Building Continents of Knowledge in Oceans of Data: The Future of Co-Created eHealth
A. Ugon et al. (Eds.)

doi:10.3233/978-1-61499-852-5-231

Standardising the Development of ODM Converters: The ODMToolBox

Iñaki SOTO-REY[a, §], Philipp NEUHAUS[a, §, 1], Philipp BRULAND[a], Sophia GEßNER[a], Julian VARGHESE[a], Stefan HEGSELMANN[a], Tobias BRIX[a], Martin DUGAS[a] and Michael STORCK[a]
[a]*Institute of Medical Informatics, University of Münster, Germany*

Abstract. CDISC's Operational Data Model (ODM) is a flexible standard for exchanging and archiving metadata and subject clinical data in clinical trials. The Portal of Medical Data Models (MDM-Portal) uses ODM to store more than 15000 medical forms. As not every electronic health system accepts ODM as input format, there is a need for conversion between ODM and other data standards and formats. This research proposes a standardised template-based process to develop ODM converters. So far, ten converters have been developed and integrated in the MDM-Portal following this process and new ones should be included soon. The template, programming utilities and an ODM test suite have been made online available and can be used to easily develop new converters.

Keywords. Semantics, CDISC ODM, interoperability.

1. Introduction

The use of standards in Medical Informatics is a common practice to enhance interoperability between information systems. Their adoption in clinical research is highly recommended due to the increasing complexity of clinical trials, number of electronic systems being used and amount of data collected [1].

The Operational Data Model (ODM) is an XML-based structural interoperability standard developed by the Clinical Data Interchange Standards Consortium (CDISC). ODM's initial use case was to facilitate the exchange of metadata and clinical data between electronic data capture systems [2]; however, ODM's flexibility allows its use in a wider range of use cases such as Electronic Health Record infrastructure, data collection, data analysis and study archival [3].

The Institute of Medical Informatics (IMI) at the University of Münster is using ODM to store medical forms in the portal of Medical Data Models (MDM-Portal) [4], which became the largest open-access clinical metadata repository in 2015; including - to date- more than 15000 forms. The MDM-Portal includes an ODM editor that allows users to create, semantically annotate in Unified Medical Language System and upload medical forms which can be commented, rated and downloaded by other users [5].

---

§ The authors contributed equally to this work.

1 Dr. rer. medic. Philipp Neuhaus, Institute of Medical Informatics, University of Münster, Albert-Schweitzer-Campus 1, Gebäude A11, 48149 Münster. Philipp.Neuhaus@uni-muenster.de

Health information systems apply different information structures, which are represented in various metadata standards. These systems are not always able to import or handle information in ODM format. Thus, there is a need for conversion between ODM and other compatible data standards and formats.

The aim of this research is to propose a standardised straightforward template-based process to develop converters from ODM to other data formats and demonstrate its feasibility with the development of several data format converters.

2. Methods

The ODMToolBox is a set of tools and utilities to convert ODM into other data formats. It has been developed using JavaEE programming language and Spring framework. The ODMToolBox framework can be deployed on a Java application server such as Tomcat and provides a standardised Representational State Transfer Application Programming Interface (REST API), which accepts client queries and forwards them to the converter engine. Generic methods to parse incoming ODM files of different versions (1.3, 1.3.1, 1.3.2) into a Java object model structure are available.

As a starting point to implement a new converter, a mapping between ODM items and the new data model is required. If the targeted data model permits it, the specific fields for semantic codes must also be mapped. Once the mapping is finished, the converter can be developed using the template within the ODMToolBox for the core implementation as well as the ODM utilities available in the framework. Developers of new converters may also use a sandbox webpage to test their prototypes and a set of ODM test cases developed and made available by the IMI team.

3. Results

Mapping the ODM structure into a target information model requires the identification of corresponding elements and attributes. If an element/attribute cannot be directly mapped, a workaround is pursued (e.g. including semantic codes in a non-specific field [6]). If this is also not possible, the element/attribute is ignored and included as a limitation of the converter. This process is performed forwards and backwards to ensure that all elements and attributes of both information models are covered.

The development of a new converter constitutes the translation of the mapping previously described into Java programming language in the dedicated function in the template. The converter template in the ODMToolBox already includes the functionality needed to upload an ODM file, validate the ODM-XML (conforms to a valid ODM structure), convert the ODM items to Java object models and generate the resulting files. The ODM form is stored in a temporary local Java object that uses an ODM Java library developed by the IMI team [7]. The method called "*convertFormToTemplate*" receives all important ODM attributes for the transformation such as form and data item definitions and returns the converted form.

Inconsistencies and/or lost information found during the transformation are added to a "Conversion Notes"-Log that is presented to the user after the conversion. Conversion notes have three different severity levels: Notice (e.g. automatically added placeholders), Warning (e.g. missing languages, discarded information) and Critical (errors that block the conversion of single elements).

Before a new converter is deployed on the production server that stores the publicly available version of the ODMToolBox, the converter must be properly tested. For this, a test suite of 20 ODM-XML data specification samples including forms with common ODM items and inconsistencies has been developed.

The template-based process has been used to develop several converters (Table 1), some of which can be openly accessed and used as a basis for new mappings and implementations [6, 8-13].

Table 1. The first column indicates the name of the information model to which ODM was converted to; the second a description of the data model and the third a reference to the publication of the converter (if exists).

Data format	Description	Reference
CDA	HL7 Clinical Document Architecture standard for clinical documents structure and semantics	[8]
FHIR (JSON / XML / RDF)	HL7 Fast Healthcare Interoperability Resources standard for clinical data exchange	[9]
MACRO-ODM	Elsevier's electronic data capture solution for clinical trials	
PDF	Portable Document Format	
ODK	Open Data Kit: open-source electronic patient reported outcome system	
OpenClinica	Open-source clinical trial management software	[6]
REDCap	Research Electronic Data Capture: Research online surveys and databases management software	[10]
ResearchKit / ResearchStack	Electronic Patient Reported Outcome development framework	[11]
SPSS	IBM Statistical Package for the Social Sciences	[12]
ADL	Archetype Definition Language for openEHR archetypes	[13]

Further converters have been developed and made available in the ODMToolBox but are not included in Table 1 because they were not developed following the process described in methods: CSV, SQL, R, PDF and Excel. The PDF converter in the ODMToolBox is used in the MDM-Portal since September 2017; until then, a built-in PDF converter was used instead.

Between the 1st October 2016 and the 30th September 2017, 2412 downloads in 19 different data formats took place. 1011 original ODM-XML forms were requested followed by 544 PDF downloads (439 with the old converter and 105 with the new template based converter) and 290 times as Excel file. MACRO-ODM was requested in 224 downloads whereas REDCap in 80. Forms in FHIR were downloaded 107 times (71 in FHIR JSON, 36 in FHIR XML). Some of the newest converters available such as ResearchKit have not been used yet. The number of downloads for every converter can be visualised in Figure 1.

ODM forms may also be converted without accessing the MDM portal, but accessing the ODMToolBox directly through the following website: https://odmtoolbox.uni-muenster.de/.

4. Discussion

A developing framework including several tools to handle ODM files together with a template for the rapid and consistent development of converters from ODM to other data formats has been developed. Following the template-based process proposed, ten converters have been implemented including 2412 forms downloaded from the MDM-Portal in different data formats.

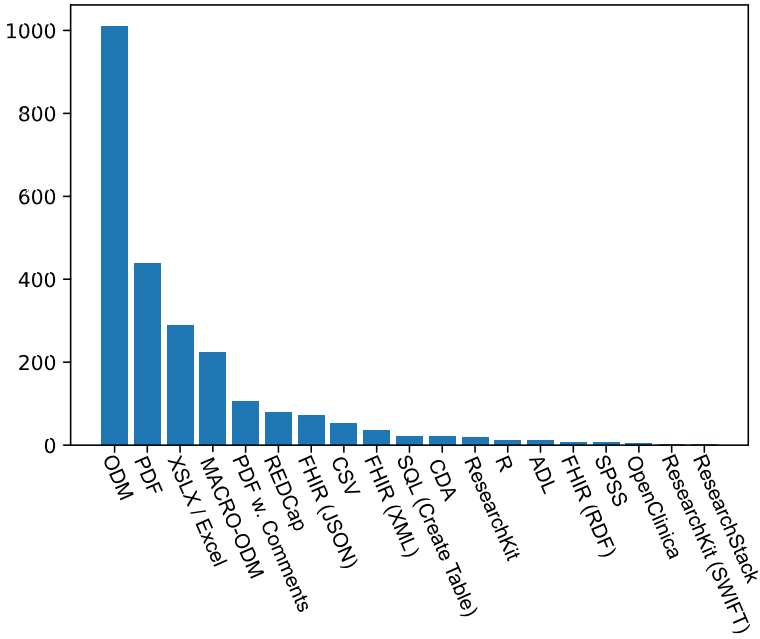

Figure 1. Number of downloads in the available data formats from the MDM-Portal (from October 2016 to September 2017).

Despite of the wide range of ODM converters available in the ODMToolBox, the most desired download data format in the MDM-Portal is ODM-XML, followed by PDF, XLSX and MACRO-ODM.

Information models and data standards normally differ on their nature and purposes. Hence, the conversion of one data format into another commonly implies some loss of information. ODM is a powerful standard for the representation of case report forms but entails some limitations when it is used for representing other clinical forms such as Patient Reported Outcomes [11]. When the XML-based standard is converted to other data formats such as SQL or R, information such as semantic annotations may be ignored or skipped. In order to provide an overview of the information lost during the transformation process, a text file containing notes for every piece of lost information is encapsulated in a .zip file, together with the transformed data model.

While it is possible to check converted files for formal correctness (e.g. using testing environments and test datasets), it is an arduous task to detect mapping errors and semantic discrepancies. Testing environments can be easily deployed locally for proprietary data formats like REDCap and OpenClinica; on the other hand, a complete testing of other data formats like FHIR JSON would require multiple test installations of different FHIR-compliant systems.

One of the advantages of the proposed framework is the ease of creating a new converter. Routines for opening ODM files and serving RESTful requests are already provided and the developer can focus on the actual conversion and mapping issues. Thanks to the standardised REST API, integrating a new data format as a download option in the portal is doable on short notice. Thus, it is reasonable to implement converter even for exotic data formats that will not be used regularly and may only be interesting for a minor number of users.

5. Conclusions

The proposed process to create ODM converters allows developers to easily implement new converters that can be integrated within the ODMToolBox. With this toolkit and the MDM-Portal, the more than 15000 available forms in the portal can be used in a great variety of electronic systems. The template for new converters, together with the ODM utilities and the ODM test suite can be freely accessed under https://imigitlab.uni-muenster.de/published/ODMToolBox.

6. Acknowledgments

This research was supported by German Research Foundation (Deutsche Forschungsgemeinschaft, DFG grant DU 352/11-1). Special thanks to Maurice Heine, who contributed to the development of the ODMToolBox.
The authors Philipp Neuhaus and Iñaki Soto-Rey contributed equally to this work. The authors declare that they have no competing interests.

References

[1] R.L. Richesson and P. Nadkarni, Data standards for clinical research data collection forms: Current status and challenges, *Journal of the American Medical Informatics Association : JAMIA* **18** (2011), 341–346.
[2] CDISC, *Operational Data Model (ODM)-XML* (https://www.cdisc.org/standards/transport/odm).
[3] S. Hume, J. Aerts, S. Sarnikar and V. Huser, Current applications and future directions for the CDISC Operational Data Model standard: A methodological review, *Journal of biomedical informatics* **60** (2016), 352–362.
[4] M. Dugas, *Portal of Medical Data Models (MDM)* (https://medical-data-models.org/).
[5] M. Dugas, P. Neuhaus, A. Meidt, J. Doods, M. Storck, P. Bruland and J. Varghese, Portal of medical data models: Information infrastructure for medical research and healthcare, *Database : the journal of biological databases and curation* **2016** (2016).
[6] S. Gessner, M. Storck, S. Hegselmann, M. Dugas and I. Soto-Rey, Automated Transformation of CDISC ODM to OpenClinica, *Studies in health technology and informatics* **243** (2017), 95–99.
[7] I. Soto-Rey, S. Geßner and M. Dugas, Operational Data Model Conversion to ResearchKit, *Studies in health technology and informatics* **245** (2017), 225–229.
[8] M. Dugas, ODM2CDA and CDA2ODM: Tools to convert documentation forms between EDC and EHR systems, *BMC medical informatics and decision making* **15** (2015), 40.
[9] J. Doods, P. Neuhaus and M. Dugas, Converting ODM Metadata to FHIR Questionnaire Resources, *Studies in health technology and informatics* **228** (2016), 456–460.
[10] P. Neuhaus, S. Gessner, S. Hegselmann, M. Dugas and J. Varghese, *Brücken statt Mauern: Proprietäre Export-Formate für mehr Interoperabilität*, 2017.
[11] I. Soto-Rey, M. Dugas and M. Storck, Implementation of an ODM and HL7 Compliant Electronic Patient-Reported Outcome System, *Studies in health technology and informatics* **228** (2016), 421–425.
[12] P. Bruland and M. Dugas, S2O - A software tool for integrating research data from general purpose statistic software into electronic data capture systems, *BMC medical informatics and decision making* **17** (2017), 3.
[13] P. Bruland and M. Dugas, Transformations between CDISC ODM and openEHR Archetypes, *Studies in health technology and informatics* **205** (2014), 1225.