Towards a semantic format for FIM: supporting German public services using the GerPS-FIM-Microverse ontology pipeline

Maximilian Raupach[1,*], Maximilian Enderling[1], Leila Feddoul[1], Hannes Legner[1], Marianne Mauch[1] and Birgitta König-Ries[1]

[1] *Friedrich-Schiller-Universität Jena, Leutragraben 1, 07743 Jena, Deutschland*

Abstract

The digitalization of the German public administration is challenging because of several barriers. For example, while federal states in Germany provide identical services for citizens, the process and data behind them can be different. To overcome this, a standardization method called Federal Information Management (FIM) was created. FIM aims to provide a standard way to digitalize services offered by the public administration. According to FIM, a service should comprise three components: the process as its core, a form to collect and store the data to be processed, and the service description, ensuring that everyone can understand it. Every part of FIM has its standards. The processes, with the XProzess-standard, and the forms, with a separate XDatenfeld-standard, are essential to digitalize a service. However, these standards exist in a non-semantic format, which prevents reasoning, querying and search. To solve these problems, we introduce as a collection of interactional microservices the GerPS-FIM-Microverse, a small web service avaible through a restfull web api. We describe the pipeline used for constructing the ontologies and describe them. Finally, we discuss the possibilities enabled by our proposed system.

Keywords

FIM, Knowledge graphs, BPMN, XProzess, XDatenfeld, Forms, Microservices, Public Services

1. Introduction

While implementing the German Online Access Law (OZG)[1], the focus is often on giving citizens access to public services via websites and other digital means. A holistic approach providing true end-to-end digitization of administrative processes is still missing. The federal government as a whole in Germany has its own state authority but is divided into 16 federal constituent states with their own constitutions, parliaments, administrative structures, and responsibilities. That makes it very complicated to find a standardized solution for digitalization for all stakeholders. Together with the city administration of Jena and jenarbeit[2], the working group of the open design of digital administrative architectures[3] is analyzing the path from a legal text to a digitized service using the example of basic income support for jobseekers in German municipalities [1].

In Germany, more than 10,000 public services require digitalization. To digitalize these services in a sustainable way, involved data should be sufficiently described in a semantic format. On January 1st 2017, the IT planning council[4] introduced the standard method called Federal Information Management (FIM)[5]. FIM is, therefore, the legally compliant translation of legal language into a standardized language for a concrete public service that is understandable to citizens, companies, and the executive branch. Because of the federal structure described above, public services vary across Germany; due to a potential difference in the underlying process to deliver them or the law they are based on. To overcome this, FIM models services in three layers.

The most general layer, called master information, is provided by the department of legislation, which directly provides detailed information about the execution of the specific law and represents the service at the most general level [2].

The second layer, called reference information, expands upon the master information by describing the information in more detail to illustrate which legal, technical, and organizational modifications need to be made to provide user-friendly administrative services for citizens and companies online [2].

The most specific layer is called local information, which encodes the service offered in only one public administration, including references to software programs and services used. When provided with local information, anyone can understand the steps involved in offering a specific service at a specific location, like a city or municipality [2].

FIM provides master information as a standard: **XZuFi**, **XProzess**, and **XDatenfeld**. This allows data reuse [3] following the one-for-all (in German: Einer für alle (EfA)) principle [4]. Every federal state should digitize services so that other federal states can reuse them and do not have to develop online services themselves, which may lead to incompatibilities. The EfA principle also applies to the development of standards.

- **XZuFi** describes the service itself.
- **XProzess** the actual process involved in offering a service.

2nd Workshop on 'Public Interest AI', September 23, 2024, Julius-Maximilians Universität Würzburg, Germany

*Corresponding author.
maximilian.raupach@uni-jena.de (M. Raupach); maximilian.enderling@uni-jena.de (M. Enderling); leila.feddoul@uni-jena.de (L. Feddoul); hannes.legner@uni-jena.de (H. Legner); marianne.mauch@uni-jena.de (M. Mauch); birgitta.koenig-ries (B. König-Ries)
0009-0002-2197-8377 (M. Raupach); 0009-0007-5039-8538 (M. Enderling); 0000-0001-8896-8208 (L. Feddoul); 0009-0007-1830-3586 (H. Legner); 0000-0003-1478-1867 (M. Mauch); 0000-0002-2382-9722 (B. König-Ries)

[1] https://www.onlinezugangsgesetz.de
[2] www.jenarbeit.de
[3] opendva.eu
[4] https://www.it-planungsrat.de/en/
[5] https://www.it-planungsrat.de/produkte-standards/produkte

- **XDatenfeld** the required data fields for creating online forms.

For the description of processes, FIM uses a format called FIM-BPMN (Business Process Model Notation) [5], which is a subset of the well-known BPMN2.0-Standard [6]. The institutions responsible of three standardized languages - the FIM-module service for XZuFi, the FIM-module processes for standardized, BPMN-2.0-based language XProzess, and the FIM-module data field for XDatenfeld - coordinate between them in the FIM Business and Coordination Centre[6]. The information for these three modules is controlled and approved according to quality criteria by centralized institutions and can be accessed in central repositories. This standardized language of public services is an ideal first step to digitizing the service. While the developed standards are structured and machine-readable, their format is neither semantically described nor machine-interpretable. The latter are needed to allow for intelligent and sustainable online services.

The EU released the *Interoperable Europe Act*[7] for better connected public services for people and businesses in the European Union (EU). The EU provides some standards to aid interoperability between semantic approaches. Firstly, the general core vocabularies for public services – the e-Government Core Vocabularies [7] is an ontology modeling many relevant components. On the other hand, the EuroVoc Thesaurus [8] is a multilingual and multidisciplinary terminology also provided by the EU. These vocabularies provide terms and phrases with descriptions in semantic formats and have started connecting to existing knowledge graphs (KG), such as Wikidata[8]. For truly machine-interpretable data, we need interlinked descriptions, which model relationships of these entities with structured information from laws regulating public services we want to digitalize. KGs provide context information about real-world entities. We introduced our first ontology *GerPS-onto* in 2023 [9], in which we converted the XProzess-information to a semantic representation.

In this paper we propose a semantic representation from the German public standards XProzess and XDatenfeld. Our project partners are working together with jenarbeit to provide us with both standardized, high-quality master information and also localized reference information for the city of Jena. To achieve a useful semantic representation, an architecture that is accessible to everyone is needed. To create this semantic representation, the first step is to create ontologies describing the schema of the data, from which, as a second step, the schema or ontology are populated with concrete data to generate a KG. For this, we propose multiple contributions:

- One of the ontologies we propose is a quality-improved version of the *GerPS-onto* from Fedoul etal. [9].
- Also, in the GerPS-onto, a more detailed way to describe the XDatenfeld-Standard is necessary, for which we propose a further ontology based on this standard (*GerPS-Datafield*).
- For the second step, in which we create instances, a pipeline for integrating FIM information was built using online services. We introduce this set of microservices as the (*GerPS-FIM-Microverse*), which can map FIM information into knowledge graphs. The GerPS-FIM-Microverse is also essential for further research in our working group and will be expanded later.
- Based on the created knowledge graphs, our working group aims to create digitalized public services [9].

The paper is structured as follows. First, we will present related work. The FIM-Informations used in the ontologies of the three FIM-modules will be explained in Section 2, alongside a brief introduction to KGs and the relevance of Artificial Intelligence (AI) in public administration. In Section 3, we introduce the structure of our proposed ontologies. Afterward, we discuss the architecture of the *GerPS-FIM-Microverse* and especially the functionality of its pipelines required to transform data into KGs in Sections 4 and 5, which is demonstrated in Section 6. Finally, in Sections 7 and 8, we conclude our paper and present future work.

2. Related Work

2.1. Federal Information Management

We need standards for translating legal language into structured, standardized information, such as FIM, for intelligent, sustainable, and legally compliant services. For faster and more sustainable digitalization of public services, we need machine-interpretable data in the form of KGs. The basis for most of our research works is the FIM standard. It consists of three different FIM modules.

The FIM module service with the standard XZuFi describes public administration services in Germany derived from law texts. It contains much information, but the most important one is the ID (also called *LeiKa-ID* or *Leistungskatalog-ID*) to identify it in the other modules of FIM.

The second FIM module process describes a detailed process for every service described in the FIM module service. The information about entries of the FIM module process is stored as standards in XProzess files. These so-called master processes consist of multiple process steps, each derived from single paragraphs of German law, and give guidelines on how services in public administration should be handled after the law comes into force. The XProzess-Standard extends a subset of the BPMN2.0 standard by adding more annotations related to specific process steps. XProzess files always contain a BPMN description of the specific process. Annane etal. [10] presented an ontology that supports the conversion from BPMN to semantic information. As explained in Section 1, the XProzess with FIM-BPMN is a subset of the BPMN2.0 standard,

[6]https://www.bmi.bund.de/DE/themen/moderne-verwaltung/verwaltungsmodernisierung/foerderales-informationsmanagement/foerderales-informationsmanagement-node.html

[7]https://ec.europa.eu/commission/presscorner/detail/en/IP_24_1970

[8]https://www.wikidata.org/wiki/Wikidata:Main_Page

enabling the use of the presented ontology and the construction of semantic information.

The third FIM module data field is necessary to obtain and store information these processes can handle. The information about entries of the FIM module data field is stored as standards in XDatenfeld files. These files contain information about the structure of forms and their elements. Each element has a specific ID, which is defined in the XDatenfeld files and can be referenced in XProzess files within process steps. The XDatenfeld files are based on a tree structure and contain three significant elements: the data fields, data field groups, as a collection for data fields, and rules, which describe the relations of the different elements within an application form. XDatenfeld is structured like a tree, with some of its nodes containing rules, e.g., restricting the valid input of a form or controlling the visibility depending on preceding information in the form. One way to model these rules was presented by Roffarello [11]. They also proposed the EUPont-Ontology containing classes and properties to design rules in the general shape of "if-this-than-that". We will apply this ontology to enable the creation of semantic information, supporting the inclusion of meta information in application forms. This semantic information can be converted to other traditional formats if needed.

2.2. AI in public administration

Zheyuan Ryan Shi et al. [12] conducted a survey about research in the area of artificial intelligence for social good. In this survey, they looked at 246 papers and divided them into 16 different categories of different AI techniques. The research of our paper would fit into the category of Knowledge representation and reasoning. This category is one of the least populated, with only 11 papers assigned to it. Any research in this field should be in the public interest since structured knowledge is essential in many areas. For example, structured knowledge is required when working with law texts, which are given in the public administration sector. The law sets very strict conditions requiring concrete data to be publicly accessed.

2.3. Knowledge Graphs

Our paper aims to provide a semantic format in the form of knowledge graphs (KG), which belongs to the category of Knowledge representation and reasoning of AI [12]. KGs create a semantic meaning of the data by relating real-world concepts to the stored data. They also provide other benefits. For example, data can be linked by concepts and relations. The technology also enables machine readability and interpretation of the data [13]. In our paper, we use the term ontology for the schema and the term knowledge graph for the populated instance based on the schema. As KGs store their schema as part of their data, they also do not require classic migrations known from relational databases [14] and are better able to keep compatibility between different versions of schemata. Semantic frameworks are needed to build this semantic format, e.g., the resource description framework (RDF). RDF is a framework for communication on the web and describes a graph via triples of internationalized resource identifiers (IRI) by describing the resources in a graph. The FIM standard describes three modules, which are specific XML formats. A main contribution of the work outlined in this paper is the conversion from data stored in an XML file to a semantic version. XML is a commonly used format. Breitling describes one of many generic transformations from XML to RDF [15]. Their outlined approach aims to provide a general solution for any XML. Files of the method FIM contain information on this particular context of digitizing public services in Germany, which will be lost when the generalized approach from Breitling is applied. Instead of taking the presented approach, we benefit from a more context-based solution that directly encodes entities unrelated to their structure in XML. The **BBO** for the BPMN processes is reused in *GerPS-onto* [9]. In addition, the **EUPont** for describing the rules will be reused in *GerPS-Datafield*.

3. The GerPS Ontologies

In this section, we propose an update to the GerPS-onto and extend the ontology to the GerPS-data field [16], which enables the modeling of FIM information as semantic information that can be stored in KGs. Namely, the GerPS-onto[9] models information encoded in XProzess and is extended by GerPS-Datafield, an ontology, to model XDatenfeld.

The new version 2.0 of the GerPS-onto, depicted in Figure 1, contains classes and relations to create a semantic standard that can be used to describe BPMN2.0-processes annotated with FIM information.

The GerPS-onto is based on FIM's XProzess information. To create entities in the GerPS-onto, process steps' universally unique identifiers (UUID) are directly converted into IRIs. The process steps' UUIDs can also be found in the BPMN process for the same process step. This way, the information about both sources can be combined in one knowledge graph. Also, the LeiKa-ID is reused to create the process IRI to identify the process. The scaling of solutions is a crucial part of a federal environment. FIM is scaling through its construction of the LeiKa-IDs. Every service is allocated its own ID depending on the locality, type, and public administration action of the service. The KGs are scaling equal to this by using these IDs in our IRIs on the instance level.

GerPS-onto version 2.0 resolves a problem that existed in the older version, which will be discussed in this paragraph. Formerly, process steps directly pointed to the agent, and its role is executing this step. However, the problem occurred when multiple instances with the same agent in different roles appeared. The role was assigned to the agent and occurred in all process steps over several processes. It was impossible to tell in which process the agent had which role. To solve this problem, PROV-O's association was used to prevent this case. Every process step is assigned to an association. The association provides information on which agent with which roles is assigned to the process step. This way, the agent is not directly referenced

[9] https://w3id.org/GerPS-onto/ontology/ date: 27. June 2024

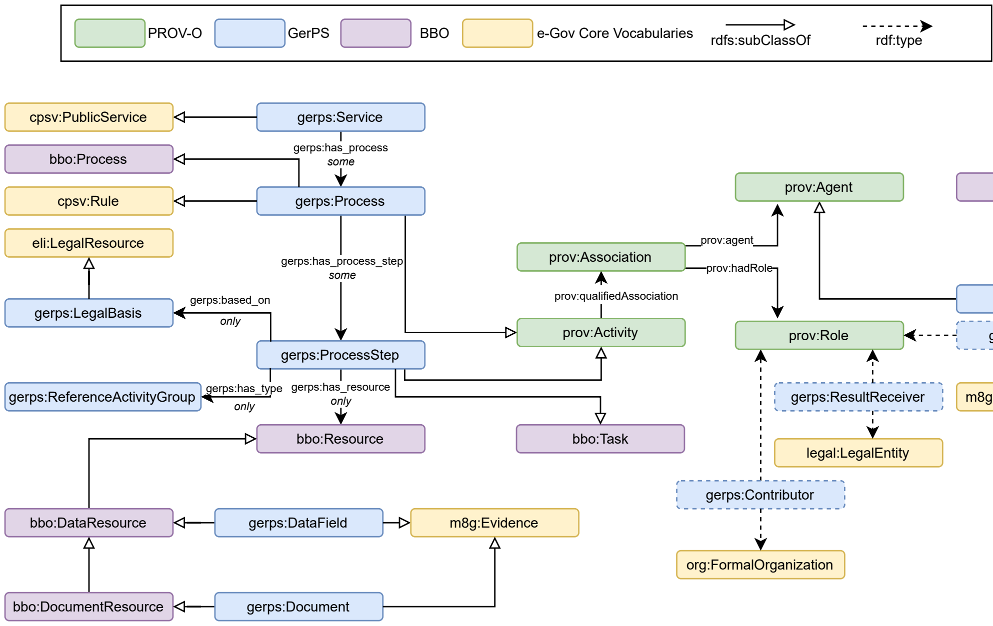

Figure 1: Structure diagram of the ontology *GerPS-onto* Version 2.0

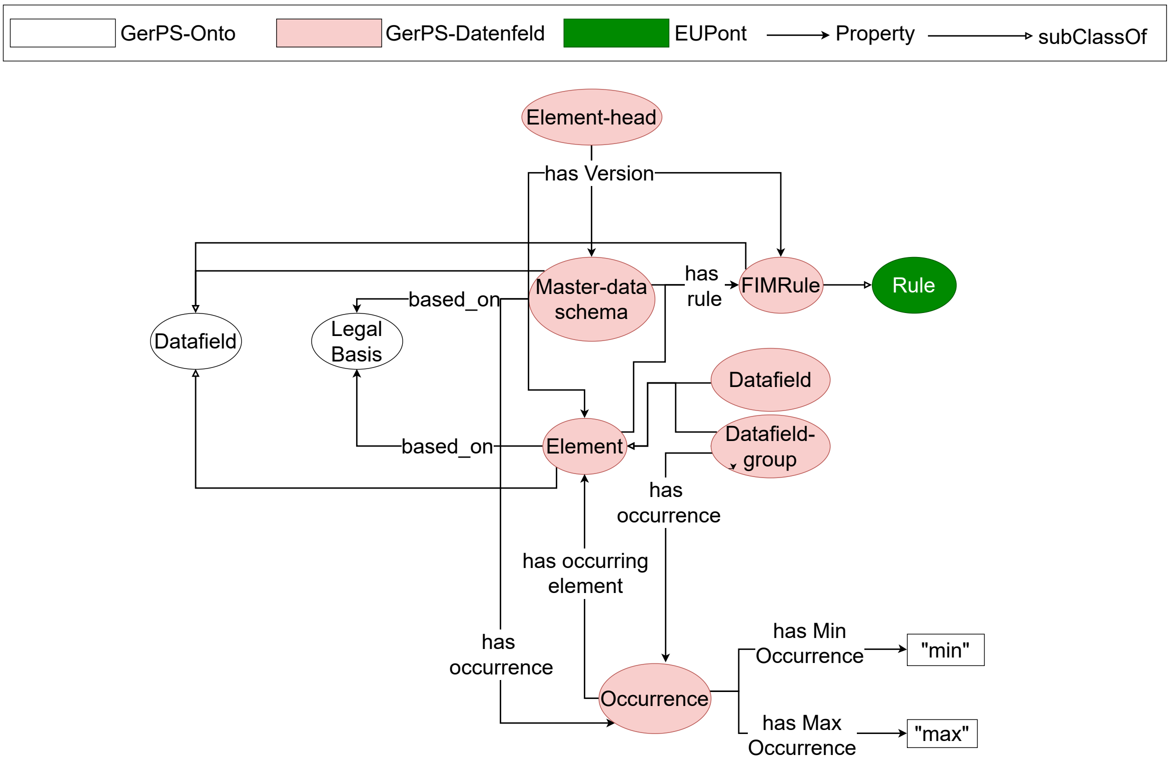

Figure 2: Structure diagram of the ontology *GerPS-Datafield*

to a role, and the problem is solved. PROV-O is an ontology that provides provenance information.

The GerPS-Datafield, shown in Figure 2, is an extension of the GerPS-onto. It contains more specific information about how the FIM module data field is modeled in FIM and which rules they follow. The ontology can be used to compare data fields, check if the content is valid, or even directly create forms from the knowledge graph. The basic data structure of XDatenfeld is a tree, and the structure is mirrored in the semantic representation. The tree consists of three major types of nodes: data field group and data field, as well as rules about them. The data field group can contain other groups and data fields. Within this tree, there are multiple data field groups and data fields. Some of them are also used in different forms. So they get harmonized within a library called *BOB* [17]. The reuse of these elements ensures equal quality of rules in every form. Therefore, these data fields also have the same ID. When it is updated, its version gets updated, too. The so-called *VersionID* is the identifier for a specific field. Out of this VersionID, the IRIs for the individuals have been created. By reusing the VersionID, the same data fields can be identified over many forms in one knowledge graph.

The ontologies are modeled for the two crucial FIM standards **XProzess** and **XDatenfeld**. The instances are currently missing. We prepared a pipeline that is shown in the following two sections.

4. Architecture of the GerPS-FIM-Microverse ontology pipeline

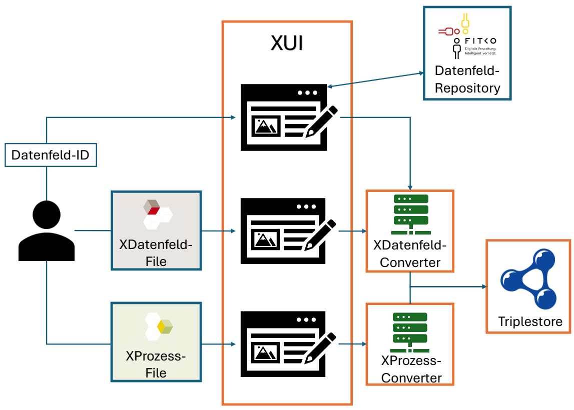

Figure 3: Architecture of GerPS-FIM-Microverse. Blocks with orange outlines indicate pre-existing resources, green outlined blocks depict newly required ones

We are designing a development kit consisting of methods and software to efficiently use the complex software around the digitization process of public services with the help of KGs. A pipeline was made to instantiate the KGs with the help of the above-described ontologies. In this section, the architecture of this pipeline is proposed. The goal of this architecture is first to provide a simple interface for everyone to use, and second to provide interfaces for additional services. Figure 3 visualizes the architecture and its different components, which will be introduced in the following.

Our architecture comprises multiple microservices, communicating with each other using internet protocols such as HTTPS.

The primary access point is a web app called *XUI*[10]. Every component in our architecture has its respective user interface as a web page. While two web pages specifically allow the upload of FIM information, the XUI also has interfaces with external data sources. The publicly available data source is provided by the FITKO[11], a German institution under public law. Our direct integration allows for the conversion of these files to semantic equivalents with minimal friction, as shown in this demonstration.

The task of converting FIM information to the semantic formats explained in more detail in Section 3 is solved by the two microservices *XDatenfeld-Converter* and *XProzess-Converter* as the *XPServer*. They are also directly responsible for transferring the semantic information to our storage. While the conventional approach would use relation databases, semantic data can benefit by being stored in a *Triplestore* [18] as they are specialized for this task. For this purpose, we also use one instance of the open-source software *Fuseki*[12].

The architecture is designed using the API-first approach, where the first step is to build the working backend-APIs (*XDatenfeld-Converter* and the *XProzess-Converter*). After that, we developed the *XUI*. This approach allows our services to use the converter over HTTP, makes the system more interoperable and easy to extend, and enables their use for external systems.

5. GerPS-FIM-Microverse ontology pipeline

In the previous section, we offered the architecture of the Architecture of the GerPS-FIM-Microverse ontology pipeline. Now, we introduce its functionality. The XUI, the XDServer, and the XPServer are the three services that are part of the GerPS-FIM-Microverse. How they work with each other is detailed in the Section 4 on architecture. A detailed overview of the workflow of the XPServer and XDServer can be seen in Figure 4. The following section will explain this workflow in further detail.

Users can input their XProzess file into the XUI, which will be uploaded to the XPServer. On this server, the file content will be parsed into Java classes. First, the server will extract metadata about the public administration service described by the file. Subsequently, information about the individual process steps is extracted. The final step extracts the BPMN file, encoded in the XProzess file. It is decoded and can be output as plain text in the XUI. With all the data from the XProzess file, the server generates RDF triples describing the process's flow. These triples can be displayed as plain text in the XUI and uploaded to a triple store.

The XDServer works similarly, allowing users to upload their XDatenfeld files via the XUI. It parses the content into Java classes, using a tree data structure described in Section 3. This tree is traversed to collect information about the structure of the form described in the XDatenfeld file. At the same time, while traversing, the server extracts data about the FIM data fields

[10] https://xui.simplex.fmi.uni-jena.de/

[11] https://www.fitko.de/

[12] Apache Software Foundation, 2021. Apache Jena, Available at: https://jena.apache.org/.

and the related groups. The corresponding descriptions will be parsed if the server encounters a rule as explained in Section 3. Like the rest of the information, these rules and their descriptions will also be parsed into RDF triples and stored as a reference in the according element. After finishing the traversal of the tree, all the triples are sent back to the XUI so they can be displayed or stored in the triple store.

The source code of our services will be hosted on the OpenCoDE GitLab[13] and Zenodo [19] with an open-source license to allow reuse and modification.

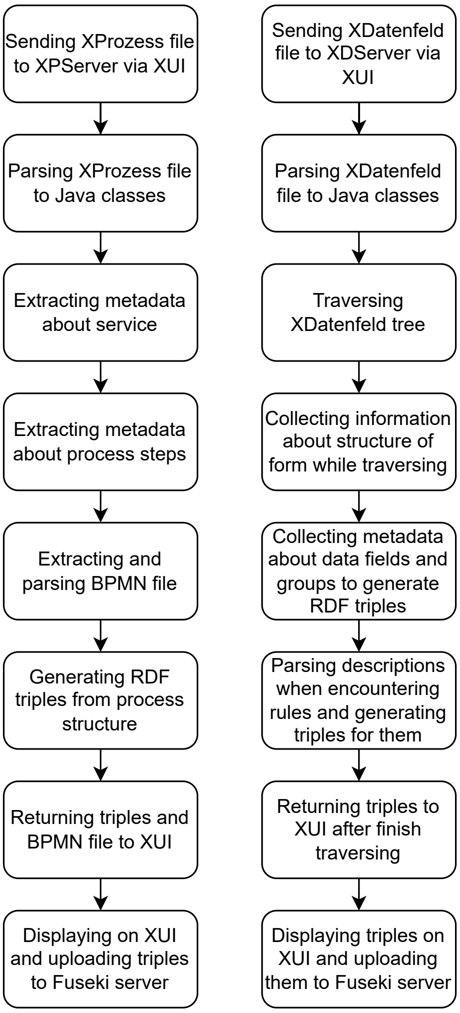

Figure 4: Workflow of the *XPServer* and *XDServer*.

6. Demonstration

We want to present the XUI from our GerPS-FIM-Microverse ontology pipeline in the demonstration. For demonstration purposes, the XUI and its pipeline can be used on the public XUI-Website[10]. Alternatively, the complete pipeline including all services can be downloaded from our release on Zenodo [19] run locally using docker-compose. In the XUI, the described pipelines are easily accessible. XImport can search in the FITKO-repository and import the FIM data fields. The Files can be used to run the pipeline using XDatenfeld-information. For the XProzess-pipeline, an XProzess file in version 2.0 is required. We provide several examples in the Zenodo repository for the XUI [19]. The file can be uploaded using drag and drop. Then, one of the provided actions can be selected in the XUI. The described actions are shown in Figure 5.

In the following section, we want to discuss some drawbacks and other issues.

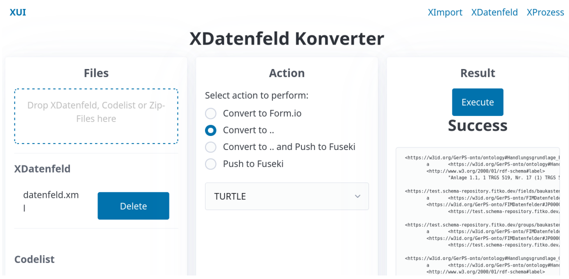

Figure 5: a screenshot of the XUI processing an XDatenfeld-file

7. Conclusion and Discussion

This section discusses the created knowledge graph, its drawbacks, and some design decisions.

7.1. Ontologies

We wanted to show that it is possible to create KGs based on XProzess and XDatenfeld. The two ontologies that can be used to create KGs are presented in Section 3.

We built a new ontology as an extension of GerPS-onto, called GerPS-Datafield. GerPS-Datafield is based on the information XDatenfeld mentioned. However, not all information from XDatenfeld can be found in the resulting knowledge graph. This is for various reasons: not all information is essential for the work, like the header of the XDatenfeld information. The header mainly contains meta information about the transfer of XDatenfeld-Information and will change for each transaction between data repositories. The GerPS-Datafield ontology contains valuable information describing forms, like the structure and meta information about all elements.

Another point to discuss is that multiple formats represent forms, like the well-known form.io-JSON[14] format, which is used to generaly describe forms. So why is the GerPS-Datafield based on XDatenfeld and not another data source? The reason for that is that a German standard for public administration was needed. Such a standard has to be an XöV-standard, a standard given by the German standardizing administration KoSIT (Koordinierungsstelle für IT-Standards), which coordinates these kinds of standards. Namely, XZuFi, XDatenfeld, and XProzess are standards of XöV. A further reason was that a standard was needed, which describes forms in general. So, the only possible option was the XDatenfeld. Every other XöV-standard describes a specific use case. The advantage is that

[13]https://gitlab.opencode.de/opendva

[14]https://www.form.io

XDatenfeld also includes provenance for laws, which is helpful when creating a system that tells you why information will be needed and on which law this is based.

7.2. GerPS-FIM-Microverse

Further, to create the KGs based on these two ontologies, we also presented a pipeline based on the GerPS-FIM-Microverse and its architecture in the Section 4 and Section 5. The purpose of the pipeline was to create a publicly accessible semantic format out of the two mentioned standards, XProzess and XDatenfeld. With the help of the pipeline, everyone can create their own instances of our semantic format based on the XProzess- and XDatenfeld-standards. Everything that is needed is the corresponding file. However, there is one obvious downside to this. The pipeline creates triples out of the file; it does not control their quality. So, the data in the file need to have high quality; if not, the pipeline does not provide high-quality triples.

A last issue that should be discussed is that there are many other pipelines that convert an XML format to a semantic format. In our case, this option was not sufficient. These pipelines generate a general knowledge graph, which is unsuitable for our purpose. We need a reasonable knowledge graph with specific connections to gather information and connections between entities. Pipelines that create a more general format like Breitling [15] do not generate such information, so, in the end, context information must be added. So, in our case to better enable automatic reasoning over KGs, it is better to implement a new, more specific semantic format instead of a general one.

One particular microservice we showed was the XUI, our user interface to control the pipeline. In Section 6 we showed the XUI and demonstrated the usage of it. We validated it with the help of several project partners.

7.3. AI for Public Interest

In their article, Theresa Züger and Hadi Asghari [20] propose a framework of requirements to classify AI-related projects and how well they serve the public interest. This framework consists of five points: public justification, serving equality, co-design process, following key technical standards, and openness for validation. The GerPS-FIM Microverse presented in our paper serves a public justification in the sense that it tries to support the public administration sector of Germany. It tries to serve equality by applying open standards in the structure of the data, which has the goal of achieving semantic interoperability. This way every local administration could use this data equally. The XUI allows everyone to contribute their own FIM-Information to the KGs, which helps public administrative staff to co-design the data. Since there are strict requirements for FIM-Informtation to be valid, the quality of the data is guaranteed by the nature of the used standards. FIM is also regularly validated and updated, which makes the GerPS-FIM Microverse open for validation by design.

8. Future work

Our introduced GerPS-FIM-Microverse ontology pipeline enables multiple possible applications of varying complexity, which will be discussed in the following section.

Our proposed knowledge graph can be used to annotate BPMN to import them into process execution platforms. This could be the basis for partially automating the digitalization of public services. In addition, our knowledge graph could serve as the basis for a range of advanced applications. For example, a simple use case could be a semantic search or automatically tracked changes to the law. Rules and regulations are frequently changed and adapted to new circumstances or requirements. A formal representation allows a simple comparison between two versions. This work is currently in progress.

In the future, the existing ontologies can be extended by the service level of FIM. Services are already represented as a class in GerPS-Onto, and can reference a service from the FIM service module. This entity can be used in a FIM-service-ontology to connect the processes with its service description. On the other hand, elements from data fields are also identified via their own IDs. These are also used at the instance level to create the IRIs. So, in a service-level ontology, these entities can also be reused to connect the ontologies.

When services can be digitalized with KGs, further information about the process could also be obtained, e.g., from the enforcement of the service, and be stored in the knowledge graph with links to digitalized services. This enables feedback for the next legislation process. It could help during the creation or revision of laws and aid, e.g., in detecting gaps in law that can be abused.

The semantic description of services with information from the enforcement could directly enable the creation of systems capable of simulating the implications of a law. This would be realized by populating a data source using several unique citizen-like profiles. Simulating the law's effects using automated engines would enable the detection of potential loopholes and costs. With a system like this, legal experts can get fast feedback and possible adjustments to the laws before they come into effect.

Besides this, a knowledge graph also benefits the modernization of registers in Germany. Registers are the official institutions that store data relating to citizens for the government. Currently, registers store data in a fashion that is similar to that of relational databases, where everything is stored in table-like structures. These structures have fixed schemas in which data is described, which leads to the need to pay special attention during changes, often requiring so-called migrations between different schema versions [14]. Another opportunity for registers arises by using the same IRIs in the registers as in XDatenfeld-information, which enables the automatic merging of information in forms and registers. When applying this process, forms can be prefilled using information stored in registers, reducing the information that citizens need to provide. Also, data from registers is assumed to be correct, reducing possible hiccups encountered when incorrect information is provided.

To summarize, the presented KGs can be used to implement many beneficial systems, making life easier for citizens. Also, feedback on the system can aid in accelerated creation of services, providing better security and robustness. Our developed knowledge graph is the first law-compliant structure of a law. The current version of the knowledge graph is the first schema of a law. An extended version could enable the generation of laws.

9. Acknowledgments

The research projects Canarėno, simpLEX, and KollOM-Fit of the working group openDVA were funded by the Federal Ministry of the Interior and Community, FITKO, and the Thuringian Ministry of Finance in Germany in the scope of the OZG implementation as well as the FITKO from the digitization budget. We would like to thank all employees, project partners, and supporters of the openDVA working group, who could not be mentioned here by name, for their great support, helpful comments, discussions, and good cooperation.

References

[1] M.Mauch, S.T. Bachinger, P.Bornheimer, S.Breidenbach, D.Erhardt, L.Feddoul, H.Legner, F.Löffler, F.Löffler, M.Raupach, S.Schindler, J.Schröder, B.König-Ries, From legal texts to digitized services for public administrations, IRIS24: International Legal Informatics Symposium 2024 (2024). URL: https://www.easychair.org/publications/preprint_english/PsVv. doi:10.38023/bf44ffb5-7a03-4334-a26b-8a6a902c4115.

[2] Arbeitsgruppe “Offenes Design Digitaler Verwaltungsarchitekturen” (openDVA), Föderales informationsmanagement (FIM), 2022. URL: https://docs.fitko.de/kompass/docs/grundlagen-und-rahmen/fim/.

[3] Bundesregierung, Entwurf eines Gesetzes zur Änderung des Onlinezugangsgesetzes sowie weiterer Vorschriften zur Digitalisierung der Verwaltung (OZG-Änderungsgesetz – OZGÄndG), 2024. URL: https://dserver.bundestag.de/btd/20/080/2008093.pdf, Drucksache 20/8093.

[4] Bundesministerium des Innern und für Heimat, Digitale Verwaltung - “Einer für Alle”-Prinzip, 2024. URL: https://www.digitale-verwaltung.de/Webs/DV/DE/onlinezugangsgesetz/efa/efa-node.html.

[5] Bundesverwaltungsamt, Der Attributkatalog für BPMN 2.0, 2023. URL: https://www.bva.bund.de/SharedDocs/Downloads/DE/Behoerden/Beratung/Prozessmanagement/PMT/Attributkatalog_PMT.pdf.

[6] O.OMG, R.Parida, S.Mahapatra, Business process model and notation (bpmn) version 2.0, Object Management Group 1 (2011) 18.

[7] European Union, e-Government Core Vocabularies handbook, 2015. URL: https://ec.europa.eu/isa2/sites/default/files/e-government_core_vocabularies_handbook.pdf.

[8] European Union Publications Office, Eurovoc, 2024. URL: https://eur-lex.europa.eu/browse/eurovoc.html.

[9] L.Feddoul, M.Raupach, F.Löffler, S.Babalou, J.Hoyer, M.Mauch, B.König-Ries, On which legal regulations is a public service based? fostering transparency in public administration by using knowledge graphs, in: INFORMATIK 2023 - Designing Futures: Zukünfte gestalten, Gesellschaft für Informatik e.V., Bonn, 2023, pp. 1035–1040. doi:10.18420/inf2023_115.

[10] A.Annane, N.Aussenac-Gilles, M.Kamel, BBO: BPMN 2.0 Based Ontology for Business Process Representation, in: 20th European Conference on Knowledge Management (ECKM 2019), volume1, Lisbonne, Portugal, 2019, pp. 49–59. URL: https://hal.science/hal-02365012.

[11] A.M. Roffarello, End user development in the iot: A semantic approach, in: 2018 14th International Conference on Intelligent Environments (IE), 2018, pp. 107–110. doi:10.1109/IE.2018.00026.

[12] Z.R. Shi, C.Wang, F.Fang, Artificial intelligence for social good: A survey, CoRR abs/2001.01818 (2020). URL: http://arxiv.org/abs/2001.01818. arXiv:2001.01818.

[13] A.Hogan, E.Blomqvist, M.Cochez, C.d’Amato, G.deMelo, C.Gutierrez, S.Kirrane, J.E.L. Gayo, R.Navigli, S.Neumaier, A.N. Ngomo, A.Polleres, S.M. Rashid, A.Rula, L.Schmelzeisen, J.Sequeda, S.Staab, A.Zimmermann, Knowledge Graphs, Synthesis Lectures on Data, Semantics, and Knowledge, Morgan & Claypool Publishers, 2021. doi:10.2200/S01125ED1V01Y202109DSK022.

[14] J.F. Roddick, A survey of schema versioning issues for database systems, Information and Software Technology 37 (1995) 383–393. doi:10.1016/0950-5849(95)91494-K.

[15] F.Breitling, A standard transformation from XML to RDF via XSLT, Astronomische Nachrichten 330 (2009) 755–760. doi:https://doi.org/10.1002/asna.200811233.

[16] M.Raupach, Gerps-datafield, 2024. doi:10.5281/zenodo.12621817.

[17] FITKO, Veröffentlichung des Baukastens BOB, 2020. URL: https://fimportal.de/news-details?tx_news_pi1%5Baction%5D=detail&tx_news_pi1%5Bcontroller%5D=News&tx_news_pi1%5Bnews%5D=11&cHash=f2570e3fe47917f615afac47b798ba26.

[18] J.J. Carroll, I.Dickinson, C.Dollin, D.Reynolds, A.Seaborne, K.Wilkinson, Jena: implementing the semantic web recommendations, in: Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters, WWW Alt. ’04, Association for Computing Machinery, New York, NY, USA, 2004, pp. 74–83. doi:10.1145/1013367.1013381.

[19] M.Raupach, H.Legner, M.Enderling, Prozess Knowledge Graph, 2024. doi:10.5281/zenodo.12569115.

[20] T.Züger, H.Asghari, AI for the public. how public interest theory shifts the discourse on AI, AI Soc. 38 (2023) 815–828. doi:10.1007/S00146-022-01480-5.