Mapping FHIR Resources to Ontology for DDI reasoning

Raees Abbas[1], Islam Fathi Hussein Al Khaldi[1], Getinet Ayele[1] and Jan Pettersen Nytun[1]

[1]Department of Information and Communication Technology, University of Agder, Grimstad, Norway

Abstract

Fast Healthcare Interoperability Resources (FHIR) specifications are used to exchange clinical and health related information between different systems. There is unfinished on-going work to represent FHIR resources using Semantic Web technology to support semantic interoperability. This same technology would then also fit applications doing reasoning. We utilize and customize the FHIR unofficial draft ontology for doing drug-drug interactions reasoning. We give one use case of such reasoning based on family history; this kind of reasoning may extend the capabilities given by Forskrivnings- og ekspedisjonsstøtte (FEST) alone. We achieve this by setting up a FHIR server and making a FHIR client that store drug and patient information to the server; we then later retrieve some of this information, translated it into Web Ontology Language (OWL) based ontology, do drug-drug interaction (DDI) reasoning exposing potential health risks.

Keywords

FHIR Ontology, DDI reasoning, Semantic Web, SPARQL

1 INTRODUCTION

Electronic health records (EHR) capture health information about patients and their medication; standards and protocols allow such records to be shared between different stakeholders such as hospitals, labs, pharmaceutical companies, etc. One of these standards is the FHIR specifications, built on Health Level 7 (HL7) which is a widely accepted set of protocols and standards used to exchange clinical and health related information between different systems (Benson, 2016).

Some consider HL7 V2 and V3 the most relevant exchange standards in healthcare (Oemig and Snelick, 2017). Our purpose is to do reasoning about Drug-Drug Interactions (DDI) based on patient information together with available information about drugs utilizing FHIR. The reasoning can help doctors making decisions, and pharmaceutical companies in manufacturing drugs. OWL (Bechhofer, 2009) being based on description logic supports reasoning and is our chosen ontology language. Many of the existing medical ontologies are expressed in OWL, e.g., Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT) (Whetzel et al., 2011), this supports our choice.

Combining data from different sources may allow enhanced reasoning to derive useful information, e.g., detect drug-drug interactions and potential health problems, which again enables early beneficial interventions. To achieve this various medical information systems, implemented on different platforms, need to exchange healthcare data in a standardized way.

This paper describes an important step towards reaching our goal. In short, we demonstrate how a FHIR client can store drug and patient information on a FHIR server; how to retrieve this information and translated it into an OWL based ontology. How to perform drug-drug reasoning on this ontology and transfer the results of the reasoning back to the FHIR server. The ontology we use for reasoning is the FHIR unofficial draft ontology (Anthony, 2016); we customize and extend this ontology for our needs. The contribution of this paper is the demonstration of how system components (EHR, Drug information, ontology for reasoning) may be tied together with help of the FHIR standard to support DDI.

Rest of the paper is organized by different sections as follows: Section II describes the background, and literature review of related concepts and work. Section III describes the solution architecture and a detailed description of solution methods and results. In Section IV, we discuss the usability of this work and provide some options for future work.

2 MATERIALS AND METHODS

2.1 Background and State of the Art

When a patient takes two or more drugs at the same time, DDI's can occur. This interaction may create unexpected side effects. The situation becomes even more complex when the side effects of two drugs together results in new symptoms falsely indicating some potentially new disease (leading to the introduction of new drugs). In some cases, the medical history of the family of a patient may contain information that helps reasoning. One of the approaches for representing the domain of Drug-Drug Interactions (DDI) is medical ontologies which provide a vital

Paper from the 15th Scandinavian Conference on Health Informatics SHI2017, Kristiansand, Norway, 29 - 30 August.

Conference Proceedings published by Linköping University Electronic Press.

integration of knowledge base and drugs data. Medical ontologies are often defined in OWL whereas reasoners such as Fact++ and HermiT (Glimm et al., 2014) can be used to infer Drug-Drug interactions. SNOMED CT is an example of a medical ontology defined in OWL where medical concepts like "organism" and "substances" are represented as classes in a hierarchical structure. Another class like "chemical" has a subclass "Inor-ganic chemical". Properties such as "has active ingredient" representing the relations and rules between individuals which are created under each class. Reasoners apply the rules against the knowledge base and determine the action that should be carried out in case a DDI is inferred.

Another form of reasoning on OWL ontologies may be performed with help of rule languages like the Semantic Web Rule Language (SWRL) (Horrocks et al., 2004).

OWL statements may be serialized by using the Resource Description Framework (RDF) where the statements are represented as triples forming RDF graphs (W3C, 2014).

Another key technology of the Semantic Web is the SPARQL Protocol and RDF Query Language (SPARQL) (W3C, 2013), which allows complex queries to retrieve data from an OWL ontology.

The Drug-Drug Interactions Ontology (DINTO) defined in OWL describes and categorizes DDIs and all the possible mechanisms that can lead to them; it demonstrates that the combination of drug-related facts in DINTO and SWRL rules can be used to infer DDIs and their different mechanisms on a large scale (Herrero Zazo et al., 2014) (Herrero-Zazo et al., 2015). Yoshikawa et al. (2004) developed Drug Interaction Ontology (DIO) with a focus of inferencing a possible drug-drug interaction based on the molecules of the drugs. It checks all possible biomolecular interactions between drugs that may lead to unexpected side effect(s). It infers not only drug-drug interaction, but also checks the individual differences in drug response or genetic susceptibility of drugs, and optimization of balance of efficacy and safety of new drugs.

“Founded in 1987, Health Level Seven International (HL7) is a not-for-profit, ANSI-accredited standards developing organization dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports clinical practice and the management, delivery and evaluation of health services.”(HL7, 2017).

HL7 has affiliates in all continents. European HL7 affiliates comprises 21 countries including Norway. FHIR (Benson, 2016) is a set of HL7 standards introduced by HL7 as Draft Standard for Trial Use (DSTU) which includes various features from different versions of HL7 such as HL7 v2, HL7 v3 and CDA.

FHIR resources are modular components that can be combined to build a solution for many problems existing in the real world of clinics and can be used to solve administrative problems as well. According to FHIR standards, resources are the common building blocks for exchanging documents and information related to healthcare information (Benson, 2016). Figure 1 shows an example of a FHIR resource in the form of a UML Diagram; the Medication resource is primarily used for the identification and definition of a medication (FHIR-HL7, 2017). As shown in the figure 1, it includes the ingredients and the packaging for a medication; communicating a medication would come as a bundle containing an instance of this pattern in one of the supported serialization formats (e.g., JSON).

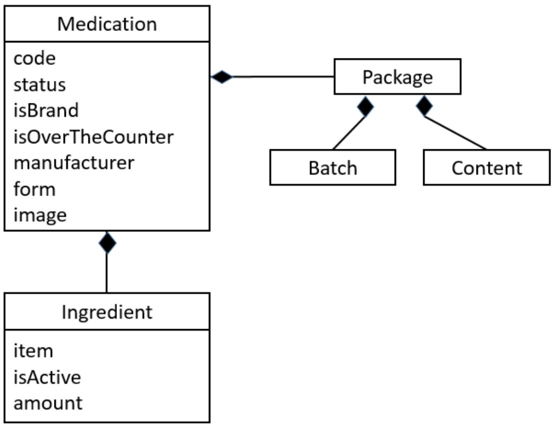

Figure 1. FHIR Medication resource (FHIR-HL7, 2017)

The main challenge for healthcare standards is the continuous need to add more fields and options to the specification which increases the cost and complexity of implementations. FHIR has defined a framework in such a way that it becomes easy enough to extend the current resources and add custom definition (Kasthurirathne et al., 2015). As FHIR mainly focuses on implementation, there are many libraries available for development with no restrictions on specification (Kasthurirathne et al., 2015).

There are also research efforts in the direction of representing FHIR data types in terms of OWL constructs which enhances semantic interoperability of FHIR resources. One key change of HL7 FHIR Release 3 is the definition of an RDF format, and how FHIR relates to Linked Data (Grieve, 2017).

An unofficial HL7 FHIR ontology draft was presented on 11th October 2016 by the ITS group (Anthony, 2016). This unofficial draft is a result of the ongoing work and its effort towards describing all the FHIR resources using OWL format. They used the same name of FHIR resources for their OWL class. Content that describe a resource is given as OWL object properties. And, some of the content is represented using OWL data properties. An OWL object property describes links going from one individual to another, e.g., the description of a specific patient (instance of class *Patient*) linked by an object property called *prescribed* to a specific drug (instance of class *Drug*). OWL data properties describe links from individuals to data values like text strings and numbers, e.g., representing name and year of birth.

There are several HAPI-FHIR FHIR RESTful servers available; HAPI FHIR is developed by the University Health Network (UHN) group. It is a Java implementation of FHIR resources, and it is an open source RESTful server which gives opportunities for researchers and academicians to use it freely and provides java libraries for all resources (UHN, 2017).
HAPI-FHIR test server has implemented all the FHIR resources and is built from numerous modules of the HAPI FHIR project. The server has uploaded sample data for all resources. It is possible to read, edit, create, update, delete and validate these resources. The output of the read operation is either in JSON or XML formats. RDF is still not defined for accessing a RESTful server.

There are several tools available for working with OWL. Protégé is an open-source ontology editor used for creating and editing OWL ontologies; it also supports reasoning. It has a plug-in architecture with a rich set of plug-ins, e.g., SPARQL, SWRL and reasoners like Fact++ and HermiT (Musen, 2015). Jena is a collection of Java libraries used for creating Semantic web and Linked data applications (McBride, 2002). It provides a framework for inferencing, storage, querying (SPARQL) and it provides a database solution, called Fuseki (Apache, 2011), for handling ontologies, e.g., through a web interface.

3 SOLUTION OVERVIEW

Figure 2 shows the architecture of our solution. We developed a customized FHIR ontology, comprised of the resources which are relevant to our drug-drug reasoning example. The part of our ontology concerning the FHIR resources coincides with ideas presented by Beredimas et al. (2015); the part concerning DDI reasoning is partly inspired by (Herrero-Zazo et al., 2015) all together with our own contribution.

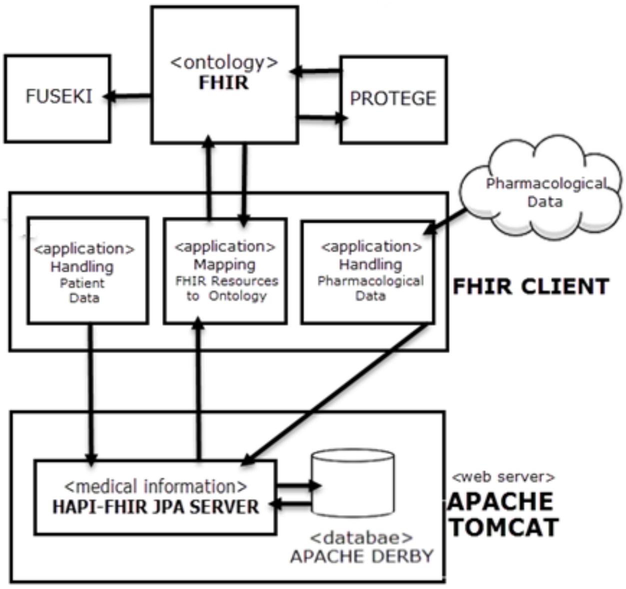

Figure 2. Architecture overview

In the ontology, we need to represent the information given by FHIR resources, and we do this in a one-to-one style. We do not represent all the FHIR resources in detail, but using a similar approach, we can extend the ontology for all the resources.
We built a FHIR client consisting of three Java modules, which can easily be made as three separate applications:

- one for inserting pharmacological data into the FHIR server.
- one for inserting patient data into the FHIR server.
- one for retrieving pharmacological and patient data; mapping of these FHIR resource instances to the corresponding FHIR ontology classes.

The server is locally hosted HAPI-FHIR server. We then used Protégé for reasoning and display of the results; to do this in the Java application is also straight forward. Also a SPARQL endpoint, e.g., Fuseki, as shown in the figure, can be used for storing the ontology.

3.1 FEST to FHIR

FEST is a database used for prescription in Norway, and have a huge collection of drug descriptions. Pharmaceutical data that we have used for the representation of drug-drug interactions is taken from FEST (i.e., from the distributed FEST xml file). For the purpose of demonstrating a drug-drug reasoning, the FEST data is handled manually by identifying relevant drugs and importing them to the FHIR server (Legemiddelvek, 2016). A different project, in the form of master thesis with title “Ontological modelling of FEST with support for DDI reasoning” (Abbas, 2017).

3.2 FHIR server

Of the FHIR resources, we mainly used Medication, Patient, Practitioner, DetectedIssues, and MedicationStatement re-sources. In the FHIR server, drugs are represented using the Medication resource. We identified 10 different medicines from FEST database where some may have potential drug interaction with each other, and created FHIR resource in-stances. Patient and Practitioner instances were created using the corresponding FHIR resources. When it comes to the identification of a medicine, two different coding systems were used: Anatomical Therapeutic Chemical (ATC) Classi-fication System and SNOMED CT.

```
<code>
    <coding>
        <system value="http://snomed.info/sct"/>
        <code value="116078005"/>
        <display value="Efavirenz (product)"/>
    </coding>
    <coding>
        <system value="https://www.whocc.no"/>
        <code value="J05AG03"/>
        <display value="efavirenz"/>
    </coding>
</code>
```

Figure 3. SNOMED CT and ATC codes example

Above is an example of a medicine, which is stored in FHIR server, identified by both using SNOMED CT and Anatomical Therapeutic Chemical (ATC) codes.

In figure 3 above, *system* value specifies the coding system that is being used and corresponding *code* displays the actual code that is used to describe the particular medicine. While *display* value describes the name of the medicine which is also specified by the coding system being used. *DetectedIssue* resource, for our use case, is used only to represent drug-drug interactions. Similar to the resource instances of *Medication*, in the FHIR server, we stored resource instances of *Practitioner, DetectedIssues, FamilyMemberHistory*. In addition, *MedicationStatement* that is used to record the drugs being used by a patient.

3.3 From FHIR to OWL

Pharmaceutical and patient data that was stored in the local FHIR server were retrieved and mapped to the FHIR ontology using JENA libraries. A part of the customized FHIR ontolo-gy has the class structure shown in figure 4.

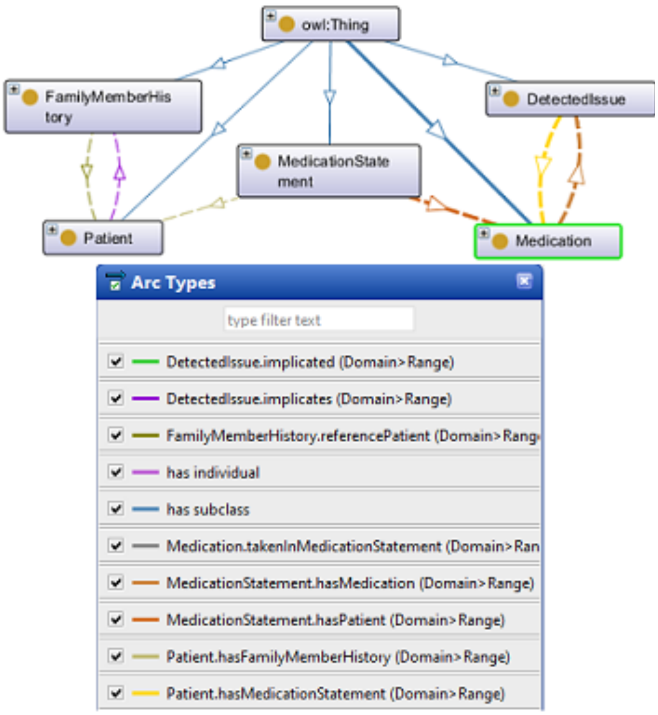

Figure 4. UML diagram of the FHIR resource Medication

We choose the approach to represent the FHIR resources (or more specifically FHIR resource types) as classes (e.g., Figure 4 shows FHIR resource Medication in a UML diagram) using same naming scheme, e.g., Medication is called Medication. The contents that describe a particular resource are represented as OWL object properties. Some of the contents are also represented using OWL data properties. Figure 5 below shows some of the data properties used to describe the details of the Medication resource.

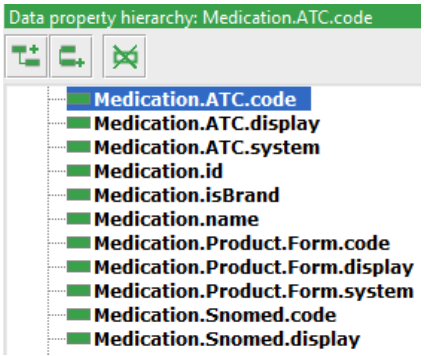

Figure 5. Data properties of the FHIR resource Medication

In this way, we mapped the details of all the selected re-sources from FHIR server to our ontology. Figure 6 shows a detailed description of the object properties along with their purpose.

When it comes to the mapping of FHIR resource instances (e.g., description of an actual drug), we accessed the resource instances from the FHIR server, and added it to the ontology. E.g., a specific drug is then seen as an individual of class Medication and inserted as such an individual into the ontology. Figure 7 shows the example of mapping instance of Medication resource named "Efavirenz".

Property	Domain	Range	Explanation
DetectedIssue.author	DetectedIssue	Practitioner	Used to describe a practitioner who worked with the particular detected issue.
DetectedIssue.detects	Practitioner	DetectedIssue	Inverse of *DetectedIssue.author*
MedicationStatement. medicationReference	Medication Statement	Medication	Used to describe a referenced medicine.
Medication. medicationStatementReference	Medication	MedicationStatement	Inverse of *MedicationStatement. medicationReference*
MedicationStatement. patient	Medication Statement	Patient	Used to describe referenced patient.
Patient. medicationStatement	Patient	MedicationStatement	Inverse of *MedicationStatement. patient*
DetectedIssue.implicated	DetectedIssue	Medication	Used to reference a medication which leads to an issue.
DetectedIssue.implicates	Medication	DetectedIssue	Inverse of DetectedIssue.implicated
FamilyMemberHistory. patientReference	FamilyMember History	Patient	Used to refer to the relative patient
FamilyMemberHistory. relationship	Patient	FamilyMember History	Inverse of FamilyMemberHistory. patientReference

Object Property description

Figure 6. Object properties

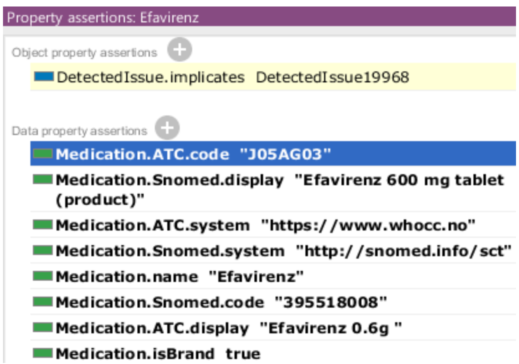

Figure 7. Medication instance mapping

As the medication is identified using two coding schemes, Display value of corresponding drug is mapped to Medication.Snomed.display property in the ontology and takes the value as “Efavirenz 600 mg tablet“. Similarly, code value from the FHIR server, in case of SNOMED, is mapped to Medication.Snomed.code in the ontology. The same mapping procedure is used for ATC code mapping.

3.4 DDI Reasoning

SPARQL queries are used to retrieve and manipulate data from the ontology, e.g., we can check whether two drugs have an interaction by the following SPARQL query:

```
SELECT ?X
  WHERE { ?X  :DetectedIssue.implicated :Fluvoxamine.
          ?X  :DetectedIssue.implicated  :Rabeprazole. }
```

X
DetectedIssue19969

Figure 8. SPARQL query result

The result shown in Figure 8 above can also be verified by running reasoner in the Protégé. Similarly, if we have a complete database from the FHIR server, we can use SPARQL queries on that existing data to generate specific results. Explicit as well as implicit results can be generated using SPARQL queries on the FHIR data, which was the purpose of this paper. One example of generating implicit results from the data is presented below.

We used the SPARQL CONSTRUCT query to generate implicit triples about unknown DDI’s by examining the existing data. For example, suppose that father of Ole Nordmann, having a heart disease, was prescribed a combination of drugs A and B; and his brother, having similar disease, was also prescribed same combination of drugs. Their statement history exhibit that both of them suffered from some side effects, possibly a DDI. But this DDI is not explicitly stated in the database nor the drugs A and B were supposed to exhibit a reaction. A hypothetical diagnose can be linked to such a situation that genes of the concerned persons may have a problem with this combination of Drugs. We can check this whole scenario by constructing a SPARQL query, and in response, Ole Nordmann can be prevented from taking the same combination as he is potentially inheriting the same genes. A possible warning can be generated for the practitioner responsible for treating Ole.

```
CONSTRUCT  {
?X  :hypotheticalDiagnose  :UnknownDDI }
WHERE   {
?X  :Patient.hasFamilyMemberHistory  ?a.
?a  :FamilyMemberHistory.relationshipCodingCode
                                          "father";
    :prescribed   :DrugA;
    :prescribed   :DrugB;
    :sideEffects  :UnknownDDI.
?X  :Patient.hasFamilyMemberHistory  ?b;
?b  :FamilyMemberHistory.relationshipCodingCode
                                          "brother";
    :prescribed :DrugA;
    :prescribed :DrugB;
    :sideEffects  :UnknownDDI.
 }
```

Subject	Predicate	Object
Patient9952	owl:hypotheticalDiagnose	UnknownDDI

Figure 9. CONSTRUCT query result

The implicit triple generated by the query above, can be inserted and stored in the ontology. The result says that *Patient9952* individual, which is Ole Nordmann, may possibly suffer from the same DDI as his family members, thus the practitioner should make further observations regarding prescription.

4 DISCUSSION AND CONCLUSION

Several clinical tools provide comprehensive lists of DDIs, often they lack the supporting scientific evidences and different tools may not give consistent results (Tari et al., 2010). One of our goals is to find out the implicit drug-drug interactions that are not explicitly stated or that are inconsistent in the currently available drug catalogs. Such reasoning would typically involve several ontologies and data sources; in this context standard protocols needs to be applied and our work success-fully demonstrate one way of doing this with help of FHIR.

We have also demonstrated some beneficial DDI reasoning based on family history. However, the reasoning potential seems huge and should be further investigated.

Our task may be simplified in the future since using RDF with the REST API is on the TODO list of HL7 (Anthony, 2016). They have already specified how each resource can be represented as a set of RDF triples represented using the Turtle syntax. Our work fits well with this format. When FHIR servers give the REST API support we can easily directly insert the RDF Graphs into our ontology that we use for reasoning.

5 REFERENCES

[1] Abbas, R. 2017, *Ontological Modelling of FEST with support for DDI Reasoning.* MSc Information and communication technology, University of Agder, Norway.

[2] Anthony, M., Prud'hommeaux E. 2016, *FHIR OWL Ontology: Unofficial Draft* [Online]. Available: http://w3c.github.io/hcls-fhir-rdf/spec/ontology.html [Accessed May 4 2017].

[3] Apache. 2011, *Fuseki: serving RDF data over HTTP* [Online]. Available: https://jena.apache.org/documentation/serving_data/ [Accessed May 5 2017].

[4] Bechhofer, S. 2009, OWL: Web ontology language. *Encyclopedia of Database Systems.* Springer.

[5] Benson, T. 2016, *Principles of Health Interoperability*, Springer.

[6] Beredimas, N., Kilintzis, V., Chouvarda, I. & Maglaveras, N. 2015, A reusable ontology for primitive and complex HL7 FHIR data types. Engineering in Medicine and Biology Society (EMBC), 37th Annual International Conference of the IEEE, 2015. IEEE, 2547-2550.

[7] FHIR-HL7. 2017, *Resource Medication - Content* [Online]. Available: https://www.hl7.org/fhir/medication.html [Accessed May 5 2017].

[8] Glimm, B., Horrocks, I., Motik, B., Stoilos, G. & Wang, Z. 2014, HermiT: an OWL 2 reasoner. *Journal of Automated Reasoning,* 53**,** 245-269.

[9] GRIEVE, G. 2017, *FHIR Release 3 Posted* [Online]. Available: https://onfhir.hl7.org/2017/03/22/fhir-release-3-posted/ [Accessed May 5 2017].

[10] Herrero Zazo, M., Hastings, J., Segura BedmarR, I., Croset, S., Martinez, P. & Steinbeck, C. 2015-04-23. An ontology for drug-drug interactions. International Workshop on Semantic Web Applications and Tools for Life Sciences (SWAT4LS), 2013 December, 11-12 2014 Edinburgh, UK. CEUR, 1-15.

[11] Herrero-Zazo, M., Segura-Bedmar, I., Hastings, J. & Martinez, P. 2015, DINTO: using OWL ontologies and SWRL rules to infer drug–drug interactions and their mechanisms. *Journal of chemical information and modeling,* 55**,** 1698-1707.

[12] HL7. 2017. *About HL7 International* [Online]. Available: http://www.hl7.org/ [Accessed May 5 2017].

[13] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., GrosofF, B. & Dean, M. 2004. SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission,* 21**,** 79.

[14] Kasthurirathne, S. N., Mamlin, B., Kumara, H., Grieve, G. & Biondich, P. 2015, Enabling better interoperability for healthcare: lessons in developing a standards based application programing interface for electronic medical record systems. *Journal of medical systems,* 39**,** 182.

[15] Legemiddelverk, S. 2016, *Fest Implementation Guidlines* [Online]. Available: https://legemiddelverket.no/Documents/Andre temaer/FEST/Hvordan bruke FEST/Implementation guide FEST v2.2.pdf [Accessed May 5 2017].

[16] Mcbride, B. 2002, Jena: A semantic web toolkit. *IEEE Internet computing,* 6**,** 55-59.

[17] Musen, M. A. 2015, The Protégé project: A look back and a look forward. *AI matters,* 1**,** 4-12.

[18] Oemig, F. & Snelick, R. 2017, *Healthcare Interoperability Standards Compliance Handbook: Conformance and Testing of Healthcare Data Exchange Standards*, Springer.

[19] Tari, L., Anwar, S., Liang, S., Cai, J. & Baral, C. 2010, Discovering drug–drug interactions: a text-mining and reasoning approach based on properties of drug metabolism. *Bioinformatics,* 26**,** i547-i553.

[20] UHN. 2017, *HAPI-FHIR. FHIR made simple.* [Online]. Available: http://jamesagnew.github.io/hapi-fhir/ [Accessed May 5 2017].

[21] W3C. 2013, *SPARQL query language for RDF* [Online]. Available: https://www.w3.org/TR/rdf-sparql-query/ [Accessed May 4 2017].

[22] W3C. 2014, *RDF 1.1 concepts and abstract syntax* [Online]. Available: https://www.w3.org/TR/rdf11-concepts/ [Accessed May 5 2017].

[23] Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T. & Musen, M. A. 2011, BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research,* 39**,** W541-W545.

[24] Yoshikawa, S., Satou, K. & Konagaya, A. 2004, Drug interaction ontology (DIO) for inferences of possible drug-drug interactions. *Medinfo,* 11**,** 454-458.

6 ADDRESS FOR CORRESPONDENCE

Jan Pettersen Nytun, Department of ICT, University of Agder, Norway, jan.p.nytun@uia.no, address: Postboks 509, 4898 Grimstad, Norway