Reinforcing the AI4EU Platform by Advancing Earth Observation Intelligence, Innovation and Adoption

D3.1: Architecture, semantics and discovery report

Grant Agreement ID	101016798	**Acronym**	AI4COPERNICUS
Project Title	Reinforcing the AI4EU Platform by Advancing Earth Observation Intelligence, Innovation and Adoption		
Start Date	01/01/2021	**Duration**	36 Months
Project URL	https://ai4copernicus-project.eu/		
Contractual due date	30/06/2022	**Actual submission date**	30/06/2022
Nature	R	**Dissemination Level**	PU
Author(s)	Manolis Koubarakis, Despina-Athanasia Pantazi, Dharmen Punjani, George Stamoulis, Eleni Tsalapati (UoA), Iraklis Klampanos, Antonis Koukourikos, Antonis Troumpoukis (NCSR-D), Vasileios Baousis, Mihai Alexe (ECMWF), Marcin Ziółkowski (CF), David Pérez (TAS)		
Contributor(s)	Martin Welß (Fraunhofer IAIS)		
Reviewer(s)	Costas Spyropoulos (NCSR-D)		

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 101016798.

Document Revision History *(including peer reviewing & quality control)*

Version	Date	Changes	Contributor(s)
v0.1	17/03/21	First version of Architecture	Despina-Athanasia Pantazi, George Stamoulis, Manolis Koubarakis
v1	06/06/22	Second version (pending review by EC)	All

Executive Summary

This is the first deliverable of WP3 (Technical positioning and architecture) and, more specifically, Tasks 3.1 (Architecture specification, tools and components) and 3.2 (Design of the semantic catalogue and the semantic search and discovery functionality). In this deliverable we present the architecture of AI4Copernicus and discuss how the software that we are developing interacts with various components of the AI-on-demand platform, and the two DIASes targeted by the project, CREODIAS and WEkEO. This deliverable also presents two original contributions to the Copernicus ecosystem: EarthQA, which is a question answering engine for discovering Copernicus data, and the Copernicus ontology.

Table of Contents

List of Figures

List of Tables

List of Terms & Abbreviations

Abbreviation	Definition
AC	Atmospheric Composition
ACD	Amplitude Change Detection
AEM	AI4EU Experiments Management
AI	Artificial Intelligence
AIoD	AI-on-demand
AIS	Automatic Identification System
AoI	Area of Interest
AQ	Air Quality
ARD	Analysis Ready Data
BFO	Basic Formal Ontology
BoA	Bottom of Atmosphere
C3S	Copernicus Climate Change Service
CAMS	Copernicus Atmosphere Monitoring Service
CD	Change Detection
CDS	Climate Data Store
CM	Collaborative Management
CO	Copernicus Ontology
CSV	Comma-Separated Values
CVA	Change Vector Analysis
DEM	Digital Elevation Model
DIAS	Data and Information Access Services
DL	Deep Learning
DTE	Digital Twin Earth

EC	European Commission
ECTL	Extract, Cleanse, Transform, Load
EO	Earth Observation
ESA	European Space Agency
EU	European Union
GA	Grant Agreement
GAN	Generative Adversarial Network
GDAL	Geospatial Data Abstraction Library
GDPR	General Data Protection Regulation
GHG	Greenhouse gas
GPS	Global Positioning System
GPT	Graph Processing Tool
GRD	Ground Range Detected
HDA	Harmonized Data Access
HKTM	Housekeeping Telemetry
IoT	Internet of Things
IW	Interferometric Wide
JRC	Joint Research Centre
JSON	JavaScript Object Notation
KG	Knowledge Graph
KM	Kilometer
L1C	Level 1C
L2A	Level 2A
LAI	Leaf Area Index
LOD	Linked Open Data
LSTM	Long Short-Term Memory
ML	Machine Learning
MSI	Multispectral Instrument
MTC	Multi-Temporal Coherence
NN	Neural Network
OE	Ontology Expert
OGC	Open Geospatial Consortium
OM	Ontology of units of Measure
OSM	Open Street Map
ORKG	Open Research Knowledge Graph
OWL	Web Ontology Language
PM	Particulate Matter
QA	Question Answering
RDF	Resource Description Framework
S1	Sentinel-1
S2	Sentinel-2
SAMOD	Simplified Agile Methodology for Ontology Development
SAR	Synthetic Aperture Radar
SLC	Single Look Complex

SM	Strip Map
SNAP	Sentinel Application Platform
SOSA	Sensor, Observation, Sampe, and Actuator
SKOS	Simple Knowledge Organization System
SLC	Single Look Complex
SRGAN	Super-resolution GAN
SRTM	Shuttle Radar Topography Misiion
SSN	Semantic Sensor Network
TOPSAR	Terrain Observation Progressive Scans SAR
UTM	Universal Transverse Mercator
W3C	World Wide Web Consortium
WG	Working Group
WGS	World Geodetic System
WKT	Well-known text
WP	Work Package
XML	eXtensible Markup Language
YAML	YAML Ain't Markup Language

1 Introduction

This is the first deliverable of WP3 (Technical positioning and architecture) and, more specifically, of Tasks 3.1 (Architecture specification, tools and components) and 3.2 (Design of the semantic catalogue and the semantic search and discovery functionality).

1.1 Purpose and Scope

The purpose of this deliverable is to present the software architecture of AI4Copernicus and to discuss how it interfaces with the various components of the AI-on-demand platform, and the two DIASes targeted by the project, CREODIAS and WEkEO. This deliverable also presents two original contributions to the Copernicus ecosystem: EarthQA, which is a question answering engine for discovering Copernicus data, and the Copernicus ontology. The development of these two contributions will be presented in more detail in D4.2.

1.2 Approach for Work Package and Relation to other Work Packages and Deliverables

Work package WP3 (Technical positioning and architecture) started on M1 and ends on M30 of the project. It is led by partner UoA with the collaboration of partners NCSR-D, TAS, CF and UNITN. WP3 positions technically AI4Copernicus in the European AI and Copernicus ecosystems. In addition, it develops the software architecture of the project.

WP3 has the following three tasks:

- Task 3.1 Architecture specification, tools and components (M1-M18, lead: UoA, contributors: NCSR-D, TAS, CF, UNITN). The technical contribution of this task is the development of the software architecture of the project with a specific emphasis to interfacing with the AI-on-demand platform, CREODIAS and WEkEO.
- Task 3.2 Design of the semantic catalogue and the semantic search and discovery functionality (M1-M9, lead: UoA, contributor: NCSR-D). The technical contributions of this task are the development of a question answering engine for discovering Copernicus data, and the development of the Copernicus ontology.
- Task 3.3 Positioning of AI4Copernicus in the European AI and Copernicus ecosystems (M7-M30, lead: UoA, contributors: TAS, UNITN). This task monitors the AI and Copernicus landscape in Europe and positions technically AI4Copernicus in this landscape.

The present deliverable D3.1 is the first deliverable of WP3 and contains the contributions of the project in Task 3.1 and Task 3.2. WP3 has two more deliverables that target Task 3.3:

- D3.2 AI4Copernicus and the European AI and Copernicus ecosystems - report I (M18, R, PU, UoA)
- D3.3 AI4Copernicus and the European AI and Copernicus ecosystems - report II - final, (M30, R, PU, UoA)

The architecture and software components designed in WP3 are implemented in WP4 (Implementation, customisation, integration and testing). WP4 started on M4 and ends on M24. It is led by partner CF with the participation of partners NCSR-D, UoA, TAS, ECMWF and UNITN.

The following tasks of WP4 are relevant to WP3:

- Task 4.1: Integration of AI-on-demand platform with CREODIAS/WEkEO (M4-M12, lead: CF, contributor: TAS). This task implements the architecture designed in Task 3.1.
- Task 4.3: Implementation of the semantic catalogue and the semantic search and discovery functionality (M4-M12, lead: UoA, contributor: NCSR-D). This task implements the question answering engine designed in Task 3.2.

The following deliverables of WP4 are relevant to WP3:

- D4.1 Integration report on AI4EU tools with DIAS platforms (M18, R, P, CF)
- D4.2 Semantic search and discovery tools (M18, D, P, UoA)

These deliverables implement the software components whose design was presented in D3.1.

1.3 Organization of the Deliverable

The rest of the deliverable is organized as follows. Section 2 presents the current state of the AI-on-demand platform. Sections 3 and 4 present CREODIAS and WEkEO. Section 5 discusses the part of the software architecture of AI4Copernicus related to CREODIAS. Section 6 presents the semantic catalogue which is the component of the AI4Copernicus architecture targeted by EarthQA. Section 7 presents EarthQA, the question answering engine itself. Section 8 presents a summary of the deliverable.

2 The European AI-on-demand Platform

The European AI-on-demand (AIoD) platform is the central reference point for the activities and achievements of the AI4EU project and its future continuation actions. Its purpose is to provide one dedicated place for accessing information on different axes of the European AI ecosystem, including people, knowledge, technology and services. Participation on the platform is based on a voluntary paradigm, and users can contribute via their participation in technical/scientific discussions, networking, announcements on jobs and relevant innovations/research results. At a more pragmatic level, the AIoD platform also hosts an extensible and comprehensive catalogue of resources, as detailed in the following subsection.

2.1 AI-on-demand Platform Resources

The AI-on-demand Platform Resource Catalogue hosts and exposes information on various resources

relevant to AI research and application. Resources of different modalities (publications, models, software, etc.) are added to the Resource Catalogue by carrying out the publication process within the platform.

The publication process foresees the provision of information for the resource, primarily including:

1. General characteristics of the resource (name, relevant research areas, relevant application areas).
2. Documentation accompanying the resource (manuals, user and developer guides, links to wikis and websites, etc.).
3. Authorship, ownership and IPR information.
4. Legal and ethical aspects, including compliance with GDPR.
5. Description on the different ways that an interested party can get support for using the resource.
6. When applicable, a general description of the performance of the resource in different experimental/application contexts.

The aforementioned information is stored in the platform's database with its organization driven by the AI4EU Data Model. The latter is informed by the general AI4EU Conceptual Model[1] and the corresponding OWL ontology, with the necessary transformations for its transition to a relational schema.

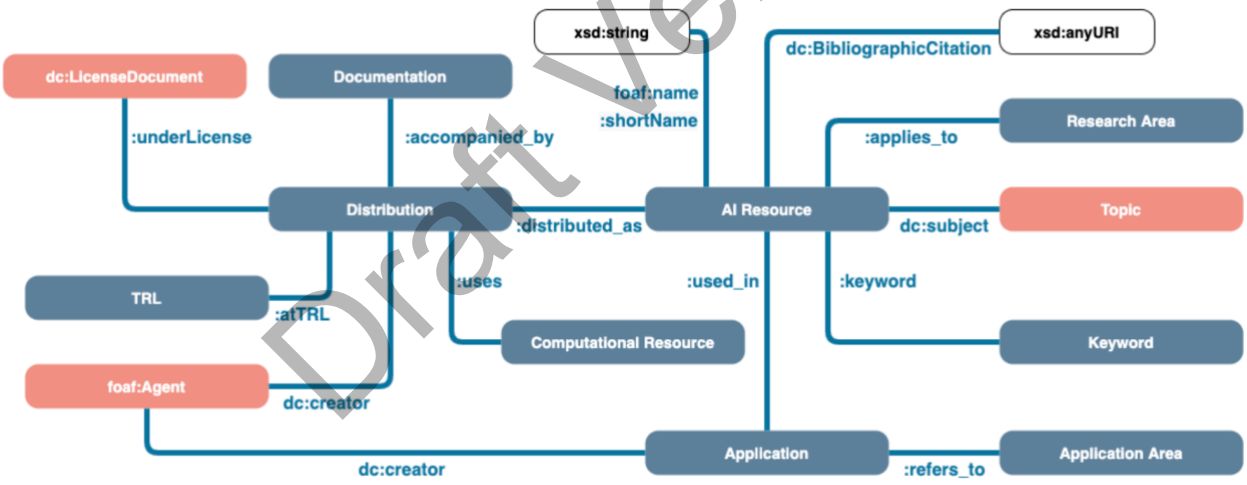


Figure 2.1: AI4EU Conceptual Model informing the AI-on-demand Platform Catalogue

As depicted in figure 2.1, the central concept for the organization of the model is the *AI Resource*. This can be specialized by various subclasses that conceptualize different types of technology e.g., Dataset, Software, Hardware, Ontology, etc. A single resource can potentially have more than one *Distribution*, depending on the packaging, licensing, access mechanisms, and other parameters. Additionally, each distribution is foreseen to be accompanied by its *Documentation*, which can also entail different objects with different characteristics (e.g., Wikis, Text Documents, Manuals, Code

[1] https://github.com/ai4eu/ai-resources-ontology/blob/master/documentation/Deliverable-AI4EU-D3.4-M24-Final.pdf

Tests and Examples, etc.). Such different types of documentation can also be modeled as specializations of the generic *Documentation* class.

Furthermore, an AI Resource is associated with different descriptive entities that help characterize the resource, i.e., the topic(s) it refers to, relevant keywords, and the computational resources to which it mainly relies for functioning.

Another aspect of a resource is its usage on applications, services, products and challenges. These are modeled through the *Application* class that will entail the properties that describe it (name, URL and so on), and is also connected to an *Application Area*. For the latter, corresponding SKOS vocabularies have been defined and incorporated in the OWL definition of the conceptual schema.

All aforementioned resources (the conceptual model, its OWL manifestation and the SKOS vocabulary for research and application areas) can be accessed at the relevant GitHub repository[2].

2.2 The AI-on-demand Platform Search Engine

According to the AI4EU deliverable D2.3 - "Search and Discovery Engine", the platform's Search Engine component is made of sub-components able to search information within given scopes and a front end which handles full scope queries by delegating sub-queries to each sub-component. The given scopes can be the Web, Collaborative Management (CM) data or AI4EU Experiments Management (AEM) data. The implementation uses the Qwant technology for web search.

For modularity reasons, as far as possible, each component manages its data in its (their) own database(s). The content was updated on a daily basis, and includes the following:

- Creation of new user profiles.
- News related to different AI content including Ethics and Women AI Heroes.
- New content in current Groups and Discussions.
- Creation of new groups and discussions by registered users.
- Publication of new AI Resources.

2.2.1 Search Engine processes

Three main processes are engaged into the search component:

1. Query analysis (that is almost identical to document analysis).
2. Document analysis to extract high level information.
3. Result generation: the ranked list of documents augmented with high-level linguistic information.

These processes are illustrated in figure 2.2.

[2] https://github.com/ai4eu/ai-resources-ontology

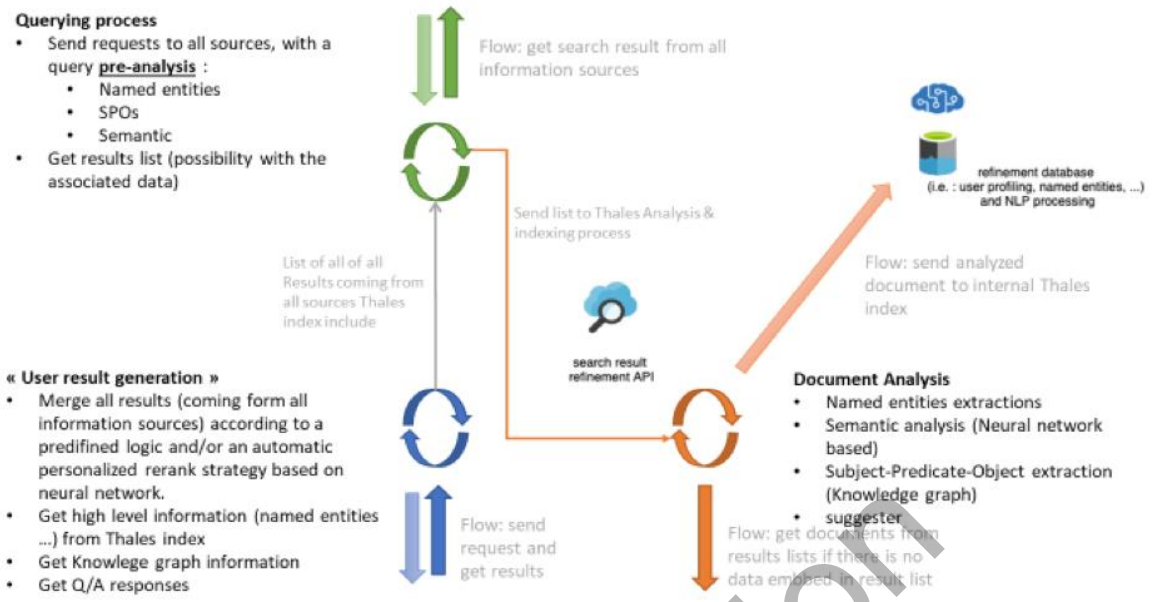


Figure 2.2: The three processes of the search engine

2.2.2 Search Engine flows

The search component API is connected to AIoD Platform's CMS (Drupal). When a user uses the search bar a query is sent, by using a REST API, to the platform's search component. Then the request is formatted for each aggregated source (for example Qwant, Mundi) and sent.

The results are analyzed, indexed, reranked and sent to the CMS:

- Content of results is indexed into an internal index after some linguistics processes. This index allows the storage of linguistic information and is close to the design of ontology proposed by the WP3 of the AI4EU project (D3.4 - "Manual on platform knowledge modeling").
- Results are merged according to a ranking policy. For future work, the reranking strategy will be based on neural networks.
- High level information, such as named entities, ontology, Q/A answers, etc, is added to the results.

The flows are presented in Figure 2.3.

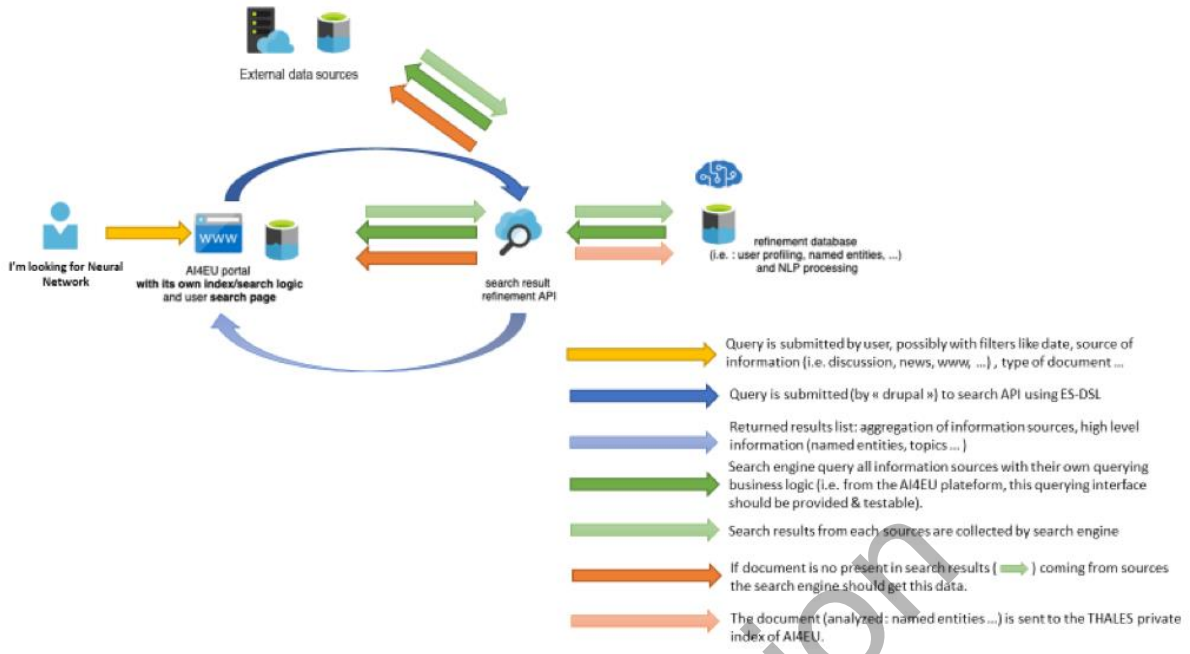
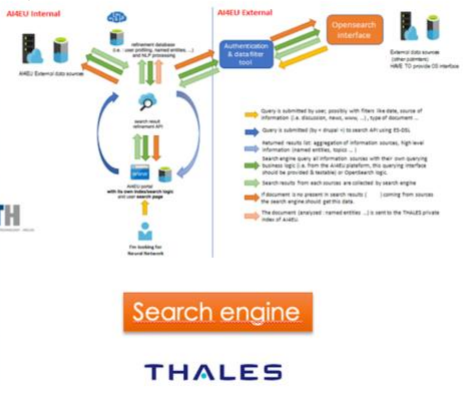


Figure 2.3: Flows in the search engine

2.2.3 Status of current available sources of information and features

At the end of the AI4EU project, the list of sources of information that are available for the search and the list of developed linguistic features is presented in figure 2.4. The list is foreseen to be extended in the context of followup projects, like AI4Europe.

Rich functionalities	TODO / Done to be evaluated
Suggesters	Done…
Automatic summary	Done…
Named entities	Done…
Knowledge graph	Done…
Sientiment analysis	Done…
Unsupervised classification	Done…
Questions/Answers	Done
Chatbot(WP3 links)	On going
Links to Mundi	On going
Semantic index	On going
See also	TODO
User/Group personalized search	TODO



Content	TODO / Done to be evaluated
AI4EU platform	Done…
The Internet Qwant	Done…
Mundi/COPERNICUS	Started
ICT-48	TODO
ICT-49	TODO
Other H2020 AI projects	TODO

Figure 2.4: Current available features

2.3 AI4EU Experiments

The AI4EU Experiments[3] service is an Acumos-based AI model management platform. It incorporates features for developing, training and deploying machine learning models and solutions along with collaboration and networking facilities for sharing and reusing the models.

The AI resources available to use and test via AI4EU Experiments are exposed via the Marketplace (figure 2.5), where the list of resources can be browsed or filtered based on their name, description or categorization.

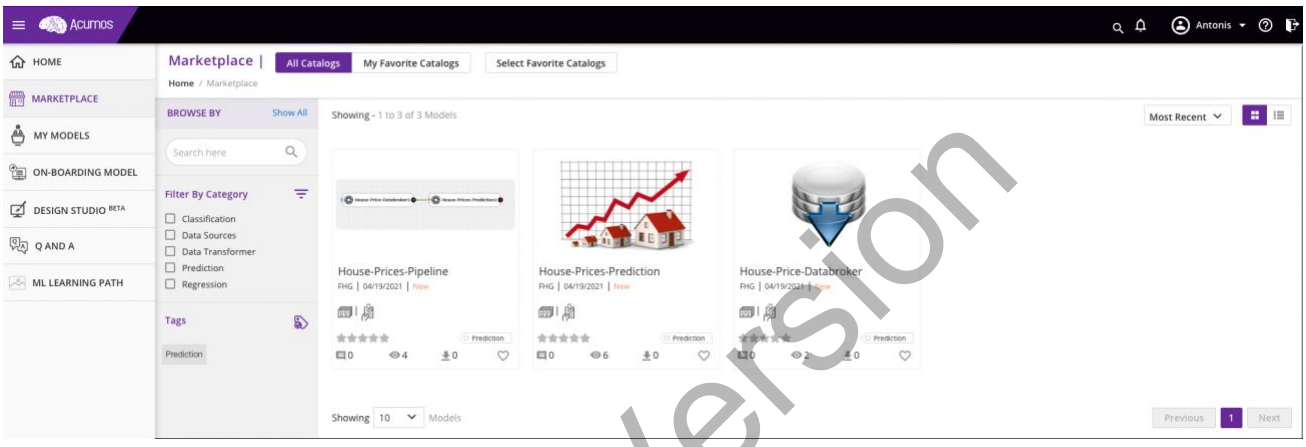


Figure 2.5: The AI4EU Experiments Marketplace

To include an externally created resource in the marketplace, users must dockerize their resource and fill-in the respective on-boarding form within AI4EU Experiments. At this stage, no further information is mandatory, but to be usable within the service, resources must carry a protobuf[4] specification and a license description. Protobuf is a widely used mechanism for serializing structured data and provides features for describing the interface (inputs and outputs) of services and modules, thus allowing the automation of compatibility and usage requirements checks.

[3] https://aiexp.ai4europe.eu/#/home

[4] https://developers.google.com/protocol-buffers/

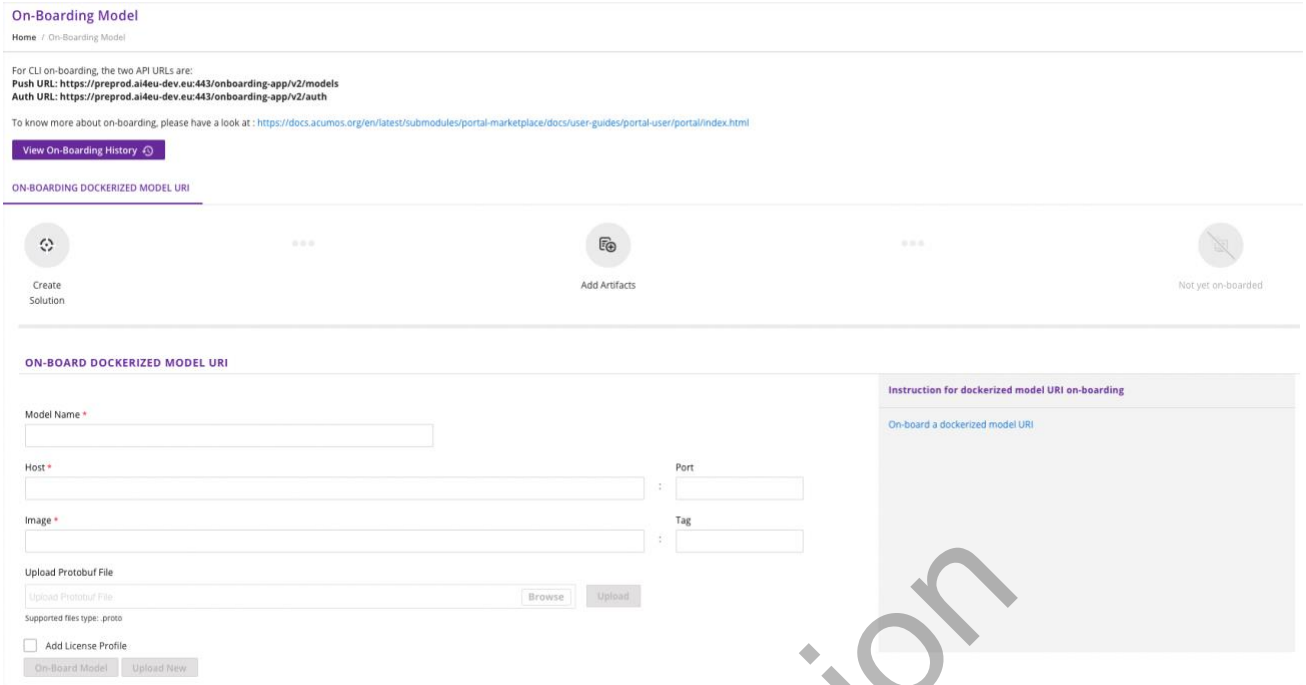


Figure 2.6: The AI4EU Experiments On-boarding form

As soon as the aforementioned information is provided and the resource is on-boarded, details on its scope and usage are visible through its respective page within AI4EU Experiments. The provided information includes licensing and ownership, documents and artifacts included in the resource, and its protobuf signature that details the functionalities exposed by the resource's package, as shown in figure 2.7, and thus the way it can be called and connected to more complex solutions.

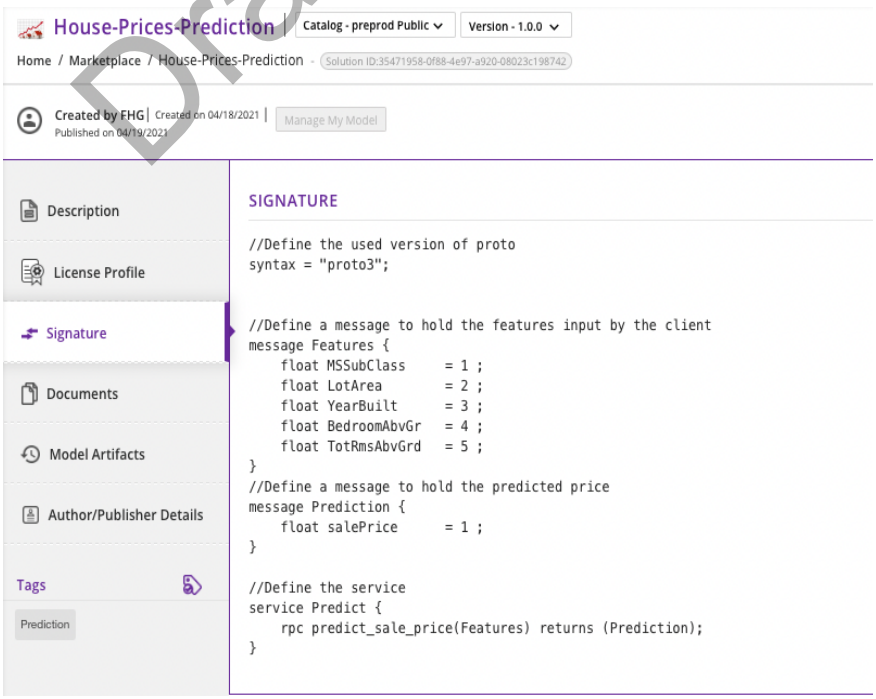


Figure 2.7: A Protobuf signature of an AI resource

Furthermore, users can download elements of the resource or download it as a package deployable in different contexts, including private clusters, as shown below.

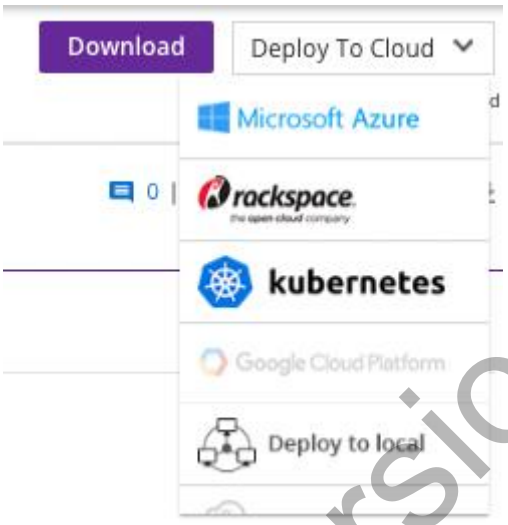


Figure 2.8: Supported deployment contexts

As long as a resource carries its protobuf description, it is available to be used in the composition of more intricate AI-driven solutions via the AI4EU Experiments Design Studio feature.

2.4 AI4EU Experiments Design Studio

The Design Studio is based on the Acumos Acu-compose, a design environment where users can visually design processing workflows (pipelines) by combining compatible marketplace resources. The main elements within the Design Studio, as shown in figure 2.9, are: the Marketplace browser where users can view available models and already published composite solutions (1); the design panel where user drag-and-drop and combine models and solutions (2); and the properties panel where further details on the selected assets are provided, along with information for models with matching inputs/outputs with the selected asset (3).

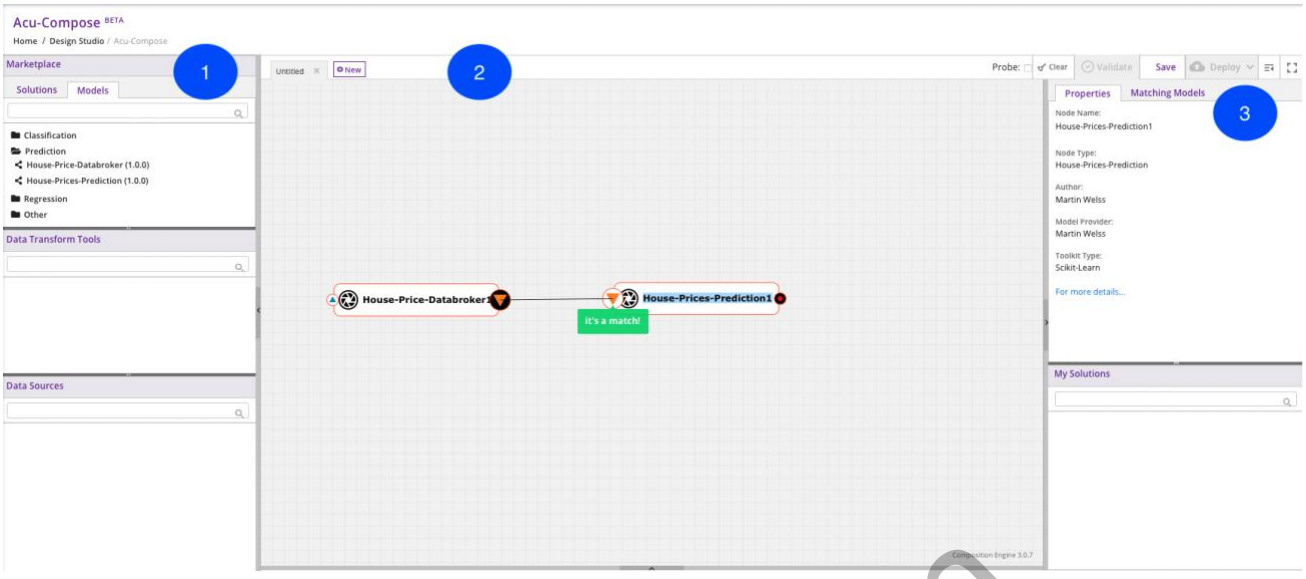


Figure 2.9: The AI4EU Experiments Design Studio

As soon as a user has designed his solution, he can save it within AI4EU Experiments and, as with any resource, retrieve a docker package deployable in different contexts.

3 The CREODIAS Data and Information Access Services

DIAS is a European Commission's concept for Data and Information Access Services to facilitate cloud based data analytics for value adding to Copernicus data. A DIAS (Data and Information Access System) is a platform where all the data related to Copernicus can be found with free and open access, from Sentinel satellites (also found in Open Access Hub[5]), to access all the Services data. It is expected to deliver from all the operative Sentinels, around 10 petabytes every year.

These platforms deliver to the users the ability to exploit Copernicus data and information without having to manage the transfer and storage in their own computer systems. In addition to images from the Sentinel satellites, each DIAS is expected to offer other extra data (images from other Earth Observation missions, in situ data or processed data, among others). The existence of several DIAS is presented as a way to offer diversity to users and to encourage competition between them with the confidence that this will result in a better offer. [6]

The available DIASes are the following[7]:

- sobloo
- Mundi Web Services

[5] Open Access Hub

[6] https://www.copernicus.eu/en/upcoming-copernicus-data-and-information-access-services-dias

[7] https://www.copernicus.eu/en/access-data/dias

- CREODIAS
- Onda
- WEkEO

CREODIAS[8] is one of the 5 DIASes. Data repository on CREODIAS is stored locally in the cloud infrastructure, which makes it easy to access and process.

CREODIAS is a seamless environment that brings processing to Earth Observation data (EODATA - EO DATA Free Archive). The CREODIAS platform contains online most of Copernicus Sentinel satellites data and Services, Envisat and ESA/Landsat data and other EODATA[9]. Its design allows Third Party Users to prototype and build their own value-added services and products. Set of pertinent tools guarantees simplicity, scalability and repeatability of any services' value chain.

Data processing is a fundamental phase in the value chain offered with CREODIAS. Pursuing most efficient and favorable solutions, the platform offers a full set of virtual resources. Virtual Machines with a choice of different operating systems available, easily mounted storage volumes with object storage solutions, virtual network and appliances like firewalls and VPN concentrators. A common authorization and authentication solution can be used to provide a single-sign-on capability to all CREODIAS services.

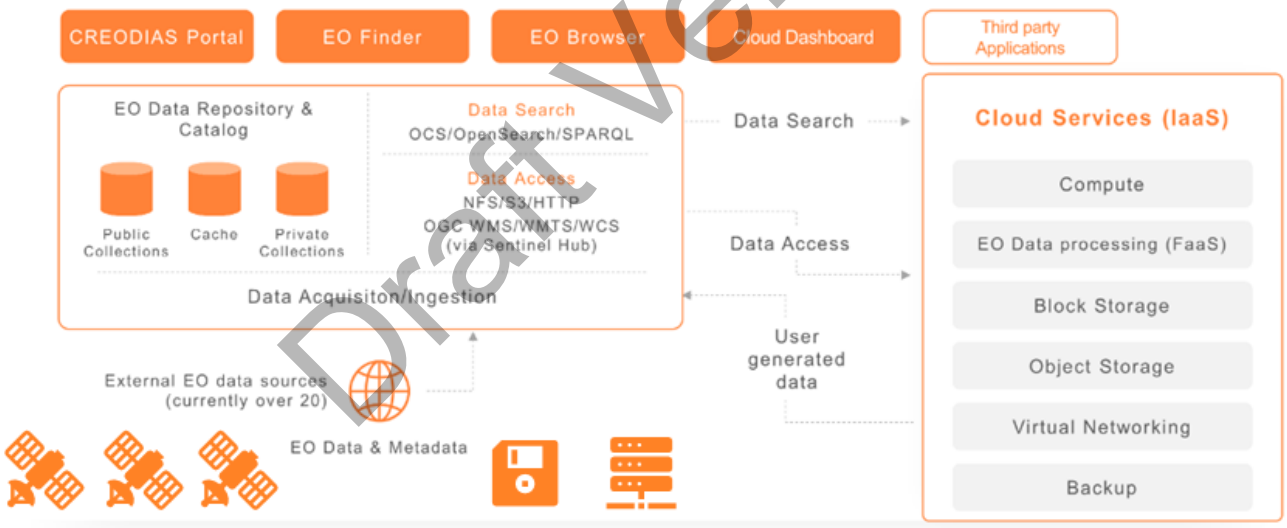


Figure 3.1: CREODIAS architecture

3.1 EO Data available in CREODIAS

A wide range of satellite data available on the CREODIAS platform can be successfully applied in many areas. Its utility depends only on the user's knowledge and creativity. Nowadays, more and more

[8] creodias.eu

[9] https://creodias.eu/data-offer

useful applications are created for analyzing agriculture, marine, weather, geology, smart-cities, industry, architecture, mining, epidemiology and many others.

The CREODIAS platform offers access to a wide range of satellite data, including the Sentinel series funded by the European Union under Earth Observation Program – Copernicus. Table below shows major characteristics. Applications depend mainly on spectral, spatial and time resolution of the sensors deployed on the satellite.

Below you can find tables describing up-to-date data offer[10] on CREODIAS.

Datasets	Products	Instrument	Locally Held
Sentinel-1A & Sentinel-1B	GRD	SAR C-BAND	Full archive
	OCN		
	RAW		Last 6 months
	SLC		- Europe: full archive - Last 6 months / orderable
Sentinel-2A & Sentinel-2B	L1C	MSI	Full archive
	L2A		- Orderable */** - Cached ***
Sentinel-3A & Sentinel-3B	L1 SLSTR	SLSTR	Full archive
	L1 OLCI	OLCI	
	L1 SRAL	SRAL	
	L2 SLSTR (LST/WST)	SLSTR	

[10] https://creodias.eu/data-offer

	L2 OLCI	OLCI	
	L2 SRAL	SRAL	
Sentinel-5P	L1B	TROPOMI	Full archive
	L2 ****		
Landsat-5	L1G, L1T, L1GT	TM	Coverage of Europe (1984-2011)
Landsat-7	L1G, L1T, L1GT	ET	Coverage of Europe (1999-2017)
Landsat-8	L1T, L1GT	OLI, OLI TIRS	Coverage of Europe
Envisat	L1	MERIS	Global (2002-2012)
SMOS	L1B, L1C, L2	MIRAS	Global (2010-present)
S2GL	-	-	Coverage of Europe (2017)

Table 3.1: Free data offer with details regarding dataset, offer and its availability online. You can find more description in the article: https://creodias.eu/utility-of-creodias-data.

* Can be ordered from ESA via CREODIAS Finder.

** If L2A does not exist in the ESA archive, it can be processed by CloudFerro with Sen2Cor.

*** Kept in rolling cache of 1 PB (mostly European coverage). Oldest data will be removed when the cache is full.

**** Sentinel-5P L2 products currently stored: Aerosol, Cloud, CO, HCHO, NO2, NP, O3, SO2.

Datasets	Products	Instrument	Estimated locally held	Data access
DEM	Mapzen	SRTM, others	Aggregation of several DEM sources	Accessible over S3 or NFS protocol

	SRTM	SRTM	Global (56S, 60N, 180W, 180E) February 2000	Accessible over S3 or NFS protocol
Jason-3	Altimeter	Altimeter	Since mission began	Accessible over S3 or NFS protocol

Table 3.2: Additional datasets available for free in CREODIAS

Datasets	Products	Data access
CAMS (Atmosphere)	All collections	Accessible over S3 or NFS protocol
CEMS (Emergency)	All collections	Accessible over S3 or NFS protocol
CLMS (Land)	All collections	Accessible over S3 or NFS protocol
CMEMS (Marine)	All collections	Accessible over S3 or NFS protocol

Table 3.3: Copernicus services data offer available of CREODIAS for free

3.2 EO Search API and SPARQL endpoint of CREODIAS

CREODIAS data can be queried using either the EO Search API or SPARQL.

The EO Search API is backed by the EO Data Finder which conforms to OpenSearch[11] standard. Data sets are organized in so-called collections, corresponding to various satellites. A query may search for data in all collections, or in one particular collection only. Queries can be executed as HTTP GET calls, and provide outputs both in JSON and XML formats. The API supports sorting, pagination, formal and natural language queries. The returned metadata is available for all products in the form provided by the original data publishers. The API is accessible free and anonymously (open for anonymous access for everyone, no authorization is used). The detailed description of all search capabilities and integration guide can be found in EO Data Finder API manual for CREODIAS[12].

The SPARQL interface is a W3C standard, Linked Data endpoint allowing RDF data to be retrieved and manipulated, based on the specialized, well-developed, semantic graph database Allegrograph; this

[11] opensearch.org

[12] https://creodias.eu/eo-data-finder-api-manual

interface is connected to the CREODIAS EO Browser, but can also be easily used by any third party SPARQL clients. Definition of the SPARQL endpoint can be found on the CREODIAS FAQ page[13].

The SPARQL endpoint of CREODIAS contains metadata of EO data. It contains metadata from Mission Sentinel-1, Sentinel-2 and Sentinel-3. The following table represents what product types it contains.

Mission	Platform	Product Type
Sentinel-1	S1A	GRD - Ground Range Detected SLC - Single Look Complex RAW OCN - Ocean
Sentinel-1	S1B	
Sentinel-2	S2A	L1C - Level-1C
Sentinel-2	S2B	
Sentinel-3	S3A	EFR - output during EO processing mode for Full Resolution ERR - output during EO processing mode for Reduced Resolution WFR - Water Full Resolution WRR - Water Reduced Resolution LAN - Land products LFR - Land Full Resolution LRR - Land Reduced Resolution LST - Land Surface Temperature RBT - Radiance and Brightness Temperature SRA - Synthetic Aperture Radar Altimeter WAP - water products WST - Water Single Temperature

Table 3.4: List of Product Type per Mission and Platform

The metadata about the collected data from different satellites are converted to RDF data using the ontology we discuss below. The overview of the ontology can be seen in figure 3.2.

[13] https://creodias.eu/faq

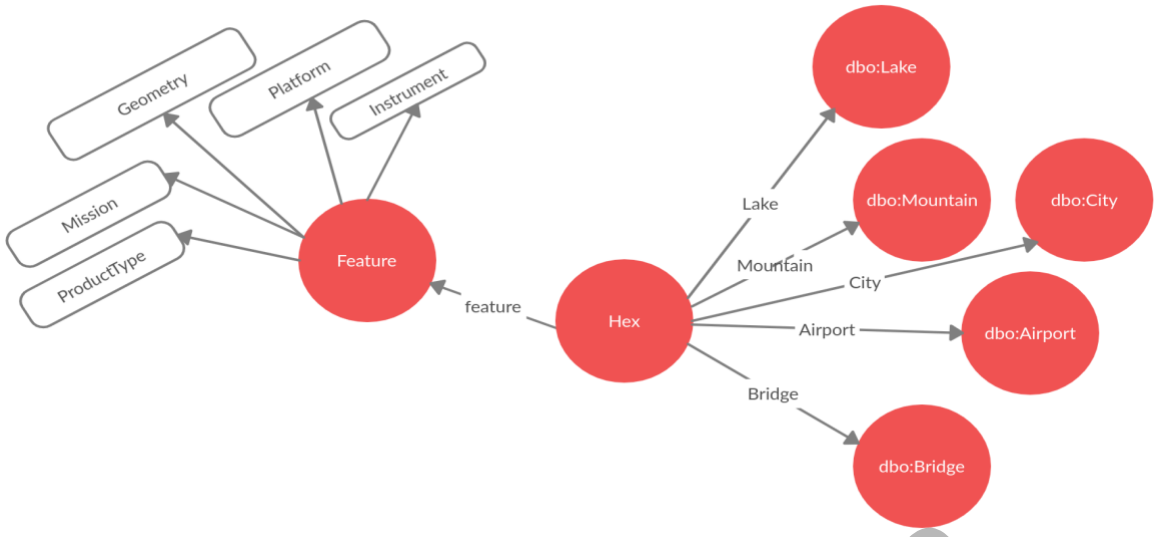


Figure 3.2: Ontology Overview of the CREODIAS SPARQL endpoint

It contains a feature class which represents products from satellites. All the metadata about the product is presented as predicates of the feature class, and can be seen in figure 3.3.

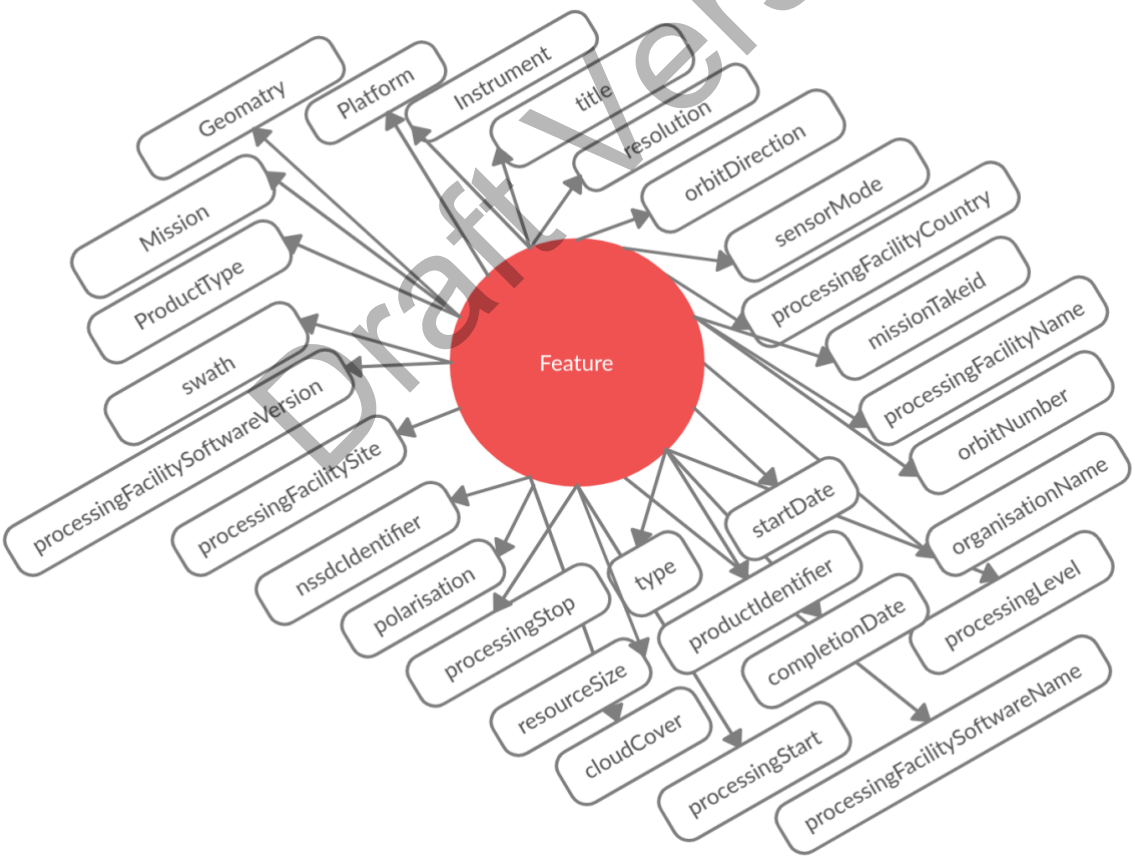


Figure 3.3: Feature Class Predicates

One of the important reasons for using Linked Open Data (LOD) in CREODIAS is the possibility of the creation of interlinks from external data sets to the CREODIAS EO data resources using the RDF graph data model. Thus, as can be seen in the figure 3.2, the feature class is linked to the entities from DBpedia using the Hex class. The Hex class is created using Uber's Hexagonal Hierarchical Spatial Index. Every feature class contains the predicate Geometry, which represents the bounding box of the area for which it contains an image. Thus, using the spatial index, instances of the feature class are connected to the instance of the Hex class. As well, instances from DBpedia contain coordinates in the form of latitudes and longitudes. Thus, all the instances from DBpedia that are present in CREODIAS SPARQL endpoint are also connected to the instance of the Hex class. The SPARQL endpoint can be found at https://sparql.creodias.eu:20035/#/repositories/creodias/.

As an example SPARQL query, if one would like to find the title of all the features that are of Sentinel-2 and covers Airports of Greece, the SPARQL query for that would be written as follows:

Natural Language Text : Find Sentinel-2 images containing airports of Greece.

SPARQL Query :

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX schema: <http://schema.org/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT distinct ?hex ?title {
{
    SERVICE <https://dbpedia.org/sparql>
    {
         SELECT ?airport
         {
          ?airport rdf:type schema:Airport .
          ?airport dbo:location  dbr:Greece .
         }
    }
}

?hex <http://ws.creodias.eu/metadata/object/airport> ?airport .
?hex <http://ws.eodias.eu/metadata/attribute#feature> ?feature .
?feature <http://ws.eodias.eu/metadata/attribute#title> ?title .
?feature <http://ws.eodias.eu/metadata/attribute#mission>
<http://ws.eodias.eu/metadata/mission/Sentinel-2> .
} LIMIT 10
```


3.3 Short description billing plan on CREODIAS

All of CREODIAS services can be purchased in several main billing modes. There are three main billing modes we provide to our Tenants:

- Per usage **(per hour)** prepaid mode where services are billed according to every hour (or even minute) used. A user can purchase a credit to be used to provision and keep the system's resources. The credit is measured in Billing Units which are valuated to 1 Euro each for the sake of the Price List. Every 10 minutes the Tenants credit is decreased with the cost of actually used resources. This is a very flexible mode that allows a user to create and remove resources at will and pay only for the actually used resources. This is a mode useful for experimental and development work or for environments with very variable resources.
- Fixed term **(long term contract)** mode where services are bought for longer periods. In this mode a user purchases well defined Platform's resources for well-defined periods of time. The long term resources can be paid directly or with the use of credit in Billing Units. One cannot then change these resources but on the other hand obtains much cheaper offering. This mode is preferable for long term usage of well-defined environments with well understood needs.
- Revenue sharing mode where cost of the service depends on the resource sold by the Third Party to its customer. This mode is used in the data marketplace. In this mode the Platform charges a Third Party with a price equal to a fixed percentage (normally 10%) of Third Party revenues acquired from the End-user.

More information is available at: https://creodias.eu/billing-modes. Moreover, the price list with all the services is available at: https://creodias.eu/price-list.

4 WEkEO

WEkEO DIAS has been designed by EUMETSAT, Mercator-Ocean International, and ECMWF, and implemented by an industry consortium where the main contributors are Thales Alenia Space and CloudFerro for the infrastructure, GMV for the data processing tools, and Telespazio for the webportal, to provide easy access to Copernicus information and processing tools, in one centralized location, so that users can develop applications for their own specific needs.[14]

This platform offers services on a free basis as well through a suite of paid advanced plans to fit the scalable needs of individuals, businesses and institutional operations. The Essential pack offers free and open access to: all Copernicus and Sentinel data, as well as Jupyter Notebooks and user support. The Advanced pack includes a competitive range of pricing plans, including virtual cloud-based processing environments and tools as well as free networking (in and out).

The WEkEO V1 cloud platform provides a comprehensive portfolio of original Copernicus Programme and Sentinel satellite data (and supporting missions), with harmonized data access, cloud infrastructure, and expert user support. The main idea is to benefit a wide range of users, including institutional bodies, the private sector, scientists, and civil society to develop their own value-added products, applications and services.

[14] https://www.mercator-ocean.fr/wekeo-dias

Among others, the services included in WEkEO DIAS are big data analysis tools, to develop applications with the required needs.

In figure 4.1, we include the available data from the Sentinel satellites:

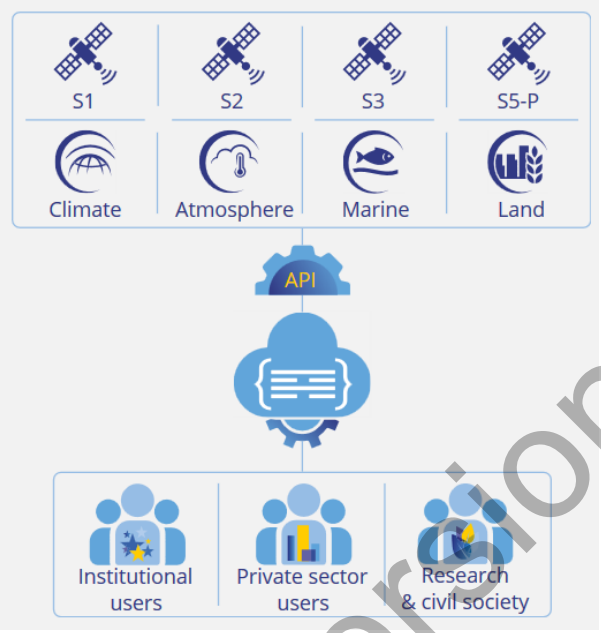


Figure 4.1: WEkEO available Sentinel data

In terms of available data, WEkEO offers direct, up-to-date and unreplicated access to the Copernicus and Sentinel data, by ensuring the connection speed required for such amount of data, and the ability to have the latest data available for the users. The catalogue available includes the Sentinel data and its services: Land, Climate, Marine, and Atmosphere. Extensive satellite data is available ranging from L1, plus processing levels such as Sentinel 1, 2, 3, as well as Copernicus contributing missions. WEkEO offers the possibility of data aggregation among their catalogues as can be seen in figure 4.2:

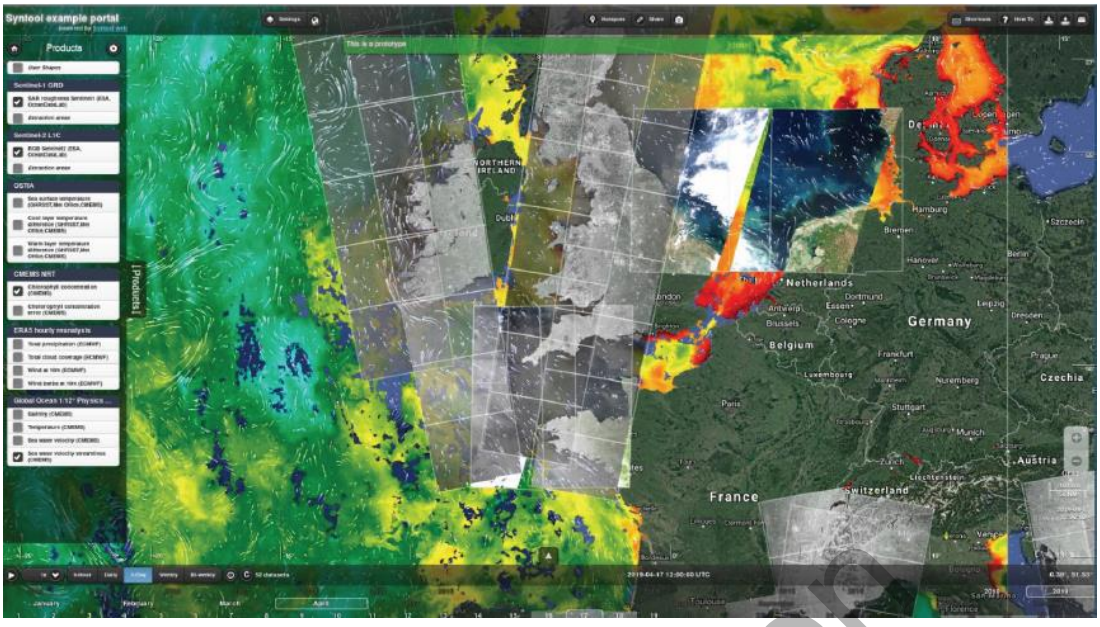


Figure 4.2: WEkEO data aggregation[15]

WEkEO supports environmental scientists in their research and publication by providing different services. As an example, the following services, among others, could be developed using WEkEO:

- SEASONAL FORECASTING: While it is generally not possible to predict the day-to-day changes in detail beyond about a week ahead, it is possible to say something about likely conditions averaged over the next few months. Seasonal forecasts provide information about these long-term averages. WEkEO provides the best available guidance on likely seasonal conditions in many parts of the world, including Europe, available to the scientific community.
- CLIMATE CHANGE: WEkEO provides climate change data to feed the scientific community, meet the requirements of a number of government and business customers, and underpin mitigation and adaptation policy formation and decision making. WEkEO provides climate datasets based on observations, and Earth System Models to sample the past environment since 1950.

In terms of infrastructure, WEkEO is built on a distributed way by a high-speed backbone, and offers cloud-based hosted processing being able to include different processing tools, including big-data tools. Those technologies allow users to compute and transform data for their own purposes, being able to create their own applications. By using software as a service (SaaS) as the main idea, some of the available services are Jupyter Notebooks, ready-to-use virtual machines pre-configured for data access, and standard tools such as SNAP, QGS and development tools for Python or R.

The main connector for the platform is the Harmonized Data Access (HDA) API, which allows uniform access to the whole WEkEO catalogue, including subsetting and downloading functionalities. The HDA protocol can be used through the API available on the WEkEO portal (web interface) or directly from a virtual machine or a Jupyther notebook by the usage of python scripts. For data sharing, the

[15] https://www.blue-cloud.org/e-infrastructures/wekeo-dias

WEkEO Drive tool can be used too, which will be available to any user once a user is connected to the platform. WEkEO's REST-based single protocol facilitates users to scale and evolve their code. It allows them to easily integrate all data sources using a virtual station, a Jupyter Notebook or an application of choice, and uses homogeneous subsetting attributes.

4.1 EO Data of WEkEO

WEkEO offers free access to Copernicus Data, including all data from Sentinel satellites, contributing missions and the Copernicus marine, land, atmosphere and climate services. More information is provided in table 4.1 below.

Dataset	Products	Sensor or Services	Area
Sentinel-1	Level 1	Synthetic Aperture Radar C-SAR Ground Range Detected (GRD) Single Look Complex (SLC)	Global
Sentinel-2	Level-1C/Level-2C	MSI	Global
Sentinel-3	Level 1B Level 2	SRAL Sea and Land Surface Temperature Radiometer (SLSTR) SRAL/MWR OLCI (Ocean and Land Colour Instrument) Visible (VIS), Near Infra-Red (NIR) and Short Wave Infra-Red (SWIR) bands (on the A and B stripe grids) Sea Surface Temperature (SST) Short Time Critical (STC)	Global
Sentinel-5P		C-SAR	Global
Copernicus	C3S (Climate)	Copernicus Services (Past or Future)	Global Europe Mediterranean Sea Black Sea
	CAMS (Atmosphere)	Copernicus Services (Past)	Global
	CLMS (Land)	Copernicus Services (Past)	Global
	CMEMS (Marine)	Copernicus Services (Past or Future)	Global Europe North Atlantic Artic Mediterranean Sea Black Sea

			Baltic Sea Atlantic-Iberian Biscay Irish Ocean European North-West Shelf Ocean

Table 4.1: WEkEO data access

4.2 Elasticity service of WEkEO

Cloud Elasticity refers to the capability of a cloud infrastructure to grow or shrink capacity for any of the provided resources such as CPU, RAM, and storage to adapt and satisfy the changing user demands. Similarly in the case of WEkEO, where the infrastructure in place (Distributed Partner Infrastructure - DPI) from CMWF, EUMETSAT, and Mercator-Ocean have limited resources, the elasticity service provides computational elasticity in support to applications and to computational needs that otherwise cannot be accommodated within the on-premises infrastructure of the WEkEO infrastructure (DPIs). Elasticity and on-demand scaling is provided by external providers or Elasticity cloud providers and in the case of WEkEO is CloudFerro. Elasticity service includes the required accounting and billing functionality such as a pay per use model.

WEkEO users that require additional computing resources or storage have the option of allocating their projects to the elasticity cluster provided on a commercial public cloud offering. This is a commercial subscription run by Elasticity cloud provider with the same Cloud orchestrator used in WEkEO i.e. Morpheus, which orchestrates and provides access to services, networking, and VM images offered on the ECMWF, EUMETSAT, and Mercator-Ocean's DPIs.

5 The AI4Copernicus Architecture related to CREODIAS

In this section we discuss the architecture of AI4Copernicus related to CREODIAS and its integration with the AIoD platform. This part of the AI4Copernicus architecture is shown in figure 5.1.

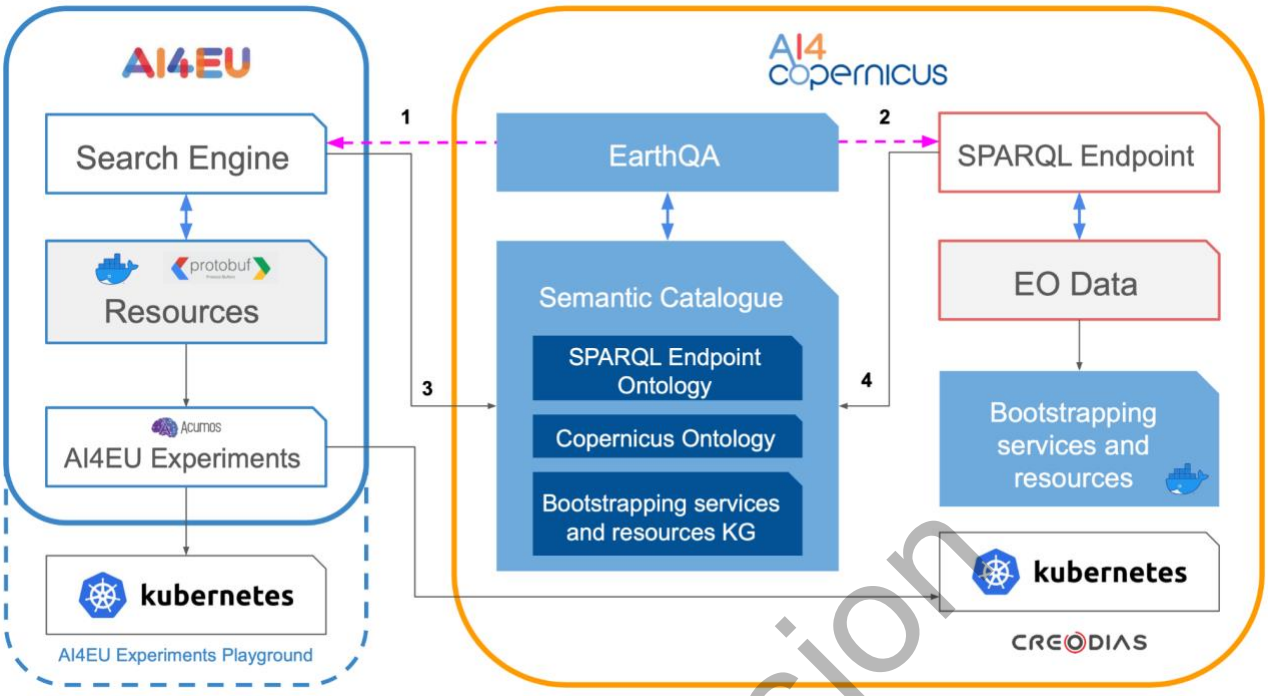


Figure 5.1: The AI4Copernicus architecture

On the left side of figure 5.1, we can see the AIoD platform. The platform has a search engine, which can be used to find resources that are available in the AIoD platform. For example, a resource can be a dataset, a model or a software tool (e.g., the implementation of a machine learning (ML) algorithm). Once the desired resources are found using the AIoD platform search engine, a user can use Acumos to build a pipeline using these resources. In a future version of the AIoD platform, the AI4EU Experiments Playground will also be provided, where users will be able to deploy Kubernetes and run their pipelines in order to develop their applications. The AI Playground will be based on standard servers without GPU acceleration to allow users the run of small pipelines that can be deployed by one click from AI4EU Experiments. The playground is suitable to try out models or make proof of concepts, but it is not suitable for any confidential data. It leverages the EU-login SSO functionality to provide seamless interaction with AI4EU Experiments and other subsystems. The description of the above AIoD platform components has already been provided in Section 2.

On the right side of figure 5.1, we can see the software components that are developed or extended in AI4Copernicus. Firstly, we have CREODIAS, as this is the first DIAS we will target, managed by partner CloudFerro. As described in Section 3 above, CREODIAS hosts EO data and provides a search API with a SPARQL endpoint that allows users to run SPARQL queries, to discover the available EO data. In AI4Copernicus, we will provide bootstrapping services and resources that will be developed in WP5, or the projects selected from the Open Calls. These services and resources will be deployed on CREODIAS, because this is the most efficient way to take advantage of the EO data, without the need to download it locally or move it to another platform.

For discovering resources in CREODIAS, a user can pose a natural language question to EarthQA, the Question Answering (QA) engine which is described in more detail in Section 7. EarthQA is targeting the semantic catalogue, which contains all the information we need for the AI4EU resources (arrow 3) and the EO data of CREODIAS (arrow 4), as explained in Section 6. The search engine of the AIoD platform will be able to send requests to EarthQA (arrow 1) as explained in Section 7 below. EarthQA will send requests to the SPARQL endpoint of CREODIAS to get information about the available EO data (arrow 2). The semantic catalogue contains the Copernicus ontology, the SPARQL endpoint ontology from CREODIAS, and the metadata of the bootstrapping services and resources of WP5. EarthQA and semantic catalogue will be deployed on CREODIAS.

Interested parties will be able to use the above AI4Copernicus architecture for (i) small-scale experimentation and (ii) demanding application developments that require big EO data and high computational power. In the first case, the users will be able to create their desired Acumos pipelines using the AI4EU Experiments playground component. There, users will have access to the AI4EU resources and other EO data. This EO data can be either some small-scale datasets, accessed externally from CREODIAS via commercial S3 block storage interface, or users can upload their own data. This implies that users can experiment while creating their Acumos pipelines. On the other hand, if users' goal is to create an application using big EO data, it can be achieved by the option to use AI4EU Experiments Playground in order to create a deployable Kubernetes docker, which user can deploy on CREODIAS (arrow 5 Figure 5.2 below) and use internal S3 API to access the data close to the source. In this way, the users will be able to take advantage of the big EO data provided by CREODIAS, without the need to download it locally.

A number of steps needs to be taken in order to access the experiments ecosystem. These include user registration, preparation of execution environment and composition of the desired pipeline. Such a user journey is depicted below and followed by an explanation of each step. Please note that the arrows are numbered showing order of undertaking required actions, and explained below:

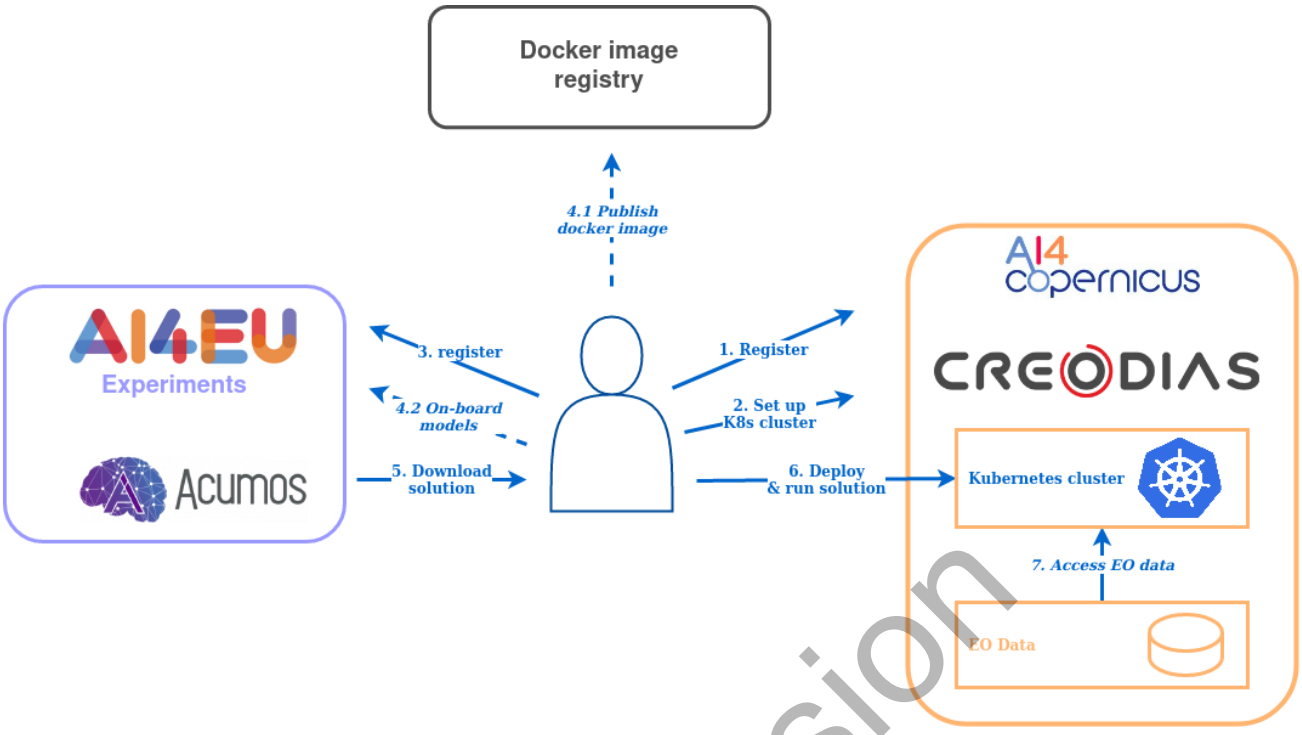


Figure 5.2: The user's journey to access the experiments ecosystem

1. A user needs to register in CREODIAS.
2. The next step for the user is to set up a Kubernetes cluster in order to prepare an execution environment for experiments.
3. Additionally, a user needs to register in the AI4EU experiments platform (Acumos) in order to access solutions and collaborate with other scientists.
4. One can compose a solution either from existing components, or create and share new models using Acumos onboarding and publication processes. It is also possible to use existing solutions published by members of the AI4EU community (this step is optional). In case of new components, docker images need to be stored in a preferred docker image repository. The address of the repository and the image identifiers are pointed during the component onboarding process.
5. Once the solution is chosen, it can be downloaded from Acumos. The solution file contains a complete set of scripts allowing AI4EU users to run them in their chosen execution environment.
6. Solution scripts deploy the pipeline to the Kubernetes cluster created in step 2.
7. Solutions run in the CREODIAS Kubernetes cluster and may access EO data. It is however also possible to execute other types of solutions if needed.

6 The Semantic Catalogue

In this section the components of the semantic catalogue are discussed. The purpose of the semantic catalogue is to enable the semantic search of the Earth Observation knowledge. Hence, as described in Section 5, EarthQA targets the semantic catalogue, which contains all the information we need for the AI4EU resources and the EO data of CREODIAS. In particular, it contains the Copernicus ontology (CO), and the metadata of the bootstrapping services and resources.

6.1 The Copernicus Ontology

CO is designed to allow describing and querying content based on semantically important aspects of the Copernicus data, therefore lowering the entry barrier and increasing the usefulness and the usability of the resources offered. Specifically, the scope of the Copernicus ontology is to capture general knowledge about Satellite Remote Sensing and its applications, to capture knowledge about EO datasets as well as about finer-detail geospatial and temporal aspects of the data. CO is an OWL ontology that contains 465 classes and nearly 1600 axioms (some of them imported from external ontologies). It is openly available[16].

The main part domains included in CO are listed below.

D1. General knowledge about Satellite Remote Sensing and its applications
D2. Knowledge about EO programmes like Copernicus and specific satellites, like the Sentinels
D3. Knowledge about EO datasets
D4. Geospatial and temporal knowledge
D5. Knowledge about publications on the domain

Next, the methodology followed for the development of CO is described in brief.

Copernicus Ontology Development Process

The last 30 years a plethora of methodologies for ontology development has been proposed in the literature (e.g., [FGJ97], [P16]). For the development of CO we will follow the methodology "*Simplified Agile Methodology for Ontology Development*" (SAMOD), developed by Peroni (2016), which is graphically depicted in Figure 6.1. In the first step, with the help of domain experts, the domain is divided into independent subdomains (modelets or microtheories) and the ontology development process is initialized. In the CO case, the different domains are defined by the subject areas D1-D5.

To specify the scope of each domain, we distributed a questionnaire (Appendix I) to domain experts requesting for questions that they would like the semantic search engine to be able to answer. The collected set of competency questions is presented in Appendix II. At the same time, for each domain, we conducted a series of interviews with the domain experts in conjunction with research publications, standards, regulations and any other source of information (e.g. ESA and CREODIAS

[16] http://pyravlos-vm5.di.uoa.gr/CopernicusOntology_BootstrappingKG.zip

websites) from which valuable knowledge was extracted. In this way, we specified the exact knowledge to represent in the ontology.

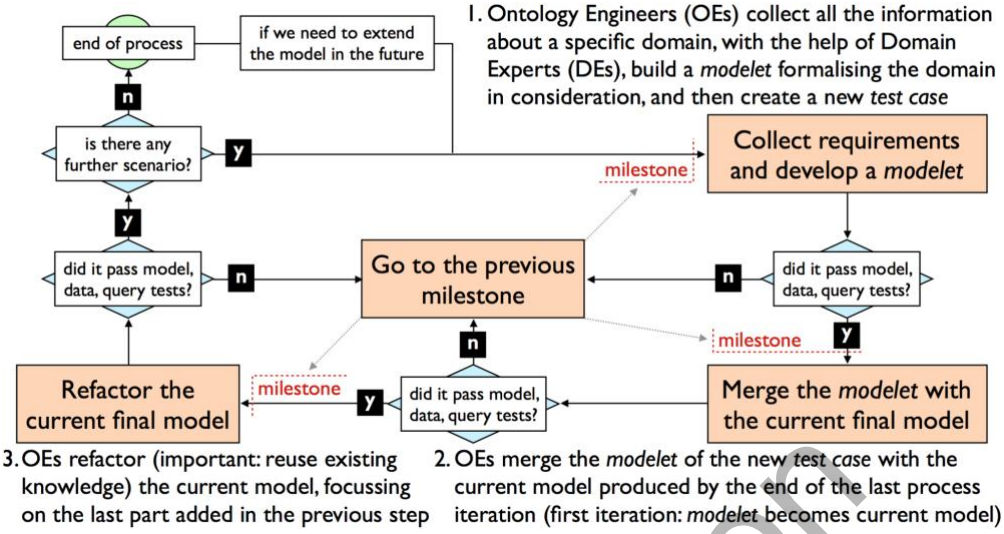


Figure 6.1: A brief summary of SAMOD, starting with the "Collect requirements and develop a modelet" step [P16]

According to the recommended practice and the linked data principles [BEM14], we, also, identified a set of external well-defined and widely reused ontologies to reuse. Also, as the purpose of AI4Copernicus is to build on top of ongoing AI4EU integration initiatives, the CO was mapped to the AI4EU ontology[17]. After this, for each domain, we formalized the remaining part of the knowledge and we checked that the newly extended ontology was consistent.

Next, we describe the part domains included in CO and their formal representation.

D1. General knowledge about Satellite Remote Sensing and its applications

The purpose of this domain is to enable the user to retrieve knowledge about the main elements of remote sensing and its applications. For instance, the following knowledge should be implied from the ontology:

- Radar remote sensing is a kind of satellite remote sensing
- Flood emergencies typically use radar images

More specifically, the following subjects are described for D1:

- the main types and components (e.g., EO process, EO equipment) of satellite remote sensing
- the main types and properties (e.g., different resolution types) of the sensors used for remote sensing organized into broader and narrower refinements.
- the properties and the types of the platforms (satellites) that carry them
- the different types of orbits
- the EO products of the observations and their primary applications

[17] https://github.com/ai4eu/ai-resources-ontology

- a classification of these applications

Related External Resources

The basic knowledge about remote sensing is based on discussions with domain experts, on the classification provided by NASA[18], and in the book of Lwin [L08] in which knowledge for the Earth Observation and Geoinformation sector is represented .

As it is illustrated in Figure 6.2, all core classes of CO, are mapped to the core ontologies SOSA, GeoSPARQL and AI4EU. *SOSA*[19] (Sensor, Observation, Sample, and Actuator), endorsed by W3C, describes sensors, their observations and their procedures. It is a well-known, widely reused ontology in a range of applications, and more recently, in the representation of satellite metadata and earth observations[20]. GeoSPARQL ontology is a standardized ontology defining a vocabulary for representing geospatial data in RDF. AI4EU ontology aims to establish the conceptual foundation for AI in the context of the AI-on-demand platform.

Finally, for a shared understanding of the domain and for a better organization of its classes, CO is also mapped to the Basic Formal Ontology (BFO). BFO is a small, upper level ontology for supporting information retrieval, analysis and integration in scientific and other domains. It has been reused in more than 250 ontologies developed for various scientific domains.

The EO Ontology[21] [T20], developed for the Candela project, represents, also, knowledge related to EO. Although our approach is similar to the one for EO ontology (it is also mapped to the ontologies SOSA/SSN and GeoSPARQL), the major difference with the EO ontology, is that CO is significantly larger, covering the domain in greater detail. There are also few differences in the modeling decisions (e.g., the footprint in CO is a geosparql:Geometry and not a geosparql:Feature).

Also, as in SOSA, we reuse the *OWL-Time* ontology to date observations.

The part of the CO related to D1

The upper-level view of the Copernicus ontology is shown in Figure 6.2, below, indicating the basic entities users will be enabled to search over. In Figure 6.2 (and in all ontology figures following), we present the object properties with dashed arrows (with labels of the form: hasY, if the arrow points to a class Y, e.g., hasEOPlatform), and with solid arrows the subclass (is-a) relationships.

Major concepts related to Satellite Remote Sensing are the EO equipment. Both EO instruments and EO platforms are regarded as EO equipment (relation SubClassOf). Any EO equipment is a material entity, hence it is characterized by a mass value and dimension values (these properties are inherited by the class *co:MaterialEntity*).

[18] https://earthdata.nasa.gov/learn/remote-sensors

[19] https://www.w3.org/TR/vocab-ssn/

[20] https://candela-h2020.eu/

[21] http://melodi.irit.fr/candela/#list-meta

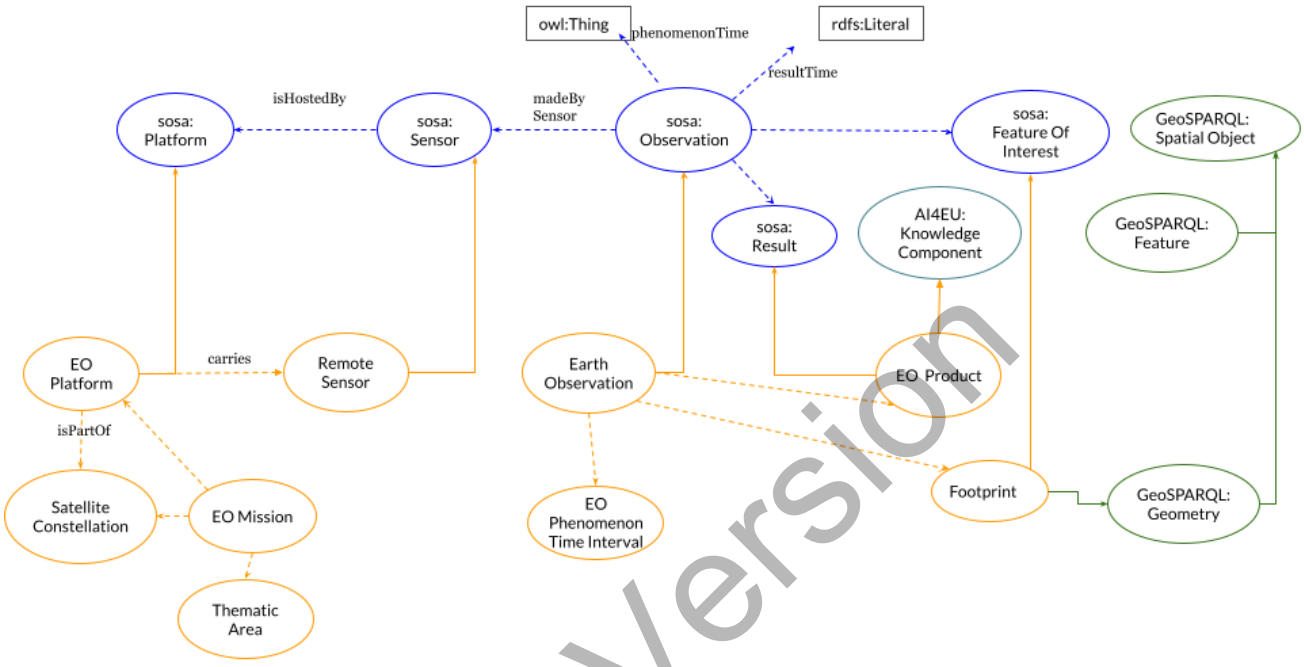


Figure 6.2: The properties related to *co:Thematic Area*

The properties related to *co:EOPlatform* (SubclassOf *sosa:Platform*) are depicted in Figure 6.3. Some of the properties (id, shortName, serial identifier, orbit) are based on the properties of the Platform object described in OGC 17-003r2 [C20], while other properties are related to the orbits that the platforms are designed for (e.g., designedForOrbit, orbitMotion, orbitalPeriod). We found the latter properties from the ESA website and we added them after discussions with domain experts. An EO platform may be part of a *co:SatelliteConstellation*, and carries some EO instruments.

The types of (subclasses) of EO instruments presented in the ontology include the classes *co:Antenna, co:GPSR, co:HousekeepingTelemetry*, and *co:RemoteSensor*. The antennas are characterized by an operational mode. The *co:RemoteSensor* class is divided into the classes *co:PassiveSensor* and *co:ActiveSensor* which are further extended by a set of specialized classes. The hierarchy of remote sensors included in the ontology, as presented in the Protégé[22] ontology development tool, is depicted in Figure 6.4. Remote sensors are characterized by their OGC 17-003r2 type (altimetric, atmospheric, etc), their *co:operationalMode* and the *co:wavelengthSpectrum* that they can operate. The imaging sensors are also characterized by radiometric, spectral and spatial resolutions.

[22] https://protege.stanford.edu

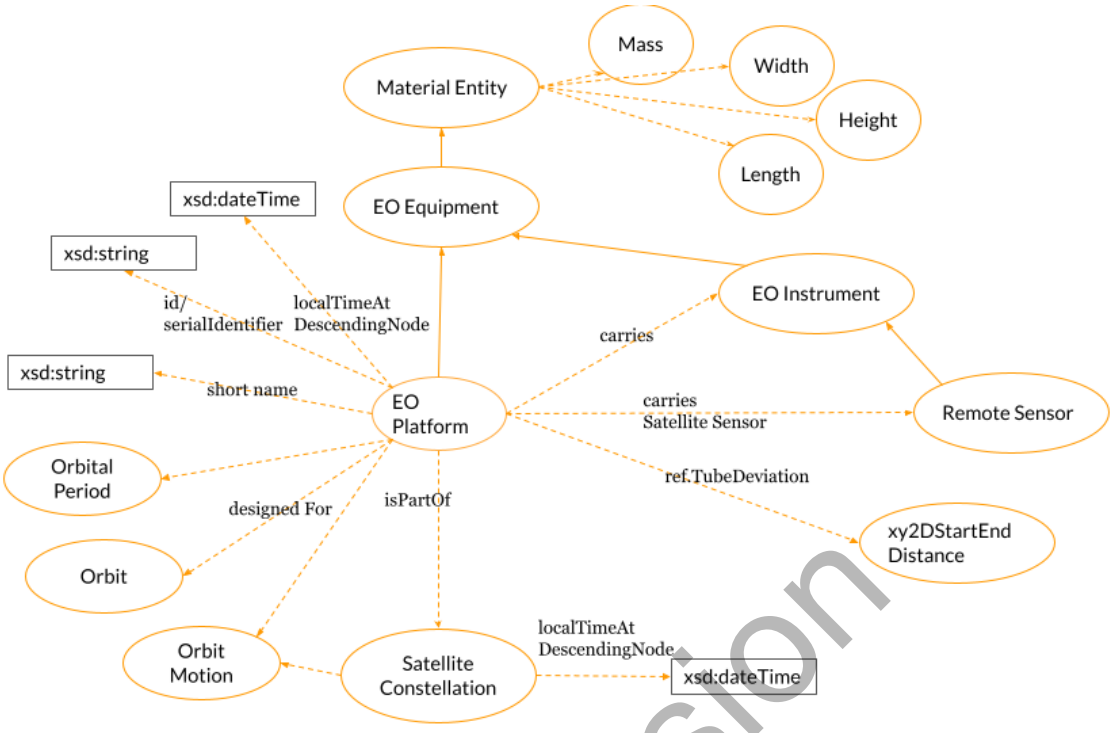


Figure 6.3: Excerpt of CO related to the class co:EOPlatform

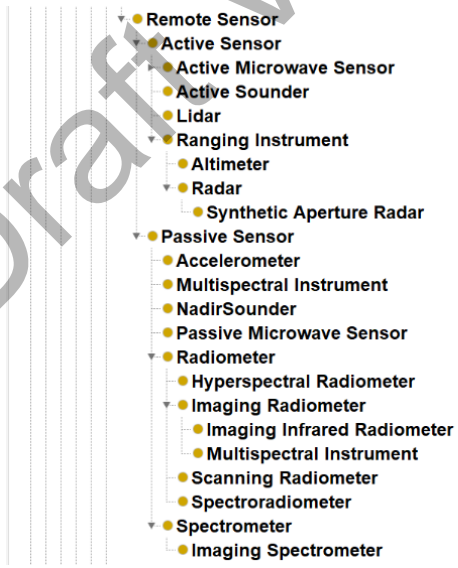


Figure 6.4: Remote Sensor hierarchy

As the orbit of the platform is also crucial for the type of EO products and their thematic areas, we have included in the ontology the several types of orbits along with their properties (Figure 6.5). For instance, *co:GeostationaryOrbit*, *co:LowEartOrbit* are subclasses of the class *co:Orbit*. While, any

orbit, in general, is characterized by the plane degrees, approximate plane degrees, eccentricity and altitude.

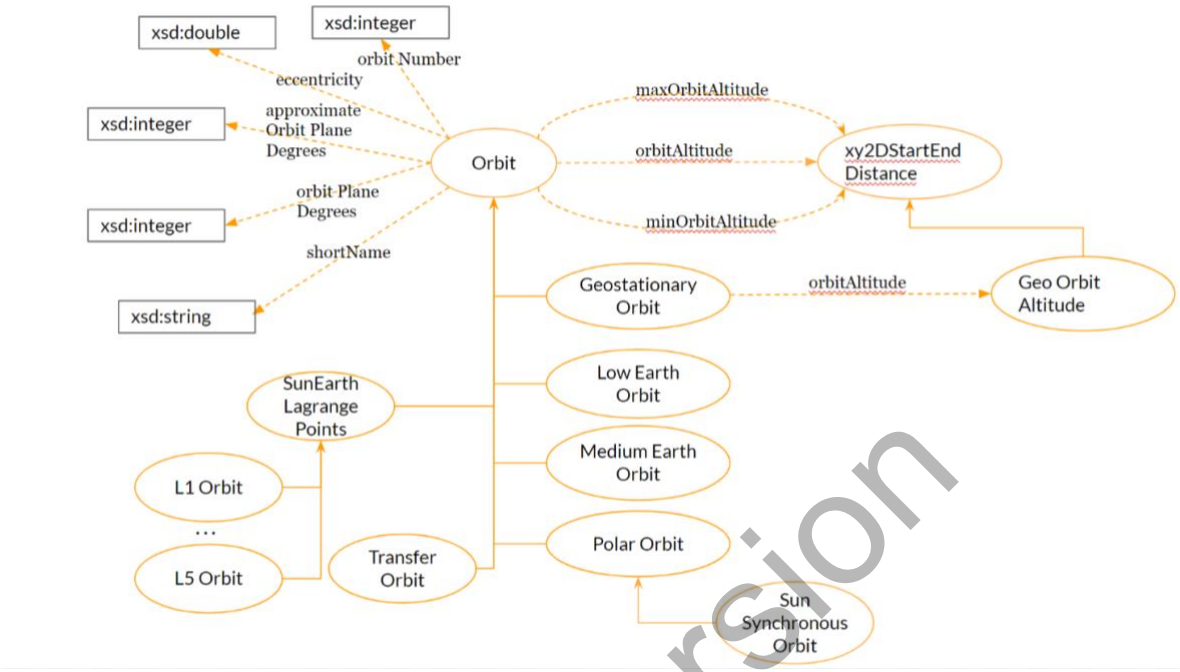


Figure 6.5: Excerpt of CO related to co:Orbit

The *thematic areas* are categorized according to ESA's technical documents, e.g., [VBJ+20]. The subclasses of the *co:ThematicArea* are the classes *co:EmergencyChange*, *co:Land, co:Maritime, co:Environment, co:Security* and *co:Topography*, which are further extended with more specialized subclasses, e.g., the class *co:Atmosphere* is subclass of co:Environment or they are instantiated, e.g., the class *co: EmergencyChange* has instances: *co:earthquakeAnalysis, co:floodMonitoring, co:forestFire, co:landslide, co:subsidence* and *co:landslideAndVolcanco*. The thematic areas are also linked to the missions (*co:Mission*), and to *co:EOProduct* and *co:Services*. This way, the user can retrieve the services offered based on the thematic area that is interested in.

Knowledge about programmes and missions is based on the information available on the websites of the European Space Agency, CREODIAS and on discussions with domain experts. As it is demonstrated in Figure 6.1, the EO mission initiates the earth observation events. The EO mission is characterized by the EO Programme that participates in, the EO platform(s) that it may involve, the thematic areas that its outputs can be used for, its temporal coverage (for how many years the data collection is taking place), its orbital period, temporal resolution, etc. The full set of properties is presented in Figure 6.6.

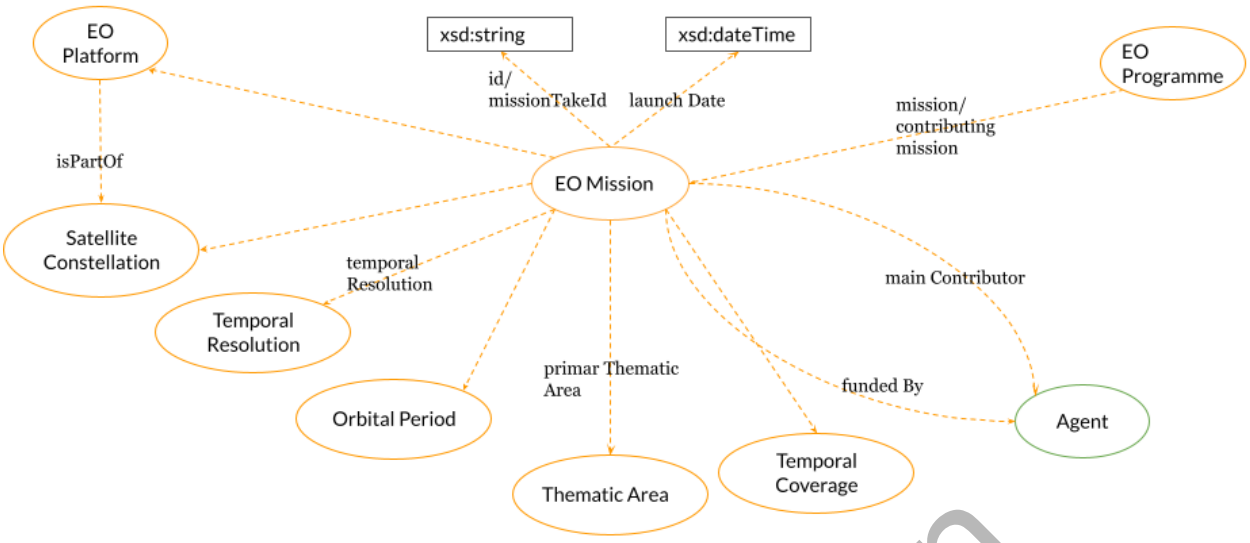


Figure 6.6: The properties related to EO Mission

D2. Knowledge about EO programmes like Copernicus and specific satellites, like the Sentinels

The representation of D2 will enable the end user to make general or specific questions about the several earth observation programmes, the earth observation missions and the satellites used for these programmes. For this domain, both generic knowledge and a set of assertions will form the final ontology. For instance, one will be able to infer automatically from the KG that Sentinel 1A is a radar satellite.

Hence, the following elements should be formally described for this domain:

- The existing programmes, their properties (e.g. objective, coordinating and managing agents) and their correlations to the missions
- Classification of on-going and past missions (e.g., Sentinel-mission is-a Copernicus mission), Population of the ontology developed for D1 with specific individuals and their properties (e.g., Sentinel1A is a Sentinel-1 platform)
- Properties of the satellite sensors for specific missions (e.g., operational modes, correlation to thematic areas)

Related External Resources To model the quantities of several measurements (e.g., the dry mass of the satellite) we have re-used the Ontology of units of Measure[23] (OM). The OM models concepts and relations important to scientific research. It has a strong focus on units, quantities, measurements, and dimensions.

The part of the ontology describing the missions, programmes, satellites etc, is presented in Figure 6.6. This part of the ontology is instantiated or further refined with subclasses. For instance, the class *co:EOMission* has subclass the class *co:CopernicusMission*, which is instantiated with the Sentinel missions (Sentinel-1, Sentinel-2, etc). A screenshot of the data related to Sentinel-1 mission is

[23] https://github.com/HajoRijgersberg/OM

presented in Figure 6.7. In particular, the related thematic areas, the satellites, and the temporal resolution of the mission are represented formally.

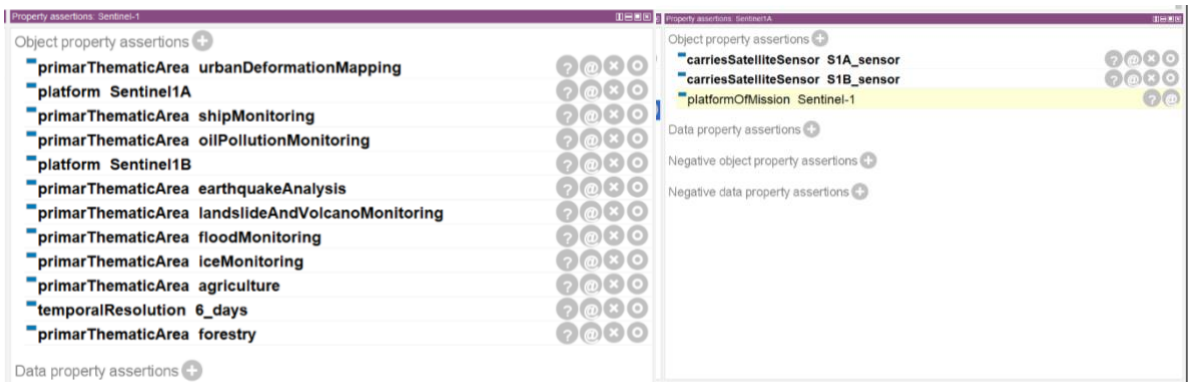


Figure 6.7: Screenshot showing the Sentinel-1 data stored in Protégé

Relevant information is also included in the ontology for the rest of the Copernicus missions.

D3. Representation of information related to Earth Observation and EO Product

The knowledge related to earth observation, and specifically, to the platforms, the sensors and their outputs is based on the OGC 17-003r2 standard. In Table 6.1, we present a part of the transformation of the standard to ontological form. As wrapping objects that usually exist in json-based metadata representation, such as AcquisitionInformation, ProductInformation, etc, included in the standard, do not make sense to appear as classes in a domain ontology (they cannot be instantiated), we reformulated OGC 17-003r2 as follows:

I. The properties related to AcquisitionInformation in the standard (i.e., the), now is directly related to the class co:EarthObservation (as the AcquisitionInformation object was related to EarthObservation object in OGC 17-003r2)
II. The properties related to ProductInformation object, now is directly related to the class *co:EO Product*
III. The properties of the objects DataIdentification, Link and Offering in the standard, now are, also, directly related to the class *co:EO Product*
IV. The information related to EO metadata documents, now is directly related to the class *co:EOMetadataDocument*
V. The properties of the OrbitParameter and WavelengthInformation object are assigned to the classes *co:Orbit* and *co:Wavelength* respectively, which are linked directly to *co:EarthObservation*.

Indicatively, an excerpt of the mapping of OGC 17-003r2 to CO is presented in Table 6.1

OGC 17-003r2		Copernicus Ontology		
		Property	Domain	Range
EO/properties/acquisitionInformation/	Platform	co:platform	co:EO	co:Platform
	Instrument	co:instrument	co:EO	co:Instrument
acquisitionParameters/	acquisitionType	co:acquisitionType	co:EO	{NOMINAL^^xsd:string, CALIBTRATION^^xsd:string,OTHER^^xsd:string}
	acquisitionSubtype	co:acquisitionSubtype	co:EO	xsd:string
	startTimeFromAscendingNode	co:startTimeFromAscendingNode	co:EO	xsd:DateTime
	…	…	co:EO	…
orbitParameters/	orbitDirection	co:orbitDirection	co:EO	{ASCEDING^^xsd:string, DESCENDING^^xsd:string}
	lastOrbitDirection	co:lastOrbitDirection	co:EO	{ASCEDING^^xsd:string, DESCENDING^^xsd:string}
	…	…	co:EO	..
TemporalInformation/…	beginningDateTime	sosa:phenomenonTime co:hasStartDate	co:EO co:EOphenomenonTimeInterval	co:EOphenomenonTimeInterval xsd:dateTimeStamp
	endDateTime	co:hasEndDate	co:EOphenomenonTimeInterval	xsd:dateTimeStamp
Wavelength Information/		co:wavelength	co:EO	co:Wavelength

verticalSpatialDomain		co:verticalSpatialDomain	co:EO	co:VerticalSpatialDomain
Acquisition Angles/		co:acquisitionAngle	co:EO	co:AcquisitionAngle
EO/properties/aProductInformation	productType	co:productType	co:Product	co:string
	size	co:sizeInBytes	co:Product	xsd:integer
	productVersion	co:productVersion	co:Product	xsd:integer
	…	..	co:Product	..

Table 6.1: Mapping of OGC 17-003r2 to CO

Schematically, this is represented with Figure 6.8.

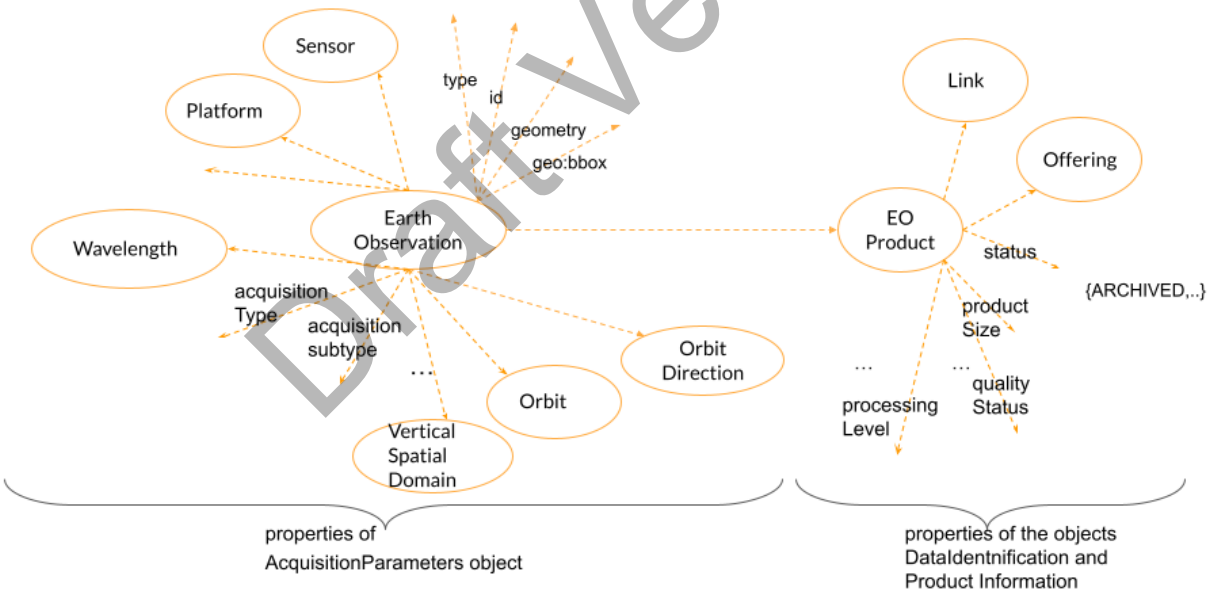


Figure 6.8: Schematic description of the representation of OGC 17-003r2 in CO

It is worth noting that the constraints of the standard are also represented in the ontology. For instance, the single multiplicity of the geometry of an earth observation process, posed by the standard, is expressed in OWL with the axiom (expressed in Manchester syntax):

co:EarthObservation **SubClassOf** co:geometry **exactly 1** co:Polygon

As it is already mentioned, the main properties of the EO Product class are based on OGC 17-003r2. Hence, besides the properties related to its identification (e.g., id, productVersion, productType),

other properties related to its processing (e.g., processing mode, processing method), quality (e.g., quality status, quality degradation tag) and coverage (cloud, snow coverage) are represented formally in the ontology. Also, technical information, including its size, spatial resolution, timeliness, EO product components, its primary thematic area, are also represented. Indicatively, in Figures 6.9 and 6.10, we provide screenshots from the Protégé ontology development tool, that presents most of the axioms related to EO Product class.

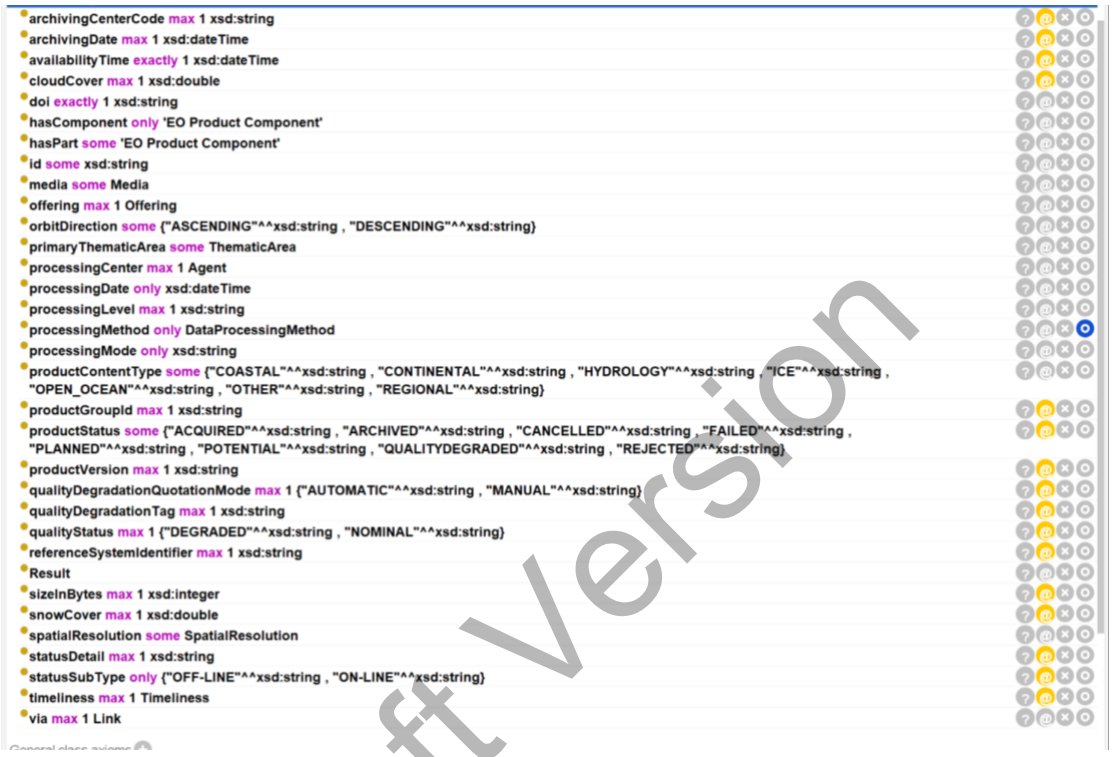


Figure 6.9: OWL axioms related to EO Product, as they appear in Protégé

Information about EO Collections is also represented in the ontology. For this the properties of the OGC 17-084r1 [C21] encoding were used. EO collections (data series) are collections of datasets sharing the same product specification, i.e., an EO collection contains a set of EO Products.

Figure 6.10: OWL axioms related to EO Collection, as they appear in Protégé

Additionally, more detailed information about the sentinel products is also included. For instance, for Sentinel-1, a full classification of its products is deployed based on the EO Equipment used (GPS, HKTM, SAR), on the processing level (L0, L1, L2), on the product type (SLC, GRD), on the sensor operation mode (EW, IW, SM, WV), on the beamID, and on the resolution class (low, medium, high- where these classes are defined with specific thresholds). This way, for instance, an EO product that belongs to the class co: S1 SM SLC L1 S1 Product, it is automatically inferred (from the respective OWL axioms) that this product is from Sentinel-1 mission, it is of processing level 1, it is of type Single Look Complex product, it is result of some SAR sensor operating in the frequency band C, in Strip Map operating mode with beamID S1. Indicatively, we present the full set of axioms related to co:S1 SM SLC L1S1 Product class in Figures 6.11, 6.12.

Figure 6.11: The axioms related to co:S1 SM SLC L1S1 Product class (part I)

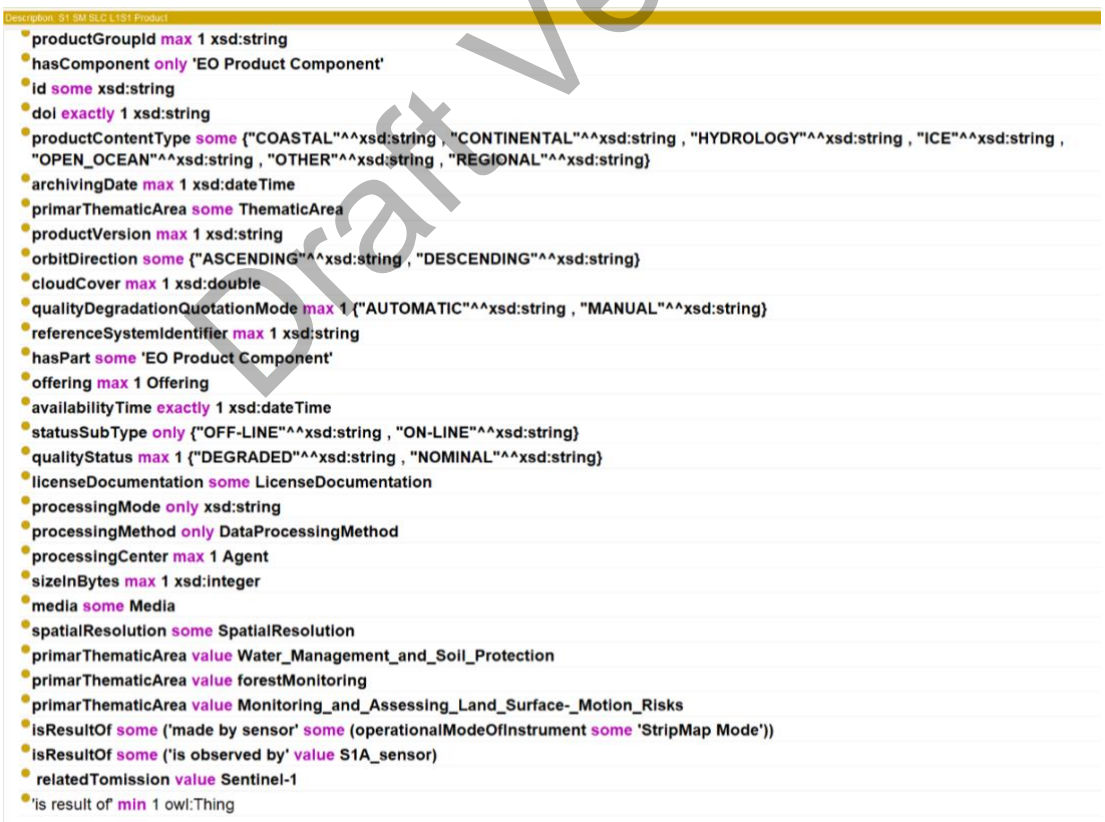


Figure 6.12: The axioms related to co:S1 SM SLC L1S1 Product class (part II)

Due to the large number of missions, we have focused only on the ones of the Copernicus programme.

D4. Geospatial and temporal knowledge

For the geospatial part of the knowledge, to be aligned with CREODIAS, the semantic search engine is linked to DBpedia. For the temporal part, as it is already mentioned, CO is linked to the Time Ontology.

D5. Knowledge about publications

For this domain the Open Research ontology (ORKG)[24]is reused, which presents the research contributions traditionally described in scholarly articles in a structured and semantic manner.

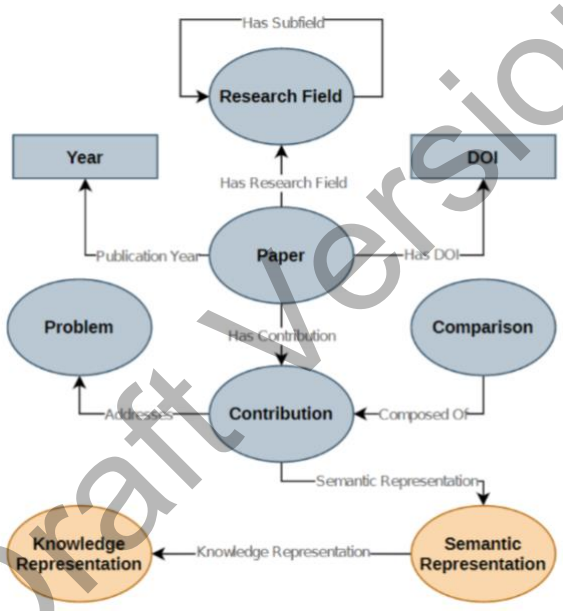


Figure 6.13: The ORKG ontology

6.2 Metadata from Bootstrapping Services and Resources

According to D5.1, the bootstrapping services and resources provided by AI4Copernicus are the following:

The datasets:

- TimeSen2Crop
- VectorDataOfHumanFeatures
- EnergyDataset

[24] https://gitlab.com/TIBHannover/orkg

The services:

- Sentinel-1 GRD pre-processing
- Sentinel-1 SLC pre-processing
- Sentinel-2 pre-processing
- Sentinel-1 Change detection – Amplitude Change Detection and Multi-temporal Coherence
- Sentinel-2 Change detection
- Deep Network for pixel-level classification of S2 patches,
- Harmonization of pre-processed Time Series of Sentinel-2 data,
- Long Short-Term Memory Neural Network for Sentinel-2,
- Pre-Trained Long Short-Term Memory for GeoTIFF samples for Agriculture
- Probabilistic downscaling of CAMS air quality model data

We have created the KG Bootstrapping Services and Resources[25] in which the aforementioned services/datasets and their metadata are described in detail. The KG is mapped to both AI4Copernicus and AI4EU.

The classification of the datasets and the services is presented in Figure 6.14:

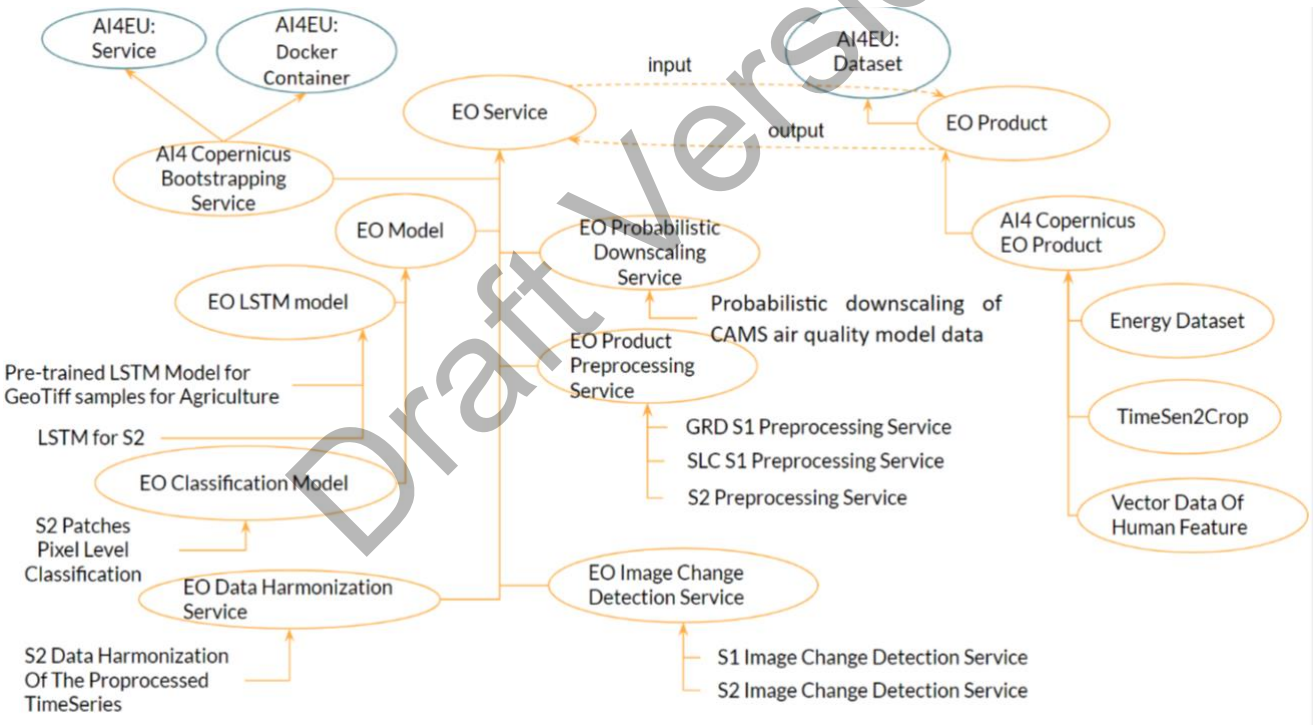


Figure 6.14: The hierarchy of the bootstrapping services in CO

All above services are instances of the *co:AICopernicusBootstrappingService* class (for clarity of the figure this is not depicted graphically) and the relevant subclasses of the Service class, therefore it

[25] https://drive.google.com/drive/folders/10UIL1mI-3HpRbXWZ36ZcF2Mz7FBvxChd?usp=sharing

inherits its properties, e.g., input, output, thematic area, etc. Additionally, all of them are also subclasses of the *AI4EU:DockerContainer* class, as they are available as dockerized applications.

In alignment to AI Asset Ontology WG, in which all ICT-49 projects are working to agree to a minimum core AI Asset Ontology, as it is depicted in Figure 6.15, we have also linked the AI4Copernicus services to core concepts such as Archetypical Problem, Algorithm and Technique.

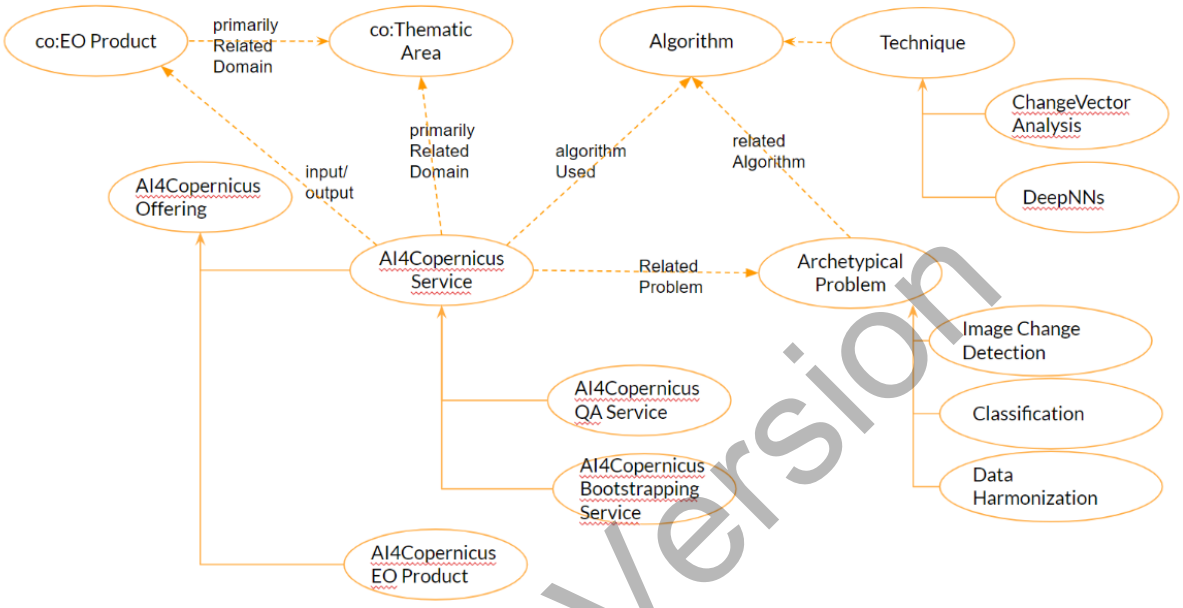


Figure 6.15: The properties of the AI4CopernicusService class related to core AI concepts

Finally, the metadata properties of the bootstrapping services are defined according to their inputs, outputs and thematic areas. Indicatively, we present in Figure 6.16 the metadata properties of Sentinel-1 GRD pre-processing.

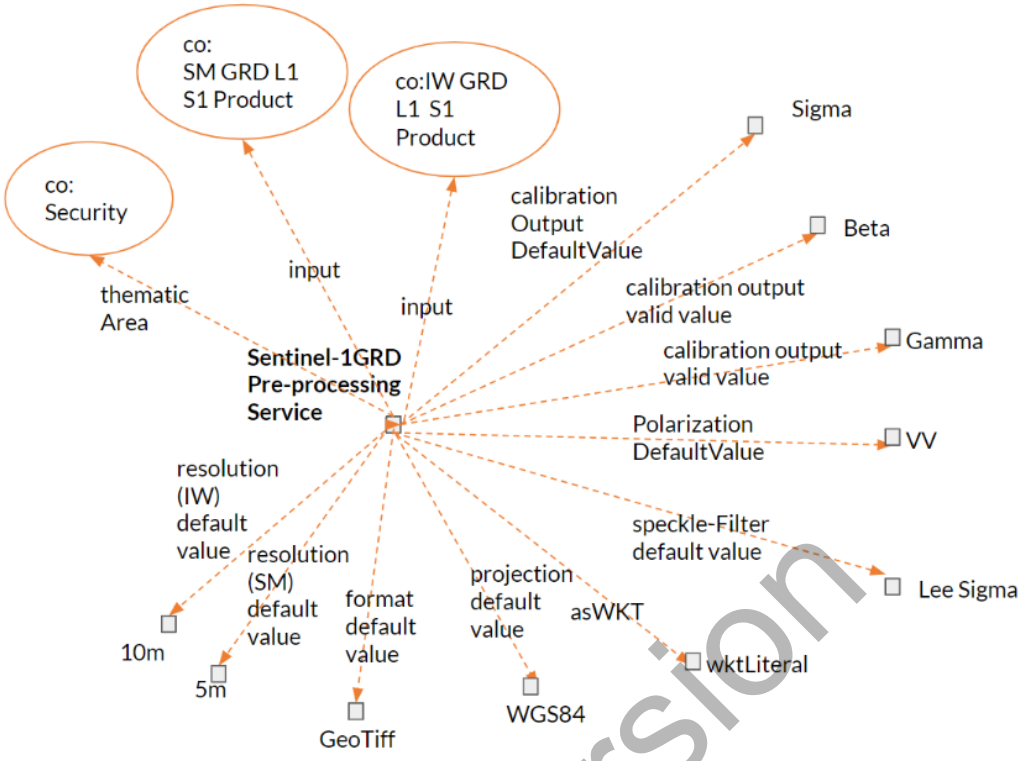


Figure 6.16: The metadata of the bootstrapping service: Sentinel-1 GRD pre-processing

7 The Question Answering Engine EarthQA

In this section we will discuss *EarthQA*, the Question Answering (QA) engine that is developed over CREODIAS SPARQL endpoint. The QA engine is developed using the Qanary methodology [BDK+16, BSD+17] and the Frankenstein platform [SRB+18] .

7.1 The QA Pipeline

Qanary is a lightweight component-based QA methodology for the rapid engineering of QA pipelines [BDS+16, BSD+17]. Frankenstein [SRB+18] is the most recent implementation of the ideas of Qanary; this makes it an excellent framework for developing reusable QA components and integrating them in QA pipelines. Frankenstein is built using the Qanary methodology developed by Both et al. [BDS+16] and uses standard RDF technology to wrap and integrate existing standalone implementations of state-of-the-art components that can be useful in a QA system. The Qanary methodology is driven by the knowledge available for describing the input question and related concepts during the QA process. Frankenstein uses an extensible and flexible vocabulary [SBD+16] for data exchange between the different QA components. This vocabulary establishes an abstraction layer for the communication of QA components. While integrating components using Frankenstein, all the knowledge associated with a question and the QA process is stored in a process-independent knowledge base using the vocabulary. Each component is implemented as an independent micro-service implementing the same RESTful interface. During the start-up phase of a QA pipeline, a

service registry is automatically called by all components. As all components are following the same service interface and are registered to a central mediator, they can be easily activated and combined by developers to create different QA systems.

Thus we take advantage of the Frankenstein framework to create QA components which collectively implement the QA pipeline reusing the components from GeoQA [PSB+18] and adding some more components that build complete QA pipeline over CREODIAS SPARQL endpoint.

The EarthQA pipeline for CREODIAS consist of the following components :

1. Concept Identifier (reused from GeoQA)
2. Instance Identifier (TagMeDisambiguate, reused from GeoQA)
3. Spatial relation Identifier (reused from GeoQA)
4. Property Identifier (reused from GeoQA)
5. Date Identifier (HeidelTime tool)
6. ProductType Identifier
7. Other Metadata Identifier
8. Query Generator (reused from GeoQA)

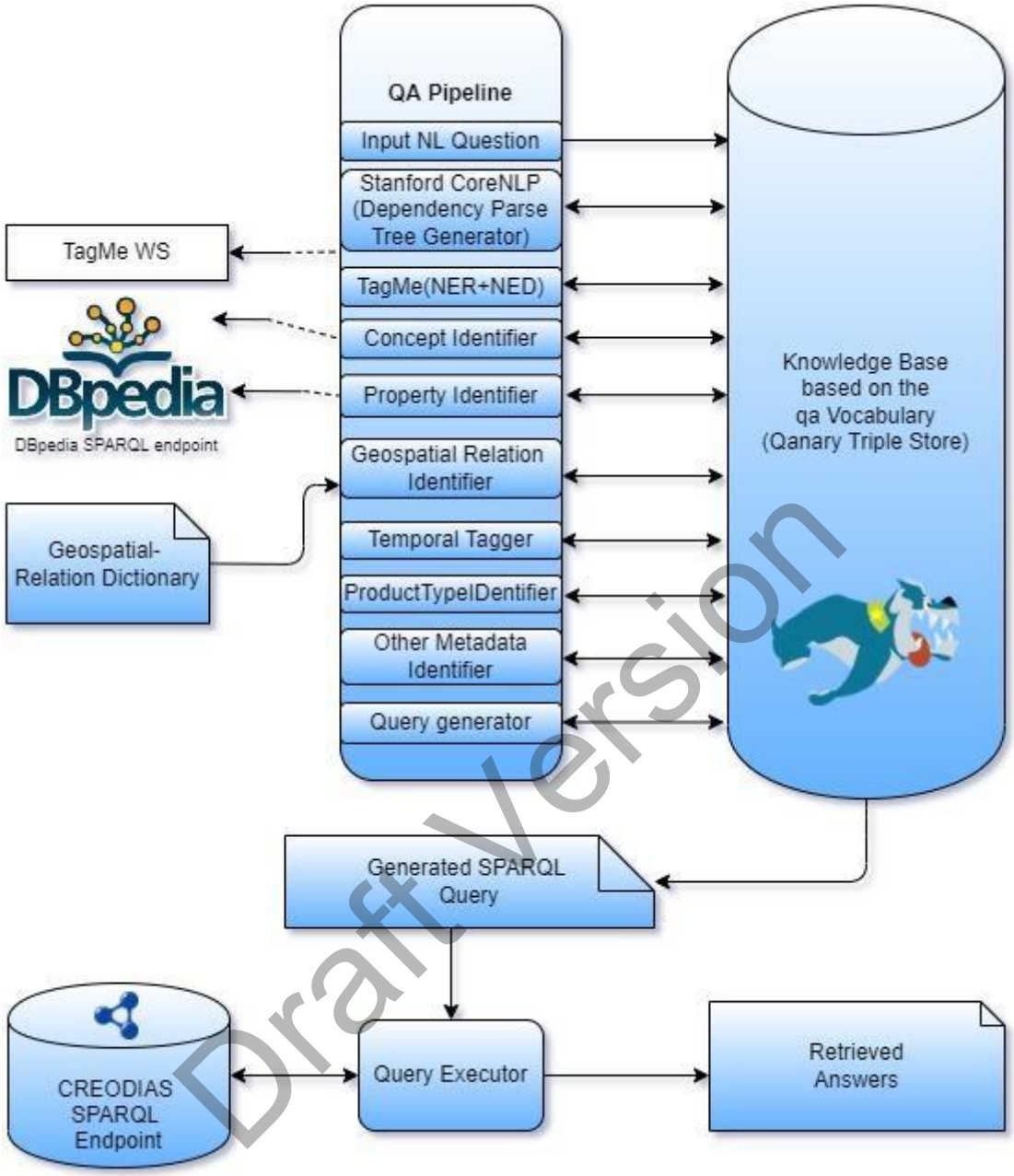


Figure 7.1 : The architecture of the implementation of EarthQA

It is to be noted that components reused from GeoQA have been modified or used as it is per requirement of task. To get a brief overview of working of the QA pipeline we will discuss an example question and output from every pipeline below. Consider the example question “Find all Sentinel-1 GRD images that show large lakes in Greece having an area greater than 100 sq km during October 2021”.

Dependency Parse Tree Generator: This module generates a dependency parse tree of the input question using StanfordCoreNLP tool.

Instance Identifier: This module identifies and maps the entity that is present in the input question to the appropriate resource of DBpedia ontology. For instance in the example question it will identify Greece and map it to `http://dbpedia.org/resource/Greece` in DBpedia ontology.

Concept Identifier: This module identifies and maps the concept(point of interest) that is present in the input question to the appropriate resource of DBpedia ontology. For instance in the example question it will identify lakes and map it to `http://dbpedia.org/ontology/Lake` in DBpedia ontology.

Spatial Relation identifier: This module identifies spatial relation present in the input question and maps it to the appropriate predicate in DBpedia. For instance in the example question it will identify in (within) and maps to `http://dbpedia.org/ontology/location` considering concept http://dbpedia.org/ontology/Lake and instance `http://dbpedia.org/resource/Greece` in the DBpedia ontology.

Property Identifier: This module identifies property of the concept or instance present in the input question and maps to the appropriate predicate of DBpedia. For instance, in the input question it will identify the area and map it to `http://dbpedia.org/property/area` considering the concept `http://dbpedia.org/ontology/Lake` that is present in the DBpedia ontology.

Temporal Tagger: This module identifies temporal keywords and annotates it with appropriate date in the input question. For instance, in the example input question it will identify "*2021-10*" and annotate it to October 2021.

Product Type Identifier: This module identifies metadata about Mission, Platform and Product type from the input question. For instance, in the example question it will identify Sentinel-1 GRD and map it to `http://ws.eodias.eu/metadata/mission/Sentinel-1` and `http://ws.eodias.eu/metadata/productType/GRD`.

Other Metadata Identifier: This module identifies other metadata about the feature/product like cloud coverage, orbit direction, processing level, swath etc. For instance, consider input question "*Find Sentinel-2 MSI products with cloud cover below 10%*" it identifies cloud cover and maps it to the predicate `http://ws.creodias.eu/metadata/attribute#cloudCover`.

Query Generator: This module takes output from all the previous modules into consideration and based on that generates the SPARQL query. For instance, for the example question it will generate following a SPARQL query:

```
select distinct ?title ?geom
where {
      select ?concept
      where {
            SERVICE <https://dbpedia.org/sparql> {
                  ?concept a <http://dbpedia.org/ontology/Lake>;
                  <http://dbpedia.org/property/area> ?property.
                  FILTER( ?property > 100000) .
                  }
            }
            ?x a <http://ws.eodias.eu/metadata/feature> .
```

```
        ?x <http://ws.eodias.eu/metadata/attribute#title> ?title .
        ?x <http://ws.eodias.eu/metadata/attribute#geometry> ?geom .
        ?x <http://ws.eodias.eu/metadata/attribute#mission>
             <http://ws.eodias.eu/metadata/mission/Sentinel-1> .
        ?x <http://ws.eodias.eu/metadata/attribute#productType>
             <http://ws.eodias.eu/metadata/productType/GRD> .
        ?x <http://ws.eodias.eu/metadata/attribute#startDate> ?date .
        bind(year(?date) as ?year) .
        bind(month(?date) as ?month) .
        FILTER(?year=2021 && ?month=10) .
    }LIMIT 1000
```

8 Conclusions

In this deliverable, we presented the software architecture of AI4Copernicus. Moreover, we discussed how it interfaces with the various components of the AI-on-demand platform, and CREODIAS and WEkEO, the two DIASes targeted by the project. This deliverable also presents two original contributions to the Copernicus ecosystem: EarthQA, a question answering engine for discovering Copernicus data and the Copernicus ontology.

9 References

[BEM14] Cody Burleson, Miguel Esteban Gutiérrez, Nandana Mihindukulasooriya Linked data platform best practices and guidelines. https://www.w3.org/TR/ldp-bp, 2014.

[BDS+16] A. Both, D. Diefenbach, K. Singh, S. Shekarpour, D. Cherix, C. Lange. Qanary - A methodology for vocabulary driven open question answering systems. Latest Advances and New Domains - 13th International Conference (ESWC 2016). Heraklion, Crete, Greece, May 29 - June 2, 2016.

[BSD+17] A. Both, K. Singh, D. Diefenbach, I. Lytra. Rapid engineering of QA systems using the light-weight qanary architecture. Web Engineering - 17th International Conference (ICWE 2017). Rome, Italy, June 5-8, 2017.

[C20] Yves Coene. OGC 17-003r2 - Earth Observation Dataset Metadata Vocabulary. Technical report, 2020.

[C21] Y. Coene, U. Voges, O. Barois. OGC EO Collection GeoJSON(-LD) Encoding Best Practice, 2021.

[FGJ97] Mariano Fernandez-Lopez, Asuncion Gomez-Perez, and Natalia Juristo. METHONTOLOGY: from Ontological Art towards Ontological Engineering. The AAAI97 Spring Symposium. Stanford, USA, 1997.

[H13] S.K. Haldar, Chapter 6 - Photogeology, Remote Sensing and Geographic Information System in Mineral Exploration, Editor(s): S.K. Haldar, Mineral Exploration, Elsevier, 2013, Pages 95-115.

[KHC+19] Janowicz Krzysztof, Armin Haller, Simon J D Cox, Danh Le Phuoc, Maxime Lefrancois. "SOSA: A Lightweight Ontology for Sensors, Observations, Samples, and Actuators." Journal of Web Semantics 56: 1–10. Crossref. Web, 2019.

[KMK19] Nikolaos Karalis, Georgios Mandilaras and Manolis Koubarakis. Extending the YAGO2 ontology with Precise Geospatial Knowledge. The 18th International Semantic Web Conference (ISWC-19). Auckland, New Zealand, 26-30 October, 2019.

[P16] Silvion Peroni. A Simplified Agile Methodology for Ontology Development. The 13th OWL: Experiences and Directions Workshop and 5th OWL reasoner evaluation workshop (OWLED-ORE 2016). Bologna, Italy, 20 November, 2016.

[PSB+18] D. Punjani, K. Singh, A. Both, M. Koubarakis, I. Angelidis, K. Bereta, T. Beris, D. Bilidas, T. Ioannidis, N. Karalis, C. Lange, D. Pantazi, C. Papaloukas, G. Stamoulis. Template-based question answering over linked geospatial data. Proceedings of the 12th Workshop on Geographic Information Retrieval (GIR'18). Seattle, Washington, USA, November 6th, 2018.

[SGA+20] Stelmaszczuk-Górska, M. A., Aguilar-Moreno, E., Casteleyn, S., Vandenbroucke, D., Miguel-Lago, M., Dubois, C., Lemmens, R., Vancauwenberghe, G., Olijslagers, M., Lang, S., Albrecht, F., Belgiu, M., Krieger, V., Jagdhuber, T., Fluhrer, A., Soja, M. J., Mouratidis, A., Persson, H. J., Colombo, R., and Masiello, G.: BODY OF KNOWLEDGE FOR THE EARTH OBSERVATION AND GEOINFORMATION SECTOR – A BASIS FOR INNOVATIVE SKILLS DEVELOPMENT, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLIII-B5-2020, 15–22, https://doi.org/10.5194/isprs-archives-XLIII-B5-2020-15-2020, 2020.

[SRB+18] K. Singh, A. S. Radhakrishna, A. Both, S. Shekarpour, I. Lytra, R. Usbeck, A. Vyas, A. Khikmatullaev, D. Punjani, C. Lange, M. Vidal, J. Lehmann, S. Auer. Why reinvent the wheel: Let's build question answering systems together. Proceedings of the 2018 World Wide Web Conference on World Wide Web (WWW 2018). Lyon, France, April 23-27, 2018.

[SBD+16] K. Singh, A. Both, D. Diefenbach, S. Shekarpour, D. Cherix, C. Lange. Qanary - the fast track to creating a question answering system with linked data technology. The Semantic Web - ESWC 2016 Satellite Events(ESWC-2016). Heraklion, Crete, Greece, May 29 - June 2, 2016.

[T20] Cassia Trojahn. D2.6 Semantic search v2. Copernicus Access Platform Intermediate Layers Small Scale Demonstrator (Candela), 2020.

[UCM12] Thomas Usländer, Yves Coene and Pier Marchetti. Heterogenous Missions Accessibility. ESA Training Manual, 2012.

[VBJ+20] Pauline Vincent, Matthieu Bourbigot, Harald Johnsen, Riccardo Piantanida, Sentinel-1 Product Specification, Ref: S1-RS-MDA-52-7441, S-1 MPC Nomenclature: DI-MPC-PB S-1, MPC Reference: MPC-0240, ESA Unclassified, 2020.

Appendix I

Collecting natural language questions for discovering Earth Observation datasets

Dear reader,

We would like to ask for your help regarding a question answering engine that we are developing in the context of the H2020 project AI4Copernicus (https://ai4copernicus-project.eu).

The question answering engine will be aimed at users of Earth Observation data, and it will enable them to discover datasets that are of interest to them by posing natural language questions (like these we are posing to search engines like Google today).

As part of the user requirements capture of our work, we are collecting interesting questions from users like you. We therefore ask you to kindly provide us with 10 questions that you may pose to a system like the Copernicus Open Access Hub, one of the five DIASes, or any other Earth Observation data portal, with the intention of discovering an EO dataset of interest to you.

Three examples of such questions that we expect our engine to be able to answer are given below:

1. Find Sentinel-3A Water Full Resolution (WFR) products with the data collected in January 2018.

2. Find Sentinel-1 products that may show Etna and areas around it in time of eruptions in March 2018.

3. Find Sentinel images taken during the summer months of 2020 which cover Athens, Greece and can be used to study air quality.

The questions that you will provide us should reflect your needs for Earth Observation data and the tasks that you do with it in your work. Optionally, you can provide your name, organization and email.

Appendix II

Natural Language Questions for discovering Earth Observation Datasets

Find images of a given urban area where we have news of unauthorized waste deposit
Find stereo-couples of Sentinel-2 images within given limits of viewing angles to create/update DEMs.
Return sentinel 2 imagery for locations in Africa affected by a drought
List all groups of satellite images where the maximum time difference between any image pair is less than <n> hours and where the best resolution of an image is better than <m> meters.
Retrieve Sentinel-2 imagery over an area of at least 20 sq. km covered at least 70% by potato cultivation and where there is no river or lake within 10km.
Retrieve the highest resolution images that cover all seas of Europe.
Retrieve all clear-sky images over a certain location between time X and time Y.
Find Sentinel-1 images covering the ice edge in the Barents Sea in March 2021.

Sentinel-2 images with fires scars of the Portuguese 2017 fires
how has vegetation activity been changing in the last 50 years in India?
Find Sentinel-1 products that may show Etna and areas around it in time of eruptions in March 2018
Find Sentinel-2 products of cotton fields in Pakistan
Find time series of Sentinel-2 images on a given urban area to monitor the urban growth. In particular this could be applied to coastal areas, where unauthorized buildings appear most often.
Return paired images of pre- and post-flooding events worldwide
Find Sentinel products that include areas in Austria where average temperature is expected to rise by at least 1C over the next 50 years based on at least two modeling forecasts from NOAA.
Retrieve all radar images for an area of interest (specified using a polygon) for the past week.
Retrieve all SAR images of floods in South-East England between 2010 and 2020
Find Sentinel-2 images with cloud-free coverage of (land)fast ice around Greenland in May 2020.
All Sentinel-2 images of crop season of 2018 (October 2017 to September 2018) with cloud cover below 60%
how much is human activity damaging forest health in the amazon?
Find all Sentinel-1 GRD images that show large lakes (and areas around) – of an area greater than 100 sq km (two SPARQL endpoints: CREODIAS and dbpedia)
Retrieve two Sentinel-1 products, before and after, a flooded event in South of France
Find time series of thermal images to monitor heat island in urban areas and their neighborhood.
Retrieve the longest available imagery timeseries of southern hemisphere glaciers
Retrieve Sentinel-2 imagery over European capitals within 50km from snowfall for at least 60 days during 2019 and not within 50km from the sea.
Retrieve all the Sentinel-1 or Sentinel-2 images for an area of interest (given the polygon).
Retrieve pairs of optical images showing significant land cover changes at a certain location [I imagine this one is very hard!]
Find Sentinel-1 images including Antarctic iceberg A68.
get the sentinel-2 images with the largest annual NDVI over Portugal during the crop year of 2020
how much forest area have we lost in the last 30 years in Greece?
Find time series (December 2017/2016) of Sentinel-1 images that show Svartisen glacier in Norway
Retrieve a Sentinel-2 product from "Okjokull", the glacier that has disappeared in Islande, and an EO products from the same area 20 years earlier
Find time series of gas profiles in urban areas to monitor gases' concentration.
Return paired imagery of locations affected by a late freeze event and on a normal year.
Find Sentinel products that include areas in Asia where the only source of fresh water within 20km is a river that flows through two or more different countries; and where these products can be used to detect crops; and find and existing crops detection model that can be applied on these Sentinel products.
Retrieve all GRD Sentinel-1 images that cover the Black Sea during the period 1/06/19-1/15/19.

Retrieve all images with <50% cloud covering the length of the Severn river in 2019
Find Sentinel-1 images for the Arctic in summer 2020 where sea ice drift exceeded 2 knots.
Find Sentinel-1 GRD images that show airports (and areas around) in Spain (two SPARQL endpoints: CREODIAS and dbpedia)
Find Sentinel-1 products from oil spill in the Black Sea
Find time series of Sea Surface Temperature images to monitor the impact of industrial discharges at sea, for example to study the correlation with the phases of industrial processes.
Retrieve Sentinel-2 imagery over an area in Scandinavia within 50km from any city with population 100000 or higher where snow cover has decreased during the last 10 years.
Retrieve Sentinel-2 images for all ports of Europe, that have less than 10% cloud coverage.
Find images where we have both an S2 optical image and an S1 SAR image taken on the same day at a given location
Find Sentinel-3 images greater than 50% cloud-free showing the Laptev Sea in December 2020 to January 2021.
Find all Sentinel-1 products that show Eight-thousanders (two SPARQL endpoints: CREODIAS and dbpedia)
Find optical products from ships on the beach (not port)
Find Sentinel products about an area anywhere in the world where current rainfall is similar to the rainfall predicted for Southern France by climate modeling forecasts for 2080; preferably (but optionally) where soil consistency is similar; and (optionally, if available) find literature about agroenvironmental experiments conducted in that area.
Retrieve Sentinel-1 or Sentinel-2 images that cover NATURA 2000 sites.
Find the most recent low-tide and high-tide image of the Thames Estuary this year
Find all Sentinel-1 images coincident with the vessel Polarstern during it's MOSAiC project drift autumn 2019 to summer 2020.
Find all Sentinel-2 images in the area of Brussels
Find Sentinel-2 products from new industrial areas
Retrieve Sentinel-1 images that cover Exclusive Economic Zones.
Find images at location X during the peak of the growing season [which might be known, or might be inferred from vegetation cover]
Find Sentinel-1 images showing icebergs within 20 nautical miles of Port Lockroy.
Find Sentinel-2 MSI products with cloud cover below 10%
Find EO products with a lot of changes in desert areas
Retrieve Sentinel-2 images that contain waterways, and have less than 10% cloud coverage.
Retrieve the most recent images of all bridges on the River Thames between Abingdon and Richmond
Find all Sentinel-1 products and display INSPIRE metadata records ID's (no geometry)
Retrieve Digital Elevation Model from Angola
Retrieve all products that contain information about sea temperature, waves, and wind speed and direction for all european sea areas.

Find Sentinel-3A Water Full Resolution (WFR) products with the data collected in January 2018
Find Sentinel-2 products with deforestation areas
Retrieve all Sentinel-1 and Sentinel-2 images that may contain vessels.
Retrieve Sentinel-2 time-series from Toulouse in France between the 1st of April and 31st of May 2021 (with less than 20% of cloud cover)
Find EO images from Greece where NDVI is less than the average of the past 10 years
Find time series on a given geographic area in which you can highlight changes from vegetation or base soil to high variance sub-images.
Find time series on a given geographic area in which you can highlight changes from vegetation or bare soil to spectral signature of concrete/asphalt/roofs. Cross-check the findings with cadastral databases.
Find thermal images of cities and surrounding areas and look for strong temperature gradients between the city and the suburbs. Cross-check the findings with city maps, with meteorological databases and with electricity consumption databases
Find images of urban areas where buildings have solar panels on the roof. Same search for extra-urban solar farms. Cross check the findings with solar panels databases.