

Are you out of your mind? CDISC mapping using a Mind Mapping tool

Johannes Ulander, S-Cubed, Copenhagen, Denmark
Niels Both, S-Cubed, Copenhagen, Denmark

ABSTRACT

The CDISC contents models such as CDASH, SDTM and ADaM as well as the CDISC controlled terminology are used all across the clinical dataflow in pharmaceutical companies. In addition internal database models and sponsor defined controlled terminology are also being used.

In theory the relationships between the various data elements should be described by electronic metadata that can be used to drive data transformations. But in practice it often turns out to be very difficult. One reason for the difficulty is that the relationships are quite complex. Another reason is that relationships need to be described correctly in every detail, in order to drive the data transformation.

The purpose of this paper is to present the idea to use a mind mapping tool like Freeplane to represent the models and relationships in a more user friendly way as we believe a mind map can help bridge the gap between human usability and machine readable metadata structures.

INTRODUCTION

Mind Map tools are flexible in their use and has an interface and visualizations which appeal to most people. But so far the use of mind map tools is quite limited to a few people who mostly use it as a personal tool on their desktop pc. It is used for brainstorming, managing meetings, structuring ideas etc. due to the visual capabilities, but the content of the information is normally moved into another format/document in the end to be spread around teams and organizations and there is few people that have the correct software installed to be able to view the result as captured during the meeting.

Another caveat is that mind mapping tools are considered to be heavily dependent on using your mouse for restructuring the mind map and creating links between nodes.

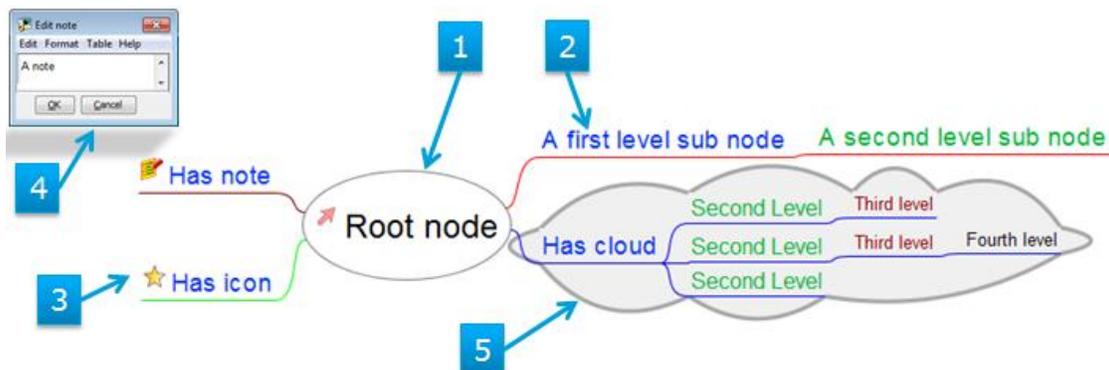
But what is usually not known, is that mind maps are XML files that can handle attributes and non-visual information that can be used for exporting and filtering the information in the mind map. You can also export this in a structured and customized way so that it is machine readable and fits other standards.

There are also mind map tools that support script programming within the tool so that you can restructure, combine and filter the mind maps while keeping its visual impact of understanding of how information fits together.

When the information is structured as XML, it can be used by software applications, for example SAS, to drive data transformations.

A simple mind map

Mind mapping and its concepts are not complicated and the below picture is just to set the scene of what is meant with a mind map. A mind map consists of a root node (1) as the center of the diagram. Sub nodes related to the root node (2) are added. The nodes then can have different styles, attributes, icons (3), notes (4) and be surrounded with clouds to group (5).



Picture 1. A simple mind map example

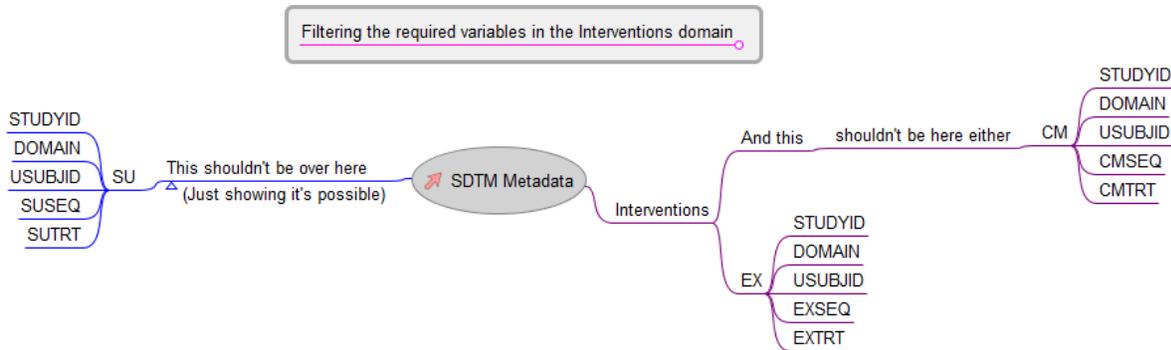
VISUALIZING CDISC STANDARDS

The standard way of visualizing how you map your raw data to CDISC SDTM today is usually in some form of table or grid, like a relational database table or excel sheet. For simple mappings this works quite well, when there is only a source variable and one target variable. Explaining more complex mappings usually involve creating custom made SAS macros or code that fits the infrastructure and systems that exist in each company's specific environment. The effect is that it is a large undertaking for anyone outside this environment to learn the syntax used and to understand the SDTM mappings. Often times many decisions for how you create them are dependent on the limitations and shortcomings of the already existing systems.

Source table	Source variable	Mapping	Target table	Targe variable
MED	TXT	Simple uppercase	CM	CMTRT
MED	DICTXT	%mbrss(input1=abcde,inpu t2=anggd,trans=Y,up=N)	CM	CMDECOD

Picture 2. A common way of expressing SDTM mappings

Another problem with the "table-way" of looking at information is that the cells and the way they are organized in the tables is what actually show the relationship. If you try to add more information, such as comments/notes or any extra information, it disturbs the visualization effect of understanding the relationship, and it is hard to show or hide information when needed or for different purposes. This problem is less in mind mapping tools as they have the ability to add attributes to nodes which can be shown or hidden. These attributes can then e.g. be used as a filtering condition which is applicable to a node independent of where in the structure the node resides.



Picture 3. Filtering the mind map on the required variables in the intervention domains

The above picture is possible due to the attributes assigned to the nodes, and it is the attributes that are used for filtering. So the information that the node actually contains looks like is this:

Variable name	Value
Seq	3
Observatio...	Interventi...
Domain	CM
Variable na...	USUBJID
Variable na...	USUBJID
Variable label	Unique Su...
Type	Char
CT	
Role	Identifier
SDTMCore	Req
Source	SDTM v1.3

Picture 4. A node with SDTM metadata as attributes

If you want to show only some of the attributes, because sometimes they are useful, that is possible as well.

PhUSE 2013

And if we take a look behind the scenes of the mind map, this is the node we just looked at, in its native format:

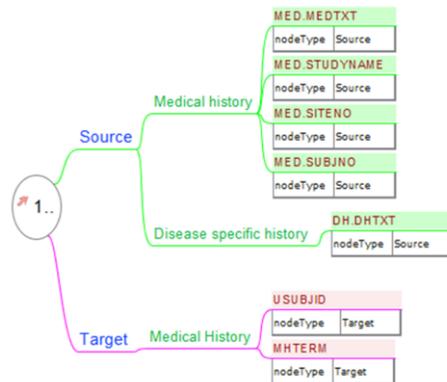
```
<node TEXT="USUBJID" ID="ID_1867308046" CREATED="1377797330782"
MODIFIED="1377800335489">
<attribute NAME="Seq" VALUE="3"/>
<attribute NAME="Observation class" VALUE="Interventions"/>
<attribute NAME="Domain" VALUE="CM"/>
<attribute NAME="Variable name short" VALUE="USUBJID"/>
<attribute NAME="Variable name" VALUE="USUBJID"/>
<attribute NAME="Variable label" VALUE="Unique Subject Identifier"/>
<attribute NAME="Type" VALUE="Char"/>
<attribute NAME="CT" VALUE=""/>
<attribute NAME="Role" VALUE="Identifier"/>
<attribute NAME="SDTMCore" VALUE="Req"/>
<attribute NAME="Source" VALUE="SDTM v1.3"/>
</node>
```

Most mind map tools are using XML as the format for storage, as the structure of the mind maps fits well with the well-formed document specification of XML files, e.g. having a single “root” element containing all other elements. In a way - the mind map tool is used very much like a dynamic style sheet to represent or edit the underlying XML file. Since a mind map is an XML file it is quite easy to export them to other formats as well as using other tools, e.g. SAS, to read the information. Of course, you can do it the other way around as well, creating the mind map from e.g. SAS.

USING MIND MAP FOR CREATING A SDTM MAPPING

Some mind mapping tools also contain scripting capabilities, so that you can programmatically interact with the information within the mind map. The scripting capability together with the attributes form an interesting combination due to the fact that the layout of the map, via a script, automatically can generate the necessary information that is used as a executable elements (e.g. XML output, a code snippet or a macro call) that work together with already existing tools

As an example you can have your source and target variables in a mind map. In this case it consist of a very limited set of variables from two different CRF pages, one for the general medical history and one for disease specific medical history. So the target variables are in this case from the SDTM MH domain for both CRF pages.



Picture 5. A mind map for creating SDTM mapping

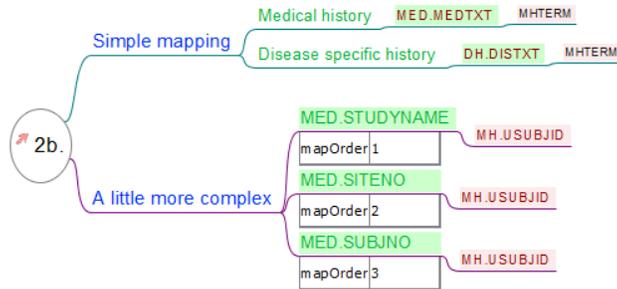
Now we restructure the information so that the target variables becomes sub nodes to the source variables.



Picture 6. Source variables “mapped” to target variables

For instructional purposes we’re grouping the mappings into simple and the one more complex one.

PhUSE 2013



Picture 7. The “mappings” reordered into simple and complex

Also showing that an attribute has been added to the source variables, telling in which order the different items should be used. (The order in the map could of course be used as well from the beginning, to assign the *mapOrder* attributes from the beginning.)

Seeing that the first simple mappings are possible to use as mapping information for a simple map is straight forward. In an object oriented language as in the below example:

```
def nodeList = c.find{ it["mapOrder"] in ['1','2','3'] | it.children[0].["NAME"] == "MHSUBJID" }
def nodeText = new ArrayList (nodeList).sort{ it["mapOrder"] }
def usubjid = nodeText.join("/")
```

the code can be used on the map to generate a variable *usubjid* with the values of the text of the nodes.

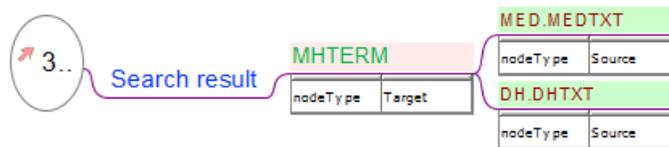
```
usubjid = MED.STUDYNAME/MED.SITENO/MED.SUBJNO
```

STANDARDS MANAGEMENT?

Everyone involved in standards management or SDTM submissions are probably familiar with these types of questions:

- What else maps to this target?
- Have we created such a variable/supplemental qualifier before?
- How did we map that CRF page?
- How often do we use that variable?
- Is that part of the domain?

If the standards are stored in a mind map, you will get the instant benefit of having a queryable XML file that can help answering the questions that is being asked..



Picture 8. Search result of “Which variables/CRF pages map to MHTERM?”

CONCLUSION

It is possible to use mind mapping tools to aid in visualizing and generation of CDISC standards and metadata. It has a form that is more user friendly than what can be done in a rectangular structure.

PhUSE 2013

RECOMMENDED READING

Some mind mapping tools to check out:

http://freeplane.sourceforge.net/wiki/index.php/Main_Page

http://freemind.sourceforge.net/wiki/index.php/Main_Page

<http://www.xmind.net/>

<http://www.thebrain.com/>

<http://www.mindjet.com/>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author(s) at:

Johannes Ulander
S-cubed Aps
Email: ju@s-cubed.dk
Web: www.s-cubed.dk

Niels Both
S-cubed Aps
Email: nb@s-cubed.dk
Web: www.s-cubed.dk

Brand and product names are trademarks of their respective companies.