# IS-ENES3 Deliverable D5.3

## Architecture Design Plans

*Reporting period: 01/07/2022 – 31/12/2022*

Authors: Philip Kershaw (UKRI)
Reviewers: Paola Nassisi (CMCC), Stephan Kindermann (DKRZ)
Release date: 20/12/2022

## ABSTRACT

The goal of this document is to outline the future plans for the software architecture for the ENES Climate Data Infrastructure, building on the work done in D10.1 (July 2020) taking into account changes introduced through the implementation of the ESGF Future Architecture and the integration of these with partners through work planned as part of Amendment 3 to the project.

| **Revision Table** | | | |
|---|---|---|---|
| **Version** | **Date** | **Name** | **Comments** |
| Release for review | 03/10/2022 | Philip Kershaw | |
| 1.0 | 18/11/2022 | Philip Kershaw | Release for issue |
| **Dissemination Level** | | | |
| PU | Public | | X |
| CO | Confidential, only for the partners of the IS-ENES3 project | | |

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

is-enes
INFRASTRUCTURE FOR THE EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

# Table of contents

## Executive Summary

Over the lifetime of the IS-ENES3 project, a major initiative has been undertaken to re-architect and re-implement the software stack for ESGF (Earth System Grid Federation). The ESGF system underpins all of the ENES CDI and therefore any change fundamentally impacts on the interfaces and functionality offered by the latter. This report describes the plans to integrate the new changes including upgrades to ESGF nodes at existing sites and work on an integration testbed to demonstrate the new capabilities. This utilises the Climate4Impact system, a web-based data analysis platform which acts as a "thick" client to ESGF services. In addition to ESGF functionality, this testbed will also exploit new data sub-setting services developed as part of work to support the Copernicus Climate Data Store. This testbed activity has been proposed and approved as part of the third amendment to the project.

# 1. References

REF1 D10.1 Architectural document of the ENES CDI software stack, 31st July 2020, https://doi.org/10.5281/zenodo.4309891

REF2 Kershaw, Philip, Abdulla, Ghaleb, Ames, Sasha, & Evans, Ben. (2020). ESGF Future Architecture Report (1.1). Zenodo. https://doi.org/10.5281/zenodo.3928223

REF3 Kershaw, P., Halsall, K., Lawrence, B.N., Bennett, V., Donegan, S., Iwi, A., Juckes, M., Pechorro, E., Petrie, R., Singleton, J., Stephens, A., Waterfall, A., Wilson, A. and Wood, A., 2020. Developing an Open Data Portal for the ESA Climate Change Initiative. *Data Science Journal*, 19(1), p.16. DOI: http://doi.org/10.5334/dsj-2020-016

# 2. Introduction

IS-ENES3 objectives state that "IS-ENES3 will support the exploitation of model data by the Earth system science community, the climate change impact community and the climate service community.". This includes work to "Maintain and develop the European component of the global Earth System Grid Federation with the aim of supporting the 6th phase of the Coupled Model Intercomparison Project (CMIP6), expected to require storing and supporting access to tens of petabytes of data", and "Invest in the operation and development of the Climate4Impact platform and the underlying services to enable customised access to data, documentation, and information about model evaluation to the climate impact community as well as climate service businesses and consultancies."

The existing description of work included the maintenance and update of ESGF nodes amongst the project partners. However, since the commencement of the project, there has been an initiative - the ESGF Future Architecture - to fundamentally re-engineer and modernise the ESGF software. This involves the adoption of completely new technologies which have not been used before with many of the project partners. These include Docker containers and Kubernetes. There is therefore a requirement for additional effort to support partners in the adoption of this new technology. CEDA, who have developed and piloted the new infrastructure will assist other partners in making their deployments.

In a collaboration between the leaders of WP5, WP7 and WP10, a work plan and roadmap were discussed and agreed in order to see how the ESGF infrastructure deployed amongst the project partners could be updated with the new system and demonstrate and test new capabilities in an

integration pilot with the Climate4Impact platform. These proposals were put forward as part of the 3rd Amendment to the project.

In doing so it is hoped to progress ESGF and the ENES CDI from a model of federated *portals* to federated *platforms*. This will entail a move beyond the traditional model of search and download for ESGF towards interactive analysis of data in-situ with where it is hosted.

## 2.1. ENES Climate Data Infrastructure (CDI)

A key goal of the IS-ENES3 project is to evolve and develop a software infrastructure to support dissemination and analysis of the outputs from CORDEX and CMIP6. This infrastructure is described in project deliverable D10.1 (REF1).

## 2.2. ESGF Background and the Future Architecture

The ENES community together plays a major role in maintaining and operating Earth System Grid Federation's software infrastructure together with our international partners. In 2019 a major initiative was started to re-architect and re-engineer the system (REF2). This has been led and co-ordinated by the ESGF Executive Committee which has overall technical oversight and was driven by a recognition of the need to update and modernise it considering that it has largely remained unchanged over the ten years of its operation.

Following a meeting of the technical representatives of the partners in ESGF, a report was compiled summarising the findings and setting out a roadmap of proposed development activities. The major findings were:

i) to adopt a more modular approach to service development and maintenance, taking advantage of modern cloud and virtualisation technologies such as containers,

ii) to adopt community standards to enable ESGF services to better interoperate with other similar systems,

iii) to centralise core federation functionality such as search and identity management and finally improve the integration of services capabilities, and

iv) to replace and improve data publishing and user frontend services.

Development activity initially focussed on re-engineering of the system to use Docker containers and Kubernetes for operation and deployment. This work was European led with US partners LLNL contributing with a refactor of the ESGF publishing system. Successful deployments of

this new system have been made by GFDL on Amazon Web Services and by CEDA on the JASMIN[1] infrastructure.

In a parallel activity, the development of a completely new search system has been undertaken by CEDA. In close collaboration with CEDA, DKRZ is also exploring this and has set up a new search system instance[2]. This adopts the popular STAC[3] search specification which originates from the Earth observation community but has been adapted for use with Earth system model data.

### 2.3. Development of Compute Services for C3S

As part of a contract for the Copernicus Climate Data Store, IS-ENES3 project partners, CEDA and DKRZ, developed "Data Reduction" services to deliver user-defined subsets of climate simulation data hosted on ESGF Data Nodes. The cost (in terms of time and storage) of bringing large amounts of this data to a client is very significant when the workflows and requirements of each user are considered. A far more efficient approach is to allow users to define and request a subset of the data from a processing service. This "data reduction" service can subset the model data before returning only the required data array back to the end-user. Services have been developed based on the OGC WPS standard[4] and implemented using DKRZ's Birdhouse[5] framework.

## 3. Design Goals Towards an Architecture Supporting In-Situ Data Access and Analysis

Over the course of the lifetime of ESGF, there has been an observable shift in paradigm in the Earth sciences community away from the traditional model of search and download, as supported by the existing ESGF system, towards models of in-situ data access and analysis. Examples include the storage of CMIP datasets alongside computing resources in facilities such as JASMIN, DKRZ, IPSL and CMCC's ENES Climate Analytics Service[6] and the use of public cloud, collocating data with analysis compute e.g. Pangeo[7].

---

[1] https://jasmin.ac.uk
[2] DKRZ STAC web service endpoint - https://stac.dkrz.de
[3] Spatio Temporal Asset Catalog (STAC), https://stacspec.org/
[4] https://opengeospatial.github.io/e-learning/wps/text/basic-main.html
[5] https://birdhouse.readthedocs.io
[6] https://ecaslab.cmcc.it/
[7] https://pangeo.io

This section outlines high-level design principles and functional areas to focus on in order to develop an architecture and accompanying implementation which better supports a model of in-situ data access and analysis for ESGF and the ENES CDI.

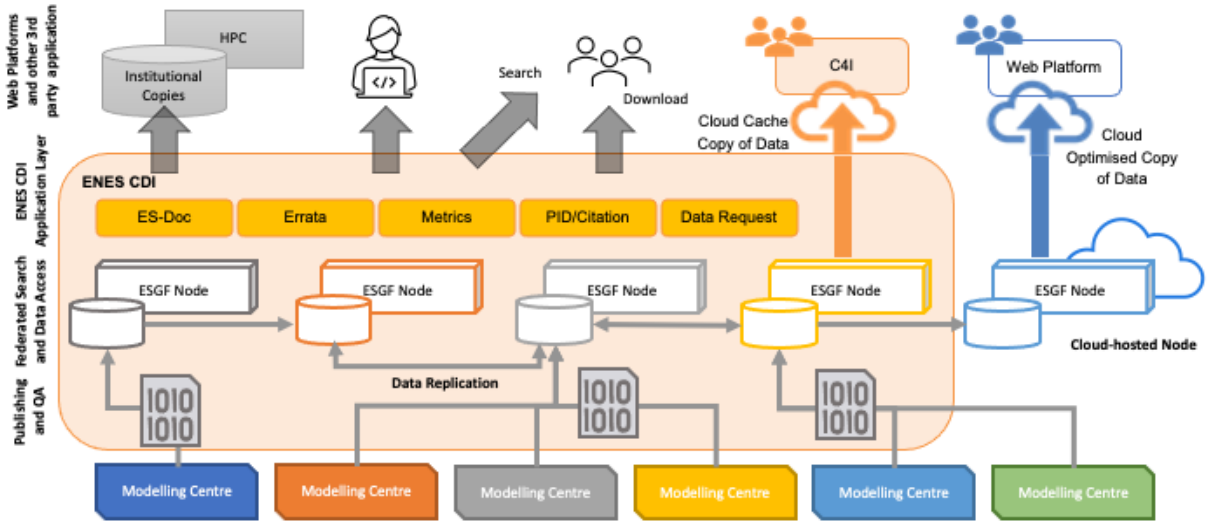



*Figure 3-1: ESGF, ENES CDI and Downstream Platforms and Services*

## 3.1. Modular Cloud-Native Deployment

The goal here is to make the ESGF software as easy to deploy as possible on a variety of hosting environments - on-premise and public cloud and thus supporting a so-called cloud-native deployment model. This provides flexibility to innovate with new hybrid deployment scenarios. Such a mixed model can take best advantage of the respective hosting options. An example of where this approach may be beneficial is the proposed centralised deployment for search and identity services set out in the Future Architecture Report. The report recommended a container-based deployment model as part of its findings. This allows cloud-agnostic deployment since all major cloud providers support the Kubernetes container orchestration system and equally data providers in the federation can select Kubernetes or the simpler Docker Compose arrangement for hosting on-premise.

## 3.2. Application of Community based Standards

Two of the major functional areas which define the interfaces for federation are data discovery and identity and access management. These need address two major concerns:

1) the need for a community standard which can be adopted widely in the Earth sciences domain and thus facilitate interoperability between ESGF and other systems and standards in use in the Earth sciences.

2) the elimination of serious security vulnerabilities introduced as a consequence of the use of legacy standards and software implementations beyond their intended operational lifetimes.

The Future Architecture Report proposed the use of established standards OpenID Connect and OAuth 2.0 for identity and access management. For search, at the time of the writing of the report no clear standard had emerged though subsequent work has demonstrated the applicability of STAC. An approach using STAC to provide *static catalogues* has gained momentum by the adoption of the PANGEO community[8] as well as commercial cloud providers (like Amazon, Google and Microsoft). A full implementation of STAC *API* for use with ESGF has been developed. In this case, STAC formatted search metadata is contained in an index searchable via a web service API[9].

## 3.3. Analysis-Ready Data Caches

Since the original development of ESGF, a new web-based platform model has been emerging whereby data access and analysis services are co-located typically on public cloud. Tools such as Jupyter Notebooks provide users with a convenient desktop-style interface for analysis whilst at the same time accessing powerful server-side compute with processing frameworks like Dask. With the popularity of object store and its relative cost-effectiveness with respect to other types of storage available on public cloud, new solutions for serialising and accessing data such as Cloud-Optimized GeoTIFF and Zarr have come to the fore. These are explored in more depth in section 5.3 later in this document. Even so, the relative cost of storage on public cloud when compared with on-premise, particularly for the hosting of large volumes of data is a limiting factor for more large-scale adoption of cloud in these scenarios.

An alternative approach is to generate smaller caches of data on cloud for analysis, copied from their primary source on-premise – for example retrieved from an ESGF data node. In this way, it

[8] see e.g https://stacindex.org/catalogs/pangeo#/
[9] https://github.com/radiantearth/stac-api-spec

is possible to take advantage of cloud for processing and analysis without incurring the costs from large-volume storage. Another motivation for creating caches is the need to transform data into a specific form suitable for analysis (e.g. chunking data in time dimension to suit time-series queries). Thus, the data is made *analysis-ready* such that it can be analysed efficiently. Software is needed to support data pipelines allowing data subsets to be transferred from primary stores in the federation (data nodes) to smaller caches co-located with cloud-hosted analysis environments.

### 3.4. Standardised Data Reduction Services

The ability to take subsets of data is important for the purposes of increased efficiency and cost effectiveness to reduce the overall volume of data transferred between server and client-side. This is an important prerequisite to support the concept of analysis-ready data caches as described in the previous section. If data and analysis environments are to be co-located on cloud, minimising the amount of data that needs to be cached makes it more cost effective. The API standard which is supported by the ENES community is based on OGC WPS[10], which is also the agreed standard for interoperability with the Copernicus Climate Data Store.

## 4. Implementation roadmap: development activities as part of the 3rd amendment

The implementation of the ESGF Architecture is being conducted in two phases (See Table 1). As part of the 3rd amendment to the project, a series of development activities will be completed in the coming months which will utilise and integrate the future architecture into the ENES CDI.

| | |
|---|---|
| Phase 1 | Cloud-native deployment system complete: ESGF can be deployed with Docker containers using Docker Ansible or Kubernetes for full container orchestration. Deployment complete at CEDA and GFDL (AWS public cloud). |
| | Existing legacy ESG Search system with Apache Solr is retained. |
| | Production deployment possible with open datasets e.g. CMIP6. Test deployment of limited features of new access control system available including OIDC/OAuth 2.0 identity provider. |

[10] Open Geospatial Consortium (OGC) Web Processing Service (WPS): https://www.ogc.org/standards/wps

| | |
|---|---|
| **Phase 2** | Cloud-native deployment system complete: ESGF can be deployed with Docker containers using Docker Ansible or Kubernetes for full container orchestration. Deployment complete at first European sites: DKRZ and LIU. |
| | New ESGF Search system available based on STAC and ElasticSearch. Deployments of search indexes limited to a very limited number of sites in line with the goal of centralising search services. |
| | Production deployment with access control available. This includes a centralised Identity Provider, means to register for access to restricted datasets e.g. CORDEX. Sites may host their own local identity provider and configure datasets for access control. |

*Table 1: ESGF Future Architecture implementation phases*

As of writing, Phase 1 is complete. From Phase 2: a prototype search index is deployed at CEDA and DKRZ populated with some CMIP search metadata. The STAC API is implemented together with Python client bindings. The central identity provider is deployed (https://login.esgf.io/). The registration system for applying to access secured resources is scheduled to be the next component to be deployed.

## 4.1. Integration Pilot with the Climate4Impact Portal

The integration pilot has been set up as a means to integrate the Future Architecture into elements of the ENES CDI.

This references the key design goals outlined in section 3 and brings together the threads of development set out in the introduction. It is centred around the use of the Climate4Impact portal as a consumer and focus for integration for the new developments.

1. Climate4Impact acts as a consumer to an ESGF node at CEDA which uses the new Future Architecture implementation (corresponding to Phase 1 of the Future Architecture development). This meets the following use cases:
    a. Climate4Impact searches and downloads open data (CMIP5/6) from CEDA node. Search is done with the existing legacy search API.
    b. Climate4Impact searches and downloads secured data (CORDEX) from CEDA node. In doing so, it interacts with the new identity and access management system which is based on OpenID Connect and OAuth 2.0.
    c. Climate4Impact utilises a new sub-setting service deployed at DKRZ to allow users to obtain a subset of data for analysis.

2. Deployment of updated release which integrates with the new STAC search system (corresponding to Phase 2 of the Future Architecture development):
    a. Climate4Impact searches and downloads data (CMIP5/6 and CORDEX) from CEDA node. CORDEX data download uses the new access control system whereby users register for access to secured resources at the Central IdP
    b. Search queries are executed with the new STAC search API. Initially integrate the Python search API via a Jupyter Notebook. If resources allow, also integrate search API into web user interface (JavaScript).
    c. Climate4Impact utilises sub-setting services from DKRZ and CEDA to allow users to obtain a subset of data for analysis.

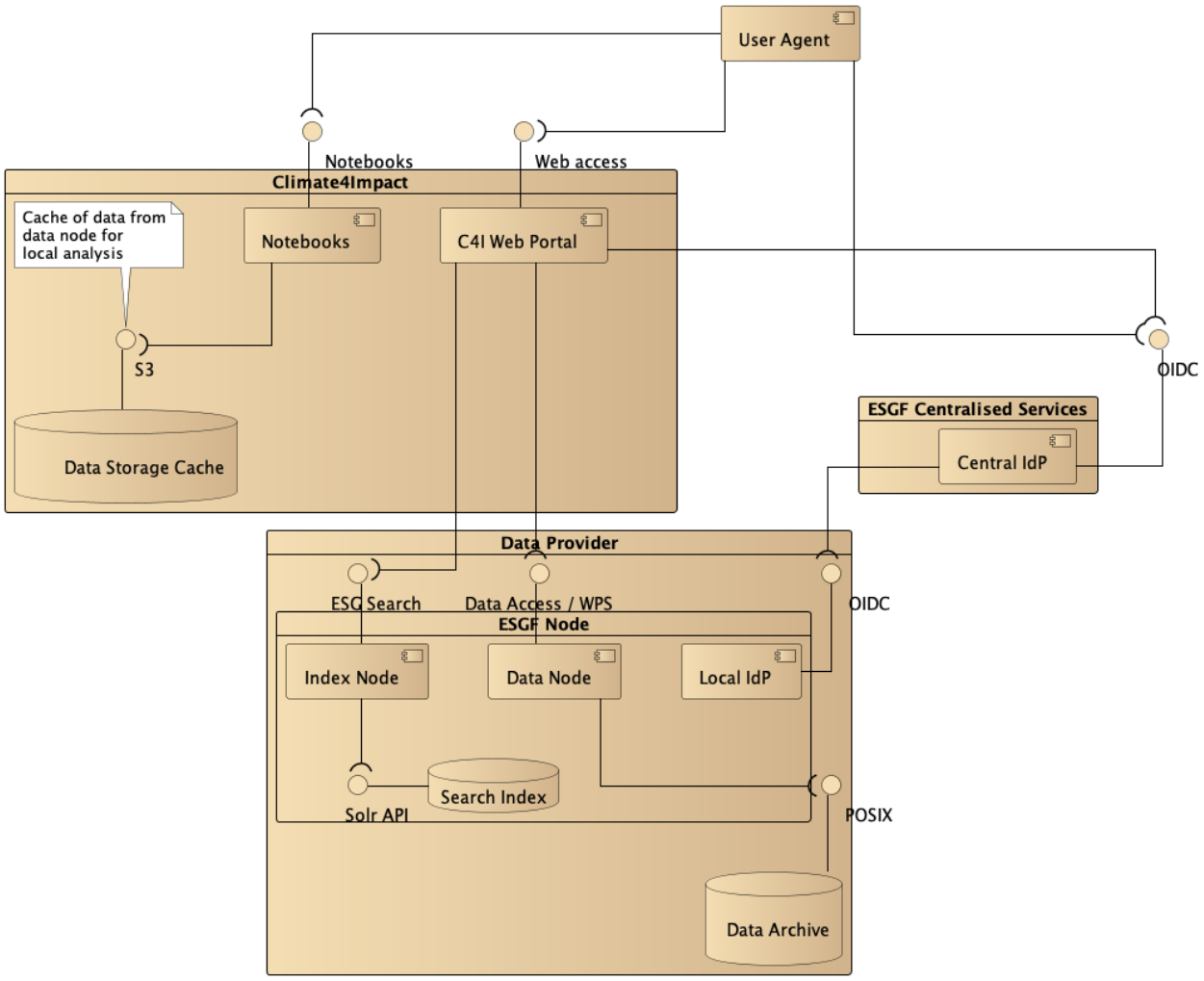



*Figure 4-1: Integration Pilot - Phase 1*

Referring to Figure 4-1, the integration pilot requires as its baseline for Phase 1:

- Climate4Impact: deployed and maintained by KNMI on public cloud
- Data Provider: ESGF Node at CEDA based on the Future Architecture release.
- Data Provider: DKRZ Node. DKRZ will provide the initial WPS based data reduction service.
- Data Provider: CMCC Node. CMCC will provide its CMIP6 data with the WPS data reduction service.
- ESGF Centralised Services:
    - The central IdP is deployed at CEDA but configured with cloud-based load balancing such that it could be deployed at alternative locations or in duplicate in order to increase resilience
    - For Phase 1, the legacy ESGF search service is used based on a deployment at CEDA.

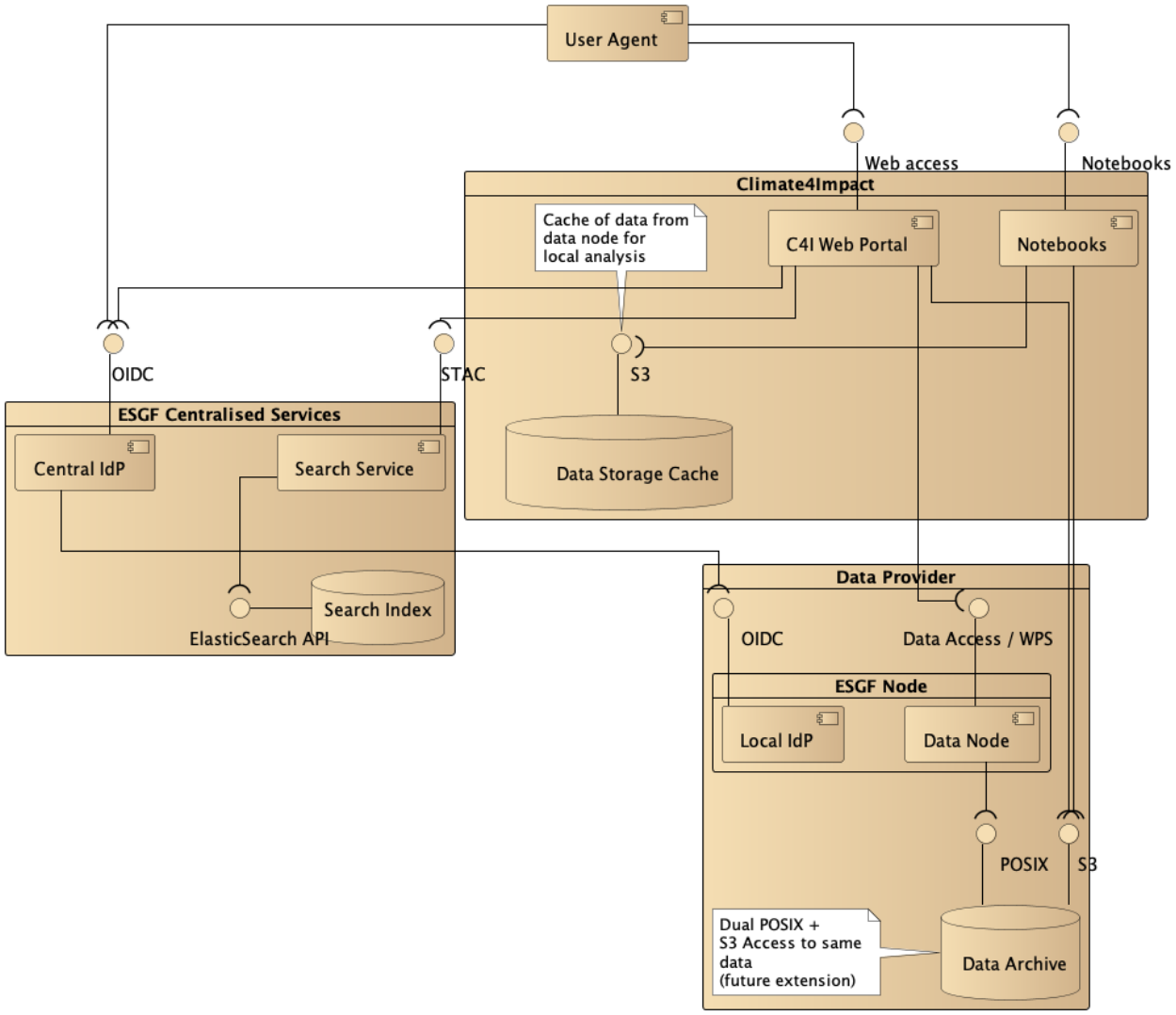



*Figure 4-2: Integration Pilot – Phase 2*

Referring to Figure 4-2, for Phase 2, the new search service based on STAC will be employed. This will be part of the centralised services.

## 4.2. Deploy ESGF future architecture release in production at European sites

This covers the operational deployment at host sites together with support from partners who have already deployed the new system. CEDA have completed the deployment and are organising a programme of work to assist other sites (initially LIU and DKRZ) to upgrade to the new system.

## 4.3. Application of Community Standards for Services

### 4.3.1. Integrate new STAC search API into infrastructure

The new system meets the requirement to apply community standards by the adoption of the STAC specification which has become a de facto standard in the Earth observation community. To fully exploit the new capability that has been developed, it is necessary to integrate the new search API with all the existing software components replacing code that currently uses the legacy ESGF system. These include:

- General Python client support – a STAC client with extensions to support ESGF data to replace esgf-pyclient. This has been implemented by CEDA (https://github.com/cedadev/esgf-stac-client).
- Integrate the ESGF STAC client into the Synda discovery module and Synda UI.
- Climate4Impact - Integrate into web search UI and use Python STAC client (previous bullet) with Climate4Impact Jupyter environment.
- ESGF Data Statistics service - to generate the data usage statistics, download logs are processed, based on information from a Solr instance of a local Index Node. This will be refactored to query the new STAC search interface. Additionally, information on the published data over the federation will be obtained. This will require the deployment of a local file system component exposing the STAC interface and the consequent adaptation of the current processing chains both for published and downloaded data.
- Promote the integration and dissemination of STAC amongst other applications which currently use ESGF Search.

### 4.3.2. Integrate identity and access management infrastructure

Section 5.14 of the ENES Software Architecture (REF1) outlines the overall architecture for the new identity and access management infrastructure, including the use of OpenID Connect (OIDC), OAuth 2.0 and the Central IdP proxy concept (REF2). As part of the integration pilot with the Climate4Impact portal, the use of OIDC has been demonstrated with a test instance of an IdP. The goal for the integration pilot is to show a complete working deployment of the access management system to secure access to CORDEX datasets. This entails:

1) Deployment of the IdP Proxy. This has been completed with the proxy deployed at CEDA with the general ESGF *.esgf.io* domain: https://login.esgf.io. This service enables login to secondary IdPs including CEDA and third-party providers Google, GitHub and ORCID.
2) Deployment of access registration system. This is a web application that enables users to register for access rights to a given secured resource. For the purposes of the pilot, we

want to demonstrate registration for access to CORDEX data. The implementation of the web application is completed and is awaiting deployment.

3) Deploy and configure the PEP (Policy Enforcement Point), PDP (Policy Decision Point) and authorisation policy settings at the CEDA data node to enforce the access policies to restrict access to CORDEX data.
4) Demonstrate flows for authenticated and authorised flows for access to CORDEX data via the Climate4Impact portal and through Python scripting (e.g. via Jupyter notebook service provisioned by the Notebook service).

## 4.4. Integrate New Web Processing Data Reduction Services into Operational ENES Infrastructure

The deployment of data reduction services at European sites (initially CEDA and DKRZ) will enable services like the Climate4Impact Portal to only download the subset of data that they need and store this on its data cache on public cloud. Tasks include:

- Refine and improve provenance information from Rooki[11]
- Integrate any necessary access control into the WPS deployments
- Integrate DKRZ, CEDA and CMCC nodes as operational WPS providing processes (Averaging and Sub-setting) and integrating these services with the Climate4Impact Portal (KNMI, DKRZ, CEDA and CMCC)
- Deploy Rooki recipes at IPSL

[11] https://rooki.readthedocs.io

# 5. Future Roadmap

This section considers ambitions for the development of the CDI architecture covering specific areas that it was not possible to include in the work for the third amendment and broader activities which should be considered for the longer term.

## 5.1. Completing ESGF Future Architecture Roadmap

Looking at the functional areas identified in the future architecture it is possible to give a summary status for each:

| Functional Area | Completed or scheduled to be complete by project end | Future work |
|---|---|---|
| Identity Management | Identity proxy with OIDC and OAuth 2.0<br>Resource registration system<br>Authorisation system | Review arrangements for hosting with other ESGF partners especially DOE ESGF2 project. |
| Data Services | File serving directly with Nginx rather than THREDDS. THREDDS OPeNDAP support continued | *Object store* integration<br>Analysis-Ready caches<br>Interface for access to data in different storage tiers |
| Search | STAC service and client bindings completed and tested as part of integration pilot | Integrate STAC with MetaGrid<br>Review arrangements for hosting with other ESGF partners especially DOE ESGF2 project – possible dual deployments for search hubs (US and European) |
| Metadata Catalogues | Work is predicated on completion of STAC server and client baseline implementations. Baseline includes all the facets that the current ESGF Search API supports. | Support for more sophisticated search by indexing content from ES-Doc. Investigate inclusion of Citation and Errata information in search index.<br>Use of *schema.org* tags to support better indexing by web search engine web crawlers. |

| Functional Area | Completed or scheduled to be complete by project end | Future work |
|---|---|---|
| | | Support for geotemporal search<br>Make associated enhancement to search clients to support the above – to MetaGrid, Climate4Impact Portal, other… |
| Publication, Replication, versioning | STAC write API (Publishing) complete | Support for nodes to publish to centralised search index<br>Publishing supports indexing of content from PID service, ES-Doc and Citation service.<br>Development of vocabulary services to give canonical definition of terms for use in search services and for publishing service to reference and enforce from |
| Compute on data | Deploy WPS data reduction services at CEDA, DKRZ and CMCC and use with integration pilot with Climate4Impact Portal | Wider deployment of WPS Data reduction services in the federation<br>Investigate support for discovery of search services through STAC. |
| Platforms and System Administration | Complete deployment for phase 1 system at three or more European sites | Review arrangements for hosting centralised services – search and identity management - with other ESGF partners especially DOE ESGF2 project. |

*Table 2: Summary status for ESGF Future Architecture work*

## 5.2. Search Services next steps

A key consideration for the ongoing development of the search services is hosting. The ESGF Future Architecture Report recommended the centralisation of search services to simplify the system and reduce the burden on individual node administrators. Practically, it is likely though that more than one instance will be needed to suit the needs of the different project consortia funding ongoing development efforts for ESGF, specifically European partners through the

ENES collaboration and US partners through the new DOE-funded ESGF2 project. One possible solution is to have one instance on each continent that is linked together. There are potentially two variants:

1) Two instances have search content which overlaps with one another, but which is not identical. There could be a common core of datasets whose catalogue records are replicated but also other datasets which are held only by one or the other. Users could access either search service according to their needs.
2) Two instances which mirror each other's content and are served through a common host name. This is achievable with DNS-based load balancing and has been demonstrated already with Copernicus Climate Data store services.

The diagram below in Figure 5-1 illustrates option 2). Note though that if we eliminated the load balancer shown, it would effectively be option 1). Should it be deemed beneficial other index nodes could be added for other centres or regions. The key difference in approach to the traditional ESGF system is that the emphasis would be on much, much fewer search indexes.

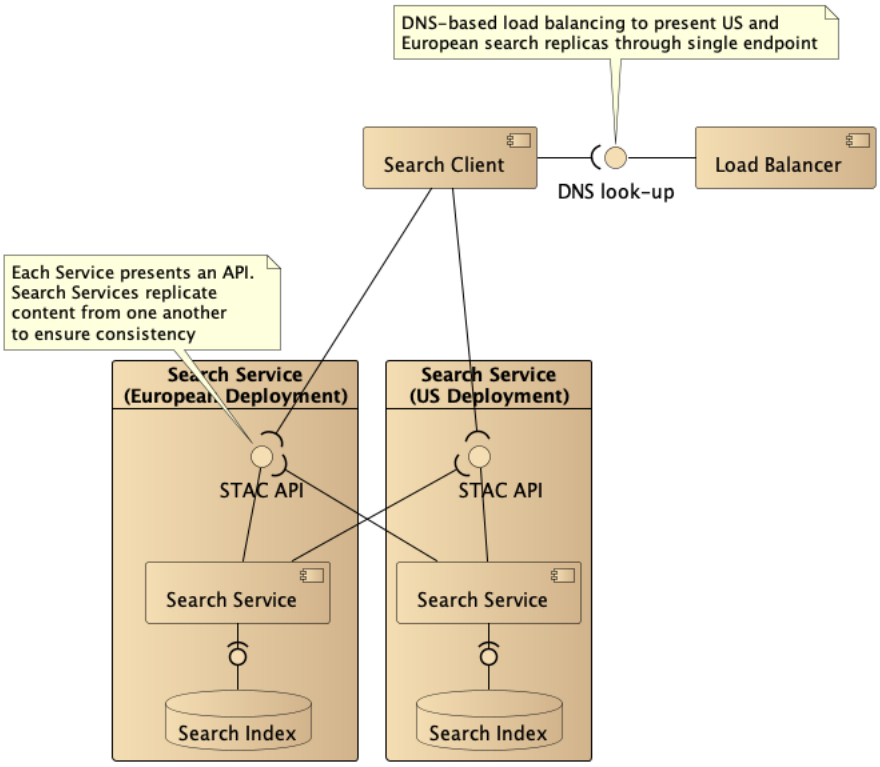



*Figure 5-1: Possible configuration for multiple search service instances*

There are a number of other tasks which were considered for the 3rd amendment work but for which time and resources didn't allow for their inclusion:

- Implement STAC static catalogues as backend to Intake[12]. Intake is a popular client-side solution for cataloguing NetCDF data and is also used at ENES partners to support the ENES Climate Analytics Service by browsable data catalogues in python notebooks. Intake supports STAC formatted metadata files as a backend. Equally, STAC formatted metadata files would be trivial to index into the ESGF STAC search system. This work would make a natural bridge between Intake and ESGF's STAC search. STAC catalogues can be also generated based on intake catalogues, this is partly supported by the current ENES STAC catalogue generation software.
- Extend the CMCC analytics hub to implement the STAC API and thus exposing its entire data catalogue, since not all data are published on the local ESGF data node. This will allow C4I users to perform advanced analysis on the data hosted at CMCC.
- Align STAC API usage with DCAT catalogue approach in order to provide alignment with EOSC (European Open Science Cloud).
- Extend search catalogue to support inclusion of WPS endpoints. This will allow a user to retrieve a WPS endpoint from given search query result metadata (e.g. for this combination of facets of data, allow the client to perform an averaging process over that data).

## 5.3. Data Access Interfaces and Analysis-Ready Data Evaluation

### 5.3.1. Background and Motivation

More widespread use of cloud computing has led to innovations in the way data is stored and accessed in the Earth sciences. This has been driven by the convenience and relative lower cost of object storage on public cloud as opposed to other storage offerings on cloud, such as block storage and parallel file systems, which are much more expensive. *Object store* APIs such as S3 popularised by AWS and OpenStack's Swift use HTTP as its access protocol, so that data may be universally accessible outside an organisation's security perimeter if so desired. This is in contrast to POSIX file systems which rely on mount semantics tied to the host operating system to mediate access.

Throughout its history ESGF has largely relied on HTTP for file access using the traditional approach of a web application server (THREDDS with Apache) to serve files from a POSIX file system together with support for file sub-setting with OPeNDAP.

[12] https://intake.readthedocs.io/

The differing nature of the access interface for *object store* have led to the development of new storage formats to best utilise it. In the Earth observation community, Cloud Optimised GeoTIFF format (COG) specially arranges the components of the data, variables, optical bands etc. and keeps a record of respective byte offsets for these components in order to use HTTP range GET operations to effectively file seek to the required portions of a given object and thus make the access more efficient.

In the climate sciences, *zarr* format together with the Python *xarray* library have been employed to allow efficient access of NetCDF data from *object store*. The underlying data model of regular NetCDF4 (HDF5) enables data to be arranged in chunks to optimise read performance by arrangement of the data into contiguous blocks. When making a *zarr* serialisation from a source NetCDF file, chunks are output as individual objects on the *object store*. Metadata is stored separately and serialised in JSON format, making it convenient to edit this information.

Additionally, it is possible to take individual source NetCDF files and convert them into a contiguous data space by joining along one of the dimensions such as time. *xarray* takes advantage of a lazy load approach to data access such that it can give the user the experience of being able to address access to a much larger data volume then would be otherwise possible. This functionality has been possible for many years using OPeNDAP aggregations. However, experience – for example with the ESA CCI Open Data Portal (REF3) – has shown that this is limited by the performance of the underlying POSIX file system when attempting to open multiple NetCDF files.

The combination of *zarr*, *xarray* and *object store* presents new possibilities but also some challenges particularly with respect to long term archiving of data. For archives containing large volumes of regular NetCDF files, it presents a huge undertaking to convert all the data to *zarr* format. There are also questions about the suitability of *zarr* as a format for data archiving: with data split out into smaller portions as objects in an *object store*, there is a danger that the original mapping of the locations may be lost and thus the data corrupted. For clients, access is trivial in a Python environment but for other languages this may not be as convenient. Fortunately, with the addition of support for *zarr* for NetCDF from Unidata, this should be addressed.

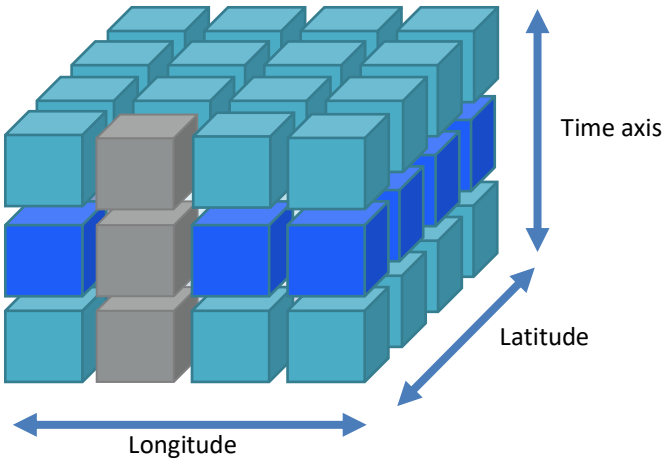
One further possibility is the use of a new library *Kerchunk*[13], which enables the use of a *zarr* layer over the top of existing NetCDF files. This avoids the need to reformat NetCDF data into *zarr* format. Kerchunk effectively creates a map of the file offsets of each of the chunks in the

[13] https://fsspec.github.io/kerchunk/

underlying NetCDF file. The client can then navigate to these chunks using HTTP range GET operations. This is analogous to the technique used by the COG file format.

### 5.3.2. Classes of Data – Analysis Ready Copies

There may be use cases for conversion of data between formats, for example where a copy of data is needed on cloud to support a given access scenario or where, in combination, a version of the data is needed that adopts an alternative chunking strategy to facilitate the predominant access pattern for given analyses. In the latter case, a copy of the data is made into an analysis-ready format.




In the above Figure 5-2, data may be output in NetCDF files, one per time step denoted by the horizontal blue slabs. However, this will be inefficient for time-based queries – vertical grey column representing a query for a pixel through the time dimension – because this involves opening files or objects for each time step. Consequently, it may be beneficial to re-chunking the data when outputting to *zarr* as a series of vertical columnar chunks. A time-based query then involves reading only a single chunk sequentially.

### 5.3.3. Classes of Data – Access Between Warm and Cold Storage Tiers

Besides a need to convert data into a format for analysis, there may also be a requirement to move archive data between warm and cold storage tiers motivated by limitations on absolute disk-based storage capacity, running costs or an increasing consideration, to meet targets for environmental sustainability. Institutional HPC resources and data centres typically have systems

to move data between these tiers (e.g. ECMWF MARS[14], CERN Tape Archive - CTA[15]). For JASMIN the Near Line Data Store (NLDS)[16] is under development incorporating *object store* and tape interfaces for data movement and tiering. At DKRZ also an object store-based solution (StrongLink[17]) is used for the tape based long term archival system.

Given the nature of ESGF as a federated archive, there may be a need to develop a common interface to enable authorised users to request data retrieval from cold storage to disk. This would be likely to need an access policy to determine quotas and limits on requests. It could provide the option to write data from tape to disk storage on the source data node or even export to a third-party location (e.g. public cloud). This would go some way towards mitigating the policy constraints that would be necessary should data be instead written to local disk at the host node.

Given the latency for retrieval, an API would be needed that supports asynchronous requests. WPS (or API Processes) would be suitable, given the existing experience in the consortium applying this standard for data reduction services. In Figure 5-3 below, an external client wishes to retrieve data which is held offline on tape storage. Ordinarily this would not be possible but in this case a WPS is deployed which has a process which allows the migration of data from tape to spinning disk. Once on disk, the data could be made available through standard data access services by the remote client.

A second use case shown in the diagram is one in which the remote client wishes to access data in an analysis-ready format. This requires transformation of the original data and staging to secondary storage either at the same site or a third-party location, for example public cloud. The client invokes the WPS' process to create the analysis-ready copy of data staging it to the required output location.

[14] https://www.ecmwf.int/en/elibrary/18124-mars-ecmwf-meteorological-archive
[15] https://cta.web.cern.ch/cta/
[16] https://github.com/cedadev/nlds and https://techblog.ceda.ac.uk/2022/03/09/near-line-data-store-intro.html
[17] https://stronglink.com/

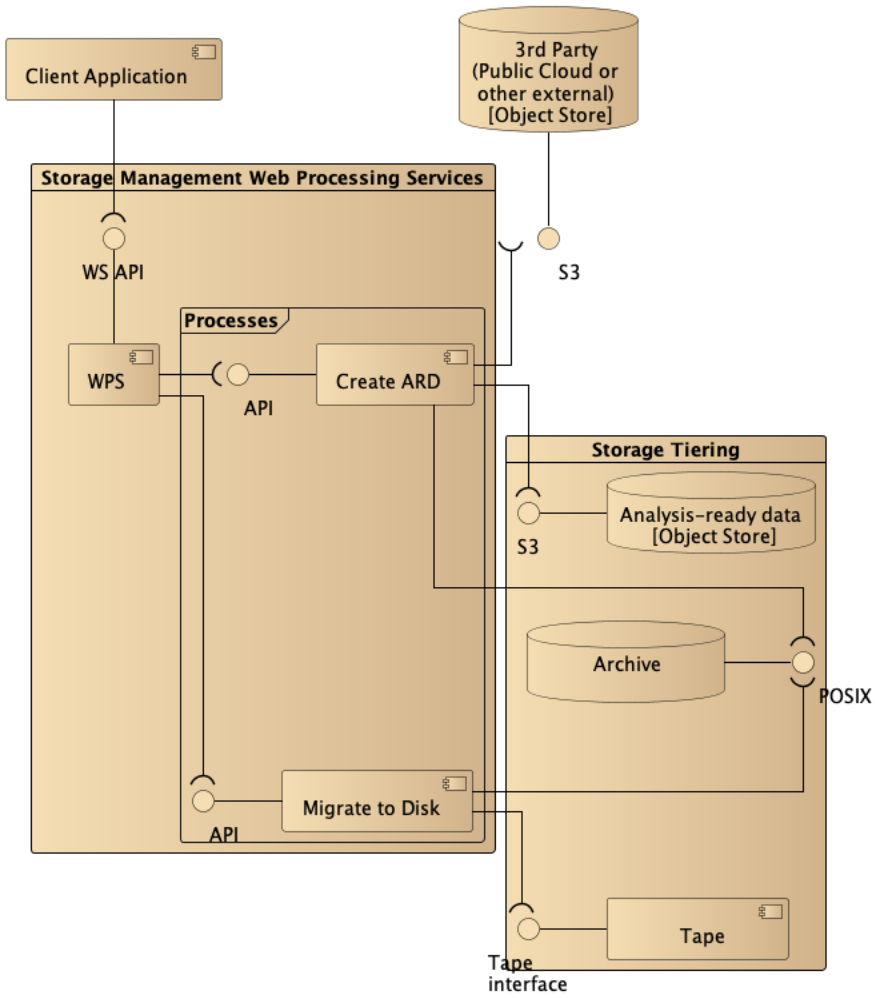



*Figure 5-3: Storage tiers and proposed WPS for managing data between tiers and generating analysis-ready data products*

### 5.3.4. Proposed Evaluation Work

Unfortunately, time and resources have not been available to investigate the use of *zarr* or similar technologies for use with the ENES CDI. However, a pilot could be investigated as follows:

- Build on existing pilot work with *zarr* on JASMIN's *object store* and on an existing zarr data generation pipeline for DKRZ's SWIFT *object store* and associated Intake/STAC catalogue generation.
- Integration with WPS developed for Copernicus. The WPS implementation uses an *xarray* interface to data and therefore could be extended to provide *zarr* formatted output on *object store* as a result of an operation. For example, apply a sub-setting/re-

gridding/averaging operation but serialise the output data to *object store* using *zarr* for the client to then access.

- Integrate these scenarios with a workflow involving the Climate4Impact portal.

## 6. Conclusions and Recommendations

This document outlines plans for the development of the ENES CDI and underpinning ESGF infrastructure. The ESGF Future Architecture is a refresh and update addressing recognised shortcomings in the existing system, such as difficulties in deployment and use of bespoke APIs. There is additionally a more fundamental change in focus required in response to observed overall shift in the provision of data infrastructures for the Earth Sciences, away from the existing model of data discovery and download towards data discovery and in-situ analysis of data on web-based platforms.

Over the remainder of the IS-ENES3 project and as the US ESGF 2.0 project commences, it will be important to prioritise efforts to complete the development of the new search system and roll out upgrades to the nodes in the federation. This is essential to avoid potential issues related to security and ongoing maintenance of the legacy search system. Following from the implementation of a baseline, some of the other proposals from the Future Architecture, around deeper integration of search with other services in the CDI such as ES-Doc, should be investigated.

Looking into the medium term, the new opportunities arising through the use of object storage should be explored further, together with solutions for management of data between different storage tiers. This, together with the developments around data reduction services, will facilitate an underpinning service layer to best support growth in the development of web-based platforms which exploit the CDI.