

Perspectives and Obstacles for Transforming Terminologies into FHIR CodeSystems Exemplified by Alpha-ID

Abdul-Mateen RAJPUT^{a,1} and Cora DRENKHAHN^{b,c}

^aKöln University Hospital, Köln, Germany

^bIT Center for Clinical Research (ITCR-L), University of Lübeck, Germany

^cInstitute of Medical Informatics (IMI), University of Lübeck, Germany

Abstract. The terminology services, defined as part of the emerging FHIR standard, yield a promising approach to finally achieve a common handling of coding systems needed for semantic interoperability. As a precondition, legacy terminology data must be transformed into FHIR-compatible resources whereby varying source formats make a manual case-by-case solution impracticable. In this work, the practicability of using CSIRO's Ontoserver and the related Snapper tool as support of the transformation process were evaluated by applying them to the German Alpha-ID terminology.

Keywords. ICD-10-GM, Alpha-ID, Semantic Interoperability, HiGHmed, FHIR Code System, Ontoserver, Terminology Server

1. Introduction

FHIR (Fast Healthcare Interoperability Resources) is an emerging open standard in healthcare. Considered the “HTML” of healthcare [1], it facilitates extensive data modeling and data exchange, irrespective of the necessity of a common Electronic Health Record (EHR) system. To enable semantic interoperability, FHIR provides a common set of terminology services based on the CodeSystem, ValueSet, and ConceptMap resources which are defined and accessible in a standardized way [1]. A code system defines the concepts and their key properties, to represent an ontology, terminology, or enumeration [2] in a unified way leading to a significant increase in their usability.

Ontoserver [3] is a syndicated clinical terminology server, which is based on the FHIR standard. It has been developed by the Australian E-Health Research Council at CSIRO to overcome some of the challenges that have hindered the adoption of standardized clinical terminologies, e.g. by providing out-of-the-box support for SNOMED CT [4] and LOINC [5]. The combination of standardized FHIR terminology services with a compatible and powerful terminology server yields the potential to finally overcome the sparse usage of medical terminologies that is due to idiosyncratic formats and isolated solutions [6, 7].

¹ Abdul-Mateen RAJPUT, Köln University Hospital, Köln, Germany, Kerpener Str. 62, 50937 Köln, Germany ; E-mail: Abdul.mateen@uni-koeln.de.

However, not all clinical terminologies are already transformed into FHIR compatible format thus not readily available. Because of this limitation, we aim to examine a generic solution to transform terminologies into FHIR CodeSystems. To reveal capabilities and difficulties in the process, the transformation is exemplarily conducted for the German Alpha-ID terminology [8] that is not yet available as a FHIR resource. After reviewing existing literature, any similar effort to deal with the challenges was not found.

2. Methods

Alpha-ID is a terminology based on the Alphabetical Index of ICD-10-GM, the German modification of ICD-10-WHO [9]. It provides a unique, immutable and semantic-free identifier for each of its entries which further differentiate the diagnosis classified below a single ICD-10 code. For example, there are 42 entries for the ICD-10-GM code “K46.0” in the Alpha-ID terminology. Each one of them is similar but has a unique Alpha-ID code (see Table 1).

Table 1. The same ICD-10-GM code is used for slightly different concepts, while Alpha-ID assigns a different code for each concept.

Alpha-ID Code	ICD-10-GM Code	Associated Text
I5635	K46.0	Eingeklemmter Bruch
I5636	K46.0	Hernia incarcerata
I81612	K46.0	Hernie mit Magenstrangulation

The Alpha-ID text file version 2021 was downloaded from the DIMDI website [10]. It contains 84746 entries. The default separator used in the file is a pipe “|” sign as semicolons are used in the “Associated Text” column. The pipe sign was replaced with a “Tab” to convert the file into tab-separated values (TSV). To make it understandable for the international scientific community, we translated the German column labels into English, according to the following table:

Table 2. The German labels of Alpha-ID tables with the English translations and the FHIR CodeSystem properties assigned to each label in the terminology.

German	English	CodeSystem item
Gültigkeit	Validity	<i>validity</i>
Stabile Identifikationsnummer mit Präfix I	ID Number with Prefix (Alpha-ID Code)	<i>code</i>
Primärschlüsselnummer 1	Primary Key Number 1 (ICD-10-GM Code)	<i>ICD-10 primary code 1</i>
Sternschlüsselnummer	Standard Key Number	<i>ICD-10 asterisk code</i>
Zusatzschlüsselnummer	Additional Key Number	<i>ICD-10 additional code</i>
Primärschlüsselnummer 2	Primary Key Number 2	<i>ICD-10 primary code 2</i>
Zugehöriger Text (Ohne eventuelle Verweise)	Associated Text (without any references)	<i>display</i>

All seven columns were considered relevant for the envisioned CodeSystem; the assigned FHIR items are displayed in Table 2 on the right. For the central “code” item the Alpha-ID code/ID number was chosen and the associated text as “display”.

Furthermore, there are four columns given for different ICD-10 codes, also including specific types of the German modification. For each of these columns an own property of type “String” is needed (see Table 2, names in italics), and another one for the validity of the entry, represented as a Boolean.

Out of 84746 concepts, 84251 (99.4%) referred to an ICD-10-GM code in the “Primary Key Number 1” column, sometimes in combination with other ICD-10 codes in the other columns. Of the remaining 495 entries, 472 had an associated “Additional ICD-10 code” whereas 23 rows didn’t have any ICD-10 relation. Please note that all of those terms without codes had “Validity” 0, meaning they are not considered valid anymore. All in all, there are 1273 invalid entries which are included here to ensure compatibility with earlier versions.

Using a simple CSV2Json [11] converter wasn’t successful as the produced JSON file is not FHIR compatible thus can’t be uploaded to the Ontoserver. To obtain a FHIR compatible JSON, Snapper was used, a tool developed by CSIRO [12] to enable the creation and maintenance of the FHIR terminology resources i.e. CodeSystem, ValueSet, and ConceptMap. A new CodeSystem resource with appropriate metadata was created in Snapper:Author and the Alpha-ID source data was added via the “Import codes from file (csv/tsv)” function (see Figure 1). Five additional properties were specified as described above to hold the validity and the ICD-10-GM codes that are associated with each Alpha-ID entry. During the import process for each source file column the suitable FHIR element was chosen from the predefined (“code”, “description”) or self-defined items (properties).

Due to Snapper’s current limitation to 25,000 rows per import, the Alpha-ID file had to be split into four files that were imported into the same CodeSystem subsequently.

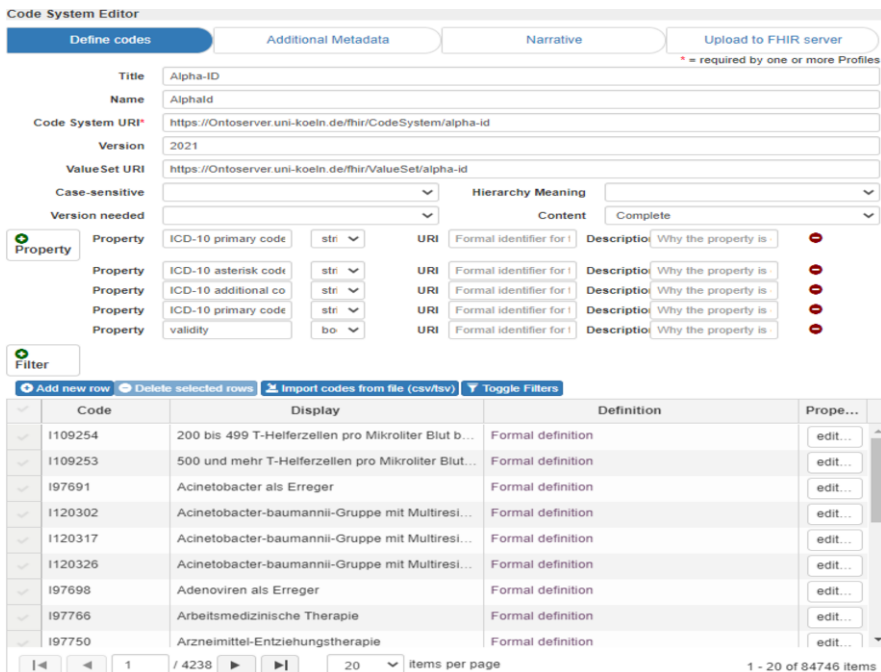


Figure 1. The interface of the Snapper tool. The top section shows the details of the metadata associated with the terminology and the bottom section shows the imported terminology. Please note that all terms were uploaded (bottom right) but only half of them were processed for the download.

The large number of terms slowed down Snapper significantly and when the Snapper-generated FHIR JSON file was downloaded only 49998 entries were included despite the previous import of all four part files with 84746 terms (Figure 2).

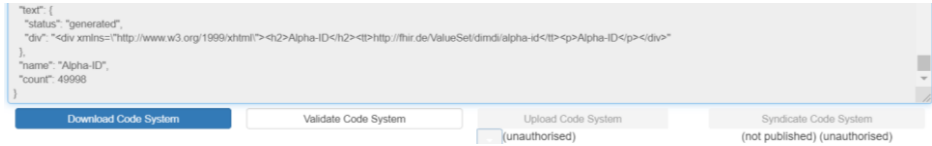


Figure 2. The window to download the FHIR JSON file. Please note the number of terms processed, mentioned in the second last row.

To circumvent that problem, two JSON files were created, each with less than 50K rows separately with the same metadata but different contents. After the download of those two files, they were merged carefully maintaining the internal structure of the file as per FHIR specification. The resulting FHIR resource was then successfully uploaded to the Ontoserver, complete with all 84746 concepts.

3. Results

In the HiGHmed consortium, semantic interoperability among different university hospitals is promised. In addition to that, the Cologne site is responsible for the administration and maintenance of the central terminology server instance including the provision of all the terminologies required for different use cases nationwide. For this purpose, it is required to transform terminologies into an Ontology Server compatible format, which is FHIR in our case.

The FHIR CodeSystem was created by using Snapper. The issue of limitation was dealt with by creating two files that had less than 50K rows. Those two small files were combined using a Text Editor. The resulting file has also been validated by FHIR Transformer [13] as well as the official FHIR validation tool [14] (Figure 3).

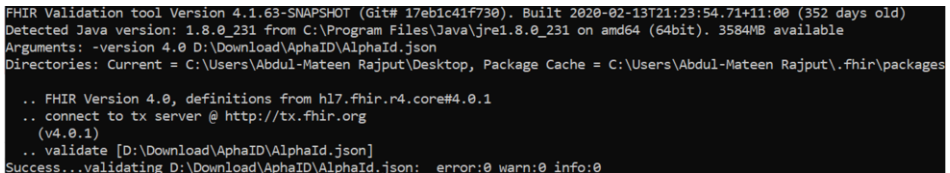


Figure 3. The validation of Alpha-ID FHIR file. The bottom line shows the successful validation.

After the validation, the terminology has been uploaded to the local terminology server. Figure 4 shows the postman query results showing the response of Ontoserver with the Alpha-ID CodeSystem.



Figure 4. The status code 200 OK (top right) shows the successful posting of the terminology on Ontoserver.

The first draft of Alpha-ID is publicly available at: <https://github.com/abdulmateenraj/AlphaID>.

4. Discussion and Conclusion

Maintenance of a terminology server and provision of the associated CodeSystem, ValueSet, and ConceptMap resources are essential but complex and time-consuming tasks. With the emerging FHIR standard, a common approach for terminology services becomes increasingly popular but often legacy terminologies are not (yet) available in a compatible format. Manual transformation of varying source data is not a sustainable solution to overcome this issue. Thus, in this work, the Ontoserver-associated tool Snapper was used to exemplarily convert the Alpha-ID coding system into a FHIR CodeSystem resource.

Snapper proved to provide the urgently needed support for creating and editing FHIR resources through a clearly structured graphical user interface with extensive functionality. Working with large datasets – that are common in the world of terminologies – quickly brings Snapper to its limits in terms of performance though. So, as of now, additional tools and manual editing are both needed to successfully evade the lacking scalability. Further improvements regarding the tool-supported creation and maintenance of FHIR resources are highly desirable.

Acknowledgment

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

Reference

- [1] Terminology-service - FHIR v4.0.1, <https://www.hl7.org/fhir/terminology-service.html>, Accessed 27 January 2021.
- [2] CodeSystem - FHIR v4.0.1, <https://www.hl7.org/fhir/codesystem.html>, Accessed 27 January 2021.
- [3] Metke-Jimenez A, Steel J, Hansen D, et al. Ontoserver: a syndicated terminology server. *J Biomed Semantics*. 2018;9:24.
- [4] SNOMED Home page. SNOMED, Accessed 27 January 2021.
- [5] Home. LOINC, <https://loinc.org/>, Accessed 4 April 2020.
- [6] Rector AL. Clinical terminology: why is it so hard? *Methods Inf Med*. 1999;38:239–252.
- [7] Cimino JJ, Zhu X. The practical impact of ontologies on biomedical informatics. *Yearb Med Inform*. 2006;124–135.
- [8] Alpha-ID, <https://www.dimdi.de/dynamic/en/classifications/icd/alpha-id/>, Accessed 27 January 2021.
- [9] ICD-10-GM, <https://www.dimdi.de/dynamic/de/klassifikationen/icd/icd-10-gm/>, Accessed 27 January 2021.
- [10] Downloads, <https://www.dimdi.de/dynamic/de/klassifikationen/downloads/?dir=alpha-id/>, Accessed 23 January 2021.
- [11] csv2json. npm, <https://www.npmjs.com/package/csv2json>, Accessed 23 January 2021.
- [12] Snapper, <https://ontoserver.csiro.au/snapper/#/codeSystem/d598db98-7d0e-455b-8cab-38defb146a88>, Accessed 25 January 2021.
- [13] Grieve G. *grahamegrieve/fhirserver*. Pascal, <https://github.com/grahamegrieve/fhirserver>, Accessed 25 January 2021.
- [14] Validation - FHIR v4.0.1, <https://www.hl7.org/fhir/validation.html#web>, Accessed 30 January 2021.