# Healthcare Data Validation and Conformance Testing Approach Using Rule-Based Reasoning

Hira Jawaid[1], Khalid Latif[1], Hamid Mukhtar[1], Farooq Ahmad[2], and Syed Ali Raza[1]

[1]School of Electrical Engineering and Computer Science, NUST, Islamabad, Pakistan
[2]Department of Computer Science, College of Computer Sciences and Information Technology (CCSIT), King Faisal University, Alahssa 31982, Kingdom of Saudi Arabia
{hira.jawaid,khalid.latif,hamid.mukhtar,ali.raza}seecs.nust.edu.pk
{hfahmad}@kfu.edu.sa

**Abstract.** HL7 community is profoundly involved in the development of standards in order to exchange, share and retrieve health related information. FHIR is an emerging standard of HL7 that encourages use of JSON as a data serialization approach. Validation of healthcare data is crucial task because errors in data can result serious consequences. Currently there does not exist a validator that can validate and conform data as per FHIR. This paper presents an approach to validate healthcare data embodied in JSON documents and to test its conformance with FHIR standard. We first developed Description Logic(DL) based schema of the FHIR. JSON data is then translated to RDF and we apply rule-based reasoning to validate data. As verified by results,design of the validation algorithm ensures that validation step is performed in sub seconds and overall system remains efficient. Further, the rule based reasoning provides aid in identifying incompatibilities in data which should be fixed to bridge the gaps in achieving interoperability.

**Keywords:** HL7, FHIR, Ontology, Validation, Conformance, Reasoning.

## 1 Introduction and Background

Validation is necessary to ensure that data is in accordance with a standard and can be processed without causing any erroneous implication [1]. According to Shanks et al [2], after constructing a conceptual model, it should be validated with requirements of stakeholders. ISO/IEC specify that conformance testing is attainment of a service [3]. Javascript Object Notation (JSON) is serialization format in the form of attribute-value pairs [5]. Different approaches are used for JSON validation. JSON Lint [6],is a reformatter and is an inline JSON validator. Logic reasoner is used to infer consequences from set of axioms. These inference rules are specified by means of Web ontology language (OWL). Reasoning can be performed on large ontologies using DLs [7]. Health Level 7 (HL7), is involved in development of standards for exchange of medical information. Fast Healthcare Interoperability Resources (FHIR) is an HL7 content standard [8]. It supports JSON based content payloads. JSON-LD is JSON based serialization format and is acronym for JSON for Linked Data. Currently there is no validator that can validate FHIR data. FHIR specifications are available in JSON and XML. For JSON, no language schema exists. This paper presents an approach for validation of FHIR data in JSON format.Semantic model of FHIR's resources is developed in OWL. In order to validate FHIR JSON against semantic model, JSON is converted into JSON-LD. Afterwards rule-based reasoning is applied to validate the data against standard data model, FHIR.

Rest of the paper is organized as follows: Proposed framework is explained in Section 2. Details of FHIR Semantic Structure is described in Section 3. In Section 4 we have described validation approaches. Evaluation strategy is explained in Section 5 and finally we conclude our work in Section 6.

## 2 Proposed Framework

For validation of FHIR JSON against schema of FHIR, we have proposed an architecture of validator that validates FHIR JSON data.The validation process is depicted in Figure 1. The validator takes FHIR JSON as an input and

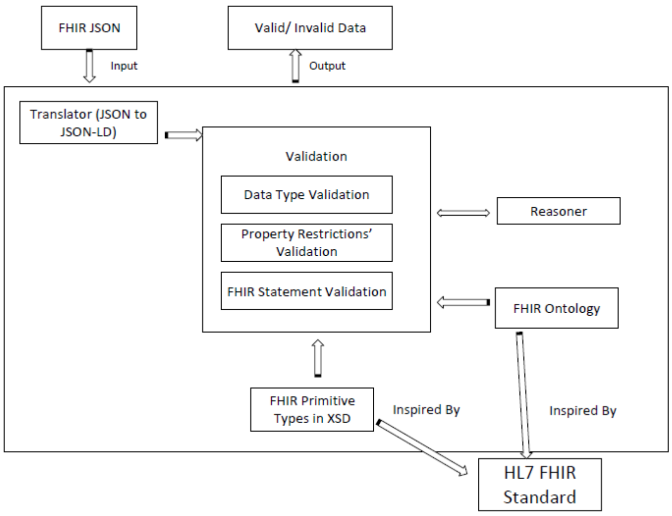


**Fig. 1.** Validation Process for FHIR JOSN

translates FHIR JSON into JSON-LD that is validated by using FHIR ontology, FHIR primitive types and resoner. The validator gives response in the form of valid or invalid data. An example of FHIR JSON is shown in Figure 2 and its translated JSON-LD is shown in Figure 3.

```
{
“name” :
“Organization”,
“publisher” : “FHIR
Project”,
“status” : “draft”,
“date” : “2013-11-26”,
}
```

**Fig. 2.** FHIR JSON

```
    "Profile.mapping" : [ "_:t2", "_:t9" ],
    "http://hl7.org/fhir#Profile.name" : {
      "@type" : "http://hl7.org/fhir#string",
      "@value" : "organization"
    },

"http://hl7.org/fhir#Profile.publisher" : {
      "@type" : "http://hl7.org/fhir#string",
      "@value" : "FHIR Project"
    },

    "http://hl7.org/fhir#Profile.status" : {
      "@type" : "http://hl7.org/fhir#code",
      "@value" : "draft"
                            },

    "@id" : "_:t10",
    "http://hl7.org/fhir#Profile.date" : {
      "@type" : "http://hl7.org/fhir#dateTime",
      "@value" : "2013-11-26"
    },
```

**Fig. 3.** JSON-LD

## 3 FHIR Schema in OWL

Ontology provide aid in inference through axioms. FHIRs semantic structure (ontology) is developed by using DL because DL provides aid in inferring. A FHIR resource of RelatedPerson is shown in Figure 4.

FHIR contains two kinds of data types. These include primitive and complex types. Primitive data types are predefined types of data. Some examples of primitive data types include date, dateTime, instant, oid, uuid. In ontology it is modelled as follows:

```
<xs:simpleTypename="oid">
<xs:restrictionbase ="xs:anyURI">
<xs:patternvalue="urn:oid:(0|[1-9][0-9]*)(\(0|[1-9][0-9]*))*"/>
<xs:minLengthvalue ="1">
</xs:restriction>
</xs:simpleType>
```

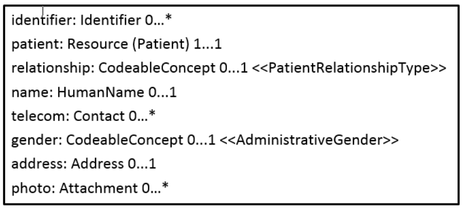


**Fig. 4.** Practitioner Resource in FHIR

This is an example of string data type oid. Several complex types are also used in FHIR. As Identifier in Figure 4 is complex type. Quantity, a complex type, is shown in Figure 5. If we consider Figure 4 then in this figure, property identifier

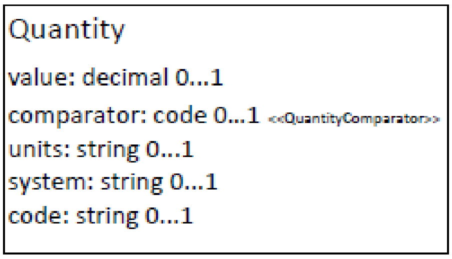


**Fig. 5.** Complex Data Type Example

contains cardinality of 0 to many.These restrictions are modelled in the following manner in ontology

1. `fhir:Practitioner.identifier rdf:type owl:ObjectProperty;`
   `rdfs:domain fhir:Practitioner ;`
   `rdfs:range fhir:Identifier.`

Ontology is developed by following modular approach where all resources are implemented as sub-classes of "Resource" and all the dependent resources are implemented as sub-classes of "DependentResource" followed by name of Resource.

## 4 Data Validation

Data is passed into validator that results in valid or invalid data. After conversion of JSON into RDF it is validated against FHIR schema. Rule based reasoning is performed to validate data as per ontology. Initially data type validation is performed that verifies individual characters are consistent with expected characters of data types that are defined in schema. For example, for instant, valid value will be 2013-04-03T15:30:10+01:00. All other values as 2013-04-03T15:30 will be invalid. After data types validation, cardinality validation is performed that checks whether data values are according to cardinalities defined in FHIR schema. If cardinality is 1 to 1 then exactly one value should be there. We have applied reasoner that checks different types of validity scenarios according to inference rules that are defined in FHIR ontology. Some of these rules are defined below:

1. `(patient:Patient(1 to 1) )`
   `∃x∃y(RelatedPerson(x)⇒ Patient(y)∧(x=y)).`
2. `name:HumanName (0 to 1)`
   `∀x (RelatedPerson(x)∧ Name(x))⇒ Name(x).`

Finally FHIR statement validation checks that all the attributes of data are as per FHIR standard. Conformance testing is performed by us in order to find out whether system validates the correct data as per FHIR standard.

## 5 Evaluation and Results

Evaluation of the system is performed on the basis of parameter that shows all valid documents are declared as valid and only valid documents should be declared as valid. Validation and conformance is performed on 266 examples of FHIR documents, out of which 221 are correct and are available on FHIR website. These examples vary in complexity from low to medium and high. The complexity is calculated on the basis of number of attribute-value pairs. Examples having more number of attribute-value pairs are considered to be more complex. Conformance and validation for 45 examples is also performed by introducing errors in valid FHIR documents. These errors include addition of extra property and data against cardinality. The evaluation is performed on a regular desktop that is Core i5, 2.53GHz, 4 GB memory, 64 bit Windows OS, Hard Disk 300 GB.

FHIR JSON is passed on to validator where Pellet reasoner is used to infer over JSON data by using FHIR schema. For any resource, that is going to be validated, its time is calculated by executing it 10 times and then average time is calculated. Validation time is shown in Table 1. Efficiency of system was

| **FHIR Resource** | **Complexity Level** | **Validation Time (sec)** |
|---|---|---|
| Patient | Medium | 126.8 |
| Organization-Profile | High | 135 |
| Location | High | 133 |

**Table 1.** Validation Time

measured by calculating response time using J-Meter. Out of 221 examples, 49 FHIR resources of high, medium and low complexity were selected and response time was calculated. Initially system was hit by 10 users and number of users are kept on increasing till 40,000. Figure 6 shows response time.

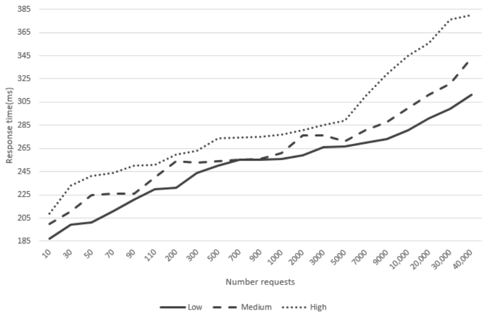


**Fig. 6.** Response Time

## 6 Conclusion

This paper presents a validation approach that is feasible for healthcare data to test its conformance as per standard. Proposed validator enables healthcare data to be validated against FHIR. Our work aims to provide accurate and consistent data among healthcare entities. It takes JSON and test its conformance with FHIR. Efficiency of this system is evaluated on the basis of validation time and response time of multiple requests that access the system simultaneously.

## 7 The References Section

## References


1. HIMDataValidation `http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1\_049164.hcsp?dDocName=bok1\_049164`
2. Graeme Shanks, Elizabeth Tansley, and Ron Weber. Using ontology to validate conceptual models. Communications of the ACM, 46(10):8589, 2003.
3. Terence P Rout. ISO/IEC 15504 evolution to an international standard. Software Process: Improvement and Practice, 8(1):2740, 2003.
4. Lynne Rosenthal, Mark Skall, and Lisa Carnahan. White paper: Conformance testing and certification framework. 2001.
5. The application/json media type for javascript object notation (json). `http://www.ietf.org/rfc/rfc4627.txt?number=4627/,2006`.
6. JSON lint. `http://zaach.github.io/jsonlint//`.
7. Anni-Yasmin Turhan. Description logic reasoning for semantic web ontologies. In Proceedings of the International Conference on Web Intelligence, Mining and Semantics, page 6. ACM, 2011.
8. FHIR standard. `http://www.hl7.org/implement/standards/fhir/overview.html/,2014`. [Online; accessed 30-September-2014].