Datenbank-Spektrum (2023) 23:107–115
https://doi.org/10.1007/s13222-023-00444-3

SCHWERPUNKTBEITRAG

Metadata Management and Asset Exchange in the Agricultural Data Ecosystem of the Project Agri-Gaia

Tobias Wamhof[1] · Ansgar Bernardi[2] · Daniel Martini[3] · Martin Leinberger[4] · Arka Sinha[2] · Heiko Tapken[1] · Andreas Schliebitz[1] · Henri Graf[1]

Received: 10 March 2023 / Accepted: 17 May 2023 / Published online: 29 June 2023

Abstract
The particularities of the agricultural ecosystem of cooperating partners suggest a highly distributed, open data architecture to support the exchange and re-use of data and data-driven applications. We describe the data strategy and the federated basic architecture of the Agri-Gaia ecosystem. We present an ontology-based approach to metadata management which extends a bucket store for arbitrary data storage by an RDF-based metadata graph store and employs widely-used domain ontologies as a conceptual basis. Data and service providers are free to extend the describing metadata at any time, according to their needs. The resulting set of interconnected platforms supports the publication, retrieval and secure sharing of exposable data under full control of their owners.

Keywords Metadata Management · SPARQL · Agri-Gaia · Gaia-X

Tobias Wamhof
t.wamhof@hs-osnabrueck.de

✉ Ansgar Bernardi
ansgar.bernardi@dfki.de

Daniel Martini
d.martini@ktbl.de

Martin Leinberger
martin.leinberger@de.bosch.com

Arka Sinha
arka.sinha@dfki.de

Heiko Tapken
h.tapken@hs-osnabrueck.de

Andreas Schliebitz
a.schliebitz@hs-osnabrueck.de

Henri Graf
h.graf@hs-osnabrueck.de

[1] Hochschule Osnabrück, Albrechtstraße 30, 49076 Osnabrück, Germany

[2] German Research Center for Artificial Intelligence—DFKI GmbH, Trippstadter Str. 122, 67663 Kaiserslautern, Germany

[3] Kuratorium für Technik und Bauwesen in der Landwirtschaft e. V. (KTBL), Bartningstraße 49, 64289 Darmstadt, Germany

[4] Corporate Research, Robert Bosch GmbH, Robert-Bosch-Campus 1, 71272 Renningen, Germany

1 Introduction

Agriculture can profit from modern AI-based applications (in particular: recognition and assessment of all kinds of structures in nature – which in turn relies on models, i.e. suitably-trained Artificial Neural Networks). Effective building of such applications needs resources and know-how for building and training the intended AI functionality – available at software developers and engineers – and the data sets on which the AI can be trained, which are to be provided by their owners. Most of the time this is a farmer, who collects the data by use of documentation systems or logging machinery, but who cannot profit from it. In practice, some elements are often lacking during a development process. The developer lacks enough data to train a model, while the farmers hesitate to share their data due to missing distribution support and fear of misuse of loss of control. To ameliorate this, support is needed to foster fair exchange of data sets and models between all participants in an cooperative ecosystem.

The pan-european initiative Gaia-X[1] envisions *"a federated and secure data infrastructure"* and aims at *"an ecosystem, whereby data is shared and made available in*

[1] See https://gaia-x.eu. Accessed 2023-06-21

a trustworthy environment". To this end, Gaia-X develops guidelines, policies, and tools to enable any participants (usually enterprises) to join a federated system where sharing agreements can be reached and assets can be exchanged. Core aspects are the unique and secure identification of every participant, the description of assets (data or services) and their listing in catalogues, the definition and execution of policies governing business and legal conditions for any particular exchange and usage of assets, and the technical transfer of data when the participants in question have reached an agreement.

While Gaia-X development is still ongoing, Gaia-X principles are used and adapted in domain-specific solutions. Some examples are CATENA-X[2] (for automotive industry), the intiative Manufacturing-X[3] (for smart industrial production) – and Agri-Gaia[4] for AI support in the agricultural domain.

Agri-Gaia makes available rich data sets, which can be used for model training. It emphasizes technical solutions for data sovereignty to ensure only users can obtain the data, if they meet certain requirements defined by the provider, and envisions new business for trained models, that shall be transferred to users who deploy them on their machines. To ensure broad usage and data offers, such ecosystem is accessible for all interested participants.

Meaningful asset sharing among participants in an open ecosystem requires a technical infrastructure and common understanding on the metadata describing the assets. We suggest to support this ecosystem by implementing a set of interconnected platforms based on GAIA-X principles. A platform shall serve as the access node of a participant which facilitates a local storage and description of their own data resources and usage conditions (thus ensuring full data sovereignty of the owner). The participants' platforms will serve as exchange tools for obtaining data or services from other platforms, will support the training steps which help to build an AI model (using obtained data) and deploy the trained model to the application-ready edge device. Own data resources are made available to other participants for mutual benefit by registering platforms at one or several market places and exhibiting their data descriptions (i.e. selected meta data). To ensure common understanding and integration within a market place, all meta data shall be based on suitable domain ontologies which are the formal basis for shared understanding. Within a market place, participants may search for data and/or services according to their needs and trigger negotiation of applicable contracts.

In the following we will present details on data storage, metadata storage, and the domain adequate ontological basis used to describe managed datasets and models. In addition to that, several workflows are described, e.g. the data storage, the metadata querying system and the data exchange based on different Gaia-X technologies.

2 Requirements for handling assets and metadata

The availability and correct and fair handling of data gains ever-increasing importance in digitized agriculture; however, the various parties who cooperate in agricultural production pursue different and sometimes conflicting interests. The need for both easy and open sharing of data as well as for control and data sovereignty is broadly discussed [12]. The full vision of universally available open data conflicts with economic interests and need for protection of privacy and business secrets. Furthermore, the growing understanding of the potential economic value of quality data [7] emphasizes the need for owner's control and fair business models. Current data platform solution approaches in agriculture [4] predominantly concentrate on the collection, transfer and usage of operational data for documentation, control and optimization of agricultural and food production work processes. A Generic support for training of AI-based, data driven solutions in agriculture is not yet widely attempted. Consequently, support for data-driven AI solutions in agriculture require

- a distributed ecosystem of individual storage entities which keep all data assets under the control of their respective owners, while facilitating a federated sharing, subject to individual usage conditions. Faced with the variety of possibly interesting data assets in agricultural scenarios, such system must be independent of the actual data formats.
- a universally usable metadata system to describe content and usage of available data assets. Common understanding across various participants (even world-wide) needs formalized, computer-processable descriptions based on established, widely usable domain ontologies.
- dynamic extension of descriptive information and data schema. To allow for the different viewpoints of participants in agricultural scenarios and to cope with varying and unforeseen needs of AI development, such metadata system needs to allow for dynamic addition of descriptive metadata by participants over time.

We thus envision a distributed, multi-participant system which unites the different views and needs of data producers/owners (each with their own viewpoints on form and purpose of collected data and their interpretation), data consumers (each with their own goals and corresponding data needs, search queries, and data processing ideas) and

[2] See https://catena-x.net. Accessed 2023-06-21

[3] see https://www.plattform-i40.de. Accessed 2023-06-21

[4] See https://www.agri-gaia.de. Accessed 2023-06-21

information systems (which require universally available, standardized data structures and meta data in order to ensure basic data management, access or usage control, and search and retrieval functionalities). However, effective support for generation and maintenance of the ontological basis, metadata for the distributed data collections, and user interaction for data ingestion, search and retrieval requires sophisticated interpretation of the ontological basis:

- A formal definition of the classes and concepts describing the various information elements and data types is the basis for the elementary system functionalities. The properties contained in such definitions are indispensible for the functioning of the system and thus considered mandatory.
- Further descriptive properties in any dataset's metadata reflect the intended usage and interpretation considered by the data generators. While technically optional, the richness of such annotation is crucial for a wide usability.
- As later data consumers' interests might not be known at the time of data generation and annotation, the foundation of any metadata in a solid domain ontology (and thus the use of both a common vocabulary and represented domain knowledge) together with suitable mechanisms for browsing and semantic search are necessary. Taking into account the dynamic modifications over time, any interactive system has to provide means to dynamically adapt its user interfaces and data entry forms to the ever-changing ontological basis.

Consequently, the envisioned metadata management system shall rely on a formal ontological basis which is used both in a prescriptive manner (as far as mandatory aspects are concerned) and in the open, descriptive interpretation, thus unifying aspects of traditional, closed-world databases with open-world knowledge bases and their dynamic reaction to reality.

3 The Agri-Gaia federated basic architecture

Agri-Gaia relies on a federated architecture comprising several platforms, a marketplace and a dataspace authority (see Fig. 1). The platforms offer sovereign services related to data storage and processing, enabling a farmer to provide their data through one of the platforms while developers can leverage the platform to process the data and train AI models. Data storage on a platform is typically provided as object storage, making a single platform essentially a data lake. Metadata management is done through an ontology-based data catalog that describes available assets within the platform. For storing the data catalog, we leverage RDF-based graph databases. Each platform then uses a Gaia-X compatible connector service that is responsible for communication and data exchange with other systems within Agri-Gaia. The connector interfaces with the data catalog and data storage, thus making assets (or subsets thereof) available for other systems. In particular, the connector provides endpoints for reading metadata about data offers stored in the data catalog, accepting policies attached to data offers as well as initiating data transfer between two connector instances. Platforms, therefore, form a federated data lake with the potential for data exchange. However, data exchange is limited to systems within the Agri-Gaia ecosystem through the dataspace authority. The dataspace authority holds basic information about participants and systems that are considered to be part of Agri-Gaia. Before data can be exchanged, connectors use the dataspace authority to verify their identity.

The marketplace comprises the unified data catalog of all data offerings within Agri-Gaia. It, therefore, collects metadata on all data offerings provided by the platforms known to the dataspace authority. A crawler is used to collect the data offers. This data is then extended with additional information (e.g., the originating platform for an asset), and the data catalog is updated. Again, the marketplace uses an ontology-based data catalog. Following standard REST prac-

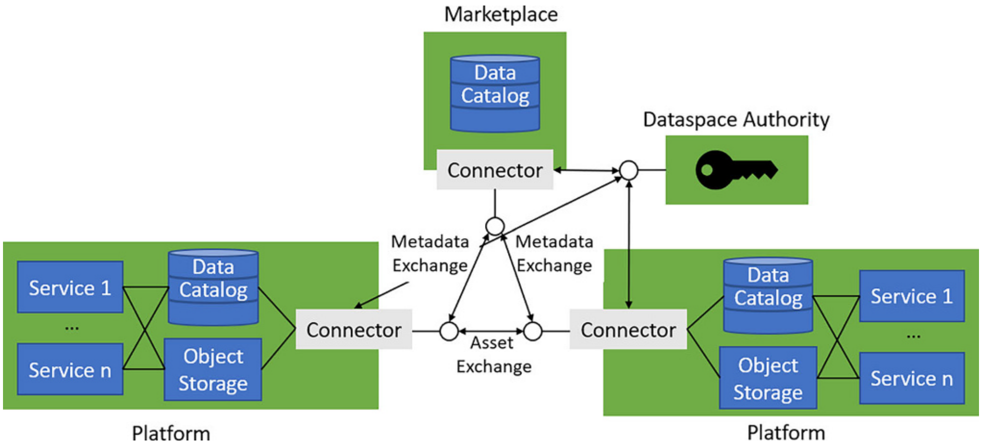

Fig. 1 Architectural overview of Agri-Gaia

tices, our endpoints return JSON data. This data is converted into JSON-LD [14] by the addition of a context, again allowing the data to be stored within a graph database or triple store based on the Resource Description Framework (RDF, [8]). This simplifies the unification of data offerings from different platforms while giving individual platforms greater flexibility in how they describe their data. Nevertheless, the basic vocabulary for data offerings is defined in the Agri-Gaia ontology which is described in Sect. 6. Similar to the platforms, the marketplace offers REST endpoints for accessing and filtering the metadata (e.g., by plant type) of available data offerings. This endpoint is for example consumed by the marketplace UI but is generally open to everyone.

4 Related Work

Since the inception of internet, Tim Berners Lee, who is often regarded as the founder of the world wide web, had a vision of connecting the data on internet. In his own words, "The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation."[5]. Since then a lot of works have been done on defining ontologies for various use cases. However, the domain of agriculture has been one of the under utilized section. Since modern technologies are incorporated in agriculture, more and more useful data are being generated which can be used for better resource management and more precise methods for farming.

An ontology specifically designed for agriculture domain can contribute to that vision and Bansal et.al. [3] created an ontology called CROPont for that purpose. For our work, in addition to agriculture related elements, we also integrated elements that are important for a fully functioning data marketplace. This includes description for the participants, identitiy for the institutions involved, description for the physical and virtual devices used for the processes, resources available for services regarding the resource etc.

Once an ontology is in place, it is also necessary to allow users to enter data based on the framework. Aydin et.al. [2] developed a tool called OWL2MVC which generates a data acquisition form which makes it easy to gather metadata from common users. While OWL2MVC focuses on data acquisition, our platform goes beyond to integrate data storage, graph storage and contract negotiations between participants

Zheng et.al. [22] developed a tool to construct agriculture based ontology by the common user. However, their work is still in research phase. In contrary, in our work we developed the ontology and then we reviewed with potential users and enhanced the ontology to make sure it caters to wide variety of use cases. This gives us more control to ensure we adhere to the standards while also expanding and modifying the ontology to make it more robust.

5 Technology decisions

This section focuses on the technologies required to enable the architecture, that was described in Sect. 3. As mentioned, it is necessary to provide technologies for storing various data objects in different formats, managing the metadata of all of those data objects and to enable the Gaia-X conform exchange of the assets between multiple participants of the ecosystem.

Regarding the storage of metadata Apache Jena Fuseki[5] is used to persist the metadata in a RDF graph format. It also supports the storage of the Agri-Gaia backend ontology and Shapes graph based on the Shapes Constraint Language (SHACL, [15])which is used to validate the incoming metadata inside the Apache Jena Fuseki storage itself. The storage can be queried using SPARQL (an RDF Query language, [11]) to retrieve information on persisted metadata as well as the previous mentioned Agri-Gaia ontology.

A MinIO S3 storage[6] is used to persist datasets and models itself. They are managed in form of buckets and located with different prefixes to simulate a file system in the MinIO frontend. By using an S3 storage it is possible to save different types of datasets. This enables the storage of image datasets alongside with e.g. tabular datasets without the need of including another new technology to the ecosystem. To communicate with the MinIO service different SDKs can be used.

To enable the Gaia-X conform exchange of data and metadata, an Eclipse Dataspace Connector[7] (EDC) is used. Alongside with the asset exchange itself, it manages the metadata of assets published in the catalogue and exchanges this metadata with the connector of the marketplace. We added an extension which enables the asset transfer from and into Minio S3 storages.

Other technologies like the IDSA Connector[8] or the Connector developed by the OCEAN protocol[9] team were also evaluated. As the OCEAN protocol connector would have been based on Distributed Ledger technology, it was discarded from the decision, as it didn't match the technologies, which already were used internally in the platforms

[5] See https://jena.apache.org/documentation/fuseki2/. Accessed 2023-06-21

[6] See https://min.io/. Accessed 2023-06-21

[7] See https://github.com/eclipse-edc/Connector. Accessed 2023-06-21

[8] See https://www.dataspaces.fraunhofer.de/de/software/connector.html. Accessed 2023-06-21

[9] See https://oceanprotocol.com/. Accessed 2023-06-21

and the marketplace. The EDC was chosen, as the Github repository showed recent progress by multiple participants on the project and other big projects like Catena-X support it as well.

6 Ontology-based metadata graph

6.1 Referenced Ontologies

The RDF approach and data model allows – in contrast to more restrictive schema technologies like XML Schema [9, 20] or JSON Schema – a flexible combination of data descriptions in the form of ontologies and vocabularies. Reusing existing, common standards and models and recombining and extending them towards a model suited for the respective applications scenarios and use cases is thus best practice among the RDF community of users. Within Agri-Gaia we thus also rely on a set of fundamental, standardized and stable ontologies, vocabularies and thesauri. This includes:

- A set of abstract base vocabularies including the RDF and RDF Schema [6] vocabularies, OWL [18] and SKOS [17] used for declaring classes, properties and concepts including their hierarchies, documentation and relations
- standard metadata and provenance vocabularies including Dublin Core, DCAT [1] and PROV [16] providing terms for describing sources, download locations and provenance of for example data sets
- vocabularies for describing people, organizations and contact information like vCard [13] and FOAF
- some datatype specific ontologies and vocabularies, for example GeoSPARQL and W3C location [19] for geospatial data, CSVW [21] for tabular data, image metadata vocabularies like EXIF [10] and XMP
- controlled vocabularies/thesauri in the form of SKOS concept schemes like AGROVOC[10] or vocabulary datasets like Geonames[11] as value spaces for some of the properties

Services designed are meant to be compliant to the Gaia-X infrastructure. Interfaces in Gaia-X partly rely on self descriptions of assets that are provided using a set of Gaia-X-specific ontologies and vocabularies. Apart from the proven and established ontologies mentioned above, our own work also relies on reuse of a number of classes and properties from the Gaia-X self description ontologies.

Terms from the above mentioned ontologies are used as building blocks for the description of the assets depending on their types. A certain subset of attributes is declared as mandatory using the Shapes Constraint Language (SHACL, see Sect. 6.2.2 for a more detailed description).

Enabling search for datasets that refer to a certain common topic is one of the application scenarios to cover. For machine learning use cases, also finding datasets that have labels on certain object classes is a common requirement. Simply assigning freetext string keywords to topic or label attributes leads to several challenges:

- spelling and/or language variants would have to be taken into account
- multilingual search is limited to the languages of keywords assigned and search will not produce any output for datasets that have been annotated using another language
- assets using synonyms of terms will not be found

The rationale behind using above mentioned controlled vocabularies and thesauri – namely AGROVOC and Geonames – is to deal with these challenges. The user is encouraged via the user interface to assign terms that are drawn from them. They model for example broader and narrower term and containment relationships explicitly so that search facilities can make use of these. For example showing all data sets for wheat when cereals are searched is possible. You can also include datasets that have been annotated to have been captured at a certain location in the result set of a search for a region, if the given location is located within that region. Also the mentioned thesauri contain lexicalizations in multiple languages taking into account synonymy by providing preferred and alternative labels for concepts.

6.2 Dynamic extensions: Growing the metadata space

6.2.1 Agri-Gaia Ontology

In a collaborative environment, as envisioned in the Agri-Gaia project, it is very important that every participant has a common understanding of the types of resources they are sharing and consuming. Therefore, while describing metadata for a resource, all the terminologies used to define a type of resource and their properties must be unambiguous and universally accepted. For this reason we decided to create an Agri-Gaia ontology which extends the Gaia-X ontology for datasets within the agriculture domain. At the top level we defined a class `AgriDataResource` which encompasses all datasets related to agriculture. This class extends from the W3C standard `Dataset` class from DCAT and `DataResource` class from the Gaia-X ontology gax-trust-framework. This ensures that any resource described in our ontology will conform to Gaia-X standards and can also utilize the rich variety of classes and properties that are connected to DCAT. Additionally, following the open

[10] See https://agrovoc.fao.org. Accessed 2023-06-21

[11] See https://www.geonames.org/. Accessed 2023-06-21

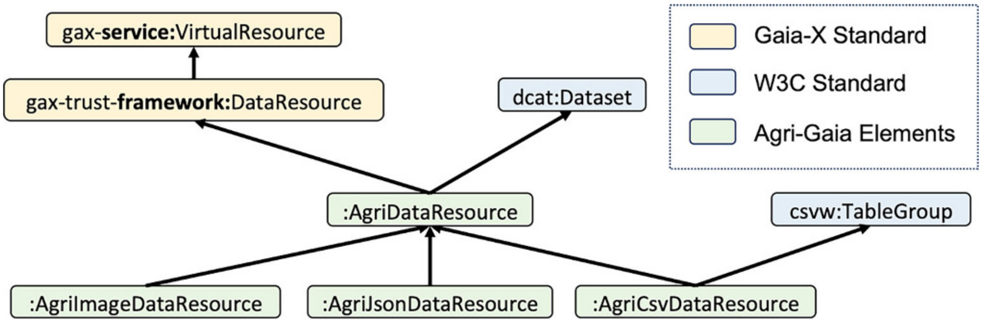

Fig. 2 Class Hierarchy in Agri-Gaia Ontology

world RDF model, it can include additional attributes from other standards for a more versatile metadata description.

We have extended the generic `AgriDataResource` class to three format specific classes, namely for image datasets, JSON datasets and CSV datasets meant for the agriculture domain (Fig. 2). When users want to provide metadata for these specific types of dataset, they will be prompted with a property list that is specific to the chosen type of dataset. For image datasets the number of images, resolution, image channels etc. might be more important but these properties do not apply to a dataset of CSV files. Furthermore, since the ontology is connected to Gaia-X and other W3C standards, many of the generic properties can be reused to enhance the quality of metadata while also conforming to W3C and Gaia-X standards. As Gaia-X expands and we continue to work with our partners to gather more information on various types of resources and how to describe them, we can build on the existing ontology to accommodate more elements to expand our coverage for metadata description.

6.2.2 Applying Constraints

The information elements and their properties (aka concepts) in the Agri-Gaia Ontology are *descriptive object definitions* used as basis for the implemented functionality and for minimal validation to exclude fatally wrong data descriptions. Besides they can be interpreted as a *prescriptive guideline* to be used in interface/form generation or as "how-to" for users when entering data.

To operationalize the verification and user guidance, we employ SHACL shapes to enforce constraints on the incoming metadata. Through a SHACL shape, we can target a particular class by defining it as a target for a NodeShape (Classes are considered as nodes in RDF graphs) and then specify constraints on it's particular properties by defining them under PropertyShape. As automated support, we use the CONSTRUCT query shown in Listing Fig. 3 to create the initial set of SHACL shapes (which are RDF graphs as well) from our ontology graph that forms the basis for our attributes recommendations to the users who want to describe their resources using the Agri-Gaia ontology.

This query creates a node shape for every class and each of those node shapes contains one property shape for each property that is defined for that class in the ontology. For example, the query will create a NodeShape named `AgriImageDataResourceShape` for the class `AgriImageDataResource`. Within that NodeShape, for each property of the class (e.g. imageCount) there will be a PropertyShape (e.g. imageCountShape). A subset of properties for AgriImageDataResource is shown in Listing Fig. 4

The resulting shapes give us a framework for further developments in two key aspects:

(1) We can adjust the property list shown to the external users by simply adding appropriate property shapes or removing non-relevant property shapes from `sh:property` without editing the ontology itself. This gives us full control of the design decisions about which attributes to show the external users for their inputs to describe their particular type of resource. Listing Fig. 5 shows an example of the shapes that were created for the class `AgriImageDataResource` using the query:

(2) We can edit the individual property shapes to introduce various constraints on attributes e.g. cardinality, accepted value type(s), mandatory or optional etc. Listing Fig. 6 is an example of how we can edit the shape for the `imageCount` property and add constraints like the

```
CONSTRUCT {?shpName rdf:type sh:NodeShape.
        ?shpName sh:targetClass ?sub .
        ?shpName sh:property ?propShpName .
        ?propShpName rdf:type sh:PropertyShape;
        sh:path ?prop.
}
WHERE
{
        ?sub rdf:type rdfs:Class.
        ?sub rdfs:subClassOf* ?superSub.
        ?prop rdfs:domain ?superSub.
        FILTER(!isBlank(?sub)).
        Bind(URI( concat(Replace(STR(?sub), "(.*/*#)",
        "https://www.Agri−Gaia.de/shacl/"),'Shape'))
        as ?shpName ).
        Bind(URI( concat(Replace(STR(?prop), "(.*/*#)",
        "https://www.Agri−Gaia.de/shacl/"),'Shape'))
        as ?propShpName ).
}
```

Fig. 3 Query to create SHACL shapes

```
:AgriImageDataResource  a rdfs:Class, owl:Class, skos:Concept .

:imageCount rdf:type owl:DatatypeProperty ;
rdfs:domain :AgriImageDataResource.

:imageColorScheme rdf:type owl:DatatypeProperty ;
rdfs:domain :AgriImageDataResource.

:imageFormat rdf:type owl:DatatypeProperty ;
rdfs:domain :AgriImageDataResource.

:avgImageSize rdf:type owl:DatatypeProperty ;
rdfs:domain :AgriImageDataResource.
```

Fig. 4 Properties for AgriImageDataResource Class

```
@prefix agsh: <https://www.Agri−Gaia.de/shacl/>
agsh:AgriImageDataResourceShape rdf:type sh:NodeShape ;
sh:property agsh:imageCountShape, agsh:imageColorSchemeShape,
        agsh:imageFormatShape, agsh:avgImageSizeShape ;
sh:targetClass agri−gax:AgriImageDataResource .
```

Fig. 5 Shapes for AgriImageDataResourceShape

```
<https://www.Agri−Gaia.de/shacl/imageCountShape>
rdf:type sh:PropertyShape ;
sh:path agri−gax:imageCount;
sh:maxCount 1;
sh:datatype xsd:integer .
```

Fig. 6 Shape for imageCount property

maximum value or the type (for example integer) for the property.

The ontological knowledge and its use for verification and guidance unifies aspects of both the closed, source-of-truth paradigm of classic database, which use stable database schemata, integrity constraints, or consistency-preserving demon mechanisms to avoid the storage of inconsistent data, with aspects of the open world paradigm of knowledge bases, which must accept whatever comes from observing the real world, and then try to make sense from the data by classification, realization of concepts, or modifying assumptions via truth maintenance algorithms.

7 Key processes

Within this section the key processes are described, which can be executed based on the defined ontologies from Sect. 6.

7.1 Storage initialization

To enable the storage, usage and exchange of metadata and data each platform instance provides their own Apache Jena Fuseki, MinIO and EDC instance. All of those services are started by packaging them into containers and run them along side with all other platform services. The Apache Jena Fuseki is used to store the metadata itself. On startup of the container pre-configured ontologies are loaded automatically, like the AGROVOC, a part of the Geonames ontology, the Agri-Gaia Ontology and the Agri-Gaia Shape files.

7.2 Data storage

The technologies from Sect. 5 are used during the upload of datasets and models to a platform instance. During this process a sequence of calls is triggered. The frontend first asks the backend via a REST call for possible subclasses of the Data Resource class, which is described in the Agri-Gaia ontologies and inherits from Gaia-X concepts. After the backend receives the request, a SPARQL query is built and sent to the Apache Jena Fuseki endpoint to search for the relevant classes from the Agri-Gaia ontology. The result is mapped and returned to the frontend and enables it to fill those possible classes into a Dropdown Selection, as shown in Fig. 7. This workflow makes it possible to add new classes to the ontology, without editing the code base of the platform.

Based on the selection of the dataset type in the dropdown a similar sequence like described before is executed, this time asking for possible properties which are attached to this class (see Listing Fig. 8).

The result is mapped into the JSON-LD format to enable automatic generation of the form in the frontend and assign the correct data types to the input fields. In a next step the SHACL shapes can be queried to enable input checks in the frontend as well.

After all mandatory inputs are filled, the dataset information and the data itself are passed to the backend, where a validation flow is initiated. First it creates a temporary dataset in the Apache Jena Fuseki Triple storage. The given metadata is saved into the temporary dataset and the Agri-Gaia shapes are loaded from the shape dataset. Afterwards the dataset is validated against the SHACL shapes and a validation report is created, which contains information on missing or wrongly given attributes, if any. In this case, the upload is canceled and the user has to refill the form.

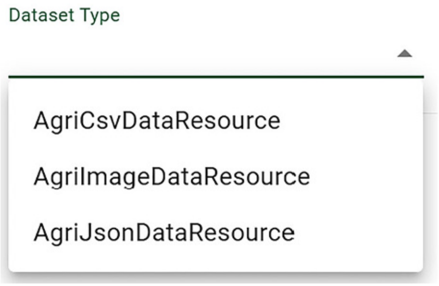

Fig. 7 Exemplary Dropdown menu with Dataset subtypes

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf−schema#>
PREFIX ag: <http://w3id.org/Agri−Gaia−X/asset#>

SELECT ?prop ?label ?range
WHERE
{
        ag:AgriImageDataResource rdfs:subClassOf∗ ?superclass .
        ?prop rdfs:domain ?superclass .
        ?prop rdfs:label ?label .
        ?prop rdfs:range ?range .
        Filter(lang(?label)=’en’)
}
```

Fig. 8 Query to receive all attributes for a class

If the check passes, the metadata is merged into the main dataset and the temporary one is deleted. This flow reduces the runtime of the shape validation, as it only has to check the new inserted information.

7.3 Data exchange

Publishing an asset in the marketplace, or retrieving an asset from another platform, both employ the EDC for transfer. On publishing an asset, all available metadata is retrieved from the triple store and serialized in JSON LD format for the EDC. The attributes needed by EDC itself (e.g. the asset name, publisher or type) are mapped to the EDC catalogue attributes, while the entire metadata set is also stored in an additional attribute to EDC’s catalogue and saved in a separate file. This file is included into a zip archive together with all asset files which should be transferred after a successful contract negotiation via a file transfer executed by the EDC.

The transfer itself uses a push mechanism. To enable this type of transfer the receiver has to provision a temporary credential to his MinIO instance in the first place. The login information will be given to the senders connector, who uses it to build a connection to the remote MinIO. Then the information will be retreived from their own storage and written to the storage of the receiver, This way the receiver gets all metadata information alongside with the asset itself saved on the provider platform and can use the asset or import the metadata into their own triple storage.

8 Conclusion

This paper showed how the project Agri-Gaia handles data and metadata in an ecosystem consisting of multiple participants. According to the needs of the agricultural application scenarios, every participant is allowed to create, add and publish any metadata they deem relevant.

The functionalities to be used by every participant, including the ability to browse, query, and use semantic search extensions, and the understanding of data content have to be ensured by a common vocabulary via a reference to suitable domain ontologies and typical data type definitions.

The approach contributes to a common understanding of metadata for datasets and models by defining an information elements ontology for the whole agricultural ecosystem. This way providers and consumers of data can be sure of what is inside an asset if it is described properly following the structure of this ontology. The metadata ontology can be updated by everyone to share specific knowledge on, for example, required attributes throughout the sector. By providing dynamic interfaces, these changes are immediately present in the forms that a user has to fill whenever uploading an asset without the need to adjust any of the platform code. The exchange of metadata alongside with the data itself between decentralized participants, based on Gaia-X principles, is shown. The paper proposed the usage of MinIO as a data storage, Apache Jena Fuseki as a graph based metadata store solution and EDC to exchange data in this complex agricultural ecosystem.

The implemented prototype is currently being evaluated in selected use cases within Agri-Gaia.

Funding The project Agri-Gaia is supported by the German Federal Ministry of Economic Affairs and Climate Action under grant no. 01MK21004.

Funding Open Access funding enabled and organized by Projekt DEAL.

Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Albertoni R, Browning D, Cox S et al (2020) Data catalog vocabulary (DCAT) – version 2. World Wide Web Consortium. https://www.w3.org/TR/vocab-dcat-2/. Accessed 2023-06-21
2. Aydin S, Aydin MN (2020) Ontology-based data acquisition model development for agricultural open data platforms and implementation of owl2mvc tool. Comput Electron Agric 175:105–589
3. Bansal N, Malik SK (2011) A framework for agriculture ontology development in semantic web. In: 2011 International Conference on Communication Systems and Network Technologies. IEEE, pp 283–286

4. Barels N, Dörr J, Fehrmann J et al (2020) Abschlussbericht Machbarkeitsstudie: Machbarkeitsstudie zu staatlichen digitalen Datenplattformen für die Landwirtschaft. No. 022.20/D Version 1.1 final in IESE Report, Fraunhofer IESE. https://www.bmel.de/SharedDocs/Downloads/DE/_Digitalisierung/machbarkeitsstudie-agrardatenplattform.pdf?__blob=publicationFile&v=3. Accessed 2023-06-21
5. Berners-Lee T, Hendler J, Lassila O (2001) The semantic web: a new form of web content that is meaningful to computers will unleash a revolution of new possibilities. Sci Am 284:1–5
6. Brickley D, Guha RV (2014) RDF schema 1.1. WWW Consortium. http://www.w3.org/TR/rdf-schema/, last accessed on 2023-06-21
7. Clasen M (2021) Über den Wert von Daten in der Landwirtschaft. In: Meyer-Aurich A, Gandorfer M, Hoffmann C et al (eds) 41. GIL-Jahrestagung. Gesellschaft für Informatik e.V., Bonn, pp 61–66
8. Cyganiak R, Wood D, Lanthaler M (2014) RDF 1.1 concepts and abstract syntax. WWW Consortium. http://www.w3.org/TR/rdf11-concepts/. Accessed 2023-06-21
9. Gao SS, Sperberg-McQueen CM, Thompson HS (2012) W3C XML schema definition language (XSD) 1.1 part 1: structures. WWW Consortium. http://www.w3.org/TR/xmlschema11-1/. Accessed 2023-06-21
10. Geo RIG (2004) Exif vocabulary workspace – RDF Schema. WWW Consortium. https://www.w3.org/2003/12/exif/. Accessed 2023-06-21
11. Harris S, Seaborne A (2013) SPARQL 1.1 query language. WWW Consortium. http://www.w3.org/TR/sparql11-query/. Accessed 2023-06-21
12. Härtel I (2020) Gutachten zum Thema "Europäische Leitlinien bzw. Regeln für Agrardaten". Bundesministerium für Ernährung und Landwirtschaft, Berlin (https://www.bmel.de/SharedDocs/Downloads/DE/_Digitalisierung/agrardaten-gutachten-haertel.pdf?__blob=publicationFile&v=2). Accessed 2023-06-21
13. Iannella R, McKinney J (2014) vcard ontology – for describing people and organizations. WWW Consortium. https://www.w3.org/TR/vcard-rdf/. Accessed 2023-06-21
14. Kellogg G, Champin PA, Longley D (2020) JSON-LD 1.1 – A JSON-based serialization for linked data. WWW Consortium. https://www.w3.org/TR/json-ld11/. Accessed 2023-06-21
15. Knublauch H, Kontokostas D (2017) Shapes constraint language (SHACL). WWW Consortium. https://www.w3.org/TR/shacl/. Accessed 2023-06-21
16. Lebo T, Sahoo S, McGuinness D (2013) PROV-O: the PROV ontology. WWW Consortium. http://www.w3.org/TR/prov-o/. Accessed 2023-06-21
17. Miles A, Bechhofer S (2009) SKOS simple knowledge organization system reference. WWW Consortium. http://www.w3.org/TR/skos-reference. Accessed 2023-06-21
18. Motik B, Patel-Schneider PF, Parsia B (2012) OWL 2 web ontology language structural specification and functional-style syntax (second edition). WWW Consortium. http://www.w3.org/TR/owl2-syntax/. Accessed 2023-06-21
19. Perego A, Lutz M (2015) ISA programme location core vocabulary. EU ISA programme core vocabularies working group, WWW Consortium. https://www.w3.org/2015/04/locn.html. Accessed 2023-06-21
20. Peterson D, Gao SS, Malhotra A et al (2012) W3C XML schema definition language (XSD) 1.1 part 2: datatypes. WWW Consortium. http://www.w3.org/TR/xmlschema11-2/. Accessed 2023-06-21
21. Pollock R, Tennison J, Kellogg G et al (2015) Metadata vocabulary for tabular data. WWW Consortium. https://www.w3.org/TR/tabular-metadata/. Accessed 2023-06-21
22. Zheng YL, He QY, Ping Q et al (2012) Construction of the ontology-based agricultural knowledge management system. J Integr Agric 11(5):700–709