C-SCALE Copernicus Data Access and Querying Design

Status:	Under EC Review	Planned due date:	30 June 2021
Version:	V1.0	Submission date:	30 June 2021
Lead Participant:	CESNET	Lead Author:	Zdeněk Šustr
Related WP:	WP2	Document Ref:	D2.1
Dissemination Level:	Public		
Document Link:	https://doi.org/10.5281/zenodo.5045317		

Deliverable Abstract
The C-SCALE Data Federation brings together European providers of Earth observation product archives to facilitate seamless access from analytic workloads running in EOSC. The architecture of the federation, and descriptions of underlying technologies and components, are presented within this document.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017529.

COPYRIGHT NOTICE

C-SCALE receives funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 101017529.

DELIVERY SLIP

Date	***Name***	***Partner/Activity***
Lead Author:	Zdeněk Šustr	CESNET
Contributors:	Sophie Herrmann Pavel Břoušek Dirk Daems	EODC CESNET VITO
Moderated by:	Zdeněk Šustr	CESNET
Reviewed by:	Enol Fernández Matti Heikkurinen	EGI EGI
Approved by:	C-SCALE Activity Management Board (AMB): Christian Briese (CB), Diego Scardaci (DS), Charis Chatzikyriakou (CC), Zdeněk Šustr (ZS), Enol Fernández (EF), Björn Backeberg (BB), Matti Heikkurinen (MH)	EODC, CESNET, EGI Foundation, Deltares

DOCUMENT LOG

Issue	***Date***	***Comment***	***Author(s)***
V0.1	10 May 2021	Initial document structure	Zdeněk Šustr
V0.2	9 June 2021	Document ready for inter-WP review	Zdeněk Šustr
V0.3	23 June 2021	Reviewers' comments implemented	Zdeněk Šustr
V1.0	30 June 2021	Final release	Zdeněk Šustr

Contents

List of Figures

List of Acronyms

Acronym	Description
AMB	Activity Management Board
API	Application Programming Interface
C-SCALE	Copernicus – eoSC AnaLytics Engine
EO	Earth Observation
EOSC	European Open Science Cloud
GEOSS	Global Earth Observation System of Systems
GOCDB	Grid Configuration Database
HPC	High Performance Computing
HTTP	Hypertext Transfer Protocol
IDP	Identity Provider
OIDC	OpenID Connect
SAFE	Standard Archive Format for Europe
STAC	Spatial Temporal Asset Catalog
URL	Uniform Resource Locator
VO	Virtual Organisation
WP	Work Package

Executive Summary

C-SCALE brings together experts in Copernicus and European Open Science Cloud (EOSC) services and solutions to provide a blueprint for solving the challenge of Copernicus full-term data archive. It has proven notoriously difficult to find an optimal solution for this problem, primarily due to the capacity requirements.

State of the Art

Existing attempts at creating such an extensive archive have been driven by their respective business cases, which put emphasis on added value, sidetracking the basic goal of redistributing Copernicus Earth observation products. That is the case for leading *state of the art* services such as Google Earth Engine [R1] or ArcGIS [R2], and even for emerging open source platforms such as OpenEO [R3]. All of them prefer to expose functional interfaces for direct analysis, rather than making the raw data accessible to users who have piloted their own analytic workflows on their "desktops" and wish to scale out their coverage without locking in with any of these specific providers, preferring to retain full control of their analytic workflows, especially if they already have adequate processing resources available locally or in shared infrastructures such as the European Open Science Cloud [R4].

They often recruit themselves from among the users of the Copernicus Open Access Hub [R5], which lies on the opposite side of the state-of-the-art spectrum, offering global coverage and a full archive of Sentinel Earth observation products, but with performance unsuitable for scaling one's analysis beyond pilot stage.

To sum up, there are flexible, agile solutions that force at least some level of lock-in, or fully-open solutions where the processing is prohibitively costly in terms of time or resource requirements. C-SCALE will provide a solution that has an ideal mix of fluidity in the data management and openness in terms of portability of the applications between different computing resource providers.

Open, Distributed Sources

While a centralised Copernicus full-term data archive may not be feasible, there are organisations who maintain various archives for limited areas of their interest. These include national archives, or archives of certain geographic regions and product types maintained, e.g., for specific research purposes. This document describes the architectural approach C-SCALE will use to integrate these heterogeneous resources into a 'system of systems' that will offer the users an interface that in most cases provides similar functionality and quality of service as a full-term data archive would.

The intended audiences for this deliverable are solution architects, researchers and maintainers of specialised Copernicus archives that are interested in using and enhancing the C-SCALE platform to increase the capacities and capabilities of their services and make them available to new audiences (including EOSC). However, we hope this deliverable will be useful also as a description of the design philosophy and rationale for the choices made, and thus useful background e.g. for training, education and policy work.

In abstract, a solution requiring data covering certain area may be able to find such an archive carrying all the products they need, or find a combination of such archives, who may satisfy such need in conjunction. It is the main purpose of C-SCALE Data federation to make such approach possible, i.e., to allow users to identify archives who might hold data relevant to their use case, and make the contents of said archives accessible in a uniform, seamless manner.

Overall, the design and the services of the C-SCALE data federation aim at allowing partners to operate their services as before, but with better discoverability and universal accessibility, especially from automated analytic platforms.

Guiding Principles of the Federation

The federation approach is based on the following principles:

1. Earth observation data will be retrieved directly from the provider that holds them, avoiding bottlenecks and duplication
2. The federation will expose a common query interface and dispatch queries "pass-through" to C-SCALE providers
3. Redundancy of data and metadata stored anywhere across the federation will be minimized

It is not the goal of the federation to make one giant storage element, and neither is it its goal to build yet another metadata database. The C-SCALE data federation wants to make products accessible directly from providers who hold them. If a given use case requires caching of data, let that be implemented with support from the federation, but not as a general feature of the federation. Federation services should be in place to help users find their way to the right provider, and then use the discovered resource as efficiently as possible.

Data Discovery

Available products are discoverable at provider's sites, reflecting the current status and contents of the site. In C-SCALE, queries for products are also received through a common query interface and dispatched in a *pass-through* fashion to sites who may hold matching products. Answers from individual sites are then collated and delivered as a response to the original query. A component to perform this discovery feature and implement the common interface (the *Metadata Query Service*) will be developed in C-SCALE. An evaluation of suitable protocols has been performed and the SpatioTemporal Asset Catalogs (STAC) suite of protocols and API specification has been selected.

The redistribution of queries among federation sites can also be extended to existing metadata catalogues such as the NextGEOSS catalogue. These may, on one hand, give more complete overview of products, especially those available from outside the federation. On the other hand, metadata catalogues are principally always delayed in providing a complete and truthful image of reality, since they must inherently overlook the freshest changes, i.e., the availability of recently acquired products as well as the unavailability of recently removed products.

Taking Data Retention Policies into Account

In the worst-case scenario, each user query is forwarded to each site participating in the federation. This is not a major issue and the federation may function in that manner. C-SCALE design includes plans to optimise this by taking the focus (e.g., geographical area of interest) and data retention policies of the sites into account and use that description to rule out candidates who are certain not to hold any data relevant to the given user query. Such sites then do not need to be queried at all, saving on communication, load, and timeout margins.

The information on site's focus and policies, being relatively stable, will be specified by participating site managers as part of their registration into the Data federation, alongside other "stable" information such as site name and service endpoints.

Data Access

With the aforementioned data discovery approach, the user is bound to receive a list of Earth observation products matching their given query. This response may include multiple listings of the same product in case it is held by multiple sites in the federation. The response may also potentially hold products available from outside the federation (in case neither partner had the product but it was reported by some metadata catalogue, such as NextGEOSS).

Authentication

Earth observation products remain available from the individual sites in the federation, and the user may proceed to download them. The sites are federated under a single *identity federation*, EOSC AAI, so that a single identity may be used to access any site in the federation.

Authorization

C-SCALE does not plan to provide a centralized solution for authorization management, and neither do current partners in the C-SCALE Data federation intend to impose authorization restrictions on federation members, yet it is technically possible to do so. If required, authorization information (mapping restricted datasets to entitled users or user groups) may be stored and decisions may be taken locally. That is, if a certain dataset is only available for a certain user group, that authorization decision is made as user tries to access the product. They will be either rejected or served the product, depending on their identity.

Access Optimisation

In a typical EOSC-based use case, a virtual organisation will have one or more contracted resource providers where resources are available and where processing takes place. Earth observation data will be available from data federation members, and may or may not be collocated with the processing resources. A specific Earth observation product may be available from multiple providers with different characteristics (e.g., available bandwidth, co-located computing capacity or storage type). C-SCALE will explore using this information to basic level optimisation of data access. However, aiming at full, globally optimised approach is beyond the scope of the current initiative. At the very least, setting a federation-wide priority ranking should be possible to clearly distinguish backup resources (those that should only be used if no other options present themselves) from regular resources.

On another level, workload managers should also make sure they prefer collocated data sources to remote ones. However, that decision takes place mostly outside the scope of the data federation.

Relationship with the Compute Federation

The C-SCALE Compute federation, deploying Copernicus analytic workflows on EOSC cloud, HTC and HPC resources, is seen as the principal “customer” of the Data federation. Indeed, the design of the data federation has been drawn with automated “robot” jobs in mind.

In essence, the main points from the perspective of a user in the Compute federation are:

1. The typical workflow (query, download, process) remains the same
2. Existing workflows unaware of the C-SCALE Data federation are not disrupted, but may be extended to use the C-SCALE data federation with local changes
3. The same identity federation (EOSC AAI) covers both the Compute and the Data federation

It is fair to acknowledge that the greater part of compute–data interoperation lies with the compute platform, which is responsible for carrying the right access keys to access EO products, and of processing query results including optimisation as mentioned in the previous section.

Relationship with EOSC

The C-SCALE Data federation positions itself as a subset of service providers offering their services through EOSC. It emphasises the use of services and federation principles offered either centrally by EOSC, or available in the wider EOSC ecosystem by other established partners. Only where such service is not already available does C-SCALE propose to contribute its own solution.

Likewise, the C-SCALE Data Federation caters primarily to EOSC users, accessing Copernicus Earth observation products from computing sites integrated with EOSC.

1 Introduction

This document outlines the overall architecture of the C-SCALE data federation, whose main purpose consists in redistribution of Earth observation products for processing in the C-SCALE analytics platform. The document consists of two main sections. Section 2 “Data Federation” describes the overall intent and structure of the federation, the interfaces and features that will be integrated across the federation, and the requirements for providers to join the federation. Chapter 3 “Core Federation Services for Data Discovery and Access” describes centralized components that will be available to the federation – either those already available from EOSC, or newly developed for the needs of C-SCALE.

2 Data Federation

The C-SCALE data federation is a federation of Earth observation product providers. Participating sites are federated in the following areas:

- *User authentication*, allowing for authenticated access to data at federation sites. In the context of C-SCALE, user authentication (see Section 3.3) is used primarily for bookkeeping and usage statistics, not to restrict access to data, although even that is possible (Section 3.3.3).
- *Information system*, publishing essential characteristics of sites in the federation. It is intended primarily for machine readability and relies on existing services available in the EOSC ecosystem, namely the GOCDB (Section 3.2).
- *Product access protocols*, whose choice is governed by existing usage patterns and compatibility with the authentication layer (Section 3.3 again).

Participation in the federation is not limited by the choice of actual storage solution used at the given site. C-SCALE project will specifically aim at providing necessary functionality in software stacks used by C-SCALE partners. They are:

- DHuS – the DataHub System
- EODC Data Access
- TerraCatalogue
- CREODIAS

The federation is not closed to third parties operating other software solutions as long as they meet technical requirements, but the C-SCALE Project will not specifically contribute to their integration beyond providing consulting, support and documentation.

Third party sites can either actively register as data providers in the federation, or have their registration – with their consent and appropriate configuration changes at their sites – filled in and "vouched for" by C-SCALE members. This may be the case, for instance, for the numerous sites running DHuS [R6] if the contributions by C-SCALE get accepted upstream and redistributed to those sites.

Figure 2.1 is taken from the C-SCALE proposal and shows the position of the C-SCALE Data federation within the EOSC ecosystem. The individual objects shown in the diagram carry the following meaning:

- *EOSC AAI*, *Portal* and *Core Services* are *existing* components in the EOSC ecosystem, through which the C-SCALE platform will be accessible.
- *HPC Federation* and *Cloud Compute Federation* integrate resource providers where analytic processing/workflows are performed. In the context of C-SCALE they are represented by the C-SCALE Compute Federation. It is an integration effort by C-SCALE, which, however, falls outside the scope of its document.

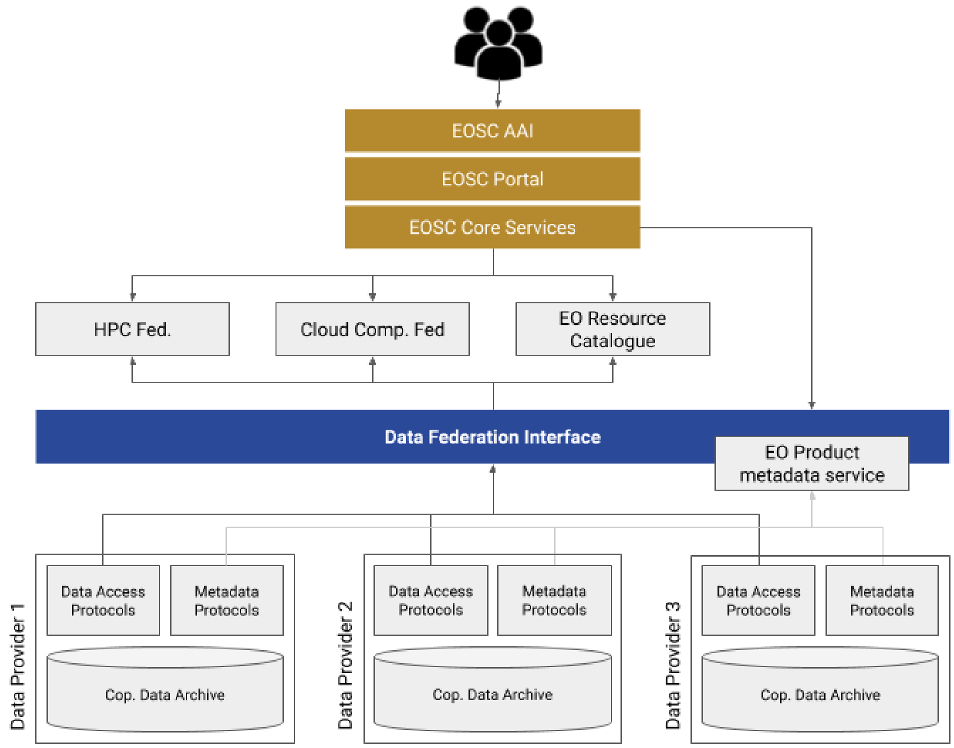

Figure 2.1: C-SCALE data federation within the EOSC ecosystem.

- *EO Resource Catalogue* is a catalogue of providers of Earth observation products, i.e., of sites where Earth observation data can be obtained. It is **not** a catalogue of products themselves. It is an *existing* catalogue in EOSC, wherein Data federation sites will be registered with a set of custom, C-SCALE specific attributes.

- The *Data Federation Interface* is a formal interface – a set of protocols – allowing search and transfer of Earth observation products. Services implementing the interface partly already exist (namely those for data access), partly will be developed in C-SCALE (namely the EO Product Metadata Service).

- *EO Product Metadata Service*, also referred to as the *Metadata Query Service* (Section 3.1), is a centralized service of the Data federation that can accept criteria-based queries for Earth observation products and locate matching products across providers within the federation.

- *Data providers* are members of the federation who typically host data accessible to users, as well as a query interface of their own, which make it possible for users to find data relevant to their queries in that provider's (and possibly also in other providers') site or sites. The providers have the bulk of the required functionality already in place, but several missing integration components especially for authentication and data discovery will be developed in C-SCALE. The three *Data Providers* are shown in the diagram for illustration purposes. In reality, all partners participating in the Data federation effort in C-SCALE are expected to become providers, plus there may be third-party providers integrated into the federation as explained above.

The federation is expected to grow as integration components for different software stacks at providers' sites are made available.

2.1 Site Requirements

The C-SCALE Data federation targets sites that perform one or (preferably) both of the following functions with regard to EO products generated in the Copernicus programme:

1. Redistribution of Earth observation products from producers towards user communities
2. Attribute-based discovery of Earth observation products

Sites that participate directly in establishing the federation are currently providing both functions, relying on their existing storage, lookup and access solutions, which they are not planning to replace but rather configure or – where necessary – extend to enable integration with the federation. A survey of participating sites and software stacks represented in the emerging federation is given in Table 2.1.

Software stack *used by WP2 partner*	**Product Access Mode**	**Query Interface**
Data Hub Software (DHuS) *CESNET, GRNET*	HTTPs, basic authentication, self-registration	OData [R7]+ OpenSearch [R8] (both HTTPs, basic authentication)
EODC Data Access *EODC*	FTP, open access	CSW [R9] (over HTTPs, open access)
TerraCatalogue *VITO*	HTTPs, OIDC authentication (through keycloak)	OpenSearch (over HTTPs, open access)
CREODIAS *CloudFerro*	HTTPs, OIDC authentication (through keycloak)	CSW + OpenSearch (both HTTPs, open access)

Table 2.1: Earth observation product redistribution software used by relevant parties in the data federation, indicating currently supported protocols for data access and lookup. Those are the two most important features to be unified across the federation.

It will be a major objective of WP2 to enable a common mode of access to EO products across all participants and under a common authentication schema, accepting federated identities provided through EOSC AAI, and to provide a common query endpoint to look up products across the federation with a single call.

The federation will rely on standards, which will simplify integration of other parties. There are in fact 3rd party services that, while not covered by C-SCALE objectives as such, might be relatively simple to integrate. They are mentioned in Table 2.2.

2.2 Interacting with the Federation

It is the principal purpose of the federation to support the Copernicus analytics platform being set up by C-SCALE. As such, the federation is there not only to allow interactive access by users, but – more importantly – by automated workflows running analytics in the platform. Figure 2.2 shows the

Software stack *used by*	Product Access Mode	Query Interface
Data Hub Software (DHuS) *ESA*	HTTPs, basic authentication, self-registration	OData + OpenSearch (both HTTPs, basic authentication)
NextGEOSS Data Hub *NextEOS, CESNET*	N/A	OpenSearch over HTTPs, open access

Table 2.2: 3rd party sites that might be considered for inclusion into the federation within the time frame of C-SCALE.

sequence diagram of a basic use case wherein an automated job queries the federation for products and accesses matching products for processing.

In Figure 2.2, the individual objects participating in the procedure are as follows:

- *User* is a physical user with their identity registered in EOSC AAI.
- *UI* is the *user interface*, which depends on use case and may range from the user's own local computer to a mobile application, interactive shell on a shared server, a web-based portal or even a portlet in the EOSC portal. Its design and implementation is up to each given use case, although C-SCALE's native and adopter use cases can expect assistance from members of both the Data and the Compute federations in setting up their UIs.
- *Compute platform* is a cloud platform or HTC resource running a job or an HPC resource running an HPC task, performing analytics in the Compute federation. In the most typical scenario it runs an automated workflow charged with analyzing a single or a series of EO products.
- *MQS* is the *Metadata Query Service* (Section 3.1), developed and operated as a central component in the C-SCALE Data federation.
- *Resource Catalogue* is the GOCDB, a central service available in EOSC through EGI.
- *Data Site(s)* are individual partner sites in the C-SCALE data federation.

2.2.1 Basic Workflow

In this basic use case, the workflow starts with the *user* initiating an analytic workflow through the *user interface* (UI). The actual processing is performed in the *analytics compute platform*. It begins by querying for products to analyse. A wide range of conditions (queryables) is supported, starting by specific product identifiers and ending with a range of spatial, temporal, provenantial and other criteria.

The query is sent through a standardized interface to the *Metadata Query Service*, which is capable of understanding the query to some extent. It uses the *EO resource catalogue* to identify (i.e., preselect) sites that might hold matching products, and forwards the query only to those sites. What results is a list of products, complete with download locations, matching the original query. This list is returned to the computing job in the analytics platform.

From the point of view of an analytics job running in the C-SCALE Compute platform, the *data sites* act as object storage services whence objects (EO products) can be downloaded in their entirety but

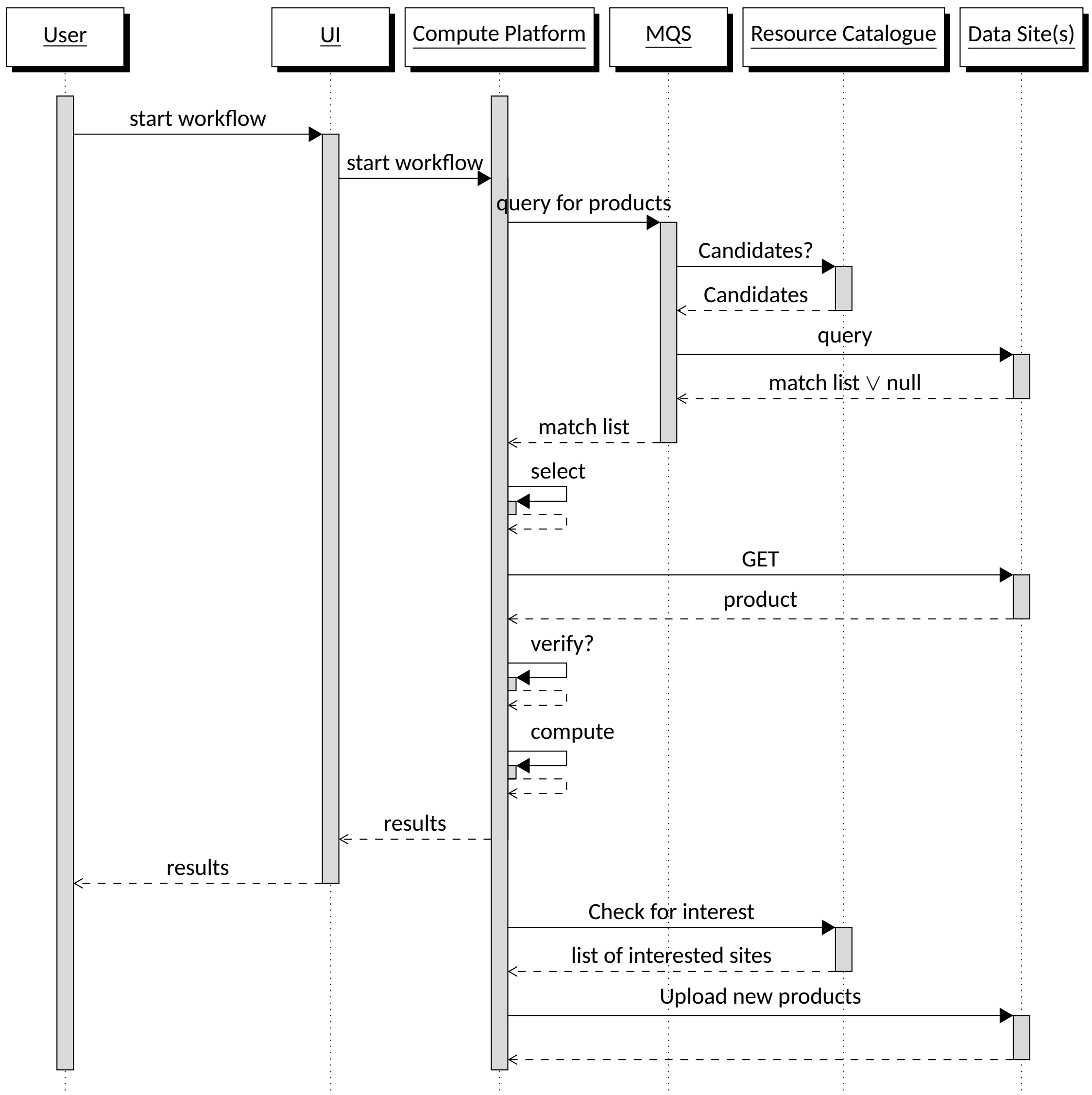

Figure 2.2: Basic use case sequence diagram

no non-sequential access is supported. Therefore a vast majority of use cases will involve downloading the required products to local scratch storage for efficient analysis. Hence the analytics job in the C-SCALE platform goes on to download the matching products, or potentially just a subset of those, based on additional criteria evaluated locally. Products are downloaded directly from the *data sites*, relying on URLs provided in the response from the Metadata query service. Once downloaded, the products are ready for analysis.

As an optional final step, if the results of the analysis fit the area of interest of any participating data site, the results may be uploaded back to that site. Potential recipients are identified through the EO resource catalogue, wherein site metadata indicate not only the site's interest in receiving certain types of input, but also the upload endpoint that would receive them (Section 3.4).

2.2.2 Alternative Workflows

The basic workflow introduced above may be modified in different ways to match the specific needs of a community. For example, the user may interact with the metadata query service directly if they only wish to perform queries and obtain results. C-SCALE does not envision in-house development of any rich graphical user interface to support that scenario, but 3rd party projects such as Intake [R10] develop STAC-enabled user tools, and using template queries with the standard programmatic interface of the query service (STAC – see Section 3.1.2) is also possible.

In direct opposition to that case, the user may also skip the query altogether, provided they supply the list of products for processing (including URLs) to the job in a different manner, either at the beginning or as the workflow progresses.

Another modification to the basic use case consists in including an additional interactive step, wherein candidate products collected from the query service are presented to the user for confirmation or prioritization. After that interactive step the analytic job may continue with processing.

Yet another modification may stem from the need to run certain analytics regularly as new products arrive. In that case, the analytics job would not be initiated interactively by a user, but rather by an automated scheduler.

2.2.3 Additional Services

The running of the analytic workflow may benefit from additional steps and procedures that may be aided or even provided by the federation.

Firstly, the services of the data federation may assist in deciding between alternative sources if a given product is available from multiple providers. Such assistance may come either in the form of a sufficiently rich description of the site in the EO resource catalogue, or even in the form of a separate service that would actively measure the performance of data federation sites with respect to different computing resource providers, and could reliably choose the best-performing endpoint to download and/or upload large data files characteristic for earth observation. This is a recurring requirement not exclusive to the Earth observation community, discussed in greater detail in Section 3.5.1.

Another service the federation could provide with respect to user access and queries would consist in understanding and evaluating its own rate of full/global coverage. It can be assumed that the federation, at least in its initial state during C-SCALE, won't be able to hold all Copernicus products generated in history. It might be, however, possible to make sure – at least for certain products such as those by ESA Sentinel – that a response to a user query, based on certain conditions, also contains a statement of how large a portion of all matching Sentinel products is actually included in the response, i.e., how many other matching products might be found outside the federation. This is a requirement identified through use case engagement, and is further discussed in Section 3.5.2.

3 Core Federation Services for Data Discovery and Access

Access to the federation, and the interoperability of user tools with its various components, will rely on several core services. Mostly, the required services are already available in the EOSC ecosystem, ready for reuse or extension. Only missing gaps will be filled in with fresh development, particularly the Metadata Query Service described in Section 3.1.

3.1 Metadata Query Service

The Metadata Query Service is a central component that will be developed and deployed within C-SCALE.

3.1.1 General Concept

The Metadata Query Service (MQS) is the central entry point to query for metadata across the federation. It is used to identify all datasets available at the different data providers matching given constraints. A number of different constraints (also called queryables) will be supported by the MQS to allow the user to restrict the results to his / her needs.

To enable this centralized search functionality the query interfaces of the single data providers will be connected to the central MQS. In this integration the central service will act as metadata query broker only. Therefore, no caching is performed but rather the queries are only redirected by the integrated query services and then the results are aggregated and returned to the user. This also ensures that always the latest datasets are found. To access only promising data providers for a given query the Federation Catalogue will be integrated (Section 3.2).

The MQS will be publicly available, so without any authentication. This will allow users to get an overview of all available datasets, even if they were not authorized to access all the data. This might pose a problem, which does not, however, impact C-SCALE with its present C-SCALE consortium, where all data providers (as well as sites considered as potential 3rd party providers) offer open data and no access restrictions are envisioned. Even if it was not the case, this design does not totally preclude the imposition of access restrictions. The way it would be done is briefly outlined in Section 3.3.3.

3.1.2 Interface Selection

As central entry point for metadata queries the MQS must define two interfaces to allow interaction with its user. Looking at Figure 3.1 those are (in the basic workflow the *Compute Platform* acts as user of the MQS service):

1. Request: *query for products* (defines: query interface)
2. Return: *match list* (defines: return format)

OpenSearch	STAC-API
+ Stable Search API + Simple & well-known Query Interface (SciHub) + Supported by most project partners, although there are different flavors → mapping required – No active development or community	+ Active community, tools (server, client), support + Active development, new features, bug fixes + Small/structured extensions + Simpler metadata structure – Currently no stable release, stable base parts but large changes in query extension

Table 3.1: Comparison between OpenSearch and STAC-API, based on their main pros and cons

3.1.2.1 Evaluation

The following protocols have been considered in C-SCALE's design stage, while choosing the most suitable protocol the MQS should expose:

- OpenSearch
- CSW
- STAC-API

All of them are in some way relevant to products already in use by C-SCALE partners. OpenSearch and CSW each are already implemented by a subset of partners in the data federation (refer back to Table 2.1), while STAC-API (Section 3.1.2.2) is used by OpenEO, which is a tool planned for deployment in the C-SCALE Compute platform.

From out of the three options, OpenSearch and STAC-API have been serious contenders, while CSW was dropped early in the evaluation process since it was found inferior to both OpenSearch, with which it is collocated, e.g., in CREODIAS, and to STAC-API, with which other partners are involved through OpenEO. Table 3.1 shows the relative merits of OpenSearch vs. STAC-API as identified by the MQS team. The final choice has been motivated primarily by STAC-API's active development, extensibility and relevance to OpenEO.

3.1.2.2 STAC

STAC-API [R11] was selected as query interface and STAC [R12] as return format. STAC stands for *Spatial Temporal Asset Catalog* and defines how geospatial information should be described to facilitate easy indexing and discovery of data. Hence, the core specification of STAC only describes the structure of a spatial temporal asset (metadata). On top of STAC, STAC-API can be used to query such catalogues. In detail for searching the item-search functionality is leveraged and for results the item specification is applied [R13]. The response from STAC-API follows the GeoJSON specification [R14] – again a well-structured, machine friendly format for encoding a variety of geographic data structures.

Both STAC and STAC-API are structured in a simple base – extension model. Base defining only a minimum set of return fields (STAC) respectively query structure and queryables (STAC-API). This allows to first implement only a minimal working product and then to extend the two interfaces depending on external requirements. In the base version STAC-API allows searching by collection, ids, time range and spatial extend (bounding box and geometry). These queryables are supported by all members of the data federation. Future extensions of queryables will be based on requirements and the availability of queryables at the different data providers (see Appendix 1 for more details).

It should be mentioned that especially STAC-API is under active development and major changes are expected in the next months concerning the support of additional queryables. As stated above, in the initial phase only base capabilities should be implemented. Hence, it is expected that stable versions of all important parts of the API will be available at the time of implementation. Additionally, the current development allows us to participate in discussions and integrate our requirements into the new standard.

The active development and active community are some of the main advantages of STAC compared to other standards such as OpenSearch or CSW. Development efforts enable both new features and bug fixes in the specification. This can also be seen in the large advances of the specification over the last years. The community already provides a number of useful tools to work with STAC [R15] and can give support if problems arise. Moreover, both the STAC-API query interface and the STAC metadata structure (used as return format) can be easily extended depending on external requirements – considering additional queryables and return fields.

In summary, although STAC/STAC-API is not yet supported by any federation member (see Table 2.1), this quite young standard has a number of advantages compared to other available ones and is therefore selected.

3.1.3 Data Provider Integration

To return all matching datasets available at the different data providers their metadata query services must be integrated with the central MQS. Generally, queries are simply forwarded to promising data providers, their results are aggregated, and returned back to the user. Fig. 3.1 presents the architecture of the MQS and connection to the data providers.

According to the interface selection described above, STAC-API queries are submitted towards the MQS and the results are returned as list of STAC items. For a data provider, two options are available to interconnect their query interface with the central MQS:

- Expose the described interfaces (STAC/STAC-API) their site as illustrated by the case of Data Provider 4 in Fig. 3.1.

- Have two additional mapping steps implemented in the backend of the MQS – one from STAC-API to the data provider's query interface and one from the data provider's return format to a list of STAC items. The mapping will be implemented as *Mapper*-microservice in the MQS, and deployed on the central infrastructure is illustrated Fig. 3.1 in the cases of Data Providers 1–3. Different implementations of the *Mapper* will be required by different data providers as they provide either completely different query interfaces (CSW, OpenSearch, OData) or at least different implementations and queryables.

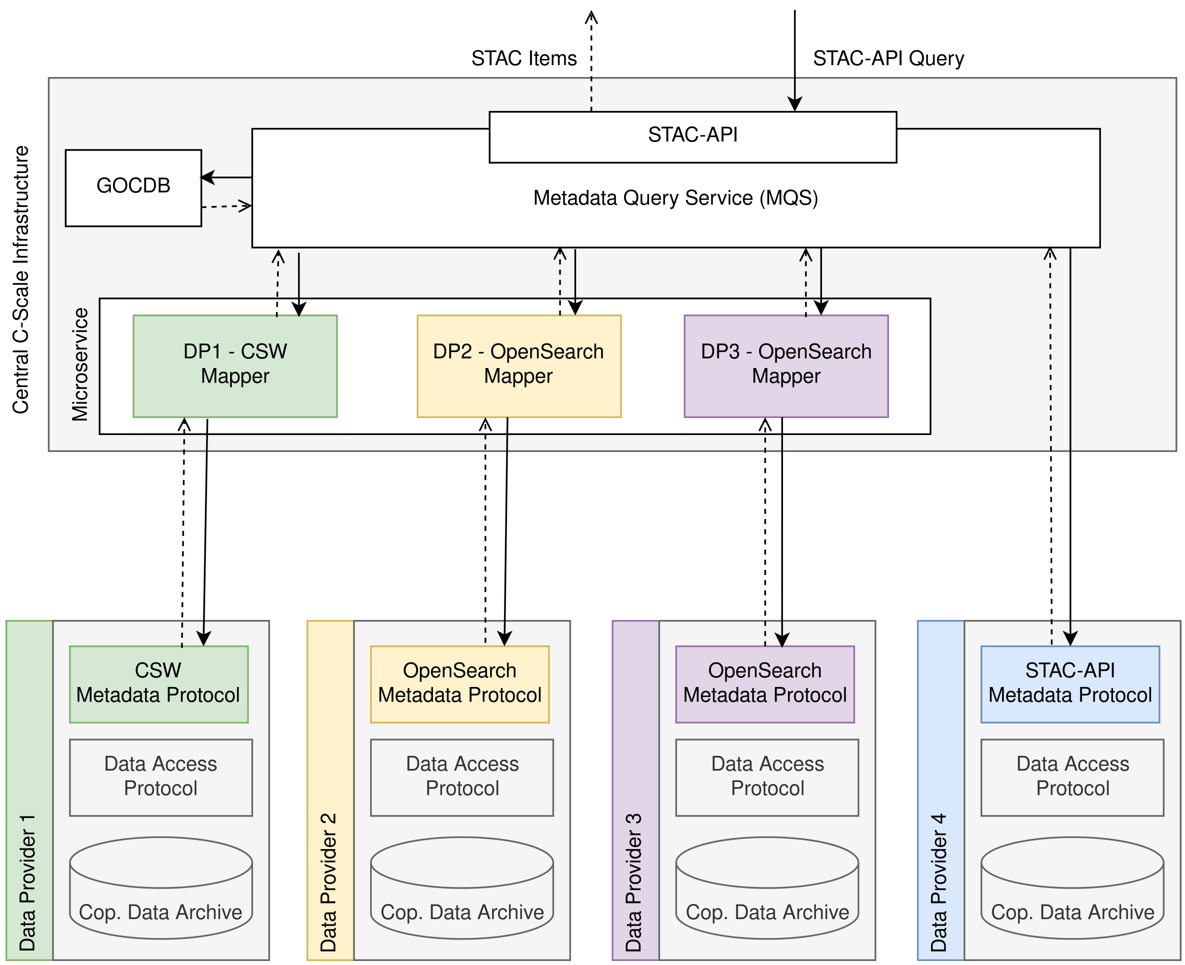

Figure 3.1: Metadata Query Service Architecture

3.1.3.1 Site Responsiveness and Timeouts

As the focus of the central MQS is to retrieve matching datasets from all federation members, it generally waits for all data providers to answer the request. Planned downtimes of a single data provider can be accessed via the Federation Catalogue. Those data providers are then skipped within the defined time frame. To account for unexpected downtimes of data providers a timeout is used. Data providers not answering within the given time frame will be skipped for this query. If a data provider is not able to answer a number of consecutive queries it will be temporarily "suspended" and only considered for incoming queries if it can answer before all other data provider returned their results. The exact duration of the timeout will be based on later evaluation. It must allow data providers to answer complex queries while assuring acceptable performance of the whole system even if some data providers are inaccessible. Indeed, an experimental evaluation of different partners' responsiveness facing various types of queries will emerge as an interesting result of the integration effort.

To give the user additional flexibility, it will be also possible for the user to specify the timeout value explicitly as a part of their query, allowing for slower queries but potentially more complete result lists.

3.1.3.2 Pagination

Returning extremely long result lists is problematic and data providers typically address that with pagination, which must be reflected by the MQS. There are two options, identified as *most responsive* and *most complete*, respectively.

The *most responsive* option performs pagination by always returning a fixed number of datasets per data provider and page, where the fixed number may vary between data providers (based on the default or maximum page size at each provider). This approach is illustrated in Fig. 3.2, where the following objects are:

- *MQS* is the *Metadata Query Service*, a central component discussed in this Section.
- *Resource Catalogue* is the GOCDB, a central service available in EOSC through EGI.
- *DP 1* through *3* are examples of Data Providers who hold actual EO product metadata and can answer queries based on their content.

With this strategy, global sorting of datasets available at different data providers is not possible. As each data provider implements sorting algorithms, datasets can be sorted per data provider, e.g., in descending order by acquisition datetime.

If global sorting is required, pagination becomes problematic. For instance, to identify the second 100 datasets, the first 100 must be known as they may be arbitrarily distributed across different data providers. The proposed flow is visualized in Fig. 3.3, with individual object representing components already explained for Fig. 3.2. Initially the first 100 datasets per data provider – sorted – are retrieved. Then global sorting starts. Once nearly all datasets of one data provider are globally sorted, the next 100 datasets are requested from this data provider. In Fig. 3.3 this first happens for data provider 3, then for data provider 2. When 400 datasets are sorted, dataset 300 to 400 are returned. This is an expensive task especially for latter datasets in large result sets. One option to mitigate this issue would

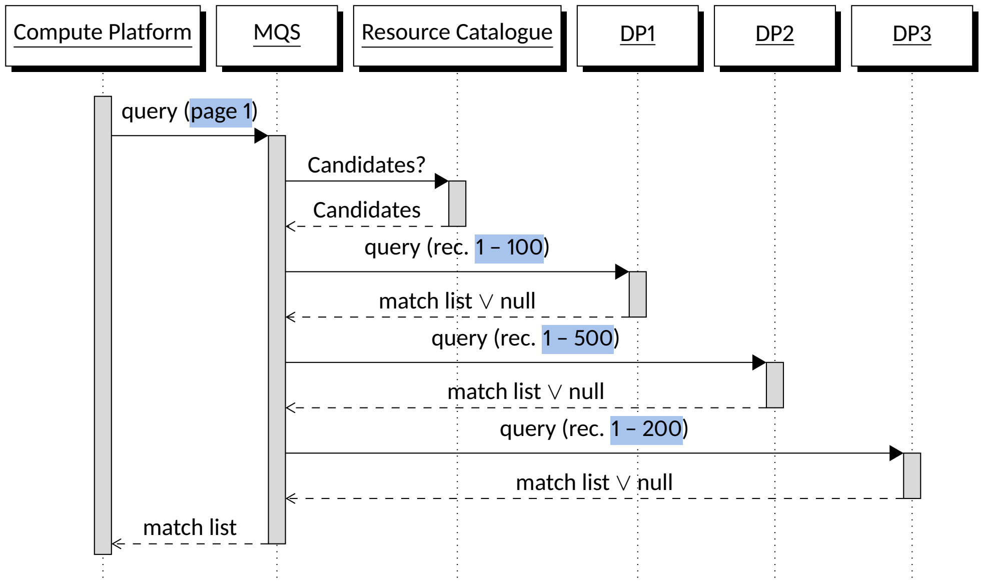

Figure 3.2: Pagination with fixed number of records per data provider.

Attribute	Mutable	Value -- Description
`DatasetsServed`	No	ISO 19115-based description of datasets served from the given endpoint.
`DatasetsReceived`	No	ISO 19115-based description of datasets received at the given endpoint if produced by C-SCALE.

Table 3.2: Custom attributes to extend GOCDB endpoint records specifically for the needs of the C-SCALE Data federation.

be to get, sort and return all matching datasets at once, e.g., leveraging streaming. Depending on requirements expressed by C-SCALE use cases, the most suitable of these approaches will be selected.

3.2 Federation Catalogue

An existing EGI service available in EOSC, the Grid Configuration Database (GOCDB) [R16] will be used to implement the federation catalogue. GOCDB is used across EGI and EOSC to catalogue participating sites and services, including specific service endpoints. This makes it a suitable choice for cataloguing EO data providers within the same ecosystem.

GOCDB supports the use of custom attributes. For the purpose of a site's inclusion into the C-SCALE Data federation, custom attributes outlined in Table 3.2 will be defined and set accordingly.

The GOCDB is not going to require any extension with respect to C-SCALE. The service will be used in its present EOSC-wide deployment. Default attributes will be used where appropriate, and custom attributes will carry the rest of metadata characterizing each given service endpoint within the federation. It is important to clarify here that metadata kept in GOCDB, i.e. the Federation Catalogue, only describe the characteristics of the given site. They do not in any way refer to actual EO products

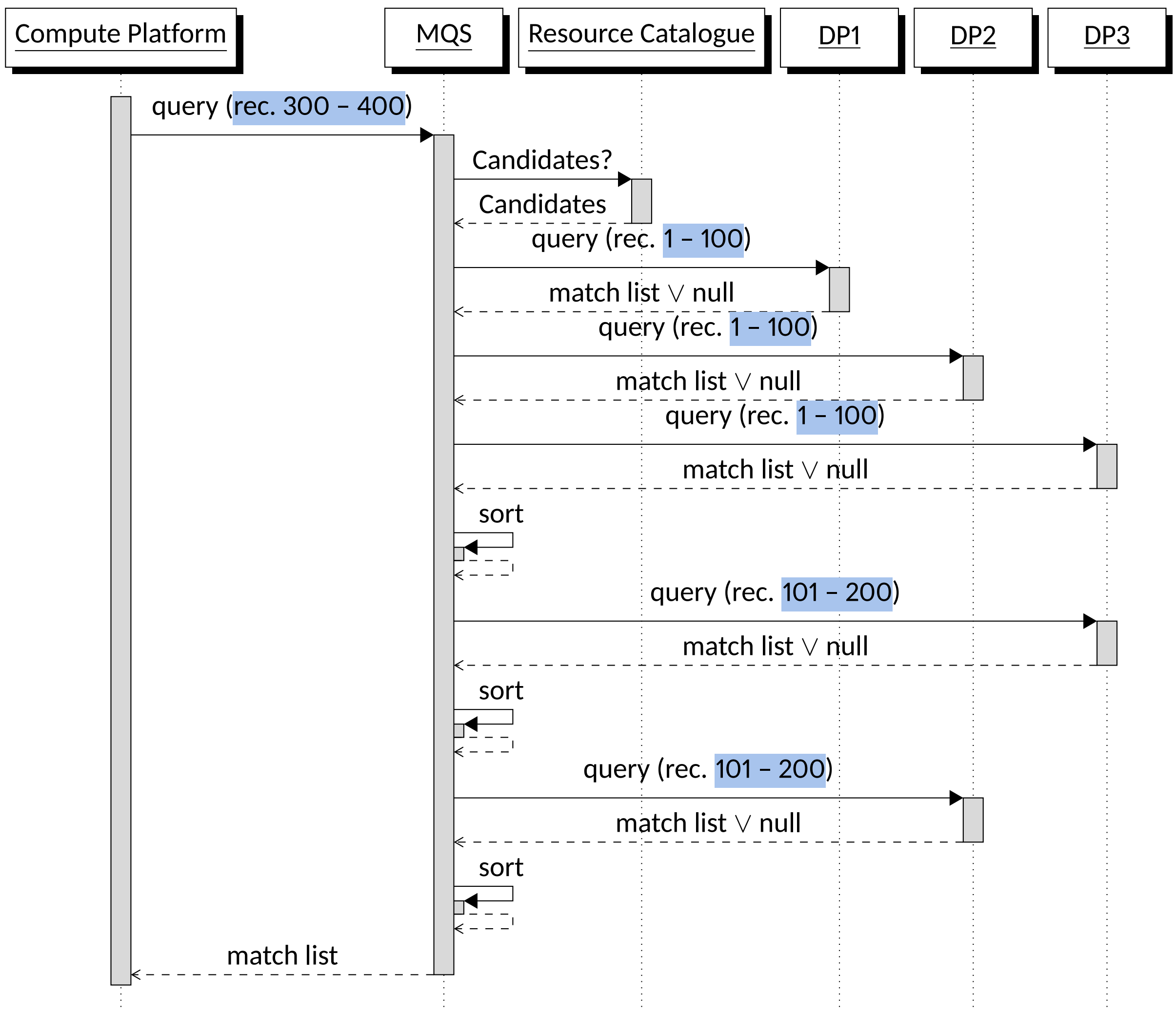

Figure 3.3: Pagination with global sorting.

stored in there since it is one of the defining features of C-SCALE that product metadata are already stored in their respective sites and will not be duplicated elsewhere.

For other types of sites in EOSC (HTC/HPC sites, cloud sites, . . .), registration in GOCDB is done and maintained by respective site administrators. This will also be the case for C-SCALE data sites. The information contained therein in semi-permanent and requires updating only on major configuration changes like change of endpoint addresses, type of services rendered, or change of focus or policies in EO product redistribution. Specifically in the context of C-SCALE, such an update would be required if the site decided, e.g., to extend the period for which the data are held, or the geographic area served.

3.3 Authenticated Access to EO Products

Data to be shared through the C-SCALE data federation are mostly open-access Earth observation products, and the authentication of users accessing said products across the federation is motivated primarily by the need to keep track of usage of the data by users representing different communities and different scientific and professional disciplines, although the option to restrict access to certain products in justified cases is also there. A federation-wide user authentication is a topic addressed already in EOSC, which provides adequate centralized services, so that C-SCALE needs to deal primarily with the integration of local services into the authentication infrastructure, not with the central services themselves.

3.3.1 OIDC and EOSC AAI

C-SCALE data providers use EOSC AAI for user authentication and authorization. EOSC AAI [R17], following a design outlined in Figure 3.4, is an infrastructure which brings together identity providers (IdPs), the EOSC service providers (SPs) and intermediary identity management proxies, such as EGI Check-in.

To authenticate users, C-SCALE sites set up OpenID Connect (OIDC) authentication with EGI Check-in, which is a proxy service that operates as a central hub to connect federated Identity Providers (IdPs) with EGI service providers (SPs). EGI Check-in follows the AARC blueprint architecture shown in Figure 3.5.

OpenID Connect authentication allows service providers to use existing identities – users log in using their identity provider. Thanks to EGI Check-in, it is enough to connect to a single OIDC provider in order to bring users from numerous IdPs.

EGI Check-in integrates with Perun to provide advanced identity management – user registration and approval, group-based authorization etc. In order to use Perun with EGI Check-in, a virtual organization (VO) is created and set up. EGI Check-in then uses data from Perun for authorization and attribute release – see Figure 3.6.

3.3.2 Mapping to Local Accounts

For the purpose of record keeping, and/or even optional authorization (Subsection 3.3.3), a local account shall be kept for users authenticating with OIDC as members of the federation. A survey among partners in data federation shows that the most important reason for maintaining local accounts is the regular generation of aggregate usage statistics.

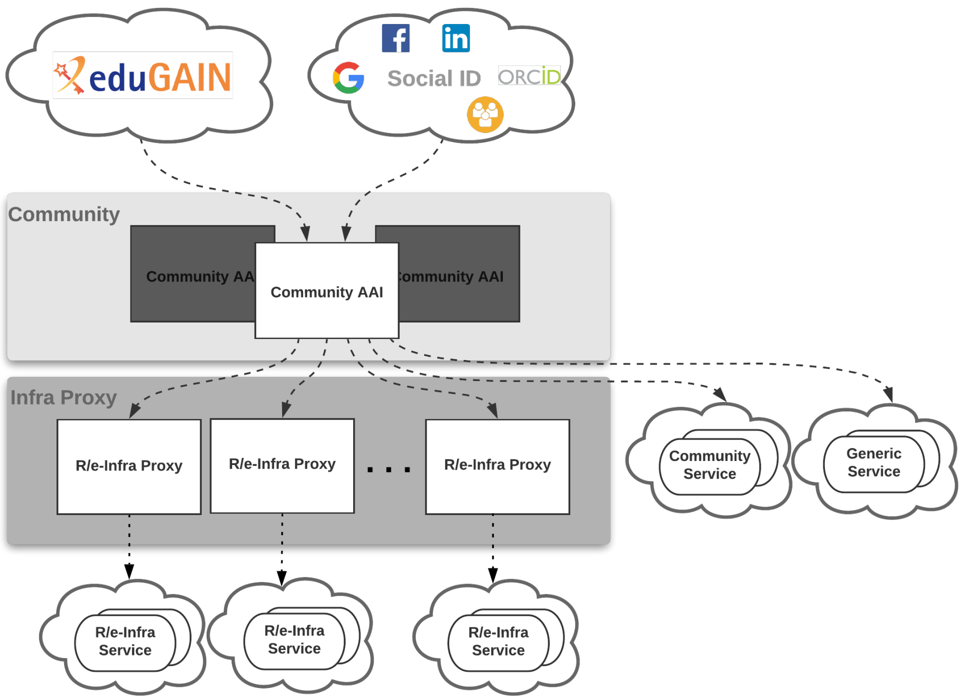

Figure 3.4: High-level view of the EOSC AAI

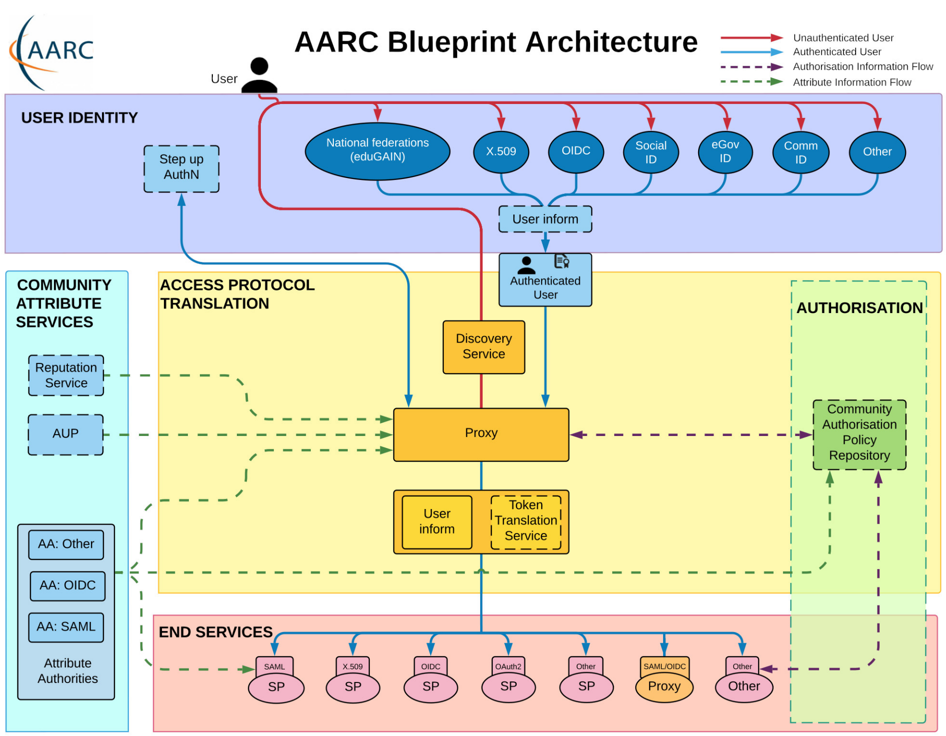

Figure 3.5: AARC blueprint architecture

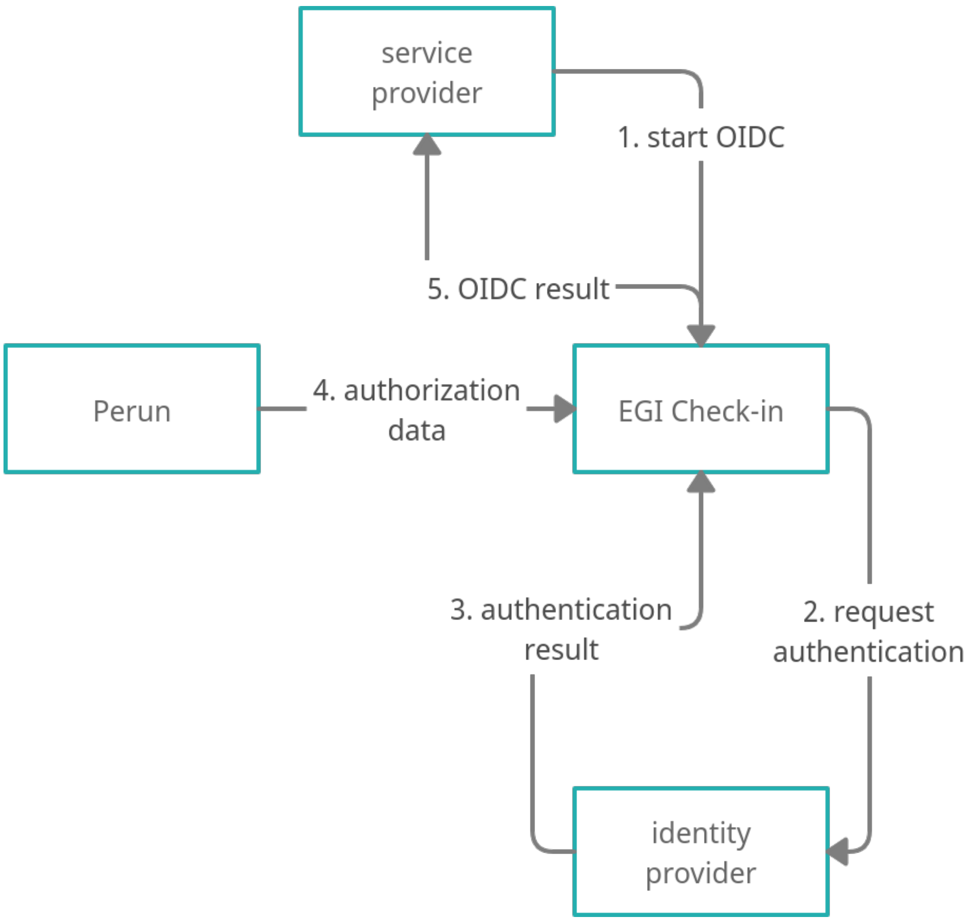

Figure 3.6: OIDC authentication with EGI Check-in and Perun

There are two possible ways of provisioning such an account, envisioned for C-SCALE, and each individual data provider site in C-SCALE will rely on one of them:

1. *On the fly*, wherein the required local account is created on the occasion the user accesses the given storage service for the first time. That is, if a user is successfully authenticated but no local record is found, it is treated as first-time access and the account is created for them. This is the *default* approach and federation members are encouraged to implement that option.

2. *Beforehand*, wherein the Perun IDP is used to manage local accounts at sites. Perun offers a feature called “Perun Services”, which allows, among others, management of local accounts in remote systems based on user identity life cycle managed by Perun. That is, e.g., whenever a user is accepted as a member of a given VO, Perun can pro-actively contact remote sites and create the required local records for that user.

Both methods may result in multiple local accounts being created for a single user, if they access the service with different credentials, e.g., if they are members of different VOs. This is considered correct, since each such local account may carry different attributes and, can in fact, contribute towards different categories in the site’s aggregate statistics. If a user, for example, contacts the site once as a member of a VO dealing with *Marine* disciplines, and on another occasion as a member of an *Atmosphere*-oriented community, it is only right to view the usage as separate in the site’s record.

3.3.3 Optional Authorization

Authorization (i.e., permission to access certain products targeted at specific users based on their identity) is not going to be addressed federation-wide. This is primarily due to the fact that no partners in the Data federation are presently applying authorization restrictions on EO data access, and the functionality is not needed. Still, ways to achieve that functionality have been considered in case it became necessary.

Similar to other models of resource consumption in EOSC, such as IaaS cloud resource access, authorization rules would be applied locally on resource sites. There would be bilateral agreements between resource providers and consumers, which would open specific datasets to selected users. The authorisation decision to either allow or reject access to a given product would be made locally based on the current user’s federated identity.

To aid in this, the MQS would also be extended to adequately mark products with restricted access and what kind of authorization is required to access them. Since there is no tangible use case to require this, however, such extension is deemed optional and no specific design for its implementation is in place.

3.4 Value Added Product Redistribution

While the primary purpose of the data federation consists in redistribution of lower-level Earth observation data towards analytic workflows, it can also be used to facilitate *uploads* of products resulting from said analysis, provided they take the form of higher-level products that can be adequately catalogued and redistributed. Unlike the *download scenario*, wherein every user should be able to download from every provider in a standardized manner, the upload scenario will typically be based on a bilateral understanding between the given producer (user) and the given receiver/redistributor (site), who is interested in redistributing higher-level products generated by the analysis. Yet there is still a possibility to open a site for uploads of all products of each type, should the operator so wish.

3.4.1 Negotiated Upload

It is the preferred approach to enter a bilateral agreement between the community producing higher-level products, and the site who wishes to redistribute their results. The use of federation technology used elsewhere in C-SCALE is encouraged: OIDC for authentication, HTTPs for data transfer. Yet it is possible to negotiate a different authentication or upload method. It is likewise possible to agree on any arbitrary format of metadata for the uploaded products. It is up to the two parties to set up a workflow that results in products finally appearing the site's catalogue.

3.4.2 Invited Upload

The other option is based on a unilateral invitation or declaration of interest. In that case, the site will declare its interest in receiving products by specifying:

- The *type* of data sought, with an ISO 19115 specification
- The upload *endpoint*, i.e., an URL where the uploads will be accepted
- The format of data to be uploaded, such as *SAFE*.

Needless to say, in this invited scenario it is much more complicated to vouch for the provenance of the data. Therefore, the negotiated scenario is preferred. Whether any of those approaches is actually put in use within the timeframe of C-SCALE will depend on the needs of supported use cases, but the technology and formats are available to support for both options.

3.5 Additional Services Identified through Community Engagement

Early in the design phase of C-SCALE, additional centralized services were identified through community engagement. These are optional services that could bring added value to users of the federation, but their actual implementation will have to be considered with respect to development capacity available.

3.5.1 Network Distance Mesh

It is a recurring requirement to maximize data transfer speeds and schedule processing such that it takes place as close to the data as possible. In C-SCALE, this can actually be achieved in two ways:

1. Deploying the analytics platform in a site that is closest to the data.
2. Choosing to download data from a location that is closest to the compute site.

The original proposal for C-SCALE plans to cover only extreme cases in terms of site prioritization: favour local (same site) resources, and/or avoid sites with arbitrarily low priority (that is sites with universally poor performance, such as ESA SciHub). It would be, however, possible to develop a tool that would actively measure the current data transfer performance across C-SCALE (both the data and compute federation) and make much more reliable choice of site with respect of performance.

This development falls under the category of additional goals identified through user community engagement, and its actual implementation will have to be considered.

3.5.2 Query Cap Service

The C-SCALE data federation is being designed as a core of a future global archive of Earth observation data. As such, it is *not* expected to hold a copy of every EO product produced so far. Certainly not in the early phases when native C-SCALE use cases and early adopters need to be supported. Hence comes the requirement for a service that could report what portion of products matching a given query *is* actually available from federation sites.

This requirement has been expressed specifically in relation to Sentinel products, which makes the design simpler as there is an authoritative, almost complete list of metadata regarding Sentinel products: The ESA SciHub. It makes the required information available in two ways: its query interfaces, and its digest of incoming products. Both could be used to implement a service that would estimate the cap to any query, i.e., compile a list of products that should appear among the query results, and compare it to the list of products actually offered by partners in the federation.

This development falls under the category of additional goals identified through user community engagement, and its actual implementation will have to be considered.

4 Conclusion

The C-SCALE data federation aims at making Earth observation data providers under EOSC findable, their metadata databases searchable, and their product storage accessible. It strives to set up a layer of order and standardization on top of the heterogeneous ecosystem of partnering sites, which comprise Sentinel Collaborative Ground Segment sites, DIAS installations, and other big players in Earth observation.

The design of the federation and its supporting services is *realistic* in the sense that it builds on existing technologies, *minimalistic* in the sense that it avoids redundancy and replication of functions, and *conservative* in the sense that it does not disrupt existing usage patterns at participating sites, but only adds a simple layer for improved discovery and seamless access.

Additional services that might be provided by the federation on top of its basic goals are being identified through community engagement, evaluated, and scheduled for implementation either within the project, or later on.

References

No	Description/Link
R1	Google Earth Engine https://earthengine.google.com/
R2	ArcGIS https://www.esri.com/en-us/arcgis/about-arcgis/overview
R3	OpenEO https://openeo.org/
R4	European Open Science Cloud – EOSC https://eosc-portal.eu/
R5	Copernicus Open Access Hub, also known as the ESA SciHub https://scihub.copernicus.eu/
R6	Data Hub Software https://sentineldatahub.github.io/DataHubSystem/
R7	OData https://scihub.copernicus.eu/userguide/ODataAPI
R8	OpenSearch https://opensearch.org/
R9	CSW https://www.ogc.org/standards/cat
R10	Intake https://intake.readthedocs.io
R11	STAC-API https://github.com/radiantearth/stac-api-spec/
R12	STAC https://github.com/radiantearth/stac-spec/
R13	STAC Item specification https://api.stacspec.org/v1.0.0-beta.1/item-search/
R14	GeoJSON https://geojson.org/
R15	STAC Tools https://stacspec.org/
R16	GOCDB https://goc.egi.eu/
R17	EOSC AAI https://confluence.egi.eu/display/EOSC/Authentication+and+Authorization+Infrastructure+-+AAI

Appendix 1 – Queryables

Taking inspiration from the OpenSearch specification, which is currently supported – to a different extent – by several partners in the Data Federation, the following is a list of queryables defined by OGC-10-032r8 and OGC 13-026r9, indicating which partner is capable of evaluating which queryable. This is namely to identify queryables that can be, or should be, universally supported across all partners and, as such, supported by the MQS.

Queryable	EODC	CESNET	VITO	CREODIAS	Description	Data type
Geo Extension `https://www.ogc.org/standards/opensearchgeo`						
`Box`	True	True	True	True	Geographic bounding box	The box is defined by "west, south, east, north" coordinates of longitude, latitude, in a EPSG:4326e decimal degrees
`geometry`	True	True	True	True	Geographic area (geometry)	The geometry is defined using the Well Known Text. The Geometry shall be expressed using the EPSG:4326
`uid`	False	True	True	True	Local identifier of the record in the repository context	String
`lat`	False	True (with lon)	True	True	The latitude of a given point	Latitude in decimal degrees in EPSG:4326
`lon`	False	True (with lat)	True	True	The longitude of a given point	Longitude in decimal degrees in EPSG:4326
`radius`	False	False	True	True	A search radius from a lat-lon point	The distance in meters along the Earth's surface.
`relation`	False (always intersects)	False (always intersects)	True (supports intersects, contains and disjoint)	False	Spatial relation to result set	String; One of "intersects", "contains", "disjoint"
`name`	False	True	True (integrates with Google Places API)	Ture	A string describing the location (place name) to perform the search	String
Temporal Extension `https://www.ogc.org/standards/opensearchgeo`						
`start`	True	True	True	True	A string describing the start of the temporal interval to search(bigger or equal to).	Datetime
`end`	True	True	True	True	A string describing the end of the temporal interval to search (smaller or equal to)	Datetime
`relation`	True (allowing additional mapping)	False	True (supports intersects, contains, during and disjoint)	False	A temporal relation to the result set	Character String: One the "intersects", "contains", "during", "disjoint", "equals"
Earth Observation Extension `http://docs.opengeospatial.org/is/13-026r8/13-026r8.html#16`						
`platform`	True	True	True	True	A string with the platform short name (e.g. Sentinel-1)	String
`platformSerialIdentifier`	True	True	True	True	A string with the Platform serial identifier (For Sentinel-1A, this would be 'A')	String
`instrument`	True	True	True, but only supported for collection search	True	A string identifying the instrument (e.g. MERIS, AATSR, ASAR, HRVIR. SAR).	String
`sensorType`	False	False	True, but only supported for collection search	True	A string identifying the sensor type. Suggested values are: OPTICAL, RADAR, ALTIMETRIC, ATMOSPHERIC, LIMB	String
`compositeType`	False	False	True, but only supported for collection search	False	Type of composite product expressed as time period that the composite product covers (e.g. P10D for a 10 day composite)	String
`processingLevel`	True	True	True, but only supported for collection search	True	A string identifying the processing level applied to the entry	String
`orbitType`	False	False	False	False	A string identifying the platform orbit type (e.g. LEO, GEO)	String
`spectralRange`	False	False	False	False	A string identifying the sensor spectral range (e.g. INFRARED, NEAR-INFRARED, UV, VISIBLE)	String
`wavelength`	False	False	False	False	A number, set or interval requesting the sensor wavelengths in nanometers.	Integer
`hasSecurityConstraints`	False	False	False	False	A string informing if the resource has any security constraints. Possible values: TRUE, FALSE	String

dissemination	False	False	False (we do support an accessedFrom parameter to distinguish between HTTP, S3, ... access methods)	False	A string identifying the dissemination method (e.g. EUMETCast, EUMETCast-Europe, DataCentre)	String
title	True	True	True	True	A name given to the resource	String
topicCategory	True	False	True, but only supported for collection search	True	Main theme(s) of the dataset (here you can find a list of INSPIRE conform topic categories)	String
keyword	True	False	True, but only supported for collection search	True	Commonly used word(s) or formalised word(s) or phrase(s) used to describe the subject	String
abstract	True	False	True, but only supported for collection search	False	Abstract	String
denominator	False	False	True, but only supported for collection search and '1' is only supported value	False	Level of detail expressed as a scale factor or a ground distance. Here: the number below the line in a vulgar fraction. This can be used instead of distanceValue + distanceUOM to describe the spatial resolution. From INSPIRE def: 'Equivalent scale: level of detail expressed as the scale denominator of a comparable hardcopy map or chart'	Integer
distanceValue	True	False	True, but only supported for collection search	False	Sample ground distance. Here: the distance as decimal value.	Double
distanceUOM	True	False	True, but only supported for collection search	False	Sample ground distance. Here: the name of the unit of measure. CodeList, one of: meter, km,...	String
organisationName	True	True	True, but only supported for collection search	True	A string identifying the name of the organization responsible for the resource	String
organisationRole	True	False	True, but only supported for collection search	False	The function performed by the responsible party	String
publicationDate	True	True	True	True	The date when the resource was issued	Datetime
lineage	True	False	True, but 'all' is only supported value	False	General explanation of the data producer's knowledge about the lineage of a dataset.	String
useLimitation	True	False	True, but only supported for collection search	False	A string identifying informing if the resource has usage limitations	String
accessConstraint	True	False	True, but only supported for collection search	False	Applied to assure the protection of privacy or intellectual property, and any special restrictions or limitations on obtaining the resource	String
otherConstraint	True	False	True, but only supported for collection search and 'None' is only supported value	False	Other restrictions and legal prerequisites for accessing and using the resource or metadata.	String
classification	False	False	True	False	Name of the handling restrictions on the resource or metadata	String
language	True	False	False	False	Language of the intellectual content of the metadata record	String
specificationTitle	True	True	True, but only supported for collection search	True	Title of the specification The three specification properties need to correspond to the same, applied specification (e.g. 'Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards interoperability of spatial data sets and services')	String
specificationDate	True	False	True, but only supported for collection search	False	Reference date of specification (e.g. 2010-12-08)	Datetime
specificationdateType	True	False	True, but only supported for collection search	False	Type reference date of specification (e.g. 'publication')	String
degree	True	False	True, but only supported for collection search and 'null' is only supported value	False	This is the degree of conformity of the resource to the related specification. [i] Possible values: true (if conformant), false (if not conformant), null (if not evaluated)	String
parentIdentifier	True	True	True	True	A string identifying the parent of the entry in a hierarchy of resources	String

`productionStatus`	False	doable	True	True (status - offline/online=to order)	A string identifying the status of the entry (e.g. ARCHIVED, ACQUIRED, CANCELLED)	String
`acquisitionType`	False	True	True	True	Used to distinguish at a high level the appropriateness of the acquisition for "general" use, whether the product is a nominal acquisition, special calibration product or other. Values: NOMINAL, CALIBRATION, OTHER This is used to differentiate e.g. calibration measurements from standard ones. I assume most datasets will only provide acquisitionType=NOMINAL	String
`orbitNumber`	Partly (not supported for all collections)	True	True	True	A number, set or interval requesting the acquisition orbit	Integer
`orbitDirection`	Partly (not supported for all collections)	True	True, but not supported for all collections	True, but not supported for all collections	A string identifying the acquisition orbit direction. Possible values are: ASCENDING, DESCENDING	String
`track`	False	False	False	False	A string identifying the orbit track	String
`frame`	False	False	False	False	A string identifying the orbit frame	String
`swathIdentifier`	False	True	False	False	Swath identifier (e.g. Envisat ASAR has 7 distinct swaths (I1,I2,I3...I7) that correspond to precise incidence angles for the sensor). Value list can be retrieved with codeSpace.	String
`cloudCover`	True	True	True (number, set or interval), but not supported for all collections	True	A number, set or interval of the cloud cover % (0-100).	Integer
`snowCover`	False	False	True (number, set or interval), but not supported for all collections	True	A number, set or interval of the snow cover % (0-100).	Integer
`lowestLocation`	False	False	False	False	A number, set or interval of the bottom height of datalayer (in meters).	Double
`highestLocation`	False	False	False	False	A number, set or interval of the top height of datalayer (in meters).	Double
`productVersion`	False	False	True	True (processingBaseline)	A string identifying the version of the Product	String
`productQualityStatus`	False	False	False	False	This optional field must be provided if the product passed a quality check. Possible values: NOMINAL and DEGRADED	String
`productQuality DegradationTag`	False	False	False	False	Keywords giving information on the degradations affecting the product. Possible values are mission specific and can be freely define (e.g. "RADIOMETRY", "MISSING_LINES")	String
`processorName`	True	True	True	False	A string identifying the processor software name (e.g. FORCE)	String
`processingCenter`	True	True	True	False	A string identifying the processing center (e.g. PDHS-E, PDHS-K, DPA, F-ACRI)	String
`creationDate`	True	True	True (in our catalogue this queryable is publicationDate)	True (PublicationDate)	The date when the metadata item was ingested for the first time (i.e. inserted) in the catalogue	Datetime
`modificationDate`	True	True (Bumps Creation)	True	True (updated)	The date when the metadata item was last modified (i.e. updated) in the catalogue.	Datetime
`processingDate`	True	True (Bumps Creation)	True	True (for acquisition dates)	A date interval requesting entries processed within a given time interval	Datetime
`sensorMode`	True	True	False	True	A string identifying the sensor mode.	String
`archivingCenter`	True	False	True	False	A string identifying the archiving center.	String
`processingMode`	True	True	False	True	Processing mode. Often referred to as Real Time, Near Real Time etc.	String