

CDISC Implementation Guides:

Full Digital Transformation Through CDISC Library and API

Authors: Anthony Chow, Executive Director, Strategy & Innovation, Data Science, CDISC, AChow@cdisc.org.
Nic Haydel, Sr. Software Engineer, CDISC, NHaydel@cdisc.org

Background: CDISC Library is composed of 1) Data Standards Browser (DSB) and 2) Application Programming Interface (API). Users can browse and retrieve normative metadata such as CDASH, SDTM, ADaM, QRS, Controlled Terminology, etc. Increasingly, CDISC is focusing on enabling end-to-end process automation and simplifying data transformation throughout the data lifecycle; CDISC makes our data standards available in PDF format. The Data Science team embarked on a proof-of-concept project to digitally transform a few Implementation Guides (IG) and make all applicable contents machine-readable, e.g., informative text, data examples, biomedical concepts, and conformance rules.

Strategy: The project team established these objectives and key results

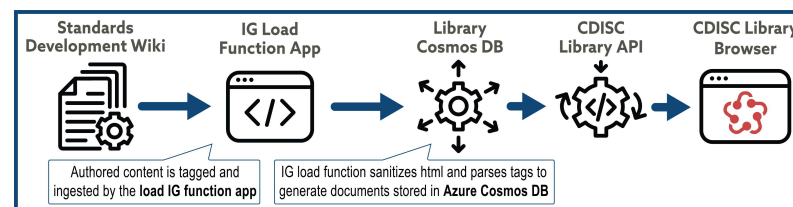
Objectives

- Support both Implementation Guides (IG) & User Guides (UG).
- Accessible to all standards implementers.
- Build on existing tech stack.
- CDISC Library serves as the reference for standard implementers.

Key Results

- Contents acquired through data pipelines are machine-readable.
- Compatible with both new and existing publications on CDISC wiki.
- Content is searchable, discoverable, and bookmarkable.
- HATEOAS-driven RESTful API design.^[1]

Design: CDISC Library makes use of an ETL pipeline to ingest and transform authored content into a generic structure. This normalized structure stores wiki content in both HTML and a stripped-down Markdown format^[2] to support the intended rendering of the content as well as the search functionality inherent in all CDISC Library content. The ETL pipeline processes wiki page labels that are supplied by the standards development team to tag stored documents. These additional metadata tags are used to filter which documents are returned by the API and subsequently displayed in the DSB.



API design was also critical in ensuring the informative content was extensible. Most informative content is well structured into specific sections associated with specific IG structures; however, there is some informative content that does not fit into these categories. Additionally, new IG versions often require new sections or documents. Storing documents in an Azure Cosmos DB, allows us to quickly query documents and filter to support any number of document types and structures. This, coupled with high level API endpoints serving supplemental data about the documents available for a given standard allows CDISC Library to be extensible for all IG types and any new IGs and UGs in the future. Below is an example UI mockup showing how this API design can support an extensible UI for multiple document sections.

Trial Design

TT

Name

Trial Stages

Structure

One record per planned Stage

Trial Stages

Examples

Assumptions

Specifications

This is an example illustrating a study with a single stage of Gestation. The stage begins with confirmation and ends at the scheduled C-section.

tt.xpt

Row	STUDYID	DOMAIN	RSTGCD	RSTAGE	TTSTRL	TTENRL	TTDUR
1	111209	TT	GEST	Gestation Confirmation of Mating Scheduled C-section	Sacrifice	P19D	



General Information



Release Notes

Discoveries: Database design is essential in determining whether a solution supports a use case and is stable enough to support new use cases. Additionally, while out of scope, using Azure Cosmos DB as a storage environment for documents could potentially support standards authoring in the future, given the right toolset. However, using wiki page labels, CDISC Library can be expanded while standards are authored in wiki documents.

Conclusion: Informative content in CDISC IGs and UGs provide needed structure and predictability but flexibility is of high importance. For example, the ability to include new content types. These require careful comprehensive review to optimize storage, efficient content retrieval, and, most importantly, usability. Complementary process updates in standards authoring, such as content label metadata, will improve this transformative feature.