Medical information-graphs, based on ontologies and FHIR

Gerhard Kober[1] and Adrian Paschke[2]

[1] Tiani Spirit GmbH, Vienna, Austria
gerhard[DT]kober[AT]tiani-spirit.com
[2] Fraunhofer FOKUS and Freie Universitaet Berlin, Berlin, Germany
adrian[DT]paschke[AT]fokus.fraunhofer.de

Abstract. Clinical relevant information about patients is stored in different locations and often hardly accessible for medical doctors and researchers. The treating doctor needs access to all available data from different sources, because not just locally available data but also data in remote locations and the relationships among different resources are relevant for patient care, e.g., the influence of a heart-rate on the medication intake. In this paper, we propose an approach for integrating FHIR (Fast Healthcare Interoperability Resources) for medical doctors' or researchers' unique needs/questions without changing the original data generated during a clinical medical process. Our approach combines semantic technologies, RDF (Resource Description Framework) data and OWL (Web Ontology Language) ontologies, with the medical standard FHIR, to transform medial information in a semantically annotated knowledge graph. The medical knowledge in the resulting graph is for the consumer (who needs the generated data) and for services of a "Distributed Medical Rule Engine" (DMRE). The service takes care of retrieving the information from different FHIR-stores, but also on the transformation to an RDF-graph. The resulting RDF-graph contains the clinical medical data, and also the connections between the entities. The knowledge graph contains applicable information for either a treating doctor or a researcher who, e.g., needs to explain observations and discover not-yet-explainable insights.

Keywords: FHIR · RDF · Ontologies.

1 Introduction

Clinical medical information about patients and the population is highly relevant for physicians and medical researchers to find new ways of treating diseases [15]. Furthermore, there is a need for collaboration of different organizations (e.g., hospitals, government) to achieve an almost complete view of patient-associated data [4], which is the basis for data-driven conclusions about an illness.

Considering the relevance of selected data sets and their relations and correlations is different for every medical researcher, an approach is needed to firstly allow a researcher to get the required information depending on the research context, secondly from multiple sources, and thirdly in an interoperable way to enable further processing. By now, data collections out of medical research studies are siloed within one organization and focused on the specific need of the research question. An integration from clinical data, generated during routine-examination is, for many reasons (e.g., data-protection, legal aspects, particular needs, ethical aspects), not done. An approach for a standardized, interoperable distribution of clinical-routine-data is required (usage for retrospective studies). Such a method can also be integrated into point-of-care information systems to allow a physician a more detailed view of patient's information.

Since data is already existing in various sources and is (partly) available for retrospective studies, the problem to be addressed is the digital curation of single unique data sets containing medical information and relationships to create value for either a doctor or a researcher. Retrospective data collection is especially useful when investigating factors that contribute to limited events and not yet known research questions since data may be collected from various sources. Moreover, retrospective data collection is less expensive than prospective collection [25]. A retrospective study relies on data that was created before beginning the study, but that does not necessarily contain dependencies [13] which are relevant for answering particular research questions.

In this paper, we contribute by practically implementing a semantic curation approach enabling the semantic mapping and ontological enrichment of secondary medical data in FHIR-format into a RDF-knowledge graph. During the semantic mapping into RDF the input FHIR-resources are enriched by an ontology that semantically represented the particular research needs. This means we semantically curate existing medical data and information about a patient to provide a medical researcher with new information or aspects of the existing data. For example, a laboratory only creates test-results. However, for the overall information about the patient there is corresponding information missing like blood-group, medications, or diagnoses. All this information together has a high impact on the laboratory-result, the diagnoses, and most important, also on the medical therapy.

This paper is structured as follows: in Section 2 we describe background and related work providing information about FHIR and RDF and the prerequisites for the targeted solution. The solution-section (3) describes the technical and implementation details. Section 4 describes an example use case. We evaluate the solution in Section 5 and discuss it in 6. Finally, we conclude this paper with the pros and cons in Section 7.

We contribute by translating FHIR into RDF and enrich the semantics of the resulting RDF knowledge graph with ontological knowledge (OWL ontology) during the curation process.

2 Background and Related Work

FHIR (Fast Healthcare Interoperability Resources) is an HL7 (Health Level 7)-Standard framework, which is based on modular components called resources. It supports RESTful architectures and is based on web-standards such as XML, JSON or HTTP and OAuth (Open Authorization) [10]. Furthermore, the defined FHIR-resources are meant to be interoperable and can be extended to fit specific use cases within a project [7] . With common standards, XPath(XML Path Language) or JSONPath, information can be queried from the JSON or XML-representation of the resources. FHIR also describes a way to convert resources to RDF-format. Since FHIR is resource-based and does not cover the relations among each resource entirely, there is a need to recreate an RDF-graph for performing operations over a complete information set. For example, if the doctor does not provide a diagnosis for the patient, which is stored in the FHIR-server, we need to find another way to apply a (potential found) diagnosis. The FHIR-standard helps in terms of interoperability and accessibility. The defined structures are standardized and allow usage for further processing. Accessing FHIR-resources is done by HTTP-Methods, and can be secured using, e.g., OAuth [12] [9][1].

The Resource Description Framework (RDF) is a standard model for data interchange on the web [22]. The basis for the data model is a defined formal semantics and directed graph structure. Data in RDF are statements about resources (URIs) (Uniform Resource Identifier), which are modeled as triples (Subject - Predicate - Object)[17]. The RDF-model allows describing relationships among resources, enabling integration of other RDF-graphs (knowledge-bases) and enrichment of the overall graph's knowledge, e.g. with ontologies. Furthermore, there is a standard for querying (SPARQL) the RDF-graphs. FHIR does not fully support the interchange of different FHIR-resources [8] and the relations between them, but RDF does.

The Web Ontology Language (OWL) [19] is a Semantic Web language representing semantic knowledge about things and their relations. OWL is designed for semantic interpretation of RDF data by applications. It adds an expressive ontological vocabulary and a formal semantics to RDF and RDF-S(RDF-Schema) [20]. OWL allows defining a structure of classes, (sub-)properties, or restrictions. Also, a description of individuals is possible (and needed).

The input for processing of FHIR-data into RDF is the generated raw-data during clinical routine processes. Today's hospital information systems (HIS) use healthcare standards like HL7- Version2 (text-based), HL7- Version3 (using a formal methodology defined in the HL7 Development Framework HDF), or HL7-FHIR. There are different use cases and aspects in a healthcare environment, which need consideration to find the correct store-location for health information. First, there are patient-centered aspects which are critical for individual patient care. Secondly, there are legal viewpoints, and a third aspect are clinical-medical research topics. For patient-centered issues, this means that an interoperable infrastructure is needed for patient treatment and doctors' collaboration, as described by "Integrating the Healthcare Enterprise" (IHE). Such an infrastructure can also take care of the legal aspects, and since the IHE-model is patient-centered, it is used for medical treatment. IHE also provides an approach for research covered in the "Quality, Research and Public Health" domain [14]. As IHE does not describe how to store information and advises to know in advance what to store, a possible solution is forwarding the clinical data in a FHIR-store (and optionally pseudonymize data during the storage process) and provide the information for research questions in such a location. This store needs to be accessible by consumers (in a secure way) [16].

FHIR-resources can be accessed by HTTP-requests and the queries can be adapted to address the information requests of a consumer. Using direct communication has a disadvantage: a consumer can not query multiple stores at the same time (and even get just a single FHIR-resource at a particular time), and the consumer needs exactly to know, where which needed resource is located in terms of endpoints (FHIR-servers). Secondly, it returns single FHIR-resources or FHIR-bundles (by using an operation called "Operation-patient-everything" [18]. This lacks in either being a dataset of only the requested research data values or being patient-centered, which is not needed in our case. The additional relations/dependencies between FHIR-resources are also missing. Therefore, the FHIR-data-structure cannot be considered to be an adequate data format for performing queries over a result-set. A different approach for sharing information is required since other potential sources for information need to be queried beside FHIR-stores and integrated into the results. [3].

In medical research, there are multiple existing patient data collection methods. Among these methods, there are "Retrospective record review", "Record review of current inpatients", and "Key Informant Interviews" [5]. These methods are based on previously stored medical records but require a senior physician's explanation of all side effects that are in the data. Also, a physician needs to decide if the given data set is part of the study data. This process is long lasting and takes a lot of resources, only for data preparation.

Another approach is using machine learning for data analytics to discover valuable patterns by analyzing massive numbers of unstructured, heterogeneous, non-standard, and incomplete healthcare data [21]. This proposal uses numerous methods like exploratory analysis, descriptive modeling, predictive modeling, discovering patterns, and retrieval by content. However, this strategy is used for preventive medicine, focusing on machine-learning, but clearly stating that interoperability is crucial for improving patient care [24], reducing errors, and saving budget. When using machine learning, the algorithm decides on the necessity of values; this means a value is probably not relevant for the algorithm, but a medical doctor or a researcher might need this certain value or result for further evaluation or inferences.

3 Proposed solution

In our solution the user specifies the requirements of an expected or needed result. A specific research question defines the requirements that are relevant to the researcher. Necessary information could be, e.g., medication intake, blood group of a patient, or the age, or the outcome of a laboratory result. The requested information is semantically defined within an ontology representing the relevant classes/concepts of the information domain and the semantic relations between them.

The defined ontology expresses the classes, which are mapped from FHIR-Resources, while datatype properties related to the classes are the attributes of the FHIR-resources. The ontology's object- properties represent the relations to other classes (e.g., a "has Medication"- property). This specific description allows the user to obtain a reduced dataset, with just the relevant content, or enrich the information available in the different FHIR-stores by relations.

Mapping-table (Table: 1) shows the general mappings from the FHIR-context to the semantic context. The particular FHIR-resource (e.g. Patient) maps to a defined class in the ontology (owl:class). The FHIR-Resource-URL is mapped to the RDF's subject, while the attribute and its values are mapped to the RDF-predicate and the RDF-object. If the FHIR-Resource-Attribute-value is not of type *PrimitiveType* (e.g., String, Integer, boolean)[6] the mapping rules are as follows: The entire object gets a new RDF statement. The object of the initial entry is then the subject of the new statement. For example, the patient has a "Name" of type "HumanName". In this case, the "Name" maps to the subject, the Attribute family maps to the Predicate and the value then to the Object.

FHIR-Context	Semantic Context
FHIR-Resource	*rdf:type*
FHIR-Resource-URL	Subject
FHIR-Resource-Attribute	Predicate
FHIR-Resource-Attribute-value	Object

Table 1. Mapping from FHIR-Context to the RDF-model

As shown in Figure 1, the consumer (e.g., a researcher) initiates a query (containing a patient-id) to an HTTP-Service located within a so-called "Distributed Medical Rule Engine" (DMRE). This DMRE is a component, which provides different services for performing SPARQL-queries, getting information from FHIR-stores and reading ontologies in OWL. After the initiated query, the DMRE calls the "OWL-read-service" to receive all the classes, attributes, and relations. This information is then used for specific FHIR-queries at the different FHIR-endpoints. Specific queries for the FHIR-stores are more efficient (in terms of timing) than "get-all-queries". Secondly, we process the returned dataset, and therefore reduce memory consumption by reducing to a specific information-collection. Once the FHIR-resources are returned to the DMRE, we generate an RDF-graph out of the information combined with the ontology defined by the consumer.

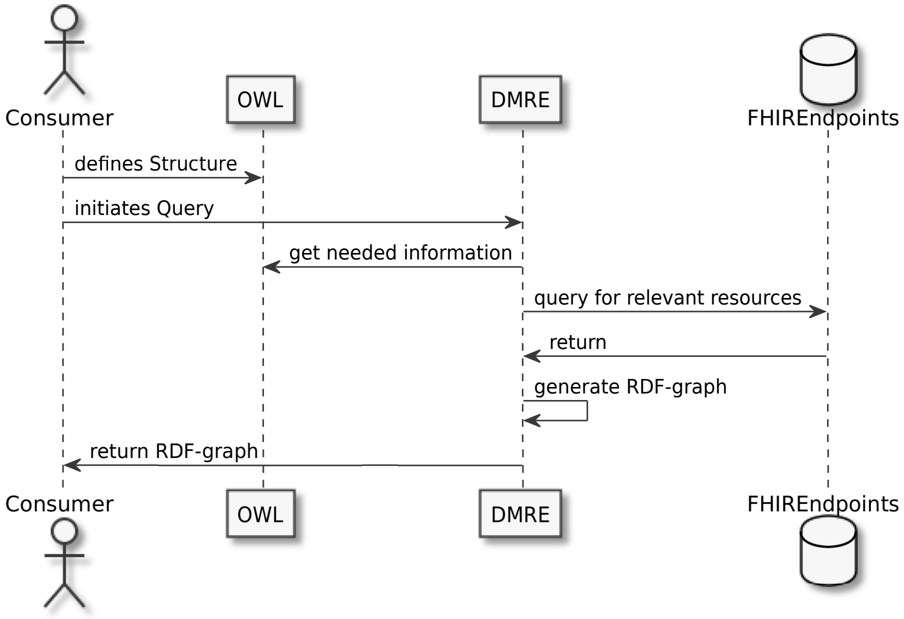


Fig. 1. Sequencediagram

For collecting all the information and making inferences about the information, we contribute with the following system architecture (Figure 2). At the top-level, there are three components: the Consumer, the Distributed Medical Rule Engine (DMRE), and the FHIR-stores.

The main component is the DMRE. The consumer initiates a query. The FHIR-stores hold the medical information. This medical information is defined by HL7-FHIR and is provided via standard-HTTP-interfaces. The DMRE applies different services to collect information from different FHIR locations and to access the ontology knowledge. The services are:

- ReadOWLAndAttachFHIR-Service
- OWL-Read-Service
- SPARQL-Service
- FHIR-Query-Service

The ReadOWLAndAttachFHIR-Service is the central coordinator, which takes care of the correct workflow execution. It is a service, which receives requests from consumers, and returns the result. In the next step, the "ReadOWL AndAttachFHIR"- Service uses the OWL-Read-Service. The OWL-Read-Service parses the predefined ontology, gets the classes, and uses a SPARQL-Service to get each defined class's properties. These classes and properties are passed to the FHIR-Query-service.

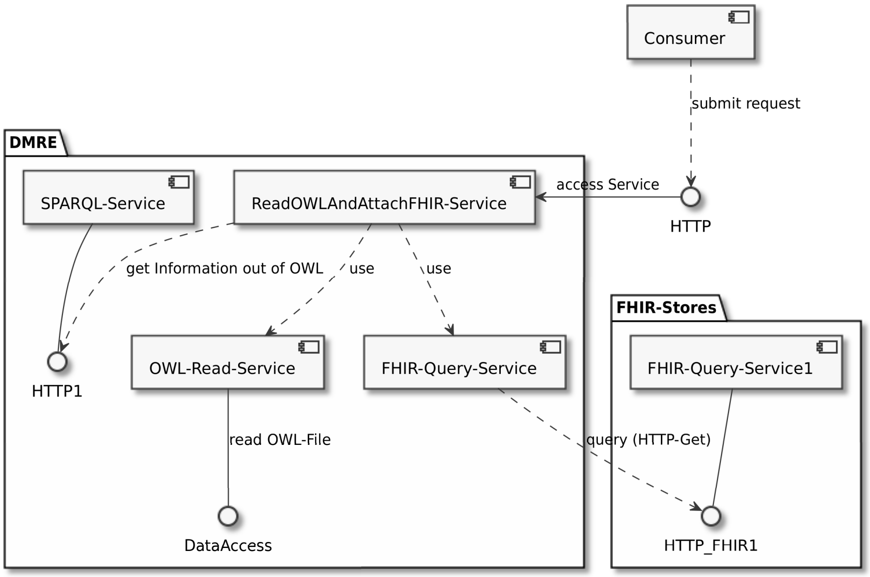


Fig. 2. Systemarchitecture

From a technical perspective, there are several essential elements. The defined ontology follows the Web Ontology Language (OWL), which is represented in RDF-syntax. The classes need to follow the HL7-defined FHIR-resources. This means a Patient-resource in FHIR needs to have the same name as the class in the ontology. The same applies to elements in the resources. As an example, a datatype-property for "unit" should be defined in the ontology by "valueQuantity.unit".

This is needed for parsing the FHIR-results for the specific values, in order to fill the RDF-graph. There is one issue: since we are using JSON-Path [11], we need to follow this specific path syntax. For example, this would apply to "name[0].family" to get the first element. Since square brackets are interpreted as blank nodes, they cannot be used within the OWL's data-property. For that reason, a modification of the brackets was needed - the definition is done using round brackets. This fact is known to the DMRE, and during processing, the brackets change to be compliant with the JSON-Path.

For reading the ontology and generating the RDF-graph, we are using the RDF4J framework for processing and handling RDF data [23]. The challenge is to map information from FHIR-stores, which is available as JSON-resources, into the resulting RDF-graph. The process is as follows: The result of the FHIR-query gets parsed, and temporarily put to an in-memory- "resources-map". This map contains the FHIR-Bundle's-resource-ID, the JSON-Path for the resource, and the value from the FHIR-resource. In a second step, the FHIR-Information from the map is placed to the RDF-graph. The basis for the RDF-graph is the defined ontology. This means we are extending the OWL with individuals out of the FHIR-server's information, but keep the defined ontology as it is. For the individuals, we create an "Internationalized Resource Identifier" (IRI) with the help of RDF4J. These individuals have an ID containing the FHIR-resource-ID (e.g., *http://hapi.fhir.org/baseR4/Observation/ 1567077/_history/1*), a relation to their class-name, and the properties. An example of a created RDF-Description is below:

```
<rdf:Description rdf:about="http://hapi.fhir.org/baseR4/
    Observation/1567077/_history/1">
   <rdf:type rdf:resource="http://dmre/Observation"/>
   <id xmlns="http://dmre/">1567077</id>
   <valueQuantity.value xmlns="http://dmre/">6.3</
       valueQuantity.value>
   <valueQuantity.unit xmlns="http://dmre/">mmol/l</
       valueQuantity.unit>
</rdf:Description>
```

Here we find the URI of the FHIR-Resource, the type of the resource, as well as the defined attributes - in this case, the ID, the unit, and the value of FHIR's Observation-valueQuantity. Finally, we store the result of the RDF-graph in a knowledge base in order to make it persistently accessible for later processing.

There also exists a proof-of-concept implementation, which is available on Github. There is also the initial ontology, as well as a resulting example provided. The code executes the process in the sequence-diagram by using the elements in the system architecture.

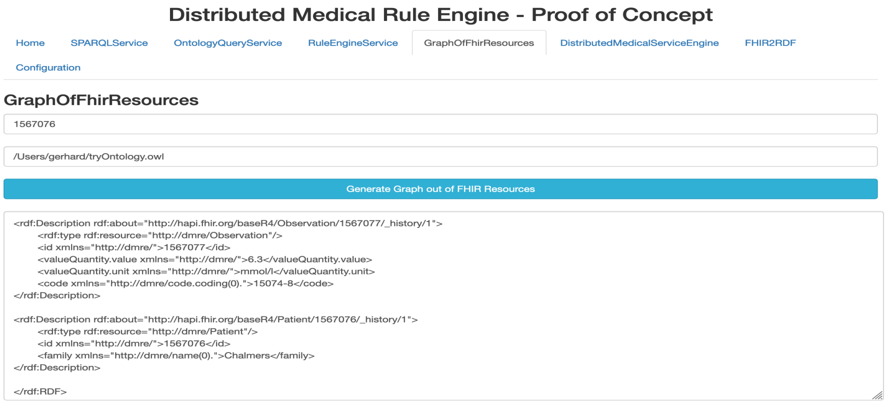
4 Example use case

A medical doctor is supposed to retrieve all available observations from a patient to get an overall picture of his current patient. There are two necessary steps: firstly, the definition of the needed dataset is given by the physician, and secondly, a manual trigger where a patient-id is sent to the DMRE-Service. For defining the needed values in advance, a person, who has knowledge about FHIR-data-structure, is needed to capture the correct values. Additionally, there is a definition needed to characterize the dependencies among the final resources. In our example, the doctor is using the FHIR-Resources "Patient", "Observation", and "Medication". In the ontology, the doctor describes the relations "a patient has a medication", and "a patient has an observation". The attributes of the Patient-class (=Patient-FHIR-Resource) are: "ID" and "name". For the Medication-class (=Medication-FHIR-Resource) it is "ID". For the Observation-class (=Observation-FHIR-Resource) the doctor likes to have the attributes "id", "value", "unit", "effectiveDateTime", and "code.coding[0].code". These prerequisites end up in an ontology provided as an ontology-file represented in XML.

Based on this ontology (which is the same for all Patients), the medical doctor can now trigger a request, having the Patient's ID. Figure 3 shows that the user

https://github.com/gkober/ReadOWLAndAttachFHIR provides the patient's ID in the web-interface and the ontology that should be applied. The result after processing is displayed in the textbox and contains the complete generated RDF-graph.

For the doctor, the complete process is transparent since the doctor only knows the patient-id, but the DMRE queries all endpoints providing information about the patient (with this patient-id). The result is not yet ready for use by a medical doctor, but it provides the technical data basis for e.g. further evaluation and further queries or for generating a graphical overview of the patient's data.


Distributed Medical Rule Engine - Proof of Concept
Home
SPARQLService
OntologyQueryService
RuleEngineService
GraphOfFhirResources
DistributedMedicalServiceEngine
FHIR2RDF
Configuration
GraphOfFhirResources
Generate Graph out of FHIR Resources

1567076

/Users/gerhard/tryOntology.owl

```
<rdf:Description rdf:about="http://hapi.fhir.org/baseR4/Observation/1567077/_history/1">
    <rdf:type rdf:resource="http://dmre/Observation"/>
    <id xmlns="http://dmre/">1567077</id>
    <valueQuantity.value xmlns="http://dmre/">6.3</valueQuantity.value>
    <valueQuantity.unit xmlns="http://dmre/">mmol/l</valueQuantity.unit>
    <code xmlns="http://dmre/code.coding(0).">15074-8</code>
</rdf:Description>

<rdf:Description rdf:about="http://hapi.fhir.org/baseR4/Patient/1567076/_history/1">
    <rdf:type rdf:resource="http://dmre/Patient"/>
    <id xmlns="http://dmre/">1567076</id>
    <family xmlns="http://dmre/name(0).">Chalmers</family>
</rdf:Description>

</rdf:RDF>
```

Fig. 3. Proof of Concept - Webview

5 Evaluation

We evaluated three crucial aspects: Firstly, the completeness of the result. Secondly, the contents of the resulting RDF-graph and as third aspect, its size. The evaluation was done using different samples of a public HAPI-FHIR-test server (http://hapi.fhir.org/baseR4/).

#	Entries FHIR-server	Entries RDF-Graph	Size-FHIR-Resources	Size-RDF-Graph
1	1	1	3 kB	4 kB
2	36	36	55 kB	28 kB
3	45	45	69 kB	34 kB
4	56	56	86 kB	41 kB
5	64	64	100 kB	46 kB

Table 2. Comparison Resource-Entries and Data-sizing

The result's completeness: we crosschecked the amount of FHIR- resources to the number of resources converted in the RDF-graph. In comparison, we found the same numbers for both (see. Table 2). This means, during conversion, no data set from the FHIR-server gets lost, and all resources belonging to a certain patient are taken into account.

Analyzing the contents of the RDF-graph was a manual and visual task. We compared the final RDF-graph with the expectations of the initial ontology and if all the given properties were translated correctly from the FHIR-resource to the graph. The visualization was done using OWLGrEd [2]. In the manual comparison of the samples and the visualization of the resulting RDF-graph, we found, the to-be-included values are correctly translated from the FHIR-resource to the RDF-graph.

With respect to the size of the resulting RDF knowledge graph: the FHIR-resources itself only needs a few kilobytes. By using the DRME, applying an ontology, and only selecting the data needed for specific questions, we were able to reduce the number of bytes. This data size reduction benefits the data transmission to the client, which needs to parse the result and display it for the end-user. As we can see in Figure 4, the sizes of the data packages in the RDF-Graph are less than the original FHIR-resources. The explanation for having a greater size in the RDF-graph in comparison to the FHIR-resource (in case of one entry) is, the additional data of the ontology-knowledge in the graph. Although there is a higher count of entries in the result, the size of the RFD-graph is still reduced because the client focuses on only a few interesting properties of an FHIR-resource.

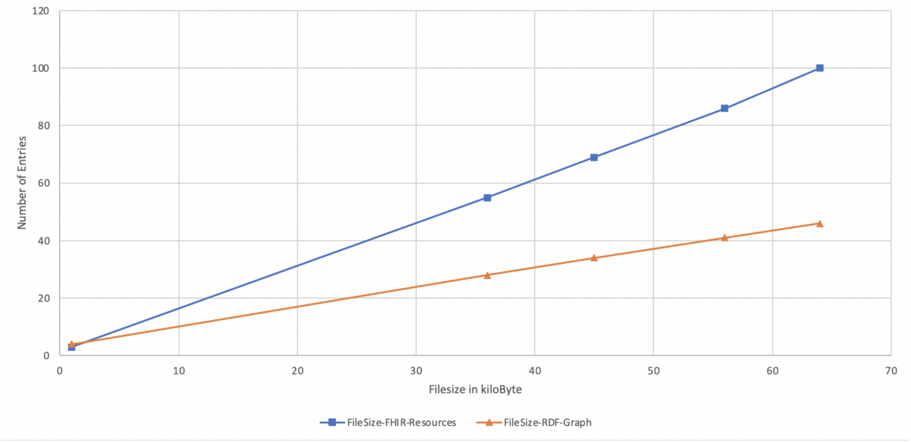


Fig. 4. Sizes

6 Results & Discussion

The solution presented in this paper semantically creates a RDF-graph by querying data values from various FHIR-resources and semantically curating and mapping these values into an ontology representation. The proof of concept implementation reads the manually generated ontology, while a consumer can trigger the RDF generation by using a ReadOWLAndAttachFHIR-Service from the Distributed Medical Rule Engine (DMRE). The ontology for the PoC contained classes for Patient, Observation, and Medication, and a subset of attributes for these classes.

We tested the approach in two scenarios: Firstly, the initiating query for evaluation is asked for only one patient having one observation. This simple task allowed us to manually check for the newly generated individuals and evaluate if they were set correctly. Secondly, we choose different patients having up to 64 observations. This also resulted in an RDF-graph with the needed instances and correctly translated medical information.

The approach is patient-centered, as we wanted to collect all information from different FHIR-stores. This might have limitations in case of other research questions, e.g. a research question could be about the whole patient-population (e.g., a particular illness). Here the ontology approach works in the same way as the patient-centered use case. However, the FHIR-queries on the FHIR-stores need alignment within the software.

Useful is the option to define which attributes of an FHIR-resource are needed. For example, defining the patient-resource without names can lead to anonymization, which is essential for research-data. Another aspect is, the ontology for the resulting graph can be aligned until a consumer has the needed data-collection as required. The limit is in the availability of the information stored by doctors, hospitals, or patients. For example, if a patient provides his pulse-rates, a researcher can get this information (if not denied by security mechanisms, like authorization).

7 Future Work and Conclusion

In this work, we proposed a semantic mapping approach that creates and curates a medical RDF-graph taking standardized FHIR-resources of daily clinical routine-processes as input and uses an ontology for semantic knowledge representation in the RDF graph. The consumer defines the information needs for the particular use-case (or research-question).

The solution can reduce and extend the amount of data according to the consumer's needs. If only some values from the FHIR-server are needed, we can provide a simplified data-structure but with the benefits of an RDF-graph. Therefore a consolidated view on the data is provided for answering research questions. If technical and legal access is granted, the patient's private information is also available in the RDF-graph. In theory, the generation-process (and the definition) allows us to form an anonymized RDF-graph. This advantage needs further investigation if it fits the needs of an anonymized data set.

Future research might apply SPARQL-queries to the resulting RDF-graph to specify the researchers' needs over the dataset. These queries could answer critical questions from researchers. Furthermore, the generated RDF-graph can be enriched with other graphs/ontologies to get more information for patients' treatment or public health questions (e.g., attaching a FOAF (Friend-of-a-friend)-Ontology).

References

1. Altamimi, A.: Secfhir: A security specification model for fast healthcare interoperability resources. International Journal of Advanced Computer Science and Applications (ijacsa) **7**(6) (2016)
2. Bārzdiņš, J., Bārzdiņš, G., Čerāns, K., Liepiņš, R., Sproģis, A.: Uml style graphical notation and editor for owl 2. In: International Conference on Business Informatics Research. pp. 102–114. Springer (2010)
3. Booth, D., Dowling, C., Fry, E., Huff, S., Mandel, J.: Rdf as a universal healthcare exchange language. SEMTECH Panel (2013)
4. Doel, T., Shakir, D.I., Pratt, R., Aertsen, M., Moggridge, J., Bellon, E., David, A.L., Deprest, J., Vercauteren, T., Ourselin, S.: Gift-cloud: A data sharing and collaboration platform for medical imaging research. computer methods and programs in biomedicine **139**, 181–190 (2017)
5. Ferencz, J., Serban, C.N.: Patient data collection methods. retrospective insights. Journal of Medical and Dental Science Research **4**(5), 49–54 (2017)
6. Datatypes - FHIR v4.0.1,
https://www.hl7.org/fhir/datatypes.html, accessed: 2021-01-16
7. Overview - FHIR v4.0.1,
https://www.hl7.org/fhir/overview.html, accessed: 2021-01-16
8. Rdf - FHIR v4.0.1,
https://www.w3.org/OWL/, accessed: 2021-01-16
9. Security - FHIR v4.0.1,
https://www.hl7.org/fhir/security.html, accessed: 2021-01-16
10. Summary - FHIR v4.0.1,
https://www.hl7.org/fhir/summary.html, accessed: 2021-01-16
11. Friesen, J.: Extracting json values with jsonpath. In: Java XML and JSON, pp. 299–322. Springer (2019)
12. Hardt, D., et al.: The oauth 2.0 authorization framework. Tech. rep., RFC 6749, October (2012)
13. Hess, D.R.: Retrospective studies and chart reviews. Respiratory care **49**(10), 1171–1174 (2004)
14. IHE International Inc.: IHE Quality , Research and Public Health (QRPH) Technical Framework Volume 1 (2011)
15. Mallappallil, M., Sabu, J., Gruessner, A., Salifu, M.: A review of big data and medical research. SAGE Open Medicine **8**, 2050312120934839 (2020)
16. Margheri, A., Masi, M., Miladi, A., Sassone, V., Rosenzweig, J.: Decentralised provenance for healthcare data. International Journal of Medical Informatics p. 104197 (2020)
17. Miller, E.: An introduction to the resource description framework. Bulletin of the American Society for Information Science and Technology **25**(1), 15–19 (1998)

18. Operation-patient-everything, https://www.hl7.org/fhir/operation-patient-everything.html/, accessed: 2021-01-16
19. OWL - Semantic Web Standards, https://www.w3.org/OWL/, accessed: 2021-01-16
20. OWL Web Ontology Language, https://www.w3.org/TR/2004/REC-owl-features-20040210/, accessed: 2021-01-16
21. Razzak, M.I., Imran, M., Xu, G.: Big data analytics for preventive medicine. Neural Computing and Applications **32**(9), 4417–4451 (2020)
22. RDF - Semantic Web Standard, https://www.w3.org/RDF/, accessed: 2021-01-16
23. Eclipse rdf4j, https://rdf4j.org/, accessed: 2021-01-16
24. Weber-Jahnke, J., Peyton, L., Topaloglou, T.: ehealth system interoperability. Information Systems Frontiers **14**(1), 1–3 (2012)
25. Weinger, M.B., Slagle, J., Jain, S., Ordonez, N.: Retrospective data collection and analytical techniques for patient safety studies. Journal of biomedical informatics **36**(1-2), 106–119 (2003)