Contribution ID: **71** Type: **TALK**

FAIRlead - A Domain Independent, Model Driven Approach to follow the FAIR Data Avenue

Monday 4 November 2024 10:50 (20 minutes)

Enriching data with describing metadata is the key-enabler for the reusability and interoperability of experimental results and thus to further research in a scientific domain. However, in order to be able to use data of former scientific work (both initial data and result data from experiments), a common understanding of the semantics of this data is essential. This understanding is typically achieved by developing or using domain-specific ontologies or conceptual models that describe the relevant entities of a domain and their relationships to each other. However, depending on the domain (photovoltaic, autonomous driving, etc.), the ontologies are different. Accordingly, the metadata in the various domains has a different structure. This also means that for the manual or (semi) automatic acquisition of metadata, each area requires its own tool to record the metadata and link it to the actual data. This is where we want to start with our research by developing a model-driven software development (MDSD) approach that enables us to provide tools for capturing metadata and linking it to the actual data. In the MDSD, source code is generated based on an input model and transformation rules. Specifically, in our case, the code generator receives an ontology and a series of templates describing the mapping of the model information to the target language or platform (PHP, Java, Django, . . .) as input. The functionality of the software to be generated includes an interactive interface for the manual collection of metadata (i.e. experimental setup of a photovoltaic system), a REST-API for the programmatic collection of metadata and as an extension point to embed application logic, an associated persistence component, as well as suitable search and export functionalities. Furthermore, a connection to the respective data is essential, whereby the widest possible range of data formats should be supported here. The non-functional requirements for the software include user-friendliness, easy integration into existing environments, expansion options, and no commitment to a specific target programming language.

Another important aspect of FAIRlead is the integration of existing data towards the domain ontology. FAIRlead shall assist users in both extending the domain ontology and reusing concepts from other existing ontologies to improve interoperability. The resulting combined ontology serves as input to the generator. To form a knowledge graph with the ontology and existing data sources, we integrate both data and metadata within a conceptual model diagram of the relevant sources (including APIs, files, databases and manual user input). To facilitate this, we use an approach to extract the conceptual model from existing (meta)data. The extracted conceptual model can be fine tuned by the user using a visual editor. It then allows metadata to be added to its respective data with a visual graph-based user interface. The generated code will also allow direct access to the integrated data in the knowledge graph.
As part of the presentation, we will go into the basic concepts for the development of the generator and also present our roadmap for the following steps in this area.

In addition, please add 3 to 5 keywords.

Code Generation, Metadata, Ontology based Engineering, FAIR

Please specify ”other”

For whom will your contribution be of most interest?

Scientists and technicians who maintain and operate research infrastructure for data generation

Please assign yourself (presenting author) to one of the following groups.

Researchers

Please specify ”other”

Primary authors: SCHMIDT, Andreas (IT4EDM/IAI); Mr KOUBAA, Mohamed Anis (Institute for Automation and applied Informatics); LIU, Nan; SCHMURR, Philipp (KIT IAI); STUCKY, Karl-Uwe (KIT); SUESS, Wolfgang (KIT)

Presenter: SCHMIDT, Andreas (IT4EDM/IAI)

Session Classification: Session A2

Track Classification: Connecting research data: 6. Interoperable semantics at domain and application level