PharmaSUG 2021 - Paper SI-046

PROC FUTURE PROOF v1.1- Linked Data

Amy Gillespie, Merck & Co., Inc., Kenilworth, NJ, USA;

Susan Kramlik, Merck & Co., Inc., Kenilworth, NJ, USA;

Suhas Sanjee, Merck & Co., Inc., Kenilworth, NJ, USA;

Jindřich Mynarz, MSD Czech Republic, Prague, CZ;

Danfeng Fu, MSD China, Shanghai, CN

ABSTRACT

As critical contributors to regulatory submissions, manuscripts, and statistical analyses, clinical trial programmers author programming code and leverage programming standards to produce deliverables in a validated, efficient, and reproducible manner. With the function having remained relatively constant for more than 20 years, there are potential opportunities and a need to transform the clinical trial programming role for continued success. Last year we published a paper that evaluated recent advances in technology and the clinical trial programming skillset to identify opportunities for improved programming efficiencies and compliance to regulatory requirements while ultimately optimizing the programming function. Use cases leveraging natural language processing and linked data were explored to determine potential value these digital technologies could add within clinical trial programming processes. This paper shares an overview of steps and challenges building a linked data proof of concept and a readout. The linked data proof of concept is linking analysis results end to end from clinical study reports to source data.

INTRODUCTION

As part of a five-year strategic initiative, our company is utilizing multiple approaches to building digital capabilities. In the previous paper, use cases were introduced for linked data and natural language processing in clinical trial programming. The use cases included document driven programming, consistency cross-checking between various submission documents and source tables, identifying standardization opportunities, and linking analysis results in CSRs and manuscripts.

As a next step at our company, we have been able to develop proofs of concepts based on those use cases. The opportunity for the proofs of concepts manifested as a partnership with our IT organization. The goals are increased operational efficiency, improved regulatory compliance, and alignment with industry activities such as PhUSE working groups and CDISC 360.

The CDISC 360 effort initiated approximately 2 years with a goal to implement standards as linked metadata with a conceptual foundation providing the additional semantics needed to support metadata-driven automation across the end-to-end clinical research data lifecycle[1]. While the CDISC 360 effort is directly relevant to our internal efforts, we have also been inspired by linked data applications across other industries. The identification of these key industry trends and precedents has framed our innovation strategy, supported our learning, and influenced our proofs of concepts. A few of the cross-industry examples we found relevant are described more fully below.

A familiar result of the application of linked data is the display returned from a Google search. The information returned is from a knowledge graph. Linked data is necessary to develop the underlying knowledge graph that ultimately results in the displayed search result.

Facebook uses an open graph protocol, which employs connections and relationships between individuals and between individuals and other entities. "It is based on the standard RDF specification for linked data and includes basic and optional metadata, as well as different types of structured data about objects, of which music and videos are the most well-defined.'[2]

Similarly, LinkedIn uses a knowledge graph. "LinkedIn's knowledge graph is a large knowledge base built upon 'entities' on LinkedIn, such as members, jobs, titles, skills, companies, geographical locations, schools, etc. These entities and the relationships among them form the ontology of the professional world and are used by LinkedIn to enhance its recommender systems, search, monetization and consumer products, and business and consumer analytics."[3]

Another example linked data application is in the ProvCaRe framework, which was developed with an aim to aid in study design and reproducibility. This is to support the principles for reproducible science put forth by NIH in 2014 and the Population, Intervention, Comparison, and Outcome (PICO) framework for clinical research questions. It was designed to show data provenance, and it could be helpful in assessing data quality.[4]

A fifth example of linked data application is in the EHR4CR project. The "EHR4CR project aims to develop an integrated reusable solution to seamlessly connect existing clinical research platforms and healthcare networks across multiple European countries and legal frameworks." [5]

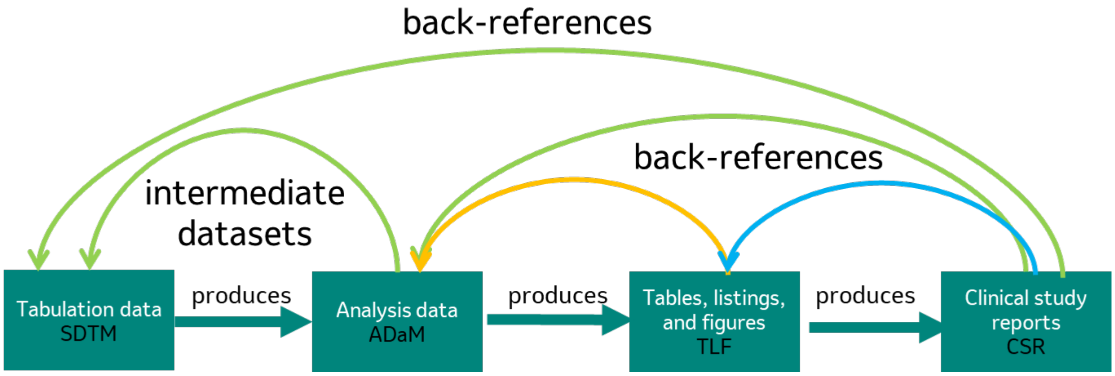
In addition to the proofs of concepts, as part of the five-year strategic initiative, our company is engaging in an academic collaboration. It is another means of assessing and building digital technologies to improve our efficiency and compliance. Natural language processing (NLP) in conjunction with knowledge graphs are being considered in the academic collaboration. The technology could potentially automate and simplify the process of generating standard analysis tables. These techniques may also be used to streamline quality and consistency checks of our deliverables by removing manual efforts to check numbers and text across tables, graphics and documents (protocol, CSR, reviewers guide, manuscripts etc.). Capitalizing on NLP within our company's current analysis and reporting process can benefit statisticians and statistical programmers by reducing the burden to complete manual, tedious tasks and improving efficiency and quality.

The building of and the readout from the proof of concept of linking analysis results to datasets and study reports is the focus of this paper. The build and readouts from the remaining proofs of concepts are planned for future papers.

THE CASE FOR LINKED DATA

Clinical trial Analysis & Reporting is a complex process involving several data transformation steps as shown in Figure 1.

Figure 1.



Transformation from SDTM to ADaM and ADaM to TLFs is programmatic whereas CSR authoring is manual. Traceability is a regulatory requirement and an important aspect in clinical trial A&R. In the current state, the traceability between SDTM and ADaM is accomplished through metadata and by including some variables and observations from SDTM in ADaM datasets. Also, there is no straightforward way in the current state to link the analysis results referred in the study reports to the TLFs delivered by clinical trial programmers. One needs to open TLFs, program specifications, metadata and datasets to be able to traceback information from study reports all the way back to SDTM.

Linked data using shared identifiers and references to CDISC standards can improve the traceability and quality of deliverables in an efficient, reliable, and automated fashion. Linking analysis results with source data and study reports improves traceability, reduces error-prone manual effort and the potential errors or inconsistencies that manual effort introduces. Linked data allows to capture the provenance of data by explicit links. These can be followed automatically to trace back through the originally unidirectional data flows of clinical data derivations.

Using linked data, we can add back-references to the A&R workflow, and this is represented by the curved arrows in Figure 1. Traceability from the CSR to TLFs (blue arrow) was demonstrated in our earlier paper[6]. Traceability from TLFs to ADaM datasets (yellow arrow) was demonstrated by the PhUSE working group project “Analysis Results and Metadata in RDF”[7]. The focus of this paper is to demonstrate automated traceability (green arrows) from

1. ADaM to SDTM by providing users with the contributing variables & observations from SDTM that were used for deriving variables in ADaM
2. CSR narratives to datasets (SDTM/ADaM) by allowing authors to insert URIs from SDTM and ADaM into the reports rather than copy/pasting the results

BUILDING THE PROOF OF CONCEPT

A high number of proofs of concepts do not make it to production. Some of the challenges in going from data-driven proof of concept to production include organizational, project, data, and infrastructure.[8] Applying linked data techniques for efficiency and traceability in clinical trial analysis and reporting is relatively new and uncharted territory. Some of the challenges in the cited paper going from proof of concept to production were also experienced during the initiation and conduct of our project. These will be discussed further.

The proofs of concepts were built in collaboration with our IT organization. As the charter was being developed, discussions with our dedicated IT account manager were fruitful. We learned that there is an IT organization in our company that works in AI/machine learning/data science. Fortunately, they were willing and eager to work with us. A rapid prototyping approach was taken.

Initially there was an analysis and reporting learning curve for our IT colleagues. Similarly, there was a learning curve in our department, understanding the rapid prototyping approach and other nuances and needs.   It required an investment of time to share domain knowledge. There was also a large investment needed in securing relevant data to use for building and testing because study data are confidential.

IMPLEMENTATION

The proof of concept used anonymized subsets of ADaM and SDTM datasets from two clinical studies. The ADaM data contained a number of derived analysis variables, and the goal was to automate their traceability to datapoints that contributed to their derivation. Derivation rules for these variables were documented in Define-XML with a mix of natural language and pseudocode. These derivations were not machine readable without nontrivial NLP. The SAS programs that generated ADaM datasets were updated to create intermediate datasets with information about the contributing rows from source datasets. These intermediate datasets helped link the derived variable with the contributing observations. These links enabled generation of dependency graph for each derived variable, mapping its data lineage.

The Define-XML defined how to map the SAS datasets to CDISC standards. In order to transform it to linked data, we converted the Define-XML files to RDF Data Cube Vocabulary's[9] (DCV) data structure definitions. This allowed the SAS datasets to be read as RDF/DCV datasets aligned with CDISC standards. This was implemented as a generic and fully automated process which could be applied to any SDTM or ADaM datasets described by the Define-XML standard. In a similar way, the intermediate datasets were transformed to intermediate RDF annotations that were resolved into links by post-processing with SPARQL UPDATE operations once they were loaded into an RDF store. Amazon Neptune was used as the RDF store, together with several additional Amazon Web Services, such S3 for storage.

The relationship between a derived ADaM variable and its contributing SDTM/ADaM variable(s) was established by providing the contributing variable names in ADaM specification and by creating an intermediate SAS dataset. This SAS dataset included key variables from both source and derived datasets. The same key variable names from source and derived datasets were distinguished by adding different prefixes. Figure 2 shows an example of intermediate dataset for derived variable RANDFL.

The intermediate datasets (example shown in Figure 2) are structured in a way that the variable naming conventions indicate whether a given variable is a key variable to be used for merging with source or derived datasets.

<u>Naming conventions for variables in intermediate datasets:</u>

Prefix “C” indicates that a variable(s) are key(s) for contributing dataset

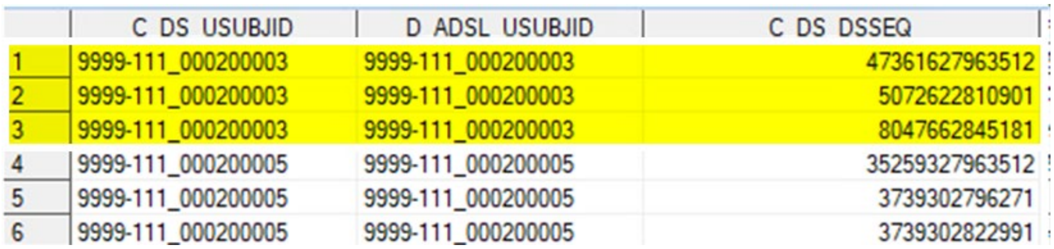
Prefix “D” indicates that a variable(s) are key(s) for derived dataset

Second portion i.e. substring between the first and second underscore indicates the dataset name

Last part i.e. after second underscore indicates the key variable names

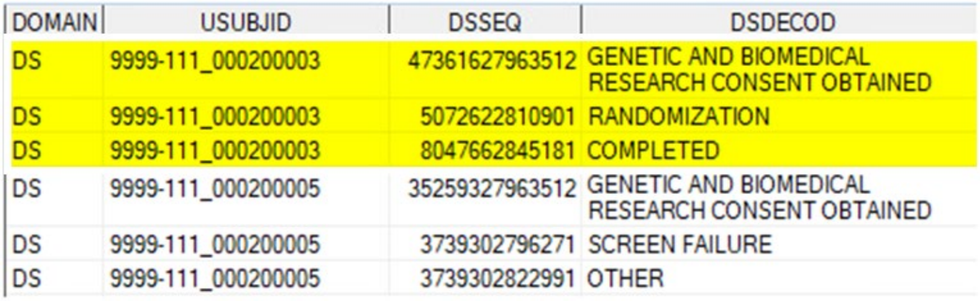
In the below example illustrated in figures 2,3, and 4, the variable names C_DS_USUBJID, and C_DS_DSSEQ indicate that the intermediate dataset needs to be merged with SDTM DS on (C_DS_USUBJID=DS.USUBJID C_DS_DSSEQ=DS.DSSEQ) to identify contributing observations. Variable name D_ADSL_USUBJID indicates that ADSL needs to be merged with the intermediate dataset on (D_ADSL_USUBJID=ADSL.USUBJID)

Figure 2. Intermediate dataset for RANDFL (RANDFL.sas7bdat)



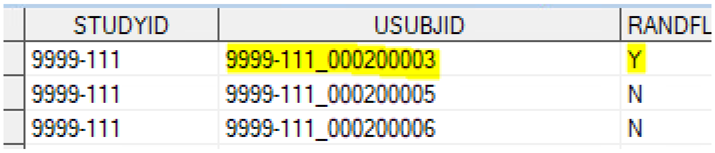
	C DS USUBJID	D ADSL USUBJID	C DS DSSEQ
1	9999-111_000200003	9999-111_000200003	47361627963512
2	9999-111_000200003	9999-111_000200003	5072622810901
3	9999-111_000200003	9999-111_000200003	8047662845181
4	9999-111_000200005	9999-111_000200005	35259327963512
5	9999-111_000200005	9999-111_000200005	3739302796271
6	9999-111_000200005	9999-111_000200005	3739302822991

Figure 3. Contributing SDTM dataset (DS)



DOMAIN	USUBJID	DSSEQ	DSDECOD
DS	9999-111_000200003	47361627963512	GENETIC AND BIOMEDICAL RESEARCH CONSENT OBTAINED
DS	9999-111_000200003	5072622810901	RANDOMIZATION
DS	9999-111_000200003	8047662845181	COMPLETED
DS	9999-111_000200005	35259327963512	GENETIC AND BIOMEDICAL RESEARCH CONSENT OBTAINED
DS	9999-111_000200005	3739302796271	SCREEN FAILURE
DS	9999-111_000200005	3739302822991	OTHER

Figure 4. Derived dataset ADSL with RANDFL



STUDYID	USUBJID	RANDFL
9999-111	9999-111_000200003	Y
9999-111	9999-111_000200005	N
9999-111	9999-111_000200006	N

Several steps need to be taken to create this intermediate dataset for RANDFL in the ADaM generation program.

1. Include USUBJID and DSSEQ form DS domain that are relevant to the derivation of RANDFL
2. Create another variable "D_ADSL_USUBJID" by assigning same value in USUBJID from DS domain
3. Rename USUBJID and DSSEQ to C_DS_USUBJID and C_DS_DSSEQ to indicate that they refer to contributing observations

In order to demonstrate the value of linked data in a more visible way we built a web application with a data browser enabling to navigate through the data in the RDF store via tabular views and arbitrary SPARQL queries for power users. The browser allowed to show contributing data for chosen values of derived analysis variables. It also provided a unique link (IRI) for each value that can be used as an unambiguously reference a datapoint in clinical study reports. We developed an automated script to resolve these references in MS Word documents back to their values via our RDF store's SPARQL endpoint. This served as a demonstration of how to improve the consistency between medical writing and analytical data that was prone to copy/paste errors.

The link between derived dataset and contributing dataset is established by below logic. Refer to figures 2, 3 and 4 above for details about how the links are established. When users click on one value of ADSL.RANDFL from the data browser, the records from ADSL for given value of key variable(s) which is USUBJID = 9999-111_000200003 (highlighted row in Figure 4) in this case will be merged with the intermediate dataset (Figure 2) for RANDFL by the key variables (ADSL.USUBJID=RANDFL.D_ADSL_USUBJID). By extracting the matching observations (shown in highlighted observations in Figure 2), we can find the key variable combinations (USUBJID & DSSEQ) to identify the contributing records from DS (highlighted rows in Figure 3). Contributing variable names are retrieved from the ADaM specification to display in the data browser.

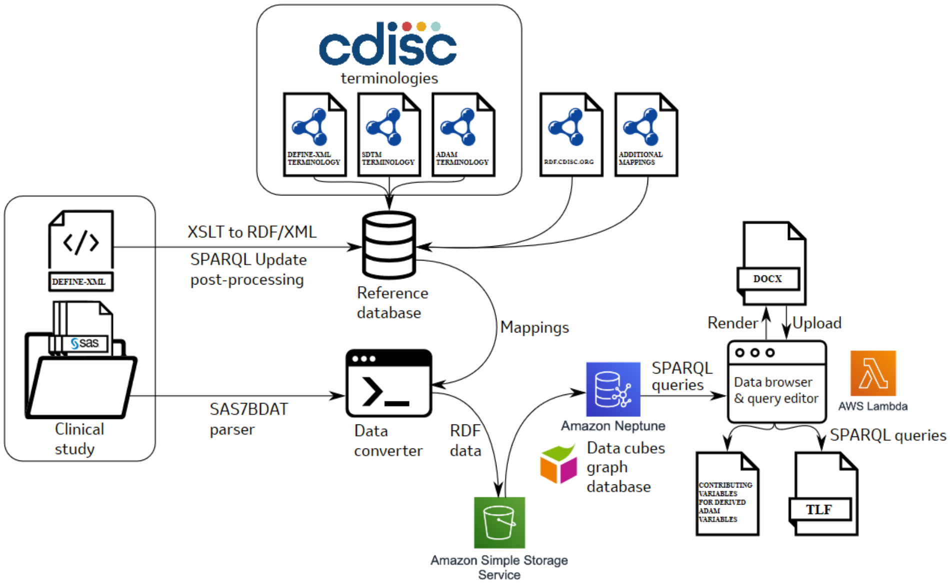
To summarize, using the information from SDTM datasets, ADaM datasets, contributing variable names in ADaM specification, and key variables in intermediate datasets, we successfully built the links between derived dataset/variables and contributing dataset/variables.

PROOF OF CONCEPT READOUT

The proof of concept ran for over 15 weeks, with monthly readouts to discuss progress. We found that frequent *ad hoc* meetings were required between the business subject matter experts and the IT technical team, to clarify and educate. These were resource-intensive, but they contributed to the success in the rapid prototyping model.

Figure 5 shows an overview of the components and the traceability in our linked data and standards approach in this proof of concept. Details from the final readout demonstrated the feasibility of implementing this type of solution and whether to pursue it longer-term.

Figure 5.



Applying current linked data technical tools and capabilities to improve data traceability and report quality is highly technical and relatively novel in clinical trial analysis and reporting. Consequenty, there were challenges.

Challenges encountered were in the design thinking and in agile experimentation. For example, where CDISC standards have complex derivations, the origial proof of concept design was not feasible. The solution using intermediate data and a data browser, described in detail in the previous section was applied to enable continuation of the project. In a production environment, a more robust and scalable method would be needed.

Not all components originally identified were achievable within the timeframe of the project plan and resource availability. Technical challenges and complexity required alternative approaches. The scope of ADaM datasets and variables linked to source needed to be limited. Paired with experimentation, the intensive partnership, communication, and flexibility between collaborators enabled success within a redefined scope.

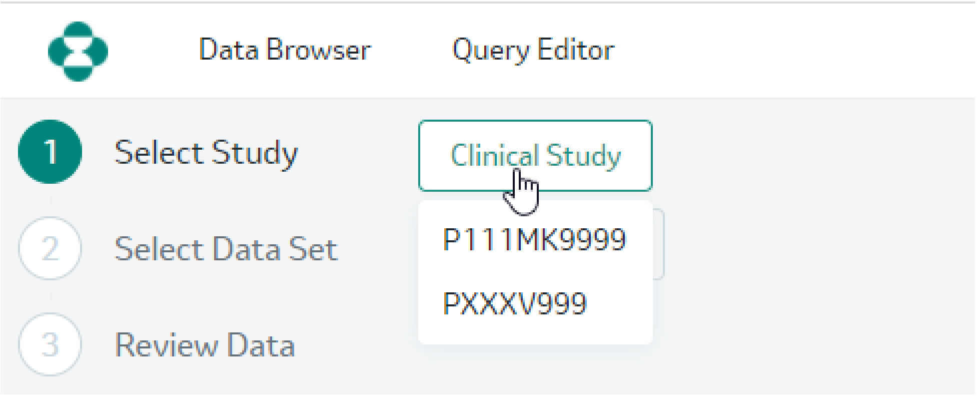
The proof of concept achieved automated traceability between SDTM and ADaM, the ability to reference and de-reference SDTM and ADaM data points in reports, and the ability for power users to perform automated checks using SPARQL queries. The practical applications of these capabilites are:

1) Derived analysis variables can be traced back to tabulation data.
2) Data references placed in reports can be resolved back to data.
3) Exploratory queries can be performed by power users.

RESULTS

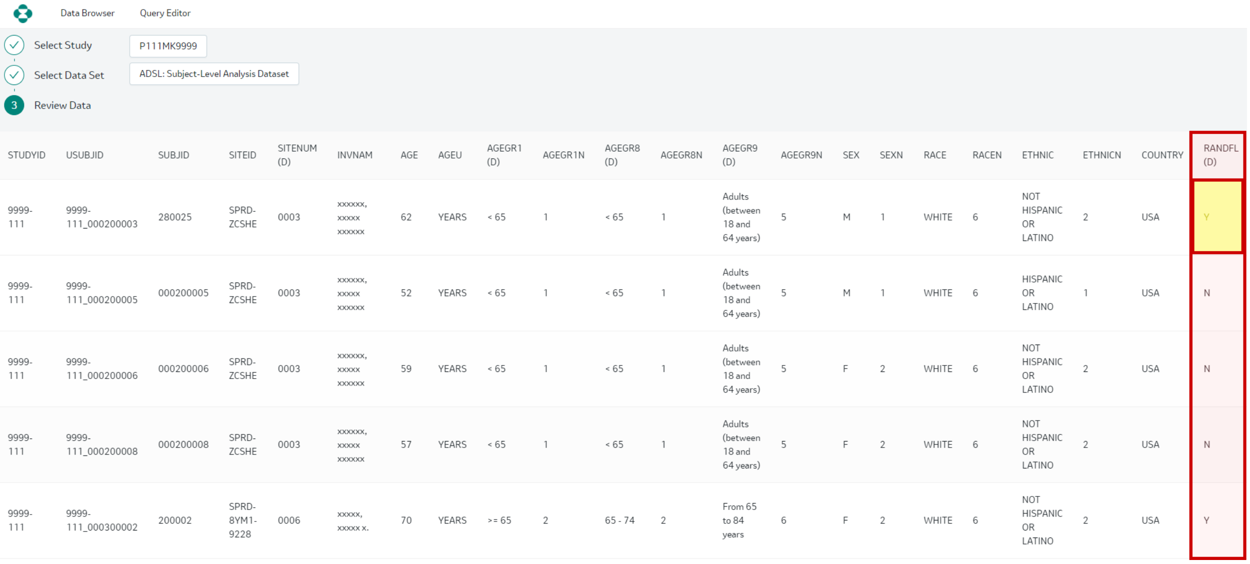
Figure 6 shows the screenshot of the data browser when you start. The user can select the study of choice and dataset within that study to browse.

Figure 6.



Once the user selects the study and dataset, the data browser displays all the variables in the selected dataset. Figure 7 shows portion of ADSL dataset from study “P111MK999”. Let’s take the derived variable RANDFL as an example.

Figure 7.



✓ Select Study: P111MK9999

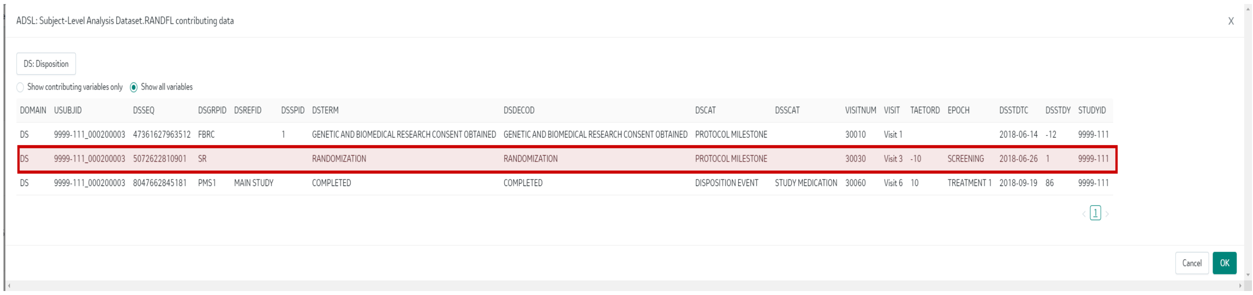
✓ Select Data Set: ADSL: Subject-Level Analysis Dataset

3 Review Data

STUDYID	USUBJID	SUBJID	SITEID	SITENUM (D)	INVNAM	AGE	AGEU	AGEGR1 (D)	AGEGR1N	AGEGR8 (D)	AGEGR8N	AGEGR9 (D)	AGEGR9N	SEX	SEXN	RACE	RACEN	ETHNIC	ETHNICN	COUNTRY	RANDFL (D)
9999-111	9999-111_000200003	280025	SPRD-ZCSHE	0003	xxxxxx, xxxxx xxxxxx	62	YEARS	< 65	1	< 65	1	Adults (between 18 and 64 years)	5	M	1	WHITE	6	NOT HISPANIC OR LATINO	2	USA	Y
9999-111	9999-111_000200005	000200005	SPRD-ZCSHE	0003	xxxxxx, xxxxx xxxxxx	52	YEARS	< 65	1	< 65	1	Adults (between 18 and 64 years)	5	M	1	WHITE	6	HISPANIC OR LATINO	1	USA	N
9999-111	9999-111_000200006	000200006	SPRD-ZCSHE	0003	xxxxxx, xxxxx xxxxxx	59	YEARS	< 65	1	< 65	1	Adults (between 18 and 64 years)	5	F	2	WHITE	6	NOT HISPANIC OR LATINO	2	USA	N
9999-111	9999-111_000200008	000200008	SPRD-ZCSHE	0003	xxxxxx, xxxxx xxxxxx	57	YEARS	< 65	1	< 65	1	Adults (between 18 and 64 years)	5	F	2	WHITE	6	NOT HISPANIC OR LATINO	2	USA	N
9999-111	9999-111_000300002	200002	SPRD-8YM1-9228	0006	xxxxx, xxxxx x.	70	YEARS	>= 65	2	65 - 74	2	From 65 to 84 years	6	F	2	WHITE	6	NOT HISPANIC OR LATINO	2	USA	Y

RANDFL is derived from SDTM dataset DS. The user can get the contributing rows for each subject by clicking on corresponding cell. Figure 8 shows the relevant rows from DS that contributed for this derivation. The user can scroll to see other variables.

Figure 8.



ADSL: Subject-Level Analysis Dataset.RANDFL contributing data

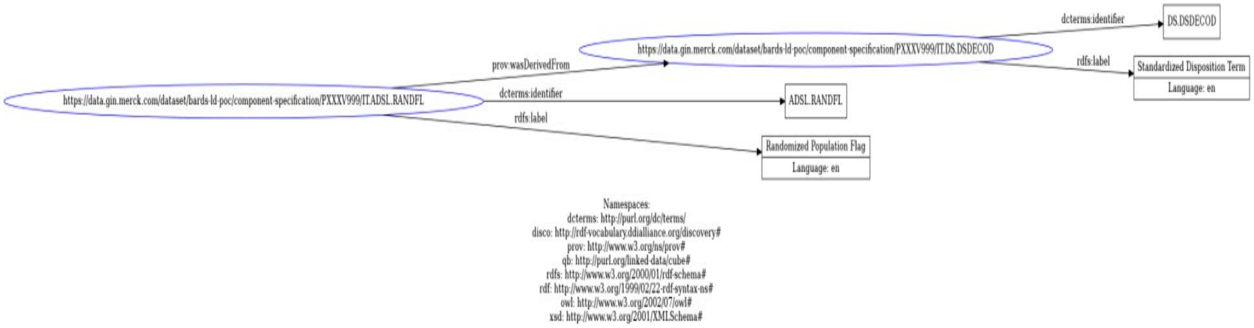
DS: Disposition

○ Show contributing variables only ● Show all variables

DOMAIN	USUBJID	DSSEQ	DSGRPID	DSREFID	DSSPID	DSTERM	DSDECOD	DSCAT	DSSCAT	VISITNUM	VISIT	TAETORD	EPOCH	DSSTDTC	DSSTDY	STUDYID
DS	9999-111_000200003	47361627963512	FBRC		1	GENETIC AND BIOMEDICAL RESEARCH CONSENT OBTAINED	GENETIC AND BIOMEDICAL RESEARCH CONSENT OBTAINED	PROTOCOL MILESTONE		30010	Visit 1			2018-06-14	-12	9999-111
DS	9999-111_000200003	5072622810901	SR			RANDOMIZATION	RANDOMIZATION	PROTOCOL MILESTONE		30030	Visit 3	-10	SCREENING	2018-06-26	1	9999-111
DS	9999-111_000200003	8047662845181	PMS1	MAIN STUDY		COMPLETED	COMPLETED	DISPOSITION EVENT	STUDY MEDICATION	30060	Visit 6	10	TREATMENT 1	2018-09-19	86	9999-111

The data browser also has a query editor that facilitated execution of SPARQL queries. A pre-defined SPARQL query was developed to return a dependency graph for a given variable. Figure 9 shows such dependency graph for ADSL.RANDFL.

Figure 9.


https://data.gin.merck.com/dataset/bards-ld-poc/component-specification/PXXXV999/IT.ADSL.RANDFL
prov:wasDerivedFrom
https://data.gin.merck.com/dataset/bards-ld-poc/component-specification/PXXXV999/IT.DS.DSDECOD
dcterms:identifier
DS.DSDECOD
rdfs:label
Standardized Disposition Term
Language: en
dcterms:identifier
ADSL.RANDFL
rdfs:label
Randomized Population Flag
Language: en
Namespaces:
dcterms: http://purl.org/dc/terms/
disco: http://rdf-vocabulary.ddialliance.org/discovery#
prov: http://www.w3.org/ns/prov#
qb: http://purl.org/linked-data/cube#
rdfs: http://www.w3.org/2000/01/rdf-schema#
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
owl: http://www.w3.org/2002/07/owl#
xsd: http://www.w3.org/2001/XMLSchema#

DISCUSSION

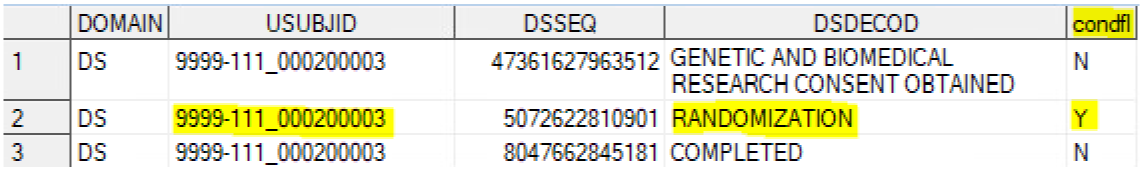
As demonstrated in the previous section, the data browser facilitated

(i) traceability for derived ADaM variables
(ii) traceability for datapoints referred in the CSR narratives to ADaM/SDTM datasets
(iii) generation of dependency graph for a given variable

This is a significant step forward in automating the traceability between SDTM and ADaM. We can think of few enhancements which are discussed here.

In the POC, all the relevant records were included in the intermediate dataset without any sub-setting applied. E.g. When a user clicks on a value of RANDFL variable for a given subject, all the records from input DS dataset for that subject were displayed. This way users could cross check the derivation and were able to determine the actual contributing records based on derivation rules from Define XML. This can be improved by adding a flag to indicate the specific observations that contributed to the derivation. In case of ADSL.RANDFL, only the record with DSDECOD = “Randomization” contributes to the derivation of RANDFL. The user interface can then be designed to display all the relevant records from the contributing dataset with specific observations contributing to the derivation highlighted as shown in in Figure 10.

Figure 10.



	DOMAIN	USUBJID	DSSEQ	DSDECOD	condfl
1	DS	9999-111_000200003	47361627963512	GENETIC AND BIOMEDICAL RESEARCH CONSENT OBTAINED	N
2	DS	9999-111_000200003	5072622810901	RANDOMIZATION	Y
3	DS	9999-111_000200003	8047662845181	COMPLETED	N

CONCLUSIONS

The linked data proof of concept began with a hypothesis that linking analysis results with source data and study reports will improve traceability and reduce error-prone manual efforts. Our overall goal was to facilitate data traceability from a clinical study report back to the initial SDTM datasets.

While we accomplished several objectives during our proof of concept, there is more experimentation needed to fully prove our original hypothesis. Some of our accomplishments included converting SDTM and ADaM datasets to RDF, building a capability to copy links to SDTM and ADaM datapoints into study reports, and developing a user interface to identify the contributing variables and observations for derived ADaM variables. Future proof of concepts will include converting tables and listings to RDF and the development of a data browser to traceback analysis results from tables and listings to contributing records in ADaM datasets.

The linked data proof of concept gave us an opportunity to explore whether some recent advances in technology can be leveraged in analysis and reporting. We feel strongly that this proof of concept is a success regardless of whether these solutions progress toward full implementation. The initial outlay in time and resources was relatively modest and the learnings through the partnership were a great asset and an investment toward future work together.

Amazon's Jeff Bezos has been quoted many times saying, "Our success is a function of how many experiments we do per year, per month, per week, per day." While we are experimenting on a much smaller scale than Amazon, the importance of experimentation through proofs of concepts and the learnings are still critical to our innovation strategy.

REFERENCES

[1]*CDISC 360* (n.d.) Retrieved from https://www.cdisc.org/cdisc-360

[2]Heller, M. (2012). *Real World Semantic Web?: Facebook's Open Graph Protocol ACRL Tech Connect.* Retrieved from https://acrl.ala.org/techconnect/post/real-world-semantic-web-facebooks-open-graph-protocol/

[3]He, Q., Chen, B., Agarwal, D. (2016, October 6) Building *the Knowledge Graph.* Retrieved from

https://engineering.linkedin.com/blog/2016/10/building-the-linkedin-knowledge-graph

[4]Sahoo, S. S., Valdez, J., & Rueschman, M. (2017) Scientific reproducibility in biomedical research: provenance metadata ontology for semantic annotation of study description. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, *2016*, 1070–1079. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5333253/

[5]Curcin, V., Miles, S., Danger. R, Chen, W., Bache, R., Taweei, A. (2014) Implementing interoperable provenance in biomedical research. *Future Generation Computing Systems.* 34, 1-16. Retrieved from https://www.sciencedirect.com/science/article/pii/S0167739X13002653

[6]Gillespie, A., Kramlik, S., Sanjee, SR. (2020) "PROC Future Proof" Presented at PharmaSUG Virtual conference, San Francisco, CA. Retrieved from https://www.lexjansen.com/pharmasug/2020/SI/PharmaSUG-2020-SI-173.pdf

[7]Anderson, M. Hungria, M, Sanjee, SR. (2016) Generating Analysis Results and Metadata – report from a PhUSE CS project. Presented at PhUSE Annual Conference 2016, Barcelona, Spain. Retrieved from https://www.lexjansen.com/phuse/2016/tt/TT05.pdf

[8] Kervizic, J. (2020, June 1) *Challenges moving data science proof of concepts (POCs) to production*. Retrieved from https://medium.com/analytics-and-data/challenges-moving-data-science-proof-of-concept-to-production-458d89b6a9a1

[9] Cyganiak, R., Reynolds, D. (2014, January 16) *The RDF data cube vocabulary*. Retrieved from https://www.w3.org/TR/vocab-data-cube/

ACKNOWLEDGEMENTS

The authors would like to recognize the following subject matter experts from our IT and statistical programming departments who partnered with us and worked diligently on this proof of concept: Raj Thota, Keyur Patel, Majdoub Haloui, Lucie Roškotová, Jan Bárta, Younes Ben Salem, and Jana Vataščinová.