

# Semi-Automated Approach to Populate any FHIR Server via POST Method

Abdul Mateen RAJPUT <sup>a,1</sup>

<sup>a</sup>Köln University Hospital, Köln, Germany

**Abstract.** FHIR (Fast Healthcare Interoperability Resources) is a specification for exchanging healthcare data electronically. We provide a relatively easy way to populate any FHIR server by using a workflow. A dataset of 25 FHIR JSON files with resource type Bundles, synthetically generated by using Synthea, has been tested for the population of the Vonk Server. The described approach facilitates population of any FHIR server with a KNIME workflow using POST method.

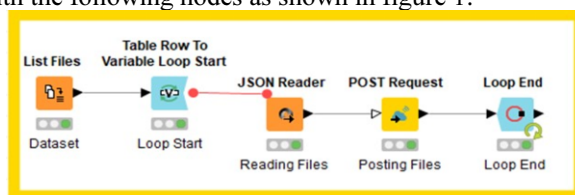
**Keywords.** FHIR, KNIME, ETL Process, RESTful API Methods

## 1. Introduction

FHIR (Fast Healthcare Interoperability Resources) is a modern specification for exchanging healthcare data electronically [1]. In recent years, a lot of development has been done in FHIR related technologies as it is considered as the foundation of modern healthcare systems. Transitioning the patient data to FHIR would make the data able to connect various applications and increases interoperability significantly [2]. Even though there has been significant progress made towards developments of FHIR related technologies, not a lot has been done to improve usability and automation. Common tasks such as data import and export require technical know-how. In current work, we provide a relatively easy way to populate any FHIR server by using a workflow. The workflow was created using KNIME (Konstanz Information Miner) [3].

## 2. Methods

Our setup was Vonk FHIR Server version 3.2.0 running on a local Windows 10 machine. KNIME version was 3.7.2. A workflow has been developed to POST FHIR JSON files on the Server with the following nodes as shown in figure 1:



**Figure 1.** KNIME workflow shows the data processing by different nodes.

<sup>1</sup> Corresponding Author, Abdul-Mateen Rajput,; E-mail: Abdul.mateen@uni-koeln.de.

- List files: Create a list of all files in a given directory, irrespective of filetype.
- Table row to Variable Loop Start: It converts rows coming from the output of the previous node to variables; makes it easier to run through a loop over them.
- JSON Reader: It is a JSON file reader and it parses JSON files.
- POST Request: This node uses a POST method to a given server with various configurations such as Time Interval etc.
- Loop End: Loop End runs a loop over the total number of rows, which are indexed by "Table Row to Variable Loop Start" and ends the loop.

The workflow starts from left, by listing all FHIR JSON files in a directory, converting individual list items into variables and beginning the loop, parsing JSON files one by one, Posting them on the FHIR Server, and looping over to the next record until all the files are processed.

### 3. Results

A dataset of 25 FHIR JSON files with resource type Bundles, synthetically generated by using Synthea, has been tested for the population of the Vonk Server. The dataset had patient-specific data of different FHIR resources including Patient, Condition, Encounter, Observation, etc. During the KNIME workflow execution, some errors were encountered as the Vonk Server was not responsive with the default Time Interval of 5 seconds but after increasing the Time Interval to 15 seconds, the POST request node posted the data successfully. The successful posting returns a response code of "200". Only one file out of 25 files was failed to create a record on the server. The same dataset was used to populate the server by using VonkLoader but encountered significantly more "InternalServerError" errors compared to the KNIME workflow. Since there was no possibility to increase the response Time Interval in VonkLoader, these failed attempts could not be prevented.

### 4. Discussion and Conclusion

The approach described here facilitates population of any FHIR server with a KNIME workflow using POST method. The performance of the approach has been tested in comparison to a popular FHIR server loader, VonkLoader, and it was found that KNIME offers more customization. Because of the various features available in KNIME, one can transform Text files, XMLs, and other file formats to JSON before posting them to a FHIR Server thus the workflow provides an important step in overall ETL processes and populating any FHIR server using POST method.

*Acknowledgment:* The project is funded by the German Federal Ministry of Education and Research (BMBF, grand id: 01ZZ1802U).

### References

- [1] Overview - FHIR v4.0.1, <https://www.hl7.org/fhir/overview.html> (accessed 4 March 2021).
- [2] Mandel JC, Kreda DA, Mandl KD, et al. SMART on FHIR: a standards-based, interoperable apps platform for electronic health records. *J Am Med Inform Assoc* 2016; 23: 899–908.
- [3] KNIME | Open for Innovation, <https://www.knime.com/> (accessed 4 March 2020).