ChinaXiv合作期刊

DATA PAPER

DAMS: A Distributed Analytics Metadata Schema

Sascha Welten[1†], Laurenz Neumann[1], Yeliz Ucer Yediel[2], Luiz Olavo Bonino da Silva Santos[3,4], Stefan Decker[1,2] & Oya Beyan[2,5]

[1]Chair Informatik 5, RWTH Aachen University, 52056 Aachen, Germany

[2]Fraunhofer Institute for Applied Information Techniques (FIT), 53757 Sankt Augustin, Germany

[3]Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500AE Enschede, The Netherlands

[4]Department of Human Genetics, Leiden University Medical Centre, Leiden 2333 ZA, The Netherlands

[5]Institute of Medical Information, Faculty of Medicine & University Hospital Cologne, University of Cologne, 50674 Cologne, Germany

chinaXiv:202211.00211v1

Keywords: Distributed analytics; Federated analytics; Personal Health Train; Metadata schema; RDF(S); SHACL; FAIR

Citation: Welten, S., et al.: DAMS: A distributed analytics metadata schema. Data Intelligence 3(4), 528-547 (2021). doi: 10.1162/dint_a_00100

Received: January 19, 2021; Revised: April 11, 2021; Accepted: May 1, 2021

ABSTRACT

In recent years, implementations enabling Distributed Analytics (DA) have gained considerable attention due to their ability to perform complex analysis tasks on decentralised data by bringing the analysis to the data. These concepts propose privacy-enhancing alternatives to data centralisation approaches, which have restricted applicability in case of sensitive data due to ethical, legal or social aspects. Nevertheless, the immanent problem of DA-enabling architectures is the black-box-alike behaviour of the highly distributed components originating from the lack of semantically enriched descriptions, particularly the absence of basic metadata for data sets or analysis tasks. To approach the mentioned problems, we propose a metadata schema for DA infrastructures, which provides a vocabulary to enrich the involved entities with descriptive semantics. We initially perform a requirement analysis with domain experts to reveal necessary metadata items, which represents the foundation of our schema. Afterwards, we transform the obtained domain expert knowledge into user stories and derive the most significant semantic content. In the final step, we enable machine-readability via RDF(S) and SHACL serialisations. We deploy our schema in a proof-of-concept monitoring dashboard to validate its contribution to the transparency of DA architectures. Additionally, we evaluate the schema's compliance with the FAIR principles. The evaluation shows that the schema succeeds in increasing transparency while being compliant with most of the FAIR principles. Because a common metadata model is critical for enhancing the compatibility between multiple DA infrastructures, our work

† Corresponding author: Sascha Welten (Email: welten@dbis.rwth-aachen.de; ORCID: 0000-0001-5570-9672).

ChinaXiv合作期刊

lowers data access and analysis barriers. It represents an initial and infrastructure-independent foundation for the FAIRification of DA and the underlying scientific data management.

1. INTRODUCTION

In recent years, Big Data analysis has become a highly optimistic research area producing promising results. Especially in the medical domain, medical data analysis has a huge potential to directly enhance the well-being of humans by improving the quality and efficiency of healthcare [1, 2]. However, the reuse of patient data for medical research is often limited to data sets available at a single medical centre where the medical scientist has authorised access. The most immanent reason why medical data is not heavily inter-institutionally shared for research relies on ethical, legal, and privacy aspects and rules [3]. Such rules are typically guarded by national and international laws, such as the General Data Protection Regulation (GDPR) in the European Union or the Data Protection Act (DPA) in the UK limiting sharing sensitive data [4, 5, 6]. An approach to address these problems is Distributed Analytics (DA). In DA, the algorithm is sent to the data—instead of *vice versa*. The analysis algorithm is then executed at the data premises and only the results, e.g., model parameters or aggregated statistics are communicated back to the requester. This ensures that the data stays within the institutional borders and therefore preserves the data sovereignty of the data owner. An approach following the concept of DA for healthcare is the Personal Health Train (PHT) [3, 7, 8]. It is a platform for the distribution of analytical tasks to data providers. In PHT terms, the analysis algorithms are represented by so-called Trains and so-called Stations represent the environments where algorithms can interact with data. It has already demonstrated several of its capabilities in several studies [7, 8, 9, 10, 11].

Nevertheless, current DA architectures, such as the PHT, still have significant shortcomings that need to be addressed to improve usability and interoperability [3]. Due to its highly distributed design and various involved components, the system operates as a black-box from a user's perspective, which hinders efficiency and transparency. Further, each PHT architecture is independently designed. Therefore, a common consent concerning the interoperability of different PHT implementations is not apparent yet.

1.1 Objective

In this work, we propose a newly created metadata schema specification for DA platforms. Simultaneously, we aim at improving the transparency of the components and activities, which should contribute to a trust-based collaboration between users and data providers. We use the PHT as a reference architecture to build a schema from scratch. However, our schema should be generic to comply with other DA implementations. We base our work on initially acquired domain expert knowledge. Further, we aim to use well-established ontology engineering methodologies to transform the unstructured information into a well-arranged and uniform format. Finally, the schema should be present in a machine-readable representation. Overall, the metadata should be able to convey information about the different components to the users and enable a better understanding of the performed analysis processes. We want to reach this objective by making the DA infrastructure and the metadata compliant with the so-called FAIR principles [3]. During this process,

chinaXiv:202211.00211v1

ChinaXiv合作期刊

we make every digital asset more Findable and Accessible. We further want to achieve cross-architectural Interoperability amongst different DA implementations, which facilitates access to additional data and the Reusability of the components.

1.2 Findings & Contributions

The contribution of this work is a requirement analysis of information, which is important for users of a DA infrastructure. The foundation for the requirement analysis is a transcript of conducted interviews with domain experts and users as interviewees. Based on this foundation, we create user stories to receive the requirements in a structured form. In the next step, we develop a metadata schema describing the two main PHT components, i.e., the Train and the Station. To enable machine-readability, the schema is modelled using the Resource Description Framework (RDF) and the Resource Description Framework Schema (RDFS). As a file format, we use the Terse RDF Triple Language (Turtle). Additionally, we provide schema constraints using the Shapes Constraint Language (SHACL). We analyse the schema according to the FAIR principles to emphasise its *FAIRness*. Finally, we validate the applicability of this schema with respect to a proof-of-concept implementation.

We find that a metadata schema is a suitable method to define requirements for a minimal set of metadata information about participating entities. This minimal set of information adapted by the stakeholders increases the level of transparency that is required for a proper function of the infrastructure. In addition, the developed schema is an enabler for cross-architectural interoperability, which contributes to the *FAIRification* of DA architectures.

The remainder of this work is structured as follows. Related work is presented in Section 2. Section 3 and Section 4 deal with the development and the content of the schema, respectively. In Section 5, we validate the schema in a proof-of-concept approach before we evaluate our schema with respect to the FAIR principles. Finally, Section 6 concludes this work and gives an outlook on future work.

2. RELATED WORK

In this section, we will present related work in the DA domain and discuss the FAIR principles as evaluation qualities for our proposed work.

2.1 Distributed Analytics

Several approaches and infrastructures have been developed for DA to perform statistical analyses or model training in terms of decentralised data [12, 13, 14, 15, 16, 17]. These approaches follow two basic paradigms of bringing the algorithm to the data. The first paradigm is the parallel execution of the analysis task, which is often referred to as Federated Learning (FL) in the literature [14, 16]. Furthermore, there is a successive execution of the analysis tasks, also known as Institutional Incremental Learning (IIL) or Weight Transfer (WT) [12, 15]. An overview of the two paradigms is given in Figure 1. Since the communication

chinaXiv:202211.00211v1

ChinaXiv合作期刊

chinaXiv:202211.00211v1

only consists of the results of the analytical task, data never leaves its origin and stays within the institutional borders. Due to this circumstance, DA architectures propose a solution for preserving data privacy and comply with the above-mentioned existing regulations.

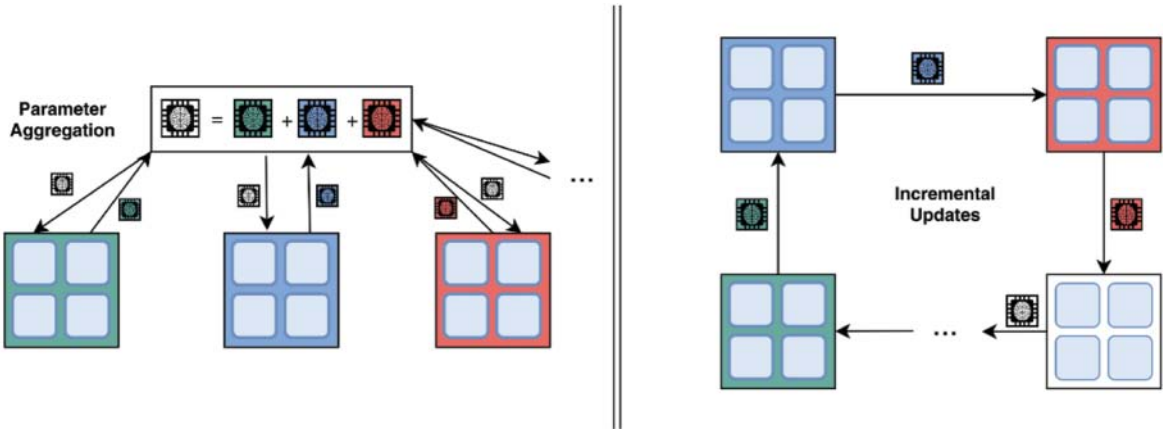

Figure 1. Federated Learning (FL) (left) and Institutional Incremental Learning (IIL) / Weight Transfer (WT) (right). In a Distributed Analytics setting, we usually consider two policies for performing an analytical task. The parallel approach, FL, distributes replicas of the analysis task to the data providers. The data provider executes the task, e.g., model training, and returns the results to the central component, which aggregates the results. If necessary, the distribution process is repeated—based on the aggregated results. An alternative is the sequential execution of the task following an incremental result update policy. In this scenario, the analytical task is step-wisely circulated in a network of data providers. If the analytical task periodically re-visits the data providers, it is denoted as cyclic IIL/WT.

One platform for DA following the IIL/WT policy is the so-called Personal Health Train (PHT), which we consider to be the reference architecture of our work [3, 7, 8]. The PHT has been fostered by the GO FAIR initiative[①] (Section 2.2) and is part of the established GO FAIR implementation networks with the intention to facilitate data-driven medicine based on decentralised data. The PHT originates from an analogy from the real world. The infrastructure reminds us of a railway system including trains and stations. The train uses the network to visit different stations to transport, e.g., several goods. Adapting this concept to the PHT ecosystem, we can draw the following similarities. The train encapsulates an analytical task, which represents the goods in the analogy. The analytical task is developed for example by a researcher, who wants to conduct a statistical study based on data located in decentralised data repositories. Each data provider takes over the role of a reachable *Station*, which can be accessed by the *Train*. Further, the Station executes the task, which processes the available data.

In general, the PHT processes the analytical task incrementally by visiting each station one by one and stores the intermediate results. Finally, after visiting every selected Station, the Train is sent back to the researcher, who inspects the calculated results. However, parallel executions are also possible by integrating a central aggregation component. Therefore, our work considers policies for both parallel and incremental

① For more information visit: https://www.go-fair.org/implementation-networks/overview/personal-health-train/

approaches. Note that we denote *Station* and data provider, as well as, *Train* and analytical task interchangeably. Other PHT architectures, which have been developed by the scientific community, have been applied to several use cases [3, 7, 8, 9, 10, 11].

Deist et al. [9] have conducted DA experiments based on decentralised lung cancer patient data. They have been able to perform several analyses and reveal insights from the data without data centralisation. Further, Shi et al. [8] have applied the PHT approach to facilitate the provision of radiomics data for DA. They emphasise the need for such a distributed infrastructure to give access to more data.

2.2 FAIR Guideline Principles

The FAIR principles are introduced by Wilkinson et al. as a guideline to improve the **F**indability, **A**ccessibility, **I**nteroperability, and **R**eusability of digital assets [18, 19, 20]. These denote desired qualities of metadata about scientific data or the scientific data themselves. Therefore, the FAIR principles are suitable evaluation criteria for our metadata schema presented in this work. Each of the four mentioned criteria is sub-divided into more precise requirements the metadata should meet:

- **Findable.** The data should be findable not only by humans but also by computers. Each asset should have a globally unique and persistent identifier (F1). Further, the data should be described with rich metadata (F2) and this metadata should clearly and explicitly include the identifier of the data they describe (F3). Finally, data should be registered in a searchable resource to be findable for search operations (F4).
- **Accessible.** The metadata should be retrievable by using a standardised communication protocol (A1), which is open, free and universally implementable. If necessary, this protocol allows for an authentication and authorisation procedure. The metadata should also be accessible although the data is no longer available.
- **Interoperable.** The metadata should use a formal, accessible, shared, and broadly applicable language for knowledge representation (I1). Further, the metadata should use vocabularies that also follow the FAIR principles (I2) and should include qualified references to other metadata (I3).
- **Reusable.** The metadata should be richly described with a plurality of accurate and relevant attributes (R1), e.g., usage licences, provenance or domain-relevant community standards.

Recently, several works have proposed metadata schemata to facilitate FAIR data management in different domains [21, 22, 23]. Specka et al. [21] proposed BonaRes, which is a meta-data schema for geospatial soil-agricultural research data. Labropoulou et al. [22] presented ELG-SHARE, which describes language resources and aims to improve the European Language Technology sector. Additionally, Franke et al. [23] proposed the metadata schema Plasma-MDS for the plasma science community to facilitate the publication of research data in that domain. Beyond these related implementations targeting the establishment of FAIR data management, other works have focused on methods or procedures to make data more FAIR [3, 19, 24]. Thompson et al. [19] gave a condensed overview of current tools and technologies in terms of FAIR data. Jacobsen et al. [24] proposed a seven-step workflow for a data FAIRification process. The workflow consists

chinaXiv:202211.00211v1

ChinaXiv合作期刊

chinaXiv:202211.00211v1

of—among others—the identification of the objective, the analysis of the available metadata, and the generation of the semantic model. Highly related is the work by Beyan et al. [3], who investigated the application of the FAIR principles to a DA architecture. Especially, they have analysed the meaning of FAIRness with respect to the components of the PHT architecture and it has been an initial foundation for the FAIRification of the DA ecosystems. We utilise the findings by Beyan et al. to develop and implement a metadata schema in order to enable an initial foundation for FAIR data management in the DA domain—analogously to the presented works mentioned above. The implementation of this metadata model will resolve several challenges. As we described earlier, the current DA architectures suffer from the non-transparency of the involved components since these architectures are highly distributed by design. Especially, such architectures are black-boxes for their users preventing an efficient analysis execution and even if informative metadata is available, it has not been standardised. The development of the metadata schema is presented in the next sections.

3. SCHEMA DEVELOPMENT

Several methodologies and guidelines have been proposed to develop ontologies or metadata schemata in a standardised workflow [25, 26]. In our work, we partially follow the process by Keet et al. and the well-established method called *Methontology* by Lopez et al. [25, 26, 27]. However, we tailor selected components of these methodologies to allow a more user-centric domain analysis. We split our schema development into a succession of four phases (Figure 2), which we will present in the following sections: Knowledge Acquisition (Section 3.1), Conceptualisation (Section 3.2), Implementation (Section 3.3), and the integration of reusable ontologies (Section 3.4). The complete schema is ultimately evaluated in Section 5. Note that we made development artefacts and supplemental materials online available[②].

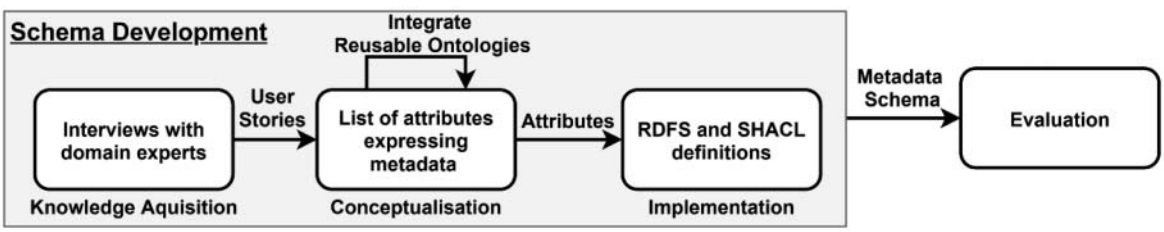

Figure 2. Schema development workflow. We first conduct requirement analyses as interviews with experts to have a valid foundation for our work. Then, we revise the results and structure the data using user stories (knowledge acquisition). The obtained information is transformed into metadata items and attributes (conceptualisation). In this phase, we additionally integrate reusable ontologies in order to enable interoperability. During the last development step, we serialise the components using RDF(S) and we define SHACL expressions for validation purposes (implementation). The resulting schema is then evaluated in the final step.

② Supplemental materials: gitlab.com/PersonalHealthTrain/implementations/germanmii/smith/phtmetadata

ChinaXiv合作期刊

3.1 Knowledge Acquisition

The first important step is to acquire domain knowledge as a foundation for our work [25]. Especially, we want to identify what kind of information is obscure to the users of a DA platform. Instead of gathering data from experts via competency questions (Keet et al. [26]), we conduct several unstructured interviews with DA domain experts and possible users to better reflect our present use case (Lopez et al. [25]). We asked the interviewees several questions about their interaction with such a system and we explicitly demanded suggestions for possible items of the metadata schema. After the interviews, we obtained interview transcripts representing our initial groundwork. At this stage, our foundation is unstructured and contains possibly redundant and irrelevant information. Therefore, we need to process this foundation into a more structured and compact format.

A suitable method for this purpose—originating from requirements analysis in the domain of software engineering—are so-called user stories due to their high capability of identifying requirements [28, 29]. A user story is a short statement expressing the interviewees' requirements of a system in a compact shape. In our case, the mentioned system represents the DA architecture and the requirements reflect the components our metadata schema should have to increase transparency and efficiency. Usually, a user story contains three types of information: a role, a story, and a benefit. The role of the user describes from which point of view the user is interacting with the architecture. The story represents the action the user wants to perform, and the benefit gives a reason why this action is performed. A generic structure of such a user story is given in [29]. We modified this proposed user story template to suit the metadata schema and DA context as follows:

As a <role of the user in the DA>, I want to have information on <characteristic of a component/ concept> so that <benefit of having that information>.

Using this template, we transform the transcript into a collection of 89 user stories (including duplicates), which represent the contained information in an easy-to-read list and natural language format. Some examples are given in the following:

- As a Data Scientist, I want to have metadata about a contact person for each data provider, so that I can contact him/her if issues arise.
- As a Data Scientist, I want to have a general description of the computing capabilities of a data provider, so that I can decide if the data provider will be able to run my analytic task.
- As a Data Scientist, I want rich information about what kind of data is provided, so that I can easily find suitable data providers for my analysis.

However, the user stories are still not semantically arranged or even machine-readable. Hence, a further processing step is needed to categorise the items into semantically consistent units. This procedure is presented in Section 3.2.

chinaXiv:202211.00211v1

ChinaXiv合作期刊
chinaXiv:202211.00211v1

3.2 Conceptualisation

In the conceptualisation phase, our goal is to organise and structure the domain information into an intermediate representation independent of the implementation language [25]. This approach also includes the identification of hierarchically structured taxonomies arranging the knowledge in semantically consistent groups [25]. Therefore, we group the user stories according to their content. This improves the ordering of the user stories, helps to discover and remove duplicates, and structures the metadata schema plausibly. Subsequently, we assign each user story to one of our two main PHT components: **Train** or **Station**. This differentiation relies on the fact that both components represent two fundamentally different concepts. The Train especially captures information about the analytical task while the Station acts as a data provider and execution environment for the Train. By introducing these two sub-schemata, we keep the information component-oriented and reduce the presence of redundant metadata attributes in each sub-schema. We further determine sub-categories, which relate to different scopes of Train or Station information. We identify three categories for the Train metadata:

1. **Train-Business Information:** This group contains metadata with a non-technical and organisational purpose, e.g., information about the author of the Train algorithm.
2. **Train-Technical Information:** This subset includes metadata that provides information about the technical aspects of the algorithm. This includes all static information needed to run the analytical task at each site. An example of such metadata is the data type the algorithm is processing.
3. **Train-Dynamic Execution Information:** This subset includes metadata instances that are created during the execution of an algorithm. An example is the log output the Train is producing.

Analogously, metadata about the Stations can also be sub-divided into three sub-groups:

1. **Station-Business Information:** The criteria for metadata in this subset are analogous to the criteria of the Train-Business Information. An example could be the location of the data provider.
2. **Station-Runtime Environment Information:** This subset includes all metadata about the computational environment a Station provides. An example would be information about the capabilities the Station has in terms of computational power.
3. **Station-Data Information:** This subset contains metadata about the data a Station provides for data analyses. An example could be the size of a data set or the data set type provided by the Station.

We create a table for each of the six above-mentioned groups, which maps each user story to one or more corresponding easy-to-read attributes. This procedure provides a first semi-formal specification of a metadata schema [25]. An extract of these tables is given in Table 1. Once we have *embedded* each user story into an intermediate attribute representation, we specify the attributes by adding a `data type` and an expected (`cardinality`). We denote the final attribute specifications as `identifier:data type` (`cardinality`). Regarding the `data type`, we distinguish between primitive (e.g., integer), enumerations (predefined values), and complex (entities or sub-schemata) data types. The (`cardinality`) is expressed as a range including a minimum and a maximum value or *n* if the maximum is unbounded (1..*n* or 0..1).

ChinaXiv合作期刊

chinaXiv:202211.00211v1

Table 1. Extract of the created attribute table.

Sub-category	User Stories	Attributes
Business (Train)	*As a general User of the PHT, I want a good versioning system, so that I can reproduce results.*	version:string (1)
Business (Station)	*As a Train User, I want to have information about the Station's geographical location, so that I know if I have to consider national regulations.*	longitude:float (1...n) latitude:float (1...n)
Technical (Train)	*As a Train User, I want to differ between the analytic task and the underlying model, so that I can properly reuse models.*	model:Model (1...n)
Runtime Environment (Station)	*As a Data Scientist, I want to have information about the computational environment, so that I can decide if the Station will be capable of running my model.*	ComputationalEnvironment .estimatedGFLOPS:float (0...n) ComputationalEnvironment .hasCUDASupport:boolean (0...n)
Dynamic Execution (Train)	*As a Station Owner, I want to know when a train will visit my Station, so that I can make preparations.*	TrainExecution.plannedRoute: ExecutionPlanStep (1...n)
Data (Station)	*As a Train User, I want to know how to access the data, so that I can ensure that my Train has suitable input interfaces.*	DataSet.accessURL:string (0...n)

Note: We have two components, i.e., Train and Station, and each is sub-divided into three sub-categories representing the type of information. Each user story is assigned to one sub-category. Finally, the semantic information contained in the user story is transformed into one or more attribute variable(s) as it is shown in the last column. In this table, we only show one user story per sub-category.

3.3 Implementation

During the implementation step, we transform the semi-formal schema specification given by Table 1 into a machine-readable representation. In our work, we use *RDF(S)* for modelling the proposed hierarchical structure. Additionally, we use *SHACL* to provide schema constraints (so-called shapes) for schema validation. We use Turtle for the serialisation of the schema. A short extract of the serialisation can be seen in Listing 1.

ChinaXiv合作期刊

chinaXiv:202211.00211v1

```
@prefix pht:<https://gitlab.com/PersonalHealthTrain/
            implementations/germanmii/smith/phtmetadata#>.
#train
pht:Train rdfs:subClassOf rdfs:Class.
pht:creator rdf:type rdfs:Property.
pht:publisher rdf:type rdfs:Property.
pht:identifier rdf:type rdfs:Property.

[...]

#station
pht:Station rdfs:subClassOf rdfs:Class.
pht:creator rdf:type rdfs:Property.
pht:responsibleForStation rdf:type rdfs:Property.
pht:certificate rdf:type rdfs:Property.

[...]
```

Listing 1. Extract of the schema serialisation in Turtle. We use RDF(S) to model Train and Station classes and the corresponding metadata attributes as properties of these classes.

To further enhance the interoperability of this first schema version, we determine attributes, which can be reused from other already implemented ontologies or vocabularies in Section 3.4.

3.4 Integration of Reused Ontologies

During the attribute definition process, it is useful to detect conceptual intersections between our attribute set and existing metadata schemata and ontologies [25, 27]. The reuse of already proposed concepts has multiple advantages. For example, the identifiers of the attributes in these schemata are often well-established and therefore widely understood. Furthermore, we increase the interoperability of our work since the reused attributes are consistent across different schemata. In the following, we briefly introduce the reused concepts and their scope of application in our metadata schema. An overview is given in Figure 3.

DataCite[3] is considered for metadata belonging to the group of business information of both Train and Station. Therefore, we are able to register our digital assets with *DataCite*, which assigns digital object identifiers (DOIs) to these assets and ensures that sufficient information is available for each component. We integrate the *friend of a friend* (FOAF[4]) ontology to express business information about social entities like the Train or the Station owner. The *Data Mining OPtimization* (DMOP) ontology and the *Software Ontology* (SWO[5]) are suitable for describing the technical information of the Train and the data information of the data provider [26]. Additionally, the *Data Catalog Vocabulary* (DCAT) is also considered. It provides pre-defined attributes describing the semantics of data sets [31].

[3] DataCite: https://schema.datacite.org
[4] FOAF: http://www.foaf-project.org
[5] SWO: http://theswo.sourceforge.net

ChinaXiv合作期刊

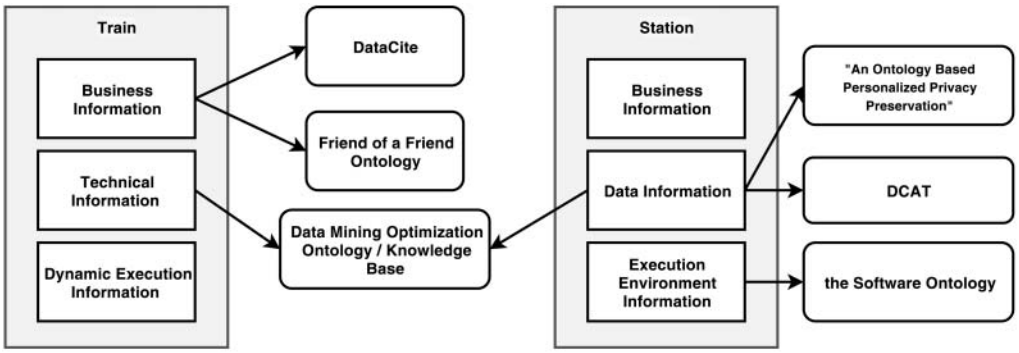

Figure 3. The reused ontologies/metadata schemata in relation to the proposed groups of metadata. DataCite, denoting general information, and FOAF, describing information about social entities, are used for the business information. The SWO is used to describe the license for a model. DMOP describes technical information about the analytical task and data sets. The ontology presented by Can et al. [30] is used to denote the grade of anonymization of a data set.

4. INTEGRATION OF REUSED ONTOLOGIES

After we have shown how we developed the proposed metadata schema, we will study several core metadata items of our architectural key components, i.e., Train and Station, in more detail. An overview is given in Figure 4.

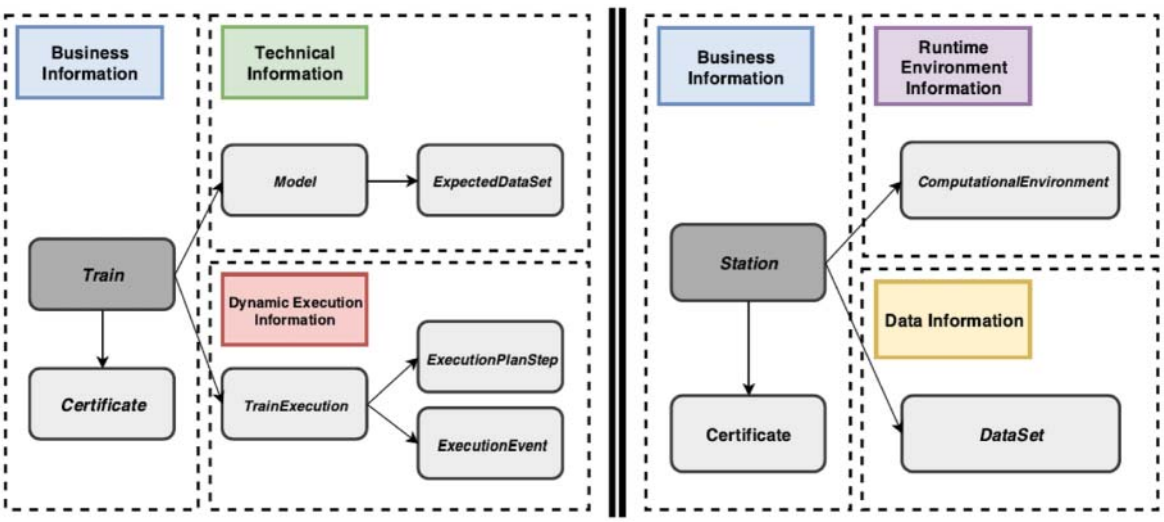

Figure 4. A top-level view of the metadata schema for the key components Train (left) and Station (right). Each component consists of three sub-categories. For the Train component, we implemented a general `Train` class to capture all Train related attributes. This hierarchical structure is continued on the sub-category level. For the technical information, we have a central `Model` class. The `TrainExecution` attribute aggregates information about the Train execution. Analogously, the `Station` attribute covers information about the Station. The sub-attributes are `ComputationalEnvironment` for the runtime environment information category and `DataSet` for the data information category.

chinaXiv:202211.00211v1

ChinaXiv合作期刊

4.1 Train Metadata

As we discussed above, we split the metadata about the Trains into three sub-categories covering different content. This partitioning can also be found in the intrinsic structure of our schema. We put the Train's metadata attributes in a hierarchical relation. Therefore, we create a central attribute `Train` representing the top-level attribute of the hierarchy (Figure 4). One advantage is that sub-attributes can easily be detached and reused without the necessity of applying the top-level attributes.

Directly related to `Train` are the business information attributes. For example, information about the author and certificate specifications are part of this group. The `creator` attribute captures the author information by using the `Agent` class of the FOAF ontology. Information about the certificate is stored in a separated `certificate` attribute. Additionally, a detailed description of the Train is part of the business information. Therefore, we add a `description` attribute with a string data type. These mentioned attributes already enable a basic level of transparency of our DA architecture since the Station admins can gain insights into the provenance of the analytical tasks.

While the business information gives more high-level and non-technical information of the Train, the technical information covers a more detailed description of the data analysis. This information is captured in a sub-attribute called `Train.Model`. The `Train.Model` sub-schema includes characteristics of the analytical task the Train is performing. Information about the data set the algorithm is expecting is stored in the `ExpectedDataSet` sub-schema, which is further sub-divided into `ExpectedTableDataSet` or `ExpectedFileDataSet` to differentiate between different types of data sources. Furthermore, our schema allows for a more detailed characterisation of the used algorithm. For this purpose, we reuse the class `AlgorithmCharacteristic` of the DMOP ontology in our `Model` sub-schema. An exemplary attribute is the `ToleranceToNoise` characteristic, for example, indicating the noise tolerance of a model. Since models can differ in their design and objective, we further introduce a `usedAlgorithm` variable specifying the algorithm type the model is using. This variable is also reused from the DMOP ontology.

Lastly, the dynamic execution information group describes the Train execution aggregated by `TrainExecution`. Associated are the sub-schemata `ExecutionPlanStep` and `ExecutionEvent`. The former contains all information about a step in the planned route of a Train and the latter gives information about execution events, e.g., the CPU consumption during runtime or a report about the Train's status (*running* or *idle*).

4.2 Station Metadata

Analogously, we follow the same hierarchical principle for the Station metadata, i.e., we introduce a `Station` attribute as a top-level schema. Instead of an attribute `creator`, Station has a mandatory attribute describing the `owner` of it. However, the data type is the same as in the `Train` schema (the `Agent` class of the FOAF ontology). In addition to the `owner`, an attribute `responsibleForStation` is introduced. This attribute represents a contact person for support requests for example. The reference to a Station certificate is equivalent to the Train `certificate` attribute (Figure 4).

chinaXiv:202211.00211v1

ChinaXiv合作期刊

Further, to ease interoperability between the Train and Station metadata, we decided to keep the data format descriptions identical. Therefore, we reused the above-introduced data description attributes and analogously added two attributes to the data provider schema called `TabularDataSet` and `FileDataSet`. This has the benefit that the Train and Station data interfaces are homogeneous and it is ensured that the Train can access the correct data source. However, we have designed the metadata about the data sets domain-agnostic. Each data set provided by the Station is equipped with a (global) persistent identifier (PID). This PID additionally references to the mentioned domain-specific metadata, which can be resolved via this introduced PID. Regarding the data protection aspects of the data provision, we added security-related attributes, e.g., `usedDifferentialPrivacy`, which states the level of data anonymisation. The third sub-schema is called `ComputationalEnvironment`. It captures detailed information about the technical capabilities of the Station. Possible characteristics of a Station could be the support of Compute Unified Device Architecture (CUDA®) for deep learning tasks (`hasCUDASupport`) or the maximum number of tasks the Station can simultaneously execute (`maximumNumberOfModelsSupported`).

In this section, we have shown basic attributes of our metadata schema to enhance the transparency of a DA architecture for its involved parties—the Train requester and the Station admins. Based on this transparency, these parties gain confidence in the architecture and each other. In the next section, we present a proof-of-concept implementation of our work and evaluate the *FAIRness* of our schema.

5. EVALUATION

In this section, we apply our metadata schema to a DA platform in a proof-of-concept approach and we evaluate the schema with the above-mentioned FAIR principles (Section 2).

5.1 Usage of DAMS: A Proof-of-Concept

We present an exemplary real-world application of our schema as a proof-of-concept approach. For this, we explore the possibilities it provides to enhance the usability and transparency of the already presented PHT architecture.

One application of the metadata schema is the provision of information via a *dashboard*, which visualises the information in a pre-processed, comprehensible, and graphical form. Such a dashboard is useful for various users of the PHT infrastructure. For example, the researcher (Train sender) can trace the Train's status at each Station and estimate the remaining execution time. Furthermore, information such as the current CPU consumption can be used to evaluate the efficiency of the Train, which enables possible optimisations for further executions. In addition, the dynamic collection of the execution time histories can be prospectively used to match the Stations' computing capacities with the real Trains' computing requirements to prevent Station overload. Additionally, the Station admin also benefits from a dashboard since our metadata schema

® CUDA: https://developer.nvidia.com/cuda-zone

chinaXiv:202211.00211v1

ChinaXiv合作期刊

captures logs emitted during the Train execution. The Station admin can detect potential malicious activities or unintended behaviour. It also facilitates troubleshooting, for example. For the dashboard development, we utilise the tool Grafana[7], which is an open-source visual analytics platform developed by *Grafana Labs*. The advantages of this platform are the following. It allows the user to customise dashboards by selecting suitable visualisation panels. Those panels support various data source interfaces, from which data can be queried. The connection to the data is established by a *plug&play* mechanism using source-depended plugins. The configuration is performed through the front-end and requires minor technical knowledge. To connect our panels with the metadata source, we have developed a custom plugin for this purpose. It consumes the metadata (i.e., the serialisation of the RDF(S) model) in JavaScript Object Notation for Linked Data (JSON-LD) format from Hypertext Transfer Protocol (HTTP) endpoints provided by the components of the PHT architecture. A possible layout of such a dashboard can be seen in Figure 5.

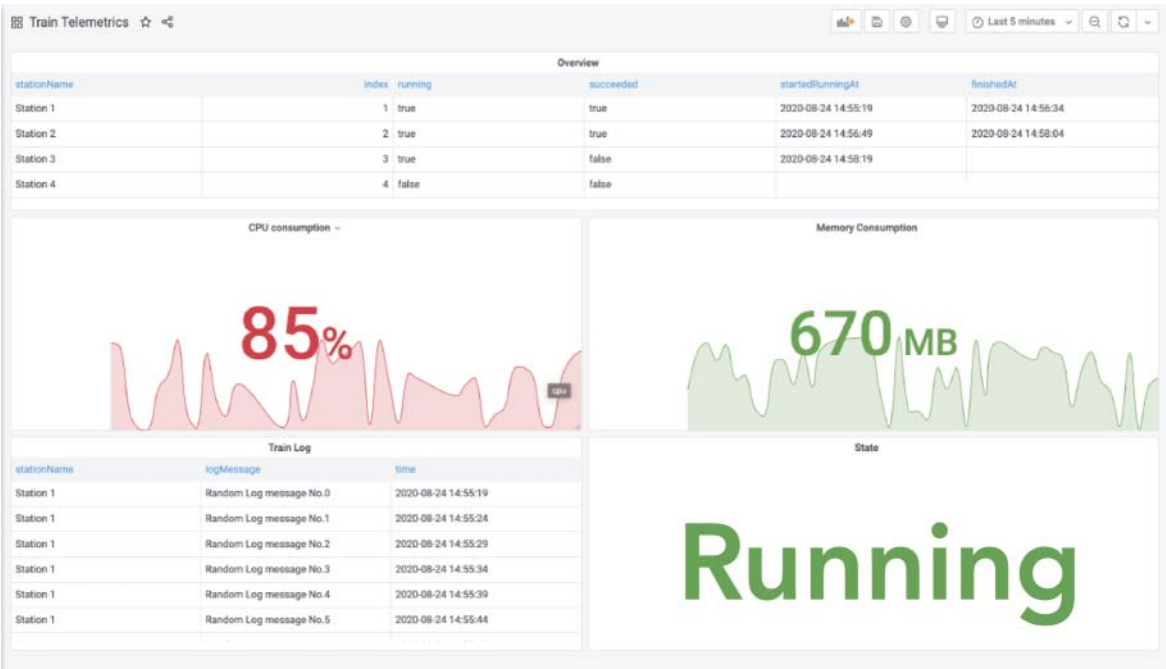

Figure 5. Screenshot of the metadata dashboard. The dashboard monitors dynamic metrics about a Train. In the upper part of the dashboard, an overview of the Train route is presented. The table indicates whether the Train has been executed (running is set to true) and terminated successfully. The two central panels depict live information about the CPU consumption and the memory consumption of the Train. The bottom left panel displays log messages the Train emits and the bottom right panel shows the current status of the Train. Possible states are transmitted, waiting or running.

[7] Grafana: https://grafana.com/

chinaXiv:202211.00211v1

ChinaXiv合作期刊

In this exemplary dashboard configuration, the CPU and memory usage of a Train at each Station is centrally displayed as a hybrid visualisation containing both the current usage and the usage over time as a graph. The dashboard also allows tracing the current location of the train using a table, which displays for each Station whether the Train was successfully executed. If the execution of the Train fails, the user is immediately notified. The remaining visualisations show the state of the Train or the emitted log messages as a live feed. Especially, the provision of this telemetric metadata enables the different users to monitor the execution and the current status of Trains. As stated above, the proposed visualisation of the metadata is a proof-of-concept approach showing that the metadata schema contributes to the transparency and usability of a DA architecture. In our presented scenario, random metadata has been generated by an emulator on a single machine. This emulator makes all necessary endpoints available and provides the generated metadata, which is queried by Grafana. We finally analyse our approach according to the above-mentioned FAIR principles in the next section.

chinaXiv:202211.00211v1

5.2 Usage of DAMS: A Proof-of-Concept

In this section, we briefly evaluate the application of a metadata schema in a DA architecture according to the FAIR guidelines (Section 2). For a complementary evaluation, we refer to the preliminary work of Beyan et al. [3].

1. **Findable.** By design, we identify objects in our metadata schema by using Uniform Resource Identifiers (URIs) (F1). For linking metadata to the corresponding digital asset of the PHT ecosystem it describes, we included attributes referencing to identifiers of those assets in the schema (F3). Further, the metadata schema includes descriptive information enhancing the findability of the digital assets (F2). Currently, the metadata items are not indexed or registered in a central and discoverable service (F4), which is part of future work.
2. **Accessible.** In our proof-of-concept, we present the accessibility of the metadata by using a dashboard, which utilizes standard communication protocols for retrieving it (A1). Along with human clients, the metadata schema can also be accessed by computation clients in terms of automation. Further, our schema incorporates versioning attributes such that the metadata is accessible even when the assets disappear (A2).
3. **Interoperable.** We have modelled the schema using RDF(S) and SHACL for validation. We have ensured the utilisation of existing and widely-established vocabularies (Section 3.4). RDF and SHACL are both standards of the W3C and used in many broadly-used applications (I1). Currently, the FAIR evaluation of the reused vocabularies is still necessary (I2) and cross-referencing is only partially included (I3).
4. **Reusable.** We based the schema content on the feedback of the domain experts to ensure the incorporation of relevant attributes attached to the data. The schema includes attributes related to licensing (R1.1) and provenance (R1.2). Additionally, we partially meet domain-relevant community standards by reusing existing ontologies or vocabularies (R1.3). The attribute names are chosen to be self-explanatory.

Overall, it can be stated that the proposed metadata schema mostly fulfils the FAIR principles or is an enabler for FAIR data management. However, we recognise room for improvements, which will be discussed in the next section as part of future work.

ChinaXiv合作期刊

chinaXiv:202211.00211v1

6. CONCLUSIONS AND FUTURE WORK

The goal of this work has been to create a foundational metadata schema to propose a solution for the present limitations of DA architectures. Current problems are the insufficient interoperability across different implementations and the black-box-alike characteristics originating from the lack of adequately described assets. These concerns have stimulated the need for semantically enriched components to enhance transparency and accessibility of information about different architectural DA components. We chose a sample PHT infrastructure as an appropriate reference DA architecture to develop our metadata schema without losing compatibility with other DA implementations. To achieve our objective, we followed a three-phase approach inspired by the well-established methodologies proposed by Keet et al. and Lopez et al. for ontology engineering [25, 26, 27]. In the first phase, we acquired domain knowledge by conducting oral expert interviews. The result of this phase has been a collection of user stories representing the knowledge in a structured but unarranged format. According to the mentioned methods, we classified the user stories based on their content in the second phase to obtain a hierarchically arranged semi-formal representation. We categorised the intermediate representations into Train- and Station-related information. Each of the two sub-schemata was further sub-divided into three sub-categories. For the Train metadata component, we proposed Business Information, Technical Information, and Dynamic Execution Information while the Station metadata was divided into Business Information, Technical Information, and Data Information. Finding these taxonomies has the advantage that we decouple the Train and Station metadata since both are autonomous entities in the DA infrastructure and therefore, we have kept them semantically consistent. In addition, this enables less complex modifications of the (sub-)schemata without changing the other. Finally, we transformed this first schema version into a machine-readable and standardised format. In our evaluation, we applied our metadata schema to a proof-of-concept dashboard implementation. We have shown that the integration of our schema enables transparency-enhancing monitoring of a DA architecture and its intrinsic activities. Finally, we evaluated our schema with the FAIR principles and highlighted that our schema mostly fulfils them or contributes to FAIR data management. Our work is an initial foundation, which leverages cross-infrastructural interoperability between different DA implementations. By opening the borders between several DA implementations, researchers have access to more digital assets, e.g., decentralised data or even available analytical task. The compliance with the FAIR principles allows for improved efficiency and usability. Using our schema, components can be equipped with semantically enriched descriptions, which mitigate the black-box characteristics of DA architectures. Additionally, it contributes to an increase in confidence since all components and activities have been made transparent, monitorable, and controllable. Therefore, our work is the first step towards FAIR decentralised data management and FAIR analytics in terms of DA.

Nevertheless, we will continually extend our metadata schema. We are working on a vocabulary extension to cover more domain-specific data sources and types, e.g., healthcare data. Further, an advanced data provenance mechanism is needed to guarantee the trustworthiness of the metadata description since the description can be misleading or malicious. Another subject is the manual entry of the metadata, which is not standardised yet. We pursue a SHACL extension to allow common form generation and user interfaces. A suitable vocabulary is the Data Shapes (DASH) namespace® that assists in such form definitions and provides a standardised user interface.

® DASH: http://datashapes.org/dash

ChinaXiv合作期刊

chinaXiv:202211.00211v1

ACKNOWLEDGEMENTS

First of all, we want to thank all interviewees for their participation in our requirement analysis, which represents the foundation of our work. Further, this work was supported by the German Ministry for Research and Education (BMBF) as part of the SMITH consortium (SW, LN, YUY, SD and OB, grant no. 01ZZ1803K). This work was conducted jointly by RWTH Aachen University and Fraunhofer FIT as part of the PHT and Go FAIR implementation network, which aims to develop a proof-of-concept information system to address current data reusability challenges occurring in the context of so-called data integration centres that are being established as part of ongoing German Medical Informatics BMBF projects.

AUTHOR CONTRIBUTIONS

S. Welten (welten@dbis.rwth-aachen.de) wrote the manuscript, designed the concept. L. Neumann (laurenz.neumann@rwth-aachen.de) developed the schema and wrote the manuscript. Y. Ucer Yediel (yeliz.ucer.yediel@fit.fraunhofer.de), L.O. Bonino da Silva Santos (luiz.bonino@go-fair.org), S. Decker (decker@informatik.rwth-aachen.de), and O. Beyan (beyan@dbis.rwth-aachen.de) reviewed the paper.

REFERENCES

[1] Murdoch, T.B., Detsky A.S.: The inevitable application of big data to health care. JAMA 309(13), 1351–1352 (2013)

[2] Mehta, N., Pandit, A.: Concurrence of big data analytics and healthcare: A systematic review. International Journal of Medical Informatics 114, 57–65 (2018)

[3] Beyan, O., et al.: Distributed analytics on sensitive medical data: The personal health train. Data Intelligence 2(1–2), 96–107 (2020)

[4] GDPR: General Data Protection Regulation (GDPR)—Official Legal Text. Available at: gdpr-info.eu. Accessed 11 April 2021

[5] Atchinson, B.K., Fox, D.M.: The politics of the Health Insurance Portability and Accountability Act. Health Affairs (Project Hope) 16(3), 146–150 (1997)

[6] DPA: Data protection. Available at: www.gov.uk. Accessed 11 April 2021

[7] Sun, C., et al.: A privacy-preserving infrastructure for analyzing personal health data in a vertically partitioned scenario. MedInfo 264, 373–377 (2019)

[8] Shi, Z., et al.: Distributed radiomics as a signature validation study using the Personal Health Train infrastructure. Scientific Data 6, Article number 218 (2019)

[9] Deist, T.M., et al.: Distributed learning on 20000+ lung cancer patients—The Personal Health Train. Radiotherapy and Oncology 144, 189–200 (2020)

[10] Jochems, A., et al.: Developing and validating a survival prediction model for nsclc patients through distributed learning across 3 countries. International Journal of Radiation Oncology, Biology, Physics 99(2), 344–352 (2017)

[11] Jochems, A., et al.: Distributed learning: Developing a predictive model based on data from multiple hospitals without data leaving the hospital—A real life proof of concept. Radiotherapy and Oncology 121(3), 459–467 (2016)

ChinaXiv合作期刊

chinaXiv:202211.00211v1

[12] Chang, K., et al.: Distributed deep learning networks among institutions for medical imaging. JAMIA 25(8), 945–954 (2018)

[13] Das, A., et al.: Collaborative filtering as a case-study for model parallelism on bulk synchronous systems. In: Conference on Information and Knowledge Management (CIKM), pp. 969–977 (2017)

[14] McMahan, H.B., et al.: Communication-efficient learning of deep networks from decentralized data. arXiv preprint arXiv:1602.05629 (2017)

[15] Sheller, M.J., et al.: Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In: Crimi, A., et al. (eds) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 92–104. Springer, Cham (2019)

[16] Su, H., Chen, H.: Experiments on parallel training of deep neural network using model averaging. arXiv preprint arXiv:1507.01239 (2015)

[17] Su, Y., et al.: Communication-efficient distributed deep metric learning with hybrid synchronization. In: International Conference on Information and Knowledge Management (CIKM), pp. 1463–1472 (2018)

[18] Wilkinson, M.D., et al.: The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data 3, Article number 160018 (2016)

[19] Thompson, M., et al.: Making fair easy with fair tools: From creolization to convergence. Data Intelligence 2(1–2), 87–95 (2020)

[20] FAIR principles by GO-FAIR. Available at: www.go-fair.org/fair-principles/. Accessed 11 April 2021

[21] Specka, X., et al.: The bonares metadata schema for geospatial soil-agricultural research data—merging inspire and datacite metadata schemes. Computers & Geosciences 132, pp. 33–41 (2019)

[22] Labropoulou, P., et al.: Making metadata fit for next generation language technology platforms: The metadata schema of the european language grid. In: Language Resources and Evaluation Conference, pp. 3428–3437 (2020)

[23] Franke, S., et al.: Plasma-MDS, a metadata schema for plasma science with examples from plasma technology. Scientific Data 7, Article number 439 (2020)

[24] Jacobsen, A., et al.: A generic workflow for the data fairification process. Data Intelligence 2(1–2), 56–65 (2020)

[25] Lopez, M.F., et al.: Building a chemical ontology using methontology and the ontology design environment. IEEE Intelligent Systems and Their Applications 14(1), 37–46 (1999)

[26] Keet, C.M., et al.: The data mining optimization ontology. Journal of Web Semantics 32, 43–53 (2015)

[27] Fernández-López, M., et al.: Methontology: From ontological art towards ontological engineering. In: AAAI Conference on Artificial Intelligence, pp. 33–40 (1997)

[28] Lucassen, G., et al.: The use and effectiveness of user stories in practice. In: International Working Conference on Requirements Engineering: Foundation for Software Quality, pp. 205–222 (2016)

[29] Cohn, M.: User stories applied: For agile software development. Addison-Wesley Professional, Hoboken (2004)

[30] Can, O., Usenmez, B.: An ontology based personalized privacy preservation. In: International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, pp. 500–507 (2019)

[31] Maali, F., et al.: Data Catalog Vocabulary (DCAT). W3C recommendation.Available at: https://w3c.github.io/dxwg/dcat/. Accessed 11 April 2021

ChinaXiv合作期刊

chinaXiv:202211.00211v1

AUTHOR BIOGRAPHY

Sascha Welten received the B.Sc. and M.Sc. degrees in Computer Science from RWTH Aachen University. He is working as a research assistant and PhD student at the Chair for Computer Science 5 at RWTH Aachen University. His research area includes distributed analytics, semantic information systems for interorganisational interoperability in the context of distributed analytics, and data quality management.
ORCID: 0000-0001-5570-9672

Laurenz Neumann received the B.Sc. degree in Computer Science from RWTH Aachen University and is pursuing the M.Sc. degree in Computer Science. He is working as a student assistant at the Chair for Computer Science 5 at RWTH Aachen, where he is helping to develop a distributed analytics infrastructure. His interests are semantic Web developing and distributed software architectures.
ORCID: 0000-0002-7106-1973

Yeliz Ucer Yediel is a Researcher and Project Manager at Fraunhofer Institute for Applied Information Technology. Her supervised projects deal with FAIR Data Management, Distributed Analytics Platforms, and PID systems.
ORCID: 0000-0002-6845-7774

chinaXiv:202211.00211v1

Luiz Olavo Bonino da Silva Santos is an Associate Professor at University of Twente and Leiden University Medical Center. Additionally, he is the International Technology Coordinator of the GO FAIR International Support and Coordination Office. His background is in ontology-driven conceptual modelling, semantic interoperability, service-oriented computing, requirements engineering and context-aware computing. In the last five years Luiz has been involved in a number of activities to realize the FAIR principles, including the development of a number of technologies and tools to support making, publishing, indexing, searching and annotating FAIR (meta)data.
ORCID: 0000-0002-1164-1351

Stefan Decker is a Full Professor heading the Chair of Databases and Information Systems, RWTH Aachen University, Aachen, Germany. Further, he is the Director of the Fraunhofer Institute for Applied Information Technology, Sankt Augustin, Germany. His research interests include semantic Web, metadata, ontologies and semistructured data, Web services, and applications for digital libraries, knowledge management, information integration, and peer-to-peer technology.
ORCID: 0000-0001-6324-7164

Oya Beyan is a Professor at the Institute for Medical Informatics at University of Cologne and group leader at Fraunhofer Institute for Applied Information Technology. Her research focuses on methods of data reusability and FAIR data, data-driven transformation and distributed analytics. Her area of expertise is in the semantic Web technologies and application of them in health care and life sciences. She actively contributes to the national and international initiatives to enable the adoption of FAIR principles and develops tools and infrastructures supporting FAIR data. With her interdisciplinary background in informatics, medical informatics and sociology, she developed a focus on societal reflections of data-driven change.
ORCID: 0000-0001-7611-3501