

Fit for Purpose: Analyzing the German Archiving and Exchange Interface for Medical Practice Management Systems

Linus BARKOW ^{a,b,1}, Oliver MEINCKE ^b, Hannes ULRICH ^c and Josef INGENERF ^{a,c}

^a*Institute of Medical Informatics, University of Lübeck, Germany*

^b*Epikur Software & IT-Service GmbH & Co. KG, Berlin, Germany*

^c*IT Center for Clinical Research (ITCR-L), University of Lübeck, Germany*

Abstract. The archiving and exchange interface for practice management systems of the *Kassenärztliche Bundesvereinigung*, defined by FHIR (Fast Healthcare Interoperability Resources) profiles with extensions, describes a new opportunity for medical practitioner to change the system provider. The expectation is to transfer an entire database of a legacy system to another system without data loss. In this paper the potential loss of data is analyzed by comparing parameters. The results show that during an import on average 75% of the parameters per profile are supported and on average only 49% of the reviewed parameters, existing in the exporting system, could be represented based on the interface specification.

Keywords. HL7 FHIR, electronic health record, health information exchange, archives

1. Introduction

The ongoing digitalization affects the German healthcare system and the German government is facilitating this process. In 2015 the “E-Health Act” was introduced which regulates the connection of all doctor's offices to the central telematics infrastructure (TI). The secure environment contains the management of insured persons’ master data synchronized with medical practice management systems (*Praxisverwaltungssystem - PVS*). Most medical institutions are using a PVS to manage daily tasks and quarterly accounting [1]. There are more than 200 different practice management systems in Germany that differ in specializations, e.g., ambulatory healthcare like psychotherapy [1]. The healthcare market offers a wide range of different systems but changing the provider of the PVS normally results in major problems. The data transfer process is error-prone and time-consuming because the entire database needs to be transferred to the new system. Furthermore, the architectures and data models of different PVS are very heterogeneous so that a complete transfer and transformation are only possible with great effort and costs [2]. Therefore, the new law (Sozialgesetzbuch (SGB) V §291d) motivates and regulates the use of open, system-neutral interfaces in medical practice management systems. The National Association of Statutory Health Insurance

¹ Corresponding Author: Linus Barkow, Institute of Medical Informatics, Ratzeburger Allee 160, 23562 Lübeck, Germany; E-mail: linus.barkow@student.uni-luebeck.de.

Physicians (*Kassenärztliche Bundesvereinigung* - KBV) is assigned for defining the interface specification. This new interface is called PVS archiving and exchange interface (*PVS-Archivierungs- und Wechselschnittstelle* - AWS) and is supposed to allow the human-readable archiving of patient related data and exchange between different systems [2]. The interface structure was defined to represent the most common data stored in a PVS. In the following, it will be analyzed how the interface can represent special features and parameters of the practice management system Epikur and examine which challenges can arise during the integration of the interface.

2. Material and Methods

2.1. Fast Healthcare Interoperability Resources (FHIR)

The KBV defines the interface using various profiles, which are based on the Fast Healthcare Interoperability Resources (FHIR) standard [3]. FHIR is an exchange format for medical data, which was introduced by Health Level 7 (HL7) in 2013. The focus is to simplify the HL7 adoption significantly and to use existing open internet standards. The profiles represent the most important units of health data during information exchange. The specification considers these profiles as distinguished building blocks, which can be integrated into many processes as a logical entity. One profile instance refers to other profiles by using the unambiguous identifier so that a data representation can be composed by making use of various building blocks. Due to general applicability and complexity, the modules of the standard are limited to the most necessary but relevant parameters. To be able to integrate all information, it is possible to define and integrate further attributes via extensions. Depending on their content, the profiles are divided into different packages as well as sub-packages. In addition to the profiles, Value Sets and Code Systems are used to aggregate codes from different terminologies, ontologies, or enumerations. To guarantee interoperability, only specified codes are permitted in a defined context. By using the *Bundle* resource, any number of profiles can be packaged and transferred in one object. Multiple reference implementations for the FHIR standard already exist. The current version is R4 (4.0.1), which was released on 30 October 2019. The previous version Standard for Trial Use (STU) 3, released in 2016, is currently available in version (3.0.2) and marked as “historical” [4].

2.2. Specification of the interface

The KBV specified a catalog of requirements to define and constrain the interface. The profiles constrain general STU 3 FHIR resources. To be able to represent all peculiarities of the German health system, many extensions are created, and Code Systems are defined. The resulting profiles are grouped into five different bundles: address book, treatment modules, patient files, office supplies and appointments. For example, the patient file bundle is created for each patient and contains all patient-relevant data. The KBV aims to use all possible profiles in order to export the complete PVSs information and marks certain elements with the property “mustSupport” to highlight the parameters importance. These elements should be supported independently of the cardinality if they are available in the information system. When importing data records, all exported data must be transferred according to the scope of the importing system. The context of the information should be retained correctly [2,5].

2.3. Calculation of parameter difference

For the quantitative and statistical evaluation of each AWS profile, it was examined whether a parameter is supported by the reviewed PVS, the AWS or both. Each parameter was manually assigned to one of three labels. In addition to “is supported”/“not supported”, a parameter can be marked with “support unknown” if the support of this parameter is unclear. The occurrence of the labels for a profile is counted and the percentage in relation to the total parameter number is calculated. Note that only parameters that contain variable data were analyzed. If a parameter is predefined or does not contain any information, this parameter is not included in the statistic. To display the results, profiles were summarized to packages and sub-packages. Additionally, all used profiles are analyzed to determine how many of the “mustSupport” elements are supported. Furthermore, the parameters stored in the tables of the PVS database are examined which and how many profiles are supported. The evaluation is to show how many of the stored parameters are affected by the structured elements of the modified FHIR profiles.

3. Results and Discussion

3.1. Interface Implementation

The AWS definition is available in XML files, which contain the structure definitions of the various profiles. According to these definitions, objects were created using the open-source Java reference implementation HAPI-FHIR [6] and filled with all available data from the database of the Epikur PVS. For the mapping of the various coding systems, the given systems were extended. At least one internal code was added in a new tag `<pvsCode>` for each code (see Fig. 1). If the KBV publishes new codes, the program is able to load these dynamically and provides updates to the PVS system. To ensure the data integrity and quality, each given bundle is validated against the structure definitions of the interface for either the import or export.

As the object structures in the PVS often do not match the profiles in the AWS interface, it is challenging to ensure that the data transfer is correct and as lossless as possible. Since a complete and structured data transfer is not possible, further data is stored in free text fields or as PDF attachments. Examples are documents such as orders, letters, prescriptions, laboratory data and medical documentation.

3.2. Import: Interface to PVS

A statistical evaluation reveals how many parameters of the interface definition the PVS supports. Figure 2 visualizes overall packages from that at least 62% of the parameters per object are supported and maximally 26% are not supported. On average, 78% of the parameters per profile are provided. The evaluation included overall 1073 different parameters. Examples of unsupported parameters are the prefix qualifier and suffix part of names, the dagger-asterisk coding and version of ICD-10 codes and the coded dosage information of drugs. Furthermore, profiles such as allergy, cure, vaccination, preventive power of attorney are not provided by the Epikur-PVS since this information are not recorded.

```

<CodeSystem xmlns="http://hl7.org/fhir">
  <url value="https://fhir.kbv.de/CodeSystem/74_CS_AW_Medikamenten_Abgabeart"/>
  <version value="1.10.002.pvs.1"/>
  <name value="74_CS_AW_Medikamenten_Abgabeart"/> [...]
  <concept>
    <code value="Grüenes_Rezept"/>
    <display value="Grünes_Rezept"/>
    <pvsCode value="REZEPT.GREEN"/>
  </concept>
  <concept>
    <code value="Privatrezept"/>
    <display value="Privatrezept"/>
    <pvsCode value="REZEPT.PRIVAT"/>
    <pvsCode value="REZEPT.PRIVAT.LANG"/>
  </concept> [...]
  <pvsEnumClass value="FormEnum"/>
  <pvsDate value="2019-11-26"/>
</CodeSystem>

```

Figure 1. Example of a modified Coding System for mapping interface defined codes to the PVS codes.

The additional evaluation showed that the PVS supports 93% of the “mustSupport” parameters. All remaining information (e.g. the dosage of a medicine or the address of a home visit) could be written in an unstructured parameter or the information is not available in the data set.

Therefore, the data loss during an import out of the FHIR structure is relatively low.

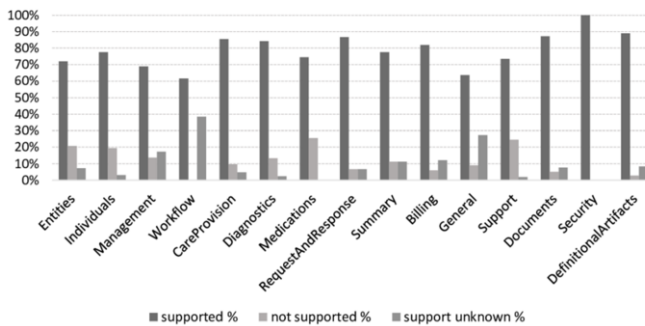


Figure 2. Percentage per package of supported and not supported FHIR interface parameters by the PVS.

3.3. Export: PVS to interface

As seen in Figure 3, many parameters are not supported by the interface representation. The diagram differentiates between five object groups and is based on the analysis of 637 parameter. For patient data only 49% could be exported into the structured parameters of the interface. These parameters often contain information like the blood type, living environment or company of the patient, data about relatives, information about births of children, assignments to groups or psychotherapy sessions that have already taken place. The patient package also includes individual patient settings like the preferred contact method, therapeutic schemata, private service factors or session counter. For the “user” class 54% of the parameters are supported but 35% are unsupported. On the one hand, information like the name, the address or the doctor number can be exported. On the other hand, data like the relation between a doctor and their assistances, the assignment to user-profiles or the settings in the PVS are not directly provided in the interface. Looking at the results of the “service” package 57% are

supported and 24% are not supported parameters. Own fee schedules, as well as service items, can only be integrated into the interface structure in a very complicated way. Information like the tax profiles, visibilities or internal marks cannot be integrated into the FHIR structure. The PVS “QuickButton” class – which corresponds to the KBV “DefinitionalArtifacts” – supports 43% and does not support 53%.

Since the interface is particularly specialized in billing objects, the billing-relevant data can be transferred with fewest losses. But some data like the tax number gets lost which is not directly relevant for archiving but for (re)import, since this information is relevant for accounting. In the “QuickButton” class the rough structure of the treatment modules can be represented well, but as soon as the parameters become more specific and individually adaptable, they can no longer be adopted. For example, key combinations, color settings or conditions for the individual components get lost. The listed aspects also explain the satisfying results for the analyzed “TimeLineElements” like diagnosis, therapy or observation elements. Hence, interoperable exchange of additional data is difficult.

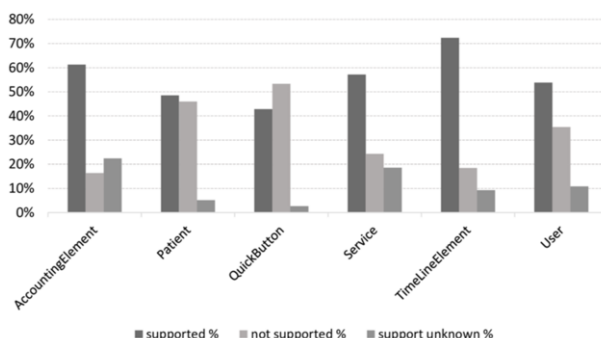


Figure 3. Percentage per package of supported and not supported PVS parameter by the interface structure.

3.4. Compatibility challenges

The Code Systems and Value Sets show clear differences to the PVS-internal values sets which challenge the compatibility and mapping. For example, the interface specifies a Code System for the family status which distinguishes between more species than in the PVS system. The necessity of the codes for annulled, interlocutory, polygamous, never married, and domestic partner may be arguable. Another difference is the attachment types of a document. The KBV specifies fewer codes than used in the PVS system, which implements a variety of types due to legacy support. However, these are still available in the exporting PVS and are meant to be exported without information loss. Thus, it is difficult to integrate old forms into the given structures. These varieties demonstrate that a lossless data exchange is hardly possible.

Another major problem is the different approach of storing information of forms and the corresponding context, e.g., a diagnosis field. The PVS is using flat structures like unstructured text to store the usual information. Whereas FHIR separates this information in its own resource with the possibility of providing more details and reference to other resources, e.g. coding systems or physicians. A flawless transformation from the PVS to FHIR objects can hardly be generated from the string parameters. Therefore, the AWS concept is suitable for data archiving, but a lossless data exchange needs a more elaborate assessment. It also should be considered that the actual

meaning of a data set is not changed during transmission. A wrong interpretation of the interface specification could have serious consequences.

One important disadvantage is evident: the interface is not downward compatible. An import can only be performed on data with the same interface version as the software component. A change of the interface version, as well as the update to the current FHIR version R4 and following updates, will entail major changes in the PVS.

4. Conclusion

This study analyzed the data loss during data export and import with the archiving and exchange interface for practice management systems of the KBV. The results show that there is a considerable difference between the PVS and the interface representation. An export of all exportable data and a reimport in the same system will result in a significant data loss. Even if all parameters that are not supported could be integrated into unstructured text fields, a reimport is much more complicated and a loss of information not avoidable. The interface is a suitable option for archiving patient-related data due to the structured data representation and the generated human-readable output of each profile. Physicians and therapists can access their data continuously even if the system which generated the data is outdated.

In the current state, the interface is missing a few users' requirements as highlighted in the evaluation and is basically a good alternative to the x-/BDT standard, which has been widely used so far. Through a collaboration between software manufacturers and KBV the interface will be updated and currently not supported parameters will be added. It remains to be seen whether this interface will simplify the change between PVSs and how far it will be used for the archiving of practice data.

Acknowledgment

The work has been carried out at Epikur Software & IT-Service GmbH & Co. KG, Berlin and supervised by the Institute of Medical Informatics, Universität zu Lübeck.

References

- [1] Kassenärztliche Bundesvereinigung. Praxisverwaltungssysteme. <https://www.kbv.de/html/pvs.php> (accessed December 18, 2019).
- [2] Dezernat Digitalisierung und IT in der Arztpraxis. Festlegung der Archiv- und Wechselschnittstelle nach § 291d Abs. 1 SGB V. 1st ed. Kassenärztliche Bundesvereinigung; 2019. <ftp://ftp.kbv.de/itaupdate/291d-Schnittstellen/PVS-Archivierungs-Wechsel-Schnittstelle/> (accessed November 06, 2019).
- [3] Bender D, Sartipi K. HL7 FHIR: An Agile and RESTful approach to healthcare information exchange. In: Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems. IEEE; 2013.
- [4] HL7 org. FHIR - Fast healthcare interoperability resources. <http://www.hl7.org/fhir> (accessed December 16, 2019).
- [5] Dezernat Digitalisierung und IT in der Arztpraxis. Anforderungskatalog der Archiv- und Wechselschnittstelle (AW-SST). 1st ed. Kassenärztliche Bundesvereinigung; 2019. <ftp://ftp.kbv.de/itaupdate/291d-Schnittstellen/PVS-Archivierungs-Wechsel-Schnittstelle/> (accessed November 06, 2019).
- [6] University Health Network. HAPI-FHIR fhir made simple; 2019. <https://hapifhir.io/> (accessed December 16, 2019).