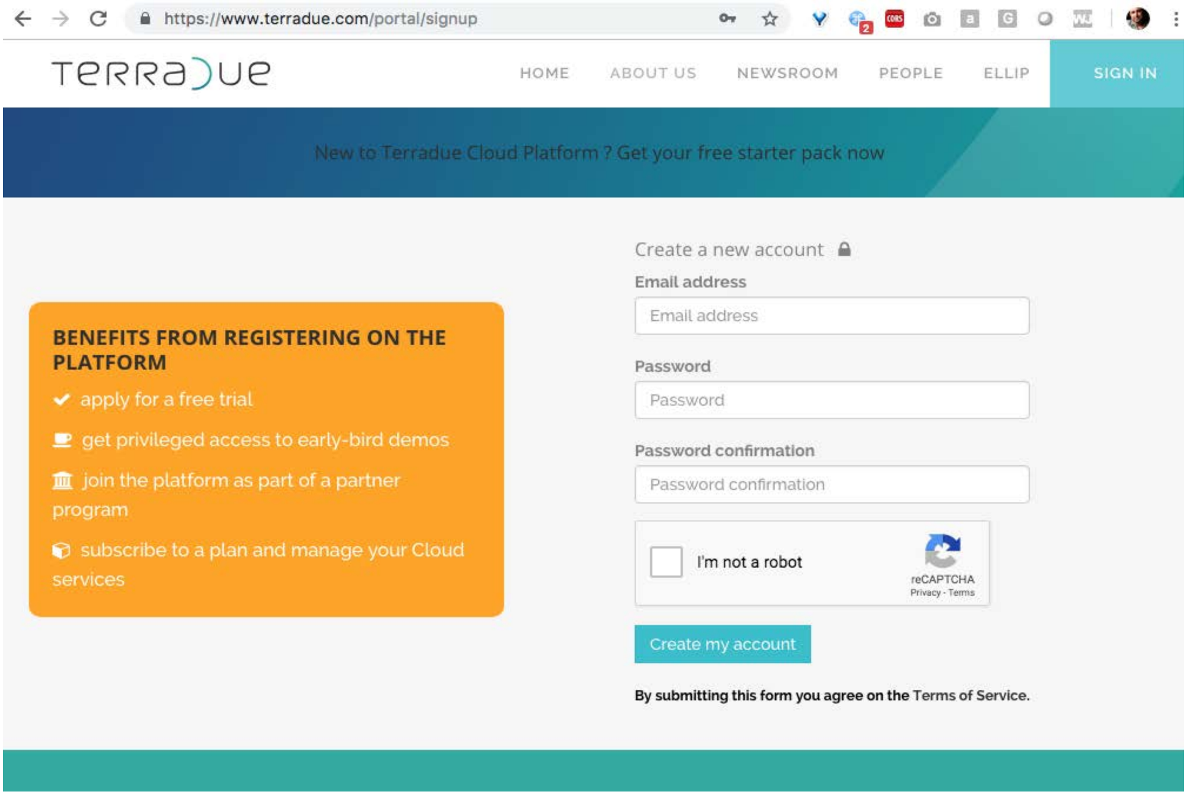
For example EGI.eu is a Terradue partner, in capacity to provide IaaS resources for the INTAROS Applications when they are validated for production. Other Cloud providers are accessible from Terradue Cloud Platform and work is on-going to connect the Copernicus DIAS providers, as additional IaaS providers capable to serve the INTAROS Partner needs for production environments on the Cloud (storage, data access, compute).

2.1. How to join the PaaS environment

Business Partners can join the PaaS environment as a service integrator, by creating a user account and getting an account upgrade.

For this, they can directly sign-up as a Terradue Cloud Platform user:

- https://www.terradue.com/portal/signup (cf. sign-up page registration form)



New to Terradue Cloud Platform ? Get your free starter pack now

BENEFITS FROM REGISTERING ON THE PLATFORM

- ✓ apply for a free trial
- get privileged access to early-bird demos
- join the platform as part of a partner program
- subscribe to a plan and manage your Cloud services

Create a new account

Email address: Email address

Password: Password

Password confirmation: Password confirmation

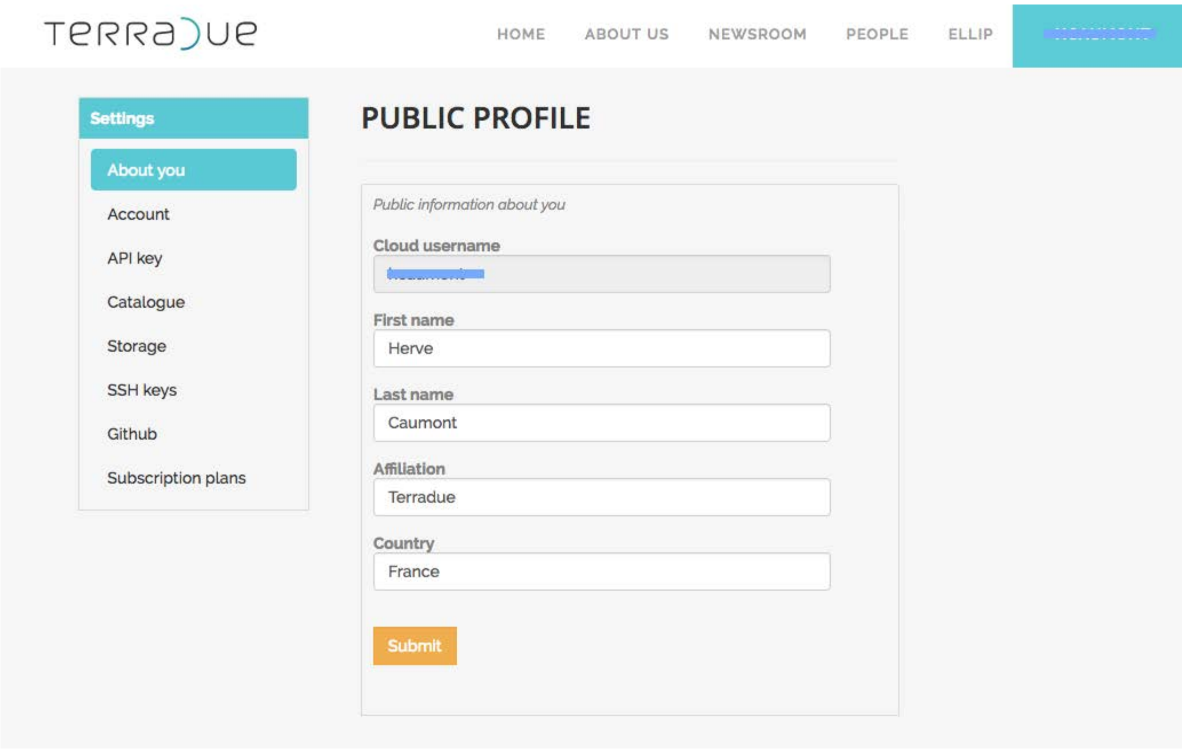
By submitting this form you agree on the Terms of Service.

Once done and signed-in, users can access their Welcome page:

- https://www.terradue.com/portal/welcome (when signed-in)

This pages provides the user with the links to the following services on the Terradue Cloud Platform:

- Access the Ellip Core Services and the Ellip Solutions that are part of your subscription level
- Request an account upgrade (see details further below)
- Access your user profile information (account settings):
    - **About you**: personal details (username, first name, last name, affiliation, country)
    - **Account**: user account details (username, email address, registered external authentication modes, password change
    - **API Key**: used to exploit the API functions on Terradue Cloud Platform (e.g. use of the catalogue API to manage a personal/company catalogue index)
    - **Catalog**: creation and management of your personal catalogue index on Terradue Cloud Platform
    - **Storage**: creation and management of your personal data storage on Terradue Cloud Platform
    - **SSH keys**: used to access via SSH protocol your Virtual Machines created on Terradue Cloud Platform
    - **Github**: to link your github account and use it as version control tool for your developments.
    - **Subscription plans**: your subscription status to Terradue Cloud Platform services, and the "Contact us" function to request an account upgrade.



PUBLIC PROFILE

Public information about you

Cloud username: [redacted]

First name: Herve

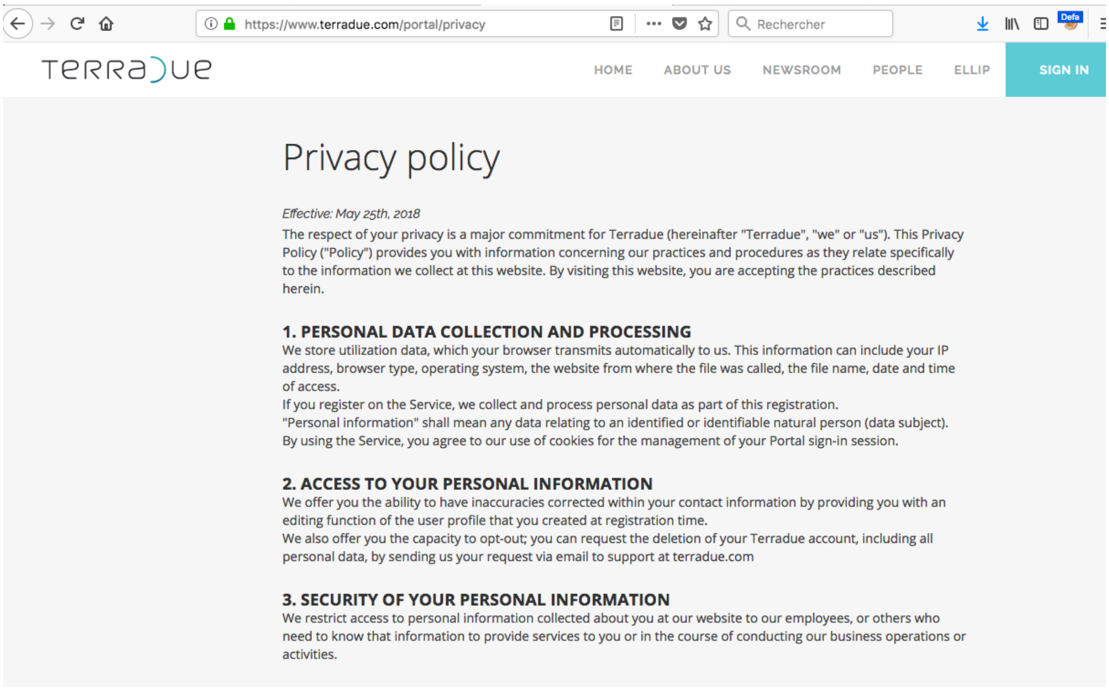
Last name: Caumont

Affiliation: Terradue

Country: France

The GDPR notice related to user control over his/her personal information managed on the Portal is accessible from the Terradue website footer:

- https://www.terradue.com/portal/privacy



Privacy policy

Effective: May 25th, 2018

The respect of your privacy is a major commitment for Terradue (hereinafter "Terradue", "we" or "us"). This Privacy Policy ("Policy") provides you with information concerning our practices and procedures as they relate specifically to the information we collect at this website. By visiting this website, you are accepting the practices described herein.

1. PERSONAL DATA COLLECTION AND PROCESSING

We store utilization data, which your browser transmits automatically to us. This information can include your IP address, browser type, operating system, the website from where the file was called, the file name, date and time of access.

If you register on the Service, we collect and process personal data as part of this registration.

"Personal information" shall mean any data relating to an identified or identifiable natural person (data subject).

By using the Service, you agree to our use of cookies for the management of your Portal sign-in session.

2. ACCESS TO YOUR PERSONAL INFORMATION

We offer you the ability to have inaccuracies corrected within your contact information by providing you with an editing function of the user profile that you created at registration time.

We also offer you the capacity to opt-out; you can request the deletion of your Terradue account, including all personal data, by sending us your request via email to support at terradue.com

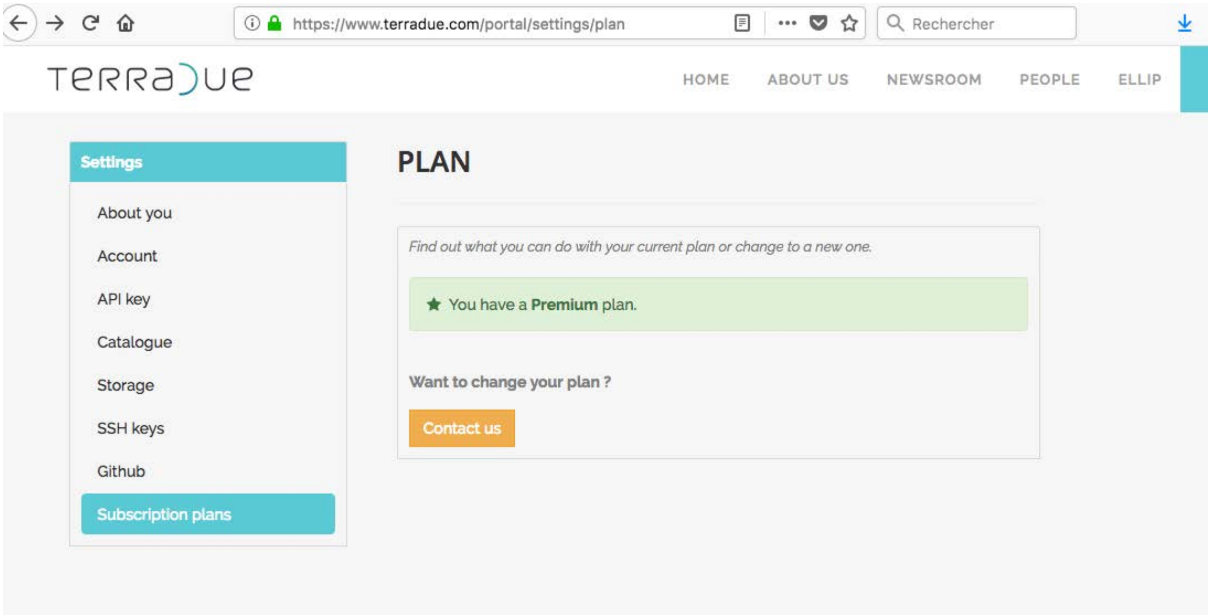
3. SECURITY OF YOUR PERSONAL INFORMATION

We restrict access to personal information collected about you at our website to our employees, or others who need to know that information to provide services to you or in the course of conducting our business operations or activities.

How to request a Terradue account upgrade ?

After the account creation, one needs to get an account upgrade, with the proper access rights granted. This can be requested via the Terradue Portal account settings page:

- https://www.terradue.com/portal/settings/plan (when signed-in, "Subscription plans" menu entry)



PLAN

Find out what you can do with your current plan or change to a new one.

★ You have a **Premium** plan.

Want to change your plan ?

2.1.1. My account (as a PaaS user)

Once signed-in (see sections above), users of the Integration environment (PaaS from Terradue) can access their user account from here:

- https://www.terradue.com/portal/settings/profile (when signed-in)

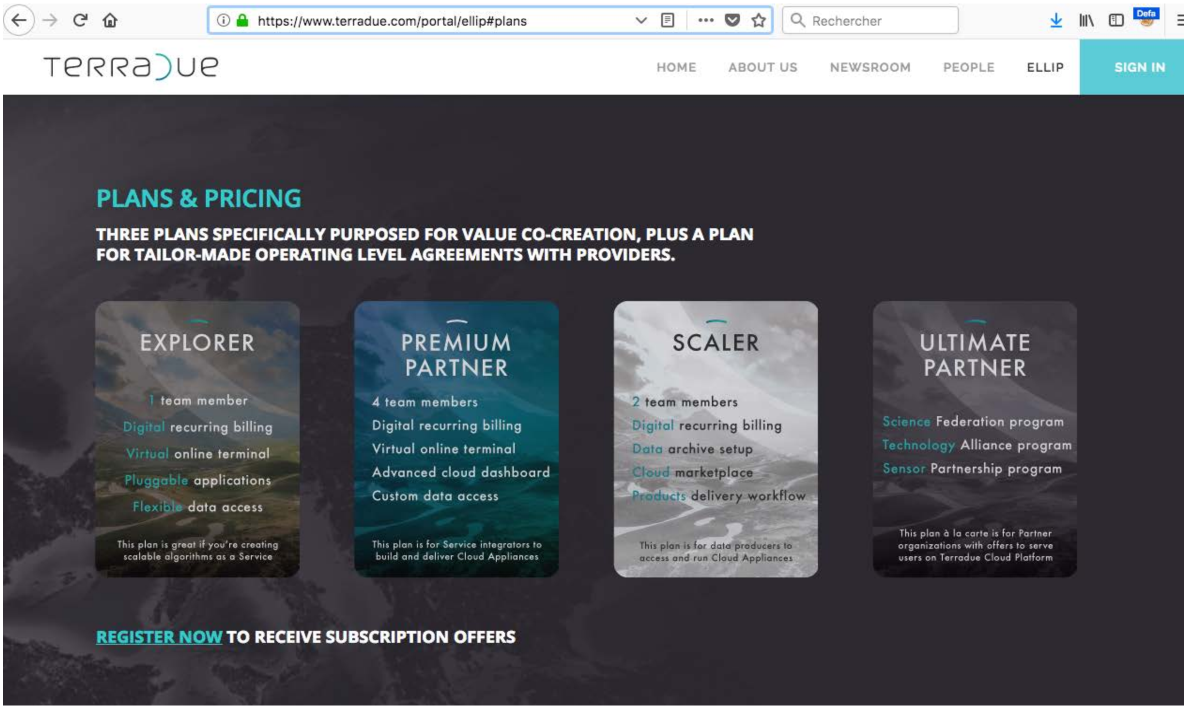
Depending on their account upgrade status, user accounts are assigned a set of credentials which are defined as follows:

- **No plan**: after initial registration, accounts do no have a subscription plan associated, they imply by default 'Visitor' accounts
- **Membership**: this is a user account upgrade meant for the users of external, Ellip-powered systems. A membership account means the possibility to create on Terradue Cloud Platform a personal Cloud storage space, a personal catalog index, as well as an API key, that are linked from and used by the Portal functions of a given external, Ellip-powered system.

- **Explorer**: this subscription plan is meant for creating processing services as Cloud Appliances, either based on Jupyter Notebook executables or on scalable algorithms packaged for a data processing framework (e.g. Hadoop)
- **Scaler**: this subscription plan is meant for data producers to access and run Cloud Appliances previously prepared on Terradue Cloud Platform
- **Premium Partner**: this subscription plan is meant for Service integrators to build and deliver client side user services (e.g. Web Portal) and server side Cloud Appliances on Terradue Cloud Platform, as well as deploy and operate these on third party IaaS providers (production environments).
- **Ultimate Partner**: this subscription plan 'à la carte' is meant for organizations with offers to serve users on Terradue Cloud Platform (data providers, technology providers, data processing module providers).

The subscription plans to access the tools and services of the Integration environment (PaaS) are presented on the Terradue website:

- https://www.terradue.com/portal/ellip#plans



PLANS & PRICING

THREE PLANS SPECIFICALLY PURPOSED FOR VALUE CO-CREATION, PLUS A PLAN FOR TAILOR-MADE OPERATING LEVEL AGREEMENTS WITH PROVIDERS.

EXPLORER

1 team member
Digital recurring billing
Virtual online terminal
Pluggable applications
Flexible data access

This plan is great if you're creating scalable algorithms as a Service

PREMIUM PARTNER

4 team members
Digital recurring billing
Virtual online terminal
Advanced cloud dashboard
Custom data access

This plan is for Service integrators to build and deliver Cloud Appliances

SCALER

2 team members
Digital recurring billing
Data archive setup
Cloud marketplace
Products delivery workflow

This plan is for data producers to access and run Cloud Appliances

ULTIMATE PARTNER

Science Federation program
Technology Alliance program
Sensor Partnership program

This plan à la carte is for Partner organizations with offers to serve users on Terradue Cloud Platform

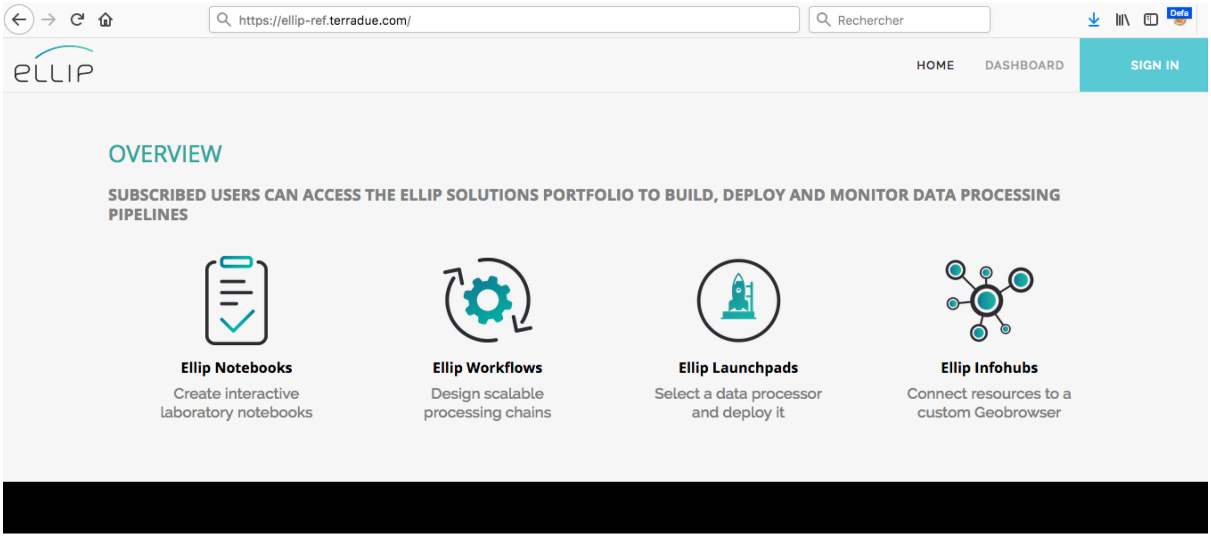
REGISTER NOW TO RECEIVE SUBSCRIPTION OFFERS

Once subscribed to a plan, you can access the Ellip Dashboard from here:

- https://ellip.terradue.com/ (Ellip Dashboard interface to access the subscribed Ellip Solutions)

As of today (this is subject to change in the future), the access control based on the user subscription plan is provided as follows:

- Solutions subscribed to by **Explorer** users:
    - Ellip Notebooks
    - Ellip Workflows
- Solutions subscribed to by **Scaler** users:
    - Ellip Launchpads
    - Ellip Infohubs
- Solutions subscribed to by **Premium** Partners:
    - Ellip Notebooks
    - Ellip Workflows
    - Ellip Launchpads
    - Ellip Infohubs



OVERVIEW

SUBSCRIBED USERS CAN ACCESS THE ELLIP SOLUTIONS PORTFOLIO TO BUILD, DEPLOY AND MONITOR DATA PROCESSING PIPELINES

Ellip Notebooks
Create interactive laboratory notebooks

Ellip Workflows
Design scalable processing chains

Ellip Launchpads
Select a data processor and deploy it

Ellip Infohubs
Connect resources to a custom Geobrowser

The online documentation for the Ellip solutions will be made available from here:

- http://docs.terradue.com (entry point for the documentation about Terradue Cloud Platform and the Ellip solutions).

2.1.2. Partner Support

Partner users with “Premium Partner” or “Ultimate Partner” subscription plans can access a dedicated support space in order to interact with Terradue’s DevOps team:

- http://support.terradue.com (entry point for the service integrators, data providers, technology providers, data processing module providers that are integrating value-added services on Terradue Cloud Platform via the provided APIs).

2.1.3. User Forum

All Terradue users can access the public user Forum, and also start contributing to it (after login). Several sections on the User Forum are already contributed by user communities. All the sections are under the editorial control of Terradue (as a moderator), and the Forum is a source of shared knowledge about using the Platform tools and services, either as part of the application integration activities, or as part of the exploitation of Ellip-powered services.

- http://discuss.terradue.com (entry point for users having a Terradue account, to sign-in and comment or ask questions on existing posts, or provide their feedback, hints & tips and advices to others about using the Platform tools and services).

2.1.4. User Helpdesk

Terradue users having a Membership status can access the User Helpdesk service as a 'customer' requesting support, for the projects they have been granted an access to an Ellip-powered application. For these projects, they can receive professional support services in relation with an Ellip-powered system they are using.
Terradue users having a Subscription plan can access the User Helpdesk service as an 'agent' delivering support, for the projects they are managing. For these projects, they can deliver professional support services in relation with an Ellip-powered system they are operating.

- http://helpdesk.terradue.com (entry point for users having a Membership or a Subscription plan to benefit from a professional service desk environment supporting their exploitation goals on Ellip-powered systems).

2.2. Using the Cloud Platform's core services

We present hereafter the list of the Ellip core services, with their access points, and their access conditions for users in order to build, test and deploy data processing services.

2.2.1. Data Agency

The Data Agency is made of two core services for the management of federated data repositories and scalable data staging operations:

- https://store.terradue.com
  Access to your personal storage on the store service follows this template:
  - https://store.terradue.com/<username>
  - <username> is provided via your Terradue user account

  The protocol to access the store functions is based on HTTP:
  - http://docs.terradue.com/t2-api/data/storage.html

- https://catalog.terradue.com
  Access to your personal index on the catalog service follows this template:
  - https://catalog.terradue.com/<username>/search?apikey=<YOUR-API-KEY>
  - <username> is provided via your Terradue user account
  - <YOUR-API-KEY> is provided via your Terradue user account

  The protocol to access the catalog functions is based on OpenSearch:
  - http://docs.terradue.com/t2-api/data/publication.html

2.2.2. Cloud Dashboard

The Cloud Dashboard service is used for the creation and management of Containers and Virtual Machines supporting the application integration work done within the PaaS environment, as well as for the deployment of Cloud Appliances in production (cloud bursting):

- https://cloud.terradue.com (for developer users to create and manage their VMs)
- https://[sandbox-ip]/dashboard (for developer users to access their VM dashboard and work on it, with [sandbox-ip] delivered to them from the Cloud dashboard, after their creation of a new VM)

2.2.3. Virtual Private Network

The Virtual Private Network service is used for establishing a secured communication from a client (the user desktop/laptop machine) to the Virtual Machines that the user has created on Terradue Cloud Platform:

- https://access.terradue.com (for developer users to activate their VPN connexion, for their access to VMs managed on Terradue's private Cloud)

2.2.4. Other Cloud Platform core services

Additional services are being configured on Terradue Cloud Platform in order to serve the PaaS users needs. They will be documented in a later stage of the Terradue Cloud Platform developments.

- geo.terradue.com (Geoserver instance)
- recast.terradue.com (Data Casting service)
- repository.terradue.com (Software repository)
- anaconda.org/Terradue (Software repository)

2.3. Cloud Platform SDK Handbook

The Ellip Software Development Kit (SDK) is providing functions to the users of the Ellip Solutions, for using in particular the iAOS and Terradue Cloud Platform APIs, in order to perform Catalogue resources discovery and data access operations, as well as workflow integration, simulation and invocation, workflow parameters retrieval and error logs management within a data processing workflow.

The new user documentation for the Cloud Platform SDK, contributed as part of the INTAROS WP5 effort, will be made available online:

- https://docs.terradue.com/ellip/developer/sdk

Definitions

SDK: Software Development Kit. A set of tools installable on the developer IDE and usable as software dependencies in the developer's application code.

SDK Code Templates: templates for quickly creating a specific type of application, from an entire application down to specific application components (cf. an illustration[2] from the well-established Android community, on using SDK tools for creating application projects).

API: an application programming interface (API) is a computing interface to a software component host or a system host, that defines how other software applications can use the hosted resources (calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow). An API can be custom, specific to a component, or it can be designed based on a standardized interface specification in order to ensure interoperability.

Rationales

The new SDK will repackage several software tools in order to present a clear offer to the Ellip subscribers (developer users) in terms of APIs made available to them on Terradue Cloud Platform and in terms of SDK tools provided to them in order to interact with these APIs from their application code.

In particular, when several SDK functions from different technical scopes are involved during an application integration work, making use of all of them shall not imply they are grouped into a same software package (installer) but rather they become part of an application template (e.g. leveraging further the cookiecutter[3] technology) that declares all the needed software import operations.

[2] https://stuff.mit.edu/afs/sipb/project/android/docs/tools/projects/templates.html

[3] https://cookiecutter.readthedocs.io

2.3.1. Data discovery - Catalog client library

These libraries are used to perform the dataset search operations, which are processed at the level of the Data Agency API on the Platform.

Classes & Methods (Python):

collection.py

- Collection
    - Register_operation
    - search
    - _intersection
    - feed
    - filter
    - correlate
    - _params

operations.py

- OperationFactory
- get_intersection
- AoiAnalysisOperation
- ReferenceOperation

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.3.2. Data access - Stage-in client library

These libraries are used to perform dataset copy operations, which are processed at the level of the Data Gateway API on the Platform.

Classes & Methods (Python):

Dataset file copy operations from distributed source repositories:

- stagein.copy

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.3.3. Data processing - Web Processing Service client library

These libraries are used to invoke data processing jobs and retrieve data processing results.

Classes & Methods (Python):

wps.get_wps_process_list
wps.get_parameters_as_dict
wps.get_parameter_string
wps.get_wps_parameters
wps.validate_parameters
wps.get_last_process_version

wps.create_job_request
wps.execute_job_request
wps.get_job_request_status
wps.check_job_request_status
wps.get_status_location
wps.get_status
wps.get_processOutputs

owslib.wps class.method list:

- WebProcessingService
- WebProcessingService.getcapabilities
- WebProcessingService.describeprocess
- WebProcessingService.execute
- monitorExecution
- WPSExecution.buildRequest
- WPSExecution.submitRequest
- WPSExecution.statusLocation
- WPSExecution.checkStatus
- WPSExecution.getStatus
- WPSExecution.parseResponse
- WPSExecution.getOutput

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.3.4. Data processing - Managed processes client library

These libraries are used to manage processes: add a processor, remove a processor from the registry of Cloud appliances on the Platform's production center.

Classes & Methods (Python):

wps_mp.deploy
wps_mp.undeploy
wps_mp.is_process_deployed

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.3.5. Workflow application - Streaming executable client library

These libraries are used to integrate a MapReduce streaming executable as part of an Ellip Workflows "standalone" application (purpose: perform data transformation operations), based on the Hadoop MapReduce framework offered by the Platform to deliver scalable data processing services.

Classes & Methods (Python):

mrse_getparam
mrse_log
mrse_publish

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.3.6. Trigger application - Production scheduling client library

These libraries are used to integrate a MapReduce streaming executable as part of an Ellip Workflows "Trigger" application (purpose: perform input data lists management), based on the Hadoop MapReduce framework offered by the Platform to deliver scalable data processing services.

Trigger:

- tg_queue
- tg_pipe

Recast:

- recast
- recast_connect
- recast_execute
- recast_monitor

DataItems:

- dataItem_create_from_single_ref
- dataItem_create_from_multiple_ref
- dataItem_create_empty

Installers will be made available on the Platform software repository for direct import from within application code integrated on Ellip Notebooks and/or Ellip Workflows environments.

2.4. Using the Cloud Platform's Ellip Solutions portfolio

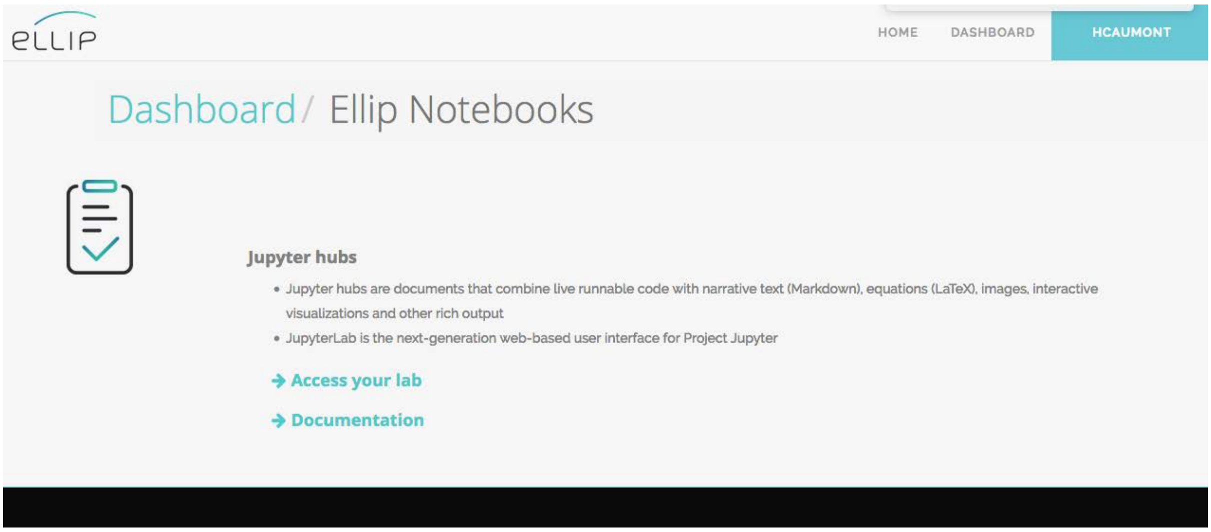
The **Ellip Solutions** are empowering the PaaS users to create value as part of the build, test and deploy phases of a data processing pipeline lifecycle. They can select the way (open / paid) to share their assets on Terradue Cloud Platform to other users, in a self-service mode.

All the value creation (assets) done by using an "Ellip Solution" will become referenced on the **Ellip Exchange**, a new cloud platform core service that will be developed during the coming year, and will enforce Ellip as "a collaborative workplace" where partners can co-design their Applications.

2.4.1. Ellip Notebooks

The Ellip Notebooks solution is based on the JupyterLab environment. It is hosted on a scalable JupyterHub server, which provides individual users with a Jupyter Notebook server and an associated permanent workspace. The solution allows for the test & validation of EO data processing modules developed in popular Data Science programming languages such as R and Python. Users typically explore new opportunities to work with data, that they can share as reproducible experiments on the Platform's marketplace (Ellip Exchange).

The Ellip Notebooks solution is made available to Explorer subscribers & Premium Partners.

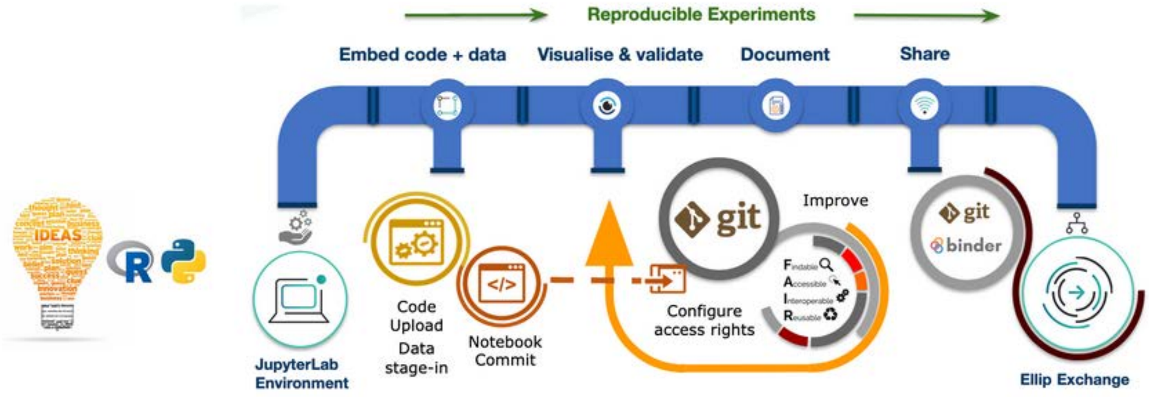


Dashboard / Ellip Notebooks

Jupyter hubs

- Jupyter hubs are documents that combine live runnable code with narrative text (Markdown), equations (LaTeX), images, interactive visualizations and other rich output
- JupyterLab is the next-generation web-based user interface for Project Jupyter

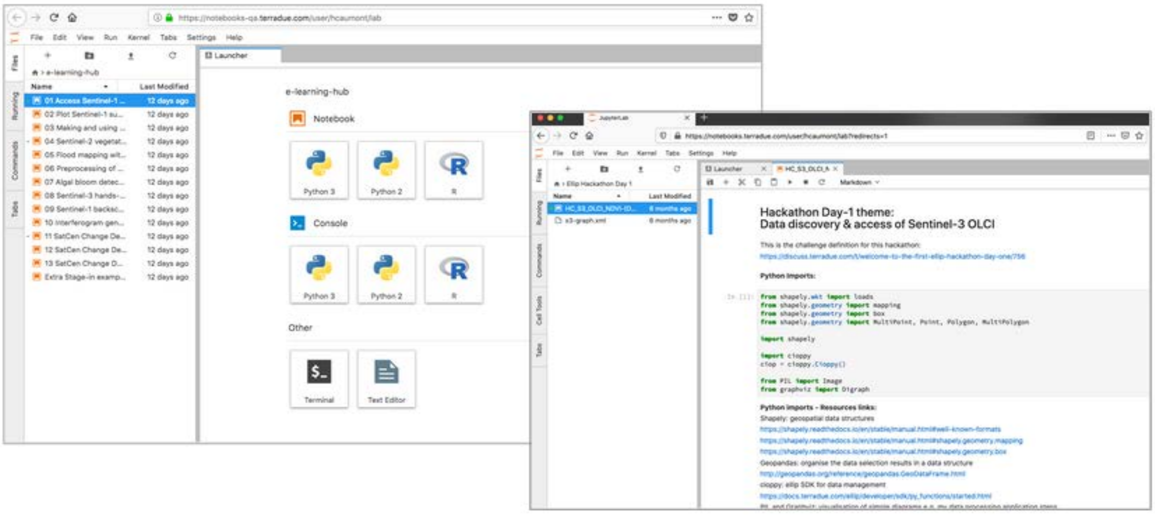
The Ellip Notebooks solution allows users to integrate, test and share simple data processing functions in a very interactive way. These functions are implemented as notebook files, which can further on become unitary nodes of a data processing pipeline.
It is also suitable for users to easily integrate client software contacting data access or data processing Web Services.



The Ellip Notebooks Solution provides a user with a dedicated access to a Jupyter Notebook server instance, managed on the Platform's JupyterHub cluster, and with the connectivity to Git repositories where notebook files can be managed and versioned.

The outcomes for an Explorer subscriber work on the Ellip Notebooks solution are Jupyter Notebook files, with application code implementing data processing and visualization functions, and accessing EO data products from well-known remote repositories referenced by the Platform's Data Agency. These notebook files are executable documents that contain live code, equations, visualizations and explanatory text. They can be designed to comply with FAIR principles (Findable, Accessible, Interoperable, Reusable) and shared with other users via the Ellip Exchange, leveraged as the core enabler of Ellip as a 'collaborative workplace'. Shared notebook file resources will typically consist here of:

- Reproducible experiments as code
- Interactive data visualizations



Name	Last Modified
01 Access Sentinel-1 ...	12 days ago
02 Plot Sentinel-1 su...	12 days ago
03 Making and using ...	12 days ago
04 Sentinel-2 vegetat...	12 days ago
05 Flood mapping wit...	12 days ago
06 Preprocessing of ...	12 days ago
07 Algal bloom detec...	12 days ago
08 Sentinel-3 hands-...	12 days ago
09 Sentinel-1 backsc...	12 days ago
10 Interferogram gen...	12 days ago
11 SatCen Change De...	12 days ago
12 SatCen Change De...	12 days ago
13 SatCen Change D...	12 days ago
Extra Stage-in examp...	12 days ago

Name	Last Modified
HC_S3_OLCI_NDVI-D...	6 months ago
s3-graph.xml	6 months ago

Hackathon Day-1 theme:
Data discovery & access of Sentinel-3 OLCI

This is the challenge definition for this hackathon:
https://discuss.terradue.com/t/welcome-to-the-first-ellip-hackathon-day-one/756

Python imports:

```
In [1]: from shapely.wkt import loads
        from shapely.geometry import mapping
        from shapely.geometry import box
        from shapely.geometry import MultiPoint, Point, Polygon, MultiPolygon

        import shapely

        import cioppy
        ciop = cioppy.Cioppy()

        from PIL import Image
        from graphviz import Digraph
```

Python imports - Resources links:
Shapely: geospatial data structures
https://shapely.readthedocs.io/en/stable/manual.html#well-known-formats
https://shapely.readthedocs.io/en/stable/manual.html#shapely.geometry.mapping
https://shapely.readthedocs.io/en/stable/manual.html#shapely.geometry.box
Geopandas: organise the data selection results in a data structure
http://geopandas.org/reference/geopandas.GeoDataFrame.html
cioppy: ellip SDK for data management
https://docs.terradue.com/ellip/developer/sdk/py_functions/started.html

In a code cell of a Jupyter Notebook file, it is possible to edit and write new code, with productivity helpers such as full syntax highlighting and tab completion. By default, the language associated to a code cell is Python, but depending on the kernel, other languages, such as R, can be handled interactively. When a code cell is executed, its code is sent to the kernel associated with the notebook and the results that are returned from this computation are displayed in the notebook as the cell's output. The output is not limited to text, with many other possible forms of output are also possible.
Users of the Ellip Notebooks solution can also design their notebook files with particular reusability goals in mind, in particular for their exploitation as streaming executables within an Ellip Workflows application (see next section about the Ellip Workflows solution).

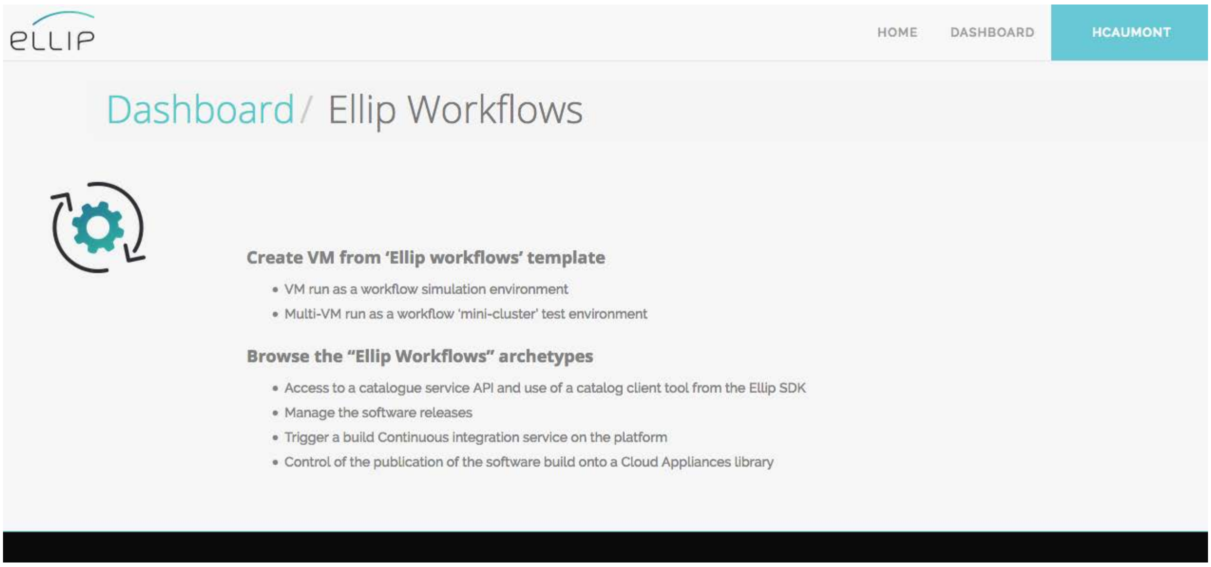
The user documentation for the Ellip Workflows solution is available online:

- https://docs.terradue.com/ellip/solutions/notebooks

2.4.2. Ellip Workflows

The Ellip Workflows solution is based on a framework including the Hadoop MapReduce streaming API as well as ad-hoc data staging tools and services ('ciop' tools) and Web Processing Service interface (WPS).

The Ellip Workflows solution is made available to Explorer subscribers & Premium Partners.



Dashboard / Ellip Workflows

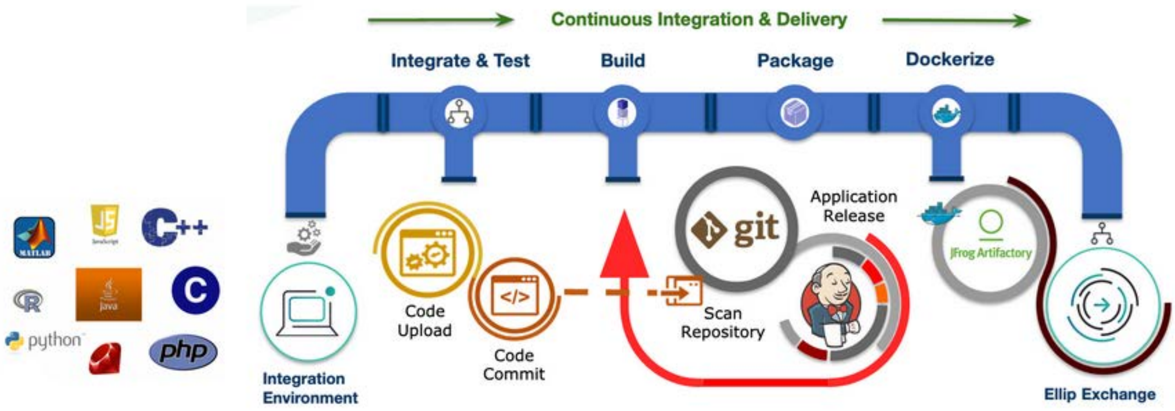
Create VM from 'Ellip workflows' template

- VM run as a workflow simulation environment
- Multi-VM run as a workflow 'mini-cluster' test environment

Browse the "Ellip Workflows" archetypes

- Access to a catalogue service API and use of a catalog client tool from the Ellip SDK
- Manage the software releases
- Trigger a build Continuous integration service on the platform
- Control of the publication of the software build onto a Cloud Appliances library

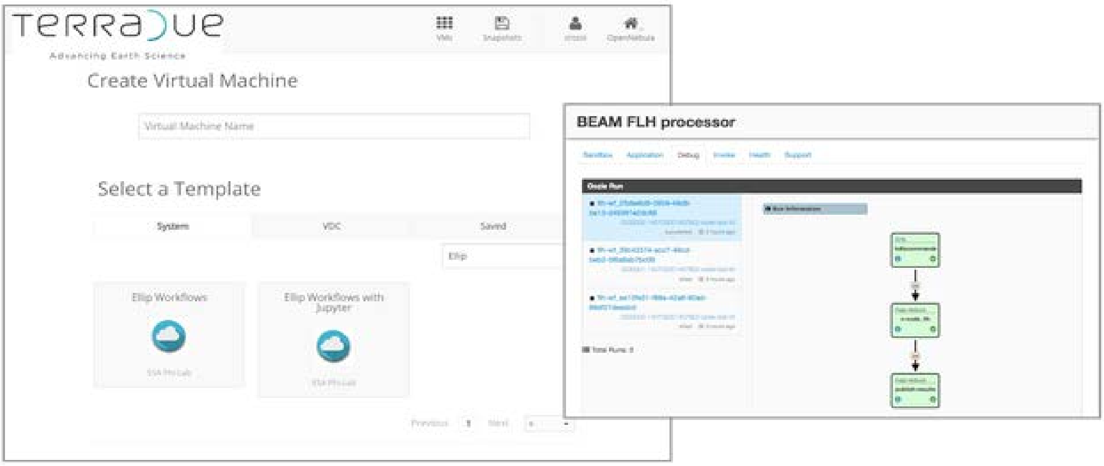
The Ellip Workflows solution allows users to integrate, test and validate/package scalable data processing workflows that can be deployed as highly interoperable data processing services. The application Integration Environment (Cloud Sandbox) is accessed by developer users within a Virtual Machine (VM). It consists in a software framework and tools for processor integration and testing, that allows to run initially the applications in a simulation environment (sandbox mode) that can readily scale when deployed in production (cluster mode) without any further modification. The auto-scalability feature enabled by the provided framework and tools is a major asset of the Terradue PaaS.



The outcomes for an Explorer subscriber work on the Ellip Workflows solution are software application packages, with application code implementing a highly scalable data processing service, and accessing EO data products from well-known remote repositories referenced by the Platform's Data Agency. These data processing applications can be designed to comply with software modularity and reusability principles, and shared with other users via the Ellip Exchange, leveraged as the core enabler of Ellip as a 'collaborative workplace'.
Shared data processing resources will typically consist here of:

- Application code
- Packaged applications
- Application Tutorials

The Cloud Sandbox environment allows users to plug scientific applications written in a variety of languages (e.g. Java, C++, IDL, Python, R), then test, validate and package these applications. The Application Packaging output is a file in RPM format that, once validated, can be easily installed on a production host or cluster, including via the management of Docker images. The package provides both the Application and the software dependencies required for executing the Application itself on the Production environment. A new interface is furthermore made available within the Ellip Workflows solution, also based on the Jupyter Notebook technology, which allows to integrate data processing services within the provided framework in a very interactive way.



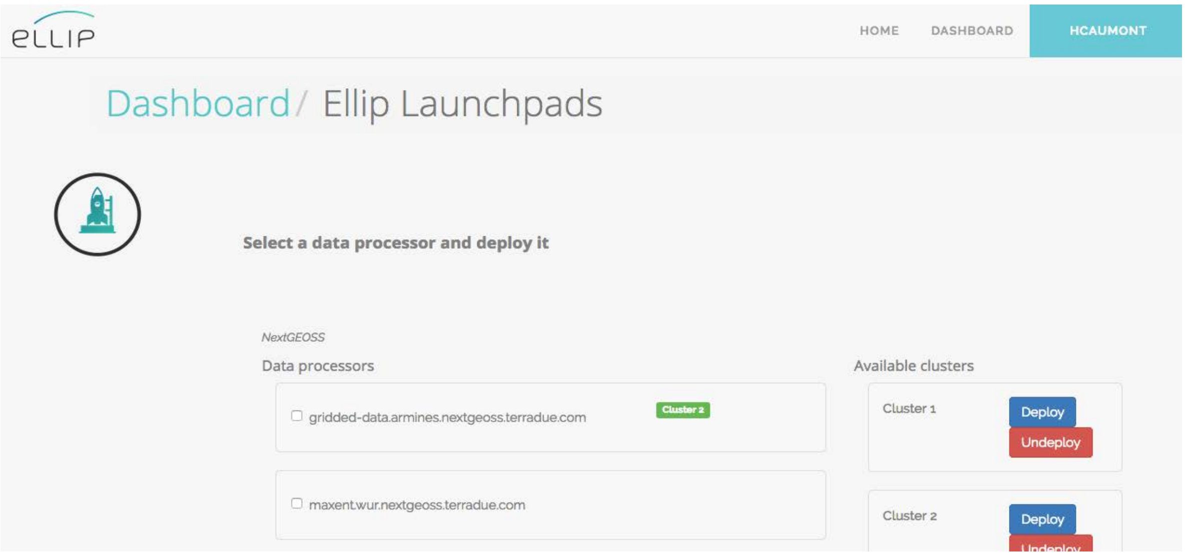
Overall, the Continuous Integration capacity on Terradue Cloud Platform provides the automation layers to manage versioned application packages, in order to feed Cloud deployments (cloud bursting), from Terradue Cloud Platform towards a selected Production environment, where such applications can be accessed as independent, scalable data processing services, in a Software-as-a-Service mode operated by the service owner.
The user documentation for the Ellip Workflows solution is available online:

- https://docs.terradue.com/ellip/solutions/workflows

2.4.3. Ellip Launchpads

The Ellip Launchpads solution is based on a deployer engine, performing Cloud bursting operations to selected Production environments (Cloud providers) in order to deliver scalable data processing capabilities. The Production environments are accessed by EO data product specialists, in order to configure the scalable data processing services that will deliver value-added products to their user communities. This way, they scale up production environments to deploy and run pre-packaged processing services on a selected Cloud Provider.

The Ellip Launchpads solution is made available to Scaler subscribers & Premium Partners.



Dashboard / Ellip Launchpads

Select a data processor and deploy it

NextGEOSS

Data processors

☐ gridded-data.armines.nextgeoss.terradue.com Cluster 2

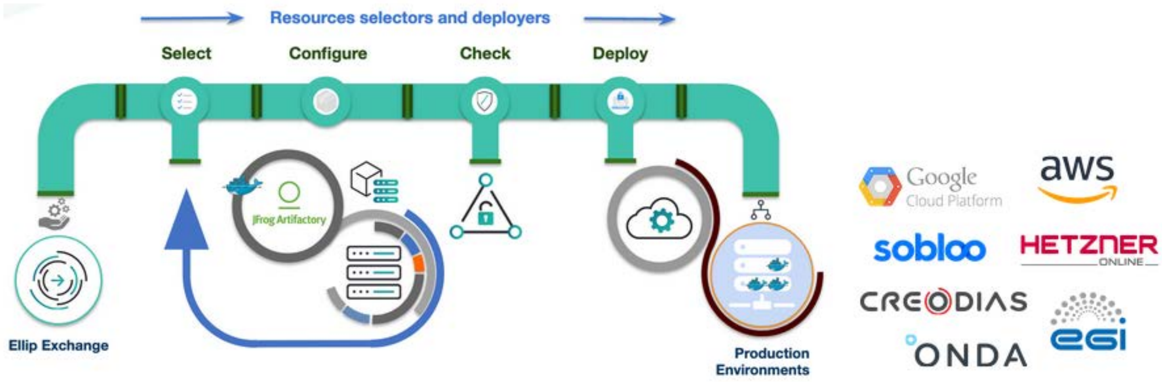
☐ maxent.wur.nextgeoss.terradue.com

Available clusters

Cluster 1

Cluster 2

The Ellip Launchpads solution will be made available to users in a later stage of the Terradue Cloud Platform developments.



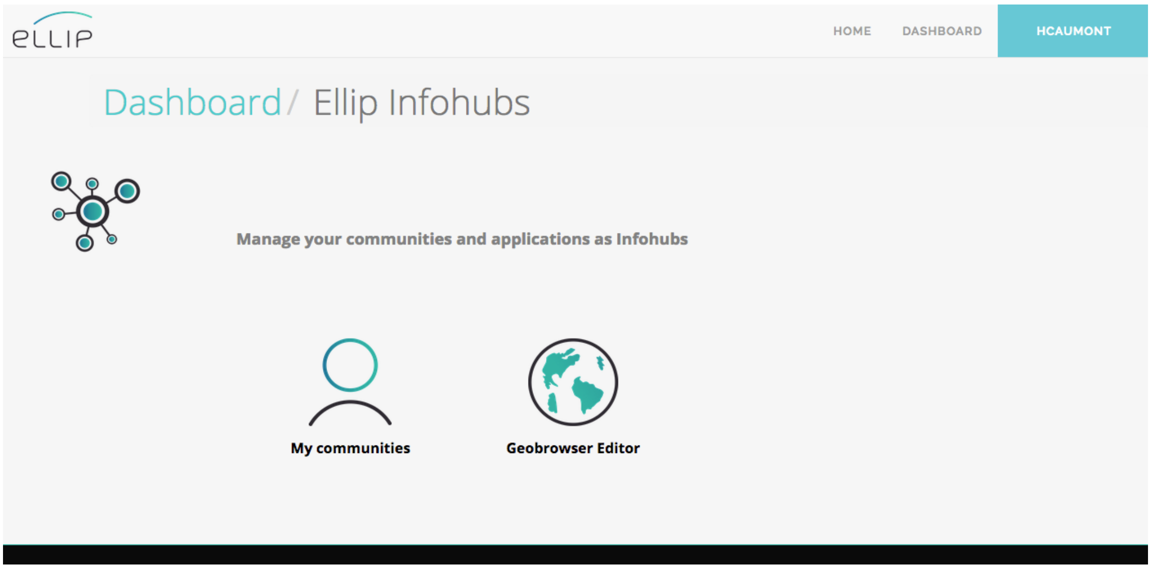
The user documentation for the Ellip Launchpads solution is not yet available online.

- Online user documentation: upcoming

2.4.4. Ellip Infohubs

The Ellip Infohubs Solution is dedicated to the creation of work environments for the engagement of stakeholders. These are defined for target user communities as "Information hubs" (Infohubs), where data analysis systems (e.g. a Geobrowser App) and data exploitation tools (incl. for the post-processing of newly generated information layers) are delivered.

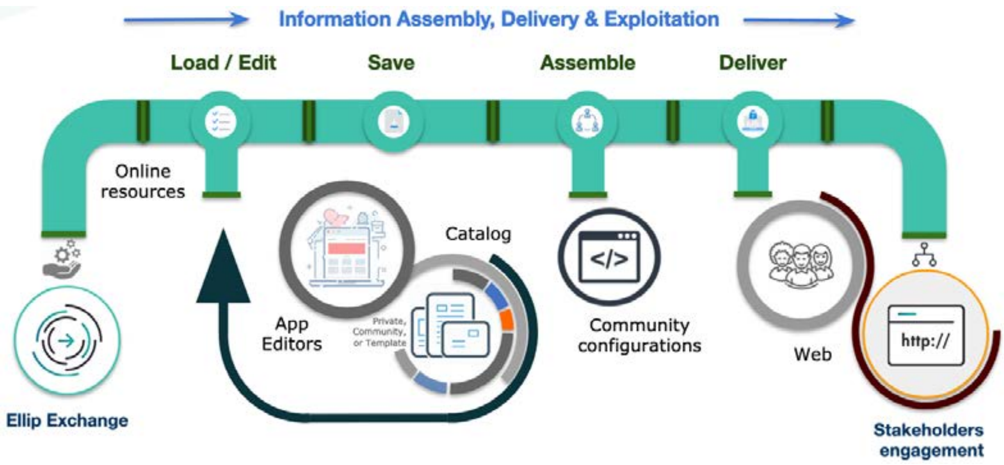
The Ellip Infohubs Solution is made available to Scaler subscribers & Premium Partners.



Dashboard / Ellip Infohubs

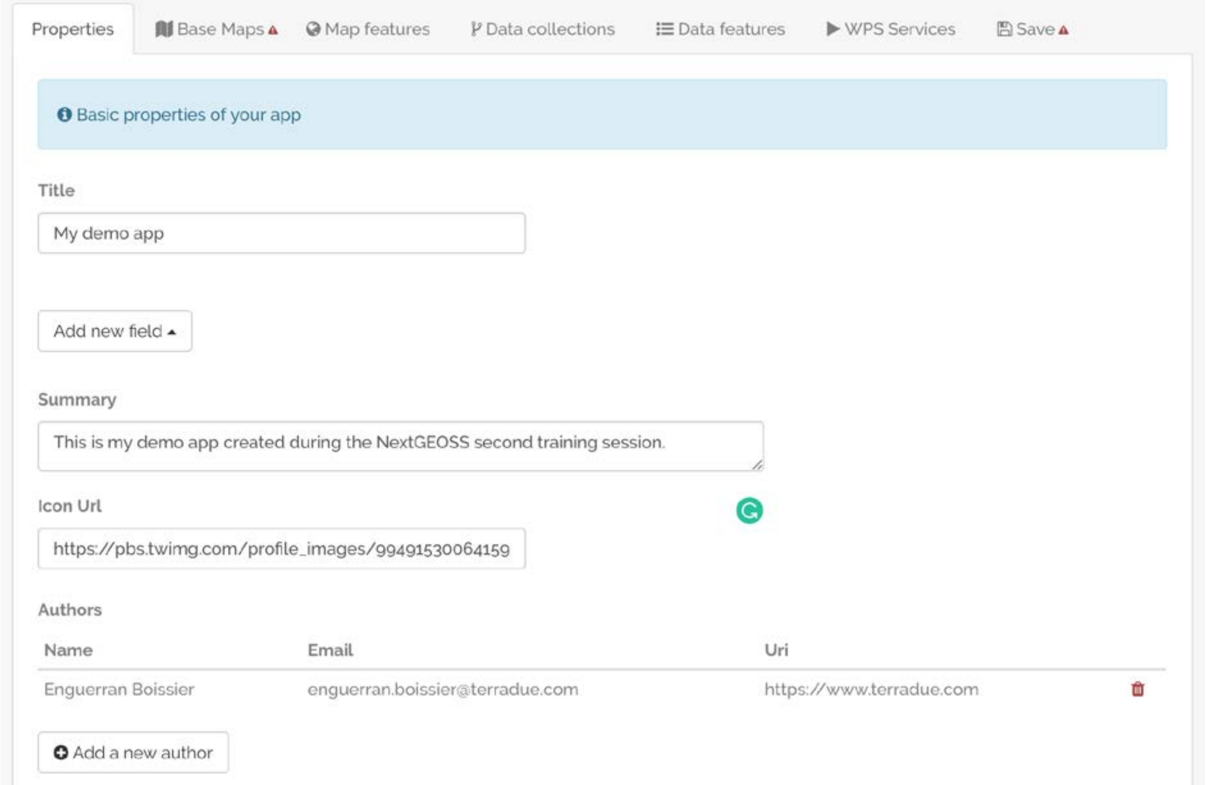
Manage your communities and applications as Infohubs

Users of the ellip Infohubs solution have the ability to create ready-to-use geobrowser apps, and save their definition as standardized OWS Context files in the Platform's catalog, allowing collaborative work to take place by customizing existing templates, and sharing the resulting Apps with selected group of users.



The user documentation for the Ellip Launchpads solution is not yet available online. We provide it hereafter as a preview.

Step 1: define properties



Basic properties of your app

Title

My demo app

Summary

This is my demo app created during the NextGEOSS second training session.

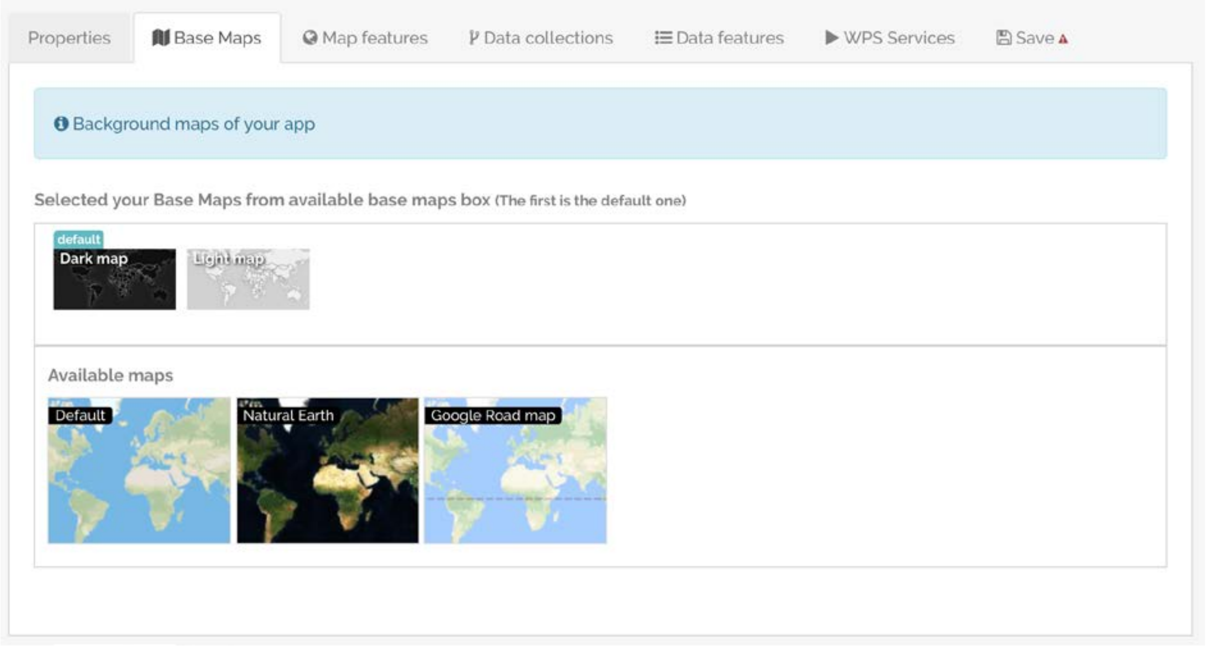
Icon Url

https://pbs.twimg.com/profile_images/99491530064159

Authors

Name	Email	Uri
Enguerran Boissier	enguerran.boissier@terradue.com	https://www.terradue.com

Step 2 & 3: select a base map and map features (like existing WMS layers)

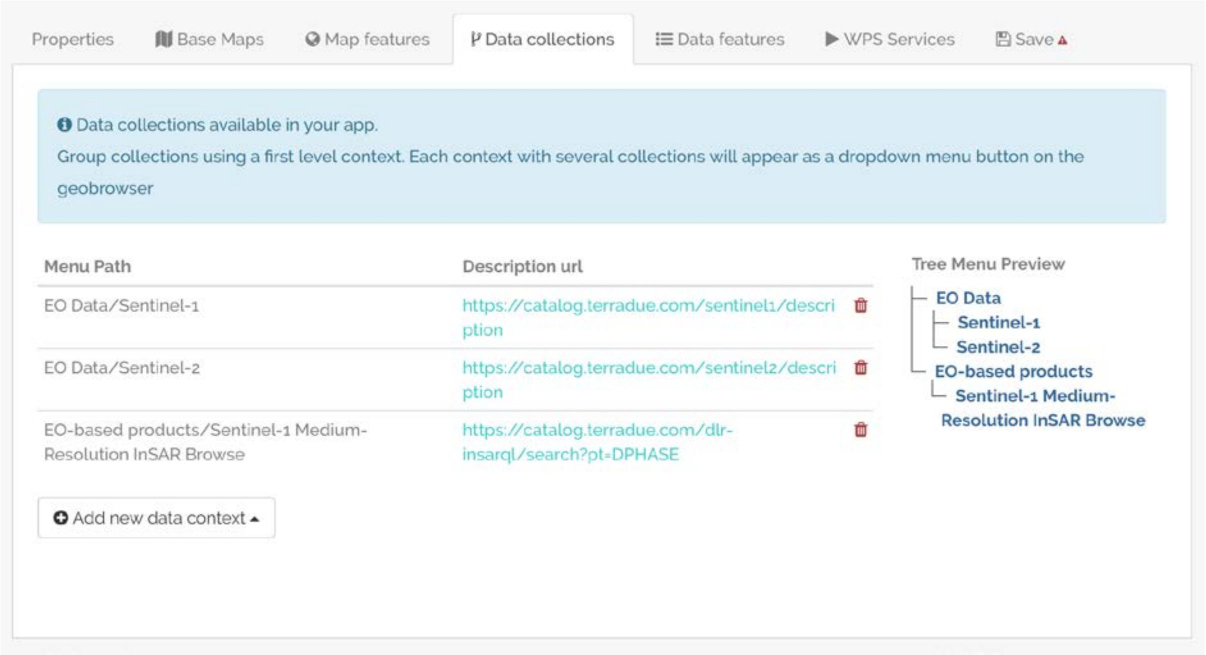


Background maps of your app

Selected your Base Maps from available base maps box (The first is the default one)

Available maps

Step 4: configure data collections



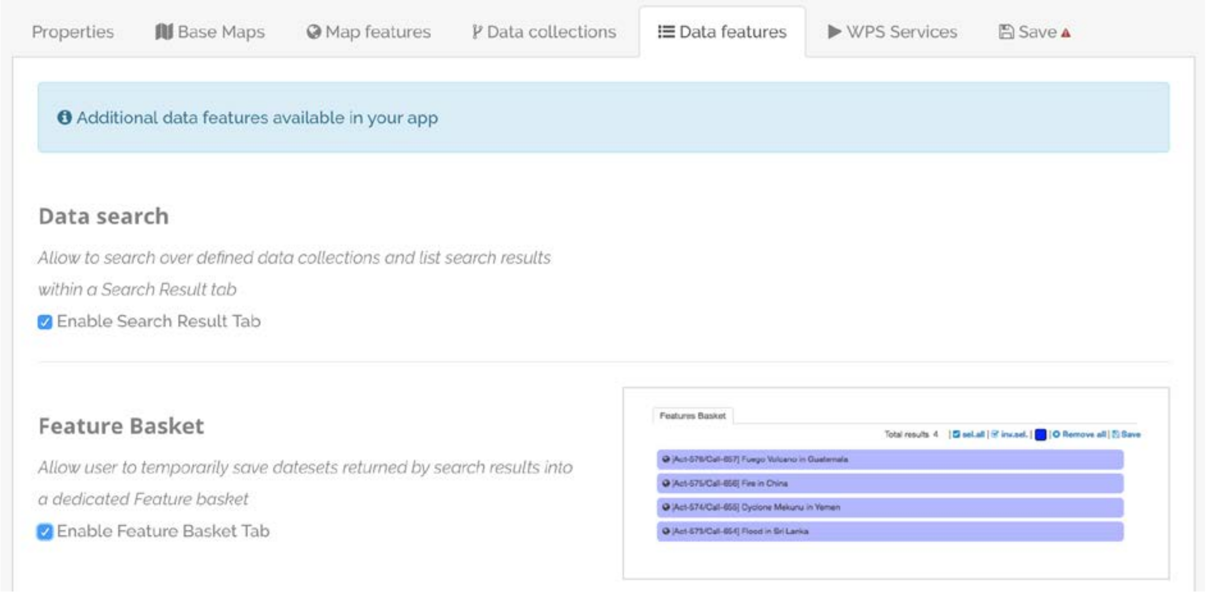
Data collections available in your app.
Group collections using a first level context. Each context with several collections will appear as a dropdown menu button on the geobrowser

Menu Path	Description url
EO Data/Sentinel-1	https://catalog.terradue.com/sentinel1/description
EO Data/Sentinel-2	https://catalog.terradue.com/sentinel2/description
EO-based products/Sentinel-1 Medium-Resolution InSAR Browse	https://catalog.terradue.com/dlr-insarql/search?pt=DPHASE

Tree Menu Preview

- EO Data
  - Sentinel-1
  - Sentinel-2
- EO-based products
  - Sentinel-1 Medium-Resolution InSAR Browse

Step 5: configure the data management features



Additional data features available in your app

Data search

Allow to search over defined data collections and list search results within a Search Result tab

☑ Enable Search Result Tab

Feature Basket

Allow user to temporarily save datesets returned by search results into a dedicated Feature basket

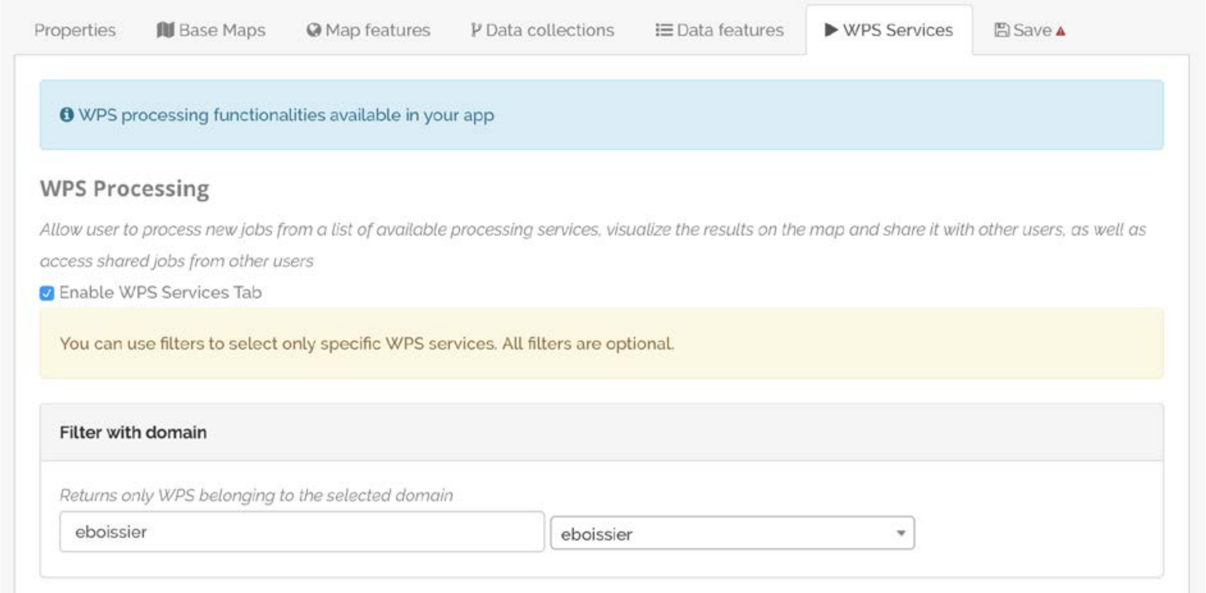
☑ Enable Feature Basket Tab

Features Basket

Total results 4 | sel.all | inv.sel. | | Remove all | Save

[Act-576/Call-657] Fuego Volcano in Guatemala
[Act-575/Call-656] Fire in China
[Act-574/Call-655] Cyclone Mekunu in Yemen
[Act-573/Call-654] Flood in Sri Lanka

Step 6: configure the WPS services to be accessed



WPS processing functionalities available in your app

WPS Processing

Allow user to process new jobs from a list of available processing services, visualize the results on the map and share it with other users, as well as access shared jobs from other users

☑ Enable WPS Services Tab

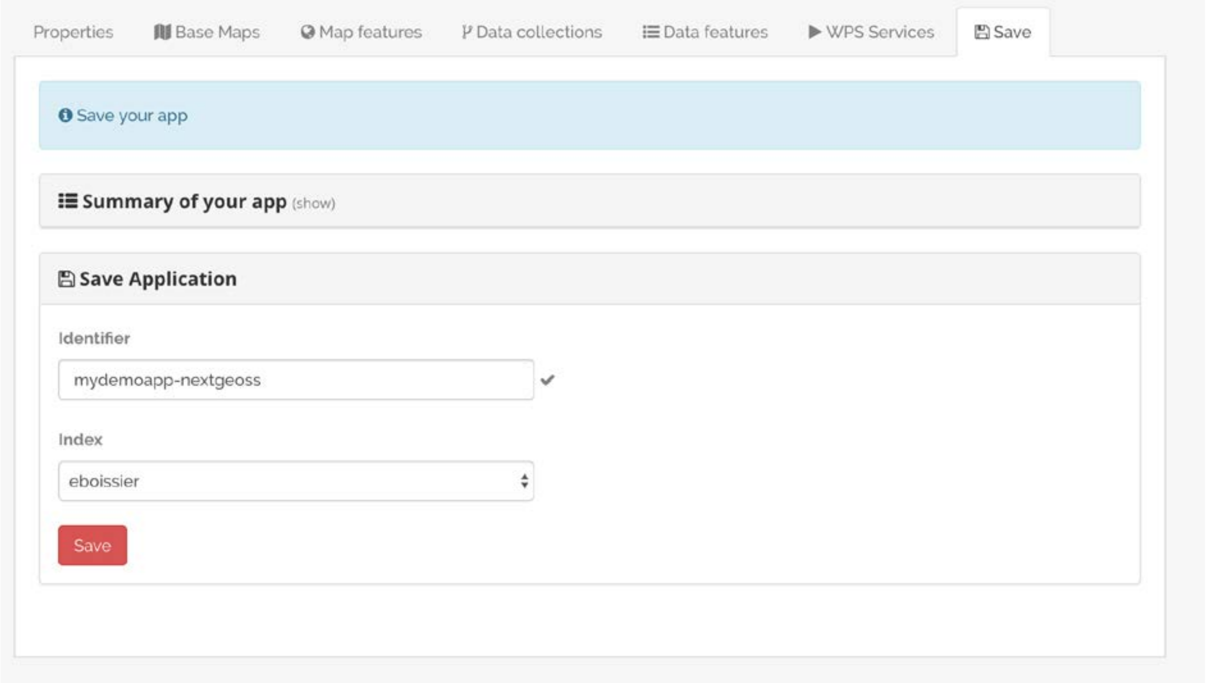
You can use filters to select only specific WPS services. All filters are optional.

Filter with domain

Returns only WPS belonging to the selected domain

eboissier | eboissier

Step 7: save the result as an OWS Context file



Save your app

Summary of your app (show)

Save Application

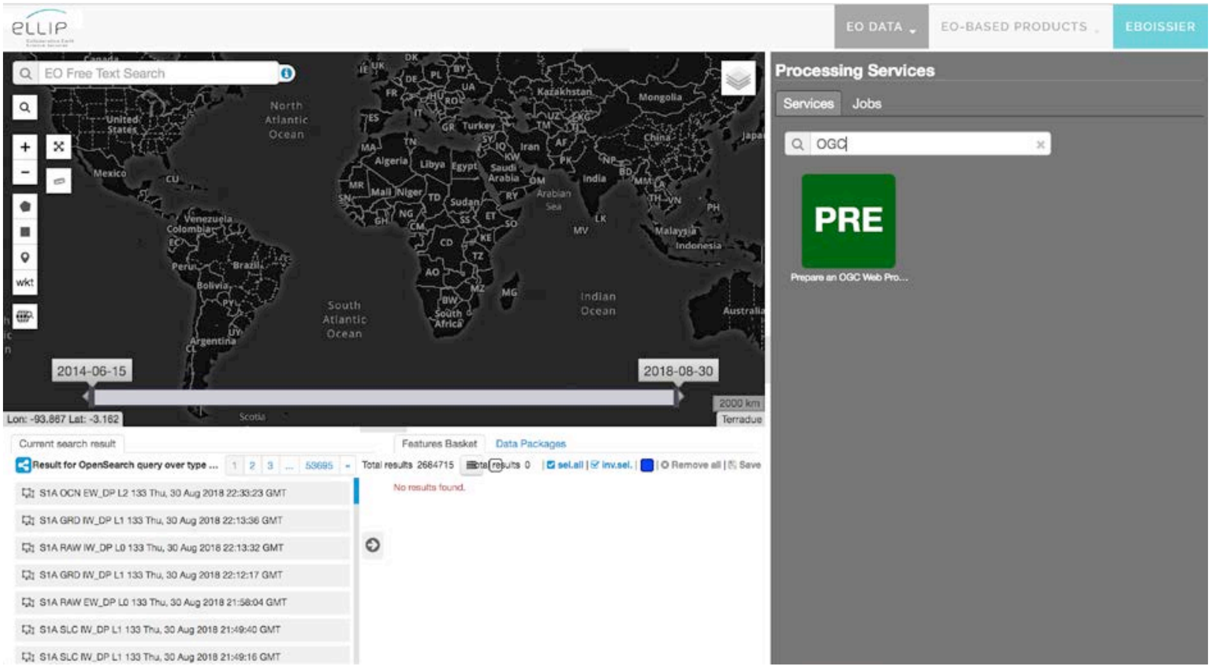
Identifier

mydemoapp-nextgeoss ✓

Index

eboissier

Finally, users can access the instantiated Infohub via a group management feature:



S1A OCN EW_DP L2 133 Thu, 30 Aug 2018 22:33:23 GMT
S1A GRD IW_DP L1 133 Thu, 30 Aug 2018 22:13:36 GMT
S1A RAW IW_DP L0 133 Thu, 30 Aug 2018 22:13:32 GMT
S1A GRD IW_DP L1 133 Thu, 30 Aug 2018 22:12:17 GMT
S1A RAW EW_DP L0 133 Thu, 30 Aug 2018 21:58:04 GMT
S1A SLC IW_DP L1 133 Thu, 30 Aug 2018 21:49:40 GMT
S1A SLC IW_DP L1 133 Thu, 30 Aug 2018 21:49:16 GMT

The user documentation for the Ellip Infohubs solution is not yet available online.

- Online user documentation: upcoming

3. Client software tools to query remote repositories

INTAROS partners have the possibility to use standard ways to discover and retrieve datasets from distributed repositories that are connected to the iAOS via an interoperability protocol.

The supported data discovery standard protocols are:

- OGC OpenSearch interface
- OpenDAP THREDDS interface
- PANGAEA data access service interface

For each protocol, the INTAROS project provides client software tools, tested and validated for working with selected distributed data repositories. All the identified client software presented hereafter are made available on Terradue Cloud Platform and can be used by the project partners from within the Ellip Solutions (the iAOS application integration environment).

3.1. OpenSearch Catalog client - OpenSearch for EO data

OpenSearch is a specification of information sharing formats that natively provisions for extensions. The specification document is available online:
https://github.com/dewitt/opensearch
https://github.com/dewitt/opensearch/blob/master/opensearch-1-1-draft-6.md

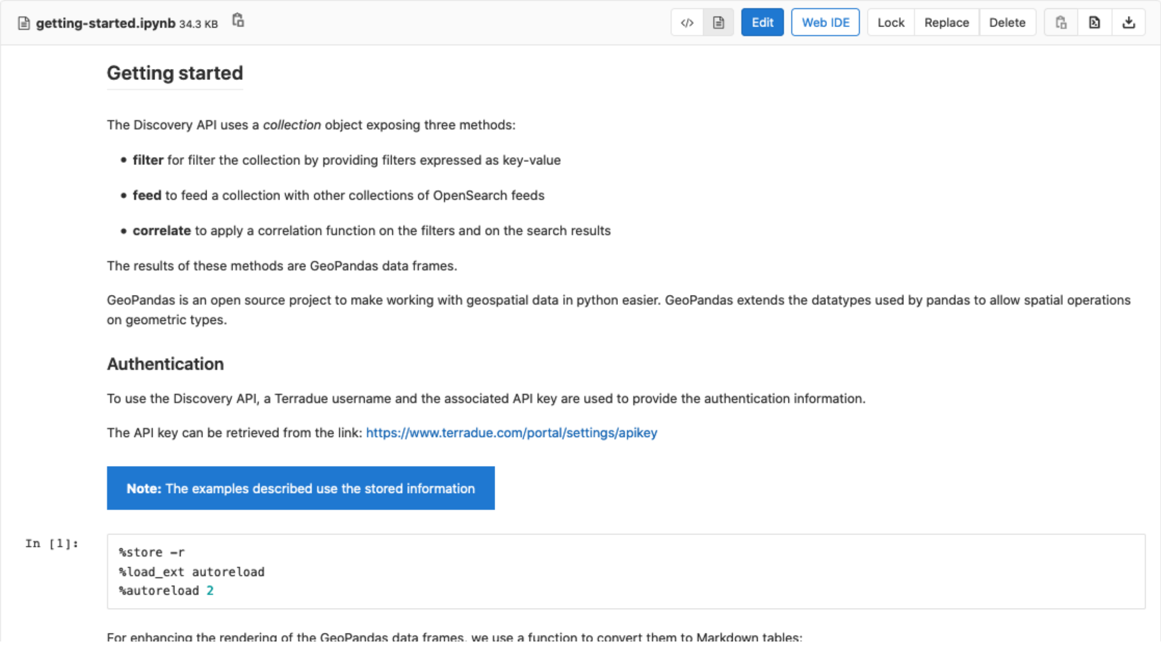
The Open Geospatial Consortium (OGC, https://opengeospatial.org) has specified a number of OpenSearch extensions in order to address the search semantics of geospatial and earth observations data products.
The technology implementing this corpus of specifications for iAOS is the catalogue for EO Data hosted on Terradue's servers. It is compliant with the OGC OpenSearch Geo and Time extensions, as well as the OGC OpenSearch extension for EO, and therefore allows the syndication of distributed repositories that contain information on Earth Observation data products.

For client application software interacting with such Catalog servers, the OpenSearch specification defines an **OpenSearch Description Document** (**OSDD**) format in order to expose the capabilities of a search engine that can be used by search client applications.

The OpenSearch catalog client provided to iAOS partners as part of Terradue's Ellip SDK uses the OSDD to learn about the public interface of the catalog server. The OSDD contains parameterized URL templates, that indicate how the search client should place search requests.

The source code of the SDK libraries for Python are provided online here:
https://gitlab.com/ellip/incubator/catalogc
Online documentation is also provided in the form of Jupyter notebook files:
https://gitlab.com/ellip/incubator/catalogc/-/tree/master/docs

Getting-started use cases guidance is provided through a dedicated Jupyter Notebook file https://gitlab.com/ellip/incubator/catalogc/-/blob/master/docs/getting-started.ipynb



Getting started

The Discovery API uses a *collection* object exposing three methods:

- **filter** for filter the collection by providing filters expressed as key-value
- **feed** to feed a collection with other collections of OpenSearch feeds
- **correlate** to apply a correlation function on the filters and on the search results

The results of these methods are GeoPandas data frames.

GeoPandas is an open source project to make working with geospatial data in python easier. GeoPandas extends the datatypes used by pandas to allow spatial operations on geometric types.

Authentication

To use the Discovery API, a Terradue username and the associated API key are used to provide the authentication information.

The API key can be retrieved from the link: https://www.terradue.com/portal/settings/apikey

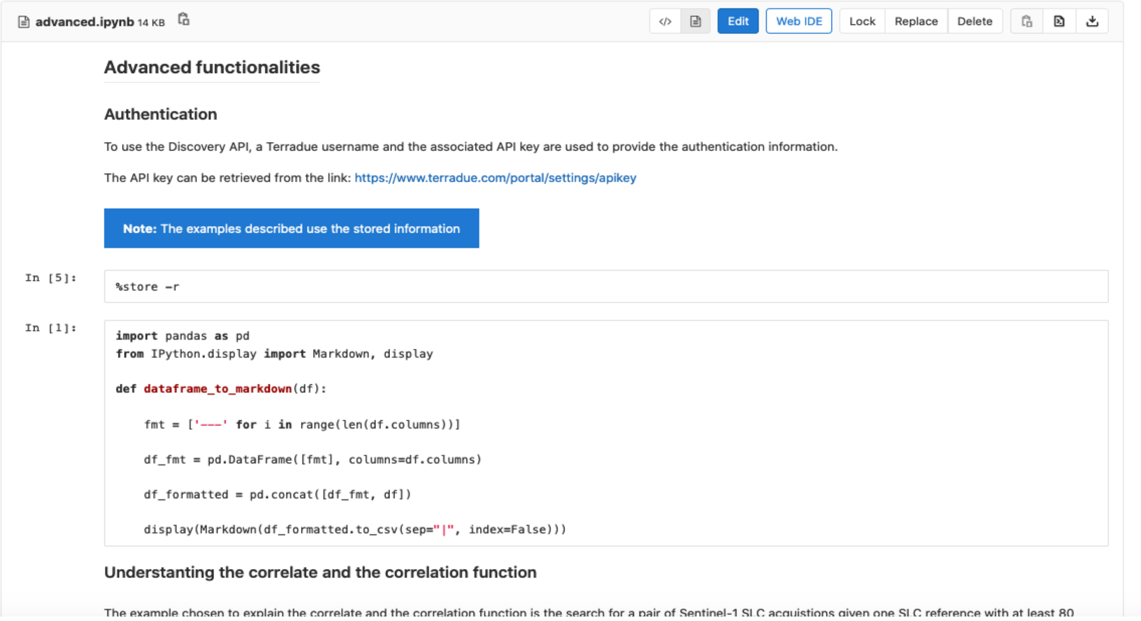
Note: The examples described use the stored information

In [1]:

```
%store -r
%load_ext autoreload
%autoreload 2
```

For enhancing the rendering of the GeoPandas data frames, we use a function to convert them to Markdown tables:

Advanced use cases guidance is provided through a dedicated Jupyter Notebook file https://gitlab.com/ellip/incubator/catalogc/-/blob/master/docs/advanced.ipynb



Advanced functionalities

Authentication

To use the Discovery API, a Terradue username and the associated API key are used to provide the authentication information.

The API key can be retrieved from the link: https://www.terradue.com/portal/settings/apikey

Note: The examples described use the stored information

In [5]:

```
%store -r
```

In [1]:

```
import pandas as pd
from IPython.display import Markdown, display

def dataframe_to_markdown(df):

    fmt = ['---' for i in range(len(df.columns))]

    df_fmt = pd.DataFrame([fmt], columns=df.columns)

    df_formatted = pd.concat([df_fmt, df])

    display(Markdown(df_formatted.to_csv(sep="|", index=False)))
```

Understanting the correlate and the correlation function

The example chosen to explain the correlate and the correlation function is the search for a pair of Sentinel-1 SLC acquisitions given one SLC reference with at least 80

A command line tool is also made available for other types of usage than Python application development. It is documented online here: https://docs.terradue.com/ellip/developer/sdk/bash_commands_functions/catalogue/opensearch-client.html

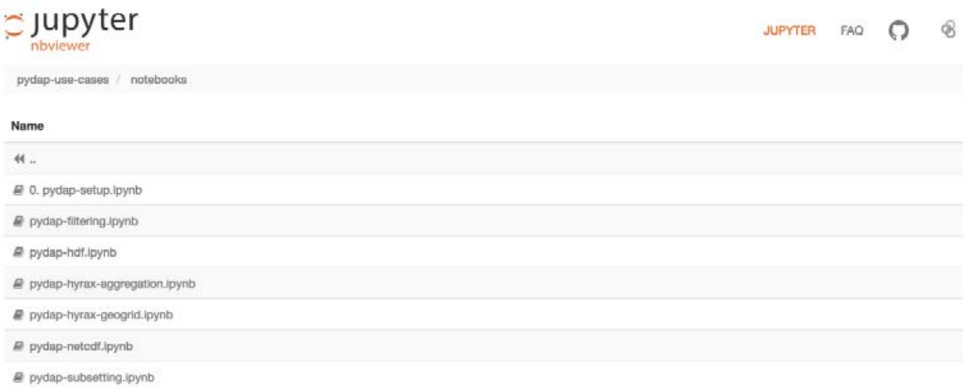
3.2. OpenDAP/THREDDS Catalog client

The PyDap client software (https://www.pydap.org/en/latest/client.html) is made available on Terradue Cloud Platform as part of the Conda repository (Python software libraries).

Hereafter, an example to query a THREDDS catalog as performed by the ARMINES partners as part of their Ellip Workflows application based on RGeostats (the example of the OpenDAP server instance is provided by the IMR partner in the context of iAOS):

```
# The following code demonstrates use of THREDDS catalog, to find list of datasets available for processing

def
load_all_data(geo_bounds,min_date,max_date,min_depth,max_depth,server_type="thredds",verbose=0,show
_time=0):
        #Download sub-catalog
        if server_type is "thredds":
        catalog_url = "http://opendap1-test.nodc.no/thredds/catalogs/physics/physics_point_yearly.xml"
        else:
        catalog_url = "http://opendap1.nodc.no/opendap/physics/point/yearly/catalog.xml"
        contents = urllib.request.urlopen(catalog_url).read()

        #Parse xml
        e = xml.etree.ElementTree.fromstring(contents)

        #Find dataset list
        root_dataset = e.find('{http://www.unidata.ucar.edu/namespaces/thredds/InvCatalog/v1.0}dataset')
        for child in root_dataset:
        url = "http://opendap1.nodc.no"+child.get('ID')
        if server_type is "thredds":
        url = "http://opendap1-test.nodc.no/thredds/dodsC/"+child.get('urlPath')
        else:
        url = "http://opendap1.nodc.no"+child.get('ID')

        min_time = date2time(min_date)
        max_time = date2time(max_date)
        time_depth_bounds = geometry.box(min_time,min_depth,max_time,max_depth)
        load_data(url,geo_bounds,time_depth_bounds,verbose,show_time)
```

A user guidance is provided online in the form of Jupyter Notebook examples here:
https://nbviewer.jupyter.org/github/ec-intaros/pydap-use-cases/tree/master/notebooks/



pydap-use-cases / notebooks

Name

..

0. pydap-setup.ipynb

pydap-filtering.ipynb

pydap-hdf.ipynb

pydap-hyrax-aggregation.ipynb

pydap-hyrax-geogrid.ipynb

pydap-netcdf.ipynb

pydap-subsetting.ipynb

A key capability supported by the DAP protocol, and illustrated in this guidance, is the "Index sub-setting". That is choosing parts of an array based on the indexes of that array's dimensions. This operation always returns an array of the same rank as the original, although the size of the return array will (likely) be smaller. Index sub-setting uses the bracket syntax described subsequently.

Sub-setting fixed-size arrays in their index space is accomplished using square brackets. For an array with N dimensions, N sets of brackets are used, even if the array is only subset on some of the dimensions. The names of array variables are fully qualified names (FQNs) so it's possible to name arrays in structures and/or Groups. Array index values are zero-based as with a number of programming languages such as C and Java. Every array has a known starting index value of zero.

Within the square brackets, several subexpressions are allowed:

[ ] return all of elements elements for a particular dimension.
[n] return only the value at a single index, where 0 <= n < N for a dimension of size N.
This slicing operator does not reduce the dimensionality of an array, but does return a dimension size of one for the dimension to which this is applied.

[start : step : last] return every value whose index is in the range start <= index <= last
and where (index - start) % step == 0.
This is the complete version of the syntax.

[start : last] return the values whose index is in the range start <= index <= last.
[start :] return the values whose index is in the range start <= index <= the dimension size - 1.
[start : step :] return every value whose index is in the range start <= index <= dimension size - 1
and where (index - start) % step == 0.

Sub-setting can be applied to any array.
It can also be applied to a scalar, but in this case, the only legal forms are [0] or [].

3.3. Pangaea Catalog client

The PANGAEA data download service allows a client application to query the FRAM data warehouse and filter by geocodes and parameters. The service returns values in tab delimited text files. The usage of the PANGAEA data download service interface is documented online here:
http://ws.pangaea.de/dds-fgp/

In substance, queries to the PANGAEA data warehouse return values in a text file, based on search criteria that (1) specify a bounding box in time and space (latitude, longitude and water depth), and (2) specify a list of parameters.

The service returns a text file containing the following columns:
Date/Time, Latitude (north), Longitude (east), Depth water [m], Parameter 1, Parameter 2, ..., Data source

In summary:

Mandatory search parameters	Optional search parameters
@westBoundLon description: west bound longitude example: westBoundLon=-96.5667 **@southBoundLat** description: south bound latitude example: southBoundLat=-13.4251 **@eastBoundLon** description: east bound longitude example: eastBoundLon=-96.9987 **@northBoundLat** description: north bound latitude example: northBoundLat=-12.9052 **@pangaeaParameterIDs** description: the parameters to retrieve (internal PANGAEA parameter ID's need to be provided) example: pangaeaParameterIDs=717,716,80103,756,128305	**@dateTimeMin** description: datetime minimum (ISO 8601) example: dateTimeMin=2011-04-14T00:00:00 **@dateTimeMax** description: datetime maximum (ISO 8601) example: dateTimeMax=2011-04-16T00:00:00 **@depthMin** description: minimal water depth (metres) example: depthMin=45 **@depthMax** description: maximal water depth (metres) example: depthMax=55 **@depthMinSed** description: minimal sediment/rock depth (metres) example: depthMinSed=0.01 **@depthMaxSed** description: maximal sediment/rock depth (metres) example: depthMaxSed=0.15 **@projectID** description: internal PANGAEA project ID example: projectID=4055

Service call examples:

Example 1:
http://ws.pangaea.de/dds-fgp/rest/dwhquery?westBoundLon=-96.9984&southBoundLat=-13.41512&eastBoundLon=-96.9987&northBoundLat=-13.41124&pangaeaParameterIDs=717

Description: service call using only the mandatory search parameters, which returns all temperature data archived at PANGAEA within the specified bounding box.

Example 2:
http://ws.pangaea.de/dds-fgp/rest/dwhquery?westBoundLon=-96.5667&southBoundLat=-13.42512&eastBoundLon=-96.9987&northBoundLat=-12.9052&pangaeaParameterIDs=717,716,80103,756,128305&depthMin=45&depthMax=55&dateTimeMin=2011-04-14T00:00:00&dateTimeMax=2011-04-16T00:00:00

Description: service call using mandatory and optional search parameters, which returns all values archived at PANGAEA for the specified PANGAEA parameter ID's within a bounding box defined in horizontal space (latitude, longitude), vertical space (water depth) and time.

Example 3:
http://ws.pangaea.de/dds-fgp/rest/dwhquery?pangaeaParameterIDs=24645&westBoundLon=-15&southBoundLat=75&eastBoundLon=15&northBoundLat=82&dateTimeMin=2000-01-01T00:00:00&dateTimeMax=2001-01-01T00:00:00&projectID=4055

Description: service call using mandatory and optional search parameters, which returns all values archived at PANGAEA for the specified PANGAEA parameter ID's within a bounding box defined in horizontal space (latitude, longitude) and time. The internal PANGAEA project ID is used as an additional filter.

Example 4:
http://ws.pangaea.de/dds-fgp/rest/dwhquery?pangaeaParameterIDs=157811,4870&westBoundLon=-4&southBoundLat=79&eastBoundLon=5&northBoundLat=80&projectID=4055&depthMinSed=0.001&depthMaxSed=0.1&dateTimeMin=2009-07-15T06:45:00&dateTimeMax=2009-007-15T06:47:00

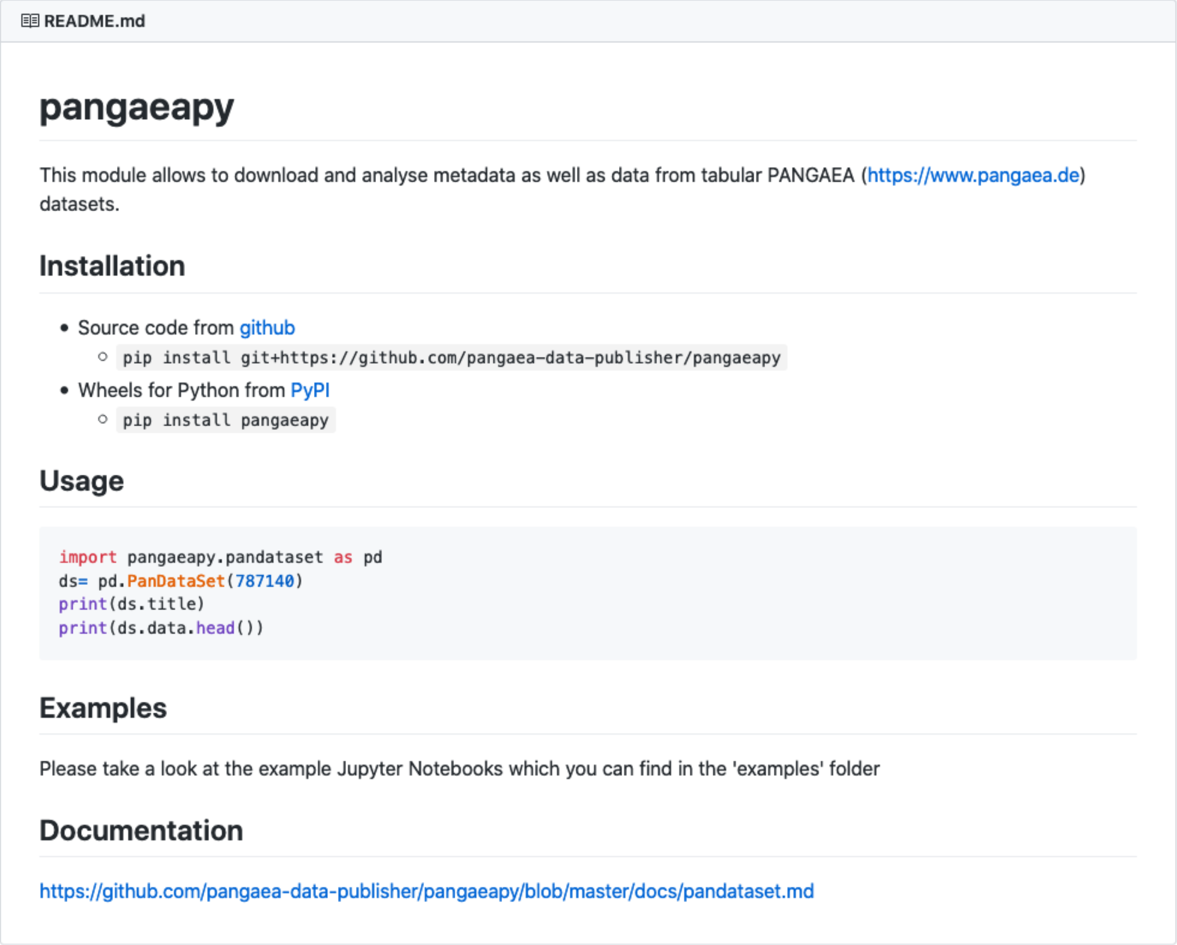
Description: service call using mandatory and optional search parameters including sediment depth, which returns all values archived at PANGAEA for the specified PANGAEA parameter ID's within a bounding box defined in horizontal space (latitude, longitude) and time. The internal PANGAEA project ID is used as an additional filter.

Finally, tools developed and made available in this regard to the INTAROS partners are available as Jupyter Notebook files, accessible from AWI’s contributed Git repositories:

1) PANGAEA Data Publisher for Earth & Environmental Science

A collection of tools curated on the PANGAEA GitHub public repository https://github.com/pangaea-data-publisher

In particular **pangaeapy** is a Python Client module that allows to download and analyse metadata as well as data from tabular PANGAEA (https://www.pangaea.de) datasets. Usage examples in Python are provided as Jupyter Notebook files (.ipynb files).



README.md

pangaeapy

This module allows to download and analyse metadata as well as data from tabular PANGAEA (https://www.pangaea.de) datasets.

Installation

- Source code from github
  - `pip install git+https://github.com/pangaea-data-publisher/pangaeapy`
- Wheels for Python from PyPI
  - `pip install pangaeapy`

Usage

```
import pangaeapy.pandataset as pd
ds= pd.PanDataSet(787140)
print(ds.title)
print(ds.data.head())
```

Examples

Please take a look at the example Jupyter Notebooks which you can find in the 'examples' folder

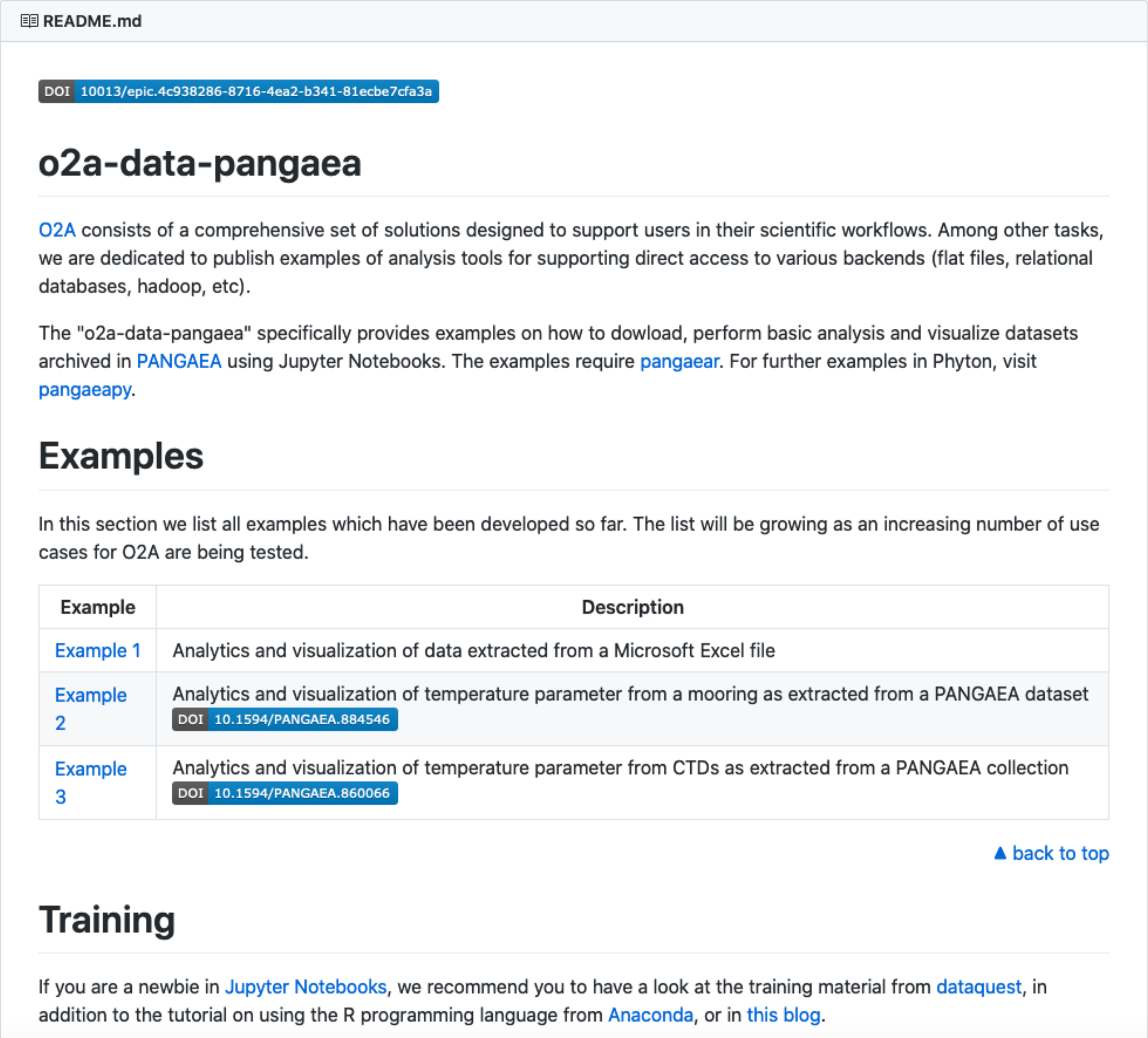
Documentation

https://github.com/pangaea-data-publisher/pangaeapy/blob/master/docs/pandataset.md

2) PANGAEA Data flow & analysis

A collection of tools designed for scientific workflows requiring direct access to various backends (flat files, relational databases, hadoop, etc).
https://github.com/o2a-data/o2a-data-pangaea

The "o2a-data-pangaea" specifically provides examples on how to download, perform basic analysis and visualize datasets archived in PANGAEA using Jupyter Notebook. The examples require “pangaear”.



README.md

DOI 10013/epic.4c938286-8716-4ea2-b341-81ecbe7cfa3a

o2a-data-pangaea

O2A consists of a comprehensive set of solutions designed to support users in their scientific workflows. Among other tasks, we are dedicated to publish examples of analysis tools for supporting direct access to various backends (flat files, relational databases, hadoop, etc).

The "o2a-data-pangaea" specifically provides examples on how to dowload, perform basic analysis and visualize datasets archived in PANGAEA using Jupyter Notebooks. The examples require pangaear. For further examples in Phyton, visit pangaeapy.

Examples

In this section we list all examples which have been developed so far. The list will be growing as an increasing number of use cases for O2A are being tested.

Example	Description
Example 1	Analytics and visualization of data extracted from a Microsoft Excel file
Example 2	Analytics and visualization of temperature parameter from a mooring as extracted from a PANGAEA dataset DOI 10.1594/PANGAEA.884546
Example 3	Analytics and visualization of temperature parameter from CTDs as extracted from a PANGAEA collection DOI 10.1594/PANGAEA.860066

▲ back to top

Training

If you are a newbie in Jupyter Notebooks, we recommend you to have a look at the training material from dataquest, in addition to the tutorial on using the R programming language from Anaconda, or in this blog.

4. Summary

The cloud platform has the objective to integrate multidisciplinary and distributed data repositories and provide a set of tools for data analysis, transformation and visualization. Addressing this goal, the report is describing the cloud platform services that are made available to project partners in order to access observations and derived parameters, including the new observations from WP2-3-4.

This report is defining the second version of the iAOS cloud platform and tools, updating the work initially reported as part of the INTAROS Deliverable 5.2. The platform services and tools improvements added from the initial iteration are covering:

- the evolutions of the Ellip Software Development Kit (SDK) Handbook;
- the user guidance for using the Cloud Platform's Ellip Solutions portfolio;
- the user guidance for using the Client software tools to query remote repositories (OpenSearch Catalog client, OpenDAP/THREDDS Catalog client, Pangaea Catalog client).

Selected applications established in WP6 will use the services and tools provided by the cloud platform, with the main purpose of demonstrating the usefulness and functionality of the platform. These applications will showcase the benefit of enhanced integration of data from Arctic observing systems.

----------- END of DOCUMENT-----------

-

INTAROS

This report is made under the project
Integrated Arctic Observation System (INTAROS)
funded by the European Commission Horizon 2020 program
Grant Agreement no. 727890.

Project partners: