CDISC Library

Sam Hume, Anthony Chow, Mike Hamidi
Data Science, CDISC

21-Feb-2019

Introducing CDISC Library

The metadata repository formerly known as CDISC SHARE 2.0

CDISC Library Launch

- Formerly known as SHARE 2.0
- Built on a new technology stack based on a linked data model
- Includes an expanded API
- Includes new content not previously available
- Broader membership access to the API
- Launch includes lots of supporting activities

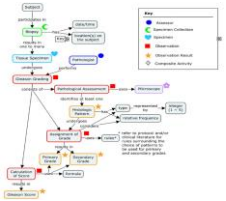
Why Would a Standards Development Organization Use an MDR?

- SDOs generally release “content standards” publications as PDF

Challenges with PDFs that Interfere with Electronic Adoption Scale

Parsing Issues

Extraction Errors



Possible Metadata Loss

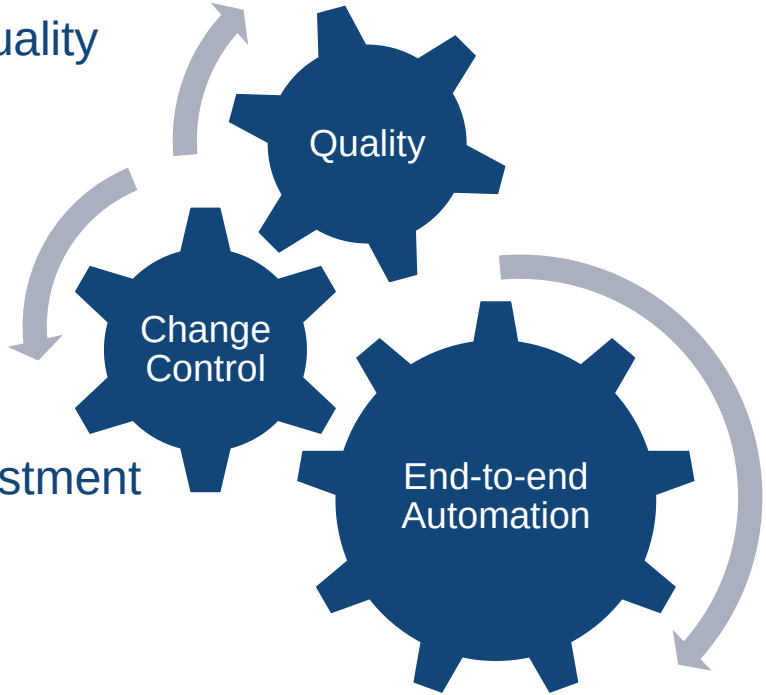
Cost

- Uniquely, CDISC provides electronic standards to support your business processes

Why Use the CDISC Library?

cdisc LIBRARY
formerly SHARE

- Single, trusted, authoritative source for CDISC standards metadata
- Model-based standards metadata improves quality
- Support for an end-to-end standards model
- Maintains version traceability
- API simplifies standards-based automation
- New content published regularly
- Only source for certain CDISC metadata
- Aids in realizing full value from standards investment



cdisc

Expanding API Access

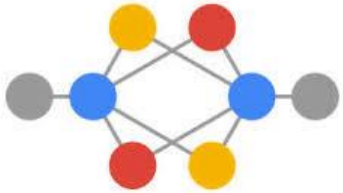
- All members will have access to the CDISC Library API
  - Bandwidth and number of accounts allocated vary by membership level
- Developers will have access to the CDISC Library API
  - Developers for pre-commercial use
  - Open source platforms and others for non-commercial use
- Non-members can access the CDISC Library API through software tools provided by a vendor
- Updating the EULA (terms of use) to make it simpler for vendors to embed the API in their applications

CDISC Library Deployment

- Early sneak peek release to Platinum members on Feb. 20th
- Release to all members on April 10th
- Webinars for all members on how to access and use the API
- CAC meeting presentation March 6th
- New examples and knowledge base articles will be added incrementally

CDISC Library

New API



New model: 3D Standards

Roadmap includes input from CDISC Blue Ribbon Commission Recommendations

Better support for standards teams

New tools

cdisc

Access, Support, & Future Development

Account process, product inquiry, knowledge base, and roadmap

Account Benefits Summary

The CDISC Library API will have key benefits based on the respective membership level and license agreement. The existing accounts and bandwidth allotment may be expanded in the future. As deemed necessary, CDISC will periodically evaluate existing baseline benefits to ensure all members and non-members are adequately pleased with the product. Moreover, to ensure that such limits are maintained for continued performance and access by other customers.

Shown here are the Platinum benefits. Similar benefits will be provided upon full release of CDISC Library on April 10, 2019.

Overview | Why Use the CDISC Library? | Request an Account | API Documentation | Benefits Summary | Upgrades & Overage Fees | Product Inquiry

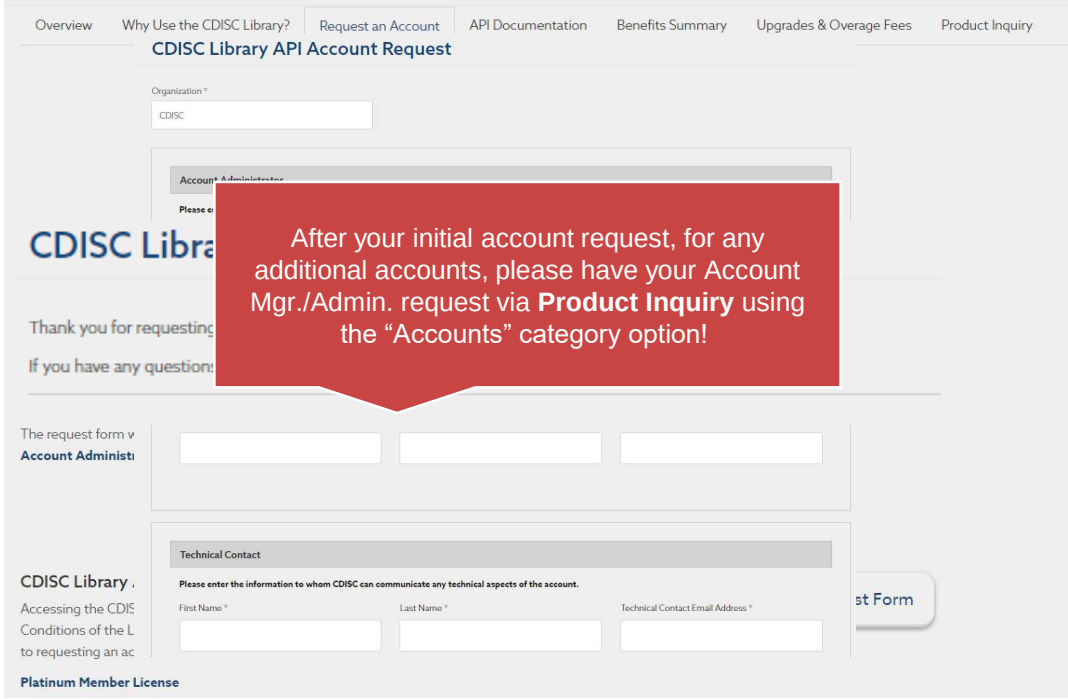
API Benefits Summary

Membership	License	Maximum Allocated Bandwidth (Per Organization)	Maximum # of Accounts (Per Organization)
Platinum	Member	2 GB/month	3
Platinum	Commercial	3 GB/month	4

Disclaimer: Images may not represent the final product and are subject to change.

Account Request Process

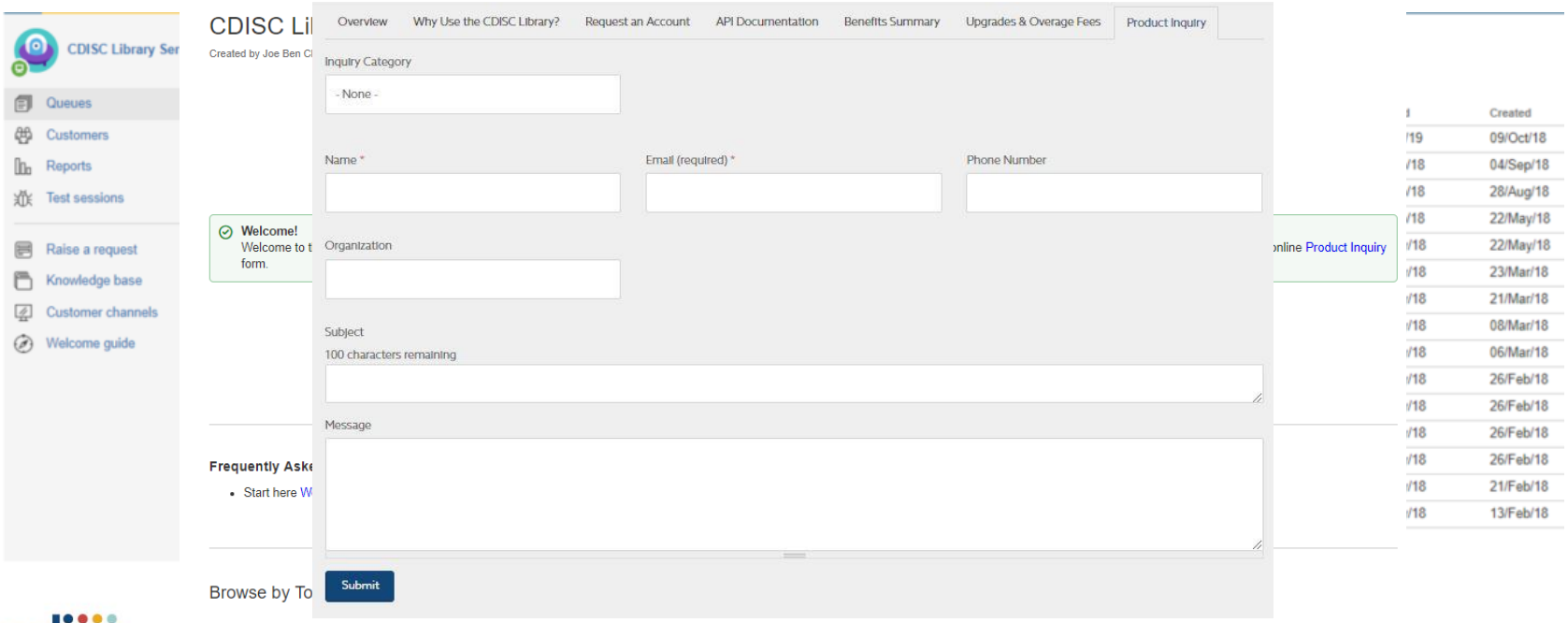
CDISC Library API will be accessible via CDISC website landing page (shown here). Specific details, upon logging into the website, will be made available to the user. It's important that an account requester (on behalf of their organization) reviews and agrees to the licensing terms prior to initiating the account request form. The account request form fields are to be completed, including the requesters' organization account admin./mgr. and IT contact. This ensure that the respective organization is made aware of an account being requested. Once complete and license agreement obtained, the requester will submit their request and receive an initial confirmation screen and email. API credentials and welcome package will also be provided via email upon account creation.



Disclaimer: Images may not represent the final product and are subject to change.

Service Desk System & Product Inquiry

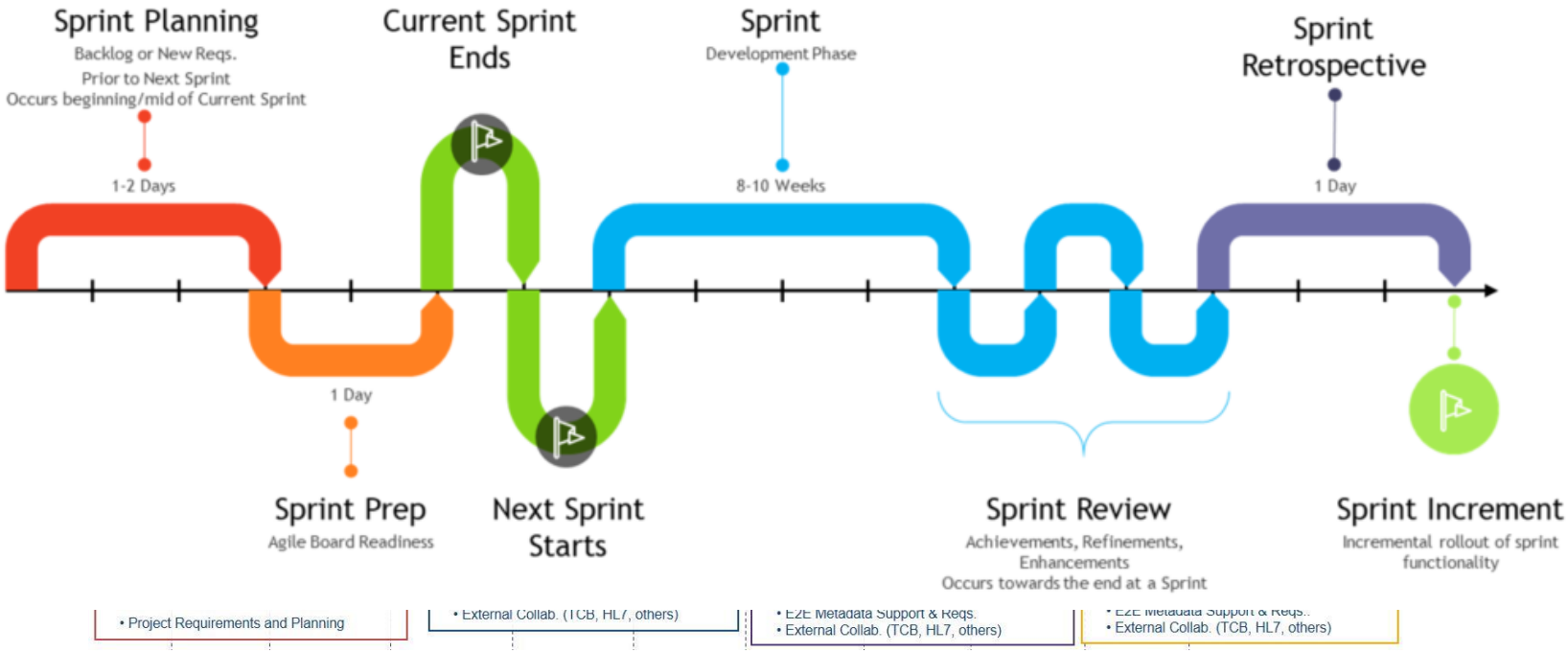
Upon completion of an account request, the requester will be notified via email once their API account has been created, which will include their username and password. CDISC will be utilizing a service desk ticketing system to manage, track and, in certain circumstances, communicate directly with the customer. As mentioned on the previous slide, a welcome package email with links to the CDISC Library service desk and knowledge base will be provided. The knowledge base will serve as a centralized repository where customers can search for solutions and locate various other supporting articles. The community can also provide inquiries (i.e., accounts, enhancements, questions, etc.) via CDISC website using the Product Inquiry form.



Disclaimer: Images may not represent the final product and are subject to change.

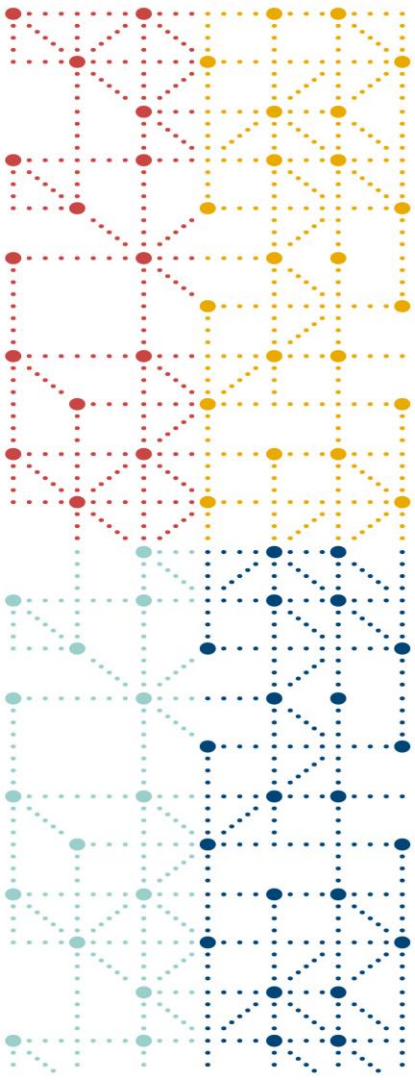
Data Science Roadmap & Scrum Development Methodology

The CDISC data science team has many product requirements we wish to fulfill for our community. A preliminary draft 2019 snapshot is shown, which is a small snippet of a longer 5-year roadmap. The content is subject to change and is not indicative of any final product. In order to achieve such a release cadence, the team will be using an agile scrum development methodology. This will ensure nimbleness and more release functionality and content at a more desirable rate.



Disclaimer: Content is subject to change.

CDISC Library API Technical Introduction



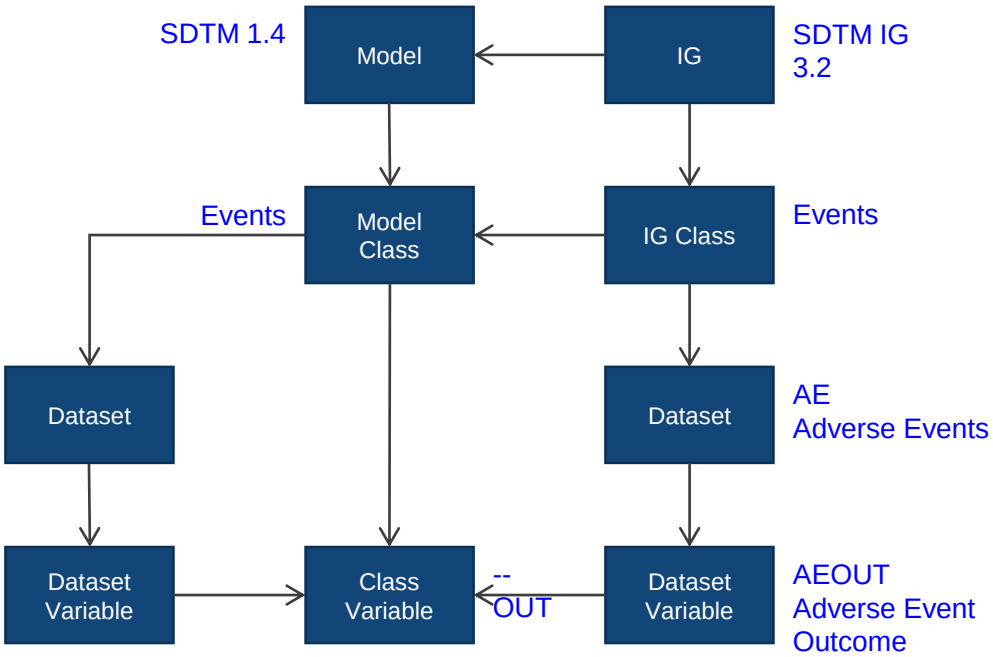
Disclaimer

All materials in this portion are for illustration purposes and may not constitute the final product. Codes, tables, and figures are often cropped for brevity.

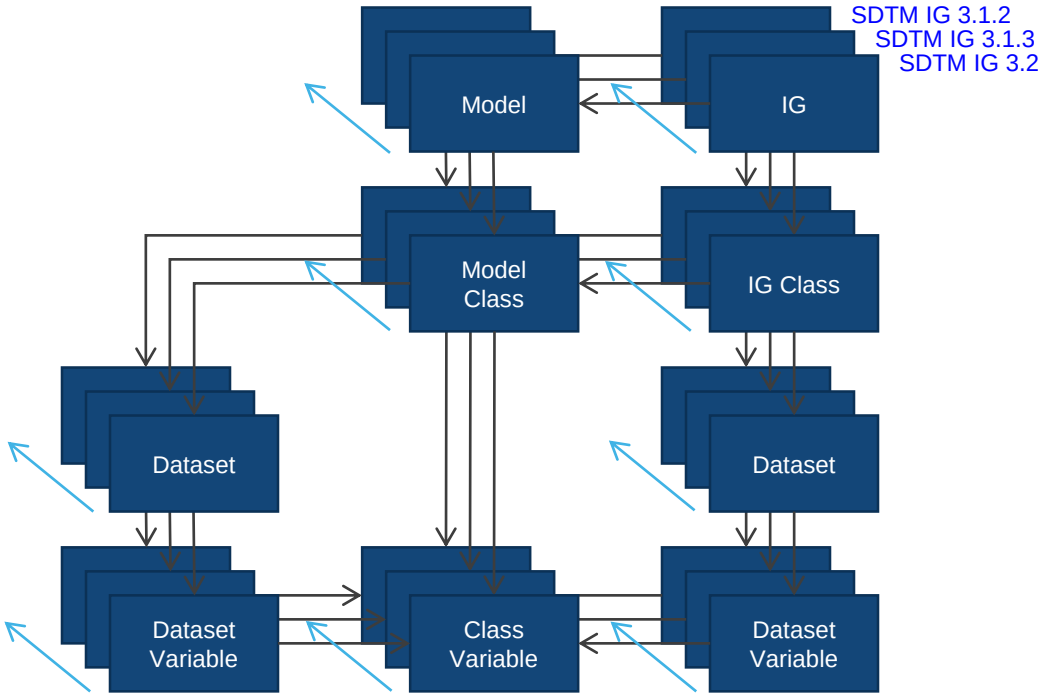
Key Changes in CDISC Library API

- Model
  - Graph based resources
  - ISO 11179 principles applied
  - Meta-model is enhanced, e.g., clear separation between Model and IG, enhanced support for data exchange such as CDISC ODM v2
  - Introduced the root concept for variables, codelists, and terms
- API
  - JSON output
  - Hypermedia deep linking (HATEOAS)
  - No longer driven by asset id

Study Data Tabulation Model (SDTM)



Study Data Tabulation Model (SDTM) Versions



What's Available?

- CDASH v1.1.1, CDASH Model v1.0, CDASHIG v2.0
- SDTM v1.2, v1.3, v1.4, v1.5, v1.6, v1.7
- SDTMIG v3.1.2, v3.1.3, v3.2, v3.3*, SDTMIG-AP v1.0, SDTMIG-MD v1.0, SDTMIG-MD v1.1*
- SENDIG v3.0*, SENDIG v3.1, SENDIG-DART v1.1*
- ADaMIG v1.0, ADaMIG v1.1, ADaM ADAE v1.0, ADaM TTE for BDS v1.0, ADaM OCCDS v1.0
- Controlled Terminology 3rd quarter 2014 (2014-09-26) and onward: 18 quarters, 65 individual parts

Transition from Spreadsheets to API Methods

- Contemporaneity
  - Spreadsheets are a point-in-time snapshot. Example: CDISC Controlled Terminology publication frequency is much higher than Model and Implementation Guide updates
- Intelligibility
  - To convey all the end-to-end aspects of metadata, it will require many worksheets and columns in a spreadsheet: Example: Biomedical concepts
- “3-D” Metadata
  - The two-dimensional nature of spreadsheet is inadequate to illustrate complex, multifaceted concepts

5 Models for Special-Purpose Domains

Demographics (DM)

DM – Description/Overview for the Demographics Domain Model

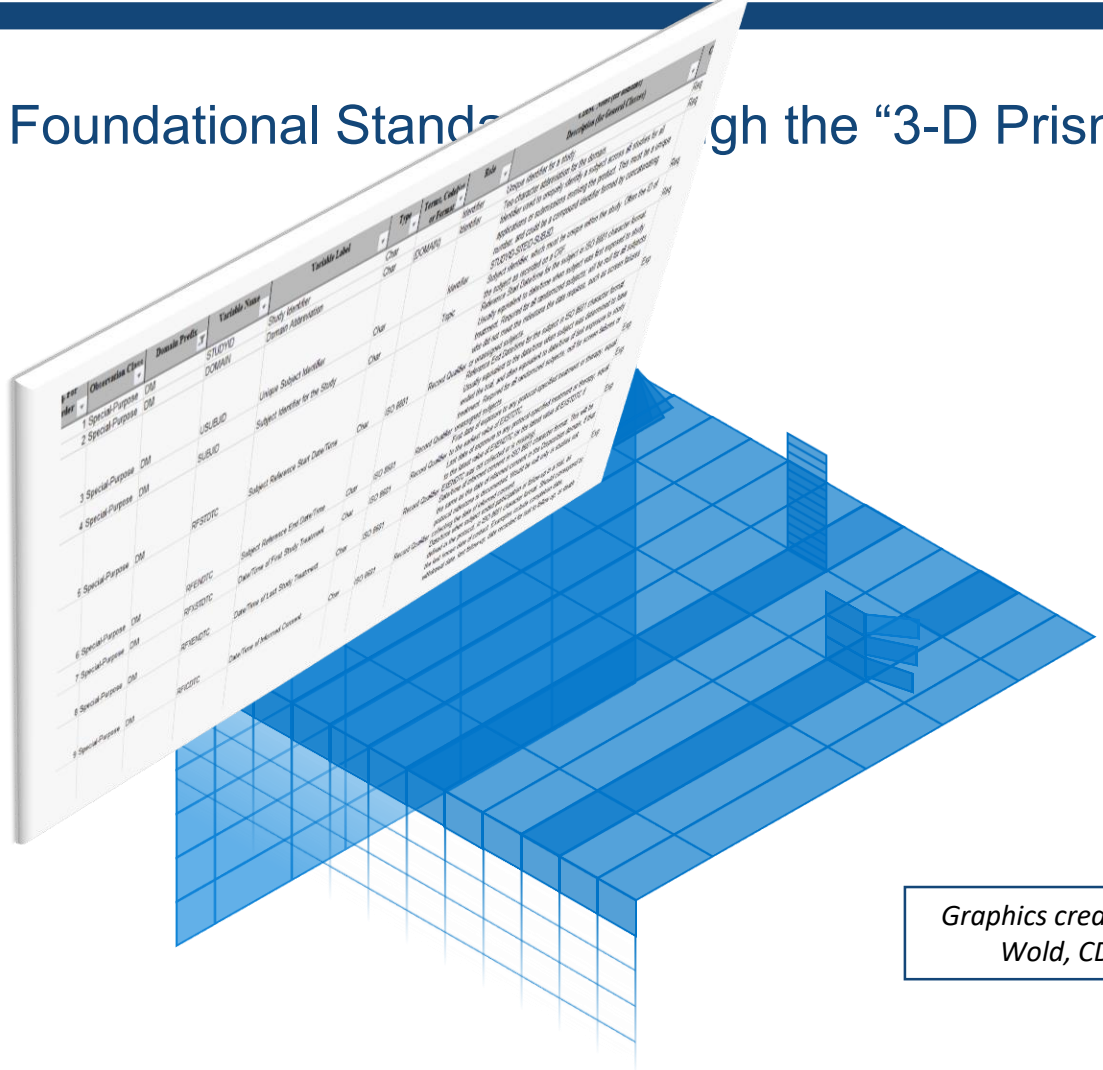
The Demographics domain includes a set of essential standard variables that describe each subject in a clinical study. It is the parent domain for all other observations for human clinical subjects.

DM – Specification for the Demographics Domain Model

dm.xpt, Demographics — Version 3.2. One record per subject, Tabulation

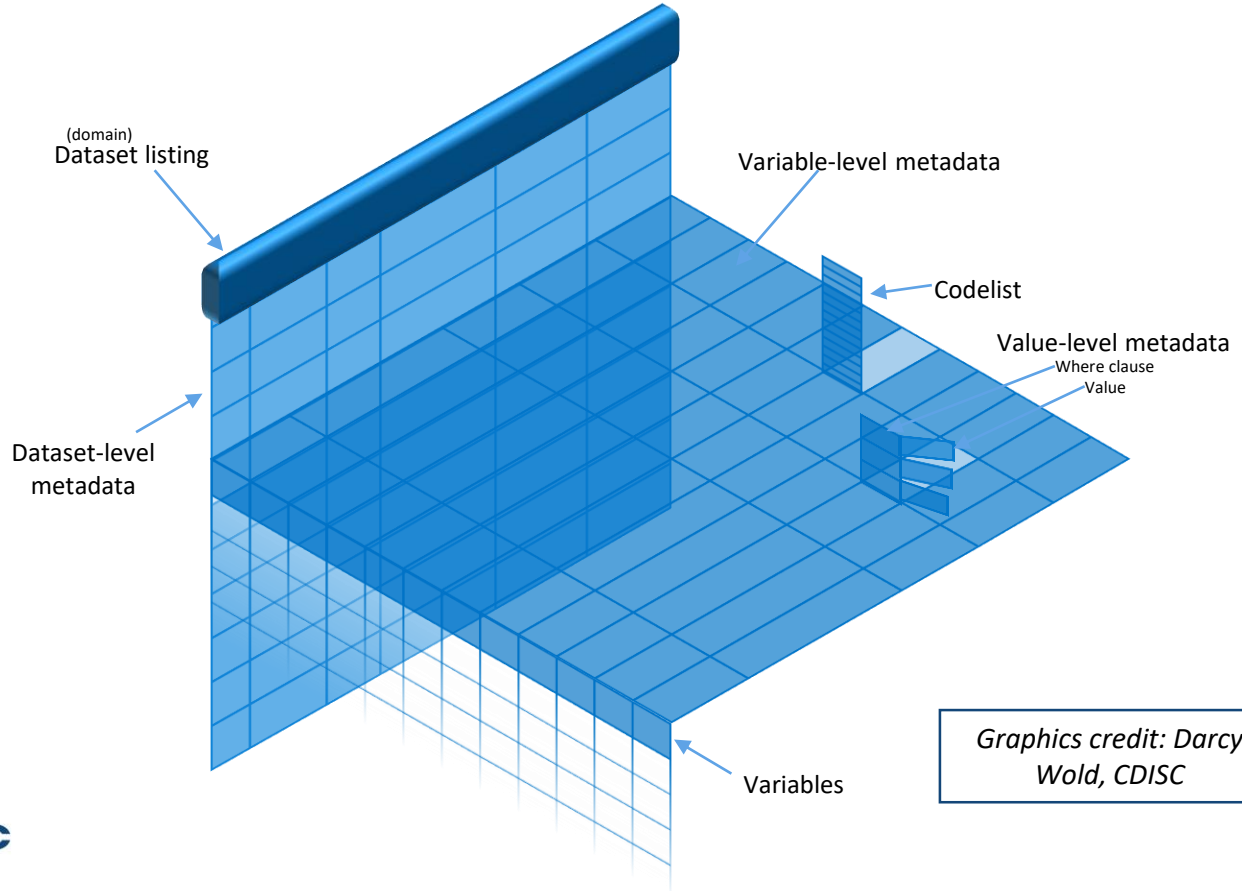
Variable Name	Variable Label	Type	Controlled Terms, Codelist or Format	Role	CDISC Notes	Core
STUDYID	Study Identifier	Char		Identifier	Unique identifier for a study.	Req
DOMAIN	Domain Abbreviation	Char	DM	Identifier	Two-character abbreviation for the domain.	Req
USUBJID	Unique Subject Identifier	Char		Identifier	Identifier used to uniquely identify a subject across all studies for all applications or submissions involving the product. This must be a unique number, and could be a compound identifier formed by concatenating STUDYID-SITEID-SUBJID.	Req
SUBJID	Subject Identifier for the Study	Char		Topic	Subject identifier, which must be unique within the study. Often the ID of the subject as recorded on a CRF.	Req
RFSTDTC	Subject Reference Start Date/Time	Char	ISO 8601	Record Qualifier	Reference Start Date/time for the subject in ISO 8601 character format. Usually equivalent to date/time when subject was first exposed to study treatment. Required for all randomized subjects; will be null for all subjects who did not meet the milestone the date requires, such as screen failures or unassigned subjects.	Exp
RFENDTC	Subject Reference End Date/Time	Char	ISO 8601	Record Qualifier	Reference End Date/time for the subject in ISO 8601 character format. Usually equivalent to the date/time when subject was determined to have ended the trial, and often equivalent to date/time of last exposure to study treatment. Required for all randomized subjects; null for screen failures or unassigned subjects.	Exp
RFXSTDTC	Date/Time of First Study Treatment	Char	ISO 8601	Record Qualifier	First date of exposure to any protocol-specified treatment or therapy, equal to the earliest value of EXSTDTC.	Exp
RFXENDTC	Date/Time of Last Study Treatment	Char	ISO 8601	Record Qualifier	Last date of exposure to any protocol-specified treatment or therapy, equal to the latest value of EXENDTC (or the latest value of EXSTDTC if EXENDTC was not collected or is missing).	Exp

Foundational Standa gh the “3-D Prism”

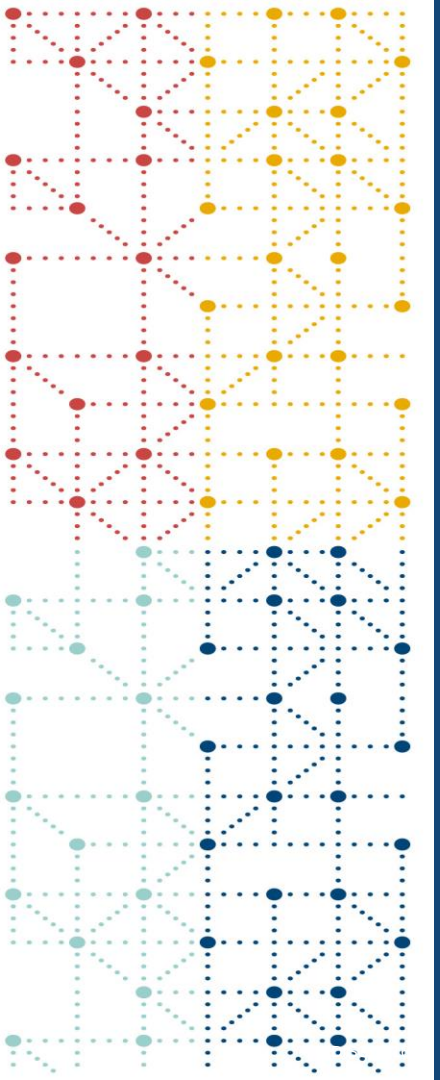


Graphics credit: Darcy Wold, CDISC

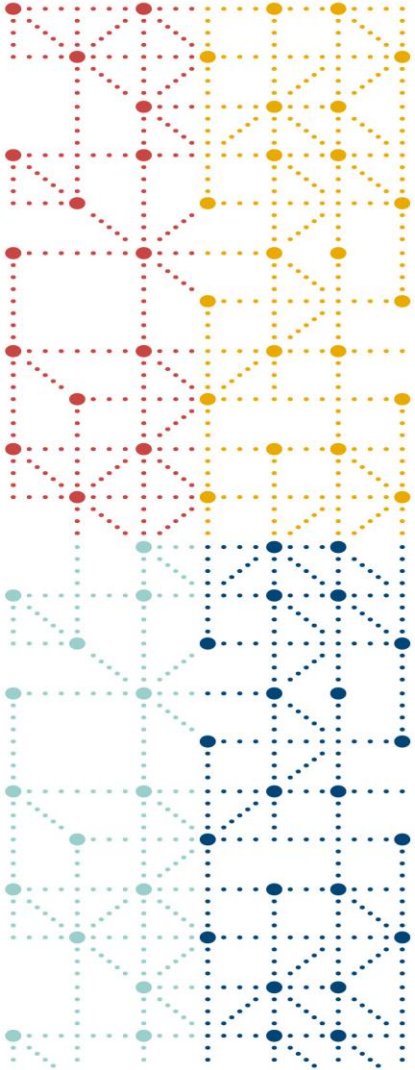
Foundational Standard Through the “3-D Prism”



Graphics credit: Darcy Wold, CDISC



CDISC Library API Deeper Dive

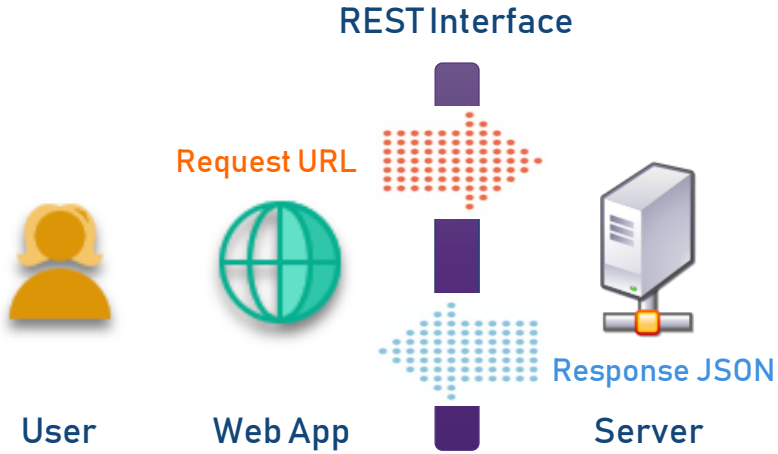


The Basics

Glossary

- **API**, short for application programming interface, is a programmatic method to query & retrieve information, such as metadata stored in the CDISC Library
- **REST API** is a set of widely adopted implementation principles. REST is a shorthand for representational state transfer
- **Base URL** is defined by scheme, host, and base path
- **Endpoint** refers to end of the request URL path, often refers to a collection in plural nouns such as `terms`, `datasets`, `datastructures`
- **JSON**, short for JavaScript object notation, is a commonly used lightweight format standard for data interchange

CDISC Library API Typical Data Flow



Construct of CDISC Library API's Request URL

Term	Meaning
operation	Method to be performed. GET is used for retrieving information
scheme	The transfer protocol of the API
host	The host (name or IP) serving the API
basePath	The base path on which the API is served, which is relative to the host
endpoint	The available paths and operations for the API

JSON response structure (1)

- Objects
  - Surrounded by curly braces { }
  - Written in key-value pair
    - ":" to separate key from value
    - comma is used separate each pair
  - Key is a string, while value must be a valid JSON data type, i.e., string, number, object, array, boolean or null
- Arrays
  - Square brackets [ ] are used to declare JSON array
  - Ordered list of values

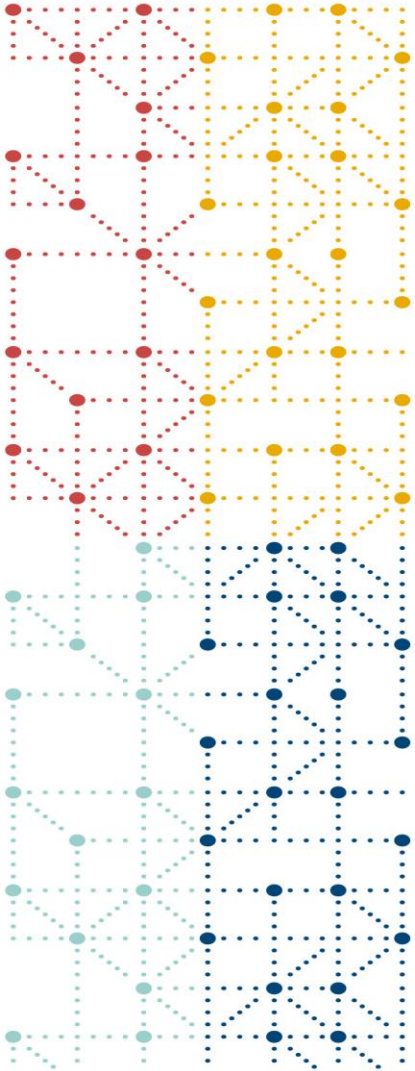
JSON response structure (2)

An API response when querying the CDISC Controlled Terminology term HOURS (C25529), showing how JSON objects and arrays are applied


```
{
   "conceptId": "C25529",
   "submissionValue": "HOURS",
   "definition": "A unit measure of time equal to 3,600 seconds or 60 minutes. It is approximately 1/24 of a med
   "preferredTerm": "Hour",
   "synonyms": [
      "Hour",
      "hr"
   ]
}
```


object; NCI preferred term

array; a list of synonyms



Technical Documentations

- API Design & Documentations
- HTTP Status Codes
- Release Notes

API Design & Documentation

default

GET /products Get SHARE Product List

GET /products/{product-group} Get SHARE Product List by Product Group

Controlled Terminology (CT)

Clinical Data Acquisition Standards Harmonization (CDASH)

CDASH Implementation Guide (CDASHIG)

Study Data Tabulation Model (SDTM)

SDTM Implementation Guide (SDTMIG)

GET /sdtmig/{version} Get SDTMIG Product

GET /sdtmig/{version}/classes Get SDTMIG Class List

GET /sdtmig/{version}/classes/{class} Get SDTMIG Class

GET /sdtmig/{version}/classes/{class}/datasets Get SDTMIG Class Dataset List

cdisc

HTTP Status Codes

These are codes used when the server responds to requests made; CDISC's implementation generally follows HTTP/1.1 guidelines.

Status Code	Meaning
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed
503	Service Unavailable

Release Notes

1. Documentation

News and updates
2019-02-15: Initial version

General

The base URL for the API is:

```
https://library.cdisc.org/api
```

The API uses Basic Auth (HTTP Basic Authentication)

For a machine-readable method to obtain a full listing of products loaded into the metadata repository:

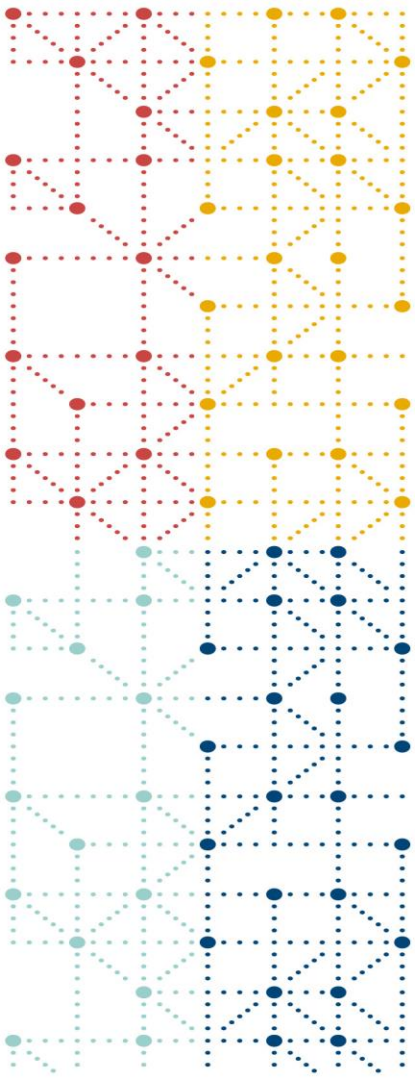
```
/mdr/products
```

Or, by product family:

```
/mdr/products/{product-family}
```

where product-family is one of these values:

- Terminology
- DataCollection
- DataTabulation
- DataAnalysis



HATEOAS

Hypermedia **a**s **t**he **e**ngine **o**f **a**pplication **s**tate

Construct of CDISC Biomedical Concept

Shown is a mindmap describing a medical or health science concept. Each node (a discrete piece of knowledge) is linked to other nodes.

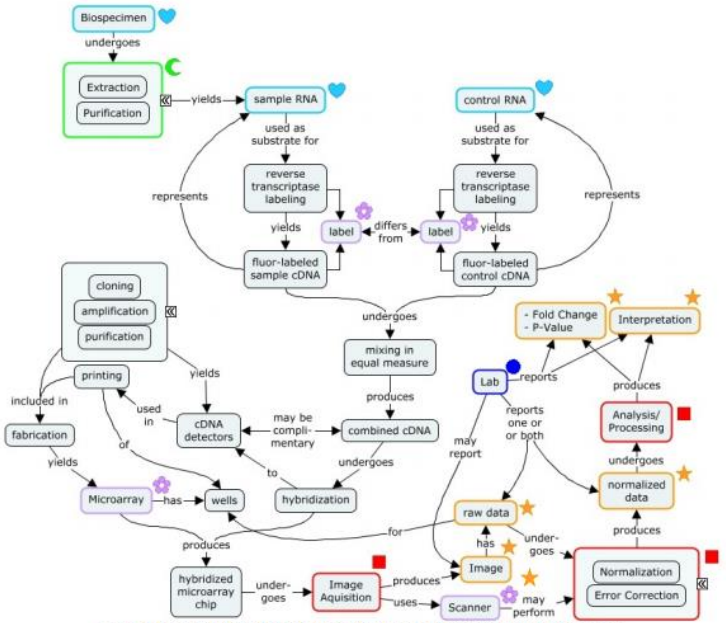


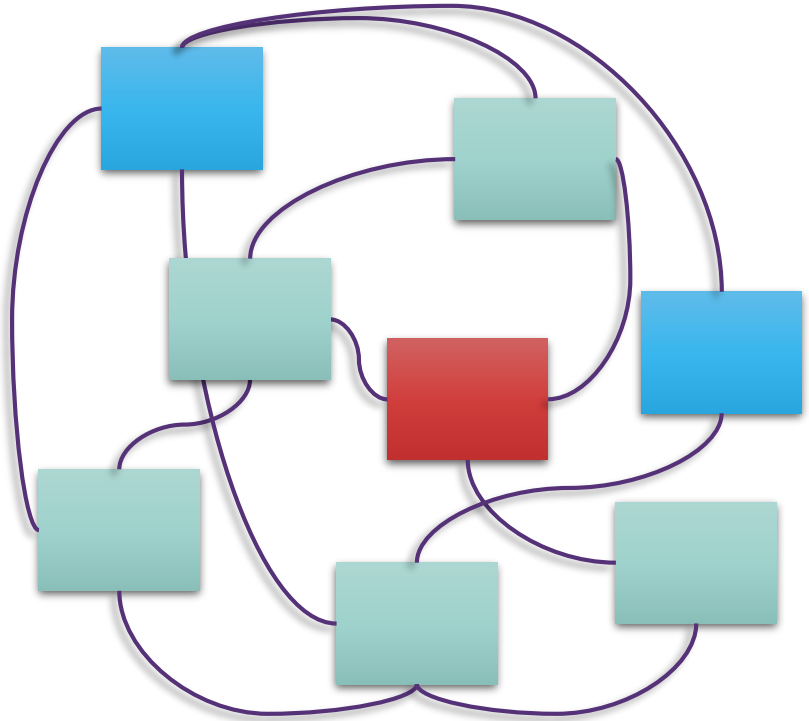
Figure 12: cDNA Microarray Two Channel Preparation and Processing

Modeling of microarray preparation is based upon protocols outlined by the National Human Genome Research Institute (NHGRI) at: http://research.nhgri.nih.gov/microarray/protocols.html. Methodologies may vary.

State diagram (1)

State diagram is the analogue in data science and system design.

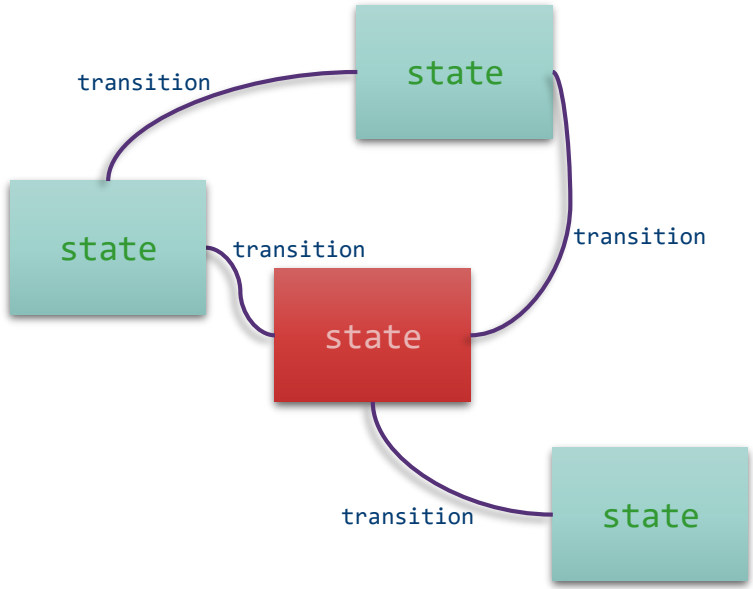
A state diagram is a model of deterministic graph with nodes and arcs.



State diagram (2)

The nodes are called state, the arcs are transition.

One may follow a transition to go from one state to another, and to navigate around the entire graph.



The Role of HATEOAS in CDISC Library API (1)

- The heart of the CDISC Library is the API
- The design is guided by REST principles by Dr. Roy Fielding
- A couple of important constrains for REST API:
  - Each response contains links for next requests
  - Server provides clients a **uniform method** for determining what contents can be retrieved, actions can be performed, and formats can be represented
- Flattened representation for the 3D world of metadata

The Role of HATEOAS in CDISC Library API (2)

An API response in JSON format when querying about SDTMIG v3.2

The Role of HATEOAS in CDISC Library API (3)

A link (**"href"**) is provided for one of the class objects accessible (**"type"**), whose name is Events Observation Class (**"title"**).

An application client (web, mobile, or programming language of choice) can reliably use this **uniform interface** for subsequent requests

The Role of HATEOAS in CDISC Library API (4)

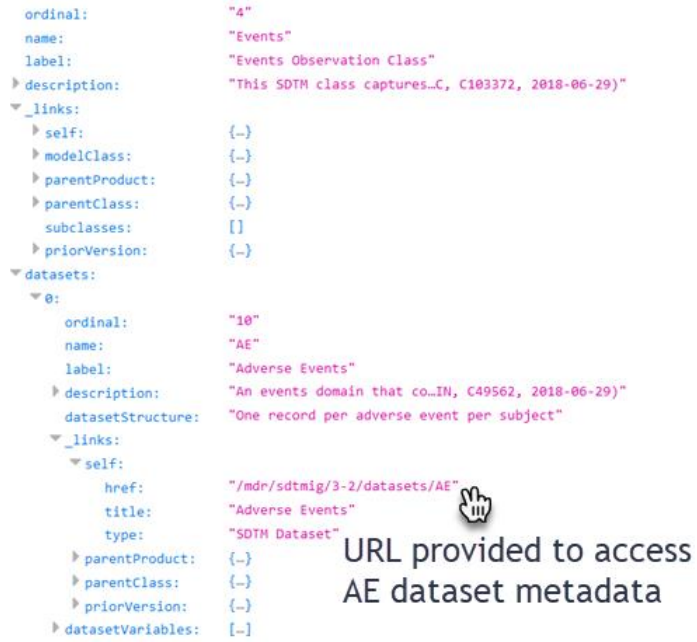
This is the response after following the hyperlink given /mdr/sdtmig/3-2/classes/Events

In other words, it is a state transition from SDTMIG v3.2 to the SDTMIG v3.2 Events class

The Role of HATEOAS in CDISC Library API (5)

A link (**"href"**) is provided for one of the SDTM dataset objects accessible (**"type"**), whose name is Adverse Events Class (**"title"**)

These links, for state transitions, aligns with the standards' hierarchy: classes > domains > variables > codelists



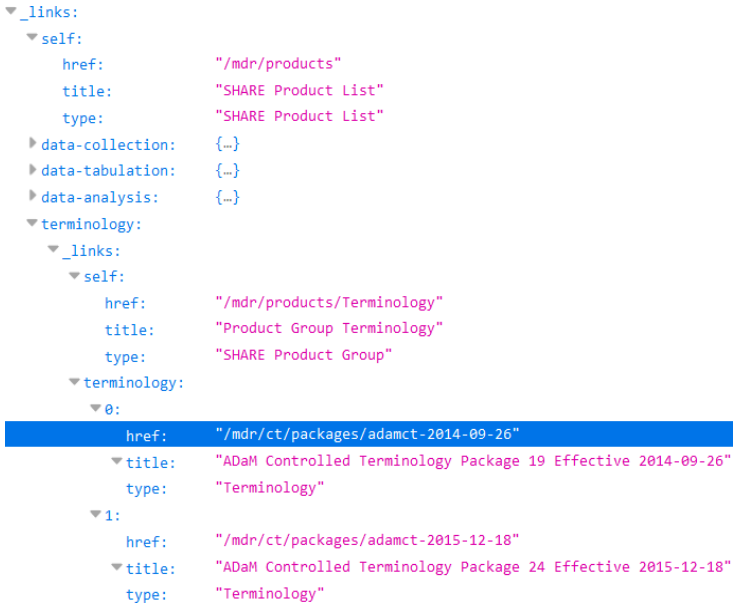
What's Available? A Machine-Readable Way (1)

Two endpoints are useful:

- /mdr/products
- /mdr/products/{product-family}

Where, product family is one of these:

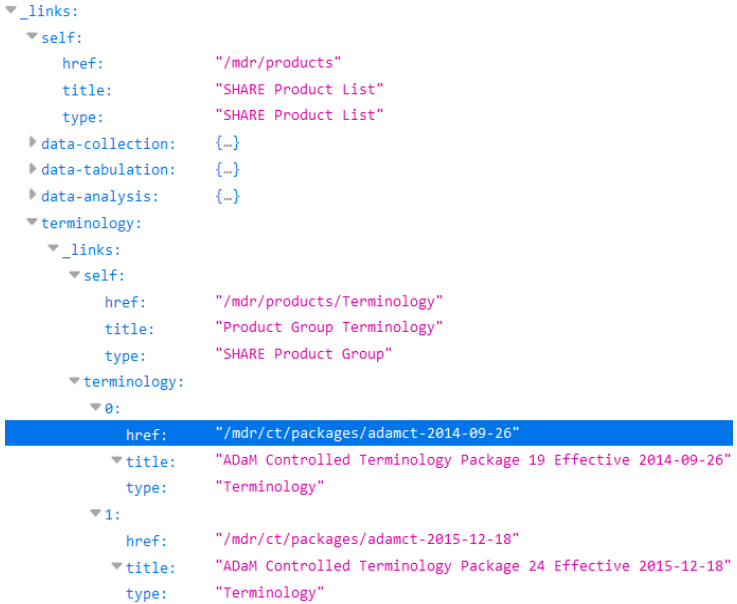
- Terminology
- DataCollection
- DataTabulation
- DataAnalysis



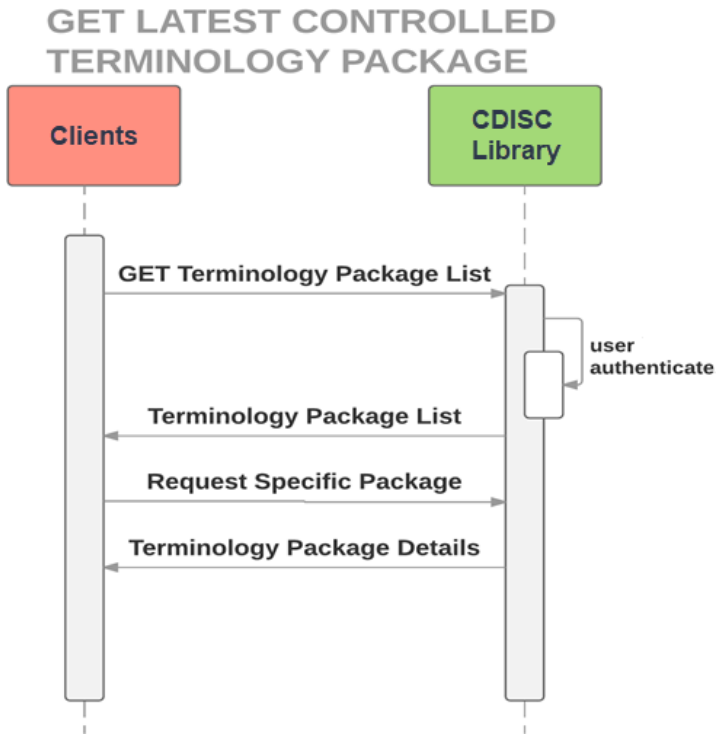
What's Available? A Machine-Readable Way (2)

Highlighted is a link to the a CDISC Controlled Terminology package after querying for a full product listing.

Notice it is using the same uniform interface for linking.



API Usage Scenarios: Get Latest Controlled Terminology Package



How CDISC Standards Are Connected

How CDISC Standards Are Connected (1)

- CDISC products are meant to be used in an end-to-end manner, and they are highly interconnected.
- Many CDISC products have published multiple versions over time.
- There are various ways the API surfaces this aspect of connectivity.

How CDISC Standards Are Connected (2)

IG and Model

Highlighted is an API response in JSON format when querying about SDTMIG v3.2, showing SDTM v1.4 is the model it implemented.


```
{
  "name": "SDTMIG v3.2",
  "label": "Study Data Tabulation Model Implementation Guide: Human Clinica
  "description": "CDISC Version 3.2 (V3.2) Study Data Tabulation Model Impl
  "source": "Prepared by the CDISC Submission Data Standards Team",
  "effectiveDate": "2013-11-26",
  "registrationStatus": "Final",
  "version": "3.2",
  "_links": {
    "self": {
      "href": "/mdr/sdtmig/3-2",
      "title": "Study Data Tabulation Model Implementation Guide: Human C
      "type": "Implementation Guide"
    },
    "model": {
      "href": "/mdr/sdtm/1-4",
      "title": "Study Data Tabulation Model Version 1.4",
      "type": "Foundational Model"
    },
    "priorVersion": {
      "href": "/mdr/sdtmig/3-1-3",
      "title": "Study Data Tabulation Model Implementation Guide: Human C
      "type": "Implementation Guide"
    }
  }
}
```

How CDISC Standards Are Connected (3)

SDTM and CT

Highlighted is an API response in JSON format when querying about SDTMIG v3.2's AGEU variable.


```
{
   "ordinal": "18",
   "name": "AGEU",
   "label": "Age Units",
   "description": "Units associated with AGE.",
   "role": "Variable Qualifier",
   "simpleDatatype": "Char",
   "core": "Exp",
   "valueList": [],
   "_links": {
      "self": {
         "href": "/mdr/sdtmig/3-2/datasets/DM/variables/AGEU",
         "title": "Age Units",
         "type": "SDTM Dataset Variable"
      },
      "codelist": [
         {
            "href": "/mdr/root/ct/sdtmct/codelists/C66781",
            "title": "Version-agnostic anchor resource for codelist C66781",
            "type": "Root Value Domain"
         }
      ]
   }
}
```

How CDISC Standards Are Connected (4)

CDASH and SDTM

Highlighted is an API response in JSON format when querying about CDASHIG v2.0's AGEU and how it maps to SDTMIG v3.2's DM.AGEU


```
{
   "ordinal": "10",
   "name": "AGEU",
   "label": "Age Units",
   "definition": "Those units of time that are routinely used to express
   "questionText": "What is the age unit used?",
   "prompt": "Age Unit",
   "core": "O",
   "_links": {
      "sdtmigDatasetMappingTargets": [
         {
            "href": "/mdr/sdtmig/3-2/datasets/DM/variables/AGEU",
            "title": "Age Units",
            "type": "SDTM Dataset Variable"
         }
      ]
   }
}
```

How CDISC Standards Are Connected (5)

Previous version (or, predecessor)

Highlighted is an API response in JSON format when querying about SDTMIG v3.2's DM.AGEU, with a link to its previous version, i.e., SDTMIG v3.1.3


```
{
   "ordinal": "18",
   "name": "AGEU",
   "label": "Age Units",
   "description": "Units associated with AGE.",
   "role": "Variable Qualifier",
   "simpleDatatype": "Char",
   "core": "Exp",
   "_links": {
      "codelist": [
         {
            "href": "/mdr/root/ct/sdtmct/codelists/C66781",
            "title": "Version-agnostic anchor resource for codelist C66781",
            "type": "Root Value Domain"
         }
      ],
      "priorVersion": {
         "href": "/mdr/sdtmig/3-1-3/datasets/DM/variables/AGEU",
         "title": "Age Units",
         "type": "SDTM Dataset Variable"
      }
   }
}
```

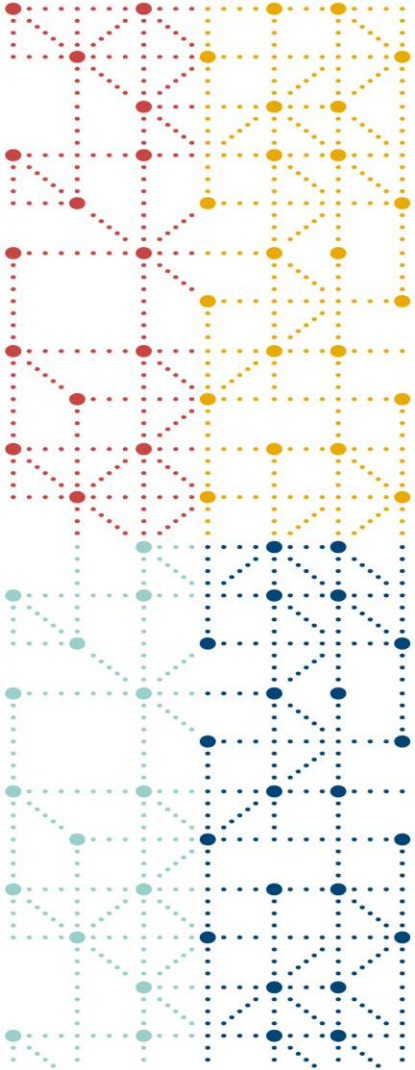
How CDISC Standards Are Connected (6)

Root items, version-agnostic resources

Highlighted is an API response in JSON format when querying about the AGEU variable in a SDTMIG dataset, showing a list of versioned variable in the metadata repository.

See API documentation for a full list where root items are applicable.


```
{
  "_links": {
    "self": {
      "href": "/mdr/root/sdtmig/datasets/DM/variables/AGEU",
      "title": "Version-agnostic anchor resource for SDTMIG variable DM.AGEU",
      "type": "Root Data Element"
    },
    "versions": [
      {
        "href": "/mdr/sdtmig/3-1-2/datasets/DM/variables/AGEU",
        "title": "Age Units",
        "type": "SDTM Dataset Variable"
      },
      {
        "href": "/mdr/sdtmig/3-1-3/datasets/DM/variables/AGEU",
        "title": "Age Units",
        "type": "SDTM Dataset Variable"
      },
      {
        "href": "/mdr/sdtmig/3-2/datasets/DM/variables/AGEU",
        "title": "Age Units",
        "type": "SDTM Dataset Variable"
      }
    ]
  }
}
```



Security and Server Availability

Security

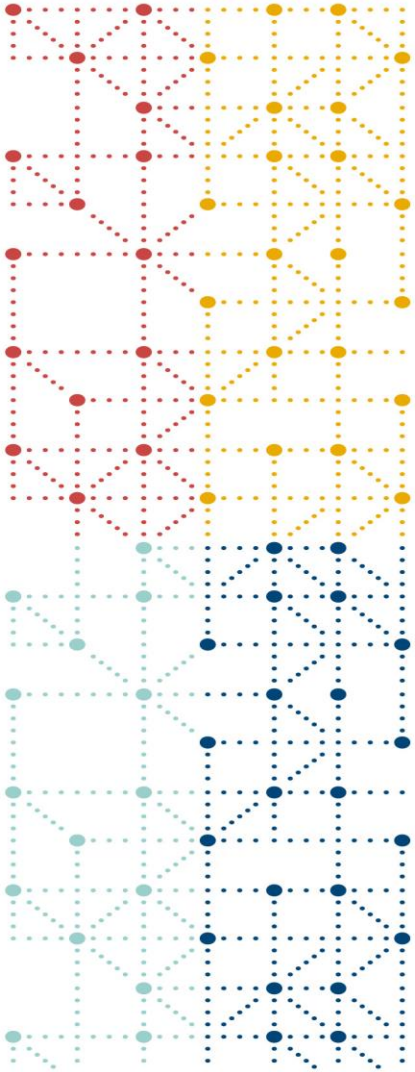
- Named user accounts with read-only access to all content loaded into the metadata repository
- HTTPS and basic web authentication based security
- Usage monitoring and event tracking
- Server is hosted on an AWS Virtual Private Cloud and security is managed by our hosting provider

Service Levels & Systems Availability

- Maintaining CDISC Library availability will be a priority
- CDISC Library will not have mission critical service levels
- Downtime will be kept to a minimum, but some downtime will occur
- Users will be notified of planned maintenance electronically
- No immediate plans to support a quantifiable service level (e.g. five 9s) in order to keep the cost down

Limitations

- XML and CDISC ODM v2 outputs are forthcoming, on top of JSON.
- Cosmetic touch-ups may be needed in the value of the “title” key part of HATEOAS.
- There are known gaps in cross-standards metadata. For example, when a CDASH’s mapping to an SDTM target that is not explicitly specified in the original publication, it is omitted in the metadata repository. CDISC will be working with standards development teams to discuss resolutions.
- Additional contents are being actively curated and tested. Watch the Release Notes page for updates.



Technical Wrap-up

Highlights (1)

- Designs are informed by Public Review and review walkthrough with each standards development team
- Access to contents do not require intimate knowledge about the API design
- Endpoints follow the standards' hierarchical stack, e.g.,
  {{base-url}}/mdr/ct/packages/sdtmct/2018-06-29/codelists/C99073/terms
  {{base-url}}/mdr/cdashig/2-0/domains/AE/fields
  {{base-url}}/mdr/sdtm/1-4/datasets/DM/variables
  {{base-url}}/mdr/adamig-1-1/structures/ADSL/variables/TRxxPGyN

  * Plural nouns indicate collections

Highlights (2)

- CDISC's implementation makes good use of REST's HATEOAS principle.
- It is a machine-reliable way of navigating the intricate collection of CDISC products. This has been referred to state transition.

```
"_links": {
    "self | <<applicable
object>>": {
      "href":
"<<value>>",
      "title":
"<<value>>",
      "type":
"<<value>>"
    }
```

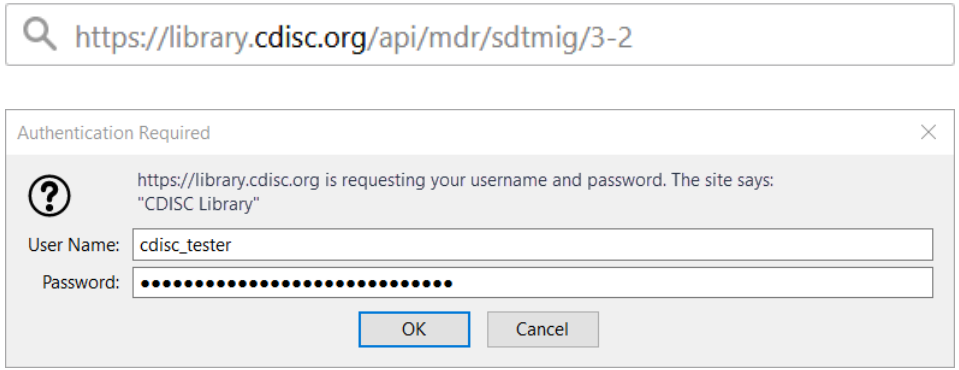
Highlights (3)

- Root items, i.e., version-agnostic anchor elements for versioned resources. Examples:
  {{base-url}}/mdr/root/ct/packages/sdtmct/codelists/C66742
  {{base-url}}/mdr/root/sdtm/datasets/DM/variables/RFSTDTC
  - Several useful applications, e.g.,
    - Linage: Dataset & variable history, retired & reinstated terms
    - Biomedical concept: Conceptual representation of medical & clinical knowledge; allow us to attach implementation information, such as variables, controlled terms, etc.
    - Immutable rule bindings: Flag variables using (NY) codelist, variables with ISO 8601-formatted values, etc.

Highlights (4)

- 85 versioned standards
- 1 million resources
  - > 90% related to CT
  - 1,200 CDASH fields, 2,000 SDTM variables, 740 ADaM variables
- 6 million relationships
- Other
  - 1,500 CDASH-to-SDTM mapping statements
  - 35 regulatory support statements
  - 100 predicates

Querying Using Web Browser



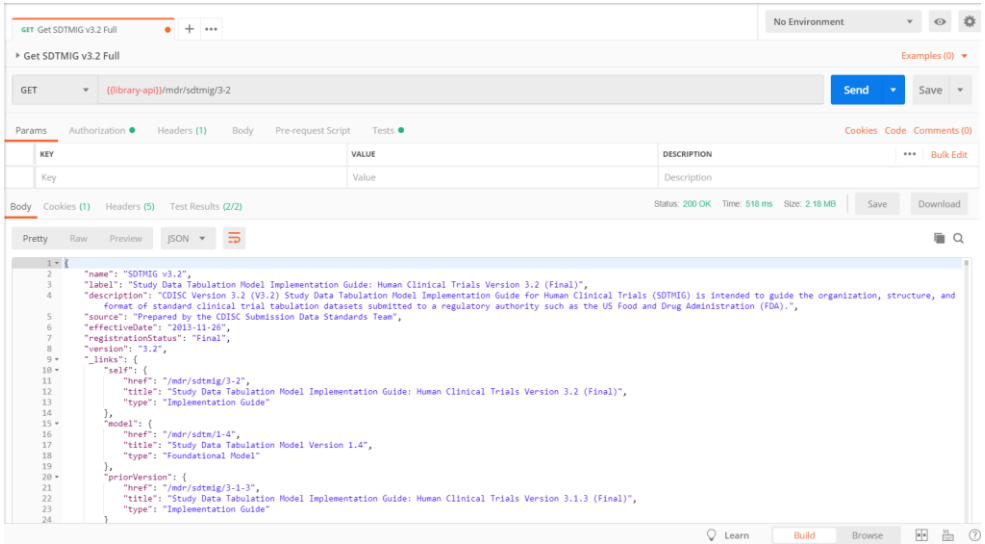
JSON Raw Data Headers

Save Copy Collapse All

```
name:               "SDTMIG v3.2"
label:              "Study Data Tabulation Mo…als Version 3.2 (Final)"
description:        "CDISC Version 3.2 (V3.2)…g Administration (FDA)."
source:             "Prepared by the CDISC Su…ion Data Standards Team"
effectiveDate:      "2013-11-26"
registrationStatus: "Final"
version:            "3.2"
_links:             {…}
classes:            […]
```

API Development Tools

Free and open-source development and testing tools are plentiful. Postman, SoupUI, Insomnia are just a few suggestions. Postman is shown below.



Sample Code Repository

Displayed is a Python code that queries one codelist and export to a CSV file.

Code repository will be an excellent way to share ideas and API use cases.

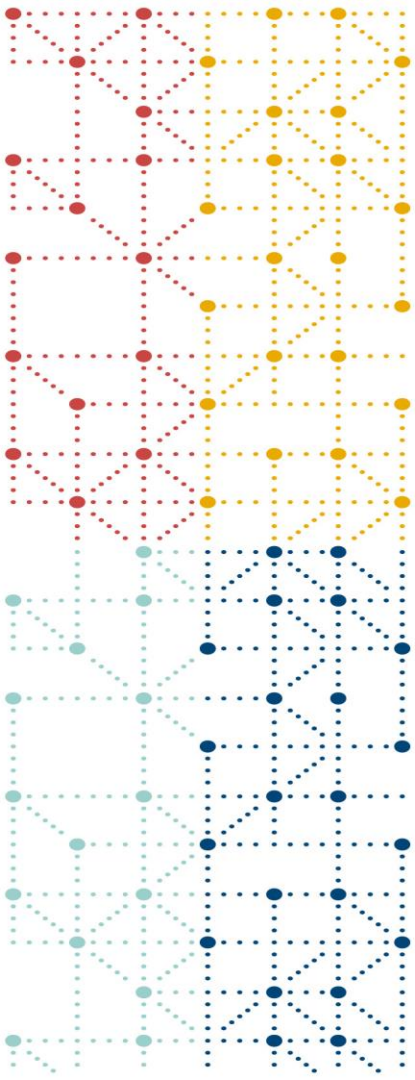
Tentative plan is to host it on CDISC Bitbucket, open to public. Details will be in future announcements.

```python
import requests
import json
import pandas as pd

# set ct package and codelist for export
ctPackage = 'sendct-2016-03-25'
ctCodelistCode = 'C74456'

try:
    # make a SHARE API request to retrieve codelist from package
    req = requests.get("https://library.cdisc.org/api/mdr/ct/packages/" +
      ctPackage + "/codelists/" + ctCodelistCode, auth=("user", "pass"),
      headers={'Connection': 'keep-alive', 'Accept': 'application/json'})

    # proceed when response status code is 200
    if req.status_code == 200:
        # deserialize response to a python dictionary object
        resp = json.loads(req.text)

        # load response to a pandas dataframe
        df_terms = pd.DataFrame.from_dict(resp['terms'])

        # sort data by submission value
        # flatten list (array) of synonyms to a semi-colon separately value list
        df_terms.sort_values(["submissionValue"], ascending=[True], inplace=True)
        df_terms['synonyms'] = df_terms['synonyms'].str.join('; ')

        # write to a CSV flat file
        df_terms.to_csv("ct_example_out.csv", index=False, sep=",")
    else:
        pass
except:
    raise
```

Useful Links

- API Documentations: https://www.cdisc.org/cdisc-library/api-documentation
- Release Notes: https://wiki.cdisc.org/display/LIBSUPRT/Release+Notes
- Further Reading
  - REST API. Fielding, Roy Thomas (2000). "Architectural Styles and the Design of Network-based Software Architectures". Dissertation. University of California, Irvine.

End of Technical Introduction



In Closing…

We would like to acknowledge all the CDISC staff who contributed to the successful development and release of the CDISC Library API.

cdisc LIBRARY
formerly SHARE

To help support our efforts, please contact us if you're interested in joining our team and paving the future.

cdisc