frontiers in Sustainable Food Systems

METHODS
published: 30 March 2022
doi: 10.3389/fsufs.2022.767863

A Flexible, Extensible, Machine-Readable, Human-Intelligible, and Ontology-Agnostic Metadata Schema (OIMS)

*Gideon Kruseman**

Sustainable Agrifood Systems Program (SAS), International Maize and Wheat Improvement Center (CIMMYT), Texcoco, Mexico

This paper presents a lightweight, flexible, extensible, machine readable and human-intelligible metadata schema that does not depend on a specific ontology. The metadata schema for metadata of data files is based on the concept of data lakes where data is stored as they are. The purpose of the schema is to enhance data interoperability. The lack of interoperability of messy socio-economic datasets that contain a mixture of structured, semi-structured, and unstructured data means that many datasets are underutilized. Adding a minimum set of rich metadata and describing new and existing data dictionaries in a standardized way goes a long way to make these high-variety datasets interoperable and reusable and hence allows timely and actionable information to be gleaned from those datasets. The presented metadata schema OIMS can help to standardize the description of metadata. The paper introduces overall concepts of metadata, discusses design principles of metadata schemes, and presents the structure and an applied example of OIMS.

Keywords: metadata, interoperability, data management, reusability, JSON (Java Script Object Notation)

OPEN ACCESS

Edited by:
James Hammond, International Livestock Research Institute (ILRI), Kenya

Reviewed by:
Anton Eitzinger, International Center for Tropical Agriculture (CIAT), Colombia
Leo Gorman, The Alan Turing Institute, United Kingdom

***Correspondence:**
Gideon Kruseman
g.kruseman@cgiar.org

Specialty section:
This article was submitted to Land, Livelihoods and Food Security, a section of the journal Frontiers in Sustainable Food Systems

Received: *31 August 2021*
Accepted: *16 February 2022*
Published: *30 March 2022*

Citation:
Kruseman G (2022) A Flexible, Extensible, Machine-Readable, Human-Intelligible, and Ontology-Agnostic Metadata Schema (OIMS). Front. Sustain. Food Syst. 6:767863. doi: 10.3389/fsufs.2022.767863

INTRODUCTION

Prior to the COVID-19 pandemic, it has been estimated that international agricultural research for development (specifically CGIAR) alone collected household survey data from a quarter-million farmers each year. Along with the data collected by other entities and published in open access, such as the World Bank LSMA ISA datasets (https://www.worldbank.org/en/programs/lsms/initiatives/lsms-ISA), the FAO household data (https://data.apps.fao.org/catalog/dataset/household-survey-data-portrait), these datasets can potentially provide valuable insights into smallholder farming and key aspects of agri-food system transformation. However, interoperability of these datasets is a major challenge. In socio-economic data, there is a notable lack of widely accepted standards. Standardization of both questions and the way they are asked is challenging. Questions are often context specific related to the location and cultural context in which the data is collected and the specific research questions underlying the data collection.

There is an increasing requirement for publicly funded research organizations to make the data they collect available as global public goods. While this is the main driver behind many open access/open data initiatives, there are other more compelling reasons to work toward well-organized data repositories. The cost of collecting data (again) often outweighs the

costs of organizing it. Once it is well-organized the data can be repurposed for other research purposes. Very often, only a part of the data collected is actually used in the research for which it was initially intended. Making the data accessible can create value beyond its original purpose for more researchers than those who originally collected and/or analyzed the data. To be able to use and re-use data effectively and efficiently, and to provide data to the world as a global public good it is imperative to implement user friendly data management systems.

Agile, data-oriented research tools can help to overcome these challenges. The term "agile" in this context is used to imply methods that are designed to be easy to use, which entail some degree of flexibility in terms of adaptation to local conditions and integration with other tools or methods. This helps to address the major challenges facing smallholders in the context of agri-food system transformation. Smallholders face complex dynamic circumstances, and the data for analysis of those circumstances are also dynamic and complex. Standardization is often lacking, and approaches are needed to ensure interoperability of the various datasets needed for actionable research.

It is very important to distinguish between data and metadata. In the original FAIR data guidelines (Wilkinson et al., 2016) data and metadata were grouped. FAIR stands for Findable, Accessible, Interoperable and Reusable. Not distinguishing between data and metadata has a certain appeal because metadata itself is a form of data. While that makes sense for the human intelligible aspects of making data FAIR it has a different connotation for the machine-readable aspects of the standards. Because we focus on the standardization of metadata instead of the data it describes, we must be meticulous about our metadata definitions.

We, therefore, need to make a distinction between different types of metadata (Cundiff, 2004; Cantara, 2005). There are three main types of metadata with some subtypes: descriptive metadata, structural metadata and administrative or technical metadata. These concepts are very often used interchangeably leading to poorly defined metadata and hindering open and FAIR data. This will be discussed in some more detail in the next section on design principles. In this context, the notion of structural metadata referring to the actual content of datasets is vitally important. Being able to readily use datasets to address issues related to agri-food system transformation requires a metadata schema that is flexible and extensible.

Both in the private sector as well as in not-for-profit organizations, the issue of data management in relation to the ever-increasing amount of data is a hot topic. In the domain of agricultural research this is no different, data from different scientific domains are present and increasingly there is a need to combine data from different domains to glean new insights. There is a heated and ongoing debate on the concept of data lakes and its usefulness for managing the ever-increasing volume, variety, and velocity of data. All organizations must adopt data management strategies that keep up with the advent of big data if we hope to conduct research effectively and accurately. In the private sector, data management is often referred to as master data management (MDM) (Rittman, 2008) which comprises the processes, governance, policies, standards, and tools that consistently define and manage an organization's critical data to provide a single point of reference. Metadata is an essential component of data management. In the context of international agricultural research for development, data management complexity is even greater as data is coming from many different sources.

Twentieth century data management strategies focused on ensuring data was made available in standard formats and structures in databases and/or data warehouses (Inmon, 1992; Russom, 2013)—a combination of many different databases across an entire enterprise (Oracle, 2002). The major drawback of the data warehouse concept is that it works like a straitjacket acting as a disincentive to corporate level data repositories.

One alternative storage and retrieval system that can handle high variety data is the data lake. It is one of the newest flavors in MDM (O'Brien, 2012; Cap Gemini Pivotal, 2013; Knowledgent, 2014; PWC, 2015). While it is still a controversial concept it is the most promising for research purposes. Data lakes are a store-everything approach to big data, and is a massive, easily accessible, centralized repository of large volumes of structured and unstructured data. The Data Lake is a data-centered architecture featuring a repository or set of repositories capable of storing vast quantities of data in various formats. Data from many different sources such as webserver logs, data bases, sensors, satellites, surveys, social media, and third-party data is ingested into the Data Lake.

However, without metadata—information that describes the data we are collecting—and a mechanism to maintain it, data lakes can become data swamps where data is murky, unnavigable, has unknown origins, and is ultimately unreliable. Every subsequent use of data means, scientists and researchers start from scratch. Metadata also allows extraction, transformation, and loading (ETL) processes to be developed and take place, which retrieve data from operational systems and process it for further analysis (Lane, 2005). The data collected in international agricultural research often resembles a data swamp instead of a data lake. Data sets often lack adequate metadata. If metadata is present, it tends to be limited to descriptive metadata. In the case some detailed structural metadata is provided, this is often in the form of an idiosyncratic data dictionary.

In international agricultural research for development focusing on the transformation of complex dynamic agri-food systems, data from many different domains are used from genomic data, remote sensing and satellite data, and crop management data to socio-economic data. Some of these data have some level of standardization like genomic data, while for instance socio-economic data consisting of high variety structured, semi-structured and unstructured data suffers from an almost complete lack of standardization.

In this paper, the first version of a light-weight metadata schema is presented that is flexible and extensible so that it can be used for the wide variety of household-level datasets used for the analysis of smallholders and agri-food system transformation. In the next section, the design principles are discussed, followed by the structure of the metadata schema. The approach and metadata schema are then used to tag a portion of a farm household dataset as an example.

DESIGN PRINCIPLES

Metadata Typology

As mentioned earlier, metadata can be subdivided into categories (Cundiff, 2004). The first type of metadata is administrative metadata. Administrative metadata relates to the technical source of a digital asset. It can be subdivided into three subtypes: technical metadata which we define as *information necessary for decoding and rendering files*; Preservation metadata which is defined as *information necessary for the long-term management and archiving of digital assets*; and rights metadata defined as *information pertaining to intellectual property and usage rights*.

Descriptive metadata is essential for discoverability and identification of digital assets. This is the most common type of metadata used for finding relevant data assets in open data repositories. It describes the data asset in terms of concepts such as "author," "title," "publisher," "abstract" and "keywords," to name a few. This is the most common type of metadata attached to research data information products. Examples include Dublin Core metadata schema (https://dublincore.org/) and the CG Core metadata schema (Devare, 2017).

Structural metadata is data that indicates how a digital asset is organized and that act as identifiers and descriptors of the data. Structural metadata facilitates content reuse by providing detailed information about the structure of the content of the digital asset. It can therefore be defined as *data defining the logical components of complex or compound objects and how to access those components*. Structural metadata comprises most of what is traditionally considered metadata that is organized as the data dictionary, and can include: data element information, table information or record structure information, depending on the data asset.

The lack of structural metadata in an easily accessible way that allows searching high variety datasets is arguably the greatest challenge to turning existing data into new actionable information (Rasmussen, 2018). Metadata schemas tend to be focused on specific domains (Canham and Ohmann, 2016), stop at a very high conceptual level (Shukair et al., 2013) or focus on descriptive metadata with a fixed structure (Devare, 2017; Labropoulou et al., 2020). Specific metadata schemas exist for specific datasets. For socio-economic datasets the Document, Discover and Interoperate (DDI) approach (https://ddialliance.org/) exists (Rasmussen, 2014).

The DDI/XML approach to managing metadata is elegant and comprehensive, but due to its complexity, very difficult for individuals to manage on their own, because it usually requires large scale projects to implement with varying success (Vardigan et al., 2015). DDI as a metadata schema for socio-economic data was first developed in the mid-1990s (Vardigan, 2014). A key example of successful implementation in the domain of smallholder agriculture is the World Bank LSMS ISA datasets that use the metadata approach. In agricultural research for development, there are seldom sufficient resources to implement a heavy weight approach like DDI. Moreover, investment in a heavy-weight approach makes more sense when the same types of data are collected on a regular basis in multiple settings by the same organizations managing the data assets. The key lesson that can be drawn from the DDI experience is that there is a need for a light-weight approach that is compatible with other approaches.

Data Entity Approach

A Data Entity, is a top level container of information (Esteva et al., 2019). From a machine perspective it is most relevant as a data object that has a unique uniform resource identifier (URI) (Berners-Lee et al., 2005) and at least some technical metadata. From the human perspective, the relevancy of a data entity is that of a data concept, something that has meaning for humans and hence has some descriptive metadata. Data objects and data concepts can coincide but do not have to do so necessarily. An example of a data entity as a concept and not an object is a data collection. A data collection is defined here as a number of datasets that are somehow related. The datasets are data objects with a distinct URI. The data collection encompasses data objects but is not a data object itself.

Data entities can have parent child relationships. An example is a dataset. A farm household-survey dataset in an open-access repository is an example of a data entity, it typically has metadata describing the study, study area, authors, and contributors. It is a parent with children. The children are for instance the various data files in the dataset. Household surveys often have numerous data files covering the various interlinked tables. It is these data files that require a flexible metadata schema to describe their contents.

Data entities can also be the various supporting documentation files as well as all the relevant metadata files.

Rich Metadata Beyond Ontologies

Structural metadata describes the contents of a data file. An example of a structural metadata file is a data dictionary. Statistical software packages such as STATA (https://www.stata.com/) commonly used for farm-household data analysis actually contain some of the basic structural metadata:

- Name: variable names
- Label: short description of the variable
- Type: data type
- Format: specific format of the variable

Other metadata is not included in the STATA metadata but is essential to understand the structure and content of files. Examples of key metadata that cannot be gleaned from the STATA data files include information on primary and foreign keys and information on controlled vocabularies when code books have been used. Some metadata fields may be relevant in some cases but not others such as the way information was captured, if a variable contains restricted information, such as personally identifiable information or information on data quality.

What is deemed useful metadata depends on context, international best practices, and organizational data policies. Therefore, the metadata schema must be flexible and extensible. The flexibility also pertains to the fact that some of the metadata may actually be included in the data file in another field. While this is perfectly understandable for a human, it can be tricky to

program machines that parse datasets. It is therefore important to include this kind of information in a standardized way in the metadata file.

In recent years, within the realm of data management for agricultural research for development, a strong focus has been placed on ontologies (Arnaud et al., 2020). Ontologies are important components of formal descriptions of knowledge. They are useful where there is strong agreement about terms and their relationships. Ontologies are important, arguably necessary but are in themselves not sufficient for data interoperability. Ontologies can provide structured content in terms of values used in the metadata.

Formalized definitions of concepts are essential for interoperability across high variety datasets. Ontologies can play an important role in that formalization. Many different pre-existing domain-agnostic standards in terms of ontologies exist as well as domain-specific ones. Creating interoperability requires ontology term mappings when different ontologies are used to tag concepts. Within a metadata schema that does not depend on a single ontology, it therefore becomes essential to identify which ontologies are used to tag concepts.

Summary of Design Principles

In summary the following design principles emerge for a metadata schema that can be used easily for the high variety datasets that characterize the domain of small holder farming and agri-food system transformation analysis.

High variety implies that the schema must be **flexible** to accommodate all kinds of data. The schema must be **extensible** to address new issues, including demands for different types of metadata that currently are not prioritized. Metadata approaches such as DDI provide that flexibility and extensibility but are cumbersome to use and the metadata is not very user-friendly or human-intelligible (Amin et al., 2012). A **lightweight** approach that is **human-intelligible** is therefore the way forward. Obviously, the approach should be **machine-readable**, requiring a formalized structure in a generally accepted format. We are not operating in a vacuum, so the metadata approach should take advantage of any work already done. Ideally allowing for the automatic incorporation of existing and new data dictionary approaches. While formalized knowledge in terms of ontologies is an essential component of interoperability the approach should not be dependent on a single ontology. Being **ontology-agnostic** and able to incorporate existing metadata approaches is part of the flexibility and extensibility already highlighted as design criteria. For **reproducibility**, a versioning system must be included. For **transparency**, the information about the schema must be available in open access with relevant documentation. Furthermore, transparency requires a method that allows comments to be included meant for humans and not machines.

The DDI/XML approach was designed to allow interoperability with other metadata schemas. The same principle was used for OIMS. This implies that in principle, metadata should be exchangeable between the two approaches. Obviously, this comes at some cost as it requires a metadata schema to be described in terms of the other schema.

STRUCTURE OF OIMS

The fundamental discussion between flexibility and standardization is at the core of the way OIMS is structured. The questions we asked are provided here.

1. Would be possible to describe datasets that already have data dictionaries or other metadata without having to redo all the work data managers have already put into the process?
2. If we wanted to add another metadata element to a data dictionary, how can that be done without overturning everything?

The metadata as description of the data itself is less domain specific and less context specific as the data itself. So, if we can standardize the way we describe the metadata in such a way that it can describe any metadata field, we have the standardization we want and the flexibility. If we want to add a metadata field to a data dictionary, the OIMS schema allows us to describe the field in a standardized way.

In the following subsections we provide some of the technical details that allow this flexibility and standardization.

Metadata Schema Format

For the metadata to be machine readable, it needs to be in a format that is standard and flexible. **JSON** (Java Script Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate (http://www.json.org/). The JSON format allows both single objects as well as arrays. See **Figure 1** for a graphical representation of JSON.

The DDI metadata approach uses XML (eXtensible Markup Language). JSON and XML are comparable in flexibility and use. The main reason for choosing JSON is that parsing JSON is much faster than parsing XML.

A Metadata Schema to Describe the Data Dictionary

As mentioned earlier metadata is data itself and hence should have metadata attached to it. In order to describe a metadata schema, we need a format for doing so.

Each metadata field can be described with the following elements

- **AttributeName**: the identifier of the metadata field is a required element both for machine-readability (MR) as well as it being human-intelligible (HI)
- **AttributeDescription**: A short description of what the metadata field entails is a required element for HI
- **DataType**: the data type of the contents of the metadata field. This allows consistency checking in automated quality assurance tools and is handy for data management purposes
- **Status**: determine if a field is: required; recommended; required if applicable; recommended if applicable; optional
- **TypeClass**: identification if a metadata field consists of multiple attributes or if the attribute has only a single value. TypeClass has two possible values: primitive and compound

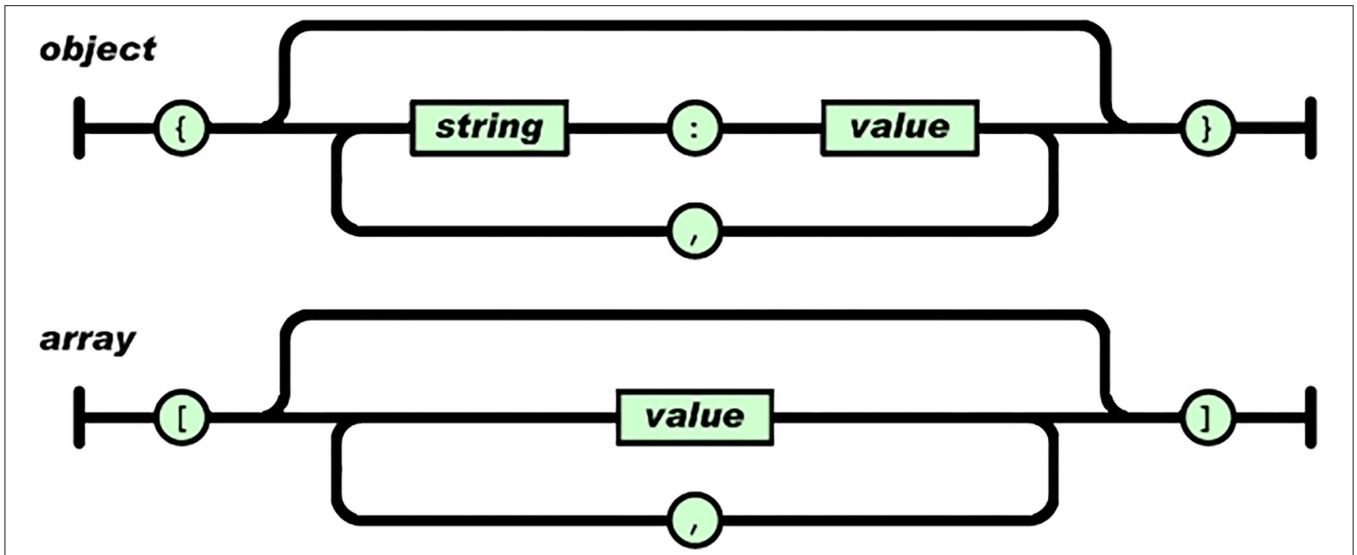


FIGURE 1 | Structure of JSON.

- **Multiple**: does a metadata field allow multiple entries or not. Multiple has two possible values: TRUE or FALSE
- **OntologyTerm**: attribute value identifier is a compound field consisting of several sub-elements.
  - **OntologyTermName**: ontology term
  - **OntologyTermDescription**: short HI description of the ontology term as defined in the relevant ontology
  - **OntologyName**: The name of the ontology in which the term is defined
  - **OntologyTermID**: This is a MR element
  - **OntologyTermURL**: This provides the link to the ontology term through a persistent identifier
  - **OntologyTermQuality**: This provides information on how well the ontology term fits the metadata field

If the data type is enumeration, then there is a controlled vocabulary linked to that field

- **ControlledVocabulary**: This is a compound element providing the description of the controlled vocabulary
  - **VocabularyElementID**: element identifier of a unique element of the controlled vocabulary
  - **VocabularyElementDescription**: description of the element
  - **OntologyTerm**: element identifier is a compound field consisting of several sub-elements. For their description see above.
    - **OntologyTermName**
    - **OntologyTermDescription**
    - **OntologyName**
    - **OntologyTermID**
    - **OntologyTermURL**
    - **OntologyTermQuality**

We can therefore describe the elements of the metadata-metadata in the same terms as well, albeit that these may contain different elements depending on the schema. In the end we can describe the elements of the metadata-metadata-metadata in terms of themselves which then becomes the basis for the description of any metadata schema.

This is a standardized approach to describing metadata, in other words a standard metadata of metadata (data dictionaries). Because it can describe any metadata field in a standardized way, the schema is both flexible and standardized.

Self-Describing Metadata

So, at the highest level of abstraction, we have a metadata schema that describes itself. We can use this schema to describe any metadata of metadata schemas that may contain additional elements. It is more intuitive and more self-contained than for instance the RDF schema (Dan Brickley and Guha, 2014). Besides describing the OIMS metadata schema, the self-describing OIMS schema can also be used to describe any other metadata schema. Besides describing already existing data dictionaries, OIMS can be used to describe for instance data entities and their relationships as well as ETL procedures.

In addition to the eight attribute attributes there is one more that we use, namely the attribute "//" in our JSON files which we use as a comment. This allows us to add user friendly information that is not needed by a machine parsing the metadata file.

For ontology terms we have mostly used the very complete and comprehensive NCI Thesaurus OBO Edition (http://www.obofoundry.org/ontology/ncit.html) accessed through the EMBL-EBI ontology look-up service (https://www.ebi.ac.uk/ols/index). In the OIMS approach multiple ontology terms can be attached to an attribute and this is expected to happen in next versions.

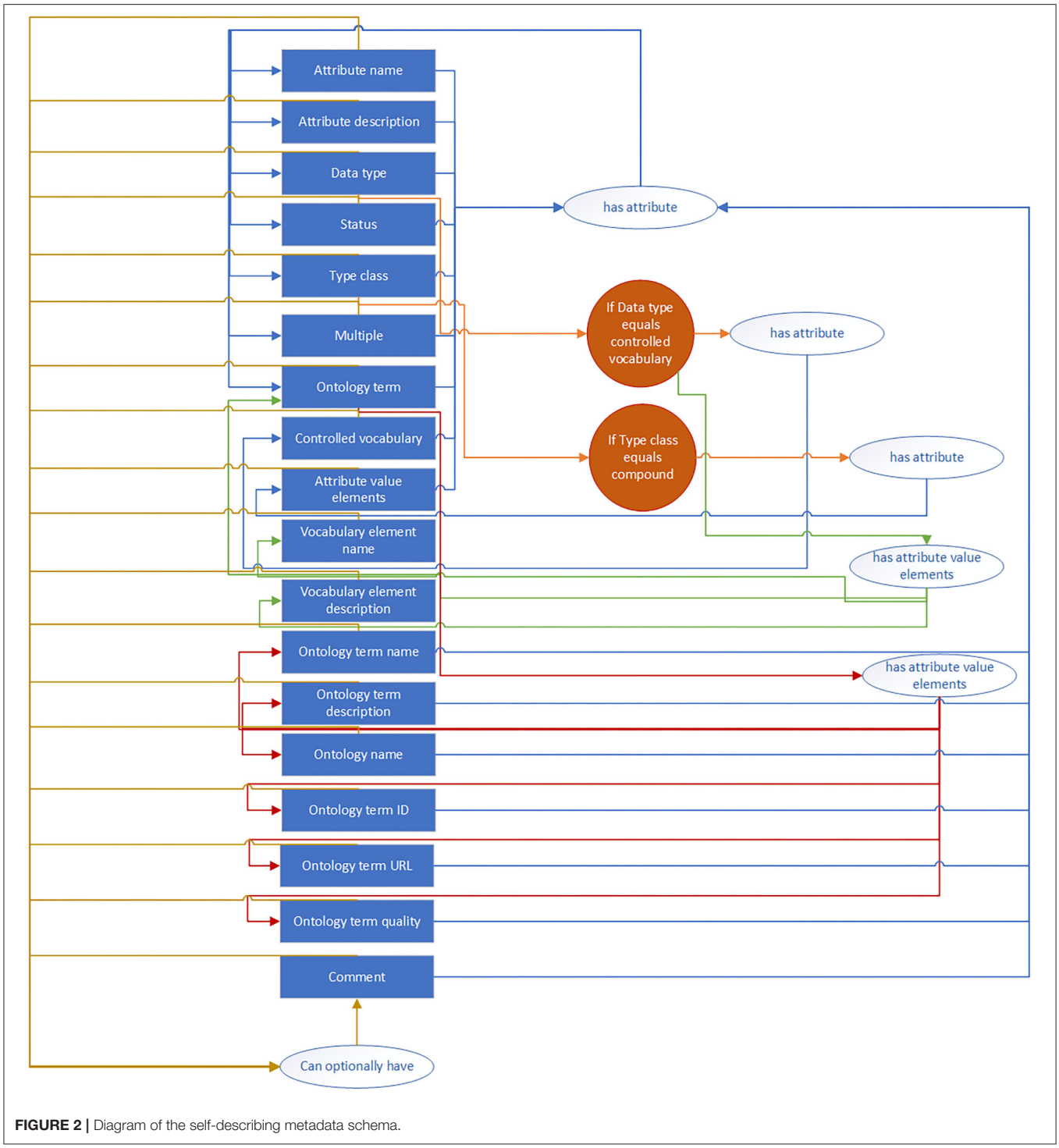


FIGURE 2 | Diagram of the self-describing metadata schema.

In the next subsections we provide a description of each of the elements of the self-describing metadata schema. For the description of the schema in JSON format, see the **Supplementary Material**: "data sheet 1.docx." In **Figure 2** we see a schematic diagram of the self-describing metadata schema.

The description of the self-describing metadata is purposely done in a way that closely resembles the structured format underlying the schema instead of a more narrative-like approach. The importance is to provide the definitions of all the elements as highlighted in **Figure 2**.

The diagram in **Figure 2** demonstrates that each attribute in the blue boxes has at least the first seven attributes. The attribute data type can have the value "controlled vocabulary." In that case an additional attribute is needed to describe the

controlled vocabulary. It has the specific attribute value elements linked to controlled vocabulary as well as the more general attribute ontology term. Ontology terms have specific attribute value elements as well-related to ontology terms. These are two examples of compound attributes and hence they have the related attribute that captures these compound elements. Note that this schema is somewhat different from a standard database schema. It defines everything that can then be used in the metadata schemas describing actual data.

Attribute Name

Take for instance the concept **AttributeName**. It can be described in terms of the major attributes:

- **AttributeName = AttributeName**
- **AttributeDescription =** the name of a metadata field
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = false
- **OntologyTerm** =
    - **OntologyTermName**: Name
    - **OntologyTermDescription**: The words or language units by which a thing is known
    - **OntologyName:** NCIT
    - **OntologyTermID**: C42614
    - **OntologyTermUR**: http://purl.obolibrary.org/obo/NCIT_C42614
    - **OntologyTermQuality**: to be confirmed

Attribute Description

In a similar vein, **AttributeDescription** can be described in terms of the major attributes:

- **AttributeName = AttributeDescription**
- **AttributeDescription =** Description of a metadata field
- **DataType** = character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = false
- **OntologyTerm** = combination of:
    - **OntologyTermName**: Description
    - **OntologyTermDescription**: A written or verbal account, representation, statement, or explanation of something.
    - **OntologyName:** NCIT
    - **OntologyTermID**: C25365
    - **OntologyTermUR**: http://purl.obolibrary.org/obo/NCIT_C25365
    - **OntologyTermQuality**: to be confirmed.

Data Type

DataType is a bit more complex as it has a controlled vocabulary that needs to be defined. It can be described as:

- **AttributeName = DataType**
- **AttributeDescription =** The datatypes of the various fields of the self-describing metadata schema
- **DataType** = Controlled vocabulary
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = true
- **ControlledVocabulary** = a set of unique elements containing
    - Simple character string defined as
        - **VocabularyElementID**: Simple character string
        - **VocabularyElementDescription**: simple machine-readable language independent sequence of characters
        - **OntologyTerm**:
            - **OntologyTermName**: Simple Character String Data Type
            - **OntologyTermDescription**: A data type comprised of a text string that can be displayed, or machine processed, and which has no language.
            - **OntologyName:** NCIT
            - **OntologyTermID:** C95682
            - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C95682
            - **OntologyTermQuality**: to be confirmed
    - String defined as
        - **VocabularyElementID**: String
        - **VocabularyElementDescription**: An expression consisting of a linear sequence of symbols (characters or words or phrases).
        - **OntologyTerm**:
            - **OntologyTermName**: String
            - **OntologyTermDescription**: An expression consisting of a linear sequence of symbols (characters or words or phrases).
            - **OntologyName:** NCIT
            - **OntologyTermID**: C45253
            - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C45253
            - **OntologyTermQuality**: to be confirmed
    - Boolean defined as
        - **VocabularyElementID**: Boolean
        - **VocabularyElementDescription**: The type of an expression with two possible values, "true" and "false."
        - **OntologyTerm**:
            - **OntologyTermName**: Boolean
            - **OntologyTermDescription**: The type of an expression with two possible values, "true" and "false."
            - **OntologyName:** NCIT
            - **OntologyTermID:** C45254
            - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C45254
            - **OntologyTermQuality**: exact

- Controlled vocabulary defined as
  - **VocabularyElementID**: Controlled vocabulary
  - **VocabularyElementDescription**: set of unique elements that are the only valid values of a variable, also known as enumeration or in R terminology a factor.
  - **OntologyTerm**:
    - **OntologyTermName**: Controlled vocabulary
    - **OntologyTermDescription**: A set of terms that are selected and defined based on the requirements set out by the user group, usually a set of vocabulary is chosen to promote consistency across data collection projects.
    - **OntologyName:** NCIT
    - **OntologyTermID**: C25704
    - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C25704
    - **OntologyTermQuality**: to be confirmed
- Text defined as
  - **VocabularyElementID**: Text
  - **VocabularyElementDescription**: sequence of strings
  - **OntologyTerm**:
    - **OntologyTermName**: Text
    - **OntologyTermDescription**: The words of something written.
    - **OntologyName:** NCIT
    - **OntologyTermID**: C25704
    - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C25704
    - **OntologyTermQuality**: to be confirmed
- HTML defined as
  - **VocabularyElementID**: HTML
  - **VocabularyElementDescription**: Hypertext Markup Language
  - **OntologyTerm**:
  - **OntologyTermName**: Hypertext Markup Language
    - **OntologyTermDescription**: A standard markup language used to display content on a web page, as specified by the World Wide Web Consortium (W3C).
    - **OntologyName:** NCIT
    - **OntologyTermID:** C142380
    - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C142380
    - **OntologyTermQuality**: to be confirmed
- Compound
  - **VocabularyElementID**: compound
  - **VocabularyElementDescription**: the datatype of the attribute is an array of elements possibly but not necessarily with different datatype combinations
  - **OntologyTerm**: no ontology term available

- **OntologyTerm** = combination of
  - **OntologyTermName**: DataType
  - **OntologyTermDescription**: An indication of the form that a value will have.
  - **OntologyName:** NCIT
  - **OntologyTermID**: C42645
  - **OntologyTermUR**: http://purl.obolibrary.org/obo/NCIT_C42645
  - **OntologyTermQuality**: to be confirmed

Note that the controlled vocabulary contains only the data types needed to describe the self-describing metadata schema.

Status

The attribute **Status** is complex as it has a controlled vocabulary that needs to be defined. It can be described as follows:

- **AttributeName = Status**
- **AttributeDescription =** identification if a metadata field is either: required; recommended; required if applicable; recommended if applicable; optional
- **DataType** = Controlled vocabulary
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **ControlledVocabulary** = a set of unique elements containing
  - required
    - **VocabularyElementID**: required
    - **VocabularyElementDescription**: indication if the attribute is mandatory
    - **OntologyTerm**:
      - **OntologyTermName**: Required Indicator
      - **OntologyTermDescription**: An indication as to whether entity is mandatory.
      - **OntologyName:** NCIT
      - **OntologyTermID:** C164599
      - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C164599
      - **OntologyTermQuality**: to be confirmed
  - recommended
    - **VocabularyElementID**: recommended
    - **VocabularyElementDescription**: indication as to whether attribute is not mandatory but recommended
    - **OntologyTerm**: no ontology term available
  - required if applicable
    - **VocabularyElementID**: required if applicable
    - **VocabularyElementDescription**: required if applicable
    - **OntologyTerm**: no ontology term available

- Recommended if applicable
  - **VocabularyElementID**: recommended if applicable
  - **VocabularyElementDescription**: recommended if applicable
  - **OntologyTerm**: no ontology term available
- Optional
  - **VocabularyElementID**: Optional
  - **VocabularyElementDescription**: optional attribute of a metadata field
  - **OntologyTerm**:
    - **OntologyTermName**: Optional
    - **OntologyTermDescription**: Possible but not necessary; left to personal choice.
    - **OntologyName:** NCIT
    - **OntologyTermID**: C25603
    - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C25603
    - **OntologyTermQuality**: to be confirmed
- **OntologyTerm** = no ontology term available

Note that some of the ontology terms are missing for this attribute. This does not imply that this is a new term. It is used in the JSON metadata file of DataVerse (https://dataverse.org/), an open access data repository system commonly used for international agricultural research for development.

Type Class

The attribute type class can be described as follows:

- **AttributeName = TypeClass**
- **AttributeDescription =** if the attribute is compound or primitive
- **DataType** = Controlled vocabulary
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **ControlledVocabulary** = a set of unique elements containing
  - primitive
    - **VocabularyElementID**: primitive
    - **VocabularyElementDescription**: the attribute does not have underlying attributes
    - **OntologyTerm**: no ontology term available
  - compound
    - **VocabularyElementID**: compound
    - **VocabularyElementDescription**: the attribute has underlying attributes
    - **OntologyTerm**: no ontology term available
- **OntologyTerm** = no ontology term available

Note that some of the ontology terms are missing for this attribute. This does not imply that this is a new term. It is used in the JSON metadata file of DataVerse (https://dataverse.org/).

Multiple

The attribute multiple can be described as:

- **AttributeName = Multiple**
- **AttributeDescription =** can the attribute have multiple values
- **DataType** = Boolean
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = combination of
  - **OntologyTermName**: Multiple
  - **OntologyTermDescription**: Having, relating to, or consisting of more than one individual, element, part, or other component; manifold.
  - **OntologyName:** NCIT
  - **OntologyTermID**: C17648
  - **OntologyTermUR**: http://purl.obolibrary.org/obo/NCIT_C17648
  - **OntologyTermQuality**: to be confirmed

This attribute is also used in the JSON metadata file of DataVerse (https://dataverse.org/).

Controlled Vocabulary

Controlled vocabulary is the code book of a specific controlled vocabulary used in an attribute. It is linked to the data type Controlled Vocabulary. It is an array of values that have multiple attributes themselves.

The attribute, controlled vocabulary, can be described as:

- **AttributeName = ControlledVocabulary**
- **AttributeDescription =** controlled vocabulary definition if data type is controlled vocabulary also known as an enumeration or a factor in R.
- **DataType** = Compound
- **Status** = required if applicable
- **TypeClass** = compound
- **Multiple** = TRUE
- **AttributeValueElements** = when a data type is compound the array elements of the compound data type must be described:
  - **VocabularyElementName**
  - **VocabularyElementDescription**
  - **OntologyTerm**
- **OntologyTerm** = combination of
  - **OntologyTermName**: Controlled vocabulary
  - **OntologyTermDescription**: A set of terms that are selected and defined based on the requirements set out by the user group, usually a set of vocabulary is chosen to promote consistency across data collection projects.
  - **OntologyName:** NCIT
  - **OntologyTermID**: C25704
  - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C25704
  - **OntologyTermQuality**: to be confirmed

Vocabulary Element Name
The specific attribute value elements linked to the compound data type of a controlled vocabulary include the identifier of a controlled vocabulary element

- **AttributeName = VocabularyElementName**
- **AttributeDescription =** the element identifier in a controlled vocabulary
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Vocabulary Element Description
The specific attribute value elements linked to the compound data type of a controlled vocabulary include the description of a controlled vocabulary element

- **AttributeName = VocabularyElementDescription**
- **AttributeDescription =** the description of an element in a controlled vocabulary in human-intelligible terms
- **DataType** = text
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Ontology Term
Ontology term is described elsewhere in section Structure of OIMS. However, ontology term in the context of a controlled vocabulary has a special significance. Ontologies are as said before the formalization of knowledge at a conceptual level. When dealing with the variable names, ontology terms provide the conceptual basis for semantic interoperability. When dealing with the values of the variables, many alternative classifications exist and are used. The lack of standardization hinders interoperability. To improve interoperability various classifications of possible values can be mapped onto each other creating the basis for interoperability of actual data. The details of the processes and procedures related to the creation of such concordances goes beyond the scope of the current paper.

Ontology Terms

Ontology terms can be added for semantic interoperability. Modern data portals such as GARDIAN (https://gardian.bigdata.cgiar.org/), rely on formalized ontology terms for enhanced data interoperability. An ontology is a formal representation of a body of knowledge within a given domain. Ontologies usually consist of a set of classes (or terms or concepts) with relations that operate between them.

Ontology terms can be described as:

- **AttributeName = OntologyTerm**
- **AttributeDescription**: the ontology term for the relevant attribute
- **DataType** = compound
- **Status** = recommended
- **TypeClass** = compound
- **Multiple** = TRUE
- **AttributeValueElements** = when a data type is compound the array elements of the compound data type must be described:
  - **OntologyTermName**
  - **OntologyTermDescription**
  - **OntologyName**
  - **OntologyTermID**
  - **OntologyURL**
  - **OntologyTermQuality**
- **OntologyTerm** = combination of
  - **OntologyTermName**: Ontology term
  - **OntologyTermDescription**: A term (name) from an ontology
  - **OntologyName:** EDAM
  - **OntologyTermID**: data:0966
  - **OntologyTermURL**: http://edamontology.org/data_0966
  - **OntologyTermQuality**: to be confirmed **Or**
  - **OntologyTermName**: Ontology concept
  - **OntologyTermDescription**: A unique entry or term in a specific ontology
  - **OntologyName:** NCIT
  - **OntologyTermID**: C89273
  - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C89273
  - **OntologyTermQuality**: to be confirmed

Note that we provide two ontology terms for the attribute ontology term.

Ontology Term Name
The attribute OntologyTermName which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyTermName**
- **AttributeDescription**: the identifier of an ontology term for the relevant attribute
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Ontology Term Description
The attribute OntologyTermDescription which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyTermDescription**
- **AttributeDescription**: the description of an ontology term for the relevant attribute in human-intelligible terms
- **DataType** = text
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE

- **OntologyTerm** = no ontology term available

Ontology Name
The attribute OntologyName which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyName**
- **AttributeDescription**: the name of the ontology that describes the ontology term
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Ontology Term Identifier
The attribute OntologyTermIdentifier which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyTermIdentifier**
- **AttributeDescription**: the unique identifier for the term within the ontology
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Ontology URL
The attribute OntologyURL which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyURL**
- **AttributeDescription =** persistent URI of the ontology term
- **DataType** = HTML
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **OntologyTerm** = no ontology term available

Ontology Term Quality
The attribute OntologyTermQuality which is part of the compound datatype of the value of an ontology term can be described as:

- **AttributeName = OntologyTermQuality**
- **AttributeDescription**: the degree to which the ontology term covers the attribute
- **DataType** = controlled vocabulary
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = FALSE
- **ControlledVocabulary**
  - Exact match
    - **VocabularyElementID**: exact match
    - **VocabularyElementDescription**: the ontology terms match the attribute exactly
    - **OntologyTerm**: no ontology term available
  - To be confirmed
    - **VocabularyElementID**: to be confirmed
    - **VocabularyElementDescription**: the quality of the ontology term in describing the attribute needs to be confirmed
    - **OntologyTerm**: no ontology term available
- **OntologyTerm** = no ontology term available

Attribute Value Elements

When a data type is compound the array elements of the compound data type must be described. The attribute AttributeValueElements provides that list. And can formally be described as:

- **AttributeName = AttributeValueElements**
- **AttributeDescription**: Attributes that are part of a compound attribute
- **DataType** = simple character string
- **Status** = required
- **TypeClass** = primitive
- **Multiple** = TRUE

Comment

As we mentioned earlier, we include a comment attribute. This allows us to add comments to improve transparency and understandability of metadata files. When parsing the JSON file, comments can be skipped by the machine reading the metadata.

- **AttributeName = //**
- **AttributeDescription**: comment
- **DataType** = text
- **Status** = optional
- **TypeClass** = primitive
- **Multiple** = TRUE
- **OntologyTerm** = combination of
  - **OntologyTermName**: comment
  - **OntologyTermDescription**: A written explanation, observation or criticism added to textual material.
  - **OntologyName:** NCIT
  - **OntologyTermID**: C25393
  - **OntologyTermURL**: http://purl.obolibrary.org/obo/NCIT_C25393
  - **OntologyTermQuality**: exact match

EXAMPLE USING OIMS TO DESCRIBE A DATA FILE

Stepwise Description of Data and Metadata

As an example, we use a small section from a household survey file containing household rosters identifying household composition and household member characteristics, see **Figure 3** for a screenshot from STATA data viewer.

We observe the following variables:

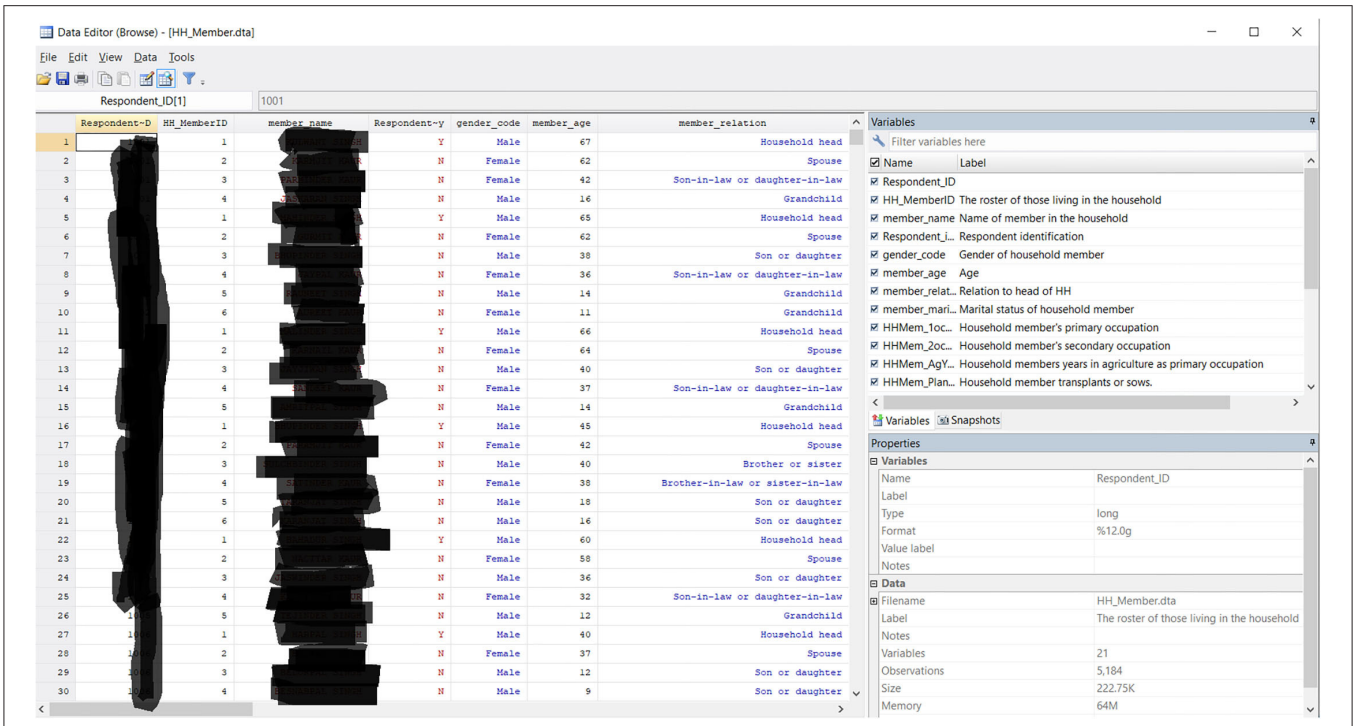
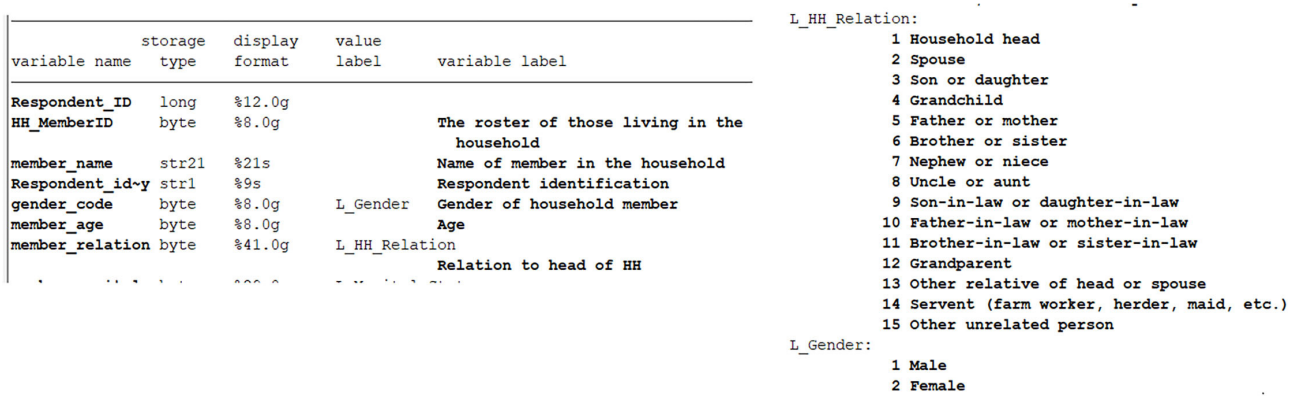


FIGURE 3 | Screen shot of an example data file of household composition. Personally identifiable information has been blacked out.



```
              storage   display    value
variable name   type    format     label      variable label
Respondent_ID   long    %12.0g
HH_MemberID     byte    %8.0g                 The roster of those living in the
                                                household
member_name     str21   %21s                  Name of member in the household
Respondent_id~y str1    %9s                   Respondent identification
gender_code     byte    %8.0g      L_Gender   Gender of household member
member_age      byte    %8.0g                 Age
member_relation byte    %41.0g     L_HH_Relation
                                              Relation to head of HH
```

```
L_HH_Relation:
           1 Household head
           2 Spouse
           3 Son or daughter
           4 Grandchild
           5 Father or mother
           6 Brother or sister
           7 Nephew or niece
           8 Uncle or aunt
           9 Son-in-law or daughter-in-law
          10 Father-in-law or mother-in-law
          11 Brother-in-law or sister-in-law
          12 Grandparent
          13 Other relative of head or spouse
          14 Servent (farm worker, herder, maid, etc.)
          15 Other unrelated person
L_Gender:
           1 Male
           2 Female
```

FIGURE 4 | Extractable metadata from STATA.

- Household_ID
- HH_memberID
- Member_name
- Respondent_identity
- Gender_code
- Member_age
- Member_relation

The STATA file itself already contains some key metadata (see **Figure 4**):

- VariableName
- VariableDescription
- DataType
- Format

Moreover, extracting the metadata from the STATA file allows us to create a list of the elements of enumerations. However, these enumerations are locally defined classifications that do not necessarily have a relationship with some standard classification.

To enhance the reusability of the data CIMMYT, the holder of this particular dataset, is encouraging tagging data with rich metadata including but not limited to:

- Key: whether a variable is a primary key, foreign key, or a regular variable. A primary key is used to ensure data in the specific column is unique. A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It uniquely identifies a record in the relational database table. Only one primary key is allowed in a table.

TABLE 1 | Data dictionary of single primitive fields and simple enumerations.

Variable in the dataset	Metadata field	Value
Household_ID	VariableName	Household_ID
	VariableDescription	Unique identifier of the household in the survey
	DataType	Long integer
	Format	12 characters
	Key	2
	Unit of measurement	NA
	Method of measurement	NA
	Sensitivity	NA
HH_memberID	VariableName	HH_memberID
	VariableDescription	Unique identifier of the household member within a household
	DataType	Integer
	Format	8 characters
	Key	0
	Unit of measurement	NA
	Method of measurement	NA
	Sensitivity	NA
Member_name	VariableName	Member_name
	VariableDescription	Name of the household member
	DataType	String
	Format	21 characters
	Key	0
	Unit of measurement	NA
	Method of measurement	Interview
	Sensitivity	PII
Respondent_ identity	VariableName	Respondent_identity
	VariableDescription	Is this the household head
	DataType	Enumeration
	Enumeration	Y = respondent N = other household member
	Format	9 characters
	Key	0
	Unit of measurement	NA
	Method of measurement	Interview
	Sensitivity	No
Gender_code	VariableName	Gender_code
	VariableDescription	Gender of the household member
	DataType	Enumeration
	Enumeration	Male Female
	Format	8 characters
	Key	0
	Unit of measurement	NA
	Method of measurement	Interview
	Sensitivity	Indirect PII
Member_age	VariableName	Member_age
	VariableDescription	Age of household member
	DataType	Numeric
	Format	8 characters
	Key	0
	Unit of measurement	Years
	Method of measurement	Interview
	Sensitivity	Indirect PII
Member_relation	VariableName	Member_relation
	VariableDescription	Relation to the household head
	DataType	Enumeration
	Format	41 characters
	Key	0
	Unit of measurement	NA
	Method of measurement	Interview
	Sensitivity	Indirect PII

- Unit of measurement
- Method of measurement
- Sensitivity of the data

The data dictionary for these seven variables therefore contains seven primitive fields and a complex one for defining any controlled vocabularies. For the description of the schema in JSON format, see the **Supplementary Material**: "data sheet 2.docx."

Note that this table does not file does not contain a primary key. Instead, a primary key can be constructed by combining the household ID and the HH member ID. There is one variable containing pure personally identifiable information, namely the name of the household member. By any ethics standard that data cannot be made public. Granular household member characteristics such as gender age education levels, marital status, and occupation can be used to reidentify households hence they are considered indirect PII that can only be made public in aggregated form. Information on data sensitivity can be added to the structural metadata of a data file.

In this example the method of measurement is interview. When collecting information about past events, such as crop yield, it can be more appropriate to use recall as the method of measurement. If actual measurements were taken such as crop cuts, this can be indicated.

Separate tables need to be provided with the code books in machine readable form.

Obviously, we can expand this data dictionary with any number of relevant metadata fields that are appropriate for the context. For the purpose of explaining OIMS we limit ourselves to these key fields. We now take a closer look at the fields in the data dictionary. For the description of the schema in JSON format, see the **Supplementary Material**: "data sheet 3.docx."

The metadata of this metadata can be readily described in terms of the self-describing metadata schema highlighted in section Self-Describing Metadata. For the description of the

```
{"Metadata Content":[
   {
    "VariableName":"Household_ID",
    "VariableDescription":"Unique identifier of the household in the survey",
    "Datatype":"Long integer",
    "Format":"12 characters",
    "Key":"2",
   },
   {
    "VariableName":"HH_memberID",
    "VariableDescription":"Unique identifier of the household member within a household",
    "Datatype":"integer",
    "Format":"8 characters",
    "Key":"0",
   },
   {
    "VariableName":"Member_name",
    "VariableDescription":"Name of the household member",
    "Datatype":"String",
    "Format":"21 characters",
    "Key":"0",
    "MethodOfMeasurement":"interview",
    "Sensitivity":"PII"
   },
   {
    "VariableName":"Respondent_identity",
    "VariableDescription":"Is this the household head?",
    "Datatype":"Controlled vocabulary",
    "Format":"9 characters",
    "ControlledVocabulary": [
       {
        "ControlledVocabularyItem":"yes",
        "ControlledVocabularyItemDescription":"yes",
       },
       {
        "ControlledVocabularyItem":"no",
        "ControlledVocabularyItemDescription":"no",
       }
      ]
    "Key":"0",
    "MethodOfMeasurement":"interview",
    "Sensitivity"="none"
```


FIGURE 5 | Portion of the JSON file containing the data dictionary in OIMS format.

schema in JSON format, see the **Supplementary Material**: "data sheet 4.docx."

Turning the Information Into Machine Readable Form

Turning the example into a machine readable JSON format requires three steps. The first step is formalizing the information in each table into JSON format. The second step is the provision of a header into the JSON file. The header is crucial as it provides the information where the metadata structure can be found. The third step is creating a parent-child relationship table that links data files in a dataset.

Basic Transformation of Metadata Into JSON Format

The JSON format as highlighted in section Metadata Schema Format is a flexible standard for data exchange across platforms and systems.

The information in **Table 1** can therefore be transformed into JSON format. In **Figure 5** we see a portion of that transformed table (see **Supplementary Material** for the full JSON file: ExampleDataDictionary.JSON).

In a similar vein, the information in the description of the data dictionary found in **Table 2** can be transformed into JSON format (see **Supplementary Material** for the full JSON file: ExampleDataDictionaryMetadata.JSON). In the example, we purposefully did not use the exact same terminology as in the self-describing schema at the

highest level of abstraction. This implies that we need one additional metadata file that links the terminology used in the data dictionary metadata to the standard OIMS terminology (see **Supplementary Material** for the full JSON file: ExampleDataDictionaryMetadata2OIMS.JSON). This has the added benefit of being able to define specifically all relevant elements such as data types that may be specific for a given dataset or metadata schema.

Obviously, the self-describing metadata schema presented in section Self-Describing Metadata is also available in JSON format (see **Supplementary Material** for the full JSON file: OIMS_vers.JSON).

Adding a Header to the JSON File

The header in any OIMS JSON file provides crucial information to place the metadata file in context. The header should contain the following information.

- Name of the metadata file
- Version of the metadata
  - Version identifier
  - Version status: is it under development, review, restricted or openly available
- Metadata schema used
  - Schema name
  - Schema type: this field can have multiple elements depending on the complexity of the metadata schema including:
    - Technical metadata
    - Descriptive metadata
    - Structural metadata
    - Entity metadata
  - Schema version
  - Schema URL
  - URI to schema documentation

Ideally the header should also contain some descriptive information on the creator, the affiliation and some contact details so interested users can get in touch.

Data Entities and Parent-Child Relationships

This paper does not deal in detail with the way the data entities are managed in terms of metadata within the context of the OIMS metadata philosophy. A separate paper on this topic is in preparation. For interoperability of datasets, it is essential, and it builds on the concepts laid out in this paper. Key element in the use of data entities in conjunction with OIMS are the parent-child relationships that exist. Datasets have one or more files containing data. These data files have associated metadata files as well as **Supplementary Material**. The data entity approach enhances data interoperability through the structuring this type of information and storing it in a way compatible with the OIMS metadata schema.

TABLE 2 | Metadata of the data dictionary.

Metadata field	Attributes	Value
VariableName	MetadataFieldName	VariableName
	MetadataFieldDescription	The name of the variable
	DataType	Simple character string
	Format	Alphanumeric
	Status	Required
	TypeClass	Primitive
	Multiple	FALSE
VariableDescription	MetadataFieldName	VariableDescription
	MetadataFieldDescription	Description of the variable: Label in STATA
	DataType	String
	Format	Alphanumeric
	Status	Required
	TypeClass	Primitive
	Multiple	FALSE
DataType	MetadataFieldName	DataType
	MetadataFieldDescription	The datatypes of the various fields of the variables
	DataType	Enumeration
	Status	Required
	TypeClass	Primitive
	Multiple	FALSE
Format	MetadataFieldName	Format
	MetadataFieldDescription	Any specific information about the format of the values in the variable
	DataType	Various
	Status	Required
	TypeClass	Primitive
	Multiple	FALSE
Key	MetadataFieldName	Key
	MetadataFieldDescription	Indicates whether a variable is a primary key, foreign key, or a regular variable. A primary key is used to ensure data in the specific column is unique. A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It uniquely identifies a record in the relational database table. Only one primary key is allowed in a table.
	DataType	Enumeration
	enumeration	0 = not a key, regular variable 1 = primary key 2 = foreign key
	Status	Required
	TypeClass	Primitive
	Multiple	FALSE
Unit of measurement	MetadataFieldName	Unit of Measurement
	MetadataFieldDescription	Unit of Measurement

(Continued)

TABLE 2 | Continued

Metadata field	Attributes	Value
	DataType	Enumeration
	Status	Required if appropriate
	TypeClass	Primitive
	Multiple	FALSE
Method of measurement	MetadataFieldName	Method of measurement
	MetadataFieldDescription	How the value of the variable was determined
	DataType	Enumeration
	Status	Required if appropriate
	TypeClass	Primitive
	Multiple	FALSE
Sensitivity	MetadataFieldName	Sensitivity
	MetadataFieldDescription	Information on the sensitivity of the information in the variable related to personally identifiable information, granular geo-spatial coordinates and or sensitive questions
	DataType	Enumeration
	Enumeration	PII Indirect PII NA = not applicable GPS = granular Geospatial information
	Status	Required if appropriate
	TypeClass	primitive
	Multiple	FALSE

DISCUSSION, CONCLUSIONS, AND NEXT STEPS

This paper presented an internally consistent approach to providing metadata for data files when standards are missing. The approach is flexible and extensible so it will not be obsolete before it is implemented at scale. The approach is based on the concept of data lakes where data is stored as is. To ensure that data lakes do not become swamps, metadata is indispensable (Ravat and Zhao, 2019). The OIMS metadata schema approach can help to standardize the description of metadata and thus can be considered the fishing gear to extract data from the data lake. Past approaches have been comprehensive but cumbersome. That could be the reason that for instance DDI is limited to some large-scale data projects.

Currently researchers can collect data which is not compatible (e.g., because questions were phrased differently, or amounts were measured using different methods). A flexible metadata schema like OIMS does not disallow this in contrast to efforts at data standardization. This begs the question whether the development of an incredibly flexible meta-data schema simply facilitate the collection of disparate data sources. Interoperability in general is much better served by data standardization. Many high variety data sets already exist and hence the highly flexible approach of OIMS serves to tag those data sets with metadata in a standardized way. The agile nature of the approach will support the uptake of OIMS.

The example in the paper illustrates the potential of the use of OIMS for making datasets interoperable and hence reusable. We are in the process of tagging several datasets with rich structural metadata and placing that in the OIMS metadata schema, this will be reported on in due time. The proof of the pudding is in the eating. Over the next years, as requirements for interoperability in relation to open and FAIR data are likely to become more stringent, the OIMS metadata schema can be a useful tool. Existing data dictionaries can be described in terms of the OIMS schema without altering the data dictionaries themselves. This implies that datasets themselves also do not need to be changed. The additional information can be provided at a fairly high level of aggregation. The next steps include demonstrating how this schema can be used to link multiple datasets, covering different topics, to use the analogy of a data lake, demonstrate how the schema can be used to fish data from the lake.

In the paper the importance of ontologies as formalized knowledge and relationships within knowledge domains was mentioned. For socio-economic household data a socio-economic ontology is under development, commonly known as SEOnt (Arnaud et al., 2020; Kim et al., under review), that initially links to a set of standardized survey questions, commonly known as 100Q (van Wijk et al., 2019), that builds on the RHOMIS approach (Hammond et al., 2017). SEOnt is a socio-economic ontology of controlled vocabularies, classifications, and concordances that allow standardization of key indicators, including gender-related indicators. The ontology has been developed by CGIAR researchers and collaborators as part of the activities undertaken in the CGIAR Platform for Big data in Agriculture.

In a setting where the data are standardized, there seems less urgency for flexibility in the metadata schema. However, evolving insights on data and its uses, can and should lead to the tagging of existing datasets even if they are based on a standardized format with additional metadata. Hence the flexibility in the metadata schema is useful for highly standardized datasets as well.

For data interoperability in general one can argue that standardization of the data is the most straightforward way of creating interoperability. However, in some domains, such as the social sciences, including economics, standardization is not a realistic option given the high variety of research questions and related data needs. If the data is high variety than the next best way of standardization for enhancing data interoperability is by standardizing metadata schemas. In summary this implies that we strive for standardization where possible and flexibility where necessary.

As part of the on-going work of the community of practice on Socio-economic data of the CGIAR Platform for Big Data in Agriculture, implementation of the OIMS metadata schema approach on datasets that can create indicators highlighted in the 100Q approach with linkages to SEOnt is envisaged. This will provide datasets with enhanced interoperability.

With more and other datasets also using the OIMS approach in the near future, it will become possible to turn what is

currently a socio-economic data swamp into a data lake that can provide timely actionable information to support the agri-food systems transformation and support efforts to assist smallholders to generate a living income while staying within planetary boundaries.

Implementing OIMS in practice requires data managers and scientists that collect the data to actively engage in providing the relevant metadata. As mentioned before, some of the metadata can be gleaned from the software solutions the scientists use already. As these are structured metadata, they can be extracted by machines. Often it does require curation by the scientist involved, especially when the software solution does not provide key information that the scientist has at hand but is not documented in a machine-readable way already.

The development of graphic user interfaces (GUIs) and tools to convert existing data dictionaries into OIMS compatible JSON format will enhance the user friendliness of the schema. We will report on the development of these tools separately.

Making data interoperable and accessible offers scope for data reuse. However, this comes with a caveat. Not all data can be reused for all purposes. Messy socio-economic datasets can come with numerous biases, including sampling bias and recall-bias to name a few. Ideally information on these issues should be included in the metadata of the dataset.

Developing standards for reporting such important issues can be helpful and the information can be added to any OIMS compatible metadata schema as the relevant fields can be described flexibly. The standardization of the way metadata is documented is the key to interoperability. It allows for reuse of efforts such as reuse of mappings between different representations and ontologies.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

GK was responsible for all aspects of the manuscript.

FUNDING

The development of the structural metadata schema OIMS was supported by the CGIAR Platform for Big Data in Agriculture through the Community of Practice on Socio-Economic Data (CoP SED). CoP SED was financially supported by the CGIAR Platform for Big Data in Agriculture that is mainly supported by the CGIAR Trust Fund (https://www.cgiar.org/funders/).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fsufs.2022.767863/full#supplementary-material

REFERENCES

Amin, A., Barkow, I., Kramer, S., Schiller, D., and Williams, J. (2012) *Representing and Utilizing DDI in Relational Databases.* RatSWD Working Paper No. 191. doi: 10.2139/ssrn.2008184

Arnaud, E., Laporte, M.-A., Kim, S., Aubert, C., Leonelli, S., Miro, B., et al. (2020). The ontologies community of practice: a CGIAR initiative for big data in agrifood systems. *Patterns* 1:100105. doi: 10.1016/j.patter.2020.100105

Berners-Lee, T., Fielding, R., and Masinter, L. (2005). *Uniform Resource Identifiers (URI): Generic Syntax.* Available online at: https://www.hjp.at/doc/rfc/rfc3986.html

Canham, S., and Ohmann, C. (2016). A metadata schema for data objects in clinical research. *Trials* 17, 557. doi: 10.1186/s13063-016-1686-5

Cantara, L. (2005). METS: the metadata encoding and transmission standard. *Cat. Classif. Q.* 40, 237–253. doi: 10.1300/J104v40n03_11

Cap Gemini and Pivotal (2013). *The Technology of the Business Data Lake*. gopivotal.com and capgemini.com. Available online at: www.gopivotal.com/businessdatalake

Cundiff, M. V. (2004). An introduction to the metadata encoding and transmission standard (METS). *Libr. Hi Tech.* 22, 52–64. doi: 10.1108/07378830410524495

Dan Brickley, and Guha, R. V. (2014). *RDF Schema 1.1: W3C Recommendation 25 February 2014*. Available online at: https://www.w3.org/TR/rdf-schema/ (accessed December 21, 2021).

Devare, M. (2017). *CG Core Metadata Schema and Application Profile - Beta Version 1.0*. Montpellier.

Esteva, M., Walls, R. L., Magill, A. B., Xu, W., Huang, R., Carson, J., et al. (2019). Identifier services: modeling and implementing distributed data management in cyberinfrastructure. *Data Inf. Manag.* 3, 26–39. doi: 10.2478/dim-2019-0002

Hammond, J., Fraval, S., van Etten, J., Suchini, J. G., Mercado, L., Pagella, T., et al. (2017). The rural household multi-indicator survey (RHoMIS) for rapid characterisation of households to inform climate smart agriculture interventions: description and applications in East Africa and Central America. *Agric. Syst.* 151, 225–233. doi: 10.1016/j.agsy.2016.05.003

Inmon, B. (1992). *Building the DataWarehouse.* Wellesley, MA: QED Information services, John Wiley & Sons.

Kim, S., Miro, B., Song, X., Arnaud, E., Laporte, M. A., Van Wijk, M., et al. (under review) Socio-Economic ONTology (SEOnt): Agile tool to label farm household survey data in the CGIAR data lake. Front. Sustain. Food Syst. Livelihoods Food Sec.

Knowledgent (2014). *How to Design a Successful Data Lake*. Knowledgent.com. Available online at: www.knowledgent.com.

Labropoulou, P., Gkirtzou, K., Gavriilidou, M., Deligiannis, M., Galanis, D., Piperidis, S., et al. (2020). "Making metadata fit for next generation language technology platforms: the metadata schema of the european language grid," in *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)* Available online at: https://schema.datacite.org/meta/kernel-4.1/doc/

Lane, P. (2005). *Oracle Database Data Warehousing Guide, 10g Release 2 (10.2)*. Available online at: http://docs.oracle.com/cd/B19306_01/server.102/b14223/title.htm (accessed November 17, 2015).

O'Brien, J. (2012). *The Definitive Guide to the Data Lake.* White Paper. Unisphere Radiant Advisors. Available online at: https://www.dbta.com/DBTA-Downloads/WhitePapers/The-Definitive-Guide-to-the-Data-Lake-5130.aspx

Oracle (2002). *Oracle9i Data Warehousing Guide*. Available online at: https://docs.oracle.com/cd/B10500_01/server.920/a96520/toc.htm (accessed November 17, 2015).

PWC (2015). *Technology Forecast Landing Page: Rethinking Integration: Data Lakes and the Promise of Unsiloed Data*. Available online at: http://www.pwc.com/us/en/technology-forecast/2014/cloud-computing/features/data-lakes.html (accessed November 17, 2015).

Rasmussen, K. B. (2014). Social science metadata and the foundations of the DDI. *IASSIST Q.* 37, 28. doi: 10.29173/iq499

Rasmussen, K. B. (2018). Metadata is key-the most important data after data. *IASSIST Q.* 42, 1–2. doi: 10.29173/iq933

Ravat, F., and Zhao, Y. (2019). "Metadata management for data lakes BT - new trends in databases and information systems," in eds T. Welzer, J. Eder, V. Podgorelec, R. Wrembel, M. Ivanović, J. Gamper, et al. (Cham: Springer International Publishing), 37–44.

Rittman, M. (2008). *Introduction to Master Data Management*. Available online at: www.rittmanmead.com.

Russom, P. (2013). *Integrating Hadoop into Business Intelligence and Data Warehousing*. TDWI Best Practices report. Renton WA, The Data Warehousing Institute.

Shukair, G., Loutas, N., Peristeras, V., and Sklar,ß, S. (2013). Towards semantically interoperable metadata repositories: the asset description metadata schema. *Comput. Ind.* 64, 10–18. doi: 10.1016/j.compind.2012.09.003

van Wijk, M., Alvarez, C., Anupama, G., Arnaud, E., Azzarri, C., Burra, D., et al. (2019). *Towards a Core Approach for Cross-Sectional Farm Household Survey Data Collection: A Tiered Setup for Quantifying Key Farm and Livelihood Indicators*. Texcoco.

Vardigan, M. (2014). The DDI matures: 1997 to the present. *IASSIST Q.* 37, 45. doi: 10.29173/iq501

Vardigan, M., Donakowski, D., Heus, P., Ionescu, S., and Rotondo, J. (2015). Creating Rich, Structured metadata: lessons learned in the metadata portal project. *IASSIST Q.* 38, 15. doi: 10.29173/iq123

Wilkinson, M. D., Dumontier, M., Aalbersberg, I., J., Appleton, G., Axton, M., et al. (2016). The FAIR guiding principles for scientific data management and stewardship. *Sci. Data* 3, 160018. doi: 10.1038/sdata.2016.18

Conflict of Interest: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.