German Medical Data Sciences: Bringing Data to Life
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI210074

Fast Healthcare Interoperability Resources (FHIR®) Representation of Medication Data Derived from German Procedure Classification Codes (OPS) Using Identification of Medicinal Products (IDMP) Compliant Terminology

Julian SASS[a,1], Susanne ZABKA[b], Andrea ESSENWANGER[a], Josef SCHEPERS[a], Martin BOEKER[b] and Sylvia THUN[a,c]
[a] *Berlin Institute of Health (BIH), Germany*
[b] *Institute for Medical Biometry and Medical Informatics, University of Freiburg, Freiburg, Germany*
[c] *Charité – Universitätsmedizin Berlin, Germany*

Abstract. Electronic documentation of medication data is one of the biggest challenges associated with digital clinical documentation. Despite its importance, it has not been consistently implemented in German university hospitals. In this paper we describe the approach of the German Medical Informatics Initiative (MII) towards the modelling of a medication core dataset using FHIR® profiles and standard-compliant terminologies. The FHIR profiles for *Medication* and *MedicationStatement* were adapted to the core dataset of the MII. The terminologies to be used were selected based on the criteria of the ISO-standard for the Identification of Medicinal Products (IDMP). For a first use case with a minimal medication dataset, the entries in the medication chapter of the German Procedure Classification (OPS codes) were analyzed and mapped to IDMP-compliant medication terminology. OPS data are available at all German hospitals as they are mandatory for reimbursement purposes. Reimbursement-relevant encounter data containing OPS medication procedures were used to create a FHIR representation based on the FHIR profiles *MedicationStatement* and *Medication*. This minimal solution includes – besides the details on patient and start-/end-dates – the active ingredients identified by the IDMP-compliant codes and – if specified in the OPS code – the route of administration and the range of the amount of substance administered to the patient, using the appropriate unit of measurement code. With FHIR, the medication data can be represented in the data integration centers of the MII to provide a standardized format for data analysis across the MII sites.

Keywords. Health Information Interoperability, Medication, Identification of Medicinal Products (IDMP), FHIR

[1] Julian Sass, Berlin Institute of Health (BIH), Anna-Louisa-Karsch-Straße 2, 10178 Berlin, Germany; E-mail: julian.sass@bihealth.de.

1. Introduction

1.1. Background

Electronic documentation of medication data in hospital information systems (HIS) is a challenge. Often, a patient's medication is captured on paper or in semi-structured format. This however makes it difficult to utilize the data in decision-support-systems or for secondary use in research.

The German Procedure Classification (Operationen- und Prozedurenschlüssel – OPS) is an adaptation of the International Classification of Procedures in Medicine (ICPM) and is mandatory for reimbursement in the inpatient sector. Besides radiological procedures, surgical procedures and so forth, the OPS is also used to encode the application of some medications.

Currently, the German university hospitals cooperate within a project called Medical Informatics-Initiative (MII). Its goal is to provide data in a unified format through data integration centers so that it can be used for research across sites. The basis of this representation is the definition of a core dataset that covers the most relevant data elements - one of which is the Medication module - for the MII use cases.

As a technical representation of the core dataset, HL7 Fast Healthcare Interoperability Resources (FHIR) was chosen by the initiative's consortia and the National Steering Committee. FHIR is a standard in development that defines a set of XML or JSON health data resources and an API to exchange, query and retrieve the data.

This paper presents how a FHIR-based specification was developed that covers medication data with terminology which is compliant to the ISO standards for the Identification of Medicinal Products (IDMP). Furthermore, experiences with the implementation and specification processes are described.

1.2. Requirements

Extract, transform and load (ETL) processes are implemented at different sites of MII partner institutions. The extracted data from heterogeneous sources shall be transformed into the FHIR format as specified within the MII core dataset. The MII requires the use of FHIR Release 4 (R4), the fourth version of the FHIR standard. As a first representation of medication data in the FHIR format, medication data derived from OPS codes were used, because these codes are reliably available in a consistent format at all German hospitals for reimbursement purposes. The requirements are met when the transformed data is loaded into a FHIR server and successfully validated against the core dataset specification.

2. State of the art

FHIR is still evolving and probably not many implementations using the FHIR medication module exist. Bloomfield et al. [1] report how they used FHIR in their electronic health record (EHR). They used FHIR in a patient-facing application for medication management and education. This work is based on the FHIR Draft Standard for Trial Use 2 (DSTU 2) and uses the *MedicationOrder* resource to cover orders of

medications for a patient. In FHIR's first normative version R4, of FHIR, *MedicationOrder* has been replaced by *MedicationRequest*.

Apple implemented FHIR medication data as part of the Health Records feature in iOS 11.3. Besides the aforementioned *MedicationOrder* resource, *Medication* and *MedicationStatement* are used, which are also available in later versions of FHIR [2].

Hong et al. [3] used natural language processing (NLP) techniques to integrate both, structured EHR data and unstructured data from clinical notes to populate elements of *MedicationStatement* resources. They used a set of different NLP tools for data annotation from clinical notes and state that they achieved good results for automated information extraction. The use of external code systems and value sets is identified as an ongoing challenge though.

Zabka et al. [4] have previously demonstrated how information from OPS codes can be mapped to standard pharmaceutical vocabularies. Here, a manual mapping approach was applied.

3. Concept

We developed an implementation guide (IG) that contains a set of profiles to specify how FHIR should be used in our particular context [5]. The IG is based on FHIR R4. For our use case, we utilize the *Medication* and *MedicationStatement* FHIR-resources. *Medication* covers the pharmaceutical product or ingredient that is prescribed, planned or administered. It can be a finished drug or a prescription. It is also possible to specify only an ingredient, should the administered product be unknown. *MedicationStatement* documents the prescription of a drug or its consumption by a patient. *MedicationStatements* can indicate that a patient is currently taking, has taken in the past, or will take a drug in the future. This can be prescription drugs as well as over-the-counter (OTC) preparations that the patient takes on his or her own responsibility.

Wherever possible, terminologies and code systems that meet international standards are intended to be used. For medicinal products and associated information, these are the ISO standards for the Identification of Medicinal Products (IDMP), which are mainly implemented in a regulatory context. IDMP-compliant code systems to identify pharmaceutical ingredients include the ASK-numbers from the German Drug Substance Catalogue (Arzneimittel-Stoffkatalog ASK) [6], the Unique Ingredient Identifier (UNII) from the US Substance Registration System [7], CAS Registry Numbers from the Chemical Abstracts Service (CAS) [8] or SNOMED CT codes. For the type of application and dosage form, the EDQM lists of Standard Terms (European Directorate for the Quality of Medicines and Health Care) "Routes and Methods of Administration" and "Pharmaceutical Dose Forms" should be used.

The OPS mapping is based on the work by Zabka et al. [4] and was extended to work with the 2020 version of the OPS classification. For each OPS code from chapter 6 - Medications, the mapping provides at least one – or, in case of combination products, more than one - coded ingredient from each of the code systems UNII, CAS, and ASK. If defined in the OPS code, structured dosage and unit information can also be included, as well as EDQM codes for routes and methods of administration.

Eventually, we developed an ETL process that connects to our local clinical data repository, transforms OPS medication codes to MII core dataset compliant FHIR resources and loads them into our FHIR server via its RESTful API.

4. Implementation

The implementation is written in Python 3 and uses the latest version of the open-source *fhirclient* library from the official GitHub project [9]. SMART's *fhirclient* contains all the FHIR data model classes in R4 and allows us to easily handle FHIR data in Python.

With the available mapping from OPS codes to structured IDMP compliant terminology as a CSV file, we were able to populate elements in FHIR resources as specified in the MII core dataset medication module. For each OPS code, a combination of a *MedicationStatement* and *Medication* resource is created. The *Medication* resource contains the coded ingredient information and is referenced by the *MedicationStatement* resource, where the start and end date of the medication intake is captured. If available from the OPS code, the structured dosage and route of administration information are also included in the *MedicationStatement* resource. Furthermore, we implemented a random generator for *Patient* resources to cover data about the subject receiving treatment. Figure 1 shows an example of how an OPS code can be represented in FHIR data structures.

Once the ETL script has transformed OPS codes into FHIR resources, it connects to our local FHIR server and loads the data via the RESTful interface. Before actually sending the data though, we use the validate operation and check if the content is valid against the MII core dataset profiles. Validation is performed on the server-side, since it was preloaded with the IG profiles. If the validation request to the server is successful, the resources are posted and created on the FHIR server. At the moment, our pipeline creates a corresponding *Patient* and *Medication* resource for each instance of *MedicationStatement*. Since we decided to take the IDs generated by our FHIR server, the instances for *Patient* and *Medication* had to be posted to the server beforehand, to retrieve the respective IDs which were to be used for the references in each of the *MedicationStatement* instances.

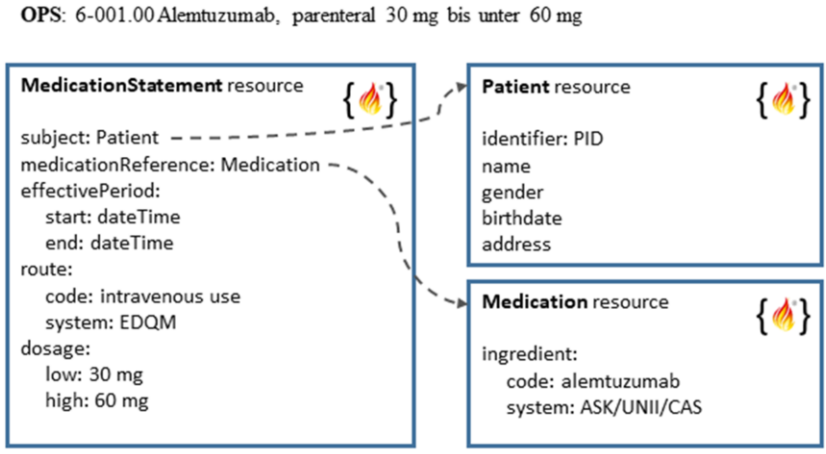

Figure 1: Simplified representation of an OPS code by FHIR resources

5. Lessons learned (Discussion)

During the planning and development phase of the MII core dataset, there was some uncertainty, whether FHIR could meet the needs for the medication module and use cases. Eventually, the FHIR resources in the medication implementation guide only needed little customization and most of the work was spent on specifying which external code systems should be used and in which place.

Hereafter, we chose medication data from OPS procedure codes as an initial use case and transformed the data into the FHIR format. At first, the idea was to parse the information from OPS codes to populate the resource elements, but we soon realized that this approach was not feasible because we could not find a way to programmatically provide ingredient or route of administration codes from standard terminologies. Hence, it was decided to build upon the existing mapping. It may be laborious and time consuming to manually annotate the OPS codes, however at the moment we consider this to be less prone to error compared to e.g. implementing an NLP based solution.

This first practical application of the medication module of the MII core dataset also brought some drawbacks of the current profiles to light. One of these being the required terminology binding to placeholder *ValueSet* and *CodeSystem* resources where the actual content is not present within the resource. These were used for the ASK system since we did not have access to the whole set of codes. As our server is configured to strictly validate every incoming resource to such a degree that it verifies the presence of a code in a value set, validation would always fail when the terminology service tried to retrieve codes from a placeholder resource. The problem here is the limited public accessibility to the official list of ASK numbers. Thus, we modified the profiles accordingly and loosened the binding strength to ensure that the MII core dataset is implementable across partnering sites.

The availability of the open-source *fhirclient* Python library facilitated the implementation. Moreover, due to the reference implementation FHIR server which was used [10], we were able to validate and exchange data in a relatively short period of time. We plan to use a similar approach and implement additional ETL processes to provide additional FHIR-enabled data.

6. Conclusion

Integrating structured medication data is challenging and requires an extensive amount of work. We benefited from the preliminary effort on the OPS mapping. Deriving medication information from OPS codes is of course only a very limited and retrospective view on some patients' pharmacological therapy. Yet, we see this work as a starting point to test our implementation and the core dataset specification. From here, we plan to continually integrate additional data sources and improve interoperability of medication data.

The implementation of the medication module has helped us gain valuable experience and we were able to test our specification on this use case. Based on the lessons learned, we were able revise the FHIR profiles and introduce improvements.

Conflict of Interest

The authors state that they have no conflict of interests.

Acknowledgements

This study was funded by the German Federal Ministry of Education and Research (BMBF) within the ‘‘Medical Informatics Funding Scheme’’ (FKZ 01ZZ1801X, FKZ 01ZZ1802X, FKZ 01ZZ1803X, FKZ 01ZZ1804X).

References

[1] R.A. Bloomfield, F. Polo-Wood, J.C. Mandel, and K.D. Mandl, Opening the Duke electronic health record to apps: Implementing SMART on FHIR. Int J Med Inform **99** (2017), 1–10.
[2] M.L. Braunstein, Health Informatics on FHIR: How HL7's New API is Transforming Healthcare, Springer International Publishing, Cham, 2018.
[3] N. Hong, A. Wen, F. Shen, S. Sohn, S. Liu, H. Liu, and G. Jiang, Integrating Structured and Unstructured EHR Data Using an FHIR-based Type System: A Case Study with Medication Data. AMIA Jt Summits Transl Sci Proc 2017 (2018), 74–83.
[4] S. Zabka, D. Ammon, T. Ganslandt, J. Gewehr, C. Haverkamp, S. Kiefer, H. Lautenbacher, M. Löbe, S. Thun, and M. Boeker, Towards a Medication Core Data Set for the Medical Informatics Initiative (MII): Initial Mapping Experience between the German Procedure Classification (OPS) and the Identification of Medicinal Products (IDMP). In: Proceedings of the Joint Ontology Workshops 2019: Episode V: The Styrian Autumn of Ontology, Vol-2518, A. Barton, S. Seppälä, and D. Porello, eds., September 23-25, 2019.
[5] Medizininformatik Initiative - Kerndatensatz - Modul Medikation, https://simplifier.net/guide/medizininformatikinitiative-modulmedikation-implementationguide/igmiikdsmodulmedikation [cited 2020 July 16].
[6] Rohdaten der Stoffbezeichnungen, https://www.dimdi.de/dynamic/de/arzneimittel/arzneimittel-recherchieren/rohdaten-stoffbezeichnungen/index.html [cited 2020 July 15].
[7] Substance Registration System - Unique Ingredient Identifier (UNII), https://fdasis.nlm.nih.gov/srs/ [cited 2020 July 15].
[8] K.A. Hamill, R.D. Nelson, G.G. Vander Stouw, and R.E. Stobaugh, Chemical Abstracts Service Chemical Registry System. 10. Registration of substances from pre-1965 indexes of Chemical Abstracts. J Chem Inf Comput Sci 28 (1988), 175–179.
[9] Python SMART on FHIR client, https://github.com/smart-on-fhir/client-py [cited 2020 March 23].
[10] Vonk FHIR Server, https://fire.ly/products/vonk/vonk-fhir-server/ [cited 2020 July 14].