NERC Environmental Data Service

Publishing Environmental Data APIs for use in AI workflows: recommendations and demonstrators of a standard approach within the NERC Environmental Data Service

27/06/2025

Authors: Chris Card[1], Rachel Heaven[1], Andrew Kingdon[1], Patrick Bell[1], Alex Baldwin[2], Jeremy Carter[3], Jonathan Coney[2], Jonathan Cooper[3], Itahisa Gonzalez Alvarez[1], Michael Hollaway[3], Matthew McCormack[2], David Poulter[5], Ag Stephens[5], Philip Trembath[3]

Contributors: Alberto Arribas Herranz[2], Timothy Barnes[4], Emma Bee[1], Andy Bevan[1], Jonathan Booth[1], Paul Breen[4], Matthew Cazaly[2], Matthew Coole[3], Thomas Gardner[2], Lily Gouldsbrough[3], Matthew Hopson[2], Henry James[2], Edd Lewis[1], Alan MacKenzie[1], Eric Orenstein[2], Roman Roth[1], Colin Sauze[2], Rod Scott[3], Michael Tso[3], Elisabeth Wetchy[2], Thomas Zwagerman[4]

[1]British Geological Survey, [2]National Oceanography Centre, [3]UK Centre for Ecology and Hydrology, [4]British Antarctic Survey, [5]Centre for Environmental Data Analysis

Acknowledgements

This research project “API 4 AI” (NERC award number: UKRI292: API 4 AI) was enabled by UKRI Digital Research Infrastructure Programme funding through the opportunity entitled ‘Enhancing digital research infrastructures by trialling approaches to skills and software’. This funding provided resources across the British Oceanographic Data Centre (BODC) hosted at National Oceanographic Centre (NOC), the Environmental Information Data Centre (EIDC) hosted at UK Centre for Ecology & Hydrology (UKCEH) and the National Geoscience Data Centre (NGDC) hosted at British Geological Survey (BGS)
Contributions in kind were made by the other NERC Data Centres, the Polar Data Centre (UK PDC) hosted at the British Antarctic Survey (BAS), and the Centre for Environmental Data Analysis (CEDA), hosted at the Science Technologies and Facilities Council (STFC).
For BGS staff this report was published with permission of the Director, British Geological Survey.

NERC Environmental Data Service

Contents

NERC Environmental Data Service

Summary

The advent of artificial intelligence as a scientific tool is driving a new demand for multi-disciplinary data analyses that crosscuts scientific domain boundaries. Developing interoperable solutions to data delivery enables new avenues for scientific investigation. This report summarises work on development of interoperability tools, for environmental research.

The NERC Environmental Data Service (EDS) consists of five domain specific data centres supplying data to environmental scientists. The data is findable through data catalogues and web search engines, thanks to decades of collaborative effort implementing standardised discovery metadata. However, the access methods, formats and content of the delivered data are varied, and users need to spend time navigating and understanding these. Data access through Application Programming Interfaces (API) are preferred over bulk data downloads, because they allow programmatic querying and repeatable workflows, and are recommended for access to data that is large, complex or being continuously updated.

This project's detailed aim was a greater level of standardisation of data access APIs across the EDS, with a particular focus on their use in AI and machine learning (ML) applications. This will reduce the effort needed by EDS as data publishers and by environmental researchers as data consumers, saving development time and easing data integration processes. This supports systematic AI analysis of multiple environmental datatypes to underpin development of predictive environmental modelling and digital twins.

Through co-design and Agile development processes, we identified and recommended mlcommons Croissant specification as a common standard to help ML consumers interface between data APIs - and bulk download - of any design. Croissant extends existing metadata standards, is understood by web search engines and AI agents to support findability and is integrated into ML python libraries and popular ML platforms to support usability.

We created a number of Croissant descriptors from each of the data centres, a new data API, and extensions of metadata APIs to serve croissant metadata. We created demonstrator ML workflow notebooks using the Croissant descriptors and data APIs and ran these on different data science platforms to demonstrate portability.

Croissant [26] is a relatively new standard and not built primarily for data access by API or for multi-dimensional spatiotemporal data. We identified areas where croissant and the implementing libraries could work better for these use cases, such as use of the

emerging geo-croissant extension, integration with OpenAPI [38] specifications, and support for authenticated data access.

At the API implementation level our recommendations were more flexible, and in line with existing EDS practices to use API standards appropriate to the data type (e.g. OGC [37], STAC [43]), and to describe APIs using OpenAPI specification.

Introduction

Background

The NERC Environmental Data Service (EDS) provides a focal point for scientific data and information spanning environmental science domains: atmosphere and climate; earth observation, polar and cryosphere; marine, terrestrial and freshwater; geoscience, and solar and space physics. Improving access to quality-assured, high-resolution environmental information and associated software tools will enable new users and communities to access environmental research by facilitating integrated analyses of environmental processes in response to societal challenges and environmental change.

NERC Data Centres funded by this project were British Oceanographic Data Centre (BODC) hosted at National Oceanographic Centre (NOC), the Environmental Information Data Centre (EIDC) hosted at UK Centre for Ecology & Hydrology (UKCEH) and the National Geoscience Data Centre (NGDC) hosted at British Geological Survey (BGS).

Contributions in kind were made by the other NERC Data Centres; the Polar Data Centre (UK PDC) hosted at the British Antarctic Survey (BAS), and the Centre for Environmental Data Analysis (CEDA), hosted at the Science Technologies and Facilities Council (STFC).

The Data Centres

The National Geoscience Data Centre (NGDC) hosted by BGS collects and preserves geoscientific data and information, making them available for the long-term to a wide range of users and communities. BGS is the UK's national geological survey and a world-leading independent research organisation, providing objective geoscientific data, information, and expertise.

The Environmental Information Data Centre (EIDC) hosted by UKCEH is the UK's national data centre for terrestrial and freshwater sciences. It curates environmental data for the long term and makes it accessible and reusable.

The National Oceanography Centre (NOC) is the UK’s leading institution for ocean research, climate studies, and marine technology. It operates research vessels, manages ocean data archives, and develops autonomous underwater vehicles. NOC

hosts the British Oceanographic Data Centre (BODC) which provides instant access to over 130,000 unique data sets and helps provide answers to both local questions such as the likelihood of coastal flooding, or global issues such as the impact of climate change.

The UK Polar Data Centre (UK PDC) is the focal point for Arctic and Antarctic environmental data management in the UK.

The Centre for Environmental Data Analysis (CEDA) serves the environmental science community by the provision of data centres, data analysis & access, and research project participation.

Context and motivation

Many environmental processes take place in multiple, environmental domains involving processes that crosscut discipline countries. For example, coastal erosion processes involve rocks (geosciences) being affected by waves (marine science) and surface erosion (hydrology), driven by wind and rain (atmospheric processes), protected by plant life (terrestrial ecology) and affecting human population and infrastructure.

Increasingly scientists are trying to provide scenario-based modelling of how environmental systems and related human infrastructure are affected by the risks of climate change. Defining these models requires quantitative input data from multiple sources.

The wholesale availability of Machine Learning (ML) and other Artificial Intelligence (AI) technologies is changing the way that environmental data is being used. They enable a new scale of processing and allow the identification of trends and signals in data streams that were previously impossible to identify or practically impossible to deliver due to a lack of computing power.

A significant portion of data science involves data wrangling (i.e. gathering, preparing, cleaning, and organising) rather than direct modelling, analysis and inference. Accessing and integrating large, complex datasets requires well-structured metadata to describe data properties, origins, formats, and accessibility. This challenge is particularly pronounced in Earth system sciences, where datasets are vast, multidimensional, and stored across diverse formats and platforms. Lowering the barriers to data access has the potential to:

1. Greatly increase the usage of key environmental datasets

2. Significantly improve the productivity of scientists and data scientists by allowing them to focus on using the data rather than getting it

The scientific method requires that data be traceable to source and that analytical processes be reproducible during peer-review for validation of outputs. Undocumented intermediate data wrangling stages that occur between data retrieval and ingestion into modelling or processing software damages the traceability and reproducibility of modelling systems, as the exact data used in analysis cannot be clarified. Ingestion of static datasets, cited using persistent identifiers such as DOI ensure accurate reproducibility,

As an alternative to providing datasets as static archive downloads, reproducibility can be achieved if data is provided via Application Programming Interfaces (API) [47]. APIs provide machine-readable access to the data for use in programmatic and repeatable workflows. They support dynamic sub-setting, re-formatting and are particularly useful for environmental monitoring data that is constantly being updated. Use of standardised APIs presents opportunities for multiple sources to be ingested through single gateways minimising data processing of different input data streams. Standardisation can be implemented at various levels:

- metadata - how the dataset and API endpoint is described and therefore discovered
- core data format - the structure of commonly shared queries and data attributes, such as spatiotemporal attributes
- full data format - all queries and data attributes are standardised
- Semantic interoperability – attributes and data values are defined by linked data identifiers

Using the greatest degree of data standardisation, changing between ingesting different environmental datasets requires flicking a virtual switch rather than complex data engineering, thereby enabling increased automation of modelling parameters. In practice, standardising on core attributes is more achievable and offers wider integration opportunities, but there will still be multiple standards applicable at this level for different data types. This forms part of the “narrow middle” proposed for EDS data commons roadmap [3].

This project aim was to share experience in creating and applying a standardised approach to making data APIs available in a way that would lower the barriers to use in AI workflows. This would be demonstrated in worked examples within existing data science platforms hosted by UKCEH (DataLabs) and NOC (Data Science Platform, DSP). The intended impact will be to widen data access to environmental researchers and

enable systematic AI analysis of multiple environmental data types to underpin the development of predictive environmental modelling and digital twins.

Scope

The data APIs under consideration in this project are Web APIs that provide read only access to environmental datasets such as earth observation data, time series sensor data, series of environmental monitoring images, geospatial features and collections of images with associated feature level attributes.

This project was not concerned with standardisation of discovery metadata of datasets, or of semantic standardisation of parameter vocabularies in either the metadata or feature level attributes of data. These are complementary research areas and the subject of other previous and current work within and between the Environmental Data Centres. [3,32]

For the purposes of this project, all datasets are assumed to be publicly available on the Web. Handling authenticated access or API keys is out of scope, though a consideration for future work.

Summary of proposed tasks and deliverables

The original project proposal defined a sequence of work packages to facilitate effective delivery of intended project outcomes. However, these were adapted through the lifespan of the project as a consequence of co-design processes and Agile development strategies [2]. This section outlines the work packages and deliverables as set out in the original project proposal.

Work Package 1: Use case identification workshop

A workshop bringing together technical experts and environmental data users to explore and co-design potential use-cases that can demonstrate the potential for API powered AI/ML workflows for environmental data

Deliverables:

- Workshop event
- Use case report and specification

Work Package 2: Documentation to enable standardised APIs

Consultation with data centres to discover existing practices and architectures for data delivery APIs. Development of guidelines and documentation that standardise the design, development, deployment and documentation of API provision for use in AI/ML workflows.

Deliverables:

- Recommendations
- API guidelines documentation

Work Package 3: Creation of API Reference Templates

Deliverables:

- API specifications and reference implementation code to support identified use cases

Work Package 4: Provision of APIs to support use in AI/ML workflows

Deliverables:

- published APIs and documentation for two datasets and use cases

Work Package 5: Integration of API workflows into existing data science platforms

This deliverable will demonstrate accessibility of APIs for AI/ML workflows though use in example data science processing notebooks, hosted on CEH and NOC data science platforms.

Deliverables:

- API used in Data Science Platform and DataLabs to produce new outputs for two use cases

Use case identification workshop

The first work package of the project was a requirements gathering and co-design workshop attended by the funded partners and contributing partners.

The workshop provided a clearer understanding of the AI platforms available for running AI workflows and led to an initial agreement on the standards for data APIs to support those workflows.

The key outcomes of the workshop were:

- Identification of 20+ AI use cases across EDS, either completed or planned.
- Development of a high-level implementation timeline for project deliverables.

The full workshop report is available through the NERC Open Research Archive [6].

Documentation to enable standardised APIs

EDS have been developing a roadmap for data interoperability [3] based on a data commons consisting of standardised data formats, metadata schemas, vocabularies and use of recognised API standards for different data types. Together these help make environmental datasets meet the FAIR principles (Findable, Accessible, Interoperable and Reusable).

Standardisation of metadata for dataset descriptions

The Findability principle of FAIR is addressed through the use of quality standardised metadata and supporting controlled vocabularies, available in a machine-readable file format.

One of the key standard formats is ISO 19115/19139 xml [19,20], which is already implemented by all the NERC data centres using the UK GEMINI [1] profile, and in for marine data the MEDIN [25] profile. This metadata used in federated data catalogues for a range of communities [10,13,25,33]. Recent collaborative work by the NERC data centres resulted in a revision (v2.0) to the NERC metadata guidance with the aim of improving the quality and usability of EDS metadata, with a greater degree of consistency between the data centres [32].

Another key standard for metadata is schema.org [41], which is used by internet search engines such as Google Dataset Search [15]. The Geospatial Commission Data Discoverability project [14] reported that “around 75% of users turn to a search engine first when searching for geospatial data. The remaining 25% eventually turn to a search engine when their first attempts are unsuccessful". Schema.org metadata can also be understood by AI agents, meaning that users can be told about the existence of such datasets, and given answers based on any of the information contained in the metadata fields. Metadata supplied from the NERC data centres is federated in the EDS data catalogue and available in schema.org format, but only some of the source catalogues are currently configured to be indexable by search engines. See recommendation API4AI-R1.

Information contained in existing published discovery metadata for EDS datasets therefore already goes someway to also addressing accessibility and interoperability of datasets for AI, by providing the online links for datasets and some indication of the file format. The recommendations made in v2.0 of the NERC metadata guidance [32] will further improve the usefulness of discovery metadata, for example through the use of controlled vocabularies for data formats and measured parameters, and online access URLs that can be used programmatically. See recommendation API4AI-R2.

Dataset descriptions for AI – Croissant

Despite these existing standards which are widely used by authoritative data providers, AI datasets published on sites such as HuggingFace [17], Kaggle [22] and Tensorflow [44] had not used a standardised machine-readable metadata scheme.

The Croissant specification [26] was designed to meet this need, improving the interoperability of AI datasets by providing a structured metadata format. It is suitable for any type of dataset, e.g. text, structured data, images, audio, video. It is based on schema.org, with extensions where required. Croissant covers more of the Interoperability aspect of FAIR than discovery metadata standards do, by describing each attribute that can be found in the dataset. It also allows datasets to be described to provide information for Responsible AI (RAI). See later section.

A python library has been built that makes it easy to use and create Croissant descriptions and work with the datasets in popular ML/AI tools. It enables developers to capture processing steps and compute RAI metrics automatically and systematically, identifying potential data quality issues to be fixed.

Standardising the _interface_ between datasets and AI workflows using the Croissant specification would meet the objectives of this project and be a flexible and scalable solution to accommodate a range of API standards and data types. See recommendation API4AI-R3.

Standardisation of access methods

The dataset description in the metadata will tell the user where and how to get hold of the data. AI datasets can be made accessible in two main ways: bulk download or via multiple parameterised API calls.

Bulk download is where users download the entire dataset at once, often as a compressed archive file. This method is suitable for offline analysis and large-scale model training but can be inefficient when dealing with massive or complex datasets from which only a portion is required, or dynamic data that is being continuously updated. W3C Data on the Web Best Practices 17 to 22 [47] should be followed for bulk data access.

The focus of this project was on access to data via a data access API, which W3C states “offers the greatest flexibility and processability for consumers of your data. It can enable real-time data usage, filtering on request, and the ability to work with the data at an atomic level. If your dataset is large, frequently updated, or highly complex, an API is likely to be the best option for publishing your data." [47]

Data Access API standards

The Data Centres are already implementing a range of API access to datasets. For example, CEDA provides metadata and indexing solutions such as SpatioTemporal Asset Catalog (STAC) [9,41] and Kerchunk [23] to enhance dataset discoverability and access. STAC provides a structured way to index large geospatial datasets, offering machine-readable catalogs with RESTful API access. Meanwhile, Kerchunk enables lightweight virtual indexing of chunked Zarr [49] and NetCDF [45] datasets, allowing efficient data access without duplication. Kerchunk can provide access over a range of protocols (such as local disk, HTTPS and S3).

BGS offers many spatial and time series datasets in OGC standard formats, increasingly using the emerging OGC API family of standards, and using pygeoapi [39] - and soon ESRI ArcGIS Server [12]- as a low-code API implementation [8] when suitable. Pygeoapi works by connecting to a database table – typically in which each row is a geometry and some attributes - and exposing it as a RESTful API conforming to OGC API standards. To work with more complex datasets from relational databases then compromises are needed to create the flattened data table, such as de-normalising, or using embedded JSON data types.

UKCEH also have a pygeoapi implementation, but many of their datasets are not appropriate for that data model.

OpenAPI specifications are already widely used in the Data Centres for describing APIs. Through use of the Croissant specification as an interface to data access APIs, it is not necessary to standardise on a single API design for the benefit of AI/ML consumers. However, there are benefits to standardisation of API design for other user communities, such as geospatial/GIS, but EDS needs to support a range of appropriate standards for those different communities. Standardisation of API design is also efficient for publishers if low-code open-source implementations such as pygeoapi [39] can be utilised.

See recommendation API4AI-R4.

Standardisation of labelled training datasets

During this project, OGC published a standard for geospatial or temporal datasets that contain labels or annotations to be used as training data for AI/ML workflows, “Training Data Markup Language for Artificial Intelligence”. This is in the process of being adopted as an ISO and British Standard [21,33].

In the design workshop, none of the datasets to be implemented in this project included labelled data, and due to the timing of the publication of these standards, this is an area for future research.

Croissant is primarily designed for describing static datasets that are available for bulk download, rather than those delivered dynamically via parameterised API queries. Croissant's intention is to provides metadata about entire datasets (structure, source, and formats), ensuring datasets can be easily discovered, shared, and loaded into ML pipelines.

Using AI datasets appropriately and responsibly

Using AI datasets responsibly requires ethical data collection, careful processing, fair model training, and accountable deployment.

Data collection for AI should be carried out ethically, ensuring informed consent when handling personal information, and comply with legal frameworks like GDPR & copyright ownership. Additionally, datasets should be diverse and representative to prevent biased AI outcomes.

During data processing, sensitive information should be anonymised to safeguard privacy. Bias detection and mitigation are crucial to prevent AI models from making unfair decisions. Ensuring data quality through accurate labelling and validation enhances the reliability of AI systems

Model training must be transparent, with clear documentation of data sources and limitations. AI models should be tested for bias and fairness, with corrective measures applied when necessary.

When deploying AI systems, accountability is key. Organisations must take responsibility for AI-driven decisions, ensuring human oversight in critical areas. Continuous monitoring and feedback loops help improve AI performance while minimising unintended consequences.

Ethical considerations should guide AI usage. AI systems should avoid causing harm, provide explainable decisions, and promote transparency. Open access to datasets, where appropriate, encourages collaboration and innovation while maintaining responsible AI development.

The Croissant specification provides a machine-readable format for capturing and publishing metadata about ML datasets. It records how datasets are created, processed, enriched and allows for quality to be assessed throughout their lifecycle, ideally this will occur via automation when integrated with ML frameworks.

Additionally, Croissant can also be extended using the Croissant RAI vocabulary [27], which supports responsible AI use cases, including data lifecycle tracking, labelling, safety, fairness, traceability, and compliance.

Creation of API Reference Templates

As described above, the original work package 3, Creation of API Reference Templates, was adapted to cover the creation of and guidelines for Croissant reference implementations.

Creation of Croissant implementations and guidelines

During the project, the use of Croissant was investigated, utilising its fundamental features: Distributions, FileObjects and RecordSets to define the structure of the data. This provided a foundational understanding of Croissant's capabilities. However, this represents only a fraction of what Croissant can support, highlighting the need for further investigation and trialling to fully assess its potential applications.

Due to time and technical constraints, many of the example Croissant descriptor files were generated statically rather than dynamically. These were served as static resources for use in the demonstrator AI/ML workflows. Nevertheless, they referenced data accessible through live APIs and data sources.

Utility and sample code were also prototyped to facilitate both the reading and writing of Croissant descriptor files. Additionally, we explored how Croissant RecordSets could be generated from SQL queries, demonstrating potential methods for automatic generation of Croissant files and how to integrate Croissant with structured databases.

Several issues with Croissant emerged during the project.

- Croissant is primarily designed for describing static datasets that are available for bulk download, rather than those accessed dynamically through parameterised API queries. Its primary purpose is to provide metadata about entire datasets—defining their structure, source, and formats—ensuring they can be easily discovered, shared, and integrated into machine learning pipelines.
- One challenge encountered was the unclear use case of FileSets. The supported behaviours of FileSets differed from those of FileObjects, and this distinction was not well documented. Our inference was that when a FileSet is the data source, the most granular level of information a Field can reference is the files

themselves. In contrast, when a FileObject is the source, the Field can reference elements within the file, depending on the file type.

- Although Croissant is still a relatively new standard, it has already gained significant support from major platforms and organisations. However, due to its early stage of development, technical documentation, detailed guidance, and mature tooling remain limited. This makes it challenging to fully explore its capabilities, particularly when working with more complex aspects of the specification.

The accompanying Python library, mlcroissant, proved highly useful for generating Croissant files and seamlessly integrating them into AI/ML workflows. However, a few limitations were identified during the project.

- Currently, the mlcroissant library supports data retrieval (via URLs) only from unauthenticated endpoints or those using basic authentication. Also, it does not support Croissant files hosted on endpoints requiring any form of authentication method.
- During experimentation with the mlcroissant library for consuming JSON data, it was observed that the selected JSONPath dependency does not fully support the complete JSONPath specification.
- The Croissant specification explicitly recommends including checksums for FileObjects that represent static data (checksums not being required for “live data”). However, when using the mlcroissant library to generate a Croissant file, it raises errors if a checksum is not provided, even when the isLiveDataset property is set to true.

The process of creating Croissant descriptor files and utilising the mlcroissant library provided valuable insights into dataset standardisation and interoperability. Throughout this experience, key findings, issues encountered, and observed limitations have been consolidated in the guidelines Croissant Guidelines document [31].

Provision of APIs to support use in AI/ML workflows

BGS

The dataset that BGS chose to deliver via a new API for this project is the BGS Mineralogy and Petrology Collection. This includes (127,000+) digital images of rock thin sections from samples referenced in the BGS Petrological Collection Database (Britrocks). Each thin section is captured in two reference images: one in Plane Polarized Light (PPL) and another in Crossed Polarized Light (XPL). The images are useful in AI/ML workflows to

characterise the subsurface at the micro-scale for a variety of science research challenges.

As a result of recent initiatives in BGS, the Britrocks collection is now managed as a subset of an aggregated collection of all geological samples, referred to as the Generic Sample Register (GSR) , therefore an API was built to deliver all the GSR data. The GSR API provides detailed descriptions of physical samples and specimens, including metadata on sample locality, geology, and associated images.

The GSR API [5] was developed using pygeoapi (Python) to leverage its rapid development capabilities and built-in support for OGC standards (e.g., Features, Records) and multiple data formats, including JSON, GeoJSON, JSON-LD, and CSV. It is released as a beta service initially.

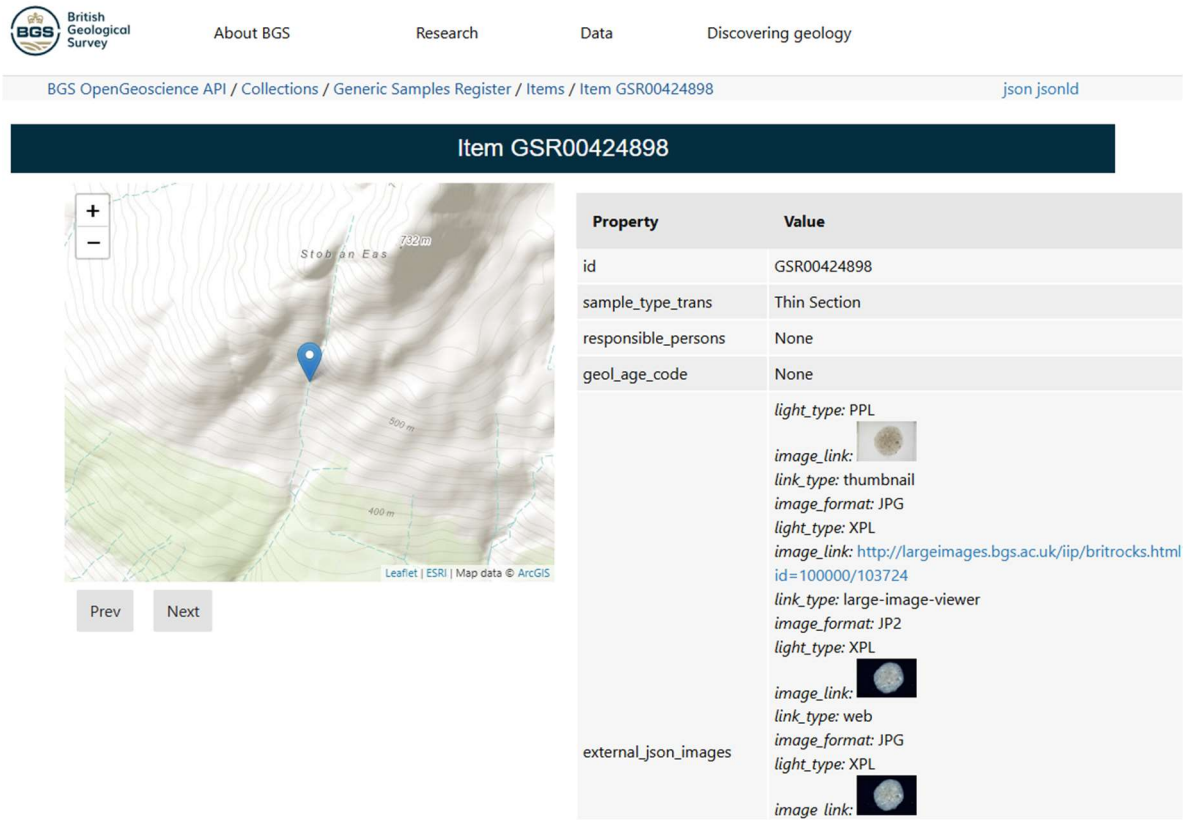


Figure 1, GSR API, HTML representation

However, some shortcomings were identified with pygeoapi during development:

- Lack of support for complex objects: This was addressed by embedding JSON objects directly within the database tables supporting the API.

- Challenges in extending pygeoapi with Croissant support: It was found that adding additional endpoints would require significant modifications to pygeoapi's source code. To circumvent this, the Croissant descriptor for the GSR API was hosted externally while referencing the GSR API via URL as the data source for AI/ML workflows.

UKCEH

UKCEH chose to extend their existing metadata catalogue API with a new croissant descriptor endpoint, thereby enabling a croissant interface to a number of existing online datasets.

UKCEH datasets can be explored in the EIDC catalogue [46]. The JSON metadata endpoints facilitate machine-readability using schema.org, JSON-LD and ro-crate [40]. These endpoints include information on dataset availability, licensing, download URLs, formats and structure. An example is the ro-crate endpoint for the COSMOS-UK daily hydrometeorological and soil moisture observation dataset.

The API implementation is a Java Spring-Boot [42] application with a backend document store and templating engine. To test this architecture's ability to support Machine Learning, the templating engine was augmented to support the croissant format.

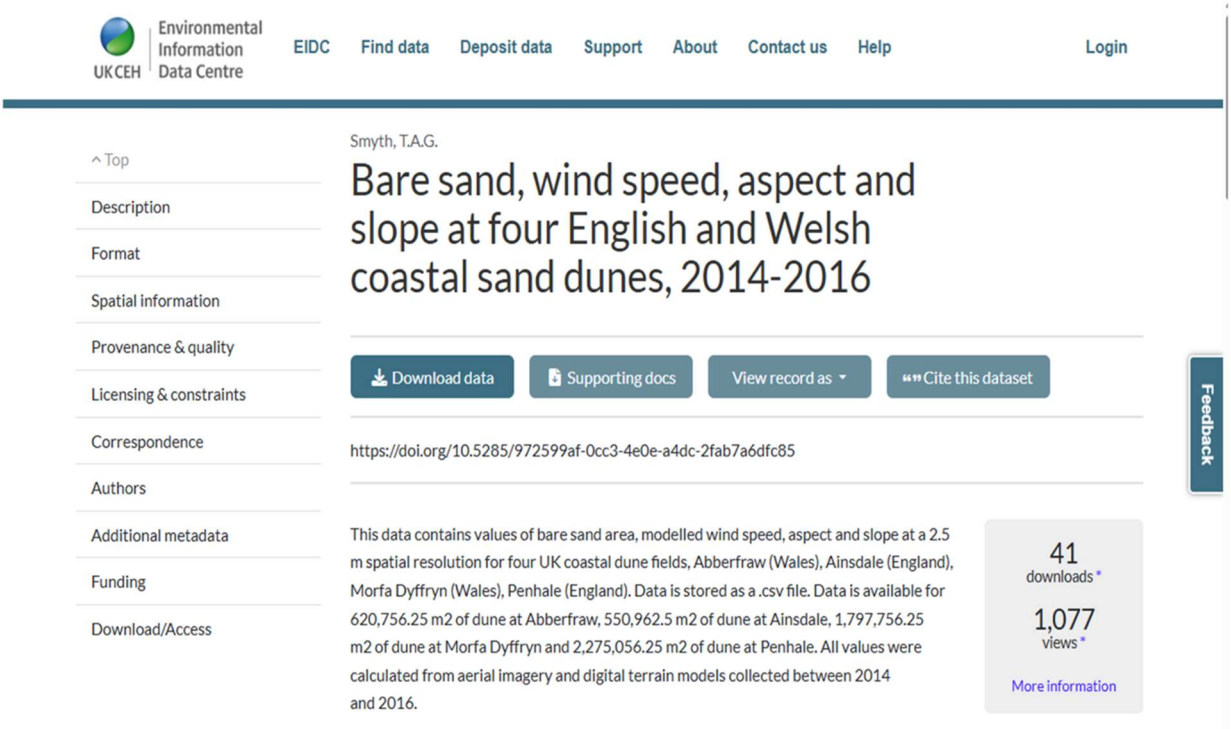


Figure 2, UKCEH EIDC catalogue, HTML representation

NOC

NOC chose to extend their existing ML dataset management and annotation software (CRAB) to enable export of croissant metadata, thereby enabling a croissant interface to a number of existing online datasets.

CRAB

Centralised Repository for Annotations and BLOBs (CRAB) [34] is a Free Open-Source Software stack for ML dataset management and annotation. It includes preconfigured profiles for importing Imaging Flow CytoBot (IFCB) [18] and LISST-Holo [24] data. CRAB manages ingesting data, providing all files in a uniform TIFF format on an S3 compatible object store. Metadata is searchable using queries to the integrated CouchDB server, allowing you to easily access data stored on the platform.

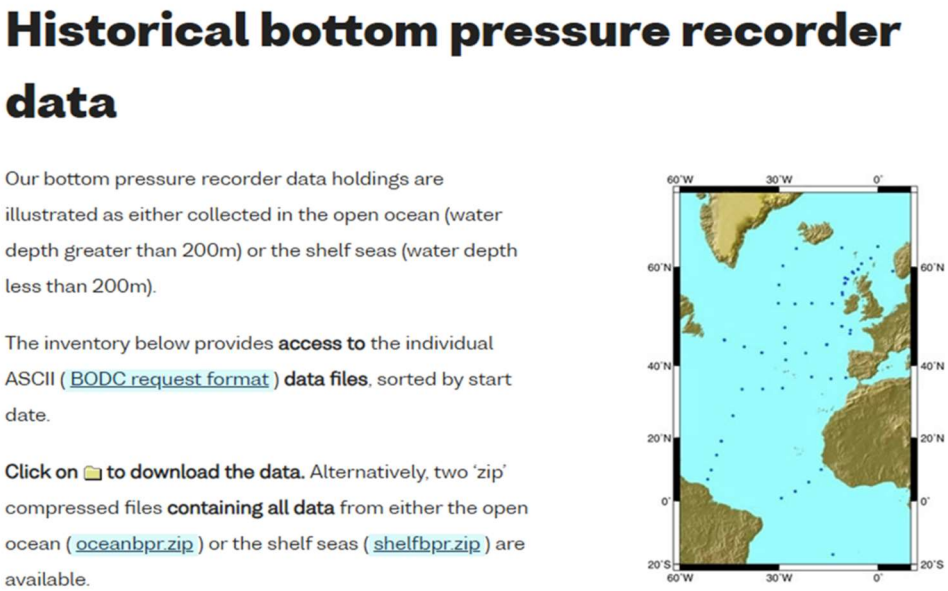
CRAB uses Croissant for standardised and machine-readable metadata for enhanced FAIR compliance, ensuring data accessibility, interoperability, and long-term usability. Its primary use cases include leveraging high-performance computing (HPC) servers for rapid development of machine learning models for image classification and automating or batch-processing large image datasets on cloud platforms for efficient analysis and scalability.

Figure 3, NOC, CRAB API (GitHub project)

Due to existing challenges with the data delivery systems, a live API could not be implemented as part of this project. However, the code for a functional implementation is publicly accessible, and several key outcomes have been achieved:

- A Python script [31] was developed to convert Schema.org JSON-LD metadata into Croissant descriptor files. This will be instrumental in generating Croissant files for the existing data catalogue.

- The Croissant outputs from this script were manually modified to reference publicly available BODC data, enabling the development of machine learning pipelines [31].
- This work lays the foundation for exposing existing machine-accessible data in Croissant format via APIs, enabling seamless integration into AI/ML workflows and enhancing interoperability and usability.



Historical bottom pressure recorder data

Our bottom pressure recorder data holdings are illustrated as either collected in the open ocean (water depth greater than 200m) or the shelf seas (water depth less than 200m).

The inventory below provides **access to** the individual ASCII (BODC request format) **data files**, sorted by start date.

Click on to **download the data.** Alternatively, two 'zip' compressed files **containing all data** from either the open ocean (oceanbpr.zip) or the shelf seas (shelfbpr.zip) are available.

Figure 4, NOC, BODC API, HTML representation

Integration of API workflows into existing data science platforms

This section provides demonstrators of how these APIs can be practically implemented into scientific AI/ML workflows hosted on data science platforms to answer a specific science question These are chosen as processes that are representative of typical challenges in each institute and /or within each scientific discipline.

BGS

Science question: Provide an estimate of a sample's mineral composition from thin section images (plane polar and crossed polar light).

The BGS Python notebooks [31]. illustrate how standardised Croissant descriptor files, in conjunction with the croissantml library [28], can be employed for Machine Learning-based data analysis in the field of mineralogy.

- For the high-resolution images use-case, a Croissant descriptor file is used to reference a small set of high-resolution thin-section image pairs.
- For the low-resolution use-case, a Croissant descriptor file references the live GSR API, which facilitates access to pairs of thin-section images for each sample.

Once the URLs for the corresponding thin-section images (captured in both plane polarised light and cross polarised light) are retrieved, the analysis process can begin.

The initial step involves isolating the rock sample in each image from the surrounding box. This is followed by a clustering process to carry out basic image segmentation, which provides an initial visual approximation of the sample's mineral composition based purely on pixel colours.

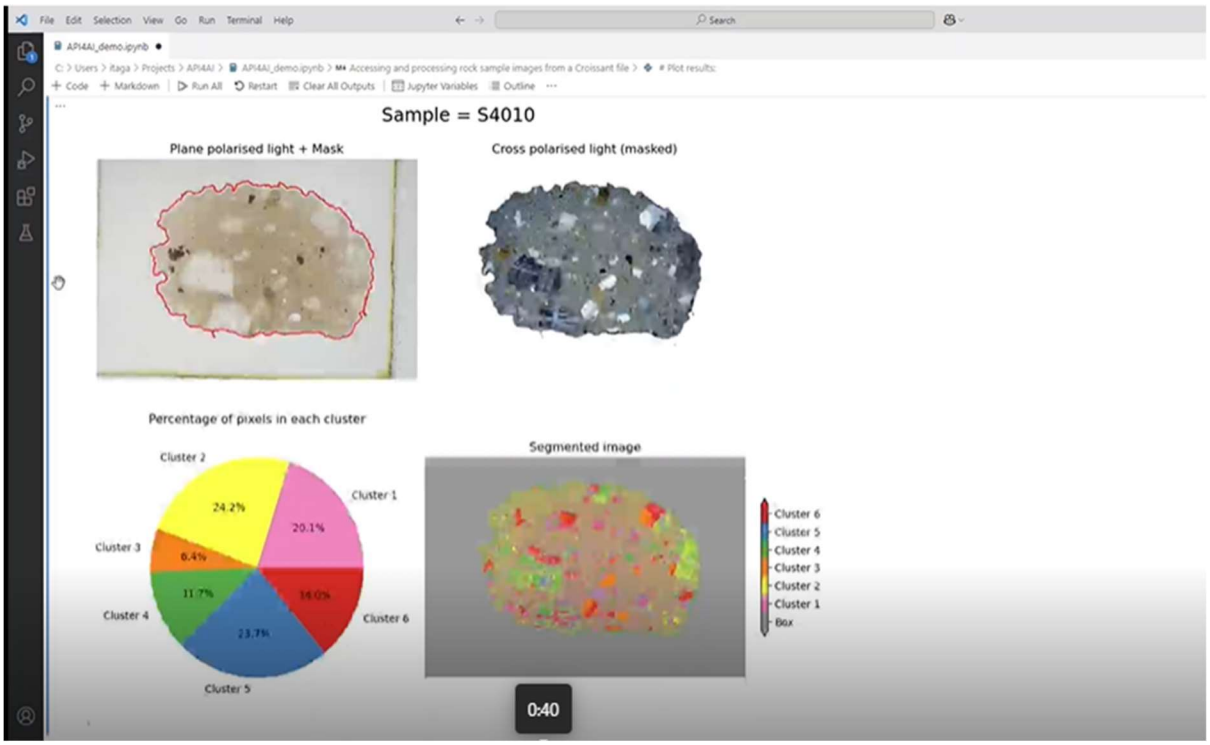


Figure 5, BGS, Python notebook: mineral composition

UKCEH

Science question: Predict the proportion of sand in an image of coastal sand dunes, from wind speed, degree of slope and aspect data.

The UKCEH dataset selected for analysis comprises values for bare sand area, modelled wind speed, aspect, and slope, all at a spatial resolution of 2.5 m. The data covers four UK coastal dune fields. The Croissant descriptor file contains metadata for these datasets, outlining the data structure and providing the necessary URLs to access the data.

The Python notebook [31]. loads the dataset via the Croissant descriptor file, that references data made available via the EIDC search catalogue, using the croissantml library [28], as well as a custom TensorFlow 'CroissantBuilder'.

Once the data is downloaded, it is processed through a Machine Learning pipeline. This pipeline demonstrates the use of an MLPRegressor neural network from scikit-learn, as well as a neural network for tabular data implemented in PyTorch to predict the proportion of sand in an image from the wind speed, degree of slope and aspect.

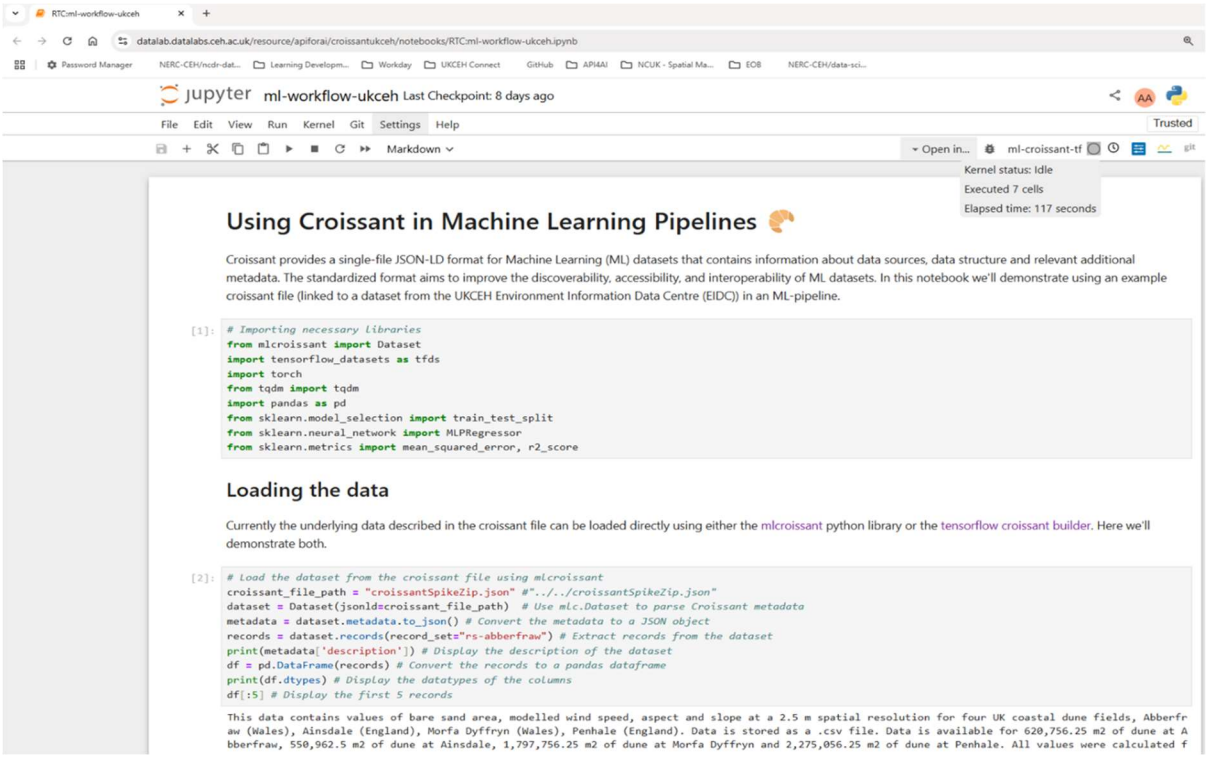


Figure 6, UKCEH, Python notebook: proportion of sand in an image of coastal sand dunes

NOC

Science question: Train a simple machine learning model to forecast sea floor pressure from existing sea floor pressure gauge data.

The NOC Python notebook [31] illustrates the process of ingesting data through a Croissant descriptor file, utilising the croissantml library [26], to train a simple machine learning model for a small forecasting task. The task focuses on existing bottom pressure recorder data maintained by the British Oceanographic Data Centre (BODC)., to train a simple machine learning model for a small forecasting task. The task focuses on existing bottom pressure recorder data maintained by the British Oceanographic Data Centre (BODC).

The notebook loads the data via the Croissant descriptor file and employs the Darts Python library to train a neural network for forecasting the bottom pressure recorder data.

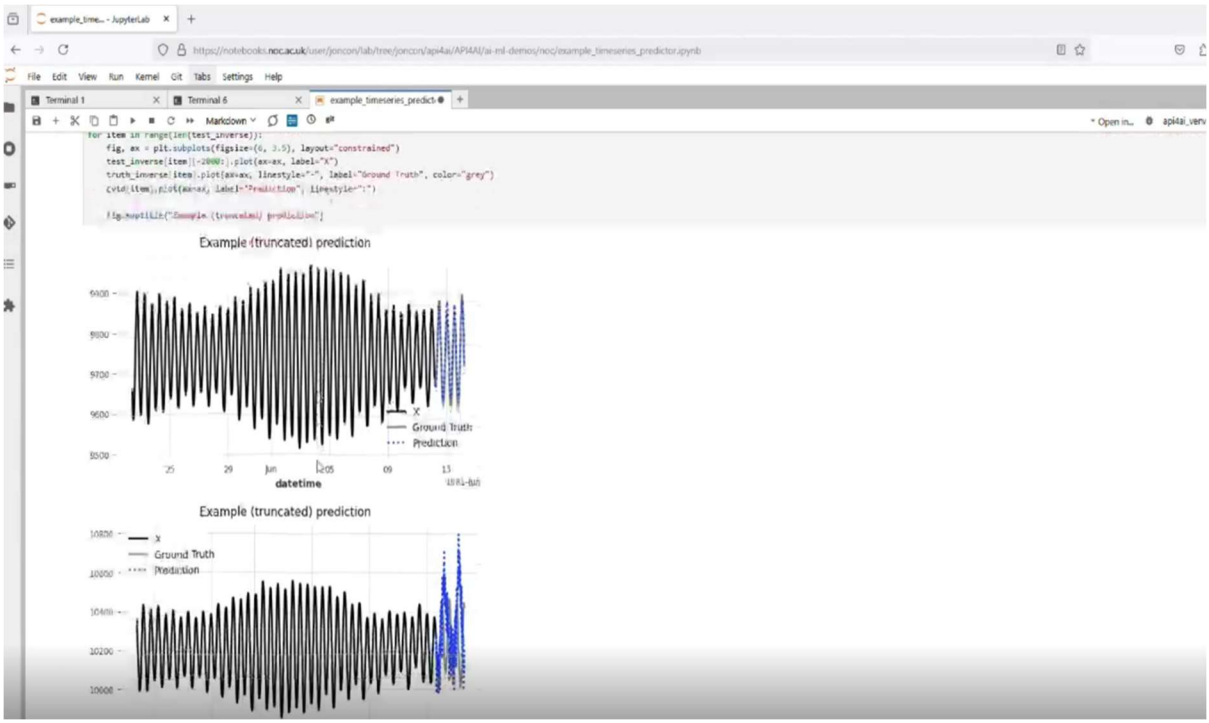


Figure 7, NOC, Python notebook: training a model to forecast sea floor pressure

Proof of Interoperability & Portability

Both UKCEH and NOC have their own platforms for executing AI/ML workflows: DataLabs and the Data Science Platform (DSP), respectively. To demonstrate portability

and interoperability of AI/ML workflows, both organisations were required to gather evidence of executing demonstration AI/ML workflows from the other organisations (BGS, UKCEH, and NOC).

Much of the interoperability was achieved through the use of analogous technology stacks, specifically Python-based environments. The adoption of Croissant further enhanced this interoperability by abstracting data access details and structure. This allows data scientists to seamlessly integrate existing workflows and datasets with Croissant descriptor files, enabling them to adapt these resources to their specific use cases without needing to manage technical complexities.

Additionally, Croissant provides easy access to contextual information, such as data lifecycle, versioning, labelling, AI safety, fairness, and compliance. This context is presented in a standardised format (JSON-LD), ensuring that essential details are readily available and can be consistently utilised across different workflows.

UKCEH, DataLabs

DataLabs is a virtual lab environment hosted through UKCEH. It provides a secure online environment to access, create and run analysis. It provides powerful analytical resources including direct access to large datasets, parallel computing, collaborative and publication tools.

The images below demonstrate the interoperability and portability of croissant-based AI workflows. Each of the workflows created by each data centre could be successfully hosted and executed within the DataLabs environment.

BGS Croissant Notebook:

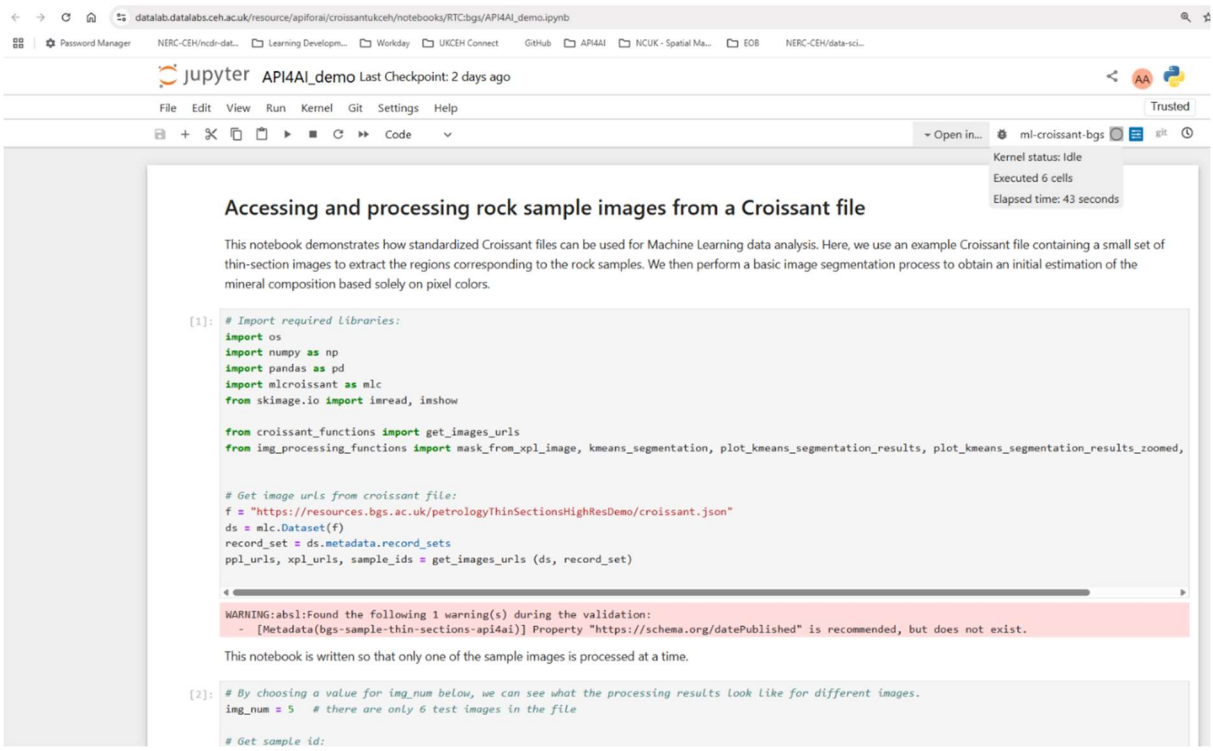


Figure 8, BGS Python notebook, executed on UKCEHDataLabs

NOC Croissant Notebook:

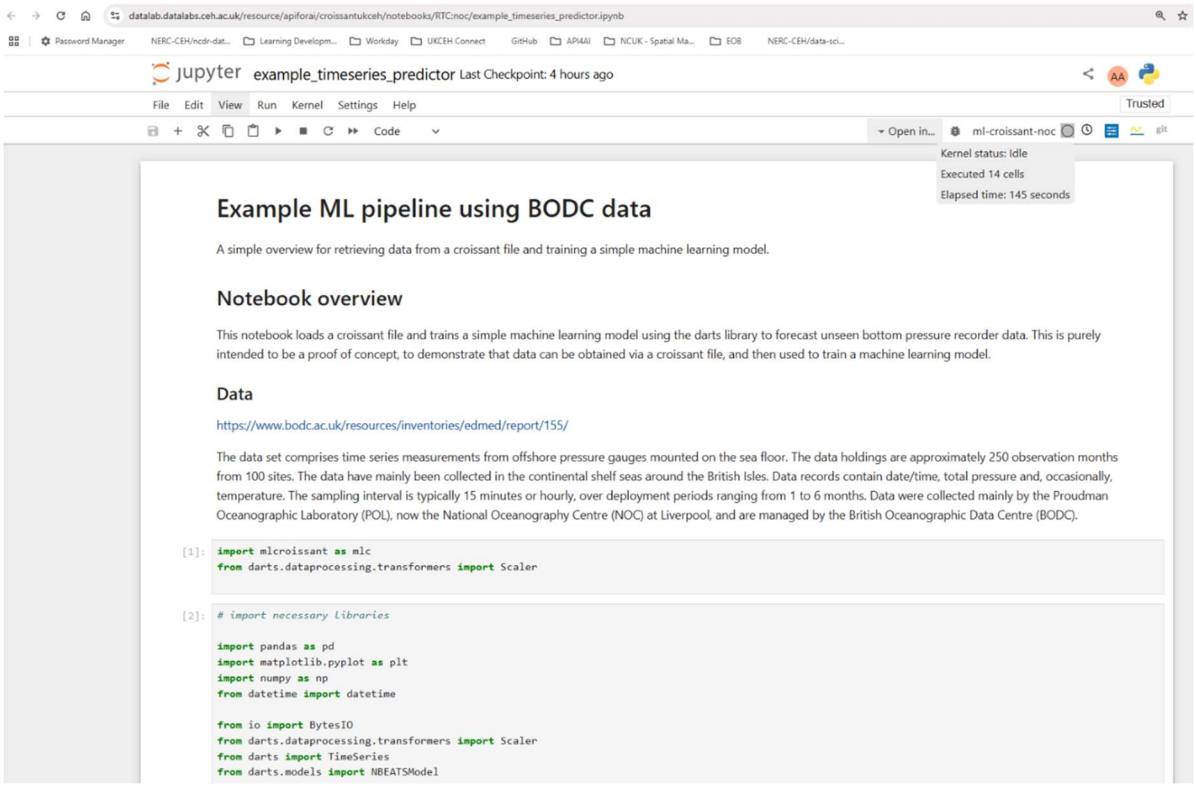


Figure 9, NOC Python notebook, executed on UKCEH DataLabs

UKCEH Croissant Notebook:

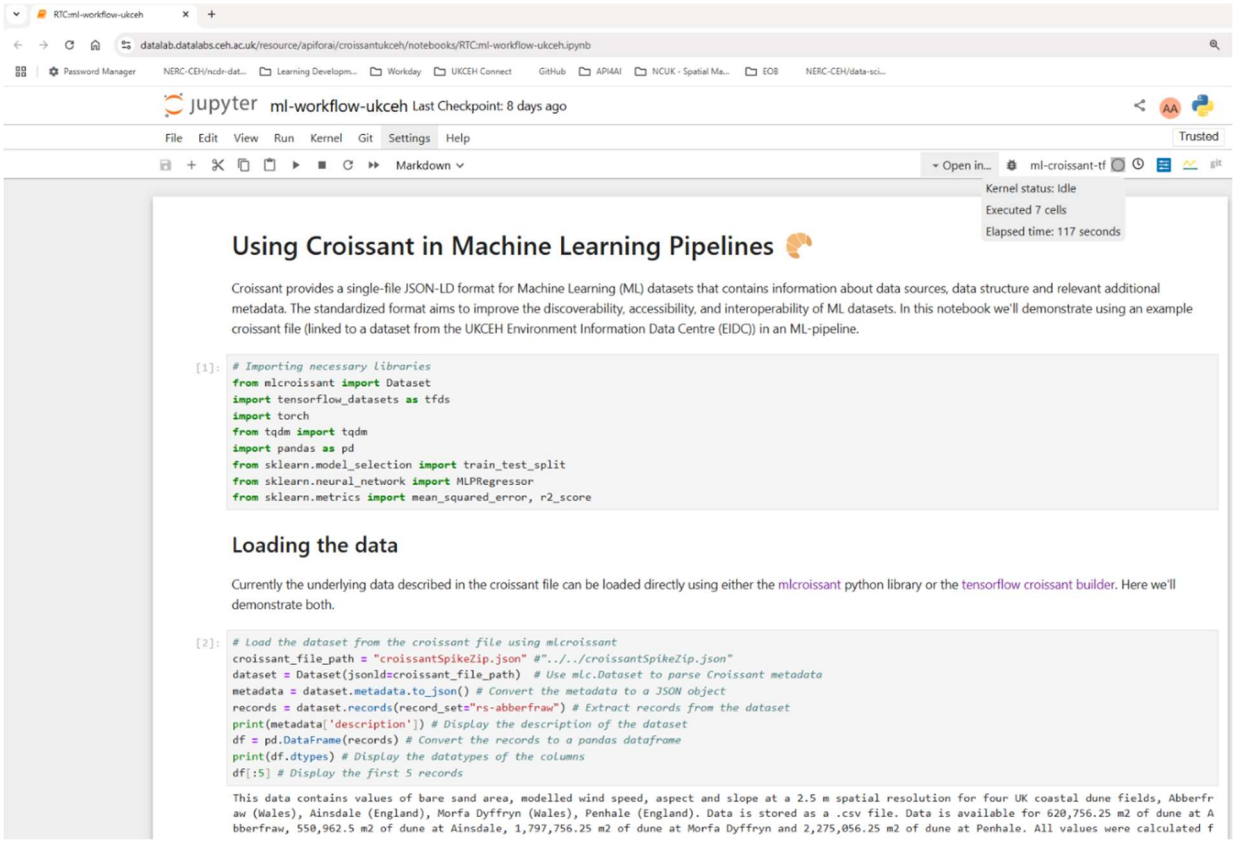


Figure 10, UKCEH Python notebook, executed on UKCEH DataLabs

NOC, Data Science Platform (DSP)

The NOC Data Science Platform (DSP) is a computing environment based on Jupyter notebooks currently hosted on-premises, but using cloud-ready architectures. The DSP is designed to allow scientists a way to access resources through a standard user environment, including pre-configured software environments and containerised workflows. Work is underway to deploy it to other cloud services (such as JASMIN). Going forward the DSP will be given access to run jobs on high memory compute servers, High-performance Computing and GPU's. At the moment the system links in with the IT supported Storage Scale (GPFS) service and we will be looking at adding some additional object storage. These high-performance virtual machines can be accessed on-demand through a web browser, offering flexible, remote computing power.

Again, all croissant-based AI workflows from each data centre were successfully hosted and executed on the DSP environment.

BGS Croissant Notebook:

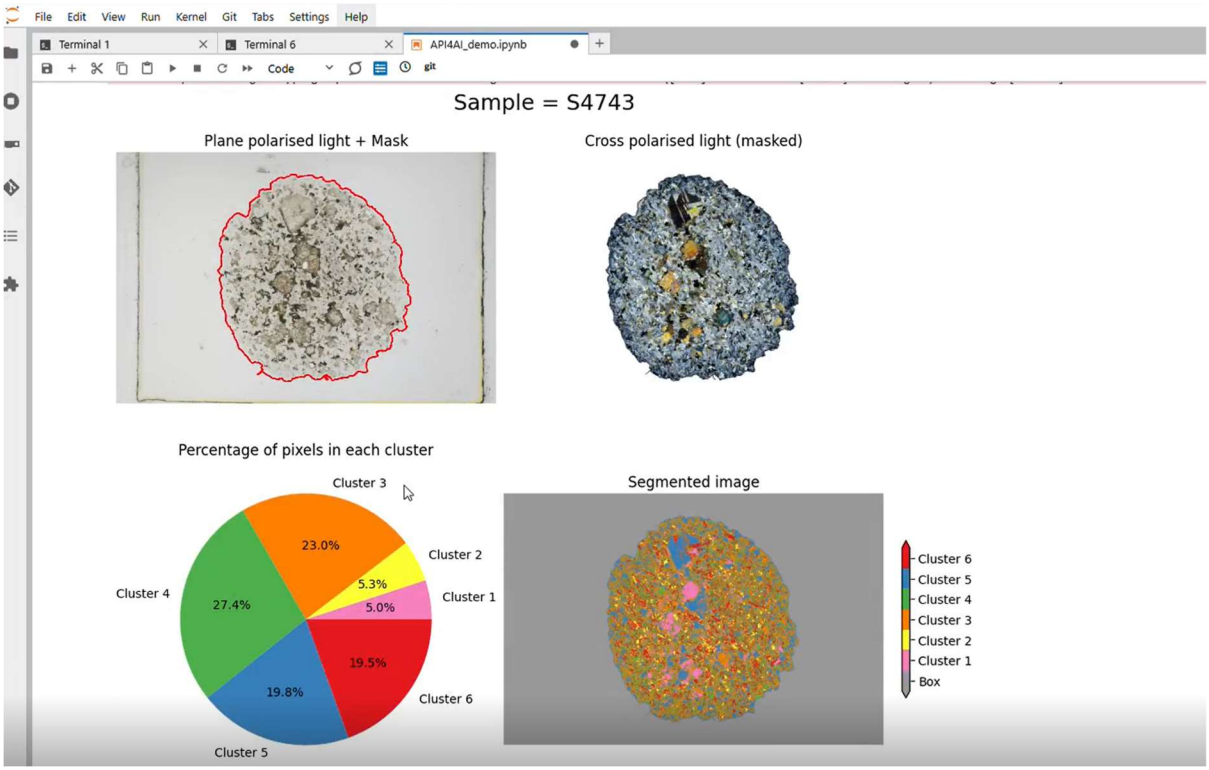


Figure 11, BGS Python notebook, executed on NOC Data Science Platform (DSP)

NOC Croissant Notebook:

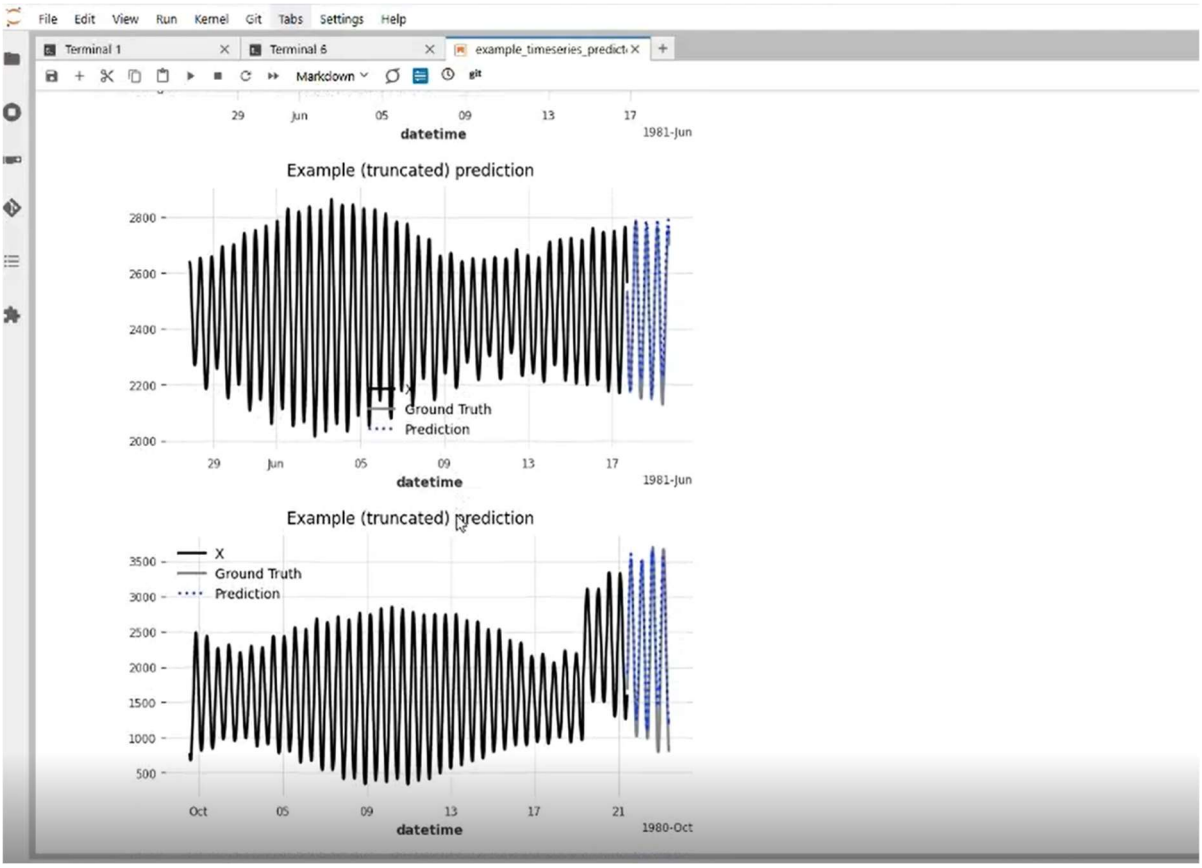


Figure 12, NOC Python notebook, executed on NOC Data Science Platform (DSP)

UKCEH Croissant Notebook:

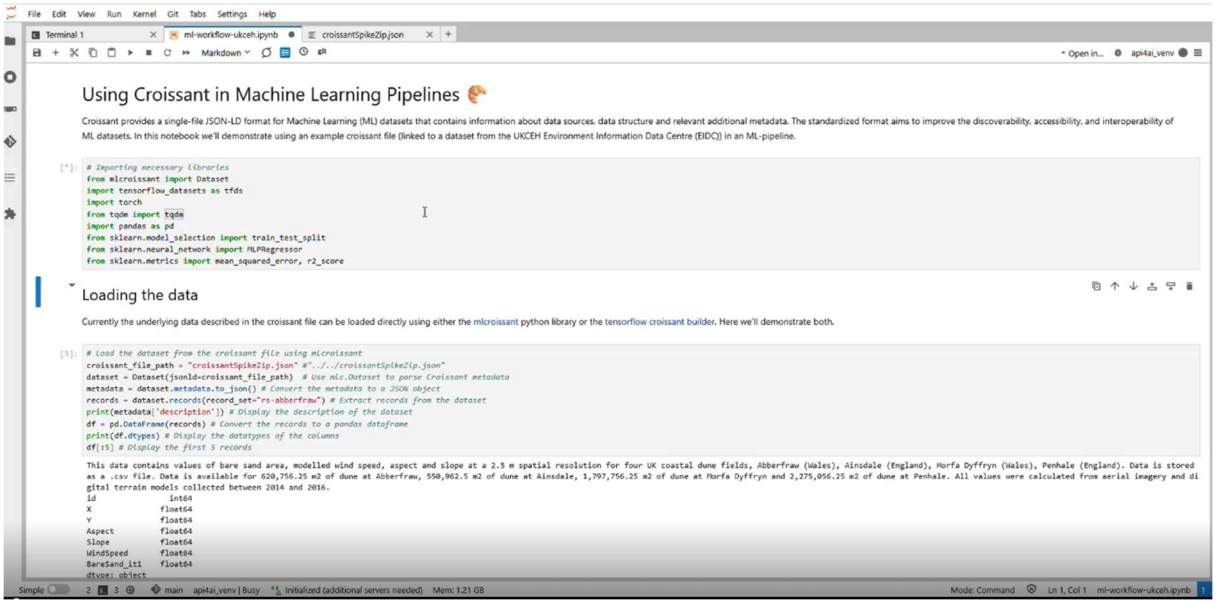


Figure 13, UKCEH Python notebook, executed on NOC Data Science Platform (DSP)

Conclusion

The Agile development methodology used in this project resulted in a different route to achieving the project goals than was originally anticipated in the proposal, and therefore a different set of outputs. Rather than all the Data Centres attempting to standardise on a single API implementation architecture, the pragmatic method of delivering standardisation was achieved through use of the Croissant specification as a common interface between data access APIs and AI workflows, i.e. standardisation was achieved at the metadata level rather than the data format level. The guidelines for API implementations therefore could allow a greater degree of flexibility and accommodate different data models and existing architectures in use.

The Croissant metadata standard is a positive step forwards for describing ML-ready datasets. It facilitates:

- Comprehensive descriptions of datasets, including their metadata, structure, content, and provenance.
- Licensing information to clarify data usage rights.
- Bias and fairness documentation to support ethical ML development.
- Storage and structure representation to aid interoperability.

Croissant metadata is an extension of standardised metadata already implemented by the Data Centres, and some real progress was made towards automating the generation of Croissant using content taken from existing metadata sources. The RecordSet elements of Croissant are beyond that provided by any discovery metadata standards, but are provided in OpenAPI response descriptions for APIs, so there is scope for further automation.

Python notebooks from each of the Data Centres demonstrated successful ingestion of data APIs into a ML workflow, via a Croissant metadata description and the python croissant library. All notebooks were run on UKCEH DataLabs platform and NOC Data Science Platform, demonstrating the interoperability of computing environments and portability achieved through standardisation.

Issues with Croissant

Some difficulties found were that Croissant documentation provides limited or no help for how datasets accessed via API should be described; it seems to be primarily intended for datasets accessed by a bulk archive download file. It currently lacks built-

in mechanisms to fully represent the storage and formatting of high-dimensional Earth system data.

Earth Observation (EO) and climate model datasets are often multidimensional, requiring extensive metadata to describe their spatial, temporal, and variable aspects. Their distributed nature and storage complexity make integration into machine learning (ML) workflows challenging, often requiring data from a multitude of files.

Geo-Croissant (i.e. geospatial enabled) is emerging as a solution to bridge Croissant with large-scale cloud-native geospatial datasets. By aligning Croissant metadata with geospatial data descriptions, it aims to improve efficiency in handling EO and climate model datasets.

Development of new data APIs in this project highlighted existing tensions within the Data Centres and Institutes between the desire embedded within UKRI funding bodies for open data, highlighted by the NERC data policy [30] to increase the accessibility and re-use of datasets, the requirement on institutes / data centres to achieve cost recovery through data licencing and the reluctance of individual scientists to hand over high value data assets they have been responsible for developing to unknown users. For example, the BGS data API serves only medium resolution version of geological thin section images rather than the high-resolution versions. This tension could be resolved through access control for example through API registration and authentication. However, Croissant can't currently handle authenticated access to datasets. This is subject to corporate level decision making.

Summary of outputs

In addition to this report, the project produced the following outputs:

- **Technical outputs from this project can be found on GitHub [31]**
    - Proof-of-concept demonstrators (Python notebooks) showcasing the use of each API within an AI workflow via a Croissant descriptor.
    - Publication of croissant descriptors of AI suitable datasets from each of the partner Data Centres
    - Guidelines on how to implement a standardised interface between APIs and AI workflow using croissant
    - Utility scripts to generate part of Croissant file from database object definitions, and convert schema.org metadata to Croissant
- API4AI Workshop report [6]
- Implementation of new data API by BGS [5], see figure 1.

- Extension of existing CEH metadata API to serve croissant format for selected datasets [47]
- Evidence of each proof-of-concept demonstration AI workflow (Python notebook) being successfully executed on both the NOC DSP and UKCEH DataLabs AI platforms (see figures 4 to 13).

Recommendations

Recommendation API4AI-R1: Allow NERC EDS Data Catalogue metadata to be indexed by search engines

This configuration change in the online catalogue software will mean search engines like Google Dataset Search can crawl and index the schema.org metadata in the catalogue and make the information available to web searches and AI agents, increasing the visibility and widening use of the data.

Recommendation API4AI-R2: Implement improvements to metadata as defined in NERC Metadata Guidelines V2.0.

These improvements will ensure all datasets, and data delivered via API, are described in with quality metadata in a consistent way across the NERC Data Centres and will provide the foundation for enhanced metadata needed specifically for APIs in AI /ML use cases.

Recommendation API4AI-R3: Publish a description of your dataset and how to access it using the Croissant specification

One of the intended objectives was to develop an API specification and reference implementations to guide the creation of APIs for AI workflows. However, through the consultation with data centres to discover existing practices and architectures for data delivery APIs, it became clear that the data centres already had mature systems for data delivery that would be hard to harmonise at the API implementation level. Standardising the _interface_ between APIs and AI workflows would be more beneficial than attempting to prescribe a single uniform structure for all APIs.

Instead, the recommendation is to focus on a higher level of standardisation—the interface between APIs and AI/ML systems. Croissant is a new standard with good uptake in the ML community to bridge this gap effectively.

Recommendation API4AI-R4: Use an API implementation standard appropriate to your dataset, described using OpenAPI specification

When it comes to API development, our recommendation it to follow (current) industry best practices, technologies, and standards while ensuring they are flexible enough to provide data that can be seamlessly integrated into AI workflows. APIs should be described using OpenAPI specification.

Future work

Tasks directly identified for potential future work are:

Croissant metadata

- Share findings of this project with the mlcommons Croissant working group
- Participate in the mlcommons Croissant working group
- Extend croissant specification or documentation to make it more applicable for data access though API, and through authenticated access
- Review outcomes of geo-croissant [16] working group and test on large geospatial datasets
- Investigate automated creation of part of Croissant metadata from OpenAPI specification descriptions of data access APIs

- Data access
  - Consider creation and delivery of bulk downloads alongside API where appropriate
  - Investigate and agree a shared approach and mechanism within EDS for API registration and authentication
  - Enable semantic interoperability by defining data attributes using linked data definitions, e.g. from the NERC Vocabulary Server
  - Follow UKCEH lead to update metadata catalogues capabilities to make Croissant metadata available as a download format option where applicable

AI/ML workflows

- Create API endpoints or data downloads of statistical data describing distribution of labels and attributes used in data to indicate clustering and degree of balance; this is helpful when using data attributes in classification tasks.
- Create a capability in BGS equivalent to UKCEH's DataLabs and NOC's Data Science Platform

- Investigate use of Croissant in workflow platforms to record at a granular level how a dataset was created, processed and enriched throughout its lifecycle, enabling the Responsible AI metrics to be shared.
- Investigate OGC/ISO standard for sharing labelled training data and implement examples

Appendices

Existing relevant data and metadata standards

OGC Web Services (Traditional)

- Web Map Service (WMS): For visualising georeferenced map images.
- Web Feature Service (WFS): For querying and updating vector geospatial features.
- Web Coverage Service (WCS): For accessing raster and gridded data.
- Web Map Tile Service (WMTS): For tiled map delivery, enhancing scalability.

OGC API Standards (Next Generation)

- OGC API – Features: RESTful access to vector features (alternative to WFS).
- OGC API – Tiles: Efficient map tile delivery (replacement for WMTS).
- OGC API – Coverages: Improved access to raster and gridded data (alternative to WCS).
- OGC API – Records: Geospatial metadata discovery and cataloguing.
- OGC API – Environmental Data Retrieval (EDR): Lightweight access to environmental spatio-temporal data.

Other API standards

- STAC - (SpatioTemporal Asset Catalog) API is a standardised way to search, access, and share geospatial data, particularly satellite imagery, remote sensing data, and other spatial datasets [43].
- Kerchunk enables lightweight virtual indexing of chunked Zarr [48] and NetCDF [45] datasets, allowing efficient data access without duplication [23].

Data Formats

- JSON: Lightweight data interchange format.
- JSON-LD: Linked data representation for enhanced interoperability.
- GeoJSON: Encoding geographic data structures using JSON.
- Well-Known Text (WKT) & Well-Known Binary (WKB): Geometry representation in spatial databases.

- GML: Flexible format for OGC-compliant services.
- netCDF: Multi-dimensional array storage for scientific and geospatial data.
- Zarr: a cloudy ready open standard for storing large multidimensional array data
- Cloud-Optimised GeoTIFF (COG): Cloud-native raster data for efficient streaming.
- Apache Parquet: Columnar storage for big data processing and machine learning.
- Apache Arrow: Columnar memory format for efficient data analysis.

Metadata schemas

- ISO19115/ ISO19139 [19,20]
- UK GEMINI [1]
- INSPIRE [13]
- MEDIN [25]
- Schema.org [41]
- DCAT [11]
- CSW: OGC Catalog Service for Web for federated discovery [35]

Standards for ML training data

- Croissant: Data format for standardised machine learning workflows. [26]
- GeoCroissant: Extension of Croissant for geospatial machine learning, addressing spatial specific requirements [16]
- Draft: ISO 19178, "Training data markup language for artificial intelligence" [21]
- OGC TrainingDML-AI “Training Data Markup Language for Artificial Intelligence” [36]

References

[1] AGI, GEMINI, https://www.agi.org.uk/groups/agi-gemini/

[2] Agile Alliance https://www.agilealliance.org/agile101/

[3] Blair G S etc al, 2024 *Contributions to the development of the next-generation NERC Environmental Data Service: Building Interoperability - a NERC Data Commons RoadMap*. UK Centre for Ecology and Hydrology, 123pp. (Unpublished)

[4] BGS OGC API Service https://ogcapi.bgs.ac.uk/

[5] BGS. Generic Samples Register https://ogcapi.bgs.ac.uk/collections/generic-samples-register

[6] Booth, J.; Bell, P.; Kingdon, A; Heaven, R.E.; Card, C.; Sauze, C.; Tso, M. 2025 NERC Environmental Data Services: API 4 AI. Project kick-off workshop report. NERC, 38pp. (OR/25/019)) https://nora.nerc.ac.uk/id/eprint/539107/ (Unpublished)

[7] BS EN ISO 19178-1 Geographic information — Training data markup language for artificial intelligence —. Part 1: Conceptual model https://standardsdevelopment.bsigroup.com/projects/2024-00454

[8] BGS OGC API https://ogcapi.bgs.ac.uk/

[9] CEDA STAC Catalogue: https://stac.ceda.ac.uk/collections/cmip6?.language=en

[10] Data.gov.uk https://www.data.gov.uk/

[11] DCAT https://www.w3.org/TR/vocab-dcat-3/

[12] ESRI, ArcGIS Server https://enterprise.arcgis.com/en/server/

[13] European Commission INSPIRE data catalogue https://inspire-geoportal.ec.europa.eu

[14] Geospatial Commission. 2020. Finding Geospatial Data. https://www.gov.uk/government/publications/finding-geospatial-data (accessed 2025/04/03)

[15] Google Dataset Search https://developers.google.com/search/docs/appearance/structured-data/dataset

[16] GRSS-IEEE GeoCroissant- A Metadata Framework for Geospatial ML-ready Datasets. 2024. (webinar) https://www.grss-ieee.org/events/geocroissant-a-metadata-framework-for-geospatial-ml-ready-datasets/

[17] HuggingFace https://huggingface.co

[18] IFCB, Imaging Flow CytoBot, A Flow Cytometry device from McLane Laboratories https://mclanelabs.com/imaging-flowcytobot/

[19] ISO 19115-1:2014 https://www.iso.org/standard/53798.html

[20] ISO 19139-1:2019 https://www.iso.org/standard/67253.html

[21] ISO/FDIS 19178-1 Geographic information — Training data markup language for artificial intelligence Part 1: Conceptual model https://www.iso.org/standard/89050.html

[22] Kaggle https://www.kaggle.com/

[23] Kerchunk specification: https://github.com/cedadev/stac-notebooks

[24] LISST-Holo, A Holographic camera from Sequoia Scientific https://www.sequoiasci.com/product/lisst-holo/

[25] MEDIN data portal https://medin.org.uk/

[26] MLCommons, Croissant Specification 1.0 https://docs.mlcommons.org/croissant/docs/croissant-spec.html

[27] MLCommons, Croissant RAI Specification 1.0 https://docs.mlcommons.org/croissant/docs/croissant-rai-spec.html

[28] MLCommons, mlcroissant library https://github.com/mlcommons/croissant/tree/main/python/mlcroissant

[29] NASA https://www.earthdata.nasa.gov/news/blog/introducing-croissant-format-machine-learning-datasets

[30] NERC Data Policy https://www.ukri.org/who-we-are/nerc/our-policies-and-standards/nerc-data-policy/

[31] NERC-EDS GitHub API4AI repository https://github.com/NERC-EDS/API4AI/tree/main/

[32] NERC EDS, Environmental Data Services: NERC Guidance for authors of discovery metadata, version 2.0, NERC. 2024 .53pp. (Unpublished)

[33] NERC EDS metadata catalogue. https://data-search.nerc.ac.uk

[34] NOC, CRAB https://github.com/NOC-OI/crab

[35] OGC, Catalog Services for the Web (CSW) Standard https://www.ogc.org/publications/standard/cat/

[36] OGC, OGC Training Data Markup Language for Artificial Intelligence (TrainingDML-AI) Standard https://www.ogc.org/publications/standard/trainingdml-ai/

[37] OGC, OGC APIs https://ogcapi.ogc.org/

[38] OpenAPI, OpenAPI Specification https://www.openapis.org/

[39] pygeoapi https://pygeoapi.io/

[40] Research Object. RO-Crate https://www.researchobject.org/ro-crate/

[41] Schema.org https://schema.org/https://schema.org/

[42] Spring, Spring Boot https://spring.io/projects/spring-boot

[43] STAC, SpatioTemporal Asset Catalogs https://stacspec.org/en

[44] TensorFlow https://www.tensorflow.org/https://www.tensorflow.org/

[45] UCAR, NetCDF https://www.unidata.ucar.edu/software/netcdf/

[46] UKCEH Environmental Information Data Centre (catalogue) https://catalogue.ceh.ac.uk/eidc/documents

[47] World Wide Web Consortium (W3C). 2017. Data on the Web Best Practices. https://www.w3.org/TR/2017/REC-dwbp-20170131/

[48] Zarr, chunked, compressed, N-dimensional arrays https://zarr.dev/