Data Catalogues – The new Showman to organize your data, A CRO perspective

Limna Salim, Genpro Research, Trivandrum, India
Mohin K Mohanan, Genpro Research, Trivandrum, India

ABSTRACT

In an era where data driven approach is prioritized by CROs, Data catalogues are an important functionality. Every team after adopting data catalogues find it productive to organize and review their data.
Here we focus on creating secure Web based interactive catalogues across use cases mentioned from a CRO perspective. Also using a standard modular template optimizes structuring and retrieving information across the organization.

Use Cases detailed in paper are as below-

- Macro Library
- Standard Study Repository
- Metadata Extractions for CDISC (SDTM/ADaM) programming
- Statistical Programming library

A good catalogue contains elements such as Data discovery tools and keyword search engines that would allow users to discover metadata. It also provides the ability to download contents enabling reusing scripts in a downloaded format.

This paper also reviews

- Different elements that catalogue contain.
- Development using open-source platforms like R ® and Python with Markdown files.
- How to Access & Extract information.

INTRODUCTION

A data scientist or statistician or programmer initially always relies on the organization's data to curate models or discover trends. They need to quickly browse through information to find answers to their questions and identify which questions needs additional data. A Data Catalogue can be the go-to-spot in such cases ensuring a holistic and navigable view of the organization's data spread across various sources and repositories be it Technical, HR, Finance etc.
A data catalogue is thus a library of all your data neatly indexed, organized, and kept ready for use. It not only lists out the data but also details it to the users.

BENEFITS OF DATA CATALOGUES

Clarity, Accuracy and Speed are three main factors that significantly improve with the implementation of Data Catalogues. Other benefits include-

- Refined Data efficiency
- Refined Data analysis
- Simplified Data governance with proper structure and definitions
- Access to metadata alongside the data
- Traceability and understanding the lineage of data
- Improved compliance/data security

DATA CATALOGUES IN HEALTHCARE

Catalogues can have a huge diversity of applications in the healthcare domain.

- Facilitate the discovery of biomedical/research data.
- Database for Genomics data

- Clinical trial database
- Database for Epidemiology
- Patient Engagement & Population Health

DATA CATALOGUES IN CRO'S

Customized catalogues can be created for each CRO's specialized departments, such as Statistics, Medical Writing, Data Analytics, Statistical Programming, Data Management, Research, and Human Resources. We can tailor the layout to meet the unique requirements of each group.

In this paper, we talk about 4 different Use Cases where data catalogues can be implemented within a CRO.

- Macro Library
- Standard Study Repository
- Standards & Guidelines Repository
- Statistical Programming library

BUILDING A DATA CATALOGUE

Before discussing further on the use cases, let's first explore how a data catalogue can be created. We currently have various tools in the market available that can easily help us develop catalogues. There are various organizations that develop catalogues and curate them according to our needs. But this comes with a continuous dependency and an associated cost. From a CRO perspective, this can be mitigated to a certain extend by building catalogues in house using our internal pool of developers.

Data catalogues can be developed using various back-end languages like JavaScript, Python, PHP, C# etc. offering plenty of frameworks and supporting several platforms. Lack of awareness across these languages becomes challenging for our clinical team. In this paper, we explore the options of creating a data catalogue using R, a language frequently used by data scientists/statistical programmers within a CRO.

DATA CATALOGUE USING R

The goal of this section is to give a concise step-by-step guide on how we can build a catalogue using R framework. We will then take you in detail through the use cases mentioned above and how these can be incorporated using the elements mentioned below.

PREREQUISITES

To get started on the development, we need to ensure the below aspects are in place.

- Install a recent version of R
- Predefined theme that you can use across catalogues
- Collection of Metadata and data to be presented across catalogues
- Getting study folder or Github repository in place.

We need to also have the below mentioned packages installed in R:

- Bookdown
- Rmarkdown

IMAGE 1: Home Page of the catalogue

ELEMENTS IN DATA CATALOGUE

There are various components or elements included as part of catalogue development. Each of these are described below step by step.

STEP 1 – CSS FILE

CSS is abbreviated as Cascading style sheet, a file that sets the theme of your Catalog. It is a stylesheet language that describes the presentation of an HTML page and describes how elements should be rendered on screen or displayed on a webpage like background color, font size etc.

The advantage is that with a css file, you can change the look of an entire page by changing just one file! It can also control the layout of multiple pages all at once.

STEP 2 – RMD FILE

RMD file or R Markdown files are the most important component of catalogue development. This is basically the section where all our data are fed. It provides an authoring framework that we can use to save and execute our code and generate high quality reports. It is a plain text file with extension '.Rmd'. As we open the file, it becomes a notebook interface for R. You can run each code chunk by clicking the run icon.

To have this generated, we need to have the 'rmarkdown' package installed.

A rmd file contains three important types of content:

- An (optional) YAML header surrounded by ---s.
- Chunks of R code surrounded by ```.
- Text mixed with simple text formatting like # heading and italics_.

Once the required contents are populated into the rmd file, we can produce a complete report containing all text, code, and results by just clicking "Knit" in the R interface. We can also do this programmatically using render command -rmarkdown::render("1-example.Rmd").

When you render your .Rmd file, R Markdown will run each code chunk and embed the results beneath the code chunk in your final report. For chunks we can also set the option to whether run the code, present the output, or just simply display it.

Rendering or Knitting will display the report in the viewer pane and create a self-contained HTML file that can be shared. R Markdown generates a new file that contains selected text, code, and results from the .Rmd file. The new file can be either a finished web page, PDF, or MS Word document.

All the data and content that we want to be displayed as part of our catalogue will be basically defined in these rmd files and rendered as separate pages.

Features of rmd

- It allows you to implement text, code, graphs, all in one interactive report that can be run and re-run with ease.
- R Markdown report can be knitted in a variety of document formats including HTML, PDF, MS Word, MS PowerPoint.
- Helps to create and reuse template documents with custom formatting.
- Helps to avoid switching between multiple tools. Data wrangling, visualizations, reporting can all be accomplished using rmd.
- RMarkdown enables reproducibility.

.

STEP 3 - BOOKDOWN

Bookdown is an R package that facilitates long-form reports with R Markdown. The next major component involved as part of the development is parsing all the created rmd's in the previous step through the bookdown package. Bookdown enables the possibility of having dynamic graphs and interactive applications and renders multiple output formats.

You can use bookdown's render book() function to combine all Rmd files into a book. The 'bookdown' package must be installed in order for this to be generated.

When you click the Knit button to compile an R Markdown document in the RStudio IDE, the default function called by RStudio is rmarkdown::render(), which is not what we want for books. We need to call the function bookdown::render_book() instead. This can be rendered through the button Build Book in the Build pane. You can click the button to build the whole book in different formats, and if you click the Knit button on the toolbar, RStudio will automatically preview the current page/session in the provided formats.

Features of bookdown

- Can be easily published to GitHub, bookdown.org, and any web servers
- Integrated with the RStudio IDE
- Multiple choices of output formats
- Flexibility of including dynamic graphics and interactive applications

USE CASES FROM A CRO PERSPECTIVE

There are various use cases where catalogues can be implemented across a CRO that too across various functions within your organization. Some of the use cases incorporated within Genpro are explained below.

To begin, a single CSS file is created that contains a standard template that can be used across all catalogues referenced within our organization. This allows us to maintain a consistent format, fonts, logos, and so on across all catalogues. Once the data has been rendered, we can make changes to the catalogue's final presentation by modifying the CSS file.

USE CASE 1 – MACRO LIBRARY

Every Statistical programming team comes with its own share of SAS macros that we utilize for our day-to-day programming. Mostly these macros would be placed within a common location where programmers can access it from. Making this retrieval smoother will also be to have a glossary of these macros available where user can understand in detail what each macro does and its outputs & dependencies.

We start by defining the data that is needed. Here we have basically accounted for the various macro categories and macros under each category will be listed out with further drill down taking us to the details of these individual macros like the input parameters, assumptions, a sample call etc.

Once data is defined, we map all these information to a rmd file, an example of which is as below. As mentioned earlier rmd file allows us to mention specifically the Headers and contents parsed under each section. Each page that needs to be displayed as part of the catalogue should be mapped to one rmd file.

Once all rmd files are ready, we simply build the book as described earlier to include all of these files in the bookdown package. We also have the option of specifying the type of output we want, such as a pdf or an html file.

Specific packages in this use case –

- SASmarkdown

Special features implemented in the page include –

- This section currently displays macro codes in their entirety. The functionality of presenting this in a box and enabling sliders has been completed.

Future scope of expansion in this use case include –

- Provision to download the macro code version and using it from specific location.
- Long term scope can also include having the macros executed from the Catalogue page.

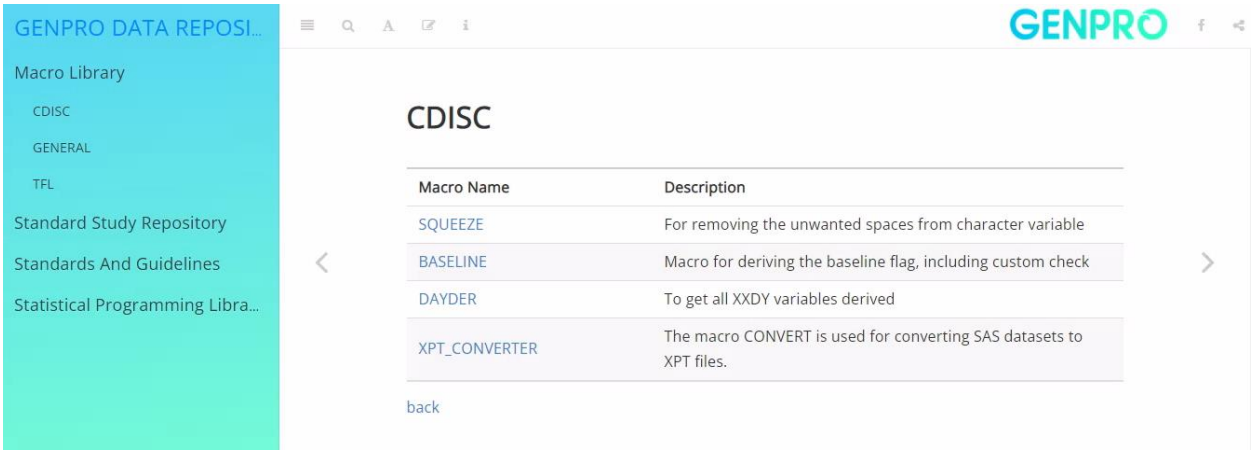

IMAGE 2: Macro Library main page

USE CASE 2 – STANDARD STUDY REPOSITORY

For every CRO, maintaining a list of studies they have contributed towards is always beneficial. This can be useful for the team to turn back to as reference, or can be used by the QA team for an audit/RFI purpose etc. Understanding the exposure across specific therapeutic areas, phase of study, number of subjects/sites as part of the study, niche areas of contribution by the team are all crucial information we would like to maintain.

Specific packages in this use case –

- Gt
- DT
- openxlsx

Special features implemented in the page include –

- The study details are captured in an excel and here we showcase the possibility of importing the excel file into our rmd file and having that presented.

Future scope of expansion in this use case include –

- Currently it's just a display of all the studies, but we can present a small summary by providing an automated count of number of studies in each section.

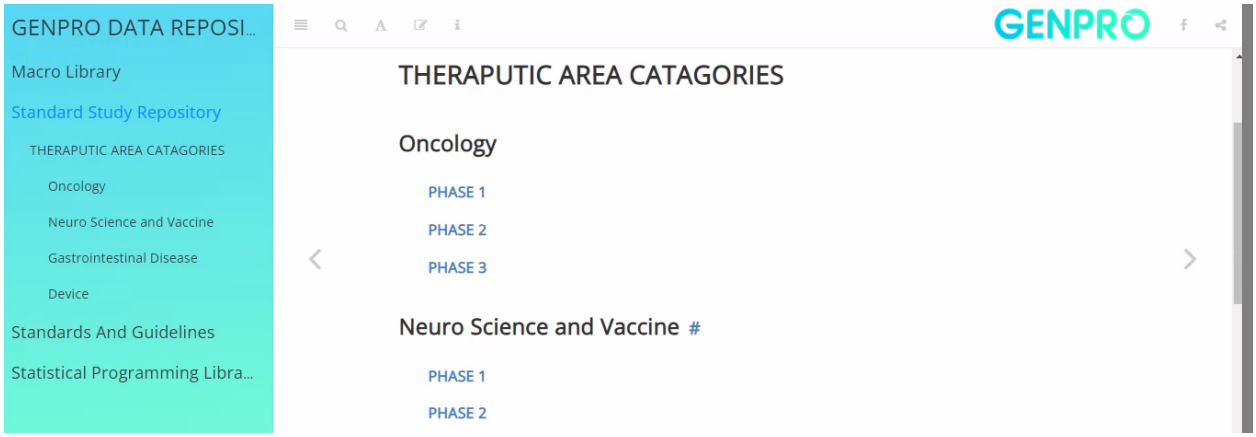

IMAGE 3: Study Repository main page

USE CASE 3 – STANDARDS & GUIDELINES REPOSITORY

As Clinical Programmers or Statisticians, we must strictly follow or rely on the standards and guidelines set forward by the FDA/CDISC for all submissions. Having these standards readily available for our team becomes crucial and this can be accomplished using catalogues.
Multiple versions of the same guideline can also be represented across pages along with information like version number, release date, links to these documentations etc.

Specific packages in this use case –

- Gt
- DT
- openxlsx

Special features implemented in the page include –

- Accounted for tabular representation of data.
- Incorporated sliders in the table that helps us to scroll the data
- Provided hyperlinks within the data that takes us to external web links.

Future scope of expansion in this use case include –

- Hyperlinks can also be expanded to include data stored within the organization's servers itself like files saved in SharePoint.
- Considering that guidelines are subject to change, we would need to account for Version changes to a particular document and this can be automated.
- Use this catalogue section to maintain a repository of standard CDISC templates used like specification template, metadata mapping, documents structure like reviewer's guide etc.

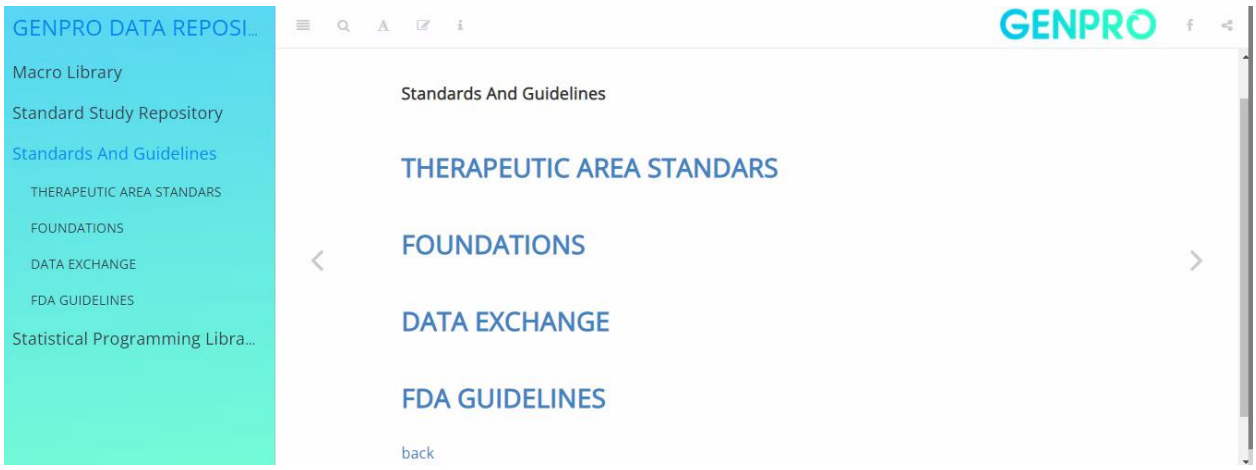

IMAGE 4: Standards & guidelines main page

IMAGE 5: Standards & Guidelines Sub page

USE CASE 4 – STATS & PROGRAMMING LIBRARY

No Statistical Programmer or Statistician comes with the full knowledge of a code snippet specifically the syntaxes or options that can be used, be it in R or SAS. To a certain extend we would always rely on web search, for eg - to see how a Stat model can be fixed, using proc MMRM or Regression, we would start to obtain a base code and then fit our data into it. Data catalogues can be implemented in this situation to reduce the research time for our team. This use case takes you through the Stat or Programming library were code snippets from complex analysis performed by the team can be saved for further use by the rest of the team. We have incorporated a code patch, a sample data where this code can be applied and the various outputs from the code patch. Thus, by creating a vast library with the help of this catalogue, our team's search time will be drastically reduced, and easy references with details will be made available.

Specific packages in this use case –

- SASmarkdown

Special features implemented in the page include –

- Having images or graphs incorporated into the presented details.

Future scope of expansion in this use case include –

- Enable links to access the data directly where users can maybe test and execute
- Have a provision to download the code from the catalogue page
- Incorporate any type of programs be it SAS/R scripts for a particular concept.

IMAGE 6: Statistical Programming Library main page

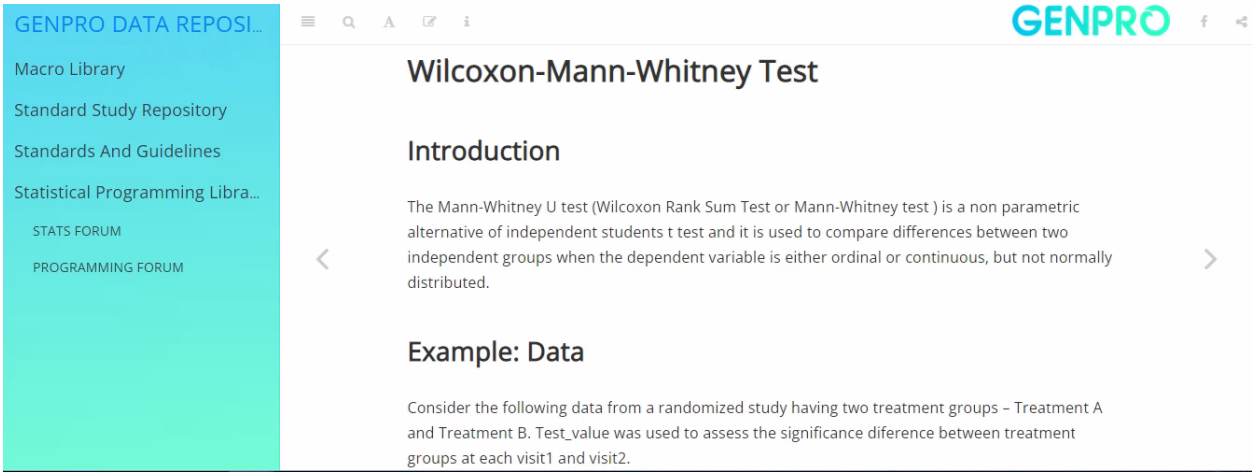

IMAGE 7: Statistical Programming Library sub page

FEATURES OF OUR CATALOGUES

It is not practical to create a data catalogue from scratch, but it is not impossible. It is critical that data be collected from multiple sources and that it be indexed and discovered. The following are some of the key features that have been implemented or will be implemented in the catalogues in the future.

- Search Capability:
  Searching the data within the catalogue using business terms, keywords, and natural language.
- Data Preview:
  Before visualizing the data, it becomes very important to provide a preview of the dataset and the quality of the data.
- Data Privacy:
  It is important for the data catalogue to not expose all the data to all users or in other words maintain data privacy.
- Expansion/Upgradation:
  Instead of having a pre-set list of catalog objects, you can add as many fields or objects as you want at any stage and organize them however you like.
- Business Glossary:
  Helps in capturing business vocabulary of the CRO

Features that can be included in future scope or improvement include-

- Implementing tags and links to other catalogues or documents within the internal server.
- Integrating capabilities from other apps using REST APIs.
- Having provisions to include notes, review feedback and ratings on data presented in the catalogue to make it more interactive.

- Having data accessed directly and seamlessly from the servers or Cloud databases.
- Data intelligence features can be explored where data catalogue users are provided with data recommendations.
- Give a Weblike interface feel for our Catalogue by expanding from Bookdown package to Blogdown package.
- The data that is captured within the catalogues can be converted to Knowledge Graphs as part of a long-term goal.

HOSTING OUR CATALOGUES

Once the Catalogue pages have been developed and built into a book, the next step is to have these shared with the users.

The first main thought would be to have the R scripts shared with the users, but this comes with the necessity of having R installed across the systems. Thus, hosting the Catalogue application as a Web Page would be the best and most appropriate option to avoid manual execution. Users will use a web browser to access the catalogue and find the app fully rendered, up to date, and ready to use.

From a CRO perspective, the best approach for hosting will be to have a server access/credentials created for our team, have all the files/scripts implemented and have the server setup ready in our environment. If familiar with web hosting, we can host it ourselves, if not our IT department should help in hosting.

For further support or easier experience, RStudio itself offers different ways to host your application.

1. RStudio Connect
2. Bookdown.org
3. Rpubs
4. GitHub

CONCLUSION

Data scientists or Statisticians or Programmers have different requirements for a data catalog. Although there is significant overlap between their workflows, they all rely on data in a data lake rather than the highly structured data in the database. Thus, it's important to have a data catalog that can meet the technical needs of all groups. There are numerous tools available to aid in the development of catalogues, but the method summarized here is a simple and cost-effective way for a CRO to temporarily kick start the process, which can then be modified and further developed.

RECOMMENDED READING

https://bookdown.org/yihui/rmarkdown/rmarkdown-site.html
https://rmarkdown.rstudio.com/lesson-1.html
https://r4ds.had.co.nz/r-markdown.html

CONTACT INFORMATION

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

Author Name : Limna Salim
Company : Genpro Research
Address : Technopark, India
City / Postcode : Thiruvananthapuram, 695581
Work Phone : 04714026700
Email : limna.salim@genproresearch.com
Web: : www.genproresearch.com

Author Name : Mohin K Mohanan
Company : Genpro Research
Address : Technopark, India
City / Postcode : Thiruvananthapuram, 695581
Work Phone : 04714026700
Email : mohin.mohanan@genproresearch.com
Web: : www.genproresearch.com