

From machine-readable CDISC Standard Specifications to the e-Protocol

Jozef Aerts, University of Applied Sciences FH Joanneum, Graz, Austria

ABSTRACT

CDISC standard specifications are still published as PDF or HTML documents, which are not very well machine-readable and surely not machine-executable. Therefore, implementation of the standards in software or of the corresponding implementation guides require a human interpretation step, leading to different interpretations of the standards. This paper describes the development of a machine-readable implementation of the "Submission Data Standard Model Implementation Guide" (SDTM-IG) in XML. First attempts have been made to also integrate SDTM validation rules into the SDTM-IG in XML itself, as statements in a pseudo machine language. This will be extended in future using the "Open Rules for CDISC Standards" rule implementations written in the XQuery language. The current SDTM-IG in XML already allows direct import in software tools, e.g. for automated generation of a template define.xml. In the second part of the paper, a protocol annotation tool is presented, allowing to annotate narrative protocols with codes from over 20 coding systems, from ATC to WHO-ART and including all CDISC controlled terminology. The tool uses a number of publicly available RESTful web services for retrieval of the best matching codes. It currently already allows to automatically generating SDTM trial design datasets as well as CTR-XML for clinical trial registries. When used in combination with a metadata repository, this methodology has the potential of generating an almost complete study design including CRFs starting from the narrative protocol. With a larger number of annotated protocols, machine learning technologies can be used to automate such protocol annotations.

INTRODUCTION

CDISC Standards have mostly been published as PDF documents. More recently, the development teams have received the possibility to use collaborative tools such as a "wiki" [1] and "Jira" issue tracking tool [2] to develop new standards, which can then be published in HTML format on the CDISC website [3].

However, this does not solve the main problem that when a new standard is published, it must be implemented in software, requiring a human step interpreting the standard and then creating software code from that human interpretation. This leads to many different interpretations of the very same standard, undermining the principles of a "standard". Ideally, the standard should be in a machine-readable format and be read by a software that then generates implementation software for that standard automatically. In the case of SDTM, one could think about the automated generation of a define.xml "template" (in XML of course) from the Implementation Guide itself.

Another major problem is that clinical research protocols are usually developed as office documents in narrative form. Also here, a human interpretation step is necessary to "translate" such a narrative protocol into a study design, either e.g. as a machine-readable CDISC ODM file (allowing automated setup of the EDC system and the e-CRFs) or to generate a study database, CRFs, instructions for sites, and so on, and ultimately regulatory submission data sets.

The latter problem is not easily solvable, as protocol writers usually have no idea about submission data standards, although an electronic submission for obtaining a marketing authorization is one of the main objectives of a clinical trial after all. This paper describes a method for annotating narrative protocols with information and codes from over 20 coding systems for the purpose of high quality study design and for electronic submissions to the FDA and PMDA, which may be a first step to come to an "e-protocol". In our tool, the retrieval of suitable codes is based on the use of publicly available RESTful web services.

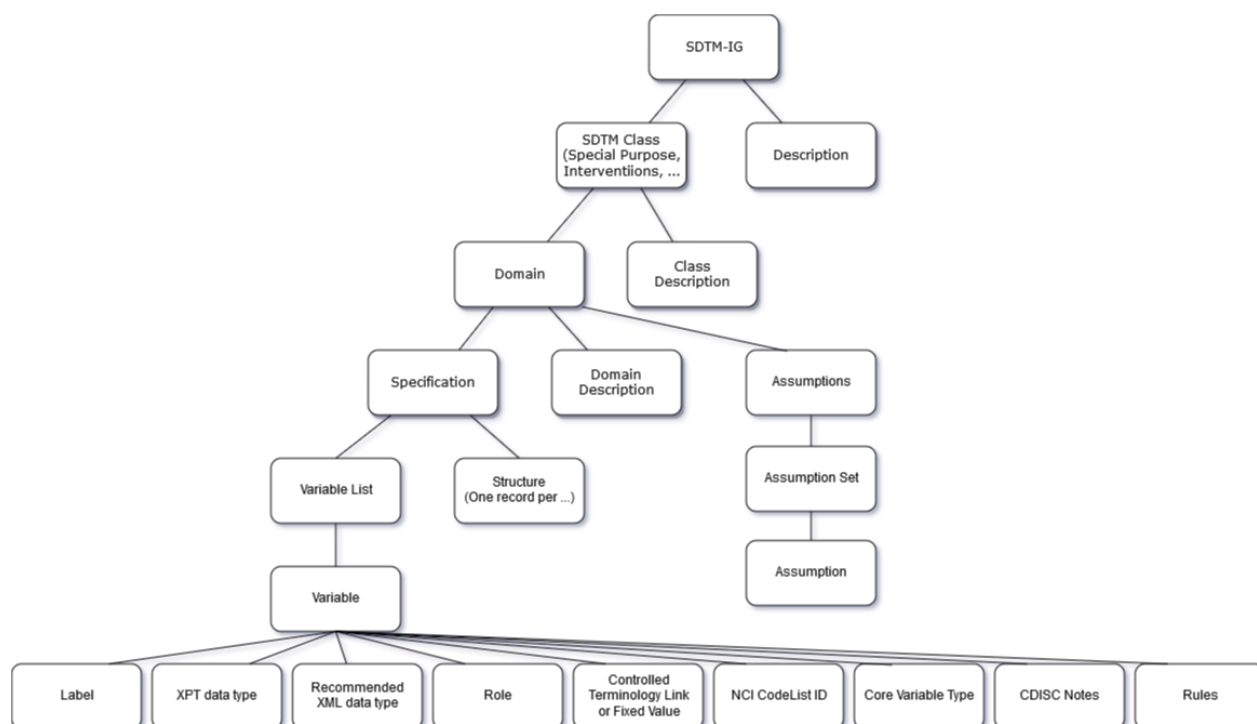
THE SDTM-IG IN XML

DEVELOPMENT OF THE SDTM-IG IN XML

CDISC still publishes standards documents and its implementation guides like the SDTM-IG as PDF or HTML, which is barely machine-readable and surely not machine-interpretable. As the contents of the SDTM-IG are highly structured, it should not be too difficult to generate an implementation guide as an XML document. Therefore, we started such an effort as a student project. At the moment, the XML-ification has been limited to all the SDTM domain descriptions (sections 5-8 in the SDTM-IG), which comprises about 55% of the volume of the SDTM-IG. These sections are also the highest structured sections in the SDTM-IG.

The current hierarchical structure of the final XML document is the following:

PhUSE EU Connect 2018



For each of the "Description" elements, we have added a single "TranslatedText" element with an "xml:lang" attribute containing the value "en", indicating that this is the English version of the description. This mechanism was borrowed from the CDISC ODM standard [3,4] and allows to add descriptions in different languages. This can be used in future to e.g. comprise the Japanese translation of the SDTM-IG [5] which has recently been published as a ... PDF document. We did not use such a structure for the "Variable Label", as the regulatory authorities, even in Japan, still require the variable labels to be submitted in English, which is surely related to the incapability of the SAS-XPT format to use non-ASCII characters.

On the SDTM variable level, we added a few new data structures. Besides the SAS-XPT data type ("char" or "num") we also added the recommended XML data type, such as "datetime" for any of the "DTC" (date-time of collection) variables. For the controlled terminology, we used a "Type" attribute to indicate whether the controlled terminology is supposed to be "sponsor defined" (indicated by a "*" in the original PDF), or whether CDISC controlled terminology needs to be used. In case of a fixed value, such as for the domain abbreviation, we used a child element "FixedValue", containing the fixed value, such as "DM" for the "DOMAIN" variable in the "Demographics" domain. Furthermore, for the case of CDISC controlled terminology, both the codelist name (e.g. "SPECCOND") as well as the NCI code for that codelist is contained.

An extra element added is "Rules". It is intended to contain a set of single "Rule" elements in a machine-readable code, or at least pseudo code. For example, this can be used to embed the "SDTM conformance rules" published by CDISC in 2017 as an Excel worksheet with a "precondition" and a "condition" in pseudo code for each rule. On the longer term however, the goal is to embed fully machine-readable rules, such as have been developed by the "Open Rules for CDISC" initiative in the XQuery language [6]. This will especially become important when FDA and PMDA finally start accepting submissions in XML format.

For now, a few rules have been added in pseudo code, for example, the rule that the combination of USUBJID and --SEQ must be unique within the data set:

PhUSE EU Connect 2018

```
<Variable Name="COSEQ">
  <VariableLabel>Sequence Number</VariableLabel>
  <SASXPTDataType>Num</SASXPTDataType>
  <RecommendedXMLDataType>positiveInteger</RecommendedXMLDataType>
  <Role>Identifier</Role>
  <ControlledTerminology/>
  <NCICodeList/>
  <Core>Permissible</Core>
  <CDISCNotes>Sequence Number given to ensure uniqueness of subject records within a domain.
    May be any valid number</CDISCNotes>
  <Rules>
    <Rule ID="RULE.COSEQ">
      <RuleText>Must be unique within the subject</RuleText>
      <FormalExpression>count(combination(COSEQ,USUBJID)) = 1</FormalExpression>
    </Rule>
  </Rules>
</Variable>
```

Further rules still need to be added both on the variable, as well as on the domain and the overall level.

The "assumptions" published in the SDTM-IG have been structured into "AssumptionSet" and "Assumption" elements. This was not very easy, as these are less well-structured parts of the IG, where it also looks as that different authors used different styles, and the emphasis was mostly on presentation (e.g. using bullets) rather than on structure. Currently, each "Assumption" element essentially only contains text, with some presentation elements such as using an "AssumptionPoint" element, for indicating this was a bullet point in the original text. No effort was yet done to transform any of the "assumptions" in machine-executable language.

For some of the domains, we also added the examples. As many of the examples are semi-structured and contain one or more tables, we decided to use XHTML for structuring and formatting. So each table is represented by an XHTML "table" element and using "tr" for table rows and "td" for table cells respectively.

USE OF THE SDTM-IG IN XML - HUMAN DISPLAY

An SDTM-IG in XML would have the disadvantage that it is not very well human-readable anymore (only for people with XML knowledge). Therefore, an XSLT stylesheet was developed that transforms the XML into HTML. The thus generated HTML can easily be displayed in a browser for human consumption. When developing the stylesheet, great care was taken that the result as displayed in the browser is extremely similar to the display in the by CDISC published PDF, with the difference that the HTML view does of course not have pages. It would also be possible to generate a stylesheet to generate a PDF, but that doesn't make sense - we do not want to encourage people to print out the implementation guide.

The resulting browser display is extremely similar to the PDF display. For example, for the specification of the DS domain:

PhUSE EU Connect 2018

Disposition (DS)

DS - Description/Overview for the Disposition Domain Model

DS - A subject domain utilized for the submission of information encompassing and representing data, vocabulary or records related to disposition. (NCI)

DS - Specification for the Disposition Domain Model

ds.xpt, Disposition - Events General Observation, Version 3.2. One record per disposition status or protocol milestone per subject, Tabulation

Variable Name	Variable Label	Type	Controlled Terms, CodeList or Format	Role	CDISC Notes	Core
STUDYID	Study Identifier	Char		Identifier	Unique identifier for a study	Required
DOMAIN	Domain Abbreviation	Char	DS	Identifier	Two-character abbreviation for the domain	Required
USUBJID	Unique Subject Identifier	Char		Identifier	Identifier used to uniquely identify a subject across all studies for all applications or submissions involving the product.	Required
DSSEQ	Sequence Number	Num		Identifier	Sequence Number given to ensure uniqueness of subject records within a domain. May be any valid number.	Required
DSGRPID	Group ID	Char		Identifier	Used to tie together a block of related records in a single domain for a subject.	Permissible
DSREFID	Reference ID	Char		Identifier	Internal or external identifier.	Permissible
DSSPID	Sponsor-Defined Identifier	Char		Identifier	Sponsor-defined reference number. Perhaps pre-printed on the CRF as an explicit line identifier or defined in the sponsor's operational database. Example: Line number on a Disposition page.	Permissible
DSTERM	Reported Term for the Disposition	Char		Topic	Verbatim name of the event or protocol milestone. Some terms in DSTERM will match DSDECOD, but others, such as "Subject moved" will map to controlled terminology in DSDECOD, such as "LOST TO FOLLOW-UP."	Required

which is almost indistinguishable from the corresponding content of the IG on page 139 of the SDTM-IG v.3.2.

This is very important, as it proves that it is possible to have a machine-readable specification without having to renounce on a human-readable specification. It also means that in future, SDTM-IGs can and should be produced not as office documents, but using a database (relational or any other type) containing all the pieces (variable definitions, rules, assumptions, ...). With a single click, the specification in XML can then be assembled as well as the human-readable representation generated. Using such a "standards development database" would also greatly enhance teamwork, as triggers can be built in for automating quality and consistency tests. This kind of database-based collaboration tool is very common in software development (version control systems) but still seems to be unknown in the CDISC community.

USE OF THE SDTM-IG IN XML - MACHINE READABILITY

The main purpose of having an SDTM-IG in XML is the machine-readability. At this moment, with our prototype, its use is still a bit limited, but will in future become very important once rules and assumptions can and will be added in machine-readable language.

Currently, the SDTM-IG in XML can already be used to:

- Automatically generate template Define-XML files for used in mapping software.
This was especially one of the reasons why for each variable, we did not only add the XPT data type, but also the recommended XML data type, as the define.xml contains metadata for the SDTM submission in a much more granular and especially more precise.
- Be imported into any software that uses the SDTM-IG as a source of information. Until now, information from the SDTM-IG was usually added to software by human inspection, interpretation, and/or copy-and-paste. This lead to very many different interpretations of the standard, which is essentially conflicting with the purpose of a "standard".

Until now, software developers have probably been spending thousands of hours copying information from the SDTM-IG into their software and scripts. This is not only contra-productive, but is also error prone. With an SDTM-IG in XML, this should not be necessary anymore.

Once rules are added to the SDTM-IG in XML in a format that is as well machine-readable as well as "developer-readable", such as the rules implemented in XQuery in the "Open Rules for CDISC Standards" initiative [6], it will finally be achieved that the rules come together with the implementation guide or standard itself, instead of being published years later, and even then in a format that is barely machine-readable (worksheets). This time gap between publication of the standard and publication of the rules has led in the past to the fact that validation rules

PhUSE EU Connect 2018

were developed by organizations strange to CDISC, and were then "promoted" to be a "standard" as regulatory authorities such as FDA and PMDA started using these in their systems, even though many of the implementations were completely wrong or were over-interpretations of the SDTM-IG. If the rules are however published in a machine-readable way within the SDTM-IG itself however and by CDISC, this would mean that CDISC remains the custodian of the rules, with the machine-readable rules becoming the reference implementation.

We currently live in an era where more and more knowledge systems are being developed, with or without machine-learning and artificial intelligence methods. For such systems, the primary requirement is that the information is available in a machine-readable form, and is being structured as much as possible. People and organizations generating SDTM data sets for regulatory submissions often struggle with the SDTM-IG in PDF format. In order to solve an issue, the only way currently available is to perform a manual and visual search in the SDTM-IG. With an SDTM-IG in XML, "smart" software and SDTM knowledge systems can finally started to be developed. Something like: "Alexa, can VSDY be zero or negative?".

CDISC Standards documents should in future not be developed using office documents anymore. Even using more modern technologies such as a Wiki or the JIRA issue tracking software do not fundamentally solve the problem. Only a well-organized, well-structured method based using a database or a metadata repository such as SHARE [7] can lead to standards specifications that are as well machine- as well as human-readable. Our proposed XML structure can also be used as a blueprint for such a database system.

ANNOTATING PROTOCOLS WITH CDISC AND OTHER CONTROLLED TERMINOLOGY

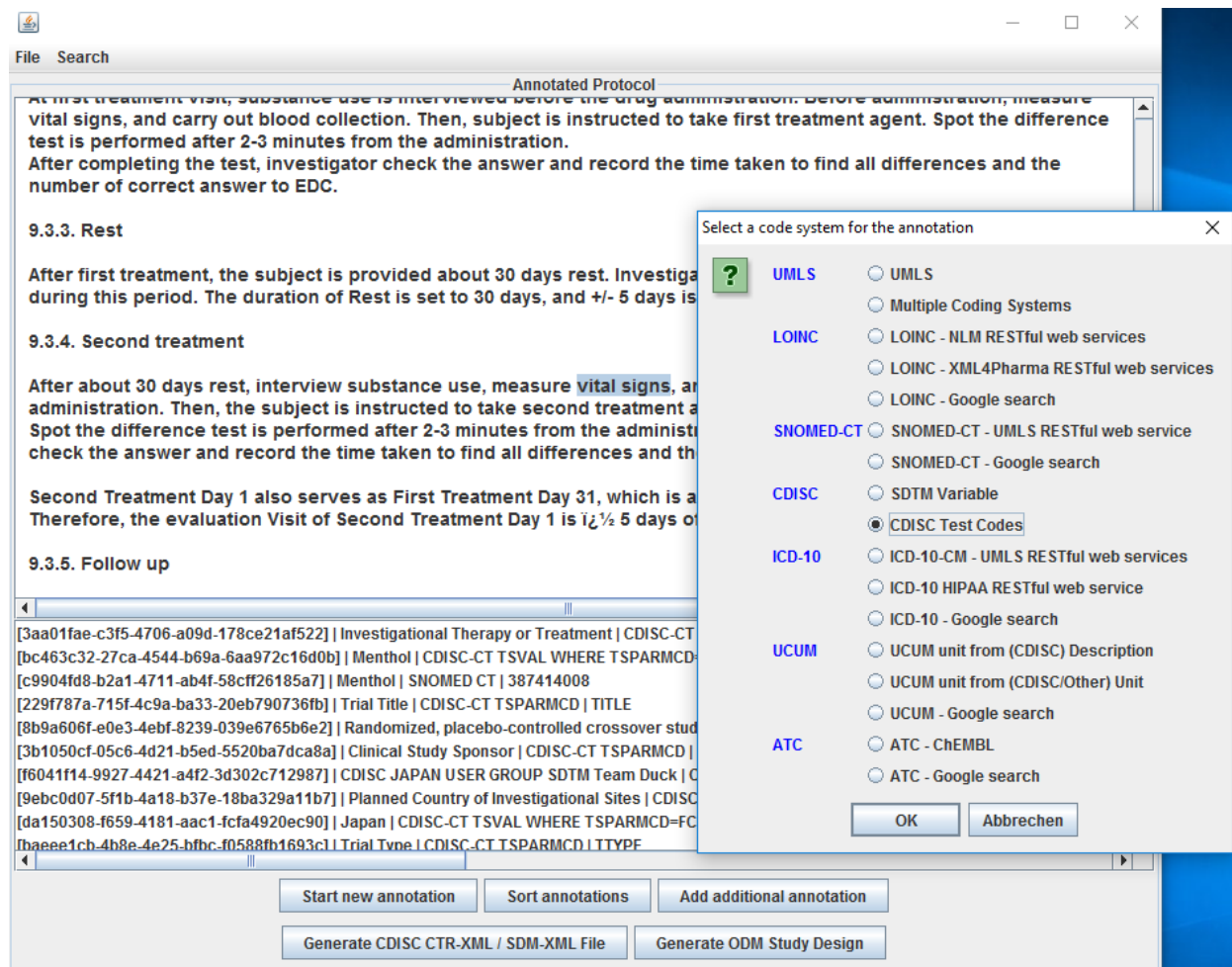
THE PROBLEM

Another problem in clinical research is that study protocols are usually developed as office documents in narrative form, with workflows often as embedded pictures. Humans then still need to translate such a protocol into a study design for the data collection. Many years later, information from the protocol, from the study design, and from the collected data must then be categorized and used in order to create SDTM submission data sets. Essentially, the latter is an ETL (Extract-Transform-Load) process. As protocol writers usually do not have any knowledge about SDTM, the question arises how protocols can be annotated with SDTM information, in order to later facilitate SDTM generation automation. Essentially, this not only applies to SDTM information. Also other types of coded information could be added in order to make the information more precise and thus later improve data quality. For example, if a protocol states that "glucose in urine" should be measured, and it is left to the labs to decide which test is exactly performed, one can expect to receive glucose values that are not comparable between sites, thus jeopardizing data quality. If however, the wording "glucose in urine" is annotated with one or more LOINC codes [8], the latter can be used as precise instructions to the labs which test exactly must be performed. This will then lead to glucose data that is comparable between labs, thus considerably improving data quality, which of course is very important for regulatory review.

METHODS

We developed a "protocol annotation tool" in the Java programming language that starts from the simple text of the protocol. In the tool, the user can select a range of text using the mouse. Once the mouse button is released, the user is asked which coding system to use:

PhUSE EU Connect 2018

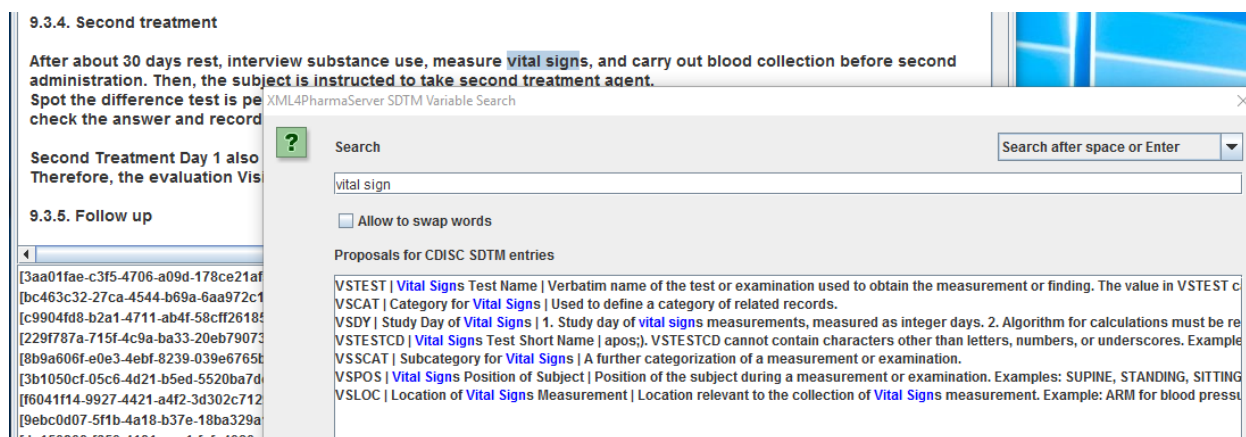


In our tool, the user can choose between over 20 coding systems, ranging from ATC (Anatomical-Therapeutical-Chemical classification system for active molecules [9]) to WHO-ART for adverse reactions [10]. For searching a suitable code for the selected text, the user can then use one of the RESTful web services that were implemented, or a Google search is started. In this paper, we will further concentrate on the use of RESTful web services.

In medical informatics, more and more RESTful web services become available [11]. A good number of them have been developed by the US NLM (National Library of Medicine), but we also developed our own ones, for example for use with LOINC and with CDISC controlled terminology [12]. A very important set of these are the UMLS RESTful web services [13] as UMLS spans a large number of medical coding systems and controlled terminologies, and even allows to use relationships between terms in different coding systems.

Depending on the coding system selected by the user, a query is submitted using the applicable RESTful web service, and descriptions of the returned codes are ranked for similarity with the selected text. The tool then proposes a list of codes and their description, ranked by similarity. In many cases, this leads to the most suitable code being presented first in the list of proposed codes.

PhUSE EU Connect 2018



The user can then select a code for the selected text, which is then stored as XML together with metadata such as the start- and end-position, and this XML is added at the end of the text.

RESULTS

Once the code assigned, the selected phrase in the text is highlighted and color coded. Some of the annotations are key value pairs, such as the text "Planned number of Arms" (key) and the number "2" (value). Others do have a single key and several values, such as "stratification factors" with the values "age" and "sex". Others are just simple codes, e.g. for the text "glucose test in urine" which will usually be assigned one or more LOINC tests. This allows to describe which lab tests exactly need to be executed, or at least suggest the most suitable tests to be performed. This is then restricting the wild variety of different tests that can be found by the human interpretation of the wording "glucose test in urine" (LOINC lists 24 different tests of glucose in urine). This means that it would not be left to the site or even to the lab anymore to decide exactly which test is performed, and which is leading to incomparable results between sites. With such LOINC annotations, the "chaos of data" resulting from different interpretations of the protocol can be overcome, considerably improving data quality, especially for regulatory review.

The current application already allows to fully automate the generation of some of the SDTM trial design data sets, especially the "trial summary" (TS) and the "trial inclusion and exclusion criteria" (TI) data sets. Furthermore, a partial CDISC CTR-XML data file for use of submission to clinical trial registries [14] can be generated. Also important is that it allows to automate the generation of lists of tests that needs to be performed in an exact way. For example, annotation of the text description of lab tests described in the protocol with LOINC annotations would make it possible to automatically generate a "laboratory test form" including the LOINC coding as precise instruction for the lab. We even see the possibility of generating a "template SDTM" from the codes in the protocol in future. The software is regularly extended to add such new features.

As the "annotated protocol" is stored as a machine-readable XML document, we see a lot of potential for its use. Especially in combination with a sponsor metadata repository (MDR), a good part of the study design could be automated, e.g. generating a CDISC ODM file with the study design. Furthermore, when a larger number of protocols of a single sponsor has been annotated in this way, machine learning (ML) technologies could be used to automatically annotate new protocols. Even when this would have a success ratio of e.g. 90%, this would reduce the amount of work to annotate a new protocol enormously.

LIMITATIONS

Protocol documents do not consist of simple text only. They also very often contain tables and figures. Especially when the workflow of the study is depicted as a table or a figure ("schedule of events"), it becomes extremely difficult to extract information from these and transform that into something a computer can read and interpret. Ideally, such workflows or schedules could be transformed into the XML representation of BPMN-2 [15], which is completely machine-readable and allows to set up the study flow in the electronic data capture (EDC) or clinical data management system (CDMS).

CONCLUSION

Our SDTM in XML example demonstrates that it is possible to generate an Implementation Guide in machine-readable form, and from that, using a stylesheet, generate the human-readable representation. It now already allows to generate Define-XML templates, which can be used as starting points for SDTM ETL software packages.

We see our "Protocol annotation tool" and the XML representation only as a first step into coming to an e-protocol. The use of RESTful web services allows to easily extend the tool for other and new terminologies, without the need

PhUSE EU Connect 2018

of locally installing a database. For some coding systems, such as MedDRA [16], no RESTful web services are however available. For this case, we provide an interface to a local MedDRA file system (this requires a license). We have not found a solution yet for parts of the protocol that are not simple text, such as e.g. for tables and figures. As the text and the annotations are stored as a machine-readable XML document, we see a lot of potential for using such annotated protocols in study automation, for automating the generation of submission trial design datasets, and for the creation of template SDTM datasets. We also expect considerably improved data quality, as tests to be performed can be annotated with codes, leading to precise instructions which tests exactly need to be performed. When a sufficient number of protocols is annotated this way, we even expect that machine learning techniques can automate this step too.

REFERENCES

1. <https://wiki.cdisc.org>. Last visited 2018-09-07
2. <https://jira.cdisc.org>. Last visited 2018-09-07
3. <https://www.cdisc.org/standards>. Last visited 2018-09-05
4. S.Hume, J.Aerts, S. Sarnikar and V.Huser, "Current applications and future directions for the CDISC Operational Data Model standard: A methodological review", J. Biomed. Inform. 2016 Apr; 60: 352-62. Available at: <https://www.ncbi.nlm.nih.gov/pubmed/26944737>
5. <https://www.cdisc.org/standards/foundational/japanese>. Last visited 2018-09-08
6. "Open Rules for CDISC Standards" initiative: <http://xml4pharmaserver.com/OpenRulesForCDISCStandards/index.html>. Last visited 2018-09-08
7. CDISC SHARE: <https://www.cdisc.org/standards/share>. Last visited 2018-05-07
8. LOINC: Logical Observation Identifier Names and Codes: <https://www.loinc.org>. Last visited 2018-09-07
9. ATC: Anatomical, Therapeutic, Chemical Classification System: <http://www.who.int/classifications/atcddd/en/>. Last visited 2018-09-07
10. WHO-ART: WHO Adverse Reaction Terminology. <https://bioportal.bioontology.org/ontologies/WHO-ART>. Last visited 2018-09-06
11. J.Aerts, "The Use of RESTful Web Services in Medical Informatics and Clinical Research and Its Implementation in Europe". Study Health Techn. Inf., 2017; 236:80-87. Available at: <http://ebooks.iospress.nl/volume/health-informatics-meets-ehealth-digital-insight-information-driven-health-care-proceedings-of-the-11th-ehealth2017-conference>
12. XML4Pharma Web Services: <http://xml4pharmaserver.com/WebServices/index.html>
13. UMLS REST API Home Page: <https://documentation.uts.nlm.nih.gov/rest/home.html>
14. CDISC Clinical Trial Registry XML: <https://www.cdisc.org/standards/data-exchange/ctr-xml>
15. Business Process Model and Notation v.2: <https://www.omg.org/spec/BPMN/2.0/>
16. MedDRA: Medical Dictionary for Regulatory Activities: <https://www.meddra.org/>

ACKNOWLEDGMENTS

Special thanks go to my CDISC colleagues, especially Sam Hume, and my colleague at NIH, Vojtech Huser, for the many inspiring discussions on the idea of an e-Protocol.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

FH-Prof. Dr. Jozef Aerts
Institute for eHealth
University of Applied Sciences FH Joanneum
Eckertstrasse 30i, 8020 Graz, Austria
Email: Jozef.Aerts@fh-joanneum.at - Jozef.Aerts@XML4Pharma.com

Brand and product names are trademarks of their respective companies.