Data resource profile: Alspac datacatalog schema and Alspac omics datacatalog

Sam Neaves

March 2023

1 Introduction

The Avon Longitudinal Study of Parents and Children (ALSPAC) (1) (2) is a long-term cohort study that has been collecting data on the health and development of children and their families since 1991. In recent years, ALSPAC has collected a large amount of omics data, including genetic array, methylation, and gene expression data. This data, combined with the deep phenotype data collected as part of the study, allows researchers to investigate the genetic factors that influence disease and well-being, and to use techniques such as Mendelian randomization to identify effective health interventions.

To make this data more accessible to researchers, ALSPAC has developed a data catalog schema and an omics data catalog using LinkML (3), which is presented as Linked Open Data (LOD) (4) in a web-accessible triple store. The catalog documents the metadata for the omics data, and LOD allows data from different sources to be connected and integrated, enabling new insights and discoveries. This is becoming increasingly important as researchers seek to integrate data from various sources, such as other longitudinal cohort studies, trials, routine health data, and commercially collected data, to maximize statistical power and answer scientific questions.

The use of LOD and LinkML in the ALSPAC data catalog has several advantages. It facilitates the integration and linking of data from multiple sources, making it easier to find and access the data. Users are able to navigate and visualise the data as knowledge graph, which aids discoverability. It enables the creation of flexible and extensible data models that can represent complex and evolving data. And it allows the use of standardized and interoperable technologies, enabling the data to be used by a wider range of tools and applications. Overall, the use of LOD and LinkML has improved the accessibility and interoperability of ALSPAC's omics meta data and offers significant advantages for researchers working with this rich and complex dataset.

2 Aims and Implementation

2.1 FAIR data aims

In implementing the omics data catalog, a primary objective is to ensure the metadata adheres to the FAIR data principles (5). These principles, which stand for Findable, Accessible, Interoperable, and Reusable, promote efficient and effective research and decision-making by facilitating the discovery and utilization of data for researchers and other stake-holders.

Findable: Data should be easy to locate and search for. This requires the use of clear and standardized metadata, as well as the use of persistent identifiers such as Digital Object Identifiers (DOIs) and Persistent uniform resource locatators to uniquely identify data sets.

Accessible: Data should be easy to access and retrieve, regardless of the user's location or technology. This requires the use of open and standardized protocols, as well as the use of clear and consistent licensing terms.

Interoperable: Data should be easily integratable with other data sources. This requires the use of standardized data formats and APIs, as well as the use of common data models and ontologies.

Reusable: Data should be able to be used and reused for multiple purposes. This requires the use of clear and concise documentation, as well as the use of standardized and open data licenses that allow for the reuse of the data.

Overall, the FAIR data principles are important because they help to make data more discoverable, usable, and valuable, which can lead to advances in research and other fields.

2.2 Data security and privacy aims

Another critical goal with the ALSPAC omics data catalog is that the actual data (as opposed to the metadata) remain secure and private and is only released to approved researchers for approved research.

Importantly having well cataloged and documented omics data helps to ensure that the security of the data can be maintained. For example a comprehensive data catalog can help ensure data security and compliance with internal policies, the General Data Protection Regulation (GDPR) (6) and ISO27001 (7)in several ways:

Data discovery: A comprehensive data catalog makes it easy to discover and understand all of the data that ALSPAC has. This can help ALSPAC identify where sensitive data is located and take steps to protect it.

Data lineage: A comprehensive data catalog can also track the lineage of data, so the ALSPAC can understand where data came from, who has access to it, and how it is being used. This can help ALSPAC identify and address any potential compliance issues, such as data that is being used in ways that are not compliant with the GDPR or approved research.

Access control: A data catalog can also help ALSPAC control access to omics data. The catalog can inform access controls so that we ensure that only authorized users can access omics data and that they are only using it for approved purposes.

Auditing and reporting: A data catalog can also help ALSPAC with auditing and reporting. The data catalog can record all access and usage of sensitive data, allowing ALSPAC to more easily produce reports and evidence of compliance if audited.

Data Governance: A comprehensive data catalog can be a assistance to data governance activities, by acting as central point of the data. This can help governance activities that include policies, standards, procedures for data management, data quality and data lineage.

By having a comprehensive understanding of all the data and how it is being used, a data catalog can help ALSPAC ensure that they are complying with internal policies, the GDPR and other regulations related to data security and privacy.

2.3 Data management

With the goal of making the omics data catalog FAIR and comprehensive in order to maintain security and aid research it was necessary to design how the existing omics data can be organised and cataloged as well as designing systems for injesting new data into the omics data stores alongside their metadata for the catalog as well as documenting how and what data is shared with approved researchers.

ALSPAC is a research project that has adapted to the changing needs of the scientific community over time. It has a data management plan (`https://www.bristol.ac.uk/media-library/sites/alspac/documents/researchers/data-access/alspac-data-management-plan.pdf`) that details its overall plan for managing data.

This includes how ALSPAC uses Standard operating procedures (SOPs) which are important for managing omics data for a number of reasons:

Quality assurance: The SOPs provide a set of guidelines for how data should be collected, processed, and analyzed, which helps to ensure that the data is of high quality. This is especially important in the field of omics, where large amounts of complex data are generated.

Reproducibility: SOPs enable researchers to reproduce the results of a study by providing a clear set of steps that were followed during the research process. This is critical for building trust in the scientific community and for advancing knowledge in a field.

Efficiency: SOPs can help to streamline data management processes, making them more efficient and reducing the risk of errors. This is especially important in large research projects where there may be multiple researchers working with the data.

Compliance: SOPs can help to ensure that an organization is compliant with relevant regulations and standards, such as those related to data privacy and security.

Overall, using SOPs helps to promote transparency, reliability, and consistency in the management of omics data.

The ALSPAC data catalog schema and the omics data catalog can act as a glue between different SOPS, practice on the ground and the data. Having formally documented requirements for metadata and descriptions of data entities as well as using semi automated tools for validating and injesting data aids the goals of the SOPS. For this reason the data catalog schema and the omics data catalog are integrated into relevant ALSPAC SOPs as key component of managing and organising the data. The schema and catalog aid clarity and detail in the SOPS around how data is organised which aids data security and discovabilty.

2.4 General requirements for organising the data

As new data collection and processing/generation techniques become available, ALSPAC has to incorporate the incoming data into the study in a number of ways.

One way is through internally-initiated data collection, such as the decision to genotype original participants in 200X, with biological samples being sent to an external lab and the resulting data returned to the study. A second way is through internal processing of existing data into a new format or the creation of new files from existing data, such as converting genotype array data from plink format to bgen format or deriving genetic principle component data.

A third way that ALSPAC receives additional data is from researchers who are MRC Integrative Epidemiology Unit (MRC-IEU) direct users of the data, ALSPAC works closely with staff members from the MRC_{IEU}. And these researchers often produce new data that is returned to ALSPAC. For example the DNA methylation data from the ARIES study (8).

A fourth is by data being returned by external researchers for example x, y,z.

Other potential ways data could be incorporated include data linkage to other studies or commercial databases.

A challenge with this constantly evolving study is keeping the data organized as it creates and ingests new data. This is also constrained by the IT infrastructure that is made availble by the University of Bristol to support the study. The IT infrastructure includes secured backed up data storage in the research storage facility (RDFS) as well as High performance computers Blue Crystal and Blue Pebble for data processing.

These different parts of IT infrastructure perform different roles and due to the large size of the omics data, multiple copies of the data need to be stored across these parts.

For example in order to process omic data files BC4 compute nodes can neither read or write directly to the RDSF. Special login nodes are used for these purposes.

Another important way that ALSPAC generates new data files is in the process of making the data availble to external researchers. A core idea with ALSPAC is that data can be applied for and reused by further researchers. However it is also a requirement of ALSPAC data governance that research is pre approved and that the smallest amount of data be shared. In general hypothesis free research is not supported. It is also a requirement that data released has participants who have withdrawn consent removed from the data that is distributed to the external researchers.

Additionally when releasing omics data to approved collaborators, efforts are made to anonymize and De-identify the data, as ALSPAC data is particularly sensitive. This is done to support researchers using the data in sticking to there approved research area by making it difficult/convoluted to process the data with unapproved linkages. Therefore, due to the constant production of new heterogeneous data and the constraints of working with IT infrastructure, multiple copies of the data are required, and the need for producing datasets for external researchers, many different related datasets are produced and held by ALSPAC. It is important to record how these datasets are created or brought into the study and how they differ amongst themselves.

There are therefore a few challenges in order to keep the data organized over time, these include:

Data growth: As data is added over time, it is necessary to plan and structure the file system and to add the required metadata on the files. Without a clear hierarchy of folders and associated metadata, it becomes difficult to find specific files or folders.

Data retention: As data accumulates, it is necessary to retain certain data for longer periods of time, This can lead to challenges in terms of storage capacity and data management. For example with different copies of data as different snap shots. Especially with the large nature of modern omics data.

Data backup and recovery: It's important to ensure core data is backed up and it is possible to recover it in the event of a disaster. This is challenging because of the large amount of data that needs to be backed up regularly.

Data security: As data accumulates over time, it becomes increasingly important to understand what data is kept and its relationships in order to ensure that it is secure.

Data access and collaboration: As data in ALSPAC grows and evolves, it is important to manage access to data effectively and ensure that it is being used in a way that is consistent with the internal policies. This is especially challenging as there are a large number of users who need to access and collaborate on data.

The omics data catalog and data catalog schema aim to aid the organisations in meeting these challenges.

2.5 Previous data organisation issues

The challenges in data management and organization previously led to difficulties in utilizing the previously existing systems and procedures. Data was loosely organized with ad-hoc and deep file hierarchy structures. The directory structures were being used as metadata but in an inconsistent way. There were also no formal requirements or guidelines for associated meta-data, leading to inconsistencies and difficulties in understanding the relationships between data sets. A significant issue was the lack of a clear concept of a data set, with informal references to datasets such as "1000 genome data" or "round 2 data," but no clear information on the specific files or number of files that constitute these datasets. This led to ambiguities in understanding what data was available and how it related to other data. The previous issues with data organization included a lack of understanding of the specifics of the data, such as the number of files, their location, format, and relationships to other data sets; as well as difficulties in determining whether files in different locations are identical or different or a derived dataset; and a lack of information on the processing that has been undertaken on a dataset, the people who have worked on it, and the scripts used to create different datasets.

2.6 Why have a schema for the metadata for the omics data?

To effectively manage and organize the metadata for omics data and enable efficient querying, we implemented a data modeling process to formally describe the data we wanted to store and the relationships between them. The result of this process is the data catalog schema.

We started by thinking about a number of questions about the proposed schema. These included:

1. What is the purpose of the schema?
2. What entities and relationships are represented in the schema?
3. How is data modeled in the schema (e.g. as tables, columns, and rows)?
4. What data types are used for each attribute ? (e.g. integers, strings,dates)
5. Are there any constraints or rules that must be followed when storing data in the schema (e.g. unique constraints, foreign keys)?
6. Are there any null values allowed in the schema, and if so, how are they handled?
7. How is data integrity maintained in the schema (e.g. through the use of primary and foreign keys)?
8. Are there any performance considerations to be aware of when using the schema (e.g. indexes or partitioning)?
9. How is the schema versioned and managed over time?
10. How is the schema documented, and how to keep the documentation up to date?

These questions allowed us to assess different technologies for creating a schema for the omics data catalog.

The most important of these questions is the first one. What is the purpose of the schema? In order to answer this it is useful to think about the questions we we want to ask about our data.

1. What data sets do we have? i.e. what collection of files should constitute a dataset that we want to reuse, refer to, and distribute to collaborators.
2. Who and how a dataset was made.
3. What versions of each dataset do we have? Individual datasets may evolve and change as errors are found and corrected or different technologies allow formats etc to change or
4. How do different data files relate to each other?
5. What size disk-space is required for a data file or collection of data files?
6. What data should someone use for there research or for processing a new dataset?
7. What processing and quality control has occurred on this data?

These questions and others allowed us to understand the purpose of the schema.

The technology we choose to model our schema in is Linkml (3), this was chosen because of the features offered. These are discussed in the following section.

Alternative technologies or techniques such as using a SQL database or writing RDF (9)directly were discounted, because they lacked these features. For example writing RDF directly can be difficult for humans due to the syntax. A disadvantage of a SQL database compared to linkml is that it is typically a centralized store, the (meta)data can not sit or travel with the data as simple small files. Some other potential disadvantages of SQL include limited data modeling (tables with fixed columns for example), lack of flexibility (Altering the schema can be a time-consuming and error-prone process that can cause data loss or inconsistency), poor support for unstructured data (e.g text), limited query capabilities (typically limited to simple JOIN operations and basic aggregation function rather than semantic inference), and lack of interoperability.

3 Linkml

LinkML (3), is a versatile modeling framework designed to facilitate collaboration between humans and computers. It is platform-agnostic, and it can be compiled down to RDF. It is user-friendly for experts in both technical and domain-specific fields. LinkML enables the development of YAML-based schemas to define data structures, offering features such as simplified schema generation, support for inheritance hierarchies, semantic enumerations, and compatibility with RDF tools. Moreover, it can produce documentation and web-sites, ensuring adaptability and seamless integration with multiple frameworks, such as JSON and RDF.

One of the advantages of LinkML's schema authoring is the use of YAML files, which are easy for people to read and write, support comments, and can be version controlled and distributed separately. Additionally an important feature is that LinkML makes it simple to utilize ontological terms for your data, meaning that your data can be consistent and interoperable with existing and future data sources, by using standard linking ontologies.

LinkML also makes it easy to work with RDF triples, by providing tooling to transform data from different formats to RDF. RDF triples are a basic unit of information in the RDF data model. They consist of a subject, predicate, and object, and provide a simple and flexible way to represent and exchange information on the web. By linking RDF triples, complex networks of interlinked data, known as the "linked open data" (LOD) cloud, can be created.

Additionally, LinkML encourages the reuse of existing ontologies and ontology terms in data schema, which can ensure consistency and interoperability, promote the sharing and reuse of data within the scientific community, and improve the overall quality of the data schema. Examples of ontologies and vocabularies that we used in our schema include DCAT (10), the Nepomuk File System Ontology (11), and Simple Knowledge Organization System (skos) (12).

In addition to its modeling capabilities, LinkML also offers advantages in project management through the use of the provided project cookie cutter template which is managed by cruft (13) and the Poetry (14) Python environment.

The cookie cutter managed by cruft allowed for the easy setup and management of the LinkML project on Github, streamlining the process of creating and maintaining the project structure. One useful feature that it provides is the suggestion of a PURL (Persistent URL) for the project, which is used when LinkML converts to RDF. A PURL is a long-term, persistent identifier for a resource, which can be resolved to the current location of the resource.

When LinkML converts to RDF, it uses the PURL as the base URI for the RDF triples, ensuring that the resources represented in the RDF data have a stable and persistent URI. This makes it easier to reference and link to the resources, both within the data and in external systems.

The use of PURLs also helps to ensure that the data remains accessible and identifiable over time, even if the location of the resource changes. This makes it easier to maintain and update the project, and ensures that the data remains accessible to others.

The poetry Python environment, on the other hand, is a dependency management tool that allows for easy installation and management of project dependencies. It creates isolated environments for each project, avoiding conflicts with other projects and ensuring that the correct versions of dependencies are being used. This can help to prevent issues related to version conflicts and make it easier to maintain and update the project.

Together, these tools make it easy to manage and organize the schema project, ensuring smooth development and maintenance, and allowing developers and project managers to focus on the core functionality of the project.

It is accepted that the schema will evolve with time as new requirements come to light. To manage and document these changes, the schema is managed using the version control system Git and the hosting platform GitHub.

Git allows for tracking changes to the schema over time and for the ability to roll back to previous versions if necessary. This can be helpful for debugging and for keeping track of the development history of the schema.

GitHub Pages is used for hosting the documentation, while the generation of the HTML documentation is done by GitHub Actions which have been set up with the LinkML cookie cutter managed by cruft. This automatic generation of schema documentation makes it easy for people to understand and use the schema. The documentation has been made searchable to further improve its usability. The documentation also includes diagrams such as a UML (Unified Modeling Language) class diagram and entity relationship diagram for the schema.

A UMLclass diagram is a type of diagram that shows the structure of a system by representing the classes, attributes, and relationships within that system. Class diagrams are often used to model the static aspects of a system, such as its class hierarchy and the relationships between classes.

An entity relationship diagram (ERD) is a type of diagram that shows the relationships between entities in a database. An ERD typically consists of entity types (such as customers, orders, and products) and the relationships between them (such as a customer placing an order for a product).

Both UML class diagrams and ERDs are useful documentation for the schema because they provide a visual representation of the schema's structure and relationships. This can make it easier for others to understand and use the schema, and can also help with debugging and maintenance. These diagrams are included in the schema documentation to help users understand how the schema is organized and how different elements of the schema are related to one another.

Overall, using a VCS and generating documentation in this way is considered best practice for schema maintenance and development. It helps to ensure the integrity, reliability, and transparency of the schema, and also makes it easier for others to contribute to the development of the schema and for users to understand and use it.

4 Design decisions and modeled entities

We first identified the necessary entities for the schema. To make the schema more versatile, we modeled entities at a higher level than required for our specific use case. This allows for easy adaptation of the schema for future uses. For this reason we model catalogs as entities themselves with the omics catalog being an instance of this class. However our main entity of interest is the 'NamedAlspacDataset', which is a type of dataset we want to name, refer to and reuse. We also have versions of NamedAlspacDataset, as well as "freezes" of these versions, which are subsets of the dataset that we distribute to collaborators when they request data. These freezes contain the core data for the dataset that an external collaborator would need to use the data. They also have different identifiers to our internal datasets and they have people who have with drawn from the study removed. In addition to this we identified that each version of a NamedAlspacDataset version or freeze may have distinct identifiable parts. For example a part might be a set of principle components or a chromosome.

These main classes were complemented by other entities which we model with classes for example people in the person class, which is used to have a minimal way to refer to people who might have built or written documentation for a dataset.

These are depicted in table 1

We then identified the relationships between these entities, as shown in Figure 2 and 3. We identified the attributes of our entities that we wanted to record, and mapped them to existing concepts in well-known ontologies or vocabularies. We used the linkml $\text{class}_{\text{uri}}$ and slot_{uri} meta slots to assign

Class	Description
AlspacDataCatalogue	Represents an alspac data catalogue
AlspacDataSetVersion	Represents a version of a named$_{\text{alspacdataset}}$
AlspacDataSetVersionFreeze	Represents a freeze of a version of named$_{\text{alspacdataset}}$
DataDistribution	A dataset distribution has a location, file type and file size
DatasetPart	Represents a part of named alspac data set, in a version or freeze
KnownIssue	Known issues for a dataset should have a description, when they are logged an…
NamedAlspacDataset	Represents a named$_{\text{alspacdataset}}$
NamedThing	A generic grouping for any identifiable entity
Paper	a scientific paper
Person	A person
QCKeyValue	A qc part with a key and a value
Script	A description and attributes of a script included in a version or freeze
UGKeyValue	A user guide entry

Table 1: Table of classes and descriptions

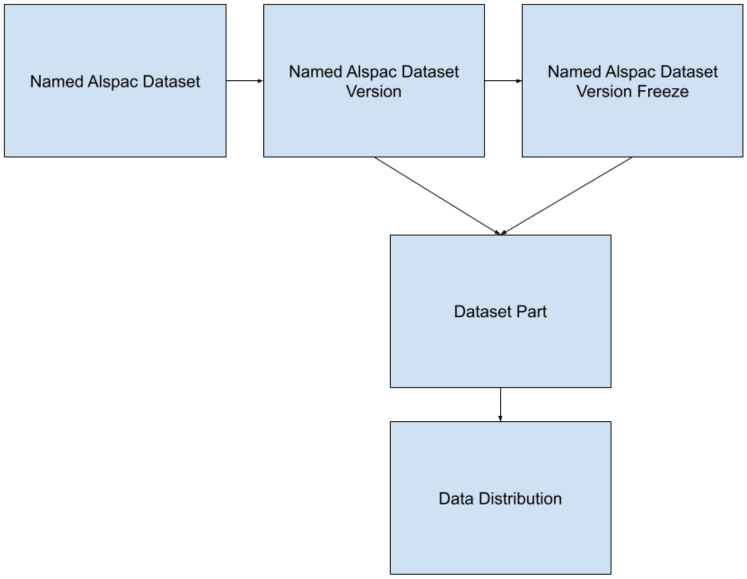

Named Alspac Dataset
Named Alspac Dataset Version
Named Alspac Dataset Version Freeze
Dataset Part
Data Distribution

Figure 1: Main diagram

this information. For classes and slots that did not have exact matches, we used the linkml meta slots x, y, z, which use the skos relationships of exact, broad, narrow, and close match. The main vocabulary used for our data catalog is the DCAT (Data Catalog Vocabulary). In this vocabulary there is a concept of data distribution, A data distribution in DCAT refers to a specific file or set of files that contains data in a specific format and can be downloaded or accessed in some way, while a dataset is a collection of data that can be described by a set of properties and may be made up of one or more data distributions. Therefore to be consistent with this we added data distributions to our schema. This means that anyone familiar with the dcat vocabulary can query our data for information about the datasets. It also means that automatic tools can find the dataset descriptions and facilitates linking the datasets in other catalogs. We added constraints and types where appropriate to the slots, such as marking x as mandatory. Additionally, we designed formats for our ID types and different entities, which can be described by regular expressions. Many times when we need to refer to a file we use slots that will be filled with md5sums which are hash values that uniquely identify a file.

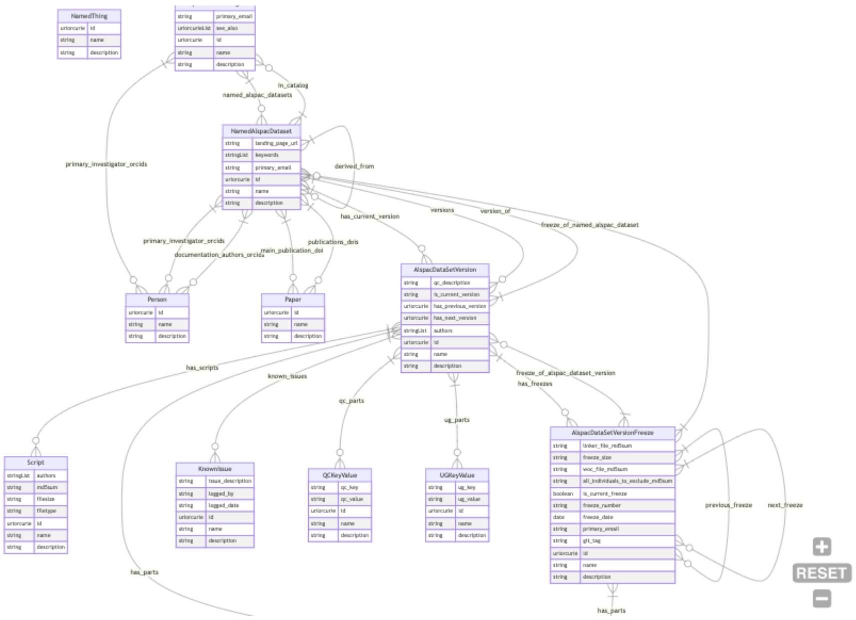


Figure 2: ER diagram

For full documentation on the designed schema please visit the schema documentation page `https://alspac.github.io/alspac-data-catalogue-schema/` which is auto generated from the schema with the github actions set up by the cruft managed cookie cutter.

Note that is is possible to view sub-diagrams of each class, and to view the raw source code as well as the inferred source code for a class. It is also possible to search the documentation using the search function. Figure 4

The schema is managed in the github repo: `https://github.com/alspac/alspac-data-catalogue-schema`

Here you can check out the different versions and see the development history of the schema.

There it is possible to raise issues about the schema. This can be to ask questions or suggest improvements. Pull requests are also welcome.

5 Populating the data for the catalog.

With the schema in place, we set out to create the necessary files that contain the metadata for existing datasets that we want to catalog. Each individual ALSPAC dataset has an associated GitHub repository, which contains three

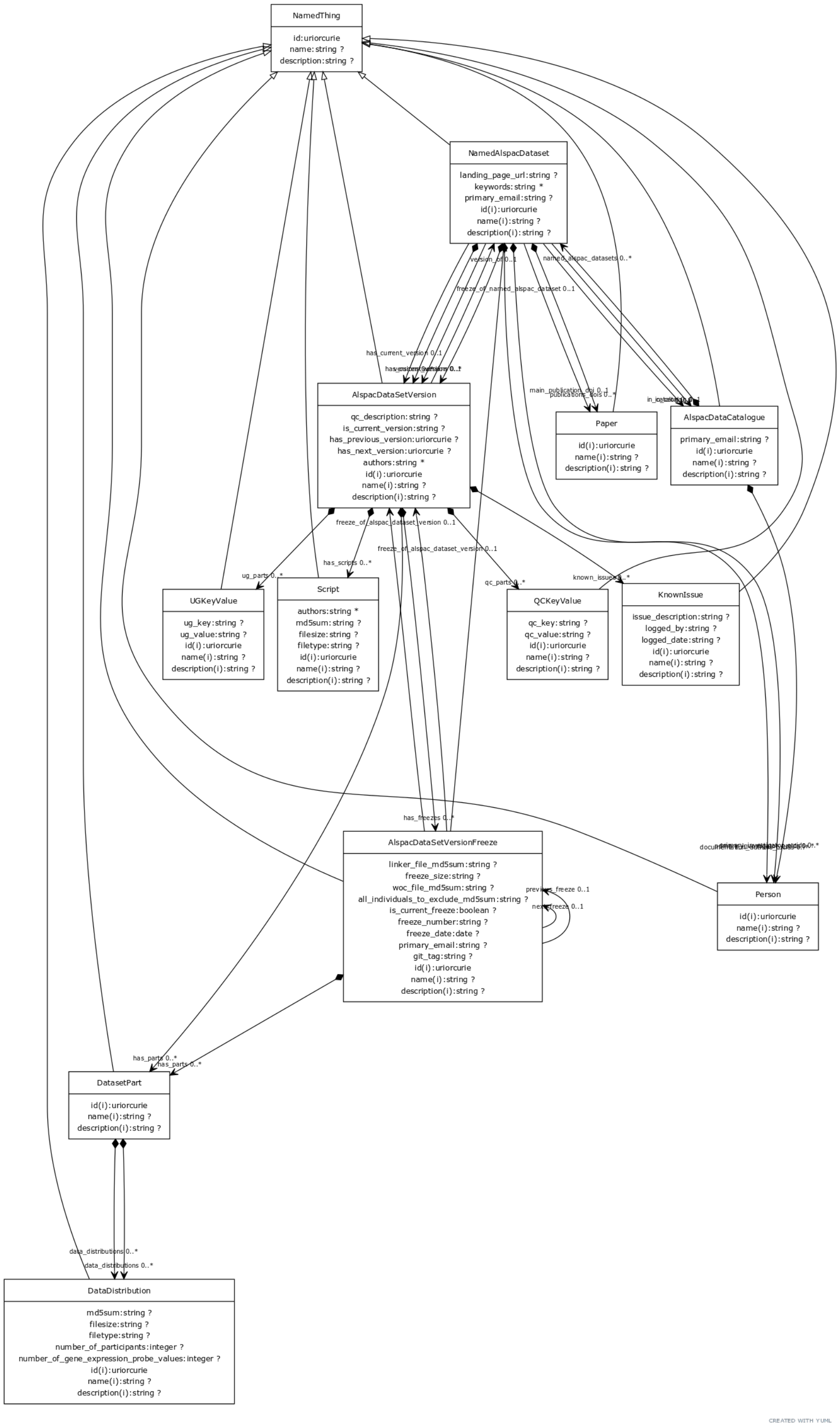
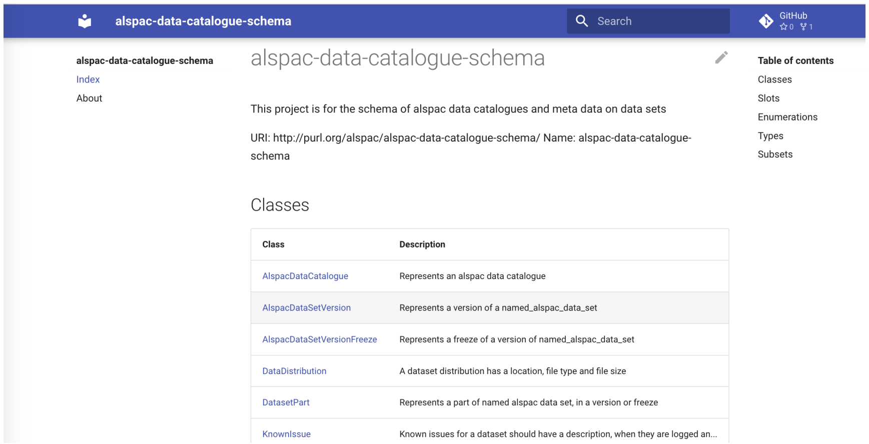


Figure 3: Uml diagram



alspac-data-catalogue-schema

This project is for the schema of alspac data catalogues and meta data on data sets

URI: http://purl.org/alspac/alspac-data-catalogue-schema/ Name: alspac-data-catalogue-schema

Classes

Class	Description
AlspacDataCatalogue	Represents an alspac data catalogue
AlspacDataSetVersion	Represents a version of a named_alspac_data_set
AlspacDataSetVersionFreeze	Represents a freeze of a version of named_alspac_data_set
DataDistribution	A dataset distribution has a location, file type and file size
DatasetPart	Represents a part of named alspac data set, in a version or freeze
KnownIssue	Known issues for a dataset should have a description, when they are logged an...

Figure 4: Screen shot of documentation website

types of YAML files for the data set's metadata: one for the dataset, one for its versions, and one for its "freezes." These YAML files provide easy-to-read versions of the data that can be distributed with the datasets.

We populated the data using a combination of manual data entry and small scripts that, for example, obtained the MD5 sum and size of data files. To ensure that data corresponds to the schema, we used validation tools provided by LinkML when creating new YAML files.

To add a new named dataset to the catalog, we first decide on a name for the dataset and create a private GitHub repository under that name. Then, we set up a space in RDSF for the data, create a standard folder structure, and set up an RDSF bin for external access. We create a "dev" directory to process the data into a release version and create documentation for the named release dataset and its versions. We convert the documentation from YAML to RDF/TTL using the linkml tools and create PNG of TTL if it is not too large, and load the TTL files into the LOD server (discussed later).

To add a new version of a named dataset to the catalog, we create a new directory with a date, use the existing "dev" directory to process the data, and create or modify the freeze scripts to be able to create new freezes based on this new version and add details to the catalog, We run the freeze creation script and populate a new YAML file documenting the freeze's contents. This file is then validated and converted to RDF/TTL with the linkml tools.

Another class we define is for a new custom dataset. This class can be used to record meta data for custom datasets that are built for a specific researcher. We add fields to this class that allow us to know the relationship to the standard named datasets and how the files are built and by who. This will also us to have provenance and the ability to reproduce these dataset at a future date in a simpler manner.

These procedures are fully detailed in the relevant internal SOPs.

6 Presenting the data

The catalog can be viewed in various ways. The first option is to view the metadata files in YAML format in the GitHub repository for each dataset. The schema for the data is also available in YAML format within the GitHub repository for the schema. Additionally, RDF files in TTL format are provided, which correspond to the YAML data combined with the schema produced by the LinkML facility. These individual subgraphs of the data can be easily analyzed by researchers using various computing resources, such as R, Python, or Java. Along with the TTL files, visual representations of the graph in PNG format are also provided for easier comprehension, but only for graphs that are not too large.

For example:

```
# This yaml file is a description of a named alspac data set.
# It should conform to the schema https://github.com/alspac/alspac-data-catalogue-schem

id: alspacdcs:ge_ht12_g1
name: Gene expression - array - G1
description: Gene expression data from Illumina Human HT-12 v3 bead array for G1 indivi
in_catalog: alspacdcs:alspac_data_catalogue_001
landing_page_url: http://www.bristol.ac.uk/alspac/researchers/our-data/biological-resou
primary_investigator_orcids:
  - ORCID:0000-0002-7141-9189 # Nic Timpson
  - ORCID:0000-0003-0663-4621 # Dave Evans
keywords:
  - genomic
  - expression
  - genome-wide
  - illumina
  - transcription
has_current_version: alspacdcs:ge_ht12_g1_2015-11-02
versions:
  - alspacdcs:ge_ht12_g1_2015-11-02
```

```
primary_email: alspac-omics@bristol.ac.uk # Who to contact with questions about this da
documentation_authors_orcids:
  - ORCID:0000-0003-0663-4621 # Dave Evans
  - ORCID:0000-0003-0920-1055 # Gibran Hemani
  - ORCID:0000-0002-4064-3794 # Sam Neaves
main_publication_doi: doi:10.1371/journal.pgen.1004461 # Cis and Trans Effects of Huma
publications_dois:
  - doi:10.1371/journal.pgen.1004461
```

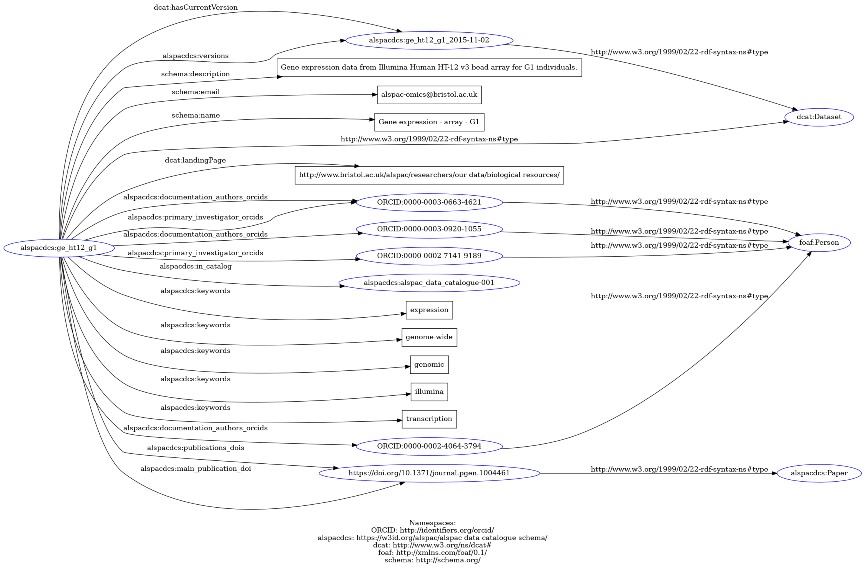

dcat:hasCurrentVersion
alspacdcs:ge_ht12_g1_2015-11-02
alspacdcs:versions
schema:description
Gene expression data from Illumina Human HT-12 v3 bead array for G1 individuals.
schema:email
alspac-omics@bristol.ac.uk
schema:name
Gene expression - array - G1
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
dcat:Dataset
dcat:landingPage
http://www.bristol.ac.uk/alspac/researchers/our-data/biological-resources/
alspacdcs:documentation_authors_orcids
alspacdcs:primary_investigator_orcids
ORCID:0000-0003-0663-4621
ORCID:0000-0003-0920-1055
ORCID:0000-0002-7141-9189
foaf:Person
alspacdcs:ge_ht12_g1
alspacdcs:in_catalog
alspacdcs:alspac_data_catalogue-001
alspacdcs:keywords
expression
genome-wide
genomic
illumina
transcription
ORCID:0000-0002-4064-3794
alspacdcs:publications_dois
alspacdcs:main_publication_doi
https://doi.org/10.1371/journal.pgen.1004461
alspacdcs:Paper
Namespaces:
ORCID: http://identifiers.org/orcid/
alspacdcs: https://w3id.org/alspac/alspac-data-catalogue-schema/
dcat: http://www.w3.org/ns/dcat#
foaf: http://xmlns.com/foaf/0.1/
schema: http://schema.org/

Figure 5: Example of RDF subgraph generated from the yaml file for named alspac data

6.1 Linked open data tripple store

The LOD tripple store that we use to present the omics data catalog is Cliopatria (15). ClioPatria is a Semantic Web toolkit that is based on the Logic Programming (LP) paradigm and is tightly connected to an efficient main-memory RDF store. This in memory tripple store supports multi language querying, including SPARQL and prolog.

ClioPatria is good for an omics data catalog because:

It provides a built in HTML site generator for navigating the loaded RDF. This presents a page for every entity with links to other pages for the other entities. This allows people to navigate the ALSPAC omics catalog data as a graph which is useful for data discovery. We have customised how each page is presented by taking advantage of the config settings made availble for ClioPatria. For example we have added a custom home page and custom menu bar. The menu bar has links to other custom pages such as a omics tips page, which is built from an emacs org markdown file. In this way tips for using ALSPAC omics data can be shared and updated by updating the tips org markdown file which is managed in the github repository.

Further, the presentation of the main data in the generated html is modified using the provided 'hook' predicates that enable us to show extra detail when displaying a link to another resource. For example providing the looked up name as well as the id of a resource. This makes it easier for a person navigating the site to understand each entity without having to click through.

ClioPatria includes a SPARQL 1.1 endpoint (16) which allows us querying the catalog using the standard SPARQL language, this can make it easier to integrate and work with a variety of omics data sources, as SPARQL supports federated queries which is when one query interacts with multiple data end points 8.

ClioPatria also includes Pengines (17), which allow for the remote execution of simple programs and can be accessed from a variety of languages. ClioPatria also has web-based interface based on pengines which is a version of SWISH (18) . This version has tight integration with the rest of the LOD tripple store (For example a query result which is a rdf tripple is presented as a clickable link to the html page for that resource).

These tools allow users to run programs and queries related to the data catalog in a web browser, This makes it easier for users to access and interact with the data catalog.

Additionally, the ability to embed SWISH in tutorial web pages or use it for collaborative development makes it easier for users to learn how to use the data catalog and work with others on analyzing the data. For example standard reports can be written as notebooks that query for the sizes of combinations of datasets or for finding details about a specific dataset or file.

6.1.1 Docker containerisation

Docker is a tool that enables the creation and deployment of applications in self-contained, isolated environments called containers. Containers allow us to package our application with all of the necessary dependencies, libraries, and configuration files, making it easy to deploy and run on any platform.

We use docker to deploy the Cliopatria tripple store.

There are several reasons why Docker is useful for the RDF triple store that describes the ALSPAC omics data catalog:

Portability: Docker containers are portable, which means that they can be easily deployed and run on any platform that supports Docker. This makes it easier to manage the deployment of the RDF triple store, especially if you need to run it on multiple environments (e.g. development, staging, production).

Isolation: Docker containers provide isolation, which means that each container runs in its own isolated environment. This can help to prevent conflicts between different applications or libraries that may be running on the same system.

Scalability: Docker containers are easy to scale up or down, which can be useful if we end up needing to handle a large volume of data or a large number of users.

Maintenance: Docker makes it easier to maintain the RDF triple store by providing a consistent environment for development, testing, and deployment. This can help to reduce the risk of errors and make it easier to roll out updates or new features. For example development can take place on one machine and then deployed to another simply.

Overall, using Docker helps to make the deployment and management of the omics data catalog as a RDF triple store more efficient and reliable.

7 Availability

- The catalog is availble here: `http://purl.org/alspac/alspac-data-catalogue-schema/alspac_data_catalogue_001`
- SPARQL end point here: `http://samneaves.ddns.net/yasgui/index.html`
- SWISH Pengine here: `http://samneaves.ddns.net/swish/`
- Github repoistory here: `https://github.com/alspac/alspac-data-catalogue-schema`
- Data schema documentation here: `https://alspac.github.io/alspac-data-catalogue-schema`

The SWISH Pengine endpoint allows querying the catalog with Prolog and other languages such as Python and Bash.

8 Using the Omics Data Catalog

In previous sections, we established that the Omics Data Catalog serves various purposes, these can be categorized based on the user types. Primarily, there are two types of users.

Researchers: These users aim to perform data discovery tasks and request actual omics data for their research. Users can explore the named ALSPAC datasets in the catalog by visiting the following page: `http://purl.org/alspac/alspac-data-catalogue-schema/alspac_data_catalogue_001`. See figure 6 By clicking on a specific dataset, users can access more details: For example to see the Gene expression - array - G12 dataset page you they would visit: `http://purl.org/alspac/alspac-data-catalogue-schema/ge_ht12_g1`

From there, users can examine a particular version of the named ALSPAC dataset and view the contents of the ALSPAC freeze version. They can then explore the various parts of the dataset and the data distributions for each part.

At any point, users can view relationships between datasets, such as how dataset X is derived from dataset Y. The search functionality can also be used to search the catalog 7.

Based on this understanding, users can decide whether to make a standard omics data request or a custom data request. To do so, they will follow the process outlined on the ALSPAC website. If they need clarification on how metadata is modeled, they can refer to the schema documentation, which includes a search functionality: 4.

Internal ALSPAC Omics Staff: This user group consists of staff, managers, and delegates who use the catalog to gain a better understanding of the stored data. This is beneficial for reporting, data provenance, and security. These users can navigate the catalog as outlined above and also query the catalog for the purpose of reporting and understanding the data, see the next section for some simple examples.

Additionally, individuals may utilise the schema to validate metadata YAML documents they generate, ensuring that the data adheres to the prescribed schema. This can be achieved using the following command, for example: `linkml-validate -s alspac_data_catalogue_schema.yaml data.yaml --target-class NamedAlspacDataset.` This command attempts to validate the data.yaml file against the NamedAlspacDataset class.

If the data is valid, confirmation will be provided; otherwise, helpful error messages will be displayed, such as identifying a missing required field. This process helps maintain consistency and usefulness of the data in the catalogue. Should a user discover their file does not conform to the schema, they can either modify the file or, in certain instances, amend the schema if they have decided to record a new type of information, for example.

9 Summary of the data availble in the catalog

The following queries give a summary of the data available in the catalog, including information on the types of data, the number of samples, and the research areas covered.

- Number of Named ALSPAC datasets

  ? -Query.

- Number of Files in the omics catalog:

  ?- Query.

- Number of triples

  ?- Query.

- Total size of files in catalog

  ?- Query.

9.0.1 How FAIR metadata can complement existing data discoverability tools.

Cohort and Longitudinal Studies Enhancement Resources (CLOSER) (19)is a consortium based at the UCL Institute of Education in the UK. It was established in 2012 to improve the integration, enhancement, and use of longitudinal data from a range of biomedical and social science studies. CLOSER brings together eight UK longitudinal studies, each of which has its own participant group and data collection methods. The aim of CLOSER is to encourage collaboration and the exchange of knowledge and skills between the different studies, in order to identify new learning opportunities and establish tools and standards that can facilitate and improve longitudinal research. The work is aimed at addressing challenges such as divergences in construct definitions, gaps in data coverage, large volumes of data, and the need for data harmonization and linkage.

The existing CLOSER discovery tool primarily aims to enable discovery of data in studies such as ALSPAC that have been collected in questionnaires

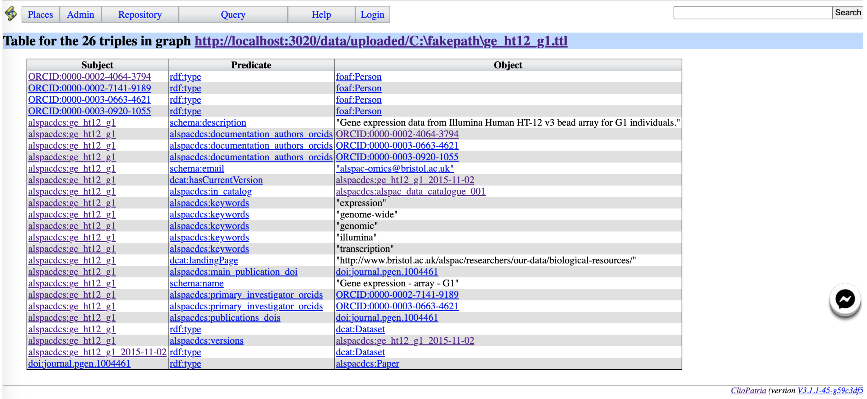
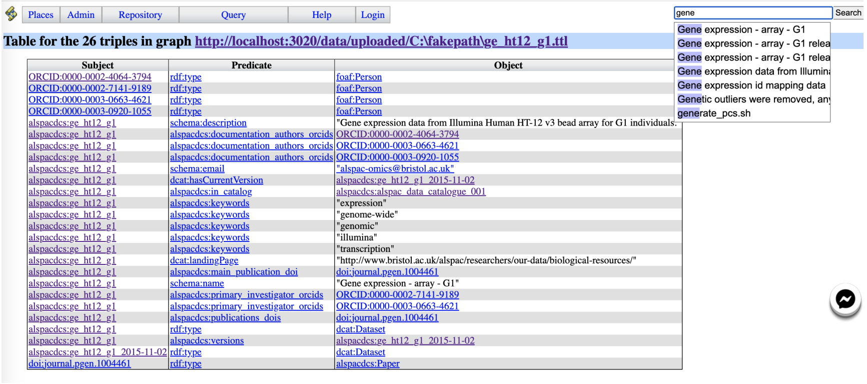


Table for the 26 triples in graph http://localhost:3020/data/uploaded/C:\fakepath\ge_ht12_g1.ttl

Subject	Predicate	Object
ORCID:0000-0002-4064-3794	rdf:type	foaf:Person
ORCID:0000-0002-7141-9189	rdf:type	foaf:Person
ORCID:0000-0003-0663-4621	rdf:type	foaf:Person
ORCID:0000-0003-0920-1055	rdf:type	foaf:Person
alspacdcs:ge_ht12_g1	schema:description	"Gene expression data from Illumina Human HT-12 v3 bead array for G1 individuals."
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0002-4064-3794
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0003-0663-4621
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0003-0920-1055
alspacdcs:ge_ht12_g1	schema:email	"alspac-omics@bristol.ac.uk"
alspacdcs:ge_ht12_g1	dcat:hasCurrentVersion	alspacdcs:ge_ht12_g1_2015-11-02
alspacdcs:ge_ht12_g1	alspacdcs:in_catalog	alspacdcs:alspac_data_catalogue_001
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"expression"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"genome-wide"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"genomic"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"illumina"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"transcription"
alspacdcs:ge_ht12_g1	dcat:landingPage	"http://www.bristol.ac.uk/alspac/researchers/our-data/biological-resources/"
alspacdcs:ge_ht12_g1	alspacdcs:main_publication_doi	doi:journal.pgen.1004461
alspacdcs:ge_ht12_g1	schema:name	"Gene expression - array - G1"
alspacdcs:ge_ht12_g1	alspacdcs:primary_investigator_orcids	ORCID:0000-0002-7141-9189
alspacdcs:ge_ht12_g1	alspacdcs:primary_investigator_orcids	ORCID:0000-0003-0663-4621
alspacdcs:ge_ht12_g1	alspacdcs:publications_dois	doi:journal.pgen.1004461
alspacdcs:ge_ht12_g1	rdf:type	dcat:Dataset
alspacdcs:ge_ht12_g1	alspacdcs:versions	alspacdcs:ge_ht12_g1_2015-11-02
alspacdcs:ge_ht12_g1_2015-11-02	rdf:type	dcat:Dataset
doi:journal.pgen.1004461	rdf:type	alspacdcs:Paper

ClioPatria (version V3.1.1-45-g59c3df5

Figure 6: Screen shot of the catalog view in ClioPatria. This screen shows the triples that describe the Gene expression data from Illumina Human HT-12 v3 bead array for G1 individuals, dataset with code name: ge$_{\mathrm{ht12g1}}$. These links will take a user around the knowledge graph so that they can explore the data.



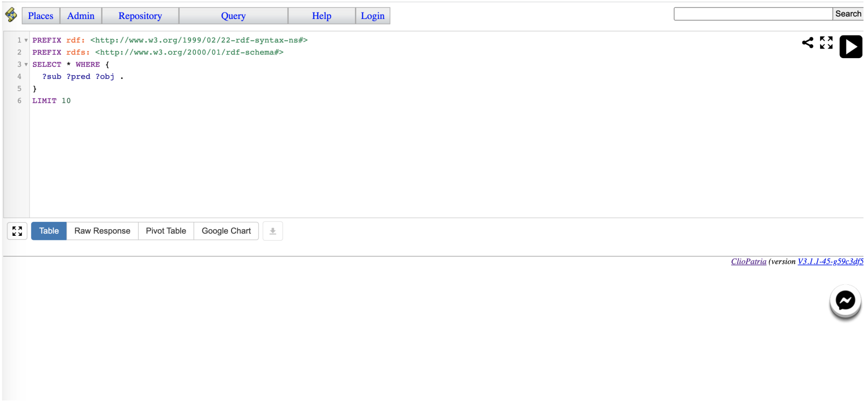
gene

Gene expression - array - G1
Gene expression - array - G1 relea
Gene expression - array - G1 relea
Gene expression data from Illumina
Gene expression id mapping data
Genetic outliers were removed, any
generate_pcs.sh

Table for the 26 triples in graph http://localhost:3020/data/uploaded/C:\fakepath\ge_ht12_g1.ttl

Subject	Predicate	Object
ORCID:0000-0002-4064-3794	rdf:type	foaf:Person
ORCID:0000-0002-7141-9189	rdf:type	foaf:Person
ORCID:0000-0003-0663-4621	rdf:type	foaf:Person
ORCID:0000-0003-0920-1055	rdf:type	foaf:Person
alspacdcs:ge_ht12_g1	schema:description	"Gene expression data from Illumina Human HT-12 v3 bead array for G1 individuals.
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0002-4064-3794
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0003-0663-4621
alspacdcs:ge_ht12_g1	alspacdcs:documentation_authors_orcids	ORCID:0000-0003-0920-1055
alspacdcs:ge_ht12_g1	schema:email	"alspac-omics@bristol.ac.uk"
alspacdcs:ge_ht12_g1	dcat:hasCurrentVersion	alspacdcs:ge_ht12_g1_2015-11-02
alspacdcs:ge_ht12_g1	alspacdcs:in_catalog	alspacdcs:alspac_data_catalogue_001
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"expression"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"genome-wide"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"genomic"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"illumina"
alspacdcs:ge_ht12_g1	alspacdcs:keywords	"transcription"
alspacdcs:ge_ht12_g1	dcat:landingPage	"http://www.bristol.ac.uk/alspac/researchers/our-data/biological-resources/"
alspacdcs:ge_ht12_g1	alspacdcs:main_publication_doi	doi:journal.pgen.1004461
alspacdcs:ge_ht12_g1	schema:name	"Gene expression - array - G1"
alspacdcs:ge_ht12_g1	alspacdcs:primary_investigator_orcids	ORCID:0000-0002-7141-9189
alspacdcs:ge_ht12_g1	alspacdcs:primary_investigator_orcids	ORCID:0000-0003-0663-4621
alspacdcs:ge_ht12_g1	alspacdcs:publications_dois	doi:journal.pgen.1004461
alspacdcs:ge_ht12_g1	rdf:type	dcat:Dataset
alspacdcs:ge_ht12_g1	alspacdcs:versions	alspacdcs:ge_ht12_g1_2015-11-02
alspacdcs:ge_ht12_g1_2015-11-02	rdf:type	dcat:Dataset
doi:journal.pgen.1004461	rdf:type	alspacdcs:Paper

Figure 7: Screen shot of the search functionality of the tripple store.



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT * WHERE {
  ?sub ?pred ?obj .
}
LIMIT 10
```

ClioPatria (version V3.1.1-45-g59c3df5

Figure 8: Screen shot of the SPARQL interface on the tripple store.

and the discovery tool only gives a high level description of the omics data that ALSPAC has stored.

The ALSPAC omics data catalog is a FAIR (5) linked open data server that provides detailed information about the omics datasets and files that are available for use in research. This catalog is designed to complement the CLOSER (Cohort and Longitudinal Studies Enhancement Resources) consortium by providing a centralized, standardized resource for accessing and using the omics data from the ALSPAC study.

The FAIR principles, which are intended to make data more findable, accessible, interoperable, and reusable, are relevant to CLOSER because they help to ensure that data from the different studies in the consortium are consistent and easy to use. By providing a FAIR linked open data server, the ALSPAC omics data catalog helps to support the integration, enhancement, and use of data from the ALSPAC study within the CLOSER consortium, and it also makes it easier for researchers to find and use the data for their own purposes.

10 Possible future improvements

There is a considerable number of potential extensions to the data catalogue and schema that could enhance its functionality. These can be categorised into two main types of extensions.

The first type of extension involves maintaining the existing schema while populating it with more information. This may include providing additional descriptions for multiple versions of the currently named ALSPAC datasets, as initially, we have only offered comprehensive YAML descriptions for the latest versions. Another possibility is the formal incorporation of supplementary datasets from our archive data, such as HLA imputed datasets.

The second type of extension entails modelling individual data files. For instance, we could utilise LinkML to describe the gene expression data files. This would involve extending the current schema or creating new schemas that characterise the classes for the actual data in each dataset. For every set of tabular data within each dataset, there will be a class that connects to semantic information about the corresponding values.

In conjunction with the LinkML meta schema, of which these classes will be instances, this approach will enable us to generate RDF triples that can be integrated into our Linked Open Data (LOD) server. Consequently, this will facilitate the search for individual variables appearing in each dataset and distribution, such as a gene expression probe that could be semantically linked to other data points using gene and sequence ontologies.

11 Conclusion:

In conclusion, the Avon Longitudinal Study of Parents and Children (ALSPAC) has amassed an immense quantity of omics data over time, encompassing genetic array, methylation, and gene expression data, as well as in-depth phenotype data, in order to explore the genetic factors influencing disease and well-being. To enhance accessibility and integration, ALSPAC has devised a data catalogue schema and an omics data catalogue using LinkML, which is presented as Linked Open Data (LOD) in a web-accessible triple store.

The employment of LOD and LinkML offers numerous advantages, such as streamlining the integration and linking of data from various locations and sources, enabling the construction of flexible and extensible data models, and promoting the use of standardised and interoperable technologies. Additionally, it provides PURLs for each data point, thereby eliminating ambiguity concerning data resources. Consequently, researchers can now easily access and utilise ALSPAC's omics metadata, offering considerable benefits when working with this intricate and extensive dataset, particularly in terms of data discovery and the integration of data from multiple sources to maximise statistical power and address scientific inquiries.

Furthermore, maintaining and enhancing the catalogue is made simple through the use of LinkML tooling and well-defined YAML files that permit comments. These files can be validated by the LinkML tooling, ensuring consistency and seamless data ingestion. This streamlined approach not only simplifies the management process but also contributes to the overall efficiency and effectiveness of the catalogue.

Other advantages include the ability to search for information about files by their md5sum, which allows users to view the contextual and semantic details of a specific file. Ultimately, this paper serves as documentation on the development of a catalogue using modern best-practice tools and features, thereby promoting further advancements in the field.

12 Acknowledgments

This work was carried out using the computational facilities of the Advanced Computing Research Centre, University of Bristol - http://www.bris.ac.uk/acrc/.

13 References

1. Boyd A, Golding J, Macleod J, et al. Cohort Profile: The 'Children of the 90s'—the index offspring of the Avon Longitudinal Study of Parents and Children. *International journal of epidemiology* [electronic article]. 2012;42(1):111–127. (https://doi.org/10.1093/ije/dys064)

2. Fraser A, Macdonald-Wallis C, Tilling K, et al. Cohort Profile: The Avon Longitudinal Study of Parents and Children: ALSPAC mothers cohort. *International journal of epidemiology* [electronic article]. 2012;42(1):97–110. (https://doi.org/10.1093/ije/dys066)

3. Moxon SA, Solbrig H, Unni DR, et al. The linked data modeling language (linkml): A general-purpose data modeling framework grounded in machine-readable semantics. In: *Icbo*. 2021:148–151.

4. Bizer C, Heath T, Berners-Lee T. Linked data: The story so far. In: *Semantic services, interoperability and web applications: emerging concepts*. IGI global; 2011:205–227.

5. Wilkinson MD, Dumontier M, Aalbersberg IJ, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*. 2016;3(1):1–9.

6. European Parliament, Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). 2016;

7. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 27001:2013, Information technology – security techniques – Information security management systems – Requirements. Geneva, Switzerland: International Organization for Standardization; 2013.

8. Relton CL, Gaunt T, McArdle W, et al. Data resource profile: accessible resource for integrated epigenomic studies (aries). *International journal of epidemiology.* 2015;44(4):1181–1190.

9. Manola F, Miller E, McBride B, et al. Rdf primer. *W3c recommendation.* 2004;10(1-107):6.

10. W3C Semantic Web Interest Group. DCAT - data catalog vocabulary. 2022;

11. Sebastian Trüg. Nepomuk File System Ontology. 2007;

12. Miles A, Bechhofer S. Skos simple knowledge organization system reference. 2009;

13. Kothari S. Cruft. 2022;

14. Poetry. 2023;

15. Jan Wielemaker MH Wouter Beek, Ossenbruggen JV. Cliopatria: A swi-prolog infrastructure for the semantic web. *Semantic web journal.* 2015;

16. Pérez J, Arenas M, Gutierrez C. Semantics and complexity of sparql. *Acm transactions on database systems (tods).* 2009;34(3):1–45.

17. Lager T, Wielemaker J. Pengines: Web logic programming made easy. *Theory and practice of logic programming.* 2014;14(4-5):539–552.

18. Wielemaker J, Lager T, Riguzzi F. Swish: Swi-prolog for sharing. *Arxiv preprint arxiv:1511.00915.* 2015;

19. O'Neill D, Benzeval M, Boyd A, et al. Data resource profile: cohort and longitudinal studies enhancement resources (closer). *International journal of epidemiology.* 2019;48(3):675–676i.