Paper TT17

Transforming Clinical Trials with Linked Data

Armando Oliva, Semantica LLC, Fort Lauderdale, USA

Tim Williams, UCB Biosciences Inc., Raleigh, USA

# Abstract

The pharmaceutical industry continues to be plagued by data integration and management challenges across the clinical trial data life cycle. Considerable progress has been made in recent years with the implementation of CDISC standards. Historically, standards focused on distinct segments of the clinical trial process: study design, submission, publication. To provide a future-proof solution, these standards must be adapted and integrated holistically and consistently across all use cases.

Linked Data provides a potential solution by representing clinical trial concepts at their atomic level, then leveraging ontological classification and rules integration. This paper reports results from the PhUSE project "Clinical Trials Data as RDF." SDTM data was converted to Linked Data based on CDISC and custom ontologies, then reassembled into high-quality, submission-ready data sets. The approach has several advantages, including the inextricable representation of data and their meaning in ways not possible in traditional approaches.

# Introduction

For more than 15 years, we have witnessed the gradual, and more recently, rapid adoption and implementation of CDISC standards. This has in general been a great success story. For example, the implementation of the Study Data Tabulation Model (SDTM) for regulatory submission datasets to the FDA has led to a new generation of automated tools to process and analyze the data resulting in improvements and efficiencies in scientific and regulatory review. During the Dark Ages before data standards, the industry was barely crawling with respect to automated data management and analysis processes, but now we can universally acknowledge that the industry is now walking, and at a fairly brisk pace! However, problems remain. Industry continues to face data integration and management challenges despite the availability of data standards. Inconsistencies in standards implementation is only one of several reasons that we fail to achieve an optimal level of computable semantic interoperability (CSO).

Linked Data is defined as a method of publishing structured data so that it can be interlined and become more useful through semantic queries.[[1]](#footnote-1) Linked Data provides a potential solution by representing clinical trial concepts at their atomic level, then leveraging ontological classification and rules integration. The Resource Description Framework (RDF), a World Wide Web Consortium (W3C) standard, is an established approach to achieve Linked Data solutions. This paper reports results from the PhUSE project "Clinical Trials Data as RDF." SDTM data was converted to Linked Data based on CDISC and custom ontologies, then reassembled into high-quality, submission-ready data sets. This approach benefits from the ability to define concepts computationally so that inconsistencies in implementation can be minimized. The result is the automated creation of highly structured, high consistent SDTM data, thereby essentially removing the high variability in SDTM implementation seen today. The approach has several additional advantages, including the inextricable representation of data and their meaning in ways not possible in traditional approaches.

In the world of clinical trials data management and analysis, Linked Data provides the ability to take the industry to the next level. From barely crawling, to now walking, Linked Data provides the capability to reach the next level. The industry can soon fly.

# The Problem

When one considers the process to create and submit SDTM datasets, one encounters an exceedingly slow, manual process. The instructions on creating valid SDTM datasets are located in human readable PDF documents. It is not unexpected that variability in implementation is widespread as different human interpretations of the instructions are quite common. Worse still, the instructions are scattered across multiple sources and organizations. One must know CDISC models, CDISC terminology, MedDRA, WHO Drug Dictionary, and other standards, and must know how to integrate them holistically. Add on top of this the fact that standards are continuously evolving, often at different paces from one another, and the implementation challenges are magnified. As an example, take the SDTM variable RACE. How to use this variable is described in the SDTM Implementation Guide published by CDISC. The permissible values for RACE are found elsewhere, in the SDTM terminology document made available by the National Cancer Institute Enterprise Vocabulary Services (NCI EVS). The ability to link RACE in one document with the permissible values for RACE in another document is exactly what Linked Data is designed to do, so that an information system can easily link the two and without having to rely on human memory for the link.

As another example, consider the SDTM reference exposure end date (RFXENDTC). This is the last known date of exposure to study medication for a given subject in a trial. It is located in the DM (Demographics) domain. In reality, this is a variable that is derived from individual subject exposure records in the EX (Exposure) domain. Because the derivation is not computable, human error results in values for RFXENDTC that are not consistent with the more granular exposure data in EX. RDF provides the ability to define this concept computationally so that its derivation is consistent and automated across studies.

A third example is the representation of SDTM concepts that have varying definitions across submissions. Two examples are the reference start date and also the treatment emergent flag for an adverse event. These sponsor-provided definitions are often included in a separate define.xml document, but sometimes the details are buried in the protocol or study report, unavailable to automated systems. RDF addresses this problem by providing the ability to link the concept to its computationally valid definition. When pooling data across studies, it is important to understand whether two variables named the same can be pooled. With RDF, the computer can now assist in that determination.

[address additional bullets below]

**The Linked Data Solution**

**Data Conversion**

Data was needed to test the model and create SDTM and DEFINE files. The team chose CDISC pilot data recently updated to SDTM 3.2 by the PhUSE Test Dataset Factory Project ( <http://www.phusewiki.org/wiki/index.php?title=WG5_Project_09> ). Source datasets are downloadable as SAS transport files from GitHub at <https://github.com/phuse-org/phuse-scripts/tree/master/data/sdtm/updated_cdiscpilot> .

The data conversion process and accompanying scripts are not considered project deliverables. They are the by-product of translating the source data necessary to develop and test the model. Having an authentic SDTM data source facilitates round-trip validation from the graph data back to the original format. The team chose CDISC pilot data recently updated to SDTM 3.2 by the PhUSE Test Dataset Factory Project2. Original SAS transport files are available on GitHub3.

The R application was chosen to convert the SAS transport source data to RDF.  First attempts relied on the package rrdf/rrdflibs from Egon Willighagen (<https://github.com/egonw/rrdf/tree/master/rrdflibs>) to create the TTL files. Unfortunately, this package is not part of the Comprehensive R Archive Network (CRAN), is not actively maintained, and can be troublesome to configure. For these reasons the team switched over to the redland R Package (<https://cran.r-project.org/web/packages/redland/index.html> ) in late 2017. The R scripts and their interdependencies proved hard to maintain and update as the model evolved. A simpler approach was needed.

Following the 2018 PhUSE CSS, the data conversion process changed to one that is more scalable and easier to maintain. An overview of the process is shown in **Figure X**. Like the previous methods, R is used to import the source XPT files and perform data manipulation and imputation. Some data necessary to validate the model is fabricated using both R and additional values in CSV files. For example, a value for death date (dthdtc) and death flag (dthdtc) was created in R for one patient in the DM domain even though no deaths were reported in the original data. *Site* and *investigator* information was not in the original data, so it was created in a CSV file since this information is part of most clinical trial data sets.

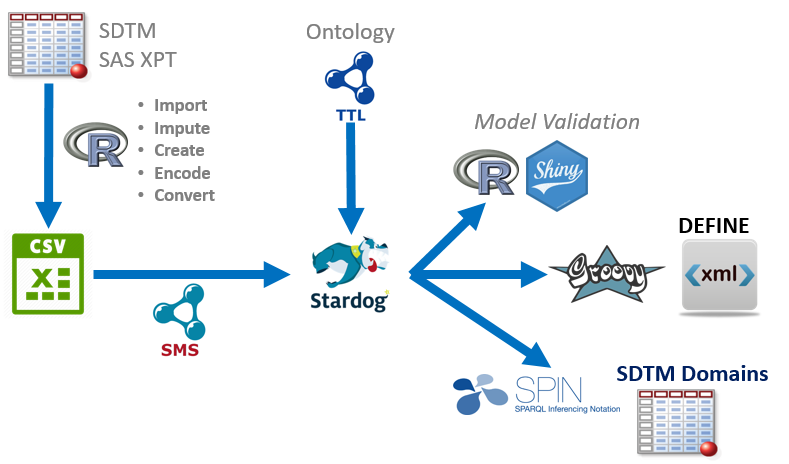


Figure 1 Data Conversion Process

Additional information required in the graph data is not found in the clinical trial results values. This graph-specific data includes information like the start rules that occur prior to making an observation or taking a measurement. Consider the case where a blood pressure is taken in the standing position (vspos=STANDING, in the VS domain) after the subject was previously supine for five minutes. This corresponds to a start rule of StartRuleLying5 in the graph. In the future, these types of rules would be present in the graph as part of the study protocol information. For this prototype project, the additional values for rules are created as part of the data conversion process. Process details are available on the project's GitHub site (<https://github.com/phuse-org/CTDasRDF/blob/master/DataMappingAndConversion.md>). After the data manipulation is complete, the R data frame for each domain is saved as a comma-separated value (CSV) file. The CSV is used as the source data file for mapping values to the graph database.

The W3C standard R2RML "Relational Database to RDF Mapping Language" (<https://www.w3.org/TR/r2rml/>) defines how to map data in a relational or row-by-column format to RDF graphs. Stardog further simplified R2RML as Stardog Mapping Syntax (<http://docs.stardog.com/#_stardog_mapping_syntax> ). SMS mappings can be converted to R2RML, allowing the project to maintain vendor neutrality for the conversion step. Because SMS is based on Turtle, the structure within map file closely resembles the desired TTL structure for the instance data **Figure X**. This allows the team to approach the data conversion from two directions, matching the instance data from the ontology and the source data from the CSV to the SMS file in the middle. When the SMS file is executes, the CVS file is processed row-by-row. CVS column names inside the curly braces {} are replaced with the corresponding value from current row. SMS lacks the ability to perform functions like concatenation, so some pre-processing occurred in the R scripts, including URL encoding of values that include spaces or other characters not valid in IRIs.

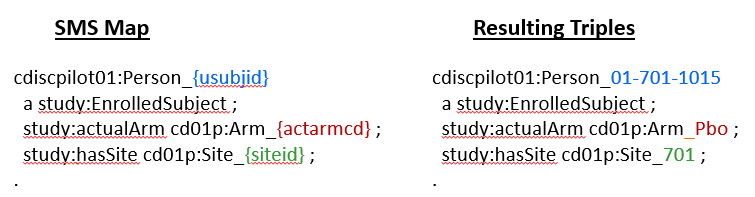


Figure x SMS Map and Resulting Triples

# Other Approaches to Graph Data Conversion

The SAS transport format is an open standard that could, in theory, be supported by SMS or R2RML import approaches. Mapping XPT files directly to the graph would avoid the intermediate steps for imputation and manipulation. Remapping and manipulation could occur within the graph environment. It is unlikely that direct import of XPT files is a desirable first. First, the approach does not match the project philosophy of identifying the atomic entities within the SDTM domains, and mapping those entities to the structure defined in the ontology. The approach of mapping a row-by-column source to the graph, where

the row identifier becomes the *subject*, column identifier the *predicate*, and the cell value the *object* (Sem web reference) is a fast and efficient way to transform data into a graph, but perpetuates many of the problems and limitations inherent in the original structure!

XPT files are commonly created at or near the end of the data lifecycle. In this project the XPT files served as the source of instance data while in an actual real-world scenario the data sources would more likely be from a relational database system (RDBMS). Data could be supplied directly from the RDBMS to the graph using a virtual graph approach, allowing the original sources to remain while the graph becomes the federated data store, bringing together multiple sources with minimal disruption to existing platforms.

# Validation of Converted Data

The process of creating the SMS files that match the project ontology is a largely manual process. SMS TTL files are created by hand using a text editor while cross referencing the ontology, instance data created from the ontology, and the source CSV files. Not surprisingly, errors occur within the SMS TTL files. In the future, it would be very beneficial to have an application that would create the SMS file by visually mapping between an ontology and data source.

The team uses a multi-faceted approach to validate the data created by the SMS mapping, starting with basic SPARQL queries after the data is uploaded to the Stardog triplestore. An R Shiny application is used graphically compare the ontology triples with the SMS-derived instance data using a collapsible tree rendering of a Stardog PATHS query **Figure X**.

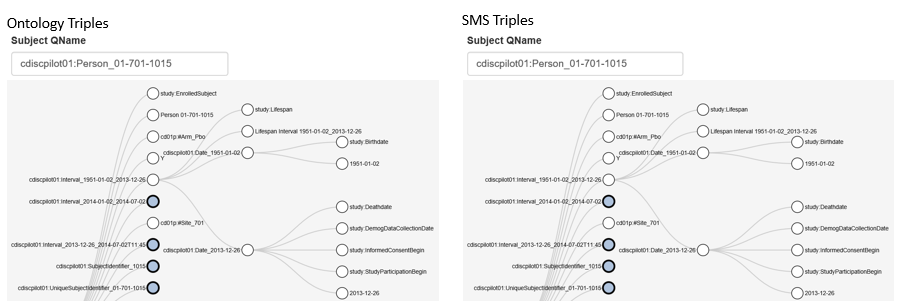


Figure x Collapisible Tree Rendering of PATHS Query

After a visual comparison of the tree structure, another R Shiny application compares the values of the triples programmatically to ensure and exact match. Future validation may include the use of Shapes Expressions (ShEx) or Shapes Constraint Language (SHACL).

# A Main Section

Next section text...

# Another Main Section.

Another section text.

# A Subsection

Subsection text as needed.......

# Another subsection

Subsection text as needed.......

# Data Conversion [TW]

Points to cover:

* original method (rrdf/rrdflibs, redland.. )
* SMS (1) method: what it is, compatible with R2RML (1)
* Data imputation and creation
* Encoding
* Development and validation tools in SPARQL, RShiny. Future: SHACL?
* Future: Why convert at all? Virtual graphs and mapping to source data

Conclusion

Conclusion text...

References [to be updated based on paper content]

1. **Stardog.** Stardog User Manual. *Stardog Mapping Syntax (SMS).* [Online] [Cited: 04 08, 2018.] https://www.stardog.com/docs/#\_stardog\_mapping\_syntax.

2. **W3C.** R2RML: RDB to RDF Mapping Language. *W3C Recommendation.* [Online] [Cited: 04 08, 2018.] https://www.w3.org/TR/r2rml/.

3. **PhUSE Emerging Trends and Technologies.** *Transport for the Next Generation.* s.l. : PhUSE, 2017.

4. *Technical Rejection Criteria for Study Data - Preliminary Findings.* **Allard, Crystal.** Boston : PhUSE Single Day Event (SDE), 2017.

5. *Managing Study Workflow Using the Resource Description Framework (RDF).* **Oliva, Armando.** Endinburgh : PhUSE Annual Conference, 2017.

6. **CDISC.** About CDISC. *CDISC Website.* [Online] [Cited: 05 01, 2017.] https://www.cdisc.org/about.

7. *State of the Union: The Crossroads of CDISC Standards and SAS' Supporting Role.* **Decker, Chris.** Las Vegas, Nevada : SAS Global Forum 2011, 2011.

ACKNOWLEDGEMENTS

The authors are indebted to the contributions from the "CTDasRDF" PhUSE project team members.

This paper is largely based on free, open-source software and the efforts of volunteers in PhUSE working groups. Please support those who donate their time and expertise through your own collaboration, participation, and promotion of these activities.

Contact Information

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

|  |  |
| --- | --- |
| Armando Oliva, M.D.  Semantica LLC  Fort Lauderdale, FL, USA  aoliva@semanticallc.com  C:\_sandbox\sas\Conferences\PhUSE-SDE-CARY2015\images\Twitter_logo_blue.png @nomini  https://www.linkedin.com/in/aolivamd | Tim Williams  UCB BioSciences, Inc  Raleigh, NC, USA  tim.williams@ucb.com  C:\_sandbox\sas\Conferences\PhUSE-SDE-CARY2015\images\Twitter_logo_blue.png @NovasTaylor  <https://www.linkedin.com/in/timpwilliams> |

All project files, data, and this paper are available from the project's Github repository: <https://github.com/phuse-org/CTDasRDF>. Study instance data: <https://raw.githubusercontent.com/phuse-org/ctdasrdf/master/data/rdf/cdiscpilot01.ttl>

Brand and product names are trademarks of their respective companies.

1. See <https://en.wikipedia.org/wiki/Linked_data> [↑](#footnote-ref-1)